# Enhancing ReaxFF for Molecular Dynamics Simulations of Lithium-Ion Batteries: An interactive reparameterization protocol (Supplementary Material)

Paolo De Angelis  $^{1,*},$  Roberta Cappabianca  $^1,$  Matteo Fasano  $^1,$  Pietro Asinari  $^{1,2,\dagger},$  and Eliodoro Chiavazzo  $^{1,\ddagger}$ 

<sup>1</sup>Department of Energy "Galileo Ferraris", Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Torino, Italy. <sup>2</sup>Istituto Nazionale di Ricerca Metrologica, Strada delle Cacce 91, 10135 Torino, Italy.

\*paolo.deangelis@polito.it

<sup>†</sup>pietro.asinari@polito.it

<sup>‡</sup>eliodoro.chiavazzo@polito.it

This is the supplementary material for the article "Enhancing ReaxFF for Molecular Dynamics Simulations of Lithium-Ion Batteries: An interactive reparameterization protocol." Here we list resources and images that aim to provide further explanation and understanding of the methods and results explained in the main text.

In line with the FAIR principles (Findability, Accessibility, Interoperability, and Reusability)<sup>1</sup>, we provide detailed documentation within the two repositories (Section 1 and Section 2) that accompany the article. These repositories are designed to help the reproducibility of our results and facilitate further exploration of the LiF configurations or other Solid Electrolytes Interphase (SEI)<sup>2</sup> compounds, extending the database and improving the ReaxFF force field<sup>3</sup>.

## Contents

1	Enhancing ReaxFF protocol (Protocol Repository)	2
1.1	Installation	2
	Minimum Requirements • Requirements for Re-running or Performing New Simulations • Requirements for Re-running ReaxFF optimization • Python project setup	ning
1.2	Folder Structure	4
2	Enhancing ReaxFF database (Database Repository)	6
2.1	Installation	9
	Minimum Requirements • Requirements for Re-running or Performing New Simulations • Python project setup	
2.2	Interacting with the Database	9
	ASE db Command-line • ASE Python Interface • Web Interface	
2.3	Folder Structure	. 12
3	DFT simulations	14
3.1	DFT vs. ReaxFF mechanical deformation	. 18
4	Diffusion MSD analysis	18
5	Reparameterized ReaxFF	21
Ref	erences	24

# 1 Enhancing ReaxFF protocol (Protocol Repository)

In this repository, we collect and organize all the steps described in the main text for reparameterizing the ReaxFF potential. To facilitate the interaction with the protocol, we have divided the workflow, illustrated in Figure S1, into four main Jupyter Notebooks (JNBs) and an auxiliary notebook: JNB1-Initial\_configurations.ipynb, JNB2-Simulations.ipynb, JNB3-Build\_trainingset.ipynb,

preJNB4-ReaxFF\_optimization.ipynb, JNB4-ReaxFF\_optimization.ipynb. These provided



Figure S1. Protocol workflow overview

Jupyter Notebooks are specifically designed to streamline the process of configuration building (JNB1), DFT simulation execution (JNB2), database preparation (JNB3), and ReaxFF potential optimization (JNB4), as outlined in the workflow in Figure S1. The conditional part of the workflow is achieved using the fourth and auxiliary notebook since the auxiliary notebook (preJNB4-ReaxFF\_optimization.ipynb) is selected the subset of ReaxFF parameters related to a specific interaction and then fed to the JNB4, which will change it to minimize the loss function.

The repository is available both in Zenodo at the permanent link https://doi.org/10.5281/zenodo. 8036775, and on version control repositories host GitHub https://github.com/paolodeangelis/ Enhancing\_ReaxFF, Figure S2.

#### 1.1 Installation

The protocol strongly relies on the commercial code Amsterdam Modeling Suite (AMS) by Software for Chemistry & Materials (SCM)<sup>4</sup>. However, depending on which part of the protocol you aim to reproduce, it is possible to install only the minimal requirements, as described in Subsubsection 1.1.1, which allow for the generation and handling of atomistic simulations. To perform the simulations, additional requirements outlined in Subsubsection 1.1.2 are necessary. For the ReaxFF optimization, the Python library ParAMS<sup>5</sup> is indispensable, and it is currently available

a) Search or jump to	7 Pull requests Issues Codespaces Marketplace Explore	¢ +• ⊜•	b) Coord Cases a Uplad Communities	🔺 paolo.deangelis@polito.it 👻
paolodeangelis / Enhancing_F     Code	ReaxFF         Public         Q         Pin         O Unwatch         1           O         Actions         Ell Projects         O Security         Insights         Settings	v ¥ fork 0 v ☆ Star 0 v	June 14, 2023 Saturer Rended Acade	CZ Edit
₽ main • ₽ 1 branch ⊚1 tag	Go to file Add file * Code *	About 🕸	© De Angelis, Paolo; © Cappablanco, Roberta; © Fasano, Matteo; © Aslant, Pietro; © Chiavazzo, Chiavazzo	New Version
🧐 paolodeangelis 🥕 zenodo DOI	× zaeżska 7 hours ago 30 commits	Jupyter notebooks used for retraining the ReaxFF force field for the inorganic	Project manager(s) O De Angelis, Paolo	0 0 • views <b>±</b> downloads
assets/img	+ workflow sketch 15 hours ago	compound LIF.	Project member(s) O Cappabianca, Roberta	See more details
🖿 data	+ ReaxFF optimized for LiF yesterday	Readme	Supervisor(s)	
tools	+ python tools 9 hours ago	E <sup>2</sup> Cite this repository -	🐵 Fasano, Matteo; 💿 Asianri, Pietro; 😰 Chiavazzo, Chiavazzo	
D flake8	+ Setup project 3 days ago	★ Activity	This repository contains the jupyter notebooks used to re-parametrize the ReavFF force field for LIF, an inorganic compound. The parameter of the set of obtaining is to familitate the approximaterization ReavFF force field for LIF. The	Available in
	a madma uastaritu	\$≩r 0 stars	results and method used were published in the article Enhancing ReavEF for Molecular Dynamics Simulations of Lithium-	<b>A</b>
	A reason of the	<ul> <li>1 watching</li> </ul>	Ion Batteries: An interactive reparameterization protocol.	GitHub
pre-commit-conig.yami	+ setup project 3 days ago	¥ 0 forks	Enhancing ReaxFF repository DFT Database.	Ultiup
.remarkrc.yaml	+ Setup project 3 days ago		The Jupyter Notebooks provided here are designed to facilitate the configuration building, execution of DFT simulations,	
🗅 .zenodo.json	✓ zenodo DOI 7 hours ago	Releases	and optimization of the ReaxFF potential. To manage the complexity of the overall operation, we have divided the process into four main Jupyter Notebooks, with an additional auxiliary notebook.	
CITATION.cff	🖌 zenodo DOI 7 hours ago	🛇 1 tags	Each notebook can be executed independently, allowing for flexibility in adapting the workflow to optimize the ReaxFF	OpenAIRE
CREDITS.md	+ auxiliar files 10 hours ago	Create a new release	potential for different compounds with minimal modifications.	
JNB1-Initial_configrations.ipynb	Add comments 15 hours ago		Ine four main Jupyter Notebooks cover various aspects of the workflow, including configuration setup, UF I simulations, and Reax/FF optimization. By following the sequence of these notebooks, you will be able to efficiently perform the entire	
JNB2-Simulations.ipynb	+ JNB2 15 hours ago	Packages	process, from initial configuration generation to the optimization of the ReaxFF potential.	Publication data:
JNB3-Build_traingset.ipynb	+ JNB3 10 hours ago	No packages published	Installation To use the database and interact with it, ensure that you have the following Python requirements installed:	June 14, 2023
JNB4-ReaxFF_optimization.ipynb	+ JNB4 10 hours ago	Publish your first package	Minimum Requirements:	D01: 10.5781/zeepedp.80.35776
	se change LICENSE GPL-3 > CC-BY-4.0 3 days ago		Python 3.8 or above	Keyword(s):
C README med	a zanodo DOI 7 hours ano	Contributors 2	Atomic Simulation Environment (ASE) library     Jupyter Lab	Density Functional Theory DFT Force Field FF Lithern fluoride LiF Reactive Force Field Read/F
B instancing		DAP93	Requirements for Re-running or Performing New Simulations:	Solid electrolyte interphase SEI Uthium-ien battery
D packagejson	× nous ago	Angelia Paolo De Angelia	SCM (Software for Chemistry & Materials) Amsterdam Modeling Suite	LIB Atomic Simulation Environment ASE BIG-MAP Reproducible rassamb Protocols
preJNB4+Keax+F_optimization.ipynb	+ JNB4 9 nours ago	<b>W</b>	PLANIS (Python Library for Automating Molecular Simulation) library  Requirements for Re-pipeled ReavEE entimization;	Grants:
pyproject.toml	+ Setup project 3 days ago		AMS (Amsterdam Modeling Suite) Python Stack (more info here)	European Commission:     BIG-MAP - Battery Interface Genome -
requirements.txt		Languages	You can install the required Python packages using pip:	Materials Acceleration Platform (957189)
E README.md	0	Jupyter Notebook 100.0%	pip install -r requirements.txt	Related identifiers: Supplement to https://piths.fp.com/panlodeancelis/Enhancing. Re
Enhancing the Re	eaxFF protocol	Suggested Workflows Based on your tech stack	Warning This do not House the ParAMS packages since is available only in AMS Python Stack. See instruction below. expert SOL_PYTRI0D1845(pack)/verv	axFF/tree/v1.0.0-alpha
pre-commt.cl (talure)  output outp	A Chronic Sector Con International Works	Automatically convert CVCD files to YAML for Github Autions. dj Django Configure	avocai wany roominstrut_venu source verview(RYOTYY).X.vervi28in/etclivate # Replace YYY).X sidth the correct 4/6 version Waning Make see to have the apoptratise loomes and installations of SCM Antiterfam Modeling Suite and any other necessary software for unrung annulations.	Version: 1.0.0-alpha Jun; 14, 2023 10:5281/zerodo: 80305776 Cite all versions? You can cite all versions by using the DOI 10:5281/zerodo: 8036775. This DOI represents all versions.
The database containing the simula	stion data from <i>ab initio</i> simulations obtained from this protocol is published	Build and Test a Django Project	Configure Patterial Project APT Get Material Project APF kg Access to Material Device and follow the intrustice on the documentation	and will aways resolve to the satest one. Head more.
The Jupyter Notebooks provided he simulations, and optimization of the manage the complexity of the over- with an additional auxiliary onethor	ere are designed to facilitate the configuration building, execution of DFT eRearF potential, as described in the workflow diagram shown below. To all operation, we have divided the process into four main Jupyter Notebooks, or	Publish Python Configure Publish a Python Package to PyPI on release.	Access to Notified Fright and John the entrybulation of the inclusion and the entrybulation of the 1. Run the script reglacing (INTERLE) PROTECT_EXP: with the Material Project API key, python tool/replacer/scs_sistup.gy -k_dWIETLE_PROTECT_EXP:	Share Cite as De Angelis, Paolo, Cappabianca, Roberta, Fasano, Matteo, Asiani, Pietro, & Chiavazzo, Chiavazzo,
	Start	More workflows Dismiss suggestions	Folder Structure The repository has the following folder structure:	(2023). Enhancing the Reax/FF protocol (1.0.0- alpha). Zenodo. https://doi.org/10.5281/zenodo.9036776
JNB 1 Using the crystal's the Material Proje stable and metast Generation of initi	c database via its API for all able unit structures.		- - CROTTS.ed - LICEOR - READER.ed - read-ream.tst - read- -	Start typing a citation style
Lusing AGE and py	inaugen inzianea.		- tools	JSON JSON-LD GeoJSON MARCXML

Figure S2. Screenshot of the protocol hosted on GitHub (a) and Zenodo (b).

only within the AMS Python Stack, as explained in Subsubsection 1.1.4.

#### 1.1.1 Minimum Requirements

- Python 3.8 or above
- Atomic Simulation Environment (ASE) library
- Jupyter Lab

#### 1.1.2 Requirements for Re-running or Performing New Simulations

- SCM (Software for Chemistry & Materials) Amsterdam Modeling Suite
- PLAMS (Python Library for Automating Molecular Simulation) library

#### 1.1.3 Requirements for Re-running ReaxFF optimization

• AMS (Amsterdam Modeling Suite) Python Stack (more info here). To utilize the AMS Python Stack for this project, we recommend creating a Python virtual environment as follow:

```
export SCM_PYTHONDIR=$(pwd)/venv
$AMSBIN/amspython --install_venv
source venv/AMSYYYY.X.venv/bin/activate # Replace YYYY.X with the correct AMS
version
```

#### 1.1.4 Python project setup

Regardless of the software available, the next step is to install all the additional Python packages (such as pymatgen, ASE, etc.) required for the protocol. This can be accomplished using the Python package manager, pip, through the following terminal command:

(venv)\$ pip install -r requirements.txt

Then it is necessary to set up the Material Project "API keys" to enable the code to query the database. To obtain the API key, follow the instructions provided in the documentation, which can be found here. After obtaining the API key, replace <MATERIAL\_PROJECT\_KEY> with your actual API key and run the script as shown below:

(venv)\$ python tools/mpinterface\_setup.py -k <MATERIAL\_PROJECT\_KEY>

Finally, to utilize the Jupyter Notebooks, it is necessary to run the Jupyter Lab server using the following command:

(venv)\$ jupyter lab

# 1.2 Folder Structure

The repository is structured as follows:

```
assets
data
ffield.reax.optimized.ff
tools
...
CREDITS.md
LICENSE
README.md
requirements.txt
JNB1-Initial_configrations.ipynb
JNB2-Simulations.ipynb
JNB3-Build_traingset.ipynb
JNB4-ReaxFF_optimization.ipynb
```

In the folder tree, we have excluded certain auxiliary files and folders that are not essential for understanding the repository, and below we explain in detail the content of each folder/file:

- assets: This folder contains any additional assets, such as images or documentation; related to the repository.
- data: This folder contains the data files resulting from this work.
  - ffield.reax.optimized.ff: This file is the optimized ReaxFF resulting from using these Jupyter notebooks, as explained in the main text, and as reported in Section 5.

- tools: This directory contains a collection of Python modules and scripts.
- CREDITS.md: This file acknowledges and credits each author who contributed to the repository.
- LICENSE: This file contains the license information for the repository (CC BY 4.0). It specifies the terms and conditions under which the repository's contents are distributed and used.
- README.md: This is the "readme" file (repository overview and instructions).
- requirements.txt: This file lists the required Python packages and their versions (see Subsection 1.1).
- JNB1-Initial\_configurations.ipynb: Jupyter Notebook where the protocol is initialized by querying the *Materials Project* database, downloading the unit crystals, and producing all the initial configurations for the DFT simulations using the *pymatgen* library.
- JNB2-Simulations.ipynb: This notebook performs the simulations using BAND and DFTB codes available in the Amsterdam Modeling Suite. The simulations are performed in parallel using the *PLAMS* library and *SLURM* scheduler.
- JNB3-Build\_trainingset.ipynb: Here, the quantities needed for the database are extracted and tuned to favor accuracy on the energy.
- preJNB4-ReaxFF\_optimization.ipynb: This is an auxiliary Notebook where the old ReaxFF is converted into a Python object, and it is possible to select the subset of coefficients related to specific interactions to change during the optimization (e.g., bond, van der Waals, angular, etc.).
- JNB4-ReaxFF\_optimization.ipynb: This notebook takes the database and the ReaxFF Python object to perform a multi-objective optimization and find the new ReaxFF potential that minimizes the Sum of Squared Errors (SSE).

# 2 Enhancing ReaxFF database (Database Repository)

In addition to the protocol, we are sharing the complete database used to reparameterize ReaxFF. This allows for testing the optimization without rerunning time-consuming simulations and enables future expansion and improvement by the community. We stored the data using the ASE SQLite3 database, which was chosen for its widespread use in the computational community of European institutions. The use of this type database provides fast and flexible querying and browsing capabilities, as described in Subsection 2.2. The database contains information on initial configurations, single-point calculations, pre-optimizations, and DFT optimizations. However, only a subset of the database is used in the training set for the ReaxFF optimization, as explained in the main text and briefly summarized in Table S1. For each entry in the database, we store various information regarding the simulation and

**Table S1.** Training set composition, with quantities calculated using DFT: Energy (E), Forces (F), and Charges (Q). The numbers in parentheses indicate the successful cases for self-consistent field (SCF) calculations that may not always converge.

Tuno		Unit cell		Q	uan	tities
туре	Fm3m	P6 <sub>3</sub> mc	Pm3m	E	F	Q
Supercells	6	6	6	$\checkmark$	$\checkmark$	$\checkmark$
Vacancies	10	24	10	$\checkmark$		$\checkmark$
Strain	39	39	39	$\checkmark$		$\checkmark$
Substitution	5	5	5	$\checkmark$		$\checkmark$
Interstitial	5	5	5	$\checkmark$	$\checkmark$	$\checkmark$
Slabs	12(10)	12(8)	12(11)	$\checkmark$	$\checkmark$	$\checkmark$
Bulk 300K	10	10	10	$\checkmark$		
Bulk 500K	10	10	10	$\checkmark$		
Amorphous	10	10	10	$\checkmark$		
Total	107(105)	121(117)	107(106)			

properties obtained from the DFT calculations. These details are listed in Table S2.

The repository is available both in Zenodo at the permanent link https://doi.org/10.5281/zenodo. 7959121, and on version control repositories host GitHub https://github.com/paolodeangelis/ Enhancing\_ReaxFF\_DFT\_database, Figure S3.



Figure S3. Screenshot of the database hosted on GitHub (a) and Zenodo (b).

name	description	type	unit
id	Uniqe row ID	integer	_
ctime	Creation time of the data (for simulation, it coincides with	float	yr
	the runtime)		
formula	Chemical formula of the system	string	_
pbc	Periodic boundary conditions	boolean	_
user	Username or full name of the user who created the data	string	_
calculator	Name of the ASE-calculator and engine used to calculate	string	_
	the system (e.g ams/band means AMS calculator with		
	BAND engine)		
energy	Total energy of the system from the calculation	float	eV
natoms	Number of atoms	integer	_
fmax	Maximum force	float	$eV Å^{-1}$
smax	Maximum stress	float	$eV/Å^3$
charge	Net charge in unit cell	float	e
mass	Sum of atomic masses in unit cell	float	au
magmom	Magnetic moment	float	$\mu_{ m B}$
unique_id	Random (unique) ID	integer	_
volume	Volume of unit cell	float	$Å^3$
functional	Exchange-and-correlation functional	string	_
fermi_energy	Fermi Energy (N.B. is not the Fermi Level), which indicates	float	eV
	the energy of non-interacting fermions in the system (Fermi		
	gas) at 0K		
homo_energy	Highest Occupied Molecular Orbital energy	float	eV
lumo_energy	Lowest Unoccupied Molecular Orbital energy	float	eV
band_gap	Band gap energy (for LiF is the HOMO energy - LUMO	float	eV
	energy)		
runtime	Simulation start date whit format %b%d-%Y %H:%M:%S	string	—
elapsed	Elapsed Time	float	S
name	System Name	string	_
sim_name	Simulation Name	string	_
subset_name	Name of the subset of configuration	string	_
run_script	Full AMS scrip for running the simulation	string	_
input_script	Full input scrip for running the simulation	string	_
success	Simulation end status	boolean	_
used_in	Indicate in which set (training-set or test-set) the data was	string	—
	used		
task	Simulation task type	string	_
space_group	Full International Space Group Symbol. The notation is a	string	—
	LaTeX-like string, with screw axes being represented by an		
	underscore		
data	Additional data calculate: Density of states(DOS) and Hys-	dictionary	—
	troy (e.g. energy, force evolution during the simulation)		

Table S2. Datailed list of properties and information store in each database entry

# 2.1 Installation

To browse and query the database, only Python and the ASE and Jupyter Lab libraries are required (see Subsubsection 2.1.1). With this minimal setup, it is possible to perform additional simulations using the preferred DFT engine. Using ASE, you can wrap the simulations and add them to the database. For an example, you can refer to the notebooks\running\_simulation.ipynb Jupyter Notebook, which demonstrates the usage of the BAND plane-wave DFT code. However, if you wish to rerun the simulations stored in the database, the commercial code Amsterdam Modeling Suite (AMS) is required (refer to Subsubsection 2.1.2).

## 2.1.1 Minimum Requirements

- Python 3.8 or above
- Atomic Simulation Environment (ASE) library
- Jupyter Lab

## 2.1.2 Requirements for Re-running or Performing New Simulations

- SCM (Software for Chemistry & Materials) Amsterdam Modeling Suite
- PLAMS (Python Library for Automating Molecular Simulation) library

## 2.1.3 Python project setup

Similar to the protocol repository, the project setup requires the installation of additional Python packages such as ASE, PLAMS, etc. These packages can be installed using the Python package manager, pip, with the following terminal command:

(venv)\$ pip install -r requirements.txt

Then it is possible to utilize the Jupyter Notebooks by starting the Jupyter Lab server using the following command:

(venv)\$ jupyter lab

## 2.2 Interacting with the Database

There are three methods available for interacting with the database: using the ASE db command line, the web interface, and the ASE Python interface.

## 2.2.1 ASE db Command-line

To interact with the database through the ASE db command line, follow these steps:

- 1. Open a terminal and navigate to the directory where the LiF.db file is located.
- 2. Execute the following command to initiate the ASE db terminal:

env)\$ ase db LiF.db

3. Now, one can utilize the available commands within the terminal to query and manipulate the database.

(venv) paol	o@DESK		PyProjects	/Enhancing_Re	axFF_DFT_		a\$ ase (	db LiF.db –L 8				
id age user		formula	calculato	r  energy nat	oms  fmax	pbc volume	charge	mass  smax				
1 18M Paol	o De A	ngelis LiF				TTT 17.022	0.000 2	25.938				
2 18M Paol	o De A	ngelis LiF	ams/band	-9.623	2 0.000	TTT 17.022	0.000 :	25.938 0.012				
3 18M Paol	o De A	ngelis Li2F2				TTT 42.224	0.000	51.877				
4 18M Paol	o De A	ngelis Li2F2	ams/band	-19.209	4 0.039	TTT 42.224	0.000	51.877 0.007				
5 18M Paol	o De A	ngelis LiF				TTT 16.768	0.000 :	25.938				
6 18M Paol	o De A	ngelis LiF	ams/band	-9.024	2 0.000	TTT 16.768	0.000 2	25.938 0.008				
7 18M Paol	o De A	ngelis LiF	ams/band	-9.626	2 0.000	TTT 16.499	0.000 2	25.938 0.001				
8 18M Paol	o De A	ngelis Li2F2	ams/band	-19.214	4 0.007	TTT 41.420	0.000	51.877 0.001				
Rows: 706 (	showin	g first 8)										
Keys: band_	gap, e	lapsed, fermi_	energy, fu	nctional, hom	no_energy,	input_scrip	t, lumo_	_energy, name,	run_script,	runtime,	sim_name,	space_
group, subs	et_nam	e, success, ta	sk, used_i	n								
(venv) paol	o@DESK		PyProjects	/Enhancing_Re	axFF_DFT_		a\$ ase o	db LiF.db id=2				
FLi:												
Unit cell i	n Ang:											
axis period	ic		Уİ		length	angle						
1  y	es	2.501	0.000	1.444	2.887	60.000						
2  y	es	0.834	2.358	1.444	2.887	60.000						
3  y	es	0.000	0.000	2.887	2.887	60.000						

Figure S4. Example of interacting with the database via the terminal.

### 2.2.2 ASE Python Interface

To interact with the database using the ASE Python interface, you can utilize the following code example:



Figure S5. Example of interacting with the database via Python, utilizing the iPython terminal interface.

For a more detailed example, refer to the notebook notebooks\browsing\_db.ipynb. To learn how to perform a simulation, check the notebook notebooks\running\_simulation.ipynb.

#### 2.2.3 Web Interface

To interact with the database using the web interface, follow these steps:

- 1. Open a terminal and navigate to the directory where the LiF.db file is located.
- 2. Execute the following command to initiate the ASE db terminal:

(venv)\$ ase db -w LiF.db

- 3. Access the web browser and connect to the local server at http://127.0.0.1:5000.
- 4. Upon accessing the interface, Figure S6.a will be displayed, showcasing all the entries in the database.
- 5. To query the database and filter the entries based on specific criteria, users can input their search parameters into the designated fields (e.g. success=True).
- 6. Once the query is executed, Figure S6.b will be displayed, presenting the selected entries that match the search criteria.
- 7. To access all the information of an individual entry, users can click on a specific entry to view a detailed visualization. Figure **S**7 demonstrates an example of a single entry displayed through the web interface.

a) <sub>Li</sub>	) FC	Datas	et for Rea	kFF pa	arar	netriza	ation								k L	) _iF	Da	tase	et for Reax	(FF pa	rar	netriza	ation							
	Se Help Togg	iarch formulae e.g. MoS2 Q with construent guidanced learch queries In fair drivey												H	succes elp with a oggie list o	s=True constructing of keys	g advanced search queries	•			٩									
Di	splayin	ig rows 1-25	out of 706									Rows: :	25 🕶 🕴 Add C	olumn 🕶		Display	ying roi	ws 1-25 o	out of 377 (direct link)									Rows:	25 🕶 🗌 Add C	Xolumn 🕶
ID ×	Age ×	Formula ×	System Name X	Calculator ×	PBC ×	Task ×	Maximum force ×	Volume ×	Charge ×	Energy ×	Simulation status X	Elapsed Time X	Subset Name X	Used in X		ID Ag × ×	e For	rmula ×	System Name X	Calculator ×	PBC ×	Task X	Maximum force X	Volume ×	Charge ×	Energy ×	Simulation status X	Elepsed Time X	Subset Name X	Used in X
1	18M	LiF	0-LiF_Fm-3m3.18		тт	initial configuration		17.022	0.000				unit cell	none		2 18	м і	LiF	0-LiF_Fm-3m3.18	ams/band	TTT	single point	0.000	17.022	0.000	-9.623	True	518.069	unit cell	none
2	18M	LIF	0-LiF_Fm-3m3.18	ams/band	TTT	single point	0.000	17.022	0.000	-9.623	True	518.069	unit cell	none		4 18 6 18	M LI M I	LZF2	2.1 iE Pm-3m -2.89	amsband	TT	single point	0.009	42.224	0.000	-19.209	True	447 538	unit cell	none
3	18M	Li2F2	1-LIF_P6_3mc3.17	Ψ.	ш	initial configuration		42.224	0.000			-	unit cell	none	5	7 187	M I	LIF	0-LiF_Fm-3m3.18	ams/band	π	geometry	0.000	16.499	0.000	-9.626	True	1126.407	unit cell	trainin
4	18M	Li2F2 LiF	1-LiF_P6_3mc3.17 2-LiF_Pm-3m -2.89	ams/band	TTT	single point	0.039	42.224	0.000	-19.209	True	547.002	unit cell	none	1	8 18	M Li	2F2	1-LIF_P6_3mc3.17	ams/band	тт	geometry	0.007	41.420	0.000	-19.214	True	1020.345	unit cell	test
						configuration										9 18/	мі	LiF	2-LiF_Pm-3m2.89	ams/band	ш	geometry	0.000	16.533	0.000	-9.024	True	745.817	unit cell	test
6	18M	LIF	2-LIF_Pm-3m2.89	ams/band	TTT	single point	0.000	16.768	0.000	-9.024	True	447.538	unit cell	none								optimization								
<i>'</i>	18M	LIF	0-LIF_FM-3M3.18	ams/band		optimization	0.000	16.499	0.000	-9.626	True	1126.407	unit cell	training		11 17	м	F4	0-F_C2c_na4	ams/band	10	single point	1.2/1	76.301	0.000	-7.120	True	60.805	elements unit cell	trainin
8	18M	Li2F2	1-LiF_P6_3mc3.17	ams/band	ш	geometry optimization	0.007	41.420	0.000	-19.214	True	1020.345	unit cell	test	1	13 17	м	Li3	1-Li_R-3m_na3	ams/band	TTT	single point	0.008	60.335	0.000	-5.470	True	132.487	pure elements	training
9	18M	LIF	2-LiF_Pm-3m2.89	ams/band	TTT	geometry optimization	0.000	16.533	0.000	-9.024	True	745.817	unit cell	test															unit cell	
10	17M	F4	0-F_C2c_na4		TTT	initial configuration	*	76.301	0.000	•			pure elements unit cell	none		14 177	м	F4	0-F_C2c_na4	ams/band	ш	geometry optimization		96.959	0.000	-7.182	True	1706.927	pure elements unit cell	none
11	17M	F4	0-F_C2c_na4	ams/band	TTT	single point	1.271	76.301	0.000	-7.120	True	60.805	pure elements	training		15 177	M	Li3	1-Li_R-3m_na3	ams/band	Π	geometry optimization		63.800	0.000	-5.477	True	3780.882	pure elements unit cell	none
10	1714	1.2	111 0.2m and			(million)		00 225	0.000				unit cell		1	17 188	M Li	2F2	1.0-0-LIF_Fm-	ams/band	TTT	single point	0.000	34.044	0.000	-19.251	True	350.506	supercell	training

**Figure S6.** Screenshot showing the web interface of the database, displaying all the entries (a), and filtering the entries after querying the database (b).



Figure S7. Screenshot displaying a entrie from the database visualized using the web interface.

## 2.3 Folder Structure

The repository is structured as follows:

```
assets
data
   -LiF.db
  — LiF.json
  └_LiF.yaml
notebooks 🗇
  -browsing_db.ipynb
  L running_simulation.ipynb
- tools
  - db
  plams experimental
   scripts
. . .
-CONTRIBUTING.md
-CREDITS.md
-LICENSE
-README.md
-requirements.txt
```

- assets: This folder contains any additional assets, such as images or documentation, related to the repository.
- data: This folder contains the data files used in the repository.
  - LiF.db: This file is the SQLite database file that includes the DFT data used for the ReaxFF force field. Specifically, it contains data related to the inorganic compound LiF.
  - LiF. json: This file provides the database metadata in a human-readable format JSON.
  - LiF.yaml: This file also contains the database metadata in a more human-readable format YAML.
- notebooks: This folder contains Jupyter notebooks that provide demonstrations and examples of how to use and analyze the database.
  - browsing\_db.ipynb: This notebook demonstrates how to handle, select, read, and understand the data points in the LiF.db database using the ASE database Python interface. It serves as a guide for exploring and navigating the database effectively.

- running\_simulation.ipynb: In this notebook, you will find an example of how to get a data point from the LiF.db database and use it to perform a new simulation. The notebook showcases how to utilize either the PLAMS library or the AMSCalculator and ASE Python library to conduct simulations based on the retrieved data and then store it as a new data point in the LiF.db database. It provides step-by-step instructions and code snippets for a seamless simulation workflow.
- tools: This directory contains a collection of Python modules and scripts that are useful for reading, analyzing, and re-running simulations stored in the database. These tools are indispensable for ensuring that this repository adheres to the principles of Interoperability and Reusability, as outlined by the FAIR principles.
  - db: This Python module provides functionalities for handling, reading, and storing data into the database.
  - plasm\_experimental: This Python module includes the necessary components for using the AMSCalculator with PLASM and the SCM software package, utilizing the ASE API. It facilitates running simulations, and performing calculations.
  - scripts: This directory contains additional scripts for advanced usage scenarios of this repository.
- CONTRIBUTING.md: This file provides guidelines and instructions for contributing to the repository. It outlines the contribution process, coding conventions, and other relevant information for potential contributors.
- CREDITS.md: This file acknowledges and credits each author who contributed to the repository.
- LICENSE: This file contains the license information for the repository (CC BY 4.0). It specifies the terms and conditions under which the repository's contents are distributed and used.
- README.md: This is the "readme" file (repository overview and instructions).
- requirements.txt: This file lists the required Python packages and their versions. (see Subsection 2.1)

# **3 DFT simulations**

In this section, we present the results of our DFT simulations, highlighting the energy obtained for various configurations.



**Figure S8.** Comparison of energy per atom obtained from DFT simulations (SP: Single Point, GO: Geometry Optimization) with those stored on Material Project (a), for each primitive unit cell of LiF (b).



Figure S9. Energy per atom obtained from DFT simulations for each crystal in the LiF supercell.



**Figure S10.** Energy per atom curves obtained from strained configurations, specifically with 1D strain  $\varepsilon_{11}$ , shear strain  $\varepsilon_{12} = \varepsilon_{21}$ , and homogeneous expansion/compression  $\varepsilon_{11} = \varepsilon_{22} = \varepsilon_{33}$ .



Figure S11. Energy per atom obtained from DFT simulations for each system where a vacancy defect was added.



**Figure S12.** Energy per atom obtained from DFT simulations for each system where a substitution defect was added.



**Figure S13.** Energy per atom obtained from DFT simulations for each system with an added interstitial defect. Due to the significant crystal deformation introduced, the optimization process was conducted in two steps: Pre-optimization (blue) and full geometry optimization (green).



**Figure S14.** Comparing energy per atom for different crystal plates ([0,0,1], [0,1,1], [1,1,1], [0,1,2]) with varying slab thickness. The plot illustrates how the energy per atom changes as a function of the inverse of the number of slabs, emphasizing the impact of surface energy.



**Figure S15.** Evolution of temperature and energy per atom in *ab initio* NVT-MD simulation at 2500K with Non-Scc-GFN1xTB DFTB model<sup>6</sup> for sampling amorphous LiF systems.



**Figure S16.** Evolution of temperature and energy per atom in *ab initio* NVT-MD simulation at 300 K with Non-Scc-GFN1xTB DFTB model<sup>6</sup> for sampling solid LiF systems.



**Figure S17.** Evolution of temperature and energy per atom in *ab initio* NVT-MD simulation at 500 K with Non-Scc-GFN1xTB DFTB model<sup>6</sup> for sampling solid LiF systems.





**Figure S18.** Comparing the mechanical response of LiF: ReaxFF and DFT Predictions. The black line represents DFT results used for training, while the blue, green, and red lines depict the energy predictions by ReaxFF models from Yun et al.<sup>7</sup>, Wang et al.<sup>8</sup>, and our proposed new reparameterization, respectively. We examine the energy variation with respect to the equilibrium crystal for three types of deformation: tensile strain deformation  $\varepsilon_{11}$  (left), shear strain deformation  $\varepsilon_{12} = \varepsilon_{21}$  (center), and homogeneous lattice length change  $V/V_0$  (where  $V_0$  is the equilibrium volume). This study was performed on all stable and metastable crystals with space groups Fm $\bar{3}$ m (a), Pm $\bar{3}$ m (b), and P6<sub>3</sub>mc (c) from the Material Project database<sup>9</sup>.

## 4 Diffusion MSD analysis

Presented here is a plot illustrating the Mean Square Displacement (MSD) analysis conducted to investigate Li diffusion using ReaxFF simulatios.



**Figure S19.** Evolution of Mean Square Displacement (MSD) during NVT-MD ReaxFF simulations of Pure Bulk LiF (LiF) and point defects: vacancy (Li<sub>0.9</sub>F) and interstitial (Li<sub>1.1</sub>F) at 300 K. The markers (square, circle, and triangular) represent the corresponding systems in the plot. The simulations were performed using ReaxFF parameterizations by Yun et al.<sup>7</sup> (blue), Wang et al.<sup>8</sup> (green), and our proposed new parameterization (red). The dashed line indicates the linear model ( $MSD = 6D \cdot t$ ) employed to compute diffusivity in each simulation.



**Figure S20.** Evolution of Mean Square Displacement (MSD) during NVT-MD ReaxFF simulations of Pure Bulk LiF (LiF) and point defects: vacancy (Li<sub>0.9</sub>F) and interstitial (Li<sub>1.1</sub>F) at 400 K. The markers (square, circle, and triangular) represent the corresponding systems in the plot. The simulations were performed using ReaxFF parameterizations by Yun et al.<sup>7</sup> (blue), Wang et al.<sup>8</sup> (green), and our proposed new parameterization (red). The dashed line indicates the linear model ( $MSD = 6D \cdot t$ ) employed to compute diffusivity in each simulation.



**Figure S21.** Evolution of Mean Square Displacement (MSD) during NVT-MD ReaxFF simulations of Pure Bulk LiF (LiF) and point defects: vacancy (Li<sub>0.9</sub>F) and interstitial (Li<sub>1.1</sub>F) at 500 K. The markers (square, circle, and triangular) represent the corresponding systems in the plot. The simulations were performed using ReaxFF parameterizations by Yun et al.<sup>7</sup> (blue), Wang et al.<sup>8</sup> (green), and our proposed new parameterization (red). The dashed line indicates the linear model ( $MSD = 6D \cdot t$ ) employed to compute diffusivity in each simulation.

# 5 Reparameterized ReaxFF

1	! ReaxFF force field C/H/O/Si/Li/F by P. De Angelis (reparameterized from ReaxFF by Yun,
	Kang-Seop, et al. (2017).).
2	39 ! Number of general parameters
3	50.0000 !Overcoordination parameter
4	9.5469 !Overcoordination parameter
5	26.5405 !Valency angle conjugation parameter
6	1.7224 !Triple bond stabilisation parameter
7	6.8702 !Triple bond stabilisation parameter
8	60.4850 !C2-correction
9	1.0588 !Undercoordination parameter
10	4.6000 !Triple bond stabilisation parameter
11	12.1176 /Undercoordination parameter
12	13.3056 !Undercoordination parameter
13	-70.5044 !Triple bond stabilization energy
14	0.0000 !Lower Taper-radius
15	10 0000 /Upper Taper-radius
16	2 8793 /Not used
17	33 8667 Valency undercoordination
18	6 0891 !Valency angle/lone pair parameter
19	1 0563 !Valency angle
20	2 0384 [Valency angle parameter
20	6 1431 Not used
21	6 9290 !Double bond/angle parameter
22	0.3989 !Double bond/angle parameter: overcoord
23	3 9954 [Double bond/angle parameter: overcoord
25	-2 4837 Not used
25	5 7796 !Torsion/BO parameter
20	10 0000 !Torsion overcoordination
27	1 9487 !Torsion overcoordination
29	-1.2327 [Conjugation 0 (not used)
30	2 1645 /Conjugation
31	1 5591 LydWaals shielding
32	0 0010 /Cutoff for bond order (+100)
33	2 1365 [Valency angle conjugation parameter
34	0 6991 !Overcoordination parameter
35	50 0000 !Overcoordination parameter
36	1 8512 !Valency/lone pair parameter
37	0 5000 !Not used
38	20.0000 !Not used
30	5 0000 !Molecular energy (not used)
39 40	0.0000 !Molecular energy (not used)
40	2 6962 !Valency angle conjugation parameter
12	6   Nr of atoms: cov r: valency: a m : Rvdw: Evdw: gammaEEM: cov r2: #el
13	l alfa: gammaydW: valency13: Eunder: Fover: chiEEm: etaEEM: n ul
43	L cov r3. Eln. Heat inc. $13B01 \cdot 13B02 \cdot 13B03 \cdot n u2 \cdot n u3$
45	$v_{\text{un: vval1: vval2: vval3: vval4: n u5: n u6: n u7}$
46	C = 1 3825 + 0000 + 12 0000 + 9133 + 01853 + 0.9000 + 1.1359 + 0.000
40	9 7602 2 1346 4 0000 33 2433 79 5548 5 8678 7 0000 0 0000
 /18	1 2104 0 0000 199 0303 8 6991 34 7289 13 3894 0 8563 0 0000
49	
50	H 0.7853 1.0000 1.0080 1.5904 0.0419 1.0206 -0.1000 1.0000
51	9.3557 5.0518 1.0000 0.0000 121.1250 5.3200 7.4366 1.0000
52	
53	-15.7683 2.1488 1.0338 1.0000 2.8793 0.0000 0.0000 0.0000
54	0 1.2450 2.0000 15.9990 2.3890 0.1000 1.0898 1.0548 6.0000
55	9.7300 13.8449 4.0000 37.5000 116.0768 8.5000 8.3122 2.0000
56	0.9049 0.4056 59.0626 3.5027 0.7640 0.0021 0.9745 0.0000
57	-3.5500 2.9000 1.0493 4.0000 2.9225 1.3000 0.2000 13.0000

58	Si	2	2.2902	4.0000	28.0600	1.	8354	0	.2110	) (	).5947	1.	2962	4.0000
59		11	L.1336	3.1831	4.0000	21.	7115	139	.9309	) 4	1.2033	5.	5558	0.0000
60		-1	L.0000	0.0000 1	04.0000	9.	0751	23	.8188	3 (	.8381	0.	8563	0.0000
61		_ 4	4.1684	2.0754	1.0338	4.	0000	2	.5791	. 1	.4000	0.	2000 1	3.0000
62	Li	-	L.9205	1.0000	6.9410	1.	8896	0	.0905	5 (	.4668	-0.	1000	1.0000
63		(	9.9084	1.0896	1.0000	0.	0000	0	.0000	) -6	5.4188	15.	0000	0.0000
64		_	1.0000	0.0000	37.5000	5.	4409	6	.9107	7 (	).1973	0.	8563	0.0000
65		-24	5 0000	2 2989	1 0338	1	0000	2	8103	× ۲	3000	0	2000 1	3 0000
66	F	2	2705	1 0000	18 998/	1	1100	0	0112	, , , ,	0135	-0	1000 1	7 0000
60	Ľ	1 -	6107	5 4401	4 0000	£.	1/72	0	1252		5 1220	17	2000	0.0000
67		- 1	1 0000	25 0000	1 5000	6	0001	0	1700	·	0561	± / •	0000	0.0000
68			2.0000	33.0000	1.0400	0.	9021	4	.1/95		0.000	0.	0000	0.0000
69	0.1	-2	2.2869	2./340	1.0493	4.	0000	3	.0013	5 (	1.0000	0.	0000	0.0000
70	21		! Nr oi	bonds; E	disi; Ed	1SZ;	Edis	33;	pbel;	pr	005; 13	corr	; ppo6;	KOV
71	_		! pbe2;	pbo3; p	bo4; n.u	1; p.	bol;	pbo	2; ov	cori	; n.u2			
72	1	1	156.595	3 100.039	7 80.00	00	-0.81	57	-0.4	1591	1.00	00	37.7369	0.4235
73			0.452	7 -0.100	0 9.26	05	1.00	000	-0.0	)750	6.83	16	1.0000	0.0000
74	1	2	170.231	6 0.000	0 0.00	00	-0.59	931	0.0	0000	1.00	00	6.0000	0.7140
75			5.226	7 1.000	0 0.00	00	1.00	000	-0.0	)500	6.83	15	0.0000	0.0000
76	1	4	94.591	2 50.119	7 0.00	00	-0.57	12	-0.5	558	1.00	00	17.2117	0.0308
77			2.395	1 -1.189	2 8.64	03	1.00	00	-0.1	028	5.42	78	1.0000	0.0000
78	2	2	156.097	3 0.000	0 0.00	00	-0.13	377	0.0	0000	1.00	00	6.0000	0.8240
79			2.990	7 1.000	0 0.00	00	1.00	00	-0.0	)593	4.83	58	0.0000	0.0000
80	1	3	224.829	3 31.884	7 89.54	56	-1.49	2.5	-0.0	850	1.00	0.0	13.4838	1,1008
81			0.738	7 -0.722	8 5.19	53	1.00	000	-0.1	175	6.43	19	0.0000	0.0000
82	З	З	1/12 285	, 0,22 8 1/5 000	0 50 82	93	0.25	06	-0 1	000	1 00	10	29 7503	0 6051
02 02	5	5	142.205	1 - 0 105	5 0.02	00	1 00		_0.1	225	5 50	20	1 0000	0.0001
85	2	S	160 000	0.105	0 0 00	00	1.00	100	-0.1	223	1 00	20	L.0000	0.0000
84	2	2	1 115	0 0.000	0 0.00	00	-0.57	20	0.0	0000	1.00	00	0.0000	0.0020
85	0		1.115	0 1.000	0 0.00	00	0.00	000	-0.0	1920	4.2/	90	0.0000	0.0000
86	2	4	101.184	0 0.000	0 0.00	00	-0.1/	51	0.0	0000	1.00	00	6.0000	1.1044
87			7.354	9 1.000	0 0.00	00	1.00	000	-0.0	)450	7.90	80	0.0000	0.0000
88	3	4	274.833	9 5.000	0 0.00	00	-0.58	884	-0.3	3000	1.00	00	36.0000	0.2131
89			9.977	2 -0.257	2 28.81	53	1.00	000	-0.1	130	8.47	90	6.0658	0.0000
90	4	4	61.112	7 85.814	6 30.00	00	-0.81	97	-0.3	3000	1.00	00	16.0000	0.1386
91			0.130	7 -0.805	5 7.12	48	1.00	00	-0.0	674	8.23	74	0.0000	0.0000
92	1	5	10.054	0.000	0 0.00	00	0.30	05	-0.3	3000	0.00	00	6.0000	0.2953
93			0.267	9 -0.253	4 12.00	19	1.00	00	-0.1	143	7.52	11	0.0000	0.0000
94	2	5	63.464	9 0.000	0 0.00	00	0.02	94	0.0	0000	0.00	00	6.0000	0.4868
95			0.309	0 0.000	0 12.00	00	1.00	000	-0.0	0080	5.10	33	0.0000	0.0000
96	5	3	78.366	6 -0.020	0 0.00	0.0	-1.00	0.00	0.3	3000	0.00	0.0	6.0000	0.3228
97	0	Ŭ	0 202	2 -0.250	0 11 99	65	1 00	000	-0 1	276	7 86	56	0 0000	
08	5	Λ	23 196	3 0 000	0 0 00	00	1 00		0.3	2000	0 00	10 10		0 5185
00	0	1	0 0.81	2 0 000	0 12 00	00	1 00		-0 1	1/2	6 05	25		
100	5	F	16 744	2 0.000	0 12.00	00	1 00	120	0.1	2000	0.00	20	26 0650	0.0000
100	5	5	1 (12	3 0.000	0 0.00 E 10.00	00	1.04	120	0.3		4 70	00	20.0005	0.0000
101	1	c	1.013	7 -0.080	5 12.00	00	1.00	100	-0.1	.040	4.78	97		1 2151
102	T	6	100.880	0 0.000		00	0.4/	53	-0.5	0000	1.00	00	35.0000	1.3151
103		_	3.130	3 -0.250	0 15.00	00	1.00	000	-0.9	94/5	/.11	88	1.0000	0.0000
104	6	2	260.689	2 0.000	0 0.00	00	-0.66	90	0.0	0000	1.00	00	6.0000	2.3211
105			8.893	7 1.000	0 0.00	00	1.00	000	-0.3	3966	9.20	31	0.0000	0.0000
106	6	3	99.906	5 0.000	0 0.00	00	-1.00	000	0.0	0000	1.00	00	6.0000	0.7194
107			0.592	8 1.000	0 0.00	00	1.00	000	-0.1	318	8.42	78	0.0000	0.0000
108	6	6	65.856	3 0.000	0 0.00	00	-0.06	530	-0.5	5267	1.01	16	30.2616	0.2245
109			-0.230	6 -0.216	1 15.39	28	1.00	000	-0.1	007	8.66	28	0.0597	0.0000
110	6	5	24.708	6 0.000	0 0.00	00	-1.37	75	-0.0	961	0.46	44	45.9272	0.6224
111			6.917	2 -0.085	1 14.80	66	1.00	000	-0.0	752	9.06	92	0.7043	0.0000
112	6	4	284.861	0 0.000	0 0.00	00	-0.86	580	-0.5	5000	1.00	00	35.0000	1.4117
113		-	3.544	9 -0.250	0 15.00	0.0	1.00	0.00	-0.1	198	6.03	80	1.0000	0.0000
114	1.5		! Nr of	off-diag	onal ter	ms:	Ediss	: R	vdw:	alfa	a: cov	r: c	ov.r2:	cov.r3
115	1	2	0 121	9 1 /00	0 9 81	42	1 10	03	-1 0	0000	_1 00	2, 0		
116	1	2	0 1 2 0	3 1 707	6 10 20	70	1 36	08	1 0	1381	1 06	46		
117	1	1	0.109	6 1 224	0 10.29	98	1 10	100	1 6	016	_1 00	10		
11/	1	4	0.007	U 1.004	J41	20	1.43	,00	τ.0	040	T.00	00		

118	2	3	0.	.0283	1.2885	10.9190	0.9215	-1.0000	-1.0000		
119	2	4	0.	.1035	1.3327	11.5963	1.3977	-1.0000	-1.0000		
120	3	4	0.	.1836	1.9157	10.9070	1.7073	1.2375	-1.0000		
121	1	5	0.	.0610	1.5665	11.4404	1.2079	-1.0000	-1.0000		
122	2	5	0.	.2966	1.2550	10.2920	1.1989	-1.0000	-1.0000		
123	5	3	0.	.0790	2.2000	9.0491	1.8165	-1.0000	1.0000		
124	5	4	0.	1271	2.0090	11.5659	1.7156	1.0000	1.0000		
125	1	6	0	0749	1.7702	12.0430	1.6315	-1.0000	-1.0000		
126	6	2	0	0702	1 1145	12 1833	1 0722	-1 0000	-1 0000		
127	6	3	0	1293	1 3773	11 4683	1 3410	-1 0000	-1 0000		
127	6	5	0	0/95	1 2573	15 6983	0 9/15	-1 0000	-1 0000		
120	6	1	0.	0995	1 5863	13 3/01	1 5077	-1 0000	_1 0000		
129	52	7	I Na	0	los. The	+ 20.0 - k2.0 l	1.3077		1.0000		
121	1	1	1	67 2326	22060	1 628		$\int P^{\sqrt{2}}$	050 15 /1/1	1 8089	
122	1	1	2	65 2527	1/ 319	5 6 2 97	7 0.00		645 0 0000	1 1 1 5 3 0	
132	1 2	1	2	70 0040	14.510	0 2 4500				2 0000	
133	1	T	2	/0.0840	23.354	3.4508				3.0000	
134	1	2	1	0.0000						1.0400	
135	T	2	T	0.0000	3.411	.0 7.7350	- 0.00			1.0400	
136	2	2	2	0.0000	) 27.921	.3 5.8635	5 0.00	00 0.00		1.0400	
137	1	1	3	94.2903	3 59.357	5.0000	0.00	00 1.8	/4/ 1.0000	1.3538	
138	3	1	3	86.8971	23.911	7 5.0000	0.00	00 0.04	499 1.0000	2.8723	
139	1	3	1	49.8324	39.327	1 5.0000	0.00	00 3.01	156 0.0000	2.7534	
140	1	3	2	70.1101	13.121	.7 4.4734	4 0.00	00 0.84	433 0.0000	3.0000	
141	1	3	3	81.9029	32.225	1.739	7 0.00	00 0.98	888 68.1072	1.7777	
142	1	2	3	0.0000	25.000	3.0000	0.00	00 1.00	0.000	1.0400	
143	3	3	3	80.7324	30.455	4 0.9953	3 0.00	00 1.63	310 50.0000	1.0783	
144	2	3	3	75.6935	50.000	2.000	0.00	00 1.00	0.000	1.1680	
145	2	3	2	85.8000	9.845	3 2.2720	0.00	00 2.8	635 0.0000	1.5800	
146	3	2	3	0.0000	15.000	2.8900	0.00	00 0.00	0.000	2.8774	
147	2	2	3	0.0000	8.574	4 3.0000	0.00	00 0.00	0.000	1.0421	
148	4	4	4	66.2921	19.519	0.9624	4 0.00	00 0.10	0.000	1.2636	
149	2	4	4	68.5501	19.423	9 2.3592	2 0.00	00 0.20	0.0000	1.0000	
150	2	4	2	70.7499	11.485	4.660	5 0.00	00 1.5	647 0.0000	1.0902	
151	3	4	4	86.3294	18.387	9 5.8529	9 0.00	00 1.73	361 0.0000	1.2310	
152	2	4	3	73.6998	40.000	0 1.8782	2 0.00	00 4.00	0.000	1.1290	
153	3	4	3	79.5581	34.914	0 1.0801	L 0.00	00 0.10	632 0.0000	2.2206	
154	4	3	4	82.3364	4.735	1.3544	4 0.00	00 1.4	627 0.0000	1.0400	
155	2	3	4	90.0000	6.685	7 1.6689	9 0.00	00 2.5	771 0.0000	1.0400	
156	3	3	4	92.1207	24.393	0.500	0.00	00 1.72	208 0.0000	3.0000	
157	2	2	4	0.0000	0.010	0 1.0000	0.00	00 1.00	0.000	2.0000	
158	4	2	4	0.0000	4.421	.6 0.8596	5 0.00	00 0.90	624 0.0000	1.0000	
159	3	2	4	0.0000	5.000	1.000	0.00	00 1.00	0,000	1.2500	
160	3	5	3	60.0000	0.000	0 1.0000	0.00	00 1.00	0,000	1.0000	
161	5	3	3	81.6233	30.000	2.000	0.00	00 1.00	0.000	1.0000	
162	5	3	5	67 5247	6 451	2 4 0000				2 8079	
163	5	3	4	62 6634	8 4 4 4	1 2 5120				1 0000	
164	1	5	3	0 0000	0.010					1 0000	
165	1	3	5	0.0000				00 3.89	836 0.0000	1 0000	
166	5	1	3	99 6390					653 0.0000	1 8902	
167	1	1	1	70 8533	2 23 281	6 2 7470				) 2 2246	
10/	1	1	1	69 0000	, 20.040		5 0.00			) 2.1105	
106	1	1	1	50 0015	10 000	1 007				2.1190	
169	4	T	4	50.9317	10.933			00 0.2			
1/0	1	4	4	70 5040	12.096	2.070		00 1.00		1 0 4 0 0	
171	1	T	4	72.5945	11.034	7 2.4952		00 1.00		1.0400	
172	1	4	2	12.5945	14.834	1 2.4952	2 0.00	00 1.00		1.0400	
173	1	2	4	0.0000	2.500	1.0000		00 1.00		1.2500	
174	Ţ	Ţ	6	35.0676	44.062	0.4803	5 0.00	00 0.13		0.6875	
175	6	Ţ	6	90.2886	41.787	1 0.4774	± 0.00	00 0.2		0.6548	
176	1	6	1	82.4766	33.578	0.1350	o 0.00	00 0.20	0.0000	0.9612	
177	1	6	6	2.0340	25.186	1.0950	0.00	00 0.13	315 0.0000	0.9638	

178	5	6	5	75	.9308	39.5	963	1.5406	0.000	00	5.0016	0.0000	1.2226
179	6	5	6	85	.2677	38.4	940	1.6737	0.000	00	5.0138	0.0000	1.3107
180	6	3	6	86	.4972	15.1	409	1.1470	0.000	00 0	0.2045	0.0000	2.4012
181	6	3	3	73	.0506	30.3	751	2.0217	0.000	00	1.7756	0.0000	2.9301
182	6	4	6	79	.1640	38.5	410	1.3735	0.000	0 0	4.9517	0.0000	2.2216
183	34		! No	r o	f torsi	ons;	V1; V	72; V3; V	/2 (BO);	vcor	nj; n.u1;	n.u2	
184	1	1	1	1	-0.250	0 1	1.5822	0.189	9 -4.	.7057	-2.2047	0.0000	0.0000
185	1	1	1	2	-0.250	0 3	1.2596	0.170	9 -4.	.6391	-1.9002	2 0.0000	0.0000
186	2	1	1	2	-0.177	0 3	0.0252	0.434	0 -5.	.0019	-2.0697	0.0000	0.0000
187	1	1	1	3	-0.500	0	5.0000	-0.500	0 -9.	.0000	-1.0000	0.000	0.0000
188	2	1	1	3	-0.500	0	8.1082	0.171	0 -8.	.1074	-1.0000	0.000	0.0000
189	3	1	1	3	-1.447	7 1	6.6853	0.646	51 -4.	9622	-1.0000	0.000	0.0000
190	1	1	3	1	-0.230	0 4	6.8253	-0.284	8 -2.	.6326	-1.0000	0.000	0.0000
191	1	1	3	2	1.204	4 8	0.0000	-0.313	9 -6.	.1481	-1.0000	0.000	0.0000
192	2	1	3	1	-2.500	0 3	1.0191	0.616	5 -2.	.7733	-2.980	0.0000	0.0000
193	2	1	3	2	-2.487	5 7	0.8145	0.758	2 -4.	.2274	-3.0000	0.000	0.0000
194	1	1	3	3	-0.000	2 2	0.1851	0.160	1 -9.	.0000	-2.0000	0.000	0.0000
195	2	1	3	3	-1.438	3 8	0.0000	1.000	0 -3.	.6877	-2.8000	0.000	0.0000
196	3	1	3	1	-1.224	4 7	7.8133	-0.473	8 -4.	.7499	-3.0000	0.000	0.0000
197	3	1	3	2	-2.500	0 7	0.3345	-1.000	0 -5.	.5315	-3.0000	0.000	0.0000
198	3	1	3	3	-0.158	3 2	0.0000	1.500	0 -9.	.0000	-2.0000	0.000	0.0000
199	1	3	3	1	0.000	2 8	0.0000	-1.500	0 -4.	4848	-2.0000	0.000	0.0000
200	1	3	3	2	-2.128	9 1	2.8382	1.000	0 -5.	.6657	-2.9759	0.000	0.0000
201	1	3	3	3	2.500	0 -2	5.0000	1.000	0 -2.	.5000	-1.0000	0.0000	0.0000
202	2	3	3	3	0.830	2 -	4.0000	-0.776	53 -2.	.5000	-1.0000	0.000	0.0000
203	3	3	3	3	-2.500	0 -	4.0000	1.000	0 -2.	.5000	-1.0000	0.0000	0.0000
204	0	1	1	0	0.000	0	0.0000	0.000	0 0.	.0000	0.0000	0.0000	0.0000
205	0	1	2	0	0.000	0	0.0000	0.000	0 0.	.0000	0.0000	0.0000	0.0000
206	0	2	2	0	0.000	0	0.0000	0.000	0 0.	.0000	0.0000	0.0000	0.0000
207	0	2	3	0	0.000	0	0.1000	0.020	0 -2.	.5415	0.0000	0.0000	0.0000
208	0	3	3	0	0.551	1 2	5.4150	1.133	50 -5.	.1903	-1.0000	0.0000	0.0000
209	2	4	4	2	0.000	0	0.0000	0.064	0 -2.	.4426	0.0000	0.0000	0.0000
210	2	4	4	4	0.000	0	0.0000	0.158	7 -2.	.4426	0.0000	0.0000	0.0000
211	0	2	4	0	0.000	0	0.0000	0.120	0 -2.	.4847	0.0000	0.0000	0.0000
212	1	1	1	6	0.000	0 4	2.3172	3.454	6 -22.	.6501	-1.7255	5 0.0000	0.0000
213	6	1	1	6	0.000	0 7	5.5402	-0.749	97 -4.	.0257	-1.7255	0.0000	0.0000
214	0	1	6	0	4.000	0 4	5.8264	0.900	0 -4.	.0000	0.0000	0.0000	0.0000
215	0	6	6	0	4.000	0 4	5.8264	0.900	0 -4.	.0000	0.0000	0.0000	0.0000
216	0	5	6	0	0.033	6	0.6333	-0.062	1 -0.	.0007	-0.0578	3 0.0000	0.0000
217	0	4	6	0	4.733	4 4	4.7693	1.236	51 -4.	.7157	0.0000	0.0000	0.0000
218	2		! N1	r o	f hydro	gen 1	bonds;	Rhb; De	hb; vł	nb1; v	vhb2		
219	3	2	3	2	.1200	-3.5	800	1.4500	19.500	00			
220	6	2	3	2	.1200	-2.0	000	1.4500	19.500	00			

#### References

- 1. Wilkinson, M. D. *et al.* The fair guiding principles for scientific data management and stewardship. *Sci. data* 3, 1–9, DOI: https://doi.org/10.1038/sdata.2016.18 (2016).
- 2. Winter, M. The solid electrolyte interphase–the most important and the least understood solid electrolyte in rechargeable li batteries. *Zeitschrift für physikalische Chemie* 223, 1395–1406, DOI: https://doi.org/10.1524/zpch.2009.6086 (2009).
- 3. Van Duin, A. C., Dasgupta, S., Lorant, F. & Goddard, W. A. Reaxff: a reactive force field for hydrocarbons. *The J. Phys. Chem. A* 105, 9396–9409, DOI: https://doi.org/10.1021/jp004368u (2001).
- 4. Amsterdam Modeling Suite, SCM website.
- Komissarov, L., Ruger, R., Hellstrom, M. & Verstraelen, T. Params: parameter optimization for atomistic and molecular simulations. *J. Chem. Inf. Model.* 61, 3737–3743, DOI: https://doi.org/10.1021/acs.jcim.1c00333 (2021).

- Celik, F. A. & Karabulut, E. A new design of borax-cannabinol nanomaterial used to strengthen concrete structures: Non-scc-gfn1xtb model. *Solid State Commun.* 358, 114995, DOI: https://doi.org/10.1016/j.ssc.2022. 114995 (2022).
- 7. Yun, K.-S. *et al.* Simulation protocol for prediction of a solid-electrolyte interphase on the silicon-based anodes of a lithium-ion battery: Reaxff reactive force field. *The J. Phys. Chem. Lett.* 8, 2812–2818, DOI: https://doi.org/10.1021/acs.jpclett.7b00898 (2017).
- 8. Wang, Y., Liu, Y., Tu, Y. & Wang, Q. Reductive decomposition of solvents and additives toward solidelectrolyte interphase formation in lithium-ion battery. *The J. Phys. Chem. C* **124**, 9099–9108, DOI: https: //doi.org/10.1021/acs.jpcc.9b10535 (2020).
- 9. Jain, A. *et al.* Commentary: The materials project: A materials genome approach to accelerating materials innovation. *APL materials* 1, 011002, DOI: https://doi.org/10.1063/1.4812323 (2013).