

NAME

OpenFEUtil

SYNOPSIS

import OpenFEUtil

DESCRIPTION

OpenFEUtil module provides the following functions:

CalculatePartialCharges, ExecuteProtocolDAG, ExecuteProtocolDAGsAndGatherResults, GenerateLigandNetwork, GetMissingPartialChargesMolCount, GetMolFromName, GetMolNamePresentCount, GetPartialChargePropName, InitializeAbsoluteBindingFreeEnergyProtocol, InitializeAbsoluteSolvationFreeEnergyProtocol, InitializeAtomMapper, InitializeAtomMapperScorer, InitializeAtomMappers, InitializeChemicalSystem, InitializeProtocolDAG, InitializeRelativeFreeEnergyHybridTopologyProtocol, InitializeRelativeFreeEnergySeparatedTopologyProtocol, InitializeSolventComponent, InitializeTransformation, IsMolNamePresent, IsMolNamePresentMultipleTimes, ListOpenFESettings, ListOpenFESettingsByGroupName, ProcessMoleculeNames, ProcessMoleculePairs, ProcessOptionOpenFEAbsoluteBindingFreeEnergyParameters, ProcessOptionOpenFEAbsoluteFreeEnergyMode, ProcessOptionOpenFEAbsoluteHydrationFreeEnergyParameters, ProcessOptionOpenFECharge, ProcessOptionOpenFEChargeParameters, ProcessOptionOpenFEExecuteDAGParameters, ProcessOptionOpenFEMapper, ProcessOptionOpenFEMapperParameters, ProcessOptionOpenFEMissingChargeMode, ProcessOptionOpenFEMoleculePairs, ProcessOptionOpenFENetwork, ProcessOptionOpenFENetworkParameters, ProcessOptionOpenFERelativeFreeEnergyChargeCorrectionParameters, ProcessOptionOpenFERelativeFreeEnergyMode, ProcessOptionOpenFERelativeFreeEnergyParameters, ProcessOptionOpenFERelativeFreeEnergySeparatedTopologyParameters, ProcessOptionOpenFERelativeFreeEnergyVacuumParameters, ProcessOptionOpenFEResultFileParameters, ProcessOptionOpenFESolventParameters, ProcessRadialCentralLigandName, ReadAndValidateMolecules, ReadPDBFile, SetupAbsoluteBindingFreeEnergySettings, SetupAbsoluteHydrationFreeEnergySettings, SetupRelativeFreeEnergySeparatedTopologySettings, SetupRelativeFreeEnergySettings, SuggestAtomMappingsForMoleculePairs, UpdateRelativeFreeEnergySettingsForChargeCorrection, UpdateRelativeFreeEnergySettingsForVacuum, WriteLigandNetworkGraphMLFile, WriteLigandNetworkImageFile, WriteMappingImageFile, WriteProtocolDAGResultFile

FUNCTIONS**CalculatePartialCharges**

CalculatePartialCharges(Mols, ChargeMethod, ChargeParamsInfo)

Calculate partial atomic charges for molecules and return a set of OpenFE charges molecule objects. The calculated charges are stored as value of the molecule property named 'atom.dprop.PartialCharge'.

The following methods are supported to calculate partial atomic charges: AM1BCC, M1-Mulliken, Espaloma, Gasteiger, MMFF94, or NAGL.

The ChargeParamsInfo parameter is a dictionary of name and value pairs for charge parameters and may be generated by calling the function named ProcessOptionOpenFEChargeParameters().

Arguments:

Mols (list): List of OpenFE molecule objects.
 ChargeMethod (str): Charge method.
 ChargeParamsInfo (dict): Parameter name and value pairs.

Returns:

list: List of OpenFE charged molecule objects.
 bool: True or False.

ExecuteProtocolDAG

ExecuteProtocolDAG(ProtocolDAG, SharedOutDirPath, ScratchOutDirPath, KeepShared=True, KeepScratch=False, NRetries=0)

Execute protocol DAG to perform simulations for calculating FE.

Arguments:

ProtocolDAG (object): OpenFE protocol DAG object.
 SharedOutDirPath (object): Pathlib path object.
 ScratchOutDirPath (object): Pathlib path object.
 KeepShared (bool): Keep shared directory.

KeepScratch (bool): Keep scratch directory.
NRetries (int): Number of times to attempt the execution. A value
0 implies only 1 try.

Returns:

object: OpenFE DAG result object.

ExecuteProtocolDAGsAndGatherResults

ExecuteProtocolDAGsAndGatherResults(MolTransformations, MolProtocolDAGs,
SharedOutDirPath, ScratchOutDirPath, KeepShared=True, KeepScratch=False, NRetries=0,
WriteResults=True)

Execute protocol DAG and gather results.

Arguments:

MolTransformations (List): List of OpenFE transformation objects.
MolProtocolDAGs (List): List of OpenFE DAG objects.
SharedOutDirPath (str): Shared results directory path.
ScratchOutDirPath (str): Scratch results directory path.
KeepShared (bool): Keep shared directory.
KeepScratch (bool): Keep scratch directory.
NRetries (int): Number of times to attempt the execution. A value
0 implies only 1 try.
WriteResults (bool): Write results to a JSON file.

Returns:

list: List of OpenFE DAG result objects.

GenerateLigandNetwork

GenerateLigandNetwork(Mols, NetworkName, NetworkParamsInfo, Mappers, MapperScorer)

Generate a ligand network for molecules using the specified atom mappers and scorer. You may specify multiple atom mappers for generating mapping between two molecules. All specified mappers are employed to identify the highest scoring edges for generating a ligand network.

Possible values for network name are: LOMAP, MinimalSpanning, or Radial.

The NetworkParamsInfo parameter is a dictionary of name and value pairs for network parameters and may be generated by calling the function named ProcessOptionOpenFENetworkParameters().

Arguments:

Mols (list): List of OpenFE molecule objects.
NetworkName (str): Network name.
NetworkParamsInfo (dict): Parameter name and value pairs.
Mapper (list): List of OpenFE atom mapper objects.
MapperScorer (Callable): OpenFE atom mapper scorer.

Returns:

object: OpenFE ligand network object.

GetMissingPartialChargesMolCount

GetMissingPartialChargesMolCount(Mols)

Get count of molecules with missing partial atomic charges.

The absence of molecule property name, atom.dprop.PartialCharge, implies missing charges for the molecule.

Arguments:

Mols (list): List OpenFE molecule objects.

Returns:

int: Molecule count with missing partial charges.

GetMolFromName

GetMolFromName(Mols, MolName)

Get the first molecule whose name matches the specified molecule name from a list of molecules.

Arguments:

Mols (list): List of OpenFE molecule objects.
MolName (str): Molecule name

Returns:

object or None: Open FE molecule object.

GetMolNamePresentCount

GetMolNamePresentCount(Mols, MolName)

Get count of molecule name present in a list of molecules.

Arguments:

Mols (list): List OpenFE molecule objects.
MolName (str): Molecule name

Returns:

int: Molecule name present count.

GetPartialChargePropName

GetPartialChargePropName()

Get partial atomic charge property name used for associating partial charges to a molecule.

Arguments:

None

Returns:

str: Property name 'atom.dprop.PartialCharge'

InitializeAbsoluteBindingFreeEnergyProtocol

InitializeAbsoluteBindingFreeEnergyProtocol(ABFESettings)

Initialize absolute binding free energy protocol.

Arguments:

ABFESettings (object): OpenFE AbsoluteBindingProtocol settings object.

Returns:

object: OpenFE AbsoluteBindingProtocol object.

InitializeAbsoluteSolvationFreeEnergyProtocol

InitializeAbsoluteSolvationFreeEnergyProtocol(AHFESettings)

Initialize absolute solvation free energy protocol.

Arguments:

AHFESettings (object): OpenFE AbsoluteSolvationProtocol settings object.

Returns:

object: OpenFE AbsoluteSolvationProtocol object.

InitializeAtomMapper

InitializeAtomMapper(MapperName, MapperParamsInfo)

Initialize an atom mapper.

Possible values for atom mapper names are: LOMAP or Kartograf.

The MapperParamsInfo parameter is a dictionary of name and value pairs for network parameters and may be generated by calling the function named ProcessOptionOpenFEMapperParameters().

Arguments:

MapperName (str): Atom mapper name.
MapperParamsInfo (dict): Parameter name and value pairs.

Returns:

object: OpenFE Atom mapper object.

InitializeAtomMapperScorer

InitializeAtomMapperScorer(ScorerName)

Initialize an atom mapper scorer.
Possible value for scorer name is LOMAP.

Arguments:

ScorerName (str): Atom mapper scorer name.

Returns:

object: Atom mapper scorer object.

InitializeAtomMappers

InitializeAtomMappers(MapperNameList, MapperParamsInfo)

Initialize atom mappers.
Possible values for atom mapper names: LOMAP or Kartograf.
The MapperParamsInfo parameter is a dictionary of name and value pairs for network parameters and may be generated by calling the function named ProcessOptionOpenFEMapperParameters().

Arguments:

MapperNameList (list): List of atom mapper names.
MapperParamsInfo (dict): Parameter name and value pairs.

Returns:

list: List of OpenFE atom mapper objects.

InitializeChemicalSystem

InitializeChemicalSystem(SmallMol=None, MacroMol=None, Solvent=None, Name="")

Initialize a chemical system.
A valid value must be specified for at least one part of the system.

Arguments:

SmallMol (object): OpenFE SMC object.
MacroMol (object): OpenFE PDB object.
Solvent (object): OpenFE solvent component object.
Name (str): Chemical system name.

Returns:

object: OpenFE ChemicalSystem object.

InitializeProtocolDAG

InitializeProtocolDAG(Transformation, Name="")

Create a protocol DAG (Directed Acyclic Graph) for a transformation to perform calculation.

Arguments:

Transformation (object): OpenFE Transformation object.
Name (str): DAG name.

Returns:

object: OpenFE DAG object.

InitializeRelativeFreeEnergyHybridTopologyProtocol

```
InitializeRelativeFreeEnergyHybridTopologyProtocol(RBFESettings)
```

Initialize relative free energy hybrid topology protocol.

Arguments:

RBFESettings (object): OpenFE RelativeHybridTopologyProtocol settings object.

Returns:

object: OpenFE RelativeHybridTopologyProtocol object.

```
InitializeRelativeFreeEnergySeparatedTopologyProtocol
```

```
InitializeRelativeFreeEnergySeparatedTopologyProtocol(RBFESettings)
```

Initialize relative free energy separated topologies protocol.

Arguments:

RBFESettings (object): OpenFE SeparatedTopologyProtocol settings object.

Returns:

object: OpenFE SeparatedTopologyProtocol object.

```
InitializeSolventComponent
```

```
InitializeSolventComponent(SolventParamsInfo)
```

Initialize a solvent component.

The SolventParamsInfo parameter is a dictionary of name and value pairs for network parameters and may be generated by calling the function named ProcessOptionOpenFESolventParameters().

Arguments:

SolventParamsInfo (dict): Parameter name and value pairs.

Returns:

object: OpenFE SolventComponent object.

```
InitializeTransformation
```

```
InitializeTransformation(StateA, StateB, Mapping, Protocol, Name="",  
Validate=False)
```

Initialize a transformation between two chemical systems represented by StateA and StateB.

Arguments:

StateA (object): OpenFE ChemicalSystem object.
StateB (object): OpenFE ChemicalSystem object.
Mapping (object): OpenFE mapping object.
Protocol (object): OpenFE protocol object.
Name (str): Transformation name.
Validate (bool): Validate inputs for transformation..

Returns:

object: OpenFE Transformation object.

```
IsMolNamePresent
```

```
IsMolNamePresent(Mols, MolName)
```

Check for the presence of a molecule name in a list of molecules.

Arguments:

Mols (list): List OpenFE molecule objects.
MolName (str): Molecule name

Returns:

bool: True or False.

IsMolNamePresentMultipleTimes

IsMolNamePresentMultipleTimes(Mols, MolName)

Check for the presence of a molecule name in a list of molecules.

Arguments:

Mols (list): List OpenFE molecule objects.
MolName (str): Molecule name

Returns:

bool: True or False.

ListOpenFESettings

ListOpenFESettings(Settings)

List setting retrieved from a protocol settings object.

Arguments:

Settings (object): OpenFE protocol settings object.

Returns:

None

ListOpenFESettingsByGroupName

ListOpenFESettingsByGroupName(Settings, SettingName)

List setting retrieved from a protocol settings object for a specified settings group name.

Arguments:

Settings (object): OpenFE protocol settings object.

Returns:

None

ProcessMoleculeNames

ProcessMoleculeNames(Mols, MoleculeNamesList=None)

Process molecule names to generate a list of molecule objects for specified names. The molecule name must be a valid name and occur only once in the list of molecules.

The first molecule in the list of molecules is returned for an unspecified list of molecule names.

Arguments:

Mols (list): List of OpenFE molecule objects.
MoleculeNamesList (list): List of molecule names.

Returns:

list: List of OpenFE molecule objects corresponding to molecule names.

ProcessMoleculePairs

ProcessMoleculePairs(Mols, MoleculePairsList=None)

Process molecule names, corresponding to pairs of molecules, to generate a list of molecule objects for these names. The molecule name must be a valid name and occur only once in the list of molecules.

The first and the second molecule in the list of molecules is returned for an unspecified list of molecule names.

Arguments:

Mols (list): List of OpenFE molecule objects.
MoleculePairsList (list): List of molecule names corresponding to pairs of molecules.

Returns:

list: List of OpenFE molecule objects corresponding to pairs of molecule

names.

ProcessOptionOpenFEAbsoluteBindingFreeEnergyParameters

ProcessOptionOpenFEAbsoluteBindingFreeEnergyParameters(ParamsOptionName,
ParamsOptionValue, ParamsDefaultInfo=None)

Process parameters for ABFE parameters option and return a map containing processed parameter names and values.

The ParamsOptionValue is a comma delimited list of parameter name and value pairs to setup ABFE calculations.

The default values are automatically updated to match settings provided by OpenFE module AbsoluteBindingProtocol.

You must specify valid OpenFE values for these parameters. An extensive validation is not performed.

The supported parameter names along with their default and possible values are shown below:

protocolRepeats, 3

Complex equil output settings:

complexEquilOutputCheckpointInterval, 1 [Units: nanosecond]
complexEquilOutputCheckpointStorageFilename, checkpoint.chk
complexEquilOutputEquilNPTStructure, equil_npt_structure.pdb
complexEquilOutputEquilNVTStructure, equil_nvt_structure.pdb
complexEquilOutputForcefieldCache, db.json
complexEquilOutputLogOutput, production_equil_simulation.log
complexEquilOutputMinimizedStructure, minimized.pdb
complexEquilOutputIndices, all [Possible value: Any valid
selection.]
complexEquilOutputPremnimizedStructure, system.pdb
complexEquilOutputProductionTrajectoryFilename, production_equil.xtc
complexEquilOutputTrajectoryWriteInterval, 20.0 [Units:
picosecond]

Complex equil simulation settings:

complexEquilSimulationEquilibrationLength, 0.5 [Units: nanosecond]
complexEquilSimulationEquilibrationLengthNVT, 0.25 [Units:
nanosecond]
complexEquilSimulationMinimizationSteps, 5000
complexEquilSimulationProductionLength, 5.0 [Units: nanosecond]

Complex lambda settings:

complexLambdaElec, 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.1 0.2 0.3 0.4 0.5 0.6
0.7 0.8 0.9 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
1.0 1.0 [Possible values: A space delimited list of values
between 0.0 and 1.0]
complexLambdaRestraints, 0.0 0.2 0.4 0.6 0.8 1.0 1.0 1.0 1.0 1.0
1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
1.0 1.0 1.0 [Possible values: A space delimited list of
values between 0.0 and 1.0]
complexLambdaVdw, 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.65 0.7 0.75 0.8 0.85
0.9 0.95 1.0 [Possible values: A space delimited list of values
between 0.0 and 1.0]

Complex output settings:

complexOutputCheckpointInterval, 1.0 [Units: nanosecond]
complexOutputCheckpointStorageFilename, complex_checkpoint.nc
complexOutputForcefieldCache, db.json
complexOutputFilename, complex.nc
complexOutputIndices, not water [Possible value: Any valid
selection.]
complexOutputStructure, alchemical_system.pdb
complexOutputPositionsWriteFrequency, 100 [Units: picosecond]
complexOutputVelocitiesWriteFrequency, None [Possible

values: > 0; Units: picosecond]

Complex simulation settings:

complexSimulationEarlyTerminationTargetError, 0.0 [Units:
kilocalorie_per_mole]
complexSimulationEquilibrationLength, 1.0 [Units: nanosecond]
complexSimulationMinimizationSteps, 5000
complexSimulationNReplicas, 30
complexSimulationProductionLength, 10.0 [Units: nanosecond]
complexSimulationRealTimeAnalysisInterval, 250.0 [Units:
picosecond]
complexSimulationRealTimeAnalysisMinimumTime, 500.0 [Units:
picosecond]
complexSimulationSamplerMethod, repex [Possible values: repex,
sams, or independent]
complexSimulationSamsFlatnessCriteria, logZ-flatness [Possible
values: logZ-flatness, minimum-visits or histogram-flatness]
complexSimulationSamsGamma0, 1.0
complexSimulationTimePerIteration, 2.5 [Units: picosecond]

Complex solvation settings:

complexSolvationBoxShape, dodecahedron [Possible values: cube,
dodecahedron, or octahedron]
complexSolvationBoxSize, None [Possible value: A triplet of space
X Y Z values; Units: nanometer]
complexSolvationSolventModel, tip3p [Possible values: tip3p, spce,
tip4pew, or tip5p]
complexSolvationSolventPadding, 1.0 [Units: nanometer]

Engine settings:

engineComputePlatform, CPU [Possible values: CPU, CUDA,
OpenCL, or Reference]
engineGpuDeviceIndex, None [Possible values: 0, 0 1, etc.]

Forcefield settings:

forcefieldConstraints, HBonds [Possible values: HBonds,
AllBonds, or HAngles]
forcefields, amber/ff14SB.xml amber/tip3p_standard.xml
amber/tip3p_HFE_multivalent.xml amber/phosaa10.xml
[Possible values: A space delimited list of valid names.]
forcefieldHydrogenMass, 3.0 [Units: amu]
forcefieldNonbondedCutoff, 0.9 [Units: nanometer]
forcefieldNonbondedMethod, PME [Possible values: PME or
NoCutoff]
forcefieldRigidWater, yes, [Possible values: yes or no]
forcefieldSmallMoleculeForcefield, openff-2.1.1 [Possible
value: A valid forcefield name.]

Integrator settings:

integratorBarostatFrequency, 25.0 * timestep [The specified value
is a multiple of integratorTimestep.]
integratorConstraintTolerance, 1e-06
integratorLangevinCollisionRate, 1.0 [Units: 1 / picosecond]
integratorNRestartAttempts, 20
integratorReassignVelocities, no [Possible values: yes or no]
integratorRemoveCom, no [Possible values: yes or no]
integratorTimestep, 4.0 [Units: femtosecond]

Partial charge settings:

partialChargeNaglModel, None [Default: Production AM1BCC model for
NAGL; Possible value: Any valid name.]
partialChargeNumberOfConformers, None [Possible value: > 0]
partialChargeOffToolkitBackend, AmberTools [Possible values:

AmberTools or RDKit]
 partialChargeMethod, AM1BCC [Possible values: AM1BCC, Espaloma,
 or NAGL]

Restraint settings:

restraintKPhiA, 334.72 [Units: kilojoule_per_mole / radian**2]
 The default value is equivalent to 80 kcal/mol/radian**2]
 restraintKPhiB, 334.72 [Units: kilojoule_per_mole / radian**2]
 The default value is equivalent to 80 kcal/mol/radian**2]
 restraintKPhiC, 334.72 [Units: kilojoule_per_mole / radian**2]
 The default value is equivalent to 80 kcal/mol/radian**2]
 restraintKR, 4184.0 [Units: kilojoule_per_mole / nanometer**2]
 The default value is equivalent to 10 kcal/mol/angstrom**2]
 restraintKThetaA, 334.72 [Units: kilojoule_per_mole / radian**2]
 The default value is equivalent to 80 kcal/mol/radian**2]
 restraintKThetaB, 334.72 [Units: kilojoule_per_mole / radian**2]
 The default value is equivalent to 80 kcal/mol/radian**2]
 restraintAnchorFindingStrategy, bonded [Possible values:
 multi-residue or bonded]
 restraintDsspFilter, yes [Possible values: yes or no]
 restraintHostMaxDistance, 1.5 [Units: nanometer]
 restraintHostMinDistance, 0.5 [Units: nanometer]
 restraintHostSelection, backbone [Possible value: Any valid
 selection.]
 restraintRmsfCutoff, 0.1 [Units: nanometer]

Solvent equil output settings:

solventEquilOutputCheckpointInterval, 1.0 [Units: nanosecond]
 solventEquilOutputCheckpointStorageFilename, checkpoint.chk
 solventEquilOutputNPTStructure, equil_npt_structure.pdb
 solventEquilOutputNVTStructure, equil_nvt_structure.pdb
 solventEquilOutputForcefieldCache, db.json
 solventEquilOutputLogOutput, production_equil_simulation.log
 solventEquilOutputMinimizedStructure, minimized.pdb
 solventEquilOutputIndices, all [Possible value: Any valid
 selection.]
 solventEquilOutputPreminimizedStructure, system.pdb
 solventEquilOutputProductionTrajectoryFilename, production_equil.xtc
 solventEquilOutputTrajectoryWriteInterval, 20.0 [Units:
 picosecond]

Solvent equil simulation settings:

solventEquilSimulationEquilibrationLength, 0.2 [Units: nanosecond]
 solventEquilSimulationEquilibrationLengthNVT, 0.1 [Units:
 nanosecond]
 solventEquilSimulationMinimizationSteps, 5000
 solventEquilSimulationProductionLength, 0.5 [Units: nanosecond]

Solvent lambda settings:

solventLambdaElec, 0.0 0.25 0.5 0.75 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
 1.0 1.0 [Possible values: A space delimited list of values
 between 0.0 and 1.0]
 solventLambdaRestraints, 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
 0.0 0.0 0.0 [Possible values: A space delimited list of values
 between 0.0 and 1.0]
 solventLambdaVdw, 0.0 0.0 0.0 0.0 0.0 0.12 0.24 0.36 0.48 0.6 0.7
 0.77 0.85 1.0 [Possible values: A space delimited list of values
 between 0.0 and 1.0]

Solvent output settings:

solventOutputCheckpointInterval, 1.0 [Units: nanosecond]
 solventOutputCheckpointStorageFilename, solvent_checkpoint.nc
 solventOutputForcefieldCache, db.json
 solventOutputFilename, solvent.nc

```

solventOutputIndices, not water [ Possible value: Any valid
    selection. ]
solventOutputStructure, alchemical_system.pdb
solventOutputPositionsWriteFrequency, 100.0 [ Units: picosecond ]
solventOutputVelocitiesWriteFrequency, None [ Possible
    values: > 0; Units: picosecond ]

Solvent simulation settings:

solventSimulationEarlyTerminationTargetError, 0.0 [ Units:
    kilocalorie_per_mole ]
solventSimulationEquilibrationLength, 1.0 [ Units: nanosecond ]
solventSimulationMinimizationSteps, 5000
solventSimulationNReplicas, 14
solventSimulationProductionLength, 10.0 [ Units: nanosecond ]
solventSimulationRealTimeAnalysisInterval, 250.0 [ Unit: picosecond ]
solventSimulationRealTimeAnalysisMinimumTime, 500.0 [ Units:
    picosecond ]
solventSimulationSamplerMethod, repex [ Possible values: repex,
    sams, or independent ]
solventSimulationSamsFlatnessCriteria, logZ-flatness [ Possible
    values: logZ-flatness, minimum-visits or histogram-flatness ]
solventSimulationSamsGamma0, 1.0
solventSimulationTimePerIteration, 2.5 [ Units: picosecond ]

Solvent solvation settings:

solventSolvationBoxShape, dodecahedron [ Possible values: cube,
    dodecahedron, or octahedron ]
solventSolvationBoxSize, None [ Possible value: A triplet of space
    X Y Z values; Units: nanometer ]
solventSolvationSolventModel, tip3p [ Possible values: tip3p, spce,
    tip4pew, or tip5p ]
solventSolvationSolventPadding, 1.5 [ Units: nanometer ]

Thermo settings:

thermoPh, None [ Possible values: > 0 ]
thermoPressure, 1.0 [ Units: bar ]
thermoRedoxPotential, None [ Possible values: A valid float.
    Units: millivolts (mV) ]
thermoTemperature, 298.15 [ Units: kelvin ]

```

A brief description of parameters, taken from OpenFE documentation, is provided below:

protocolRepeats: Number of completely independent repeats of the entire sampling process.

Complex settings:

Complex parameters for the system, including the solvent model and the solvent padding.

Complex equil output settings:

Parameters controlling simulation output during equilibration phase of complex transformation.

```

complexEquilOutputCheckpointInterval: Frequency to write the
    checkpoint file.
complexEquilOutputCheckpointStorageFilename: Checkpoint filename.
complexEquilOutputEquilNPTStructure: NPT structure filename.
complexEquilOutputEquilNVTStructure: NVT structure filename.
complexEquilOutputForcefieldCache: Filename for caching small
    molecule residue templates.
complexEquilOutputLogOutput: Simulation log filename.
complexEquilOutputMinimizedStructure: Minimized structure filename.
complexEquilOutputIndices: Selection string for selecting
    coordinates to write.

```

complexEquilOutputPremnimizedStructure: Initial structure filename.
complexEquilOutputProductionTrajectoryFilename: Trajectory filename.
complexEquilOutputTrajectoryWriteInterval: Frequency for writing velocities to trajectory file.

Complex equil simulation settings:

Parameters controlling simulation during equilibration phase of complex transformation.

complexEquilSimulationEquilibrationLength: Length of the NPT equilibration phase.
complexEquilSimulationEquilibrationLengthNVT: Length of the NVT equilibration phase.
complexEquilSimulationMinimizationSteps: Maximum number of minimization steps to perform.
complexEquilSimulationProductionLength: Length of the NPT production phase.

Complex lambda settings:

Lambda protocol parameters for complex transformation.

complexLambdaElec: List of lambda values for electrostatics. The values of 0 and 1 imply state A and state B respectively.
complexLambdaRestraints: List of lambda values for restraints. The values of 0 and 1 imply state A and state B respectively.
complexLambdaVdw: List of lamda values for van der Waals. The values of of 0 and 1 imply state A and state B respectively.

Complex output settings:

Parameters controlling simulation output during final phase of complex transformation.

complexOutputCheckpointInterval: Frequency to write the checkpoint file.
complexOutputCheckpointStorageFilename: Checkpoint filename.
complexOutputForcefieldCache: Filename for caching small molecule residue templates.
complexOutputFilename: Trajectory filename.
complexOutputIndices: Selection string for selecting coordinates to write.
complexOutputStructure: Topology structure filename.
complexOutputPositionsWriteFrequency: Frequency for writing positions to trajectory file.
complexOutputVelocitiesWriteFrequency: Frequency for writing velocities to trajectory file.

Complex simulation settings:

Parameters controlling simulation during final phase of complex transformation.

complexSimulationEarlyTerminationTargetError: Target error for the real time analysis measured in kcal/mol. Once the MBAR error of the free energy is at or below this value, the simulation will be considered complete. The suggested value of 0.12 has shown to be effective in both hydration and binding free energy benchmarks.
complexSimulationEquilibrationLength: Length of the equilibration phase. The specified value must be divisible by 'integratorTimestep'.
complexSimulationMinimizationSteps: Maximum number of minimization steps to perform.
complexSimulationNReplicas: Number of replicas to use.
complexSimulationProductionLength: Length of the production phase. The specified value must be divisible by 'integratorTimestep'.
complexSimulationRealTimeAnalysisMinimumTime: Time interval for

performing analysis of the free energies. At each interval, real time analysis data will be written to a yaml file named <outputFileName>_real_time_analysis.yaml. The current error in the estimate will also be assessed and the simulation will be terminated when it drops below 'complexSimulationEarlyTerminationTargetError'.

complexSimulationSamplerMethod: Alchemical sampling method to use: REPEX (Hamiltonian REPLICA EXchange), SAMS (Self-Adjusted Mixture Sampling), or Independent (Independently sampled lambda windows).

complexSimulationSamsFlatnessCriteria: Method for assessing when to switch to asymptotically optimal scheme for SAMS.

complexSimulationsamsGamma0: Initial weight adaptation rate for SAMS.

complexSimulationTimePerIteration: Simulation time between each MCMC move attempt

Complex solvation settings:

Solvation parameters for the system, including the solvent model and the solvent padding.

complexSolvationBoxShape: Shape of the periodic solvent box.

complexSolvationBoxSize: Lengths of the unit cell for a solvent box.

complexSolvationSolventModel: Forcefield water model to use during solvation and defining the model properties.

complexSolvationSolventPadding: Minimum distance from any solute bounding sphere to the edge of the box.

Engine settings:

Parameters configuring the compute platform used by the OpenMM to perform the simulation.

engineComputePlatform: Platform to use for running OpenMM MD calculations.

engineGpuDeviceIndex: Space delimited list of device indices to use for running OpenMM MD calculations.

Forcefield settings:

forcefieldConstraints: Constraints to use.

forcefields: List of valid forcefield paths for all components except small molecules.

forcefieldHydrogenMass: Mass to be repartitioned to hydrogens from neighboring heavy atoms.

forcefieldNonbondedCutoff: Cutoff for short range nonbonded interactions.

forcefieldNonbondedMethod: Method for treating nonbonded interactions.

forcefieldRigidWater: Use a rigid water model.

forcefieldSmallMoleculeForcefield: A valid forcefield name to use for small molecules.

Integrator settings:

Parameters controlling the LangevinSplittingDynamicsMove integrator used for simulation.

integratorBarostatFrequency: Frequency at which volume scaling changes should be attempted.

integratorConstraintTolerance: Tolerance for constraint solver.

integratorLangevinCollisionRate: Collision frequency.

integratorNRestartAttempts: Number of attempts to restart from Context in case there are NaNs in the energies after integration.

integratorReassignVelocities: Reassign velocities from the Maxwell-Boltzmann distribution at the beginning of each Monte Carlo move.

integratorRemoveCom: Remove the center of mass motion.
integratorTimestep: Size of the simulation timestep.

Partial charge settings:

Parameters for automatically assigning missing partial charges to small molecules, including the partial charge method.

partialChargeNaglModel: Model to use for partial charge assignment. A value of None implies the use of the latest available production AM1BCC model.

partialChargeNumberOfConformers: Number of conformers to generate as part of the partial charge assignment. A value of None implies the use of the existing conformer.

partialChargeOffToolkitBackend: OpenFF toolkit registry backend to use for calculating partial charges.

partialChargeMethod: Method to use for calculating partial charges.

Restraint settings:

Parameters to configure Boresch-style restraint between two groups of atoms named host (Hx) and guest (Gx).

restraintKPhiA: Equilibrium force constant for the dihedral formed by H2-H1-H0-G0.

restraintKPhiB: Equilibrium force constant for the dihedral formed by H1-H0-G0-G1.

restraintKPhiC: Equilibrium force constant for the dihedral formed by H0-G0-G1-G2.

restraintKR: Bond spring constant between H0 and G0.

restraintKThetaA: Spring constant for the angle formed by H1-H0-G0.

restraintKThetaB: Spring constant for the angle formed by H0-G0-G1

restraintAnchorFindingStrategy: Boresch atom picking strategy to use. bonded: pick host atoms that are bonded to each other. multi-residue: pick host atoms which can span multiple residues.

restraintDsspFilter: Apply DSSP filter to the host atoms.

restraintHostMaxDistance: Minimum distance between any host atom and the guest G0 atom.

restraintHostMinDistance: Xaximum distance between any host atom and the guest G0 atom

restraintHostSelection: A valid selection string to sub-select the host atoms which will be involved in the restraint.

restraintRmsfCutoff: Cutoff value for filtering atoms by their root mean square fluctuation. Atoms with values above this cutoff are ignored.

Solvent equil output settings:

Solvent equil simulation settings:

Solvent lambda settings:

Solvent output settings:

Solvent simulation settings:

Solvent solvation settings:

The solvent settings are similar to the complex settings already described under various sections for complex. The prefix 'solvent' is used for the names of the parameters instead of the prefix 'complex.'

Thermo settings:

Thermodynamic parameters, including the temperature and the pressure of the system.

thermoPh: Simulation pH

thermoPressure: Simulation pressure.

thermoRedoxPotential: Simulation redox potential.

thermoTemperature: Simulation temperature.

Arguments:

ParamsOptionName (str): Command line OpenFE RFE parameters option name.
 ParamsOptionValue (str): Comma delimited list of parameter name and value pairs.
 ParamsDefaultInfo (dict): Default values to override selected parameters.

Returns:

dictionary: Processed parameter name and value pairs.

ProcessOptionOpenFEAbsoluteFreeEnergyMode

ProcessOptionOpenFEAbsoluteFreeEnergyMode(OptionName, OptionValue)

Process absolute FE mode command line option and return a valid canonical value.

Valid values names are: FirstMolecule, AllMolecules, or MoleculeNames.

Arguments:

OptionName (str): Command line missing charge mode option name.
 OptionValue (str): Command line missing charge mode option value.

Returns:

str: Canonical value for missing charge mode.

ProcessOptionOpenFEAbsoluteHydrationFreeEnergyParameters

ProcessOptionOpenFEAbsoluteHydrationFreeEnergyParameters(ParamsOptionName,
 ParamsOptionValue, ParamsDefaultInfo=None)

Process parameters for AHFE parameters option and return a map containing processed parameter names and values.

The ParamsOptionValue is a comma delimited list of parameter name and value pairs to setup AHFE calculations.

The default values are automatically updated to match settings provided by OpenFE module AbsoluteSolvationProtocol.

You must specify valid OpenFE values for these parameters. An extensive validation is not performed.

The supported parameter names along with their default and possible values are shown below:

protocolRepeats, 3

Integrator settings:

integratorBarostatFrequency, 25.0 * timestep [The specified value
 is a multiple of integratorTimestep.]
 integratorConstraintTolerance, 1e-06
 integratorLangevinCollisionRate, 1.0 [Units: 1 / picosecond]
 integratorNRestartAttempts, 20
 integratorReassignVelocities, no [Possible values: yes or no]
 integratorRemoveCom, no [Possible values: yes or no]
 integratorTimestep, 4.0 [Units: femtosecond]

Lambda settings:

lambdaElec, 0.0 0.25 0.5 0.75 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
 1.0 [Possible values: A space delimited list of values
 between 0.0 and 1.0]
 lambdaRestraints, 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
 0.0 0.0 [Possible values: A space delimited list of values
 between 0.0 and 1.0]
 lambdaVdw, [0.0 0.0 0.0 0.0 0.0 0.12 0.24 0.36 0.48 0.6 0.7 0.77
 0.85 1.0 [Possible values: A space delimited list of values
 between 0.0 and 1.0]

Partial charge settings:

partialChargeNaglModel, None [Default: Production AM1BCC model for
 NAGL; Possible value: Any valid name.]
 partialChargeNumberOfConformers, None [Possible value: > 0]
 partialChargeOffToolkitBackend, AmberTools [Possible values:

```

    AmberTools or RDKit ]
partialChargeMethod, AM1BCC [ Possible values: AM1BCC, Espaloma,
    or NAGL ]

Solvation settings:

solvationBoxShape, dodecahedron [ Possible values: cube,,
    dodecahedron, or octahedron ]
solvationBoxSize, None [ Possible value: A triplet of space
    X Y Z values; Units: nanometer ]
solvationSolventModel, tip3p [ Possible values: tip3p, spce, tip4pew,
    or tip5p ]
solvationSolventPadding, 1.5 [ Units: nanometer ]

Solvent engine settings:

solventEngineComputePlatform, CPU [ Possible values: CPU, CUDA,
    OpenCL, or Reference ]
solventEngineGpuDeviceIndex, None [ Possible values: 0, 0 1, etc. ]

Solvent equil output settings:

solventEquilOutputCheckpointInterval, 1.0 [ Units: nanosecond ]
solventEquilOutputCheckpointStorageFilename, checkpoint.chk
solventEquilOutputNPTStructure, equil_npt_structure.pdb
solventEquilOutputNVTStructure, equil_nvt_structure.pdb
solventEquilOutputForcefieldCache, db.json
solventEquilOutputLogOutput, equil_simulation.log
solventEquilOutputMinimizedStructure, minimized.pdb
solventEquilOutputIndices, not water [ Possible value: Any valid
    selection. ]
solventEquilOutputPreminimizedStructure, system.pdb
solventEquilOutputProductionTrajectoryFilename, production_equil.xtc
solventEquilOutputTrajectoryWriteInterval, 20.0 [ Units: picosecond ]

Solvent equil simulation settings:

solventEquilSimulationEquilLength, 0.2 [ Units: nanosecond ]
solventEquilSimulationEquilLengthNVT, 0.1 [ Units: nanosecond ]
solventEquilSimulationMinimizationSteps, 5000
solventEquilSimulationProductionLength, 0.5 [ Units: nanosecond ]

Solvent forcefield settings:

solventForcefieldConstraints, HBonds [ Possible values: HBonds,
    AllBonds, or HAngles ]
solventForcefields, amber/ff14SB.xml, amber/tip3p_standard.xml
    amber/tip3p_HFE_multivalent.xml amber/phosaa10.xml
    [ Possible values: A space delimited list of valid names. ]
solventForcefieldHydrogenMass, 3.0 [ Units: amu ]
solventForcefieldNonbondedCutoff, 0.9 [ Units: nanometer ]
solventForcefieldNonbondedMethod, PME [ Possible values: PME or
    NoCutoff ]
solventForcefieldRigidWater, yes, [ Possible values: yes or no ]
solventForcefieldSmallMoleculeForcefield, openff-2.1.1 [ Possible
    value: A valid forcefield name. ]

Solvent output settings:

solventOutputCheckpointInterval, 1.0 [ Units: nanosecond ]
solventOutputCheckpointStorageFilename, solvent_checkpoint.nc
solventOutputForcefieldCache, db.json
solventOutputFilename, solvent.nc
solventOutputIndices, not water [ Possible value: Any valid
    selection. ]
solventOutputStructure, hybrid_system.pdb
solventOutputPositionsWriteFrequency, 100.0 [ Units: picosecond ]
solventOutputVelocitiesWriteFrequency, None [ Possible
    values: > 0; Units: picosecond ]

```

Solvent simulation settings:

```

solventSimulationEarlyTerminationTargetError, 0.0 [ Units:
    kilocalorie_per_mole ]
solventSimulationEquilibrationLength, 1.0 [ Units: nanosecond ]
solventSimulationMinimizationSteps, 5000
solventSimulationNReplicas, 14
solventSimulationProductionLength, 10.0 [ Units: nanosecond ]
solventSimulationRealTimeAnalysisInterval, 250.0 [ Units:
    picosecond ]
solventSimulationRealTimeAnalysisMinimumTime, 500.0 [ Units:
    picosecond ]
solventSimulationSamplerMethod, repex [ Possible values: repex,
    sams, or independent ]
solventSimulationSamsFlatnessCriteria, logZ-flatness [ Possible
    values: logZ-flatness, minimum-visits or histogram-flatness ]
solventSimulationsamsGamma0, 1.0
solventSimulationTimePerIteration, 2.5 [ Units: picosecond ]

```

Thermo settings:

```

thermoPh, None [ Possible values: > 0 ]
thermoPressure, 1.0 [ Units: bar ]
thermoRedoxPotential, None [ Possible values: A valid float.
    Units: millivolts (mV) ]
thermoTemperature, 298.15 [ Units: kelvin ]

```

Vacuum engine settings:

```

vacuumEngineComputePlatform, CPU [ Possible values: CPU, CUDA,
    OpenCL, or Reference ]
vacuumEngineGpuDeviceIndex, None [ Possible values: 0, 1, etc. ]

```

Vacuum equil output settings:

```

vacuumEquilOutputCheckpointInterval, 1.0 [ Units: nanosecond ]
vacuumEquilOutputCheckpointStorageFilename, checkpoint.chk
vacuumEquilOutputNPTStructure, equil_structure.pdb
vacuumEquilOutputNVTStructure, None
vacuumEquilOutputForcefieldCache, db.json
vacuumEquilOutputLogOutput, equil_simulation.log
vacuumEquilOutputMinimizedStructure, minimized.pdb
vacuumEquilOutputIndices, not water [ Possible value: Any valid
    selection. ]
vacuumEquilOutputPreminimizedStructure, system.pdb
vacuumEquilOutputProductionTrajectoryFilename, production_equil.xtc
vacuumEquilOutputTrajectoryWriteInterval, 20.0 [ Units: picosecond ]

```

Vacuum equil simulation settings:

```

vacuumEquilSimulationEquilLength, 0.2 [ Units: nanosecond ]
vacuumEquilSimulationEquilLengthNVT, None [ Units: nanosecond ]
vacuumEquilSimulationMinimizationSteps, 5000
vacuumEquilSimulationProductionLength, 0.5 [ Units: nanosecond ]

```

Vacuum forcefield settings:

```

vacuumForcefieldConstraints, HBonds [ Possible values: HBonds,
    AllBonds, or HAngles ]
vacuumForcefields, amber/ff14SB.xml, amber/tip3p_standard.xml
    amber/tip3p_HFE_multivalent.xml amber/phosaa10.xml
    [ Possible values: A space delimited list of valid names. ]
vacuumForcefieldHydrogenMass, 3.0 [ Units: amu ]
vacuumForcefieldNonbondedCutoff, 0.9 [ Units: nanometer ]
vacuumForcefieldNonbondedMethod, nocutoff [ Possible values: PME
    or NoCutoff ]
vacuumForcefieldRigidWater, yes, [ Possible values: yes or no ]
vacuumForcefieldSmallMoleculeForcefield, openff-2.1.1 [ Possible
    value: A valid forcefield name. ]

```


Vacuum output settings:

```
vacuumOutputCheckpointInterval, 1.0 [ Units: nanosecond ]
vacuumOutputCheckpointStorageFilename, vacuum_checkpoint.nc
vacuumOutputForcefieldCache, db.json
vacuumOutputFilename, vacuum.nc
vacuumOutputIndices, not water [ Possible value: Any valid
    selection. ]
vacuumOutputStructure, hybrid_system.pdb
vacuumOutputPositionsWriteFrequency, 100.0 [ Units: picosecond ]
vacuumOutputVelocitiesWriteFrequency, None [ Possible
    values: > 0; Units: picosecond ]
```

Vacuum simulation settings:

```
vacuumSimulationEarlyTerminationTargetError, 0.0 [ Units:
    0.0 kilocalorie_per_mole ]
vacuumSimulationEquilibrationLength, 0.5 [ Units: nanosecond ]
vacuumSimulationMinimizationSteps, 5000
vacuumSimulationNReplicas, 14
vacuumSimulationProductionLength, 2.0 [ Units: nanosecond ]
vacuumSimulationRealTimeAnalysisInterval, 250.0 [ Units: picosecond ]
vacuumSimulationRealTimeAnalysisMinimumTime, 500.0 [ Units: picosecond]
vacuumSimulationSamplerMethod, repex [ Possible values: repex,
    sams, or independent ]
vacuumSimulationSamsFlatnessCriteria, logZ-flatness [ Possible
    values: logZ-flatness, minimum-visits or histogram-flatness ]
vacuumSimulationSamsGamma0, 1.0
vacuumSimulationTimePerIteration, 2.5 [ Units: picosecond ]
```

Thermo settings:

```
thermoPh, None [ Possible values: > 0 ]
thermoPressure, 0.98692327 [ Units: standard_atmosphere ]
thermoRedoxPotential, None [ Possible values: A valid float.
    Units: millivolts (mV) ]
thermoTemperature, 298.15 [ Units: kelvin ]
```

A brief description of parameters, taken from OpenFE documentation, is provided below:

protocolRepeats: Number of completely independent repeats of the entire sampling process.

Integrator settings:

Parameters controlling the LangevinSplittingDynamicsMove integrator used for simulation.

```
integratorBarostatFrequency: Frequency at which volume scaling
    changes should be attempted.
integratorConstraintTolerance: Tolerance for constraint solver.
integratorLangevinCollisionRate: Collision frequency.
integratorNRestartAttempts: Number of attempts to restart from
    Context in case there are NaNs in the energies after
    integration.
integratorReassignVelocities: Reassign velocities from the
    Maxwell-Boltzmann distribution at the beginning of each
    Monte Carlo move.
integratorRemoveCom: Remove the center of mass motion.
integratorTimestep: Size of the simulation timestep.
```

Lambda settings:

Lambda protocol parameters, including number of lambda windows and

lambdaElec: List of lambda values for electrostatics. The values of 0 and 1 imply state A and state B respectively.

lambdaRestraints: List of lambda values for restraints. The values of 0 and 1 imply state A and state B respectively.

lambdaVdw: List of lamda values for van der Waals. The values of
of 0 and 1 imply state A and state B respectively.

Partial charge settings:

Parameters for automatically assigning missing partial charges to
small molecules, including the partial charge method.

partialChargeNaglModel: Model to use for partial charge assignment.
A value of None implies the use of the latest available
production AM1BCC model.

partialChargeNumberOfConformers: Number of conformers to generate
as part of the partial charge assignment. A value of None
implies the use of the existing conformer.

partialChargeOffToolkitBackend: OpenFF toolkit registry backend to
use for calculating partial charges.

partialChargeMethod: Method to use for calculating partial charges.

Solvation settings:

Solvation parameters for the system, including the solvent model and
the solvent padding.

solvationBoxShape: Shape of the periodic solvent box to create.

solvationBoxSize: Lengths of the unit cell for a solvent box.

solvationSolventModel: Forcefield water model to use during
solvation and defining the model properties.

solvationSolventPadding: Minimum distance from any solute bounding
sphere to the edge of the box.

Solvent engine settings:

Parameters configuring the compute platform used by the OpenMM to
perform the simulation.

solventEngineComputePlatform: Platform to use for running OpenMM MD
calculations.

solventEngineGpuDeviceIndex: Space delimited list of device indices
to use for running OpenMM MD calculations.

Solvent equil output settings:

Parameters controlling simulation output during equilibration
phase of solvent transformation.

solventEquilOutputCheckpointInterval: Frequency to write the
checkpoint file.

solventEquilOutputCheckpointStorageFilename: Checkpoint filename.

solventEquilOutputNPTStructure: NPT structure filename.

solventEquilOutputNVTStructure: NVT structure filename.

solventEquilOutputForcefieldCache: Filename for caching small
molecule residue templates.

solventEquilOutputLogOutput: Simulation log filename.

solventEquilOutputMinimizedStructure: Minimized structure filename.

solventEquilOutputIndices: Selection string for selecting
coordinates to write.

solventEquilOutputPreminimizedStructure: Initial structure filename.

solventEquilOutputProductionTrajectoryFilename: Trajectory filename.

solventEquilOutputTrajectoryWriteInterval: Frequency for writing
velocities to trajectory file.

Solvent equil simulation settings:

Parameters controlling simulation during equilibration phase of
solvent transformation.

solventEquilSimulationEquilLength: Length of the NPT equilibration
phase.

solventEquilSimulationEquilLengthNVT: Length of the NVT equilibration

phase.
solventEquilSimulationMinimizationSteps: Maximum number of minimization steps to perform.
solventEquilSimulationProductionLength: Length of the NPT production phase.

Solvent forcefield settings:

Parameters to set up the force field with OpenMM Force Fields equilibration phase of solvent transformation.

solventForcefieldConstraints: Constraints to use.
solventForcefields: List of valid forcefield paths for all components except small molecules.
solventForcefieldHydrogenMass: Mass to be repartitioned to hydrogens from neighboring heavy atoms.
solventForcefieldNonbondedCutoff: Cutoff for short range nonbonded interactions.
solventForcefieldNonbondedMethod: Method for treating nonbonded interactions.
solventForcefieldRigidWater: Use a rigid water model.
solventForcefieldSmallMoleculeForcefield: A valid forcefield name to use small molecules.

Solvent output settings:

Parameters controlling simulation output during final phase of solvent transformation.

solventOutputCheckpointInterval: Frequency to write the checkpoint file.
solventOutputCheckpointStorageFilename: Checkpoint filename.
solventOutputForcefieldCache: Filename for caching small molecule residue templates.
solventOutputFilename: Trajectory filename.
solventOutputIndices: Selection string for selecting coordinates to write.
solventOutputStructure: Hybrid topology structure filename.
solventOutputPositionsWriteFrequency: Frequency for writing positions to trajectory file.
solventOutputVelocitiesWriteFrequency: Frequency for writing velocities to trajectory file.

Solvent simulation settings:

Parameters controlling simulation during final phase of solvent transformation.

solventSimulationEarlyTerminationTargetError: Target error for the real time analysis measured in kcal/mol. Once the MBAR error of the free energy is at or below this value, the simulation will be considered complete. The suggested value of 0.12 has shown to be effective in both hydration and binding free energy benchmarks.
solventSimulationEquilibrationLength: Length of the equilibration phase. The specified value must be divisible by 'integratorTimestep'.
solventSimulationMinimizationSteps: Maximum number of minimization steps to perform.
solventSimulationNReplicas: Number of replicas to use.
solventSimulationProductionLength: Length of the production phase. The specified value must be divisible by 'integratorTimestep'.
solventSimulationRealTimeAnalysisMinimumTime: Time interval for performing analysis of the free energies. At each interval, real time analysis data will be written to a yaml file named <outputFileName>_real_time_analysis.yaml. The current error in the estimate will also be assessed and the simulation will be terminated when it drops below 'simulationEarlyTerminationTargetError'.
solventSimulationSamplerMethod: Minimum simulation time after

which the real time analysis is performed.

`solventSimulationSamplerMethod`: Alchemical sampling method to use: REPEX (Hamiltonian REPlICA EXchange), SAMS (Self-Adjusted Mixture Sampling), or Independent (Independently sampled lambda windows).

`solventSimulationSamsFlatnessCriteria`: Method for assessing when to switch to asymptotically optimal scheme for SAMS.

`solventSimulationSamsGamma0`: Initial weight adaptation rate for SAMS.

`solventSimulationTimePerIteration`: Simulation time between each MCMC move attempt

Vacuum engine settings:

Parameters configuring the compute platform used by the OpenMM to perform the simulation.

`vacuumEngineComputePlatform`: Platform to use for running OpenMM MD calculations.

`vacuumEngineGpuDeviceIndex`: Space delimited list of device indices to use for running OpenMM MD calculations.

The rest of the vacuum settings are similar to the solvent settings already described under various sections for solvent. The prefix 'vacuum' is used for the names of the parameters instead of the prefix 'solvent.'

Thermo settings:

Thermodynamic parameters, including the temperature and the pressure of the system.

`thermoPh`: Simulation pH
`thermoPressure`: Simulation pressure.
`thermoRedoxPotential`: Simulation redox potential.
`thermoTemperature`: Simulation temperature.

Arguments:

`ParamsOptionName (str)`: Command line OpenFE RFE parameters option name.
`ParamsOptionValue (str)`: Comma delimited list of parameter name and value pairs.
`ParamsDefaultInfo (dict)`: Default values to override selected parameters.

Returns:

dictionary: Processed parameter name and value pairs.

ProcessOptionOpenFECharge

`ProcessOptionOpenFECharge(OptionName, OptionValue)`

Process charge command line option and return a valid canonical charge method name.

Valid network names are: AM1BCC, AM1-Mulliken, Espaloma, Gasteiger, MMFF94 or NAGL

Arguments:

`OptionName (str)`: Command line charge option name.
`OptionValue (str)`: Command line charge option value.

Returns:

str: Canonical charge method name.

ProcessOptionOpenFEChargeParameters

`ProcessOptionOpenFEChargeParameters(ParamsOptionName, ParamsOptionValue, ChargeMethod, ParamsDefaultInfo=None)`

Process parameters for charge option and return a map containing processed parameter names and values.

The `ParamsOptionValue` is a comma delimited list of parameter name and value pairs to setup platform.

The supported parameter names along with their default and possible values are shown below:

```

naglModel, auto [ Possible value: A valid NAGL model name. By
                  default, it corresponds to the latest AM1BCC production model ]
toolkit, auto [ Possible values: RDKit or AmberTools. Default value:
                RDKit for Gasteiger and MMFF94; AmberTools for AM1BCC and
                AM1-Mulliken; Not used for Espaloma and NAGL. ]

numProcessors, 1 [ Only used for AM1BCC, AM1-Mulliken, Espaloma,
                  and NAGL ]

precision, 4
lineSize, 90

useConformer, auto [ Use current conformer. Possible values: yes or
                    no. Default value: no for Gasteiger using AmberToolkit;
                    otherwise, yes. ]

```

A brief description of parameters is provided below:

```

naglModel: NAGL model name. The latest AM1BCC NAGL production
           model is used by default. You must specify it explicitly in case no
           production model is available.
toolkit: Toolkit name. RDKit for Gasteiger and MMFF94; AmberTools
         for AM1BCC, AM1-Mulliken, and Gasteiger.

numProcessors: Number of processors. This is only used during the
              calculation of AM1BCC, AM1-Mulliken, Espaloma, and NAGL
              employing OpenFE method bulk_assign_partial_charges().

precision: Floating point precision for writing the calculated
          partial atomic charges.
lineSize: Line size for writing the calculated partial atomic
         charges to SD file as a string value for data field label
         'atom.dprop.PartialCharge'.

useConformer: Use current conformer. The current conformer is
             always used to calculate AM1BCC, Espaloma and NAGL charges
             using OpenFE method bulk_assign_partial_charges() and this
             option is ignored. In addition, the option value is passed to
             OpenFF method assign_partial_charges() during the calculation
             of AM1-Mulliken, Gasteiger and MMFF94 charges employing
             AmberTools or RDKit. The RDKit functions, however, ignore the
             conformer during the calculation of Gasteiger and MMFF94
             charges. The current conformer appears not used to calculate
             Gasteiger charges employing AmberTools.

```

Arguments:

```

ParamsOptionName (str): Command line OpenFE network parameters option name.
ParamsOptionValue (str): Comma delimited list of parameter name and value pairs.
ChargeMethod (str): Charge method name.
ParamsDefaultInfo (dict): Default values to override for selected parameters.

```

Returns:

```

dictionary: Processed parameter name and value pairs.

```

ProcessOptionOpenFEExecuteDAGParameters

```

ProcessOptionOpenFEExecuteDAGParameters(ParamsOptionName, ParamsOptionValue,
ParamsDefaultInfo=None)

```

Process parameters for protocol DAG execution and return a map containing processed parameter names and values.

The ParamsOptionValue is a comma delimited list of parameter name and value pairs to setup execution of protocol DAG.

The supported parameter names along with their default and possible values are shown below:

```

keepShared, yes [ Possible values: yes or no ]
keepScratch, no [ Possible values: yes or no ]
nRetries, 2 [ Possible values: >= 0. A value of 0 implies only 1 try. ]

```

A brief description of parameters is provided below:

keepShared: Keep shared directories after the execution of DAG.
 keepScratch: Keep scratch directories after the execution of DAG.
 nRetries: Number of times to attempt the execution.

Arguments:

ParamsOptionName (str): Command line execute DAG parameters option name.
 ParamsOptionValues (str): Comma delimited list of parameter name and value pairs.
 ParamsDefaultInfo (dict): Default values to override for selected parameters.

Returns:

dictionary: Processed parameter name and value pairs.

ProcessOptionOpenFEMapper

ProcessOptionOpenFEMapper(OptionName, OptionValue)

Process mapper command line option and return a list of valid mapper names.

Valid atom mapper names are: LOMAP or Kartograf

Arguments:

OptionName (str): Command line mapper option name.
 OptionValue (str): Comma delimited list of mapper option values.

Returns:

list: List of valid canonical mapper names.

ProcessOptionOpenFEMapperParameters

ProcessOptionOpenFEMapperParameters(ParamsOptionName, ParamsOptionValue, ParamsDefaultInfo=None)

Process parameters for mapper option and return a map containing processed parameter names and values.

The ParamsOptionValue is a comma delimited list of parameter name and value pairs to setup platform.

The supported parameter names along with their default and possible values are shown below:

```
lomapTime, 20, [ Units: seconds ]
lomapThreeD, yes [ Possible values: yes or no ]
lomapMax3D, 1.0 [ Units: Angstrom ]
lomapElementChange, yes [ Possible values: yes or no]
lomapSeed, None [ Possible value: A string. An empty string causes
  MCS search to start from scratch ]
lomapShift, no [ Possible values: yes or no]

kartografAtomMaxDistance, 0.95 [ Units: Angstrom ]
kartografAtomMapHydrogens, yes [ Possible values: yes or no ]
kartografMapHydrogensOnHydrogensOnly, No [ Possible values: yes or
  no ]
kartografMapExactRingMatchesOnly, yes [ Possible values: yes or no ]
kartografAllowPartialFusedRings, yes [ Possible values: yes or no ]
```

A brief description of parameters is provided below:

lomapTime: Time out for MCS algorithm.
 lomapThreeD: Use atom positions to prune symmetric mappings.
 lomapMax3D: Forbid mapping between atoms with distance more than specified value.
 lomapElementChange: Allow mappings that change an atom element.
 lomapSeed: An Empty SMARTS string causes MCS search to start from scratch.
 lomapShift: Keep pre-aligned atom positions for 3D position checks.

 kartografAtomMaxDistance: Geometric criteria for two atoms corresponding to maximum distance between them.
 kartografAtomMapHydrogens: Map hydrogens.
 kartografMapHydrogensOnHydrogensOnly: Map hydrogens only on hydrogens.

kartografMapExactRingMatchesOnly: Map rings with only matching ring size and bond orders. In addition, ring breaking is not permitted.
kartografAllowPartialFusedRings: Allow mapping of partially fused rings.

Arguments:

ParamsOptionName (str): Command line OpenFE mapper parameters option name.
ParamsOptionValue (str): Comma delimited list of parameter name and value pairs.
ParamsDefaultInfo (dict): Default values to override for selected parameters.

Returns:

dictionary: Processed parameter name and value pairs.

ProcessOptionOpenFEMissingChargeMode

ProcessOptionOpenFEMissingChargeMode(OptionName, OptionValue)

Process missing charge mode command line option and return a valid canonical value.

Valid values are: Calculate or Stop.

Arguments:

OptionName (str): Command line missing charge mode option name.
OptionValue (str): Command line missing charge mode option value.

Returns:

str: Canonical value for missing charge mode.

ProcessOptionOpenFEMoleculePairs

ProcessOptionOpenFEMoleculePairs(OptionName, OptionValue)

Process molecule pairs command line option and return a list of molecule names.

Arguments:

OptionName (str): Command line molecule pairs option name.
OptionValue (str): Command line molecule pairs option value.

Returns:

list or none: List of molecule names.

ProcessOptionOpenFENetwork

ProcessOptionOpenFENetwork(OptionName, OptionValue)

Process network command line option and return a valid canonical network name.

Valid network names are: LOMAP, MinimalSpanning, or Radial.

Arguments:

OptionName (str): Command line network option name.
OptionValue (str): Command line network option value.

Returns:

str: Canonical network name.

ProcessOptionOpenFENetworkParameters

ProcessOptionOpenFENetworkParameters(ParamsOptionName, ParamsOptionValue, ParamsDefaultInfo=None, RadialNetworkStatus=False)

Process parameters for network option and return a map containing processed parameter names and values.

The ParamsOptionValue is a comma delimited list of parameter name and value pairs to setup platform.

The supported parameter names along with their default and possible values are shown below:

lomapDistanceCutoff, 0.4
lomapMaxPathLength, 6
lomapRequireCycleCovering, yes [Possible values: yes or no]

```

minimalSpanningProgress, no [ Possible values: yes or no ]

radialCentralLigand, None [ Possible values: Valid ligand name ]

outputEdges, no [ Possible values: yes or no ]
outputNetworkFormat, svg [ Possible values: Any valid format. ]

```

A brief description of parameters is provided below:

```

lomapDistanceCutoff: Maximum distance/dissimilarity between two
                     molecules for an edge to be accepted.
lomapMaxPathLength: Maximum distance between any two molecules in
                     the resulting network
lomapRequireCycleCovering: Add cycles into the network

minimalSpanningProgress: Show progress using tqdm.

radialCentralLigand: Name of central ligand. A valid ligand name
                     must be specified to generate a radial ligand network.

outputEdges: Generate PNG image files for all edges in a ligand
              network.
outputNetworkFormat: Valid image file format for ligand network.
                     You must specify a valid format supported by Python module
                     Matplotlib. For example: PNG (.png), SVG (.svg), PDF (.pdf),
                     etc. In addition, the graphml file is always generated.

```

Arguments:

```

ParamsOptionName (str): Command line OpenFE network parameters option name.
ParamsOptionValue (str): Comma delimited list of parameter name and value pairs.
ParamsDefaultInfo (dict): Default values to override for selected parameters.
RadialNetworkStatus (bool): Radial network status.

```

Returns:

```

dictionary: Processed parameter name and value pairs.

```

ProcessOptionOpenFERelativeFreeEnergyChargeCorrectionParameters

```

ProcessOptionOpenFERelativeFreeEnergyChargeCorrectionParameters(ParamsOptionName,
ParamsOptionValue, ParamsDefaultInfo=None)

```

Process parameters for RBFE charge correction option and return a map containing processed parameter names and values.

The ParamsOptionValue is a comma delimited list of parameter name and value pairs to setup charge correction for RBFE calculations.

The supported parameter names along with their default and possible values are shown below:

```

alchemicalExplicitChargeCorrection, yes
simulationProductionLength = 20 * unit.nanosecond
simulationNReplicas, 22
lambdaWindows, 22

```

Arguments:

```

ParamsOptionName (str): Command line OpenFE RBFE charge correction
                        parameters option name.
ParamsOptionValue (str): Comma delimited list of parameter name and value pairs.
ParamsDefaultInfo (dict): Default values to override for selected parameters.

```

Returns:

```

dictionary: Processed parameter name and value pairs.

```

ProcessOptionOpenFERelativeFreeEnergyMode

```

ProcessOptionOpenFERelativeFreeEnergyMode(OptionName, OptionValue)

```

Process relative FE mode command line option and return a valid canonical value.

Valid values names are: MoleculePairs or MoleculeNetwork.

Arguments:

OptionName (str): Command line missing charge mode option name.
 OptionValue (str): Command line missing charge mode option value.

Returns:

str: Canonical value for missing charge mode.

ProcessOptionOpenFERelativeFreeEnergyParameters

ProcessOptionOpenFERelativeFreeEnergyParameters(ParamsOptionName,
 ParamsOptionValue, ParamsDefaultInfo=None)

Process parameters for RFE parameters option and return a map containing processed parameter names and values.

The ParamsOptionValue is a comma delimited list of parameter name and value pairs to setup RFE calculations.

The default values are automatically updated to match settings provided by OpenFE module RelativeHybridTopologyProtocol.

You must specify valid OpenFE values for these parameters. An extensive validation is not performed.

The supported parameter names along with their default and possible values are shown below:

protocolRepeats, 3

Alchemical settings:

alchemicalEndstateDispersionCorrection, no [Possible values:
 yes or no]
 alchemicalExplicitChargeCorrection, no [Possible values:
 yes or no]
 alchemicalExplicitChargeCorrectionCutoff, 0.8 [Units: nanometer]
 alchemicalSoftcoreLJ, Gapsys [Possible values: Gapsys or Beutler]
 alchemicalSoftcoreAlpha, 0.85
 alchemicalTurnOffCoreUniqueExceptions, no [Possible values:
 yes or no]
 alchemicalUseDispersionCorrection, no [Possible values: yes or no]

Engine settings:

engineComputePlatform, CPU [Possible values: CPU, CUDA, OpenCL,
 or Reference]
 engineGpuDeviceIndex, None [Possible values: 0, 0 1, etc.]

Forcefield settings:

forcefieldConstraints, HBonds [Possible values: HBonds, ALLBonds or
 HAngles]
 forcefields, ['amber/ff14SB.xml', 'amber/tip3p_standard.xml',
 'amber/tip3p_HFE_multivalent.xml', 'amber/phosaa10.xml']
 [Possible values: A space delimited list of valid names.]
 forcefieldHydrogenMass, 3.0 [Units: amu]
 forcefieldNonbondedCutoff, 0.9 [Units: nanometer]
 forcefieldNonbondedMethod, PME [Possible values: PME or NoCutoff]
 forcefieldRigidWater, yes [Possible values: yes or no]
 forcefieldSmallMoleculeForcefield, openff-2.1.1 [Possible value:
 A valid forcefield name.]

Integrator settings:

integratorBarostatFrequency, 25.0 * timestep [The specified value
 is a multiple of integratorTimestep.]
 integratorConstraintTolerance, 1e-06
 integratorLangevinCollisionRate, 1.0 [Units: 1 / picosecond]
 integratorNRestartAttempts, 20
 integratorReassignVelocities, no [Possible values: yes or no]
 integratorRemoveCom, no [Possible values: yes or no]
 integratorTimestep, 4.0 [Units: femtosecond]

Lambda settings:

lambdaFunctions, default [Possible values: Default, namd, or quarters]
 lambdaWindows, 11

Output settings:

outputCheckpointInterval, 1.0 [Units: nanosecond]
 outputCheckpointStorageFilename, checkpoint.chk
 outputForcefieldCache, db.json
 outputFilename, simulation.nc
 outputIndices, not water [Possible value: Any valid selection.]
 outputStructure, hybrid_system.pdb
 outputPositionsWriteFrequency, 100.0 [Units: picosecond]
 outputVelocitiesWriteFrequency, None [Possible values: > 0; Units: picosecond]

Partial charge settings:

partialChargeNaglModel, None [Default: Production AM1BCC model for NAGL; Possible value: Any valid name.]
 partialChargeNumberOfConformers, None [Possible value: > 0]
 partialChargeOffToolkitBackend, AmberTools [Possible values: AmberTools or RDKit]
 partialChargeMethod, AM1BCC [Possible values: AM1BCC, Espaloma, or NAGL]

Simulation settings:

simulationEarlyTerminationTargetError, 0.0 [Units: kilocalorie_per_mole]
 simulationEquilibrationLength, 1.0 [Units: nanosecond]
 simulationMinimizationSteps, 5000
 simulationNReplicas, 11
 simulationProductionLength, 5.0 [Units: nanosecond]
 simulationRealTimeAnalysisInterval, 250.0 [Units: picosecond]
 simulationRealTimeAnalysisMinimumTime, 500.0 [Units: picosecond]
 simulationSamplerMethod, repex [Possible values: repex, sams, or independent]
 simulationSamsFlatnessCriteria, logZ-flatness [Possible values: logZ-flatness, minimum-visits or histogram-flatness]
 simulationSamsGamma0, 1.0
 simulationTimePerIteration, 2.5 [Units: picosecond]

Solvation settings:

solvationBoxShape, dodecahedron [Possible values: cube, dodecahedron, or octahedron]
 solvationBoxSize, None [Possible value: A triplet of space X Y Z values; Units: nanometer]
 solvationSolventModel, tip3p [Possible values: tip3p, spce, tip4pew, or tip5p]
 solvationSolventPadding, 1.5 [Units: nanometer]

Thermo settings:

thermoPh, None [Possible values: > 0]
 thermoPressure, 1.0 [Units: bar]
 thermoRedoxPotential, None [Possible values: A valid float. Units: millivolts (mV)]
 thermoTemperature, 298.15 [Units: kelvin]

A brief description of parameters, taken from OpenFE documentation, is provided below:

protocolRepeats: Number of completely independent repeats of the entire sampling process.

Alchemical settings:

Parameters controlling the creation of the hybrid topology system, including various parameters ranging from softcore parameters to whether or not to apply an explicit charge correction for systems with net charge changes.

`alchemicalEndstateDispersionCorrection`: Employ extra unsampled endstate windows for long range correction.
`alchemicalExplicitChargeCorrection`: Explicitly account for a charge difference during the alchemical transformation by transforming a water to a counterion of the opposite charge of the formal charge difference.
`alchemicalExplicitChargeCorrectionCutoff`: Minimum distance from the system solutes from which an alchemical water can be chosen.
`alchemicalSoftcoreLJ`: Use LJ softcore function as defined by Gapsys [Ref 181] or Buetler [Ref 182].
`alchemicalSoftcoreAlpha`: Softcore alpha parameter.
`alchemicalTurnOffCoreUniqueExceptions`: Turn off interactions for new exceptions (not just 1,4s) at lambda 0 and old exceptions at lambda 1 between unique atoms and core atoms.
`alchemicalUseDispersionCorrection`: Use dispersion correction in the hybrid topology state.

Engine settings:

Parameters configuring the compute platform used by the OpenMM to perform the simulation.

`engineComputePlatform`: Platform to use for running OpenMM MD calculations.
`engineGpuDeviceIndex`: Space delimited list of device indices to use for running OpenMM MD calculations.

Forcefield settings:

Parameters to set up the force field with OpenMM Force Fields, including the general force fields, the small molecule force field, the nonbonded method, and the nonbonded cutoff.

`forcefieldConstraints`: Constraints to use.
`forcefields`: List of valid forcefield paths for all components except small molecules.
`forcefieldHydrogenMass`: Mass to be repartitioned to hydrogens from neighboring heavy atoms.
`forcefieldNonbondedCutoff`: Cutoff for short range nonbonded interactions.
`forcefieldNonbondedMethod`: Method for treating nonbonded interactions.
`forcefieldRigidWater`: Use a rigid water model.
`forcefieldSmallMoleculeForcefield`: A valid forcefield name to use for small molecules.

Integrator settings

Parameters controlling the LangevinSplittingDynamicsMove integrator used for simulation.

`integratorBarostatFrequency`: Frequency at which volume scaling changes should be attempted.
`integratorConstraintTolerance`: Tolerance for constraint solver.
`integratorLangevinCollisionRate`: Collision frequency.
`integratorNRestartAttempts`: Number of attempts to restart from Context in case there are NaNs in the energies after integration.
`integratorReassignVelocities`: Reassign velocities from the Maxwell-Boltzmann distribution at the beginning of each Monte Carlo move.
`integratorRemoveCom`: Remove the center of mass motion.
`integratorTimestep`: Size of the simulation timestep.

Lambda settings:

Lambda protocol parameters, including number of lambda windows and lambda functions.

lambdaFunctions: Function name to use for alchemical mutation.
lambdaWindows: Number of lambda windows to calculate.

Output settings:

Parameter controlling simulation output, including the frequency to write a checkpoint file, the selection string for writing selected coordinates, and the paths to the trajectory and output structure files.

outputCheckpointInterval: Frequency to write the checkpoint file.
outputCheckpointStorageFilename: Checkpoint filename.
outputForcefieldCache: Filename for caching small molecule residue templates.
outputFilename: Trajectory filename.
outputIndices: Selection string for selecting coordinates to write.
outputStructure: Hybrid topology structure filename.
outputPositionsWriteFrequency: Frequency for writing positions to trajectory file.
outputVelocitiesWriteFrequency: Frequency for writing velocities to trajectory file.

Partial charge settings:

Parameters for automatically assigning missing partial charges to small molecules, including the partial charge method.

partialChargeNaglModel: Model to use for partial charge assignment. A value of None implies the use of the latest available production AM1BCC model.
partialChargeNumberOfConformers: Number of conformers to generate as part of the partial charge assignment. A value of None implies the use of the existing conformer.
partialChargeOffToolkitBackend: OpenFF toolkit registry backend to use for calculating partial charges.
partialChargeMethod: Method to use for calculating partial charges.

Simulation settings:

Parameters controlling the simulation plan and the alchemical sampler, including the number of minimization steps, lengths of equilibration and production runs, the sampler method (e.g. Hamiltonian REplica EXchange (repex), and the time interval at which to perform an analysis of the free energies.

simulationEarlyTerminationTargetError: Target error for the real time analysis measured in kcal/mol. Once the MBAR error of the free energy is at or below this value, the simulation will be considered complete. The suggested value of 0.12 has shown to be effective in both hydration and binding free energy benchmarks.
simulationEquilibrationLength: Length of the equilibration phase. The specified value must be divisible by 'integratorTimestep'.
simulationMinimizationSteps: Number of minimization steps to perform.
simulationNReplicas: Number of replicas to use.
simulationProductionLength: Length of the production phase. The specified value must be divisible by 'integratorTimestep'.
simulationRealTimeAnalysisInterval: Time interval for performing analysis of the free energies. At each interval, real time analysis data will be written to a yaml file named <outputFileName>_real_time_analysis.yaml. The current error in the estimate will also be assessed and the simulation will be terminated when it drops below

```
'simulationEarlyTerminationTargetError'.
simulationRealTimeAnalysisMinimumTime: Minimum simulation time
after which the real time analysis is performed.
simulationSamplerMethod: Alchemical sampling method to use:
    REPEX (Hamiltonian REpLica EXchange), SAMS (Self-Adjusted
    Mixture Sampling), or Independent (Independently sampled lambda
    windows).
simulationSamsFlatnessCriteria: Method for assessing when to switch
to asymptotically optimal scheme for SAMS.
simulationSamsGamma0: Initial weight adaptation rate for SAMS.
simulationTimePerIteration: Simulation time between each MCMC move
attempt
```

Solvation settings:

Solvation parameters for the system, including the solvent model and the solvent padding.

```
solvationBoxShape: Shape of the periodic solvent box to create.
solvationBoxSize: Lengths of the unit cell for a solvent box.
solvationSolventModel: Forcefield water model to use during
solvation and defining the model properties.
solvationSolventPadding: Minimum distance from any solute bounding
sphere to the edge of the box.
```

Thermo settings:

Thermodynamic parameters, including the temperature and the pressure of the system.

```
thermoPh: Simulation pH.
thermoPressure: Simulation pressure.
thermoRedoxPotential: Simulation redox potential.
thermoTemperature: Simulation temperature.
```

Arguments:

```
ParamsOptionName (str): Command line OpenFE RFE parameters option name.
ParamsOptionValue (str): Comma delimited list of parameter name and value pairs.
ParamsDefaultInfo (dict): Default values to override selected parameters.
```

Returns:

```
dictionary: Processed parameter name and value pairs.
```

ProcessOptionOpenFERelativeFreeEnergySeparatedTopologyParameters

```
ProcessOptionOpenFERelativeFreeEnergySeparatedTopologyParameters(ParamsOptionName,
ParamsOptionValue, ParamsDefaultInfo=None)
```

Process parameters for RBE parameters option and return a map containing processed parameter names and values.

The ParamsOptionValue is a comma delimited list of parameter name and value pairs to setup RBE calculations using a separated topologies approach.

The default values are automatically updated to match settings provided by OpenFE module SepTopProtocol.

You must specify valid OpenFE values for these parameters. An extensive validation is not performed.

The supported parameter names along with their default and possible values are shown below:

```
protocolRepeats, 3
```

Complex equil output settings:

```
complexEquilOutputCheckpointInterval, 1 [ Units: nanosecond ]
complexEquilOutputCheckpointStorageFilename, checkpoint.chk
complexEquilOutputEquilNPTStructure, equil_npt.pdb
complexEquilOutputEquilNVTStructure, None
complexEquilOutputForcefieldCache, db.json
complexEquilOutputLogOutput, equil_simulation.log
```

```

complexEquilOutputMinimizedStructure, minimized.pdb
complexEquilOutputIndices, all [ Possible value: Any valid
selection. ]
complexEquilOutputPreminimizedStructure, system.pdb
complexEquilOutputProductionTrajectoryFilename, production_equil.xtc
complexEquilOutputTrajectoryWriteInterval, 20.0 [ Units:
picosecond ]

```

Complex equil simulation settings:

```

complexEquilSimulationEquilibrationLength, 0.1 [ Units: nanosecond ]
complexEquilSimulationEquilibrationLengthNVT, 0.1 [ Units:
nanosecond ]
complexEquilSimulationMinimizationSteps, 5000
complexEquilSimulationProductionLength, 2.0 [ Units: nanosecond ]

```

Complex lambda settings:

```

complexLambdaElecA, 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.25 0.5 0.75
1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 [ Possible values: A space
delimited list of values between 0.0 and 1.0 ]
complexLambdaElecB, 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 0.75 0.5 0.25
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 [ Possible values: A space
delimited list of values between 0.0 and 1.0 ]
complexLambdaRestraintsA, 0.0 0.05 0.1 0.3 0.5 0.75 1.0 1.0 1.0
1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 [ Possible values: A
space delimited list of values between 0.0 and 1.0 ]
complexLambdaRestraintsB, 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
1.0 1.0 1.0 0.75 0.5 0.3 0.1 0.05 0.0 [ Possible values: A space
delimited list of values between 0.0 and 1.0 ]
complexLambdaVdWA, 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.143 0.286 0.429 0.572 0.715 0.857 1.0 [ Possible values: A
delimited list of values between 0.0 and 1.0 ]
complexLambdaVdWB, 1.0 0.857 0.715 0.572 0.429 0.286 0.143 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 [ Possible values: A
delimited list of values between 0.0 and 1.0 ]

```

Complex output settings:

```

complexOutputCheckpointInterval, 1.0 [ Units: nanosecond ]
complexOutputCheckpointStorageFilename, complex_checkpoint.nc
complexOutputForcefieldCache, db.json
complexOutputFilename, complex.nc
complexOutputIndices, not water [ Possible value: Any valid
selection. ]
complexOutputStructure, alchemical_system.pdb
complexOutputPositionsWriteFrequency, 100.0 [ Units: picosecond ]
complexOutputVelocitiesWriteFrequency, None [ Possible
values: > 0; Units: picosecond ]

```

Complex restraint settings:

```

complexRestraintKPhiA, 334.72 [ Units: kilojoule_per_mole/radian**2
The default value is equivalent to 80 kcal/mol/radian**2 ]
complexRestraintKPhiB, 334.72 [ Units: kilojoule_per_mole/radian**2
The default value is equivalent to 80 kcal/mol/radian**2 ]
complexRestraintKPhiC, 334.72 [ Units: kilojoule_per_mole/radian**2
The default value is equivalent to 80 kcal/mol/radian**2 ]
complexRestraintKR, 4184.0 [ Units: kilojoule_per_mole/nanometer**2
The default value is equivalent to 10 kcal/mol/angstrom**2 ]
complexRestraintKThetaA, 334.72 [ Units:kilojoule_per_mole/radian**2
The default value is equivalent to 80 kcal/mol/radian**2 ]
complexRestraintKThetaB, 334.72 [ Units:kilojoule_per_mole/radian**2
The default value is equivalent to 80 kcal/mol/radian**2 ]
complexRestraintAnchorFindingStrategy, bonded [ Possible values:
multi-residue or bonded ]
complexRestraintDsspFilter, yes [ Possible values: yes or no ]
complexRestraintHostMaxDistance, 1.5 [ Units: nanometer ]
complexRestraintHostMinDistance, 0.5 [ Units: nanometer ]

```

complexRestrainedHostSelection, backbone [Possible value: Any valid selection.]
 complexRestrainedRmsfCutoff, 0.1 [Units: nanometer]

Complex simulation settings:

complexSimulationEarlyTerminationTargetError, 0.0 [Units: kilocalorie_per_mole]
 complexSimulationEquilibrationLength, 1.0 [Units: nanosecond]
 complexSimulationMinimizationSteps, 5000
 complexSimulationNReplicas, 19
 complexSimulationProductionLength, 10.0 [Units: nanosecond]
 complexSimulationRealTimeAnalysisInterval, 250.0 [Units: picosecond]
 complexSimulationRealTimeAnalysisMinimumTime, 500.0 [Units: picosecond]
 complexSimulationSamplerMethod, repex [Possible values: repex, sams, or independent]
 complexSimulationSamsFlatnessCriteria, logZ-flatness [Possible values: logZ-flatness, minimum-visits or histogram-flatness]
 complexSimulationSamsGamma0, 1.0
 complexSimulationTimePerIteration, 2.5 [Units: picosecond]

Complex solvation settings:

complexSolvationBoxShape, dodecahedron [Possible values: cube, dodecahedron, or octahedron]
 complexSolvationBoxSize, None [Possible value: A triplet of space X Y Z values; Units: nanometer]
 complexSolvationSolventModel, tip3p [Possible values: tip3p, spce, tip4pew, or tip5p]
 complexSolvationSolventPadding, 1.0 [Units: nanometer]

Engine settings:

engineComputePlatform, CPU [Possible values: CPU, CUDA, OpenCL, or Reference]
 engineGpuDeviceIndex, None [Possible values: 0, 1, etc.]

Forcefield settings:

forcefieldConstraints, HBonds [Possible values: HBonds, AllBonds, or HAngles]
 forcefields, amber/ff14SB.xml amber/tip3p_standard.xml
 amber/tip3p_HFE_multivalent.xml amber/phosaa10.xml
 [Possible values: A space delimited list of valid names.]
 forcefieldHydrogenMass, 3.0 [Units: amu]
 forcefieldNonbondedCutoff, 0.9 [Units: nanometer]
 forcefieldNonbondedMethod, PME [Possible values: PME or NoCutoff]
 forcefieldRigidWater, yes [Possible values: yes or no]
 forcefieldSmallMoleculeForcefield, openff-2.1.1 [Possible value: A valid forcefield name.]

Integrator settings:

integratorBarostatFrequency, 25.0 * timestep [The specified value is a multiple of integratorTimestep.]
 integratorConstraintTolerance, 1e-06
 integratorLangevinCollisionRate, 1.0 [Units: 1 / picosecond]
 integratorNRestartAttempts, 20
 integratorReassignVelocities, no [Possible values: yes or no]
 integratorRemoveCom, no [Possible values: yes or no]
 integratorTimestep, 4.0 [Units: femtosecond]

Partial charge settings:

partialChargeNaglModel, None [Default: Production AM1BCC model for NAGL; Possible value: Any valid name.]

```
partialChargeNumberOfConformers, None [ Possible value: > 0 ]
partialChargeOffToolkitBackend, AmberTools [ Possible values:
    AmberTools or RDKit ]
partialChargeMethod, AM1BCC [ Possible values: AM1BCC, Espaloma,
    or NAGL ]
```

Solvent equil output settings:

```
solventEquilOutputCheckpointInterval, 1.0 [ Units: nanosecond ]
solventEquilOutputCheckpointStorageFilename, checkpoint.chk
solventEquilOutputEquilNPTStructure, equil_npt.pdb
solventEquilOutputEquilNVTStructure, None
solventEquilOutputForcefieldCache, db.json
solventEquilOutputLogOutput, equil_simulation.log
solventEquilOutputMinimizedStructure, minimized.pdb
solventEquilOutputIndices, all [ Possible value: Any valid
    selection. ]
solventEquilOutputPreminimizedStructure, system.pdb
solventEquilOutputProductionTrajectoryFilename, equil_npt.xtc
solventEquilOutputTrajectoryWriteInterval, 20.0 [ Units:
    picosecond ]
```

Solvent_equil_simulation_settings:

```
solventEquilSimulationEquilibrationLength, 0.1 [ Units: nanosecond ]
solventEquilSimulationEquilibrationLengthNVT, 0.1 [ Units:
    nanosecond ]
solventEquilSimulationMinimizationSteps, 5000
solventEquilSimulationProductionLength, 2.0 [ Units: nanosecond ]
```

Solvent lambda settings:

```
solventLambdaElecA, 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.125
    0.25 0.375 0.5 0.625 0.75 0.875 1.0 1.0 1.0 1.0 1.0 1.0
    1.0 1.0 1.0 [ Possible values: A space delimited list of values
        between 0.0 and 1.0 ]
solventLambdaElecB, 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 0.875
    0.75 0.625 0.5 0.375 0.25 0.125 0.0 0.0 0.0 0.0 0.0 0.0
    0.0 0.0 0.0 [ Possible values: A space delimited list of values
        between 0.0 and 1.0 ]
solventLambdaRestrainsA, 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
    0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
    0.0 [ Possible values: A space delimited list of values between
        0.0 and 1.0 ]
solventLambdaRestrainsB, 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
    0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
    0.0 [ Possible values: A space delimited list of values between
        0.0 and 1.0 ]
solventLambdaVdwA, 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
    0.0 0.0 0.0 0.0 0.0 0.0 0.15 0.23 0.3 0.4 0.52 0.64 0.76 0.88
    1.0 [ Possible values: A space delimited list of values between
        0.0 and 1.0 ]
solventLambdaVdwB, 1.0 0.85 0.77 0.7 0.6 0.48 0.36 0.24 0.12 0.0
    0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
    0.0 [ Possible values: A space delimited list of values between
        0.0 and 1.0 ]
```

Solvent output settings:

```
solventOutputCheckpointInterval, 1.0 [ Units: nanosecond ]
solventOutputCheckpointStorageFilename, solvent_checkpoint.nc
solventOutputForcefieldCache, db.json
solventOutputFilename, solvent.nc
solventOutputIndices, not water [ Possible value: Any valid
    selection. ]
solventOutputStructure, alchemical_system.pdb
solventOutputPositionsWriteFrequency, 100.0 [ Units: picosecond ]
solventOutputVelocitiesWriteFrequency, None [ Possible
    values: > 0; Units: picosecond ]
```


Solvent restraint settings:

```
solventRestraintCentralAtomsOnly, No [ Possible values: yes or no ]
solventRestraintSpringConstant, 1000.0 [ Units: kilojoule_per_mole /
    nanometer ** 2. The default value is equivalent to 2.40
    kilocalorie_per_mole / angstromg ** 2 ]
```

Solvent simulation settings:

```
solventSimulationEarlyTerminationTargetError, 0.0 [ Units:
    kilocalorie_per_mole ]
solventSimulationEquilibrationLength, 1.0 [ Units: nanosecond ]
solventSimulationMinimizationSteps, 5000
solventSimulationNReplicas, 27
solventSimulationProductionLength, 10.0 [ Units: nanosecond ]
solventSimulationRealTimeAnalysisInterval, 250.0 [ Unit: picosecond ]
solventSimulationRealTimeAnalysisMinimumTime, 500.0 [ Units:
    picosecond ]
solventSimulationSamplerMethod, repex [ Possible values: repex,
    sams, or independent ]
solventSimulationSamsFlatnessCriteria, logZ-flatness [ Possible
    values: logZ-flatness, minimum-visits or histogram-flatness ]
solventSimulationSamsGamma0, 1.0
solventSimulationTimePerIteration, 2.5 [ Units: picosecond ]
```

Solvent solvation settings:

```
solventSolvationBoxShape, dodecahedron [ Possible values: cube,
    dodecahedron, or octahedron ]
solventSolvationBoxSize, None [ Possible value: A triplet of space
    X Y Z values; Units: nanometer ]
solventSolvationSolventModel, tip3p [ Possible values: tip3p, spce,
    tip4pew, or tip5p ]
solventSolvationSolventPadding, 1.5 [ Units: nanometer ]
```

Thermo settings:

```
thermoPh, None [ Possible values: > 0 ]
thermoPressure, 1.0 [ Units: bar ]
thermoRedoxPotential, None [ Possible values: A valid float.
    Units: millivolts (mV) ]
thermoTemperature, 298.15 [ Units: kelvin ]
```

A brief description of parameters, taken from OpenFE documentation, is provided below:

protocolRepeats: Number of completely independent repeats of the entire sampling process.

Complex settings:

Complex parameters for the system, including the solvent model and the solvent padding.

Complex equil output settings:

Parameters controlling simulation output during equilibration phase of complex transformation.

```
complexEquilOutputCheckpointInterval: Frequency to write the
    checkpoint file.
complexEquilOutputCheckpointStorageFilename: Checkpoint filename.
complexEquilOutputEquilNPTStructure: NPT structure filename.
complexEquilOutputEquilNVTStructure: NVT structure filename.
complexEquilOutputForcefieldCache: Filename for caching small
    molecule residue templates.
complexEquilOutputLogOutput: Simulation log filename.
complexEquilOutputMinimizedStructure: Minimized structure filename.
complexEquilOutputIndices: Selection string for selecting
    coordinates to write.
```

complexEquilOutputPremnimizedStructure: Initial structure filename.
complexEquilOutputProductionTrajectoryFilename: Trajectory filename.
complexEquilOutputTrajectoryWriteInterval: Frequency for writing velocities to trajectory file.

Complex equil simulation settings:

Parameters controlling simulation during equilibration phase of complex transformation.

complexEquilSimulationEquilibrationLength: Length of the NPT equilibration phase.
complexEquilSimulationEquilibrationLengthNVT: Length of the NVT equilibration phase.
complexEquilSimulationMinimizationSteps: Maximum number of minimization steps to perform.
complexEquilSimulationProductionLength: Length of the NPT production phase.

Complex lambda settings:

Lambda protocol parameters for complex transformation.

complexLambdaElecA: List of lambda values for electrostatics. The values of 0 and 1 imply state A and state B respectively.
complexLambdaElecB: List of lambda values for electrostatics. The values of 0 and 1 imply state A and state B respectively.
complexLambdaRestraintsA: List of lambda values for restraints. The values of 0 and 1 imply state A and state B respectively.
complexLambdaRestraintsB: List of lambda values for restraints. The values of 0 and 1 imply state A and state B respectively.
complexLambdaVdWA: List of lamda values for van der Waals. The values of of 0 and 1 imply state A and state B respectively.
complexLambdaVdWB: List of lamda values for van der Waals. The values of of 0 and 1 imply state A and state B respectively.

Complex output settings:

Parameters controlling simulation output during final phase of complex transformation.

complexOutputCheckpointInterval: Frequency to write the checkpoint file.
complexOutputCheckpointStorageFilename: Checkpoint filename.
complexOutputForcefieldCache: Filename for caching small molecule residue templates.
complexOutputFilename: Trajectory filename.
complexOutputIndices: Selection string for selecting coordinates to write.
complexOutputStructure: Topology structure filename.
complexOutputPositionsWriteFrequency: Frequency for writing positions to trajectory file.
complexOutputVelocitiesWriteFrequency: Frequency for writing velocities to trajectory file.

Complex restraint settings:

Parameters to configure Boresch-style restraint between two groups of atoms named host (Hx) and guest (Gx).

complexRestraintKPhiA: Equilibrium force constant for the dihedral formed by H2-H1-H0-G0.
complexRestraintKPhiB: Equilibrium force constant for the dihedral formed by H1-H0-G0-G1.
complexRestraintKPhiC: Equilibrium force constant for the dihedral formed by H0-G0-G1-G2.
complexRestraintKR: Bond spring constant between H0 and G0.
restraintKThetaA: Spring constant for the angle formed by H1-H0-G0.
complexRestraintKThetaA: Spring constant for the angle formed by

H1-H0-G0.
complexRestraintKThetaB: Spring constant for the angle formed by
H0-G0-G1.
complexRestraintAnchorFindingStrategy: Boresch atom picking strategy
to use. bonded: pick host atoms that are bonded to each other.
multi-residue: pick host atoms which can span multiple residues.
complexRestraintDsspFilter: Apply DSSP filter to the host atoms.
complexRestraintHostMaxDistance: Maximum distance between any
host atom and the guest G0 atom.
complexRestraintHostMinDistance: Minimum distance between any
host atom and the guest G0 atom
complexRestraintHostSelection: A valid selection string to
sub-select the host atoms which will be involved in the
restraint.
complexRestraintRmsfCutoff: Cutoff value for filtering atoms by their
root mean square fluctuation. Atoms with values above this
cutoff are ignored.

Complex simulation settings:

Parameters controlling simulation during final phase of complex
transformation.

complexSimulationEarlyTerminationTargetError: Target error for the
real time analysis measured in kcal/mol. Once the MBAR error of
the free energy is at or below this value, the simulation will
be considered complete. The suggested value of 0.12 has shown to
be effective in both hydration and binding free energy
benchmarks.
complexSimulationEquilibrationLength: Length of the equilibration
phase. The specified value must be divisible by
'integratorTimestep'.
complexSimulationMinimizationSteps: Maximum number of minimization
steps to perform.
complexSimulationNReplicas: Number of replicas to use.
complexSimulationProductionLength: Length of the production phase.
The specified value must be divisible by 'integratorTimestep'.
complexSimulationRealTimeAnalysisMinimumTime: Time interval for
performing analysis of the free energies. At each interval, real
time analysis data will be written to a yaml file named
<outputFileName>_real_time_analysis.yaml. The current error
in the estimate will also be assessed and the simulation will
be terminated when it drops below
'complexSimulationEarlyTerminationTargetError'.
complexSimulationSamplerMethod: Alchemical sampling method to use:
REPEX (Hamiltonian REpLica EXchange), SAMS (Self-Adjusted
Mixture Sampling), or Independent (Independently sampled lambda
windows).
complexSimulationSamsFlatnessCriteria: Method for assessing when to
switch to asymptotically optimal scheme for SAMS.
complexSimulationSamsGamma0: Initial weight adaptation rate for
SAMS.
complexSimulationTimePerIteration: Simulation time between each
MCMC move attempt

Complex solvation settings:

Solvation parameters for the system, including the solvent model and
the solvent padding.

complexSolvationBoxShape: Shape of the periodic solvent box.
complexSolvationBoxSize: Lengths of the unit cell for a solvent box.
complexSolvationSolventModel: Forcefield water model to use during
solvation and defining the model properties.
complexSolvationSolventPadding: Minimum distance from any solute
bounding sphere to the edge of the box.

Engine settings:

Parameters configuring the compute platform used by the OpenMM to perform the simulation.

engineComputePlatform: Platform to use for running OpenMM MD calculations.

engineGpuDeviceIndex: Space delimited list of device indices to use for running OpenMM MD calculations.

Forcefield settings:

forcefieldConstraints: Constraints to use.

forcefields: List of valid forcefield paths for all components except small molecules.

forcefieldHydrogenMass: Mass to be repartitioned to hydrogens from neighboring heavy atoms.

forcefieldNonbondedCutoff: Cutoff for short range nonbonded interactions.

forcefieldNonbondedMethod: Method for treating nonbonded interactions.

forcefieldRigidWater: Use a rigid water model.

forcefieldSmallMoleculeForcefield: A valid forcefield name to use for small molecules.

Integrator settings:

Parameters controlling the LangevinSplittingDynamicsMove integrator used for simulation.

integratorBarostatFrequency: Frequency at which volume scaling changes should be attempted.

integratorConstraintTolerance: Tolerance for constraint solver.

integratorLangevinCollisionRate: Collision frequency.

integratorNRestartAttempts: Number of attempts to restart from Context in case there are NaNs in the energies after integration.

integratorReassignVelocities: Reassign velocities from the Maxwell-Boltzmann distribution at the beginning of each Monte Carlo move.

integratorRemoveCom: Remove the center of mass motion.

integratorTimestep: Size of the simulation timestep.

Partial charge settings:

Parameters for automatically assigning missing partial charges to small molecules, including the partial charge method.

partialChargeNaglModel: Model to use for partial charge assignment. A value of None implies the use of the latest available production AM1BCC model.

partialChargeNumberOfConformers: Number of conformers to generate as part of the partial charge assignment. A value of None implies the use of the existing conformer.

partialChargeOffToolkitBackend: OpenFF toolkit registry backend to use for calculating partial charges.

partialChargeMethod: Method to use for calculating partial charges.

Solvent equil output settings:

Solvent equil simulation settings:

Solvent lambda settings:

Solvent output settings:

Solvent restraint settings:

Solvent simulation settings:

Solvent solvation settings:

The solvent settings are similar to the complex settings already described under various sections for complex. The prefix 'solvent' is used for the names of the parameters instead of the prefix 'complex.'

Thermo settings:

Thermodynamic parameters, including the temperature and the pressure of the system.

thermoPh: Simulation pH
 thermoPressure: Simulation pressure.
 thermoRedoxPotential: Simulation redox potential.
 thermoTemperature: Simulation temperature.

Arguments:

ParamsOptionName (str): Command line OpenFE RFE parameters option name.
 ParamsOptionValue (str): Comma delimited list of parameter name and value pairs.
 ParamsDefaultInfo (dict): Default values to override selected parameters.

Returns:

dictionary: Processed parameter name and value pairs.

ProcessOptionOpenFERelativeFreeEnergyVacuumParameters

ProcessOptionOpenFERelativeFreeEnergyVacuumParameters(ParamsOptionName, ParamsOptionValue, ParamsDefaultInfo=None)

Process parameters for RBFE vacuum option and return a map containing processed parameter names and values.

The ParamsOptionValue is a comma delimited list of parameter name and value pairs to setup charge correction for RBFE calculations.

The supported parameter names along with their default and possible values are shown below:

forcefieldNonbondedMethod, NoCutoff [Possible values: PME or NoCutoff]

Arguments:

ParamsOptionName (str): Command line OpenFE RBFE vacuum parameters option name.
 ParamsOptionValue (str): Comma delimited list of parameter name and value pairs.
 ParamsDefaultInfo (dict): Default values to override for selected parameters.

Returns:

dictionary: Processed parameter name and value pairs.

ProcessOptionOpenFEResultFileParameters

ProcessOptionOpenFEResultFileParameters(ParamsOptionName, ParamsOptionValue, ParamsDefaultInfo=None)

Process parameters for result file and return a map containing processed parameter names and values.

The supported parameter names along with their default and possible values are shown below:

precision, 4 [Possible values: > 0]
 delimiter, comma [Possible values: comma or tab]

Arguments:

ParamsOptionName (str): Command line result file parameters option name.
 ParamsOptionValues (str): Comma delimited list of parameter name and value pairs.
 ParamsDefaultInfo (dict): Default values to override for selected parameters.

Returns:

dictionary: Processed parameter name and value pairs.

ProcessOptionOpenFESolventParameters

ProcessOptionOpenFESolventParameters(ParamsOptionName, ParamsOptionValue, ParamsDefaultInfo=None)

Process parameters for solvation option and return a map containing processed parameter names and values.

The ParamsOptionValue is a comma delimited list of parameter name and value pairs to setup solvation parameters for creating OpenFE SolventComponent.

The supported parameter names along with their default and possible values are shown below:

```
positiveIon, Na+ [ Possible value: Li+, Na+, K+, Rb+, or Cs+ ]
negativeIon, Cl- [ Possible values: Cl-, Br-, F-, or I- ]
neutralize, yes [ Possible values: yes or no ]
ionConcentration, 0.15 [ Units: molar ]
```

A brief description of parameters is provided below:

```
positiveIon, negativeIon: Pair of ions used to neutralize and bring
the solvent to required ionic concentration.
neutralize: Neutralize the net charge on the chemical state by the
ions in the solvent component.
ionConcentration: Ionic concentration.
```

Arguments:

```
ParamsOptionName (str): Command line solvation parameters option name.
ParamsOptionValue (str): Comma delimited list of parameter name and value pairs.
ParamsDefaultInfo (dict): Default values to override for selected parameters.
```

Returns:

```
dictionary: Processed parameter name and value pairs.
```

ProcessRadialCentralLigandName

```
ProcessRadialCentralLigandName(Mols, RadialCentralMolName)
```

Check for the presence of the central ligand name, used for generating a radial ligand network, in a list of molecules and make sure it occurs only once in the list.

Arguments:

```
Mols (list): List of OpenFE molecule objects.
CentralMoleculeName (str): Molecule name.
```

Returns:

```
Object or none: OpenFE molecule object or None.
```

ReadAndValidateMolecules

```
ReadAndValidateMolecules(FileName, **KeywordArgs)
```

Read molecules from an input file, validate all molecule objects, and return a list of valid OpenFE SmallMoleculeComponent objects along with the count of valid and non-valid molecule objects.

Arguments:

```
FileName (str): Name of a file with complete path.
**KeywordArgs (dict) : Parameter name and value pairs for reading
and processing molecules.
```

Returns:

```
list or None: List of valid OpenFE molecule objects.
int : Number of total molecules in input file.
int : Number of valid molecules in input file.
```

The file extension is used to determine type of the file and set up an appropriate file reader.

ReadPDBFile

```
ReadPDBFile(PDBFile, Name="")
```

Read molecule from a PDB file.

The supported PDB file formats are: PDB(.pdb) and CIF (.cif)

Arguments:

```
PDBFile (str): PDB file path.
Name (str): Name of macromolecule.
```

Returns:

object: OpenFE PDB object.

SetupAbsoluteBindingFreeEnergySettings

SetupAbsoluteBindingFreeEnergySettings(ParamsOptionName, ParamsInfo)

Setup absolute binding free energy protocol settings to calculate ABFE.

The ParamsInfo is a comma delimited list of parameter name and value pairs returned by ProcessOptionOpenFEAbsoluteBindingFreeEnergyParameters().

Arguments:

ParamsOptionName (str): Command line OpenFE RFE parameters option name.
ParamsInfo (dict): Parameter name and value pairs.

Returns:

object: OpenFE AbsoluteBindingProtocol settings object.

SetupAbsoluteHydrationFreeEnergySettings

SetupAbsoluteHydrationFreeEnergySettings(ParamsOptionName, ParamsInfo)

Setup absolute hydration free energy protocol settings to calculate AHFE.

The ParamsInfo is a comma delimited list of parameter name and value pairs returned by ProcessOptionOpenFEAbsoluteHydrationFreeEnergyParameters().

Arguments:

ParamsOptionName (str): Command line OpenFE RFE parameters option name.
ParamsInfo (dict): Parameter name and value pairs.

Returns:

object: OpenFE AbsoluteSolvationProtocol settings object.

SetupRelativeFreeEnergySeparatedTopologySettings

SetupRelativeFreeEnergySeparatedTopologySettings(ParamsOptionName, ParamsInfo)

Setup relative binding free energy protocol settings to calculate RBFE using separated topology.

The ParamsInfo is a comma delimited list of parameter name and value pairs returned by ProcessOptionOpenFERelativeBindingFreeEnergySeparatedTopologyParameters().

Arguments:

ParamsOptionName (str): Command line OpenFE RFE parameters option name.
ParamsInfo (dict): Parameter name and value pairs.

Returns:

object: OpenFE SepTopProtocol settings object.

SetupRelativeFreeEnergySettings

SetupRelativeFreeEnergySettings(ParamsOptionName, ParamsInfo)

Setup relative free energy protocol settings to calculate RBFE.

The ParamsInfo is a comma delimited list of parameter name and value pairs returned by ProcessOptionOpenFERelativeFreeEnergyParameters().

Arguments:

ParamsOptionName (str): Command line OpenFE RFE parameters option name.
ParamsInfo (dict): Parameter name and value pairs.

Returns:

object: OpenFE RelativeHybridTopologyProtocol settings object.

SuggestAtomMappingsForMoleculePairs

SuggestAtomMappingsForMoleculePairs(MoleculePairs, Mappers, MapperScorer)

Suggest atom mapping between a pair of molecules using specified mappers and a scorer.

You may specify multiple mappers for generating mapping between pair of molecules. All specified mappers are employed to identify the highest scoring mapping between a pair of molecules.

Arguments:

Mols (list): List of OpenFE molecule objects.
Mappers (list): List of OpenFE mapper objects.
MapperScorer (object): OpenFE scorer object.

Returns:

list: List of OpenFE mapping objects.

UpdateRelativeFreeEnergySettingsForChargeCorrection

UpdateRelativeFreeEnergySettingsForChargeCorrection(ParamsOptionName, ParamsInfo, RBFESettings)

Update relative free energy protocol settings for charge correction.

The ParamsInfo is a comma delimited list of parameter name and value pairs returned by ProcessOptionOpenFERelativeFreeEnergyChargeCorrectionParameters().

Arguments:

ParamsOptionName (str): Command line OpenFE RBFE charge correction parameters option name.
ParamsInfo (dict): Parameter name and value pairs.
RBFESettings (dict): OpenFE RelativeHybridTopologyProtocol settings object.

Returns:

None

UpdateRelativeFreeEnergySettingsForVacuum

UpdateRelativeFreeEnergySettingsForVacuum(ParamsOptionName, ParamsInfo, RBFESettings)

Update relative free energy protocol settings for vacuum.

The ParamsInfo is a comma delimited list of parameter name and value pairs returned by ProcessOptionOpenFERelativeFreeEnergyVacuumParameters().

Arguments:

ParamsOptionName (str): Command line OpenFE RBFE vacuum parameters option name.
ParamsInfo (dict): Parameter name and value pairs.
RBFESettings (dict): OpenFE RelativeHybridTopologyProtocol settings object.

Returns:

None

WriteLigandNetworkGraphMLFile

WriteLigandNetworkGraphMLFile(LigandNetwork, GraphMLOutfile)

Write ligand network to a GraphML file.

Arguments:

LigandNetwork (object): OpenFE ligand network object.
GraphMLOutfile (str): GraphML file path.

Returns:

None

WriteLigandNetworkImageFile

WriteLigandNetworkImageFile(LigandNetwork, ImageOutfile)

Write ligand network to an image file.

You must specify a valid format supported by Python module Matplotlib. For example: PNG (.png), SVG

(.svg), PDF (.pdf), etc.

Arguments:

LigandNetwork (object): OpenFE ligand network object.
ImageOutfile (str): Image file path.

Returns:

None

WriteMappingImageFile

WriteMappingImageFile(Mapping, ImageOutfile)

Write mapping to an image file.

You must specify PNG (.png) format for the image file.

Arguments:

Mapping (object): OpenFE mapping object.
ImageOutfile (str): Image file path.

Returns:

None

WriteProtocolDAGResultFile

WriteProtocolDAGResultFile(ProtocolDAGResult, ProtocolResult, ResultsFilePath)

Write DAG results to a JSON file.

The file format and contents are based on the OpenFECLI code in quickrun.py.

Arguments:

ProtocolDAGResult (object): OpenFE DAG result object.
ProtocolResult (object): OpenFE protocol result object.
ResultsFilePath (str): File path.

Returns:

None

AUTHOR

Manish Sud <msud@san.rr.com>

COPYRIGHT

Copyright (C) 2026 Manish Sud. All rights reserved.

The functionality available in this script is implemented using OpenFE, an open source package for alchemical free energy calculations.

This file is part of MayaChemTools.

MayaChemTools is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 3 of the License, or (at your option) any later version.