Genarris

Release 2.0

Rithwik Tom, Timothy Rose, Imanuel Bier

Sep 20, 2019

Contents

CHAPTER 1

Introduction

1.1 Configuration File

Genarris is a random crystal structure generation code that can be adapted to perform *ab initio* crystal structure prediction. The modularity of Genarris is achieved through the sequential execution of procedures. The execution of Genarris is controlled by a configuration file. Below is a small example of a configuration file for Genarris.:

```
[Genarris_master]
procedures = ["Pygenarris_Structure_Generation"]

[pygenarris_structure_generation]
# Path to the single molecule file to used for crystal structure generation
molecule_path = relaxed_molecule.in
# Number of cores (MPI ranks) to run this section with
num_cores = 56
# Number of OpenMP Threads
omp_num_threads = 2
num_structures = 5000
Z = 4
sr = 0.85
tol = 0.00001
max_attempts_per_spg_per_rank = 1000000000
geometry_out_filename = glycine_4mpc.out
output_format = json
output_dir = glycine_4mpc_raw_jsons
```

Sections of the configuration file are denoted by square brakets, `[...]`. All parameters that are specified below a section are called **options**. The workflow of Genarris can be precisely controlled by the user by specifying the order of the desired procedures in `[Genarris_master]`. The user must also include the corresponding section for each procedure listed in `[Genarris_master]`. Each section may have many options which are required, optional, or inferred.

This document details the options for procedures that are executed in the Genarris 2.0 *Robust* workflow. In order these are:

```
["Relax_Single_Molecule",
 "Estimate_Unit_Cell_Volume",
 "Pygenarris_Structure_Generation",
 "Run_Rdf_Calc",
 "Affinity_Propagation_Fixed_Clusters",
 "FHI_Aims_Energy_Evaluation",
 "Affinity_Propagation_Fixed_Clusters",
 "Run_FHI_Aims_Batch"]
```

There are many options that can be specified and modified for each section. All of these options are specified in this document under the **Configuration File Options** section of each procedure.

1.2 Option Category

There are three *categories* of **Configuration File Options**. These are *required*, *optional*, and *inferred*. In the **Configuration File Options**, these categories are specified after the *type* of the option, such as *int*, *float*, or *bool*.

1. *Required* options have no category placed after the type in the documentation. These options are required to be in the configuration file for execution of Genarris.
2. *Optional* arguments are specified after the option *type*. These aregument have default settings built into the code perform well in general. The user may specify these *optional* arguments in the configuration file to have more control over the program executing.
3. *Inferred* options are specified after the option *type*. These options may be present in multiple different procedures. For example, the option `aims_lib_dir` is needed in the `Relax_Single_Molecule`, `FHI_Aims_Energy_Evaluation`, and `Run_FHI_Aims_Batch`. But, because it is an inferred parameter, it only needs to be specified once in the earliest procedure in which occurs and then it will be inferred by all further procedures. Options which are inferred are thus optional in all proceeding sections.

CHAPTER 2

Genarris 2.0 Procedures for Robust Workflow

2.1 Description

This section details all arguments and configuration file options for the procedures executed by the Robust Genarris 2.0 workflow. Each procedure is a class function of the of the `Genarris` master class. The documentation follows a standard format for each procedure. The name of the procedure is given first followed by a short description of the function the function it performs. Below the description is the the configuration file options subsection. This section gives the name, the data type, the *Option Category*, and a description of each option which is accepted by the procedure. By referencing this documentation, the user can obtain precise control over the execution of Genarris procedures.

2.2 Genarris Procedures

class `Genarris.genarris_master.`**Genarris**(*inst_path*)

Master class of Genarris. It controls all aspects of the Genarris workflow which can be executed individually or sequantially. Begins by reading and intepreting the configuration file. Calls the defined procedures with the options specified in the configuration file. Some options may be inferred from previous sections if they are not present in every section.

Affinity_Propagation_Fixed_Clusters(*comm*)

AP that explores the setting of preference in order to generate desired number of clusters.

Configuration File Options

output_dir [str] Path to the directory where the chosen structures will be stored.

preference_range [list] List of two values as the [min, max] of the range of allowable preference values.

structure_dir [str, inferred] Path to the directory of files to be used for the calculation. Default is to infer this value from the previous section.

dist_mat_input_file [str, inferred] Path to the distance matrix output from the descriptor calculation. Default is to infer this value from the previous sections.

output_format [str, optional] Format the structure files should be saved as. Default is both.

cluster_on_energy [bool, optional] Uses energy values to determine examplars. Structures with the lowest energy values from each cluster are selected. Default is False.

plot_histograms [bool, optional] If histogram plots should be created of the volume and space groups. Default is False.

num_of_clusters [int or float, optional] Float, must be less than 0. Selects a fraction of the structures. Int, selects specific number of structures equal to int. Default is 0.1.

num_of_clusters_tolerance [int, optional] Algorithm will stop if it has generated the number of clusters within the number of desired clusters and this tolerance. Default is 0.

max_sampled_preferences [int, optional] Maximum number of preference values to try.

output_without_success [bool, optional] Whether to perform output procedures if the algorithm has reached the maximum number of sampled preferences without finding the correct number of clusters. Default is False.

affinity_type [list, optional] List of [type of afinity, value] argument Scikit-Learn AP alogrithm.

affinity_matrix_path [str, optional] Path to the affinity matrix to use for the AP algorithm. Default is `affinity_matrix.dat`.

damping [float, optional] damping argument for Scikit-Learn AP algorithm. Default is 0.5.

convergence_iter [int, optional] convergence_iter argument for Scikit-Learn AP algorithm. Default is 15.

max_iter [int, optional] max_iter argument for Scikit-Learn AP algorithm. Default is 1000.

preference [int, optional] preference argument for Scikit-Learn AP algorithm. Default is None.

verbose_output [bool, optional] verbose argument for Scikit-Learn AP algorithm. Default is False.

property_key [str, optional] Key which the AP cluster will be stored in the properties of each structure object. Default is `AP_cluster`.

output_file [str, optional] Path where info about the AP alogrithm execution will be stored. Default is `./AP_cluster.info`.

exemplars_output_dir [str, optional] If provided, will output the examplars of each cluster to this folder. Default is None.

exemplars_output_format [str, optional] File format of structures to be output. Default is both.

structure_suffix [str, optional] Suffix to apply to structure files which are written. Default is `.json`.

output_dir_2: str, inferred Code automatically looks for the option output_dir_2 if the output directory already exists. This is how the code currently identifies that AP is running for a second time. Default behavior is to not use this option if output_dir does not already exist.

num_of_clusters_2: int or float, optional num_of_clusters for second clustering step. Default value is 0.1.

output_file_2 [str, inferred] Use if running AP algorithm twice, such as in the Robust workflow. Default is to use output_file.

exemplars_output_dir_2 [str, inferred] Exemplars output directory if second clustering step is used. Default is to use exemplars_output_dir.

cluster_on_energy_2 [str, inferred] How to choose examplars for the second clustering step. Default is to use cluster_on_energy value.

energy_name_2 [str, inferred] Energy name to use for second clustering step. Default is to use energy_name.

Estimate_Unit_Cell_Volume(*comm*)
Performs volume estimation using a machine learned model train on the CSD and based on Monte Carlo volume integration and topological molecular fragments. See Genarris 2.0 paper for full description.

Configuration File Options

volume_mean [float, optional] If provided, uses this value as the volume generation mean without using the ML model to etimate the volume.

volume_std [float, optional] If provided, uses this value for structure generation, otherwise a default value of 0.075 multiplied by the prediction volume per unit cell is provided.

FHI_Aims_Energy_Evaluation(*comm*, *world_comm*, *MPI_ANY_SOURCE*, *num_replicas*)
Runs Self-Consistent Field calculation on a pool of structures.

Configuration File Options

See *Run_FHI_Aims_Batch()*

Pygenarris_Structure_Generation(*comm*)
Uses the Genarris module written in C to perform structure generation. This module enables generation on special positions.

Configuration File Options

molecule_path [str] Path to the relaxed molecule geometry.

output_format [str, optional, default="json"] Determines the type of file which will be output for each structure. Can be one of: json, geo, both.

output_dir [str] Path to the directory which will contain all generated structures which pass the intermolecular distance checks.

num_structures [int] Target number of structures to generate.

Z [int] Number of molecules per cell to generate.

volume_mean [float, optional] See *Estimate_Unit_Cell_Volume()*

volume_std [float, optional] See *Estimate_Unit_Cell_Volume()*

sr [float, optional] Defines the minimum intermolecular distance that is considered physical by multiplying the sum of the van der Waals radii of the interacting atoms by sr. Default value is 0.85.

tol [float, optional] Tolerance to be used to identify space groups compatible with the input molecule.

max_attempts_per_spg_per_rank [int] Defines the maximum number of attempts the structure generator makes before moving on to the next space group.

num_structures_per_allowed_SG_per_rank [int] Number of structures per space group per rank which will be generated by Pygenarris.

geometry_out_filename [str] Filename where all structures generated by Pygenarris will be found.

omp_num_threads [int] Number of OpenMP threads to pass into Pygenarris

truncate_to_num_structures [bool] If true, will reduce pool to exactly the number defined by num_structures.

Run_Rdf_Calc(*comm*)
Runs RDF calculation for the pool of generated structures. RDF descriptor is similar to that described in Behler and Parrinello 2007. Then calculates the structure difference matrix.

Configuration File Options

structure_dir [str, inferred] Path to the directory of structures to evaluate.

dist_mat_fpath [str] Path to file to write distance matrix to.

output_dir [str] Path of directory to write structures to (will create if it DNE). If 'no_new_output_dir' then input structures will be overwritten.

normalize_rdf_vectors: bool,optional Whether to normalize the rdf vectors over the columns of the feature matrix before using them to compute the distance matrix. Default is Falase.

standardize_distance_matrix: bool If True, standardizes the distance matrix. The method is to divide all elements by the max value in the distance matrix. Because it is a distance matrix and thus all elements are positive, the standardized elements will be in the range [0, 1]. Default is False.

save_envs: bool, optional Whether to save the environment vectors calculated by the RDF method in the output structure files. Default is False.

cutoff [float, optional] Cutoff radius to apply to the atom centered symmetry function. Default is 12.

n_D_inter [int, optional] Number of dimensions to use for each type of pair-wise interatomic interaction found in the structure. Default is 12.

init_scheme [str, optional] Can be centered or shifted, as described in Gastegger et al. 2018. Default is shifted.

eta_range [list, optional] List of two floats which define the range for eta parameter in Gastegger et al. 2018. Default is [0.05,0.5].

Rs_range [list, optional] List of two floats which define the range for Rs parameter in Gastegger et al. 2018. Default is [[0.1,12].

pdist_distance_type [str,optional] Input parameter for the pdist function. Default is Euclidean.

Relax_Single_Molecule(*comm*, *world_comm*, *MPI_ANY_SOURCE*, *num_replicas*)
Calls run_fhi_aims_batch using the provided single molecule path.

Configuration File Options

See *Run_FHI_Aims_Batch()*

Run_FHI_Aims_Batch(*comm*, *world_comm*, *MPI_ANY_SOURCE*, *num_replicas*)
Runs FHI-aims calculations on a pool of structures using num_replicas.

Configuration File Options

verbose [bool] Controls verbosity of output.

energy_name [str] Property name which the calculated energy will be stored with in the Structure file.

output_dir [str] Path to the directory where the output structure file will be saved.

aims_output_dir [str] Path where the aims calculation will take place.

aims_lib_dir [str, inferred] Path to the location of the directory containing the FHI-aims library file.

molecule_path [str] Path to the geometry.in file of the molecule to be calculated if called using harris_single_molecule_prep or relax_single_molecule.

structure_dir [str, inferred] Path to the directory of structures to be calculated if calculation was called not using harris_single_molecule_prep or relax_single_molecule.

Z [int, inferred] Number of molecules per cell.