

Fort-BPsym Documentation

Adam Duster

December 17, 2018
v1.0

Abstract

Fort-BPsym is a program that takes geometry information about a molecule from an existing HDF5 file, generates the Behler-Parinello radial and angular symmetry functions for the geometries, and writes the results to a new HDF5 file.

Contents

1	Introduction	1
2	Usage	2
2.1	Installation and Execution	2
2.2	Testing the program	2
2.3	Input HDF5 Structure	2
2.4	Output HDF5 Structure	3
2.5	Basis vector layout	4
2.6	Creating an HDF5 File	4
3	Symmetry Function Definitions	5
3.1	Input Vectors	5
3.2	Graphs and Explanation	6
3.3	Symmetry Function Gradient	6

1 Introduction

Fort-BPsym is a program that takes geometry information about existing molecules, generates the Behler-Parinello (BP) radial and angular symmetry functions for the geometries, and writes the results to a new HDF5 file. Symmetry function vectors "scan" cartesian space for the presence of specific elements and can be used as input to a neural network.

Elements of the symmetry function vector correspond to scanning for elements at a specific radius from the i -th atom or the presence of a specific

combination of elements at a given angle, called radial and angular basis functions respectively. The number of angular and basis functions can be increased to an arbitrary value, increasing the resolution of the input vector at the cost of exponentially increasing the computational expense of training and evaluating the network, as well as increasing the number of parameters of the system.

The symmetry functions used are the

2 Usage

2.1 Installation and Execution

In order to install the program, you must first compile or install the HDF5 file library. On CentOS, the command is

```
yum install hdf5-devel
```

On Ubuntu:

```
apt install libhdf5-dev
```

Then add the location of the library to the `LD_LIBRARY_PATH` variable or edit the makefile with the location of the installed library. Next do the following

```
cd src
make all
```

The executable should be compiled and ready to run by typing:

```
./gen_symfunc_parallel [input_hdf5.h5] [output_hdf5.h5]
```

2.2 Testing the program

The program can be tested by executing the following commands from the program root directory:

```
cd test_files
cd test_parallel
./test.sh
```

The testing script should compile the program, and run it to generate the the output file `./2_waters-symfuncs.h5` from the input file

```
./test_files/2_water_clusters.h5.
```

It will then call the script `./test_files/compare_basis_functions.py` to compare the elements of the newly generated basis functions with the file `./test_files/reference.h5`. The script will output whether the reference elements are bigger, smaller, or equal to the newly generated one.

2.3 Input HDF5 Structure

This program requires the hdf5 file to have the following as top-level datasets. Note that `max-atoms` is a parameter that should be the same between all of the datasets and the dimensions are in row-major order (so should be backwards if this file is to be created by a FORTRAN program).

Dataset Name	Type	Description
<code>atomic_number</code>	<code>int1</code>	The atomic numbers of the atoms. Dimensions should be (num-geoms, max-atoms).
<code>cartesian_coordinates</code>	<code>real*8</code>	The cartesian coordinates in . Should have dimensions of (num-geoms, max-atoms, 3).
<code>num_atoms</code>	<code>int*2</code>	The number of atoms for a given geometry. Should have dimensions of (num-geoms).

2.4 Output HDF5 Structure

The program currently outputs the following hdf5 datasets. Note that the dimensions for each file are given in row major order and thus must be inverted for use with FORTRAN programs:

<code>o_radial_sym_funcs</code>	Oxygen radial symmetry function vector. Should have dimensions of (number of oxygens in all geometries, 48 = number of radial basis elements)
<code>h_radial_sym_funcs</code>	Hydrogen radial symmetry function vector. Should have dimensions of (number of hydrogens in all geometries, 48 = number of radial basis elements)
<code>h_angular_sym_funcs</code>	Hydrogen angular symmetry function vector. Should have dimensions of (number of hydrogens in all geometries, 36 = number of radial basis elements)
<code>o_angular_sym_funcs</code>	Oxygen angular symmetry function vector. Should have dimensions of (number of hydrogens in all geometries, 54 = number of radial basis elements)
<code>o_mol_index</code>	Key for linking basis function indices with a given molecule. This should have dimensions of (number of geometries, 2) For example, in a file which contains the symmetry function vectors for two geometries with 6 hydrogens in each geometry, this dataset would contain the following array: <code>[[0, 3],[3, 6]]</code> indicating that the rows 0, 1, and 2 belong to the first molecule and that rows 3, 4, and 5 belong to the second.
<code>h_mol_index</code>	See above

2.5 Basis vector layout

In the current implementation, the program has a fixed layout for the basis vectors. The used bond and angle types and the order of the types in this implementation are:

O bond	(H, O)
H bond	(H, O)
O angle	([O, O], [H, O], [H, H])
H angle	([H, H], [H, O])

As each bond type is described by 24 η and R_s parameters, this means that the oxygen symmetry function vector (has length of 48 for given atom) will have the first 24 elements devoted to the O-H bond and the last 24 elements to the O-O bond. This is the same for the angles. The η and R_s parameters for each bond type are as follows:

#	η	R_s	#	η'	ζ	λ
1	0.800	19.531	1	0.001	1	-1
2	1.113	10.090	2	0.001	1	1
3	1.426	6.146	3	0.001	4	-1
4	1.739	4.133	4	0.001	4	1
5	2.052	2.968	5	0.001	16	-1
6	2.365	2.234	6	0.001	16	1
7	2.678	1.743	7	0.010	1	-1
8	2.991	1.397	8	0.010	1	1
9	3.304	1.145	9	0.010	4	-1
10	3.617	0.955	10	0.010	4	1
11	3.930	0.809	11	0.010	16	-1
12	4.243	0.694	12	0.010	16	1
13	4.557	0.602	13	0.050	1	-1
14	4.870	0.527	14	0.050	1	1
15	5.183	0.465	15	0.050	4	-1
16	5.496	0.414	16	0.050	4	1
17	5.809	0.370	17	0.050	16	-1
18	6.122	0.334	18	0.050	16	1
19	6.435	0.302				
20	6.748	0.275				
21	7.061	0.251				
22	7.374	0.230				
23	7.687	0.212				
24	8.000	0.195				

2.6 Creating an HDF5 File

While any program can be used to create the input HDF5 file, one can be generated with a helper script included at the following location:

`./scripts/xyz2hdf5.py`

Instructions are included at the beginning of the script

3 Symmetry Function Definitions

3.1 Input Vectors

Elements of the symmetry vector correspond to scanning for elements at a specific radius from the i -th atom or the presence of a specific combination of elements at a given angle, called radial and angular basis functions respectively. The number of angular and basis functions can be increased to an arbitrary value, increasing the resolution of the input vector at the cost of exponentially increasing the computational expense of training and evaluating the network, as well as increasing the number of parameters of the system. A radial basis function has the following expression:

$$G_i^1 = \sum_{j \neq i}^{all} e^{-\eta(R_{ij}-R_s)^2} f_c(R_{ij}) \quad (1)$$

where R_s is the distance to scan, R_{ij} is the distance between the i -th and j -th atom, η is a parameter controlling the width of the gaussian function, and $f_c(R_{ij})$ is the smoothing function:

$$f_c(R_{ij}) = \begin{cases} \frac{1}{2} \cos\left(\frac{\pi R_{ij}}{R_s} + 1\right) & \text{for } R_{ij} \leq R_c \\ 0 & \text{for } R_{ij} > R_c \end{cases} \quad (2)$$

In this expression, R_{ij} is the distance between the i -th and the j -th atom and R_c is a parameter which represents the cutoff distance for the symmetry functions. The cutoff function is necessary to ensure that the basis functions do not change abruptly when atoms enter the largest distance that is to be scanned. It smoothly and monotonically decreases to 0 as R_{ij} approaches R_s .

The angular symmetry functions describe the angular distribution of two atom types surrounding a given atom by the following expression:

$$G_i^2 = 2^{1-\zeta} \sum_{j,k \neq i}^{all} (1 + \lambda \cos \theta_{ijk})^\zeta e^{-\eta'(R_{ij}^2 + R_{ik}^2 + R_{jk}^2)} f_c(R_{ij}) f_c(R_{ik}) f_c(R_{jk}) \quad (3)$$

OR:

$$G_i^2 = 2^{1-\zeta} \sum_{j,k \neq i}^{all} (1 + \lambda \cos \theta_{ijk})^\zeta e^{-\eta'(R_{ij} + R_{ik} + R_{jk})^2} f_c(R_{ij}) f_c(R_{ik}) f_c(R_{jk}) \quad (4)$$

Here, θ is the angle formed by the vectors from the i -th to the j -th and i -th to the k -th atoms, λ is a parameter used to invert the cosine function, η' is a parameter used to control the gaussian width, and ζ increases the specificity of the basis function for the given angle.

For each atom, a predetermined number of radial and symmetry functions must be selected and the symmetry function parameters do not change throughout the course of the NN weight optimization. For the proton transfer through water experiment proposed later, the symmetry functions will be based on ref and include 24 radial Gaussian-shape filters, with R_s values distributed evenly between 0.8 and 8.0. The width of the associated *eta* parameter will be proportional to the center's position by the expression $1/\sqrt{2\eta} = 0.2R_s$. The angular probe will use $\zeta = [1, 4, 16]$ for the filter widths, $\lambda = [-1, 1]$ for switching the filter's center between 0 and π , and $\eta' = [0.001, 0.01, 0.05]$ ($^{-2}$) for the levels of separation dependence. This will yield a feature of length 82 for H atoms, and 100 for O. Further details about the basis vectors can be found in appendix ??

3.2 Graphs and Explanation

As stated in the main text, a radial basis vector element is determined for the following expression:

$$G_i^1 = \sum_{j \neq i}^{all} e^{-\eta(R_{ij}-R_s)^2} f_c(R_{ij}) \quad (5)$$

with the cutoff function defined as:

$$f_c(R_{ij}) = \begin{cases} \frac{1}{2} \cos\left(\frac{\pi R_{ij}}{R_s}\right) + 1 & \text{for } R_{ij} \leq R_c \\ 0 & \text{for } R_{ij} > R_c \end{cases} \quad (6)$$

The affect of changing the parameters is shown in 1. Note that panel D is a representation of the way the radial symmetry functions will be used in this work.

Again for convenience, the equation for an angular basis vector component is reproduced here.

$$G_i^2 = 2^{1-\zeta} \sum_{j,k \neq i}^{all} (1 + \lambda \cos \theta_{ijk})^\zeta e^{-\eta'(R_{ij}^2 + R_{ik}^2 + R_{jk}^2)} f_c(R_{ij}) f_c(R_{ik}) f_c(R_{jk}) \quad (7)$$

The effect of changing of changing the parameters is shown in Figure 2

3.3 Symmetry Function Gradient

The gradients of the symmetry function terms are presented here in order of increasing complexity.

The gradient of the distance R_{ij} with respect to i is the unit vector along the line between the two points:

$$\frac{\partial R_{ij}}{\mathbf{X}_i} = -\frac{\mathbf{X}_j - \mathbf{X}_i}{\|\mathbf{X}_j - \mathbf{X}_i\|} \quad (8)$$

$$\frac{dR_{ij}}{d\mathbf{X}_j} = -\frac{\partial R_{ij}}{\mathbf{X}_i} \quad (9)$$

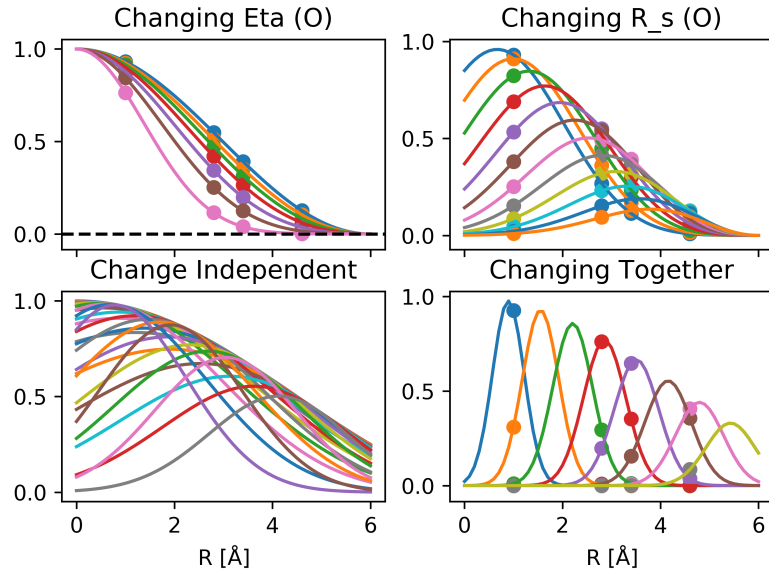


Figure 1: Changes in radial basis function while adjusting parameters. The colored points represent the value of the symmetry function for an atom located at R away from the given atom.

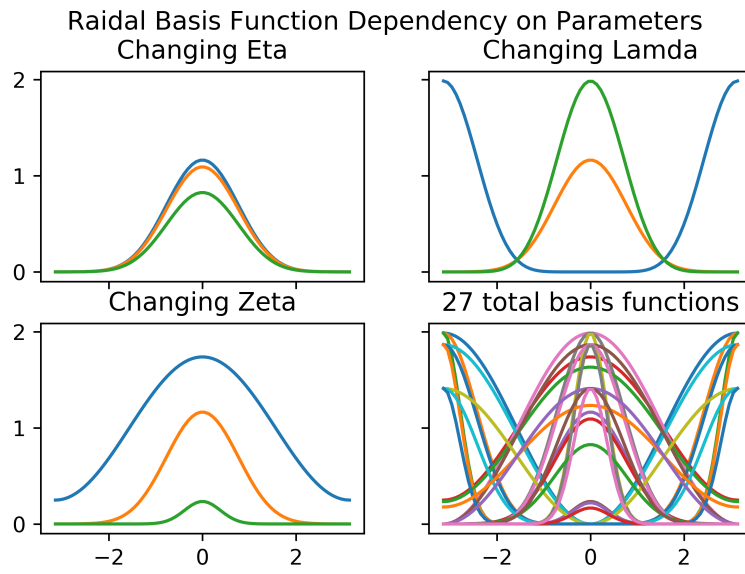


Figure 2: Changes in angular basis function while adjusting parameters. The colored points represent the value of the symmetry function for an atom located at R away from the given atom.

The gradient of the smoothing function with respect to a cartesian coordinate:

$$\frac{\partial f_c(R_{ij})}{\mathbf{X}} = -\frac{\pi}{2R_s} \sin\left(\frac{\pi R_{ij}}{R_s} + 1\right) \quad (10)$$

The gradient of the j -th component of the i -th radial symmetry function:

$$\frac{\partial G_{i,j}^2}{\mathbf{X}} e^{-\eta(R_{ij}-R_s)^2} \left[-2\eta(R_{ij}-R_s)f_c(R_{ij})\frac{dR_{ij}}{dR_{ij}} + \frac{df_c(R_{ij})}{dR_{ij}} \right] \quad (11)$$

The angular symmetry function terms and gradients:

$$\frac{\partial}{\partial \mathbf{X}} e^{-\eta'(R_{ij}^2+R_{ik}^2+R_{jk}^2)} = -2\eta' R_{ij} e^{-\eta'(R_{ij}^2+R_{ik}^2+R_{jk}^2)} \frac{dR_{ij}}{d\mathbf{X}} \quad (12)$$

$$\frac{\partial}{\partial \mathbf{X}} (1 + \lambda \cos \theta_{ijk}) = \zeta(1 + \lambda \cos \theta_{ijk})^{\zeta-1} \frac{\partial \cos \theta_{ijk}}{\partial \mathbf{X}} \quad (13)$$

As $\cos \theta$ can be expressed as:

$$\cos \theta_{ijk} = \frac{\mathbf{R}_{ij} \cdot \mathbf{R}_{ik}}{\|\mathbf{R}_{ij}\| \|\mathbf{R}_{ik}\|} \quad (14)$$

The gradient with respect to \mathbf{X}_j is:

$$\frac{\partial \cos \theta_{ijk}}{\partial \mathbf{X}_j} = -\frac{\|\mathbf{R}_{ik}\| \mathbf{R}_{ij} \cdot \mathbf{R}_{ik}}{\|\mathbf{R}_{ij}\| \|\mathbf{R}_{ik}\|^2} \frac{\partial \|\mathbf{R}_{ij}\|}{\partial \mathbf{X}_j} + \frac{\mathbf{R}_{ik}}{\|\mathbf{R}_{ij}\| \|\mathbf{R}_{ik}\|} \quad (15)$$

The gradient with respect to \mathbf{X}_i is:

$$\frac{\partial \cos \theta_{ijk}}{\partial \mathbf{X}_k} = -\frac{\mathbf{R}_{ij} \cdot \mathbf{R}_{ik}}{\|\mathbf{R}_{ij}\| \|\mathbf{R}_{ik}\|^2} \|\mathbf{R}_{ik}\| \left(\|\mathbf{R}_{ik}\| \frac{\partial \|\mathbf{R}_{ij}\|}{\partial \mathbf{X}_i} + \|\mathbf{R}_{ij}\| \frac{\partial \|\mathbf{R}_{ik}\|}{\partial \mathbf{X}_i} \right) + \frac{2\mathbf{X}_i - \mathbf{X}_j - \mathbf{X}_k}{\|\mathbf{R}_{ij}\| \|\mathbf{R}_{ik}\|} \quad (16)$$