High-Dimensional Neural Network Potential for the Malonaldehyde Molecule

Brandon-Lee De Bruyn

1 Introduction

This project implements a second-generation high-dimensional neural network potential (HDNNP) as proposed by Behler–Parrinello, where molecular energies are represented as sums of element-specific atomic contributions predicted from local, symmetry-preserving descriptors [1, 2]. A structural overview of the second-generation HDNNP pipeline is shown in Figure 1.

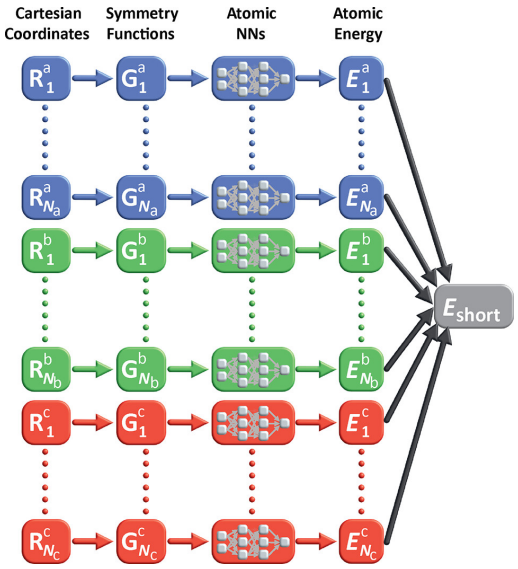

Figure 1: Second-generation HDNNP structure: Cartesian coordinates $\mathbf{R}_i^\mu \rightarrow$ ACSF descriptors $\mathbf{G}_i^\mu \rightarrow$ element-specific atomic NNs $\rightarrow$ atomic energies summed to total energy [1, 2].

2 Data Curation

For more efficient training, data splitting was performed at the frame/molecule level: unique frames were partitioned into training and validation sets using a 90/10 split (`val_size=0.1`, `random_state=42`), thus resulting in a train/validation/test frame distribution of 16200/1800/3600.

2.1 Atom-centered symmetry functions (ACSFs)

Since the HDNNP requires descriptors invariant to translation, rotation, and permutation of identical atoms, each atom environment was encoded using radial and angular ACSFs [1, 2]. All symmetry functions were localized using the smoothly decaying cosine cutoff function [2]:

$$f_c(R_{ij}) = \begin{cases} \frac{1}{2}\left[\cos\left(\pi \frac{R_{ij}}{R_{\text{cut}}}\right)+1\right], & R_{ij} \le R_{\text{cut}}, \\ 0, & R_{ij} > R_{\text{cut}}, \end{cases} \tag{1}$$

with $R_{\text{cut}} = 5.5$ and $R_{ij} = \|\mathbf{R}_j - \mathbf{R}_i\|$.

2.1.1 Element-resolved radial functions

Radial ACSFs describe the neighbor distribution around atom i. Element-resolved functions were used [2]:

$$G_{i,m}^{\text{rad}} = \sum_{\substack{j \neq i \\ R_{ij} \le R_{\text{cut}}}} \exp\left[-\eta_m (R_{ij} - R_s)^2\right] f_c(R_{ij}). \tag{2}$$

The parameter grid used in code was $\eta \in \{0.05,\ 0.5,\ 1.0,\ 2.0,\ 4.0,\ 8.0\}$, $R_s \in \text{linspace}(0, R_{\text{cut}}, 8)$.

Thus, $6 \times 8 = 48$ radial functions were computed *per neighbor element type* $Z \in \{\text{C,H,O}\}$, giving $3 \times 48 = 144$ radial features per atom.

2.1.2 Element-pair-resolved angular functions

Radial functions alone cannot distinguish environments with identical neighbor radii but different angular arrangements, therefore angular ACSFs were also computed [2]:

$$G_{i,m}^{\text{ang}} = 2^{1-\zeta_m} \sum_{\substack{j<k \\ R_{ij}, R_{ik} \le R_{\text{cut}}}} \left(1 + \lambda_m \cos\theta_{ijk}\right)^{\zeta_m} \exp\left[-\eta_m \left(R_{ij}^2 + R_{ik}^2 + R_{jk}^2\right)\right] f_c(R_{ij}) f_c(R_{ik}), \tag{3}$$

where θ_{ijk} is the angle at the central atom i. The code applies the cutoff only to R_{ij} and R_{ik} (central–neighbor distances), and includes R_{jk} only inside

the exponential. The parameter grid was $\eta \in \{0.0005, 0.005\}, \zeta \in \{1, 2, 4\}, \lambda \in \{-1, 1\}$.

This yields $2 \times 3 \times 2 = 12$ angular functions per unordered element pair. With three elements, the unordered pairs are (C,C), (C,H), (C,O), (H,H), (H,O), (O,O) (6 pairs), giving $6 \times 12 = 72$ angular features per atom.

Thus the final feature dimension is $D = 144 + 72 = 216$.

3 Model Design and Training

As mentioned previously, the implemented model follows the second-generation HDNNP architecture shown in Figure 1. Each atomic network maps the local ACSF descriptor of an atom to a scalar atomic energy contribution, and the total molecular energy is obtained by summation:

$$\hat{E} = \sum_{i=1}^{N} \hat{E}_i = \sum_{\mu} \sum_{i \in \mu} \mathrm{NN}_{\mu}(\mathbf{G}_i^{\mu}). \tag{4}$$

For the malonaldehyde molecule, three element-specific atomic neural networks were constructed corresponding to carbon (C), hydrogen (H), and oxygen (O). Each atomic network used an input dimension of $D = 216$, two fully connected hidden layers with 64 neurons each and `tanh` activation function.

To reduce overfitting and improve generalization, two forms of regularization were applied. First, L2 weight regularization (weight decay) was used on all dense layers, including the output layer, with a regularization coefficient of $\lambda = 10^{-6}$. Second, dropout regularization was applied after each hidden layer with a dropout rate of 0.05, randomly deactivating neurons during training to prevent co-adaptation[3].

3.1 Normalization and optimization

Feature and Energy standardization was applied per feature dimension using statistics computed on the training set, $X' = \frac{X - \mu_X}{\sigma_X}$ and $y' = \frac{y - \mu_y}{\sigma_y}$.

The model was trained with a mean-squared error (MSE) loss function using Adam [4] with learning rate 10^{-4}. Training used `batch_size=32` for up to 500 epochs. Two callbacks were used, namely, Early stopping on validation loss (`patience=30`, restoring best weights) and Reduce-on-plateau (`factor=0.25`, `patience=30`, `min_lr=`10^{-6}).

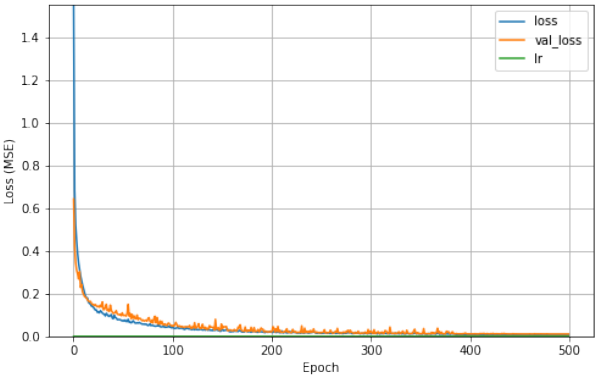

Figure 2: Training and validation MSE vs epoch.

4 Testing and Results

Performance was evaluated using MAE $= \frac{1}{F}\sum_f |\hat{y}_f - y_f|$, RMSE $= \sqrt{\frac{1}{F}\sum_f (\hat{y}_f - y_f)^2}$, and R^2 on the held-out test set. The final test performance (in the native energy units) was:

$$\mathrm{MAE} = 0.737, \mathrm{RMSE} = 0.996, R^2 = 0.721.$$

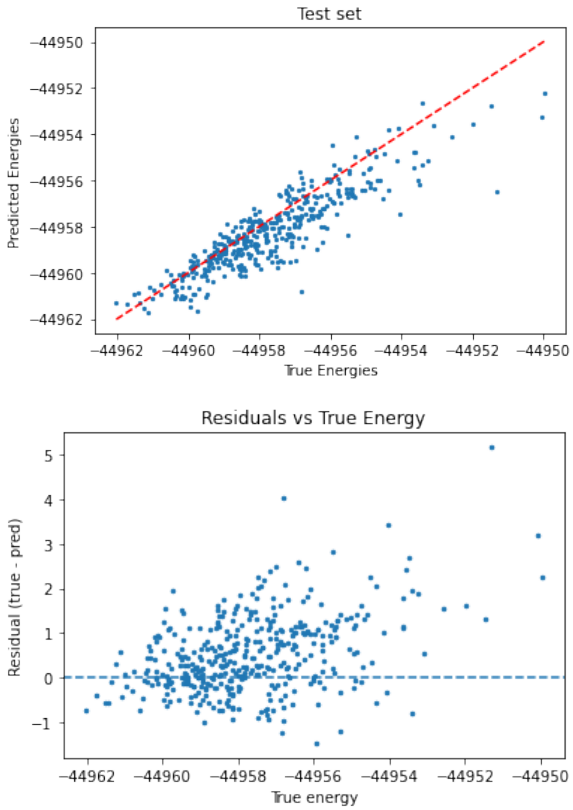

Figure 3: (Top) Predicted vs. true test energies. (Bottom) Residuals (true−pred) vs. true energy on the test set.

Figure 3 shows strong agreement between predicted and reference energies ($R^2 \approx 0.72$), with residuals broadly centered around zero, indicating that the ACSF-based second-generation HDNNP captures the dominant structure of the malonaldehyde energy landscape.

References

[1] Jörg Behler and Michele Parrinello. Generalized Neural-Network Representation of High-Dimensional Potential-Energy Surfaces. *Physical Review Letters*, 98(14):146401, April 2007.

[2] Jörg Behler. Four Generations of High-Dimensional Neural Network Potentials. *Chemical Reviews*, 121(16):10037–10072, August 2021.

[3] Geoffrey E. Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors, July 2012. arXiv:1207.0580 [cs].

[4] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. 2014. Version Number: 9.