ACEMD 4: protein-ligand complex simulations with a hybrid method of neural network potential and molecular mechanics

Galvelis, R.[1], Varela, A[1,2]., Doerr, S.[1], Fino, R.[1], Eastman, P.K.[3], De Fabritiis, G.[1,2,4]

[1]Acellera Labs, C/ Doctor Trueta 183, 08005 Barcelona, Spain

[2]Computational Science Laboratory, Universitat Pompeu Fabra, PRBB, C/ Doctor Aiguader 88, 08003 Barcelona, Spain

[3]Department of Chemistry, Stanford University, 11 Mill Site Road, Scotts Valley, CA95066, USA

[4]Institució Catalana de Recerca i Estudis Avançats (ICREA), Passeig Lluís Companys 23, 08010 Barcelona, Spain

1. Introduction

In 2008, ACEMD was the first and fastest code utilizing the graphical processing units (GPUs) to accelerate molecular dynamics (MD) simulations [Harvey2009]. This innovation has increased the speed of MD simulations by 1-2 orders of magnitude.

The remaining problem is the accuracy of the molecular mechanics (MM) and its force fields (FFs), but the recent advance of the neural network potential (NNP) has a potential to change that. NNP is based on an idea that a neural network (NN) is a *universal approximator*, which can be trained to predict the results of quantum mechanics (QM) calculations. Numerous NNPs have been proposed, but the most successful for organic molecules is ANI-2x [Devereux2020]. It is ~10^6 times faster than its reference QM calculations (ωB97X/6-31G*), but there are several limitations: no long-range interactions, only 7 elements (H, C, N, O, F, S, Cl), and only neutral molecules.

Despite these limitations, a hybrid method of NNP and MM (NNP/MM) has been proposed [Lahey2020]. The main idea is similar to QM/MM. An important region of a system is modeled with a more accurate method, while a less accurate and computationally cheaper one is used for the rest of the system. For example, the binding free energies of the Tyk2 congeneric ligand benchmark series have been computed with the alchemical free energy method and NNP/MM, which reduces the errors from 1.0 kcal/mol to 0.5 kcal/mol [Rufa2020]. However, the speed of NNP/MM is a critical limitation. Although NNP is much faster

than QM, it is still slower than MM. For example, the reported simulation speed is 3.4 ns/day and the longest simulation is 20 ns [Lahey2020]. It is ~2 orders of magnitude worse than the typical simulations with MM.

In this work, we present the release of ACEMD 4 with an optimized implementation of NNP/MM. The implementation is validated by performing MD simulations of 4 protein-ligand complexes with a combined sampling of 1 μs for each complex.

2. Method

We have implemented NNP/MM as proposed by Lahey et al. [Lahey2020]. A system is partitioned into two regions (NNP and MM) and coupled with following term:

$$V_{\text{NNP-MM}}(\vec{r}) = \sum_{i}^{N_{\text{NNP}}} \sum_{j}^{N_{\text{MM}}} \left(\frac{q_i q_j}{4\pi\epsilon_0 r_{ij}} + 4\epsilon_{ij} \left[\left(\frac{\sigma_{ij}}{r_{ij}} \right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}} \right)^{6} \right] \right),$$

where N_{NNP} and N_{MM} are the number of atoms, respectively; q_i and q_j are the atomic charges; ϵ_{ij} and σ_{ij} are the Lennard-Jones parameters; r_{ij} is the distance between the atoms; and ϵ_0 is the vacuum permittivity.

We have optimized the performance of NNP/MM in three ways. First, all the terms are computed on a GPU. Neither atomic position, nor atomic force need to be transferred between the CPU and GPU. Second, the featurizer of ANI-2x has been implemented as custom CUDA kernels. Third, the computation of the atomic NNs has been batched to reduce the number of CUDA kernels taking advantage that the same molecule is computed repeatedly. The custom kernels and the batched NNPs are available in the NNPOps library (https://github.com/openmm/NNPOps).

3. Results

We have selected 4 protein-ligand complexes from PDBbind 2019, where the ligand contains only elements supported by ANI-2x, there are no charged functional groups, the ligand has <100 atoms and at least one rotatable bond.

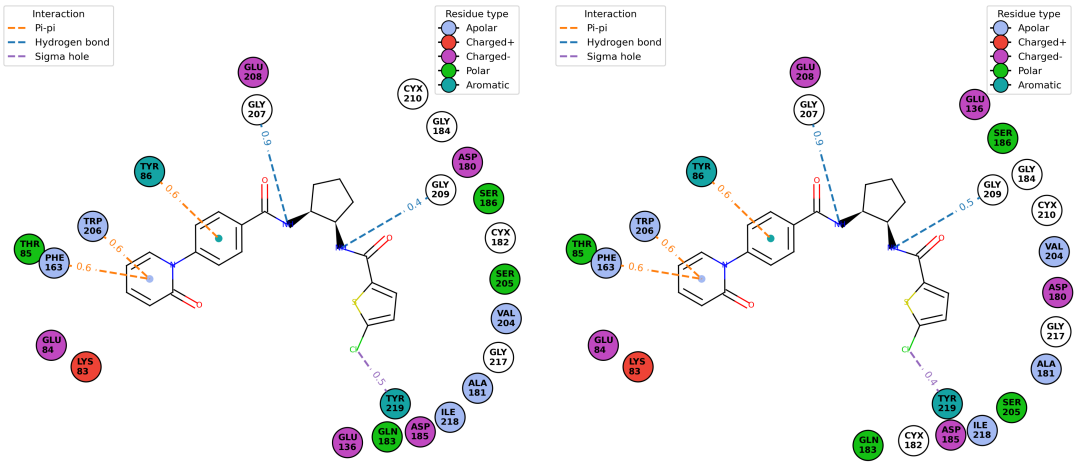
We have performed MD simulations (NVT, T = 310 K) of each complex with two methods MM (the ligands are modelled with GAFF2 parameters) and NNP/MM (the ligands are modelled with ANI-2x). In both cases, the proteins are modelled with AMBER ff14SB parameters. In total each complex with each method has been simulated 10 times for

100 ns (1 µs of combined sampling).

A comparison of the simulation speed (Table 1) shows that the performance of NNP/MM has been improved 3.9 times (NVIDIA GTX 1080 GPU). A comparison of protein-ligand interactions (Figure 1) shows comparable interactions and their probabilities for the 2P95 complex. The results for the rest complexes are similar.

Table 1: Comparison of MD simulation speed (ns/day) of NNP/MM using the original TorchANI and the TorchANI accelerated with NNPOps (TorchANI/NNPOps). For reference, MM speed is included. The results obtained with NVIDIA GTX 1080 GPU.

System	MM	NNP/MM (TorchANI)	NNP/MM (TorchANI/ NNPOps)	Speed up
1AJV	308	5.01	17.5	3.5
1HPO	254	5.04	18.8	3.7
2P95	168	4.35	19.8	4.6
3BE9	253	5.79	22.1	3.8
Average				3.9



(a) 2P95 with MM *(b) 2P95 with NNP/MM*

Figure 1 *Protein-ligand (2P95) interaction observed in the (a) MM and (b) NNP/MM simulations. The circles represent the protein residues in the binding pocket. The dotted lines represent specific interactions (i.e. hydrogen bonds, pi-pi interactions, and sigma holes) and their probabilities.*

4. Conclusions

We have implemented NNP/MM in ACEMD 4. The implementation is ~5 times faster than previously reported and the simulations of protein-ligand complexes are 50 times longer than previously reported. The analysis of the simulations show that NNP can successfully replace the conventional MM force field, but, due to lack of experimental data, we cannot verify the improvement of accuracy.

Nevertheless, ACEMD 4 represents a significant software improvement and feasibility demonstration of NNPs for MD simulations. This will become a platform to integrate other emerging NNPs (e.g. SchNet, TorchMD-Net, etc) which have a potential to bring more accuracy and remove the current limitations. Also, further speed improvements are possible. Currently, the NNP graphs are executed with PyTorch, but TensorRT, a low-latency inference library, is much faster and will be adopted in the future.

ACEMD 4, along with necessary tools and tutorials to set up an NNP/MM simulation, will be available at https://software.acellera.com/.

5. References

[Devereux2020] Devereux, C.; Smith, J. S.; Davis, K. K.; Barros, K.; Zubatyuk, R.; Isayev, O.; Roit-berg, A. E. Extending the Applicability of the ANI Deep Learning Molecular Potential to Sulfur and Halogens. *J. Chem. Theory Comput.* 2020, 16, 4192–4202.

[Galvelis2019] Galvelis, R.; Doerr, S.; Damas, J. M.; Harvey, M. J.; De Fabritiis, G. A Scalable Molecular Force Field Parameterization Method Based on Density Functional Theory and Quantum-Level Machine Learning. *J. Chem. Inf. Model* 2019, *59*, 3485–3493.

[Harvey2009] Harvey, M. J.; Giupponi, G.; Fabritiis, G. D. ACEMD: accelerating biomolecular dynamics in the microsecond time scale. *J. Chem. Theory Comput.* 2009, *5*, 1632–1639.

[Lahey2020] Lahey, S.-L. J.; Rowley, C. N. Simulating protein–ligand binding with neural network potentials. *Chem. Sci.* 2020, 11, 2362–2368.

[Rufa2020] Rufa, D. A.; Macdonald, H. E. B.; Fass, J.; Wieder, M.; Grinaway, P. B.; Roitberg, A. E.; Isayev, O.; Chodera, J. D. Towards chemical accuracy for alchemical free energy calculations with hybrid physics-based machine learning/molecular mechanics potentials. *BioRxiv* 2020.