Active learning for excited states dynamics simulations to discover molecular degradation pathways

Chen Zhou
AiMat group
Karlsruhe Institute of Technology
Engler-Bunte-Ring 8, 76131 Karlsruhe
chen.zhou@kit.edu

Prashant Kumar
Computational Photochemistry Lab
KU Leuven
3001 Heverlee, Belgiu
prashant.kumar@kuleuven.be

Daniel Escudero
Computational Photochemistry Lab
KU Leuven
3001 Heverlee, Belgiu
daniel.escudero@kuleuven.be

Pascal Friederich
AiMat group
Karlsruhe Institute of Technology
Engler-Bunte-Ring 8, 76131 Karlsruhe
pascal.friederich@kit.edu

Abstract

The demand for precise, data-efficient, and cost-effective exploration of chemical space has ignited growing interest in machine learning (ML), which exhibits remarkable capabilities in accelerating atomistic simulations of large systems over long time scales. Active learning is a technique widely used to reduce the cost of acquiring relevant ML training data. Here we present a modular, transferrable, and broadly applicable, parallel active learning orchestrator. Our workflow enables data and task parallelism for data generation, model training, and ML-enhanced simulations. We demonstrate its use in efficiently exploring multiple excited state potential energy surfaces and possible degradation pathways of an organic semiconductor used in organic light-emitting diodes. With our modular and adaptable workflow architecture, we expect our parallel active learning approach to be readily extended to explore other materials using state-of-the-art ML models, opening ways to AI-guided design and a better understanding of molecules and materials relevant to various applications, such as organic semiconductors or photocatalysts.

1 Introduction

Data science and machine learning have been brought into the spotlight of education, research, and industry of chemistry and material science [1]. Applications such as the generation and selection of molecule candidates [2], molecular property prediction [3, 4, 5, 6], reaction condition screening [7, 8] and product prediction[9, 10] have shown superior capacity of ML on accuracy and efficiency over conventional methods that are based on human intuition or quantum calculations.

To tackle the challenge related to the (computational) cost of data acquisition, active learning (AL) has become increasingly popular. Active learning allows the targeted identification of informative but unlabeled instances by querying information sources with a variety of strategies (e.g. query-by-committee [11]) and aims to reduce the amount of data needed to train highly accurate ML models, thereby minimizing the labeling cost [12] and maximizing data efficiency. Active learning has been applied successfully in fields such as molecular dynamics simulation [3, 13, 14, 15, 16] and reaction property prediction [17, 18].

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

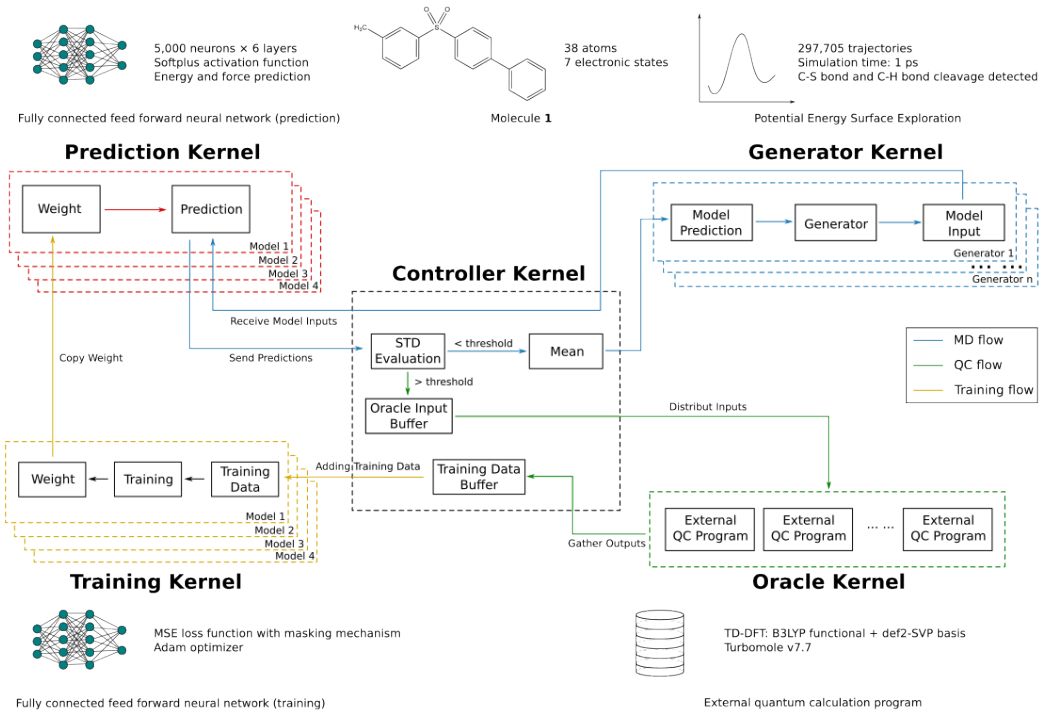


Figure 1: The computational architecture of the parallel active learning orchestration workflow.

Despite the improved performance, most current active learning algorithms still suffer from the overhead of serial execution of different tasks. For example, the model training process may halt and wait for new data while the information querying process is running. As the time required for querying and ML training processes could range from minutes to hours [13, 17, 18], this serial workflow usually fails to take full advantage of modern computational resources. To address this issue, we design a parallel active learning orchestrator that enables both data and task parallelism on computer clusters (Figure 1). The Message Passing Interface (MPI) [19] based workflow includes parallel execution of multiple learning, prediction, exploration, and data generation processes.

We demonstrate the capabilities of our approach by the application to dynamics simulations of molecule **1** (Figure 1) with 6 excited electronic states to explore its potential energy surface (PES) and investigate its degradation pathways. We deployed our active learning orchestrator on a hybrid CPU-GPU system parallelized across two nodes, with 4 ML training and 4 ML prediction processes on GPU coupled in parallel with 89 molecular dynamics (MD) simulations processes, and 30 quantum calculation (QC) processes based on CPU hardware. We achieve accurate predictions matching low-statistics reference calculations. Due to the speedup gained through the use of fully trained neural network (NN) potentials, we are able to explore possible degradation pathways of **1**, including C-S bond and C-H bond cleavage reactions.

2 Parallel active learning orchestrator

The active learning workflow we propose in this study consists of five kernels working in parallel: 1. prediction kernel, 2. generator kernel, 3. oracle kernel, 4. training kernel, and 5. controller kernel (see Figure 1). **Prediction kernel:** The NN models in the prediction kernel perform energy and force predictions for the same set of inputs. The average predictions are distributed to each simulation in the generator kernel. The model weights are updated by copying weights from the corresponding models in the training kernel after a given number of training epochs, to keep the prediction models as updated as possible. **Generator kernel:** An arbitrary number of simulations are running in the generator kernel, taking energy and force predictions to propagate MD trajectories which explore the input space to find unseen molecular geometries. **Oracle kernel:** To evaluate the

prediction uncertainty, the standard deviation of energy predictions of NNs in the prediction kernel is evaluated in each step. Predictions with a standard deviation above a given threshold are distributed to quantum chemistry calculations in the oracle kernel, to generate new labels for retraining. This strategy is known as query-by-committee [11] and is widely used for active learning. **Training kernel:** Quantum chemistry results are collected to enlarge the training set of the NN models in the training kernel (in our case 50 new data points per active learning iteration), which undergo continual training until new data points are added or early stopping is triggered to prevent overfitting. The weights of training kernel models are regularly copied to the prediction kernel, and training is restarted, in our case without resetting model weights or the learning rate scheduler. **Controller kernel:** Data communication as well as standard deviation calculation, and metadata storage (oracle input buffer and training data buffer) in the workflow are managed by a controller kernel. However, to ensure highly efficient and uninterrupted communication between the prediction and generator kernel, updated model weights are transferred directly from the training kernel to the prediction kernel. For the purpose of executing the parallel active learning workflow on distributed- and hybrid systems, data communication processes were implemented with the Message Passing Interface (MPI) based on the Python package mpi4py [20] to leverage parallel computing resources across multiple computational nodes. Further details on each kernel and on measures to ensure that the labeling of redundant data points during parallel execution is avoided can be found in the appendix.

In the specific active learning application presented in this work, active learning assisted molecular dynamics (ALMD) based surface hopping simulations with Zhu–Nakamura theory of surface hopping (ZNSH) [21] were used in the generator kernel to explore the excited state dynamics and degradation pathways of molecule **1**. Training data, i.e. electronic properties of the respective molecular geometries, is generated in the oracle kernel using time-dependent DFT (TD-DFT) calculations (B3-LYP functional and def2-SV(P) basis set). We used the NewtonX (v2.2) [22] and PyRAI2MD [3] packages for MD simulations in the generator kernel, respectively, and Turbomole (v7.7) [23] for QC calculations in the oracle kernel.

ALMD is initialized by sampling geometries from non-adiabatic molecular dynamics simulations (NAMD) trajectories from an initial data set. The fewest Switches Surface Hopping (FSSH) method was used for initial NAMD calculations that resulted in 29 trajectories and 94,419 data points with geometries, energies, and forces for 6 excited states (see appendix for more details).

As our use case aims at high accuracy and speed, rather than generalizability to other molecules, we use a fully connected neural network (NN) with an inverse distance representation inspired by prior work[14, 3]. The NN is trained to predict energies using a combined energy and force mean squared error loss function, with forces trained as derivatives of energies. The NN models consist of 6 softplus activated hidden layers, trained using the Adam optimizer.

3 Results and discussion

3.1 Accuracy and speed of the trained ML potential

To test and benchmark the accuracy and speed of ML potentials trained on initial and AL-generated data, we trained two times four neural in a bootstrapping manner, with 5,000 and 10,000 neurons per layer, respectively (see Table 1). For both NN sizes, training with the initial data and additional data from active learning results in $R^2 > 0.99$ for energy predictions, suggesting an almost linear correlation between NN predictions and QC ground truth labels. Due to memory limitations on the GPUs used here, we restricted the parallel active learning workflow to the NNs with 5,000 neurons per layer. Larger and potentially even more accurate models would be possible with appropriate data loaders, but the use of (equivariant) graph neural networks is a more promising alternative for further development.

The forward pass of a single molecular geometry of a single NN with 5,000 (10,000) neurons per layer to predict energies and forces takes 178.7 ms (461.2 ms) on a single CPU. As a comparison, a TD-DFT calculation requires on average 754 seconds for a single molecular geometry, which indicates a 4.2×10^3 (1.6×10^3) acceleration of using a NN to propagate MD compared to a DFT calculation. A further speedup can be obtained when using GPUs and parallelizing over many molecular geometries in a batch-wise way. A forward pass of 89 geometries in parallel with the 5,000 neurons per layer model on a GPU takes on average 51.4 ms (37.4 ms for a single geometry).

Table 1: NN sizes (neurons per layer) and MAE with R^2 of predicted energies (eV) and forces (eV Å^{-1}). The initial data set consists of 53,112 data points, and the size of the training set is 71,524 after the AL workflow with an additional 18,412 data points generated during active learning.

	With initial data		With AL data	
NN size	5,000	10,000	5,000	10,000 (on CPU)
Energy MAE (R^2)	0.0385 (0.9986)	0.0328 (0.9988)	0.1059(0.9907)	0.0356(0.9986)
Force MAE (R^2)	0.0507 (0.9979)	0.0515 (0.9978)	0.1443 (0.9839)	0.0483 (0.9981)

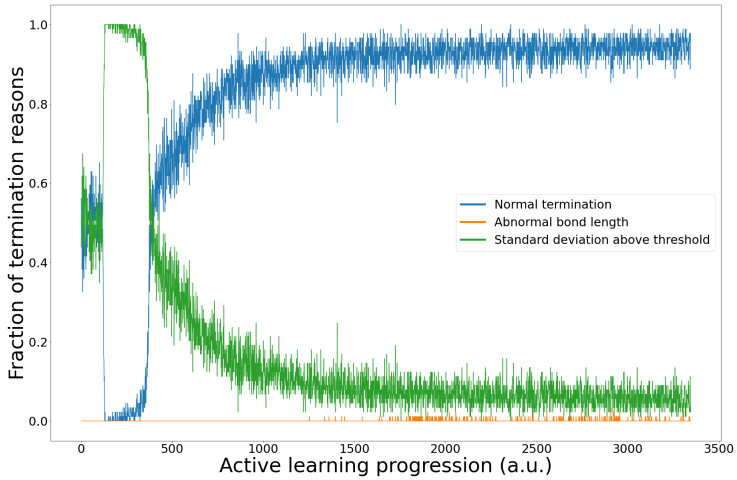


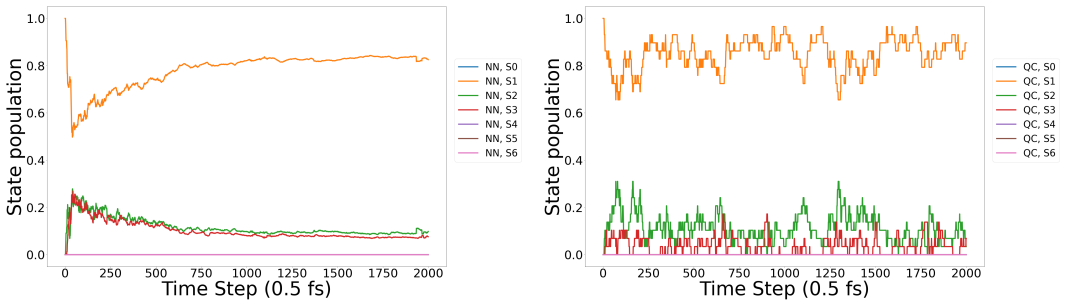
Figure 2: The time development of the trajectory termination reasons as a function of the progression of the active learning workflow. The termination events are counted for 3,345 trajectories for each of the 89 MD generators (297,705 trajectories in total).

3.2 Convergence analysis of the parallel active learning workflow

As the initial training data set was constructed from only a few ab initio MD trajectories, it is to be expected that it does not cover the relevant input space sufficiently well. To investigate the capacity of active learning to explore the conformational space outside of the initial training set distribution, we analyzed different events that led to terminations of MD trajectories in the generator kernel. As shown in Figure 2, initially only half of the trajectories terminated normally after reaching 2,000 steps at the beginning stage of active learning. This drastically changed as more of the input space was explored and most trajectories terminated early with standard deviations of the energy predictions exceeding the threshold. We then observed that the number of normally terminated trajectories slowly increased again and converged to almost 100% as the active learning process converged. This implies a strong change in the parameter distribution and thus robustness of the NNs due to a more general data distribution in the training set, even though only approximately 18,000 data points were added to the initial 53,000 data points. We refer to Section 5.8 of the appendix for additional analysis of the development of the NNs during active learning iterations. AL runs with smaller initial datasets are currently under investigation.

3.3 PES exploration of aryl sulfone oxide

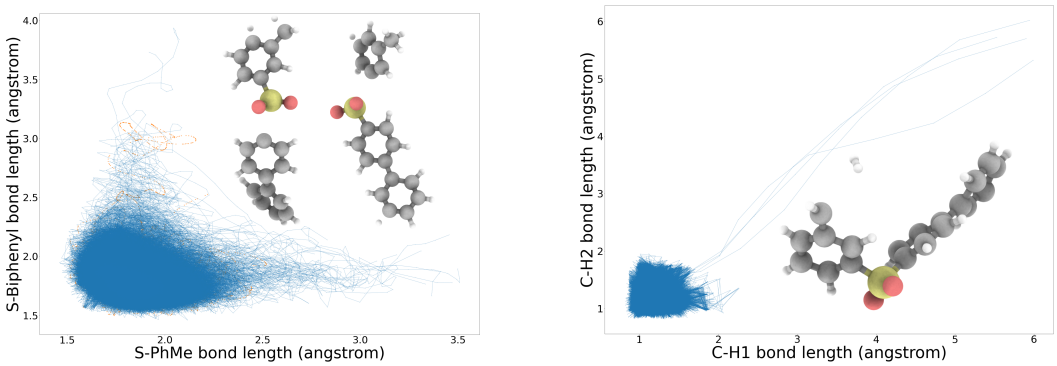
In order to validate the trained ML models, the electronic state distribution of trajectories resulting from ALMD and NAMD simulation were compared (see Figure 3 and the appendix). In Figure 3, the trajectories were initialized from the first excited state. Due to the larger amount of trajectories explored, ALMD trajectories show a smooth change of state population compared the the 29 available NAMD trajectories. The agreement between ML-predicted and QC-calculated state populations,



(a) ALMD trajectories (ours).

(b) NAMD trajectories (reference).

Figure 3: State populations for trajectories initialized at S_1. Populations average 25,810 ALMD trajectories and 29 NAMD trajectories.



(a) C-S bond length.

(b) C-H bond lengths (methyl group).

Figure 4: Abnormal bond lengths detected with ALMD.

especially after convergence at approximately 500 fs suggests sufficiently accurate energy and force predictions by the ML model. The remaining differences might be attributed to the difference between the ZNSH and the FSSH method.

Due to its speed, the ALMD method is able to uncover possible degradation pathways of **1** by tracking bond lengths of thousands of trajectories, potentially also over longer time scales. As shown in Figure 4a, trajectories with cleavage of C-S bonds were observed by ALMD, matching the results of previous studies in literature [24, 25]. In contrast to that, NAMD only captured the cleavage of one C-S bond (Figure 4a, orange trajectories). A further, unexpected reaction was detected with two hydrogens detached from the methyl group, forming a hydrogen molecule (Figure 4b). No correlation between C-S and C-H bond cleavage reactions was observed (see appendix).

4 Conclusion, limitation and ongoing work

In summary, we introduced a modular parallel active learning orchestrator that achieves independent execution of high-throughput MD simulations, QC calculations, and ML model training. The workflow efficiently generates training data to obtain ML models with high accuracy for energy and force prediction and rapidly explores the PES of a 38-atom organic molecule **1** with 6 excited electronic states. We were able to detect rare cleavage reactions of C-S bonds and C-H bonds, which potentially lead to the degradation of **1**. With the adaptable architecture, we expect this parallel active learning workflow to be readily extended to other application scenarios, including atomistic simulations and beyond.

One limitation of the current work includes the use of fully connected NN, which brings challenges when applied to other molecular systems. Furthermore, in this work, we consider the standard deviation of multiple neural networks as the only criterion for data selection, while the performance of the workflow could benefit from other heuristics such as molecular or geometrical similarity.

Besides, in this work, the generators were then restarted by sampling geometries from the initial NAMD data set. To enhance exploration, failed geometries could be buffered and revisited later by generators to explore spaces more frequently that are unfamiliar to the ML models. To test the limits of our workflow, we are currently running ALMD simulations of up to 10 ns (which takes approximately 2 weeks). For degradation reactions found by the workflow, we plan to investigate the relationship between the bond lengths and energies to identify, better understand, and potentially manipulate transition states by changing the molecular structure.

The active learning orchestrator is written in a modular and non-application-specific way. The kernels (oracle, training, prediction, generator) can be easily replaced and adapted for other scenarios and applications, within but also outside of atomistic simulations. Further use cases are being developed currently and will be published in the form of a customizable parallel active learning library.

References

[1] O Anatole von Lilienfeld. Introducing machine learning: Science and technology. *Machine Learning: Science and Technology*, 1(1):010201, feb 2020. doi: 10.1088/2632-2153/ab6d5d. URL https://dx.doi.org/10.1088/2632-2153/ab6d5d.

[2] Camille Bilodeau, Wengong Jin, Tommi Jaakkola, Regina Barzilay, and Klavs F. Jensen. Generative models for molecular discovery: Recent advances and challenges. *WIREs Computational Molecular Science*, 12(5):e1608, 2022. doi: https://doi.org/10.1002/wcms.1608. URL https://wires.onlinelibrary.wiley.com/doi/abs/10.1002/wcms.1608.

[3] Jingbai Li, Patrick Reiser, Benjamin R. Boswell, André Eberhard, Noah Z. Burns, Pascal Friederich, and Steven A. Lopez. Automatic discovery of photoisomerization mechanisms with nanosecond machine learning photodynamics simulations. *Chem. Sci.*, 12:5302–5314, 2021. doi: 10.1039/D0SC05610C. URL http://dx.doi.org/10.1039/D0SC05610C.

[4] Kevin Yang, Kyle Swanson, Wengong Jin, Connor Coley, Philipp Eiden, Hua Gao, Angel Guzman-Perez, Timothy Hopper, Brian Kelley, Miriam Mathea, Andrew Palmer, Volker Settels, Tommi Jaakkola, Klavs Jensen, and Regina Barzilay. Analyzing learned molecular representations for property prediction. *Journal of Chemical Information and Modeling*, 59(8):3370–3388, 2019. doi: 10.1021/acs.jcim.9b00237. URL https://doi.org/10.1021/acs.jcim.9b00237. PMID: 31361484.

[5] Jonas Busk, Peter Bjørn Jørgensen, Arghya Bhowmik, Mikkel N Schmidt, Ole Winther, and Tejs Vegge. Calibrated uncertainty for molecular property prediction using ensembles of message passing neural networks. *Machine Learning: Science and Technology*, 3(1):015012, dec 2021. doi: 10.1088/2632-2153/ac3eb3. URL https://dx.doi.org/10.1088/2632-2153/ac3eb3.

[6] Connor W Coley, Regina Barzilay, William H Green, Tommi S Jaakkola, and Klavs F Jensen. Convolutional embedding of attributed molecular graphs for physical property prediction. *Journal of chemical information and modeling*, 57(8):1757—1772, August 2017. ISSN 1549-9596. doi: 10.1021/acs.jcim.6b00601. URL https://dspace.mit.edu/bitstream/1721.1/116837/1/Coney%20Manuscript.pdf.

[7] Hanyu Gao, Thomas J. Struble, Connor W. Coley, Yuran Wang, William H. Green, and Klavs F. Jensen. Using machine learning to predict suitable conditions for organic reactions. *ACS Central Science*, 4(11):1465–1476, 2018. doi: 10.1021/acscentsci.8b00357. URL https://doi.org/10.1021/acscentsci.8b00357. PMID: 30555898.

[8] Zhenpeng Zhou, Xiaocheng Li, and Richard N. Zare. Optimizing chemical reactions with deep reinforcement learning. *ACS Central Science*, 3(12):1337–1344, 2017. doi: 10.1021/acscentsci.7b00492. URL https://doi.org/10.1021/acscentsci.7b00492. PMID: 29296675.

[9] Mikołaj Sacha, Mikołaj Błaż, Piotr Byrski, Paweł Dąbrowski-Tumański, Mikołaj Chromiński, Rafał Loska, Paweł Włodarczyk-Pruszyński, and Stanisław Jastrzębski. Molecule edit graph attention network: Modeling chemical reactions as sequences of graph edits. *Journal of Chemical Information and Modeling*, 61(7):3273–3284, 2021. doi: 10.1021/acs.jcim.1c00537. URL https://doi.org/10.1021/acs.jcim.1c00537. PMID: 34251814.

[10] Wengong Jin, Connor Coley, Regina Barzilay, and Tommi Jaakkola. Predicting organic reaction outcomes with weisfeiler-lehman network. *Advances in neural information processing systems*, 30, 2017.

[11] H. S. Seung, M. Opper, and H. Sompolinsky. Query by committee. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, COLT '92, page 287–294, New York, NY, USA, 1992. Association for Computing Machinery. ISBN 089791497X. doi: 10.1145/130385.130417. URL https://doi.org/10.1145/130385.130417.

[12] Burr Settles. Active learning literature survey. 2009.

[13] Michael Gastegger, Jörg Behler, and Philipp Marquetand. Machine learning molecular dynamics for the simulation of infrared spectra. *Chemical science*, 8(10):6924–6935, 2017.

[14] Julia Westermayr, Michael Gastegger, Maximilian FSJ Menger, Sebastian Mai, Leticia González, and Philipp Marquetand. Machine learning enables long time scale molecular photodynamics simulations. *Chemical science*, 10(35):8100–8107, 2019.

[15] Tom A Young, Tristan Johnston-Wood, Volker L Deringer, and Fernanda Duarte. A transferable active-learning strategy for reactive molecular force fields. *Chemical science*, 12(32):10944–10955, 2021.

[16] Shi Jun Ang, Wujie Wang, Daniel Schwalbe-Koda, Simon Axelrod, and Rafael Gómez-Bombarelli. Active learning accelerates ab initio molecular dynamics on reactive energy surfaces. *Chem*, 7(3):738–751, 2021.

[17] Tom A. Young, Joseph J. Silcock, Alistair J. Sterling, and Fernanda Duarte. autode: Automated calculation of reaction energy profiles— application to organic and organometallic reactions. *Angewandte Chemie International Edition*, 60(8):4266–4274, 2021. doi: https://doi.org/10.1002/anie.202011941. URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/anie.202011941`.

[18] Tom A Young, Tristan Johnston-Wood, Hanwen Zhang, and Fernanda Duarte. Reaction dynamics of diels–alder reactions from machine learned potentials. *Physical Chemistry Chemical Physics*, 24(35): 20820–20827, 2022.

[19] D W Walker. Standards for message-passing in a distributed memory environment. 8 1992. URL `https://www.osti.gov/biblio/7104668`.

[20] Lisandro Dalcin and Yao-Lung L. Fang. mpi4py: Status update after 12 years of development. *Computing in Science & Engineering*, 23(4):47–54, 2021. doi: 10.1109/MCSE.2021.3083216.

[21] Le Yu, Chao Xu, Yibo Lei, Chaoyuan Zhu, and Zhenyi Wen. Trajectory-based nonadiabatic molecular dynamics without calculating nonadiabatic coupling in the avoided crossing case: trans - cis photoisomerization in azobenzene. *Phys. Chem. Chem. Phys.*, 16:25883–25895, 2014. doi: 10.1039/C4CP03498H. URL `http://dx.doi.org/10.1039/C4CP03498H`.

[22] Mario Barbatti, Mattia Bondanza, Rachel Crespo-Otero, Baptiste Demoulin, Pavlo O. Dral, Giovanni Granucci, Fábris Kossoski, Hans Lischka, Benedetta Mennucci, Saikat Mukherjee, Marek Pederzoli, Maurizio Persico, Max Pinheiro Jr, Jiří Pittner, Felix Plasser, Eduarda Sangiogo Gil, and Ljiljana Stojanovic. Newton-x platform: New software developments for surface hopping and nuclear ensembles. *Journal of Chemical Theory and Computation*, 18(11):6851–6865, 2022. doi: 10.1021/acs.jctc.2c00804. URL `https://doi.org/10.1021/acs.jctc.2c00804`. PMID: 36194696.

[23] TURBOMOLE V7.7 2022, a development of University of Karlsruhe and Forschungszentrum Karlsruhe GmbH, 1989-2007, TURBOMOLE GmbH, since 2007; available from `https://www.turbomole.org`.

[24] Huifang Li, Minki Hong, Annabelle Scarpaci, Xuyang He, Chad Risko, John S. Sears, Stephen Barlow, Paul Winget, Seth R. Marder, Dongwook Kim, and Jean-Luc Brédas. Chemical stabilities of the lowest triplet state in aryl sulfones and aryl phosphine oxides relevant to oled applications. *Chemistry of Materials*, 31(5):1507–1519, 2019. doi: 10.1021/acs.chemmater.8b04235. URL `https://doi.org/10.1021/acs.chemmater.8b04235`.

[25] Na Lin, Juan Qiao, Lian Duan, Liduo Wang, and Yong Qiu. Molecular understanding of the chemical stability of organic materials for oleds: A comparative study on sulfonyl, phosphine-oxide, and carbonyl-containing host materials. *The Journal of Physical Chemistry C*, 118(14):7569–7578, 2014. doi: 10.1021/jp412614k. URL `https://doi.org/10.1021/jp412614k`.

[26] John C Tully. Molecular dynamics with electronic transitions. *The Journal of Chemical Physics*, 93(2): 1061–1071, 1990.

[27] Mario Barbatti. Nonadiabatic dynamics with trajectory surface hopping method. *Wiley Interdisciplinary Reviews: Computational Molecular Science*, 1(4):620–633, 2011.

[28] Simon Axelrod, Eugene Shakhnovich, and Rafael Gómez-Bombarelli. Excited state non-adiabatic dynamics of large photoswitchable molecules using a chemically transferable machine learning potential. *Nature communications*, 13(1):3440, 2022.

[29] Justin S. Smith, Ben Nebgen, Nicholas Lubbers, Olexandr Isayev, and Adrian E. Roitberg. Less is more: Sampling chemical space with active learning. *The Journal of Chemical Physics*, 148(24):241733, 05 2018. ISSN 0021-9606. doi: 10.1063/1.5023802. URL `https://doi.org/10.1063/1.5023802`.

5 Appendix

5.1 Distribution of energy and force data in the initial data set

As described in Section 5.3, in this work, the initial data set was generated with non-adiabatic molecular dynamics simulations (NAMD) that were started from S_1, S_5 and S_6 respectively. This led to an unbalanced distribution of energy (Table 2) and force (Table 3) data for different electronic states. As summarized in Table 2, the number of quantum calculation (QC) energy data of S_0 to S_3 was significantly larger than S_0 to S_5 and S_0 to S_6, as the number of corresponding trajectories was much larger in the initial data set. Besides, the number of force data of S_1 also greatly outperformed other states while there was no force data for S_0 (see Table 3) due to the NAMD setting that left out the coupling between S_0 and other states thus no relaxation to S_0. To address this issue, we employed a masking mechanism for the loss function explained in Section 5.5.

Table 2: Distribution of quantum calculated energies for different states in the initial data set.

States	S_0 - S_3	S_0 - S_5	S_0 - S_6
Energy data amount	43,598	10,837	16,380

Table 3: Distribution of quantum calculated forces for different states in the initial data set.

State	S_0	S_1	S_2	S_3	S_4	S_5	S_6
Force data amount	0	56,211	9,297	3,827	1,273	984	367

5.2 Oracle buffer updates to avoid redundancies in data

To ensure the efficiency of the workflow and to avoid labeling redundant data points, input candidates in the oracle buffer are evaluated by retrained NNs in the training kernel every time training is interrupted by the arrival of new data. The standard deviation of energy predictions is calculated and coordinates with standard deviation below the threshold are discarded from the oracle input buffer. The remaining atom coordinates are sorted according to prediction standard deviation with the most uncertain geometries being sent to oracle first in order to minimize the amount of costly DFT calculations. An MD trajectory propagated in the generator kernel is terminated if it runs into a molecular geometry with abnormal bond length or the standard deviation of the energy predictions exceeds the threshold.

5.3 Generator and oracle kernels

For active learning assisted molecular dynamics (ALMD) calculations incorporated in the generator kernel of our parallel active learning workflow, we applied Zhu–Nakamura theory of surface hopping (ZNSH) [21] instead of the more widely used Fewest Switches Surface Hopping (FSSH) method[26, 27], as the non-adiabatic couplings are challenging for ML prediction [3, 28]. ZNSH estimates the probability of surface hopping based on energies only and has been successfully applied to excited-state dynamics studies.

5.4 Initial training data and reference calculations

ALMD is initialized by sampling geometries from NAMD trajectories of the initial data set. FSSH was used for initial NAMD calculations that resulted in 29 trajectories initialized in the first excited state (S_1), 11 trajectories initialize in the fifth excited state and 8 trajectories initialized in the sixth excited state. Each trajectory was simulated for a duration of 1 picosecond, employing a time-step of 0.5 femtoseconds. 94,419 data points were collected from all trajectories with atom coordinates, forces for corresponding states, i.e. S_0 to S_3, S_0 to S_5, and S_0 to S_6.

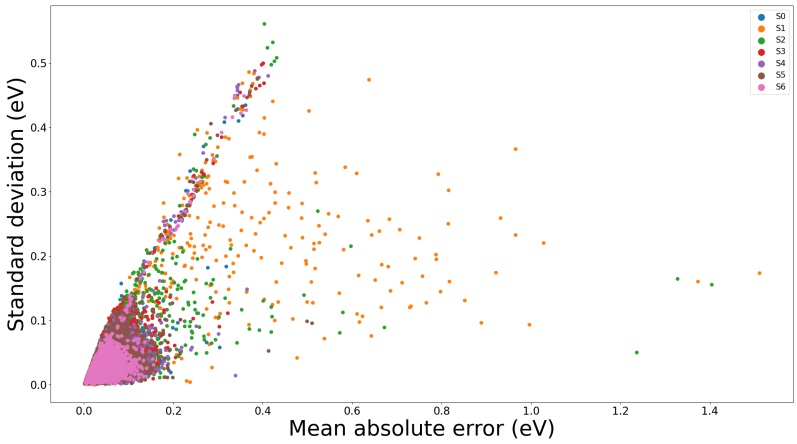


Figure 5: Standard deviation vs. mean absolute error of energy predictions. Results are shown for 23,605 data points that were drawn out of the initial data set, separated from the initial training set, and used as the test set.

5.5 Training and prediction kernels

As our use case aims at high accuracy and speed, rather than generalizability to other molecules, we use a fully connected neural network (NN) inspired by prior work[14, 3]. The NN is trained to predict energies using a combined energy and force mean squared error loss function, with forces trained as derivatives of energies. To handle the incomplete energy and force data for some electronic states in the initial training set, we incorporate a masking mechanism in the loss function to leave out missing labels during the training process. The NN takes as the input an inverse distance-based feature representation of $\mathbf{1}$, which consists of 703 features resulting from $N = 38$ atoms. There are 7 electronic states (ground state and 6 excited states; k = 6) that lead to 7 energy values and $k \times 3N = 798$ force components. The NN models consist of 6 layers with e.g. 5,000 neurons per layer and a softplus activation function. The training process is carried out with the Adam optimizer. For initial training, we adopted an exponential decrease of learning rate from 10^{-6} over 1000 epochs, while the learning rate was fixed to 10^{-7} during the active learning iterations. The NN models were implemented using TensorFlow/Keras (v2.10).

5.6 MD step timing and communication overhead

The most time-critical element of the active learning approach is the generator kernel, which is closely linked with the prediction kernel, with communications happening multiple times per second. We recorded the time required by different components of the ALMD simulation and found the bottleneck being the energy and force predictions in the prediction kernel, which took on average 51.4 ms per prediction and thus per MD step, in comparison with the MPI communications and trajectory propagation that required 9.1 ms. The total time to propagate one MD step for 89 trajectories adds up to 60.5 ms. Removal of the oracle- and training kernels did not affect this result, indicating that additional communication and data processing does not reduce the performance of the rate-limiting step.

5.7 Standard deviation vs. model error

The standard deviation vs. corresponding mean absolute error (MAE) of energy predictions on the test set of the initial distribution is plotted in Figure 5, indicating a positive correlation between the two metrics. This finding matches both the results from previous studies [29] as well as the hypothesis of query-by-committee strategy [12] to estimate model errors through prediction uncertainty evaluations.

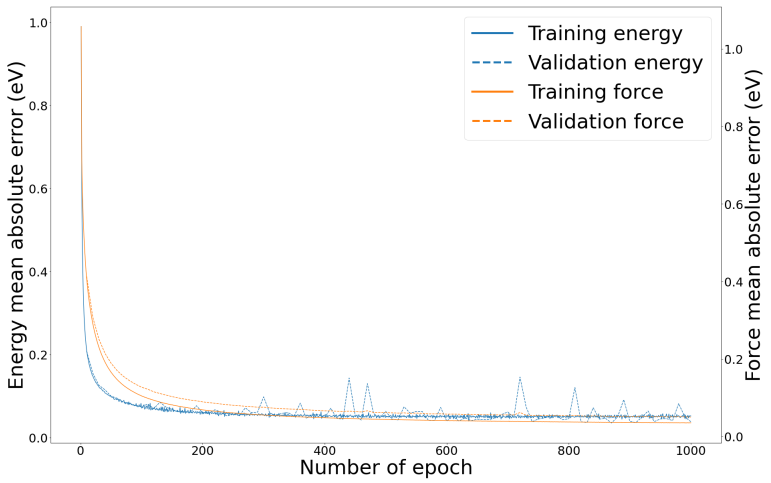


Figure 6: Energy and force training curve for NN model with 5,000 neurons.

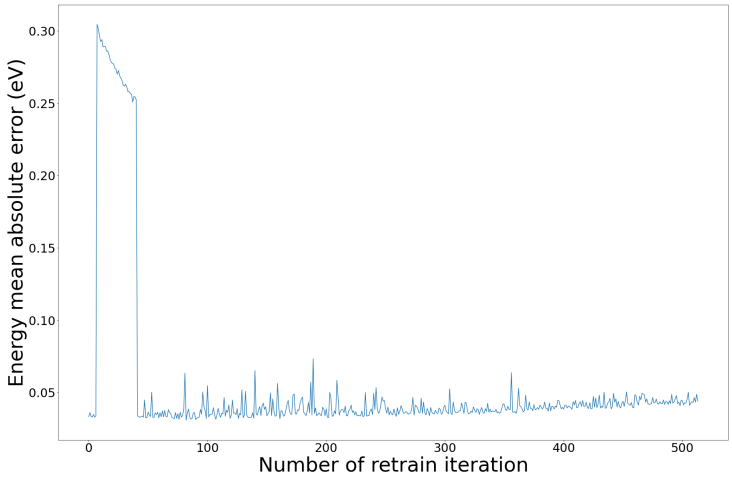


Figure 7: Energy prediction MAE of NNs with 5,000 neurons on a fixed test set after each retrain iteration. The test set with a size of 23,605 was drawn from the initial dataset and separated from the initial training.

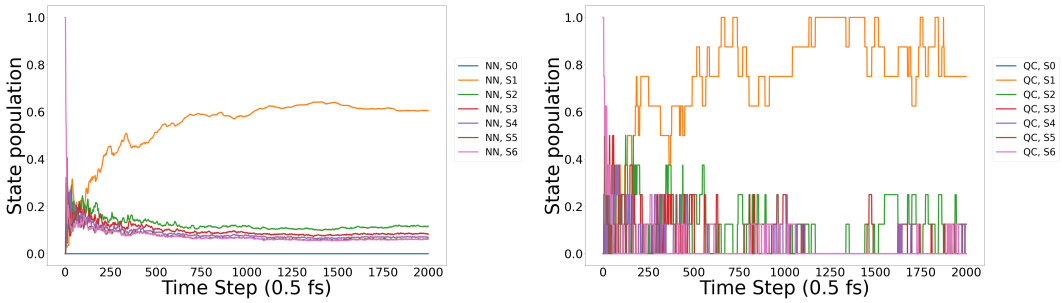
5.8 History of the initial training and testing results of NNs during active learning

Training and validation mean absolute error metrics for energy and force are summarized in Figure 6, which demonstrate the capacity of NNs with 5,000 neurons to fit the energy and force data without overfitting.

Figure 7 displays testing results of NNs with 5,000 neurons after each training iteration on the energy test set from the initial data set (separated from the initial training set). The noteworthy change in the MAE, especially the increase in the beginning of the active learning iterations, suggests a shift in data distribution during the active learning workflow. After a sufficient amount of new training data is accumulated, the trained neural networks seem to substantially change their parameter distribution, leading again to much lower mean absolute errors. As the training dataset distribution puts more emphasis on newly explored parts of the conformational space, the mean absolute error on a fixed test set from the initial distribution then slightly increases over time.

5.9 ALMD for aryl sulfone oxide

Figure 8 serves as a supplement of Section 3.3. The results of trajectories initialized from S_6 displayed good agreement between ML predictions and QC calculations, as discussed in this work.



(a) ALMD trajectories.

(b) NAMD trajectories.

Figure 8: State populations for trajectories initialized at S_6. Populations average 153,220 ALMD trajectories and 8 NAMD trajectories.