Distributed Reinforcement Learning for Molecular Design: Antioxidant case

Huanyi Qin[1], Denis Akhiyarov[2], Sophie Loehle[2], Kenneth Chiu[1], and Mauricio Araya-Polo[2]

[1]Department of Computer Science, SUNY Binghamton {hqin4, kchiu}@binghamton.edu
[2]TotalEnergies {denis.akhiyarov, sophie.loehle, mauricio.araya}@totalenergies.com

Abstract

Deep reinforcement learning has successfully been applied for molecular discovery as shown by the Molecule Deep Q-network (MolDQN) algorithm. This algorithm has challenges when applied to optimizing new molecules: training such a model is limited in terms of scalability to larger datasets and the trained model cannot be generalized to different molecules in the same dataset. In this paper, a distributed reinforcement learning algorithm for antioxidants, called DA-MolDQN is proposed to address these problems. State-of-the-art bond dissociation energy (BDE) and ionization potential (IP) predictors are integrated into DA-MolDQN, which are critical chemical properties while optimizing antioxidants. Training time is reduced by algorithmic improvements for molecular modifications. The algorithm is distributed, scalable for up to 512 molecules, and generalizes the model to a diverse set of molecules. The proposed models are trained with a proprietary antioxidant dataset. The results have been reproduced with both proprietary and public datasets. The proposed molecules have been validated with DFT simulations and a subset of them confirmed in public "unseen" datasets. In summary, DA-MolDQN is up to 100x faster than previous algorithms and can discover new optimized molecules from proprietary and public antioxidants.

1 Introduction

Antioxidants are compounds that inhibit oxidation and are critical in a variety of industrial applications, such as fuel additives [34], lubricants [1], and polymer stabilization [24]. They can prevent oxidation and degradation, improving the performance and longevity of materials and products. Traditional antioxidant discovery approaches are time-consuming and costly, because these approaches typically require time-consuming iterative approaches from chemical researchers and rely on expensive Density Functional Theory (DFT) [22] calculations. Reinforcement learning (RL) algorithms [23, 5, 31, 17] are often used to optimize and generate new molecules based on their chemical properties, such as quantitative estimation of drug-likeness (QED) or synthesizability (SA) scores.

However, new challenges arise while using RL to optimize antioxidants. Bond dissociation energy (BDE) and ionization potential (IP) indicate the reactivity and stability of the antioxidant molecules, respectively [18, 15]. Unfortunately, the estimation of BDE and IP is more demanding than QED and SA scores because DFT calculation is borderline prohibitive— it takes hours or even days to calculate a single BDE value while using DFT [37]. In this paper, Alfabet [30] and AIMNet-NSE [40] machine learning models are integrated to predict and estimate the BDE and IP properties of antioxidants, which are state-of-the-art BDE and IP predictors based on deep learning.

Another challenge of antioxidant properties optimization is that the Pareto optimization front between BDE and IP is often a trade-off. Molecules with good BDE properties usually have poor IP properties and vice versa. By balancing the weight of BDE and IP, the proposed RL agent is able to optimize

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

both BDE and IP properties of antioxidants, so that the optimized molecules have stronger antioxidant properties and are more stable.

MolDQN [38] algorithm is a well-known RL model for optimizing molecules. Subsequently, multi-task (MT)-MolDQN was developed [9], which has a major advantage over the original MolDQN algorithm in that it is able to train the model with multiple initial molecules simultaneously. Their optimizations start with one or several initial molecules, but their explorations are restricted to the neighborhood of initial molecules, which prevents them from exploring larger antioxidant datasets and proposing well-optimized antioxidants. To address this, the distributed DA-MolDQN was proposed, which is capable of training and optimizing hundreds of antioxidants simultaneously.

In this paper, extra methods and optimizations are also necessary to further improve the performance and efficiency. They include: (1) additional protection mechanism of O-H bond, (2) an efficient method that avoids invalid 3D conformers for new molecules, (3) improved C++ RL environment ported from original Python implementation, (4) incremental Morgan fingerprint algorithm, (5) a BDE property cache, (6) a filtering algorithm to constrain the search space, (7) and the optional fine-tuned steps for further optimizing outlier molecules.

Specifically, we make the following contributions:

- DA-MolDQN integrates state-of-the-art BDE and IP predictors.
- The molecules optimized by DA-MolDQN have both good BDE and IP, which have strong antioxidant properties and are stable.
- The model trained by DA-MolDQN is generalized and can optimize antioxidants on both proprietary and public data sets.
- DA-MolDQN is efficient and is 2.6 - 106x faster than MT-MolDQN and MolDQN.

2 Background & Related Work

2.1 Chemical Properties of Antioxidants

BDE Property of O-H bond BDE is used to measure the strength of a chemical bond. BDE of the O-H bond is one of the most important characteristics of an antioxidant [8, 2, 18, 36]. In this paper, BDE refers to the lowest BDE values of all O-H bonds in the molecule. The lower the BDE, the easier it is for the antioxidant to yield its hydrogen atom, and the more promising the compound is. For the antioxidant to be effective, the BDE of the antioxidant must be well below the BDE of peroxide radicals (88-90 kcal/mol) [7]. Specifically in this work, the BDE of optimized antioxidants should be lower than 76 kcal/mol, which are thought to have good BDE properties.

IP Property IP is the energy required to remove an electron from an atom. It tells how tightly the electron is bound and how stable the antioxidant is. If the IP is too low, the compound becomes unstable in air and undergoes natural oxidation with the dioxygen present in the air, losing its antioxidant activity and making it useless for antioxidants. The IP of generated antioxidants should be higher than 145 kcal/mol.

Trade-off between BDE and IP The BDE can be efficiently lowered by incorporating electron-donor substituents on the ring such as methyl, methoxy, tertiobutyl, or hydroxide, preferably in ortho and para position. It allows stabilization of the formed radical, leading to a lower BDE. However, it's not possible to stack five dimethyl amino groups on the ring and hope to find the best antioxidant ever made. This doesn't work because introducing electron-donating substituents lowers both the BDE and the IP [18]. As mentioned above, when IP is too low, the compound becomes unstable and useless. There is a trade-off between BDE and IP, and the RL agent needs to balance the BDE and IP so that the generated antioxidants are effective and stable.

2.2 State-of-the-art Property Predictors

BDE Predictor: Alfabet Alfabet is based on graph neural networks (GNNs) and has been shown to be accurate in predicting BDE for a wide range of molecules. When using Alfabet to predict the BDE of generated molecules, it accepts SMILES representation of molecules as input. Then the lowest BDE is found among all O-H bonds and it is termed as BDE in this paper.

IP Predictor: AIMNet-NSE AIMNet-NSE is also a machine learning architecture that can predict molecular energies including IP, and it achieves high accuracy from training with over 100k molecules. The AIMNet-NSE uses the 3D conformer of molecules to predict IP properties, which brings new challenges, because the molecules generated by MolDQN algorithm may not have a valid and stable 3D conformer.

Both predictors have high accuracy on the proprietary antioxidant data set. Their average relative error is less than 5%. Although their performance is much faster than DFT, they are still the main bottlenecks for RL optimization. Their performance is further improved by introducing a cache and reducing the needed 3D conformers.

2.3 MolDQN & MT-MolDQN

MolDQN is an innovative deep reinforcement learning model that treats molecules as undirected graphs, optimizing the structure of molecules by adding or removing new atoms and bonds. The modifications restart from the initial molecules in every episode and the molecule is slightly optimized in each step. So the explorations are around the initial molecules. Each modification will take care of the valence of different atoms, and the modified molecules are only guaranteed to be valid in 2D graph representations. The modified molecules may not be valid in 3D space and have valid 3D conformers because these modifications do not consider the 3D structures, such as the torsion angles [27] and aromatic ring. After each modification, the modified molecules are inputted into property predictors, then their rewards are calculated. The molecules and other needed information are stored as samples to train the model.

MolDQN has achieved excellent results on both single or multiple objective optimization tasks to find molecules with better or specific properties, such as maximizing penalty logP values and QED. The following MT-MolDQN is the parallel version and it is implemented with Pytorch Distributed Data-Parallel (DDP) [16]. Compared to MolDQN, MT-MolDQN proposes better-optimized molecules and is more efficient, because more initial molecules can be used in training.

3 Distributed Antioxidant Optimizer: DA-MolDQN

Figure 1 illustrates the workflow of DA-MolDQN, providing an overview of the training process for the general model. The flowchart commences with an introduction to distributed processes, followed by a depiction of the optimization and evaluation of molecules within these processes. The flowchart also includes various other elements, encompassing the integration of property predictors, distributed training, reward function, filter script, and fine-tuning. This section elaborates on the intricacies of the workflow and also presents performance optimizations.

3.1 Distributed Overview & Details in Process

Distributed Overview The MT-MolDQN uses Pytorch DDP as a framework, and DA-MolDQN extends it to a distributed training script, so the DA-MolDQN is no longer limited by the computational resources within a node. The distributed processes (workers) are launched by SLURM [14] and each process will receive one initial molecule. The processes will work on optimizing their initial molecules independently, but they will cooperate to propose a well-trained and generalized model.

Details in Process: Step by Step Optimization The molecule is modified and optimized by the RL agent in each process. This actually generates an optimization path to the final proposed molecule, and the initial molecule is optimized step by step. In short, one step of a molecule includes: generating all valid action molecules based on chemical valence, choosing an action by decaying epsilon greedy method [32], and predicting the properties. The details are explained in Section 2.3 and the MolDQN paper [38].

Details in Process: Batched Modification Batched modifications (blue parallelograms in Figure 1) are needed so that each process can process multiple initial molecules, which in turn promote parallelism. Each process receives multiple initial molecules while launching. In each step, the batched RL algorithm will operate on all molecules in that process one by one. It will not go to the next step until all molecules in the current step finished their operations. By using batched modification,

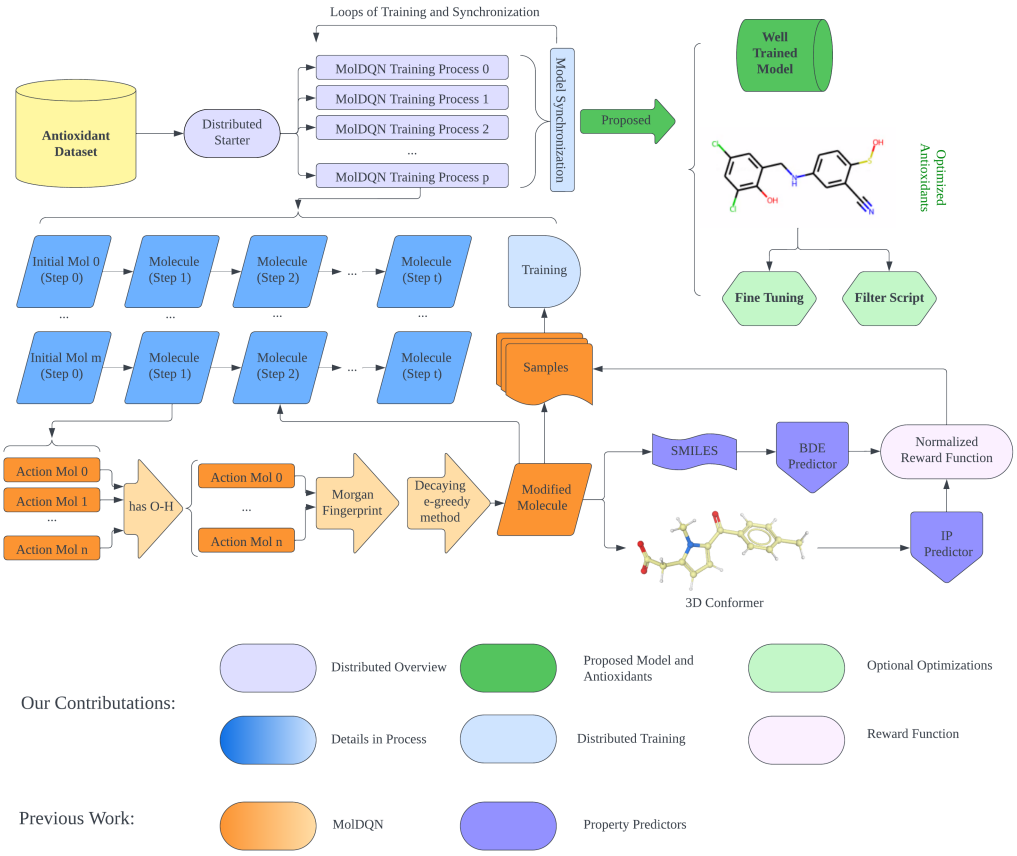

Figure 1: Workflow of DA-MolDQN

the number of initial molecules further increases, and diverse samples could be generated by the RL agent. The diversity of samples is crucial to train a general model. The batched modification also reduces the required computational resources because the multiple molecules in a process share the expensive BDE and IP property predictors.

3.2 Distributed Training

Each process has a replay buffer that stores all modified molecules and their useful information, such as fingerprints [28], and rewards. The samples from the replay buffer are reused to train the model. The losses are collected from all processes during training. Then, the general model is synchronized among all processes at the end of every episode, so the experience learned from other processes will be utilized in the next episodes.

3.3 Extra Efforts on Integrating Property Predictors

BDE: Protecting O-H Bond Since the BDE property is the lowest BDE among all O-H bonds, an implicit restriction is that the generated molecules must have at least one O-H bond, otherwise, the BDE properties are undefined and the molecules are not needed. This restriction is not guaranteed in the MolDQN and MT-MolDQN, because the O-H bond may be accidentally broken by the modifications. These modifications are not allowed, and the molecules always have at least one O-H bond in this paper. The approach removes only a few invalid modifications (e.g. Action Mol 1 in Figure 1) from over a hundred action molecules, which has a trivial impact on the exploration of molecular space. The examples are presented in Appendix A.

IP: Avoiding Invalid 3D Conformers AIMNet-NSE uses the 3D conformers of the molecules as its input. The RDKit [26] is used to calculate the 3D coordinates of all atoms in the molecules [4]. However, as mentioned in Section 2.3, some generated molecules may not have a valid 3D conformer. The MolDQN has done some work on this, such as limiting the size of new rings, but their efforts are still far from being complete. In order to avoid generating molecules without any valid 3D conformations, the reward of invalid molecules is set to -1000, which is much less than the normal rewards (0.8-2.5). Then RL agent successfully learns how to avoid these invalid conformers, without employing a huge number of human handwritten chemistry rules. An example of an invalid molecule and the results of the avoiding approach are shown in Appendix B

3.4 Normalized Reward Function

As mentioned in Section 2, the reward function needs to combine the normalized BDE and IP properties. In this paper, min-max normalization is used for both BDE and IP. The lower bound and upper bound are minimal and maximum properties in the proprietary data set. The weights of the normalized rewards are also useful to further balance the BDE and IP, depending on the optimization target. They are set to 0.8 and 0.2 in this paper because the BDE property is related to the effectiveness of antioxidants and is more important than IP. Different combinations of reward weights are also tested and the above weights are found to be optimal.

The molecules with fewer atoms and bonds are preferred when seeking new antioxidants, so the γ is also added to Equation 1. It represents the relatively reduced atoms and bonds from the initial molecule.

$$\text{Reward} = -w_1 \cdot \text{nBDE} + w_2 \cdot \text{nIP} + w_3 \cdot \gamma \tag{1}$$

3.5 Optional Optimizations

Filter Script Although the antioxidants are significantly optimized by the general model, the molecules are not 100% guaranteed to fit the BDE and IP constraints. Further, there are several other properties that chemists are interested in, such as Tanimoto similarity [3] of fingerprints and SA score. To address this, an extra script filters out molecules without good BDE and IP properties. The molecules are also filtered out if their SA scores are higher than 3.5 or if they are identical to existing antioxidants.

Fine-Tuning The experiences that a general model learns from the most antioxidant optimizations may prevent the RL agent from doing some unusual but effective optimizations for the outlier molecules. Inspired by [39, 12], a few fine-tuning episodes could significantly improve the performance of a model in special environments. An optional fine-tuning step is introduced to the workflow. The fine-tuning starts with the pre-trained general model, and the initial epsilon threshold is 0.5. Fine-tuning is independent for each molecule and the properties of irregular molecules are further improved with trivial overhead.

3.6 Non-trivial Performance Improvement

General Performance Optimization The MolDQN and MT-MolDQN are profiled with py-spy [25] and the result demonstrates that the molecule modification and Morgan fingerprint [28] calculation are the main performance bottlenecks of MT-MolDQN. To address this, the modification functions are re-implemented in C++ instead of Python, and a fast incremental Morgan fingerprint algorithm is developed. With the help of the above optimizations, DA-MolDQN is 2.6x faster than MT-MolDQN while using QED and SA score rewards.

Property Predictor Performance Optimization Although Alfabet and AIMNet-NSE are much faster than the traditional DFT method, they are still 466.8x and 32.6x slower than QED calculation. The estimated computation time without any performance optimization will be over 16 days. To address this, a Least Recently Used (LRU) [21] cache is introduced to store the predicted BDE values. The AIMNet-NSE proposed 5 trained models and suggested using the average predicted properties. However, only one model is used in this paper and the accuracy is still good enough. The performance of predictors is significantly improved during training and fine-tuning.

4 Experiments & Results

4.1 Antioxidant Optimization Experiment

The models in Table 1 are trained on a random subset of 256 antioxidants that are from a proprietary data set [20] of over 500 antioxidant molecules.

Model	Initial Mols/Model	Trained Models	Episodes	Nodes	Modification Batch
Individual	1	256	8000	1	1
Parallel	8	32	8000	1	1
General	256	1	250	4	4
Fine-Tuned	1	256	200	1	1

Table 1: This table summarizes the number of initial molecules, computation resources, and episodes to train the models. Each node has 4 $\times$ Tesla A100 GPUs. The fine-tuning process starts with the pre-trained general model and needs 200 extra episodes. More parameters are in Appendix C.

Summary of Expected Properties The optimized molecules are expected to have the following properties or constraints: (A) lower BDE than the original antioxidants ($<$ 76 kcal/mol). (B) higher IP than the original antioxidants ($>$ 145 kcal/mol). (C) fewer atoms and bonds. (D) similar but not identical to the original antioxidants. (E) low SA score. A, B, and C are satisfied in the reward function (Equation 1).The extra filter script in Section 3.5 is needed for D and E.

Optimization Failure Rate An optimization is successful if the BDE of the generated molecule is less than 76 kcal/mol and the IP is greater than 145 kcal/mol. Otherwise, the optimization fails and training resources are wasted. The optimization failure rate (OFR) is defined in Equation 2 where S is the number of successful optimizations and A is the total number of optimizations.

$$\text{OFR} = 1 - S/A \tag{2}$$

4.2 Optimization Results of Different Models

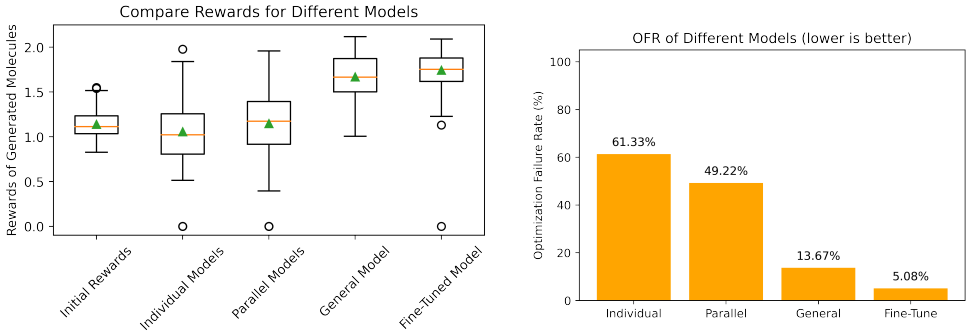

Figure 2: Left: This figure compares the rewards of antioxidants optimized by 4 different models. The rewards of a few molecules are out of range (0.0 - 3.0) because of invalid 3D conformers. Their rewards are set to 0. Right: The figure compares the OFR of different models.

Figure 2 shows the rewards of antioxidants optimized by the individual, parallel, general, and fine-tuned models. In the left figure, the rewards of individuals and parallel models are not improved compared to initial molecules. 61.33% and 49.22% of their optimizations are failed. The rewards of the general model are significantly higher than the rewards of initial molecules, individual models, and parallel models, and the OFR is also much lower than the individual models and parallel models. For the fine-tuned models, the rewards are only slightly improved, and the OFR of fine-tuned models is further reduced. The results indicate that most antioxidants are successfully optimized by the general model and fine-tuned models. In Figure 3, when the models are trained with more and more fine-tuned episodes, the fine-tuned models are eventually individual models. The results show that 100 or 200 extra episodes are enough for the fine-tuned models to optimize the irregular antioxidants.

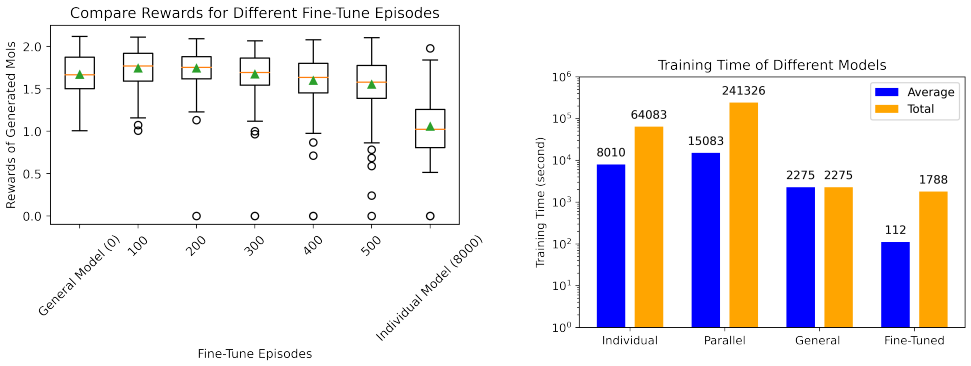

Figure 3: The left figure compares rewards of different fine-tune episodes. The right figure shows the computation time for training each model. The episodes of each model are in Table 1. The blue bars are the average training time per model and the orange bars are the total computation time to train all models while using all 4 nodes.

Computation Time of Training and Fine-tuning In Figure 3, the computation time of the general model is 3.5x and 6.6x faster than individual models and parallel models, while the general model is more powerful in optimizing antioxidants. Note that it needs only 1 general model to optimize the 256 training molecules, but it needs 256 individual models and 32 parallel models to optimize all 256 molecules. Although the 4 nodes with 16 A100 GPUs could train 16 individual models or 4 parallel models simultaneously, their optimizations of 256 molecules are still 28.1x and 106x slower than the general model. Figure 3 also shows that the extra computation time of fine-tuning is trivial compared to training an initial model from scratch.

4.3 Optimization of Unseen Proprietary Antioxidants

128 testing molecules are randomly selected from the rest antioxidants and are optimized by the trained models. 16 independent models and 16 parallel models are also randomly picked from the 256 and 32 trained models. These models, the general model, and the fine-tuning models are used to optimize all 128 testing antioxidants. Their optimized rewards and OFR are shown in Figure 4. Individual models and parallel models are not able to optimize unseen molecules. The general model still outperforms individual and parallel models, but the effect of fine-tuning is more significant for optimizing unseen molecules. The results show that the proposed DA-MolDQN model could still well optimize the unseen antioxidants, under the help of 200 fine-tuning episodes.

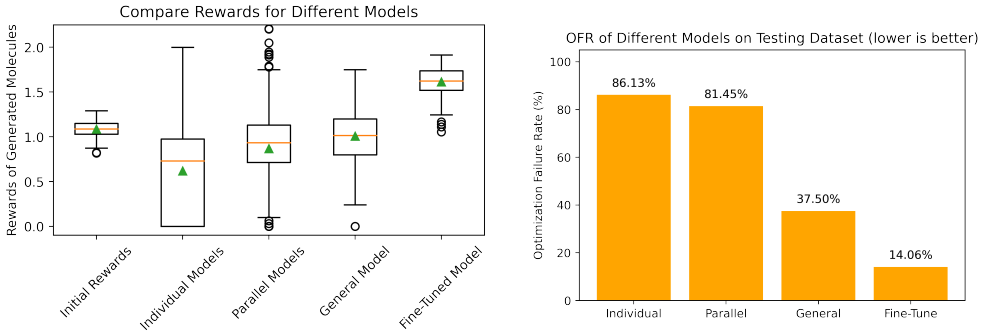

Figure 4: Left: This figure compares the rewards of 128 test antioxidants optimized by 4 different models. Right: The figure compares the OFR of the test antioxidants.

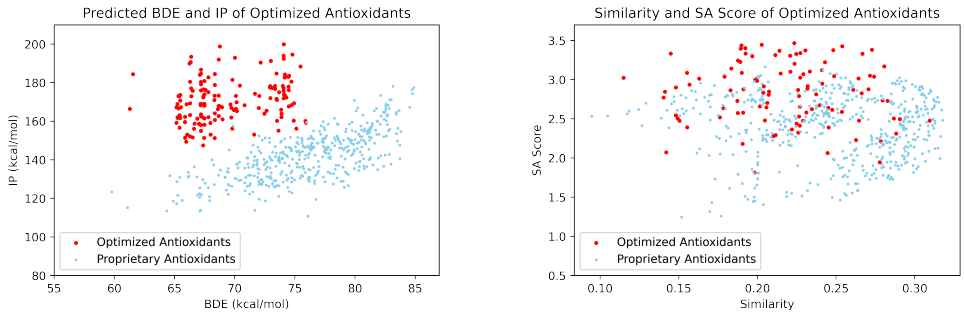

Figure 5: Left: The predicted BDE and IP of optimized molecules (red) are better than the initial molecules (blue). Right: The Similarity and SA Score of generated molecules (red) are close to the initial molecules (blue).

4.4 Proposed Antioxidants

Optimized BDE and IP Properties Figure 5 compares the initial molecules and the molecules generated by the trained DA-MolDQN model. The molecules are filtered out if the properties required in Section 4.1 are not satisfied. The figure shows that the generated molecules have significantly lower BDE and higher IP, compared to all initial molecules in the antioxidant data set. The proposed molecules have both good BDE and IP properties, indicating that the optimizer successfully balances and optimizes these chemical properties.

Similarity and SA Score Because similarity and SA score are not included in the RL reward function, there are some optimized molecules that have SA scores greater than 3.5. These optimizations are still counted as successful optimizations, even though they are unlikely to be good antioxidant candidates in practice. The distribution of the remaining optimized molecules in Figure 5 is close to the original antioxidants, which is exactly what the chemist expected.

Example of Proposed Antioxidant and Generating Path The molecules in the proprietary antioxidant data set and the molecules generated from them are commercially confidential. The experiments are replayed for some public molecules in the ChEMBL [6] and AODB [11] data sets. An example of the optimization path is shown in Figure 6. More molecules are shown in Appendix E.

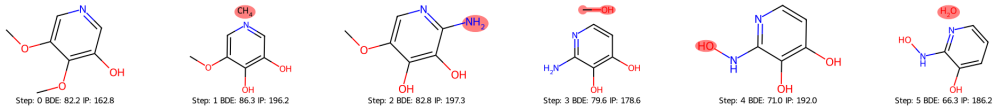

Figure 6: The figure shows one optimized antioxidant and its generating path. The molecules in Step 0 are the initial molecules and the optimized molecules are in Step 5. The modifications are highlighted and the unconnected atoms are removed from the current molecules.

5 Validation of ML Results with DFT Simulations

5.1 DFT validation

The geometry of both initial molecules and optimized molecules, their radical structures, and their radical cation structures were calculated by means of DFT, which is implemented in the Gaussian 16 [13] computer program using B3LYP hybrid functional with *6-311++G(d,p)* basis set in Polarizable Continuum Model of solvation [10] where the solvent, Toluene in the present work, is described by its dielectric constant.

The following equations are used to calculate BDE and IP on optimized structure [33, 19]:

$$BDE = H(R^{\bullet}) + H(H^{\bullet})\breve{}H(R-H) \tag{3}$$

$$IP = H(RH^{+\bullet}) + H(e-) - H(R-H) \tag{4}$$

Where $H(R^{\bullet})$ is the enthalpy of the radical formed after abstraction of the hydrogen from the phenol group and after geometry optimization of the structure. $H(H^{\bullet})$ is the enthalpy of a single H atom. $H(R-H)$ is the enthalpy of the neutral molecule. $H(RH^{+\bullet})$ is the enthalpy of the radical cation formed when the most excited electron is abstracted and $H(e-)$ is the enthalpy of an electron. Enthalpy of the hydrogen atom is $-312,44$ kcal/mol [19] and enthalpy of the electron is $-55,61$ kcal/mol [33].

7 proposed molecules are selected from the antioxidants optimized by the parallel models for evaluation with DFT simulations. The DFT results show that the proposed molecules have significantly improved properties and are close to the predicted values. The table below shows the comparison of predicted and DFT results, for BDE and IP. The results indicate that the error is within tolerance for optimizing these molecules. The details of DFT results are in Appendix F

Figure 7 shows the classification of the proposed 7 molecules to their stability and performance both with DFT and DA-MolDQN algorithms. The comparison shows that 5 of out 7 molecules match the classification for stability and performance.

Figure 7: The figure shows the optimized antioxidants ordered by their properties. The DFT and ML properties are generated by DFT simulation and the machine learning predictors.

6 Discussion

The distributed training algorithm and the batched modification trained the model with many more initial molecules, compared to the MolDQN and MT-MolDQN. The general model explores a wider range of the molecular universe, resulting in the RL agent learning from various candidate antioxidants. The experiment results of both training and testing data sets show that the DA-MolDQN achieved significantly better optimization ability than the previous work. The effectiveness of DA-MolDQN was proved by the DFT validation results and the fact that we did find some of the discovered antioxidants in PubChem and AODB datasets. The general models with 512 and 1024 initial molecules are also tested, there is no significant improvement compared to the model with 256 molecules.

By speeding up the modifications, fingerprinting, and property predictions, the training performance of DA-MolDQN is greatly improved and the distributed script could optimize 256 antioxidants within an hour, instead of taking 16 days to optimize 1 molecule. The distributed training framework of DA-MolDQN can be used to optimize various drug-like molecules by introducing other property predictors. The performance optimization methods could also be helpful to other applications.

7 Conclusions & Future Work

In this paper, state-of-the-art BDE and IP predictors were integrated and the training performance was greatly improved. The molecules proposed by DA-MolDQN have similarity and synthesizability scores close to the original molecules while their BDE and IP properties are improved, and the results are validated by DFT and public datasets. In the future, more optimization will be applied so that the general model for 256-1024 molecules can be trained using less computational resources. Another approach is to find a method which can avoid 100% invalid 3D conformers without unnecessary RL actions. Additional DFT validations are underway.

References

[1] Z. X. Alimova, N. A. Kholikova, S. O. Kholova, and K. G. Karimova. Influence of the antioxidant properties of lubricants on the wear of agricultural machinery parts. *IOP Conference Series: Earth and Environmental Science*, 868(1):012037, oct 2021. doi: 10.1088/1755-1315/868/1/012037. URL https://dx.doi.org/10.1088/1755-1315/868/1/012037.

[2] N. A. Amran, U. Bello, and M. S. H. Ruslan. The role of antioxidants in improving biodiesel's oxidative stability, poor cold flow properties, and the effects of the duo on engine performance: A review. *Heliyon*, page e09846, 2022.

[3] D. Bajusz, A. Rácz, and K. Héberger. Why is tanimoto index an appropriate choice for fingerprint-based similarity calculations? *Journal of cheminformatics*, 7(1):1–13, 2015.

[4] J. M. Blaney and J. S. Dixon. *Distance Geometry in Molecular Modeling*, pages 299–335. John Wiley & Sons, Ltd, 1994. ISBN 9780470125823. doi: https://doi.org/10.1002/9780470125823.ch6. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/9780470125823.ch6.

[5] J. Born, M. Manica, A. Oskooei, J. Cadow, G. Markert, and M. Rodríguez Martínez. Paccmannrl: De novo generation of hit-like anticancer molecules from transcriptomic data via reinforcement learning. *iScience*, 24(4):102269, 2021. ISSN 2589-0042. doi: https://doi.org/10.1016/j.isci.2021.102269. URL https://www.sciencedirect.com/science/article/pii/S2589004221002376.

[6] ChEMBL. "https://www.ebi.ac.uk/chembl/", 2022.

[7] G. da Silva, C.-C. Chen, and J. W. Bozzelli. Bond dissociation energy of the phenol oh bond from ab initio calculations. *Chemical Physics Letters*, 424(1):42–45, 2006. ISSN 0009-2614. doi: https://doi.org/10.1016/j.cplett.2006.04.022. URL https://www.sciencedirect.com/science/article/pii/S0009261406004982.

[8] C. Daguet-Schott. New methods to characterize fuels' oxidation and screening of novel antioxidants. Master's thesis, Department of Chemistry Technical University of Denmark.

[9] Z. DAI, D. Akhiyarov, R. Alami, D. Pantano, M. Araya-Polo, and C. Pereira. Multi-task deep reinforcement learning for molecular optimization. *ELLIS Machine Learning for Molecule Discovery Workshop*, 2019.

[10] M. M. Dehkordi, M. H. Asgarshamsi, A. Fassihi, and K. K. Zborowski. A comparative dft study on the antioxidant activity of some novel 3-hydroxypyridine-4-one derivatives. *Chemistry & Biodiversity*, 19(3):e202100703, 2022.

[11] W. Deng, Y. Chen, X. Sun, and L. Wang. Aodb: A comprehensive database for antioxidants including small molecules, peptides and proteins. *Food Chemistry*, 418:135992, 2023. ISSN 0308-8146. doi: https://doi.org/10.1016/j.foodchem.2023.135992. URL https://www.sciencedirect.com/science/article/pii/S030881462300609X.

[12] A. Fickinger, H. Hu, B. Amos, S. Russell, and N. Brown. Scalable online planning via reinforcement learning fine-tuning. *Advances in Neural Information Processing Systems*, 34:16951–16963, 2021.

[13] M. J. Frisch, G. W. Trucks, H. B. Schlegel, G. E. Scuseria, M. A. Robb, J. R. Cheeseman, G. Scalmani, V. Barone, G. A. Petersson, H. Nakatsuji, X. Li, M. Caricato, A. V. Marenich, J. Bloino, B. G. Janesko, R. Gomperts, B. Mennucci, H. P. Hratchian, J. V. Ortiz, A. F. Izmaylov, J. L. Sonnenberg, D. Williams-Young, F. Ding, F. Lipparini, F. Egidi, J. Goings, B. Peng, A. Petrone, T. Henderson, D. Ranasinghe, V. G. Zakrzewski, J. Gao, N. Rega, G. Zheng, W. Liang, M. Hada, M. Ehara, K. Toyota, R. Fukuda, J. Hasegawa, M. Ishida, T. Nakajima, Y. Honda, O. Kitao, H. Nakai, T. Vreven, K. Throssell, J. A. Montgomery, Jr., J. E. Peralta, F. Ogliaro, M. J. Bearpark, J. J. Heyd, E. N. Brothers, K. N. Kudin, V. N. Staroverov, T. A. Keith, R. Kobayashi, J. Normand, K. Raghavachari, A. P. Rendell, J. C. Burant, S. S. Iyengar, J. Tomasi, M. Cossi, J. M. Millam, M. Klene, C. Adamo, R. Cammi, J. W. Ochterski, R. L. Martin, K. Morokuma, O. Farkas, J. B. Foresman, and D. J. Fox. Gaussian~16 Revision C.01, 2016. Gaussian Inc. Wallingford CT.

[14] M. Jette, C. Dunlap, J. Garlick, and M. Grondona. Slurm: Simple linux utility for resource management. 7 2002. URL https://www.osti.gov/biblio/15002962.

[15] D. M. Kasote, S. S. Katyare, M. V. Hegde, and H. Bae. Significance of antioxidant potential of plants and its relevance to therapeutic applications. *International journal of biological sciences*, 11(8):982, 2015.

[16] S. Li, Y. Zhao, R. Varma, O. Salpekar, P. Noordhuis, T. Li, A. Paszke, J. Smith, B. Vaughan, P. Damania, et al. Pytorch distributed: Experiences on accelerating data parallel training. *arXiv preprint arXiv:2006.15704*, 2020.

[17] S. A. Meldgaard, J. Köhler, H. L. Mortensen, M.-P. V. Christiansen, F. Noé, and B. Hammer. Generating stable molecules using imitation and reinforcement learning. *Machine Learning: Science and Technology*, 3(1):015008, dec 2021. doi: 10.1088/2632-2153/ac3eb4. URL https://dx.doi.org/10.1088/2632-2153/ac3eb4.

[18] A. Mohajeri and S. S. Asemani. Theoretical investigation on antioxidant activity of vitamins and phenolic acids for designing a novel antioxidant. *Journal of Molecular Structure*, 930(1): 15–20, 2009. ISSN 0022-2860. doi: https://doi.org/10.1016/j.molstruc.2009.04.031. URL https://www.sciencedirect.com/science/article/pii/S002228600900252X.

[19] A. Mohajeri and S. S. Asemani. Theoretical investigation on antioxidant activity of vitamins and phenolic acids for designing a novel antioxidant. *Journal of Molecular Structure*, 930(1-3): 15–20, 2009.

[20] C. Moussa, H. Wang, M. Araya-Polo, T. Back, and V. Dunjko. Application of quantum-inspired generative models to small molecular datasets. *ArXiv*, abs/2304.10867, 2023. URL https://api.semanticscholar.org/CorpusID:258291519.

[21] E. J. O'Neil, P. E. O'Neil, and G. Weikum. The lru-k page replacement algorithm for database disk buffering. *SIGMOD Rec.*, 22(2):297–306, jun 1993. ISSN 0163-5808. doi: 10.1145/170036.170081. URL https://doi.org/10.1145/170036.170081.

[22] R. G. Parr, S. R. Gadre, and L. J. Bartolotti. Local density functional theory of atoms and molecules. *Proceedings of the National Academy of Sciences*, 76(6):2522–2526, 1979.

[23] M. Popova, O. Isayev, and A. Tropsha. Deep reinforcement learning for de novo drug design. *Science Advances*, 4(7):eaap7885, 2018. doi: 10.1126/sciadv.aap7885. URL https://www.science.org/doi/abs/10.1126/sciadv.aap7885.

[24] J. Pospíšil. Exploitation of the current knowledge of antioxidant mechanisms for efficient polymer stabilization. *Polymers for Advanced Technologies*, 3(8):443–455, 1992. doi: https://doi.org/10.1002/pat.1992.220030805. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/pat.1992.220030805.

[25] py spy. "https://github.com/benfred/py-spy", 2022.

[26] Rdkit. "http://www.rdkit.org/", "https://github.com/rdkit/rdkit", 2022.

[27] S. Riniker and G. A. Landrum. Better informed distance geometry: using what we know to improve conformation generation. *Journal of chemical information and modeling*, 55(12): 2562–2574, 2015.

[28] D. Rogers and M. Hahn. Extended-connectivity fingerprints. *Journal of chemical information and modeling*, 50(5):742–754, 2010.

[29] C. Shi, M. Xu, Z. Zhu, W. Zhang, M. Zhang, and J. Tang. Graphaf: a flow-based autoregressive model for molecular graph generation, 2020.

[30] P. C. St. John, Y. Guan, Y. Kim, S. Kim, and R. S. Paton. Prediction of organic homolytic bond dissociation enthalpies at near chemical accuracy with sub-second computational cost. *Nature communications*, 11(1):2328, 2020.

[31] N. Ståhl, G. Falkman, A. Karlsson, G. Mathiason, and J. Boström. Deep reinforcement learning for multiparameter optimization in de novo drug design. *Journal of Chemical Information and Modeling*, 59(7):3166–3176, 2019. doi: 10.1021/acs.jcim.9b00325. URL https://doi.org/10.1021/acs.jcim.9b00325. PMID: 31273995.

[32] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, second edition, 2018. URL http://incompleteideas.net/book/the-book-2nd.html.

[33] M. H. Vakarelska-Popovska and Z. Velkov. Monohydroxy flavones. part iv: Ehthalpies of different ways of o–h bond dissociation. *Computational and Theoretical Chemistry*, 1077: 87–91, 2016.

[34] K. Varatharajan and D. Pushparani. Screening of antioxidant additives for biodiesel fuels. *Renewable and Sustainable Energy Reviews*, 82:2017–2028, 2018. ISSN 1364-0321. doi: https://doi.org/10.1016/j.rser.2017.07.020. URL https://www.sciencedirect.com/science/article/pii/S1364032117310870.

[35] J. You, B. Liu, Z. Ying, V. Pande, and J. Leskovec. Graph convolutional policy network for goal-directed molecular graph generation. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/d60678e8f2ba9c540798ebbde31177e8-Paper.pdf.

[36] H.-Y. Zhang and L.-F. Wang. Theoretical elucidation on structure–antioxidant activity relationships for indolinonic hydroxylamines. *Bioorganic & Medicinal Chemistry Letters*, 12(2): 225–227, 2002. ISSN 0960-894X. doi: https://doi.org/10.1016/S0960-894X(01)00724-7. URL https://www.sciencedirect.com/science/article/pii/S0960894X01007247.

[37] H.-Y. Zhang, Y.-M. Sun, and D.-Z. Chen. O–h bond dissociation energies of phenolic compounds are determined by field/inductive effect or resonance effect? a dft study and its implication. *Quantitative Structure-Activity Relationships*, 20(2):148–152, 2001.

[38] Z. Zhou, S. Kearnes, L. Li, R. N. Zare, and P. Riley. Optimization of molecules via deep reinforcement learning. *Scientific reports*, 9(1):1–10, 2019.

[39] D. M. Ziegler, N. Stiennon, J. Wu, T. B. Brown, A. Radford, D. Amodei, P. Christiano, and G. Irving. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*, 2019.

[40] R. Zubatyuk, J. S. Smith, B. T. Nebgen, S. Tretiak, and O. Isayev. Teaching a neural network to attach and detach electrons from molecules. *Nature Communications*, 12(1):4870, 2021.

Appendix A Examples of Action Molecules

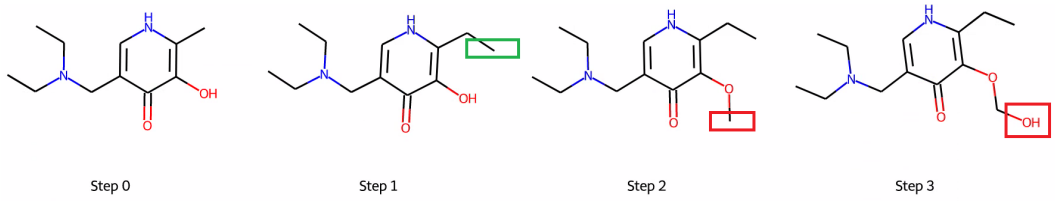

Figure 8: Examples of valid modification (green rectangle) and invalid modifications (red rectangles). Each molecule has added an atom to the previous molecule on the left. The modifications that break the last O-H bond in step 2 are invalid. The atom addition in step 3 is also invalid because it happens after the invalid atom addition in step 2.

Appendix B Examples of Invalid 3D conformers & Results of Avoiding Method

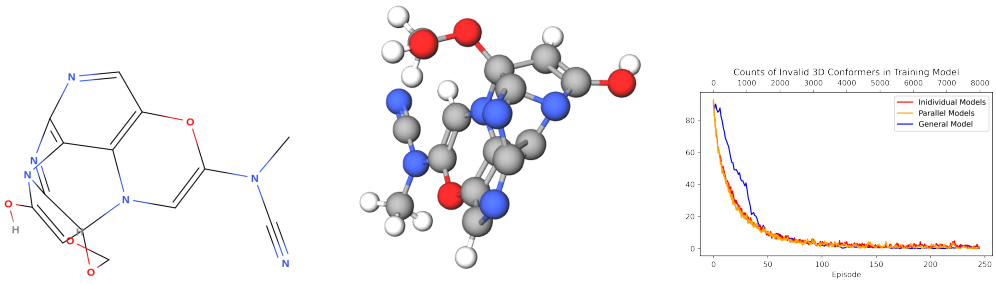

Figure 9: Left: The 2D representation of the generated molecule with invalid 3D conformers. Middle: An example of the invalid 3D conformer. Right: The RL agent learns to avoid invalid 3D conformers for individual models, parallel models, and general model. The individual and parallel models were trained with 8000 episodes and the general model was trained with 250 episodes.

Appendix C Experiments Parameters

Model	Initial Epsilon	Epsilon Decay	Max Training Batch Size	Starter
Individual	1.0	0.999	128	Torchrun
Parallel	1.0	0.999	128	Torchrun
General	1.0	0.970	512	Slurm
Fine-Tuned	0.5	0.961	128	Torchrun

Table 2: This table summarizes additional experiment parameters that are **different** among models.

Max Steps/Episodes	10	Allowed Atoms	C, O, N
Update Episodes	1	Allowed Rings	3, 5, 6
DDP Backend	gloo	Fingerprint Radius	3
Replay Buffer Size	4000	Fingerprint Length	2048
BDE Weight	0.8	BDE Factor	0.9
IP Weight	0.2	IP Factor	0.8
γ Weight	0.5	Optimizer	Adam
Discount Factor	1.0	Learning Rate	1e-4

Table 3: This table summarizes the experiment parameters that are the **same** among models.

Appendix D Comparison with Related Works

	QED			Penalized LogP		
	1st	2nd	3rd	1st	2nd	3rd
MolDQN	0.948	0.944	0.943	11.84	11.84	11.82
DA-MolDQN	0.948	0.948	0.947	7.12	7.07	6.94
GCPN	0.948	0.948	0.947	6.56	6.46	6.40
GraphAF	0.948	0.947	0.947	5.63	5.60	5.44

Table 4: The top 3 molecules are collected from both training and testing. The QED properties of top molecules are similar among the 4 models. The Plogp of MolDQN significantly outperforms DA-MolDQN and other algorithms because the PlogP is maximized by simply adding carbon atoms. As a result, the generated molecules are obviously not drug-like [38].

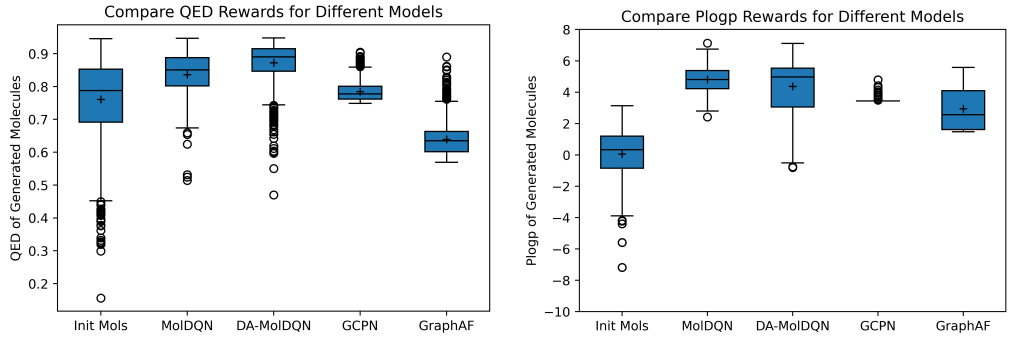

Figure 10: The figure shows the molecule QED and PlogP rewards for MolDQN, DA-MolDQN, GCPN, and GraphAF. 256 molecules are optimized by MolDQN, while 1024 molecules are generated by the DA-MolDQN fine-tuned models. For GCPN and GraphAF, the molecules are generated in the same wall clock time and only show the top 1024 molecules.

In the comparison experiments, the QED and penalized logP (PlogP) rewards were tested. The results were compared with GCPN [35] and GraphAF [29], which are state-of-the-art molecule generation algorithms based on GNN. All models are trained with the Zinc250k data set. 256 and 1024 molecules

are randomly selected to train for MolDQN and DA-MolDQN. The training parameters are the same as antioxidant experiments. GCPN and GraphAF are trained with the whole Zinc250k data set. The models are trained with (1-10) epochs, and the best results are present. The results indicate that our DA-MolDQN achieves similar performance in optimizing top molecules, generates molecules with higher QEDs and avoids failures in PlogP optimization.

Appendix E More Examples of Proposed Molecules & Generating Path

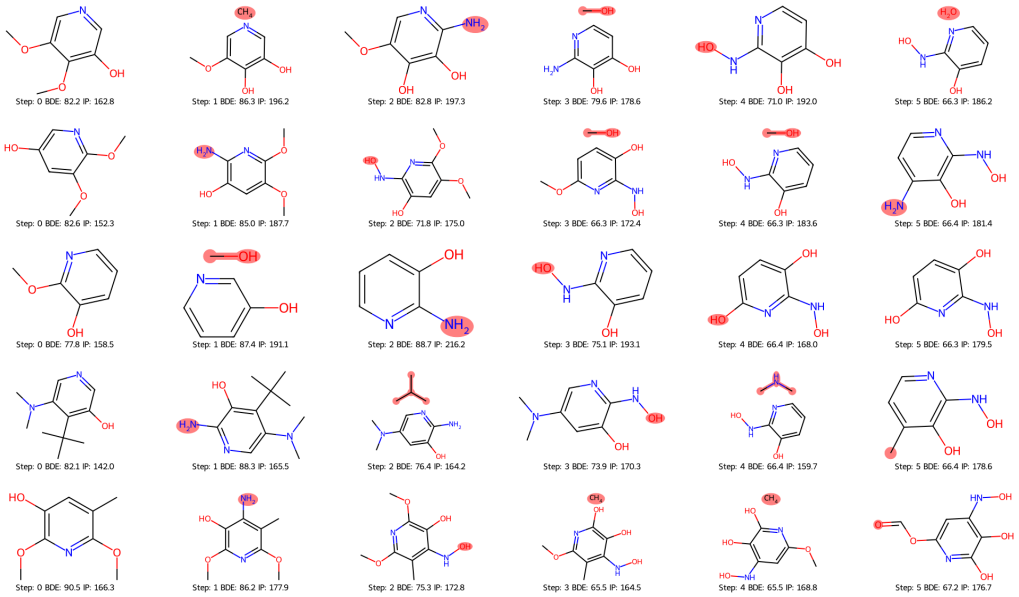

Figure 11: The figure shows several optimized antioxidants and their generating paths. The molecules in Step 0 are the initial molecules and the optimized molecules are in Step 5. The modifications are highlighted and the unconnected atoms are removed from the current molecules.

Appendix F Result of DFT Validation

No.	Initial BDE	BDE_{ML}	BDE_{DFT}	Initial IP	IP_{ML}	IP_{DFT}	Similarity	SA Score
1	76.9	72.4	77.91	139.2	164.6	168.4	0.13	2.49
2	76.9	74.4	76.04	139.2	155.9	147.87	0.12	2.53
3	67.1	75.7	81.19	113.8	174.3	187.99	0.18	2.83
4	67.1	73.0	77.82	113.8	166.9	163.86	0.18	2.80
5	64.3	72.5	77.61	113.4	161.5	166.05	0.19	2.79
6	64.3	69.7	74.85	113.4	156.8	150.65	0.18	2.88
7	64.3	67,8	71.53	113.4	158.1	141.45	0.19	2.52

Table 5: The table shows the DFT validation results of new antioxidants generated by parallel models. The initial BDE and IP are generated by DFT. BDE_{ML} and IP_{ML} are predicted by the property predictors.