Machine Learning to Accelerate Screening for Marcus Reorganization Energies

Omri D. Abarbanel[1] and Geoffrey R. Hutchison[1, 2, a)]

[1)] *Department of Chemistry, University of Pittsburgh, 219 Parkman Avenue, Pittsburgh, Pennsylvania 15260, United States*

[2)] *Department of Chemical and Petroleum Engineering, University of Pittsburgh, 3700 O'Hara Street, Pittsburgh, Pennsylvania 15261, United States*

(Dated: 7 March 2021)

Understanding and predicting the charge transport properties of π-conjugated materials is an important challenge for designing new organic electronic applications, including solar cells, plastic transistors, light-emitting devices, and chemical sensors. A key component of the hopping mechanism of charge transfer in these materials is the Marcus reorganization energy, which serves as an activation barrier to hole or electron transfer. While modern density functional methods have proven to accurately predict trends in reorganization energy, such calculations are computationally expensive. In this work, we outline active machine learning methods to predict computed intramolecular reorganization energies of a wide range of polythiophenes and their use towards screening new compounds with low reorganization energies. Our models have an overall root mean square error of $\pm$0.113 eV, but a much smaller RMSE of only 0.036 eV on the new screening set. Since the larger error derives from high-reorganization energy compounds, the new method is highly effective to screen for compounds with potentially efficient charge transport parameters.

PACS numbers: 72.20.Jv, 31.15.A, 72.80.Le, 36.20.Kd

I. INTRODUCTION

Polythiphenes are a class of π-conjugated conductive and semi-conductive organic materials which can be used in many electronic devices, such as field-effect transistors[1,2], organic solar cells[3–5], chemical sensors[6–10], and more[11]. The electronic properties of polythiophenes can be tuned across a wide range by various synthetic substitutions of the parent thiophene ring, which has enabled both fundamental studies and many applications.

The vast majority of polythiophenes derivatives are p-type, with the charge transfer mediated by a hole transfer process[12–14]. Marcus-Hush charge transfer theory shows that the internal reorganization energy (λ), which describes the energy change required to distort geometry upon a charge transfer, is one important factor in the charge transfer rate and resulting charge mobility[12–14].

The internal reorganization energy λ of molecules undergoing hole transfer can be calculated from four energies - the energy of the neutral molecule in the lowest energy geometry (E_0, "Neutral"), the energy of the cation at its lowest energy geometry (E_+, "Cation"), the energy of the cation at the geometry of the neutral species (E_+^*, "Cation@Neutral"), and the energy of the neutral molecule at the geometry of the cation (E_0^*, "Neutral@Cation")[12,14] (Figure 1). The λ can then be calculated from those energies according to the following formula 1:

$$\lambda = \lambda_0 + \lambda_+ = (E_0^* - E_0) + (E_+^* - E_+) \quad (1)$$

[a)] Electronic mail: geoffh@pitt.edu

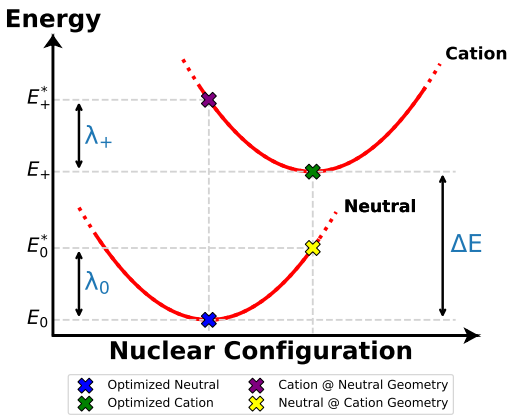


FIG. 1: Internal reorganization energy for hole transfer.

Calculating the λ of polythiophenes using density functional theory (DFT) calculations requires two geometry optimizations (of both the neutral and cationic species) and can be computationally expensive, as the calculation time increases with the length the polythiophene chain. Recent work on the approximate density functional GFN2 method[15] has shown accurate geometries and excellent correlation with coupled-cluster methods for conformers[16]. We attempted to correlate reorganization energies computed with GFN2 with those computed with the B3LYP DFT method[17,18]. As discussed below, no significant correlation was found.

In this work, we instead focus on predicting internal reorganization energies λ using machine learning (ML) methods, using a minimal amount of B3LYP-calculated λ as a training set.

In recent years ML has been applied widely, with a goal of accelerating quantum chemical calculations that would otherwise have large computational costs. Calculating electronic properties with traditional methods can be computationally expensive and take between hours to weeks to finish, depending on the size of the system and the type of calculation. ML has shown a great potential in calculating electronic structure properties, drug discovery, materials research, and more[19–25]. Training a ML model is also time consuming as well, since it requires a large data set for training and finding an accurate ML method and representation for that specific application can be exhaustive, but once a model has been properly trained, evaluation for new calculations can be performed in seconds or less.

In this work, we have developed a machine learning filter for predicting the internal reorganization energy of organic electronic materials. At present, we find the accuracy to be greater for compounds with low reorganization energy - as such it proves more useful for ignoring compounds expected to have high barriers for charge transport than as a fully accurate surrogate across the entire range considered. Nevertheless, since key applications require efficient charge transport, and thus low reorganization energies, we demonstrate its use in efficiently screening a pool of possible copolymers. Finally, we discuss frequent chemical motifs among compounds with low predicted reorganization energies.

II. METHODS

A. Computational Methods

Input files for each oligomer were created by combining the corresponding SMILES strings of its monomers (Table S1) and using OpenBabel version 3.1.0[26] to generate a 3D geometry.[27] All GFN2 calculations were performed using xTB version 6.0[15]. All DFT calculations were done using the B3LYP functional[17,18] with the 6-31G* basis set,[28] calculated with Gaussian 09,[29] for comparison with previously published internal reorganiztion energies.

Random forest and gradient boosted trees models were implemented using Scikit-Learn version 0.20.0[30]. Neural network model was implemented using Keras version 2.3.1[31] with TensorFlow version 2.1.0[32] backend.

All Python code and notebooks are pro-

vided in the Supporting Information and at `https://github.com/hutchisonlab/ReorganizationEnergy`

B. Data Set

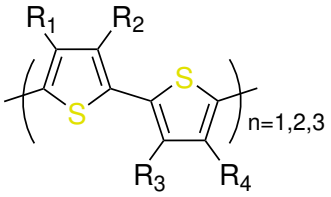


FIG. 2: Thiophene based oligomers, length of 2, 4, and 6 monomers - named dimers, tetramers, and hexamers, respectively.

Our data set consists of 253 thiophene-based monomers (Table S1). The monomers had different functional groups at the 3 and 4 positions, while connected to other monomers at the 2 and 5 positions (Figure 2), yielding a total of 31,878 possible copolymers, plus 253 homopolymers, to a total of 32,131 possible oligomer families. We used our available monomers to create a list of possible oligomers made from two, four, and six monomers - dimers, tetramers, and hexamers, respectively (Figure 2).

Calculating the λ of long oligomer chains using traditional DFT methods can be time-consuming and computationally expensive. However, previous studies have claimed that six-membered oligomer chains can closely estimate the λ of longer chains[12]. However, quantum mechanical calculations, especially when optimizing molecular geometries, drastically increase with the length of the oligomer (Figure 3a). We therefore have explored different ways to minimize the calculation time, such as using a different computational method, GFN2, and using shorter oligomers.

At first, we calculated the λ of all the oligomers using GFN2-xTB, an approximate density functional tight-binding method developed by the Grimme group[15], to see if it can be used as an accurate surrogate for B3LYP-computed reorganization energies. This method is produces accurate geometries, and is considerably faster than B3LYP calculations (Figures S7 and 3a). However, the λ calculated using GFN2-xTB does not correlate well with the λ calculated using B3LYP (Figure S2).

In contrast, while the energies have little correlation, we have found that the geometries of both the neutral and cation species calculated using GFN2-xTB did had a significant correlation with those calculated using B3LYP. Specifically, the average dihedral angle and the average inter-ring bond length. Therefore, instead of using GFN2-xTB to calculate the λ, we considered using ML methods using the geometrical descriptors obtained from the GFN2-xTB calculations. Likely, while the geometric minima correlate well between GFN2 and B3LYP, the shape of the potential energy surfaces differ substantially away from the local minima.

In addition, we considered a correlation of λ between shorter and longer oligomers, since shorter oligomers are faster to optimize (Figure S1). We did not find such a correlation between the B3LYP-computed λ of the dimer and tetramers, or the dimers and hexamers. We did find, however, a good correlation between the tetramers and hexamers (Figure 3b). Thus, to develop an adequate training set, more tetramers than hexamers can be used, considerably reducing the calculation time. We therefore used a data set made up of mainly tetramers plus a small number of hexamers. We increased the training set in batches until we saw no significant improvement in the model score (Figure 3c) and decrease in RMSE. Our final data set consisted of 7020 tetramers and 408 hexamers with B3LYP-calculated λ between 0 and 2 eV. We chose this range as we assumed that oligomers with λ larger than 2 eV are irrelevant to our study.

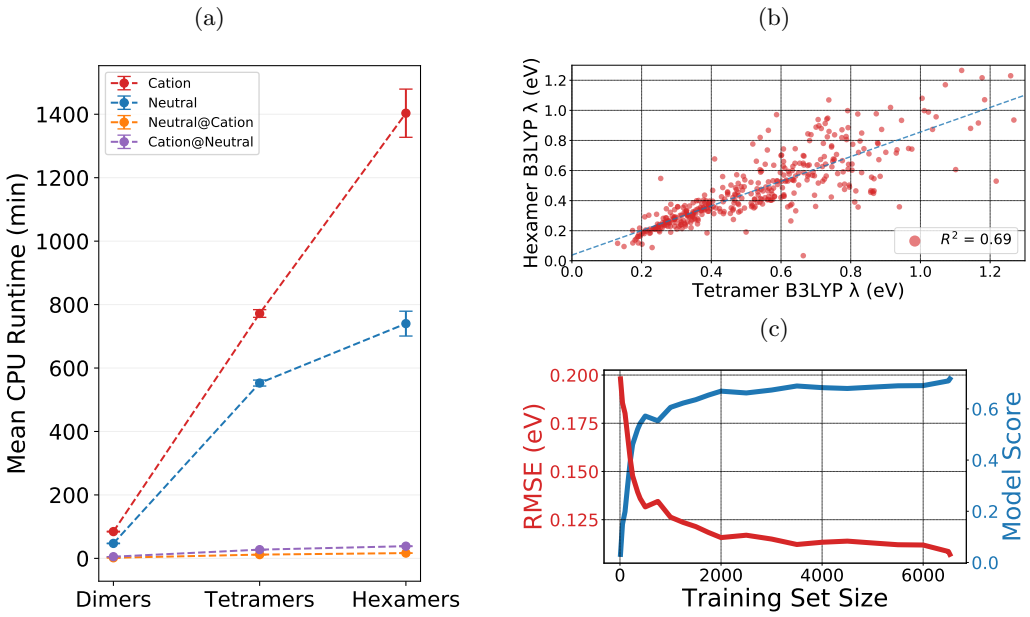


FIG. 3: **(a)** Mean CPU run time of the 4 different calculations using B3LYP, **(b)** Correlation between tetramers λ and hexamers λ, trendline indicated robust linear regression fit, **(c)** training set size effect on the RF model score.

C. Representation and Model Selection

For the representation of the oligomers in the ML model we began with the monomer id and the oligomer length. In addition, we added the average dihedral angle between the monomers of each oligomer, and the average inter-ring bond length of each oligomer, for both the neutral and cation species of each oligomer, as calculated using GFN2. We saw correlation (e.g., R^2 between 0.57 and 0.74) between those geometric values calculated with GFN2 and with B3LYP (Figure S3). Using those starting features gave us decent preliminary results. Next, we added an extended circular finger print (ECFP4) 2048 bit representation[33] using RDKit[34], which increased the R^2 and decreased the RMSE of the model significantly, likely by describing local functional group effects on reorganization energies. The final step was to add a new feature to represent the size of the π-system in the oligomers (Figure S10), as we hypothesized that a highly-conjugated oligomer will contribute to a lower λ. Adding this final feature moderately improved the model (Table I).

Using our best representation, we trained three different ML models with our data set: a random forest model, a gradient boosting trees model, and a neural network model. The first two are ensemble methods based on decision trees, which combine a several weighted trees into one model. The random forest model builds a large number of random sets of decision trees[35], hence the name, while the gradient boosted trees model builds nested decision tree one at a time, improving over the previous tree[36]. The third ML model, a neural network, has been widely used in many classification and regression applications. Neural networks are built in layers, where each node in each layer is connected to all the nodes in the next layer.

	Geometrical Data		Geometrical Data + π-System Size		Geometrical Data + ECFP4		Geometrical Data + ECFP4 + π-System Size	
	R^2	RMSE (eV)	R^2	RMSE (eV)	R^2	RMSE (eV)	R^2	RMSE (eV)
Run 1	0.533	0.138	0.536	0.138	0.706	0.109	0.719	0.107
Run 2	0.526	0.140	0.548	0.137	0.677	0.116	0.681	0.115
Run 3	0.521	0.140	0.559	0.134	0.661	0.118	0.663	0.118
Average	**0.526** ± **0.003**	**0.139** ± **0.001**	**0.548** ± **0.007**	**0.136** ± **0.001**	**0.681** ± **0.013**	**0.114** ± **0.003**	**0.688** ± **0.017**	**0.113** ± **0.003**

TABLE I: Cross-validated R^2 and RMSE, averages and standard errors, to show model development improvement as new features are added to the representation.

A mathematical loss function is dictating how much each node is contributing to the network, creating a complex structure that can predict values or classify objects[37].

In a random forest model, the key hyperparameter is only the number of trees in the forest. The greater the number of trees is likely to yield a better predictions, but also increases the time it takes to train. Moreover, at point, the model comes to a prediction ceiling, and adding more trees will not improve the model. We optimized the number of trees in the random forest model, ranging from 10 to 1500, and recorded the training time, the Scikit-Learn built-in *score* function value for random forest models, which is comparable to the coefficient of determination, R^2, and the root mean square error (RMSE) for to the test set. As indicated in Figure S11, 50 trees are the optimal number for the random forest, as it gives the optimal training time, of about 25 seconds, while having the highest score and lowest RMSE.

For the gradient boosting trees model, there are several hyperparameters to optimize — including the number of trees, maximum tree depth, minimum sample split, learning rate, and the loss function. Optimization started using the common starting parameters of 1000 trees, unlimited maximum tree depth, a minimum sample split of 2, learning rate of 0.01, and the *least squares* loss function. Parameters were manually sampled, comparing the mean square error (MSE) score. This initial sampling did not noticeably affect the performance relative to the random forest and the neural network model. Therefore a more exhaustive grid search over these hyperparameters was not performed.

As for the neural network model, we used Bayesian optimization using the HyperOpt and Hyperas python packages[38,39], to find the optimal number of hidden layers, the number of nodes in each layer, and the dropout amount. We searched over a space of 1 to 3 hidden layers, 20 to 200 nodes per layer, and dropout between 0 to 0.5. We found that 2 hidden layers, size 127 and 109 nodes receptively, and the dropout amount of 0.005 for the first hidden layer and 0.448 for the second hidden layer, are the optimal values for this neural network. We used the Continuously Differentiable Exponential Linear Units (CELU) activation function[40], as implemented in the EchoAI python package[41], for the input and both hidden layers, as it outperformed other functions - including the widely used Rectified Linear Unit (ReLU) function. The output layer consists of a single node as standard for regression, with a linear activation function. Training was performed using the Adam optimizer[42] using the mean square error (MSE) loss function. In order to reduce the number of hyperparameters needed to optimize, the *ReduceLROnPlateau* and the *EarlyStopping* Keras functions[31] were used to tune the learn-

	Random Forest		Neural Network		Gradient Boosting Trees	
	R^2	RMSE (eV)	R^2	RMSE (eV)	R^2	RMSE (eV)
Run 1	0.716	0.108	0.631	0.122	0.645	0.121
Run 2	0.662	0.118	0.653	0.120	0.575	0.134
Run 3	0.685	0.114	0.623	0.125	0.612	0.129
Average	**0.687** $\pm$ **0.016**	**0.113** $\pm$ **0.003**	**0.636** $\pm$ **0.009**	**0.122** $\pm$ **0.001**	**0.611** $\pm$ **0.020**	**0.128** $\pm$ **0.004**

TABLE II: R^2 and RMSE results for three runs for each machine learning method used. The training and test sets in each run were the same for each method. Averages with standard error.

ing rate during the training of the model and stop the training once there is no further improvement. Effectively optimized the learning rate and the number of epochs for the neural network training.

For cross-validation, each model was trained on three different train-test split sets, using the Scikit-learn *train_test_split* function[30] using random state values of 0, 42, and 420. We saw that the random forest model outperform both other models in both the R^2 value and the root mean square error (RMSE) (Table II). We therefore used the random forest model as our model of choice for the remaining work..

III. RESULTS AND DISCUSSION

In order to see how well the final random forest model can predict the λ of unseen oligomers we split the data set into a 85%-15% train-test sets, respectively, comparing the trained model to predict the λ of the test set and compared it to the B3LYP calculated λ. The correlation graph between the predicted and calculated energies (Figure 4) shows good correlation with unitless coefficient of determination, $R^2 = 0.717$ and root mean square error, $RMSE = 0.105$eV for the tetramers, $R^2 = 0.737$ and $RMSE = 0.140$eV for the hexamers, and $R^2 = 0.719$ and $RMSE = 0.107$eV in total.

Moreover, as figure 4 shows, the correlation also shows heteroscedasticity, where there is better correlation for oligomers with lower λ, and worse for compounds with greater reorganization energies. This shows that predicting the λ for oligomers with geometric differences between the neutral and cation species is a complex task. In all likelihood there are many possible geometric changes between neutral and cation geometries, and as such the limited training set makes it challenging for the model to properly account for all reorganization in compounds with large λ. Similar heteroscedastic behavior can be seen in the correlation between the tetramers and hexamers (Figure 3b. In principal, some of the heteroscedasticity in the predictions could be reduced by using more, or even only, hexamers in the training of the model. However, calculating the B3LYP *lambda* for hexamers is computationally expensive — which runs counter to the benefit of the ML model as a surrogate for the calculations.

For screening, where the intent is to find candidates with low λ, the larger heteroscedastic error for higher λ compounds has only a small effect — there is better correlation for compounds with small internal reorganization energies. Therefore we can use the random forest model as a first, rapid screening tool to find oligomers with low λ.

The random forest regression model, as implemented in the scikit-learn package, has a *feature_importance_* function[30], which enables exploration of the features in the representation that contribute the most to the model (Fig-

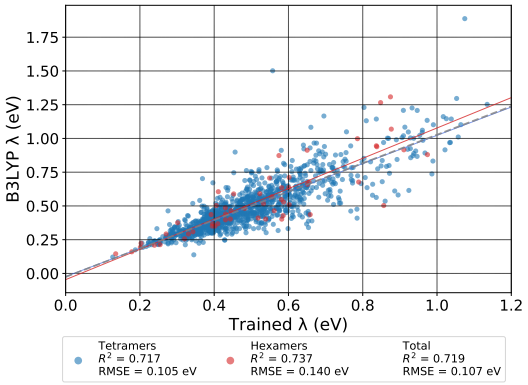


FIG. 4: Correlation plot between the random-forest predicted λ and the B3LYP calculated λ for the tetramers and hexamers in the test set.

ure S12). It is clear that the most important feature is the average inter-ring bond length of the neutral oligomer, as calculated using xTB-GFN2. While the bond lengths are expected to change going from the neutral to cation geometries, this is a surprising effect as the correlation between the neutral inter-ring bond length and the B3LYP-calculated λ is weak ($R^2 = 0.233$, Figure S14). Much like the overall reorganization energies, the correlation between GFN2-computed and B3LYP-computed inter-ring bond lengths shows the same heteroscedasticity, which may explain some of the feature importance. The second most important feature is the π-system size descriptor, which agrees with the hypotheses that bigger π-conjugated systems promote lower λ. The third most important feature is the ECFP bit number 1019, which indicates the existence of an sp^3 hybridized carbon in the oligomer (Figure S13). Two possible explanations exist for this feature — that a sp^3 hybridized carbon breaks conjugation, and as discussed below, the CH_2 group may promote a less planar conformation. Interestingly, the monomer numbers, although used as a categorical feature with a seemingly arbitrary assignment meant for naming only, do appear to contribute to the model as the fourth and sixth most important features. The rest of the features are the other geometrical information we encoded into the representation, followed by the rest of the ECFP bits which minimally contribute to the model.

After using 85% of the training set to train the model for testing purposes, the final random forest model was trained using the full data set for screening a larger validation set to predict the λ of 24,853 tetramers and 31,722 hexamers that were not part of the original data set. From those new predicted λ, oligomers with $\lambda < 0.3$ eV were filtered to compute the full B3LYP λ, including 660 tetramers and 1753 hexamers with low λ (Figure 5 a, b). The increase in the number of oligomers with $\lambda < 0.3$ev from the tetramers to the hexamers agrees with assessment of the inverse relationship between the length of the oligomer and its reorganization energy[12].

We also looked to trends in the predictions in order to see if there are monomers that repeatedly contribute to oligomers with low λ (Figure 5 c, d). For both tetramers and hexamers, the monomers number 47, 110, 158, 213, 258, and 283 are found frequently (Figure 6). As it can be seen, all the best performing monomers have a fused aromatic system on the thiophene backbone, supporting our hypothesis that a larger π-system contributes to low λ. Moreover, excluding monomer 253 which only has one, all of the monomers have two aromatic nitrogen atoms in the 3- and 4- positions on the thiophene ring. We hypothesize that steric considerations contribute, as CH_2 groups in these positions increase the steric repulsion between neighboring monomers forcing a non-planar, twisted chain conformation, and increasing λ.[12]

In order to validate the accuracy of the full random forest model the 300 tetramers and 150 hexamers with the lowest predicted λ were selected, and the B3LYP λ was computed to compare with the RF model prediction (Figure 7). While the predicted values are not perfect, the low RMSE (0.036 eV) of the prediction versus the calculated λ, indicates the that model is robust and accurate at this new validation set,

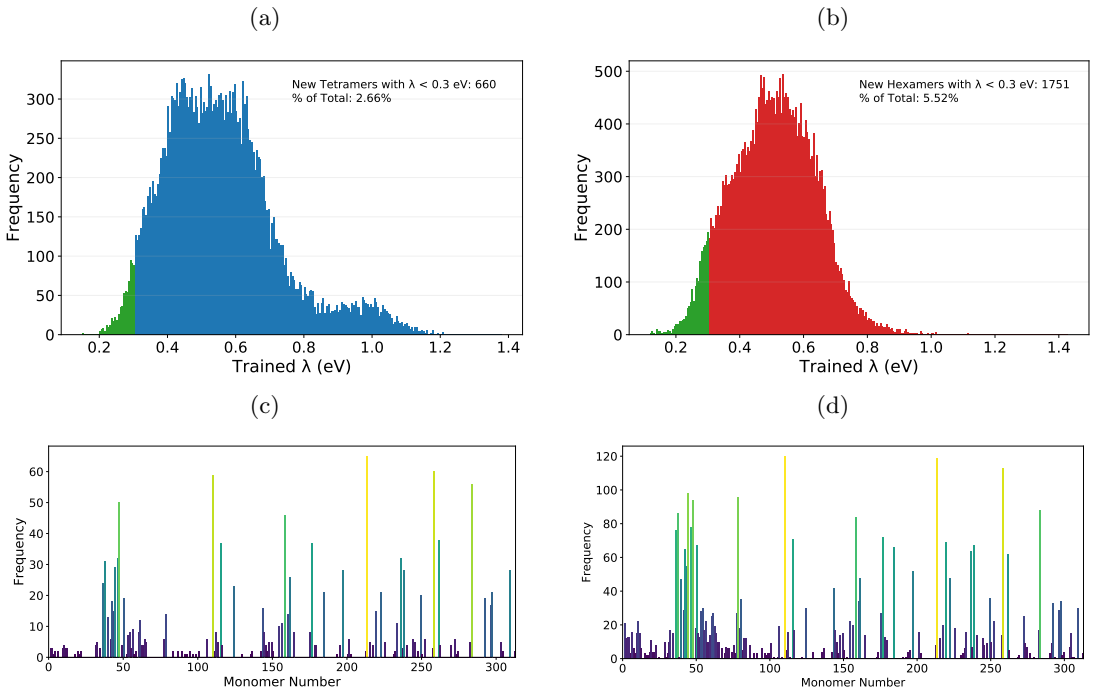


FIG. 5: λ predictions for new tetramers and hexamers. Histogram of predicted λ for the new tetramers **(a)** and hexamers **(b)**, with tetramers and hexamers with $\lambda < 0.3$ eV colored in green. Histograms of the common monomers for the tetramers **(c)** and hexamers **(d)** with $\lambda < 0.3$ eV.

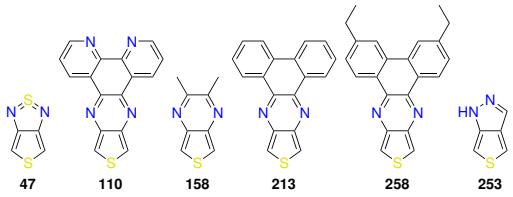


FIG. 6: The top 6 common monomers in oligomers with predicted $\lambda < 0.3$ eV.

and thus can be used as a first step in finding conjugated materials with better charge transport properties. Interestingly, of the 50 hexamers with the lowest B3LYP λ, 44 oligomers had monomer 47 as one of their monomers, and the hexamer with the lowest B3LYP λ consists of the homo-oligomer of monomer 47, with $\lambda = 0.051$ eV (Figure 7b). This fragment, and related monomers are frequently used in top organic photovoltaic materials.

Moreover, the dihedral angle between the best performing hexamers is close to 180°(Table S2), or in other words - flat, and is only minimally changing between the neutral and cation species (Table III). This further strengthen the hypothesis, that in addition to a large π-system, for low λ, better conjugation and planar chain conformations contribute to the low λ. In addition to the dihedral angle, the best performing hexamers exhibit a minimal change between the neutral and the cation bond lengths.

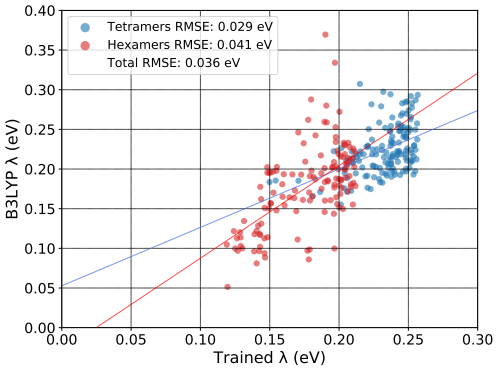
IV. CONCLUSION

In this work we have shown that a random forest model can be used as a screening tool to

Monomer 1	Monomer 2	Predicted λ	B3LYP λ	GFN2 ΔAngle (°)	B3LYP ΔAngle (°)	GFN2 ΔBond Length (Å)	B3LYP ΔBond Length (Å)
47	47	0.120	0.051	0.063	0.005	0.004	0.005
47	116	0.141	0.081	0.034	0.097	0.005	0.008
47	156	0.178	0.086	0.415	0.019	0.006	0.010
47	247	0.147	0.088	0.352	1.628	0.005	0.010
47	217	0.146	0.094	0.032	4.460	0.006	0.012

TABLE III: The monomer numbers, the predicted and calculated B3LYP λ, The GFN2 and B3LYP geometrical data of the average change in dihedral angles between the neutral and cation species, and average change in the inter-ring bond length of both neutral and cation species for the five hexamers with the lowest B3LYP λ.

(a)



(b)

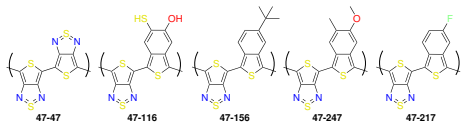


FIG. 7: **(a)** Correlation plot for the tetramers and hexamers with low λ, trendline indicated robust linear regression fit, tetramers in blue, and hexamers in red. **(b)** The top 5 hexamers with the lowest B3LYP calculated λ. The numbers represent the two monomers in the chain.

find thiophene based oligomers with low and high λ. Our goal was to train a model by minimizing the calculation time required to generate the training set by calculating the λ of shorter oligomers (i.e., tetramers), correlating with the λ of longer lengths, saving considerable computational time. The resulting random forest regression model can predict thousands of new oligomers in a matter of seconds, yielding a list of potential oligomers with low λ for further screening. Comparing the time required to generate the test and validation sets to the possible time required to calculate all 31,878 tetramers and 31,878 hexamers, the RF model yields a $\sim$13$\times$ speedup.

From the predictions of the model and the relative feature importance, it is clear that oligomers with large, conjugated, π-system have lower internal reorganization energies. In addition to a large π-system size, monomers with low steric bulk, which minimally change conformation upon a hole transfer, and also yield a high degree of delocalization and π orbital overlap between the monomers, also contributes to low λ. One monomer in particular, with a thiadiazole group is frequently observed in compounds with low internal reorganization energy. All the top oligomers also share similar geometries, i.e. being almost completely flat, and only exhibit minimal changes in geometry upon a hole transfer. Future work can consider a similar method for internal reorganization energies of n-type electron transfer, or other calculated properties requiring multiple time-intensive computational steps.

ACKNOWLEDGMENTS

We acknowledge support from Department of Energy-Basic Energy Sciences Computational and Theoretical Chemistry (Award DE-SC0019335) and the University of Pittsburgh Center for Research Computing through the computational resources provided, and to Dakota Folmsbee for thoughtful discussions.

[1] Z. Bao and A. J. Lovinger, Chemistry of Materials **11**, 2607 (1999).
[2] R. Porrazzo, S. Bellani, A. Luzio, C. Bertarelli, G. Lanzani, M. Caironi, and M. R. Antognazza, APL Materials **3**, 014905 (2015).
[3] Y. Kim, S. A. Choulis, J. Nelson, D. D. C. Bradley, S. Cook, and J. R. Durrant, Journal of Materials Science **40**, 1371 (2005).
[4] M. Zhang, X. Guo, W. Ma, H. Ade, and J. Hou, Advanced Materials **26**, 5880 (2014).
[5] Z. G. Zhang, S. Zhang, J. Min, C. Cui, H. Geng, Z. Shuai, and Y. Li, Macromolecules **45**, 2312 (2012).
[6] F. Wang, H. Gu, and T. M. Swager, Journal of the American Chemical Society **130**, 5392 (2008).
[7] P. Schottland, M. Bouguettaya, and C. Chevrot, Synthetic Metals **102**, 1325 (1999).
[8] L. Wang, Q. Feng, X. Wang, M. Pei, and G. Zhang, New Journal of Chemistry **36**, 1897 (2012).
[9] B. H. Barboza, O. P. Gomes, and A. Batagin-Neto, Journal of Molecular Modeling **27**, 17 (2021).
[10] S. K. Kang, J. H. Kim, J. An, E. K. Lee, J. Cha, G. Lim, Y. S. Park, and D. J. Chung, Polymer Journal **36**, 937 (2004).
[11] A. L. Ding, J. Pei, Y. H. Lai, and W. Huang, Journal of Materials Chemistry **11**, 3082 (2001).
[12] G. R. Hutchison, M. A. Ratner, and T. J. Marks, Journal of the American Chemical Society **127**, 2339 (2005).
[13] J. Cornil, D. Beljonne, J. P. Calbert, and J. L. Brédas, Interchain interactions in organic π-conjugated materials: Impact on electronic structure, optical response, and charge transport (2001).
[14] S. Zade and M. Bendikov, Chemistry - A European Journal **14**, 6734 (2008).
[15] C. Bannwarth, S. Ehlert, and S. Grimme, Journal of Chemical Theory and Computation **15**, 1652 (2019).
[16] D. Folmsbee and G. Hutchison, International Journal of Quantum Chemistry **121**, 10.1002/qua.26381 (2020).
[17] C. Lee, W. Yang, and R. G. Parr, Physical Review B **37**, 785 (1988).
[18] A. D. Becke, The Journal of Chemical Physics **98**, 5648 (1993).
[19] H. Sahu and H. Ma, Journal of Physical Chemistry Letters **10**, 7277 (2019).
[20] M. Rinderle, W. Kaiser, A. Mattoni, and A. Gagliardi, Journal of Physical Chemistry C **124**, 17733 (2020).
[21] D. Padula, J. D. Simpson, and A. Troisi, Materials Horizons **6**, 343 (2019).
[22] D. Padula and A. Troisi, Advanced Energy Materials **9**, 1902463 (2019).
[23] C. Chen, Y. Zuo, W. Ye, X. Li, Z. Deng, and S. P. Ong, Advanced Energy Materials **10**, 1903242 (2020).
[24] T. Sato, T. Honma, and S. Yokoyama, Journal of Chemical Information and Modeling **50**, 170 (2010).
[25] J. Vamathevan, D. Clark, P. Czodrowski, I. Dunham, E. Ferran, G. Lee, B. Li, A. Madabhushi, P. Shah, M. Spitzer, and S. Zhao, Applications of machine learning in drug discovery and development (2019).
[26] N. M. O'Boyle, M. Banck, C. A. James, C. Morley, T. Vandermeersch, and G. R. Hutchison, Journal of Cheminformatics **3**, 33 (2011).
[27] N. Yoshikawa and G. R. Hutchison, Journal of Cheminformatics **11**, 10.1186/s13321-019-0372-5 (2019).
[28] V. A. Rassolov, J. A. Pople, M. A. Ratner, and T. L. Windus, Journal of Chemical Physics **109**, 1223 (1998).
[29] M. J. Frisch, G. W. Trucks, H. B. Schlegel, G. E. Scuseria, M. A. Robb, J. R. Cheeseman, G. Scalmani, V. Barone, B. Mennucci, G. A. Petersson, H. Nakatsuji, M. Caricato, X. Li, H. P. Hratchian, A. F. Izmaylov, J. Bloino, G. Zheng, J. L. Sonnenberg, M. Hada, M. Ehara, K. Toyota, R. Fukuda, J. Hasegawa, M. Ishida, T. Nakajima, Y. Honda, O. Kitao, H. Nakai, T. Vreven, J. A. Montgomery, J. E. Peralta, F. Ogliaro, M. Bearpark, J. J. Heyd, E. Brothers, K. N. Kudin, V. N. Staroverov, R. Kobayashi, J. Normand, K. Raghavachari, A. Rendell, J. C. Burant, S. S. Iyengar, J. Tomasi, M. Cossi, N. Rega, J. M. Millam, M. Klene, J. E. Knox, J. B. Cross, V. Bakken, C. Adamo, J. Jaramillo, R. Gomperts, R. E. Stratmann, O. Yazyev, A. J. Austin, R. Cammi, C. Pomelli, J. W. Ochterski, R. L. Martin, K. Morokuma, V. G. Zakrzewski, G. A. Voth, P. Salvador, J. J. Dannenberg, S. Dapprich, A. D. Daniels, Ö. Farkas, J. B. Foresman, J. V. Ortiz, J. Cioslowski, and D. J. Fox, Gaussian 09 Revision A.2 (2009).
[30] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, Journal of Machine Learning Research **12**, 2825 (2011).
[31] F. Chollet *et al.*, Keras, `https://keras.io` (2015).
[32] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng,

TensorFlow: Large-scale machine learning on heterogeneous systems (2015), software available from tensorflow.org.
[33]D. Rogers and M. Hahn, Journal of Chemical Information and Modeling **50**, 742 (2010).
[34]RDKit: Open-source cheminformatics, `http://www.rdkit.org` (2020), [Online; accessed 1-Mar-2021].
[35]L. Breiman, Machine Learning **45**, 5 (2001).
[36]J. Ye, J.-H. Chow, J. Chen, and Z. Zheng, in *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, CIKM '09 (Association for Computing Machinery, New York, NY, USA, 2009) p. 2061–2064.
[37]W. S. McCulloch and W. Pitts, The bulletin of mathematical biophysics **5**, 115 (1943).
[38]J. Bergstra, D. Yamins, and D. C. B. T. P. o. t. t. I. C. o. M. Learning, Making a Science of Model Search: Hyperparameter Optimization in Hundreds of Dimensions for Vision Architectures (2013).
[39]M. Pumperla, Hyperas, `https://github.com/maxpumperla/hyperas` (2020).
[40]J. T. Barron, Continuously differentiable exponential linear units (2017), arXiv:1704.07483.
[41]D. Misra, Echo, `https://github.com/digantamisra98/Echo` (2020).
[42]D. P. Kingma and J. L. Ba, in *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings* (International Conference on Learning Representations, ICLR, 2015) arXiv:1412.6980.

Supplementary Information

Monomer	SMILES
1	c(s1)ccc1
2	c(s1)cc(F)c1
3	c(s1)cc(Cl)c1
4	c(s1)cc(Br)c1
5	c(s1)cc(C(F)(F)(F))c1
6	c(s1)cc(C(#N))c1
7	c(s1)cc(N(=O)=O)c1
8	c(s1)cc(N)c1
9	c(s1)cc(C)c1
10	c(s1)cc(O)c1
11	c(s1)cc(OC)c1
12	c(s1)cc(S)c1
14	c(s1)cc(C(=O)O)c1
15	c(s1)cc(C=O)c1
16	c(s1)cc(C(C)=O)c1
17	c(s1)cc(C(C(F)(F)(F))=O)c1
18	c(s1)cc(c2ccccc2)c1
19	c(s1)c(F)c(F)c1
20	c(s1)c(Cl)c(Cl)c1
21	c(s1)c(Br)c(Br)c1
22	c(s1)c(C#N)c(C#N)c1
23	c(s1)c(N(=O)=O)c(N(=O)=O)c1
24	c(s1)c(C)c(C)c1
25	c(s1)c(OC)c(OC)c1
26	c(s1)c(OC)c(N)c1
27	c(s1)c(OC)c(C#N)c1
28	c(s1)c(N)c(N(=O)(=O))c1
29	c(s1)c(C(#N))c(C(F)(F)(F))c1
30	c(s1)c(O)c(C(=O)O)c1
31	c(s1)c2CCCc2c1
32	c(s1)c2OCOc2c1
33	c(s1)c2NCNc2c1
34	c(s1)c2SCSc2c1
36	c(s1)c2occc2c1
37	c(s1)c2Nccc2c1
38	c(s1)c2N(C(F)(F)(F))ccc2c1
39	c(s1)c2sccc2c1
41	c(s1)c2Cccc2c1
42	c(s1)c2ocnc2c1
43	c(s1)c2scnc2c1
44	c(s1)c2Ncnc2c1
45	c(s1)c2onnc2c1
46	c(s1)c2snnc2c1
47	c(s1)c2N=S=Nc2c1
49	c(s1)c2c(=C)ccc2c1
50	c(s1)c2c(=O)ccc2c1
51	c(s1)c2s(=O)ccc2c1
52	c(s1)c2s(=O)(=O)ccc2c1
53	c(s1)c2C(=O)CC(=O)c2c1
54	c(s1)c2C(=O)NC(=O)c2c1
55	c(s1)c2C(=O)N(C(F)(F)(F))C(=O)c2c1
56	c(s1)c2C(=O)OC(=O)c2c1
57	c(s1)c2C(=O)SC(=O)c2c1
58	c(s1)c2oc(=O)oc2c1
59	c(s1)c2oc(=S)oc2c1
60	c(s1)c2NC(=O)Nc2c1
61	c(s1)c2NC(=S)Nc2c1

Monomer	SMILES
62	c(s1)c2sc(=O)sc2c1
63	c(s1)c2sc(=S)sc2c1
64	c(s1)c2OCCOc2c1
65	c(s1)c2OCCCOc2c1
66	c(s1)c2NCCNc2c1
67	c(s1)c2SCCSc2c1
69	c(s1)c2CC=CCc2c1
70	c(s1)c2occoc2c1
71	c(s1)c2NC=CNc2c1
72	c(s1)c2SccSc2c1
74	c(s1)c2oc(=O)c(=O)oc2c1
75	c(s1)c2Nc(=O)c(=O)Nc2c1
76	c(s1)c2Sc(=O)c(=O)Sc2c1
77	c(s1)c2ccccc2c1
78	c(s1)c2nccnc2c1
79	c(s1)c2cnncc2c1
80	c(s1)c2ncncc2c1
81	c(s1)c2cc(OC)c(OC)cc2c1
82	c(s1)c2cc(C#N)c(C#N)cc2c1
83	c(s1)c2cc(F)c(F)cc2c1
84	c(s1)c2c(F)c(F)c(F)c(F)c2c1
85	c(s1)c2c(F)ccc(F)c2c1
86	c(s1)c2cc(N(=O)=O)c(N(=O)=O)cc2c1
89	c(s1)c2CCCCc2c1
91	c(s1)cc(C=C)c1
92	c(s1)c(C)c(C=C)c1
93	c(s1)cc(C(=O)OC)c1
94	c(s1)c(S)c(O)c1
95	c(s1)c(OC)c(C(F)(F)(F))c1
96	c(s1)cc(c2ccc(N)cc2)c1
97	c(s1)cc(c2ccc(OC)cc2)c1
98	c(s1)cc(c2ccc(F)cc2)c1
99	c(s1)cc(c2ccc(N(=O)=O)cc2)c1
100	c(s1)c2OCCSc2c1
101	c1sc(c(c1C(F)(F)F)C(F)(F)F)
102	c1sc(c(c1CC)CC)
105	c1sc(c2c1SCCCS2)
106	c1sc(c2c1cc(C(=O)C)cc2)
107	c1sc(c2c1c(C#N)ccc2C#N)
108	c1sc(c2c1CCC[C@H]2O)
109	c1sc(c2c1c(C)ccc2)
110	c1sc(c2c1nc1c3cccnc3c3ncccc3c1n2)
111	c(s1)c2NCOc2c1
112	c1sc(c2c1sc(C(=O)CC)c2F)
113	c1sc(c2c1C(=O)c1ccccc1C2=O)
114	c1sc(c2c1C[C@H](F)[C@@H](F)C2)
115	c1c2c(ncc(CC)n2)c(s1)
116	c1sc(c2c1cc(S)c(O)c2)
118	c1sc(c2c1c(N(=O)=O)ccc2N)
120	c1sc(c2c1C[C@H](C#N)[C@@H](C#N)C2)
121	c1sc(c2c1C[C@H](S)CC2)
124	c1sc(c2c1sc(C(=O)OCC)c2)
127	c1sc(c2c1c(S)ccc2O)
129	c1sc(c2c1c(C(=O)OC)ccc2)
132	c1sc(c2c1CC[C@@H](N(=O)=O)C2)
133	c1sc(c2c1[C@H](C#N)CC[C@H]2OC)
135	c1sc(c2c1SCC(=O)CO2)
136	c(s1)c2ccs(=N)(=O)c2c1
137	c1sc(c2c1[C@H](C)CCC2)

Monomer	SMILES
138	c1sc(c2c1c(C=C)ccc2C)
140	c1sc(c2c1[C@H](C=C)CCC2)
142	c1sc(c(c1O)C(=O)O)
143	c1sc(c2c1nc(OCC)c(CN)n2)
144	c1sc(c2c1sc(=S)o2)
145	c1sc(c2c1cc(C(F)(F)F)cc2)
146	c1c(CC)cc(s1)
147	c1sc(c2c1cc(C)c(C)c2)
149	c1sc(c2c1cc(S)cc2)
150	c1c2c(OCN2)c(s1)
151	c1sc(c2c1C[C@H](OC)[C@@H](OC)C2)
152	c1sc(c2c1S(=O)(=O)CCC2)
154	c1sc(c2c1cc(C=C)cc2)
156	c1sc(c2c1cc(C(C)(C)C)cc2)
157	c1sc(c2c1c(C(F)(F)F)ccc2C#N)
158	c1c2nc(C)c(C)nc2c(s1)
159	c1sc(c2c1CC[C@@H](N)C2)
160	c1sc(c2c1nc(OCC)c(OCC)n2)
161	c1c2c(nccc2)c(s1)
163	c1sc(c2c1c(C(F)(F)F)ccc2)
164	c1sc(c2c1sc(=O)s2)
166	c1sc(c2c1[C@H](C)CC[C@H]2C)
167	c1sc(c2c1C[C@H](C(=O)OC)CC2)
169	c1sc(c2c1CCS(=O)(=O)C2)
170	c1sc(c2c1CC[C@@H](O)C2)
171	c1sc(c2c1[C@H](N(=O)=O)CC[C@H]2N(=O)=O)
172	c1sc(c2c1c(C=C)ccc2)
173	c(s1)c(O)c(NN)c1
174	c1sc(c(c1C(=O)O)OC)
175	c1sc(c2c1cc(N(=O)=O)c(N)c2)
176	c1c2c(ncc(C)n2)c(s1)
178	c1sc(c2c1ncnc2C#N)
179	c1sc(c2c1cc(C(=O)C(F)(F)F)cc2)
183	c1sc(c(c1N(=O)=O)N(=O)=O)
184	c1sc(c2c1nc(CCO)c(CN)n2)
185	c1sc(c(c1O)OC)
187	c1sc(c2c1C[C@H](C(F)(F)F)[C@@H](C(F)(F)F)C2)
188	c1sc(c2c1[C@H](C(=O)C(F)(F)F)CCC2)
190	c1sc(c2c1CC(=O)C(=O)C2)
191	c1sc(c(c1OCC)ON(=O)=O)
192	c1sc(c(c1N(=O)=O)C#N)
193	c1sc(c2c1oc(=O)c(=O)s2)
194	c1sc(c2c1[C@H](C(=O)OC)CCC2)
195	c(s1)c2c(=O)N(CC)c(=O)c2c1
197	c1sc(c2c1oc(C#N)c2)
198	c1sc(c2c1[C@H](C(F)(F)F)CC[C@H]2C(F)(F)F)
200	c1sc(c2c1C(=O)CCC2=O)
201	c1sc(c2c1cc(C=C)c(C)c2)
202	c1sc(c2c1sc(=O)c(=O)s2)
204	c1sc(c(c1OCC)OCC)
205	c1sc(c2c1C[C@H](C#N)[C@@H](OC)C2)
206	c1sc(c2c1C[C@H](N(=O)=O)[C@@H](N(=O)=O)C2)
207	c(s1)c(CCN)c(CCN)c1
211	c1sc(c2c1[C@H](C(F)(F)F)CCC2)
212	c1sc(c2c1cccc2F)
213	c1sc(c2c1nc1c3ccccc3c3ccccc3c1n2)
214	c1sc(c2c1c(N(=O)=O)ccc2)
215	c1sc(c2c1sc(=S)s2)
217	c1sc(c2c1ccc(F)c2)

Monomer	SMILES
218	c1sc(c2c1C[C@H](S)[C@@H](O)C2)
219	c1sc(c2c1nc(CN)c(CN)n2)
220	c1sc(c2c1C[C@H](C)[C@@H](C)C2)
222	c1sc(c2c1oc(N(=O)=O)c2)
223	c1sc(c2c1c(C#N)ccc2)
225	c1sc(c2c1cc(OC)cc2)
226	c1sc(c2c1C[C@H](N)[C@@H](N(=O)=O)C2)
227	c1sc(c2c1[C@H](C#N)CC[C@H]2C#N)
229	c1sc(c2c1OCSS2)
230	c1sc(c2c1c(S)ccc2)
231	c1sc(c2c1OCC[C@@H](SC)O2)
232	c1sc(c2c1OSCS2)
233	c1c2c(sc(C(=O)OCC)c2F)c(s1)
234	c1sc(c2c1[C@H](F)[C@H](F)[C@@H](F)[C@H]2F)
236	c1sc(c2c1[nH]c(N(=O)=O)c2)
237	c1sc(c2c1oc(=O)o2)
238	c1c2nc(CC)c(CC)nc2c(s1)
239	c1sc(c2c1[C@H](C(=O)C)CCC2)
240	c1sc(c2c1cc(C=O)cc2)
241	c1sc(c2c1C[C@H](C(=O)C)CC2)
242	c1sc(c2c1[C@H](S)CC[C@H]2O)
244	c1sc(c(c1O)C#N)
245	c1sc(c2c1cc(C#N)c(OC)c2)
247	c1sc(c2c1cc(C)c(OC)c2)
248	c1sc(c2c1[C@H](F)CC[C@H]2F)
249	c1sc(c2c1sc(C(=O)CC)c2)
250	c1sc(c2c1oc(=S)o2)
251	c1sc(c2c1CCC[C@H]2N(=O)=O)
252	c1sc(c2c1cc(C)cc2)
254	c1sc(c2c1c(C)ccc2C)
255	c1sc(c2c1CC[C@@H](C)C2)
256	c1sc(c2c1[C@H](C)CC[C@H]2C=C)
257	c1sc(c2c1cc(N(=O)=O)cc2)
258	c1sc(c2c1nc1c3ccc(CC)cc3c3cc(CC)ccc3c1n2)
259	c1sc(c2c1CC[C@@H](C=O)C2)
261	c1sc(c2c1[nH]c(C#N)c2)
262	c1sc(c2c1cc1C(=O)N(CC)C(=O)c1c2)
263	c1sc(c2c1[C@H](C(F)(F)F)CC[C@H]2OC)
264	c1sc(c2c1C[C@H](C(F)(F)F)CC2)
266	c1sc(c2c1CCC[C@H]2C=O)
267	c1sc(c(c1C=C)OC)
268	c(s1)c2C(NCC)OC(O)c2c1
269	c1sc(c2c1C(=O)c1ccc(CC)cc1C2=O)
270	c1sc(c2c1C[C@H](C(=O)C(F)(F)F)CC2)
271	c1sc(c2c1CC[C@@H](OC)C2)
272	c1sc(c2c1cc(C(=O)O)c(O)c2)
273	c1sc(c2c1cc(C(F)(F)F)c(OC)c2)
274	c1sc(c2c1cc(C(F)(F)F)c(C#N)c2)
275	c1sc(c2c1[C@H](OC)CC[C@H]2OC)
276	c1sc(c2c1c(C(=O)C(F)(F)F)ccc2)
277	c1sc(c2c1C[C@H](C(F)(F)F)[C@@H](C#N)C2)
279	c1sc(c2c1OCCCC2)
280	c1sc(c2c1CC[C@@H](C(=O)O)C2)
281	c1sc(c2c1[C@H](OC)CCC2)
282	c1sc(c2c1cc(C#N)cc2)
283	c1sc(c2c1[nH]nc2)
284	c1sc(c2c1[C@H](N)CC[C@H]2N)
285	c1sc(c2c1C[C@H](C(=O)O)[C@@H](O)C2)
286	c1sc(c2c1[C@H](S)CCC2)

Monomer	SMILES
288	c1sc(c2c1c(C(F)(F)F)ccc2OC)
290	c1sc(c(c1O)C(F)(F)F)
291	c1sc(c2c1CN2)
292	c1sc(c2c1cc(C(=O)OC)s2)
293	c1sc(c2c1c(C(=O)O)ccc2O)
296	c1sc(c2c1cc(C)s2)
297	c1sc(c2c1sc(N(=O)=O)c2)
298	c1sc(c2c1oc(=O)c(=O)o2)
299	c1sc(c2c1c(C(=O)C)ccc2)
301	c1sc(c2c1[C@H](C(F)(F)F)CC[C@H]2C#N)
303	c1sc(c2c1cc(SCC)c(SCC)c2)
304	c1sc(c2c1cc(C(=O)OC)cc2)
305	c1sc(c2c1SC(=O)CC(=O)O2)
306	c1sc(c2c1SCC(=O)CS2)
307	c1sc(c2c1c(C#N)ccc2OC)
309	c1sc(c2c1sc(C#N)c2)
310	c1sc(c2c1[C@H](N)CC[C@H]2OC)
311	c1sc(c2c1c(C=O)ccc2)
312	c1sc(c2c1OCCCS2)

TABLE S1: List of monomers numbers and their correspondent SMILES string

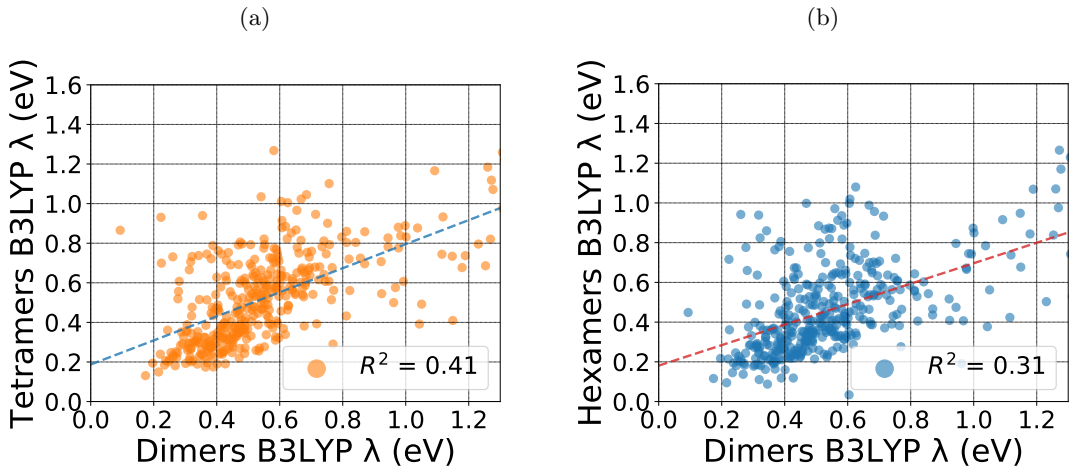


FIG. S1: Correlation of B3LYP calculated λ between **(a)** dimers and tetramers, and **(b)** dimers and hexamers. Trendlines indicated robust linear regression fit.

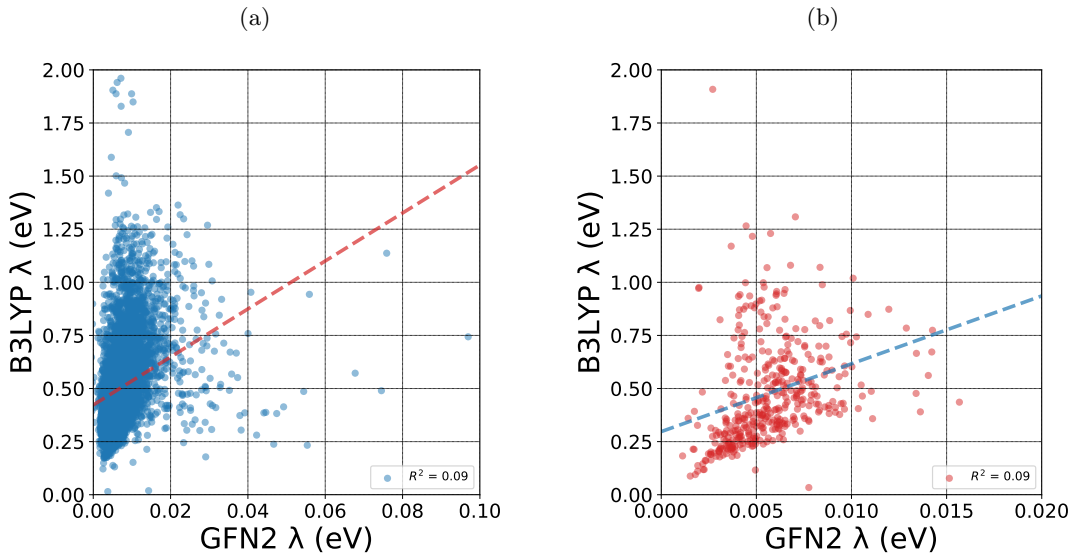


FIG. S2: Correlation between λ calculated using B3LYP vs. λ calculated using GFN2 for **(a)** tetramers and **(b)** hexamers. Trendlines indicated robust linear regression fit.

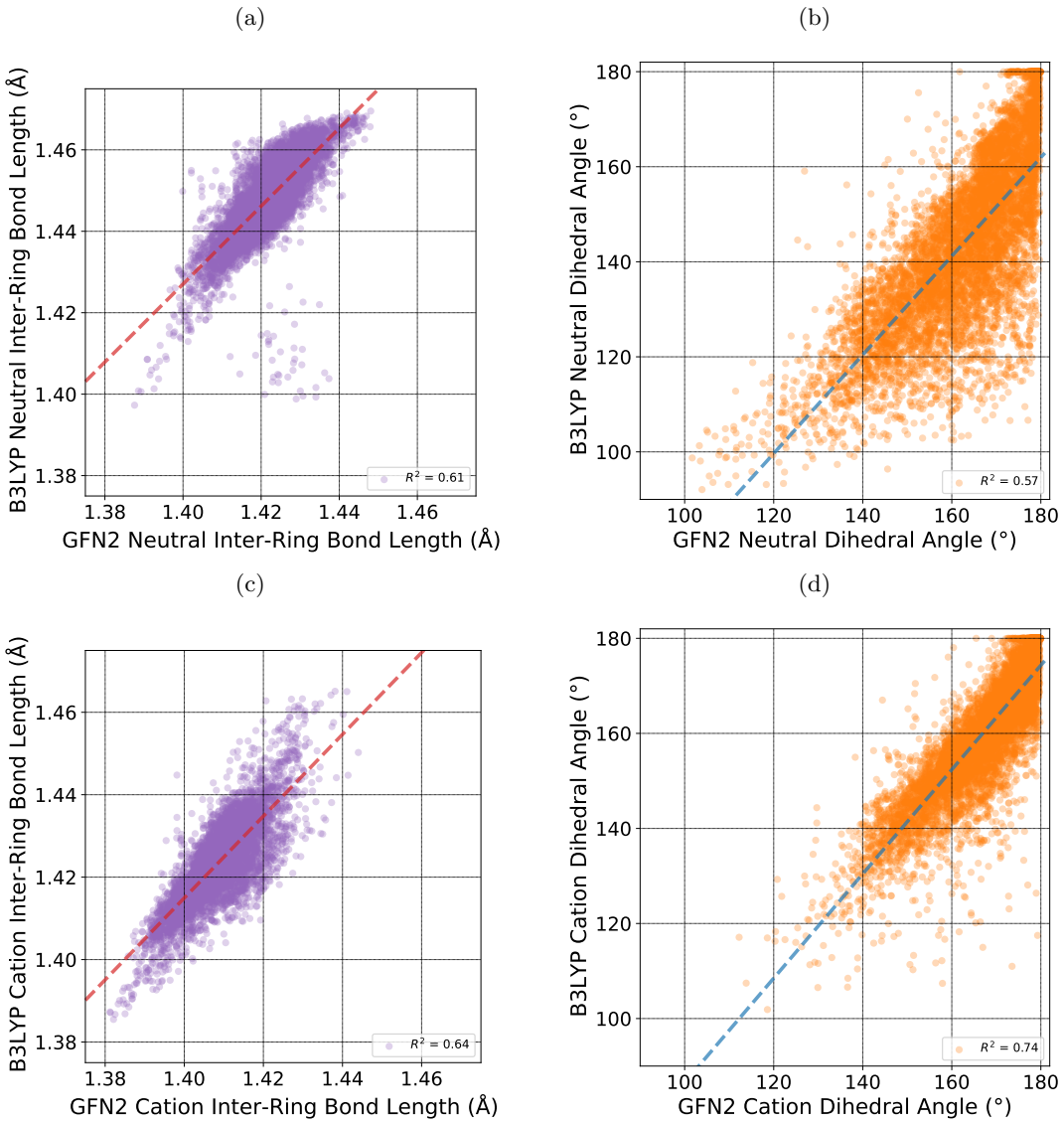


FIG. S3: Correlation between the dihedral angle (**b, d**) and the inter-ring bond length (**a, c**) between the monomers calculated using B3LYP vs. GFN2 for the neutral (**a, b**) and cation (**c, d**) species. Trendlines indicated robust linear regression fit.

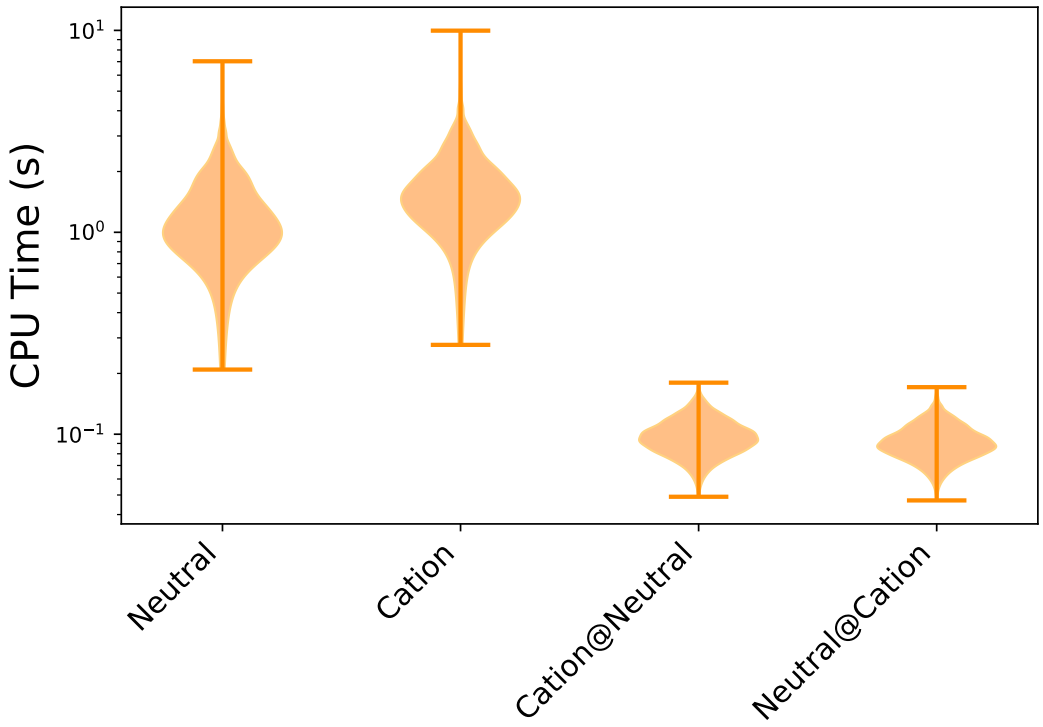


FIG. S4: Calculation run time of the 4 different calculation for the dimers using GFN2. Note the logarithmic y-axis.

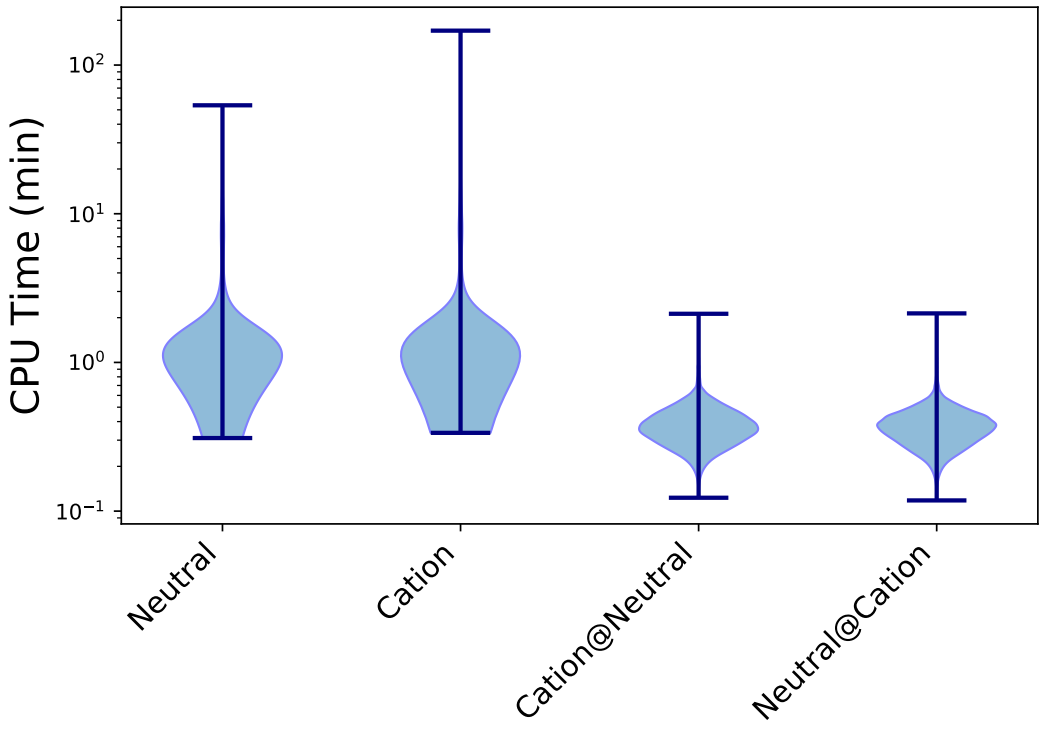


FIG. S5: Calculation run time of the 4 different calculation for the tetramers using GFN2. Note the logarithmic y-axis.

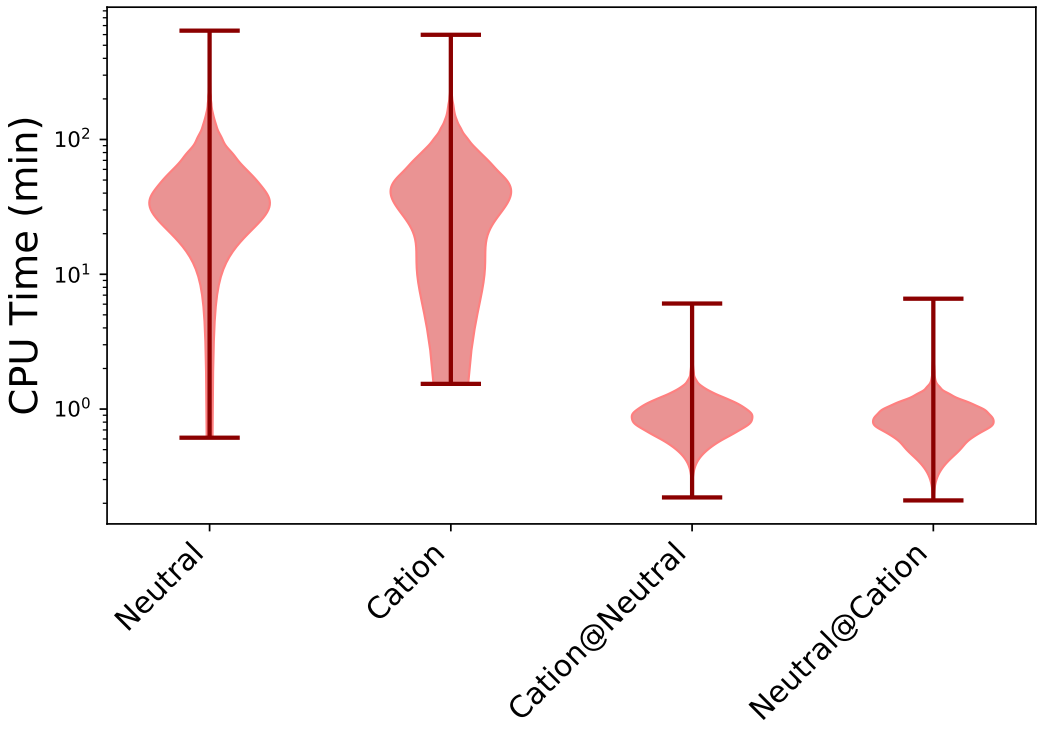


FIG. S6: Calculation run time of the 4 different calculation for the hexamers using GFN2. Note the logarithmic y-axis.

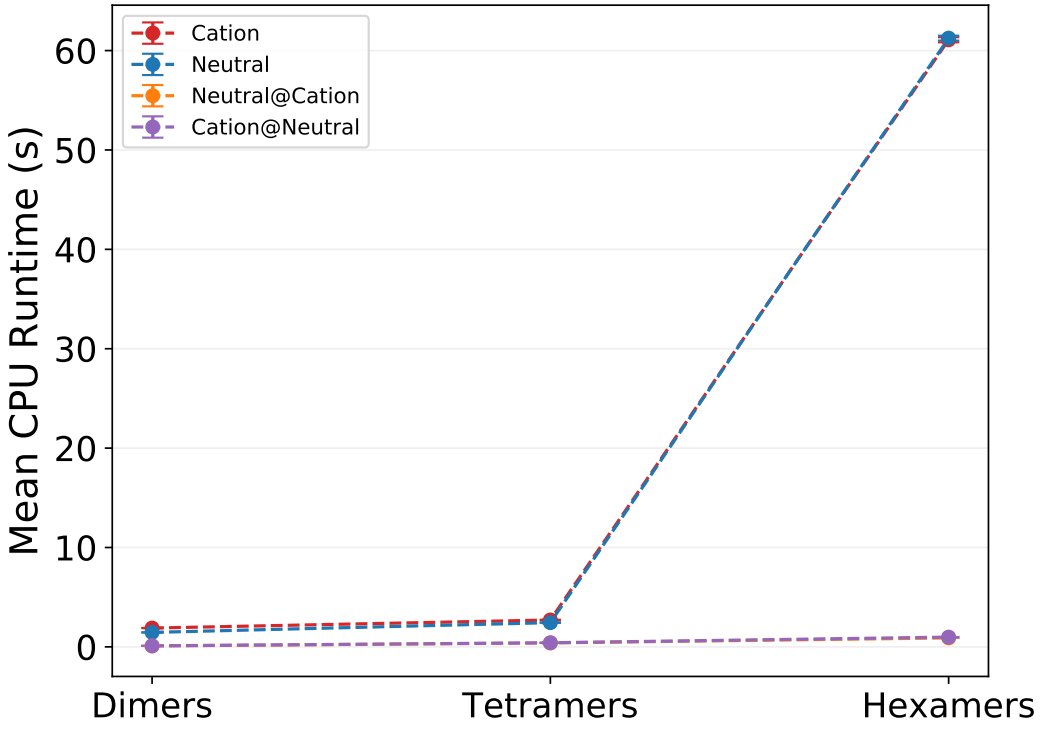


FIG. S7: Mean run time for each of the 4 calculations for the dimers, tetramers, and hexamers using GFN2.

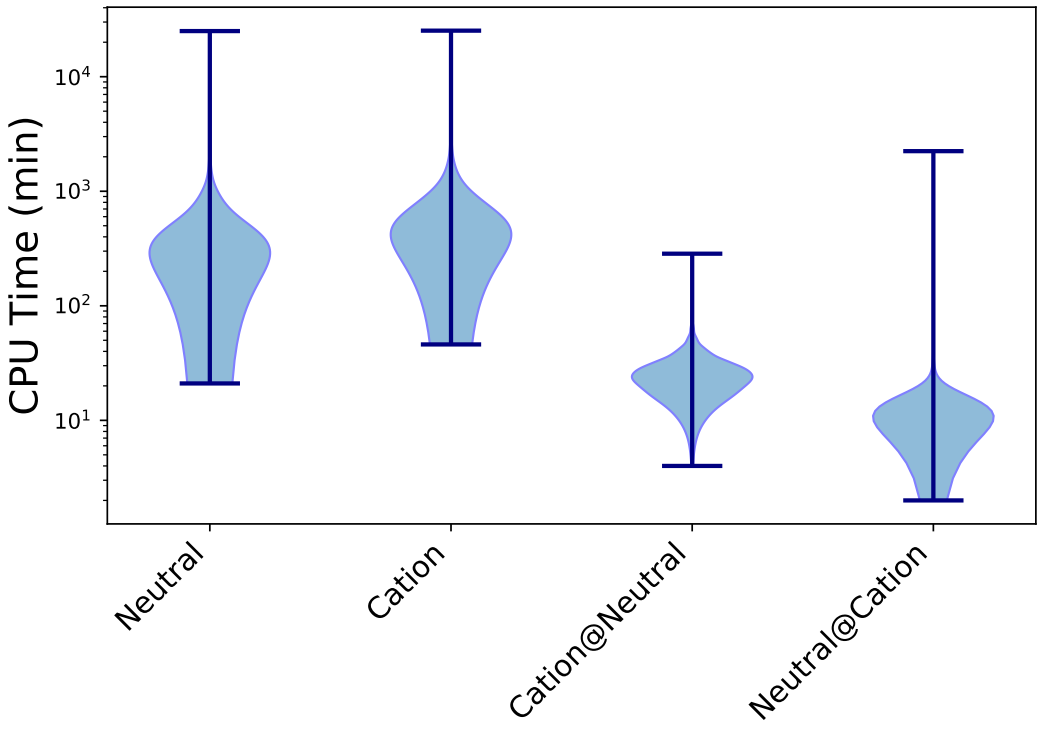


FIG. S8: Calculation run time of the 4 different calculation for the tetramers using B3LYP. Note the logarithmic y-axis.

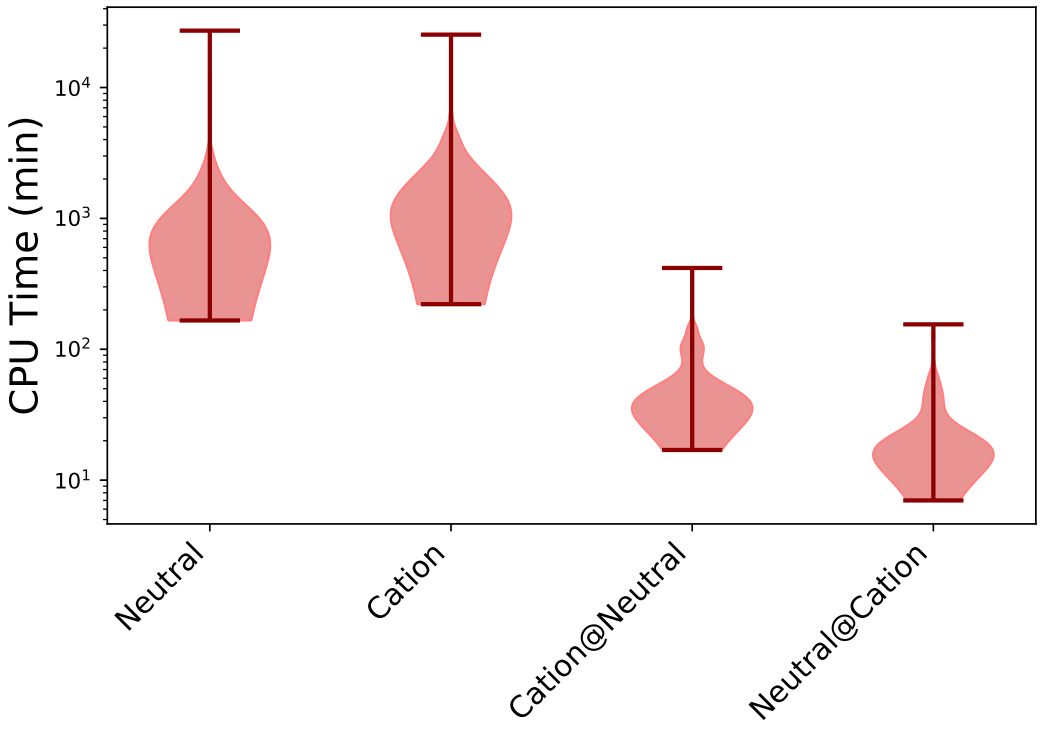


FIG. S9: Calculation run time of the 4 different calculation for the hexamers using B3LYP. Note the logarithmic y-axis.

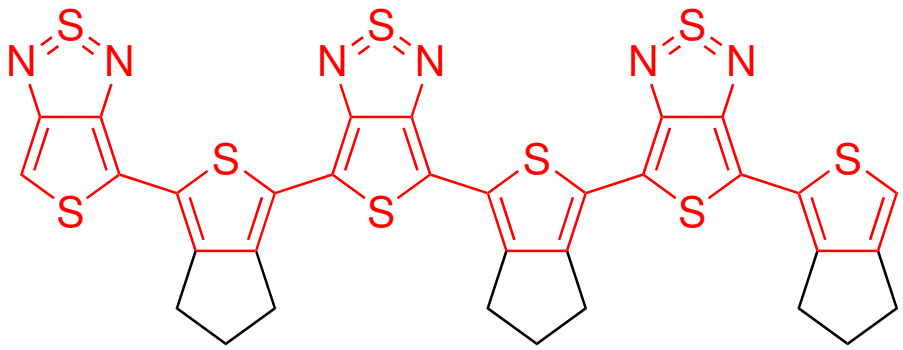


FIG. S10: Example for the PiSystemSize feature, which counts the number of atoms in the longest continuous conjugated π-system. In this example of the hexamer of monomers 31 (cyclopentathiophene) and 47 (thiadiazolthiophene) - the 39 highlighted atoms in red are counted.

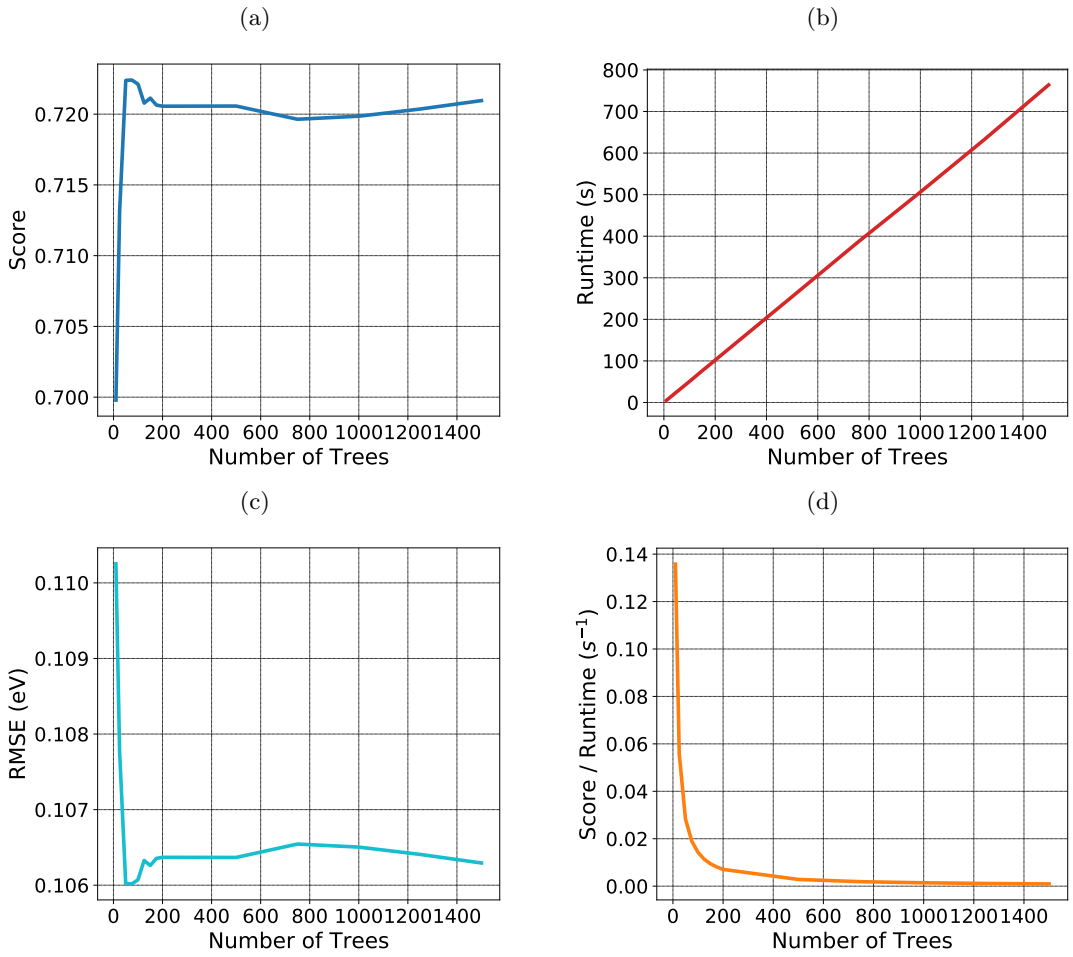


FIG. S11: Random Forrest regression optimization, **(a)** score vs. number of trees, **(b)** run time vs. number of trees, **(c)** RMSE vs. number of trees, **(d)** score/run time vs. number of trees.

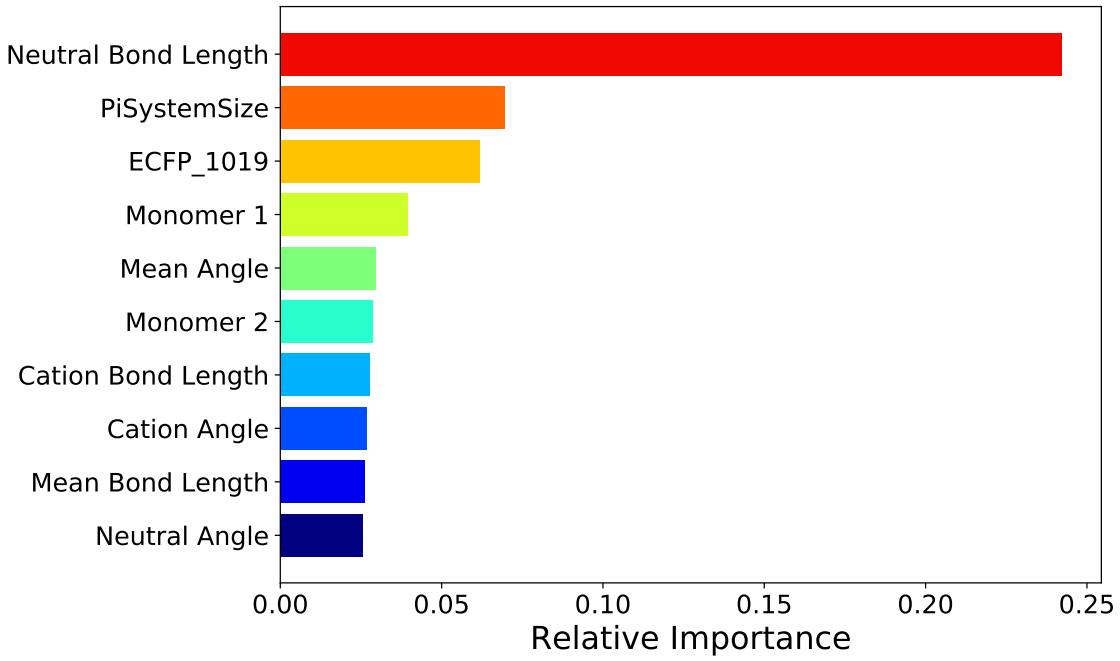


FIG. S12: Relative feature importance of the top 10 features in the random forest model.

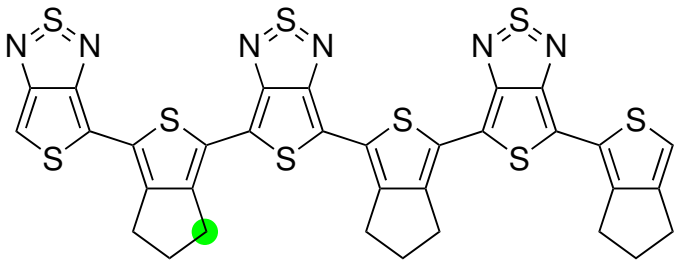


FIG. S13: Example of the ECFP bit number 1019 which indicates the existence of an sp^3 hybridized carbon in the oligomer.

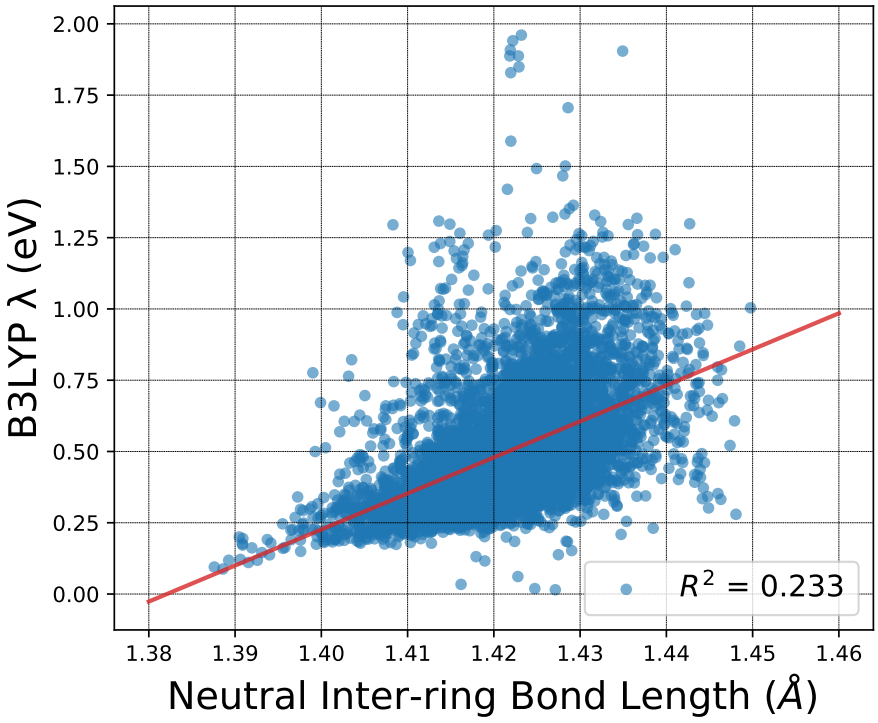
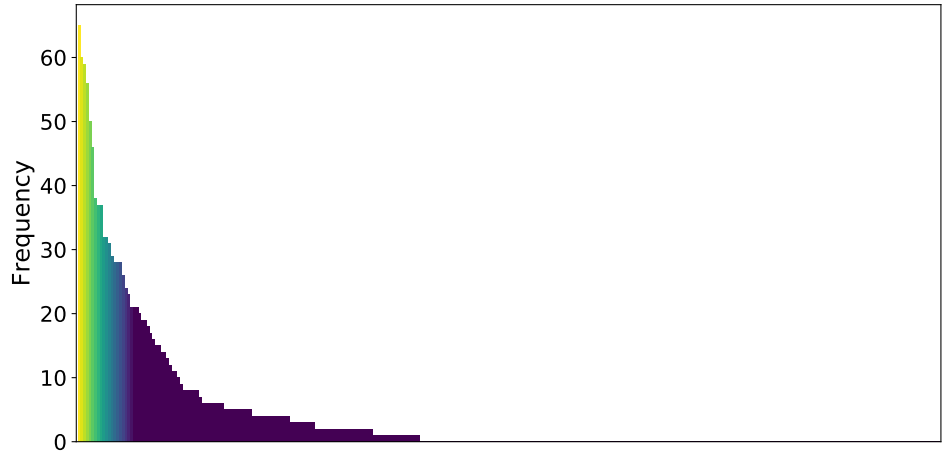


FIG. S14: Correlation between the average neutral inter-ring bond length of the oligomers versus the B3LYP calculated λ.

(a)



(b)

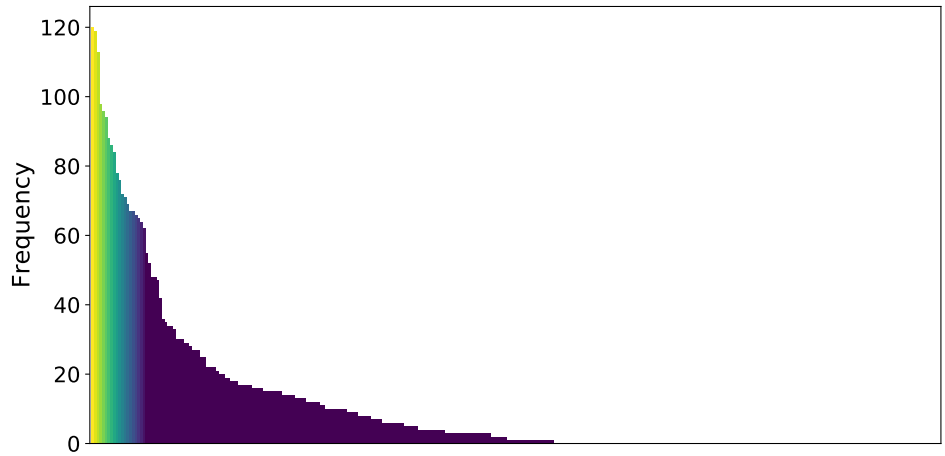


FIG. S15: Histogram of monomers, sorted by frequency, for (a) tetramers and (b) hexamers with $\lambda < 0.3$ eV illustrating that only a small number of monomers are found frequently (compare to sorting by arbitrary monomer number in **(a)** 5c, and **(b)** 5d).

Monomer 1	Monomer 2	GFN2 Neutral Dihedral Angle (°)	GFN2 Cation Dihedral Angle (°)	B3LYP Neutral Dihedral Angle (°)	B3LYP Cation Dihedral Angle (°)	GFN2 Neutral Bond Length (Å)	GFN2 Cation Bond Length (Å)	B3LYP Neutral Bond Length (Å)	B3LYP Cation Bond Length (Å)
47	47	179.308	179.371	179.999	179.994	1.381	1.378	1.379	1.374
47	116	179.693	179.727	178.576	178.674	1.390	1.385	1.397	1.389
47	156	177.520	177.935	179.979	179.997	1.388	1.383	1.397	1.387
47	247	179.201	178.849	178.099	179.726	1.388	1.383	1.401	1.390
47	217	179.460	179.493	175.533	179.994	1.393	1.387	1.403	1.391

TABLE S2: The monomer numbers, the predicted and calculated B3LYP λ, the dihedral angles of the neutral and cation species, and the inter-ring bond length of both neutral and cation species for the 5 hexamers with the lowest B3LYP λ.