Machine Learning Classification can Significantly Reduce the Cost of Calculating the Hamiltonian Matrix in CI Calculations

Chen Qu,[*,†] Paul L. Houston,[*,‡] Qi Yu,[*,¶] Riccardo Conte,[§] Priyanka Pandey,[¶] Apurba Nandi,[¶,‖] and Joel M. Bowman[*,¶]

†*Independent Researcher, Toronto, CA*

‡*Department of Chemistry and Chemical Biology, Cornell University, Ithaca, New York 14853, U.S.A. and Department of Chemistry and Biochemistry, Georgia Institute of Technology, Atlanta, Georgia 30332, U.S.A*

¶*Department of Chemistry and Cherry L. Emerson Center for Scientific Computation, Emory University, Atlanta, Georgia 30322, U.S.A.*

§*Dipartimento di Chimica, Università degli Studi di Milano, via Golgi 19, 20133 Milano, Italy*

‖*Department of Physics and Materials Science, University of Luxembourg, L-1511, Luxembourg City, Luxembourg*

E-mail: szquchen@gmail.com; plh2@cornell.edu; qyu28@emory.edu; jmbowma@emory.edu

Abstract

Hamiltonian matrices in electronic and nuclear contexts are highly compute-intensive to calculate, mainly due to the cost for the potential matrix. Typically these matrices contain many off-diagonal elements that are orders of magnitude smaller than diagonal elements. We illustrate that here for vibrational H-matrices for H_2O, C_2H_3 (vinyl) and $C_2H_5NO_2$ (glycine) using full-dimensional *ab initio*-based potential surfaces. We then show that many of the these small elements can be replaced by zero with small errors of the resulting full set of eigenvalues, depending on the threshold value for this replacement. As a result of this empirical evidence, we investigate three machine learning approaches to predict the zero elements. This is shown to be successful for these H-matrices after training on a small set of calculated elements. For one vinyl and glycine H-matrices, of order 15 552 and 8 828, respectively, training on a percent or so of elements is sufficient to obtain all eigenvalues with a mean absolute error of roughly 2 cm^{-1}.

The most general quantum approach to obtain accurate energies and wavefunctions of bound states of electrons (in electronic structure theory) and nuclei (vibrational dynamics) is "configuration interaction" (CI). This approach typically leads to a large Hamiltonian matrix, the cost of which to obtain scales exponentially or factorially with the size of the underlying basis. This bottleneck has stimulated numerous approaches to reduce the size of the basis, which of determines the size of the H-matrix. For approaches in electronic structure, see for example refs. 1–4. Also note the recent "heat-bath" CI approach,[5,6] which has also been using in vibrational CI, denoted VCI.[7,8]

Here we focus on vibrational dynamics, which has some prominent differences with electron motion; perhaps the biggest one is the need to obtain hundreds to thousands of eigenvalues from nuclear CI compared to a much smaller number in electronic CI.

To begin, recall that in the widely-used direct-product (Hartree) basis for f-modes the H-matrix is given by

$$\langle n'_1, n'_2, ..., n'_f|H|n_1, n_2, ..., n_f\rangle, \tag{1}$$

where the quantum numbers are the usual ones for this basis. For example, these could be harmonic-oscillator functions in normal coordinates of a potential minimum. More sophisticated 1-mode bases are used currently, but the details of these are not relevant. The interested reader can consult relevant chapters in a recent book on computational approaches to vibrational dynamics.[9] In applications, the values of these quantum numbers for each 1-mode basis is restricted. In addition, the sum of quantum numbers of all modes is typically bounded. This is probably the most common method to restrict the basis and goes back 40-50 years.[10,11] More fine-grained restriction methods are used in the code MULTIMODE,[12–14] where the excitation space is restricted, as in electronic structure methods, to "singles", "doubles", "triples", "quadruples", etc. Then separate excitation bounds are applied to each level of excitation. Also, when applicable, point-group symmetry is also used to block-diagonalize the H-matrix.

The above (and perhaps other) approaches all reduce the basis compared to an unre-

stricted basis. And this does result in a major reduction in the size of the H-matrix. We might refer to this as a basis-restricted H-matrix. It is possible to go further and directly eliminate elements of the matrix. This has been done, albeit rarely, in the vibrational dynamics literature. One example uses a method suggested by Handy and Carter[15] and used for vibrational energies of C_2H_4 and CH_3NO_2.[16] The method is based on second-order perturbation theory to eliminate elements and borrows from similar ideas in electronic structure theory. A direct approach has been given in ref. 17 as a means to reduce the storage of the H-matrix, illustrated for C_2H_4. This is simply to neglect Hamiltonian matrix elements below an absolute value of about 0.00002 cm^{-1}. Doing so resulted in a major reduction is the number of elements stored in memory.

While it may be possible to exploit the presence of very small matrix elements in memory storage and even reduce the size of the matrix, it still requires the calculation of the H-matrix. Clearly, it would be advantageous if the determination of very small elements could be done without calculating them. Here we report several machine learning classification approaches to achieve this. This is illustrated for H_2O, C_2H_3 (vinyl), and $C_2H_5NO_2$ (glycine).

To proceed, we give histograms of five H-matrix elements for these molecules obtained, with one exception, using MULTIMODE, which uses the exact Watson Hamiltonian with a direct product basis and the n-mode representation[18,19] of full-dimensional *ab initio* potential energy surfaces. Potential matrix elements were calculated using optimized numerical quadrature. In one case, 4-mode vinyl, the H-matrix was obtained using a direct-product of numerical basis functions in the normal coordinates of the saddle-point separating the two equivalent minima in vinyl. Details of the steps used in these calculations have been given previously for H_2O,[12] vinyl,[20,21] and glycine[22] and the interested reader if referred to those references for details.

First, consider the H-matrix for H_2O using a high-level potential.[23] Restriction of the basis was done using an excitation space up to triples. Point group symmetry was also used. The resulting H-matrix consists of two blocks (A_1 and B_2) of orders 161 and 125,

respectively. (Even with these small matrices, the eigenvalues for the fundamentals and the bend overtone are within less than 1.0 cm^{-1} of "exact" results.) A histogram of the matrix elements for the A_1 block is given in Figure 1. As seen many elements are small; 1 192 of the 13 041 elements are less than 10^{-10} hartree (0.00002 cm^{-1}) in magnitude and so can be classified as negligible. Of course many more elements are also small, e.g., 7 310 have magnitude less than 10^{-5} hartree (2 cm^{-1}). Whether these can classified as negligible, i.e., assigned the value zero, or not requires quantitative investigation and that is done after histograms for vinyl and glycine are given.

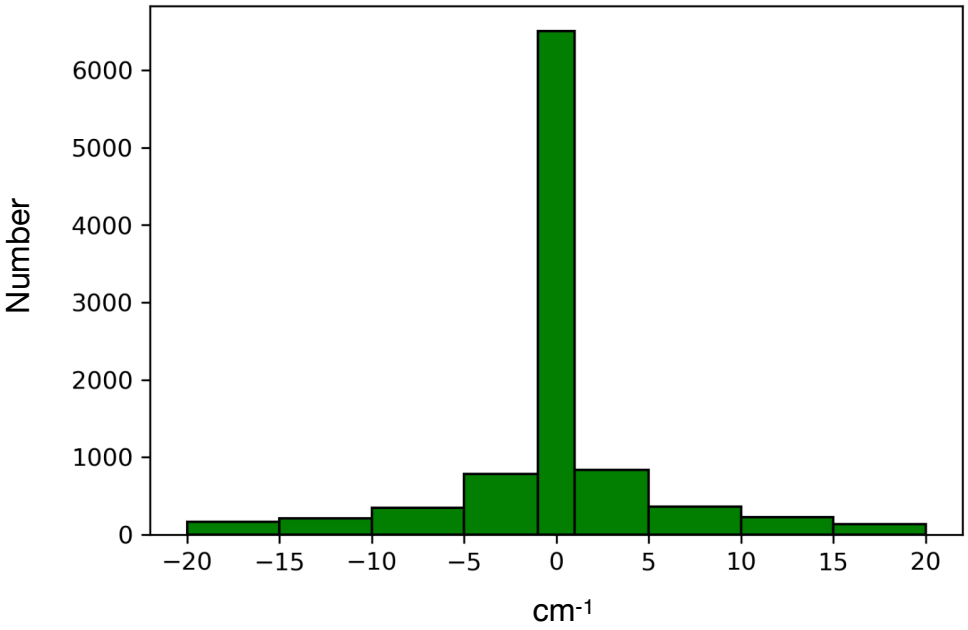

Figure 1: Histogram of H-matrix elements in cm^{-1} for the A_1 symmetry block of for H_2O. See text for more details.

Next consider the histogram of H-matrix elements from MULTIMODE for 5-mode vinyl, shown in Figure 2, using an accurate *ab initio*-based potential.[24] Note we used an option in MULTIMODE to consider a subset of modes. In this case 5 modes and a quintuples excitation space. As seen the vast majority of elements have a magnitude of order 0.01 cm^{-1}.

The final histogram we show is for 10-mode glycine in Fig. 3, where it appears that the

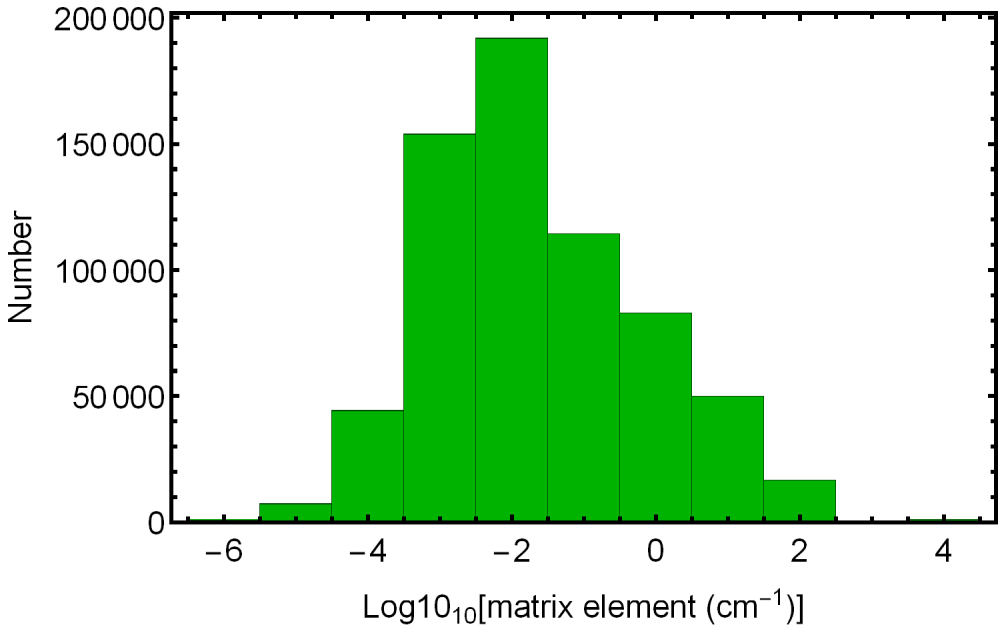

Figure 2: Histogram of H-matrix elements in hartree for 5-mode vinyl on a semi-log scale

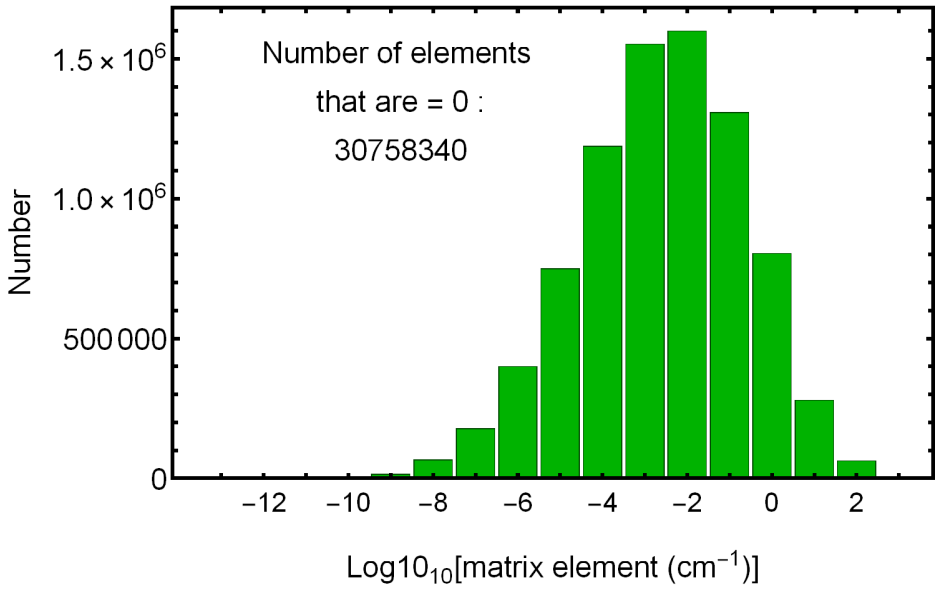

Figure 3: Histogram of H-matrix elements for the A'' symmetry block for 10-mode glycine on a semi-log scale

vast majority of elements are either very small or exactly zero. The "exact-zeros" occur only for 10-mode glycine among the five H-matrices considered here. We explain this below.

To proceed we determine the quantitative relationship between the magnitude of matrix elements that are be set to zero, based on a threshold, and the resulting mean absolute error (MAE) of all the eigenvalues for each H-matrix. (The number is the same as the order of the matrix and this is given for convenience in the second column of Table 1.) This relationship is shown in Figure 4 for five H-matrices. The two additional ones are for 3-mode and 4-mode vinyl. The latter one is the exception mention above that was not obtained with MULTIMODE. It was obtained used saddle-point normal modes, following earlier work.[21] As seen, and as expected, the fraction of elements set to zero and the MAE of all eigenvalues increases with increasing threshold. For the threshold of 10^{-5} hartree (2.2 cm^{-1}) the MAE is at most 1 cm^{-1}. Also at that threshold the percentage of zeroed elements increases from about 25 to 90+ percent. For glycine with a percentage just below 100 %, the histogram includes numerous elements that are set to zero in MULTMODE without calculating them. Glycine is the only example where this occurs and as noted earlier we provide the explanation below, where we examine the n-mode representation of the potential.

We now describe the machine learning classification (MLC) approaches we use to predict zero matrix elements. This is a standard 0/1 classification problem: in the training set, elements with a magnitude greater than the threshold are labelled 1, and elements with smaller magnitudes are labelled 0. The ML model then will predict 0 or 1 for an unknown matrix element; if it predicts 0, the matrix element can be set to zero so the computation can be skipped, and otherwise if the prediction is 1, the actual value matrix element will be calculated.

For demonstration purposes, we use a threshold of 1×10^{-5} hartree, 2.2 cm^{-1}, as this achieves good accuracy in the eigenvalues and considerable savings in the computations. Using this threshold, we label H-matrix elements as 0 or 1 to create a binary dataset that can be used in machine learning classification. The features ($\boldsymbol{X}$) for this classification prob-

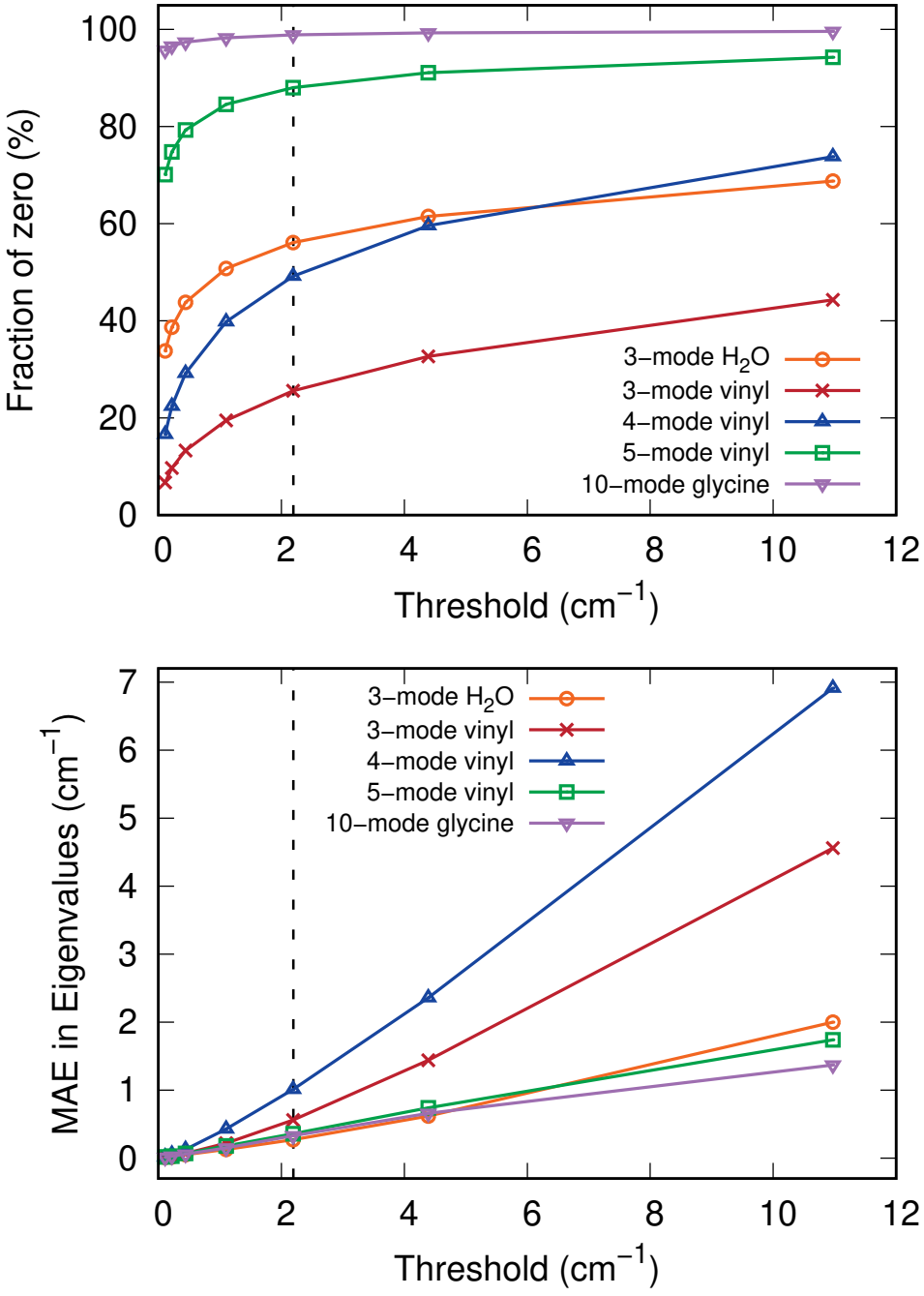

Figure 4: Fraction of matrix elements classified as zero as a function of the threshold for the 5 H-matrices studied in this work (upper); mean-absolute-error (MAE) of all eigenvalues as a function of the threshold (lower). The dashed line shows the threshold we use for the ML classification, 1×10^{-5} hartree (2.2 cm^{-1}).

lem are the quantum numbers of the 1-mode basis in *bra* and *ket*, $\langle n'_1, n'_2, ..., n'_{nmode}|$ and $|n_1, n_2, ..., n_{nmode}\rangle$. We investigate three widely used classification algorithms on the water H-matrix: random forest (RF), support vector machine (SVM), and multilayer perceptron (MLP). All ML models were trained using scikit-learn package.[25]

First consider the smallest H-matrix (of order 161) for water, with distribution of elements shown in Fig. 1. There are 13 041 matrix elements. These are randomly split into 40% training and 60% testing. For MLP and SVM models, the feature $\boldsymbol{X}$ is standardized to zero mean and unit variance by using the `StandardScaler` from scikit-learn, while for the RF model, we did not scale the features because it is invariant to feature scaling. The RF uses 20 estimators (also known as "decision trees") with maximum depth of 10. The MLP model uses two hidden layers with 15 neurons in each, and `'relu'` as the activation function. Early stopping is used to mitigate overfitting when the validation score (which is based on 20% of the training data, set aside as the validation set) does not improve for 10 consecutive training epochs. The SVM model uses C-Support Vector with radial basis function (RBF) kernel.

For 5-mode and 3-mode vinyl the H-matrices are of 1 152 and 1 960, respectively, so the numbers of matrix elements are 664 128 and 1 921 780. The SVM algorithm implemented in scikit-learn becomes too expensive, so only RF and MLP classifiers are considered. In these two cases, only 10% of the data set is used as the training data. This is produces sufficient data to train on. And, more importantly since the goal of the ML approach is to accurately predict negligible matrix elements and thus avoiding calculating them, we would like to keep the training set as small as possible. For these matrices the MLP classifier uses 20 neurons in each hidden layer and The RF classifier use 50 estimators with maximum tree depth of 20.

Moving on to much larger H-matrices, consider 4-mode vinyl using a saddle-point normal-mode basis and the B_2-symmetry block of the 10-mode glycine. The respective H-matrices are order 15 552 and 8 828. Thus there are tens or hundreds of millions of matrix elements,

and for vinyl each matrix element requires 4D quadrature. Note, for glycine, although we included 10 normal modes in the basis, we used 4-mode representation of the potential so the highest dimension for the numerical quadrature of potential matrix elements is still 4D. For glycine, as explained in detail below, because the number of modes is larger than than 4, many matrix elements are known to be exactly zero and so not calculated. In these two cases, only 1% of the matrix elements are used to train the ML models. The MLP models for these two examples have three hidden layers of 20 neurons, while the RF models use 50 estimators with maximum depth of 25. For 4-mode vinyl that uses the saddle-point normal-mode basis, the H-matrix is of order 15 552 and only the MLP classifier is used, since the random forest also becomes too expensive.

Now we present the training and testing accuracy of the ML models on the five H-matrices in Table 1. It can be seen that all ML models give accurate predictions for the labels, with an accuracy close to or greater than 90%. The random forest models slightly overfit the training set, indicated by the difference in the training and testing accuracy.

Table 1: Training/testing accuracy of the ML predicted labels of the matrix elements for the five H-matrices.

H-matrix	Order	SVM	RF	MLP
H_2O	161	0.891 / 0.885	0.963 / 0.933	0.889 / 0.882
5-mode vinyl	1 152	-	0.994 / 0.970	0.966 / 0.963
3-mode vinyl	1 960	-	0.984 / 0.93	0.915 / 0.914
10-mode glycine	8 828	-	0.999 / 0.992	0.994 / 0.993
4-mode SP vinyl	15 552	-	0.975 / 0.879	0.872 / 0.872

Next we reconstruct the H-matrices using the predicted labels from various ML models. If the predicted label for a matrix element is 1, we keep the exact value of that element, and if the predicted label is 0, we simply set that matrix element to zero. Then we diagonalize the reconstructed H-matrices and compare the resulting eigenvalues with the exact eigenvalues of the exact H-matrices. In Table 2, we present the mean-absolute-error (MAE) of all the eigenvalues. We can see that the MAE of the eigenvalues are about 1–2 cm^{-1}, except for

the RF model on glycine. This is probably due to the very unbalanced 0/1 label in this H-matrix. When using a threshold of 10^{-5} hartree, 98.9% of the matrix elements are labelled 0, so a small amount of wrong predictions may cause relatively large error. Nevertheless, this level of accuracy in the eigenvalues is satisfactory compared to the the errors caused by the underlying electronic structure theory, fitting precision, etc.

Table 2: Mean-absolute-error (in cm^{-1}) in the eigenvalues of the approximate H-matrices based on the predicted labels from different ML models on the matrix elements.

H-matrix	SVM	RF	MLP
H_2O	1.12	0.67	1.42
5-mode vinyl	-	1.31	0.84
3-mode vinyl	-	0.94	1.37
10-mode glycine	-	7.37	1.62
4-mode SP vinyl	-	2.23	2.17

Finally, we present the timings for training these ML models and make predictions on the matrix elements. The goal of pruning the matrix elements is to save computer times for "zeros", so the training and prediction of the ML models must at least be more efficient than actually computing those matrix elements. We only test the timings on the largest H-matrix, i.e., from 4-mode saddle point vinyl, as the matrix elements are expensive to compute in this example. The ML approach may not have the advantage if the matrix size is small to medium, since computing all matrix elements in a small or medium H-matrix is still fast, while the overheads of training ML models make this approach slower than simply computing all the elements. For this 4-mode vinyl H-matrix, training the RF model takes only 48 seconds and predicting the off-diagonal elements take 389 seconds, and the training and prediction of the MLP model takes 151 and 36 seconds, respectively. As a comparison, computing the H-matrix exactly takes hours, so the overhead of the ML model is small. The advantage of the ML approach will become more significant as the size of the H-matrix increases.

In summary, the ML classification approaches are quantitatively accurate for the H-

matrices examined here by training on just 1% of the elements with a major reduction on the number of predicted non-negligible elements to be calculated. For the largest examples, 4-mode vinyl with the saddle point basis and 10-mode glycine the reductions are around 90%. Also the results here show that the percentage reduction increases with the number of modes. Note the matrices considered here were not so large as to need special routines that can exploit sparse matrices. However, there are routines that do exploit this sparsity, both in terms of greatly reducing the memory needed to store the non-zero elements and also to obtain eigenvalues. Examples are scipy.sparse.linalg in Scipy[26] and sparse solver routines in the MKL library. Also, iterative routines that multiply the H-matrix by a trial vector clearly benefit from sparsity.

Figure 5 shows a flow chart of our proposed approach that would be implemented for a calculation of the H-matrix. As usual the starting point is the restricted basis for the calculation of the H-matrix.

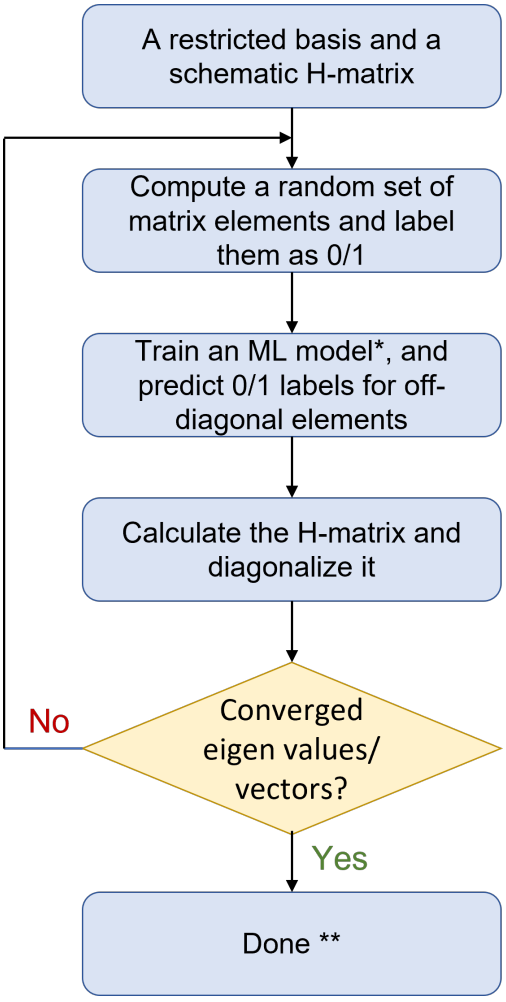

Figure 5: Flow chart of the machine-learning approach to prune the H-matrix elements. Note: * Could train committee ML models; ** Optionally could try other thresholds.

Finally, to provide insight into the large number of very small matrix elements consider the n-mode representation of the potential,[12,14,18] which some readers may know is used in MULTIMODE. In brief it represents the potential by

$$V(Q_1, Q_2, \cdots, Q_f) = \sum_i V_i^{(1)}(Q_i) + \sum_{i,j} V_{ij}^{(2)}(Q_i, Q_j) + \sum_{i,j,k} V_{ijk}^{(3)}(Q_i, Q_j, Q_k) + \sum_{i,j,k,l} V_{ijkl}^{(4)}(Q_i, Q_j, Q_k, Q_l) + \cdots, \tag{2}$$

where $V_i^{(1)}(Q_i)$ is the one-mode potential, i.e., the 1D cut through the full-dimensional PES in each mode, one-by-one, $V_{ij}^{(2)}(Q_i, Q_j)$ is the intrinsic 2-mode potential among all pairs of modes, etc. Here, intrinsic means that any n-mode term is zero if any of the arguments is zero. This representation is exact if there are f terms. In MULTIMODE this level goes up to a maximum of $V_{ijklmn}^{6)}$ terms. For H_2O, C_2H_3 with 3, 4, and 5 modes this expansion was taken to the level of being exact. For the VCI calculation using saddle point normal modes the full 4-mode potential was used without expansion as above. The 10-mode glycine example the 4-mode representation of the potential was used,[22] as many applications have indicated that this level produces eigenvalues that converged to with a few cm^{-1}. For the present work the accuracy of the representation is not relevant. However, the 10-mode glycine example is insightful. A general potential matrix is given by

$$\langle n_1', n_2', ..., n_{10}'|V|n_1, n_2, ..., n_{10}\rangle, \tag{3}$$

Using a 4-mode representation of V, it should be clear that many of these elements are exactly zero. And indeed MULTIMODE takes note of these and sets them to zero. Referring to the n-mode representation, it's clear that only by going to $V_{ijklmn...}^{(10)}$ would these analytical zeros disappear. If the n-mode representation is not used then the potential for 24-mode glycine (the actual number of vibrational modes) would be expressed as usual by $V(Q_1, ..., Q_{24})$ and the idea of doing general matrix would be truly infeasible and nonsensical. Clearly,

at a minimum the excitation space would have to be severely limited. However, even with the evaluating the matrix elements by even the most efficient quadrature methods would be prohibitive in 24 dimensions.[27] This fact notwithstanding, it does appear that the ML approach here will be powerful means to avoid calculating very small matrix elements. And, based on the empirical evidence given here and the discussion above, the number of elements increases rapidly with the number of degrees of freedom. Thus, the ML method proposed will become even more "needed" and effective as a means to deal with this rapid increase.

We note that non-direct product bases are also used in vibrational dynamics. These generally come from partitioning the full Hamiltonian into two sub-Hamiltonian in non-overlaping modes plus an interaction part. Diagonalization of the sub-Hamiltonians yields a new basis in the corresponding modes, which can then be used in new direct-product representation. Matrix elements in this new basis are then done and a final H-matrix is obtained. The goal here is to reduce the order of this matrix, however, a reasonable expectation is that there also many small elements in this matrix. Major examples of this approach are in refs. 28–30 It does appear that the machine learning classification approach described here for a direct-product basis should be directly applicable to non-direct product bases. That is work for the future.

Acknowledgement

JMB and PP acknowledge support from NASA grant (80NSSC22K1167).

References

(1) Sherrill, C. D.; Schaefer, H. F. In *The Configuration Interaction Method: Advances in Highly Correlated Approaches*; Löwdin, P.-O., Sabin, J. R., Zerner, M. C., Brändas, E., Eds.; Advances in Quantum Chemistry; Academic Press, 1999; Vol. 34; pp 143–269.

(2) Schriber, J. B.; Evangelista, F. A. Communication: An adaptive configuration interaction approach for strongly correlated electrons with tunable accuracy. *J. Chem. Phys.* **2016**, *144*, 161106.

(3) Coe, J. P. Machine learning configuration nteraction. *J. Chem. Theory Comput.* **2018**, *14*, 5739–5749.

(4) Mniszewski, S.; Dub, P.; Tretiak, S.; Anisimov, P. M.; Zhang, Y.; Negre, S.; A., C. F. Reduction of the molecular hamiltonian matrix using quantum community detection. *Sci Rep* **2021**, *11*, 4099.

(5) Holmes, A. A.; Tubman, N. M.; Umrigar, C. J. Heat-Bath Configuration Interaction: An Efficient Selected Configuration Interaction Algorithm Inspired by Heat-Bath Sampling. *J. Chem. Theory Comput.* **2016**, *12*, 3674–3680.

(6) Sharma, S.; Holmes, A. A.; Jeanmairet, G.; Alavi, A.; Umrigar, C. J. Semistochastic Heat-Bath Configuration Interaction Method: Selected Configuration Interaction with Semistochastic Perturbation Theory. *J. Chem. Theory Comput.* **2017**, *13*, 1595–1604.

(7) Fetherolf, J. H.; Berkelbach, T. C. Vibrational heat-bath configuration interaction. *J. Chem. Phys.* **2021**, *154*, 074104.

(8) Bhatty, A. U.; Brorsen, K. R. An alternative formulation of vibrational heat-bath configuration interaction. *Mol. Phys.* **2021**, *119*, 1936250.

(9) Bowman, J. M., Ed. *Vibrational Dynamics of Molecules*; World Scientific Press, 2022.

(10) Whitehead, R. J.; Handy, N. C. Variational calculation of vibration-rotation energy levels for triatomic molecules. *J. Molec. Spect.* **1975**, *55*, 356–373.

(11) Christoffel, K. M.; Bowman, J. M. Investigations of self-consistent field, SCF CI and virtual statec onfiguration interaction vibrational energies for a model three-mode system. *Chem. Phys. Lett.* **1982**, *85*, 220–224.

(12) Bowman, J. M.; Carter, S.; Huang, X. MULTIMODE: A code to calculate rovibrational energies of polyatomic molecules. *Int. Rev. Phys. Chem.* **2003**, *22*, 533.

(13) Carter, S.; Bowman, J. M.; Handy, N. C. Extensions and tests of "Multimode": A code to obtain accurate vibration/rotation energies of many-mode molecules. *Theor. Chem. Acc.* **1998**, *100*, 191–198.

(14) Yu, Q.; et al., In *Vibrational Dynamics of Molecules*; Bowman, J. M., Ed.; World Scientific Press, 2022.

(15) Handy, N. C.; Carter, S. Large vibrational variational calculations using 'multimode' and an iterative diagonalization technique. *Mol. Phys.* **2004**, *102*, 2201–2205.

(16) Wang, X.; Carter, S.; Bowman, J. M. Pruning the hamiltonian matrix in MULTIMODE: Test for C_2H_4 and application to CH_3NO_2 using a new *ab initio* potential energy surface. *J. Phys. Chem. A* **2015**, *119*, 11632–11640.

(17) Schröder, B.; Rauhut, G. In *Vibrational Dynamics of Molecules*; Bowman, J. M., Ed.; World Scientific Press, 2022.

(18) Carter, S.; Culik, S. J.; Bowman, J. M. Vibrational self-consistent field method for many-mode Systems: A new approach and application to the vibrations of CO adsorbed on Cu(100). *J. Chem. Phys.* **1997**, *107*, 10458–10469.

(19) Bowman, J. M.; Carrington, T.; Meyer, H.-D. Variational Quantum Approaches for Computing Vibrational Energies of Polyatomic Molecules. *Mol. Phys.* **2008**, *106*, 2145–2182.

(20) Sharma, A. R.; Braams, B. J.; Carter, S.; Shepler, B. C.; Bowman, J. M. Full-dimensional ab initio potential energy surface and vibrational configuration interaction calculations for vinyl. *J. Chem. Phys.* **2009**, *130*, 174301.

(21) Kamarchik, E.; Wang, Y.; Bowman, J. Reduced-dimensional quantum approach to tunneling splittings using saddle-point normal coordinates. *J. Phys. Chem. A* **2009**, *113*, 7556–7562.

(22) Qu, C.; Houston, P. L.; Conte, R.; Nandi, A.; Bowman, J. M. MULTIMODE calculations of vibrational spectroscopy and 1d interconformer tunneling dynamics in glycine using a full-dimensional potential energy surface. *J. Phys. Chem. A* **2021**, *125*, 5346–5354.

(23) Partridge, H.; Schwenke, D. W. The determination of an accurate isotope dependent potential energy surface for water from extensive *ab initio* calculations and experimental data. *J. Chem. Phys.* **1997**, *106*, 4618–4639.

(24) Chen, L.; Shao, K.; Chen, J.; Yang, M.; Zhang, D. H. Full-dimensional quantum dynamics study of the $H_2 + C_2H \rightarrow H + C_2H_2$ reaction on an *ab initio* potential energy surface. *J. Chem. Phys.* **2016**, *144*, 194309.

(25) Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V. et al. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* **2011**, *12*, 2825–2830.

(26) Virtanen, P.; Gommers, R.; Oliphant, T. E.; Haberland, M.; Reddy, T.; Cournapeau, D.; Burovski, E.; Peterson, P.; Weckesser, W.; Bright, J. et al. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods* **2020**, *17*, 261–272.

(27) Avila, G.; Carrington, T. Using multi-dimensional Smolyak interpolation to make a sum-of-products potential. *J. Chem. Phys.* **2015**, *143*, 044106.

(28) Zou, S.; Bowman, J. M.; Brown, A. Full-dimensionality quantum calculations of acetylene–vinylidene isomerization. *J. Chem. Phys.* **2003**, *118*, 10012–10023.

(29) Wang, X.-G.; Carrington, T. Computing excited OH stretch states of water dimer in 12D using contracted intermolecular and intramolecular basis functions. *J. Chem. Phys.* **2023**, *158*, 084107.

(30) Felker, P. M.; Bačić, Z. HF trimer: 12D fully coupled quantum calculations of HF-stretch excited intramolecular and intermolecular vibrational states using contracted bases of intramolecular and intermolecular eigenstates. *J. Chem. Phys.* **2023**, *158*, 234109.