JCTC Journal of Chemical Theory and Computation

Article

Cite This: J. Chem. Theory Comput. 2018, 14, 5739−5749

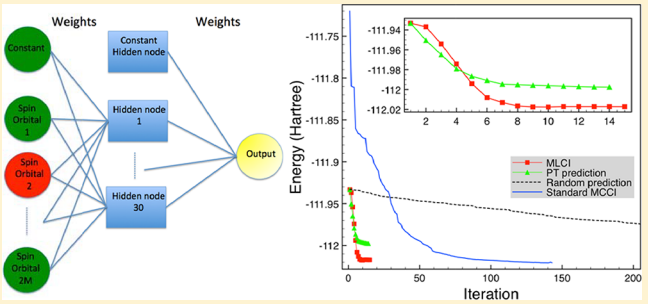
pubs.acs.org/JCTC

Machine Learning Configuration Interaction

J. P. Coe*

Institute of Chemical Sciences, School of Engineering and Physical Sciences, Heriot-Watt University, Edinburgh, EH14 4AS, United Kingdom

ABSTRACT: We propose the concept of machine learning configuration interaction (MLCI) whereby an artificial neural network is trained on-the-fly to predict important new configurations in an iterative selected configuration interaction procedure. We demonstrate that the neural network can discriminate between important and unimportant configurations, that it has not been trained on, much better than by chance. MLCI is then used to find compact wave functions for carbon monoxide at both stretched and equilibrium geometries. We also consider the multireference problem of the water molecule with elongated bonds. Results are contrasted with those from other ways of selecting configurations: first-order perturbation, random selection, and Monte Carlo configuration interaction. Compared with these other serial calculations, this prototype MLCI is competitive in its accuracy, converges in significantly fewer iterations than the stochastic approaches, and requires less time for the higher-accuracy computations.



1. INTRODUCTION

Machine learning has become increasingly popular and successful as a tool for quantum chemistry, partly due to the advent of graphical processing unit training for deep neural networks. Impressive applications have included training artificial neural networks, using energies from electronic structure calculations, to construct empirical potential energy surfaces for molecular dynamics, see, for example, ref 1 and references therein. Another approach is to choose molecular descriptors as inputs then train the machine learning algorithm on density-functional theory (DFT) data to predict quantum chemical properties, for example spin-state gaps of transition-metal complexes.[2] Interatomic distances and nuclear charges have been used as inputs in a deep tensor neural network that was trained to accurately predict energies for molecules as large as salicylic acid.[3] The correlation energy has also been predicted using machine learning for large sets of organic molecules.[4] Furthermore, corrections to the energy of a variant of second-order Møller−Plesset perturbation theory have been successfully demonstrated using deep neural networks with inputs from quantum chemistry calculations.[5]

At a more fundamental level of quantum chemistry calculation, a deep neural network has been developed and trained to predict the Kohn−Sham kinetic energy within DFT when using the density as an input.[6] A powerful machine learning approach was later developed[7] to predict the density directly from the potential for DFT calculations thereby avoiding the need to solve the Kohn−Sham equations. In ref 8, machine learning was used to predict exact ground-state energies for one-electron systems using their two-dimensional potentials. There the deep neural network approach was found to be more efficient and could achieve chemical accuracy.

Much of machine learning for quantum chemistry uses DFT to generate the training data. Although DFT is exact in principle, in practice it depends on the choice of the approximate functional. Elegant methods using coupled-cluster theory, CCSD[9] and CCSD(T),[10] offer more reliable accuracy for systems that can be well-described by small corrections to a single Slater determinant. However, their computational cost is greater than DFT and, in common with existing approximate functionals, they can perform very poorly when confronted with systems that have multiple important determinants. Such systems are often termed multireference problems and encompass stretched geometries, molecules containing transition metals, and excited states. The powerful approach of complete active space self-consistent field (CASSCF)[11] allows qualitatively correct results on multireference problems and can be followed by second-order perturbation (CASPT2)[12] or truncated configuration interaction (MRCI)[13] for quantitative accuracy. This, however, can come at a very high computational price and the results are now dependent on the choice of orbitals for the active space which can cause bias in, for example, the computation of potential energy surfaces.

By capitalizing on the common paucity of configurations that contribute substantially to the full configuration wave function (FCI), selected configuration interaction methods iteratively construct a compact wave function that can approach the energy of FCI but do not require the user to choose an active space. If we could efficiently select only the important configurations then we could quickly converge to a compact and accurate wave function for multireference

Received: August 14, 2018
Published: October 4, 2018

DOI: 10.1021/acs.jctc.8b00849
J. Chem. Theory Comput. 2018, 14, 5739−5749

problems. With this in mind, we take a different approach to machine learning in quantum chemistry by proposing an artificial neural network, trained on-the-fly, that predicts the important configurations for inclusion in a iterative selected configuration interaction calculation with the goal of improving accuracy and accelerating convergence.

Early work in selected configuration interaction (CI) included using first-order perturbation theory for the coefficients of the wave function to select new configurations (CIPSI),[14] or the contribution to the energy from perturbation theory to second order.[15] Later, the Monte Carlo configuration interaction (MCCI) method was developed,[16−18] which stochastically adds configurations to build up the wave function where configurations that are found to have absolute coefficients less than the control parameter (c_{min}) are removed. Another approach is to assume a structure of the wave function and then use the coefficients from a truncated CI calculation to estimate those of other configurations. For example, in ref 19, a structure similar to coupled cluster was used to predict which configurations could be neglected when the excitation level for the truncated CI calculation was increased. Furthermore, the @CC method[20] can be viewed as selected coupled cluster where the orbital importance in truncated CI determines the terms to include in the cluster operator.

Recently there has been a resurgence of interest in selected CI. For example, MCCI has been built upon and successfully applied to other challenges in quantum chemistry including crossings of potential curves for excited states,[21] molecular tunnel junctions formed of gold atoms,[22] hyperpolarizabilities,[23] perturbative corrections for the dissociation of diatomics,[24] X-ray absorption in multireference molecules,[25] and spin−orbit coupling.[26] The Λ-CI method, developed in ref 27, chooses configurations that are within a prescribed energy from the lowest energy configuration and was demonstrated to successfully calculate dissociation curves for N_2 and the carbon dimer. Adaptive configuration interaction (ACI) was later created[28] which uses both energy and coefficients as criteria to select configurations and was shown to give good agreement with DMRG results for singlet−triplet splittings in acenes. ACI was then built upon to give accurate results for excited states.[29] Adaptive sampling CI (ASCI)[30] improves upon CIPSI by only generating single and double substitutions from configurations with the largest coefficients. Instead of using the full expression of first-order perturbation to select configurations as in CIPSI, heat-bath CI[31] employs a simpler quantity and its impressive results are in agreement with DMRG for the chromium dimer. The MCI3 method was created in ref 32 by using projector or diffusion Monte Carlo in configuration space[33,34] to approximate the first-order perturbative corrections to the wave function. These are then used to guide the selection of new configurations thereby allowing larger systems to be considered than for CIPSI. MCI3 was applied to the ground and excited states of the carbon dimer where it gave accurate potential curves when compared with FCI but used a very small fraction of the configurations. CIPSI has also been recently used to create trial wave functions for diffusion Monte Carlo calculations that give the correct ground state of FeS.[35]

In this Article, we first discuss the methods employed beginning with the artificial neural network, then the MCCI program which is used as the framework for the other selected-CI approaches. This leads in to the description of machine learning configuration interaction (MLCI) which uses the artificial neural network trained on-the-fly to select important configurations. We also describe replacing the neural network predictions with predictions from first-order perturbation or random prediction. The accuracy of the neural network predictions are then investigated on stretched carbon monoxide in the 3-21G basis and this multireference system is then used to compare final energies, iterations to convergence and timings for the four selected-CI approaches. We then consider the molecule at its equilibrium bond length to assess the prototype MLCI method on a problem that is not multireference. Finally, we check that the form of the neural network can also work well for selected-CI calculations on other molecules by applying the methods to the water molecule with a cc-pVDZ basis and stretched bonds as an example of another multireference problem.

2. METHODS

2.1. Artificial Neural Network. The general form of the artificial neural network used is depicted in Figure 1 where we have n_i inputs plus a constant input, a single layer of n_h hidden nodes plus a constant hidden node, and one output. Hidden nodes are just nodes that are neither an input nor an output. For a basis set of size M, we use $2M$ inputs, corresponding to the spin orbitals, and one constant input for the neural network. An input is 1 if that spin orbital is occupied in the configuration of interest and 0 otherwise. After testing various numbers of hidden nodes, we settled on 30 hidden nodes plus one constant.

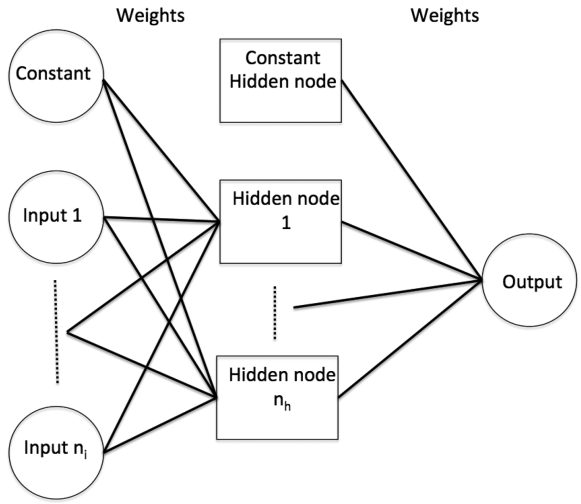


Figure 1. Schematic of the artificial neural network with n_i inputs plus a constant input, a single layer of n_h hidden nodes plus a constant hidden node, and one output.

We label the weights from input i to hidden node j as W_{ij}^{in}, while those from the hidden layer j to the output are labeled W_{j0}^{out}. The value of hidden node j is given on (0,1) by a logistic or sigmoid function

$$\text{hidden}_j = \frac{1}{1 + e^{-\sum_{i=0}^{n_i} W_{ij}^{in} \text{input}_i}} \quad (1)$$

where i runs from 0 to n_i as input node 0 is constant. The output is between zero and one, and is calculated by

$$\text{output} = \frac{1}{1 + e^{-\sum_{j=0}^{n_h} W_{j0}^{out} \text{hidden}_j}} \quad (2)$$

DOI: 10.1021/acs.jctc.8b00849
J. Chem. Theory Comput. 2018, 14, 5739−5749

Without a hidden layer, then finding the weights for a given set of input and output values would just be logistic regression which could be transformed to linear regression when the training output is on (0,1). However, a single hidden layer, in principle, allows the neural network to approximate essentially any function when sufficient nodes are used.[36] Many of the recent successful applications of neural networks have employed multiple hidden layers which are known as deep or convoluted neural networks. It is not clear whether this is because deep networks are less dependent on the form of the input or are easier to train with current techniques than using a single hidden layer with many nodes. Interesting work has shown that shallow neural networks can give similar accuracy when trained using the outputs of previously trained deep neural networks.[37] For this proof-of-concept work, we restrict the neural network to a single hidden layer.

We train the neural network to approximate the outputs of a training set of input and output values o_t using back-propagation with stochastic gradient descent, see, for example, ref 38. The weights are initially set to small random values then for each training set example the output is calculated and compared with the desired result o_t. Stochastic gradient descent is used to minimize the error

$$\text{error} = \frac{1}{2}(\text{output} - o_t)^2 \tag{3}$$

through updating the weights for every training example error rather than using gradient descent on the total error for the whole training set. By using this approach the chance of being trapped in a local minimum is hoped to be reduced as it is tantamount to introducing noise into gradient descent. When the errors are propagated back to update the weights, the change in the weights on each iteration is controlled by the learning rate parameter. This balances speed of training against likely accuracy. We use an initial learning rate of 0.1 that drops to 0.01 from the third iteration of machine learning configuration interaction.

Sufficient training of the neural network can require many passes through the entire training set. However, the neural network needs to perform well on unseen data not just the training set. So, to avoid very high quality results on the training set but very poor quality predictions for new data (overtraining), the neural network is applied to a verification set of values that it has not been trained on at each iteration. One technique is then to stop training the neural network once its error on the verification set begins to increase. For this work, the maximum number of passes through a training set is fixed at 2000 and we use the weights that give the lowest error on the verification set from these 2000 training passes.

We implement an artificial neural network for selected configuration interaction within the framework of the Monte Carlo configuration interaction program,[16−18] which we discuss next.

2.2. Monte Carlo Configuration Interaction. Monte Carlo Configuration Interaction (MCCI)[16−18] builds up a compact wave function by repeatedly randomly adding interacting configurations to the current set then diagonalizing the Hamiltonian matrix and eventually removing configurations whose absolute coefficient is less than the cutoff $c_{\min}$.

For MCCI in this work, we begin with the Hartree−Fock−Slater determinant and a brief description of the algorithm is presented as follows:

- Symmetry-preserving single and double substitutions are used to stochastically enlarge the current configuration space.
- The Hamiltonian matrix is constructed and diagonalized to give the energy and coefficients of the wave function.
- If a configuration is newly added but has an absolute coefficient less than $c_{\min}$ then it is deleted (pruned).
- All configurations in the wave function are considered for deletion (full pruning) every ten iterations.
- This process is iterated until the energy after a full pruning step satisfies a convergence criterion.[39]

The convergence threshold[39] is set to $c_{\min}$ here and uses the energy change between full pruning steps averaged over the last three full pruning steps and compares the maximum of the last three values of this with the threshold. For all the work in this Article, we use Slater determinants as the configurations and run the calculations in serial. The Hartree−Fock molecular orbitals are calculated using Molpro,[40] as are the one-electron and two-electron integrals.

2.3. Machine Learning Configuration Interaction. To facilitate the training of the neural network to predict important configurations in machine learning configuration interaction (MLCI) we transform the coefficients c_i of configurations to $|\tilde{c}_i|$ for the neural network. With the aim of making the difference between important and unimportant configurations more apparent. We set absolute coefficients on $[0,c_{\min})$ to zero and linearly map absolute coefficients on $[c_{\min},1]$ to $[0.6,1]$. This latter transformation is achieved using

$$|\tilde{c}_i| = \frac{0.4|c_i| + 0.6 - c_{\min}}{1 - c_{\min}} \tag{4}$$

We note that if the weights are all zero then the logistic function will give a value of 0.5 so we required a value greater than this and found that 0.6 as the threshold was satisfactory. These $|\tilde{c}_i|$ and their configurations are respectively the outputs and inputs for the training set.

The neural network requires initial data for training so the first step in MLCI is to add all single and double substitutions permitted by spatial symmetry to the Hartree−Fock wave function, that is, we create the CISD wave function.

Pruning is then implemented to remove new configurations in the wave function with $|c_i| < c_{\min}$. As all information on the importance of configurations is useful for training and we do not want the neural network to "forget" what an unimportant configuration looks like, we store pruned configurations as a reject set without duplicates and with zero coefficients. If a pruned configuration later becomes important it is removed from this set. That the neural network does not reproduce the training data perfectly may help the calculation: the training data contains a snapshot of the importance of configurations for the current wave function and so is an evolving approximation as the importance may change when configurations are included and removed. Hence just because a configuration is in the reject set at one point in the calculation does not mean it should never be added again.

The transformed coefficients and their configurations in both the wave function and the reject set are then shared randomly and equally between the training set and the verification set. The neural network is then trained with 2000 passes through the training set and the weights that give the lowest error on the verification set are used for the next step.

DOI: 10.1021/acs.jctc.8b00849
J. Chem. Theory Comput. 2018, 14, 5739−5749

The neural network is then applied to all single and double substitutions from the current wave function and, for a wave function of L configurations, adds the L configurations it considers to be most important, that is, they have the largest predicted coefficients. As the number of single and double substitutions from the current set can become very large we use the method of ref 41 to efficiently generate all single and double substitutions without duplicates. We note that the neural network's weights are random at the start of MLCI but then are retained between iterations. The neural network, in a sense, pulls itself up by its bootstraps as after the first iteration in MLCI it both predicts the configurations to be added and learns, on-the-fly, from the output of the diagonalization.

The MLCI procedure is summarized below where initially we construct the CISD wave function from the Hartree−Fock single determinant.

- Create and diagonalize the Hamiltonian matrix in the current configuration space.
- Newly added configurations with $|c_i| < c_{\min}$ are pruned and stored in the reject set. Every ten iterations the entire wave function is pruned (full prune).
- The neural network is trained to predict transformed coefficients $|\tilde{c}_i|$ by combining the reject set with the wave function and sharing these data equally and randomly between training and verification sets.
- All symmetry-allowed single and double substitutions without duplicates are efficiently generated[41] from the pruned wave function using the quicksort algorithm.
- For a wave function of L configurations the neural network then predicts the best L new configurations, i.e. those with the largest predicted $|\tilde{c}_i|$, to enlarge the current configuration space.
- This procedure is iterated until the energy satisfies a convergence criterion.[39]

We note that MLCI reaches convergence much sooner than MCCI so in this work MLCI looks at every iteration, not just every ten, to check for convergence. This ensures that the MLCI procedure is not repeated unnecessarily many times.

2.4. Prediction By First-Order Perturbation. We also consider first-order perturbation theory to select configurations, which we term PT prediction. This therefore uses the same approach for selection as CIPSI.[14] The procedure is that of MLCI except instead of using the neural network for the predictions we use the absolute coefficients $|c_I|$ in the first-order correction $\Psi^{(1)}$ to the current wave function $\Psi^{(0)}$. In this case

$$\Psi^{(1)} = \sum_I c_I |I\rangle \tag{5}$$

and

$$c_I = \frac{\langle I|\hat{H}|\Psi^{(0)}\rangle}{E^{(0)} - \langle I|\hat{H}|I\rangle} \tag{6}$$

where the $|I\rangle$ are all the symmetry-allowed single and double substitutions when duplicates are excluded. For real coefficients then the wave function to first-order is $\Psi^{(0)} + \Psi^{(1)}$, which we normalize. We then add the L configurations with the largest absolute c_I values to the current configuration space. We found that the calculation of the c_I is somewhat slower than applying the neural network in this work, but convergence is rapid so again we check for convergence on every iteration.

2.5. Random Prediction. To check that MLCI is doing better than random selection we finally replace the predicted values in the algorithm with random numbers on $[0,1]$. This random prediction is not the same as MCCI as it is created by modifying MLCI so generates the CISD wave function on the first iteration, and creates all single and double substituted configurations from the current wave function without duplicates then randomly selects L of these configurations. In contrast, MCCI uses random generation on the first iteration and attempts to add an adaptive number of configurations where the probability of single or double substitution is the same.[18] Hence random prediction enables a fairer comparison between the MLCI program and stochastic selection. As the convergence when using random prediction is rather slow then the convergence check only looks at full pruning steps as in MCCI, but unlike MLCI and PT prediction.

3. RESULTS

3.1. Stretched Carbon Monoxide. We initially test the MLCI approach on carbon monoxide with a stretched bond of 4 bohr, the 3-21G basis set and two frozen orbitals. This system, with four frozen orbitals, was shown to be strongly multireference in ref 42 when using the canonical Hartree−Fock orbitals. A cutoff of $c_{\min} = 10^{-3}$ is used within the MCCI framework when we first consider the ability of the neural network to predict important configurations. Where, as discussed in the Methods section, absolute coefficients on $[0,c_{\min})$ are set to zero and absolute coefficients on $[c_{\min},1]$ are linearly transformed to $[0.6,1]$ to improve the training of the neural network in discerning important configurations.

The neural network results from the first iteration of MLCI are presented in Figure 2 where we order the configurations in the verification set by the size of their transformed coefficients and compare these coefficients with the neural network predictions when values less than 0.6 are set to zero. At this early stage of the computation, the neural network improved its accuracy by running through the training set 772 times

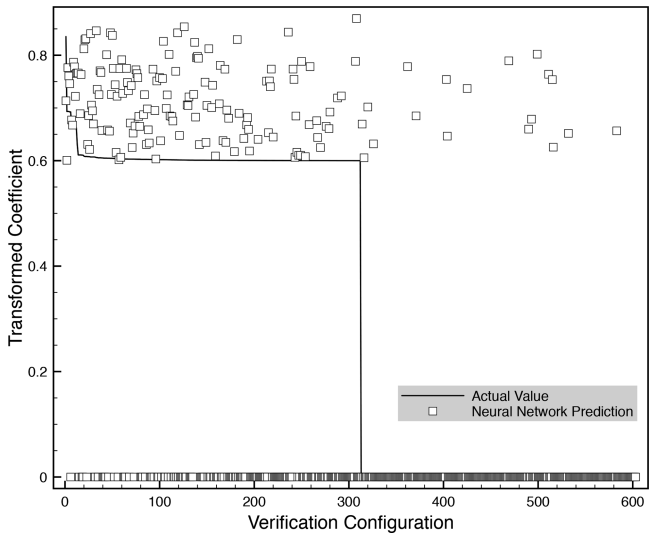


Figure 2. Transformed coefficients $|\tilde{c}_i|$ of the verification set configurations compared with the neural network predictions, when values below 0.6 are set to zero, for carbon monoxide using a bond length of 4 bohr, the 3-21G basis set, two frozen orbitals and a cutoff of $c_{\min} = 10^{-3}$ on the first iteration in MLCI.

DOI: 10.1021/acs.jctc.8b00849
J. Chem. Theory Comput. 2018, 14, 5739−5749

before its error on the verification set began to increase. The training set consisted of 603 configurations, and the verification set was the same size. At this point the root-mean-square error on the verification set was 0.27. The neural network can be seen (Figure 2) to be predicting the importance of the configurations fairly well although a number of important configurations are not identified and a few unimportant configurations are classified incorrectly.

The predictive ability of the neural network on the verification set from the first iteration of MLCI is quantified in Table 1. There we see that, at this point, the ability of the neural network to include important configurations has room for improvement as using

$$\text{sensitivity} = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}} \quad (7)$$

then the sensitivity, or true positive rate, is 46%. Yet we emphasize that the MLCI algorithm limits the number of configurations to be added using the size of the current wave function, so it does not necessarily matter if we miss some important configurations. At this early stage of developing MLCI we would be satisfied to find significantly more important configurations than would be expected by chance, particularly as this advantage will accumulate over multiple iterations. With regards to this the neural network is much better at not including unimportant configurations as using

$$\text{specificity} = \frac{\text{true negatives}}{\text{true negatives} + \text{false positives}} \quad (8)$$

then the specificity, or true negative rate, is 94%. We note that of the 162 configurations suggested to be included by the neural network, 144 (89%) are actually important. Around half of the verification set are important in the wave function so one would expect that if the 162 configurations were instead chosen randomly then only about 80 would be important.

We now plot the neural network predictions of the second iteration in Figure 3. In this case we use 0.51 rather than 0.6 as the threshold below which neural network predictions are mapped to zero in the figure. The MLCI algorithm includes the L configurations with the predicted largest coefficients where L is the number of configurations in the current wave function. Hence it is the relative importance of the predictions that matters and we can see from Figure 3 that with the lower threshold there are many true positives with only a small number of false positives. However, it can also be seen that there are many configurations that are important but not classified as such. For these results, the neural network weights were taken after 89 passes through the training set as after this the verification error began to increase. The root-mean-square error on the verification set is now 0.20 hence the neural network has continued to improve from the first iteration of MLCI despite the increase in the number of configurations in the verification set.

Table 1. Neural Network Importance Predictions[a]

	important	unimportant
predicted important	144	18
predicted unimportant	168	273

[a]Compared with whether the configuration is important in the verification set's transformed coefficients $|\tilde{c}_i|$ for the first iteration of MLCI for carbon monoxide using a bond length of 4 bohr, the 3-21G basis set, two frozen orbitals, a cutoff of $c_{min} = 10^{-3}$ and when values less than 0.6 are considered unimportant.

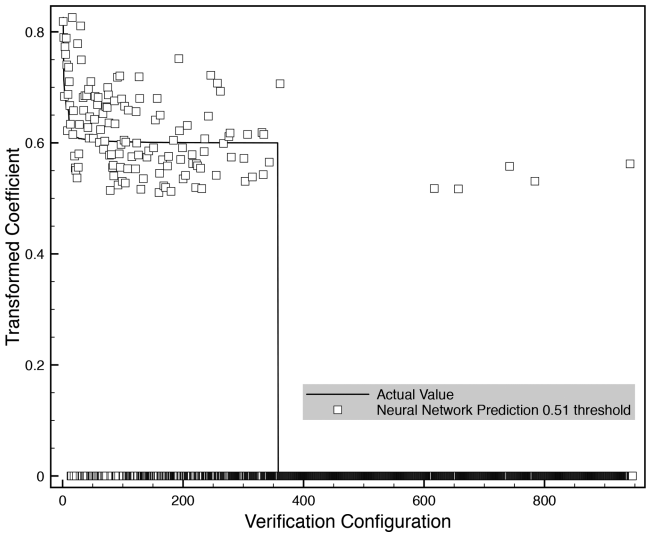


Figure 3. Transformed coefficients $|\tilde{c}_i|$ of the verification set configurations compared with the neural network predictions with a threshold of 0.51 for carbon monoxide using a bond length of 4 bohr, the 3-21G basis set, two frozen orbitals, and a cutoff of $c_{min} = 10^{-3}$ on the second iteration in MLCI.

This change in accuracy on lowering the threshold is quantified in Table 2, where we see that the number of false positives only increases from 1 to 5 but the number of true positive goes up from 72 to 129 as the threshold drops to 0.51. In this case, the sensitivity increases from 20.2% to 36.1%, while the specificity only slightly decreases from 99.8% to 99.0%. For the lower threshold, 135 configurations would be predicted to be important and 95.6% are indeed important in the current wave function. In the second iteration for MLCI, 37.9% of the verification set were important in the wave function so by picking 135 configurations randomly one would only expect around 51 to be important.

Table 2. Neural Network Importance Predictions[a]

	important	unimportant
predicted important (≥0.6)	72	1
predicted unimportant (<0.6)	285	588
Predicted Important (≥0.51)	129	6
predicted unimportant (<0.51)	228	583

[a]Compared with whether the configuration is important in the verification set's transformed coefficients $|\tilde{c}_i|$ on the second iteration of MLCI, when values less than 0.6 are considered unimportant for the verification set and 0.6 then 0.51 for the neural network. Results are for carbon monoxide using a bond length of 4 bohr, the 3-21G basis set, two frozen orbitals, and a cutoff of $c_{min} = 10^{-3}$.

We also consider varying the threshold for the neural network importance predictions from one to zero. The resulting ROC (receiver operating characteristic) curve is displayed in Figure 4 where sensitivity is plotted against 1−specificity. For small enough thresholds (less than 0.1) then the neural network can get over 90% sensitivity, but at the cost of reduced specificity. The ROC curve shows that the sensitivity could be increased to almost 40% without strongly affecting specificity and this corresponds to a threshold of 0.5. Importantly, for reasonable thresholds, the curve for the neural

DOI: 10.1021/acs.jctc.8b00849
J. Chem. Theory Comput. 2018, 14, 5739−5749

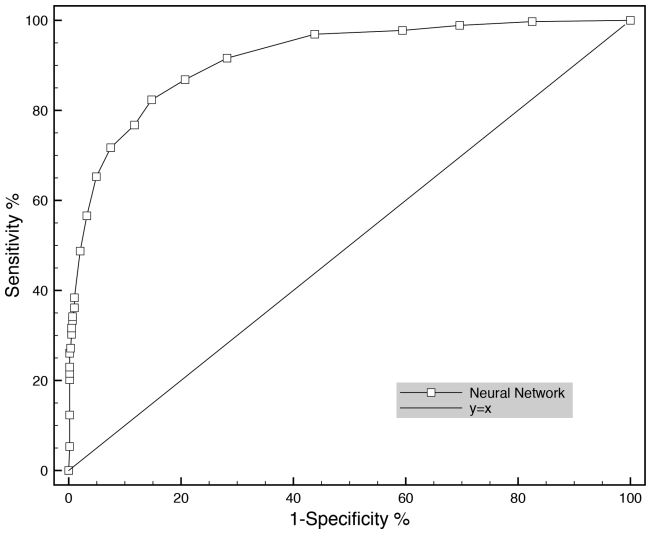


Figure 4. Sensitivity against 1−specificity for the neural network predictions with various thresholds from one to zero on the verification set for carbon monoxide using a bond length of 4 bohr, the 3-21G basis set, two frozen orbitals, and a cutoff of $c_{min} = 10^{-3}$ on the second iteration in MLCI.

network is significantly above the line $y = x$, which represents random selection.

3.1.1. Energy Calculations. We have seen that the neural network can perform well on verification sets, but the crucial test is its ability, when confronted with the much larger set of single and double substitutions, to accelerate the convergence of a selected configuration interaction calculation. Figure 5 shows that MLCI converges in around one tenth of the iterations required for MCCI when $c_{min} = 10^{-3}$ and the convergence tolerance for the energy is 10^{-3} Hartree. We note that all calculations are run in serial and by running MCCI in parallel then convergence would require fewer iterations.

Although the MLCI convergence required one more iteration than using PT prediction, the final energy is noticeably lower using MLCI (inset of Figure 5). As MCCI does not include all single and doubles on the first iteration then for a direct comparison of machine learning versus random additions we also modify MLCI so that the predictions from the neural network are replaced by random numbers. This random prediction approach is also presented in Figure 5 and we can see that, although the energy is similar at the start due to the addition of all singles and doubles on the first iteration, the convergence is very much slower than MLCI and also of MCCI. It could be that building up the wave function more slowly at the beginning is advantageous when randomly adding configurations. Perhaps as this reduces the chance of configurations that will eventually become unimportant being in the wave function and so the singles and doubles space can be smaller and also more likely to have important configurations. We note that MCCI has other differences with random prediction which could contribute to its faster convergence rather than just the number of configurations added on the first iteration. For example, there is an equal chance of adding a single or double substituted configuration in MCCI while in the singles and doubles space the doubles are much more numerous.

As we construct the three predictive approaches to always add the same number of configurations as in the current wave function, it is not the case that MLCI adds more configurations than PT or random prediction on an iteration. Rather the configurations suggested by the neural network are more likely to turn out to be important in the wave function. We see in Figure 6 that predicting important configurations randomly means that the size of the reject space increases almost linearly with the number of iterations as very many added configurations are pruned. In fact this causes the calculation to end as the size of the wave function and the size of the reject space are limited to 2×10^5 configurations. MLCI, in contrast, has fewer of its predicted configurations pruned from the wave

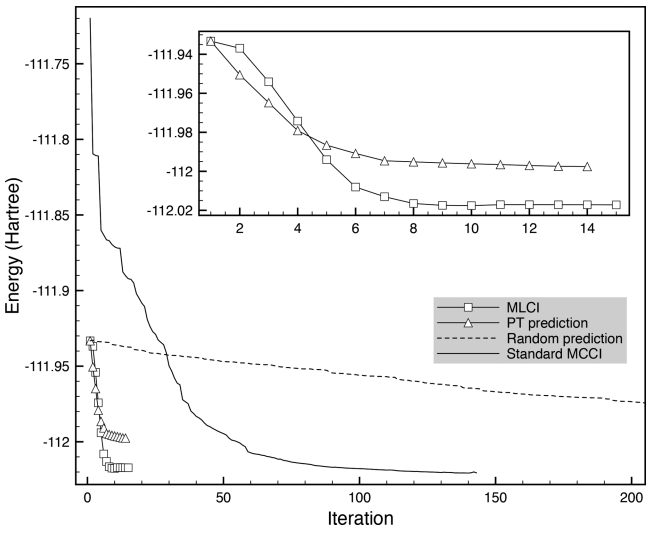


Figure 5. Energy (Hartree) against iteration number for serial calculations for carbon monoxide with a bond length of 4 bohr, the 3-21G basis set, two frozen orbitals, and a cutoff of $c_{min} = 10^{-3}$ using machine learning predictions to select configurations (MLCI), PT prediction, random prediction or standard MCCI. Inset: Enlargement of the MLCI and PT prediction curves.

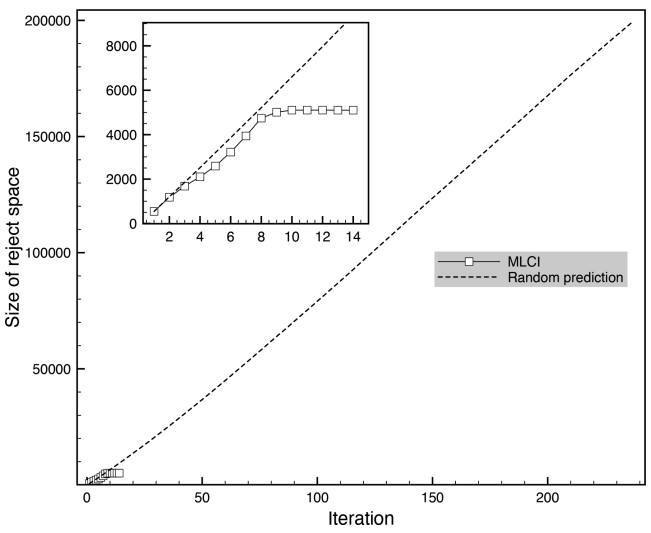


Figure 6. Size of the reject space against iteration number for serial calculations for carbon monoxide with a bond length of 4 bohr, the 3-21G basis set, two frozen orbitals, and a cutoff of $c_{min} = 10^{-3}$ using machine learning predictions to select configurations (MLCI) or random prediction. Inset: Enlargement of the results until convergence of MLCI.

DOI: 10.1021/acs.jctc.8b00849
J. Chem. Theory Comput. 2018, 14, 5739−5749

function on each iteration and as it converges the size of the reject space changes very little as the neural network will be predicting essentially the same configurations on each iteration if both the wave function and neural network have converged.

In Figure 7, we plot the energy against the number of configurations on each iteration. The curves can be rather

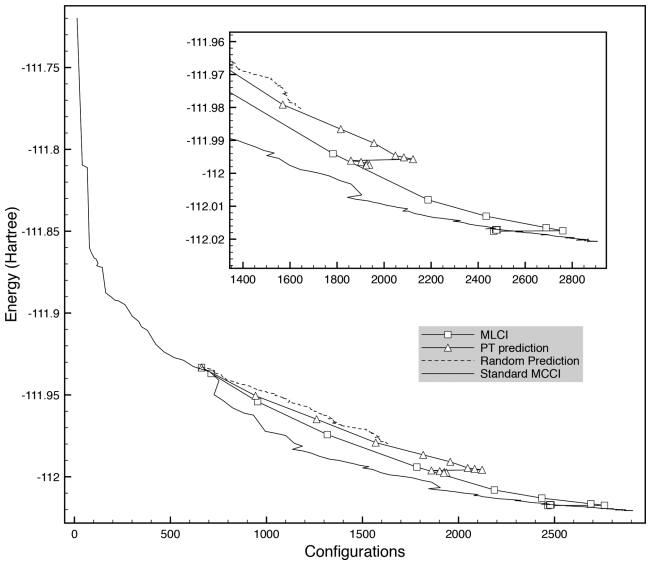


Figure 7. Energy (Hartree) against number of configurations on each iteration for serial calculations for carbon monoxide with a bond length of 4 bohr, the 3-21G basis set, two frozen orbitals, and a cutoff of $c_{min} = 10^{-3}$ using machine learning predictions to select configurations (MLCI), PT prediction, random prediction, or standard MCCI. Inset: Enlargement of the curves when close to convergence.

jagged due to full pruning causing a noticeable drop in the number of configurations every ten iterations. We see that the lowest energy and most configurations occur for MCCI, and that random prediction has the highest energy for a given number of configurations although this is often not by much. Until convergence is reached PT prediction tends to have a higher energy than the MLCI curve, which in turn is higher than MCCI. However, at convergence the result of PT prediction is approximately in line with MLCI results from earlier iterations (see inset to Figure 7), while the MCCI curve almost passes through the converged MLCI value. In this case the MLCI calculation could be viewed as being able to use significantly fewer iterations than MCCI to reach a point on the MCCI curve that is reasonably close to the converged MCCI value.

We emphasize that these are proof-of-concept calculations for predicting important configurations so that the algorithms and code could be made more efficient, but we present indicative timing information in Table 3. We note that recent developments have resulted in faster programs that use perturbation theory to select configurations. These include ref 43, which employs a semistochastic approach and efficiently prevents duplicates by checking if a prospective configuration can be created from the previously considered reference determinants. Stochastic sampling of the singles and doubles space is used to accelerate heat-bath CI[31] in ref 44. Very recently, developments,[45] including calculating matrix elements before sorting to remove duplicates, have impressively improved the efficiency of deterministic perturbation in the

Table 3. Percentage of the FCI Correlation Energy Recovered, Iterations, Time, and Number of Configurations for Converged Serial Calculations[a]

	% correlation energy	iterations	time (s)	configurations
MLCI	93.9	15	325	2477
random prediction	87.7	333	1763	1897
PT prediction	88.0	14	4349	1924
MCCI	95.3	143	291	2853

[a]Using MLCI, random prediction, PT prediction or MCCI for carbon monoxide with a bond length of 4 bohr, the 3-21G basis set, two frozen orbitals, and $c_{min} = 10^{-3}$.

context of ASCI.[30] The results for random prediction here differ to those plotted in Figures 5 and 6 in that they limit the number of rejects to the length of the wave function so that the number of rejects does not cause the calculation to stop and convergence can be reached. The time required for MLCI is not much longer than MCCI despite MLCI generating all the singles and doubles without duplicates. This also slows down random prediction and PT prediction, where in addition the former suffers from the large number of iterations with little energy change while the latter has the cost of evaluating $\langle I|\hat{H}|\Psi^{(0)}\rangle$ for all members $\langle I|$ of the singles and doubles space. This means that they take noticeably more time to converge than the other two approaches. PT prediction also calculates the normalization for the first-order wave function which is not strictly necessary if we are only interested in the relative magnitude of coefficients in this wave function and will contribute to the time cost of the calculation. The FCI energy was calculated with Molpro[40] and required ∼4.8 million Slater determinants. The MCCI wave function used the most configurations, but also captured the most correlation energy while MLCI was not far behind. Using a multireference indicator (MR) for configuration interaction wave functions,[42,46] we find a value of 0.94 for the MLCI wave function compared with that of 0.95 for MCCI. MR approaches one as the wave function becomes very strongly multireference so this system is indeed a strongly multireference problem and MLCI is capturing the multireference character.

In Table 4, we see that, when the cutoff is lowered to $c_{min} = 5 \times 10^{-4}$, the MLCI calculation is around twice as fast as the MCCI example run although MLCI captures slightly less of the

Table 4. Percentage of the FCI Correlation Energy Recovered, Iterations, Time, and Number of Configurations for Converged Serial Calculations[a]

	% correlation energy	iterations	time (s)	configurations
	$c_{min} = 5 \times 10^{-4}$			
MLCI	96.9	15	703	5638
random prediction	94.7	523	7714	5146
PT prediction	91.4	17	16947	4114
MCCI	97.7	143	1431	6451
	$c_{min} = 2 \times 10^{-4}$			
MLCI	98.3	16	2079	12971
MCCI	99.1	133	7293	16014

[a]Using MLCI, random prediction, PT prediction, or MCCI for carbon monoxide with a bond length of 4 bohr, the 3-21G basis set, two frozen orbitals, and $c_{min} = 5 \times 10^{-4}$ or $c_{min} = 2 \times 10^{-4}$.

DOI: 10.1021/acs.jctc.8b00849
J. Chem. Theory Comput. 2018, 14, 5739−5749

correlation energy using fewer configurations. Again, very few iterations are needed for the convergence of MLCI or PT prediction and the MLCI energy is closer to the FCI result than using random or PT predictions. Similar to the $c_{min} = 10^{-3}$ result, we see that random prediction requires very many iterations for convergence although its final energy is now more accurate than using PT prediction. Table 4 also shows that if we lower the cutoff further to $c_{min} = 2 \times 10^{-4}$ then MLCI is around 3.5 times faster than MCCI. However, again, slightly less of the correlation energy is recovered by MLCI in its 16 iterations although this is not at odds with the fewer configurations that it requires to give 98.3% of the correlation energy for this multireference problem.

3.2. Equilibrium Carbon Monoxide. We now consider carbon monoxide using, again, the 3-21G basis set and two frozen orbitals, but at its equilibrium bond length[47] of 2.1316 bohr. With four frozen orbitals such a system was previously found[42] to not have significant multireference when using the canonical Hartree−Fock orbitals. For problems that are not multireference then one would expect approaches based on small perturbative corrections to be more efficient when starting from the Hartree−Fock determinant and PT prediction should be more accurate.

In Figure 8, the convergence in the energy is displayed only for MLCI and PT prediction, as the number of iterations were

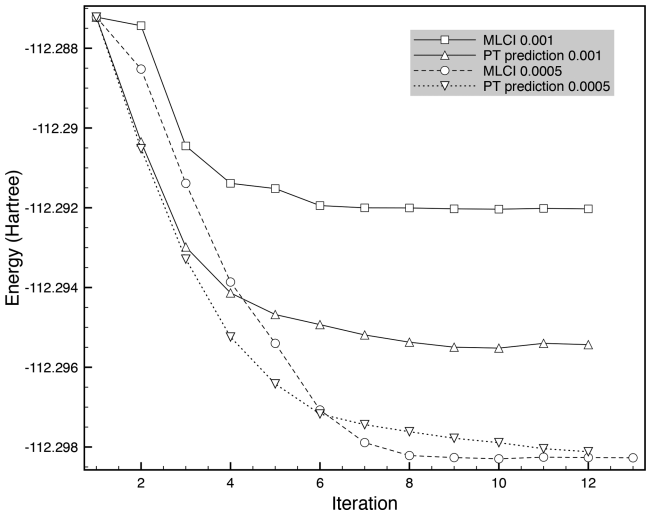


Figure 8. Energy (Hartree) against iteration number for serial calculations for carbon monoxide with a bond length of 2.1316 bohr, the 3-21G basis set, two frozen orbitals, and a cutoff of $c_{min} = 10^{-3}$ or $c_{min} = 5 \times 10^{-4}$ with c_{min} also the convergence tolerance, using machine learning predictions to select configurations (MLCI) or PT prediction.

so much lower for these methods than the others. For $c_{min} = 10^{-3}$, it is indeed the case that PT prediction gives a lower converged energy than MLCI. The energy scale of the graph amplifies this difference to a degree as PT prediction captures 93.7% of the correlation energy while the MLCI is just a little lower at 92.3%. Interestingly, on lowering the cutoff to $c_{min} = 5 \times 10^{-4}$, we see in Figure 8 that MLCI gives a very slightly lower energy than PT prediction, although PT prediction required one fewer iteration.

Table 5 shows that, for $c_{min} = 5 \times 10^{-4}$, the MCCI run gave the closest energy to FCI and required a similar time to MLCI, which was the quickest approach here. Again the number of

Table 5. Percentage of the FCI Correlation Energy Recovered, Iterations, Time, and Number of Configurations for Converged Serial Calculations[a]

	% correlation energy	iterations	time (s)	configurations
MLCI	95.2	13	255	2366
random prediction	92.9	103	653	1995
PT prediction	95.1	12	3927	2278
MCCI	96.9	113	290	3350

[a]Using MLCI, random prediction, PT prediction, and standard MCCI for carbon monoxide with a bond length of 2.1316 bohr, the 3-21G basis set, two frozen orbitals, and $c_{min} = 5 \times 10^{-4}$.

iterations required for convergence were substantially more for the stochastic methods than for MLCI or PT prediction and, in this case, random prediction gave the highest energy. A multireference indicator (MR) gives 0.16 for both MCCI and MLCI suggesting that the wave function would not be considered multireference. Despite this, the neural network in MLCI performs similarly to PT prediction in the accuracy of the energy (although PT prediction used slightly fewer configurations) and is only a little less accurate than MCCI, which required noticeably more configurations.

3.3. Stretched Water. To demonstrate that the form of neural network used can tackle not only the particular case of carbon monoxide (10 electrons and 16 orbitals) but also other multireference problems, we finally look at water with stretched bonds. We use the cc-pVDZ basis with one frozen orbital resulting in 8 electrons in 23 orbitals. The bond length is set to 4.8 bohr and the angle to 104.5 degrees.

Figure 9 displays the energy against iteration for MLCI and PT prediction as the other approaches required many more iterations. For this system we see that MLCI gives a lower energy at convergence than PT prediction. In addition, the MLCI result at the larger cutoff ($c_{min} = 10^{-3}$) is actually slightly

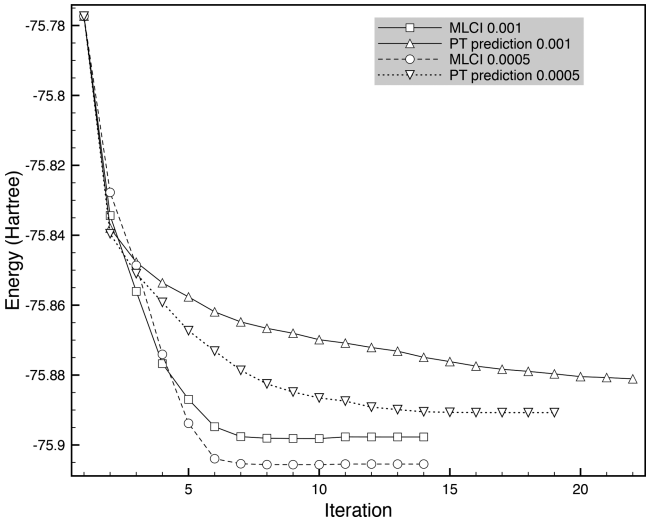


Figure 9. Energy (Hartree) against iteration number for serial calculations for water with bond length 4.8 bohr, angle 104.5 degrees, the cc-pVDZ basis set, one frozen orbital, and a cutoff of $c_{min} = 10^{-3}$ or $c_{min} = 5 \times 10^{-4}$ with the same values for the convergence tolerance using machine learning predictions to select configurations (MLCI) or PT prediction.

DOI: 10.1021/acs.jctc.8b00849
J. Chem. Theory Comput. 2018, 14, 5739−5749

more accurate than PT prediction at the lower cutoff (c_{min} = 5 × 10^{-4}).

We see in Table 6 that, for c_{min} = 10^{-3}, MLCI and MCCI both capture the most correlation energy to one decimal place

Table 6. Percentage of the FCI Correlation Energy Recovered, Iterations, Time, and Number of Configurations for Converged Serial Calculations[a]

	% correlation energy	iterations	time (s)	configurations
	c_{min} = 10^{-3}			
MLCI	96.2	14	425	2086
random prediction	73.4	63	320	893
PT prediction	93.1	22	14468	1791
MCCI	96.2	193	283	2231
	c_{min} = 5 × 10^{-4}			
MLCI	98.0	14	686	3967
random prediction	94.1	713	9469	3328
PT prediction	95.1	19	35526	3558
MCCI	98.3	173	917	4368

[a]Using MLCI, random prediction, PT prediction, and MCCI for water with bond length 4.8 bohr, angle 104.5 degrees, the cc-pVDZ basis set, one frozen orbital, and c_{min} = 10^{-3} or c_{min} = 5 × 10^{-4}.

but we mention that to two decimal places MLCI recovers marginally more (96.23% versus 96.20%) and uses slightly fewer configurations at 2086. However, in this case, the MLCI calculation was slower than MCCI. Random prediction performs poorly due to a much earlier convergence than in the runs on the previous systems.

On lowering the cutoff to c_{min} = 5 × 10^{-4}, we find that this is a strongly multireference problem as MCCI gives an MR value of 0.94, while for MLCI the value is lower, but still indicative of strong multireference character, at 0.83. However, the MLCI value for multireference character could be the more accurate: when running MCCI with c_{min} = 10^{-4}, we found a wave function of 23259 configurations and an MR value of 0.84. Returning to c_{min} = 5 × 10^{-4} and Table 6, we see that random prediction gives an energy that is not so different to the other approaches but in doing so requires many more iterations than the other methods. MLCI, now the quickest approach, captures only slightly less of the correlation energy than MCCI using fewer configurations: 3967 versus 4368 to give around 98% of the correlation energy where we note that the FCI wave function for this system comprises around 19.6 million configurations.

4. SUMMARY

In this Article, we put forward the idea of machine learning configuration interaction (MLCI) for quantum chemistry. Here, an artificial neural network is trained on-the-fly to select important configurations as it iteratively builds up a wave function by choosing configurations for inclusion in a selected configuration interaction scheme.

For stretched carbon monoxide, we demonstrated how the chosen form of neural network could discriminate between important and unimportant configurations, that it was not trained on, much better than by chance. The MLCI procedure was applied to this multireference problem and shown to converge in significantly fewer iterations than when predicting important configurations randomly. We found Monte Carlo configuration interaction (MCCI)[16−18] to give the most accurate results for this system for a given cutoff (c_{min}), which defines the minimum coefficient a configuration needs in the wave function to not be eventually deleted. However, MLCI was only a little less accurate, used fewer configurations and, even for the proof-of-concept MLCI program, required significantly less time for serial computation as c_{min} decreased. However, we noted that the number of iterations to convergence for MCCI could be reduced if it was run in parallel. Using first-order perturbation theory (PT prediction) to predict configurations instead of the neural network also resulted in significantly fewer iterations to convergence than stochastic approaches. However, compared with the neural network approach much more computation time was necessary and, perhaps in keeping with this being a multireference problem, the energy was less accurate than MLCI.

We then considered carbon monoxide at its equilibrium geometry as an example of a system that is not significantly multireference and should be well-described by methods built around small corrections to a single determinant. For larger c_{min}, corresponding to less accurate calculations, we saw that PT prediction captured a little more of the correlation energy than MLCI. Interestingly, when c_{min} was lowered, the more accurate calculations had MLCI and PT prediction giving similar energies and MLCI was noticeably faster. Again MCCI required many more iterations, but gave a slightly more accurate energy using more configurations than MLCI.

To verify that the form of the neural network was not just appropriate for carbon monoxide, we finally investigated the MLCI approach on the multireference problem of water with stretched bonds. For c_{min} = 5 × 10^{-4}, we found that fewer than 20 iterations were needed for convergence of MLCI or PT prediction, while the MCCI run required 173. When replacing the neural network predictions in MLCI with random predictions 713 iterations were necessary. MLCI, again the fastest method, captured only slightly less of the correlation energy (98%) than MCCI and more than the other approaches. To do this it used around 4000 configurations compared with approximately 19.6 million in the full configuration interaction wave function.

We have seen that machine learning configuration interaction (MLCI) can use on-the-fly training of an artificial neural network to iteratively build up a compact wave function that can capture much of the accuracy of full configuration interaction but using a very small fraction of the configurations. Despite MLCI being implemented as a prototype at this stage, we found that compared with the other ways considered here of selecting configurations its accuracy was competitive, it could converge in fewer than 20 iterations and it required less time for the higher-accuracy serial calculations on small molecules including systems with, and without, significant multireference character.

Although on-the-fly training of the artificial neural network did not disadvantage the relative speed of the calculations here, for larger basis sets and more configurations this could become a bottleneck. Furthermore, accurate calculations with larger basis sets may require more hidden nodes or more hidden layers. Future work will investigate using graphical processing units for training deep neural networks to enable MLCI to efficiently and accurately calculate ab initio potential energy surfaces of larger molecules.

DOI: 10.1021/acs.jctc.8b00849
J. Chem. Theory Comput. 2018, 14, 5739−5749

AUTHOR INFORMATION

Corresponding Author

*E-mail: J.Coe@hw.ac.uk.

ORCID

J. P. Coe: 0000-0002-9642-8399

Notes

The author declares no competing financial interest.

ACKNOWLEDGMENTS

J.P.C. thanks the EPSRC for support via the platform grant EP/P001459/1.

REFERENCES

(1) Gastegger, M.; Marquetand, P. High-Dimensional Neural Network Potentials for Organic Reactions and an Improved Training Algorithm. *J. Chem. Theory Comput.* **2015**, *11*, 2187.

(2) Janet, J. P.; Kulik, H. J. Predicting electronic structure properties of transition metal complexes with neural networks. *Chem. Sci.* **2017**, *8*, 5137.

(3) Schütt, K. T.; Arbabzadah, F.; Chmiela, S.; Müller, K. R.; Tkatchenko, A. Quantum-chemical insights from deep tensor neural networks. *Nat. Commun.* **2017**, *8*, 13890.

(4) Ramakrishnan, R.; Dral, P. O.; Rupp, M.; von Lilienfeld, O. A. Big Data Meets Quantum Chemistry Approximations: The Δ-Machine Learning Approach. *J. Chem. Theory Comput.* **2015**, *11*, 2087−2096.

(5) McGibbon, R. T.; Taube, A. G.; Donchev, A. G.; Siva, K.; Hernández, F.; Hargus, C.; Law, K.-H.; Klepeis, J. L.; Shaw, D. E. Improving the accuracy of Møller-Plesset perturbation theory with neural networks. *J. Chem. Phys.* **2017**, *147*, 161725.

(6) Yao, K.; Parkhill, J. Kinetic Energy of Hydrocarbons as a Function of Electron Density and Convolutional Neural Networks. *J. Chem. Theory Comput.* **2016**, *12*, 1139.

(7) Brockherde, F.; Vogt, L.; Li, L.; Tuckerman, M. E.; Burke, K.; Müller, K.-R. Bypassing the Kohn-Sham equations with machine learning. *Nat. Commun.* **2017**, *8*, 872.

(8) Mills, K.; Spanner, M.; Tamblyn, I. Deep learning and the Schrödinger equation. *Phys. Rev. A: At., Mol., Opt. Phys.* **2017**, *96*, 042113.

(9) Purvis, G. D., III; Bartlett, R. J. A full coupled-cluster singles and doubles model: The inclusion of disconnected triples. *J. Chem. Phys.* **1982**, *76*, 1910−1918.

(10) Raghavachari, K.; Trucks, G. W.; Pople, J. A.; Head-Gordon, M. A fifth-order perturbation comparison of electron correlation theories. *Chem. Phys. Lett.* **1989**, *157*, 479−483.

(11) Siegbahn, P. E. M.; Almlöf, J.; Heiberg, A.; Roos, B. O. The complete active space SCF (CASSCF) method in a Newton-Raphson formulation with application to the HNO molecule. *J. Chem. Phys.* **1981**, *74*, 2384−2396.

(12) Andersson, K.; Malmqvist, P.-Å.; Roos, B. O.; Sadlej, A. J.; Wolinski, K. Second-Order Perturbation Theory with a CASSCF Reference Function. *J. Phys. Chem.* **1990**, *94*, 5483.

(13) Werner, H.-J.; Knowles, P. J. An efficient internally contracted multiconfiguration-reference configuration interaction method. *J. Chem. Phys.* **1988**, *89*, 5803.

(14) Huron, B.; Malrieu, J. P.; Rancurel, P. Iterative perturbation calculations of ground and excited state energies from multiconfigurational zeroth-order wavefunctions. *J. Chem. Phys.* **1973**, *58*, 5745.

(15) Harrison, R. J. Approximating full configuration interaction with selected configuration interaction and perturbation theory. *J. Chem. Phys.* **1991**, *94*, 5021.

(16) Greer, J. C. Consistent treatment of correlation effects in molecular dissociation studies using randomly chosen configurations. *J. Chem. Phys.* **1995**, *103*, 7996.

(17) Greer, J. C. Monte Carlo Configuration Interaction. *J. Comput. Phys.* **1998**, *146*, 181.

(18) Tong, L.; Nolan, M.; Cheng, T.; Greer, J. C. A Monte Carlo configuration generation computer program for the calculation of electronic states of atoms, molecules, and quantum dots. *Comput. Phys. Commun.* **2000**, *131*, 142. Monte Carlo Configuration Interaction. https://github.com/MCCI/mcci.

(19) Bytautas, L.; Ruedenberg, K. A priori identification of configurational deadwood. *Chem. Phys.* **2009**, *356*, 64.

(20) Lyakh, D. I.; Bartlett, R. J. An adaptive coupled-cluster theory: @CC approach. *J. Chem. Phys.* **2010**, *133*, 244112.

(21) Coe, J. P.; Paterson, M. J. State-averaged Monte Carlo configuration interaction applied to electronically excited states. *J. Chem. Phys.* **2013**, *139*, 154103.

(22) Szepieniec, M.; Yeriskin, I.; Greer, J. C. Quasiparticle energies and lifetimes in a metallic chain model of a tunnel junction. *J. Chem. Phys.* **2013**, *138*, 144105.

(23) Coe, J. P.; Paterson, M. J. Approaching exact hyperpolarizabilities via sum-over-states Monte Carlo configuration interaction. *J. Chem. Phys.* **2014**, *141*, 124118.

(24) Kelly, T. P.; Perera, A.; Bartlett, R. J.; Greer, J. C. Monte Carlo configuration interaction with perturbation corrections for dissociation energies of first row diatomic molecules: C_2, N_2, O_2, CO, and NO. *J. Chem. Phys.* **2014**, *140*, 084114.

(25) Coe, J. P.; Paterson, M. J. Multireference X-ray emission and absorption spectroscopy calculations from Monte Carlo configuration interaction. *Theor. Chem. Acc.* **2015**, *134*, 58.

(26) Murphy, P.; Coe, J. P.; Paterson, M. J. Development of spin-orbit coupling for stochastic configuration interaction techniques. *J. Comput. Chem.* **2018**, *39*, 319.

(27) Evangelista, F. A. Adaptive multiconfigurational wave functions. *J. Chem. Phys.* **2014**, *140*, 124114.

(28) 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.

(29) Schriber, J. B.; Evangelista, F. A. Adaptive Configuration Interaction for Computing Challenging Electronic Excited States with Tunable Accuracy. *J. Chem. Theory Comput.* **2017**, *13*, 5354.

(30) Tubman, N. M.; Lee, J.; Takeshita, T. Y.; Head-Gordon, M.; Whaley, K. B. A deterministic alternative to the full configuration interaction quantum Monte Carlo method. *J. Chem. Phys.* **2016**, *145*, 044112.

(31) 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.

(32) Ohtsuka, Y.; Hasegawa, J. Selected configuration interaction using sampled first-order corrections to wave functions. *J. Chem. Phys.* **2017**, *147*, 034102.

(33) Ohtsuka, Y.; Nagase, S. Projector Monte Carlo method based on configuration state functions. Test applications to the H_4 system and dissociation of LiH. *Chem. Phys. Lett.* **2008**, *463*, 431.

(34) Booth, G. H.; Thom, A. J. W.; Alavi, A. Fermion Monte Carlo without fixed nodes: A game of life, death, and annihilation in Slater determinant space. *J. Chem. Phys.* **2009**, *131*, 054106.

(35) Scemama, A.; Garniron, Y.; Caffarel, M.; Loos, P.-F. Deterministic Construction of Nodal Surfaces within Quantum Monte Carlo: The Case of FeS. *J. Chem. Theory Comput.* **2018**, *14*, 1395.

(36) Cybenko, G. Approximation by Superpositions of a Sigmoidal Function. *Math. Control Signals Systems* **1989**, *2*, 303.

(37) Ba, L. J.; Caruana, R. Do Deep Nets Really Need to be Deep? *Advances in Neural Information Processing Systems 27*. **2014**.

(38) Mitchell, T. M. *Machine Learning*; McGraw-Hill, 1997.

(39) Győrffy, W.; Bartlett, R. J.; Greer, J. C. Monte Carlo configuration interaction predictions for the electronic spectra of Ne, CH_2, C_2, N_2, and H_2O compared to full configuration interaction calculations. *J. Chem. Phys.* **2008**, *129*, 064103.

(40) Werner, H.-J.; Knowles, P. J.; Knizia, G.; Manby, F. R.; Schütz, M.; Celani, P.; Györffy, W.; Kats, D.; Korona, T.; Lindh, R.; Mitrushenkov, A.; Rauhut, G.; Shamasundar, K. R.; Adler, T. B.;

DOI: 10.1021/acs.jctc.8b00849
J. Chem. Theory Comput. 2018, 14, 5739−5749

Amos, R. D.; Bernhardsson, A.; Berning, A.; Cooper, D. L.; Deegan, M. J. O.; Dobbyn, A. J.; Eckert, F.; Goll, E.; Hampel, C.; Hesselmann, A.; Hetzer, G.; Hrenar, T.; Jansen, G.; Köppl, C.; Liu, Y.; Lloyd, A. W.; Mata, R. A.; May, A. J.; McNicholas, S. J.; Meyer, W.; Mura, M. E.; Nicklass, A.; O'Neill, D. P.; Palmieri, P.; Peng, D.; Pflüger, K.; Pitzer, R.; Reiher, M.; Shiozaki, T.; Stoll, H.; Stone, A. J.; Tarroni, R.; Thorsteinsson, T.; Wang, M. MOLPRO, version 2015.1, 2015. http://www.molpro.net.

(41) Coe, J. P.; Paterson, M. J. Development of Monte Carlo configuration interaction: Natural orbitals and second-order perturbation theory. *J. Chem. Phys.* **2012**, *137*, 204108.

(42) Coe, J. P.; Paterson, M. J. Investigating Multireference Character and Correlation in Quantum Chemistry. *J. Chem. Theory Comput.* **2015**, *11*, 4189.

(43) Garniron, Y.; Scemama, A.; Loos, P.-F.; Caffarel, M. Hybrid stochastic-deterministic calculation of the second-order perturbative contribution of multireference perturbation theory. *J. Chem. Phys.* **2017**, *147*, 034101.

(44) 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.

(45) Tubman, N. M.; Levine, D. S.; Hait, D.; Head-Gordon, M.; Birgitta Whaley, K. An efficient deterministic perturbation theory for selected configuration interaction methods. 2018, arXiv:1808.02049. arXiv.org e-Print archive. https://arxiv.org/abs/1808.02049.

(46) Coe, J. P.; Murphy, P.; Paterson, M. J. Applying Monte Carlo configuration interaction to transition metal dimers: Exploring the balance between static and dynamic correlation. *Chem. Phys. Lett.* **2014**, *604*, 46.

(47) Muenter, J. S. Electric dipole moment of carbon monoxide. *J. Mol. Spectrosc.* **1975**, *55*, 490.

DOI: 10.1021/acs.jctc.8b00849
J. Chem. Theory Comput. 2018, 14, 5739−5749