Active Learning in Gaussian Process Interpolation of Potential Energy Surfaces

Elena Uteva [a], Richard S. Graham [b], Richard D. Wilkinson [c] and Richard J. Wheatley[a].[1]

[a]*School of Chemistry, University of Nottingham, Nottingham NG7 2RD, UK.*

[b]*School of Mathematical Sciences, University of Nottingham, Nottingham NG7 2RD, UK.*

[c]*School of Mathematics and Statistics, University of Sheffield, Sheffield, S10 2TN, UK.*

(Dated: 16 October 2018)

Three active learning schemes are used to generate training data for Gaussian process interpolation of intermolecular potential energy surfaces. These schemes aim to achieve the lowest predictive error using the fewest points and therefore act as an alternative to the status quo methods involving grid-based sampling or space-filling designs like Latin hypercubes (LHC). Results are presented for three molecular systems: CO_2-Ne, CO_2-H_2 and Ar_3. For each system, two of the active learning schemes proposed notably outperform LHC designs of comparable size, and in two of the systems, produce an error value an order of magnitude lower than the one produced by the LHC method. The procedures can be used to select a subset of points from a large pre-existing data set, to select points to generate data de-novo, or to supplement an existing data set to improve accuracy.

PACS numbers: XXX, XXX, XXX

I. INTRODUCTION

Potential energy surfaces (PES) are a central concept in physical chemistry and are used extensively in silico to obtain relevant information about the structural, spectral and dynamical properties of different molecular systems. Since the quantitative accuracy of the information extracted from applications depends directly on the quantitative accuracy of the potential energy, a lot of research has been devoted into developing novel methods of generating high quality ab initio potential energy surfaces without suffering an excessive computational penalty.

Recently, the methods adopted by the scientific community have focussed on the use of machine learning techniques to generate global potential surfaces through simple data mapping approaches[1]. The adoption of machine learning models in the quantum chemistry domain marks a change in the nature of the proposed solution, stepping away from the previous heavy reliance on chemical and physical knowledge of the molecular system being modelled, and instead trying to map the input-output space as accurately as possible given pre-existing data known as the training set.

For any given model, the choice of training set dictates both the overall predictive accuracy of the model and the total amount of points needed to reach a certain error threshold. With the computational cost of calculating potential energies increasing with both the size of the basis set and the complexity of the system, generating an optimal choice of points that yields the lowest overall error using the small data set possible becomes more and more imperative, the longer each single point calculation takes to run.

One possible way to achieve an optimal training set is through the use of active learning - a family of machine learning methods with a strong anchoring in information theory that has been applied to domains such as chemoinformatics[2], financial crime detection[3] and speech recognition problems[4]. The overall aim of active learning is to make models more economical and time-efficient (in terms of data use and generation) by allowing the model to make queries as to where to add more training data rather than remaining passive to the data acquisition process. In the context of regression-based machine learning models, active learning (alternatively known as sequential design, online fitting or adaptive fitting), is the process of using previously learned information obtained during the training phase, to guide the process of new point placement through iterative methods that add points either one-by-one or in batches.

Active learning (AL), in the domain of PES generation, is an alternative to the current status quo of relying on either grid-based sampling[5] (where points are sampled in intervals between a range of values of each geometric variable, also known as factorial design) or space filling designs[6] (where points are spread out across design space to get approximate uniform coverage) to produce training data. This work applies active learning to Gaussian processes (GPs), which are a flexible non-parametric family of models capable of approximating functions using relatively small data sets. Gaussian processes have been successfully applied to model the PES of several systems, including the generation of a reactive PES for the lowest triplet state of SH_2[7], a ground state global PES for N_4[6], and numerous bimolecular Van der Waals complexes including $CO_2{-}Ne$, $CO_2{-}H_2$, $HF{-}HF$, $CH_4{-}N_2$, $CO_2{-}CO$[8] and $CO_2{-}N_2$[9]. In all these instances, however, training data were generated using space filling designs (usually Latin hypercube sampling), which despite surpassing grid-based sampling methods for GPs, does not achieve an optimal trade-off between accuracy and computational cost.

Relative to batch designs (where batch design is a broad term encompassing experimental designs that are produced in bulk, rather than sequentially), there are numerous potential benefits of active learning that make it an attractive prospect in the pursuit of optimal training sets:

1. A design might not be able to achieve the required level of accuracy for a given number of data points, so additional points may be needed to reduce the error of predictions. Since some information has already been gained, it follows naturally that active learning of subsequent points will lead to the best improvement in predictive accuracy relative to a non-adaptive data point selection.

2. AL offers the potential to build functional non-stationarity into the experimental design, by including more points in regions of space where there is a greater functional variation, which is hard to do *a priori* as is needed for batch design.

3. AL offers a way to reduce a large pre- existing data set to a smaller subset of training points while retaining a controlled level of prediction accuracy.

4. When working with a model with over-specified variables, finding optimal space-filling

> designs is nontrivial. Specifically, it can be difficult to propose space-filling designs for which all points lie in the configuration subspace that corresponds to physically valid molecular geometries. In contrast the AL strategies presented herein choose new points from a set of candidate points already in the physically valid subspace.

For general non-GP active learning problems, there have been many heuristics to guide point selection, including selection of points where there is a lack of data[10], low confidence[11], the biggest expected change in the model[12] and where there has been previous selection of data that has resulted in learning[13].

Gaussian process regression models give probabilistic predictions (i.e. a best guess and the uncertainty about that guess), making them ideal candidates for sequential design methodology. The use of variance in Gaussian process sequential design tasks was first introduced by McKay[14], who proposed a method that aims to maximize the expected information gain about the parameter values of the model by choosing data with the highest predictive variance. This was then later expanded on by Cohn[12], who proposed a method which aimed to maximize the expected information gain about the parameter values of the model by selecting points that lead to the biggest reduction in average variance instead of just finding the highest single-point predictive variance.

Although there has been some prior research dedicated to active learning in the context of both Gaussian process regression and general computer emulation tasks, there has been relatively little work done on the use of active learning techniques in the generation of intermolecular potentials for different molecular systems. Active learning techniques in the field of PES generation have so far been most prominent in instances of 'on the fly' molecular dynamics, where the data points are generated over the course of the sampling. Rupp *et al.*[15] used an adaptive learning scheme alongside a hybrid quantum mechanics and machine learning method based on Gaussian process regression to run molecular dynamics simulations of the complex natural product Archazolid A, and found that using the model's inherent predictive variance to decide whether to carry out additional electronic structure calculations led to a reduction of the amount of calculations carried out by 40%. Research was also done by Podryabinkin *et al.*[16] on linearly parametrized interatomic potentials. An active learning scheme based on a D-optimality criterion was applied to moment tensor potentials, querying whether the machine learning value at the sampled configuration was expected to be a good enough estimate or whether new ab initio data needed to be generated instead. The results were then compared to the classical on-the-fly method proposed by De Vita et al[17]. The classical method has a general framework of starting with an initial short AIMD trajectory (a few picoseconds long) and then adding on new ab initio points after a fixed number of steps have been taken. The simulations by Podryabinkin *et al.*[16] were terminated if they produced unphysically low atomic separations. The failure time was the time after which half of the trajectories had terminated. Hence, they used failure time as a measure of PES accuracy. It was found that whereas the classical method increased the failure time from 15ps to 150ps following the addition of 1500 ab initio points, the active learning method increased simulation time to 0.5μs following the addition of only 50 new points.

Although these methods are promising in the domain of molecular dynamics, there has still been little work done on active learning of global intermolecular potentials. Since one of the main factors underpinning the predictive power of the model is training point location, finding a way to select an optimal selection of data points is a critical step in generating chemically useful potential energy surfaces at a fraction of the computational cost of extensive ab initio calculations. Recently, Zhang et al.[18] used predictive variances (weighted to bias lower energy point selection) to generate training data for H_3 and two prototypical reactive systems. Although this lays out the first framework for sequential design of global potentials, only methods involving predictive variance were considered, and the active learning scheme was not compared to space filling design methods of equal size.

II. GAUSSIAN PROCESS MODELLING

Gaussian processes are flexible non-parametric models of functions. They are used widely in machine learning and statistics as they form a closed family of models under Bayesian updating, so that a Gaussian process conditioned upon observations is still a Gaussian process, making them a convenient tractable family of models to use. Like neural networks, they are capable of fitting highly flexible functions without relying on theory-based fits of complex data sets. In comparison to neural networks however, GPs are mathematically tractable and interpretable, and allow for prior information (such as symmetry, differentiability, and conditioning on derivative information) to be directly incorporated into the model itself. Gaussian processes are called *non-parametric* as although they may have parameterised components (typically the mean and covariance functions - see below), the dimension of the ‘parameter’ that defines a GP model grows with the size of the training set. In this sense, GPs ‘carry the training data’ with them.

A GP model of an unknown function $f(\mathbf{x})$ is fully defined by its mean function $m(\mathbf{x}) := \mathbb{E}(f(\mathbf{x}))$, which is often set to zero, and a covariance function $k(\mathbf{x}, \mathbf{x}') := \mathbb{C}\mathrm{ov}(f(\mathbf{x}), f(\mathbf{x}'))$. Training data, consisting of observations of f at different values of $\mathbf{x}$, i.e. pairs $(\mathbf{x}_i, f(\mathbf{x}_i))$, can be used to update the prior mean and covariance functions to produce a posterior model (i.e. posterior mean and covariance functions) which can be used to predict $f(\mathbf{x})$ for any value of $\mathbf{x}$. The computational cost of evaluating the mean of the GP (i.e. the prediction for $f(\mathbf{x})$) is proportional to the size of the training set. The covariance function is the key component of Gaussian process learning, as it defines the space of functions from which we choose, i.e., the covariance function encodes functional properties such as continuity, stationarity and differentiability. Both the mean and covariance functions may be parametrised by ‘hyper-parameters’ of fixed dimension. These can be estimated using any standard procedure, such as maximum likelihood, Bayes, or cross-validation. A more detailed overview of Gaussian processes can be found in the seminal book by Rasmussen and Williams[19], which also presents different theoretical ways to conceptualise them and a more extensive background on covariance and mean functions.

III. METHODOLOGY

A. Molecular systems

Three molecular systems are investigated: $CO_2{-}Ne$, $CO_2{-}H_2$ and Ar_3. The two CO_2 systems are chosen because they produced impressive benchmark results in previous work[8] using exclusively a batch design strategy, namely Latin hypercube (LHC) sampling. Thus any further improvements in the root mean square error (RMSE) would be a significant result. The Ar_3 system was chosen in order to test the algorithms on a non-additive potential, which are generally harder to fit due to greater functional variation.

The intermolecular interaction energies of these complexes are calculated as a function of their configurational geometry. All molecules are approximated as linear rigid rotors in their vibrational ground state, with fixed bond lengths, although the interpolation method can be extended straightforwardly to non-rigid molecules. Energy calculations are carried out in Molpro[20] using second-order Møller-Plesset (MP2) perturbation theory for the two CO_2 systems and CCSD(T) perturbation theory for Ar_3. In all cases the augmented correlation-consistent triple-zeta (aug-cc-pVTZ) basis set is used and basis set superposition errors are accounted for by the counterpoise correction technique.

B. Data sets

The data sets contain a number of cluster geometries, along with their corresponding intermolecular interaction energies. The method considers two types of data set, training sets and reference sets. The training set is used to train the GP. The reference set serves two purposes: to provide candidate data points to be added to the training set during the active learning, and to compute the root mean square error (RMSE) of the GP.

C. Co-ordinates

TABLE I. Co-ordinates for the reference data (grid or LHC) for each system.

System	Coordinate	Range	Spacing	Reference points
$CO_2{-}Ne$	r	1.5-10 Å	0.116 Å	1122
	$\cos\theta$	0-1	0.05	
$CO_2{-}H_2$	r	1.5-10 Å	0.5 Å	12844
	$\cos\theta_1$	0-1	0.111	
	$\cos\theta_2$	0-1	0.111	
	ϕ	0-180°	20°	
Ar_3	r_{12}, r_{13}, r_{23}	2.88-9.0Å	LHC	5476

All reference data are generated using the distance and angular coordinates presented in Table I. In all binary cases r is the distance between the molecular centres. For $CO_2{-}Ne$, θ is the angle between r and the CO_2 axis. For $CO_2{-}H_2$, θ_1 is the angle between r and the CO_2 axis, θ_2 is the angle between r and the H_2 axis, and ϕ is the torsional angle of the H_2 axis. For Ar_3, r_{ij} is the distance between Ar atoms i and j. For $CO_2{-}Ne$ and $CO_2{-}H_2$, the reference data are positioned on an evenly spaced grid, as generated for previous work[8] and with the co-ordinates detailed in table I. The Ar_3 reference data are newly generated from a LHC design strategy, using an LHC algorithm[8], with adaptions described in Appendix A. For the dimer systems energy points above the high energy cut off (0.005 E_h) are excluded from the reference set. For the three body system, Ar_3, the non-additive energy is not used to exclude points. Instead points are excluded if the pairwise interaction potential between any of the 3 pairs of atoms exceeds the high energy cut off. The most straightforward way to impose this high energy cut off is to pre-calculate the reference data set and exclude points above the cut off, as is done here. However, this pre-calculation can be avoided, in principle, if a reliable way of accounting for the high energy barrier can produced. Such a method is straightforward for non-additive interactions if a PES is available for all dimers in the system.

D. Active learning and GP training

During active learning, points are removed sequentially from the reference set and added to the training set in accordance with an acquisition rule. Each method starts with a small initial set of data points and adds one point at a time, updating the model between each addition. The following three acquisition rules for selecting new training points are each tested in each of the three physical systems:

- Highest variance search, which adds points on the basis of the highest predictive variance over a pool of input data, as detailed in section III D 1. **(Method A)**.
- Absolute highest error search, which uses the reference data and adds points with the highest absolute error iteratively, as detailed in section III D 1. **(Method B)**
- Two set search, which uses two different GPs, each trained on different training sets. The test point with the largest discrepancy between the two model predictions is then added into the training set, as detailed in section III D 2. **(Method C)**

Note that method B is not a genuine active learning approach as it requires the outputs, and thus defeats the purpose of AL. It is included purely for comparison purposes. GP learning is carried out using the GPy Python package[21], modified to include symmetric covariance functions. The co-ordinates in table I are converted to inverse interatomic distances and these are used as over-specified covariates in the GP. The covariance function, a symmetrised squared exponential function, is used alongside a zero mean function. Zero-mean Gaussian observation error, with standard deviation σ_n, is assumed on the function outputs, which is often known as a nugget term. All of these approaches are identical to previous work[8]. After each new training point the RMSE is computed against the reference data. Points that have been moved from the reference to the training set no longer contribute to the RMSE.

1. Highest variance and absolute highest error search

One data point is chosen at random from the reference set, and used as the initial training set. Predictions are made over the reference set. One point is added to the training set on the basis of the highest predictive variance (Method A) or the highest absolute error of the predictions (Method B). The point is removed from the reference set and the GP hyperparameters are re-optimised (using 20 random restarts of the optimizer). The process is iterated until the desired number of points is reached.

2. Two set search

The two set search method utilises two training data sets. These two training sets are each initialised with a single point, chosen randomly from the reference set. The starting point is different for each set and both points are removed from the reference set. To add a new point, each training set is used to train a GP. The predictions for both GPs are then made over the reference data and the discrepancy between the predictions of the two GPs is calculated at each point. The point with the biggest absolute discrepancy in energy value is then removed from the reference data and is inserted into both training sets. Further points are added by repeating this process until the desired number of training points has been reached. Note that, in this algorithm, as in Method A, it is not essential to precompute the reference data. Instead, computations of the intermolecular interaction energy can be performed sequentially as new points are added. Thus these potentially expensive calculations need only be performed for the much smaller training set.

To understand the rationale for this method, note its similarity to using a jackknife estimate of the variability. A known problem with GP models, particularly when the hyper-parameters are estimated and then fixed, is that they often under-estimate uncertainty[22]. The jackknife or bootstrap[23] can be used to get more accurate estimates of the prediction uncertainty[22]. They work by training multiple GPs on multiple different subsets of the data, and then averaging predictions across these different GP models. Method C is not quite a jackknife approach (but could be made so by using more than two training sets) but is motivated by it. It adds points to the training set in places where the mean prediction from the different GPs differs most. This is the region in our parameter space most sensitive to the choice of hyper-parameters, and where interpolation is least reliable. See the work by Kleijnen *et al.*[24] for a related approach.

E. Comparison to a batch space-filling design

The effectiveness of these three acquisition rules in an active learning scheme is assessed by comparing the resulting RMSE of the models for each method against a model trained on data generated by a non-active approach[8]. These non-active results are obtained by training GPs to independently generated 'maximin' Latin hypercube data of increasing size, based on the co-ordinates in table I. For the two CO_2 systems the series of LHC data sets generated previously[8] is used. For Ar_3 a new series of LHC training sets is generated using an LHC algorithm[8], with adaptions described in Appendix A.

IV. RESULTS

For $CO_2{-}H_2$ and Ar_3, the predictive performance is measured using the root mean square error (RMSE) of the GPs on the reference set. Equal weighting is used for all data points, including positive interaction energies up to the potential energy cut off. Since the reference set is the data set from which new points are selected during the active learning algorithms, any points that have been added to the actively learned set are discarded from the reference set prior to the RMSE calculation. For fair comparison with the batch design method based on Latin hypercubes, the same reduced test set is consequently used to calculate the error of the LHC of equal size to the actively learned set. In the $CO_2{-}Ne$ case, the RMSE is calculated using a new 2095-point Latin hypercube set, using the same algorithm, geometric constraint and energy cut off as our previous work[8]. Since this test set is completely independent of the set from which points are added, only points above the energy cut off are removed during error calculation.

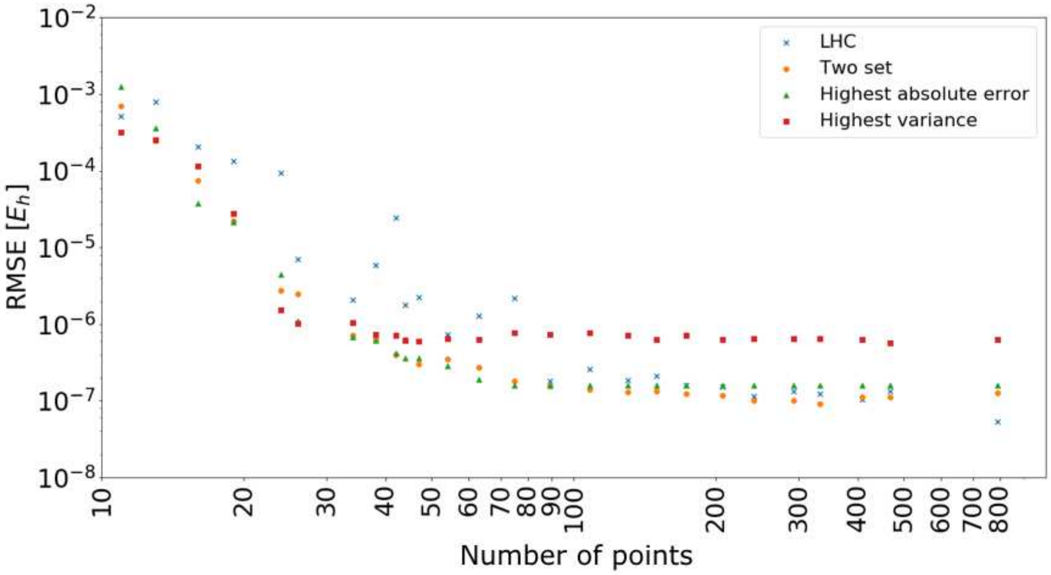

FIG. 1. RMSE against training set size for CO_2−Ne. The lowest and highest energies in the reference data are -2.90×10^{-4} E_h and 0.005 E_h and the root mean square over the reference data is 6.24×10^{-4} E_h

The results for CO_2−Ne are shown in figure 1 (E_h is the Hartree energy). The graphs show the RMSE of all three active learning methods and the error produced by the non-active Latin hypercube method. The data sets generated using active learning outperform the Latin hypercube sets until around the 90 point mark, after which the LHC produces RMSE similar to both the highest actual error and the two set methodology. Although the highest variance method performs well with small training set sizes, producing the best RMSE in three cases, it plateaus at $\sim 10^{-6}$ E_h despite the addition of many new data points. This may be due to an accumulation of different factors. As mean variance approaches the value of the nugget term, the algorithm may act more erratically in its point addition process. The highest variance algorithm also tends to bundle points around the border, which have higher uncertainty than central points. This bundling of points produces a high density coverage of a region where such point density is not needed, hence not producing the improvement in accuracy expected for a large addition of points, and causing numerical problems for the optimiser in its search for the best hyperparameter due to points being too close to each other making the covariance matrix ill-conditioned.

Figure 2 shows results for $CO_2{-}H_2$. When the number of points is very low, there is no clear advantage of the active learning over the LHC in terms of predictive accuracy. However after the addition of about 30 points, all three of the active learning methods outperform the LHC sets. The error of the highest variance method again plateaus around 10^{-6} E_h, similar to the error of the large LHC sets, whereas the RMSE of the two set and the highest absolute error method keep reducing towards $\sim 10^{-7}$ E_h and significantly outperforms the LHC design. Since the two set method adds points based on mean discrepancy and does not require energy data for the entire reference set, the fact that it performs the best is a significant result. Note that the LHC has been transformed to account for the fact that a greater density of points at short range will improve the result, so the actively learned data is being compared to an already somewhat 'intelligent' design that allocates more points in subregions of space that have more variation.

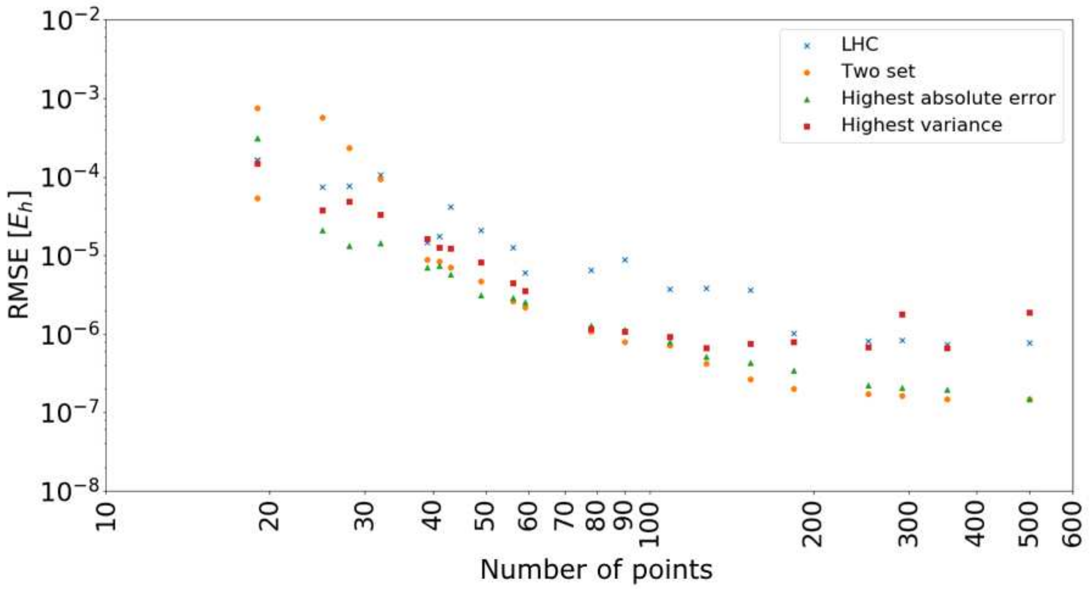

FIG. 2. RMSE against training set size for $CO_2{-}H_2$. The lowest and highest energies in the reference data are -8.25×10^{-4} E_h and 0.005 E_h and the root mean square over the reference data is 7.54×10^{-4} E_h.

Results for the three body non-additive potential of argon are presented in figure 3. The two set and the highest absolute error method outperform the space filling design, especially for training sets with a large number of design points where the AL methods produce errors an order of magnitude lower. The highest variance method performs erratically and is the worst out of all the models.

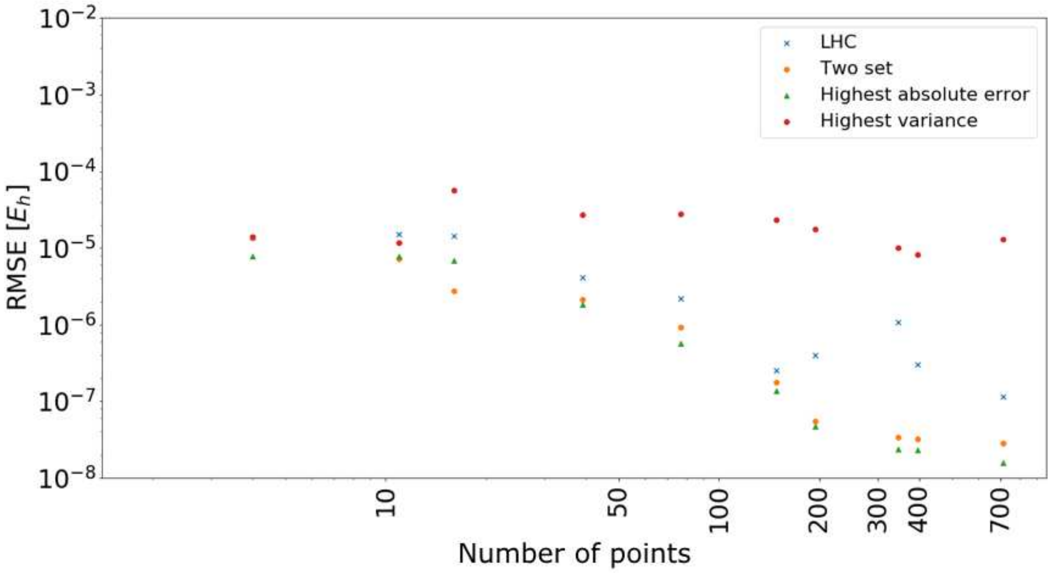

FIG. 3. RMSE against training size for Ar_3. The lowest and highest energies in the reference data are -2.10×10^{-4} E_h and 5.31×10^{-5} E_h and the root mean square over the reference data is 1.28×10^{-5} E_h.

In all 3 cases our GP approach captures very well the interaction data within the geometric constraint. However, our GPs are not ideally suited to extrapolate to low energies at long distances, outside our geometric constraint. For long distances we suggest the use of a long-range asymptotic expansion function, obtained from intermolecular perturbation theory, as in our previous work[8].

Highest absolute error relies on having a complete set of calculated energy points prior to conducting active learning, so it is limited exclusively to producing an optimal smaller subset of points from a large pre-existing data set. The other two methods, the highest variance and the two set learning, can both be expanded to generating data sets de-novo if a way of accounting for the high energy region can be introduced to prevent the active learning algorithm adding points that are then excluded by the high energy cut-off. One possible way of doing this is to use an initial smaller Latin hypercube to train a GP and in the active learning, only add points predicted to be below the high energy threshold. Another possible way of excluding the high energy points is by using a classifier that gets more accurate as the number of points increases, allowing for a complete 'de novo' generation of data and a more refined high energy boundary. An additional consideration when modifying the algorithm to produce training sets de novo is to automate the energy point generation and the insertion of the point back into the GPy model for re-training. This may create computational inefficiency, as new data points need to be calculated one at a time.

V. FUTURE WORK

There is a large scope for work that builds on these results. An obvious extension is the application of the algorithms to larger chemical systems, where the need for smaller data sets becomes more imperative due to the computational unfeasibility of carrying out extensive calculations at different geometries. All of our methods could be applied to larger chemical systems, to some extent. For particularly large systems generating the LHC will become computationally impractical. Here, method C has a particular advantage as it does not require a large LHC reference set and needs only the generation of the training data. Increasing computational cost per point also motivates work producing data sets completely de novo, using iterative design alongside a classifier that aims to predict high energy points and exclude them from the search over the course of the active learning algorithm. The algorithms that we have used rely on a random point selection for the initial training point, and one possible way of improving the robustness of the algorithm is by devising a more systematic way to choose the location of the starting point.

VI. CONCLUSIONS

Active learning of training data in the field of potential energy surface interpolation is a promising development, as it allows for the production of more time-efficient global potential energy surfaces that can be systematically improved until the desired level of accuracy has been achieved. Since the main appeal of potential energy surfaces is their consequent application in different techniques to extract macroscopic and atomistic properties of different molecules, subregions of space where the model fared badly due to point sparsity may cause problems in the methodology of Monte Carlo and molecular dynamics simulations.

This can be rectified by the use of active learning, where the goal is to place points in the most problematic areas, and hence can be used to identify these subregions of space and fill them with additional data prior to application. A further advantage of the sequential design methodology is that, having selected N points, the data points for all $n \leq N$ designs are also given and are subsets of the N data points. This is useful in applications to allow a convenient trade-off of accuracy and computational expense by selecting a suitable n. Of the three methods presented, the highest error search (method B) and two set search (method C) produced the lowest RMSEs for a given number of training points, with these two methods producing very similar RMSE values. Method B is preferable if the large reference LHC dataset can be computed (or already exists), as this method always returns an RMSE value. However, if the cost of computing the interaction potential prohibits the generation of a large reference set then method C must be used as it does not rely upon an initial reference set. Since the ultimate aim of machine learning methods in potential energy surface generation is to produce high-accuracy potentials at the lowest possible computational cost, active learning based methods of experimental design are a promising tool in achieving optimality in smaller training sets.

ACKNOWLEDGEMENTS

The authors are grateful to the EPSRC for the award of a studentship to EU, and to the University of Nottingham for the use of the 'Minerva' high-performance computing facility.

Appendix A: LHC generation for $\mathbf{Ar_3}$

The method for generating Ar_3 LHC data is similar to our previous work[8], with the following clarifications. The trimer system is described by three distances (r_{12}, r_{13}, r_{23}). These distances must obey the triangle constraint $r_{12} + r_{13} + r_{23} \geq 2\max(r_{12}, r_{13}, r_{23})$. Also, as the atoms are identical, the following symmetry constraint is used to restrict the size of the space: $r_{12} \leq r_{13} \leq r_{23}$. LHC data are generated via a 3-dimensional unit LHC and scaling this to a LHC on ($1/r_{12}$, $1/r_{13}$, $1/r_{23}$) using the ranges in table I. These are converted to (r_{12}, r_{13}, r_{23}) and geometries that disobey the triangle or symmetry constraint are rejected. Taking 2.88Å as the minimum value for the r_{ij} ensure that no pairwise interaction exceeds the high energy cut off. If a LHC rejects more than the mean number of rejected geometries then the entire LHC is rejected. The minimum distance within a LHC is calculated as in our previous work and a long series of candidate LHC is generated by this algorithm.

The LHC with the largest minimum distance is returned as the 'best' LHC. The test LHC data for Ar_3 are generated on $1/r$ as above. However, the training LHCs are generated using $1/r^2$ as previous work showed that this scaling gave a slightly improved RMSE for the CO_2-CO dimer.

REFERENCES

[1] C. M. Handley and P. L. A. Popelier, *The Journal of Physical Chemistry A*, 2010, **114**, 3371–3383.

[2] M. Karthikeyan and R. Vyas, in *Machine Learning Methods in Chemoinformatics for Drug Discovery*, Springer India, New Delhi, 2014, pp. 133–194.

[3] X. Deng, V. R. Joseph, A. Sudjianto and C. F. J. Wu, *Journal of the American Statistical Association*, 2009, **104**, 969–981.

[4] G. Riccardi and D. Hakkani-Tur, *IEEE Transactions on Speech and Audio Processing*, 2005, **13**, 504–511.

[5] K. Toyoura, D. Hirano, A. Seko, M. Shiga, A. Kuwabara, M. Karasuyama, K. Shitara and I. Takeuchi, *Phys. Rev. B*, 2016, **93**, 054112.

[6] J. Cui and R. V. Krems, *Journal of Physics B: Atomic, Molecular and Optical Physics*, 2016, **49**, 224001.

[7] B. Kolb, P. Marshall, B. Zhao, B. Jiang and H. Guo, *The Journal of Physical Chemistry A*, 2017, **121**, 2552–2557.

[8] E. Uteva, R. S. Graham, R. D. Wilkinson and R. J. Wheatley, *The Journal of Chemical Physics*, 2017, **147**, 161706.

[9] A. J. Cresswell, R. J. Wheatley, R. D. Wilkinson and R. S. Graham, *Faraday Discuss.*, 2016, **192**, 415–436.

[10] S. D. Whitehead and D. H. Ballard, *Machine Learning*, 1991, **7**, 45–83.

[11] S. Thrun, in *Handbook for Intelligent Control: Neural, Fuzzy and Adaptive Approaches*, Van Nostrand Reinhold, Florence, Kentucky, 1992.

[12] L. Atlas, D. Cohn, R. Ladner, M. A. El-Sharkawi and R. J. Marks, II, in *Advances in Neural*

Information Processing Systems 2, ed. D. S. Touretzky, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1990, ch. Training Connectionist Networks with Queries and Selective Sampling, pp. 566–573.

[13] J. Schmidhuber, J. Storck and S. Hochreiter, *Proc ICANN'95 (Paris)*, 1995, **2**, 159–164.

[14] D. J. C. MacKay, *Neural Computation*, 1992, **4**, 590–604.

[15] M. Rupp, M. R. Bauer, R. Wilcken, A. Lange, M. Reutlinger, F. M. Boeckler and G. Schneider, *PLOS Computational Biology*, 2014, **10**, 1–8.

[16] E. V. Podryabinkin and A. V. Shapeev, *Computational Materials Science*, 2017, **140**, 171 – 180.

[17] Z. Li, J. R. Kermode and A. De Vita, *Phys. Rev. Lett.*, 2015, **114**, 096405.

[18] Y. Guan, S. Yang and D. H. Zhang, *Molecular Physics*, 2017, **0**, 1–12.

[19] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*, The MIT Press, 2005.

[20] H.-J. Werner, P. J. Knowles, G. Knizia, F. R. Manby, M. Schütz *et al.*, *MOLPRO, version 2015.1, a package of ab initio programs*, 2015.

[21] GPy, *GPy: A Gaussian process framework in python*, `http://github.com/SheffieldML/GPy`, since 2012.

[22] D. Den Hertog, J. P. Kleijnen and A. Siem, *Journal of the Operational Research Society*, 2006, **57**, 400–409.

[23] B. Efron, in *Breakthroughs in statistics*, Springer, 1992, pp. 569–593.

[24] J. P. Kleijnen and W. C. Van Beers, *Journal of the operational research society*, 2004, **55**, 876–883.