Optimal construction of a fast and accurate polarisable water potential based on multipole moments trained by machine learning†

Chris M. Handley,[ab] Glenn I. Hawe,[ab] Douglas B. Kell[ab] and Paul L. A. Popelier*[ab]

To model liquid water correctly and to reproduce its structural, dynamic and thermodynamic properties warrants models that account accurately for electronic polarisation. We have previously demonstrated that polarisation can be represented by fluctuating multipole moments (derived by quantum chemical topology) predicted by multilayer perceptrons (MLPs) in response to the local structure of the cluster. Here we further develop this methodology of modeling polarisation enabling control of the balance between accuracy, in terms of errors in Coulomb energy and computing time. First, the predictive ability and speed of two additional machine learning methods, radial basis function neural networks (RBFNN) and Kriging, are assessed with respect to our previous MLP based polarisable water models, for water dimer, trimer, tetramer, pentamer and hexamer clusters. Compared to MLPs, we find that RBFNNs achieve a 14–26% decrease in median Coulomb energy error, with a factor 2.5–3 slowdown in speed, whilst Kriging achieves a 40–67% decrease in median energy error with a 6.5–8.5 factor slowdown in speed. Then, these compromises between accuracy and speed are improved upon through a simple multi-objective optimisation to identify Pareto-optimal combinations. Compared to the Kriging results, combinations are found that are no less accurate (at the 90th energy error percentile), yet are 58% faster for the dimer, and 26% faster for the pentamer.

1. Introduction

The importance of water with respect to life on Earth, biological systems, environmental issues and chemistry as a whole is well documented. Water is a chemical anomaly and is the subject of a vigorous and prolonged research program. Both experimental and theoretical studies aim at a deeper understanding of this apparently simple molecule. A number of reviews cover the expansive topic of water, detailing the possible causes of why water acts in the curious ways that it does.[1–5] Over the last eight decades that water has been studied theoretically, the degree of sophistication of the models has increased. In spite of an abundance of water models[6] the need for accuracy and realism in potentials remains. The issue of an accurate and transferable water model is still not resolved, hampered by the shortcuts and simplifications still in use today, which were first introduced a quarter of a century ago.

It is expected that water models will become more accurate and transferable if polarisation is explicitly represented in the model.[6–10] Mahoney and Jorgensen also note that flexibility (i.e. non-rigidity of the intramolecular geometry) should only be introduced once polarisation can be properly modelled.[11]

Related to the issue of correctly representing how the electron density of water responds to an external field, there is the need for more accurate representations of the electron density. The water molecule has an anisotropic electron density in the gas phase, and with the condensed phase the interactions with other molecules will cause further deformations of the electron density that can only be described adequately if the use of point charges is abandoned.[12–14] A popular alternative is to represent the electron density by means of multipole moments derived from *ab initio* wavefunctions.[15–17]

There are now a number of chemical potentials that make use of machine learning methods. These methods allow the functions describing the potential to emerge from the data, ideally coping with a high number of independent variables. A commonly used method is a type of artificial neural network called multilayer perceptron (MLP).[18,19] Hundreds to thousands of *ab initio* calculations[20–23] provide the data (typically energies) to teach the MLP the potential.

Here we expand on our previous work[24–26] involving MLPs and introduce two other machine learning methods called radial basis function neural networks (RBFNN) and Kriging. These two methods generate models that are capable of accurately predicting the multipole moments (output) of a water molecule in the centre of a water cluster, solely from the configuration of the cluster (input). To the best of our knowledge this is the first time that Kriging is used in the context of (inter)molecular potentials. These new models offer two alternatives for predicting the multipole moments, and in turn the polarisation response of the molecule, with respect

to a given water cluster configuration. Using the above learning methods, models are built using the same *ab initio* data that we have previously used[26] to train and test a MLP-based water model. In this work we create models for water dimer, trimer, tetramer, pentamer and hexamer clusters. The current method generates instantaneously fluctuating multipole moments that allow us to combine dynamic polarisation and charge transfer effects into a single dynamic Coulomb term. In addition, as we show below, we consider[27] water molecules appearing in water clusters as non-overlapping. The finite boundaries of the water molecules preclude both overlap and spatial gaps between the atoms (or molecules). As a consequence, there is no need to use any form of correction, such as damping functions, to correct for the so-called penetration effect.[28,29]

2. Background

2.1 Water potentials

There are two general categories of water potentials: *ab initio* potentials and empirical potentials. *Ab initio* models are those which are parameterised to fitted potential energy surfaces found by *ab initio* calculations.[30] Empirical potentials are parameterised to reproduce the bulk phase thermodynamic properties, with recent models including TIP5P[11] and TIP4P/2005.[31] The vast majority of water models use point charges, though some make use of multipole moments. The simplest models are rigid and non-polarisable ones that make use of point charges located at the nuclei and at the off-nuclear sites.[32–35] These simplifications are still followed in the SPC series of models,[36–38] the TIPS series[11,39,40] and the recent model called NvdE.[41] Another simplification followed in the past is to assume that the water molecule is non-polarisable. In the bulk phase the dipole moment of water is enhanced, by mutual polarisation, from the gas phase value of 1.85 D[42] to between 2.3 and 3.1 D in the liquid.[43–47] To account for this polarisation in the bulk phase most water models have an artificially enhanced dipole moment, such as in SPC/E and TIP4P.[36,37] An anisotropic polarisation response of water, however, is important if we consider the water molecule in heterogeneous conditions, such as on surfaces, interfaces and very polar environments. This makes non-polarisable models with implicit polarisation only accurate for bulk, homogeneous water.[6,48] It is generally agreed that polarisation needs to be modelled *explicitly* for more accurate, transferable, potentials.

2.2 Quantum chemical topology and electrostatics

It is widely accepted that multipole moments are better at representing the electron density of water (and other molecules). Gresh *et al.*,[12] Kaminsky and Jensen,[13] and Rasmussen *et al.*[14] demonstrated that a multipolar representation of electron density is vital for modelling electrostatic interactions accurately. Expressing the moments in terms of (irreducible) spherical harmonics reduces the computational cost of evaluating multipol-multipole interactions, compared to using the less compact Cartesian tensors. The former are used in distributed multipole analysis (DMA),[29] which provides multipole moments

for the ASP potential,[15,49] the AMOEBA[17] water model and the effective fragment potential (EFP) method.[50] Within the "sum of interactions between fragments *ab initio*" (SIBFA)[51,52] potential, the multipole moments are determined by the partitioning method of Vigné-Maeder and Claverie.[53] A further partitioning method that has grown from SIBFA is the Gaussian electrostatic model (GEM),[54,55] though it relies on density fitting rather than multipole moments.

Here we partition the electron density, denoted by ρ, according to the method of the quantum theory of "Atoms in Molecules",[56–58] which is part of the quantum chemical topology (QCT) approach. A justification for QCT is given in ref. 59 and the appendix of ref. 60. QCT defines topological atoms by the so-called gradient paths in ρ. Gradient paths originating at infinity follow the direction of steepest ascent. Typically, they terminate at nuclei. An atomic volume is defined by the three-dimensional bundle of gradient paths that terminate at a given nucleus. A different (two-dimensional) bundle of gradient paths form an interatomic surface that marks the boundary between two atoms. It is important to note that there are no gaps between topological atoms. Secondly, collectively the atoms take up all space. Integration of property densities over the atomic volume yields the atomic properties. If the integrand of the volume integral is a regular spherical harmonic multiplied by ρ, then the corresponding electrostatic multipole moment emerges.

Eqn (1) expresses the electrostatic interaction energy between two topological atoms,[61]

$$E^{AB} = \sum_{l_A=0}^{\infty} \sum_{l_B=0}^{\infty} \sum_{m_A=-l_A}^{l_A} \sum_{m_B=-l_B}^{l_B} T_{l_A m_A l_B m_B}(\boldsymbol{R}) Q_{l_A m_A}(\Omega_A) Q_{l_B m_B}(\Omega_B)$$

(1)

where the multipole moments of atom A, $Q_{l_A m_A}(\Omega_A)$, and atom B, $Q_{l_B m_B}(\Omega_B)$, interact through the tensor $T(\boldsymbol{R})$. $\boldsymbol{R}$ is the vector from nucleus A to nucleus B, which are the origins of the local frames of atoms A and B. It is convenient to collect the terms of eqn (1) by their power of $R = |\boldsymbol{R}|$, gathering terms of the same rank, L, defined as $\ell_A + \ell_B + 1$, where ℓ is the rank of the multipole moment. For example, R^{-4} dependence consists of interactions between a monopole moment ($\ell = 0$) and an octupole moment ($\ell = 3$) and a dipole moment ($\ell = 1$) and a quadrupole moment ($\ell = 2$). We monitor the convergence of the multipole expansion by varying L. We implemented Hättig's recurrence formula[62] to enable calculation of interaction tensors of arbitrarily high rank. The exact Coulomb interaction energy can be obtained *via* a six-dimensional integration over the two participating atoms,

$$E^{AB} = \int_{\Omega_A} d\boldsymbol{r}_A \int_{\Omega_B} d\boldsymbol{r}_B \frac{\rho_{tot}(\boldsymbol{r}_A)\rho_{tot}(\boldsymbol{r}_B)}{r_{AB}} \quad (2)$$

where r_{AB} is the distance between two infinitesimally small charge elements, and ρ_{tot} is the total charge density (which includes the nuclear charge). Obviously, eqn (2) does not impose a convergence condition, and can hence be used as a point of reference for eqn (1), which does not necessarily converge. Secondly, we have made a distinction before[63] between the terms "electrostatic" and "Coulomb". The former

6366 | *Phys. Chem. Chem. Phys.*, 2009, **11**, 6365–6376

This journal is © the Owner Societies 2009

term is only well defined in the context of (long range) intermolecular perturbation theory, while the latter applies to the interaction of any charge densities, whether in an intra- or intermolecular context. Since we will sample the electron density from atoms in supermolecules (*i.e.* water clusters) we are not working in a perturbation context and hence the term Coulomb is more appropriate. However, some texts use the two terms interchangeably.

QCT multipole moments are successful in MD simulations of liquid hydrogen fluoride and water[16,64,65] and aqueous solutions of imidazole as well as neat liquid imidazole.[66]

2.3 Polarisation

Polarisation causes anywhere up to a 70% increase in the dipole moment of water, and it is often quoted that $\sim$15% of the total interaction energy[67,68] is due to polarisation. However, energy contributions have been stated as being as high as 50%.[69] Polarisation can easily be incorporated into a model implicitly by fitting the model parameters so that the experimental bulk phase properties are recovered. This way of including polarisation fails when a dynamic anisotropic response of the electron density is required, when the molecule is in heterogeneous environments. Polarisation can be modelled by polarisable point dipoles. This method can suffer from the "polarisation catastrophe", where the dipoles respond in such a way that the interaction energy becomes infinite. This is prevented by a damping function, which limits the response of the dipole moments.[70–73] This method of representing polarisation is popular and appears in a number of force fields.[17,51,74–77] A second method is the charge-on-spring method.[78–81] Here polarisation is modelled by a negative point charge connected by a harmonic spring to another site that bears a positive charge. Finally, the fluctuating charge method models the polarisation by allowing the charges at atomic sites to change in response to the external field. This method is seen in a modification of the TIP4P water model, in the POL5 model[82,83] and in the SIBFA model.[84]

In the context of QCT, we should mention here that topological distributed (multipole) polarisabilities[85] have been computed and analysed before for the water dimer.[86] In this work, we abandon this route for the current alternative of machine learning, capturing how multipole moments respond to variations in their environment. The latter approach does not yield polarisabilities; instead, the polarisation is embedded in the mapping between input (environment coordinates) and output (multipole moments) provided by the machine learning solution. Previously, Houlding *et al.* proposed a novel method for including polarisation within a simulation of a hydrogen fluoride dimer[24] using dynamic QCT multipole moments. MLPs were trained on the *ab initio* data of thousands of HF dimer configurations. The MLPs then predicted the atomic multipole moments of HF in response to changes in the local orientation of the HF molecules. Building upon this new methodology, we demonstrated that the same method could be used in the construction of a polarisable water potential.[26] In that work, polarisation was modelled for a water molecule in centre of water dimer, trimer, tetramer, pentamer and hexamer clusters. In conjunction with the water model, we

also presented a method[25] to tackle *intramolecular* polarisation in a similar manner. Here polarisation was the electronic response of the molecule to changes in the molecular geometry, for *N*-methylacetamide and glycine.

Here we introduce two new machine learning methods, RBFNNs and Kriging, which are both capable of modelling polarisation for water clusters. We also compare the ability of these two new methods and MLPs, to correctly predict multipole moments that reproduce the correct electrostatic interaction energies. The relative merits and limitations of all three methods are discussed. We consider the option of a 'best of all worlds' approach for constructing the best possible model.

2.4 Radial basis function neural networks (RBFNN)

In previous work[26] MLPs were used for the prediction of the multipole moments in water. Here we use an RBFNN,[87,88] which is a three-layer feedforward artificial neural network. The RBFNN performs a *nonlinear* mapping between the *d*-dimensional feature space and a hidden layer, followed by a linear mapping from the hidden layer to the output. A *feature* is a descriptor of the system whose properties are being predicted. In our case the features are the angles and distances that fully define a water cluster configuration; the value of *d* depends on the number of water molecules in the cluster, and how the features are represented—see section 3.2 for more details. Fig. 1 shows the architecture of an RBFNN, suitable for a water dimer. There are six input nodes (or neurons) in the input layer, four nodes in the (single) hidden layer and one node in the output layer. The connections between the nodes are associated with adjustable weights. During training, the weights of the linear mapping are varied so as to minimise the difference between the known output values (in the training set) and their predicted values. However, exact interpolation of the training data can be achieved as we show in the ESI, Section A.†
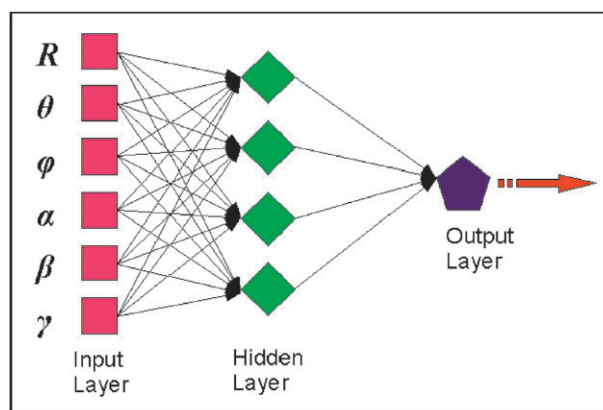

Fig. 1 Diagram of a feedforward neural network with one hidden layer. The pink squares are input nodes, the green diamonds hidden nodes and the purple pentagon is an output node. In this work, the output is a multipole moment of a given atom and the inputs are the polar and Euler coordinates of the neighbouring water molecules, shown here for the water dimer.

This journal is © the Owner Societies 2009

Phys. Chem. Chem. Phys., 2009, **11**, 6365–6376 | 6367

2.5 Kriging

Kriging[89,90] (also known as Gaussian Process regression[91]) is a machine learning method that models observations as realisations of an underlying probabilistic model. This method is rarely used in computational chemistry but featured in QSAR recently.[92] In our case, the observations are values of a particular multipole moment. In this work, the probabilistic model is a *probability density function* (PDF) that gives the probability of observing a set of values of a particular multipole moment within a certain domain. We assume that two moment values arising from different water cluster configurations are correlated in a way that may be expressed through the difference in feature values of the configurations of the two water clusters. The parameters of the PDF are chosen to be most consistent with the training data, *i.e.* they are chosen to maximise *the likelihood of the model given the observed data*. The likelihood can be described as follows. Let $f(y|\Theta)$ denote the PDF that gives the probability of observing an *m*-dimensional vector of data $y = (y_1, y_2, \ldots, y_m)$, given the parameter Θ. This parameter is a vector defined in a *k*-dimensional parameter space, $\Theta = (\Theta_1, \Theta_2, \ldots, \Theta_k)$. Given a particular set of parameter values, the PDF will be higher for some sets of observations than others. In other words, given a probabilistic model, and a set of parameters for that model, some sets of observations will be more probable than others. However, *the data have already been observed*, *i.e.* the moment values have already been determined for a range of water cluster configurations. Hence there is no point in finding a most probable set of observations given a particular model (with a fixed set of parameters). Instead, we solve the following problem: given the set of observed multipole moments and a probabilistic model of interest (shortly to be defined), *find the one PDF (i.e. Θ) that is most likely to have produced the data*. The likelihood function, defined as $L(\Theta|y) = f(y|\Theta)$, allows us to solve this problem. It gives the "likelihood" of the parameter Θ given the observed data y. Note that the PDF $f(y|\Theta)$ is a function of the data y, given a particular parameter vector Θ. However, *the likelihood $L(\Theta|y)$ is a function of the parameters of the probabilistic model, given a particular set of observations*.

Returning specifically to Kriging, N observations (of a particular multipole moment) are modelled as N observations coming from Gaussian random variables indexed over feature space (a *Gaussian process*). In other words, the training data are modelled as a realisation of an *N*-variate Gaussian distribution. By maximising the likelihood of the training data, the optimal parameters of this multivariate Gaussian distribution are found. Using the master formula eqn (B12), given and derived in Section B of the ESI,† predictions can be made for unseen combinations of features. This means that, in our case, moment values may be predicted for unseen configurations of water molecules.

The following section deals with some of the computational details behind the methods discussed, such as the selection of appropriate training sets and feature selection.

3. Computational details

3.1 Molecular local frame

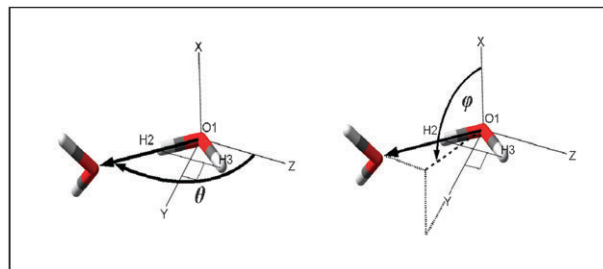Multipole moments allow for an anisotropic description of the electron density of atoms. Multipole moments must be

Fig. 2 The molecular local frame (MLF) of a central water molecule. The left frame defines the angle θ while the right frame defines the angle ϕ, subtended by the x axis and the projection of the line segment connecting the two oxygen atoms onto the xy plane. These two polar angles appear as features in Fig. 1.

correctly aligned in space relative to each other because the electrostatic interaction between moments is orientation dependent. In our previous work[25] on glycine and *N*-methylacetamide we introduced an atomic local frame (ALF). In this work we use a molecular local frame (MLF) because the water molecules are considered to be rigid bodies. As a result the atomic multipole moments for a particular water molecule all have the same local frame. Fig. 2 shows the details of a MLF installed on a central water molecule, at the origin, in a water dimer. The MLF is simply defined by aligning the positive *y*-axis along the HOH bisector, with both hydrogen atoms in the positive *y* direction. The *yz*-plane is then defined by the HOH plane, with the H2 hydrogen atom lying at the negative side of *z*-axis. The angles θ and ϕ, which will be discussed below, enable the positioning of the second water molecule (or in general, other non-central water molecules).

This orientation convention determines *both* the generation of the training data for the machine learning methods *and* the prediction of the moments by them. The MLF defines the rotation of the moments from the MLF into the global frame. The training data, that is, the multipole moments, are generated with respect to the MLF for every training example. Consequently, the machine learning methods will predict moments with respect to the MLF. This means that *the orientation of the MLF is embedded in the training data*.

3.2 Feature representation and feature selection

The inputs for the machine learning methods are generated using a transformation from the Cartesian coordinates of the cluster to a set of non-redundant coordinates following the method laid out by Stone.[29] For systems of rigid and non-linear molecules only $6(N - 1)$ coordinates are needed, where N is the number of molecules. For example, a water trimer is completely described by $6(3 - 1) = 12$ coordinates, and a dimer by only $6(2 - 1) = 6$ coordinates. With the central molecule at the origin of the MLF and aligned as described above (see Fig. 2), the position of the neighbouring water molecule is described by three polar coordinates and three Euler angles. The polar coordinates (referred to in Fig. 1) are the distance R_{OO} between the central water oxygen atom and the oxygen atom of the neighbouring water, the angle θ spanning the vector R_{OO} and the z axis, and the angle ϕ spanning the (positive) x axis and the projection of the vector

R_{OO} onto the xy plane. The three Euler angles, α, β and γ describe the water neighbouring the central water molecule as follows. Position the neighbouring water in the axis system of the central water such that the HOH plane lies in the xz plane, with the hydrogen atoms in the negative z direction. Then a succession of three rigid body rotations corresponding to the three Euler angles is applied. Finally this water is translated by R_{OO}. This procedure can be easily generalised to an arbitrary number of neighbouring water molecules.

After defining the internal coordinates of each configuration (for a given cluster size), the data are normalised, that is, transformed to lie between 0 and 1, for subsequent exploitation by the machine learning methods. Training operates on these transformed input and output data. As the number of features increases with cluster size, the training of the machine learning methods becomes computationally more expensive. For RBFNNs, the radial distances that need to be calculated between training points become computationally more expensive. In Kriging, the situation is worse, as training is an optimisation problem with $2d$ variables, where d is the number of features describing the water cluster in question. Therefore, for a water cluster of N molecules, constructing a Kriging model becomes an optimisation problem with $2d*6(N-1) = 12d(N-1)$ variables. To alleviate this increase in computational cost for both RBFNNs and Kriging models, feature selection—the identification and removal of unimportant features—is applied.

With RBFNNs, the backward sequential selection method was used.[93] Starting with all features present, an RBFNN is built and the mean absolute error (MAE) on an external test set is evaluated. Then, each feature is removed in turn, and the RBFNN rebuilt. The feature whose absence leads to the largest decrease in MAE is then removed permanently from the set of features. Then the process iterates: features are sequentially removed, until no further improvements in MAE are gained. The set of features that minimises the MAE is the set used to build each RBFNN. This process is applied to each RBFNN built, so different features may be selected for different moments. The backward sequential selection method is an example of a "wrapper" approach: each iteration in the search for the optimal subset of features involves retraining and then using the machine learning model (in this case RBFNN) to determine the performance of a particular subset.

With Kriging, due to the computational cost of building each model, a wrapper approach was not appropriate. Instead, a method involving the analysis of the Kriging parameters[94] was used. At a relatively early stage during the sequential building of a Kriging model, when the training set size is 250, the values of the Kriging parameter θ are examined. Recall that associated with each feature is a component of the vector θ. In our application all features have the same range (between 0 and 1). When this is the case, the relative values of θ reflect the relative importance of the features.[94] After optimisation of the Kriging parameters θ and p, components of θ whose value are very small relative to the other components indicate that the features they correspond to are relatively unimportant, and are thus removed. More precisely, after optimisation, the components $\theta_1, \theta_2, \ldots, \theta_d$ of θ are examined in turn. If each

feature is equally important, then $\theta_1 = \theta_2 = \cdots = \theta_d$, i.e., $\theta_i = \frac{1}{d} \sum_{j=1}^{d} \theta_j$, where $i = 1, 2, \ldots, d$. However, if for some i the following holds $\theta_i < \frac{\varepsilon}{d} \sum_{j=1}^{d} \theta_j$ (where $0 < \varepsilon \ll 1$) then the value of θ_i is set to zero, corresponding to the removal of the i-th feature. The threshold used in this paper is $\varepsilon = 0.01$.

3.3 Data generation and selection of training sets

Previously,[26] configurations were taken from molecular dynamics simulations[66] of pure liquid water at ambient conditions performed with non-polarised (gas phase) QCT multipole moments. For each water molecule in the simulation, the nearest neighbours were found that would form the water dimer, trimer, tetramer, pentamer or hexamer clusters. Thus, the clusters were built in a hierarchical manner about a central water molecule, and as a consequence, the dimer clusters lie within their trimer clusters, the trimer clusters lie within the tetramer clusters, and so forth. The program GAUSSIAN03[95] generated wavefunctions at B3LYP/aug-cc-pVTZ level, for each cluster configuration, without geometry optimisation. However, due to the modular nature of the approach, the electron density may be obtained from wavefunctions generated at other levels of theory, possibly more advanced, future computing power allowing. For the central water molecule (which lies in the MLF as described above) the program MORPHY[96–98] generated the multipole moments for each of the atoms.

The number of cluster configurations (or geometries) produced for the training in our previous work[60] was determined by the MLP's architecture, more specifically, the number of weights. We applied the rule of thumb to have approximately 10 training examples (i.e. cluster configurations) for every weight being trained, giving rise to data-sets of around 5000 configurations for each cluster size. It is from these data-sets that appropriate training sets are to be chosen for the RBFNN and Kriging models.
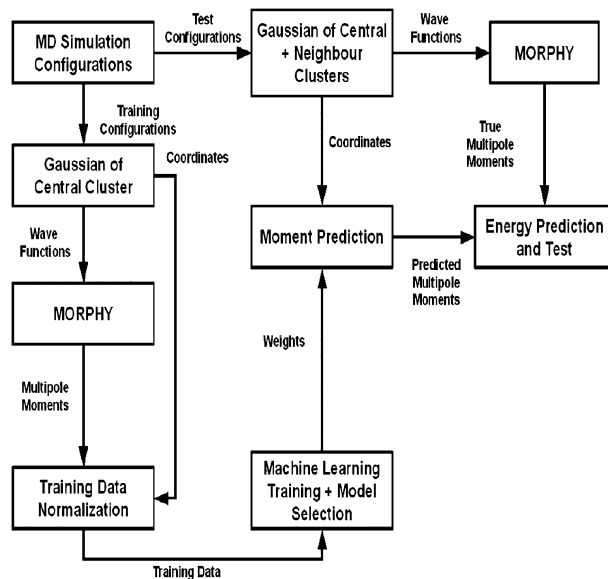

Fig. 3 Schematic of the sequence of processes followed to generate and test the models.

This journal is © the Owner Societies 2009

Phys. Chem. Chem. Phys., 2009, **11**, 6365–6376 | 6369

Fig. 3 shows a schematic of the sequence of processes followed to generate and test the models. The starting point in this scheme is the upper left corner, where the training and test configurations are sampled from the same MD simulation. Wavefunctions are computed for the training configurations ("Gaussian of Central Cluster") and for the test configurations ("Gaussian of Central + Neighbour Clusters"). The atomic multipole moments are then computed from the wavefunctions via the electron density ("MORPHY"). The training data for the machine learning methods (both input and output) are then normalised prior to training ("Training Data Normalisation"). Subsequently, the training data are used to train a machine learning method for each moment of each atom. Note that the machine learning methods predict the multipole moments solely based upon the coordinates of the nearest neighbours of a given water molecule. The predicted moments are then compared with the true moments for a given test configuration (middle right of diagram in Fig. 3). The quality of the training is then assessed by the correlation between the predicted and true moments, and atom-atom electrostatic interaction energies.

Following suggestions in ref. 99, we use a method of sequential selection for the construction of the training sets. This procedure is valid for both Kriging models and RBFNNs. The SS1040 approach is ambiguous as to what size the initial training set should be, how many rebuilds to carry out, and what stopping criteria to use. Indeed Rennen concludes[99] that the scope remains "to develop methods which dynamically determine a suitable subset size". Thus we use the following variation:

1. Define a space-filling set of points S in the feature-space using a Sobol[100] sequence of size 50, and determine the points in the data-set that are closest to the points in S. Use these points to construct an initial training set.

2. Construct a Kriging (or RBFNN) model using the training set. For each point not in the training set, calculate the absolute value of prediction error.

3. Determine the 200 points with highest absolute prediction errors, and from these, use a MAXMIN algorithm to select 50 well-spaced points. The MAXMIN algorithm attempts to maximise the minimum distance between any 2 of the 50 points chosen. Add these 50 points to the training set, and go to step 2, until the training set size exceeds 1000.

4. Twenty rebuilds have taken place in total, giving rise to 20 models (each built using a training set of a different size). The model with the lowest mean absolute error (MAE) (calculated using all the points not in the training set), is the model chosen for use.

3.4 External validation

For *external* testing of the performance of the machine learning methods, a further 1000 configurations were generated. For the dimer clusters, the moments for each of the molecules were generated. These are the *true* moments. Using these moments we can judge the ability of the machine learning methods to predict moments for the water molecules for a given dimer configuration and also the electrostatic interaction energy. For larger clusters we follow the procedure in ref. 26, the main points of which are summarised here for convenience.

For the trimer clusters and larger, 1000 clusters of 50 molecules were generated to act as the test configurations. The arbitrary number 50 turned out to be large enough to ensure that each water molecule that is a member of the central clusters we investigate (trimer, tetramer, etc.) can see its own first solvation shell. For the central water, the $n - 1$ nearest neighbours are identified ($n \geq 3$ where n is the total number of molecules in the cluster). From the coordinates of the central molecule and these neighbours the *true* moments are found for the central water molecule. The process is then repeated for each of the $n - 1$ neighbours of the central cluster, where for these neighbours we find their own nearest neighbours and predict moments for these clusters. This ensures that we have the true moments for each of the water molecules based upon their own $n - 1$ nearest neighbours as these are the configurations that are seen when the machine learning methods predict multipole moments for each water molecule in the central cluster of n molecules. For further details, the reader is referred to ref. 26.

The performance of the machine-learning methods are to be judged not on the accuracy of the multipole moment predictions themselves, but rather on the more meaningful total and individual atom-atom electrostatic interaction energies, the values of which emerge as a consequence of the machine-learning predictions. Upon making predictions over an external set of 1000 configurations, statistical values relating to the accuracy of the methods may be deduced.

3.5 Computational cost of predictions and Pareto-optimal selection of models

Recall that in order to predict electrostatic energies using multipole moments up to rank 5, we need 75 machine learning models in total (25 for each atom in the water molecule). In this paper, three different machine learning methods are being used to predict multipole moments: RBFNNs and Kriging models, along with the previously investigated MLPs. Using only one particular machine learning method to build all 75 models offers three different approaches, each representing a different compromise between accuracy and speed. However, nothing restricts us to using just one type of machine learning method to build all 75 models; rather, we have the freedom to use any combination of the machine learning methods at our disposal. Indeed, as shall be seen in section 4, using combinations of different machine learning methods allows combinations to be formed which dominate (see below for a definition) the three "one-model only" compromises, and so is a worthwhile exercise. The question arises then, how to determine the best combinations.

In terms of making predictions, Kriging models have a computational overhead that depends on both the number of points N in the training set, and on the dimension d of feature space, $O(Nd)$. Both MPLs and RBFNNs have a computational overhead that depends on the number of nodes in the hidden layer M and on the number of features d, $O(Md)$. For RBFNNs, the number of nodes in the hidden layer is equal to the size of the training set $M = N$, giving a computational cost of $O(Nd)$.

For MLPs, however, $M \neq N$ in general. This discrepancy in computational effort ($O(Nd)$ versus $O(Md)$) to make a prediction can be exploited along with any discrepancy in accuracies between the models, to give combinations of their use such that predictions of electrostatic energies are both accurate *and* fast. The problem to be solved is thus a combinatorial multi-objective optimisation problem (MOOP): determine which machine learning method to use to predict each multipole moment on each atom in the water molecule, such that the time taken to predict electrostatic energy is minimised, whilst the accuracy of the predictions is maximised.

We define the 90th *energy error percentile*, E_{90}, as the error in energy that 90% of predictions are within, as determined over a set of predictions of approximately 1000 configurations. Formally, the 90th energy error percentile of the electrostatic energy predictions is a function of 75 different models, or $E_{90}(m_1, m_2, \ldots, m_{75})$. Each model corresponds to a particular moment on a particular atom: in our case there are 3 atoms each with 25 moments, hence leading to a total of 75 different models. The average time taken to make one energy prediction is also a function of these models, or $T(m_1, m_2, \ldots, m_{75})$. We now want to solve the following:

$$\text{Minimise } E_{90}(m_1, m_2, \ldots, m_{75}) \text{ and } T(m_1, m_2, \ldots, m_{75})$$

where each of $m_1, m_2, \ldots, m_{75}$ are MLPs, RBFNNs or Kriging models. The models $m_1, m_2, \ldots, m_{75}$ are the decision variables of the problem whilst E_{90} and T are the competing objectives. In our case there are 3^{75} different possible combinations of machine learning models to search over (75 separate models are to be built with 3 different types of methods). Such a large search space would lead to difficulties for conventional algorithms such as the NSGA-II algorithm,[101] so we simplify the problem in two ways. First, the same method is used for all multipole moments sharing the same rank ℓ. For example, all three components of the dipole moment must be modelled by say Kriging. This reduces the number of decision variables per atom from 25 to 5, since ℓ adopts values between 0 and 4. Hence, the *total* number of decision variables is reduced from 75 ($= 3 \times 25$) to 15 ($= 3 \times 5$). Secondly, for each particular moment, the same method is used for each atom. This further reduces the total number of decision variables from 15 ($= 3 \times 5$) to 5 ($= 1 \times 5$). With these two simplifications, the search space is reduced to 243 ($= 3^5$) combinations, which is small enough to perform an exhaustive search. To understand the set of solutions returned, the definition of *Pareto-optimality* is required, which itself requires the definition of *dominance*:

A solution x_1 to a MOOP is said to *dominate* another solution x_2 if and only if x_1 is strictly better in one objective than x_2 and no worse in the remaining objectives. For any two solutions x_1 and x_2, it is the case that either x_1 dominates x_2, x_2 dominates x_1, or neither solution dominates the other, in which case they are said to be *equivalent*. The concept is illustrated for a two-objective problem in Fig. 4.

If, for a particular solution x^* to a MOOP, no other solution exists in the entire search space which dominates it, then x^* is said to be Pareto-optimal. The set of all such solutions is called the *Pareto-optimal set*, and its image in objective-function space is called the *Pareto-optimal front*.
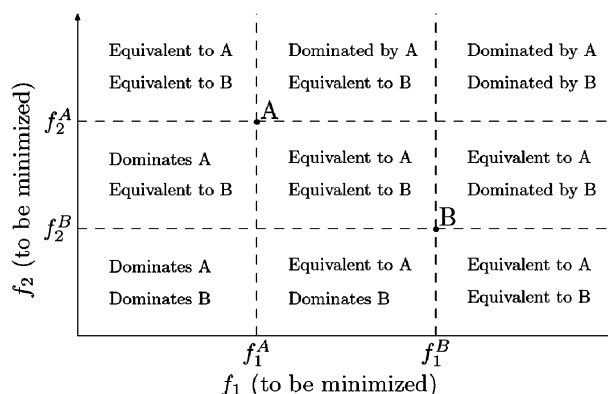

Fig. 4 Illustration of the concept of dominance in multi-objective optimisation. Depending on where a given solution lies in objective function space in relation to another solution (in this case A or B), a solution is worse than A ("is dominated by A"), is better than A ("dominates A"), or is equivalent to A ("neither dominates A, nor is dominated by A"). Similarly for B.

Fig. 4 illustrates the concept of dominance for two solutions, A and B, to a two-objective problem, where both objectives f_1 and f_2 are to be minimised ("lower is better"). For example, solutions that fall within the bottom left hand box in objective function space have *both* f_1 and f_2 values less than those of both A and B. Thus, solutions in this region dominate both A and B. In the box immediately above it, both the objective function values are still less than those of A, and so solutions in this box also dominate A. However, solutions in this region do not dominate B as their f_2 values are not better than those of B; as their f_1 values are better than those of B, B does not dominate these solutions either. Hence solutions in this region are equivalent to B. Similar logic can be applied to each of the regions to determine whether a solution dominates, is equivalent to, or is dominated by another solution.

Thus, our search attempts to identify the Pareto-optimal set of combinations of machine-learning methods to predict the multipole moments with. The selection of which particular combination to use is then determined by individual preference of the level of trade-off between accuracy and speed.

4. Results and discussion

Table 1 shows the performance of the three machine learning methods when predicting the multipole moments of the central water molecule in the dimer, trimer, tetramer, pentamer and hexamer clusters. For the central water molecule all the multipole moments of each atom in the molecule are predicted using the same machine learning method. We can judge the performance of each machine learning method in two ways: accuracy and the time it takes for the method to predict the multipole moments. The accuracy of each method is determined using the 50th, 90th and 99th interaction energy error percentiles (E_{50}, E_{90}, E_{99}) when using approximately 1000 test configurations. The *total interaction energy error* is the absolute value of the difference between the *total* inter-molecular electrostatic interaction energy when using the predicted moments and the true moments for a particular configuration. The time, T, is the average time it takes for the

This journal is © the Owner Societies 2009

Phys. Chem. Chem. Phys., 2009, **11**, 6365–6376 | **6371**

Table 1 The 50th, 90th and 99th energy error percentiles E_{50}, E_{90}, and E_{99} (kJ mol^{-1}), and the time taken for one energy prediction T (s), using the three different types of machine learning methods, for the water dimer, trimer, tetramer, pentamer and hexamer

Method	Dimer				Trimer				Tetramer				Pentamer				Hexamer			
	E_{50}	E_{90}	E_{99}	T	E_{50}	E_{90}	E_{99}	T	E_{50}	E_{90}	E_{99}	T	E_{50}	E_{90}	E_{99}	T	E_{50}	E_{90}	E_{99}	T
Krig	0.4	1.0	3.4	0.11	1.2	3.3	8.5	0.32	2.3	6.3	12.5	0.68	3.7	10.0	16.7	1.41	4.9	13.4	29.6	2.06
RBFNN	0.4	1.4	3.1	0.06	1.6	4.4	9.3	0.14	4.6	10.2	18.5	0.25	8.5	19.0	27.5	0.46	11.0	23.0	39.7	0.66
MLP	0.4	1.2	3.6	0.01	2.0	5.3	11.4	0.05	4.3	11.4	21.3	0.09	9.9	21.0	30.6	0.18	15.0	27.7	41.9	0.24

prediction and evaluation of the multipole moments and total interaction energy of a single configuration (note that all times quoted are using an Intel Core 2 Quad Q6600 @ 2.40 GHz with 3.25 GB of RAM). We find that E_{99} is not the most suitable judge of the performance of the prediction method since this percentile is almost equivalent to finding the maximum energy error for each method. Very high percentiles (*e.g.* E_{99}, E_{98}, E_{95}) can be heavily influenced by a low number of very poor predictions. This means that E_{99} is not the best discriminator to use when determining which method is most accurate. However, the other two percentiles E_{50}, E_{90} are far more useful. Qualitatively, for all cluster sizes we find that Kriging gives lower (energy) errors than RBFNN, which in turn gives lower errors than MLP. This trend is not followed for the dimer cluster. The general trend in performance is reversed when we consider the time it takes to perform a prediction of the multipole moments. MLP is faster than RBFNN, which is faster than Kriging. Thus Kriging sacrifices speed compared to MLP and RBFNN but it is more accurate. Quantitatively, beyond the dimer, there is a 40–67% improvement in E_{50} (*i.e.* a 40–67% decrease in median energy error) using Kriging models, taking the MLP performance as baseline. The improvements are greater for greater cluster size: for example, for the trimer there is a 40% decrease in the median (E_{50}) energy error (1.2 kJ mol^{-1} compared to 2.0 kJ mol^{-1}), whilst for the hexamer there is a 67% decrease in the median energy error (4.9 kJ mol^{-1} compared to 15.0 kJ mol^{-1}). With the exception of the tetramer, there is a 14–26% improvement in E_{50} using RBFNNs. At the E_{90} level, the improvement is 38–52% for Kriging (again with greater improvements for greater cluster sizes), whilst for RBFNNs the improvement is in the range 10–17%. The penalty paid for these improvements is a slowdown factor of 6.5–8.5 for Kriging models (again, taking MLPs as a baseline), and a typical slowdown factor of 2.5–3 for RBFNNs. The choice of which model to use, and thus the exact trade-off between accuracy and speed, is ultimately subjective. A more precise balance involving combinations of different methods, is discussed later.

We can also judge the performance of each method by considering the accuracy of the dipole moments for each water molecule, given in Table 2. Using each prediction method the molecular dipole moment of the central water molecule in the pentamer cluster is reconstructed from the predicted atomic multipole moments. In general we find that Kriging shows a lower average and maximum absolute error, followed by RBFNN and then MLP. Quantitatively, Kriging shows an 8% improvement in the average absolute error, and a 16% improvement in the maximum absolute error, compared to the MLP. The prediction of the molecular dipole moment is a

Table 2 Errors (a.u.) in the dipole moment of the central water molecule in pentamer clusters for different machine learning methods

Machine learning method	Ave Abs[a]	Max Abs[a]
Kriging	0.066	0.307
RBFNN	0.071	0.309
MLP	0.072	0.366
Pareto 1 (Kriging, RBFNN)[b]	0.068	0.336
Pareto 2 (RBFNN, RBFNN)	0.071	0.309
Pareto 3 (Kriging, RBFNN)	0.068	0.336
Pareto 4 (RBFNN, MLP)	0.071	0.342
Pareto 5 (Kriging, RBFNN)	0.068	0.336

[a] Average and maximum of the absolute values. [b] Models used for each Pareto combination are (charge model, dipole model).

good test as it relies on the prediction of the atomic monopoles and dipole moments. These moments account for the majority of the electrostatic interactions in water. This is because the interaction L is equal to $\ell_A + \ell_B + 1$, and ℓ values of 0 and 1 appear in more combinations making up all L values. For example, $\ell = 0$ and $\ell = 1$ can appear together in $L = 2$, $L = 3$, $L = 4$ and $L = 5$. Conversely, $\ell = 3$, for instance, can only appear in $L = 4$ and $L = 5$. A detailed study[102] on the long-range convergence behaviour of multipole expansions with topological atoms yielded a relationship between the internuclear distance, the rank L, atom types and the energy accuracy. Finally, we note that the accurate prediction of molecular dipole moment is critical for accurate simulation of water due to the important influence of the dipole moment of water on the bulk liquid thermodynamic and structural properties.

Fig. 5a shows all the energy error percentiles (E_n, $0 < n \leq 100$) in the water dimer. The typical S-shaped curves complete the information shown in Table 1. Fig. 5 presents the S-shaped curve associated with the best MLP achieved by varying all training parameters and training, early stopping and validation sets to obtain the net with the best generalisation and performance. Fig. 5 also shows the curves found using the unpolarised QCT model and TIP3P, from our previous work.[26] We also show the best S-shaped curves when using the RBFNNs and Kriging methods. RBFNN and Kriging models outperform the unpolarised QCT model and TIP3P. All three machine learning methods give models that show a similar performance, with Kriging showing the better accuracy. The inset of Fig. 5 highlights that the relative performance of Kriging, RBFNN and MLP remains conserved in the very high percentiles ($E_{>88}$).

Fig. 5b is the equivalent plot of Fig. 5a but now for the pentamer. In our previous work[26] we had found that the unpolarised QCT and TIP3P models showed a better performance than MLP. Further investigation into the
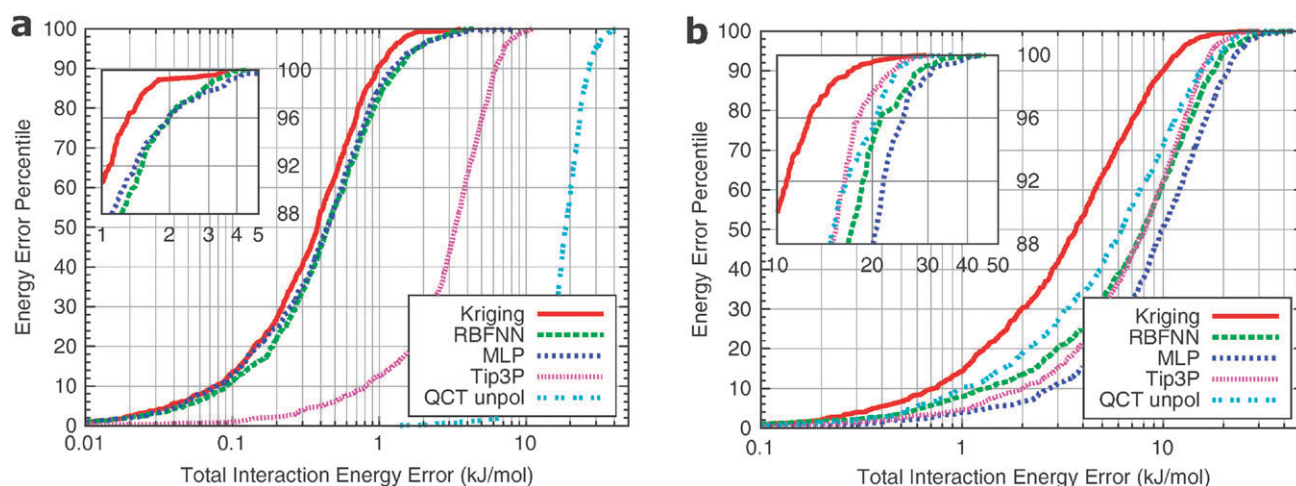
Fig. 5 The energy error percentiles for the water (a) dimer and (b) pentamer.

interaction energy errors revealed that the S-shaped curves were misleading for the pentamer. When we consider the energy errors for the *individual* atom pair intermolecular interactions we find that the unpolarised QCT and TIP3P models are on average in error by 50%. However, this error disappears for the *total* interaction energies in the cluster due to fortuitous cancellation of individual errors. However, non-polarisable models are only useful for simulations of water in the bulk liquid and not in other heterogeneous conditions. It would appear that the pentamer cluster mimics the bulk liquid (a central water molecule surrounded by 4 nearest neighbours is much like a water molecule with its first solvation shell). However, Fig. 5b shows that Kriging and RBFNN models show a dramatic improvement in the prediction of the total interaction energies while having accurate atom pair intermolecular interaction energies.

Using a selection of energy error percentiles we are able to summarise all the S-shaped curves for all three methods. Fig. 6 shows the 50th and 90th energy error percentiles (E_{50} and E_{90}) for each method for each cluster size. From Fig. 6 it is clear that Kriging has a shallower slope, followed by RBFNN and MLP. This figure highlights that Kriging is less affected by increasing cluster size. This means that Kriging methods are better suited to larger dimensional input spaces compared to the other two methods.

From Table 1 it was clear that the Kriging method, though accurate, was slower than the other two methods. Trading off

accuracy for computational speed, we have developed a variety of models that use different combinations of methods for the prediction of all the multipole moments (as discussed at the end of section 3). Fig. 7 shows the performance for all the model combinations investigated for the water dimer and pentamer, in terms of the time to predict the multipole moments of a single water molecule, and the accuracy of the model given by the 90th energy error percentile. The green crosses denote dominated models while red crosses are the Pareto-optimal models. Among the 243 combinations in each figure, the three which use only one particular type of machine learning method to build all 75 models are highlighted. With the exception of the combination that uses MLPs for every model (which always has the fastest speed, and so is never dominated), these combinations are *not* Pareto-optimal, thus justifying the investigation into combining the use of the three different machine learning methods. The Pareto-optimal models represent a series of models from which we can select, depending on how quick or accurate we wish our predictions to be. Of course we wish our models to be more accurate than the unpolarised QCT and TIP3P models. This means that the Pareto models must fall below this upper energy error marked in Fig. 7b (pentamer).

Fig. 8a shows the S-shaped curves for 5 Pareto models for the water dimer, selected from Fig. 7a. These Pareto models show a range of speed and accuracy. Going from left to right on the inset of Fig. 8a, the accuracy of the Pareto models
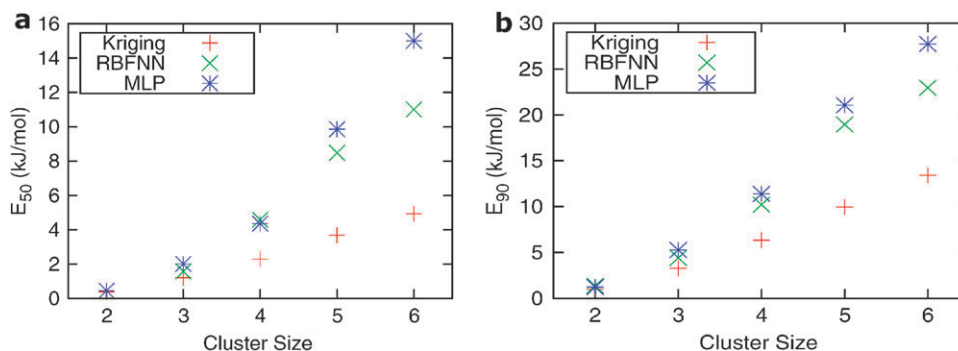

Fig. 6 The energy error percentiles (a) E_{50} and (b) E_{90} (kJ mol^{-1}) *versus* cluster size for Kriging, RBFNN and MLP.

This journal is © the Owner Societies 2009

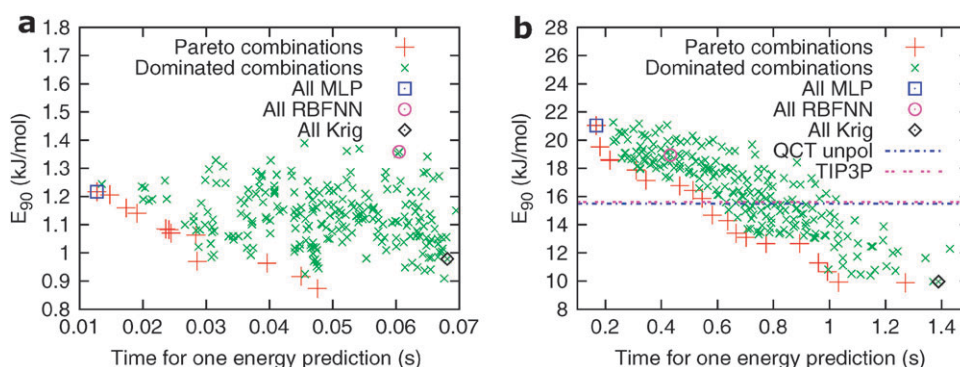*Phys. Chem. Chem. Phys.*, 2009, **11**, 6365–6376 | 6373

Fig. 7 The trade-off between the speed of predictions and the 90th energy error percentile (E_{90}) using combinations of different machine learning methods for the water (a) dimer and (b) pentamer.
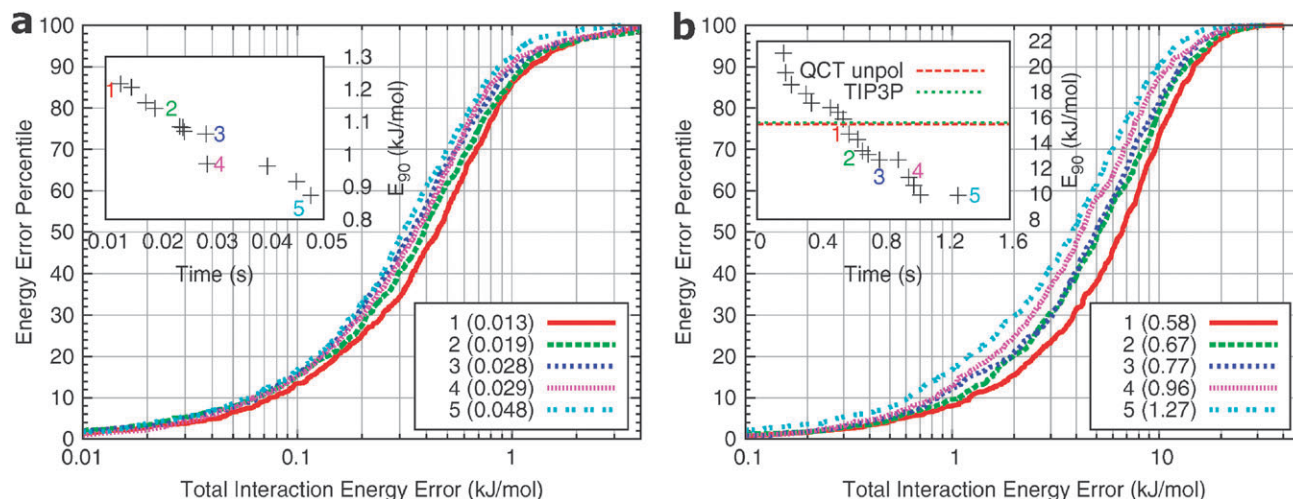

Fig. 8 The energy error percentiles for a selection of the Pareto-optimal combinations of models for the water (a) dimer and (b) the pentamer. The number in parentheses in the key denotes the time taken (s) for one energy prediction.

increases as the time of a single prediction increases. This is reflected by the S-shaped curves. Moving from left to right in the inset, the S-shaped curves move from right to left, *i.e.* the models get more accurate. This means that accuracy and speed of the model for a given cluster size can be tailored to the problem for which the model is required. Considering Pareto combination 4 in the inset of Fig. 8a, we can see for the dimer that a 58% increase in speed may be obtained relative to a pure Kriging approach (0.028 s *vs.* 0.068 s for a single energy prediction), without a decrease in performance in E_{90} (0.970 kJ mol^{-1} *vs.* 0.978 kJ mol^{-1}). Fig. 8b shows the same information as Fig. 8a but now for the pentamer, but with the addition of the unpolarised QCT and TIP3P energy errors in the inset graph. This time, a 26% increase in speed may be obtained relative to a pure Kriging approach (1.033 s *vs.* 71.389 s for a single energy prediction), without a decrease in performance in E_{90} (9.921 kJ mol^{-1} *vs.* 9.961 kJ mol^{-1}).

The Pareto models can also be assessed by the predicted molecular dipole moment that they give for the central water molecule of the pentamer. Table 2 also shows average and maximum absolute errors for the molecular dipole moment for each of the Pareto models. The models that show the lowest average absolute error are the models that have the charge of the atoms predicted by Kriging and the dipoles by RBFNN.

However, the models that show higher average errors but lower maximum errors use RBFNNs to predict the monopoles. Pareto models 1 and 5, though the same with regard to dipole moment prediction, differ in speed because model 1 uses less demanding models to predict the higher order multipole moments compared to model 5. The largest dipole moment maximum error is found when using a model, Pareto model 4, where the charge is found using RBFNN and the dipole moments are predicted using MLPs. Thus, our search attempts to identify the Pareto-optimal set of combinations of machine-learning methods with which to predict the multipole moments. The selection of which particular combination to use is then determined by individual preference of the level of trade-off between accuracy and speed.

Finally, we should mention that we have implemented the Kriging formula (eqn B13 in the ESI†) in the molecular dynamics program DLMULTI.[103] This development facilitates the availability of our potentials to other interested users.

5. Conclusions

The atomic polarisation of water inside water clusters is investigated according to a novel scheme. There are three key ingredients to this polarisation model. First, the

6374 | *Phys. Chem. Chem. Phys.*, 2009, **11**, 6365–6376

This journal is © the Owner Societies 2009

polarisation is expressed *via* atomic multipole moments, by focusing directly on how they change rather than by introducing polarisabilities. Secondly, the atoms are defined as finite portions of space, which change their shape and hence multipole moments in response to a changing environment. Thirdly, machine learning methods model this change. The inputs consist of coordinates of water molecules neighbouring the central water of interest, while the outputs are the atomic multipole moments. Here we compare the performance of three machine learning methods: multilayer perceptrons, radial basis function neural network and Kriging. Water clusters, sampled from multipolar MD simulations, are generated at B3LYP/aug-cc-pVTZ level, from the water dimer up to the hexamer. Kriging is the most accurate method to predict interatomic Coulomb interaction energies. It displays a superior performance for both small and large clusters and can therefore be used to model water in bulk as well as a heterogeneous context. Furthermore, it is observed that its prediction errors increase linearly with cluster size. However, this comes at an increase in computational cost compared to the two other methods. By modelling the trade-off between prediction accuracy and speed as the three methods are used in combination, a range of Pareto-optimal combinations can be identified. This enables force field designers to pick the optimal balance between these objectives, according to their own priorities.

Acknowledgements

References

1 J. L. Finney, *Philos. Trans. R. Soc. London, Ser. B*, 2004, **359**, 1145.
2 R. Ludwig, *Angew. Chem., Int. Ed.*, 2001, **40**, 1808.
3 G. W. Robinson, S.-B. Zhu, S. Singh and M. W. Evans, *Water in Biology, Chemistry and Physics*, World Scientific Publishing, Singapore, 1996.
4 F. Franks, *Water: A Matrix of Life*, Royal Society of Chemistry, Cambridge, GB, 2000.
5 F. H. Stillinger, *Science*, 1980, **209**, 451.
6 B. Guillot, *J. Mol. Liq.*, 2002, **101**, 219.
7 P. Paricaud, M. Predota, A. A. Chialvo and P. T. Cummings, *J. Chem. Phys.*, 2005, **122**, 244511.
8 B. Chen, J. Xing and J. I. Siepmann, *J. Phys. Chem. B*, 2000, **104**, 2391.
9 C. Vega, E. Sanz and J. L. F. Abascal, *J. Chem. Phys.*, 2005, **122**, 114507.
10 R. Bukowski, K. Szalewicz, G. C. Groenenboom and A. van der Avoird, *Science*, 2007, **315**, 1249.
11 M. W. Mahoney and W. L. Jorgensen, *J. Chem. Phys.*, 2000, **112**, 8910.
12 N. Gresh, S. A. Kafafi, J.-F. Truchon and D. R. Salahub, *J. Comput. Chem.*, 2004, **25**, 823.
13 J. Kaminsky and F. Jensen, *J. Chem. Theor. Comput.*, 2007, **3**, 1774.
14 T. D. Rasmussen, P. Ren, J. W. Ponder and F. Jensen, *Int. J. Quantum Chem.*, 2006, **107**, 1390.
15 C. Millot and A. J. Stone, *Mol. Phys.*, 1992, **77**, 439.
16 S. Liem and P. L. A. Popelier, *J. Chem. Phys.*, 2003, **119**, 4560.
17 P. Ren and J. W. Ponder, *J. Phys. Chem. B*, 2003, **107**, 5933.
18 K. Gurney, *An Introduction to Neural Networks*, Taylor and Francis, London, GB, 1997.
19 S. Haykin, *Neural Networks: A Comprehensive Foundation*, Prentice-Hall, New Jersey, USA, 2nd edn, 1999.
20 F. V. Prudente, P. H. Acioli and J. J. Soares Neto, *J. Chem. Phys.*, 1998, **109**, 8801.
21 H. Gassner, M. Probst, A. Lauenstein and K. Hermansson, *J. Phys. Chem. A*, 1998, **102**, 4596.
22 K. T. No, B. H. Chang, S. Y. Kim, M. S. Jhon and H. A. Scheraga, *Chem. Phys. Lett.*, 1997, **271**, 152.
23 K.-H. Cho, K. T. No and H. A. Scheraga, *J. Mol. Struct.*, 2002, **641**, 77.
24 S. Houlding, S. Y. Liem and P. L. A. Popelier, *Int. J. Quantum Chem.*, 2007, **107**, 2817.
25 M. G. Darley, C. M. Handley and P. L. A. Popelier, *J. Chem. Theory Comput.*, 2008, **4**, 1435.
26 C. M. Handley and P. L. A. Popelier, *J. Chem. Theory & Comput.*, 2009, DOI: 10.1021/ct800468h.
27 M. Devereux and P. L. A. Popelier, *J. Phys. Chem. A*, 2007, **111**, 1536.
28 M. A. Freitag, M. S. Gordon, J. H. Jensen and W. J. Stevens, *J. Chem. Phys.*, 2000, **112**, 7300.
29 A. J. Stone, *The Theory of Intermolecular Forces*, Clarendon, Oxford, UK, 1996.
30 G. C. Lie and E. Clementi, *Phys. Rev. A*, 1985, **33**, 2679.
31 C. Vega, J. L. F. Abascal, M. M. Conde and J. L. Aragones, *Faraday Discuss.*, 2008, **141**, 1.
32 J. S. Rowlinson, *Trans. Faraday Soc.*, 1951, **47**, 120.
33 J. A. Barker and R. O. Watts, *Chem. Phys. Lett.*, 1969, **3**, 144.
34 J. D. Bernal and R. H. Fowler, *J. Chem. Phys.*, 1933, **1**, 515.
35 A. Rahman and F. H. Stillinger, *J. Chem. Phys.*, 1971, **55**, 3336.
36 H. Berendsen, J. Postma, W. Van Gunsteren and J. Hermans, *Interaction Models for Water in Relation to Protein Hydration*, Reidel, the Netherlands, 1981.
37 H. J. C. Berendsen, J. R. Grigera and T. P. Straatsma, *J. Phys. Chem.*, 1987, **91**, 6269.
38 A. Glättli, W. F. van Gunsteren and X. Daura, *J. Chem. Phys.*, 2002, **116**, 9811.
39 W. L. Jorgensen, *J. Am. Chem. Soc.*, 1981, **103**, 335.
40 W. L. Jorgensen, J. Chandrasekhar, J. D. Madura, R. W. Impey and M. L. Klein, *J. Chem. Phys.*, 1983, **79**, 926.
41 C. L. Bishop, D. Pan, L. M. Liu, G. A. Tribello, A. Michaelides, E. G. Wang and B. Slater, *Faraday Discuss.*, 2008, **141**, 1.
42 S. A. Clough, Y. Beers, G. P. Klein and L. S. Rothman, *J. Chem. Phys.*, 1973, **59**, 2254.
43 C. A. Coulson and D. Eisenberg, *Proc. R. Soc. London, Ser. A*, 1966, **291**, 445.
44 P. L. Silvestrelli and M. Parrinello, *Phys. Rev. Lett.*, 1999, **82**, 3308.
45 J. K. Gregory, D. C. Clary, K. Liu, M. G. Brown and R. J. Saykally, *Science*, 1997, **275**, 814.
46 A. V. Gubskaya and P. G. Kusalik, *J. Chem. Phys.*, 2002, **117**, 5290.
47 C. M. Handley and P. L. A. Popelier, *Synth. and React. in Inorg., Metal-Organic, and Nano-Metal Chem.*, 2008, **38**, 91.
48 P. A. Kollman, *Acc. Chem. Res.*, 1996, **29**, 461.
49 C. Millot, J.-C. Soetens, M. T. C. Martins-Costa, M. P. Hodges and A. J. Stone, *J. Phys. Chem.*, 1998, **102**, 754.
50 M. S. Gordon, L. Slipchenko, H. Li and J. H. Jensen, *Annu. Rep. Comput. Chem.*, 2007, **3**, 177.
51 N. Gresh, *J. Comput. Chem.*, 1995, **16**, 856.
52 J.-P. Piquemal, B. Williams-Hubbard, N. Fey, R. Deeth, N. Gresh and C. Giessner-Prettre, *J. Comput. Chem.*, 2003, **24**, 1963.
53 F. Vigne and P. Claverie, *J. Chem. Phys.*, 1988, **88**, 4934.
54 J.-P. Piquemal, G. A. Cisneros, P. Reinhardt, N. Gresh and T. A. Darden, *J. Chem. Phys.*, 2006, **124**, 104101.
55 N. Gresh, G. A. Cisneros, T. A. Darden and J.-P. Piquemal, *J. Comput. Chem.*, 2007, **3**, 1960.
56 R. F. W. Bader, *Atoms in Molecules. A Quantum Theory*, Oxford University Press, Oxford, Great Britain, 1990.
57 P. L. A. Popelier, *Atoms in Molecules. An Introduction*, Pearson Education, London, UK, 2000.
58 C. F. Matta and R. J. Boyd, *The Quantum Theory of Atoms in Molecules*, Wiley-VCH, Weinheim, Germany, 2007.
59 P. L. A. Popelier and F. M. Aicken, *ChemPhysChem*, 2003, **4**, 824.

This journal is © the Owner Societies 2009

Phys. Chem. Chem. Phys., 2009, **11**, 6365–6376 | 6375

60 M. Devereux, P. L. A. Popelier and I. M. McLay, *J. Comput. Chem.*, 2009, **30**, 1300.
61 P. L. A. Popelier, L. Joubert and D. S. Kosov, *J. Phys. Chem. A*, 2001, **105**, 8254.
62 C. Haettig, *Chem. Phys. Lett.*, 1996, **260**, 341.
63 P. L. A. Popelier and D. S. Kosov, *J. Chem. Phys.*, 2001, **114**, 6539.
64 S. Liem, P. L. A. Popelier and M. Leslie, *Int. J. Quantum Chem.*, 2004, **99**, 685.
65 S. Y. Liem and P. L. A. Popelier, *J. Chem. Theory Comput.*, 2008, **3**, 353.
66 M. S. Shaik, PhD thesis, The University of Manchester, 2008.
67 R. A. Friesner, *Adv. Prot. Chem.*, 2006, **72**, 79.
68 M. Hodges, A. J. Stone and E. Cabaleiro Lago, *J. Phys. Chem.*, 1998, **102**, 2455.
69 H. Yu and W. F. van Gunsteren, *Comput. Phys. Commun.*, 2005, **172**, 69.
70 B. T. Thole, *Chem. Phys.*, 1981, **59**, 341.
71 I. Soteras, C. Curutchet, A. Bidon-Chanal, F. Dehez, J. G. Ángyán, M. Orozco, C. Chipot and F. J. Luque, *J. Chem. Theory Comput.*, 2007, **3**, 1901.
72 J. W. Caldwell and P. A. Kollman, *J. Phys. Chem.*, 1995, **99**, 6208.
73 J. Gao, D. Habibollazadeh and L. Shao, *J. Phys. Chem.*, 1995, **99**, 16460.
74 J.-P. Piquemal, N. Gresh and C. Giessner-Prettre, *J. Phys. Chem. A*, 2003, **107**, 10353.
75 J.-P. Piquemal, R. Chelli, P. Procacci and N. Gresh, *J. Phys. Chem.*, 2007, **111**, 8170.
76 M. Ledecq, F. Lebon, F. Durant, C. Giessner-Prettre, A. Marquez and N. Gresh, *J. Phys. Chem. B.*, 2003, **107**, 10640.
77 W. Chen and M. S. Gordon, *J. Chem. Phys.*, 1996, **105**, 11081.
78 E. Harder, A. V. M., I. V. Vorobyov, P. E. M. Lopes, S. Y. Noskov, A. D. MacKerell, Jr and B. Roux, *J. Chem. Theory Comput.*, 2006, **2**, 1587.
79 H. Yu and W. F. van Gunsteren, *J. Chem. Phys.*, 2004, **121**, 9549.
80 M. Yang, P. Senet and C. V. Alsenoy, *Int. J. Quantum Chem.*, 2005, **101**, 535.
81 H. Yu, D. P. Geerke, H. Liu and W. F. van Gunsteren, *J. Comput. Chem.*, 2006, **27**, 1494.
82 S. W. Rick, S. J. Stuart and B. J. Berne, *J. Chem. Phys.*, 1994, **101**, 6141.
83 H. A. Stern, F. Rittner, B. J. Berne and R. A. Friesner, *J. Chem. Phys.*, 2001, **115**, 2237.
84 N. Gresh, P. Claverie and A. Pullman, *Int. J. Quantum Chem.*, 1982, **22**, 199.
85 J. G. Angyan, G. Jansen, M. Loos, C. Haettig and B. A. Hess, *Chem. Phys. Lett.*, 1994, **219**, 267.
86 M. in het Panhuis, P. L. A. Popelier, R. W. Munn and J. G. Angyan, *J. Chem. Phys.*, 2001, **114**, 7951.
87 D. S. Broomhead and D. Lowe, *Complex Syst.*, 1988, **2**, 321.
88 R. Goodacre, E. Timmins, R. Burton, N. Kaderbhai, A. M. Woodward, D. B. Kell and P. J. Rooney, *Microbiology UK*, 1998, **144**, 1157.
89 N. Cressie, *Statistics for Spatial Data*, Wiley-Interscience, New York, USA, 1993.
90 D. G. Krige, *J. Chem., Metal. Mining Soc. South Africa*, 1951, **52**, 119.
91 C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, The MIT Press, Boston, MA, USA, 2006.
92 K.-T. Fang, H. Yin and Y.-Z. Liang, *J. Chem. Inf. Comput. Sci.*, 2004, **44**, 2106.
93 R. Kohavi and G. H. John, *Artificial Intelligence*, 1997, **97**, 273.
94 W. J. Welch, R. J. Buck, J. Sacks, H. P. Wynn, T. J. Mitchell and M. D. Morris, *Technometrics*, 1992, **34**, 15.
95 M. J. Frisch, G. W. Trucks, H. B. Schlegel, G. E. Scuseria, M. A. Robb, J. R. Cheeseman, J. Montgomery, J. A., T. Vreven, K. N. Kudin, J. C. Burant, J. M. Millam, S. S. Iyengar, J. Tomasi, V. Barone, B. Mennucci, M. Cossi, G. Scalmani, N. Rega, G. A. Petersson, H. Nakatsuji, M. Hada, M. Ehara, K. Toyota, R. Fukuda, J. Hasegawa, M. Ishida, T. Nakajima, Y. Honda, O. Kitao, H. Nakai, M. Klene, X. Li, J. E. Knox, H. P. Hratchian, J. B. Cross, V. Bakken, C. Adamo, J. Jaramillo, R. Gomperts, R. E. Stratmann, O. Yazyev, A. J. Austin, R. Cammi, C. Pomelli, J. W. Ochterski, P. Y. Ayala, K. Morokuma, G. A. Voth, P. Salvador, J. J. Dannenberg, V. G. Zakrzewski, S. Dapprich, A. D. Daniels, M. C. Strain, O. Farkas, D. K. Malick, A. D. Rabuck, K. Raghavachari, J. B. Foresman, J. V. Ortiz, Q. Cui, A. G. Baboul, S. Clifford, J. Cioslowski, B. B. Stefanov, G. Liu, A. Liashenko, P. Piskorz, I. Komaromi, R. L. Martin, D. J. Fox, T. Keith, M. A. Al-Laham, C. Y. Peng, A. Nanayakkara, M. Challacombe, P. M. W. Gill, B. Johnson, W. Chen, M. W. Wong, C. Gonzalez and J. A. Pople, *GAUSSIAN 03 (Revision C.02)*, Gaussian, Inc., Wallingford, CT, USA.
96 MORPHY98, a program written by P. L. A. Popelier with a contribution from R. G. A. Bone, UMIST, Manchester, England, EU (1998).
97 P. L. A. Popelier, *Mol. Phys.*, 1996, **87**, 1169.
98 P. L. A. Popelier, *Chem. Phys. Lett.*, 1994, **228**, 160.
99 G. Rennen, *Subset selection from large datasets for Kriging modeling*, Working Paper Series {2008–26}, Department of Econometrics and Operations Research, Tilburg University, The Netherlands, 2008.
100 I. M. Sobol, *USSR Computational Mathematics and Mathematical Physics*, 1967, **7**, 86.
101 K. Deb, *Multi-objective Optimization using Evolutionary Algorithms*, Wiley, 2001.
102 M. Rafat and P. L. A. Popelier, *J. Comput. Chem.*, 2007, **28**, 832.
103 M. Leslie, *Mol. Phys.*, 2008, **106**, 1567.