**FFLUX: Transferability of Polarizable Machine-learned Electrostatics in Peptide Chains**

Timothy L. Fletcher and Paul L.A. Popelier\*

Manchester Institute of Biotechnology (MIB), 131 Princess Street, Manchester M1 7DN, Great Britain

and

School of Chemistry, University of Manchester, Oxford Road, Manchester M13 9PL, Great Britain

**ABSTRACT:**

The fully polarizable, multipolar and atomistic force field protein FFLUX is being built from machine learning (i.e. kriging) models, each of which predicts an atomic property. Each atom of a given protein geometry needs to be assigned such a kriging model. Such a knowledgeable atom needs to be informed about a sufficiently large environment around it. The resulting complexity can be tackled by collecting the 20 natural amino acids into a few groups. Using substituted deca-alanines we present the proof-of-concept that a given atom’s charge can be modelled by a few kriging models only.

**Corresponding author: E-mail:** [**pla@manchester.ac.uk**](mailto:pla@manchester.ac.uk)

1. **Introduction**

Force fields are still the predominant way to power molecular dynamics simulations in the life sciences. While such force fields have already delivered challenging applications, the literature accumulates evidence that these force fields need to be improved in order for their predictions to be more reliable compared to experiment. Moreover, there is a need for the structural and dynamic outcome of various current force fields to be more consistent. A disconcerting recent example1 is that of the divergence of structural ensembles of intrinsically disordered proteins. The differences in chain dimensions, hydrogen bonding and secondary structure content were unexpectedly large between the eight all-atom empirical force fields investigated. An older but equally alarming example is that of Stock *et al.*2 who found that lifetimes of the conformational states of the paradigm oligopeptide trialanine differ by more than an order of magnitude, depending on which of the six different force fields (versions of AMBER, CHARMM, GROMOS and OPLS) was used.

Work towards the goal of increased consistency and reliability started many years ago culminating in next-generations force field such as XED3, AMOEBA4 and SIBFA5. These force fields all share multipolar electrostatics, which delivers much improved accuracy6 compared to the electrostatics represented by the traditional point charge approach of CHARMM and AMBER. Further research into the accuracy of force-field potentials, particularly electrostatic terms, has been urged7 already more than a decade ago. However, a force field’s architecture can be aligned even better with quantum mechanics, which ultimately governs matter at atomic scale. This is the strategy behind the development of a novel force field called FFLUX8 formerly called QCTFF9 because of its foundation upon Quantum Chemical Topology (QCT)10-12. This methodology partitions any system, whether a single molecule or an assembly of molecules (in the presence of ions, and with or without solvent molecules) into so-called topological atoms. This partitioning is parameter-free and the resulting atoms have finite boundaries. Topological atoms are space-filling: they do not overlap and leave no gaps between them. Their interior electron density is characterised by a set of atomic multipole moments. As one moves away from an atom, fewer multipole moments are necessary to describe their electron density, and hence electrostatic potential or electrostatic interaction energy with other atoms. In the limit, atomic monopole moments suffice at sufficiently long range. In a protein the vast majority of interatomic interactions are long range and hence calculated correctly with only “topological point charges” (i.e. monopole moments of topological atoms). However, at short range, the atomic multipole moments are available such that the correct electrostatics are recovered.

Unlike popular force fields, such as AMBER or CHARMM, FFLUX “sees the electrons”. In other words, FFLUX “is aware” of an atom’s electron density (through a set of multipole moments, up to the hexadecupole moment). Moreover, FFLUX is aware of an atom’s kinetic energy, the unique value of which is a paramount property that drove the original development13 of QCT. In principle, FFLUX is also aware of the second-order reduced density matrix, which contains all exchange and electron correlation effects. As a result, FFLUX incorporates all necessary information to predict the energetic behaviour of covalent bonds, as well as a range of weaker inter-atomic interactions. FFLUX also has the capacity14 to handle dispersion effects, without invoking a Lennard-Jones-like potential, for example, which amounts to an *ad hoc* add-on energy term. Finally, FFLUX stores the internal energy of an atom, which consists of its kinetic energy and the Coulomb, exchange and correlation energy between its electrons, as well as their interaction energy with the given atom’s nucleus. In summary, at the heart of FFLUX are topological atoms supplemented with their intra-atomic energy and their full inter-atomic energy (both with specific other atoms or the full environment of the atom of interest). The various energy components are provided by a QCT method called Interacting Quantum Atoms (IQA)15, which has been outlined just above.

Note that in the current work we will only use the Hartree-Fock level of theory, in view of the computational cost of generating many thousands of wave functions and IQA energy components. IQA has recently16 been extended to operate in conjunction with DFT/B3LYP, which allows FFLUX to incorporate of exchange-correlation. However, it is known that Hartree-Fock suffices to capture the essence of atomic transferability, which is the main point of the current work.

Equipped with atomic energies for atoms appearing in a sufficient number of different environments, FFLUX then invokes the machine learning method kriging17. The latter learns how an atom’s energies change with this atom’s changing environment. After sufficient training18, FFLUX creates “knowledgeable atoms”, which adjust their energies in response to the precise environment they are in. In particular, kriging creates a model that links a given atom’s property (output) to the coordinates (input) of the atoms surrounding this atom. A kriging model essentially interpolates an atomic property (e.g. charge, energy, …) between the geometries of its training set. A kriging model for an atom’s charge is thus able to return the value of this charge value when the atom resides in a previously unseen molecular geometry. Charge transfer is thus incorporated in FFLUX, but (dipolar) polarization as well because the same idea applies to the intra-atomic dipole moment. We note that kriging is able to interpolate successfully in the very high-dimensional spaces presented by the internal coordinates of a typical atomic environment and performs better than neural networks in our context19, 20.

With the architecture of FFLUX set up (and briefly explained here) the next challenge is to construct realistic kriging models and to carry this out efficiently. Recently we obtained21 accurate kriging models for all 20 natural amino acids, peptide-capped at both termini, and in their most prominent Ramachandran-map local energy minima. This achievement signifies a step change in realism in the representation of an amino acid in a protein force field. However, there is evidence that the energy of an atom in a protein is influenced by a larger environment than that of the single amino acid that it is part of. In fact22, for seven different homogeneous oligopeptides we found that the C*α*, H*α*, N, O and S atoms in a triamino-acid system are energetically comparable to those in the corresponding penta-amino-acid configurations, within 2.1 kJmol-1 (in absolute value). On average, the tri-peptide sample systems represent a ∼8 Å atomic horizon around the central atoms of interest.

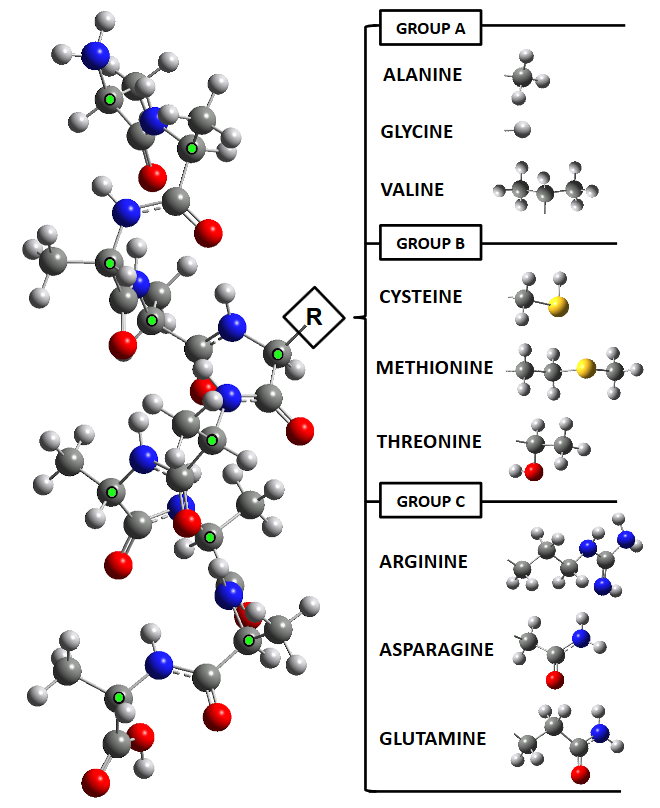
We present this work as the third part of a series of papers that have explored transferability within FFLUX through transferable kriging models. In the first paper23 we outlined the need for triamino acids compared to merely single amino acids as a data source representative of long-chain peptides. The first transferable models studied there proved that fragment models can be applied successfully to larger systems such as deca-alanine. In the second paper24 we demonstrated the advantage and feasibility of grouping amino acids according to the influence a neighbouring residue has on the charge and kinetic energy of alpha carbons (Cα). This grouping enabled the creation of kriging models that will then be more specific to an amino acid’s local environment within a peptide chain without needing 8000 (=20x20x20) potential models required for all triamino acid combinations. In the current and third paper we present a proof-of-concept of this assertion, utilizing transferable kriging models to predict the atomic charge of alpha carbons appearing in any oligopeptide. We work again with deca-alanines because they were used in the previous two publications and are well understood in the context of transferable kriging models. Here, we utilize deca-alanines containing amino acid substitution using any of 9 different amino acids, as detailed in the next Section.

While popular point-charge force fields tend to have a large list of atom types, each with a parameter set corresponding to a potential, FFLUX introduces a large list of kriging models. A given oligopeptide or protein backbone needs to be “dressed up” with the “quantum-knowledgeable” atoms that FFLUX provides. This means that a kriging model must cover a sufficiently large environment of the atom that it provides properties for. This paper proposes methods for the creation and selection of these kriging models.

**2. Methods and Materials**

In the first and second paper of this series, we showed that kriging models built with decapeptide data outperform those built with single amino acid or tripeptide data when predicting decapeptide properties. Thus, for the eventual goal of dressing up oligopeptides, here we have chosen to build kriging models using decapeptide data only.

A single deca-alanine (310) helix geometry was extracted from the Brookhaven Protein Data Bank25 (IL36.pdb). The helix termini were capped with methyl groups and the structure optimized using GAUSSIAN0926 at HF/6-31+G(d,p) level27. This deca-alanine geometry is then the basis for all other geometries in this work. Eight more decapeptides were created from a single substitution at the 6th residue (counting from the C terminus), using sidechain geometries from optimized amino acids. Figure 1 illustrates the resulting set of 9 helical decapeptides.



**Figure 1.** The nine single-substituted decapeptide helices studied in this work, each consisting of a deca-alanine with up to a single substitution at the 6th residue (counting from the C terminus). Thus, each structure differs only by its R group or the substituted residue’s sidechain. Carbon, hydrogen, nitrogen and oxygen atoms are respectively dark grey, grey, blue and red. Green dots signify an alpha carbon (Cα).

Each of these 9 decapeptides is referred to by their substituted residue, denoted *R* in Figure 1. For example, the decapeptide (ALA5-GLY-ALA4) is referred to as the ‘Glycine’ or ‘GLY’ decapeptide. Each decapeptide belongs to one of 3 *groups* (of a maximum possible number of 5 groups, denoted *A* to *E*, see below): Ala, Gly and Val belong to group *A*; Cys, Met and Thr belong to group *B*; and Arg, Asp and Glu belong to group *C*. Each amino acid pairs its substituted residue with others of a similar group as discussed in our second paper24 in this series. We take alanine (A) as a reference amino acid in the definition of such a group. In particular, our second paper24 proposed as group decision criteria ∆Q00 and ∆T, which refer respectively to the charge and kinetic energy of Cα. More precisely, ∆ refers to the difference between the central Cα in a doubly-substituted tripeptide (XAX where *X* is a substituted residue) and the central Cα in an unsubstituted ‘AAA’ tripeptide. In particular:

* ∆Q00 < 0.005 au and ∆T< 10 kJmol-1 then group A.
* ∆Q00 < 0.005 au and ∆T> 10 kJmol-1 then group B.
* ∆Q00 > 0.005 au and ∆T< 10 kJmol-1  then group C.
* ∆Q00 > 0.005 au and ∆T> 10 kJmol-1 then group D.
* If Proline then group E.

Due to the large volume of *ab initio* calculations already involved in this work (over 20,000 =10x2000 *ab initio* decapeptide calculations, see below) no representative of group *D* was investigated. We note that this group contains the large amino acids tryptophan, tyrosine and histidine. In future work, this approach can be readily extended to the full set of 20 natural amino acids in order to predict the atomic properties of any peptide or even protein. The desired proof-of-concept aimed for in this paper is amply demonstrated with the current data.

The groups above are proposed with the intention of capturing an atom’s (in this case, an alanine Cα) local environment beyond its immediate bonded atomic neighbours. It is common to create atom types28 by treating atoms of similar local environments as the same type. Here, by some analogy, we extend this concept toward neighbouring residues in a peptide chain. By doing so, we are able to create Cα kriging models specific to certain types of environment while avoiding the need for a kriging model for each alanine Cα. This principle is analogous to that applied to an atom type, which avoids the unique parameterisation of each Cα.

Each of the 9 decapeptides are optimized using GAUSSIAN09 at HF/6-31+G(d,p) level. The optimized geometries are then distorted through their normal modes of vibration using the in-house program TYCHE29, 30. The program TYCHE uses an optimized geometry as a seed for generating 2000 unique geometries where no angle or bond length in the created geometries differs from the seed’s by more than 15 %. For example, a bond length of 1 Å in a given optimized geometry cannot be distorted beyond 0.85 Å or 1.15 Å. Each of the 9 decapeptides is treated independently, giving 2000 unique geometries each. The 2000 geometries have their wavefunctions calculated also at HF/6-31+G(d,p) level. The program AIMAll31 was used to calculate the atomic charges from the system’s electron density obtained from the wavefunctions.

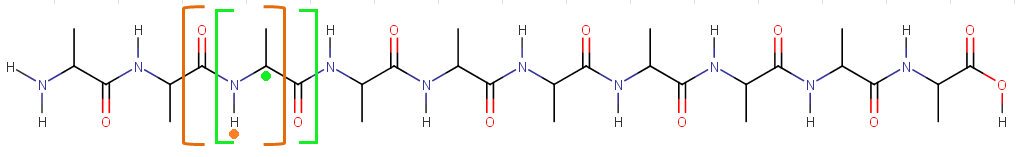
We convert the Cartesian coordinates into atom-centred polar spherical coordinates termed *features*32, 33, which give a full description of the system. Thus, each atom has its own set of features where a given geometry is described ‘as seen’ by that atom. When the features of a geometry are supplemented with the charge of a target atom, a single training example is created for this atom. With a single example per geometry, a *data set* of 2000 examples (i.e. geometries) is constructed. This data set is split into a *training set*, which can be used by a machine learning method to build a predictive model from, and a *test set* that contains examples to be predicted by the model. It is important to remember that we never test a model for its predictions for geometries that have been included in the training data. In other words, the test set is always a true *external* test set. The data set is filtered for ‘undesirable’ entries where geometries with large integration errors34, 35 (L()) are filtered alongside those whose molecular net charge deviates significantly (> 0.001 au) from the desired value, which is zero in the case of the decapeptides in this work.

This training set can be utilized by kriging to find relationships between an atom’s charge and its corresponding molecular geometry. The machine learning method Kriging36 has been explained in our earlier publications (following Jones *et al.*36-38), which is why we review it here only very briefly. From the training data39, kriging creates models that can predict atomic charges using molecular geometries only. Of course, the output can also be an atomic multipole moment. Equation (1) shows how kriging creates a relationship between a molecule’s features (geometry) and output (atomic charge), , which can be expressed as the error it carries (the summed term) plus the global term (),

 (1)

where is the *i*th element of , where is a matrix of error correlations between training points, and **1** is a column vector of ones. Thus, we take into account the prediction example’s correlation with all of our training examples and assign importance to these correlations accordingly with . Indeed, if the prediction example is very close to a pre-existing example in the training set, they are highly correlated and we can expect that they share a similar output value. In fact, the kriging predictor passes exactly through training points and a ‘perfect’ prediction is achieved when attempting to predict the outputs for a known geometry. The prediction of any given point is entirely dependent on the training points that directly surround it in the feature space rather than those at increasing distances from the prediction point. If we cannot find a well-correlated example in the training set, the output will tend toward the global term, . This is a very useful consequence of the kriging method when applied to chemical systems, giving mean charge values, which are still valuable, in the absence of any better predictions. Some studies present kriging as a method for making good predictions on sparse data sets40, and so one could also claim that these uncorrelated predictions are still valuable.

A kriging model can be made to describe any atomic charge using the entire system’s geometry as input. However, a kriging model can also be constructed from a fragment of this molecular system, as described in Figure 2.



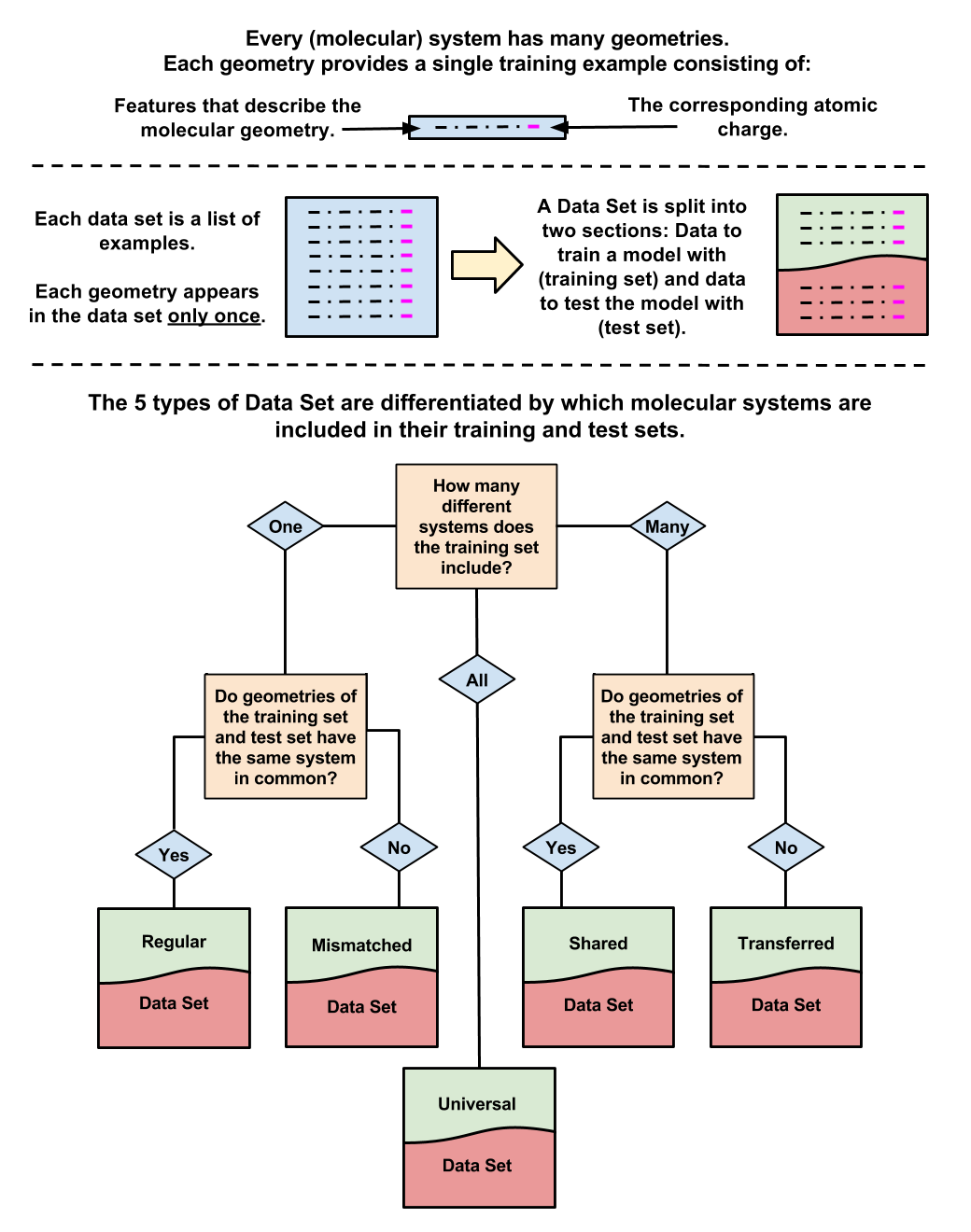
**Figure 2**. A schematic of deca-alanine with a proposed molecular fragment outlined in green. The Cα fragment (inside green brackets) fragment is a collection the atoms Namide, Hamide, Cα, Cbeta, Camide, and Oamide, which captures Cα’s (denoted by a green dot) immediate environment. Thus, data describing this fragment can be isolated and used to create a kriging model for Cα. The atom Hamide (denoted by an orange dot) uses a different fragment (inside orange brackets) consisting of Oamide, Camide, Namide, Hamide, Cα and Cbeta. It is desirable for a fragment to include atoms from all directions with respect to the atom being modelled.

A kriging model made from a fragment of the molecular system is likely to be more generally applicable than a kriging model that has full knowledge of the molecular system. Since the fragment kriging model only requires knowledge of the fragment geometry, it can be applied to any other molecular system where this fragment is found as described in past work24, 41. In order to create a data set for a fragment, features describing the positions of atoms outside of a fragment should be discarded, keeping only a description of the molecular fragment. An identical fragment in another molecular system will then have the same corresponding set of features, different only in value. Indeed, it is possible to have an identical fragment from two entirely different systems in the same data set. When the fragment’s system of origin is not represented in any example in the test set, we are using a kriging model to predict a system that it has no direct knowledge of. This application of a kriging model to a foreign system is a basic example of transferability. Similar fragments from different systems can be combined into a single training set, resulting in a more general kriging model that can be transferred to other systems.

In this work, we describe 5 different types of data set that are made from the training sets previously described and that are defined by the systems making up a data set. The various types can be defined as follows:

* If trained by a single molecule and predicting for the *same* single molecule *with a full set of features* the data set is called *Regular.*
* If trained by a single system but predicting for another system the data set is called *Mismatched.*
* If trained by multiple systems and predicting for one of the systems in the training set the data set is called *Shared.*
* A data set using multiple systems for its training data and different systems in the test data is termed a *Transferred* data set. This is the basis of transferable models in FFLUX.
* A *Universal* data set includes every applicable system in the training and test data. This type is an expansion of the Shared data set.

These five data sets are illustrated in Figure 3 with details on how to easily identify each type.

****

**Figure 3.** Anatomy of a data set and how to differentiate between each of the 5 data set types used in this study. Regular and Mismatched data sets use kriging models with a single system’s training data. Shared and Transferred data sets use kriging models comprised of multiple systems’ training data. Universal data sets can be thought of as a special type of Shared data set that contains data for *all* available molecular systems.

It should be noted that a Regular data set has no removed features and thus contains a complete description of the system. A Regular data set is a ‘gold standard’ in terms of kriging prediction accuracy but it is computationally expensive because it does not benefit from feature reduction (compared to the other types of data set, which do). Of course, a Regular data set does not invoke transferability. On the other hand, Mismatched, Shared, Transferred and Universal data sets all have a reduced number of features that describe a transferable fragment.

When the kriging model predicts atomic charges, we can calculate the error between the predicted moments and their original (*ab initio*) counterparts. The error of a single atomic charge is given in equation 2,

 (2)

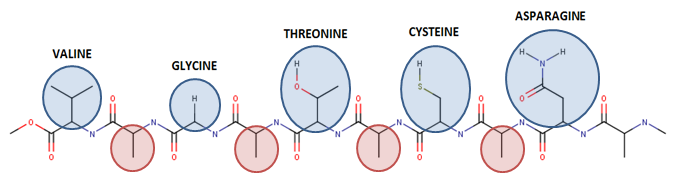
where Q00 is the notation for a charge, used in the context of multipole moments Qlm where *l* is the rank and *m* the component. The focus of this work revolves around the charge42 (rather than higher multipole moments) for two reasons. First, it can be shown33 mathematically that any effort to improve the accuracy of multipolar electrostatics (which is exact in the limit, provided the multipole expansion converges) must focus on the lower rank multipole moments; the monopole (i.e. charge) has the lowest rank. Secondly, the charge-charge contribution to the electrostatic energy is the longest “surviving” one at very long range, and the vast majority of atom-atom interactions in a protein are of that range. The error involved with the reproduction of atomic charges is used to assess the quality of a model and thus the success of implemented transferability.

Equation 3 translates an error in a charge of atom *A* into an electrostatic energy, as this atom interacts with atom *B*, at a distance *RAB* , that is

 (3)

For convenience, this equation takes the atomic charge in milli-electron (me), which is one thousandth of an atomic unit of charge. Equation (3) takes the distance in Å, and returns the energy in kJmol-1. For example, if the charge error is 10 me then the energy of its interaction with a test charge of 1000 me that is 10 Å away will be 1.4 kJmol-1.

Finally, we introduce a *Custom (*deca)peptide: **VAL**-ALA-**GLY**-ALA-**THR**-ALA-**CYS**-ALA-**ASN**-ALA, where the bold emphasises the substituted alanines. This decapeptide is created in identical fashion to the first set of 9 peptides in that it is a substituted alanine helix. In the *Custom peptide*, five different but now simultaneous substitutions are made at the 1, 3, 5, 7 and 9 positions along the alanine helix to create a new decapeptide, a schematic for which is shown in Figure 4.



**Figure 4.** Schematic view of the *Custom peptide*, which is a highly substituted deca-alanine helix. Substitutions for natural amino acids occur at residues 1, 3, 5, 7 and 9 (V, G, T, C, R, respectively, in blue circles). This results in 4 alanine residues (positions 2, 4, 6, 8, in red circles) that are flanked by two substituted residues and are used for modelling in this work. The final (10th) alanine residue (utmost right) is not modelled in this work (and hence not marked in red) due to its proximity to the terminal atoms.

The Custom peptide is the final test for transferable models, simulating 4 different scenarios of an alanine residue in a random peptide sequence. In a manner identical to previously-described data sets, 2000 geometries of this helix are generated and have their wavefunctions and atomic charges calculated before training sets were made. Note that 20,000 wavefunctions are generated because there are 2,000 geometries for each of the 10 possible systems, consisting of the Custom peptide and the 9 single-substituted deca-alanines.

It is our ultimate goal to create simple, transferable models using data from the 9 single-substituted decapeptides and use those models to predict atoms in the Custom decapeptide. In this work, alanine Cα and Hamide atoms are investigated as examples of atoms in the same system that can be handled using different data sets.

1. **Results**

**3.1 Alpha Carbons (Cα)**

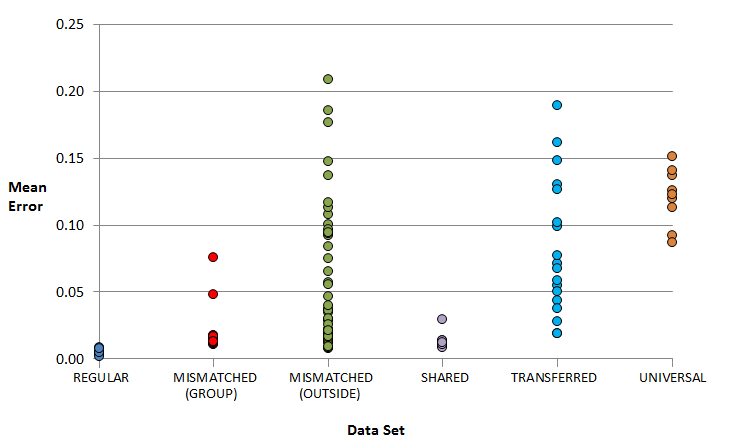
Nine kriging models for the Q00 of an alanine Cα atom are created, one for each of the decapeptides in Figure 1. The alanine Cα in question is always on the 5th residue (neighbouring the substituted 6th residue) from the C-Terminus of the alanine helix. Table 1 shows the accuracy that each model (first column) predicts for the same alanine Cα Q00 on each of the 9 decapeptides (first row), resulting in 81 combinations in total.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  | **TEST** |  |  |  |  |
|  |  | **ALA** | **GLY** | **VAL** | **CYS** | **MET** | **THR** | **ARG** | **ASN** | **GLN** |
| **TRAINING** | **ALA** | ***0.005*** | *0.015* | *0.017* | 0.018 | 0.019 | 0.017 | 0.097 | 0.040 | 0.095 |
| **GLY** | *0.013* | ***0.004*** | *0.011* | 0.065 | 0.022 | 0.016 | 0.016 | 0.113 | 0.209 |
| **VAL** | *0.017* | *0.013* | ***0.003*** | 0.013 | 0.037 | 0.017 | 0.008 | 0.147 | 0.177 |
| **CYS** | 0.017 | 0.100 | 0.007 | ***0.005*** | *0.013* | *0.016* | 0.047 | 0.030 | 0.055 |
| **MET** | 0.014 | 0.019 | 0.185 | *0.012* | ***0.002*** | *0.013* | 0.036 | 0.025 | 0.084 |
| **THR** | 0.018 | 0.018 | 0.014 | *0.012* | *0.014* | ***0.008*** | 0.009 | 0.021 | 0.117 |
| **ARG** | 0.035 | 0.013 | 0.108 | 0.014 | 0.018 | 0.057 | ***0.005*** | *0.048* | *0.013* |
| **ASN** | 0.012 | 0.029 | 0.013 | 0.021 | 0.029 | 0.037 | *0.011* | ***0.008*** | *0.076* |
| **GLN** | 0.022 | 0.137 | 0.092 | 0.008 | 0.075 | 0.094 | *0.012* | *0.013* | ***0.007*** |

**Table 1.** Mean errors (au) of 9 models predict the charge of an alanine Calpha on 9 different decapeptides, resulting in 81 predictions. Each model was created using ‘TRAINING’ data for a particular decapeptide, listed down the first column. Each model predicts ‘TEST’ data for each decapeptide, listed across the first row. The rows and columns are sectioned into 3 groups that correspond to the amino acid groups shown in Figure 1 (A (ALA, GLY, VAL); B (CYS, MET, THR) and C (ARG, ASN, GLN) ).

Values along the diagonal (top-left to bottom-right, in bold italics) of Table 1 indicate Regular predictions, where a kriging model is trained with a given decapeptide and predicting for a single atom in the same decapeptide. The Regular predictions display the lowest errors due to high specificity and the presence of a full set of features in the training set. Outside of this diagonal, prediction errors are larger because a kriging model trained on one decapeptide attempts to predict an atomic charge for a different decapeptide.

Table 1 also includes 9 sections that outline each of the three groups (A, B, C) in Figure 1. Again, the top-left to bottom-right ‘block diagonal’ of these groups (in italics) show where a kriging model is attempting to predict properties for a decapeptide within its own group (henceforth called Mismatched (GROUP)). Since decapeptides within the same group are chemically similar, a kriging model can be transferred to these molecules with some success, giving small prediction errors relative to predicting decapeptides outside of their group (called Mismatched (OUTSIDE)). Kriging models for each group can also be made by sharing the training data of decapeptides in each of the respective groups (A, B, C). These group kriging models can then predict each decapeptide within their own group (Shared data set) and outside of their group (Transferred data set). Additionally, a single kriging model can be made from all of the single-substituted decapeptides (Universal data set) and can make predictions for any decapeptide. The resulting predictions of the above data sets are shown in Figure 5.

****

**Figure 5.** Mean prediction errors (au) of the alanine residue’s Calpha Q00 from Table 1, sorted according to the type of data set the predictions belong to. The Mismatched data set is further broken down by differentiating between Mismatched data sets that predict within the *same* amino acid group (‘GROUP’) as the training data and those that predict *outside* of that group (‘OUTSIDE’).

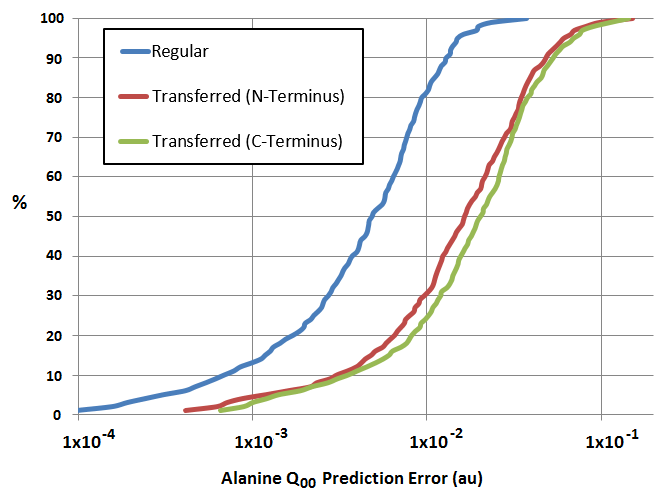
As anticipated, Regular data sets give the lowest prediction errors, with Shared data sets returning approximately double the error but still remaining consistently more accurate than other data sets. Mismatched data sets, which predict within their own group, are similar in predictive accuracy to Shared sets (which also predict within their own group) but with a larger error. This larger error is significant in group *C*, perhaps because ARG, ASN and GLN are not as similar as amino acids of other groups are (for example, GLY, ALA, VAL in group A).

In the case of Figure 5, the Transferred data sets are identical to the Shared data sets in terms of the training data and differ only in their test data. Thus, when the kriging model from a Shared set is used to predict for a molecule that it has no training data for, we then have a Transferred set. The mean prediction error for a Transferred set turns out to be 7 times higher than that of the Shared data set and ~15 times higher than that of the Regular set. Transferred sets in this context offer little value over Mismatched data sets that predict outside of their amino acid group because their kriging models have no training data that is similar enough to the predicted system. We can conclude that a training set needs some common ground with the molecule its model is predicting, because errors are substantially lower in all sets that include some amount of data on the molecule being predicted, or in sets that predict within the same amino acid group. A ‘Universal’ data set can be constructed where all 9 single-substituted deca-alanines are present in the training data and should hypothetically give acceptable predictions for every molecule. While the Universal data set does outperform data sets that predict outside of their own group, the prediction errors are significantly higher than for the Shared and even Mismatched (Group) sets. A Universal data set returns lower errors for the worst predictions, because at least some training data are present for each tested system, meaning nothing is truly ‘foreign’ to the kriging model. However, a Universal data set also means higher errors for the best predictions because the kriging model’s specificity is diluted by training data for other decapeptides. We conclude that a Universal data set is too general to be usefully accurate in predicting Cα charges.

It is important to note that a Transferred Set is a very general label, one that has shown excellent predictions in past work but large errors in Figure 5. It might be erroneously surmised that a kriging model cannot predict an atomic charge within a molecule outside of its training data. However, the Transferred data set’s poor predictions occur primarily because we have chosen to predict a decapeptide that is not similar enough to one in the training set. We set to prove a kriging model’s transferability by taking models trained on the 9 decapeptides in Figure 1 and using them to predict charges on the Custom decapeptide in Figure 4. Each alanine residue in the Custom decapeptide is neighboured by two different substituted residues, but our kriging models contain training data where the alanine is only neighboured by a single residue at a time. An ideal scenario would be having kriging models for every combination of alanine neighboured by residues of different amino acid groups, with combinations such as *A-ALA-A, A-ALA-B, A-ALA-C, B-ALA-A* and so on. This would mean creating 9 kriging models if we were to account for all combinations, plus calculating the corresponding *ab initio* data with which to train them, a situation best avoided if possible. Instead, we use the kriging models that correspond to the amino acid groups of the C-Terminus neighbour and that of the N-Terminus neighbour of an alanine residue on the Custom decapeptide. This chosen residue dictates which 3 of the 9 single-substituted deca-alanine systems a kriging model is trained on, so that the predicted residue has a neighbour of the same amino acid group as the chosen single-substituted deca-alanines.

For example, the first alanine residue (counting from the C-Terminus, see Figure 4) has neighbouring residues VAL and GLY, that is, VAL on the C-Terminus side and GLY on the N-Terminus side. In this case, both VAL and GLY belong to group *A*, and so the same kriging model is used in both situations. That kriging model is trained on single-substituted deca-alanines from group *A*. However, the second alanine residue has neighbours GLY and THR, in which case one must choose which kriging model (group *A* or *B*) to use. The consequence of this choice can be assessed by an error analysis, through the so-called S-curve. This curve shows the cumulative error distribution (y-axis in percentage) for the error (x-axis in au of charge) incurred in a test set of geometries.

Figure 6 shows the S-curve summarizing the prediction errors of the charges of the four alanine residuesin the Custom decapeptide across test 100 geometries (400 total predictions per S-curve). As anticipated, Regular data sets (blue curve, ~0.006 au mean error) give significantly lower errors than the Transferred data sets (green and red curves, ~0.025 au mean error). It is no surprise that the 4 kriging models specific to the Custom decapeptide, which the Regular data set uses, are preferable to the 3 general kriging models built on single-substituted decapeptide data. Perhaps surprising is that the Transferred data sets have only around 4 times the error of the Regular data sets for the Custom decapeptide, comparing the curves in Figure 6. However, when comparing the predictions for single-substituted decapeptides in Figure 5, we noted an almost 15 times increase of error for Transferred data sets over the Regular data sets. The difference is that in the case of the single-substituted decapeptide prediction errors in Figure 5, the Transferred data sets only contained test data of decapeptides that were outside the amino acid group of those in the training data. For the Custom decapeptide prediction errors in Figure 6, kriging models were carefully selected for each of the alanine residues. Thus, Transferred data sets were guided toward success by their training and test data sharing the same amino acid type. It should be noted that choosing the amino acid type of the N-terminus residue generally gives the best predictions (Supporting Information, Figure S1), probably due to the kriging models used being trained on single-substituted decapeptides where the substituted residue lies on the N-terminus side of the modelled alanine residue.

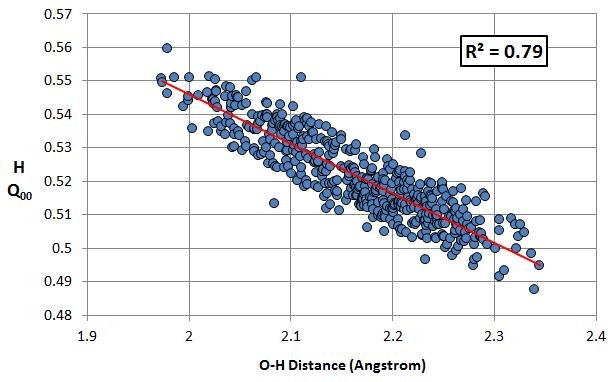
****

**Figure 6.** Summary ofprediction errors for the Cα Q00 on the four different alanine residues in the Custom decapeptide (VAL-ALA-GLY-ALA-THR-ALA-CYS-ALA-ASP-ALA). A set of Regular data set predictions is displayed alongside two different Transferred data set predictions, called C-Terminus and N-Terminus, which indicate which side of the alanine residue is considered as the ‘neighbouring residue’ for kriging model selection purposes.

**3.2 Hamide and hydrogen bonds**

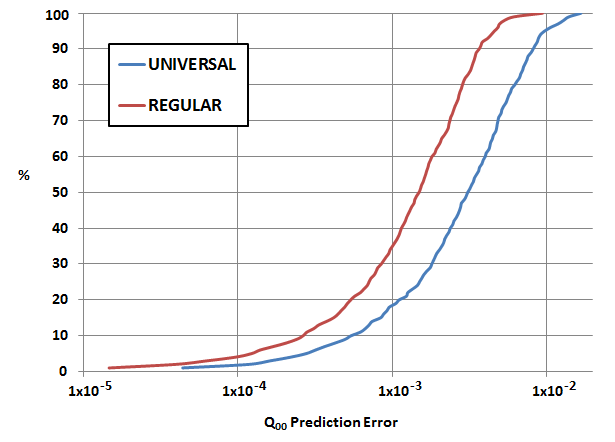
The Transferred data sets give an acceptable level of accuracy and should lead to the ability to model charges in biological systems in the near future. However, in order to progress toward these biological systems, it is important to apply transferable models to hydrogens in hydrogen bonds, which we have neglected until now.

The Hamide (amide hydrogen) atoms in our helical decapeptides form hydrogen bonds with Oamide atoms of other residues. By including the O···H bond length as a feature in the training data, an Hamide charge can be accurately modelled. For the O···H bond length to be useful as a feature for the kriging machine learning, there must be some correlation between it and Q00 of the Hamide. Figure 7 plots the Q00 for each of the 4 alanine residue Hamide atoms in the Custom decapeptide against the corresponding O···H bond lengths.

****

**Figure 7.** Hamide Q00 values (au) versus the corresponding O···H hydrogen bond lengths. Each of the 4 alanine residues in the Custom decapeptide contributes 100 hydrogen bond examples to the plot. The total set of examples has a line of best fit with an r2 of 0.79.

There is a moderate correlation between the hydrogen bond length and the Hamide atomic charge. It is not expected that the correlation between hydrogen bond length and Hamide Q00 is perfect as this would ignore the significant contributions to the Hamide Q00 coming from the N-H bond length and fluctuations from polarization caused by all other atoms in the system. Since the range of Hamide Q00 values (~0.07 au) is significantly smaller than that of Cα charges (~0.36 au), we contest that Hamide atoms are more similar to one another than Calpha atoms are, and can more easily share a single kriging model. Thus, we choose to use a Universal model to describe Hamide, using its hydrogen bonded distance as a feature of the model. The resulting prediction results are given in Figure 8.

****

**Figure 8.** Hamide Q00 prediction error on the four alanine residues across the Custom decapeptide, shown as a percentile of the range of results.

As with Cα Universal models, the Hamide Universal data set gives prediction errors significantly larger than the corresponding Regular data set. However, unlike the Calpha data sets, the Hamide Universal data set still gives usefully accurate predictions with a mean error of ~0.004 au compared to the ~0.002 au error of the Hamide Regular data set. It appears as though a Universal data set for Hamide is more akin to a Cα’s Shared data set than Cα’s Universal Data set. In other words, regardless of the neighbouring residue, all alanine Hamide atoms appear to be of a similar ‘group’, sharing similar atomic charges.

1. **Conclusion**

We present a proof-of-concept for the transferability of kriging models and their application to arbitrary peptide chains. In this work, we have investigated Cα and Hamide in decapeptides and shown how the force field FFLUX can provide a solution to modelling their atomic charges.

Amino acids can be sorted into ‘groups’ according to their influence on a neighbouring alanine residue, leading to kriging models that can predict for an entire group. The result is a small set of accurate kriging models that use the alanine Cα’s local environment to model its charge but also take into account more distant factors such as a neighbouring residue. When tested on an arbitrary decapeptide, Transferable models (~0.025 au) give approximately 4 times the error of Regular kriging models trained specifically for the same system (0.006). However, the benefit of a Transferrable model is that it can be used for potentially any possible peptide chain. If we consider the mean error against the range of charges the alanine Cα atoms in our custom decapeptide geometries span (~0.6 au), then the Transferable models predict those charges with about 96% mean accuracy.

It was found that for Hamide atoms, the range of charges is significantly smaller than those found in Calpha atoms (~0.08 and ~0.6 au, respectively). Since Hamide atoms tend to be relatively similar to one another, they are excellent candidates for a ‘Universal’ kriging model that can be used on any Hamide atom in a peptide regardless of differing neighbouring residues. Adding hydrogen bond lengths as an additional descriptor for these Hamide atoms, a Universal kriging model’s prediction errors (~0.004 au) are only two times larger than those of kriging models trained specifically for the same system (~0.002 au). Considering the mean error against the range of Hamide charges, Universal models for Hamide atoms predict with about 95% mean accuracy.

We conclude that kriging machine learning can provide transferable models that are usefully accurate in arbitrary peptide systems.

**Acknowledgements**

P.L.A.P. acknowledges the EPSRC for funding through the award of an Established Career Fellowship (grant EP/K005472), which funds both authors.

**Supporting Information**

**Figure S1:** S-Curves for the Cα Q00 prediction errors of the four alanine residues (Ala1, Ala2, Ala3, Ala4 as counted from the C-Terminus) in the Custom decapeptide (VAL-ALA-GLY-ALA-THR-ALA-CYS-ALA-ASN-ALA).

**References**

[1] S. Rauscher, V. Gapsys, M. J. Gajda, M. Zweckstetter, B. L. de Groot and H. Grubmüller, *J.Chem.Theor.Comput.* **2015**, *11*, 5513.

[2] Y. Mu, D. S. Kosov and G. Stock, *J.Phys.Chem.B* **2003**, *107*, 5064.

[3] J. G. Vinter, *J.Comput.Aided Mol. Des.* **1994**, *8*, 653.

[4] P. Y. Ren, C. J. Wu and J. W. Ponder, *J.Chem.Theory Comput.* **2011**, *7*, 3143.

[5] N. Gresh, G. A. Cisneros, T. A. Darden and J.-P. Piquemal, *J.Chem.Theory Comput.* **2007**, *3*, 1960.

[6] S. Cardamone, T. J. Hughes and P. L. A. Popelier, *Phys.Chem.Chem.Phys.* **2014**, *16*, 10367.

[7] J. W. Ponder and D. A. Case, *Adv. Protein Chem.* **2003**, *66*, 27.

[8] P. L. A. Popelier, *Phys.Scr.* **2016**, *91*, 033007.

[9] P. L. A. Popelier, *Int.J.Quant.Chem.* **2015**, *115*, 1005.

[10] P. L. A. Popelier, in *Challenges and Advances in Computational Chemistry and Physics dedicated to "Applications of Topological Methods in Molecular Chemistry"*, eds. R. Chauvin, C. Lepetit, E. Alikhani and B. Silvi, Springer, Switzerland, ch. 2, pp. 23, **2016.**

[11] P. L. A. Popelier, in *The Nature of the Chemical Bond Revisited*, eds. G. Frenking and S. Shaik, Wiley-VCH, Chapter 8, ch. 8, pp. 271, **2015**.

[12] R. F. W. Bader, *Atoms in Molecules. A Quantum Theory.*, Oxford Univ. Press, Oxford, Great Britain, **1990.**

[13] R. F. W. Bader and P. M. Beddall, *J.Chem.Phys.* **1972**, *56*, 3320.

[14] J. L. McDonagh, M. A. Vincent and P. L. A. Popelier, *Chem.Phys.Lett.* **2016**, *662*, 228.

[15] M. A. Blanco, A. Martín Pendás and E. Francisco, *J.Chem.Theory Comput.* **2005**, *1*, 1096.

[16] P. Maxwell, A. Martín Pendás and P. L. A. Popelier, *PhysChemChemPhys* **2016**, *18*, 20986.

[17] N. Cressie, *Statistics for Spatial Data*, Wiley, New York, USA, 1993.

[18] N. Di Pasquale, M. Bane, S. J. Davie and P. L. A. Popelier, *J.Comput.Chem.* **2016**, *37*, 2606.

[19] C. M. Handley and P. L. A. Popelier, *J.Chem.Theory Comput.* **2009**, *5*, 1474.

[20] M. G. Darley, C. M. Handley and P. L. A. Popelier, *J.Chem.Theory Comput.* **2008**, *4*, 1435.

[21] T. L. Fletcher and P. L. A. Popelier, *J.Chem.Theor.Comput.* **2016**, *12*, 2742.

[22] P. Maxwell and P. L. A. Popelier, *Molec.Phys.* **2016**, *114*, 1304.

[23] T. L. Fletcher and P. L. A. Popelier, *Theor.Chem.Acc.* **2015**, *134*, 135:1.

[24] T. L. Fletcher and P. L. A. Popelier, *J.Comput.Chem.* **2017**, *38*, 336.

[25] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov and P. E. Bourne, *Nucleic Acids Research* **2000**, *28*, 235.

[26] R. B. M. J. F. Gaussian 09, G. W. Trucks, H. B. Schlegel, G. E. Scuseria, M. A. Robb, J. R. Cheeseman, G. Scalmani, V. Barone, B. Mennucci, G. A. Petersson, H. Nakatsuji, M. Caricato, X. Li, H. P. Hratchian, A. F. Izmaylov, J. Bloino, G. Zheng, J. L. Sonnenberg, M. Hada, M. Ehara, K. Toyota, R. Fukuda, J. Hasegawa, M. Ishida, T. Nakajima, Y. Honda, O. Kitao, H. Nakai, T. Vreven, J. A. Montgomery, Jr., J. E. Peralta, F. Ogliaro, M. Bearpark, J. J. Heyd, E. Brothers, K. N. Kudin, V. N. Staroverov, R. Kobayashi, J. Normand, K. Raghavachari, A. Rendell, J. C. Burant, S. S. Iyengar, J. Tomasi, M. Cossi, N. Rega, J. M. Millam, M. Klene, J. E. Knox, J. B. Cross, V. Bakken, C. Adamo, J. Jaramillo, R. Gomperts, R. E. Stratmann, O. Yazyev, A. J. Austin, R. Cammi, C. Pomelli, J. W. Ochterski, R. L. Martin, K. Morokuma, V. G. Zakrzewski, G. A. Voth, P. Salvador, J. J. Dannenberg, S. Dapprich, A. D. Daniels, Ö. Farkas, J. B. Foresman, J. V. Ortiz, J. Cioslowski, and D. J. Fox, Gaussian, Inc., Wallingford CT, USA, **2009**.

[27] Y. Zhao, J. Pu, B. J. Lynch and D. G. Truhlar, *PhysChemChemPhys* **2004**, *6*, 673.

[28] P. L. A. Popelier and F. M. Aicken, *J.Am.Chem.Soc.* **2003**, *125*, 1284.

[29] S. Cardamone and P. L. A. Popelier, *J.Comput.Chem.* **2015**, *36*, 2361.

[30] T. J. Hughes, S. Cardamone and P. L. A. Popelier, *J.Comput.Chem.* **2015**, *36*, 1844.

[31] T. A. Keith, AIMAll, TK Gristmill Software, Overland Park KS, USA, (aim.tkgristmill.com) **2016.**

[32] M. J. L. Mills and P. L. A. Popelier, *Comput.Theor.Chem.* **2011**, *975*, 42.

[33] M. J. L. Mills and P. L. A. Popelier, *Theor.Chem.Acc.* **2012**, *131*, 1137.

[34] M. Rafat and P. L. A. Popelier, *J.Comput.Chem.* **2007**, *28*, 2602.

[35] F. M. Aicken and P. L. A. Popelier, *Can.J.Chem.* **2000**, *78*, 415.

[36] G. Matheron, *Econ. Geology* **1963**, *58*, 1246.

[37] D. R. Jones, *J.Global Optim.* **2001**, *21*, 345.

[38] D. R. Jones, M. Schonlau and W. J. Welch, *J.Global Optim.* **1998**, *13*, 455.

[39] N. Di Pasquale, S. J. Davie and P. L. A. Popelier, *J.Chem.Theor.Comp.* **2016**, *12*, 1499−1513.

[40] G. M. Laslett, *J.Amer.Stat.Assoc.* **1994**, *89*, 391.

[41] T. L. Fletcher, PhD thesis, School of Chemistry, University of Manchester, Great Britain, **2014.**

[42] P. Bultinck, R. Vanholme, P. L. A. Popelier, F. De Proft and P. Geerlings, *J.Phys.Chem.A* **2004**, *108*, 10359.