FARADAY

GULP: A computer program for the symmetry-adapted simulation of solids

Julian D. Gale

*Department of Chemistry, Imperial College of Science, Technology and Medicine,
South Kensington, UK SW7 2AY*

Algorithms for the symmetry-adapted energy minimisation of solids using analytical first and second derivatives have been devised and implemented in a new computer program GULP. These new methods are found to lead to an improvement in computational efficiency of up to an order of magnitude over the standard algorithm, which takes no account of symmetry, the largest improvement being obtained from the use of symmetry in the generation of the hessian. Accelerated convergence techniques for the dispersion energy are found to be beneficial in improving the precision at little extra computational cost, particularly when a one centre decomposition is possible or the Ewald sum weighting towards real-space is increased.

1 Introduction

Over the past decade, computer simulation techniques have become an increasingly valuable tool in science, as an aid to the interpretation of experimental data and as a means of yielding an atomic-level model.[1,2] The scope of such methods has advanced alongside the developments in computational hardware, as has their accuracy, to the point where predictions can now be made ahead of experiment.[3]

Computer modelling of bulk crystal structures has many applications such as prediction of the relative energetics of different polymorphs,[4] determination of the mechanical properties of solids and, recently, generation of possible atomic arrangements to assist in the solution of crystal structures from diffraction techniques.[5]

The development of the methodology for the simulation of inorganic and organic materials has largely evolved independently to date. For oxides and halide salts, the most successful models used have been based on a fully ionic description with a shell-model treatment of ion polarisation.[6] This approach in the UK has been embodied in a series of computer codes, initially originating from the Harwell Laboratory and later developed within the academic community, such as PLUTO,[7] METAPOCS[8] and THBREL.[9]

For organic materials, interatomic potential calculations have utilised the natural connectivity of covalent systems to develop the molecular mechanics approach. The pioneering programs in this field, such as WMIN of Busing[10] and PCK6 of Williams,[11] were able to simplify the problem by working with rigid molecules and, therefore, only intermolecular potentials had to be considered. However, varying degrees of intramolecular flexibility could also be introduced by defining molecules as a series of coupled rigid fragments.

A major difference between the two communities has been their approach to the use of crystal symmetry. The early organic codes all made use of the space group to constrain the symmetry, while the equivalent inorganic programs disregarded the symmetry once the crystal structure had been generated. One reason for this difference lies in the fact that much of the early organic work made use of numerical derivatives; the use of symmetry therefore introduces no complications.

One recent work has favoured the removal of symmetry constraints during minimisation.[12] However, if the program is sufficiently numerically accurate a material will not lower its space group during a minimisation, regardless of whether the symmetry is explicitly used or not, since the force acting to break the symmetry will be zero. The stability of a particular space group can readily be tested following a constrained minimisation, by checking for imaginary phonon modes and then, if necessary, a distortion can be applied along the direction of the imaginary eigenvector.

Hence, the debate as to whether it is worth using the space group or not depends on the relative merits of mildly increased complexity *vs.* any potential computational speed-up. The benefits of symmetry to computational speed are twofold. First, the number of variables is reduced in the minimisation procedure and given that, for Newton–Raphson techniques, the rate of convergence is proportional to the number of variables (at least in the absence of exact second derivatives), this will lead to fewer optimisation cycles. Also, the inversion of the hessian will be considerably more economical given that this is an N^3 process. Secondly, the actual calculation of the energy and its derivatives can be performed using a symmetry-adapted algorithm which should be more efficient than the standard approach since the calculation of equivalent interactions need not be duplicated.

In this paper the aim is to demonstrate that the use of symmetry can lead to greatly improved efficiency giving, in some extreme cases, up to an order of magnitude speed increase. These algorithms have been embodied in a new code for the simulation of three-dimensional periodic systems, the general utility lattice program (GULP), which is suitable for the treatment of both inorganic and organic systems with fully flexible molecules.

One of the principal applications of the program has been to the derivation of empirical potential parameters through least-squares fitting. Of particular utility in this case is the ability to treat multiple structures within the same run. The topic of force-field derivation in relation to GULP has been described in a previous publication[13] and any further discussion of this aspect will be omitted here.

2 Method

The calculation of the energetics of a three-dimensional system theoretically involves the evaluation of interactions between all species, be they cores, shells or united atom units, within the unit cell and their periodic replications to infinity. As this is clearly not feasible, some finite cut-off must be placed on computation of the interactions. We can decompose the components of the lattice energy into two classes: long- and short-range potentials. These categories can then be treated differently.

The summation of the short-range forces can normally be readily converged directly in real space until the terms become negligible within the desired accuracy. However, other terms may decay slowly with distance, particularly since the number of interactions increases as $4\pi r^2 N_\rho$, where N_ρ is the particle number density. In particular, the electrostatic energy is conditionally convergent since the number of interactions increases more rapidly with distance than the potential (which is proportional to $1/r$) decays. Hence, the two classes of energy components will be considered separately.

2.1 Long-range potential

The electrostatic energy is the dominant term for many inorganic materials, particularly oxides, and therefore it is important to evaluate it accurately. For small- to moderate-sized systems this is most efficiently achieved through the Ewald summation[14] in which the inverse distance is rewritten as its Laplace transform and then split into two rapidly convergent series, one in reciprocal-space and one in real-space. The distribution of the summation between real- and reciprocal-space is controlled by a parameter η. The resulting expression for the energy is:

$$E_{\text{recip}} = \left(\frac{1}{2}\right)\frac{4\pi}{V}\sum_{G}\frac{\exp(-G^2/4\eta)}{G^2} \times \sum_{i}\sum_{j} q_i q_j \exp(-iGr_{ij})$$

$$E_{\text{real}} = \frac{1}{2}\sum_{i}\sum_{j}\frac{q_i q_j \operatorname{erfc}(\eta^{1/2} r_{ij})}{r_{ij}}$$

The Ewald sum has a scaling with system size of $N^{3/2}$. This is achieved when the optimal value of η is chosen.[15] Selection of this value can be made based on the criterion of minimising the total number of terms to be evaluated in real- and reciprocal-space, weighted by the relative computational expense for the operations involved, w:

$$\eta_{\text{opt}} = \left(\frac{nw\pi^3}{V^2}\right)^{1/3}$$

where n is the number of species in the unit cell, including shells, and V is the unit cell volume.

The above formula is derived by Jackson and Catlow,[16] except that the value of w is implicitly assumed to be unity. It is generally found that the parameter, w, which reflects the ratio of the computational expense in reciprocal- and real-space, is not constant as a function of system size, owing to implementational factors. Later, in the Results section, the question of the optimum value of w will be further examined.

Because the summation of the real-space terms is performed concurrently with the short-range potentials, it can be beneficial to match the real-space cut-off to the short-range cut-off and also to keep it at less then the shortest unit cell vector for moderate to large systems, as this leads to greater efficiency in the search for translational image interactions.

The maximum electrostatic cut-offs in real- and reciprocal-space can then be written in terms of the optimum value of η:

$$R_{\max} = f/\eta_{\text{opt}}^{1/2}\,; \quad G_{\max} = 2f\eta_{\text{opt}}^{1/2}\,; \quad f = -(\ln A)^{1/2}$$

where A is an accuracy parameter which controls the magnitude of terms to be neglected in the Ewald sum. A value of 10^{-8} for A is found to give sufficiently accurate results for most systems, though those with large unit cells may require an increased value.

Recently, there has been increasing interest in many techniques which achieve linear or $N \log N$ scaling for the evaluation of the electrostatic contributions, such as the fast multipole method[17] and particle mesh approaches.[18] These methods are clearly beneficial for very large systems, but have a larger prefactor, and there is some debate as to where the crossover point with the Ewald sum occurs. The best estimates indicate that this happens at close to 10 000 ions. Since GULP is currently aimed at crystalline materials, most systems to be studied will be considerably smaller than this, and so the Ewald technique represents the most efficient solution.

The only remaining issue is how to select the charges for the electrostatic energy. For the majority of ionic inorganic materials, particularly oxides and halides, formal charges are a natural choice. Even for materials which are clearly not fully ionic, based on the results of *ab initio* electronic structure calculations, such as silicates, formal charges work well in practice, provided that a shell model is employed. For low-symmetry structures, a dipolar shell model is sufficient to absorb most of the effects of partial covalency, whereas for high-symmetry systems a breathing shell (where the shell has a finite variable radius) may be needed in conjunction with formal charges.[19]

For molecular crystals, the charges may be determined independently, for example by fitting to a quantum mechanical electrostatic potential-energy surface[20] for the isolated species, or may be empirically fitted if there is sufficient experimental data for the crystal. An attractive alternative is to use electronegativity equalisation methods[21] to determine the charges *in situ*. This option has been implemented within GULP.

2.2 Interatomic potentials

For many ionic materials the predominant short-range potential description used is the Buckingham potential, which consists of a repulsive exponential and an attractive dispersion term between pairs of species. For more general systems, such as molecular organics, semiconductors, metals and inert gases, a wider range of functional forms is required. GULP contains a variety of standard two-, three- and four-body potentials (Table 1). Additionally, there is the option to input potentials as a series of energies *vs.* distance with a spline function to interpolate between the points.

For the Lennard-Jones potential it is possible to input the parameters for each pair of atoms or combination rules can be used based on one-centre coefficients.

In the most commonly used interatomic potentials, the so called 'short-range' cut-off is controlled by the dispersion term as represented by $-C/r^{-6}$ as the exponential repulsion and terms dependent on higher powers of the distance decay more rapidly. Unfortunately, however, these dispersion terms can often be significant, even when summed out to twice the distance needed to converge the repulsive terms; such truncation of the dispersion terms generally leads to small, but noticeable, discontinuities in the energy surface which can lead to termination of an optimisation before the gradient norm falls below the required tolerance.

As pointed out by Williams,[22] it is straightforward to accelerate the convergence of the dispersion energy by the same procedure as for the electrostatic energy. When transformed partially into reciprocal space the resulting expressions for the dispersion energy are:

$$E^{\text{C6}}_{\text{recip}} = \frac{1}{2}\sum_{j}\sum_{j} - C_{ij}\left(\frac{\pi^{3/2}}{12V}\right)\sum_{G}\exp(iGr)G^3 \times \left[\pi^{1/2}\operatorname{erfc}\left(\frac{G}{2\eta^{1/2}}\right) + \left(\frac{4\eta^{3/2}}{G^3} - \frac{2\eta^{1/2}}{G}\right)\exp\left(-\frac{G^2}{4\eta}\right)\right]$$

$$E^{\text{C6}}_{\text{self}} = \frac{1}{2}\sum_{i}\sum_{j} - \frac{C_{ij}}{3}\,[(\pi\eta)^{3/2}] + \sum_{i}\frac{C_{ij}\eta^3}{6}$$

Table 1 Functional forms for interatomic potentials incorporated into GULP

potential name	formula	units for input
Buckingham	$A\exp(-r/\rho) - C\,r^{-6}$	A in eV, ρ in Å, C in eV Å^6
Lennard-Jones (combination rules permitted)	$A\,r^{-m} - B\,r^{-n}$ or $\varepsilon[c_1(\sigma/r)^m - c_2(\sigma/r)^n]$ $c_1 = [n/(m-n)]*(m/n)**[m/(m-n)]$ $c_2 = [m/(m-n)]*(m/n)**[n/(m-n)]$	A in eV Å^m, B in eV Å^n ε in eV, σ in Å
harmonic (k_3/k_4 optional)	$1/2\ k_2(r-r_0)^2 +$ $1/6\ k_3(r-r_0)^3 +$ $1/12\ k_4(r-r_0)^4$	k_2 in eV Å^{-2}, r_0 in Å k_3 in eV Å^{-3} k_4 in eV Å^{-4}
Morse	$D(\{1-\exp[-a(r-r_0)^2]\}^2 - 1)$	D in eV, a in Å^{-2}, r_0 in Å
Spring (core-shell)	$1/2\ k_2 r^2 + 1/24\ k_4 r^4$	k_2 in eV Å^{-2}, k_4 in eV Å^{-4}
general	$A\exp(-r/\rho)r^{-m} - C\,r^{-n}$	A in eV Å^m, ρ in Å, C in eV Å^n
Stillinger–Weber (2-body)	$A\exp[\rho/(r-r_{\text{max}})]\ (B\,r^{-4} - 1)$	A in eV, ρ in Å, B in Å^4
Stillinger–Weber (3-body)	$K\exp[\rho/(r_{12}-r_{\text{max}}) + \rho/(r_{13}-r_{\text{max}})]$ $[\cos(\theta_{213}) - \cos(\theta_0)]^2$	K in eV, ρ in Å
three-body harmonic	$1/2\ k_2(\theta-\theta_0)^2 +$ $1/6\ k_3(\theta-\theta_0)^3 +$ $1/12\ k_4(\theta-\theta_0)^4$	k_2 in eV rad^{-2}, θ_0 in degrees k_3 in eV rad^{-3} k_4 in eV rad^{-4}
three-body harmonic + exponential	$1/2\ k_2(\theta_{213}-\theta_0)^2 \times$ $\exp(-r_{12}/\rho)\exp(-r_{13}/\rho)$	k_2 in eV rad^{-2}, θ_0 in degrees, ρ in Å
Axilrod–Teller	$K(1 + 3\cos\theta_{213}\cos\theta_{123}$ $\cos\theta_{132})/(r_{12}r_{13}r_{23})^3$	K in eV Å^9
three-body exponential	$A\exp(-r_{12}/\rho)\exp(-r_{13}/\rho)\exp(-r_{23}/\rho)$	A in eV, ρ in Å
Urey–Bradley	$1/2\ k(r_{23}-r_0)^2$	k in eV Å^{-2}, r_0 in Å
four-body	$k[1+\cos(n\phi-\phi_0)]$	k in eV, ϕ_0 in degrees
Ryckaert–Bellemans	$\Sigma\ k_n(\cos\phi)^n$	k_n in eV

r represents the distance between two atoms *i* and *j*, θ_{ijk} represents the angle between the two interatomic vectors *i*–*j* and *j*–*k* and ϕ_{ijkl} is the torsional angle between the planes *ijk* and *jkl*.

$$E_{\text{real}}^{\text{C6}} = \frac{1}{2}\sum_i \sum_j \sum_{\text{cells}} -\frac{C_{ij}}{r^6}\left(1 + \eta r^2 + \frac{\eta^2 r^4}{2}\right)\exp(-\eta r^2)$$

The additional computational overhead to perform this summation is small and, when combined with the reduction in the real-space cut-off, the CPU time taken to achieve a particular target accuracy should be greatly diminished. Two algorithms have been implemented, depending on whether all the dispersion *C* coefficients can be factorised into one-centre parameters according to a simple geometric mean combination rule:

$$C_{ij} = (C_i C_j)^{1/2}\,; \quad \text{for all } i \text{ and } j$$

When such a factorisation can be performed there is a significant increase in efficiency of the calculation in reciprocal-space, since the loop over *i* and *j* can be transformed into a single sum:

$$\sum_i \sum_j C_{ij}\exp(iGr_{ij}) = \sum_i\ [C_i\exp(iGr_i)]^2$$

For molecular crystals, it is important to distinguish between intra- and inter-molecular potentials as they often take different functional forms. Hence, cut-offs need to be controlled, not by distance, but by molecular connectivity. Similarly, many standard molecular mechanics force-fields require 1–2 and 1–3 interactions to be Coulomb-subtracted, so that the r_0 and θ_0 values closely resemble the physically meaningful parameters of bond length and bond angle, respectively. GULP has the option to automatically locate bond lengths based on the sum of covalent radii and thus determine the connectivity. Based on this, all the required functionality is present to handle molecular systems.

2.3 Energy minimisation

Efficient minimisation of the energy is an essential part of the simulation of solids as it is a prerequisite for any subsequent evaluation of physical properties and normally represents the computationally most demanding stage. Several types of standard minimisations are available in GULP, the most commonly used being to optimise at constant pressure, in which all internal and cell variables are included, or at constant volume, where the unit cell remains frozen. However, a range of other possibilities exist, such as a shell optimisation where only the shell coordinates and radii vary. This is useful in analysing the electronic polarisation contribution to the relaxation about an impurity, for example.

The most efficient minimisers are those which are based on the Newton–Raphson method, in which the hessian or some approximation to it is used. The minimisation search direction, ***x***, is then given by;

$$\boldsymbol{x} = -\boldsymbol{H}^{-1}\boldsymbol{g}$$

where ***H*** is the hessian matrix and ***g*** is the corresponding gradient vector. The default minimiser in GULP uses the exact second-derivative matrix, calculated analytically, to initialise the hessian for the minimisation variables and then subsequently updates it using the Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm, though the Davidon–Fletcher–Powell update is also an option.[23] The hessian is explicitly recalculated when either the energy drops by more than a certain criterion in one step (which usually only happens at the start of a minimisation, when the system is in a non-quadratic region) or the angle between the gradient and search vectors becomes too large. At each cycle, a line search is performed to obtain the optimum step length along the search vector.

The above approach generally leads to rapid convergence within a few cycles for most systems, except where there are particularly soft modes in the hessian. To deal with such cases, the rational function optimiser (RFO)[24] is available, which attempts to remove imaginary modes from the hessian, thus forcing it to be positive definite. The use of RFO can lead to rapid convergence in cases where the default minimiser has difficulty. The downside is that it is much more expensive per cycle.

In many cases, the exact second-derivative matrix is not needed at the start of an optimisation as the system may be in a non-quadratic region of the potential-energy surface. The hessian can then be started as a unit matrix and updated subsequently using the BFGS procedure, with a switch to the exact hessian occurring once the gradient has dropped below some threshold value. When running very large systems it is necessary to use conjugate gradients instead of a hessian-based technique as the memory requirements for storing even a lower half triangular second derivative matrix become prohibitive and matrix operations start to dominate the computational expense of the calculations.

2.4 Calculation of properties

Once a structure has been optimised, there is a wide range of properties that can be calculated in the solid state for comparison with experiment. Conversely, these properties can also be used in the derivation of interatomic potentials from such data. The properties that can currently be calculated within GULP include the elastic constants, high-frequency and static relative permittivities, piezoelectric constants and phonon frequencies. All these are readily derived from the second derivative matrix and formulae can be found, along with expressions for the second derivatives, in an article by Catlow and Mackrodt.[25]

Phonon frequencies may be evaluated in GULP at any general $\boldsymbol{k}$ vector within the Brillouin zone. Hence, dispersion curves and phonon densities of states may readily be generated. In order to compute accurate thermodynamic properties, it is necessary to integrate across the Brillouin zone, for which the special points scheme of Monkhorst and Pack[26] is implemented, based on a regular grid whose density is controlled by a shrinking factor along each cell vector. For systems with small unit cells, a large number of k points are required to achieve convergence. However, these also tend to be systems with a high degree of symmetry. Hence, the program will automatically use the Patterson group of the Brillouin zone to reduce the number of k points to those unique ones of the asymmetric wedge.[27]

2.5 Symmetry adaptation

Symmetry information, which is supplied by the user to GULP in the form of either the space group symbol or the sequence number in International Tables,[28] is utilised in two main ways. First, it is used to identify the minimum set of variables for the minimisation procedure and secondly, it is used to accelerate the calculation of the energy and its first and second derivatives during optimisations.

In building the list of variables for minimisation, we need only consider the fractional coordinates of the asymmetric unit for the internal parameters. This is because all atomic positions, $\boldsymbol{X}_n$, can be related to those of the asymmetric unit, $\boldsymbol{X}_a$, by a roto-translation which for the nth operator of the space group is given by;

$$\boldsymbol{X}_n = \boldsymbol{R}_n \boldsymbol{X}_a + \boldsymbol{t}_n$$

where $\boldsymbol{R}_n$ is a 3×3 matrix and $\boldsymbol{t}_n$ is a translational vector.

However, even within the asymmetric unit, not all coordinates can be permitted to vary in an unconstrained fashion since, if an atom is situated on a special position, the site multiplicity may change for a particular displacement. Hence, special values of the coordinates, typically zero, a half, or a third, must be held fixed where necessary. Similarly, the program automatically searches for any constraints between the x, y and z coordinates that must be imposed to preserve a special position.

Minimisation of the unit cell in a constant pressure calculation occurs through the use of a strain matrix, rather than by direct consideration of the cell parameters. This automatically eliminates the three unwanted rotational degrees of freedom with respect to the cartesian components of the cell vectors. Although this can also be achieved by fixing three of the cartesian components, working with strains has the advantage that the derivatives are readily calculated with little more complexity than for the internal cartesian ones. Furthermore, when calculating the elastic and piezoelectric constants the second derivatives need to be formulated in terms of the strains.

The strain matrix, which scales the matrix of cell vector components, contains six unique components and takes the form;

$$\begin{pmatrix} 1+\varepsilon_1 & 1/2\varepsilon_6 & 1/2\varepsilon_5 \\ 1/2\varepsilon_6 & 1+\varepsilon_2 & 1/2\varepsilon_4 \\ 1/2\varepsilon_5 & 1/2\varepsilon_4 & 1+\varepsilon_3 \end{pmatrix}$$

The unit-cell type (triclinic, monoclinic, orthorhombic, tetragonal, cubic, hexagonal or rhombohedral) is also considered when determining the allowed variables. It is possible to select the strains that can be varied to preserve the cell symmetry, provided the correct cell orientation is chosen with respect to the cartesian axes. For example, for an orthorhombic cell the on-diagonal strains ε_1, ε_2 and ε_3 are allowed to optimise independently while numbers ε_4, ε_5 and ε_6 are fixed at zero, whereas for a cubic system all three principal axis strains would be constrained to be equal.

Reducing the number of variables has several benefits, including lowering the number of optimisation cycles required and elimination of problems associated with numerical noise artificially decreasing the symmetry during minimisation. When the number of atoms in the unit cell is large, one of the most expensive parts of the computation is the inversion of the hessian matrix which goes as the third power of the number of variables. Consideration of only the asymmetric unit can remove this bottleneck from the calculation.

Interatomic potential calculations can scale as the square of the number of atoms and so anything that reduces the number of atoms will accelerate the method. A simple use of symmetry to achieve this occurs if the structure is input as a centred unit cell (*i.e.* A, B, C, F, I or R); this can be transformed to the primitive form during the optimisation, and then transformed back to the user's preferred cell afterwards. This can immediately reduce the number of atoms by up to a factor of four.

One of the main aims of this work has been to produce a symmetry-adapted algorithm to improve the speed of each function evaluation. Initially we consider the case of a two-body real-space potential; the reciprocal-space case will be discussed later.

In a conventional energy calculation unique pairs of species are considered in a lower half triangular fashion, plus any interaction between a specie and its own periodic replications. This leads to a total number of interacting pairs (neglecting multiple image interactions since this does not affect the argument) given by;

$$N_{\text{pair}} = \tfrac{1}{2} N_{\text{full}} (N_{\text{full}} + 1)$$

where N_{full} is the total number of species within the unit cell. However, many of these pairs will be identical, for systems which are not of $P1$ symmetry, except for a rotation and/or translation operation. An alternative approach is to consider the interaction between each of the symmetry-unique species and all other species within the unit cell. If the asymmetric unit contains N_{asym} species then total number of pairs to be considered can now be written as:

$$N_{\text{pair}} = N_{\text{asym}} N_{\text{full}}$$

Hence, this symmetry-adapted algorithm will be more efficient than the conventional one provided the following inequality is satisfied:

$$N_{\text{full}} > 2N_{\text{asym}} - 1$$

This will be the case for the vast majority of space groups where there is a multiplicity higher than two for the general position.

So far, we have only invoked the symmetry equivalence between different sites within the structure. There is also the possibility of point group symmetry about each site within the asymmetric unit. In trying to find the optimal symmetry-adapted algorithm we must consider the trade-off between complexity and gain in computational efficiency. Although we have yet to implement fully the use of site rotational symmetry, preliminary results suggest that the benefits are marginal. This is because, although there are fewer pairs of atoms between which to calculate the interactions, the number of operations required to generate the derivatives increases as does the information to be stored. If the site point symmetry is ignored, then each pairwise energy calculation turns out to be no more expensive than in the non-symmetry adapted algorithm.

High levels of rotational symmetry about each atomic site only tend to be found for simple crystal structures, such as rock salt, which are trivial to calculate in terms of computational expense anyway. For larger, more expensive systems, there seems to be little or no benefit to be gained from more extensive use of symmetry. This differs from the experience found for solid-state Hartree–Fock calculations, for instance, because of the low cost of each pairwise evaluation for an interatomic potential relative to the evaluation of a quantum mechanical integral.

2.5.1 First derivatives. Evaluation of the first derivatives in the symmetry-adapted algorithm is straightforward. The forces are calculated as per the conventional expressions, except that only those acting on the asymmetric unit species are needed. Since the degrees of freedom of the reduced asymmetric unit are used as the minimisation variables, the first derivatives for these ions need to be scaled by the number of symmetry-equivalent species or, in crystallographic terms, the site multiplicity. This simple way of obtaining the symmetry-reduced derivatives results from the fact that the first derivatives transform in the same way as the fractional coordinates under the rotation operators of the space group.

Unfortunately, the strain first derivatives are sensitive to the rotational orientation of pairs of species in the crystal and cannot be evaluated between only the asymmetric unit and all other species and then scaled by the site multiplicities. For example, in a cubic system the symmetry-reduced summation gives strain derivatives with respect to *xx*, *yy*, *zz* which are not equal, as they should be. The sum of the three derivatives is correct, it is just that the contributions have not been averaged by symmetry when only one of several equivalent pairs are considered. However, we can use the constraints on the strain derivatives that arise from symmetry to arrive at the correct values again without resorting to explicit use of rotation operators.

2.5.2 Second derivatives. The generation of the hessian matrix is more complex. If we first consider the case in which the second-derivative matrix is calculated without the use of symmetry, then there are three steps involved in obtaining the hessian for the minimal set of variables. First, since the second derivatives are initially calculated in Cartesian space (which is necessary for the evaluation of phonon frequencies and other curvature-related properties) we must transform the matrix into fractional space by multiplication of each 3 × 3 block by the cell vector matrix and its transpose. This leads to a square matrix, $\boldsymbol{D}_{\text{ff}}$, of side $3N_{\text{full}}$.

The second step involves reduction of $\boldsymbol{D}_{\text{ff}}$ to $\boldsymbol{D}_{\text{aa}}$, the second-derivative matrix, based only the asymmetric unit atoms, using the transformation matrix $\boldsymbol{T}_{\text{fa}}$ and its transpose, $\boldsymbol{T}_{\text{af}}$:

$$\boldsymbol{D}_{\text{aa}} = \boldsymbol{T}_{\text{af}} \cdot \boldsymbol{D}_{\text{ff}} \cdot \boldsymbol{T}_{\text{fa}}$$

The transformation matrix above is sparse and contains 3 × 3 blocks between each asymmetric unit atom and its symmetry-equivalent images, which are just the rotation matrices, $\boldsymbol{R}_n$, which generated those images. The final step is to reduce $\boldsymbol{D}_{\text{aa}}$ according to the matrix of constraints, if required. In practice, the second and third steps can be combined to yield a single step with a transformation matrix which is independent of the cell vectors and thus needs only to be generated once at the start of the calculation.

Generating the full second-derivative matrix is clearly wasteful and becomes increasingly expensive relative to the evaluation of the energy and forces as the system size increases. All the information needed in the second derivative matrix is contained in the columns between the asymmetric unit and all other species within the unit cell. Hence, we need only calculate the corresponding submatrix $\boldsymbol{D}_{\text{fa}}$, which has dimensions of $3N_{\text{full}} \times 3N_{\text{asym}}$. The transformation matrix which maps this $\boldsymbol{D}_{\text{aa}}$ is the symmetry-adapted case, $\boldsymbol{S}_{\text{af}}$, now contains 3 × 3 blocks between an asymmetric unit species *a* and its symmetry equivalent images *f* in the full cell, which are given by:

$$S_{\text{af}} = \frac{1}{N_{\text{eqv}}} \sum_{\alpha=1}^{N_{\text{eqv}}} \boldsymbol{R}_{\alpha}^{-1} \cdot \boldsymbol{R}_{\text{f}'}$$

where the summation is over all the symmetry-equivalent species to *a*, and the atom *f′* is the species to which *f* maps under the transformation $\boldsymbol{R}_{\alpha}^{-1}$.

This symmetry-adapted approach to generating the hessian, benefits from the same computational advantages, as does the evaluation of the gradients, in that the calculation scales as $N_{\text{asym}} N_{\text{full}}$. Similar considerations also apply to the generation of the strain–strain second-derivative matrix which is, however, initially incorrect, but corrected through the application of symmetry-related constraints. Finally, the mixed strain–internal second-derivative matrix only requires multiplication by the strain-constraint matrix and site multiplicity to achieve the proper form.

Symmetry is also readily used to reduce the number of three- and four-body terms that have to be evaluated, when these are required. Although the algorithm for looping over species is a little more complex than for pair potentials the approach to symmetry adaptation is the same as above and will, therefore, not be considered in further detail here.

In evaluating terms in reciprocal-space there is often no need to use symmetry in the evaluation of the energy. This is because for interactions which can be expressed as a product of two one-centre terms, such as charges, the most efficient means of calculation involves a single loop over the species within the unit cell rather than a pairwise summation, as discussed earlier. However, the calculation of the derivatives can again be restricted to the asymmetric unit for the forces and interactions between the asymmetric unit and full cell for the second derivatives.

Finally, the algorithm for the calculation of charges according to electronegativity equalisation has been implemented with the inclusion of symmetry to improve the efficiency of this process.

2.6 Program features

All the above facilities, and more, have been coded into a new program, GULP, which is designed to be as widely applicable as possible to solid-state simulation problems. This article largely concentrates on its use for optimisation of structures and calculation of physical properties. However, GULP also contains the facility to derive interatomic potentials, either empirically from experimental data or by fitting quantum mechanically-determined energy surfaces. In order to facilitate the fitting of potentials, the program is designed to accept multiple structures within each input deck. After the first line of keywords which control the run type the input is free format.

GULP was originally written in FORTRAN 77, however a FORTRAN 90 version is now also available to make use of the dynamic memory features. The memory requirements are dominated by the second-derivative matrix, the coordinate transformation matrix, and the lower-half triangular hessian. In the dynamic memory version, if the optimiser selected does not use second derivatives then these matrices are not generated, thus enabling much larger systems to be considered within a given amount of memory.

The results produced by this new program have been validated against existing programs, where possible, and also by numerical checks in the case of derivatives.

3 Results and Discussion

Having introduced the principles behind the program GULP, and in particular the ways in which symmetry is used, the following section will examine the efficiency of the algorithms implemented.

3.1 Symmetry-adapted optimisation

In order to assess the benefits of the use of symmetry, we need a range of test systems which can readily be compared. For this purpose, the microporous materials, zeolites, which are of wide general interest, represent an ideal family for study. This is because there are many different crystal structures with varying degrees of symmetry and unit cell sizes which range from 18 atoms up to in excess of a thousand.

Owing to the high level of interest in zeolites, there have been many different interatomic potentials derived for simulating their structures and properties.[29,30] By far the most successful class of potentials has been those based on the shell model. The parameters developed originally by Sanders *et al.*,[31] but with a slightly modified oxygen shell charge and spring constant, have been the most widely applied and, hence, we have chosen these for this study. The potential model consists of formal (*i.e.* integral) charges, with a shell model for the oxide anion (where the charge is split between two particles coupled by a harmonic spring and Coulomb-screened from each other) to mimic polarisation, exponential short-range repulsions and dispersive attraction in the form of a Buckingham potential, plus a harmonic three-body angle bending term for O—Si—O. Full details and parameters can be found in the article by Jackson and Catlow.[16]

For the purposes of this comparison, an accuracy factor of 10 has been used, which is higher than the default of 8, and the dispersion energy has been summed using the combined real and reciprocal space approach. The summation of the exponential repulsions is truncated when their magnitude becomes negligible relative to the accuracy of the Ewald sum. Cut-offs for the three-body potential are based on the Si—O bond length, plus a margin of tolerance.

Timings are given in Table 2 for the optimisation of a variety of zeolite structures on a single node of a Silicon Graphics Power Challenge R8000 processor with a clock speed of 90 MHz. Three times are quoted, corresponding to different levels of the use of the symmetry. In all cases symmetry is used to reduce the centred cell, where applicable, to the primitive cell and to determine the optimisation variables. The first timing is for an optimisation where no further use is made of the symmetry. In the second case, the symmetry-adapted algorithm is used to calculate the first derivatives, but not the second. Finally, in the third case, symmetry is employed at all levels of derivatives. Table 2 also contains relevant information, such as the space group, the number of species in the asymmetric unit and the number in the full primitive unit cell. Note that the number of species does not equal the number of atoms as each oxygen comprises two species, a core and a shell.

In addition to the timings, the number of cycles to achieve convergence is given. These are, on the whole, very low because the theoretical optimised structures accurately reproduce the experimental ones (which were the starting point) and the BFGS scheme is very efficient once the system begins to enter the harmonic region of the energy surface. One clear exception to the rapid convergence is the MFI, or silicalite, structure which requires 63 cycles. For most of these cycles, the gradient norm is low and close to convergence. However, the hessian contains a number of relatively soft modes which the BFGS scheme finds hard to deal with. In this case, switching to RFO minimisation once the gradient norm drops below a certain threshold greatly accelerates the convergence, and the job then completes in 25 cycles and less than half the CPU time. The case of silicalite is also noteworthy since, while the hessian within the restrictions of the space group is real, if the calculation is repeated without these constraints then imaginary modes are found. Hence, a monoclinic distortion, as previously reported,[32] is required to achieve the true minimum at 0 K.

The actual results of the zeolite optimisations will not be discussed here as there have been numerous papers already

Table 2 Illustration of the effect of symmetry on CPU times for optimising a range of zeolite structures

structure code	space group	N_{asym}	N_{full}	$D0$/s	$D1$/s	$D2$/s	cycles for convergence
ANA	$Ia\bar{3}d$	3	120	128.8	70.1	8.7	4
CHA	$R\bar{3}m$	9	60	27.5	19.0	9.9	4
FAU	$Fd3m$	9	240	720.6	364.3	62.1	5
FER	$Immm$	20	90	74.7	52.0	37.3	6
KFI	$Im\bar{3}m$	9	240	708.4	341.6	59.9	5
LTN	$Fd\bar{3}$	40	960	69606.6	36763.9	6161.7	16
MCM-22	$P6/mmm$	34	360	3021.8	1707.5	593.4	7
MEL	$I\bar{4}m2$	37	240	1966.7	1103.9	657.8	15
MFI	$Pnma$	64	480	38194.1	14017.9	8595.4	63
MOR	$Cmc2_1$	32	120	224.7	165.5	136.3	8
MTN	$Fd\bar{3}$	11	170	203.9	110.6	35.6	3
NON	$Fmmm$	25	110	297.9	215.1	157.1	12
RHO	$Im\bar{3}m$	5	120	161.4	97.5	16.8	4
SOD	$P\bar{4}3n$	4	60	28.2	13.0	4.4	7

Timings are given for a calculation which uses (a) no use of symmetry during evaluation of derivatives, $D0$, (b) symmetry for the energy and first derivatives only, $D1$ and (c) symmetry for the energy, gradients and hessian, $D2$. All values are for a single node of a Silicon Graphics Power Challenge R8000 running at 90 MHz.

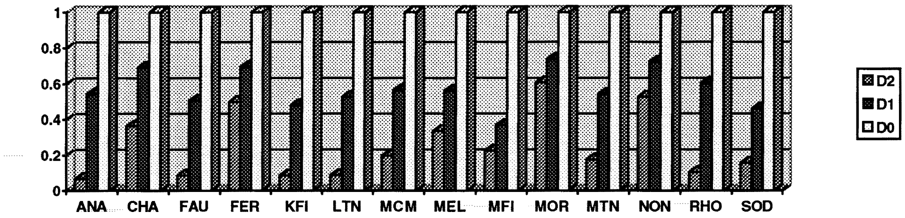

Fig. 1 Relative timings for zeolite optimisations as a function of the use of symmetry given as a ratio *vs.* the time for no use of symmetry ($D0$ = no use of symmetry in energy or derivative evaluation; $D1$ = symmetry is used for energy and first derivatives only; $D2$ = symmetry is used for the energy and all derivatives

analysing the performance of the potentials for structure reproduction and lattice energies.[4,33–35]

To enable analysis of the data in Table 2, the ratios of the three times, as a function of symmetry, are illustrated in Fig. 1. It can be seen that, in some cases, the use of symmetry can lead to a speed-up in excess of an order of magnitude. The greatest benefit tends to occur for the largest structures, which is obviously a desirable feature, as calculation of the energy and its derivatives accounts for the majority of the computer time. While there is a general correlation between improved performance in the symmetry-adapted algorithm and the ratio between the number of species in the asymmetric unit and full cell, it is not a simple function. This is because the calculation of the three-body energy and matrix diagonalisation have different scalings from the two-body terms, and from each other. Furthermore, an optimisation is a combination of points which utilise different levels of derivatives according to whether the hessian is being re-calculated exactly, updated or whether a line search is in progress.

Symmetry-adapted generation of the hessian matrix is more complex than for the first derivatives. Hence, we must consider whether this extra complexity is worthwhile. From Fig. 1 it is clear that symmetry adaptation of only the first derivatives rarely gives a speed up of greater than a factor of 2. On the whole, the use of symmetry for the second derivatives gives a much larger performance enhancement than for the first derivatives, thus justifying the use of this approach.

3.2 Convergence of dispersion energy

In this section, the aim is to compare the results obtained for calculating the dispersion energy, resulting from any $-C/r^{-6}$ terms in the potentials, either purely in real space or, alternatively, partially in real space and partially in reciprocal space. Both approaches have been applied to the case of the zeolite chabazite which was used as one of the examples in Section 3.1. As the unit cell size of chabazite is small, the default accuracy factor of 8 is sufficient and has been used in this comparison. For the purposes of this test, a system with small unit cell dimensions is preferable. Because a large amount of the computer time is spent searching for valid distances, rather than calculating potential expressions, systems with large unit cells are prone to large discontinuities in the efficiency with increasing cut-off radius when the search has to be extended to a new shell of neighbouring unit cell images.

Fig. 2 shows the total optimised energy for chabazite calculated for increasing cut-off radii in a purely real-space summation, as well as the converged energy from the expressions analogous to the Ewald sum. It is found that to achieve convergence to with 0.001 eV purely in real space requires a large cut-off of *ca.* 40 Å. To reach a high degree of convergence in this way is extremely difficult and thus justifies the extra complexity of implementing the Ewald-style summation for the dispersion terms.

An important consideration is the relative computational expense of the two approaches to the dispersion energy. In Fig. 3, the timings are illustrated as a function of cut-off radius. Not surprisingly, for small cut-off radii the purely real-space approach is faster than the more accurate method. However, the discrepancy in the energy in this region is larger than desirable. If numerical precision of 0.001 eV or greater is required then the real/reciprocal-space summation is competative.

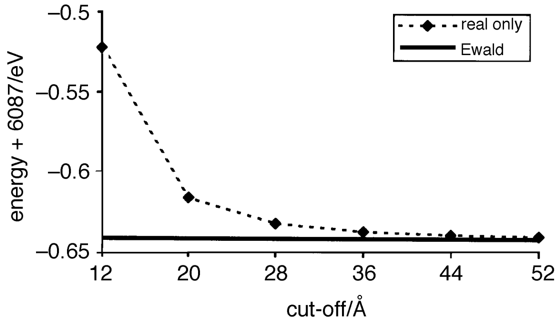

Fig. 2 Total energy for chabazite as a function of real-space cut-off for the dispersion term. The converged energy from Ewald-type summation is included for comparison.

This example actually represents an unfavourable case for the more precise method. Because both the Si—O and O—O potentials have a dispersion term, but there is none between Si—Si, it is impossible to decompose the *C* coefficients into products of one-centre terms, which considerably slows down the reciprocal-space component. To highlight this effect, the above optimisation of chabazite has been repeated with the Si—O *C* term put equal to zero, so that the program can utilise the more efficient algorithm. Timings for this are shown in Fig. 4. Note that these values are larger than those in Fig. 3 as more cycles of optimisation are needed to achieve convergence since neglecting one dispersion term leads to poorer agreement with experiment. Strictly speaking, the potential should be re-fitted to correct for this, however, here the objective is only to compare relative timings.

From the timings in Fig. 4, it can be seen that the dispersion energy can be accurately evaluated for almost negligible overhead when a one-centre decomposition is possible. Any extra expense per pairwise interaction is largely countered by the fact that the real space cut-off is very low. Pure real space

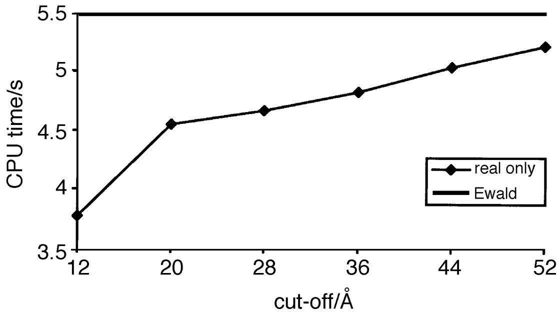

Fig. 3 Total CPU time for optimisation of chabazite as a function of real-space cut-off for the dispersion term. The equivalent value for a calculation using a Ewald-type summation is also included.

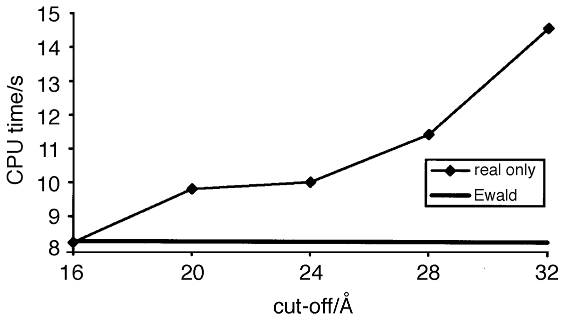

Fig. 4 Total CPU time for optimisation of chabazite as a function of real-space cut-off when the dispersion coefficients can be factorised into one-centre terms. The equivalent value for a calculation using a Ewald-type summation is also included.

summation can only compete computationally when errors of greater than 0.01 eV can be tolerated.

Converged evaluation of the dispersion energy is particularly important when performing empirical potential derivation by fitting to experimental data. If *C* terms are allowed to vary as fitted parameters then they can occasionally tend to unphysically large magnitudes with small finite cut-offs. Improved convergence damps out this tendency.

3.3 Optimisation of Ewald summation

As previously mentioned in Section 2.1, the choice of η, which controls the proportion of the lattice sums performed in real and reciprocal space, should be made so as to minimise the overall computer time. One approach to this problem is to choose the value of η that minimises the total number of terms to be evaluated. However, this does not allow for the relative computational expense of real- and reciprocal-space terms being different. Hence, in the earlier formula for η_{opt} we introduced a weighting factor *w* which can be varied to correct for this factor.

All the timings given in Table 2 for the various zeolitic materials correspond to an unoptimised value of $w = 1.0$. In this section, we investigate how close this is to the optimal choice for this parameter. For small system sizes, the timings are relatively insensitive to the value of *w*. Hence, the two most computationally demanding cases, LTN and MFI, are taken as examples, with the total CPU time being plotted against the negative logarithm of *w* for convenience (Fig. 5).

For both LTN and MFI, the minimum in the computational expense comes when *w* is less than 1.0, *i.e.* the Ewald sum is weighted in favour of the real-space summation. Although the curve for LTN contains some level of noise, it is clear that, for both systems, the optimal value of $-\log(w)$ lies in the region 1.5–2.0, leading to a value of *w* between 0.01 and 0.0316. The speed-up that results, when the optimal value of *w* is chosen, is greater than a factor of two in both cases, emphasising that it is important to tune the weighting factor for large systems.

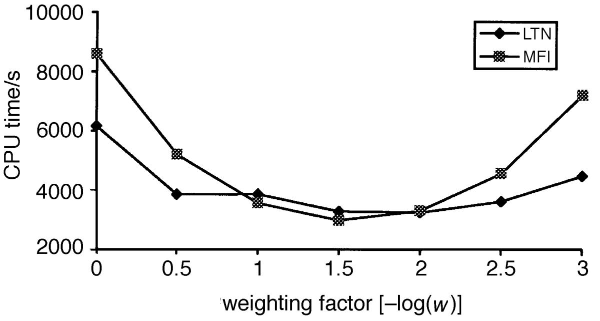

Fig. 5 Total CPU time for optimisation of the LTN and MFI structures as a function of the weighting factor, *w*, between reciprocal and real space. A value of *w* less than 1.0 implies an increased weighting towards real space.

In the absence of using the combined real/reciprocal-space sum for the dispersion terms, the optimum value of *w* tends to be much larger and can often exceed 1.0 rather than being smaller than this. The reason for this can again be traced back to the fact that the dispersion coefficients cannot always be decomposed into the products of one-centre terms. Hence, the reciprocal-space energy has to be evaluated pairwise rather than in a single loop over the species, making it far more expensive than for the standard electrostatic sum. This is highlighted by the fact that, when the dispersion terms can be decomposed to the one-centre form, much larger optimal values of *w* are obtained, close to those for the electrostatic-only case.

If the weighting factor between real and reciprocal space is optimised the accelerated convergence of the dispersion term becomes far more competitive with the pure real-space approach in the unfavourable case where the coefficients cannot be expressed as a product of one-centre terms.

4 Conclusions

In the present work it has been demonstrated that the use of symmetry in the optimisation of crystal structures is highly beneficial and can often yield an order of magnitude speed increase, with the largest increase normally arising in the generation of the hessian. Although the results presented are for derivatives calculated from interatomic potentials, the same algorithm and advantages will apply to any solid state calculation, particularly quantum mechanical ones where the maximisation of computational speed is especially crucial.

For large systems, where a series of calculations is to be performed, it is found to be well worth optimising the value of η, through tuning the reciprocal/real-space weighting, in the Ewald summation with respect to the CPU time since improvements of a factor of two or more can be obtained. Similarly, it is found to be beneficial to implement two different algorithms for the dispersion sum when using an Ewald-style summation to take advantage of one-centre factorisation when possible.

The algorithmic improvements described above have been embodied in the new program, GULP, which is designed to be a versatile tool for the simulation of solids based on interatomic potential models. A wide range of examples of the application of this program to specific problems is already in the literature.[36–41] In addition to the ability to perform optimisations and calculate second derivative-related properties, the program has many other facilities, such as the capability of Mott–Littleton defect calculations, genetic algorithms for local minimum searching, potential fitting (either empirically or to energy surfaces) and a basic molecular dynamics capability. These facilities will be described in subsequent publications.

I would like to thank Prof. C. R. A. Catlow for his valuable advice and encouragement in this project. I would also like to thank the many other colleagues who have used the program for their feedback and discussion, in particular Drs. A. L. Rohl, T. S. Bush, A. R. George and Prof. L. Glasser. This work was supported by the Royal Society through the provision of a University Research Fellowship.

References

1 C. R. A. Catlow, R. G. Bell and J. D. Gale, *J. Mater. Chem.*, 1994, **4**, 781.

2 P. A. Wright, S. Natarajan. J. M. Thomas, R. G. Bell, P. L. Gai-Boyes, R. H. Jones and J. Chen, *Angew. Chem., Int. Ed. Engl.*, 1992, **31**, 1472.

3 J. W. Couves, R. H. Jones, S. C. Parker, P. Tschaufeser and C. R. A. Catlow, *J. Phys. Condens. Matter*, 1993, **5**, 329.
4 N. J. Henson, A. K. Cheetham and J. D. Gale, *Chem. Mater.*, 1994, **6**, 1647.
5 T. S. Bush, C. R. A. Catlow and P. D. Battle, *J. Mater. Chem.*, 1995, **5**, 1269.
6 B. G. Dick and A. W. Overhauser, *Phys. Rev.*, 1958, **112**, 90.
7 C. R. A. Catlow and M. J. Norgett, UKAEA Report-M2936, 1976.
8 C. R. A. Catlow, A. N. Cormack and F. Theobald, *Acta Crystallogr., Sect. B*, 1984, **40**, 195.
9 M. Leslie, Daresbury Laboratory, UK.
10 W. R. Busing, WMIN, A Computer Program to Model Molecules and Crystals in Terms of Potential Energy Functions; ORNL-5747; Oak Ridge National Laboratory; Oak Ridge, 1981.
11 D. E. Williams, *QCPE Bull.*, 1984, **4**, 82.
12 K. D. Gibson and H. A. Scheraga, *J. Phys. Chem.*, 1995, **99**, 3752.
13 J. D. Gale, *Phil. Mag. B*, 1996, **73**, 3.
14 P. P. Ewald, *Ann. Phys.*, 1921, **64**, 253.
15 J. W. Perram, H. G. Petersen and S. W. de Leeuw, *Mol. Phys.*, 1988, **65**, 875.
16 R. A. Jackson and C. R. A. Catlow, *Mol. Simul*, 1988, **1**, 207.
17 H. G. Petersen, D. Soelvason, J. W. Perram and E. R. Smith, *J. Chem. Phys.*, 1994, **101**, 8870.
18 U. Essmann, L. Perera, M. L. Berkowitz, T. Darden, H. Lee and L. G. Pedersen, *J. Chem. Phys.*, 1995, **103**, 8577.
19 U. Schröder, *Solid State Commun.*, 1966, **4**, 347.
20 B. H. Besler, K. M. Merz Jr. and P. A. Kollman, *J. Comput. Chem.*, 1990, **11**, 431.
21 K. A. van Genechten, W. J. Mortier and P. Geerlings, *J. Chem. Phys.*, 1987, **86**, 5063.
22 D. E. Williams, *Cryst. Rev.*, 1989, **2**, 3 and 163.
23 *Numerical Recipes*, W. H. Press, S. A. Teukolsky, W. T. Vetterling and B. P. Flannery, Cambridge University Press, Cambridge, 2nd edn., 1992.
24 A. Banerjee, N. Adams, J. Simons and R. Shepard, *J. Phys. Chem.*, 1985, **89**, 52.
25 C. R. A. Catlow and W. C. Mackrodt, *Computer Simulation of Solids*, Lecture Notes in Physics no. 166, Springer-Verlag, Berlin 1982, ch. 1.
26 H. J. Monkhorst and J. D. Pack, *Phys. Rev. B*, 1976, **13**, 5188.
27 R. Ramirez and M. C. Böhm, *Int. J. Quant. Chem.*, 1988, **34**, 571.
28 *International Tables for Crystallography*, Kluwer Academic Publishers, Dordrecht, The Netherlands, vol. A, 1987.
29 B. W. H. van Beest, G. J. Kramer and R. A. van Santen, *Phys. Rev. Lett.*, 1990, **64**, 1955.
30 E. de Vos Burchart, V. A. Verheij, H. van Bekkum and B. van de Graaf, *Zeolites*, 1992, **12**, 183.
31 M. J. Sanders, M. Leslie and C. R. A. Catlow, *J. Chem. Soc., Chem. Commun.*, 1984, 1271.
32 R. G. Bell, R. A. Jackson and C. R. A. Catlow, *J. Chem. Soc., Chem. Commun.*, 1990, 782.
33 K. de Boer, A. P. J. Jansen and R. A. van Santen, *Phys. Rev. B*, 1995, **52**, 12579.
34 J. D. Gale and A. K. Cheetham, *Zeolites*, 1992, **12**, 674.
35 K-P. Schröder and J. Sauer, *J. Phys. Chem.*, 1996, **100**, 11043.
36 J. Breu and C. R. A. Catlow, *Inorg. Chem.*, 1995, **34**, 4504.
37 N. J. Henson, A. K. Cheetham and J. D. Gale, *Chem. Mater.*, 1996, **8**, 664.
38 P. Zapol, R. Pandey, M. Ohmer and J. D. Gale, *J. Appl. Phys.*, 1996, **79**, 671.
39 A. Jentys and R. W. Grimes, *J. Chem. Soc., Faraday Trans.*, 1996, **92**, 2093.
40 N. L. Allen, A. L. Rohl, D. H. Gay, C. R. A. Catlow, R. J. Davey and W. C. Mackrodt, *Faraday Discuss.*, 1993, **95**, 273.
41 T. S. Bush, J. D. Gale, C. R. A. Catlow and P. D. Battle, *J. Mater. Chem.*, 1994, **4**, 831.

Paper 6/06455H; *Received* 18*th September*, 1996