research papers

electronic Ligand Builder and Optimization Workbench (eLBOW): a tool for ligand coordinate and restraint generation

Nigel W. Moriarty,[a]* Ralf W. Grosse-Kunstleve[a] and Paul D. Adams[a,b]

[a]Lawrence Berkeley National Laboratory, One Cyclotron Road, Mailstop 64R0246, Berkeley, CA 94720, USA, and [b]Department of Bioengineering, UC Berkeley, CA 94720, USA

Correspondence e-mail: nwmoriarty@lbl.gov

The *electronic Ligand Builder and Optimization Workbench* (*eLBOW*) is a program module of the *PHENIX* suite of computational crystallographic software. It is designed to be a flexible procedure that uses simple and fast quantum-chemical techniques to provide chemically accurate information for novel and known ligands alike. A variety of input formats and options allow the attainment of a number of diverse goals including geometry optimization and generation of restraints.

1. Introduction

Recent developments in high-throughput crystallography make the automatic handling of ligands in all stages of structure solution increasingly important. A key component of the drug-design process is structure-based inquiry (Davis *et al.*, 2003), which requires the generation of a large number of protein–ligand complexes. The typical protocol for a novel ligand begins with Cartesian coordinate generation from the available chemical information. This can be a SMILES string (Weininger, 1988), a coordinate database entry, a modified molecular structure with estimated coordinates or an accurate geometry from a detailed quantum-chemical calculation. Docking of the ligand with the protein can be performed with a three-dimensional geometry or with additional information such as internal coordinates and X-ray data. The final step, refinement of the protein–ligand complex, requires the generation of geometry restraints that provide refinement packages with the relevant ideal geometry information.

The generation of Cartesian coordinates can be achieved in several ways. The use of a force-field potential to minimize geometries, such as *GROMACS* (Lindahl *et al.*, 2001), is used by *PRODRG* (Schüttelkopf & van Aalten, 2004) to generate good geometries within the limits of the empirical information. Similarly, the ligand-fitting program *AFITT* (Wlodek *et al.*, 2006) uses an empirical force field for geometry determination. Alternatively, geometries can be derived from experimental data or obtained from prior research (Kleywegt *et al.*, 2003). However, the geometries of ATP, which are very similar to adenosine, a common library entry, have been shown to be poor (Kleywegt *et al.*, 2003) in the structures deposited in the Protein Data Bank.

Refinement relies on the generation of accurate geometry restraints from ligand coordinates and their correct use and implementation in refinement. Considerable care must be taken in the generation of complex restraints such as dihedral angles, chiralities and planarities. Here, we describe methods for the computationally efficient calculation of coordinates

from molecular topologies or other descriptions and the subsequent calculation of geometry restraints for use in macromolecular refinement.

2. Methods

2.1. Overview

The *electronic Ligand Builder and Optimization Workbench* (*eLBOW*) is a suite of modules written in Python within the framework of *PHENIX* (Adams *et al.*, 2002). It is designed for the reliable generation of both Cartesian coordinates and geometry restraints.

2.2. Inputs

A flowchart of procedures in *eLBOW* is shown in Fig. 1. Many chemical formats can be used to describe a small molecule. These fall into two broad categories: those with Cartesian coordinates and those without. The latter group contains the SMILES string (Weininger, 1988) and the Mol2D format, which provide only topology information. The standard geometry restraints CIF file can lack Cartesian coordinates but contains the topology information and is therefore in the class of chemical input where the geometric information of the molecule is contained in the internal coordinates, including bonds and angles. This is a chemically intuitive coordinate system that *eLBOW* uses directly to generate the restraints and Cartesian coordinates. Chemical formats containing Cartesian coordinates include the Protein Data Bank (PDB) format, the Mol3D formats and the simple XYZ format.

eLBOW is able to process all the ligands in a PDB file that contains both the protein model and ligands. In this mode, *eLBOW* cycles over the ligands in the input, processing them in turn, and delivers a file containing restraint information for all the ligands. This reduces the number of files needed to perform subsequent structure refinement.

2.3. Bond determination

A subset of the chemical input formats that contain Cartesian coordinates do not necessarily explicitly define the bond connectivity between the atoms. The PDB format is in this subset and thus requires the detection of bonds *via* distance criteria. The PDB format is also limited because bond-order information is not provided. Bond connectivity does not lead to unambiguous bond-order determination; therefore, the bond orders must be calculated using atomic valency methods and other heuristics. The inclusion of H atoms in the input greatly reduces this problem. In contrast, the Mol3D input format requires the specification of bond connections and bond orders, leading to a much more reliable and accurate topology determination.

If the chemical format does not support a description of bond connectivity or, as in the case of the PDB format, the connectivity is optional, the bonds need to be determined directly from the Cartesian coordinates. In *eLBOW*, accurate values of all diatomic bond distances are used to determine the possible bonds in a molecule. The tabulated values were

determined by optimizing the geometry of molecules containing the two relevant non-H atoms using the Hartree–Fock quantum-chemical method with the 6-311G(d,p) basis set (Hehre *et al.*, 1971). To determine connectivity, the tolerance for possible bonds is initially set to 15% of the optimal bond distance. If not all atoms are connected after this first pass, a search for possible bonds between the existing bonded fragments is performed with a 2 Å tolerance using only the shortest bonds between the fragments.

eLBOW handles resonance structures using heuristics to determine them from a hydrogen-free PDB file. Tautomerism can be handled by supplying the complete ligand including H
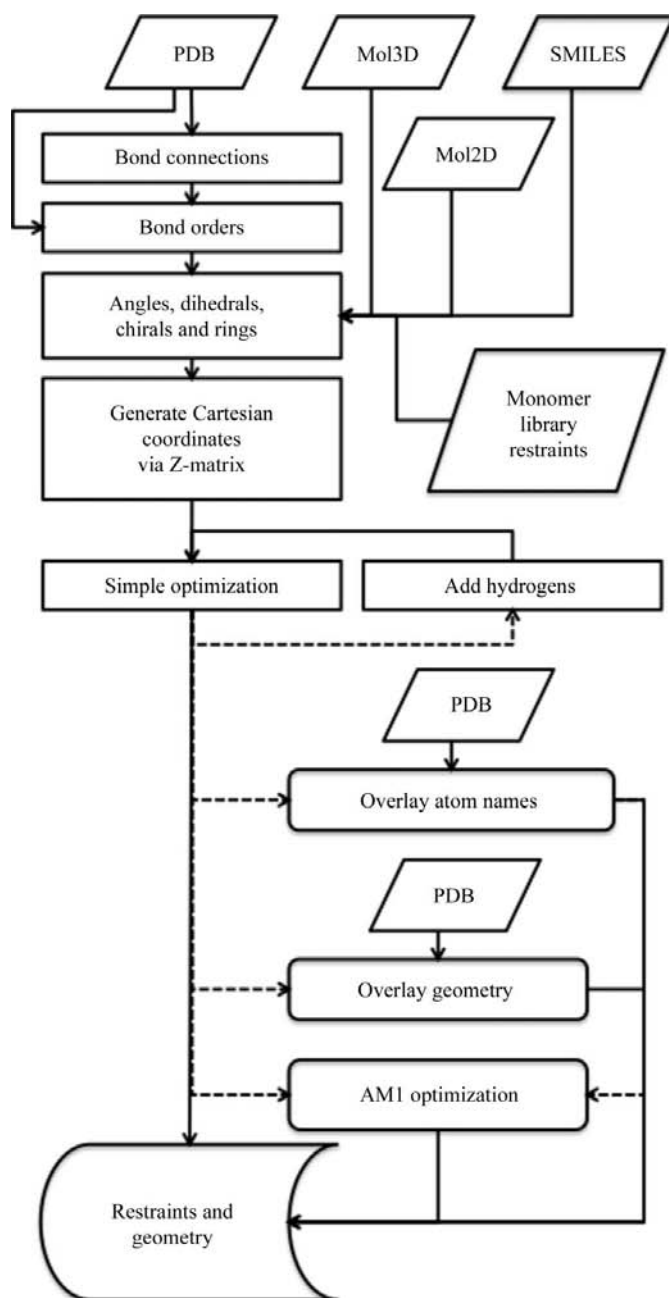

Figure 1
A flowchart of the program progression of *eLBOW*, including several of the possible alternative input and algorithmic options.

atoms. Partial charges can be determined by the semi-empirical AM1 method and included in the output.

Clearly, a poor input geometry in a format relying solely on the Cartesian coordinates has far-reaching consequences for restraint determination owing to difficulties in determining the correct connectivity in a novel ligand. This is also the case when determining bond orders.

2.4. Dependent internal coordinates

Once the bond connectivities have been determined, the dependent internal coordinates, such as angles and dihedrals, can be enumerated. The presence of loops in the geometry can also be determined. A graph-search algorithm is used to find closed loops in the molecular topology as determined by the bond connections. Typically, a closed loop of less than ten atoms could be aromatic or otherwise structured to be rigid, while longer closed loops are more flexible. Extreme examples of each are the benzene ring and a cyclic peptide chain.

The same closed loops, called rings in the following, can be fused into larger groups which can be included in bond-order determination. In the case where the bonding determination relies on the Cartesian coordinate input and where a ring or group of rings is determined to be planar with a tolerance of 0.3 Å, we infer that the rings are aromatic. Groups of rings are divided into single rings to determine the planarity of each ring. The aromaticity of a ring is determined using the Hückel rule (Hückel, 1931a,b, 1938). This is complicated by the presence of N atoms donating lone pairs to the π-bond around the ring or with one lone pair in the equatorial position and donating the remaining lone pair to the π-bond. This can change the internal bond orders and the number of H atoms bound to the ring. All cases are calculated and if one case permits an aromatic ring then it is used in further calculations.

2.5. Bond-order determination complications

Reliable bond-order determination is challenging. Reliance on the Cartesian coordinates can be misleading if the geometry of a non-aromatic ring is provided as planar. Additionally, rings can be planar but not aromatic, for example when there are double bonds connected to the ring and a double bond internal to the ring. For this reason, manual checking of output bond orders can be prudent for complex structures described using coordinate-only formats. Ideally, the molecular topology should be specified with a more content-rich chemical input format that enumerates bond orders. A mechanism is provided in *eLBOW* to use a user-editable Python script to specify the bond orders. In addition, there is a graphical restraints editor, *REEL*, available in the *PHENIX* suite.

The valence of an atom can also be used to determine bond orders. If the maximum number of bonds allowable by the atomic valence is present then the bond orders can be calculated. This is complicated by atoms such as nitrogen and phosphorus which have multiple valence states. Once the bonds that can be determined in this fashion have been found, several common chemical motifs are checked against the bond

topology. One example is the peptide linkage, which is relevant in the case of nonstandard amino acids. Bonds that are not constrained by valence are candidates for higher order bonds. Bond orders are increased independently and checked for consistency. Valence information and cues from the Cartesian coordinates are used in this step.

If the input geometry contains the H atoms (or even only a small fraction of the H atoms) then the bond-order determination is greatly simplified and the aromaticity of rings can be more readily determined from the number of H atoms bound to the ring.

2.6. Addition of H atoms

The addition of H atoms serves a number of purposes. In macromolecular crystallography, the addition of H atoms to the protein and ligand can highlight problems with packing in both the protein region and the interaction between the protein and ligand (Lovell *et al.*, 2003; Davis *et al.*, 2004). The growth of very high-resolution X-ray diffraction experiments and neutron diffraction experiments at medium to high resolution has increased the importance of modeling H atoms, as they can directly make a significant contribution to the experimental observations. H atoms are also required for the semi-empirical AM1 optimization method used within *eLBOW* and enable improved validation of the model after refinement.

2.7. Internal coordinates

Two complementary parametrizations of molecules are in wide use: Cartesian coordinates and internal coordinates. Which of these is most suitable depends on the context. Therefore, it is important to be able to convert between the parametrizations.

Internal coordinates define the molecule as a number of atoms connected by a number of bonds of certain lengths and bond orders. Angles, dihedrals, planes and chiral centers can also be included. A commonly used internal coordinate system is the Z-matrix, which is also used in *eLBOW*. It is a non-redundant system. More specifically, for a molecule of N atoms there are $N - 1$ bonds, $N - 2$ angles and $N - 3$ dihedrals. One nuance of the construction of a Z-matrix involves the final bond in a ring or loop. This bond is not specified explicitly, but is implied by the bonds, angles and dihedrals in the rest of the system.

It is straightforward to convert from Z-matrix coordinates to Cartesian coordinates. The reverse conversion is more involved. In *eLBOW*, construction of the Z-matrix is performed after the determination of ring groups and chiral centers.

By design, the Z-matrix coordinates are of course the most suitable for the manipulation of internal coordinates; for example, the change of bond lengths. Although this does not extend directly to the chiral volume or the planarity of a group of atoms, it is simple to imply planes by using the dihedral values of the atoms involved. In the case of *cis–trans* isomer-

ization, manipulation of the dihedral can be used to enforce either isomer.

The Z-matrix description generally has six fewer degrees of freedom than the corresponding Cartesian coordinate description. This can expedite the convergence of geometry optimization. In the case of the techniques used in quantum-chemical geometry optimization the improvement is marked.

2.8. Structure optimization

The values of the internal coordinates are initially estimated for input formats that lack Cartesian coordinates. However, the internal coordinate values are affected by the electronic interactions between atoms in the molecular environment. Thus, it is necessary to perform an optimization of the geometry using a technique that permits deviation of the co-ordinates from an initial assignment and ideally it should be based on internal coordinates.

There are a number of geometry-optimization options available in *eLBOW*. The default is to perform a geometry optimization using a simple force field with the optimal bond lengths derived from quantum-chemical calculations. The optimal bond lengths for the simple potential are taken from calculations for all main-group atoms from hydrogen to bromine. Two-atom molecules were constructed and H atoms were added to produce single, double and triple bonds where appropriate for all atom pairs. The minimized geometries were calculated using the Hartree–Fock method with a 6-31G(d,p) standard Gaussian basis set (Hehre *et al.*, 1971). Bond lengths for atom pairs containing the main-group elements from caesium to iodine were also calculated in a similar fashion except that the basis set used was STO-3G (Hehre *et al.*, 1969). Transition-metal bond lengths were approximated using the bond lengths for M^+-OH_2 (M = Sc–Zn) from high-level quantum-mechanical calculations as described in Magnusson & Moriarty (1993). The geometry optimization method used in this *eLBOW* option is the *L-BFGS* (Liu & Nocedal, 1989; Nocedal & Wright, 1999) minimiser in the *Computational Crystallography Toolbox* (*CCTBX*; Grosse-Kunstleve *et al.*, 2002).

It can be the case that the ligand coordinates are already known with reasonable accuracy and all that is required is restraint generation based on these input coordinates. *eLBOW* can be used to perform this task by overlaying the geometry at the end of the procedure (Fig. 1). The algorithm that determines the output restraints simply uses the values from the supplied geometry rather than the optimized geometry.

Another optimization option is the use of the AM1 semi-empirical quantum-chemical method (Dewar *et al.*, 1985). Like all quantum-chemical methods, AM1 is an all-valence-electron method and thus requires that all H atoms be included. The AM1 method uses an empirical fit based on a set of 138 molecules that replaces the most computationally expensive aspects of the quantum-chemical calculation of the molecular wavefunction. Thus, the AM1 method closely approaches the Hartree–Fock method for the single-atom contributions to the

molecular wavefunction and the majority of the two-atom interactions. The remaining two-center interactions in *ab initio* quantum-chemical calculations are parametrized, while the three-center and four-center terms are set to zero. There are a number of accurate experimental data such as ionization potentials involved in the calculations and 24 parameters that are fitted to each atom type. There is also a set of parameters that are fitted to the atom-pair interactions. The single exception to this general approach is boron, for which there is a function that relies on the element type of the bonded atom.

Another optimization of the parameters used in the AM1 algorithm was recently performed against a larger group of experimental data and denoted RM1 (Rocha *et al.*, 2006). This re-parameterization was only performed for the elements H, C, N, P, S, F, Cl, Br and I. *eLBOW* automatically substitutes the newer parameters if the molecule contains only these elements. The AM1 method and in particular the improved RM1 method have been demonstrated to provide chemical accuracy for the class of molecule generally used as macro-molecular ligands (Rocha *et al.*, 2006).

The AM1 minimization engine is a quasi-Newton procedure that uses a quadratic approximation to the potential energy hyper-surface, using a redundant internal coordinate set (Pulay & Fogarasi, 1992; Peng *et al.*, 1996; Fogarasi *et al.*, 1992; Pulay *et al.*, 1979) and a modified DIIS method (Farkas & Schlegel, 2002; Pulay, 1980, 1982; Császár & Pulay, 1984) to choose the optimal step. The step size is limited by the pseudo-linear conversion of internal coordinates to Cartesian co-ordinates and the trust radius method. The optimization is conditioned with an approximate Hessian matrix derived from the updates from each previous minimization step using the *BFGS* update algorithm (Broyden, 1970; Fletcher, 1970; Goldfarb, 1970; Shanno, 1970). Close to the minimum the method uses a simple Newton–Raphson approach in place of the DIIS method.

It is important to note that even though the AM1 method is a parametrized method, the internal coordinates are not parametrized. The geometry is minimized using the energy calculated as a function of the interactions of the nuclei and electrons of the molecule. The quantum-chemical calculation removes the limitation of force-field optimization that arises from the uniform treatment of all bonds of a certain type. The local environment of an atom or bond in a molecule influences many geometric features in the molecule and the AM1 method can include these small changes in the optimized geometry.

2.9. Restraint generation

The CCP4 monomer library format is used by *eLBOW* to describe the geometry restraints. Detailed topology information is required to determine how restraints are produced to ensure that the ligand retains a chemically meaningful geometry during refinement. For each of the restraint types an ideal value is required. These are taken from the internal optimized molecular geometry or a user-supplied geometry as mentioned in the previous section. For bonds and angles it is a

straightforward procedure to generate the restraints from the current structure.

The definition of dihedral restraints requires assignment of the period of the potential function and appropriate consideration of the interplay with planar restraints. The former is handled in *eLBOW* in two ways. The hybridization of an atom in the dihedral can limit the choice of the period. If the choice is not clear using empirical rules it is determined by rotating the molecule around rotatable bonds and using the simple force-field potential to calculate the period. The coupling of dihedrals and planar restraints requires that any dihedral in a planar group must have a period of zero. By convention, the labels of the dihedrals with a period of zero begin with 'CONST_' and the other dihedrals are labeled 'Var_XX'.

Planar restraints are generated for a number of chemical entities: aromatic rings, peptide and amide groups as well as any group of atoms surrounding a double bond. At this stage of the process the chemical information rather than the Cartesian coordinates determines planar restraints. The e.s.d. values are taken from typical examples, with atoms in the radial positions having larger values.

Chiral restraints are written using the three labels, 'positiv', 'negativ' or 'both', to denote the sign of the chiral volume. If the chiral value is not specified by the input format, 'both' is used to allow either sign. The user can request that a restraints file be written for each possible combination of chiral values at each chiral center.

2.10. Input flexibility

eLBOW can combine various chemical inputs to obtain all of the information required to build a useful restraint set. Because there are richer chemical formats for bonding information, the user can provide additional input files to specify the other required information. For example, combining a PDB file containing the desired geometry with a SMILES string will ensure both accurate coordinates and an accurate chemical topology. The two chemical inputs are matched using simple graph-matching procedures, thus removing the need to align the input order of the atoms in each input file.

Graph matching is also important in atom naming. Structure refinement requires that the atom names in the PDB file match the names in the geometry restraints definition for lookup purposes. To minimize the need for file editing, *eLBOW* can overlay an input file containing atom names (PDB, CIF) and transfer the atom naming.

2.11. Outputs

Once the molecular geometry has been minimized, the outputs, including a restraint file in CIF format and a PDB file, are written as described previously. The *eLBOW* PDB output uses a CONECT record convention to retain the bond orders. A single bond is output in the usual manner: a CONECT record for each atom with a list of bonded atoms. An atom with a double bond has the CONECT record lines duplicated for the bond in question. Other output formats are available, including Mol2, XYZ and PDB HET dictionary formats.

eLBOW is integrated with the *PHENIX* refinement package (*phenix.refine*) to handle covalently bound ligands. The input required for this mode is a PDB file minimally containing the ligand, the residue bound to the ligand and a CONECT record linking the two bound atoms. The CIF link and atom selections required are output to a file for subsequent refinement.

Appropriate output files can also be generated for metal coordination in refinement. A related feature converts any LINK records in the PDB file to a format compatible with *phenix.refine*.

2.12. Implementation details

As with most of the other components of *PHENIX*, *eLBOW* is written in Python (http://python.org) with the computationally intensive portions written in C++ and linked together using Boost.Python (Abrahams & Grosse-Kunstleve, 2003). This enables, for example, the AM1 code to be written in a compiled language (C++) while enabling accelerated development of higher level code using Python. As an example of the timings, a run without AM1 optimization takes about 35 s when calculating the restraints for ATP on a 2.9 GHz Xeon. For an AM1 optimization, an energy and gradient calculation takes 9 s, with the total time depending on the number of steps taken. Typically, ATP takes less then 200 s in total.

The inclusion of the AM1 code in the *eLBOW* package enables the user to generate a minimized geometry and a restraints file for a large class of molecules. The AM1 code is based on an open-source quantum-chemical package called *PyQuante* (Muller, 2005), which is included in the *eLBOW* distribution. *PyQuante* also uses Python as a scripting language and its computationally intensive portions are written in the C programming language.

Until recently, a limitation of the AM1 method was the small set of atoms that could be included in a calculation. The scope of atoms was increased when a systematic and comprehensive extension of the AM1 parameters to all main-group elements was performed by the original author (Stewart, 2004). This expanded set of atoms is available in *eLBOW*.

For compounds that contain atoms that are not parametrized by the AM1 method, *eLBOW* can call external quantum-chemical programs including *GAMESS* (Schmidt *et al.*, 1993), *MOPAC*7 (J. J. P. Stewart; http://openmopac.net), *Gaussian* (Frisch *et al.*, 2004) and *QChem* (Shao *et al.*, 2007) if they are available. Other geometry-minimization packages can be added to *eLBOW* in a modular fashion or from the command line using user-defined scripts. The external quantum-chemical calculation packages can be used to minimize the geometry with very high accuracy at the cost of increased computational expense. It is noteworthy that a far more computationally intensive quantum-chemical method is typically required to provide accurate geometries for transition-metal compounds.

3. Results

One of the main goals of *eLBOW* is to provide restraints for novel ligands in the structure refinement of macromolecules. Testing of the restraints generated by *eLBOW* was carried out in the following manner. The Protein Data Bank was searched for entries with a high-resolution data limit better than 1.2 Å containing at least one ligand which also have experimental diffraction data deposited. This resulted in 177 entities. Each of the ligands in the entries was processed by *eLBOW* using the AM1 optimization method to generate a set of restraints. The macromolecule and ligand were refined with the default settings of *phenix.refine* using the deposited model and data. This resulted in a ligand geometry that relied on the reflection data and the restraints generated by *eLBOW*. To further test the restraints, the refinement was repeated with reflection data truncated to a high-resolution limit of 2.5 Å.

Comparisons between the deposited ligand geometry and the newly refined geometries were performed using electron-density correlations and the r.m.s.d. of the Cartesian coordinates and the bond lengths. Each of the refined geometries in the present work was compared with the deposited PDB geometry. The comparison between the deposited and the refined geometries is important to note since it is different 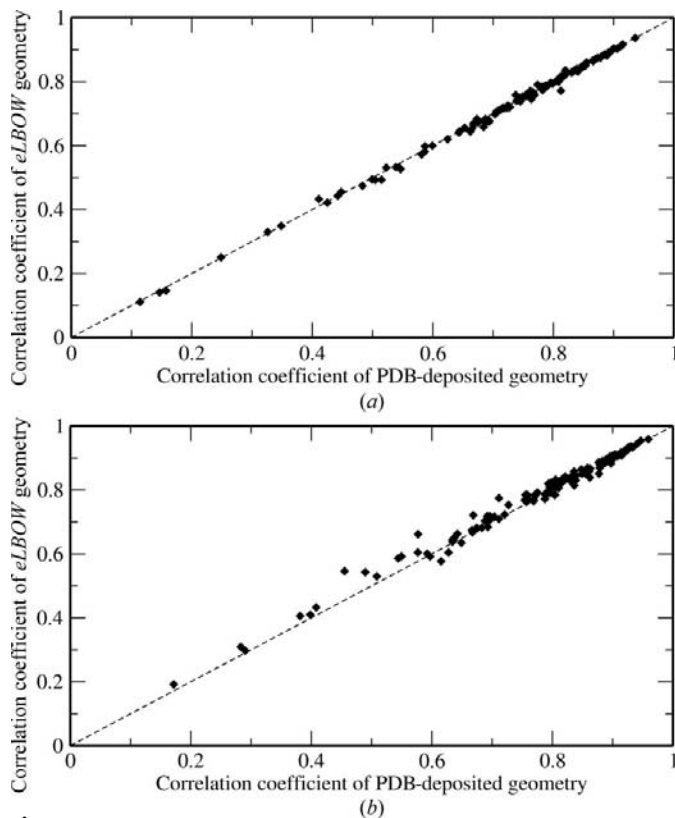from the usual geometry r.m.s.d. values quoted, which are a comparison between the refined geometry and the library used to perform the refinement. Using the high-resolution map for each ligand, the electron-density correlation coefficient (EDCC) of the deposited geometry and the EDCC of the refined ligand geometry were calculated and plotted (Fig. 2). The EDCC was calculated using maps generated by *phenix.refine* (Afonine *et al.*, 2005), determining the list of grid points within a sphere of each atomic center and using the standard correlation method. The ligand EDCC is plotted as a function of the PDB EDCC, so if the EDCC of the ligand refined using the *eLBOW* restraints is better than the deposited geometry then the data point is above the diagonal line. The same procedure was repeated for the low-resolution refinement results and plotted (Fig. 2).

The EDCCs in the high-resolution plot are very similar. This is expected because the high-resolution data determine the positions of the ligand atoms very well. In contrast, in the low-resolution refinements the restraints have more weight. In the majority of the cases the EDCC after refinement with *eLBOW*-generated restraints is improved compared with the EDCC obtained with the deposited coordinates.

The geometry of each ligand was compared with the deposited geometry by calculating the r.m.s.d. of the corresponding bonds. The r.m.s.d. for the bonds was 0.036 and 0.031 Å for the low and high-resolution refinements, respectively. The corresponding values for the absolute geometry position are 0.214 and 0.087 Å, respectively. The bond r.m.s.d. values are comparable to the accuracy of the AM1 method optimizations.

Figure 2
Electron-density correlation coefficients (EDCCs) of a number of ligand geometries refined using restraints generated by *eLBOW* plotted as a function of the EDCC for the ligand geometry as deposited in the Protein Data Bank. The graph in (*a*) corresponds to the maps using the deposited resolution, all of which are better than 1.2 Å. The graph in (*b*) shows the results of the same calculations with the resolution truncated at 2.5 Å.

4. Discussion

The *eLBOW* package is able to generate geometry-restraint information for novel ligands from a variety of chemical input formats. This includes molecules containing any main-group elements by using the built-in semi-empirical AM1 optimization and can be extended to the transition metals *via* an external quantum-chemical geometry-optimization package or using a given geometry without further geometry optimization. The ability to process novel molecules containing all the main-group elements distinguishes it from other ligand-restraint generators currently available.

The flexibility of the package allows the use of *eLBOW via* command line or Python scripts and provides many methods for manipulation of the chemical information after the initial automated analysis. The command-line interface is particularly useful in the high-throughput situations common in academia and industry.

The restraints automatically generated by *eLBOW* typically have a positive impact in structure refinement, for example in *phenix.refine*. Performing an AM1 geometry optimization provides a better r.m.s.d. between the refined geometry and the geometry deposited in the Protein Data Bank than using simple geometry optimization based on force-field restraints.

research papers

4.1. Availability and documentation

The program *eLBOW* is freely available to academic institutions as sources and precompiled binaries from http://www.phenix-online.org as part of the *PHENIX* suite.

References

Abrahams, D. & Grosse-Kunstleve, R. W. (2003). *C/C++ Users J.* **21**, 29–36.

Adams, P. D., Grosse-Kunstleve, R. W., Hung, L.-W., Ioerger, T. R., McCoy, A. J., Moriarty, N. W., Read, R. J., Sacchettini, J. C., Sauter, N. K. & Terwilliger, T. C. (2002). *Acta Cryst.* D**58**, 1948–1954.

Afonine, P. V., Grosse-Kunstleve, R. W. & Adams, P. D. (2005). *Acta Cryst.* D**61**, 850–855.

Broyden, C. G. (1970). *J. Inst. Math. Appl.* **6**, 76–90.

Császár, P. & Pulay, P. (1984). *J. Mol. Struct. (Theochem)*, **114**, 31–34.

Davis, A. M., Teague, S. J. & Kleywegt, G. J. (2003). *Angew. Chem. Int. Ed.* **42**, 2718–2736.

Davis, I. W., Murray, L. W., Richardson, J. S. & Richardson, D. C. (2004). *Nucleic Acids Res.* **32**, W615–W619.

Dewar, M. J. S., Zoebisch, E. G., Healy, E. F. & Stewart, J. J. P. (1985). *J. Am. Chem. Soc.* **107**, 3902–3909.

Farkas, Ö. & Schlegel, H. B. (2002). *Phys. Chem. Chem. Phys.* **4**, 4–10.

Fletcher, R. (1970). *Comput. J.* **13**, 317–322.

Fogarasi, G., Zhou, X., Taylor, P. W. & Pulay, P. (1992). *J. Am. Chem. Soc.* **114**, 8191–8201.

Frisch, M. J. *et al.* (2004). *Gaussian*03. Gaussian Inc., Wallingford, Connecticut, USA.

Goldfarb, D. (1970). *Math. Comput.* **24**, 23–26.

Grosse-Kunstleve, R. W., Sauter, N. K., Moriarty, N. W. & Adams, P. D. (2002). *J. Appl. Cryst.* **35**, 126–136.

Hehre, W. J., Ditchfield, R. & Pople, J. A. (1971). *J. Chem. Phys.* **56**, 2257–2261.

Hehre, W. J., Stewart, R. F. & Pople, J. A. (1969). *J. Chem. Phys.* **51**, 2657–2664.

Hückel, E. (1931*a*). *Z. Phys.* **70**, 204–286.

Hückel, E. (1931*b*). *Z. Phys.* **72**, 310–337.

Hückel, E. (1938). *Grundzüge der Theorie ungesättiger und aromatischer Verbindungen.* Berlin: Verlag Chimie.

Kleywegt, G. J., Henrick, K., Dodson, E. J. & van Aalten, D. M. F. (2003). *Structure*, **11**, 1051–1059.

Lindahl, E., Hess, B. & van der Spoel, D. (2001). *J. Mol. Model.* **7**, 306–317.

Liu, D. C. & Nocedal, J. (1989). *Math. Program.* **45**, 503–528.

Lovell, S. C., Davis, I. W., Arendall, W. B. III, de Bakker, P. I. W., Word, J. M., Prisant, M. G., Richardson, J. S. & Richardson, D. C. (2003). *Proteins*, **50**, 437–450.

Magnusson, E. & Moriarty, N. W. (1993). *J. Comput. Chem.* **14**, 961–969.

Muller, R. (2005). *PyQuante*. http://pyquante.sourceforge.net.

Nocedal, J. & Wright, S. J. (1999). *Numerical Optimization.* New York: Springer.

Peng, C. Y., Ayala, P. Y., Schlegel, H. B. & Frisch, M. J. (1996). *J. Comput. Chem.* **17**, 49–56.

Pulay, P. (1980). *Chem. Phys. Lett.* **73**, 393–398.

Pulay, P. (1982). *J. Comput. Chem.* **3**, 556–560.

Pulay, P. & Fogarasi, G. (1992). *J. Chem. Phys.* **96**, 4–8.

Pulay, P., Fogarasi, G., Pang, F. & Boggs, J. E. (1979). *J. Am. Chem. Soc.* **101**, 2550–2560.

Rocha, G. B., Freire, R. O., Simas, A. M. & Stewart, J. J. P. (2006). *J. Comput. Chem.* **27**, 1101–1111.

Schmidt, M. W., Baldridge, K. K., Boatz, J. A., Elbert, S. T., Gordon, M. S., Jensen, J. H., Koseki, S., Matsunaga, N., Nguyen, K. A., Su, S. J., Windus, T. L., Dupuis, M. & Montgomery, J. A. (1993). *J. Comput. Chem.* **14**, 1347–1363.

Schüttelkopf, A. W. & van Aalten, D. M. F. (2004). *Acta Cryst.* D**60**, 1355–1363.

Shanno, D. F. (1970). *Math. Comput.* **24**, 647–656.

Shao, Y. *et al.* (2007). *Q-Chem*, v.3.1. http://www.q-chem.com.

Stewart, J. J. P. (2004). *J. Mol. Model.* **10**, 155–164.

Weininger, D. (1988). *J. Chem. Inf. Comput. Sci.* **28**, 31–36.

Wlodek, S., Skillman, A. G. & Nicholls, A. (2006). *Acta Cryst.* D**62**, 741–749.