Journal of
Applied Crystallography
ISSN 0021-8898

computer program abstracts

This category provides a rapid means of communicating up-to-date information concerning both new programs or systems and significant updates to existing ones. Submissions should follow the standard format given in *J. Appl. Cryst.* (1985). **18**, 189–190, also available from **Crystallography Journals Online** at http://journals.iucr.org/j/services/authorservices.html.

FINDSYM: program for identifying the space-group symmetry of a crystal

Harold T. Stokes and Dorian M. Hatch*

Department of Physics and Astronomy, Brigham Young University, Provo, Utah 84602, USA. Correspondence e-mail: hatchd@byu.edu

Received 15 September 2004
Accepted 29 November 2004

Keywords: space group symmetry; standard setting; computer programs; Wyckoff positions

1. The crystallographic problem

Given the lattice parameters of a crystal and the positions of atoms in the unit cell, it is often useful to identify the space-group symmetry of the crystal in a standard setting along with the Wyckoff positions to which the atoms belong. This can be a very difficult problem to do by hand, especially if the unit cell is given in a non-standard setting or if there are many atoms in the unit cell. The aim of this program is to identify the space-group symmetry and give the lattice parameters and Wyckoff positions of the atoms in a standard setting, no matter what setting the original information is given in.

We have compared *FINDSYM* with other programs that address the same general problem.

MISSYM (Le Page, 1987) and its implementation as *ADDSYM* in the program *PLATON* (Spek, 2003) searches for possibly missed additional symmetry in a given coordinate set. Two examples of its use are given at the internet site for *ADDSYM* (1998). *FINDSYM* failed in both cases to find the additonal symmetry found by *ADDSYM*. *FINDSYM* requires that all atomic positions be reasonably close to their idealized positions in the space-group symmetry we are seeking.

TRS, *RGS* and *SFND* implemented in the program *KPLOT* (Hannemann *et al.*, 1998; Hundt *et al.*, 1999) use algorithms very similar to those in *FINDSYM*. Two examples of its function are given by Hannemann *et al.* (1998) and two more by Hundt *et al.* (1999). Each of these four examples consists of structures found during the investigation of the energy landscape of a system using global optimization techniques. In each of the four examples, *FINDSYM* successfully found the same space-group symmetry as that found by the *KPLOT* routines (this required using a tolerance of 0.03–0.06 Å).

FINDSYM was not developed for the purpose of extracting a space-group symmetry from noisey data, although it can accomplish this if the data are not too noisey, as illustrated by its success with the *KPLOT* examples. Its limitations in this regard are illustrated by its failure with the *ADDSYM* examples. The main strength of *FINDSYM* is its ease of use. It can be run over the internet by filling out a simple and user-friendly html form.

2. Method of solution

The program first finds the primitive unit cell by looking for additional lattice vectors within the unit cell given in the input. It chooses one of the atoms and tests each vector from that atom to every other atom of the same type in the unit cell. If that vector will take us from each atom in the unit cell to another atom of the same type (not necessarily in the same unit cell), then we have found a new lattice vector. This process is repeated until we have found a complete list of lattice vectors. From that list, we find three which can serve as primitive basis vectors $\mathbf{t}_i$ of the lattice, *i.e.* every lattice vector can be expressed as an integer linear combination of these basis vectors.

We next find the point group of the lattice. We make a list of all lattice vectors with lengths less than or equal to the longest basis vector. From this list, we try every possible mapping of the three basis vectors $\mathbf{t}_i$ onto three lattice vectors $\mathbf{v}_i$ in the list. We require that $\mathbf{v}_i \cdot \mathbf{v}_j = \mathbf{t}_i \cdot \mathbf{t}_j$ for every i, j. This mapping is represented by a three-dimensional matrix M of integers. We can identify the type of point operator (twofold rotation, reflection, *etc.*) from the determinant and trace of M. Every mapping found represents one of the operators in the point group of the lattice.

We next find the representative space-group operators of the crystal. Each operator consists of a point operation followed by a translation. We try each point operator in the point group of the lattice. We operate on an atom and then try every translation that brings us to an atom of the same type in the unit cell. We test each space-group operator to see if it will bring every atom in the unit cell to some other atom of the same type (not necessarily in the same unit cell).

We finally identify the space group and its standard setting in the *International Tables of Crystallography* (1995, Vol. A, 3rd ed.) using an algorithm similar to that described by Hatch & Stokes (1985). From the previous step, we have obtained a collection of space-group operators, each one consisting of a point operator (represented by a matrix M_i) and a translation $\mathbf{v}_i$ in terms of the basis vectors $\mathbf{t}_1, \mathbf{t}_2, \mathbf{t}_3$ of the primitive lattice. From the number of each type of point operator, we identify the crystal class. Finding the standard setting is a four-step process.

(*a*) Choose one of the 230 space groups G^m with the same crystal class as our unknown space group. Each space-group operator in G^m consists of a point operator (represented by a matrix $M_i^{(m)}$ of integers) and a translation $\mathbf{v}_i^{(m)}$ in terms of the basis vectors of its primitive lattice.

(*b*) Map each point operator in our space group onto one of the point operators in G^m (of the same type) so that they obey the same multiplication table. If none of the possible mappings is successful in (*c*) and (*d*), try another space group in (*a*).

(*c*) Find a lattice transformation S (a matrix consisting of all integers and with determinant 1) which obeys $SM_i^{(m)} = M_iS$ for every point operator in the crystal class. If unsuccessful, try another mapping in (*b*).

(*d*) Find a change in origin $\boldsymbol{\tau}$ which obeys $S\mathbf{v}_i - \boldsymbol{\tau} + M_i\boldsymbol{\tau} = \mathbf{v}_i^{(m)} + \mathbf{t}^{(m)}$, where $\mathbf{t}^{(m)}$ is any lattice vector of $G^{(m)}$. If unsuccessful, try another solution for S in (*c*).

 doi:10.1107/S0021889804031528

3. Software environment

FINDSYM is written in Fortran 90 and has been implemented under the Linux operating system.

4. Program specification

Input: (i) lattice parameters (a, b, c, α, β, γ) describing a unit cell of the crystal; (ii) any known centering of the lattice; (iii) number of atoms in the unit cell; (iv) type of each atom; (v) dimensionless coordinates of each atom; (vi) tolerance, indicating the accuracy of the lattice parameters and atomic positions; (vii) space-group setting desired.

Output: (i) space-group symmetry (1–230); (ii) lattice parameters of the space group in the standard setting given in *International Tables of Crystallography* (1995, Vol. A, 3rd ed.); (iii) Wyckoff positions of the atoms in this standard setting.

5. Documentation and availability

The program can be run over the internet using any computer with an internet browser. The URL of the program is http://stokes.byu.edu/findsym.html. Running the program requires no local installation. Documentation (including examples) is available at this URL. A Linux executable file can also be downloaded as part of the *ISOTROPY* software package at http://stokes.byu.edu/isotropy.html and run on any Linux operating system.

References

ADDSYM (1998). http://www.csb.yale.edu/userguides/graphics/pluton/html/pl000401.html.
Hannemann, A., Hundt, R., Schön, J. C. & Jansen, M. (1998). *J. Appl. Cryst.* **31**, 922–928.
Hatch, D. M. & Stokes, H. T. (1985). *Phys. Rev. B*, **31**, 2908–2917.
Hundt, R., Schön, J. C., Hannemann, A. & Jansen, M. (1999). *J. Appl. Cryst.* **32**, 413–416.
Le Page, Y. (1987). *J. Appl. Cryst.* **20**, 264–269.
Spek, A. L. (2003). *J. Appl. Cryst.* **36**, 7–13.