COSET: a program for deriving and testing merohedral and pseudo-merohedral twin laws

Paul D. Boyle


Many research topics in condensed matter research, materials science and the life sciences make use of crystallographic methods to study crystalline and non-crystalline matter with neutrons, X-rays and electrons. Articles published in the Journal of Applied Crystallography focus on these methods and their use in identifying structural and diffusion-controlled phase transformations, structure-property relationships, structural changes of defects, interfaces and surfaces, etc. Developments of instrumentation and crystallographic apparatus, theory and interpretation, numerical analysis and other related subjects are also covered. The journal is the primary place where crystallographic computer program information is published.
**COSET: a program for deriving and testing merohedral and pseudo-merohedral twin laws**

Paul D. Boyle

Department of Chemistry, University of Western Ontario, 1151 Richmond Street, London, Ontario, Canada N6A 5B7. Correspondence e-mail: pboyle@uwo.ca

**COSET** is a program written in ISO C99 with POSIX extensions which uses left coset decompositions to determine possible merohedral and pseudo-merohedral twin laws. In addition to a stand-alone program, the code may be compiled as a Python extension module. The program can create **SHELXL** instruction files which incorporate the appropriate TWIN and BASF instructions for the possible twin law(s). **COSET** may also be directed to execute a locally installed copy of the **SHELXL** binary executable to test the candidate twin laws in trial refinements. This facilitates the quick screening and assessment of possible twin laws.

1. Introduction

The widespread deployment of CCD and other area-detector systems has greatly facilitated the recognition and analysis of twinned structures. Derivation of the twin law is an essential step in the process. The complexity of this task can depend on the type of twin and, possibly, on the point group of the crystal. In recent years a number of software tools have been developed to facilitate the determination of twin laws. There have been several programs for indexing non-merohedral twins, such as **DIRAX** (Duisenberg, 1992), **GEMINI** (Sparks, 1999) and **CELL_NOW** (Sheldrick, 2004). There are also programs such as **ROTX** (Cooper et al., 2002) and the routine **TwinRotMat** in **PLATON** (Spek, 2009) which use differences between $F^2$ and $F^2$ values to detect previously unnoticed non-merohedral twinning. Other programs such as **XPREP** (Sheldrick, 2001) and a recent spreadsheet application (Flack & Wörrle, 2013) can be used to derive merohedral twin laws.

Software for the convenient derivation of twin laws involving pseudo-merohedry is less well represented. Local versions of the **XRAY76** system (Stewart et al., 1976) implemented a left coset decomposition algorithm several decades ago (Flack, 1987), but this program package, renamed the 'Gnu Xtal' system, does not seem to have been actively developed since 2003 (http://sourceforge.net/projects/xtal/files/). **TWINLAWS** (Schlessman & Litvin, 1995) and the coset decomposition routine on the Bilbao crystallographic server (Aroyo, Kirov et al., 2006; Aroyo, Perez-Mato et al., 2006; Aroyo et al., 2011; http://www.cryst.ehu.es/) are two other programs that can be used to derive twin laws for twinning by merohedry or pseudo-merohedry. The present article reports **COSET**, a small stand-alone program that can be used for deriving and testing twin laws in cases of twinning by merohedry or pseudo-merohedry. **COSET** implements the left coset decomposition algorithms discussed by Flack (1987) and can be used in conjunction with **SHELXL** (Sheldrick, 2008) to test candidate twin laws.

For twinning by pseudo-merohedry, the twin law is calculated by (Sands, 1982a)

$$ T = M^{-1} S M, \quad (1) $$

where $T$ is the twin law, $S$ is the symmetry operator of a metrically available apparent higher-symmetry lattice and $M$ is the transformation matrix that transforms the lower-symmetry lattice basis vectors to the metrically available higher-symmetry basis. While manual multiplication of $3 \times 3$ matrices is not difficult, it can be tedious and susceptible to error, especially when there are multiple possible twin laws to be considered. In addition, manually transforming the symmetry matrices to a common basis adds another opportunity for error to be introduced. **COSET** automates this process of deriving and testing twin laws.

The user can specify which Flack left coset decomposition algorithm is to be used. The program may be used in a strictly geometric mode, which gives all possible twin laws for a given lower-symmetry point group with respect to an apparent higher-symmetry point group. Alternatively, the program may be run in trial mode when crystallographic intensity data are available. In trial mode, the program tests each twin law by modifying an existing **SHELXL** input file specified by the user. The program inserts the appropriate TWIN and BASF statements into the .ins file which are needed to test the candidate twin laws. Either these **SHELXL** input files may be used in individual refinements after the **COSET** program exits, or the program can be directed to execute a series of **SHELXL** trial refinements automatically. A trial refinement containing an identity twin law (i.e. no twin law) is run simultaneously as a control refinement, against which the other refinements may be directly compared.

2. Usage

2.1. Installation

The source code was designed to be portable and is written mostly in ISO C99, along with POSIX.1-2008 extensions for the fork() system call and the exec() family of functions. On a reasonably modern Linux or Unix system, the program will most likely compile with only the following commands typed on the shell’s command line:

```
make
make install
```

The program successfully compiles and runs on the 32-bit version of Windows 7 using the MinGW port of the GNU C compiler gcc (http://www.mingw.org), as well as on Mac OS X 10.6.8 using gcc-4.2.1. Details for compiling on different platforms or as a Python extension module are covered in the README.txt file included with the source code distribution.

© 2014 International Union of Crystallography

2.2. Input file

The COSET program is designed both to be easy to use and to require a minimum of input. No assumption need be made with regard to the settings of the actual point group of the structure. Typing the program name on the command line without an input file name returns a help message. The input file is a plain text file which contains a number of directives and parameters used to govern the execution of the program. The program can process multiple coset analyses in a given execution, which are designated as ‘tasks’; the program is not limited in the number of tasks in a given execution. The directives and the parameter requirements are given in Table 1.

A ‘&’ character at the beginning of a line denotes a comment and is ignored by the program. The directives are not case sensitive and are explained below:

TITLe must be the first directive for a given task. The parameter for this directive is a short description of less than 75 characters. ALGORITHM allowed parameters are ‘A’ or ‘B’ (without quotes). SUPERGROUP represents the metricaly available higher symmetry, and this directive must contain one of the following character strings: -1, 2/m, mmm, 4/mmm, -3m, 6/mmm, m-3m.

SUBGROUP describes the lower (true) point symmetry of the crystal and takes two parameters. The first is a character string designation for the crystal’s point group, e.g. -3 or mm2. The second parameter is an integer, which is equal to the number of symmetry operators for the crystal’s point group, including the identity operator. The SUBGROUP directive must precede any RMAT directives.

RMAT takes nine numerical parameters, which are the matrix elements for the symmetry operators of the point group. The order of the matrix elements is \( r_1, r_2, r_3, r_{21}, r_{22}, r_{31}, r_{32}, r_{33} \). Each symmetry operator takes a separate RMAT directive. The number of RMAT directives must equal the numerical value given in the SUBGROUP directive. When constructing the RMAT directives, simply take the equivalent positions given in International Tables for Crystallography,Vol. A (Hahn, 2002), and convert them by inspection to the local system’s point symmetry and the identity operator. For example, if the crystal’s space group is \( P4/m \), drop the lattice-centring symbol and convert the translational symmetry elements to the corresponding non-translational equivalents. Thus, in this case, one would use the equivalent positions for \( P4/mb \). The identity operator must be included and must be the first RMAT statement in the input file.

TRANS takes nine numerical elements which transform the unit-cell parameters of the crystal to the metrically available supergroup cell. The order of the elements is \( t_{11}, t_{12}, t_{13}, t_{21}, t_{22}, t_{23}, t_{31}, t_{32}, t_{33} \). These elements are normally obtained from a cell-reduction program. If TRANS is omitted the identity matrix is used.

INSFILE takes a single character string, which is the name of the SHELXL .ins file for the structure. This file is not altered by the program but provides the basis for new .ins file(s) used to perform trial refinements.

OUTFILE takes a single character string, which is the name of a general output from COSET. If this directive is not specified, the program writes these results to stdout (‘standard output’, i.e. the terminal).

NEWINS takes a single character string, which is the base name for the new set of .ins files that incorporate the SHELXL BASF and TWIN instructions for twinned refinement.

EXEC takes a single character string, which is the full pathname of the local system’s SHELXL(T) executable. With UNIX systems, symbolic links are created to the actual .hkl file. With systems in which symbolic links are unavailable, the .hkl file is copied to each <new basename>.hkl file. The output SHELXL normally sends to the terminal is trapped into a so-called ‘screen’ file for convenient post-execution examination by the user.

END takes no parameters and should be the last line of the file. Examples of COSET input files are given in §4. It should be noted that not all directives need to be used. If crystallographic data or SHELXL files are not available, a coset decomposition can still be performed with only the TITLe, SUPERGROUP, SUBGROUP, RMAT and, if necessary, TRANS directives. This will give potential merohedral or pseudo-merohedral twin laws for any subgroup-supergroup relationship.

Table 1

<table>
<thead>
<tr>
<th>Directive</th>
<th>Parameters</th>
<th>Requisite</th>
</tr>
</thead>
<tbody>
<tr>
<td>TITLe</td>
<td>Character string</td>
<td>Mandatory and must be the first directive of a task</td>
</tr>
<tr>
<td>ALGORITHM</td>
<td>Single character</td>
<td>Mandatory</td>
</tr>
<tr>
<td>SUPERGROUP</td>
<td>Character string</td>
<td>Mandatory</td>
</tr>
<tr>
<td>SUBGROUP</td>
<td>Character string and integer</td>
<td>Mandatory, must precede RMAT</td>
</tr>
<tr>
<td>RMAT</td>
<td>Nine numerical elements</td>
<td>Mandatory</td>
</tr>
<tr>
<td>TRANS</td>
<td>Nine numerical elements</td>
<td>Optional</td>
</tr>
<tr>
<td>OUTFILE</td>
<td>Character string</td>
<td>Optional but requires TRANS</td>
</tr>
<tr>
<td>NEWINS</td>
<td>Character string</td>
<td>Optional but requires TRANS and INSFILE</td>
</tr>
<tr>
<td>EXEC</td>
<td>Character string</td>
<td>Optional and must be TRANS and NEWINS</td>
</tr>
<tr>
<td>END</td>
<td>None</td>
<td>Must be last</td>
</tr>
</tbody>
</table>

3. Implementation

COSET is a ‘free software’ (http://www.gnu.org/philosophy/free-sw.html) program released under the GNU General Public License (http://www.gnu.org/licenses/gpl.html). It is hoped that users will modify and enhance the program. For this reason, some details regarding the implementation of the program are given.

3.1. Input

The input for the program is read using a finite state machine (FSM). The FSM was chosen for its flexibility and modularity. There are very few requirements regarding the order of the directives in the input file. In addition, if new directives are to be added, it is a simple matter to include the directive and its callback function to the struct keyword_table table[]. array declaration in input.c. The struct keyword_table is defined in input.h. The body of the callback function should be located in the input.c file for consistency. Depending on the nature of the new directive, the struct task in task.h may also need to be modified.

3.2. Representation of symmetry matrices

The code regarding the storing and manipulation of symmetry matrices is given in the files symm_mat.h and symm_mat.c. The key data structure for the symmetry matrices is the C struct symm_op, which is declared as

```c
struct symm_op {
    bool truefalse;
    double mat[3][3];
    unsigned int bcm;
    int n_fold;
    float rotation_angle;
};
```

3.4. Creating the SHELXL TWIN and BASF instructions

The code for setting up the TWIN and BASF instructions is found in the source file shelx.c. For an n-fold axis as a candidate twin law, the program assumes n twin domains and sets the tenth parameter of the TWIN instruction to this value. The BASF instruction incorporates n − 1 fractional volume parameters. Each of these is set to a starting value of 1.0/n.

4. Case studies

COSET has been used to derive twin laws for pseudo-merohedral twins as illustrated by two case studies, 4-(1-allyl-4,5-diphenyl-1H-imidazol-2-yl)-N,N-dimethylaniline (Akkurt et al., 2013) and 2,2’-(piperazine-1,4-diyl)diethanamines dibenzoate (Cukrowski et al., 2012). In each case, the CIF and the reflection data were downloaded from the publisher’s web site (respectively, http://dx.doi.org/10.1107/S1600536813006326 and http://dx.doi.org/10.1107/S1600536812030115) and edited into a form suitable for SHELXL refinement. For the coset analysis, an appropriate COSET input file and SHELXL .ins file were constructed. The .ins file included all non-H atoms with isotropic displacement parameters. H atoms from the original structure determination were deleted. The trial refinement was set up to include appropriate HFIX commands, so H atoms would be introduced at idealized positions and allowed to ride on the parent C or N atoms. The trial refinements were run with 12 cycles of least squares each. For the first four of these, the non-H atoms were refined isotropically, and for the last eight cycles non-H atoms were included anisotropically. The weights were set to the default SHELXL weighting scheme.

4.1. 4-(1-Allyl-4,5-diphenyl-1H-imidazol-2-yl)-N,N-dimethylaniline

This structure was reported recently (Akkurt et al., 2013) and found to be twinned by pseudo-merohedry. The authors successfully derived the twin law [001, 011] and refined the twin fraction to a value of 0.513 (3).

Cell reduction of the reported cell parameters yields a metrically available C monoclinic cell with unit-cell parameters of \(a = 11.643\), \(b = 88.685\), \(c = 9.426\) Å, \(\alpha = 90.00\), \(\beta = 123.24\), \(\gamma = 89.98\)°. The transformation matrix to the metrically available C monoclinic lattice is \([T_0, T_2, 100]\). \(R_{\text{sym}}\) for the monoclinic lattice is 0.029. Using the results from the cell reduction and the point group of the crystal allows following COSET input file to be constructed:

```plaintext
TITLE Akkurt et al. Acta Cryst (2013), E69 o527-o528
ALGORITHM A
SUPERGROUP 2/m
SUBGROUP -1 2
TRANS -1 -1 0 -1 -1 -2 1 0 0
RMAT 1 0 0 0 1 0 0 0 1
RMAT -1 0 0 0 -1 0 0 0 -1
OUTFILE COSET.OUT
INFILE aj5302.ins
NEWINS twin
EXEC /usr/local/bin/shelx
END
```

COSET performed the left coset decomposition and recovered the twin law found by Akkurt et al. (2013). The trial refinements gave a clear indication that the twin law was indeed the correct one. The weighted R factor, \(wR_2\), of the ‘no twin’ control refinement started at 0.547 and decreased slightly to 0.509 over the course of 12 cycles of refinement.
least squares. In contrast, \( wR_2 \) for the trial refinement including the twin law started at 0.370 and decreased to 0.146. Inclusion of the twin law thus gave a marked and immediate improvement to the refinement model. The twin fraction parameter, BASF, converged to a value of 0.514 (1), which compares well with the previously reported value of 0.513 (3).

4.2. 2,2’-(Piperazine-1,4-diyl)diethanaminium dibenzoate

This compound crystallizes as a monoclinic structure, which is twinned by pseudo-merohedry (Cukrowski et al., 2012). The authors report the twin law as \([010, 0\bar{T}0, 100]\) and twin fractions of 0.8645 (8) and 0.1355 (8).

Cell reduction of the reported unit-cell parameters yields a metrically available \( C \) orthorhombic cell with unit-cell parameters of \( a = 20.748, b = 33.197, c = 6.669 \) \( \text{\AA} \), \( \alpha = 90.00, \beta = 90.00, \gamma = 89.71^\circ \). The transformation matrix to the metrically available \( C \) orthorhombic lattice is \([01\bar{T}, 0\bar{T}0, 0\bar{T}0]\). \( R_{\text{sym}} \) for the orthorhombic lattice is 0.318.

Using the results from the cell reduction and the point group of the crystal, the following \( COSET \) input file was created:

```
TITLE Cukrowski, et al. (2012), Acta Cryst E68, o2389
ALGORITHM A
SUPERGROUP mm
SUBGROUP 2/m 4
TRANS 1 0 1 -1 0 1 0 -1 0
RMAT 1 0 0 0 1 1 0 0 1
RMAT -1 0 0 -1 1 0 0 -1 0
RMAT 1 0 0 -1 1 0 0 -1 0
RMAT -1 0 0 1 0 0 0 -1 0
OUTFILE COSET.OUT
INSFILE j5j2134.ins
NEWINS twin
EXEC /usr/local/bin/shelxl
END
```

In this case, performing the left coset decomposition with algorithm \( A \) finds a twin law \([001, 0\bar{T}0, 0\bar{T}0]\), which corresponds to a 180° rotation about \([\bar{T}01]\), while the twin law given by Cukrowski et al. (2012) corresponds to a 180° rotation about \([\bar{T}01]\). Using algorithm \( B \) recovers the twin law given by Cukrowski et al. The twin law obtained from algorithm \( A \) was used for the trial refinements.

Unlike the previous example, \( wR_2 \) for the trial refinement including the twin law starts at a slightly higher value than the ‘no twin’ control refinement. This is due to the disparity between the starting value of the twin fraction (0.50) and the actual value of 0.13. Nevertheless, \( wR_2 \) for the trial refinement drops smoothly to a value of 0.121 over the course of 12 cycles, while \( wR_2 \) for the control refinement remains at about 0.41 for all 12 cycles.

5. Conclusion

\( COSET \) is a small stand-alone program useful for deriving merohedral or pseudo-merohedral twin laws. It can be used in conjunction with the \( SHELXL \) refinement program to evaluate potential twin laws quickly. The input files for \( COSET \) are typically small and easily created using a text editor.

6. Download

The \( COSET \) program can be downloaded from http://xray.chem.uwo.ca/COSET/. The source code is available as either a ZIP archive or a GZIP compressed tar file. A 32-bit statically compiled binary executable file for Windows is also available on the web site. The Windows executable file was produced using MinGW’s gcc compiler on Windows 7 Professional edition.

The author is grateful to Dr Sean Parkin for helpful comments regarding the code and the manuscript, as well as for testing the program on the Mac OSX platform. The author is also grateful to Dr David Watkin and Professor Nicholas Payne for their helpful comments regarding an early draft of the manuscript.

References