

Patterson Map Solution Program

Program PATSOL

Liang Tong[†]

DESCRIPTION OF INPUT COMMANDS

All input commands to the program are keyword-based and free-formatted. The input parser converts lower case characters to upper case so the program commands are not case dependent (though the program keeps the cases of the file names and the title unchanged). Each input line can contain at most 80 characters. The following characters are recognized as delimiters between words – space, comma, tab, and equal sign. More delimiters can be implemented by inserting them in the array SPACER in subroutine PARSER (and change the variable NSP at the same time). Comments in the input can be introduced by using the COMMEnt command or using the special character “!” — in the input parser, all characters in the input line beginning at the “!” are ignored.

Presently, the program uses five logical I/O units, labelled by the variables LIN, LOG, LPRT, LMAP, and LSCRCH, which are initialized to be 5, 6, 3, 1, and 2, respectively. (The default values of most of the input variables are initialized in subroutine INITSS.) All input commands are read from unit LIN. The output of the program will be written to unit LPRT. A name can be specified for this print file by using the PRINt command. The Patterson map values (if any) are read from unit LMAP. LSCRCH is a scratch file and is used for echoing the input commands at the end of the output file. Program messages are written to unit LOG. They are informational messages, warning error messages (*Warning*), or fatal error messages (*Fatal*). Each message is preceded by the name of the subroutine that produced it.

What follows is a description in alphabetical order of all the input commands that are currently supported by the program. A new command can be incorporated by inserting the command name in array COMMND in subroutine INITSS, and by inserting a segment in subroutine INTPRT that defines the input variables (if any) of the command. The program identifies each command by its first four letters, although more can be input for readability. In the following listing, the command keyword will be given in **bold face**, the names of input variables associated with the command will be given in UPPER case. The expected length of the character variables are also given (for example, MAPTYP*5 means MAPTYP is a character variable with 5 letters). Variables whose names begin with the letter ‘Q’ are logical variables. A ‘True’ or ‘False’ input is expected for these variables. The defaults (if any) of the variables are given in square brackets. These default values will be used by the program if the command is omitted from input, or if a

[†] Please send comments and bug reports to : b4w@mace.cc.purdue.edu.

Reference: L. Tong & M. G. Rossmann *J. Appl. Cryst.* **26**, 15–21, (1993).

value of 0 is input to the command.

CELL-dimensions (CELL(i), i=1,6) [1,1,1,90,90,90]

The cell dimensions of the crystal are given by this command. They are used for packing checks in cross-vector searches (with or without non-crystallographic symmetry).

COMMENT none [none]

Comments introduced with this command will be echoed at the end of the print file. Comments can also be introduced with the special character '!'.

CROSS-vector X, Y, Z [none]

This inputs known cross-vectors between heavy atom sites. The coordinate can be either fractional or on the grid of the Patterson map. The program will carry out two-atom searches based on these input cross-vectors, using the single-sites from SINGLE. The maximum number of cross-vectors that can be input by this command is given by the MAXCRS parameter.

DBLCut DBLCUT, MINX [0.25, 1]

This inputs cut-off value for two-atom search solutions. It can be either based on the σ value of the Patterson map (in which case DBLCUT should be less than 10) or in absolute value. The program sorts all the cross-vector heights between any two sites and removes the MINX-1 lowest values. Therefore, MINX-1 specifies how many cross-vectors can be missing between any pair of sites. Setting MINX to 2 or higher might be helpful for high-symmetry space groups.

DBLFit X1, Y1, Z1, X2, Y2, Z2 [0]

The program will attempt to maximize the cross-vector heights between the two input sites. The coordinates can be either fractional or on the grid of the input Patterson map. Each site will be moved by NTOL/2 (see command TOLerance) around its input position. The maximum number of two-atom solutions that can be input by this command is given by the MAXSHO parameter.

DBLShow X1, Y1, Z1, X2, Y2, Z2 [0]

The program will produce a list of the self-vectors of and the cross-vectors between the two input sites. If a Patterson map has been input, the corresponding map values at these vectors will be output as well. It is recommended that the user should use this command at least once, for a pair of general sites, to make sure that the program is calculating the self- and the cross-vectors correctly. The maximum number of two-atom solutions that can be input by this command is given by the MAXSHO parameter.

DOUBle-search DBL [F]

MAPType

MAPTYP*5, QFORM

[HASSP]

This specifies the map format. The default format, HASSP, is a file that has been created by the FFT program in the HASSP program package (T. Terwilliger, S.-H. Kim & D. Eisenberg *Acta Cryst.* **A43**, 1–5, (1987)). The map file is expected to be a binary file with no header. Though a CAB sectioning (see command MAPSection) is required by HASSP, this program can support general sectioning for this format.

A FSFOR formatted map file (FSFOR is a program in the B.C. Wang solvent-flattening package) can be read in by specifying FSFOR for the MAPType. The FSFOUR map file (in W. Furey's Phases package) is also supported. The map grid and sectioning information (see commands MAPGrid and MAPSection) will be taken from the map header. The symmetry of the space group should still be defined by the user. FSFOR and FSFOUR maps cover the entire unit cell. The command MAPRegion should be used to instruct the program to store only the region for the asymmetric unit.

An X-Plor formatted map file can be read in by specifying XPLOR for the MAPType. The X-Plor map MUST be calculated with 'EXTEND=UNIT'. The asymmetric unit of the Patterson map should be defined with MAPRegion. For X-Plor maps, QFORM specifies whether the map is written as formatted or unformatted file. If the program issues an error message about the number of grid points along cell edges, the FFT GRID command in X-Plor should be modified so the number of grid points is acceptable by this program. In any event, FFT GRID 0.3 END is the default in X-Plor. This should preferably be changed to FFT GRID 0.2 END to get a finer sampling of the Patterson map.

Currently, CCP4 maps can be input by first converting to X-PLOR map format using MAPMAN (by Gerard Kleywegt). Make sure, however, that the resulting X-PLOR map covers the entire unit cell (0 to 1 inclusive). For example, if you have 100 grids along an axis, the map should cover grid points 0 (corresponding to fractional coordinate 0.000) through 100 (fractional coordinate 1.000), not 0 through 99, of that axis. In other words, there are 101 grid points (not 100) stored in the map along that axis.

The 3D01 map format at Purdue is no longer supported in this version.

NCRSymmetry

OPTION*4

[]

This command is used to specify the non-crystallographic symmetry information and also to define parameters that will be used in the self- or cross-vector searches with the specified local symmetry.

The program will carry out a self-vector search to locate the heavy atom position relative to the center of the local symmetry elements if QSG1 has been set to true (see command SINGle). If QDBL is true (see command DOUBle), the program will carry out a cross-vector search to locate the center of the local symmetry elements in the crystal unit cell. In both cases, the average height of the vectors in the Patterson map (or the sum function) will be used. SGLCUT or DBLCUT will be used to remove search grid points that are too low. See Appendix A for some theoretical discussions.

OPTION is a four-letter keyword that defines the action of this command. Currently-supported OPTION keywords are —

CENTER — (CENTER(i), i=1, 3)

This defines the center of the NCS assembly in the crystal unit cell. Sometimes, due to alignment of NCS axis with crystallographic axis, the center of the NCS can be determined. If this center is given, the self-vector search (see command `SINGLE`) will cover all vectors (self-vectors within the NCS assembly, and cross-vectors among the crystallographically-related assemblies). If this center is not specified (which is the default), the self-vector search will only cover those within the NCS assembly.

CHECK-all — **QCKALL**

This defines whether during the search calculation vectors between all atoms will be used (if `QCKALL` is true) or only vectors between a reference atom and other atoms will be used (if `QCKALL` is false). If there are `NNCR` local symmetry operators, $\text{NNCR}(\text{NNCR}-1)/2$ vectors will be used if `QCKALL` is true whereas only `NNCR-1` vectors will be used if `QCKALL` is false for self-vector searches. The default for `QCKALL` is true.

CUTLow — **CUTLO**

This specifies that, at a search position, if the height of one of the vectors is less than `CUTLO`, the position will be skipped. The `CUTLO` value can be given either in terms of the σ value of the map or as an absolute value. The default for `CUTLO` is -2.

DUMP-map-value — **QDUMP**

This specifies whether the self- or cross-vector search map will be dumped to the output print file. The map will be dumped with Z across, Y down, and X sections. Default value for `QDUMP` is false.

EULER-angle-convention — **EULER*3**

This specifies the convention of Eulerian angles. See the documentation for the `GLRF` program for details.

LOCExpand — **QLCXP**

This specifies whether the local symmetry operations input with the `LOCSymmetry` option will be expanded by pairwise multiplication.

LOCOrient — **(ALCROT(i), i=1, 3), TLCROT*1**

This specifies a set of rotation angles (`ALCROT`) which will bring the standard orientation (as defined with `LOCSymmetry` option) to that of the molecular assembly in the crystal. `TLCROT` specifies whether the angles are Eulerian or polar.

LOCSymmetry — **(RLCAXS(i), i=1, 3), MLCAXS, TLCAXS*1**

This specifies the local symmetry elements in a standard orientation. Please see the documentation to the `GLRF` program for details.

MULTiplicity-correction — **QMUNCR**

This specifies whether multiplicity due to self- or cross-vector overlap will be corrected. The equation to calculate the average height of the vectors is $P = \sum_i P_i/n_i$ if `QMUNCR` is true and $P = \sum_i P_i$ if `QMUNCR` is false. The summation goes over all the unique vectors, and n_i is the number of occurrences of the vector. The default for `QMUNCR` is true.

ORTHogonalization-convention — ORDOR*5

This specifies the orthogonalization convention to convert fractional coordinates into Ångstrom coordinates, or *vice versa*. See the documentation for the GLRF program.

POLAR-angle-convention — POLAR*3

This specifies the polar angle convention. See the documentation for the GLRF program.

PXYZ-input — PX, PY, PZ

This inputs a known heavy atom position, in Cartesian Ångstrom coordinates, relative to the center of the local symmetry elements. Such a position must be defined for a cross-vector search as there are no default values.

SITE — SITEX, SITEY, SITEZ

The program will take the input coordinates as the position of the center of the particle (from cross-vector search) and calculate a set of heavy atom positions in the unit cell based on the site specified with the PXYZ option.

SIZE — SIZMOL

For self-vector searches in Cartesian coordinates (QSPHER is false), the maximum distance between any search position and the origin is limited to SIZMOL. For cross-vector searches, SIZMOL is the radius of the molecular assembly used in packing calculations. The default for SIZMOL is 25Å.

SHOW — SHOWX, SHOWY, SHOWZ

All the self- (or cross-) vectors for the site given will be listed in the output print file. If a map has been input, the vector heights will be output as well. Self-vectors will be calculated if QSG (see command SINGLe) is true, whereas cross-vectors will be calculated if QDBL (see command DOUBle) is true. The maximum number of positions that can be input for showing is given by the parameter MAXSHO.

SLIMits — IAXIS, BEGIN, END, INCMNT

This defines the limits of the self- or cross-vector search. The limits should be specified as beginning value, ending value, and increment value for the three directions. For self-vector searches, IAXIS values of 1, 2, and 3 correspond to X, Y, Z (or ϕ, ψ, R) directions if POLAR is false (or true), respectively. For cross-vector searches, IAXIS values of 1, 2, and 3 correspond to a, b, c directions, respectively, in deorthogonalized fractional units.

SPHERical-polar-search — QSPHER

This specifies whether the self-vector search will be carried out in spherical polar coordinates (ϕ, ψ, R) or Cartesian coordinates (P, Q, R). The default for QSPHER is false. If POLAR is 'XYK', ψ will be measured from the Cartesian Q axis. If POLAR is 'XZK', ψ will be measured from the Cartesian R axis. ϕ is measured from the P axis in both cases.

The cross-vector search is carried out in fractional deorthogonalized coordinates along the unit cell edges.

SYMMetry-operator — ISYMOP

This specifies which symmetry-related molecule will be used in the cross-vector search. The cross-vectors between a reference molecule and the molecule related by this crystallographic symmetry operator will be used in the cross-search. The default for ISYMOP is 2, *i.e.*, the second crystallographic symmetry operator listed by the program.

ORIGin-box (IORIGN(i), i=1, 3) [4, 4, 4]

This defines a box around each of the origins in the Patterson map and all self- or cross-vectors lying in these boxes will be ignored during the calculation. The size should be defined based on the grid of the input Patterson map. The default values specify a cube with 9 grid points on each edge centered at each of the Patterson map origins.

PATCut PATCUT [1.5]

This specifies the cut-off value for grid points to be included in the two-atom search with the command PATX. All grid points in the Patterson map with values greater than PATCUT will be used as trial cross-vectors in the search.

PATPeak-search QPAT [F]

This specifies whether Patterson map peaks are to be searched. If true, all unique peaks above DBLCUT (see command DBLCut) will be identified and listed in the output.

PATX-search QPATX [F]

This specifies whether the program will carry out two-atom searches using as cross-vectors all Patterson map grid points with value greater than PATCUT (see command PATCut). This method is similar to that in HASSP, with two differences. One is that all positions greater than a cut-off value, rather than only the peaks, are used as possible cross-vectors. The other difference is that the program will work only with the list of single-site positions with height greater than SGLCUT rather than going over all the positions in the asymmetric unit.

PRINT-file PRTFIL*60 [PATSOL.PRT]

This specifies a name for the print file for the current run.

SGLCut SGLCUT, MINHK [0.5, 1]

Similar to the command DBLCut, this specifies the single-site solution cut-offs.

SGLFit X, Y, Z []

Similar to command DBLFit, this maximizes the self-vector height of an input site.

SGLShow X, Y, Z []

The sum function (rather than the minimum function) will be used in the single and two-atom searches if QSUM has been set to true. Default for QSUM is false and minimum function will be used. Generally the minimum function is preferred.

SYMMetry []

This inputs the space group symmetry elements in the format of the International Table. The identity operation is assumed and need not be given. The lattice centering should be specified with the LATTice command.

The space group symmetry can also be specified by giving the symbol of the space group (for example P212121) or the number of the space group (for example 19). For monoclinic space groups, only the *b*-unique setting is supported. For rhombohedral space groups, only the hexagonal setting is supported.

SYMShow X, Y, Z []

The program will produce a list of the space group symmetry-related positions based on the input site. All the symmetry-related positions will be translationally reduced to the region [0, 1). The maximum number of sites is given by the MAXSHO parameter.

TITLE TITLE*132 [PROGRAM PATSOL]

This inputs a title that will be put at the top of each output page. The program will insert the current time and date between columns 2 and 20 of TITLE and a version flag at the end of TITLE.

TOLErance NTOLX, NTOLY, NTOLZ [3, 3, 3]

This specifies a tolerance limit, in grid points, so that two sites within (NTOLX, NTOLY, NTOLZ) grid points of each other will be considered to be the same.

TRIPle NTRIPL, TPLSGL, TPLDBL [10, -0.5, 0]

NTRIPL specifies that the top NTRIPL two-atom solutions (from DOUBle, PATX or CROSS) will be used to look for additional sites. During the search, positions with single-site height less than TPLSGL, or with cross-vector heights with the known positions less than TPLDBL will be ignored. The newly-found sites are sorted based on the cross-vector heights with the known solutions. Note that in this search no check is made on the cross-vector heights between the newly-found sites. The maximum number of solutions that can be saved by the program is MAXSLN.

PROGRAM STORAGE LIMITS

Parameter	Limit
MAPDIM	300,000
MAXCRS	100
MAXDBL	500
MAXNCR	60
MAXSGL	1000
MAXSHO	10
MAXSLN	10
MAXVEC	3600

Appendix A.

Heavy Atom Search with Non-Crystallographic Symmetry

Given a molecular assembly conforming to a point group symmetry, we should have the following relationships,

$$P_n^0 = [\rho_n^0]P_1^0$$

assuming the center of the point group is located at (0,0,0) of the Cartesian coordinate system. The symmetry operations of the point group are defined by the rotation matrices $[\rho_n^0]$, in a ‘standard’ orientation. P_n^0 and P_1^0 are positions that are related by the point group symmetry in this orientation.

The orientation of the molecular assembly in the crystal unit cell is related to the standard orientation by a rotation, $[E]$, which can be determined from rotation function calculations. Assuming the center of the molecular assembly is located at position s in the unit cell, we should have

$$x_n = [\alpha][E]P_n^0 + s$$

and

$$x_1 = [\alpha][E]P_1^0 + s$$

where x_1 and x_n are two positions in the molecular assembly that are related by the non-crystallographic symmetry. $[\alpha]$ is the deorthogonalization matrix that converts Ångstrom Cartesian coordinates to fractional deorthogonalized units. Therefore, the vector between positions x_n and x_1 , called a self-vector because it arises from atoms in the same molecular assembly, is given by,

$$x_n - x_1 = [\alpha][E]([\rho_n^0] - [I])P_1^0$$

where $[I]$ is the identity matrix. More generally, a self-vector can be calculated between any pairs of atoms from an assembly,

$$x_n - x_m = [\alpha][E]([\rho_n^0] - [\rho_m^0])P_1^0$$

Therefore, a self-vector search can be carried out to determine P_1^0 by maximizing the average of the self-vector heights,

$$SV = \sum_i P_i/n_i$$

where the summation goes over all the unique self-vectors. P_i is the value of the Patterson map corresponding to the vector and n_i is the number of occurrences of the unique vector (multiplicity correction).

To locate the center of the molecular assembly, s , the vectors between atoms in one assembly and those in a crystallographically-related mate, the cross-vectors, will be needed. Suppose the symmetry operator is defined by,

$$x' = [T_k]x + d_k$$

we can derive the following equation for the cross-vectors,

$$x'_n - x_m = ([T_k][\alpha][E][\rho_n^0] - [\alpha][E][\rho_m^0])P_1^0 + d_k + ([T_k] - [I])s$$

therefore, given P_1^0 , a search can be carried out for s . For example, in space group $P2_12_12_1$, three cross-vector searches can be carried out, by setting k to 2, 3, and 4, respectively. The correct particle center is the solution that is consistent with the results from these three sections.

Appendix B[†]. The Single-Site Solutions of HASSP

As has been discussed in Chapter 2, the program HASSP derives a list of single-site solutions to a difference Patterson map by a systematic search through the asymmetric unit of the space group and checking the map for the minimum value of the self-vectors for each position searched. The positions with the highest minimum values are taken as the single-site solutions to the difference Patterson map.

It is clear from this algorithm that any two sites that produce identical set of self-vectors can not be distinguished from each other, because they will all produce exactly the same minimum self-vector values. As will be shown later, a set of self-vectors can be produced by more than one atomic sites. These positions are equivalent to each other as far as the Patterson map is concerned, but they may not be equivalent to each other by the symmetry of the space group.

It can be easily shown that sites that are related by space group symmetry produce the same set of self-vectors. This does not cause any problem in interpreting the difference Patterson map, because these sites do occupy equivalent positions in the unit cell.

However, there are sites that are not equivalent by the space group symmetry but produce the same set of self-vectors. These include sites that are related to each other by alternative origin translations and sites that are related to each other by inversion through the origin of the unit cell (except when the space group is enantiomorphic, see below). For example, in space group $P2_12_12_1$, there are four symmetry operations; therefore, each site should produce three non-zero self-vectors. There are eight alternative origins in this space group, at $(0,0,0)$, $(1/2,0,0)$, $(0,1/2,0)$, $(0,0,1/2)$, $(1/2,1/2,0)$, $(1/2,0,1/2)$, $(0,1/2,1/2)$ and $(1/2,1/2,1/2)$. The following calculation is carried out for the alternative origin at $(1/2,0,0)$, although the same conclusion can be reached using any of the alternative origins. Two sites, at (x, y, z) and $(1/2+x, y, z)$, are related to each other by the alternative origin translation of $(1/2,0,0)$. As shown in the following table, these two sites produce the same set of self-vectors.

No.	Symmetry Positions	Self-vectors	Symmetry Positions	Self-vectors
1	x, y, z		$1/2+x, y, z$	
2	$1/2+x, 1/2-y, -z$	$1/2, 1/2+2y, 2z$	$x, 1/2-y, -z$	$1/2, 1/2+2y, 2z$
3	$1/2-x, -y, 1/2+z$	$1/2+2x, 2y, 1/2$	$-x, -y, 1/2+z$	$1/2+2x, 2y, 1/2$
4	$-x, 1/2+y, 1/2-z$	$2x, 1/2, 1/2+2z$	$1/2-x, 1/2+y, 1/2-z$	$2x, 1/2, 1/2+2z$

In the table, the four equivalent positions for sites (x, y, z) and $(1/2+x, y, z)$ are generated based on the space group symmetry, respectively. The three non-zero self-vectors are calculated by subtracting symmetry position 1 from positions 2, 3 and 4, respectively. Patterson map symmetry (in this case, $Pm\bar{3}m$) is applied to the resulting self-vectors to arrive at the final results shown in the table.

The following table shows that sites (x, y, z) and $(-x, -y, -z)$, which are related to each other by inversion, also produce the same set of self-vectors in space group $P2_12_12_1$.

[†] From L. Tong, Ph. D. Thesis, University of California, Berkeley, July, (1989).

No.	Symmetry Positions	Self-vectors	Symmetry Positions	Self-vectors
1	x, y, z		$-x, -y, -z$	
2	$1/2+x, 1/2-y, -z$	$1/2, 1/2+2y, 2z$	$1/2-x, 1/2+y, z$	$1/2, 1/2+2y, 2z$
3	$1/2-x, -y, 1/2+z$	$1/2+2x, 2y, 1/2$	$1/2+x, y, 1/2-z$	$1/2+2x, 2y, 1/2$
4	$-x, 1/2+y, 1/2-z$	$2x, 1/2, 1/2+2z$	$x, 1/2-y, 1/2+z$	$2x, 1/2, 1/2+2z$

Therefore, in space group $P2_12_12_1$, there are 4 (for space group equivalent positions) \times 8 (for alternative origins) \times 2 (for enantiomorphs) = 64 positions in the unit cell that produce the same set of self-vectors. Further calculation can show that one of these sites will always be located within $[1/4, 1/4, 1/4]$, which is $1/64$ of the unit cell volume. An inspection of the actual single-site solution lists produced by HASSP confirms this calculation — all the single sites lie within $[1/4, 1/4, 1/4]$.

The crystals of catalytic domain p21 are in space group $P6_122$, or its enantiomorph $P6_522$. There are two alternative origins for this space group, at $(0,0,0)$ and $(0,0,1/2)$. Two sites that are related by this alternative origin translation produce the same set of self-vectors. However, two sites related by inversion, (x, y, z) and $(-x, -y, -z)$, do not produce the same set of self-vectors in this space group, as shown in the following table, calculated for $P6_122$.

No.	Symmetry Positions	Self-vectors	Symmetry Positions	Self-vectors
1	x, y, z		$-x, -y, -z$	
2	$-y, x-y, 1/3+z$	$x+y, -x+2y, 1/3$	$y, -x+y, 1/3-z$	$x+y, -x+2y, 1/3$
3	$-x+y, -x, 2/3+z$	$2x-y, x+y, 2/3$	$x-y, x, 2/3-z$	$2x-y, x+y, 2/3$
4	$-x, -y, 1/2+z$	$2x, 2y, 1/2$	$x, y, 1/2-z$	$2x, 2y, 1/2$
5	$y, -x+y, 5/6+z$	$x-y, x, 5/6$	$-y, x-y, 5/6-z$	$x-y, x, 5/6$
6	$x-y, x, 1/6+z$	$-y, x-y, 1/6$	$-x+y, -x, 1/6-z$	$-y, x-y, 1/6$
7	$y, x, 1/3-z$	$x-y, -x+y, 1/3-2z$	$-y, -x, 1/3+z$	$x-y, -x+y, 1/3+2z$
8	$-x, -x+y, 2/3-z$	$2x, x, 2/3-2z$	$x, x-y, 2/3+z$	$2x, x, 2/3+2z$
9	$x-y, -y, -z$	$y, 2y, 2z$	$-x+y, y, z$	$y, 2y, 2z$
10	$-y, -x, 5/6-z$	$x+y, x+y, 5/6-2z$	$y, x, 5/6+z$	$x+y, x+y, 5/6+2z$
11	$x, x-y, 1/6-z$	$0, x-2y, 1/6-2z$	$-x, -x+y, 1/6+z$	$0, x-2y, 1/6+2z$
12	$-x+y, y, 1/2-z$	$2x-y, 0, 1/2-2z$	$x-y, -y, 1/2+z$	$2x-y, 0, 1/2-2z$

As can be seen from the table, 4 (numbers 7, 8, 10 and 11) of the 11 non-zero self-vectors are not equivalent for sites at (x, y, z) and $(-x, -y, -z)$. The explanation for this inequivalence lies in the fact that the space group itself is enantiomorphic. Therefore, the inversion of an atomic site is accompanied by a change of the space group to its enantiomorph. Some more calculation shows that this inequivalence between (x, y, z) and $(-x, -y, -z)$ is true for the following pairs enantiomorphic space groups — $P3_121$ ($P3_221$), $P3_112$ ($P3_212$), $P4_122$ ($P4_322$), $P6_122$ ($P6_522$), $P6_222$ ($P6_422$), and $P4_132$ ($P4_332$). Therefore, the single-site solutions of the difference Patterson map from these space groups all have the same handedness.

Due to the involvement of symmetry operations that are not part of the space group (origin translations and inversion), the single-site solutions can not be used directly in a MIR phase refinement. They have to be correlated to each other first, to the same origin and the same enantiomorph. This can be done by the two-atom cross-vector searches, because sites that are related by origin translations or inversion do not produce the same cross-vectors. For example, suppose two single-site solutions, at (x_1, y_1, z_1) and (x_2, y_2, z_2) , are obtained from a difference Patterson map for a crystal in space group $P2_12_12_1$. Site 1 can be chosen as the reference, which fixes the origin and the enantiomorph of the atomic positions. Relative to this reference, there are 16 possible positions (8 origins \times 2 enantiomorphs) where site 2 can be located, for example, at (x_2, y_2, z_2) or at $(1/2+x_2, y_2, z_2)$. The cross-vectors between the two sites and site 1 are $(x_2-x_1, y_2-y_1, z_2-z_1)$ and $(1/2+x_2-x_1, y_2-y_1, z_2-z_1)$, respectively. The two vectors are not equivalent positions in the Patterson map, and will generally assume different map values. Similar calculations can be carried out for the case of inversion. Therefore, the site combination that produces the highest set of cross-vector map values should be the correct origin/enantiomorph correlation.