

The marIndex Manual

By Dr. Klaus S. Bartels and Dr. Claudio Klein

The screenshot shows the automar (V1.0.6) software interface. The main window displays a list of Bravais lattices with their respective error values and cell parameters. The 'P tetragonal' lattice is highlighted in pink, indicating it is the selected best cell. Below the list, the 'Best Cell (after d²-refinement)' is summarized with its lattice type (P tetragonal), space group (89 (P 4 2 2)), and cell axes (a = 79.329, b = 79.329, c = 37.959). A 3D diagram of the unit cell is shown with axes labeled a, b, and c.

Lattice	Error	a	b	c	α	β	γ
P triclinic	0.0	37.92	79.20	79.29	89.90	89.89	89.98
P monoclinic	0.1	79.20	37.92	79.29	90.00	90.10	90.00
C monoclinic	0.2	111.98	112.17	37.92	90.00	89.94	90.00
P orthorhombic	0.3	37.92	79.20	79.29	90.00	90.00	90.00
C orthorhombic	0.2	111.98	112.17	37.92	90.00	90.00	90.00
I orthorhombic	13.1	21.99	111.98	109.99	90.00	90.00	90.00
F orthorhombic	13.3	21.21	156.98	157.16	90.00	90.00	90.00
P hexagonal	60.3	50.69	50.69	79.29	90.00	90.00	120.00
R hexagonal	48.9	96.54	96.54	118.46	90.00	90.00	120.00
P tetragonal	0.3	79.25	79.25	37.92	90.00	90.00	90.00
I tetragonal	13.3	111.06	111.06	21.21	90.00	90.00	90.00
P cubic	48.9	68.31	68.31	68.31	90.00	90.00	90.00
I cubic	88.6	65.68	65.68	65.68	90.00	90.00	90.00
F cubic	121.1	68.39	68.39	68.39	90.00	90.00	90.00

NOTE: Selecting a lattice (left mouse button) overwrites cell and spacegroup in 'Setup'

Cell axes permutation: Standard

Best Cell (after d²-refinement)

Lattice : P tetragonal
Space group: 89 (P 4 2 2)

Cell axes : 79.329 79.329 37.959
Cell angles: 90.000 90.000 90.000
Orientation: 11.827 39.072 19.116

Distance : 120.220 [mm]
Tilt&twist : 0.038 [deg.]
Beam center: 1159.430 1146.060 [pixels]
Mosaicity : 0.070 [deg.]
Spot size : 1.370 1.370 [mm]

Axis error : 0.290 [%]
Ang. error : 0.290 [deg.]

3D Unit Cell Diagram:
a = 79.329
b = 79.329
c = 37.959

The *marIndex* Manual

Version 1.4

March 2004 / May 2005

(*automar* 1.8 - June 2005)

Written by Dr. Klaus S. Bartels and Dr. Claudio Klein

Copyright 2002-2005 Marresearch G.m.b.H.

All rights reserved.

This document contains proprietary and confidential information of Marresearch G.m.b.H.. The contents of this document may not be disclosed to third parties, copied or duplicated in any form, in whole or in part, without the prior written consent of Marresearch G.m.b.H..

Possession, use, or copying of the software described in this publication is authorized only pursuant to a valid written license from Marresearch G.m.b.H.

marresearch, *automar*, *desktop beamline (dtb)* and the *marresearch* logo are registered trademarks of Marresearch G.m.b.H..

Note for users in a hurry

This manual is a detailed tutorial for users who want to understand what happens. You may - quite legitimately - only be interested in making it work. This is exactly the situation ***automar*** is made for. In this case, turn directly to

[section 2.1 "Going the easy way – ***automar*** GUI"](#)

and follow the instructions and hints. There is a good chance that you succeed !

Table of contents:

1. Introduction	5
1.1 Context	5
1.2 Coordinate Systems	5
1.3 Method	6
1.4 References	7
2. Modes of Operation	8
2.1 Going the Easy Way - <i>automar</i> GUI.....	8
2.1.1 Peak search.....	8
2.1.1.1 Hints	8
2.1.2 Auto-indexing	8
2.1.2.1 <i>Hints</i>	9
2.1.3 Prediction of pattern	9
2.2 More Details - the Log Files	9
2.3 Command Line Options	12
2.3.1 <i>mar</i> Peaks.....	12
2.3.2 <i>mar</i> Index.....	12
2.3.3 Options	12
2.3.4 Input files.....	13
2.3.5 Keywords	14
2.3.5.1 <i>Hints</i>	14
2.3.5.2 Syntax	14
2.3.5.3 Table of input parameters	14
2.3.6 Output file of <i>mar</i> Peaks	19
2.3.7 Output files of <i>mar</i> Index	20

1. Introduction

When starting data collection with a new crystal, either a new sample of a supposedly known crystal type or a totally unknown specimen, the first task is to locate the cell axes, before any indices can be assigned to the reflexion spots and their intensities - prerequisite for any crystallographic structure analysis. Finding the crystal lattice and its orientation needs an experienced crystallographer doing it interactively from suitably selected images, or a computer program to perform "auto-indexing". This achievement is the objective of this manual.

1.1 Context

marIndex is part of **automar**, a program suite for reduction of crystallographic reflexion data images. Designed to be as general and to work as fully automatic as ever conceivable, **marIndex** code concentrates on numerical mathematics and crystallography; it does not deal with images by itself, nor does it produce any graphics output.

These features are available separately by way of **marPeaks** and the **automar** graphics user interface (GUI).

marIndex reads a list of reflexion spot coordinates (from one or more images), and a number of basic experimental parameters necessary to convert those spot coordinates to reciprocal space. These parameters are usually available in the image headers (**marPeaks** includes those values in the [spot list header](#)) but can be supplied as separate input file(s), possibly overriding header values.

These parameters are wave-length, sample-to-detector distance, approximate beam position, and goniometer angle(s)/spindle axis setting(s) of the image(s).

(If the spots are given in pixel coordinates, as by **marPeaks**, then the pixel size is needed as well, or the distance would have to be measured in pixels.)

marIndex does not need a guess even for the order of magnitude of the lattice constants; nor is **marIndex** necessarily lost if the beam position is off by more than half the spot distance. **marIndex** will usually come up with the right answers if at least 20 spots are available; in favourable cases it may succeed with as few as a dozen.

1.2 Coordinate Systems

The *mar* images are visualized (e.g. by **marView**) as seen from the crystal, or along the beam, in an upright position. Its coordinates are interpreted as

X : across (left to right)

Y : up

as is usual for 2-dimensional systems (except pixel addresses on the video screen).

(There is an exception, though, for the "old" scanner format whose coordinates were imposed by the way DENZO assigned them: X up, Y to the right in **marView**; **marPeaks** writes therefore an additional keyword "format" now which defaults to "AUTOMAR", along with the coordinate file name extension ".pks", whereas old *marPeaks* files without the "format" keyword have the outdated convention and the file name extension ".mar"; **marIndex** recognizes both formats.)

Detector tilt and rotation are defined by 3 rotation angles (in deg.) around X, Y, and the detector normal Z (axis not used otherwise), in the math. +ve sense. Positive tilt_X or tilt_Y bring the upper or left half of the detector closer to the sample. Positive AD_rot (might be called turn_Z) rotates the area detector plane counter-clockwise as seen from the sample.

To avoid confusion, crystal coordinates x,y are defined parallel to X,Y, making the z-axis anti-parallel to the beam direction.

Crystal axes are defined by permutation indices l_1, l_2, l_3 with values $\pm(1,2,3)$ where "1" denotes the a^* -axis, "2"= b^* , "3"= c^* ; "+" means parallel and "-" anti-parallel to a coordinate axis; in this way l_1 gives the reciprocal crystal axis along the x-axis; l_2 that one "up" in the x-y-plane. The l_3 index (towards the radiation source) is redundant in principle, but may be forced to yield a left-handed system.

This may sound complicated but allows missetting angles to be always < 45 deg. so you need not stand on your head and twist the fingers while imagining the orientation of your crystal.

(Mis-)setting angles are given in degrees and are numbered the same way but are applied from right to left:

- ϕ_3 rotates around the z-axis (counter-clockwise when looking along the beam). Applying this setting first allows easy setting of e.g. a crystallographic face diagonal along the rotation axis by $\phi_3 = 45. + \delta\phi_3$, where $\delta\phi_3$ is the true setting angle. The 45 deg. are not altered by any of the other $\phi_{1,2}$
- ϕ_2 rotates around the y-axis, again in the math. +ve sense;
- ϕ_1 is a mere correction of the spindle axis (if it is horizontal) and may thus be used as an offset for the PHI axis reading.

Example: hexagonal c^* -axis along the (horizontal) PHI-axis, a^* up $\Rightarrow (3,1,2)$; $\phi_1=30$ brings the b^* -axis exactly against the beam.

Note the alternative settings that **mar Index** suggests for the primitive trigonal case, because there is no way to discern between them at this stage (without dealing with an extensive set of reflexion intensities yet).

mar Index always produces setting parameters with setting angles < 45 deg. (except for trigonal alternatives) and assigns permutation indices accordingly, although an easier permutation of axes may exist - especially in the case of equivalent axes (e.g. a cubic crystal might always be described by "axes 1 2 3" but may be assigned "3 1 2").

1.3 Method

Like most auto-indexing programs, **mar Index** is based on difference vectors in reciprocal space (although it has its own way of choosing a subset of spots and combining them to become reciprocal space vectors). From them it constructs and refines possible zone-axes which are well known in precession photography as real space vectors normal to reciprocal lattice planes ... not necessarily main cell axes.

The result of the vector analysis is the reduced primitive cell, and - on a purely geometrical basis - the most probable Bravais lattice, i.e. the highest symmetry that is acceptable within tolerance limits of refined zone-axes.

Zone-axes in the neighbourhood of the beam are used to sort spot coordinates into lunes, this way refining the beam centre. (Intuitively, the centre is the intersection of all zero-layer lunes.) The radius of convergence depends on how many zone-axes can be found and how close to the beam; in general it is more than a spot distance (unless one long axis is so well aligned that its zero-layer lune vanishes within the beam-stop area, and all other low-indexed zones are far apart - in this case it is advisable to start with a different PHI setting).

mar Index repeats the zone-axis refinement and analysis with updated reciprocal vectors. (At this stage, an orthorhombic Bravais lattice may be found where the initial vector analysis had suggested a monoclinic C-centred cell, etc.)

The most reliable non-coplanar zones are now used to index all spots in the input list. Analysis of their reciprocal point coordinates w.r.t. the sphere of reflexion (both start and end position in case

of rotation images) yields an approximate value for mosaicity/beam-divergence and updated setting angles. A full fit of calculated spot positions refines detector tilt and distance.

The final Bravais lattice analysis is based on all 25 equivalent "nearly Buerger-reduced" cells and the 44 possible transformations from primitive to Bravais lattices, implemented in the formalism of 6-dimensional "Gruber-space" (**G6**, based on the Niggli tensor components). The output list of derived Bravais lattices includes an error index which is the 6-dimensional tangent between auto-indexed primitive cell G6-vector and its projection onto the 6-dimensional hyperplane or hyperline which describes the symmetry of the primitive cell underlying the particular Bravais lattice. The preferred choice at the end is based both on these error indices and the tolerances previously found for the primitive axes and angles.

Note on monoclinic space-groups:

Unfortunately, monoclinic space-groups are ambiguous in several ways. (Interestingly, only cubic #228 with 2 choices for 192 equivalent atomic positions takes as many pages in Int.Tabl.A as monoclinic #5, C2, with just 4 positions !!)

- (a) **mar Index** chooses the canonical 2nd setting (unique b-axis); there is no option to enforce the modern 1st setting (unique axis c)
- (b) One of the standard rules for the choice of the unit cell says that oblique angles be as close as possible to 90°, and also be between 60° and 120°. (Hence, the "non-acute" monoclinic β must be in the range of 90° and 120°). In a fair amount of centred monoclinic cells, this rule leads to the choice of Bravais lattice **ml**. The **mC** alternative(s) may or may not have a "legal" $\beta < 120^\circ$. The space group number (e.g. #5) is not indicative of this choice (e.g. C121 or I121), however the rules for systematic absent ("extinct") reflexions are different !!! As if to make the problem worse, some well known programs seem to be totally ignorant of the monoclinic **ml** lattice. It is for this reason that – as a compromise - **mar Index** silently converts a monoclinic **ml** to **mC**, as long as any of the **mC** alternatives has $\beta < 120^\circ$. (Between several possibilities, the smallest β is selected). Only the log file will indicate the correct lattice, then. On the other hand, no $\beta > 120^\circ$ is regarded acceptable; but if the **ml** alternative is output, the log file will still present the (best though "illegal") **mC** cell.

1.4 References

Auto-indexing:

(Not a complete bibliography on the topic; these papers partly describe ideas that bear resemblance to those implemented in **mar Index**.)

- A.J.M.Duisenberg, J. Appl. Cryst.(1991) **25**, 92-96
 W.Kabsch, J. Appl. Cryst.(1988) **21**, 67-71
 W.Kabsch, J. Appl. Cryst.(1993) **26**, 795-800
 S.Kim, J. Appl. Cryst.(1989) **22**, 53-60

Cell reduction and Bravais lattices:

- A.D.Mighell, A.Santoro & J.D.H.Donnay, "old" Int.Tables I,5.1 (Kynoch Press)
 H.Burzlaff, H.Zimmermann & P.M.deWolff "new" Int.Tables I,9 (Kluwer Ac.Publ.)
 I.Krivy & B.Gruber, Acta.Cryst.(1976), **A32**, 297-298
 W.Clegg, Acta Cryst.(1981) **A37**, 913-915 (mind sign errors in his matrices !)
 L.C.Andrews & H.J.Bernstein, Acta Cryst.(1988) **A44**, 1009-1018
 W.A.Paciorek & M.Bonin, J. Appl. Cryst.(1992) **25**, 632-637

2. Modes of Operation

mar Index may be run in a variety of ways, differing both in the level of input control and the amount of output information.

2.1 Going the Easy Way - *automar* GUI

2.1.1 Peak search

Start the **automar** graphics user interface, and use its Setup window (F2 key) to define image and working directories; open the image directory folder, select an image file name (usually the first), and click on "Load Image Header" (or double click the file name right away); this will load the "Name template" and experimental parameter fields.

Select a range of images (one or more). The default ("first image only") may be o.k. – otherwise use the Peak Search — Edit menu. If needed, use the "Exclude detector areas" menu (F10 key). Then click on "**Peak Search**". This will run **mar Peaks** to select spots, storing the coordinates along with related header information in a peaks file, and displays the first image with superimposed spot coordinates.

2.1.1.1 Hints

- (1) Check the display to make sure it "looks right". Problems might arise if a large number of non-spots (very grainy background or satellite crystals) have been found, or if in more than a few cases neighbouring spots have been merged into one, or non-spots are lined up along the beam stop shadow. If so, you may use the "Peak Search — Edit" menu to change the significance level or the spot separation parameters; possibly adapt R_{\min} & R_{\max} , or use the "Exclude detector areas" page (function key F10, = Setup window bottom line) to exclude problem areas.
 - ➔ Do **mar Index** the favour to leave "Desired no. of spots/image" at "?" - otherwise **mar Peaks** stores the most significant spots only, which may happen to exclude next neighbours.
- (2) **automar** saves the values of the most recent session; so make sure that the numbers in the Setup window are applicable or at their defaults. In particular you should check the values of >Distance<, >PHI range<, >Wave-length< and >Beam-centre<. The safest way is to click on the related "Default" buttons recalling the current header parameters (at least if the detector was mounted on a *mar*-base so that the header values are reliable).
 - ➔ Make sure that >Space group< displays a "?" unless you know exactly which space group you have.
(The same menu is available by the "Index — Edit" button.)

2.1.2 Auto-indexing

Now click on "**Index**" program button to run **mar Index**. The clock ticks again...

As soon as **mar Index** has finished, **automar** displays the naked results in the "**Index**" window (which may at any time be recalled by F4).

The upper part is a short version of the Bravais lattice table with the 'best' choice highlighted ([cf. last paragraph of the above description of 'Method'](#)). Check the >Error< column for possible alternatives. If you are suspicious (or even sure) you want a different Bravais lattice: click on the corresponding line, and run "**Index**" again (mainly to make the orientation consistent).

Compare the r.m.s.-errors of the vector analysis in the bottom lines of this window. If these tolerances are unacceptably high, then virtually any compact cell may be regarded as 'nearly' cubic... surely something has been wrong from the outset! distance? centre?

Note that the lattice constants in the table are computed on the basis of the nominal (input) distance. By contrast, the lower part displays the adapted numbers for modified distance (resulting from the spot fit). Mind that changes of >Distance< are restricted to min(1%,1mm) by default.

Watch >Beam centre< and suggested >Mosaicity<.

2.1.2.1 *Hints*

- (1) **automar** uses the results to update the contents of the "Setup" menu, high-lighting any changed values. It is a game to re-run **mar Index** (you might reset space-group to "?" before hand) and see whether the auto-indexing results are stable and the spot-fit refinement converges (possibly driving beyond the in-built restriction for distance).
- (2) "Restore" buttons in the "Setup" (or "Index — Edit") menu recall the most recent previous results.
- (3) Selecting "Edit input file" option (just below "First image", "Last image") prevents program execution from starting right away; instead the input control file, as it is prepared by **automar**, is presented in an editor window, ready for modification. (You may choose your favourite editor in the [File → Preferences](#) menu). Exiting from the editor causes the program to start. This feature allows to make nearly all conceivable parameter modifications, or addition of options. For details, see the full description of input keywords below.
- (4) If your cell is monoclinic, see the ["Note on monoclinic space-groups" in section 1.3](#) above

2.1.3 Prediction of pattern

Click on "**Predict**" program button; **automar** returns to the image display and superimposes the calculated pattern that follows from the **mar Index** result This is a very good means to check whether the auto-indexing was successful.

It is not necessary to be too critical, though, since there is more refinement ahead in the evaluation process; but at low and medium resolution any cross ought to be inside the closest spot area (there may be zero intensity, of course, for a limited number of spots).

[For users in a hurry – that's it !](#)

2.2 More Details - the Log Files

If you want some more information on the auto-indexing progress, click on the **automar** "Text" button (or F3 key); the text window displays the screen output of **mar Index** (medium length). The maxi-version is available as `index.marIndex` log-file in the working directory (not in **automar**, but using any text editor).

Both listings give intermediate results of the program steps described above in [section "Method"](#). Here you find details like:

- how many spot coordinates have been used in which step; e.g.

```
308 reflexion coordinates read from file.
102 reflexion coordinates used in the first run.
```

- how many zone-axes have been found, how well did they refine on average;
- which preliminary primitive cells have been constructed from them;
- how many vectors refined how well in the (extended) W.Clegg-type analysis;
- which were the associated tolerances in x,y,z; e.g.:

```
43 out of 52 zone-axes in total, thereof :
14 out of 14 Clegg-vectors, contributed in least-squares refinement;
average r.m.s. indexing error of used zone-axes = < 0.1173 >
```

(The number in pointed brackets deserves a comment:

Every selected reciprocal space vector is projected onto each zone axis yielding a Miller index which is most probably non-integer. The maximum -absolute- index error is 0.5 of course. An average of <0.288675> is the expectation value for a random, or flat, distribution without any rejection. Or, after rejection of $|e| > 0.35$, the expectation value is <0.202073>; which tells us that a reasonably well oriented zone axis should have an average indexing error at least as small as this. The r.m.s. value over all accepted zone-axes ought to be better. If it is only slightly smaller, or not at all, it may be indicative of a split or twinned crystal.)

- how far was the centre shifted by each of lune and spot fit; e.g.

```

Lune refinement :
=====
                          Centre X                Centre Y                / oblique
Estimated radius      [mm] [pixels]          [mm] [pixels]          / elongation
of convergence :      2.715   18.10            1.925   12.83            37.0 %

-----

Shift      :      -0.406   -2.70            -0.025   -0.16
Estimated error :      0.020
Old beam centre :      90.000   600.00        90.000   600.00

-----

New beam centre :      89.594   597.30        89.975   599.84

```

The "radius of convergence" is approximated as an oblique ellipse and may rather be slightly over-estimated; so be careful if any one of the applied shifts is a substantial fraction thereof...!

(The "oblique elongation" will be positive if the elongation tends to be in the direction of the main diagonal $X=Y$, negative otherwise, its magnitude being an estimate of the deviation from the average radius; it is close to 0 if the orientation of the ellipse is along the main axes X and Y . But you are probably up to no good anyway if elongation and direction matter.)

Consult any of the listings for

- the permutation indices of the crystal setting (hidden parameters in **automar** GUI) and for alternative settings in the primitive trigonal/hexagonal case; e.g. :

```

Setting angles = -9.92  -1.22  -3.42  for axes = { c*  a*  b* }, i.e.
                                                direction of c* along spindle axis X,
                                                a* upwards in X-Y-plane,
                                                b* towards X-ray source.

alternatively  50.10  -3.57  -0.65  (= trigonal ambiguity;
or : -69.82   2.35  -2.76  with same axes choice)

```

- detector tilt;
- estimated errors of refined parameters;
- the relative contributions of linear and non-linear shifts in the spot-fit (i.e. distance vs. wave-length / cell-axes).

This may again deserve an explanation:

Let R be the reflexion spot's distance from the centre; $R/D=\tan(2\theta)$, with Bragg-angle θ (theta) defined by " $2\sin\theta=\lambda/d$ ". For small θ angles, $\tan(2\theta)$, 2θ (in rad.) and $2\sin\theta$ are approximately the same: $R/D=\lambda/d=|d^*|$; the observed low-resolution spot can be described by any combination of D and d^* as long as $D\cdot|d^*| = D\cdot\lambda/d = R = \text{const.}$; in particular, for a fixed wave-length (X-ray tube) : $D/d=\text{const.}$; or for any given crystal lattice at the synchrotron : $D\cdot\lambda=\text{const.}$; all this is true if R is small enough for $D\cdot\tan(2\theta)$ and $D\cdot 2\sin\theta$ to differ by less than, say, 1/2 pixel and thus be indistinguishable.

In other words, distance D and lattice constants d^* are 100% correlated for small 2θ , or more precisely if $(0.7\cdot R)^3 < (8/3)\cdot\text{pixel_size}\cdot(D^2)$. Only if spots are observed at resolutions beyond this limit is there any chance of refining D and d^* simultaneously. Exactly this limit is implemented in **mar Index** (except for eigen-value filtering):

In the first stages, lattice constants are derived on the basis of D being given. Then, if there are high-resolution spots available, D may be refined and the cell constants modified by an appropriate balance factor which will be found in the listings, in the section "Spot refinement". The D refinement may, however, be suppressed by definition of 0 tolerance, see keyword definitions below.

The log-file also includes

- the Niggli constants of the primitive cell, or rather the G6-values which differ from the conventional Niggli tensor by a factor of 2 in the mixed products: $(\mathbf{a}\cdot\mathbf{a})$ $(\mathbf{b}\cdot\mathbf{b})$ $(\mathbf{c}\cdot\mathbf{c})$ $2(\mathbf{b}\cdot\mathbf{c})$ $2(\mathbf{a}\cdot\mathbf{c})$ $2(\mathbf{b}\cdot\mathbf{a})$; e.g.

```
Primitive cell metric tensor [G6] :    7700.40    7741.88    9267.74
                                       33.52      4.64      61.85
Primitive cell volume : 743297.1
```

- a display of the angular distribution of the reciprocal space lattice points (\Rightarrow mosaicity), e.g.

```

      psi1 : within dPHI 234          psi2 : "near axis" 19
< -1.5*OSC 0   used total 290        2   used total 74
...
(weighted, scaled)
-0.80  0 | : | 0 | :
-0.70  2 | * | 0 | :
-0.60 14 | : | 0 | :
-0.50 22 |-----*-----| 0 | :
-0.40 24 | : | 2 | * | :
-0.30 28 | : | 0 | :
-0.20 27 | : | 8 | * | :
-0.10 25 | : |13 | * | :
 0.00 22 | = = = = = : = = * = = | 33 | = = = = = = = = = = = *
 0.10 24 | : |12 | * | :
 0.20 23 | : | 2 | * | :
 0.30 20 | : | 3 | * | :
 0.40 21 | : | 1 | * | :
 0.50 17 |-----*-----| 0 | :
 0.60  7 | * | 0 | :
 0.70  0 | : | 0 | :
 0.80  0 | : | 0 | :
 0.90  1 | * | 0 | :
 1.00  0 | : | 0 | :
.....
> 1.5*OSC 1   0
```

The bars in the left histogram denote the bounds of the rotation/oscillation with any spots "outside" defining the 'vertical' mosaicity; the 'horizontal' mosaicity (the right-left-rotation ψ_2 that is necessary to bring any rec. lattice point onto the Ewald sphere) is virtually independent of Φ , hence no $\Delta\Phi$ bars in the right hand histogram. (This statement is true for the "Rotation Method" only, not for the precession geometry !) The two mosaicities are considered the same if they are closer than 2:3 and only their mean value is output then.

The log-file Bravais lattice table is more elaborate, specifying also

- the parameters of each related primitive cell (which is in fact the cell that is measured by the 'Error' index, i.e. the above-mentioned 6-dim. tangent; cf. section "Method"), and
 - the transformation number (0 through 44, as specified in Table 9.3.1 of Int. Tabl. Cryst. [Vol.A], 4th ed.);
 - it also indicates which of the nearly-reduced Buerger-cells was closest to this result (its running number only).
- In case vector analysis and [G6-space](#) projection give different 'best' cells, or if the user-defined lattice/space-group is different, the alternative matrices and setting parameters are listed.

2.3 Command Line Options

automar GUI is the most convenient way of running **mar Index** and will usually be the method of choice. There are very few options, though, that are available only in command-line execution mode. Therefore the full description will follow under this heading.

Set your default to the working directory. (It may, but should not in general, be the same as the image directory.)

2.3.1 *mar Peaks*

As a first step, type:

```
> marPeaks [image_directory]image_file.name
```

The output will be → image_file.pks (in the working directory.)

If non-default parameters are needed, consult the **mar Peaks** manual. If uncertain about the situation, use **mar View** to display the image and superimpose the **mar Peaks** spots.

2.3.2 *mar Index*

Then type:

```
> marIndex [-f or /F] input-file [options]
```

It is these [options] that **automar** may not generally be supporting.

As usual the options consist of one letter, preceded by either "-" (UNIX) or "/" (VMS); **mar Index** accepts both conventions, letters in upper or lower case are equivalent. The option parameter(s) may follow immediately or (VMS-like) be appended by "=", or may simply be separated by blanks.

(The "-f" or "/f" switch is only needed if input-file is not the first or last command-line parameter.)

2.3.3 Options

- ▶ switch **v** ('verbose') has only one out of many debugging options that may be of interest to few users, which is "p" (for protocol) which will produce output listings with detailed tables of all zone-axes that the program tries to refine and use.
- ▶ switch **s** ('service') NONE DENZO MOSFLM XDS MARXDS AUTOMAR makes **mar Index** convert its results, and output them in a format that is suited as input to these programs. The current default is "NONE" for screen output (or => **automar** "Text" window), and "DENZO MOSFLM XDS AUTOMAR" for the log file. Specify "-s none" to suppress this 'foreign' output in the log-file. In some distributions the default may be changed to NONE for all output listings.

Example:

automar-derivative "**mar FLM**" is a gateway to data processing by way of MOSFLM. It therefore submits this command-line:

```
"marIndex automar.ctrl -s mosflm"
```

which will produce 2 extra files:

- ▶ automar.mosflm,
- ▶ automar.umat,

the first of which contains keyworded MOSFLM input, and the second holds the formatted setting parameters & matrices.

Unless the `-s` option explicitly specifies foreign formats, **mar Index** writes an internally consistent `".amp"` ([AutoMarParameters](#)) file which is suited both for re-running **mar Index** or for subsequent **automar** prediction and integration programs, with keywords as described below.

If `-s` specifies any foreign format, and if the `".amp"`-file is wanted as well, then AUTOMAR needs to be in the options list.

Comment: 'service' is also available as input keyword. If both are specified, command line definitions have priority.

2.3.4 Input files

The "input-file" in the command line may in its simplest form be the [output of mar Peaks](#), "image_file.pks", if all header information is o.k. and sufficient to determine the spot transformation to reciprocal space - which is usually the case for **mar**-detectors on **mar**-bases.

Else "input-file" may be some control file (any name, default: "automar.ctrl") which defines the **mar Peaks** output file by means of the "peak" keyword and otherwise specifies all missing or wrong parameters. (Values in the control-file will override parameters in the peaks file header.) See the keyword list below.

Mind that **mar Index** writes the `".amp"` ([AutoMarParameter](#)) file which is suited as an input control file to **mar Index**. If you choose to do so under UNIX, the new output file will destroy the previous one; VMS will create a new version.

Going one level further. You may have the situation that synchrotron data have the wrong distance and/or wavelength in their header. Other (image-specific) parameters may be o.k. Instead of preparing a separate control file for each image that you want to auto-index - just for the sake of the one or two wrong values - you may specify one "common parameter file" with the replacement(s) only, which will automatically be read by **mar Index**, then immediately going on with the **mar Peaks** file, or else with image-specific control files. For this mechanism to work, the "input-file" (peaks or control file) must have the standard form of image numbering, and the "common parameter file" must have the ".prm" extension attached to the root of the image names (i.e. without the running number); or else the (image-specific) control file may be any name, say "any_name.ext", with an accompanying common parameter file "any_name.prm".

Example: Let the images be named "Lyso_2_001.mar2000", "Lyso_2_002.mar2000", ..

Then **mar Peaks** would produce "Lyso_2_001.pks", etc.

Prepare the "common parameter file" named "Lyso_2.prm" and then type

```
> marIndex Lyso_2_001.pks
```

or edit another "Lyso_2_###.ctrl" for image ### and then type

```
> marIndex Lyso_2_001.ctrl
```

In either case, **mar Index** will first find "Lyso_2.prm" and read its parameters which will take priority over peaks-file headers (but will be overridden by any values in an input control file). The idea behind these priorities is the following: Any error that may have sneaked into the image headers should be replaced once for all by the ".prm"-file; still the user ought to have full freedom in playing with input parameters by way of some image-specific control files.

Warning !! Once you have a "common parameter file" which fits the name convention it will be read and used !!! So watch out what files you keep on your working directory.

2.3.5 Keywords

All **automar** keyworded input is consistent. The same keywords, parameters and options may appear in the *mar* Peaks file header, in any control file, or in a "common parameter file".

2.3.5.1 Hints

The following list specifies mandatory parts only of valid key-words (any length is acceptable as long as the typed string is recognizable as one word). Valid separators are blanks and/or tabs. To avoid frequent (though unnecessary) errors, some composite keywords are accepted with or without embedded - and/or blank such as :

wavelength | wave length | wave - length | wave-length;

or similarly :

beam | beam centre | beam center |...

This latter keyword causes the only exception such as to avoid ambiguity: "beamstop" must at least have the "stop" as keyword, i.e.:

beamstop | beam stop | beam-stop | stop

2.3.5.2 Syntax

- > *different fonts denote:* ► key-words ► input-values ► explanations
- > [optional] text in square brackets may or may not be typed;
- > [parameter \ default] (back-slash)
- > key1 | key2 alternatives with same meaning separated by vertical bar, e.g. :
set[ting-angle]s | ori[entation] ;

2.3.5.3 Table of input parameters

► Basic parameters

[automar parameters]		! optional heading
[-----]		! (obsolete by way of <i>comments</i>)
[[unit]-cell]	a,b,c, α , β , γ [, Bravais ext.group \ 0]	unitcell;latt
[axes]	l_{axes} , l_{up} , l_{in}	axes permutaion
[ori set]	φ_1 , φ_2 , φ_3	orientation (setting angles)
	→► for both these 2 lines, cf. section 1.2 "Coordinate systems" above.	
[wave[-length]]	wavelength ₁₁ [...wavelength ₂₂]	wave-length
	→► cf. synonymous keyword lam[da] below	
[dist[ance]]	Sample-to-Detector_Distance	distance
[rad[ius]]	R_{min} [, R_{max} [, reso_limit]]	radii [mm]; optional resolution [A]
[imag[e-range]]	first_image, last_image, [START] start_oscill, [OSC] OSC_width [, mosaicity, spot_size(x,y)]	image range, & PHI range [optional; obsolete, see <i>hint</i>]
[file[-name] name]	image_file_name (→► <i>specify the correct name, not a template</i>)	file-name cf. keywords "data image"
[scan]	image[_x]_size [, image_y_size]	detector max. number of pixels
[beam[-cent]]	c[x]_image [, cy_image]	beam centre [in pixel]
[pix [el]]	p[x]_image [, py_image]	pixel-size [in mm]

Hint: For historical reasons, the above parameters are acceptable without keywords as long as they appear in exactly this order.

The remainder is available by way of keywords only :

➤ Comments

... !./ . ! everything behind an exclamation mark, on any line, is ignored (i.e. treated as comment);
 # ./ . lines with # or * as 1st non-blank character are ignored,
 * ./ . such that comment[ed-out] lines may start with ! or # or *
 remark ... or with key-word "remark" which is assumed to terminate the key-word lines of interest for parameter reading, see *mar* Peaks file-header.

➤ Keywords

▶ Crystal :

latt[ice] Bravais-lattice or extinction-group
 —▶ only the Bravais-lattice definition is relevant in *marIndex*: Two characters as defined in new Int.Tables (Vol.I, table 9.2.2), e.g. oP, cI, etc. Usually this is more adequate when auto-indexing, but space group definition is acceptable, too:

spac[e-group] name or number in Int.Tables
 (—▶ implies the default Bravais-lattice)

reso[lution] resolution-limit = $\lambda/(2 \cdot \sin\theta_{\max})$ [Å]
 —▶ alternatively input as 3rd value with 'rad'

mosa[ic..] mosaicity(x) [, (y)] [in deg.]
 —▶ ignored as an input value, but will be passed on to the ".amp"-file overriding any *marIndex* results.

spot[-size] spot_size[_hor., spot_size_vert.] [mm]
 —▶ same comment as for mosaicity.

▶ Goniometer :

wave[-length] wavelength₁₁ [...wavelength 22]
 lam[[b]da] —▶ synonymous with 'wave[-length]'

—▶ may either be given as 'Cu|Mo|Ag' for stored values of $K\alpha_{1/2}$ and will consider α_1/α_2 split in refinement; or as numbers.
 —▶ Up to 4 numbers may have one of two different meanings :
 (1) possibly composite wave length (also true if keyworded Cu|Mo|Ag)
 <nominal λ > [, $\delta\lambda/\lambda$ | def=0.003 [, $K\alpha_1$, $K\alpha_2$]]; or
 (2) two superimposed wave lengths (e.g. $K\alpha$ - $K\beta$; or synchrotron 1st and 3rd order Ge111 - Ge333) :
 λ_1 , $\delta\lambda_1/\lambda_1$, λ_2 , $\delta\lambda_2/\lambda_2$

- div[ergence] divergence(x) [, (y)] [in deg.]
- pol[arization] sub-keyword "mir[rors]" implies no polarization, and if wave-length Cu is specified, *marIndex* will also try to index badly fitting spots by K_{β} - radiation.
 ► Alternatively:
 sub-keyword "mon[ochromator]" may be accompanied by Ge[111] | Si[220] | default: graphite; *automar* assumes the monochromator to deflect the beam horizontally and knows the monochromator's lattice constants; so for each wave-length it can work out the degree of polarization (immaterial for auto-indexing).
- dist[ance [fix]] Sample-to-Detector_Distance [mm] [, D-tolerance [%]]
- The optional keyword *fix[ed]* on the same line has the same effect as \Rightarrow D-tolerance=0% \Rightarrow no refinement of Distance
- omg | omeg[a] OMG [, OMG_off] [deg.]
- !! within *marIndex* not yet consistently implemented
- chi CHI [, CHI_off] [deg.]
- tth | twot[heta] TTH [, TTH_off] [deg.]
- detector position
- Detector :
- tilt tilt_X [, tilt_Y] [deg.]
- rot AD_rotation [deg.]
- detector-rotation w.r.t. projection of TTH-axis
- skew tilt_X [, tilt_Y [, AD_rot]]
- for all 3 detector rotations, see section 1.1 "Coordinate systems" above
- [beam -]stop X,Y, R | X1,Y1, X2,Y2 [, W]
 !!! "stop" is mandatory !!!
- **marIndex** rejects **marPeaks** coordinates within this shadowed circle or rectangle (in [pixel] as read from **marView**), rectangle either given by lower left and upper right corners, or by 2 end points and width of central line.
 —► Similarly,
- [excl[ude]] shad[ow] | ice | ring[s] | a.s.
 can be specified to exclude spots from certain areas, just like in the other **automar** programs, but of course it is preferable not to have then written by **marPeaks** in the first place.
- beam[centre] c[x]_image [, cy_image] [fix]
- beam centre, like distance, may be fixed, i.e. not included in refinement.

► Options

fix [beam[centre] | cent[re]], [dist[ance]]

- this keyword specifies which parameters not to include in refinement; the values are not numbers but rather keywords. This way of definition is synonymous with typing "fix[ed]" on each of the keyword lines concerned.

serv[ice] [NONE] [DENZO] [MOSFLM] [XDS] [MARXDS] [AUTOMAR]

→ These **automar** options are not used in *marIndex* :

list [list_code | all_extra_listings]
 plot [plot_code | all_plot_options]
 titl[e] ... (1st and 2nd title allowed, defined by their sequence)

► File name specification :

root root-of-image-file-names
 mar | peak[s] marPeaks-file-name
 ext image_file_name-extension (with or without preceding ".")
 imag[e] | data template of image_<001>_file

where the digits for the running number is enclosed by pointed brackets.

This is a generalization of the standard *mar* image file name convention to allow more than 3 digits to appear at any place in the name string. It is the pointed brackets <....> that discern this use of the "image" keyword from the above one to define image and PHI ranges.

- If image-file-name only is specified but *marPeaks* file is required, then its name is constructed with extension ".pks"
- If neither "peaks" nor "image" is specified, then any required file-name is constructed as 'root'_<nnn>.'ext' with <nnn> = first_image, formatted as (I3.3)
- If image-file-name is required but not explicitly specified, its extension defaults to ".mar<scan>" where <scan> is the (I4.4)-formatted image[_x]_size

➤ Keywords and parameters, sorted in groups:

(This is an extract from the input routine: Leading blanks or tabs on the input *line* are skipped, the next is interpreted as the **keyword**.)

Comments or titles:

```
keyword(:4).eq.'rema' .or. line(1:1).eq. '#' .or. line(1:1).eq. '*' .or.
line(:4).eq. '----' .or. INDEX(line, '!').ne.0
keyword(:4).eq. 'titl'
```

```
! )
```

```
! (
```

Options:

```
keyword(:4).eq. 'list'
keyword(:4).eq. 'plot'
```

```
! \ ignored by
```

```
! / marIndex
```

```
! (
```

```
! )
```

Scanned image:

```
keyword(:4).eq. 'root'
keyword(:4).eq. 'file' .or. keyword (:4).eq. 'name'
keyword(:3).eq. 'ext'
keyword(:3).eq. 'mar' .or. keyword (:4).eq. 'peak'
keyword(:4).eq. 'imag'
keyword(:4).eq. 'scan'
keyword(:3).eq. 'pix'
```

```
! "mar-peaks"
```

Goniometer:

```
keyword(:4).eq. 'wave' .or. keyword (:4).eq. 'leng' .or. keyword (:3).eq. 'lam'
keyword(:4).eq. 'dist'
keyword(:4).eq. 'beam'
keyword(:4).eq. 'tilt'
keyword(:3).eq. 'rot'
keyword(:4).eq. 'skew'
keyword(:3).eq. 'omg' .or. keyword(:4).eq. 'omeg'
keyword(:3).eq. 'tth' .or. keyword(:4).eq. 'twot'
keyword(:3).eq. 'chi'
```

```
! beam [cent]
```

```
! = 'tilt' + 'rot'
```

```
! = to_lower('OMG')
```

```
! = to_lower('TTH')
```

```
! = to_lower('CHI')
```

Crystal & diffraction pattern:

```
keyword(:3).eq. 'set' .or. keyword(:3).eq. 'ang' .or. keyword(:3).eq. 'ori'
keyword(:4).eq. 'axes'
keyword(:4).eq. 'unit' .or. keyword(:4).eq. 'cell'
keyword(:4).eq. 'latt'
keyword(:4).eq. 'spac'
keyword(:4).eq. 'mosa'
keyword(:4).eq. 'spot' .or. keyword(:4).eq. 'size'
keyword(:3).eq. 'rad'
keyword(:4).eq. 'reso'
keyword(:4).eq. 'stop'
keyword(:4).eq. 'shad'
keyword(:4).eq. 'ring'
keyword(:3).eq. 'ice'
keyword(:4).eq. 'a.s.' .or. keyword(:4).eq. 'ammo'
```

```
! unit[ - ]cell
```

```
! [beam - ]stop
```

2.3.6 Output file of *mar* Peaks

At this stage, the format of the *mar* Peaks file should be self-explanatory. Some comments may be helpful concerning the usage of data columns by *mar* Index, see below. Here is an example :

```

program      MARPEAKS 4.1
date         Thu May 20 14:48:47 1999
title        MARPEAKS: Spots search
root         /home/klaus/data/gbf/img/gbf_100
format       AUTOMAR
coordinates  x horizontal  y vertical  origin lower left
image        1  1          START    0.000  OSC    1.000
scan         2000 x 2000
distance     400.000
wavelength   0.99384
omega        0.000
chi          0.000
twotheta     0.000
pixelsize    0.150
remark       flag=0: fully recorded spot
remark       flag=1: partially recorded spot
remark       flag=2: spot consists of lambda1 and lambda2 components
remark       flag=3: 2 separate spots are lambda1 and lambda2 components
remark       flag=4: bad intensity/sigma ratio
remark       flag=5: bad ratio of width(x)/width(y)
remark       flag=6: bad ratio of #signal/#background pixels
remark       flag=7: spot is too large in PHI
# FLAG X Y PHI INTENSITY FFRM LFRM NP I/BKG
1 0 1596.30 1775.20 0.50 1568.0 1 1 6 7.08
2 4 1625.00 1774.00 0.50 17.0 1 1 5 0.79
3 4 991.40 1770.80 0.50 1739.0 1 1 6 3.15
4 0 1247.80 1762.90 0.50 8131.0 1 1 22 7.84
.....

```

- ▶ For "**FLAG**", see "remark"-lines. Depending on the situation, *mar* Index ignores spots with FLAG>0, >1, or >3
- ▶ "**X**", "**Y**", see "format" & "coordinates" header lines and "Coordinate systems" section above; measured in [pixel] units.
- ▶ "**PHI**" is the mean value of the oscillation range, START + OSC/2, except for spots that are found on two or more adjacent images in the range of [first image, last image] (see "image" header line), when PHI is a weighted mean angle of first (FFRM) and last (LFRM) appearance.
- ▶ "**INTENSITY**" is not used by *mar* Index
- ▶ "**NP**" counts the pixels above threshold for this spot. *mar* Index analyses several histogrammed distributions in order to detect uneven distributions of spot sizes and may e.g. reject a substantial fraction of very tiny spots if they form a separate cluster in the histogram. Also the "spot size" value in the ".amp"-file is derived from the average NP of accepted spots. This is meant as a starting value for the integration only; mind that *mar* Index cannot analyse any spot shapes since it does not deal with the image.
- ▶ "**I/BKG**" = significance of spot; not used by *mar* Index because weak spots have "FLAG=4" anyway.

2.3.7 Output files of *marIndex*

File names are constructed from the "input-file" on the command-line, retaining its part up to the dot and appending these extensions:

".amp" for the output parameter file
".marIndex " for the log-file

This way *automar* GUI requires a minimum of house-keeping because it generates a control file with the standard name "index.ctl" which always leads to log-file "index.marIndex", and "index.amp" parameter output.

Comparing auto-indexing results from different images may therefore sometimes be easier when running *marIndex* from the command line using peaks-files with standard name conventions, with individually named output parameter and log-files ... at the expense of many files to be deleted in the end of the survey.

Here is an example of a typical ".amp"-file, hopefully self-explanatory now:

```
# automar parameters
# -----
cell      87.294  87.294  96.384   90.000  90.000  90.000
lattice   tP
orient    -6.920  -14.595  20.270
axes      3  2  -1
mosaic    0.21
wavelength 0.99384  0.0030
distance  400.00
rad        7.50  150.00   0.00
spot      0.95   0.95
beam      1010.89 1006.03
tilt      -0.024   0.118
image     001  001  START   0.00  OSC  1.00
file      gbf_100_001.mar2000
scan      2000 x 2000
pixel     0.150  0.150
```

► Note that no space-group had been input; consequently, the output specifies "lattice" only, which is all that can be said at this stage.

(The log-file lists the lowest space-group number compatible with this Bravais lattice, as well.)

► This is another example of axes permutation indices not at their very simplest form: with the tetragonal 4-fold along the spindle axis, a* and b* axes are equivalent and "axes 3 1 2" would be equally appropriate.