J. Appl. Cryst. (1992). **25**, 92–96

Indexing in Single-Crystal Diffractometry with an Obstinate List of Reflections

BY ALBERT J. M. DUISENBERG

Laboratorium voor Kristal- en Structuurchemie, Bijvoet Center for Biomolecular Research, Padualaan 8, NL-3584 CH Utrecht, The Netherlands

(*Received* 1 *July* 1991; *accepted* 6 *September* 1991)

Abstract

An indexing method for single-crystal diffractometry is described which is applicable to especially difficult cases such as twin lattices, incommensurate structures, fragmented crystals, long axes and unreliable data. Finding the reciprocal lattice from a cloud of reciprocal-lattice points (reflections) is reduced to finding elementary periods in one-dimensional rows, obtained by projecting all observed points onto the normal to the plane formed by any three of these points. Row periodicity and offending reflections are easily recognized. Each row, by its direction and (reciprocal) spacing, defines one direct axis vector, based upon all cooperating observations. From the direct vectors so obtained a primitive direct cell is chosen and refined against the fitting reflections. The result is one main lattice, or a main lattice and a set of alien reflections. The method operates semi-automatically in the program *DIRAX* and has been tested, without failure, on hundreds of CAD4 reflection files, among which there were many auto-indexing-resistant lists.

Introduction

It is common practice in three-dimensional indexing to choose a suitable initial reciprocal cell (three shortest independent vectors) from the observed reflection vectors and their differences. Reciprocal-space methods such as the CAD4, 'INDEX' (*CAD4 Manual* 1989) start with indexing the reflection list with the initial cell. Because the initial reciprocal axes are (unknown) integer linear combinations of the true reciprocal axes, the ratio of the volumes of the initial and true reciprocal cell is an integer. Therefore, reflection indices calculated with the inverse orientation matrix from the initial cell are exact fractions, with denominators equal to (a divisor of) the integer volume ratio. This ratio can be derived from the fractional indices, and a linear transformation matrix with determinant equal to the ratio is set up, such that all indices are converted to integers and thereby the initial orientation matrix to the final one. Each reflection is treated individually and their coherence as being elements of a translation group is used implicitly only once when looking for the common denominator in the initial indices.

Clegg (1984) describes an enhancement of the direct-space method, based upon studies by Jacobson (1976) and Sparks (1976, 1982). An initial cell is used to produce systematically direct-lattice vectors. In order to verify a generated vector as a true direct vector the condition is applied that scalar multiplication of a true direct vector and any true reciprocal vector results in an integer. If a great majority of the products of a tentative direct vector and each of the available true reciprocal vectors (*i.e.* the observed reflection vectors) are integers, the direct vector is accepted as a true one. From the set of accepted direct vectors a final cell can be obtained. In this method the cohesion between the reflections is repeatedly exploited since all reciprocal vectors are used to confirm one direct vector.

Reciprocal- and direct-space methods are both initial-cell methods. The initial cell, however, is not based on all information but only on a subset, *viz* of three (difference) vectors. Choosing a wrong initial cell is fatal and this may easily happen with alien (non-fitting) reflections in the list. If from 25 reflections 9 are aliens (Example 1) the chance of choosing a false initial cell is 74%. Even with a single lattice the process may be obstructed when inaccurate or ill-distributed data are provided (see Example 2). Then the initial cell may be unfit to calculate reliable fractional indices or to generate reliable direct vectors. With incommensurate structures (with satellite reflections at regular distances close to the main lattice points) usual indexing procedures may find only a superlattice (see Example 3).

Sometimes indexing problems can be solved by manipulating the data. Rearrangement of the reflection list, averaging of (multiples of) difference vectors, exclusion or replacement of seemingly unfavourable reflections, adapting parameters or tolerances and manual selection of trial vectors are common examples, apart from still more ingenious, mostly undocumented, activities. If unreliable setting angles originate from heavily structured profiles from fragmented crystals an experimental method (Duisen-

0021-8898/92/020092-05$03.00

Example 1. *Automatic solution for a twin lattice from 16 regulars and 9 aliens*

LevelFit is set more critical than usual to increase the discrimination between main lattice reflections and aliens. Note that all l indices are integer.
λ: 0.70930 Å; IndexFit: 0.10, LevelFit: 1/48; *D*max: 80; **t** vectors: 157; ACL: 11.

nF statistics
(*101* unique t-vectors with exactly 5 fitting reflections, etc.)

5	*101*	8	*3*	11	*2*	14	-	17	-	20	-	23	-
6	*32*	9	*5*	12	*1*	15	*1*	18	-	21	-	24	-
7	*9*	10	*1*	13	-	16	-	19	-	22	*1*	25	*1*

nr	theta	phib	chib	h	k	l	dth	dom	dch
1	16.20	-106.45	34.88	-8.017	2.012	-0.001	0.052	-0.011	0.092
2	17.91	-51.98	30.50	-6.995	1.995	4.998	-0.018	0.007	-0.040
*3	17.97	-106.30	31.12	-9.187	1.970	0.007			
*4	12.41	-132.04	22.53	-5.570	0.990	-1.996			
5	12.49	-93.65	23.45	-6.938	0.995	0.999	-0.107	0.095	0.068
*6	12.44	-118.74	22.95	-6.359	0.986	-0.993			
7	12.85	-106.23	22.99	-7.020	1.004	-0.001	0.039	-0.006	0.021
8	12.91	-81.41	22.88	-7.003	1.004	1.999	0.009	-0.022	0.063
*9	12.64	28.00	35.60	2.900	1.925	3.004			
10	15.93	0.77	49.66	-0.001	2.997	4.000	-0.009	-0.008	-0.020
11	11.33	5.32	20.16	0.909	1.000	3.995	-0.063	-0.725	0.190
12	10.93	-2.92	21.26	0.008	0.984	4.001	-0.018	0.039	-0.322
*13	10.90	-4.00	21.73	-0.109	0.998	3.994			
14	11.29	57.53	41.76	4.001	2.006	1.004	0.024	-0.041	0.063
*15	10.68	74.74	42.17	4.211	1.928	0.005			
16	11.21	-26.17	48.06	-2.001	2.003	3.000	0.010	-0.005	0.042
17	15.98	87.85	45.29	5.981	3.011	-1.001	0.012	0.037	0.188
18	16.27	74.56	25.50	8.052	2.006	-0.008	0.089	0.066	-0.085
*19	12.61	121.21	34.83	4.540	1.945	-2.994			
20	10.98	88.10	16.74	5.995	0.993	-1.002	-0.019	0.033	-0.126
21	11.33	-177.85	21.13	-0.953	0.984	-4.004	-0.039	-0.370	-0.306
*22	9.79	129.14	50.90	2.234	1.969	-1.989			
23	8.43	128.97	68.16	0.946	1.984	-0.987	-0.085	0.381	0.355
24	13.14	145.40	65.47	1.000	3.000	-2.000	-0.000	0.005	-0.002
*25	14.06	-160.31	63.21	-2.145	3.036	-1.989			

reciprocal axes matrix **R**

-0.081290	-0.016286	0.010505
0.023237	0.001830	0.123872
-0.006756	0.196690	0.000074

direct axes matrix **D**

-11.9294	1.0123	-0.9972
-0.4106	0.0318	5.0498
2.2439	7.8825	0.1124

cell dimensions

a b c:	12.0137	5.0666	8.1964	volume
$\alpha\beta\gamma$:	90.142	101.067	90.099	489.63

Example 2. *Automatic solution for a small protein known to be tetragonal*

The data are ill-distributed (narrow **phib** region) and suffer from serious systematic errors. Therefore the criteria IndexFit and LevelFit are set wider than usual. (With IndexFit 0.2 all reflections fit, but to a still more inferior cell.)
λ: 1.54056 Å; IndexFit: 0.15; LevelFit: 1/12; *D*max: 160; **t** vectors: 1691; ACL: 22.

nF statistics
(*378* unique t-vectors with exactly 5 fitting reflections, etc.)

5	*378*	9	*149*	13	*10*	17	*1*	21	*2*	25	-
6	*371*	10	*81*	14	*1*	18	-	22	*1*	26	*11*
7	*362*	11	*35*	15	*1*	19	*1*	23	*2*		
8	*270*	12	*14*	16	*1*	20	-	24	-		

nr	theta	phib	chib	h	k	l	dth	dom	dch
*1	4.43	6.86	-46.71	-0.018	6.849	3.884			
2	10.99	11.92	-16.45	3.005	10.881	14.924	-0.069	-0.010	0.214
3	16.44	17.32	-13.36	4.008	13.961	23.989	-0.013	-0.001	0.073
*4	5.20	6.50	-29.61	0.990	6.833	5.880			
*5	10.91	12.29	21.73	5.994	-0.170	14.843			
6	14.87	15.52	-5.08	4.992	9.995	21.973	-0.018	-0.001	-0.017
7	18.19	19.35	-13.41	4.013	14.966	26.998	-0.006	-0.012	0.076
8	17.16	17.95	8.17	7.000	5.013	25.996	-0.001	-0.015	-0.020
9	19.03	22.39	28.70	8.996	-6.076	26.915	-0.035	0.009	0.167
10	18.92	19.50	-1.61	6.008	10.021	29.096	0.057	0.040	-0.003
11	16.07	16.85	-6.32	4.986	11.017	23.973	-0.016	-0.000	-0.068
12	19.07	22.41	-28.34	0.987	21.016	25.982	-0.003	0.008	-0.060
*13	10.25	12.59	32.19	6.002	-3.195	12.881			
14	10.59	11.20	-5.79	3.998	7.893	14.909	-0.068	-0.026	0.225
15	4.60	6.64	39.71	2.994	-2.147	4.914	-0.013	0.041	1.181
16	6.31	8.93	40.21	3.996	-3.150	6.911	-0.012	0.043	0.885
17	6.46	8.04	28.91	3.995	-1.134	7.910	-0.032	0.000	0.756
18	15.46	16.56	12.55	6.999	2.944	22.903	-0.052	-0.052	0.133
19	10.94	21.12	57.22	5.981	-11.050	9.925	-0.020	0.057	0.266
20	9.35	10.41	14.90	4.994	1.852	12.868	-0.072	-0.041	0.523
21	13.55	14.38	7.21	5.996	4.929	19.896	-0.061	-0.051	0.155
22	18.46	21.04	-23.74	2.004	18.983	25.975	-0.017	-0.017	0.012
23	18.56	20.18	15.88	8.005	1.002	27.990	-0.002	-0.023	0.007
24	19.66	21.05	13.55	7.997	2.089	30.086	0.046	0.006	-0.161
25	9.49	10.64	-13.40	2.995	8.888	12.905	-0.077	-0.020	0.191
26	12.49	16.03	34.61	6.992	-5.142	15.868	-0.042	0.008	0.467

reciprocal axes matrix **R**

0.015486	0.004218	-0.009478
0.016503	0.005236	0.008421
0.013334	-0.010972	0.000619

direct axes matrix **D**

22.2955	23.6353	19.8490
23.7965	31.6969	-66.8653
-58.4873	52.7221	2.6733

cell dimensions

a b c:	38.0749	77.7299	78.7879	volume
$\alpha\beta\gamma$:	89.059	90.092	90.919	233115.7

berg, 1983) can be useful to enhance the profile quality and thereby the accuracy of the data.

The method presented here does not need an initial cell nor does it require data manipulation. It accepts the reflection list as such, constructs direct-lattice vectors directly from the data, in the same process recognizing and avoiding contributions from alien or unreliable reflections, and produces a final primitive cell. It could be called a final-cell method.

Method

The distinction between alien reflections from other lattices or fragments and non-fitting reflections from inaccurate data is irrelevant in what follows, so we will use 'alien' for all sorts of reflections not belonging to a regular main lattice. Notions such as 'equal', 'integer' and 'fitting' are to be understood in the experimental, not the mathematical sense.

(*a*) *Generation of one-dimensional lattices*

'Triplets' are essential – a triplet is a triangle the corners of which are formed by the end points of three observed reflection vectors. (The origin 'reflection' is considered too.) As follows directly from the definition of the reciprocal lattice, the normal to a triplet defines the direction of a direct-lattice vector if the three reflections belong to the same reciprocal lattice. If we image such a direct vector in the vertical position, all reciprocal-lattice points will be seen in equidistant horizontal layers (similar to an axial rotation photograph on a cylindrical film, ignoring the increase of the interlayer distance with the level number). Therefore the projection of the regular observed reflection-vector end-points on the 'rotation axis' will result in a one-dimensional lattice. Alien reflections generally will end somewhere between these lattice points, or, by accident, close to a lattice point, so they are either recognized or harmless. A projection on the normal to a mixed triplet of regular and alien reflections will generally exhibit no period or an extremely small period, because this normal does not constitute a genuine direct-lattice-vector direction.

Example 3. *Primitive cell for an incommensurate structure after the second cycle, using only fitting reflections from the first cycle with a lower ACL*

IndexFit and especially LevelFit are set narrower to detect satellite reflections. Normal indexing procedures (and *DIRAX* with a high ACL) find a supercell fitting all reflections, with basic vectors (−1, −1, 0), (0, 1, −4), (6, −6, −1) and volume 3942.00 = 49 × 80.44. λ: 0.70930 Å; IndexFit: 0.08; LevelFit: 1/48; *D*max: 80; **t** vectors: 132; ACL: 13.

nF statistics
(*15* unique t-vectors with exactly 5 fitting reflections, etc.)

5	*15*	8	*12*	11	*6*	14	*40*
6	*11*	9	*9*	12	*3*	15	*16*
7	*8*	10	*5*	13	*7*		

nr	theta	phib	chib	h	k	l	dth	dom	dch
*1	8.43	-28.25	1.51	1.203	-0.203	1.447			
2	6.74	25.22	-6.62	0.000	0.999	1.000	-0.003	-0.026	0.025
3	6.94	-62.10	20.97	1.000	-1.000	0.999	-0.003	-0.023	-0.017
4	9.71	17.06	11.03	-0.000	0.999	1.999	-0.004	-0.002	0.006
5	8.04	-1.26	40.94	0.000	-0.000	1.999	-0.005	-0.011	0.000
6	9.86	42.32	33.39	-1.000	1.000	2.000	0.000	-0.013	-0.007
7	6.94	61.55	23.60	-0.999	0.999	0.999	-0.006	0.018	0.007
8	12.11	-1.26	40.94	0.000	-0.000	2.998	-0.007	-0.011	0.000
*9	8.44	-71.85	58.57	0.205	-1.205	1.447			
*10	13.20	58.14	50.52	-1.796	0.795	2.447			
11	12.65	69.78	45.51	-2.000	1.001	2.000	0.001	0.001	-0.012
12	12.13	-1.21	40.94	-0.001	0.001	3.003	0.013	0.027	0.000
13	13.41	-32.75	35.60	1.000	-1.000	3.001	0.002	0.009	-0.004
14	13.30	-13.61	19.17	1.000	0.000	3.000	0.000	0.001	-0.005
*15	13.40	-2.93	20.55	0.593	0.408	3.104			
16	13.30	12.67	19.84	-0.000	1.000	2.999	-0.003	0.007	-0.005
*17	11.42	-9.24	16.31	0.796	0.204	2.553			
18	9.72	-45.14	63.57	0.000	-1.001	2.000	0.003	-0.031	-0.000
*19	11.62	-43.93	58.11	0.204	-1.204	2.446			
20	13.99	-62.09	20.96	2.001	-2.000	1.998	-0.001	-0.014	-0.027
*21	13.20	34.82	13.92	-0.797	1.796	2.449			
*22	11.63	-21.20	14.28	1.205	-0.204	2.449			
23	9.72	17.07	11.03	-0.000	1.001	2.001	0.006	0.008	0.006
24	9.71	-17.53	10.14	1.000	-0.000	2.000	-0.001	-0.010	-0.005
*25	7.94	-12.30	4.03	0.796	0.204	1.552			

reciprocal axes matrix **R**

0.134462	-0.143490	0.003240
0.148448	0.148413	0.149005
-0.174793	-0.167593	0.129283

direct axes matrix **D**

3.5520	1.4485	-1.7585
-3.6387	1.4438	-1.5729
0.0855	3.8300	3.3185

cell dimensions

a b c:	4.2198	4.2188	5.0684	volume
αβγ:	90.002	89.957	116.946	80.44

Therefore interference from aliens is detected in this case as well.

(b) Calculation of direct vectors

Each line projection is searched for the one-dimensional lattice with the shortest period (above a given minimum) that represents the maximum number of projected reflections. In an iterating refinement with all cooperating reflections (*i.e.* projecting to a lattice point) the elementary reciprocal distance vector **d*** is established. This defines a potential direct-lattice vector **t**, with direction along **d*** and length $1/|\mathbf{d}^*|$. (Note that **d*** is a vector in reciprocal space but not a reciprocal-lattice vector.) The vector **t** is stored, together with the number of cooperating fitting reflections, n_F. It is called a potential direct vector, because finding a particular vector is not sufficient proof for it being a true direct vector from the lattice we are looking for. The triplet may consist of three aliens (or two and the origin) and then an alien vector may be found, or one or more

Example 4. *One axis much longer than the others and rather inaccurate data*

IndexFit and LevelFit are set wider than normal. Note that from the 1610 **t** vectors only 8 are supported by all (25) reflections. λ: 1.54056 Å; IndexFit: 0.20; LevelFit: 1/12; *D*max: 120; **t** vectors: 1610; ACL: 23.

nF statistics
(*288* unique t-vectors with exactly 5 fitting reflections, etc.)

5	*288*	8	*212*	11	*50*	14	*28*	17	*11*	20	*7*	23	*8*
6	*314*	9	*143*	12	*58*	15	*11*	18	*9*	21	*4*	24	*19*
7	*291*	10	*90*	13	*29*	16	*10*	19	*16*	22	*4*	25	*8*

nr	theta	phib	chib	h	k	l	dth	dom	dch
1	11.21	55.14	0.05	0.003	3.021	21.055	0.043	-0.095	0.081
2	12.36	58.43	-0.68	-0.002	3.008	23.993	0.004	-0.076	-0.023
3	11.65	56.25	-0.23	-0.000	3.037	22.138	0.092	-0.149	0.024
4	11.95	57.42	-0.49	-0.002	3.004	22.978	-0.004	-0.067	-0.028
5	12.22	33.73	4.04	0.003	5.003	15.992	0.003	-0.021	0.057
6	14.17	53.26	0.34	-0.000	4.003	25.950	-0.016	-0.072	0.007
7	13.27	78.13	-4.36	-0.004	1.011	28.971	-0.010	-0.112	-0.051
8	14.18	79.65	-4.43	0.008	0.892	31.030	-0.014	0.868	-0.027
9	9.91	86.60	-5.80	-0.003	0.008	21.966	-0.015	-0.102	-0.065
10	12.00	-50.12	29.92	1.001	3.988	-17.981	-0.021	-0.058	0.047
11	11.70	-48.35	30.49	1.000	3.992	-16.989	-0.015	-0.039	0.022
12	11.73	-29.23	30.69	1.001	4.994	-10.996	-0.008	-0.015	0.044
13	11.56	-26.90	30.81	0.995	5.007	-10.024	0.009	-0.023	-0.119
14	11.88	105.80	34.13	2.001	-0.997	18.975	-0.006	-0.003	0.055
15	12.40	93.81	33.73	1.999	0.001	20.990	-0.006	-0.006	0.003
16	13.26	102.86	29.31	1.999	-0.995	22.963	-0.016	-0.021	0.043
17	13.11	93.22	31.22	1.997	0.002	22.980	-0.014	-0.023	-0.008
18	14.18	-69.42	25.49	1.002	2.985	-27.000	-0.010	-0.111	0.041
19	12.38	-64.20	-30.91	-2.002	1.997	-18.999	0.004	-0.018	-0.045
20	12.30	-51.31	-30.14	-2.002	3.000	-16.010	0.007	-0.013	-0.014
21	12.28	-35.24	-29.70	-2.001	3.995	-11.047	0.005	-0.107	-0.003
22	12.57	-52.90	-29.16	-2.000	2.998	-17.013	0.001	-0.028	0.013
23	11.94	-30.83	-31.13	-2.001	4.003	-9.010	0.009	-0.005	0.005
24	12.69	-6.66	-29.54	-1.998	5.006	-0.013	0.003	-0.019	0.065
25	13.12	-42.65	-26.76	-2.000	3.994	-14.995	-0.009	-0.025	-0.022

reciprocal axes matrix **R**

-0.009634	0.001878	-0.010104
-0.013834	0.043713	0.000582
0.088099	0.007069	-0.001016

direct axes matrix **D**

-1.2006	-1.7192	10.9496
0.9207	22.2555	3.5954
-97.6575	5.7759	-9.7718

cell dimensions

a b c:	11.1486	22.5628	98.3150	volume
αβγ:	89.910	89.983	90.000	24730.51

aliens may by accident have been projected to an exact fraction of the true period and then the length of **t** is a multiple of the correct length or, finally, a mixed triplet may have resulted in a completely nonsensical vector.

The triplets are processed in a random sequence, interrupted at regular intervals for final cell calculations (described later). The randomness ensures equal opportunity for all reflections at the outset. This is especially effective if far more than 25 reflections have been measured (*e.g.* from an area detector), in which case a regular sequence might cause a set of fundamental triplets to be treated at the very end. With a random sequence the lattice may be obtained long before all triplets have been examined.

Those reflections which seem alien in one projection may well be regular in another. This situation is not uncommon with incommensurate structures and with twins. Therefore, reflections not fitting in one projection are not excluded from the following vector search processes. Each process stands on its own and may yield a potential direct vector **t**. Prior to the final cell calculations the **t** vectors are reduced to a unique

set by properly averaging vectors of the same length and with the same or opposite direction.

(*c*) *Final cell from the potential direct vectors*

Most **t** vectors reach only low n_F values, because the correct one-dimensional lattice is far from always being recognized and because the search for a **t** vector is not continued beyond a given limit.

The final cell is chosen from **t** vectors with high n_F values, *i.e.* from vectors supported by many reflections. 'Many' is quantified as ACL, the 'acceptance level'. The three shortest independent **t** vectors with $n_F \geq$ ACL determine the final direct cell. Usually, but not always (see Examples 1, 2, 3, 4), the highest occurring n_F can be taken as ACL. If this does not succeed lower ACLs are tried stepwise.

(*d*) *Indexing the reflections and refinement*

The complete list of reflections is indexed with the proposed final cell and reflections are accepted if three integer indices appear. The cell and orientation matrix are refined with all accepted reflections.

If there are many non-fitting reflections the total process may be repeated with the accepted reflections only, starting with a new set of triplets. This is recommended for incommensurate lattices, misfits and twins, where all reflections may fit in only one direction, which will lead to a non-primitive cell. The smaller correct primitive cell will be found by repeating with the reflections fitting in all directions. Because the bad influence of strange reflections is now eliminated, earlier rejected reflections may be accepted and *vice versa.*

The routine ends when the reflection list remains stable, *i.e.* with one all-fitting primitive lattice, or one main primitive lattice and a set of aliens. It is at this point that reflections may definitely be rejected. Checks for higher Bravais symmetry, identification of aliens or satellites, establishing of twin lattice relations and analysis of fragmentation are performed by other programs.

If the aliens do not constitute one single lattice, the main lattice is found even in the presence of a majority of outsiders. Here the qualification 'aliens' does not refer to the relative number of them but to a low degree of organization.

From 2600 triplets (obtained from 25 reflections plus the origin) the resulting number of useful direct vectors varies from several hundred, with ideal lattices (for which, in fact, *DIRAX* is not needed and any indexing method will do), to a few, as in Example 2. In the end we use only three of them, but it is not before the final calculation that we know which ones.

Formulas, definitions and parameters

The principle applied in *DIRAX* is very simple: to search for periodicity in directions perpendicular to triplets. This section explains practical details about the calculations as implemented in the program.

The unit normal **N** to the plane formed by the end points of three vectors $\mathbf{v}_i$, $\mathbf{v}_j$ and $\mathbf{v}_k$ is given by

$$\mathbf{N} = \frac{\mathbf{v}_i \times \mathbf{v}_j + \mathbf{v}_j \times \mathbf{v}_k + \mathbf{v}_k \times \mathbf{v}_i}{|\mathbf{v}_i \times \mathbf{v}_j + \mathbf{v}_j \times \mathbf{v}_k + \mathbf{v}_k \times \mathbf{v}_i|}$$

with $0 \leq i < j < k \leq P$, P being the number of observed reflections and $\mathbf{v}_0$ the origin 'reflection'. If $|\mathbf{v}_i \times \mathbf{v}_j + \mathbf{v}_j \times \mathbf{v}_k + \mathbf{v}_k \times \mathbf{v}_i| < 0.0001$ (an arbitrary 'zero') the triplet (i, j, k) is discarded.

In the **t** vector searching process no reciprocal distances shorter than $d^*_{\text{min}} = 1/d_{\text{max}}$ are tried. d_{max} should be about twice the maximum expected axis length, in order not to miss the longest axis (an overly large value leads to unnecessary calculations). **t** vectors with less than five supporting reflections are neglected.

The projections of the P reflection vectors onto **N** are defined as $a_p = \mathbf{v}_p \cdot \mathbf{N}$, with $p = 1, 2, \ldots, P$. The greatest distance between two consecutive a_p (after sorting) is called d^*_{start}. If $d^*_{\text{start}} > d^*_{\text{min}}$ residues are calculated (sum of absolute distances between a_p and the nearest proposed one-dimensional lattice point), for d^*_{start}/n, $n = 1, 2, \ldots$, as long as $d^*_{\text{start}}/n > d^*_{\text{min}}$ and $n < 13$. The n giving the lowest residue determines $d^* = d^*_{\text{start}}/n$. Finally, the **t** vector is found from $\mathbf{t} = \mathbf{N}/d^*$.

A projected reflection fits a one-dimensional lattice with elementary lattice vector $\mathbf{d}^*$ if its absolute distance to the nearest lattice point is less than $|\mathbf{d}^*|/24$. The adjustable parameter 1/24 is called 'LevelFit'. A reflection fits the final cell if all indices differ less than 0.1 ('IndexFit', adjustable) from an integer. The n_F statistics give the number of **t** vectors with exactly 5, 6, 7, ... *etc.* fitting reflections, from which the acceptance level ACL is derived as described before. The values for LevelFit and IndexFit are not very critical. From our tests, values of respectively 1/24 and 0.1 proved to be adequate and these are always tried by default. If necessary, with very stubborn cases, optimal values for LevelFit and IndexFit are found from trial and error (see Examples).

DIRAX exists in two stand-alone versions.

(1) One version for personal computers (IBM PC-AT and compatibles) written in Borland Turbo Pascal 5.5, for program development, delicate problems and demonstration, with ample graphical output on the color screen which greatly facilitates the inspection of data and of intermediate and final results. It takes from one to five minutes, dependent on the operator's (re)actions, on an Olivetti PCS 286 with mathematical co-processor Intel80287.

The up-to-date executable file DIRAX.EXE is available from the author (e-mail: duisenberg@HUTRUU54.BITNET).

(2) Another version is for practical use with twins, fragments, misfits, ill-shaped reflections *etc.*, written in Fortran77 and running under DEC VMS 4.7 on the CAD4's Digital Equipment Corporation MicroVAXII. It reads and writes directly from and to reflection files and requires about 60 s CPU time.

Delft Instruments (Enraf–Nonius) intends to integrate this version into the CAD4 program system, but it can already be obtained from the author in its present form.

Both versions expect an (inter)active user. We do not feel this is a disadvantage. On the contrary, the sort of problems *DIRAX* solves is often of a specialized crystallographic nature, where an expert's supervision is desirable.

Explanation of the examples

The observed setting angles for a reflection vector are expressed in bisecting mode: θ, φ_B, χ_B (**theta**, **phib**, **chib** in the Examples, as in the program output), comparable to polar coordinates. These are more easily visualized than Eulerian angles, not to mention CAD4 angles.

The bisecting angles transform to a Cartesian system XYZ (X to the X-ray source, Z to the Zenith and Y completing the right-handed sYstem; CAD4 convention) as

$$c_x = -s \cos \chi_B \sin \varphi_B,\ c_y = +s \cos \chi_B \cos \phi_b,$$

$$c_z = +s \sin \chi_B,$$

with $s = 2(\sin \theta)/\lambda$, the length of the reflection vector.

The final reciprocal and direct orientation matrices are $\mathbf{R}$ and $\mathbf{D} = \mathbf{R}^{-1}$, respectively.

The fractional **h**, **k**, **l** (given for non-fitting reflections too) are obtained from $\mathbf{h} = \mathbf{Dc}$, with **h** and **c** the vector representation of (**h**, **k**, **l**) and (c_x, c_y, c_z) respectively.

The deviations $\Delta\theta$, $\Delta\omega$, $\Delta\chi$ (**dth**, **dom**, **dch** in the Examples) refer to the differences between the observed reflection angles and those calculated with integer **h**, **k**, **l**, for fitting reflections. The great-circle angle $\Delta\omega$ is given instead of $\Delta\varphi$, because it is independent of χ: $\Delta\omega = \Delta\varphi \cos \chi_B$.

An asterisk indicates a non-fitting reflection.

The resulting cell is primitive, higher Bravais symmetry is ignored.

IndexFit: a reflection fits the final main lattice if all indices differ less than IndexFit from an integer.

LevelFit: a reflection fits a reciprocal level (parallel to a triplet) if its fractional distance to that level is less than LevelFit.

*D*max: direct vectors longer than *D*max are not calculated.

ACL: for the final cell calculation only **t** vectors supported by at least ACL reflections were used.

nF statistics (see Example 1): '6 32' means: 32 unique **t** vectors were found each with exactly six fitting reflections, '16 -' means: no **t** vectors with exactly 16 fitting reflections were found *etc.* (Which reflections did fit a particular **t** vector is not shown in the table, but with the *DIRAX* PC version is shown in optional screen pictures. Exploitation of this knowledge for particular problems is a subject for further studies.)

Concluding remarks

DIRAX proves to be a robust easily manageable indexing program suitable for a wide variety of indexing problems. It extracts the maximum of information from the input with the minimum of assumptions. No reflections are excluded at the outset and no initial cell is used.

The high level of performance of the method must be attributed to the many checks for consistency of intermediate results with the whole body of data taken before drawing conclusions.

I thank Dr Jan Louwert de Boer from the University of Groningen for stimulating and enlightening discussions, especially concerning problems such as Example 3 for which, moreover, he kindly supplied the data. The Fortran version of the program was written by Dr A. M. M. Schreurs of our laboratory.

References

CAD4 Manual. (1989). Enraf–Nonius, Scientific Instruments Division, PB 483, NL-2600 AL, Delft, The Netherlands.

CLEGG, W. (1984). *J. Appl. Cryst.* **17**, 334–336.

DUISENBERG, A. J. M. (1983). *Acta Cryst.* A**39**, 211–216.

JACOBSON, R. A. (1976). *J. Appl. Cryst.* **9**, 115–118.

SPARKS, R. A. (1976). In *Crystallographic Computing Techniques*, edited by F. R. AHMED, K. HUML & B. SEDLÁCEK, pp. 452–467. Copenhagen: Munksgaard.

SPARKS, R. A. (1982). In *Computational Crystallography*, edited by D. SAYRE, pp. 1–18. Oxford Univ. Press.