

# A new parallel version of a dichotomy based algorithm for indexing powder diffraction data

Ivan Šimeček\*, Aleksandr Zaloga† \*Department of Computer Systems, Faculty of Information Technology, Czech Technical University in Prague, Prague, Czech Republic †Department of Chemistry, Siberian Federal University, Krasnoyarsk, Russia

(Original Paper)

**Abstract**—One of the key parts of the crystal structure solution process from powder diffraction data is the determination of the lattice parameters from experimental data shortly called indexing. The successive dichotomy method is the one of the most common ones for this process because it allows an exhaustive search. In this paper, we discuss several improvements for this indexing method that significantly reduce the search space and decrease the solution time. We also propose a combination of this method with other indexing methods: grid search and TREOR. The effectiveness and time-consumption of such algorithm were tested on several datasets, including orthorhombic, monoclinic, and triclinic examples. Finally, we discuss the impacts of the proposed improvements.

**Index Terms**—indexing powder diffraction, dichotomy algorithm, TREOR algorithm, grid search algorithm.

## I. INTRODUCTION

The unit-cell dimensions (also called lattice parameters)  $(a, b, c, \alpha, \beta, \gamma)$  determine the positions of peaks in a powder diffraction pattern. The aim of indexing is to determine the correct lattice parameters by using the information on the peak positions in the experimental pattern. Although there are many implementations, the number of existing methods is relatively small:

- Monte Carlo method (simulated annealing search) — McMaille [1],
- ITO [2],
- SVD-Index [3],
- genetic algorithm [4],
- Conograph [5], [6],
- trial-and-error methods — TREOR [7], [8]
- grid search [9],
- dichotomy methods — DICVOL [10], X-Cell [11]).

This paper aimed mainly at the last one method (and the combination of the last three methods). This method represented by two main implementations has been continuously improved from its beginning to the current state, for example, DICVOL package has evolved into DICVOL91 [12], then into DICVOL04 [13], then into DICVOL06 [14], then into DICVOL14 [15], but even the newest version inherit some drawbacks of their predecessors.

### A. Related works

As far as we know, the most related software to our work, X-Cell and DICVOL14 traverse the search space inefficiently,

so it is difficult to use them in very complex indexing situations (high number of spurious lines or low symmetries).

### B. Used technologies

Brute force algorithms need significant computing power. Such power can be attained by parallel computing and the paper considers shared memory environment with OpenMP [16], [17]. OpenMP is a cross-platform standard for parallel processing. The OpenMP API specification is defined as a collection of compiler directives, library routines, and environment variables extending the C, C++, and Fortran programming languages. They can be used to create portable parallel (multithreaded) programs utilizing shared memory.

## II. CURRENT STATE-OF-ART

### A. Description of the powder pattern

The measured powder diffraction pattern represents the reciprocal space and contains information about dimensions of the reciprocal unit cell. Each position of the observed diffracted line  $\theta_{hkl}$  corresponds to one or more inter-planar distance  $d_{hkl}$  of the reciprocal space, where  $h, k$  and  $l$  are Miller indices (small integer numbers). The relationship between the inter-planar spacing  $d_{hkl}$  of the crystal lattice and the position of the diffracted line  $\theta_{hkl}$  in the powder pattern is ruled by Bragg's law:

$$n\lambda = 2d_{hkl} \sin \theta_{hkl}, \quad (1)$$

where

- $n$  is an integer that denotes the order of reflection (it is usually assumed to be 1),
- $\lambda$  is the wavelength of the incident ray,
- $d_{hkl}$  is the inter-planar spacing.

Let  $a^*, b^*, c^*, \alpha^*, \beta^*, \gamma^*$  denote the parameters of the reciprocal unit cell. The relationship between these parameters and the inter-planar distance  $d_{hkl}$  is:

$$1/d_{hkl}^2 = a_{11}h^2 + a_{22}k^2 + a_{33}l^2 + a_{12}hk + a_{23}kl + a_{13}hl, \quad (2)$$

where

$$\begin{aligned} a_{11} &= (a^*)^2, & a_{22} &= (b^*)^2, & a_{33} &= (c^*)^2, \\ a_{12} &= 2a^*b^* \cos \gamma^*, & a_{23} &= 2b^*c^* \cos \alpha^*, & a_{13} &= 2a^*c^* \cos \beta^*. \end{aligned}$$

## B. Indexing

The direct lattice parameters (unit-cell dimensions)  $a, b, c, \alpha, \beta, \gamma$  define the parameter space of the problem and the goal of the indexing process is to find their correct values (or values of a subset of these parameters, depending on the crystal system) for a given experimental powder diffraction pattern. For powder indexing, the expression  $1/d_{hkl}^2$  is usually denoted by  $Q_{hkl}$ . So, the input data are usually transformed using the Eq. 1 into  $Q_{hkl}$  values for each selected observed diffraction line of the pattern.

## C. Terminology and notation

In the following text, we will use this terminology and notation:

- The parameter  $N$  denotes the number of input reflexes.
- The parameter  $DOF$  denotes the number of unknown lattice parameters (degree of freedom). Depending on the crystal system, some values  $a_{ij}$  are zero. For example, in the cubic system, the expressions (1) and (2) are simplified to:

$$Q_{hkl} = (h^2 + k^2 + l^2)/a^2.$$

- The parameter  $max_{imp}$  denotes the number of impurities in the input reflexes.
- To distinguish between different types of values of  $Q$ , we further denote by  $Q^{calc}$  only the values of  $Q$  obtained from the theoretical model and by  $Q^{obs}$  the measured (observed) values of  $Q$ .
- A box  $B \subseteq \mathbb{R}^s$  is defined as a product of closed real intervals. Thus, every box is characterized by  $2 \cdot s$  real numbers  $B_{1\dots s}^L$  and  $B_{1\dots s}^U$ ,  $B_k^L$  is the lower bound in  $k$ th dimension, and  $B_k^U$  is the upper bound in  $k$ th dimension. We assume only nonempty boxes, thus  $\forall k \in \{1 \dots s\}, B_k^L < B_k^U$ .

## III. METHODS FOR INDEXING

### A. Grid search method

The grid search method is the simplest brute force indexing method based on systematic testing of search space. The following study [18] had demonstrated that brute force algorithms could better solve some complex problems in comparison to the old fine tuned sophisticated algorithms.

### B. TREOR method

The basic idea of the TREOR method (for details see [7], [8]) follows: the reciprocal cell relationship from Eq.(2) may be written as

$$M\vec{A} = \vec{Y},$$

where

- $M$  is a square matrix containing the Miller indices (derived from  $hkl$  triplets),
- $\vec{A}$  is a vector containing the parameters of the reciprocal unit cell,
- $\vec{Y}$  is a vector with  $DOF$  values from  $Q^{obs}$ .

The dimensions of all vectors and matrices are equal to  $DOF$ , and the process of indexation can be solved by this SLE, but the matrix  $M$  and the vector  $\vec{Y}$  are unknown, so we must try all possible permutations.

### C. Successive dichotomy method

The successive dichotomy (SD) method can be viewed as an application of interval arithmetic [19], [20] on the indexing process.

The key idea of the SD method is simple:

- 1) It starts with one initial box that is derived from the user-defined intervals of unit cell parameters.
- 2) The current box is tested if it can or cannot contain solutions.
  - If it cannot, then the current box (called ‘‘unfavourable volume element’’) is rejected from further investigation.
  - If it can, then the current box (called ‘‘favourable volume element’’) is divided into smaller boxes and assigned for further investigation.
- 3) The whole process (point 2) recursively repeats until one or all solutions are found.

The whole process of the SD method can be illustrated by a dichotomy tree (DT). To precise the aforementioned algorithm, we define the following functions:

- *Generate*( $B, HKL$  set) that for the given box  $B$  and the set of possible  $hkl$  trials ( $HKL$  set) returns the set of  $Q$ .
- *Verify*( $Q^{calc}, Q^{obs}$ ) that for the given sets  $Q^{calc}$  and  $Q^{obs}$  verifies whether or not the  $Q^{calc}$  may contain an indexing solution i.e., at least  $n - max_{imp}$  intervals of  $Q^{obs}$  is overlapped by intervals from  $Q^{calc}$ . This function returns a logical value.
- *Divide*( $B$ ) that for the given box  $B$  returns the set of sub-boxes (dissection of the box  $B$ ).
- *SmallEnough*( $B$ ) that for the given box  $B$  returns if the widths of the box  $B$  are ‘‘small enough’’. This function returns a logical value.
- *CheckForSolution*( $B$ ) that for the given (small) box  $B$  returns if the box is the indexing solution. This function returns a logical value.

With the mentioned functions, the algorithm for SD can be rewritten as follows:

---

#### Algorithm 1 Dichotomy base idea

---

```

1: procedure DICHOTOMY1
Input:  $B$  = initial box
Input:  $HKL$  set = set of possible  $hkl$  trials
Output:  $solution$  = set of the indexing solutions
2:   push  $B$  onto the stack
3:   while  $solution$  is not found AND stack is not empty
   do
4:     pop  $B$  from the stack    ▷  $B$  is the current box
5:      $Q^{calc} \leftarrow Generate(B, HKL\ set)$ 
6:     if ( $Verify(Q^{calc}, Q^{obs})=true$ ) then    ▷ the box  $B$ 
   can contain solution
7:       if ( $SmallEnough(B)=true$ ) then
8:         if ( $CheckForSolution(B)=true$ ) then
9:            $solution \leftarrow solution \cup B$ 
10:      else
11:         $B_{new} \leftarrow Divide(B)$ 
12:        push all  $B_{new}$  onto the stack
13:   return  $solution$  or ‘‘solution is not found’’

```

---

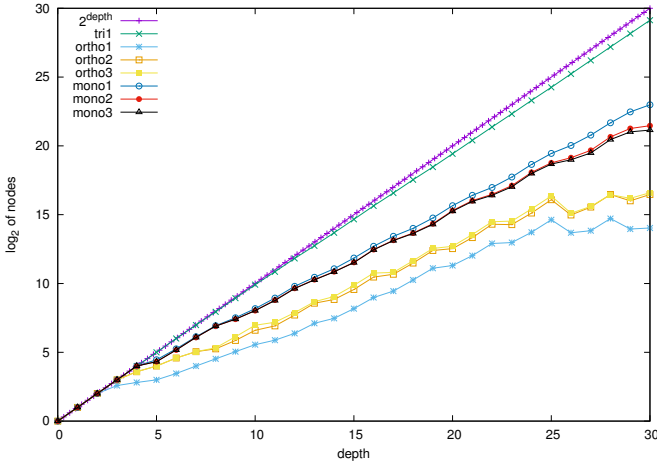


Fig. 1. The number of nodes in the given depth (dependence of the crystal system).

Figure 1 illustrated the number of nodes (of dichotomy tree) in the given depth of the tree. Please note the exponential growth of nodes (mainly for low-symmetric systems). Therefore, the recursion depth must be regulated.

1) *Existing improvements:* In previous dichotomy-based implementations are several improvements of the basic Algorithm 1:

- The set of  $hkl$  trials (denoted as  $HKL$  set) is not fixed. If any trial doesn't overlap any  $Q^{obs}$ , it is useless to test it again in child nodes (in deeper levels of dichotomy tree). To optimize the solution process, we move to/from stack not only boxes but also reduced subset of the initial  $HKL$  set.
- There is an assumption that the first  $f$  lines are indexes with a limited set of  $hkl$  trials (e.g.,  $|h|, |k|, |l| \leq c_1$  or  $|h| + |k| + |l| \leq c_2$ , where  $c_1$  and  $c_2$  are chosen constants). We decide not to include this feature. It can dramatically accelerate the indexing process (if it is set up properly), but we lost the main property of the dichotomy method: exhaustive search.
- The following optimization was introduced in X-Cell implementation. Every  $hkl$  trial can index at most one reflection. Consequently, if the cardinality of the  $HKL$  set is less than  $n - max_{imp}$ , then this box is rejected during the execution of  $Verify(Q^{calc}, Q^{obs})$ .

2) *Unsolved questions:* The key idea is relatively simple, but there are several aspects that can be improved. In Section IV, we will discuss the following questions and explain the theoretical consequences of possible solutions and measure their impacts.

- In what space we should operate? (Real or reciprocal, see Section IV-C)
- When the recursion should be stopped? (see Section IV-E)
- How to test if the current box can or cannot contain solutions? (see Section IV-D)
- Sometimes is better to switch to another approach to dissection (see Section IV-F)

## IV. OUR NEW IMPROVEMENTS

In this section, we discussed the impact of our improvement (VI) to existing implementations (X-Cell or DICVOL) proposed in Section III-C2.

### A. Experiments configuration

We have implemented all algorithms in C/C++ using OpenMP.

#### 1) HW and SW configurations:

- Testing configuration 1: Some experiments were performed on a small university cluster called "star". The execution times were measured on a server with following HW and SW parameters:

- $2 \times$  CPU Intel Xeon Processor E5-2620 v2,
- CPU cores: 6 per CPU, 12 in total,
- Memory size: 32 GB RAM,
- OS Linux, C++ compiler (g++) version 4.8.3

#### 2) Indexing data files:

The following data files were used:

- Ortho1: test3b.dat ( $B_5H_{12}NO_{12}$ , CIF 4065373,  $N = 20$ , the orthorhombic crystal system, the correct solution:  $a = 11.304\text{\AA}$ ,  $b = 11.02\text{\AA}$ ,  $c = 9.152\text{\AA}$ , published in [21],
- Ortho2: test1b.dat ( $Cd_3(OH)_5(NO_3)$ ,  $N = 20$ , the orthorhombic crystal system, the correct solution:  $a = 3.4203(3)\text{\AA}$ ,  $b = 10.0292(6)\text{\AA}$ ,  $c = 11.0295(6)\text{\AA}$ , published in [22],
- Mono1: cim.dat (Cimetidine,  $C_{10}H_{16}N_6S$ ,  $N = 21$ , the monoclinic crystal system, the correct solution:  $a = 6.821(1)\text{\AA}$ ,  $b = 18.818(3)\text{\AA}$ ,  $c = 10.374(2)\text{\AA}$ ,  $\beta = 106.42(1)^\circ$ , published in [23],
- Mono2: CSA.dat (TECRIC : Cyclosporin A dimethylisorbide solvate, Deposition Number: 1268484,  $N = 24$ , the monoclinic crystal system, the correct solution:  $a = 15.521(2)\text{\AA}$ ,  $b = 20.833(3)\text{\AA}$ ,  $c = 12.949(3)\text{\AA}$ ,  $\beta = 100.21(1)^\circ$ , published in [24],
- Mono3: Taxol.dat ( $C_{45}H_{49}NO_{13} \cdot 3(C_4H_8O_2)$ ,  $N = 20$ , the monoclinic crystal system, the correct solution:  $a = 16.329(2)\text{\AA}$ ,  $b = 17.704(2)\text{\AA}$ ,  $c = 17.504(1)\text{\AA}$ ,  $\beta = 100.61(1)^\circ$ , published in [25],
- Tri1: cap\_3-100\_mcmly.dat ( $C_{22}H_{28}CuIN_2O_4$ ,  $N = 48$ , the triclinic crystal system, the correct solution:  $a = 5.164309\text{\AA}$ ,  $b = 9.46518\text{\AA}$ ,  $c = 11.38433\text{\AA}$ ,  $\alpha = 73.173^\circ$ ,  $\beta = 89.1931^\circ$ ,  $\gamma = 87.91^\circ$ ,

3) *Initial parameters:* We have used the following values as initial:

- all lengths are between  $3\text{\AA}$  and  $30\text{\AA}$ .
- all angles are between  $90^\circ$  and  $120^\circ$ .
- minimal cell volume  $V_{min} = 30\text{\AA}^3$  and maximal cell volume  $V_{max} = 3000\text{\AA}^3$ .
- initial  $hkl$ -triplet set contains 1000 elements.

### B. Evaluation of general results

In this Section, we evaluate general results of the successive dichotomy method.

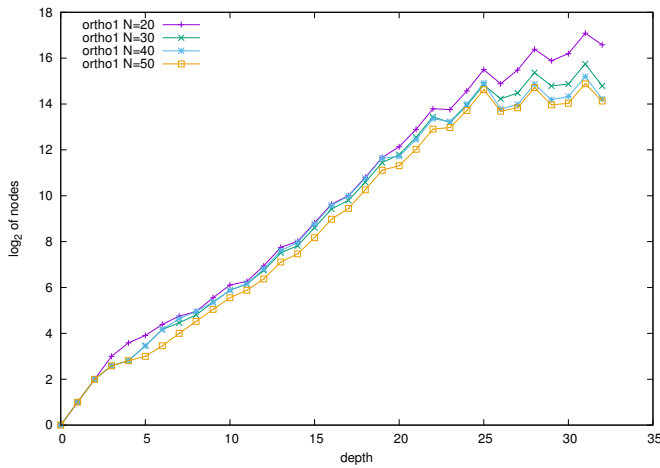


Fig. 2. The number of nodes in the given depth for `ortho1` (dependence of the number of input reflexes).

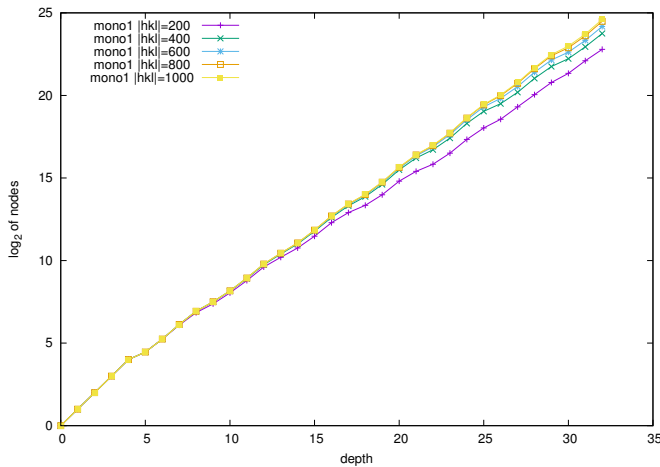


Fig. 3. The number of nodes in the given depth for `mono1` (dependence of the number of assumed  $hkl$  trials).

1) *Impact of the number of lines:* We can change the number of input lines. The results are shown in Figure 2. Obviously, with growing  $N$  (i.e., more conditions are checked within  $Verify(Q^{calc}, Q^{obs})$  in procedure `Dychotomy1`, we get fewer nodes in the DT.

2) *Impact of the number of HKL trials:* We can change the number of assumed initial  $hkl$  trials. For orthorhombic and monoclinic cells, the values (the maximal values of  $hkl$ -triplets) used for time measurement are much higher than the minimal possible values. The results are shown in Figure 3. Obviously, with growing  $|hkl|$  (i.e., it is easier to satisfy conditions within  $Verify(Q^{calc}, Q^{obs})$ , we get more nodes in the DT.

3) *Impact of the “first lines” optimization:* We have also measured the effect “first lines” optimization, but as was discussed in Section III-C1, we don’t include this optimization.

If we apply the following restrictions on the first 9 lines (reflexes) on `mono1`:

$$h, k \leq 3, \quad |l| \leq 1, \quad h + |k| + |l| \leq 4$$

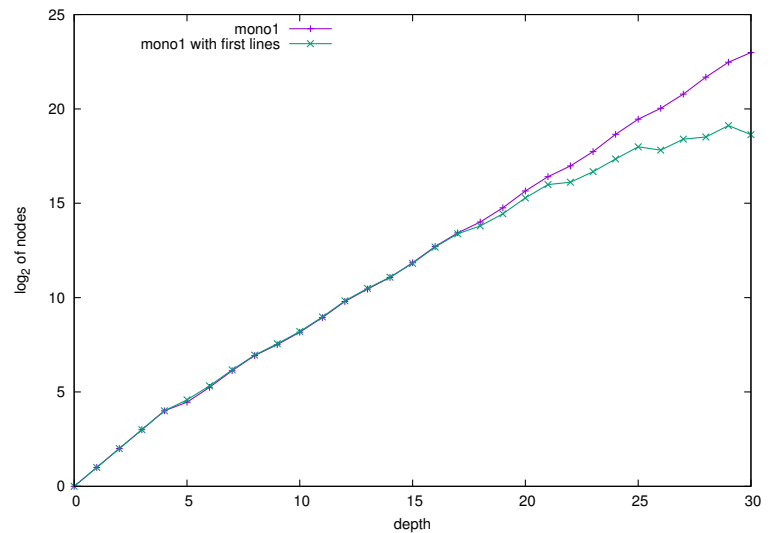


Fig. 4. The number of nodes in the given depth for `mono1` (impact of “first lines” optimization).

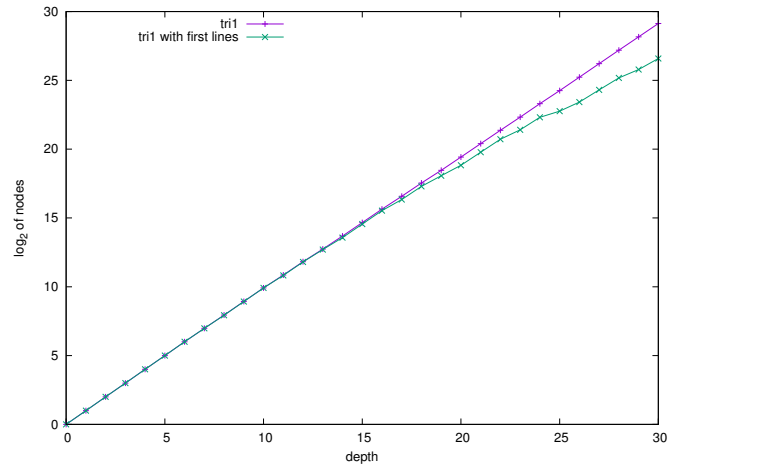


Fig. 5. The number of nodes in the given depth for `tri1` (impact of “first lines”).

Results are shown at Figure 4. With “first lines” optimization we get 20 times less nodes at depth=30.

If we apply the following restrictions on the first 6 lines (reflexes) on `tri1`:

$$h \leq 1, \quad |k| \leq 1, \quad |l| \leq 2, \quad h + |k| + |l| \leq 2$$

Results are shown at Figure 5. With “first lines” optimization we get 5 times less nodes at depth=30.

### C. Search space

As far as we know, DICVOL uses for searching the direct space, X-Cell uses the reciprocal space. The big advantage of real space usage is the transparency to the user and the fairness of dissection. Let us assume an example of a cubic system for an explanation of this phenomenon. We know that the correct solution ( $a$ ) is in  $[2, 30]$ . We also assume that the probability that any value in this interval is equal to the correct solution is the same along with the whole interval.

If we divide the starting interval in real space, we get two intervals  $[2, 16]$  and  $[16, 30]$  with the same probability that the solution lies in. If we divide the starting interval in reciprocal space ( $[2, 30]$  is transformed to  $[1/30^2, 1/2^2]$ ), we get two intervals  $[1/30^2, 113/900]$  and  $[113/900, 1/2^2]$  that is equal to  $[2, 2.822]$  and  $[2.822, 30]$  in real space. So, the probability for the first interval (that the solution lies in) is about 2.9% and about 97.1% for the second interval.

On the other hand, the advantages of reciprocal space are the following:

- It eliminates the overhead of transformation (from real to reciprocal and vice versa) because reciprocal values are needed to decide if the box would be rejected or not in function  $VERIFY(Q^{calc}, Q^{obs})$ .
- It “convergent” more quickly because transformation (from real to reciprocal) increases the “volume” of boxes. For the demonstration, let us assume an example of a monoclinic system, the input box  $B$  with the parameters  $a$ ,  $b$ , and  $c$  are in  $[2, 10]$ , and  $\beta$  is in  $[45^\circ, 90^\circ]$ . The dissection in real space results in 16 boxes. We will discuss further only the first two boxes:  $[2, 6] \times [2, 6] \times [2, 6] \times [45^\circ, 67.5^\circ]$  and  $[2, 6] \times [2, 6] \times [2, 6] \times [67.5^\circ, 90^\circ]$ . Since  $a_{11} = \frac{1}{a^2 \sin^2(\beta)}$  these two intervals are transformed into two intervals in reciprocal space for  $a_{11}$ :  $[0.03254, 0.5]$  and  $[0.02778, 0.2928]$ . Please note that these intervals are not disjoint<sup>1</sup>. Therefore, the efficiency of searching is decreased.

Based on this observation, we decide to mix both approaches: on the top level, we dissect (divide) boxes in real space (to satisfy fairness of dissection), and in deeper levels, we divide boxes in reciprocal space (to increase the efficiency of traversing).

Table I illustrated the impact of this mixed strategy on the number of nodes of DT. The optimal value is if the number of layers (dissected in real space) is approximately equal to the number of layers (dissected in reciprocal space).

1) *Sensitivity of indexing*: Since it is very heavy or impossible to derive the value of the proper maximal depth of DT, we have done an experiment:

- 3 data files, each with different crystal system (Ortho, Mono, Tri),
- We know the exact correct solution for each data file,
- We want to evaluate the size of “favourable volume element” in every dimension: We start with correct parameters. Then, we decrease or increase cell parameters until the current box is changed into “unfavourable volume element”. In other words,  $DL$  and  $DU$  represent the maximal difference from correct solution while unit cell parameters are still valid, i.e., all lines are indexed.

Tables II and III can be used to estimate the proper maximal depth of DT (denoted by  $depth_{min}$ ) with dissection by the following equation:

$$depth_{min} = \log_{arity} \prod \frac{U(p_i) - L(p_i)}{DL(p_i) + DU(p_i)} \quad p_i \in \{a_{11}, \dots, a_{33}\},$$

<sup>1</sup>Strictly speaking the intervals in reciprocal space are not also disjoint but they share only one point.

where

- *arity* is the arity of the DT.

The results for real space and  $arity = 2$ :

- $depth_{min}(\text{orthorhombic}) = 22.7$ ,
- $depth_{min}(\text{monoclinic}) = 27.1$ ,
- $depth_{min}(\text{triclinic}) = 38.2$ ,

The results for reciprocal space and  $arity = 2$ :

- $depth_{min}(\text{orthorhombic}) = 29.1$ ,
- $depth_{min}(\text{monoclinic}) = 36.4$ ,
- $depth_{min}(\text{triclinic}) = 52.8$ ,

These values are rather optimistic since we change each parameter separately. But we can conclude that the SD method is (without further optimizations) unreal for a triclinic case. And we can also conclude that the depth of DT in real space is lower than in reciprocal. This is another significant reason to use a mixed strategy (see SectionIV-C).

#### D. Reduction of intervals

In our implementation of function  $VERIFY(Q^{calc}, Q^{obs})$ , we not only test the box would be rejected or not, but we also try to reduce the widths of intervals. It can be done if one reflection  $Q_i^{obs}$  is overlapped by exactly one  $hkl$  trial (we call this situation a **unique overlap**). For the sake of simplicity, we assume that the crystal system is orthorhombic, so the following statement is valid:

$$[a_{11}^L, a_{11}^H]h^2 + [a_{22}^L, a_{22}^H]k^2 + [a_{33}^L, a_{33}^H]l^2 \in [q_i^L, q_i^H]$$

Under the assumption that  $h \neq 0$ , we can compute temporary values of  $a_{11}^L$  and  $a_{11}^H$  as follows:

$$temp\_a_{11}^L = \frac{q_i^L - a_{22}^H k^2 - a_{33}^H l^2}{h^2}$$

$$temp\_a_{11}^H = \frac{q_i^H - a_{22}^L k^2 - a_{33}^L l^2}{h^2}$$

and similarly for  $temp\_a_{22}^L$ ,  $temp\_a_{22}^H$ ,  $temp\_a_{33}^L$ , and  $temp\_a_{33}^H$ . Then, we can try to reduce intervals:

$$[new\_a_{11}^L, new\_a_{11}^H] = [temp\_a_{11}^L, temp\_a_{11}^H] \cap [a_{11}^L, a_{11}^H]$$

The second effect of unique overlap is caused by the fact that every  $hkl$  trial can index at most one reflection. Consequently, if for  $q_i$  unique overlap by the  $hkl$  trial occurs, then this  $hkl$  trial is removed from the  $HKL$  set for all other lines.

#### E. Early solution

The existing implementations try to solve the box  $B$  only if the maximal level of dichotomy tree is reached or if  $B$  is small enough (function  $SmallEnough(B)$  in Algorithm 1). We additionally try to solve if the number of unique overlaps is equal or higher to  $DOF$ , so these unique overlap construct system of linear equations (SLE) in the form (for an orthorhombic system):

$$h^2(a_{11}^L + a_{11}^H)/2 + k^2(a_{22}^L + a_{22}^H)/2 + l^2(a_{33}^L + a_{33}^H)/2 = Q^{obs} \quad (3)$$

We use Gramm-Schmidt or LSQ method to solve such SLE. It corresponds to TREOR indexing algorithm (see Sec. III-B). If this method successfully finds the solution (i.e., the number

TABLE I  
EFFECT OF DEPTH TO SWITCH FROM REAL TO RECIPROCAL SPACE FOR MONO1.

no. of layers in real no. of layers in reciprocal	0	4	8	12	16	20	24	28
no. of nodes in DT	$5.60 \cdot 10^7$	$1.21 \cdot 10^8$	$9.83 \cdot 10^7$	$5.50 \cdot 10^7$	$4.10 \cdot 10^7$	$4.04 \cdot 10^7$	$4.6 \cdot 10^7$	$1.66 \cdot 10^8$
time to solve [s]	235	592	588	222	146	132	213	445

TABLE II  
SENSITIVITY DATA FOR REAL SPACE.

parameter	$L(\text{ortho1})$	$U(\text{ortho1})$	$L(\text{mono1})$	$U(\text{mono1})$	$L(\text{tri1})$	$U(\text{tri1})$
$a$ [Å]	3	30	3	30	3	30
$b$ [Å]	3	30	3	30	3	30
$c$ [Å]	3	30	3	30	3	30
$\alpha$ [°]					90	120
$\beta$ [°]			90	120	90	120
$\gamma$ [°]					90	120
parameter	$DL(\text{ortho1})$	$DU(\text{ortho1})$	$DL(\text{mono1})$	$DU(\text{mono1})$	$DL(\text{tri1})$	$DU(\text{tri1})$
$a$ [Å]	0.04877	0.01709	0.07856	0.07141	0.02502	0.02502
$b$ [Å]	0.04434	0.1265	0.1150	0.1265	0.08641	0.06492
$c$ [Å]	0.1852	0.07856	0.04031	0.07856	0.07856	0.06492
$\alpha$ [°]					0.2983	0.6395
$\beta$ [°]			0.6395	0.3282	0.8512	0.5285
$\gamma$ [°]					0.5814	0.6395

TABLE III  
SENSITIVITY DATA FOR RECIPROCAL SPACE.

parameter	$L(\text{ortho1})$	$U(\text{ortho1})$	$L(\text{mono1})$	$U(\text{mono1})$	$L(\text{tri1})$	$U(\text{tri1})$
$a_{11}$	$1.111 \cdot 10^{-3}$	$111.1 \cdot 10^{-3}$	$1.111 \cdot 10^{-3}$	$148.1 \cdot 10^{-3}$	$1.111 \cdot 10^{-3}$	$222.2 \cdot 10^{-3}$
$a_{12}$			0	$148.1 \cdot 10^{-3}$	0	$222.2 \cdot 10^{-3}$
$a_{13}$					0	$222.2 \cdot 10^{-3}$
$a_{22}$	$1.111 \cdot 10^{-3}$	$111.1 \cdot 10^{-3}$	$1.111 \cdot 10^{-3}$	$111.1 \cdot 10^{-3}$	$1.111 \cdot 10^{-3}$	$222.2 \cdot 10^{-3}$
$a_{23}$					0	$222.2 \cdot 10^{-3}$
$a_{33}$	$1.111 \cdot 10^{-3}$	$111.1 \cdot 10^{-3}$	$1.111 \cdot 10^{-3}$	$148.1 \cdot 10^{-3}$	$1.111 \cdot 10^{-3}$	$222.2 \cdot 10^{-3}$
parameter	$DL(\text{ortho1})$	$DU(\text{ortho1})$	$DL(\text{mono1})$	$DU(\text{mono1})$	$DL(\text{tri1})$	$DU(\text{tri1})$
$a_{11}$	$87.8 \cdot 10^{-6}$	$0.2505 \cdot 10^{-3}$	$0.1414 \cdot 10^{-3}$	$0.1414 \cdot 10^{-3}$	$0.3668 \cdot 10^{-3}$	$0.36683 \cdot 10^{-3}$
$a_{12}$			$0.1285 \cdot 10^{-3}$	$0.1169 \cdot 10^{-3}$	$0.3668 \cdot 10^{-3}$	$0.4883 \cdot 10^{-3}$
$a_{13}$					$0.4035 \cdot 10^{-3}$	$0.3668 \cdot 10^{-3}$
$a_{22}$	$0.1169 \cdot 10^{-3}$	$41.0 \cdot 10^{-6}$	$37.2 \cdot 10^{-6}$	$33.8 \cdot 10^{-6}$	$0.1414 \cdot 10^{-3}$	$0.2278 \cdot 10^{-3}$
$a_{23}$					$0.1556 \cdot 10^{-3}$	$0.2278 \cdot 10^{-3}$
$a_{33}$	$13.0 \cdot 10^{-6}$	$30.8 \cdot 10^{-6}$	$0.4883 \cdot 10^{-3}$	$0.3032 \cdot 10^{-3}$	$96.6 \cdot 10^{-6}$	$0.1169 \cdot 10^{-3}$

of linearly independent equations is equal to  $DOF$ ) then we test if the solution is the correct indexing solution and cancel this branch of DT.

#### F. Different approach to dissection

We have also observed that for some boxes, the dissection does not reduce the numbers of  $hkl$  trials overlapping  $q$  intervals and methods mentioned in Sections IV-D or IV-E can not be applied. These boxes usually reached the maximal depth of recursion (and the number of nodes in the dichotomy tree grows rapidly). The alternative solution (in this situation) is that we do not dissect the box, but we construct the child nodes as unique overlaps. For example, we assume that one line  $q_i$  is overlapped by 3  $hkl$  trials:  $(h_1, k_1, l_1)$ ,  $(h_2, k_2, l_2)$ , and  $(h_3, k_3, l_3)$ . The next level of dichotomy tree consists of 3 child nodes: in the first node, the line  $q_i$  is assumed be overlapped by only  $(h_1, k_1, l_1)$  trial, and so on. Since the

unique overlap has two effects (see Section IV-D), it can reduce the number of nodes that reached the maximal depth of recursion.

#### G. Results of our improvements

Table IV illustrated the number of nodes of the dichotomy tree without and with our impromenets (VIs), the sum of the number of nodes with VIs is approximately 10 times less than without them.

#### H. Using volume intervals

The search-space can be dramatically reduced also if the interval of unit cell volume is known. There are some methods for estimation of unit cell volume (see [26], [27], [28]), but these methods can fail, and we lost the main property of the dichotomy method: exhaustive search. Therefore, we use the same method that is commonly used in the crystallography

TABLE IV

EFFECT OF OUR IMPROVEMENTS *orthol*, *monol*, *tril*,  $|DT|$  DENOTES THE NUMBER OF NODES IN DT, F1 MEANS THE APPLICATION OF THE TECHNIQUE DESCRIBED IN IV-D, F2 MEANS THE APPLICATION OF THE TECHNIQUE DESCRIBED IN IV-E, F3 MEANS THE APPLICATION OF THE TECHNIQUE DESCRIBED IN IV-F.

parameter	orthol		monol		tril	
no. of layers in real	15		16		18	
no. of layers in reciprocal	15		16		9	
optimization applied?	without	with	without	with	without	with
$ DT $	$150 \cdot 10^3$	$36 \cdot 10^3$	$41 \cdot 10^6$	$14 \cdot 10^6$	$158 \cdot 10^6$	$57 \cdot 10^6$
no. of F1		$43.9 \cdot 10^3$		$42 \cdot 10^6$		$11.2 \cdot 10^6$
no. of F2		$1.8 \cdot 10^3$		$5.2 \cdot 10^6$		$0.03 \cdot 10^6$
no. of F3		$9.4 \cdot 10^3$		$3.2 \cdot 10^6$		$2.4 \cdot 10^6$

community: we divided the initial unit cell volume interval (VI) into smaller disjoint intervals (sometimes called “volume slices”). Then, we check them in ascending order until the solution is found. Table V illustrated the number of nodes (of dichotomy tree) without and with VIs, the sum of the number of nodes with VIs is approximately 10 times less than without them.

### J. Comparison with DICVOL and X-Cell

Our program for indexing is called ParaCell and it can be obtained from this URL [29]. It has tolerance for impurity lines, and may search for a zero-point (see details in the readme file on the program website). It support multiple method of indexing, for example TREOR method [8].

Straightforward comparison with other implementations of SD method is not possible: e.g., number of nodes is an internal information, we are unable to get this information from DICVOL or X-Cell program, some important program settings (e.g. used *hkl*-triples) of DICVOL and X-Cell are not known, program usually try to find any solution with the smallest volume, etc.

The main difference to the previous versions of the SD algorithm is in the basic idea that the program simply searches all combinations instead of trying to apply the crystallographic experience in code.

### J. Comparison with other indexing software

For comparison with all well-known indexing software we use Powder Diffraction Indexing Benchmarks [30].

Benchmarks consist of 10 indexing tests, result of each test can be number +1,0, or -1

- The 1 point note to tests means that the correct cell was found in first FoM position among the proposals.
- The zero point note means that the correct cell is mixed with uncorrect ones, not at the head of the list. The order of the true solution in the list is given as a subscript :  $O_6$  means that it was the sixth cell proposal. But the order has to be  $< 10$ , otherwise, there is a -1 point note given.

- The -1 point note means that the correct cell was not found at all, or at a position larger than 10 in the lists.

The results of our program are in Table VI, results of other indexing software are in Table VII

Our setup: Manual mode, dichotomy method is chosen, max. 5 unindexed lines are tolerated. The peak position error tolerance is  $0.15^\circ$ , exploration is made only in monoclinic in a restricted cell volume range ( $800 - 1200 \text{ \AA}^3$ ). That version has tolerance for impurity lines, and may search for a zeropoint.

Overall results of our program (with dichotomy method) are quite satisfactory.

## V. CONCLUSIONS

We found the idea behind the successive dichotomy rewarding. Our program is not an extension to the existing programs, it was build up from the ground. In the current version, it is a “pure” implementation of the successive dichotomy idea. There has remained an interest in extending this idea further to better handle data in low-symmetries (the monoclinic and triclinic) systems. In this paper, we propose some optimizations to reduce the indexing search space by using three indexing methods.

## ACKNOWLEDGMENTS

This work was supported by the Grant Agency of the Czech Technical University in Prague, grant No. SGS20/212/OHK3/3T/18.

## REFERENCES

- [1] A. Le Bail, “Monte carlo indexing with mcmaille,” *Powder Diffraction*, vol. 19, pp. 249–254, 2004.
- [2] J. W. Visser, “A fully automatic program for finding the unit cell from powder data,” *Journal of Applied Crystallography*, vol. 2, no. 3, pp. 89–95, Aug 1969. [Online]. Available: <http://dx.doi.org/10.1107/S0021889869006649>
- [3] A. A. Coelho, “Indexing of powder diffraction patterns by iterative use of singular value decomposition,” *Journal of Applied Crystallography*, vol. 36, no. 1, pp. 86–95, Feb 2003. [Online]. Available: <http://dx.doi.org/10.1107/S0021889802019878>
- [4] B. M. Kariuki, S. A. Belmonte, M. I. McMahon, R. L. Johnston, K. D. M. Harris, and R. J. Nelmes, “A new approach for indexing powder diffraction data based on whole-profile fitting and global optimization using a genetic algorithm,” *Journal of Synchrotron Radiation*, vol. 6, no. 2, pp. 87–92, Mar 1999. [Online]. Available: <http://dx.doi.org/10.1107/S0909049598017932>
- [5] R. Oishi-Tomiyasu, “Robust powder auto-indexing using many peaks,” *JOURNAL OF APPLIED CRYSTALLOGRAPHY*, vol. 47, no. 2, pp. 593–598, APR 2014.

TABLE V  
EFFECT OF VOLUME INTERVALS FOR MONO1,  $|DT|$  DENOTES NUMBER OF NODES IN DT

parameter	without VI	VI no.1	VI no.2	VI no.3	VI no.4	VI no.5	$\sum VI_{1...5}$
$V_{min}$ [ $\text{\AA}^3$ ]	30	30	300	600	900	1200	30
$V_{max}$ [ $\text{\AA}^3$ ]	3000	300	600	900	1200	1500	1500
$ DT $	$5.61 \cdot 10^6$	$4.59 \cdot 10^4$	$5.95 \cdot 10^4$	$7.91 \cdot 10^4$	$1.56 \cdot 10^5$	$2.57 \cdot 10^5$	$5.97 \cdot 10^5$
a solution found?	YES	NO	NO	NO	NO	YES	YES

TABLE VI  
DETAILED RESULTS OF OUR PROGRAM IN POWDER DIFFRACTION INDEXING BENCHMARKS

A1	A2	B1	B2	C1	C2	D1	D2	E3	F4
-1	+1	+1	+1	-1	+1	0 <sub>6</sub>	+1	+1	+1

TABLE VII  
RESULTS OF OTHER INDEXING SOFTWARE IN POWDER DIFFRACTION INDEXING BENCHMARKS

program	score	note
ITO13	-6	
DICVOL91	-2	
DICVOL04 (a)	+1	fast default and manual tests made by A. Le Bail
TREOR90	-2	
DICVOL04 (b)	+9	manual tests made by D. Louër
McMaille	+7	
ParaCell	+5	

- [6] —, “Distribution rules of systematic absences on the Conway topograph and their application to powder auto-indexing,” *ACTA CRYSTALLOGRAPHICA SECTION A*, vol. 69, no. 6, pp. 603–610, NOV 2013.
- [7] P.-E. Werner, L. Eriksson, and M. Westdahl, “*TREOR*, a semi-exhaustive trial-and-error powder indexing program for all symmetries,” *Journal of Applied Crystallography*, vol. 18, no. 5, pp. 367–370, Oct 1985. [Online]. Available: <http://dx.doi.org/10.1107/S0021889885010512>
- [8] I. Šimeček, J. Rohlíček, T. Zahradnický, and D. Langr, “A new parallel and gpu version of a *treor*-based algorithm for indexing powder diffraction data,” *Journal of Applied Crystallography*, vol. 48, no. 1, pp. 166–170, Feb 2015. [Online]. Available: <http://dx.doi.org/10.1107/S1600576714026466>
- [9] M. Hušák, I. Šimeček, and J. Rohlíček, “Powder data indexation by parallel gpu accelerated grid search method,” *Acta Crystallographica Section A*, vol. 69, no. a1, p. s272, Aug 2013. [Online]. Available: <http://journals.iucr.org/a/issues/2013/a1/00/a50933/a50933.pdf>
- [10] D. Louer and R. Vargas, “AUTOMATIC INDEXATION OF POWDER DIAGRAMS BASED ON SUCCESSIVE DICHOTOMIES,” *JOURNAL OF APPLIED CRYSTALLOGRAPHY*, vol. 15, no. OCT, pp. 542–545, 1982.
- [11] X-Cell, “X-cell - a novel and robust indexing program for medium-to high-quality powder diffraction data,” *J. Appl. Cryst.*, vol. 36, pp. 356–365, 2003.
- [12] A. Boulif and D. Louer, “INDEXING OF POWDER DIFFRACTION PATTERNS FOR LOW-SYMMETRY LATTICES BY THE SUCCESSIVE DICHOTOMY METHOD,” *JOURNAL OF APPLIED CRYSTALLOGRAPHY*, vol. 24, no. 6, pp. 987–993, DEC 1 1991.
- [13] D. Louer and A. Boulif, “Indexing with the successive dichotomy method, DICVOL04,” *ZEITSCHRIFT FÜR KRISTALLOGRAPHIE*, no. 1, 23, pp. 225–230, 2006, 9th European Powder Diffraction Conference, Prague, CZECH REPUBLIC, SEP 02-05, 2004.
- [14] —, “Powder pattern indexing and the dichotomy algorithm,” *ZEITSCHRIFT FÜR KRISTALLOGRAPHIE*, no. 1, 26, pp. 191–196, 2007, 10th European Powder Diffraction Conference, Univ Geneva, Geneva, SWITZERLAND, SEP 01-04, 2006.
- [15] —, “Some further considerations in powder diffraction pattern indexing with the dichotomy method,” *POWDER DIFFRACTION*, vol. 29, no. 2, pp. S7–S12, DEC 2014.
- [16] N. H. F. Beebe, “Openmp: overview and resource guide,” 2014. [Online]. Available: <http://www.math.utah.edu/~beebe/openmp/>
- [17] OpenMP Architecture Review Board, “Openmp application program interface,” online, 2013. [Online]. Available: <http://www.openmp.org/mp-documents/OpenMP4.0.0.pdf>
- [18] J. Bergmann, A. Le Bail, R. Shirley, and V. Zlokazov, “Renewed interest in powder diffraction data indexing,” *Zeitschrift für Kristallographie*, vol. 219, pp. 783–790, December 2004.
- [19] H. Dawood, *Theories of Interval Arithmetic: Mathematical Foundations and Applications*. LAP Lambert Academic Publishing, Germany, 2011.
- [20] A. Gomes, I. Voiculescu, J. Jorge, B. Wyvill, and C. Galbraith, *Interval Arithmetic*. Springer, London, 2009.
- [21] G. C. Fortman, A. M. Z. Slawin, and S. P. Nolan, “Highly active iridium(iii)?nhc system for the catalytic b?n bond activation and subsequent solvolysis of ammonia?borane,” *Organometallics*, vol. 30, no. 20, pp. 5487–5492, 2011.
- [22] J. Plévert, M. Louër, and D. Louër, “The *ab initio* structure determination of  $\text{Cd}_3(\text{OH})_5(\text{NO}_3)$  from X-ray powder diffraction data,” *Journal of Applied Crystallography*, vol. 22, no. 5, pp. 470–475, Oct 1989. [Online]. Available: <http://dx.doi.org/10.1107/S002188989006291>
- [23] E. Hadicke, F. Frickel, and A. Franke, *Chemische Berichte*, vol. 111, p. 3222, 1978.
- [24] M. Hušák, B. Kratochvíl, A. Jegorov, V. Mat’ha, M. Stuchlik, and T. Andrysek, “The structure of a new cyclosporin a solvated form,” *Zeitschrift für Kristallographie*, vol. 211, no. 5, pp. 313–318, 1996.
- [25] G. Qi and W. Parker, *Tetrahedron*, vol. 52, p. 2291, 1996.
- [26] G. S. Smith, “Estimating unit cell volumes from powder diffraction data: the triclinic case,” *Journal of Applied Crystallography*, vol. 9, no. 6, pp. 424–428, Dec 1976. [Online]. Available: <https://doi.org/10.1107/S0021889876011850>
- [27] —, “Estimating the unit-cell volume from one line in a powder diffraction pattern: the triclinic case,” *Journal of Applied Crystallography*, vol. 10, no. 4, pp. 252–255, Aug 1977. [Online]. Available: <https://doi.org/10.1107/S0021889877013442>
- [28] W. Paszkowicz, “On the estimation of the unit-cell volume from powder diffraction data,” *Journal of Applied Crystallography*, vol. 20, no. 3, pp. 161–165, Jun 1987. [Online]. Available: <https://doi.org/10.1107/S0021889887086904>
- [29] I. Šimeček, A. Zaloga, and J. Rohlíček. (2019) Paracell home page. [Online]. Available: <https://sourceforge.net/projects/paracell/>
- [30] (2004) Powder diffraction indexing benchmarks. [Online]. Available: <http://www.cristal.org/uppw/benchmarks/>