

**DEVELOPEMENT OF A NEW COMBINATORICAL MODEL  
FOR THE INVERSE PROTEIN FOLDING PROBLEM**

**DISSERTATION**

zur Erlangung des  
akademischen Grades

**Doktor rerum naturalium**

vorgelegt von

**Mag. rer. nat. Josef E. Frömcke**

Wien, im Juni 1997

am Institut für Theoretische Chemie und Strahlenchemie  
der Universität Wien

**No new principle will declare itself from below a heap of facts !**

*Sir Peter Medawar*

Diese Arbeit entstand zwischen Oktober 1994 und Juni 1997 am Institut für Theoretische Chemie. Mein Dank gilt in erster Linie meinen Eltern, ohne die mein Studium nicht möglich gewesen wäre.

Bei Herrn Professor Peter Schuster möchte ich mich für seine wissenschaftliche Betreuung und Förderung, die stets rechtzeitig zur Stelle waren, bedanken.

Den ungezwungenen wissenschaftlichen Gesprächen mit Walter Fontana verdankt diese Arbeit viele wertvolle Impulse.

Die Kollegen am Institut für Theoretische Chemie, Bärbel Krakhofer, Ronke Babajide, Susanne Rauscher, Tanja Ukrainczyk, Waltraud Pagitsch, Norbert Tschulenk, Alexander Renner, Herbert Kratky, Thomas Griesmacher, Erich Bornberg-Bauer, Robert Happel, Jan Cupal, Martin Fekete, Christoph Flamm, Christian Haslinger, Stephan Kopp, Stephan Müller, Stefan Wuchty, Ivo Hofacker, Walter Fontana und Peter Stadler haben für ein kreatives und angenehmes Arbeitsklima gesorgt.

## Abstract

The present work studies the inverse folding problem for a class of “block-world” model proteins. The primary structure of the model proteins consists of a sequence of space filling blocks of various shapes and chain attachment points. Modelling a hydrophobic core a native structure is considered to be a cube of appropriate size entirely filled with blocks.

- In this context the inverse folding problem is approached in two stages.
  - (1) The set of possible native structures (given a set of blocks) is viewed as the problem of tiling a cube. The resulting tiles were denoted as microconfigurations in this thesis.

For some instances the tilings of a cube could be systematically generated. For large configuration spaces an ergodic move set was defined to convert one tiling into another.
  - (2) Once the set of tilings had been generated, it was tested if a chain could actually be laid through them.
- It was possible to calculate the folding probability of sequences to form hydrophobic cores for sets of blocks in the  $3 \times 3 \times 3$  and  $4 \times 4 \times 4$  cube. Folding probabilities were obtained by exhaustive folding of all sequences on a given set of blocks.

## Zusammenfassung

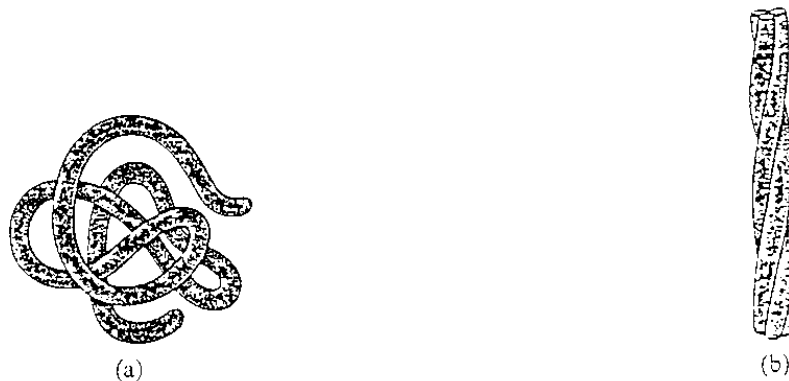
In der vorliegenden Arbeit wurde das “inverse protein folding problem” anhand von Modellproteinen betrachtet. Die Primärstruktur der angewandten Modellproteine besteht aus einer Sequenz von raumfüllenden Blöcken unterschiedlicher Form und Größe. Jeder Block repräsentiert dabei eine Aminosäure. Der hydrophobe Kern der nativen Proteinstruktur ist im Modellprotein ein Würfel, welcher kompakt mit Blöcken ausgefüllt ist.

- Das “inverse protein folding problem” wurde im Modell in zwei Schritten behandelt.
  - (1) Die nativen Strukturen einer gegebene Menge von Blöcken wurden als Würfelzerlegungen, in dieser Arbeit als Mikrokonfigurationen bezeichnet, betrachtet.  
Für einige Blockmengen konnten alle Mikrokonfiguration berechnet werden. Für große Konfigurationsräume wurde ein ergodisches Moveset festgelegt mit dessen Hilfe alle Mikrokonfigurationen durch einen “random walk” ineinander umwandelbar sind.
  - (2) Für die ermittelten Mikrokonfigurationen konnten die faltenden Sequenzen berechnet werden.
- Die Faltungswahrscheinlichkeit der Sequenzen konnte für Blockmengen des  $3 \times 3 \times 3$  und  $4 \times 4 \times 4$  Würfels berechnet werden. Bei diesen Berechnung wurde die Kette für alle Sequenzen einer gegebenen Menge von Blöcken einem “exhaustive folding”-Algorithmus unterworfen.

# 1. Introduction

## 1.1. Proteins

Proteins are biopolymers. Their ribosomal synthesis regularly works with 20 different amino acids. They can have a large variety of functions in biological systems and thus their localisations, concentrations and activities have to be controlled carefully. Enzymes catalyze very specific reactions and speed up reaction rates by factors of up to  $10^6$  and more. The transport of  $O_2$  by hemoglobin or myoglobin are examples for transport processes of small molecules mediated by protein. The cytoskeleton of cells consists of different protein polymers or filaments. Keratin, collagen, actin filaments, or microtubuli are examples. Keratin and collagen stabilize cells and tissues against external forces. Actin filaments are important for motility of cells and, together with myosin filaments, for contraction of muscles. Motorproteins like kinesin and dynein transport vesicles to the peripherie (anterograd direction) or to the cellcenter (retrograd direction). If proteins are classified according to their shape and solubility in water fibrous and globular proteins can be distinguished. Figure 1 shows the overall shape of globular (coiled) and fibrous (rodlike) proteins.



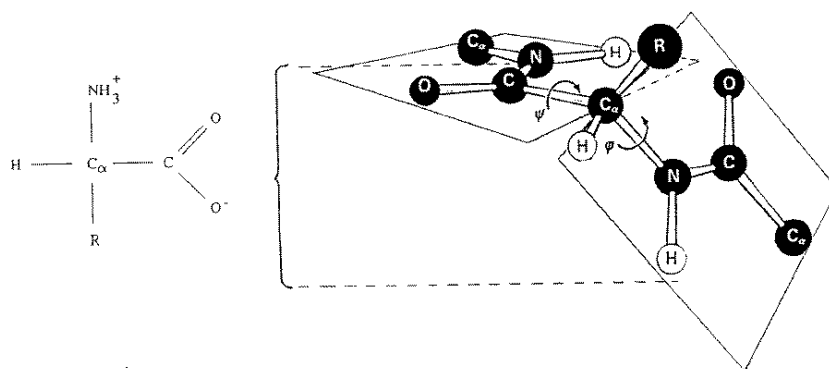
**Figure 1:** Shapes of proteins: a) shows a coiled globular protein and b) a typical rodlike fibrous protein

Fibrous proteins are long molecules of rodlike structure that are insoluble in water. Keratin, and collagen can be classified as fibrous proteins. Globular proteins in contrast have spherical shape, forming a coil, and are in general soluble in water. They are no random coils but have well defined unique structures.

Enzymes, hemoglobin (myoglobin in muscles), antibodies are globular proteins. Transmembrane proteins like receptors or ionic channels form a third class of proteins because they are insoluble in water without being fibrous proteins.

## 1.2. The structure of proteins

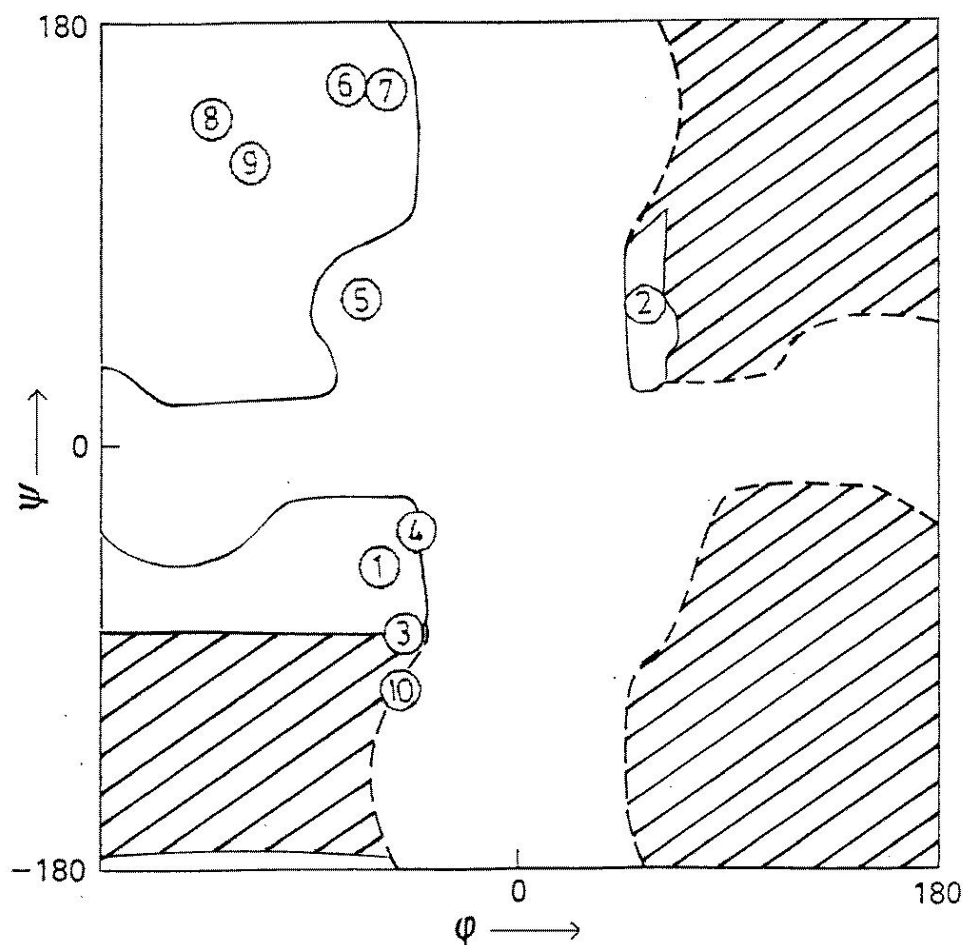
Proteins consist of amino acids. The  $\alpha$ -amino group and the  $\alpha$ -carboxyl group of two amino acids are covalently linked to form the peptide bond. There is no free rotation around the CO-NH peptide bond, because it has partial double bond character. This is indicated by the two planes in figure 2. In figure 2 a definition of the two dihedral angles ( $\varphi$ ,  $\psi$ ) is given.



**Figure 2:** Dominant structure of amino acids at  $\text{pH} \simeq 7$  and definition of the two dihedral angles determining the conformation of the peptide bond

The primary structure of a protein is the covalent structure of the polypeptide chain of this protein, excluding disulfide bonds between cystein residues. It is commonly expressed as the sequence of amino acid residues. Rotation about covalent bonds ( $\varphi$  and  $\psi$ ) gives different conformations of a protein. Certain regular conformations of the polypeptide backbone build the secondary structure. The backbone of a polypeptide chain forms a linear group if its dihedral angles are repeated. Every linear group is a helix. Conformations like  $\alpha$ -helices as well as  $\beta$ -sheets have sets of repeating backbone dihedral angles and contain helices. A pair of amino acids in a given secondary structure element has characteristic ( $\varphi$ ,  $\psi$ ) values.

The Ramachandran diagram (figure 3) gives the backbone dihedral angle sets that are possible in a protein under the assumption that atoms can be treated as hard spheres. For alanin or any other amino acid except of glycin and prolin three different regions in the Ramachandran diagram are accessible (figure 3). The left upper region of figure 3 contains the  $(\varphi, \psi)$  values of parallel and antiparallel  $\beta$ -sheets as well as the collagen helix. A second region contains the right handed  $\alpha$ -helix (=1) in figure 3). In a third region the left handed  $\alpha$ -helix (=2) is theoretically, according to the hard sphere model, possible. Because of the energetically unfavorable situation it is not realized in real proteins. Glycin has, with a proton as residue, relatively free rotation and can access an additional fourth region (hatched in figure 3). Prolin in contrast has a very restricted rotation because the  $\varphi$ -value of the  $C_\alpha - N$  bond is fixed at  $-65^\circ$ .



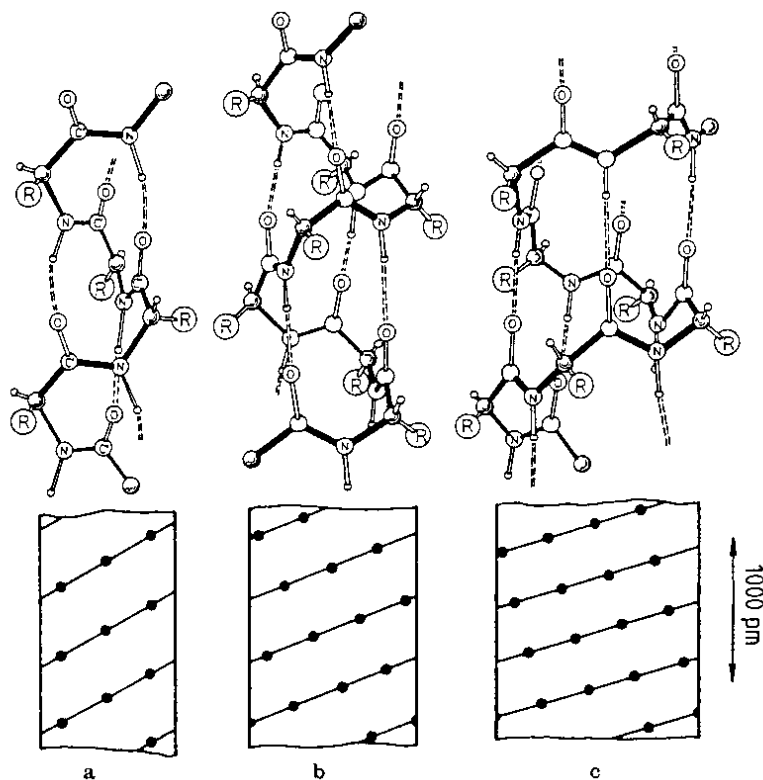
**Figure 3:** Ramachandran diagram. (1)  $\alpha_R$ -helix (The polypeptide of the backbone chain traces a right handed helical path.) (2)  $\alpha_L$ -helix (The polypeptide of the backbone traces a left handed helical path.) (3)  $\pi$ -helix (or  $4,4_{16}$ -helix) (4)  $3,0_{10}$ -helix (5) flat  $2,2_7$ -helix (6) polyprolin-helix (7) kollagen-helix (8) antiparallel and (9) parallel  $\beta$ -sheet (10) plain cyclopentapeptide ring



## 1.2.1. Elements of secondary structure

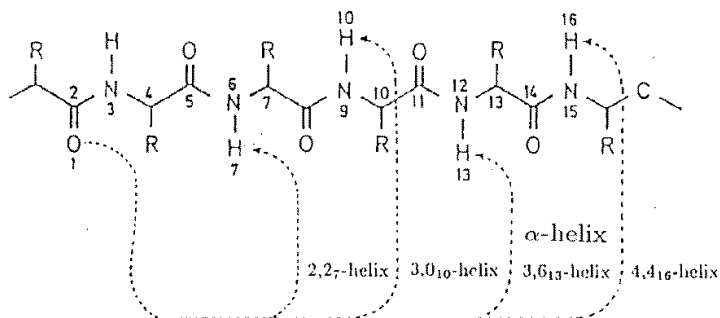
### 1.2.1.1. Helices

Figure 4 shows that different average numbers of amino acids per helix turn (3,0, 3,6, and 4,4) are possible.



**Figure 4:** Comparison of different hydrogen bond systems in different helices. a:  $3,0_{10}$ -helix; b:  $\alpha$ -helix ( $3,6_{13}$ -helix); c:  $\pi$ -helix ( $4,4_{16}$ -helix)

The indices 10, 13, and 16 of figure 4 are defined as the number of atoms per helix turn to build the corresponding hydrogen bond (figure 5).

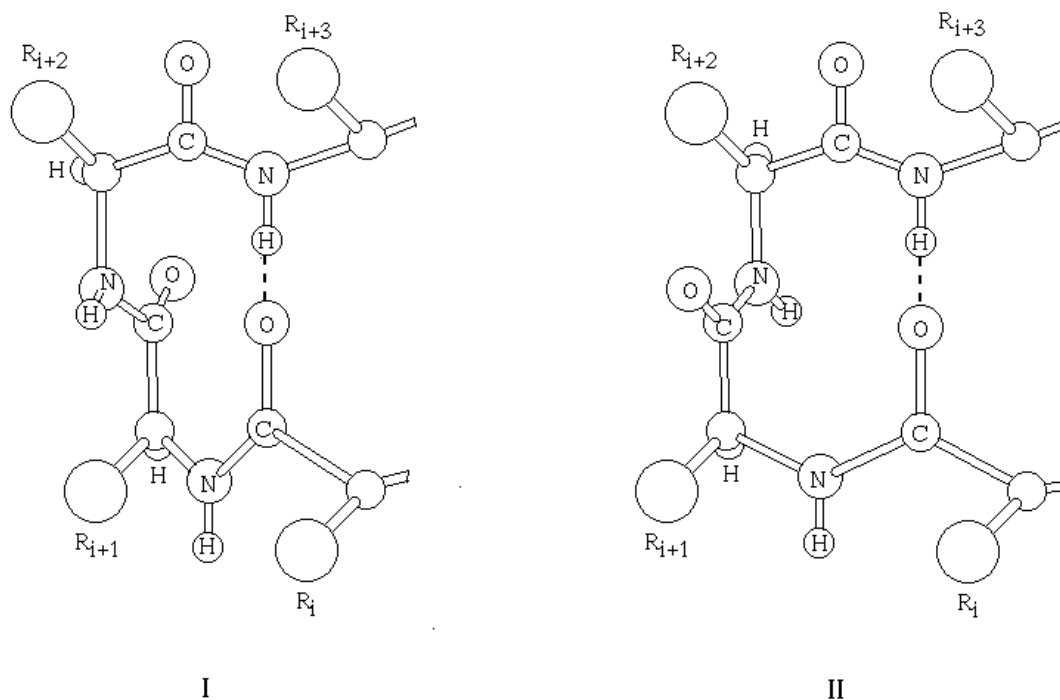


**Figure 5:** Determination of indices in figure 4 as the number of atoms per helix-turn (7, 10, 13, and 16)

The  $3_10$ -helix has an unfavorable side chain packing. Small pieces, about one or two turns, can be observed that tend to be at the N- and C-termini of  $\alpha$ -helices. An  $\alpha$ -helix has on average 3.6 amino acids per helix turn. The  $\pi$ -helix has never been observed. It would form an axial hole which cannot be filled with water and the van der Waals attractions would be reduced. Side chain interaction disfavor the left handed  $3_10$ ,  $\alpha$ , and  $\pi$ -helices which have never been observed yet.

### 1.2.1.2. Reverse turns

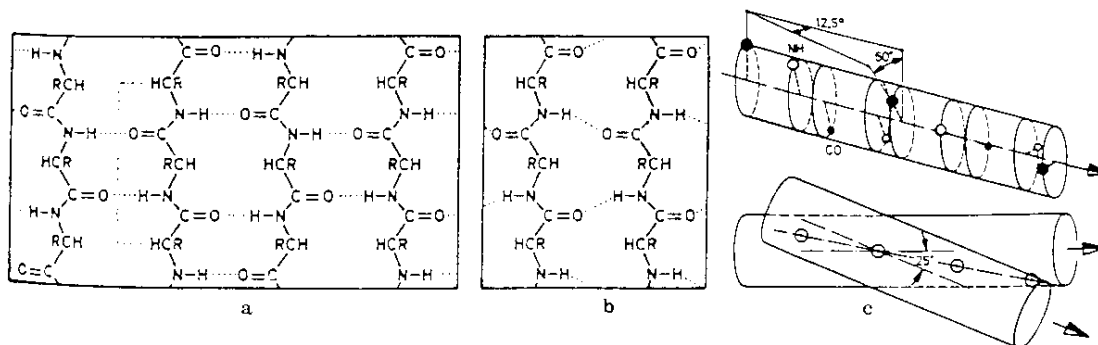
The peptide chain can form sharp reverse turns which contain a hydrogen bond. Three favorable conformations of three consecutive peptide units with a hydrogen bond between  $O_i$  and  $N_{i+3}$ , named reverse turn I, II, and III, can be found (Venkatachalon). While reverse turn III is a piece of  $3_10$ -helix, reverse turn I is a deformed  $3_10$ -helix. In reverse turn II the peptide unit between residues  $i + 1$  and  $i + 2$  of reverse turn I has flipped over (figure 6). Because of the steric hindrance between side chain  $R_{i+2}$  and  $O_{i+1}$  residue  $i + 2$  has to be a gly.



**Figure 6:** Reverse turns of type I and II. In type II residue  $i+2$  has to be a gly.

## 1.2.1.3. Sheets

Pauling and Corey postulated the parallel planar and antiparallel planar  $\beta$ -sheet as regular hydrogen bond structures for polypeptide chains. A planar (antiparallel) sheet can be found in glutathione reductase. Globular proteins contain about 15% sheet structure. The side chains of a sheet point alternatively to either side of the sheet. The  $C_\alpha - C_\beta$  bond of the side chain are perpendicular to the sheet plane. No preference for parallel or antiparallel sheets is observed. There is a clear preference of twisted sheets compared to planar sheets (figure 7).

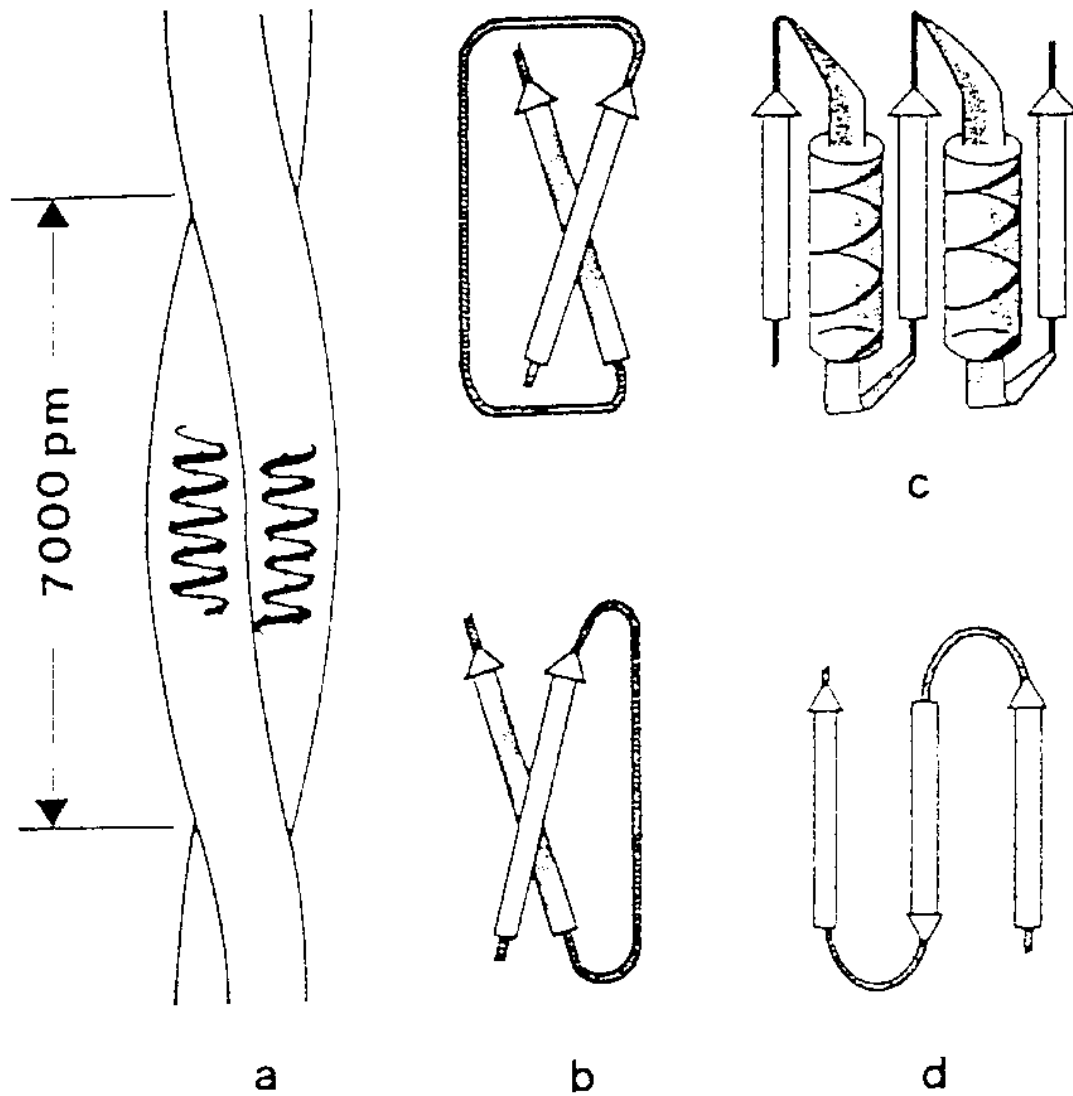


**Figure 7:** a) and b) show two possible forms of  $\beta$ -sheet-conformations; a: antiparallel  $\beta$ -sheet; b: parallel  $\beta$ -sheet; c: most observed sheets are twisted

The reason for this preference of twisted sheets is local optimization of hydrogen bonds. In either type of sheet, parallel as well as antiparallel, the chain forms a linear group with one residue as group element. Every linear group has to be a helix and in the case of  $\beta$ -sheets it is a very extended left handed helix. This helix corresponds to a right handed rotation of carbonyl and amide groups of about  $60^\circ$  per two residues. For an optimal hydrogen bond formation between neighbouring strands it is necessary that the strands form an angle of about  $25^\circ$  with each other. The twisted single strands are the reason for the twisted sheet. If the sheet is viewed along the plane perpendicular to the sheet strands it has a left handed twist.

### 1.2.2. Elements of supersecondary structure

Frequently observed combinations of secondary structure, also called motifs, form elements of supersecondary structure. Some of these elements are shown in figure 8.



**Figure 8:** Elements of supersecondary structure: a: two  $\alpha$ -helices build a superhelix - a coiled coil  $\alpha$ -helix; b: two possibilities for a  $\beta\xi\beta$ -unit; c: successive  $\beta\alpha\beta$ -units (Rossmann-fold); d:  $\beta$ -Mäander

## 1.3. States in protein folding

### 1.3.1. General considerations

How does a given sequence of amino acids fold into its functional active tertiary structure? A random search of the right conformation would mean that every already folded region of the protein could unfold again at any time during the folding process. Because there are so many possible conformations of an amino acid chain a random search is out of discussion. It would take years for such a globally exhaustive search to find the native conformation [30]. Consequently some correctly folded parts of the protein have to be conserved during the folding process by cumulative selection. The free energy difference between the unfolded and the native state of a typical protein is not very large. For a protein with hundred amino acids it is at room temperature on average only 40 kJ/mol. Each residue has therefore an average stabilisation of 0,4 kJ/mol which is lower than the thermal energy ( $RT = 2,5kJ/mol$ ). This makes clear that cumulative selection does not mean a fixation of correct but randomly positioned short stretches of the amino acid chain because such short folded parts could easily be disrupted by thermal energy. The meaning of cumulative selection has to be conservation of cooperatively folded, and therefore larger stretches, of the protein.

### 1.3.2. The native state as free energy minimum

Anfinsen recognized that reduced ribonuclease regained its characteristic biological activity on removal of the denaturing agent in the absence of other macromolecules. Ribonoclease, with 8 cystein amino acids, could also have been paired to 104 wrong configurations. The native form was build as the thermodynamically most stable form and it turned out to be the only form with enzymatic activity. So it became clear that the information for the tertiary structure of ribonuclease is determined by the amino acid sequence. Fraenkel-Conrat & Williams (1955) reassembled infectious tobacco mosaic virus by incubating together the seperated purified virion components.

The self-assembly principle, stemming from these works, states that all the information required to specify structure and function of a protein, by folding of newly synthesized polypeptides and association into oligomers, is determined by the amino acid sequences of the polypeptides comprising that protein. Protein assembly means the formation of secondary, tertiary, and quaternary structure. Being in no contradiction to the self-assembly principle in some cases pre-existing proteins are present to assist protein assembly. Proteins like disulphide isomerase perform covalent post-translational modifications to some proteins. Known molecular chaperons on the other hand build a class of unrelated families of protein that assist the correct non-covalent assembly of other polypeptide containing structures *in vivo*. They do not convey steric information essential for correct assembly. Their binding to interactive protein surfaces prevents incorrect interactions that would otherwise produce non-functional structures. Chaperons are not components of the assembled structures.

Several features are characteristic for the native state of a protein.:

- The native state of small proteins is in most cases thermodynamic stable [28].
- Proteins have characteristic secondary and tertiary structure [19].
- The protein core is very tightly packed. It is usually devoid of the simple spatial regularity of a crystal [24, 25].

### 1.3.3. The molten globule state

#### 1.3.3.1. $\alpha$ -Lactalbumin

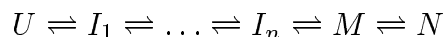
The refolding kinetic of  $\alpha$ -lactalbumin was studied by following the time-dependent changes in the circular dichroism spectra (CD) in the aromatic and the peptide regions [18]. This was done by 20-fold dilution of the unfolded protein in 6M guanidinium hydrochloride (Gdn.HCl). In the cited work an early folding intermediate was found being still unfolded when measured by the aromatic bands (CD values at 220 nm) but had folded secondary structure as measured by the peptide bands (CD values at 270 nm). This intermediate with presence of secondary structure but absence of rigid tertiary structure was called molten globule.

Human [8, 9] and bovine [9, 20]  $\alpha$ -lactalbumines show a first-order temperature transition. This transition is called all-or-none temperature transition because at the transition temperature there are only native states and denatured states but no partly native states. The temperature-denatured states are molten globules [8, 9]. So these native-molten globule transitions are comparable to the melting of a crystal. If proteins could denature noncooperatively they could be destroyed by thermal motions at all temperatures. A first-order phase transition requires a denaturation temperature that is large enough to destroy the structure as a whole. This resulted in the scheme:



### 1.3.3.2. Cytochrome c

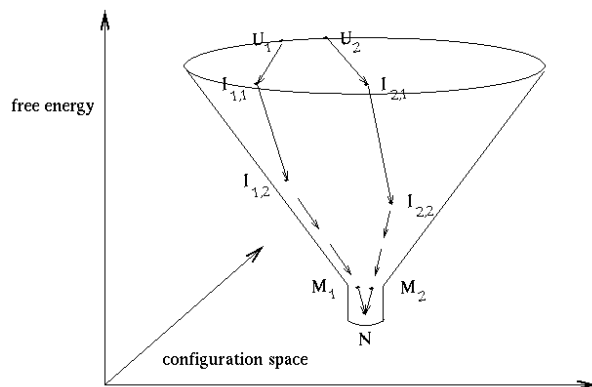
After sudden dilution of unfolded horse ferricytochrome c in 5M Gdn.HCl the refolding kinetic was investigated by stopped-flow methods, using far-UV circular dichroism (CD), near UV CD, and tryptophan fluorescence [11]. A partially condensed intermediate with a fluctuating core and no stable hydrogen bonds was found. This intermediate contained a significant amount of helical secondary structure, 44% of the total change associated with refolding, was formed in less than 4ms. The remaining 56% of  $\alpha$ -helical structure were formed in a time interval from 10ms to 1s. The compact tertiary structure only began to appear in a 400ms step and was completed in a final 10s phase. Consequently one had to write:



### 1.3.3.3. Lysozyme

Lysozym has four  $\alpha$ -helices and one  $\beta$ -sheet. Using the stopped-flow method and far-UV CD, near-UV CD as well as tryptophan fluorescence the investigation of a detailed folding kinetic was possible [23]. The complete folding of  $\alpha$ -helices took 60ms. Compared to the folding of the  $\beta$ -sheet domain, which was completed after 600ms, the  $\alpha$ -helical domain folded faster. A faster folding of  $\alpha$ -helical domains compared to  $\beta$ -sheet domains can be observed in most molecules. It is after the formation of the  $\alpha$ -helix domain and  $\beta$ -sheet domain that tertiary structure can be formed. This work stressed another important fact. No prescribed sequence of intermediates were existent but many alternative folding pathways could be shown.

These multiple pathways can be visualized as paths starting at an unfolded state and “funneling” down to the native state (figure 9).



**Figure 9:** Protein folding as paths leading down the energy landscape of possible conformations of the protein

#### 1.3.3.4. General properties

In most cases the activity of a protein is destroyed by different mild denaturation conditions (0,5M Gdn.HCl or 4M urea, low or high pH, by high temperature, and by the influence of  $\text{LiClO}_4$ ). These conditions result in the molten globule state of the protein. The molten globule was postulated as an equilibrium state at mild denaturing conditions [9] and as kinetic intermediate of protein folding [10, 22]. In contrast to the molten globule state there is a puzzle like specific tight packing in the native state. There are several experimental findings that support this model.

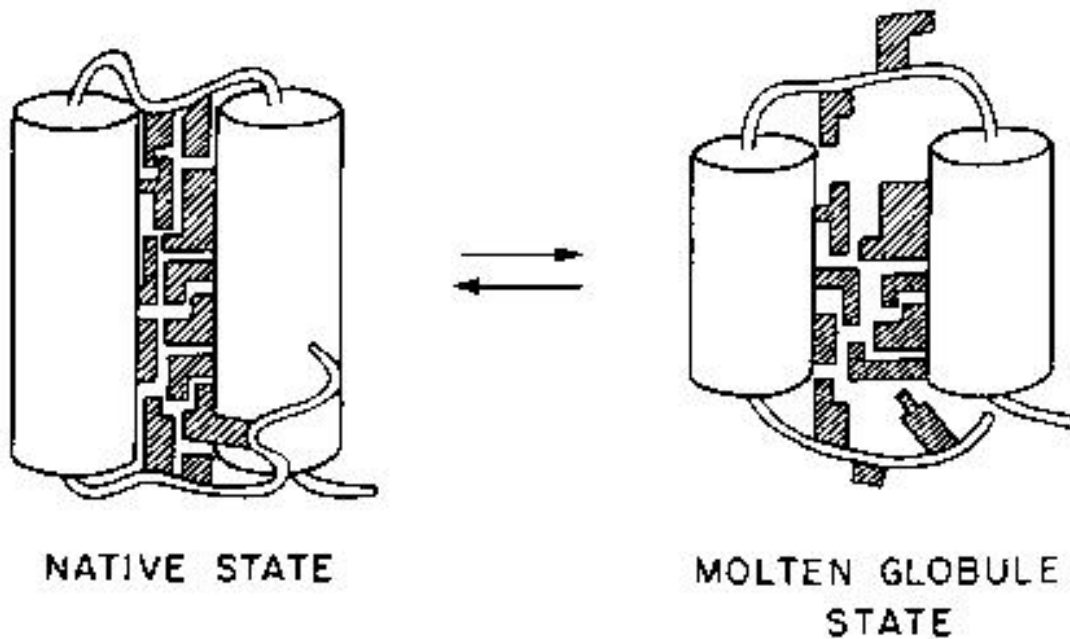
- (1) The molten globule state is stabilized mainly by hydrophobic interactions. Site-directed mutagenesis of apomyoglobins by Hughson and Baldwin [1, 15, 16] brought more light into this question. By replacements of  $\text{cys} \rightarrow \text{leu}$ ,  $\text{ala} \rightarrow \text{leu}$ ,  $\text{phe} \rightarrow \text{trp}$ ,  $\text{ser} \rightarrow \text{leu}$ , and  $\text{ser} \rightarrow \text{phe}$  they showed that an increase in side chain hydrophobicity stabilized the molten globule state against unfolding. The same mutations destabilized almost always the native state.
- (2) NMR and near-UV CD measurements show that the environment of many side chains is much more rigid in the native state than in the molten globule. The mobility of aliphatic side chains is increased at the native-molten globule transition [13]. The motion of aromatic side chains in the molten globule state is hindered [26].



The molten globule has the following properties:

- compactness
- the presence of secondary structure
- the absence of rigid tertiary structure

Moreover the molten globule shows in many cases, e.g. apomyoglobin [21], native-like tertiary fold (figure 10).



**Figure 10:** The molten globule is compact but has no tight puzzle like packing.

Robert T. Sauer [27] used the method of combinatorial mutagenesis to the N-terminal domain of the  $\lambda$ -repressor. The result showed that the information most important for folding is carried by residues in the hydrophobic core. The tight complementary packing of hydrophobic residues in the protein core seems to play a major role in specifying structure and stability.

## 1.4. Models for protein folding

### 1.4.1. Molecular dynamics simulation

Molecular dynamics is the science of simulating the motions of a system of particles [17]. Energy landscapes can be explored by molecular dynamics techniques where Newton’s laws of motion, with interaction energies obtained from smaller molecules, are solved. The problem is that these brute-force calculations are limited by computational restrictions in calculation time and accuracy of potentials. In molecular dynamics the relevant conformations are sampled locally or in the case of Monte Carlo methods sparsely. Moreover high resolution models require arbitrary parameters and permit in most cases only limited sampling of conformational space. Atomic-level simulations can currently explore only the small conformational changes occurring in the range of picoseconds to nanoseconds. For molecular properties of simulation models entropies, energies, and free energies have to be computed from statistical mechanical partition functions. Partition functions are gained from counting of possible conformations.

### 1.4.2. Lattice models

The use of lattice models in protein folding has important consequences. With lattice models the full conformational space can be investigated. In a possible conformation the “excluded volume” condition has to be respected and this means for lattice models that no lattice site may be occupied twice. It is clear that certain disadvantages are linked with lattice models. For example resolution is lost and details of protein structure as well as details of energetics are not accurately represented.

#### 1.4.2.1. Collaps models

##### (A) Homopolymers

Homopolymers are polymers being composed of a single species of monomer. Paul Flory asked for the reason of polymer compactness (1949). A polymer chain consisting of hydrophobic monomers would ball up in water. The fewer compact than expanded conformations lead to a lower conformational entropy in the compact state of the molecule resulting in a force opposing collapse. According to the homopolymer collapse theory of Oleg Ptitsyn and Yuili Eisner (in 1965) a change in the strength of the monomer-monomer attraction leads to a sharp collapse from open to compact conformations. Homopolymers collapse to large ensembles of compact conformations. Some form elements of secondary structure like  $\alpha$ -helices and  $\beta$ -sheets.

## (B) Two polymer class lattice models

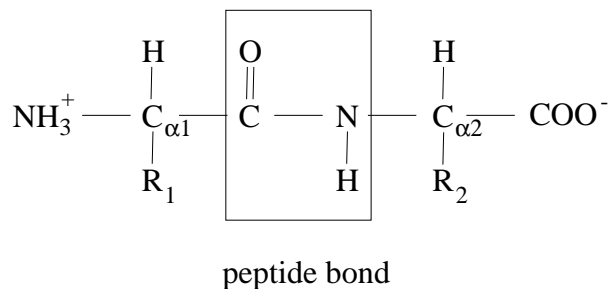
Two polymer class lattice models can collapse to a very small number of compact states. Shakhnovich and Gutin developed a heteropolymer model with  $B_{ij}$  as Gaussian distributed monomer-pair interaction strength and  $B$  as width of the heterogeneity distribution  $B_{ij}$ . They found that  $B$  plays a critical role in determining the number of degenerated ground states of the energy landscape. If  $B$  is large in their model, meaning that the sequences are sufficiently heterogenous, the number of lowest-energy states in the low-temperature phase is low.

The HP-Dill model [4] is, like the “perturbed homopolymer” model, a simple exact lattice model. In simple exact lattice models different residues of the amino acids at their corresponding  $\alpha$ -C atoms are omitted [14]. The amino acids are simplified to beads. Space is divided by the lattice into monomer-sized units or lattice sites. Those lattice sites may either be empty or filled by one bead. As lattice types in most cases 2D or 3D lattices are chosen. The different amino acids are classified as hydrophobic monomers (denoted by H) and polar or charged monomers (denoted by P). The energy landscape [2, 5] of a given conformation is regarded as function of H-H and H-P monomer contacts. Each interaction between two H monomers that are adjacent in space but not covalently linked is favored by a contact energy  $\epsilon < 0$ . All other interaction energies are zero. A maximum of H-H contacts gives the native state [6, 7, 12, 29].

Exhaustive enumeration is only possible for chain lengths up to about 30 monomers on the two-dimensional square lattice. The two-dimensional HP lattice model is a good simulation for the general properties of globular proteins. With small H-H attraction the chains populate a large ensemble of conformations, corresponding to denatured proteins, is obtained. With increasing H-H attraction a small ensemble of compact conformations with nonpolar core is formed. HP lattice models behave therefore like real proteins. In globular proteins H monomers tend to be hidden from water in the densely packed protein core while the P monomers, interacting favorably with water, tend to build the protein surface [3, 5, 19].

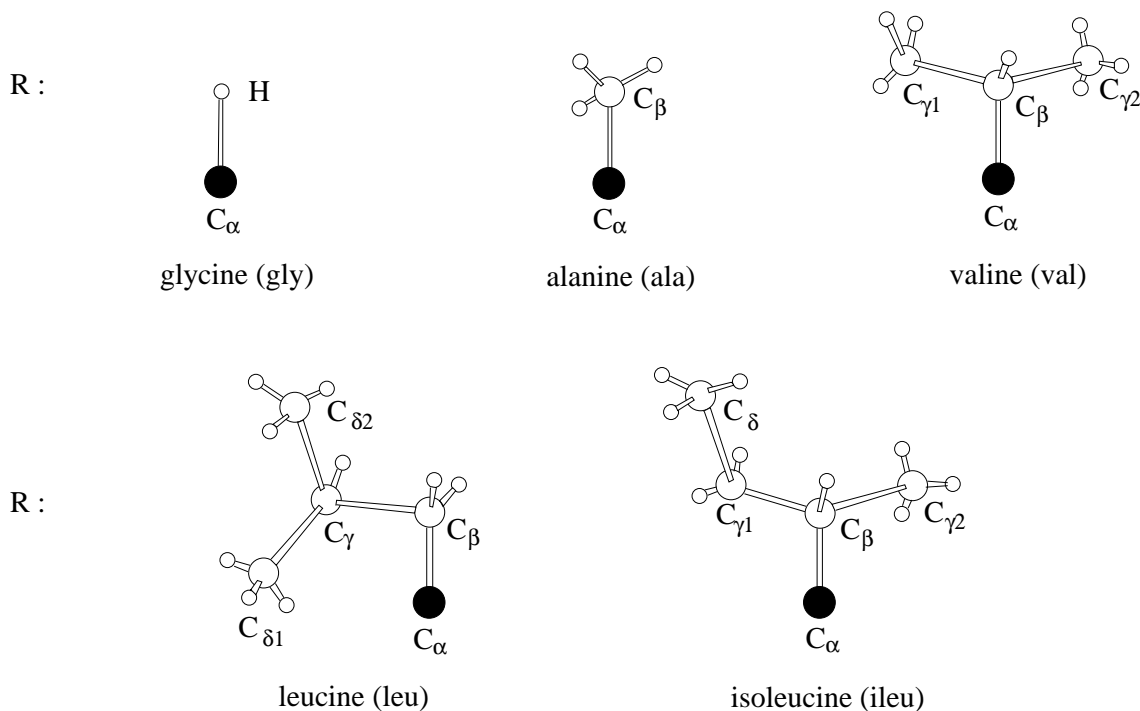
## 1.5. Present Work

At the inverse protein folding problem one is interested in the prediction of amino acid sequences that can adopt a given fold. The present work deals with the inverse folding problem in a hypothetical cubic shaped protein on a cubic lattice. If the third dimension of the problem is omitted a quadratic shaped protein on a quadratic lattice is regarded. Proteins contain, like the dipeptid in figure 11, a sequence of amino acids with certain residues linked to their  $C_\alpha$  atoms.






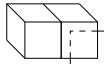
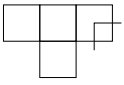
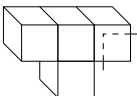
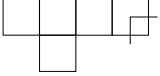
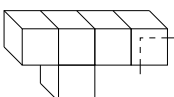

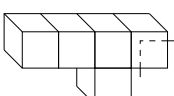
**Figure 11:** Dipeptide. The residues are linked to the peptide backbone at the  $C_\alpha$  atoms

Figure 12 shows the threedimensional pictures of frequent aliphatic amino acid residues.








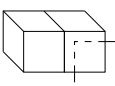
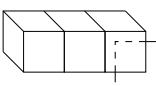
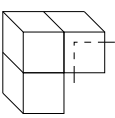
**Figure 12:** Three dimensional pictures of the most frequent aliphatic amino acids

The cubic or square shaped protein of the used model consists of one folded amino acid chain with absolutely compact amino acid residues in the protein core. Motivated by figure 12 each C-atom of an amino acid is given the same volume of one unit cell. This results in a block that represents the rough overall form of the corresponding amino acid (figure 13).

ALIPHATIC AMINO ACIDS (major amino acids contributed to a hydrophobic region)		2 dim. blocks	3 dim. blocks
glycine (gly)	$\text{H}_2\text{N}-\text{CH}_2-\text{COOH}$		
alanine (ala)	$\begin{array}{c} \text{H} \\   \\ \text{CH}_3-\text{C}-\text{COOH} \\   \\ \text{NH}_2 \end{array}$		
valine (val)	$\begin{array}{c} \text{CH}_3-\text{CH}-\text{CH}-\text{COOH} \\   \quad   \\ \text{CH}_3 \quad \text{NH}_2 \end{array}$		
leucine (leu)	$\begin{array}{c} \text{CH}_3-\text{CH}-\text{CH}_2-\text{CH}-\text{COOH} \\   \quad \quad   \\ \text{CH}_3 \quad \quad \text{NH}_2 \end{array}$		
isoleucine (ileu)	$\begin{array}{c} \text{CH}_3-\text{CH}_2-\text{CH}-\text{CH}-\text{COOH} \\ \quad \quad   \quad   \\ \quad \quad \text{CH}_3 \quad \text{NH}_2 \end{array}$		

**Figure 13:** Simulation of amino acids by similar structured blocks

To investigate the characteristic properties of the model, the set of blocks in the last figure is applied in a simplified form (figure 14).

dimension	$r_1$	$r_2$	$r_{3a}$	$r_{3b}$
2 dim.				
3 dim.				

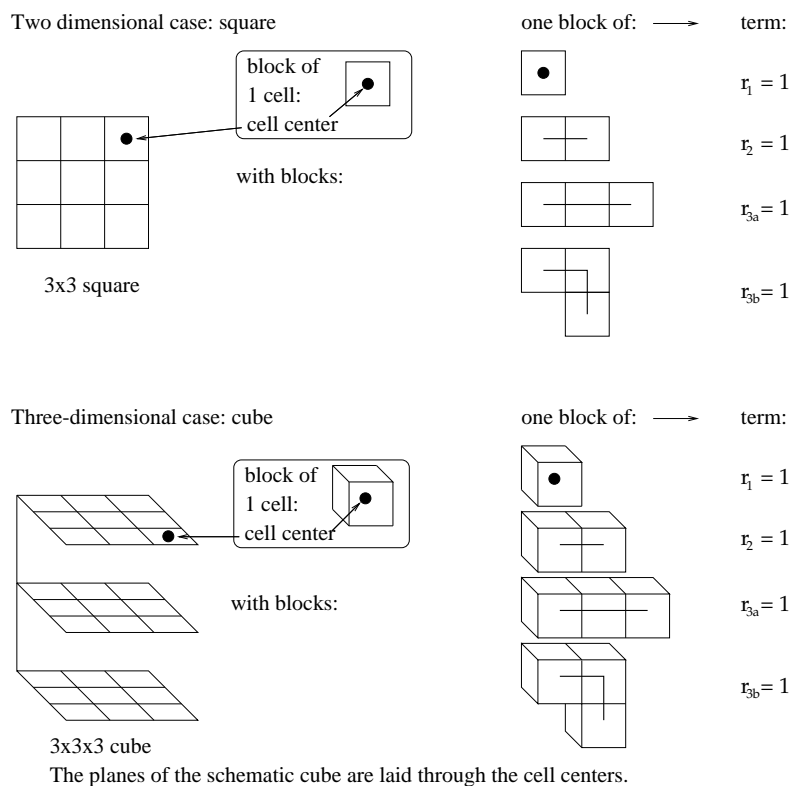
**Figure 14:** Simplified set of blocks that is used in the present work.

## 2. The Compact Block Model

### 2.1. Terminology

#### 2.1.1. Blocks

In the compact block model (CBM) amino acids in a protein core are represented by blocks fitting into each other like pieces of a puzzle. They block out a square (two dimensional case) or a cube (three dimensional case) of give side length  $n$  resulting in quadratic or cubic cells of equal size. This work deals with the three-dimensional case. Sometimes the square model is used to visualize general principles in an easy way. Each block may contain a maximum of three cells and the number of different sized blocks is given by  $r_1$ ,  $r_2$ ,  $r_3$ . A block with three neighboured cells exists in a linear or angled form which is denoted by  $r_{3a}$  or  $r_{3b}$  for the second case. Figure 15 shows the shape of  $r_1$ ,  $r_2$ ,  $r_{3a}$ , and  $r_{3b}$  blocks in a square and cube model.



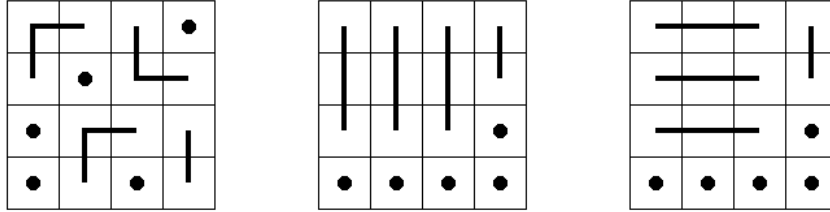
**Figure 15:** Kind of blocks for square and cube models

**2.1.2. Partition, subpartition, and microconfiguration**

In the beginning it is necessary to agree on a basic terminology. The meaning of partition, subpartition, and microconfiguration is now defined.

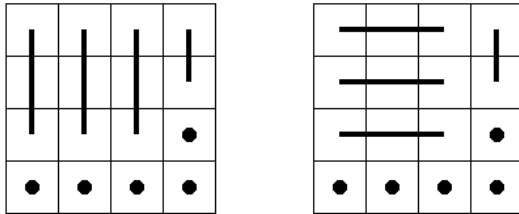
- (i) In a partition of a square or cube with given side length  $n$  the number of different sized blocks  $r_1, r_2, r_3$  are specified. These values will be given by  $\{r_1, r_2, r_3\}$ .

e.g.:  $r_1=5, r_2=1, r_3=3$  ( $=\{5, 1, 3\}$ ) ... in a  $4 \times 4$  square



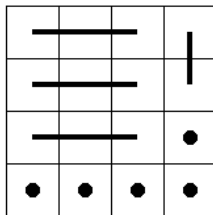
- (ii) In a subpartition linear  $r_{3a}$  and angled  $r_{3b}$  blocks are distinguished. The number of different sized and shaped blocks will be given by  $\{r_1, r_2, r_{3a}, r_{3b}\}$ .

e.g.:  $r_1=5, r_2=1, r_{3a}=3, r_{3b}=0$  ( $=\{5, 1, 3, 0\}$ ) ... in a  $4 \times 4$  square



- (iii) In a microconfiguration (= MC) the position and typ of each block is specified.

e.g.:  $r_1=5, r_2=1, r_{3a}=3, r_{3b}=0$  ( $=\{5, 1, 3, 0\}$ ) ... in a  $4 \times 4$  square



### 2.1.3. Considerations on arithmetical restrictions

#### 2.1.3.1. Number of partitions and subpartitions

Let  $n$  be the side length of a given square (dimension  $d = 2$ ) or a cube ( $d = 3$ ). With  $b$  as total number of blocks and  $r_1, r_2, r_3$  as number of blocks (residues) that occupy one, two, or three neighboured cells one can write:

$$r_1 + r_2 + r_3 = b \quad (1)$$

The total number of cells  $n^d$  equals the number of cells occupied by the different blocks. Because all cells of the cube (square) have to be occupied it holds that

$$r_1 + 2r_2 + 3r_3 = n^d \quad (2)$$

The number of two-celled blocks  $r_2$  equals at least half the number of all cells  $n^d$ .

$$0 \leq r_2 \leq \left\lfloor \frac{n^d}{2} \right\rfloor \quad (3)$$

... where  $\lfloor \cdot \rfloor$  is the floor function. ( $\lfloor x \rfloor$ : largest integer smaller or equal than  $x$ )

For the choice of  $r_3$  equation (2) determines the boundaries

$$0 \leq r_3 \leq \left\lfloor \frac{n^d - 2r_2}{3} \right\rfloor \quad (4)$$

For given  $r_2$  in a  $3 \times 3 \times 3$  cube  $r_3$  can be chosen from the intervalls of table 1.

$r_2$	$r_3$	combinations
0	$0 \leq r_3 \leq 9$	10
1	$0 \leq r_3 \leq 8$	9
2	$0 \leq r_3 \leq 7$	8
3	$0 \leq r_3 \leq 7$	8
4	$0 \leq r_3 \leq 6$	7
5	$0 \leq r_3 \leq 5$	6
6	$0 \leq r_3 \leq 5$	6
7	$0 \leq r_3 \leq 4$	5
8	$0 \leq r_3 \leq 3$	4
9	$0 \leq r_3 \leq 3$	4
10	$0 \leq r_3 \leq 2$	3
11	$0 \leq r_3 \leq 1$	2
12	$0 \leq r_3 \leq 1$	2
13	$r_3 = 0$	1

table 1

Table 1 shows that there are 75 different partitions in a  $3 \times 3 \times 3$  cube.



In mathematical terms the number of different partitions  $p$  can be expressed by

$$p = \sum_{r_2=0}^{r_2=\lfloor \frac{n^d}{2} \rfloor} \sum_{r_3=0}^{r_3=\lfloor \frac{n^d-2r_2}{3} \rfloor} 1 = \sum_{r_2=0}^{r_2=\lfloor \frac{n^d}{2} \rfloor} \left( \left\lfloor \frac{n^d-2r_2}{3} \right\rfloor + 1 \right) \quad (5)$$

This sum is equivalent to

$$p = \begin{cases} 1 + \frac{n^d}{2} + \lfloor \frac{n^d}{6} \rfloor + \lfloor \frac{n^d}{6} \rfloor^2 + \frac{1}{2} \left( \lfloor \frac{n^d-1}{3} \rfloor + \lfloor \frac{n^d-1}{3} \rfloor^2 \right) & \dots \text{ if } n \text{ is even} \\ \frac{n^d+1}{2} + \lfloor \frac{n^d+3}{6} \rfloor^2 + \frac{1}{2} \left( \lfloor \frac{n^d-1}{3} \rfloor + \lfloor \frac{n^d-1}{3} \rfloor^2 \right) & \dots \text{ if } n \text{ is odd} \end{cases} \quad (6)$$

If all floor brackets are ommited, resulting in  $p_{omm}$ , and a overall floor function is applied on  $p_{omm}$  for even  $n$ ,  $p_{app}$  is recieved as very good approximation for  $p$ .

$$p_{omm} = \begin{cases} \frac{3n^{2d}+26n^d+32}{36} & \dots \text{ if } n \text{ is even} \\ \frac{3n^{2d}+26n^d+32}{36} - \frac{1}{4} & \dots \text{ if } n \text{ is odd} \end{cases} \Rightarrow p_{app} = \left\lfloor \frac{3n^{2d} + 26n^d + 32}{36} \right\rfloor \quad (7)$$

In a subpartition it holds that  $r_3 = r_{3a} + r_{3b}$ . In cubes with  $n > 2$  (squares,  $n > 3$ ) are for fixed  $r_2$  all values for  $r_{3a}$  possible. For these cubes (squares) the number of different subpartitions  $s$  can be calculated.:

$$s = \sum_{r_2=0}^{r_2=\lfloor \frac{n^d}{2} \rfloor} \sum_{r_3=0}^{r_3=\lfloor \frac{n^d-2r_2}{3} \rfloor} \sum_{r_{3a}=0}^{r_{3a}=r_3} 1 = \frac{1}{2} \sum_{r_2=0}^{r_2=\lfloor \frac{n^d}{2} \rfloor} \left( \left\lfloor \frac{n^d-2r_2}{3} \right\rfloor + 1 \right) \left( \left\lfloor \frac{n^d-2r_2}{3} \right\rfloor + 2 \right) \quad (8)$$

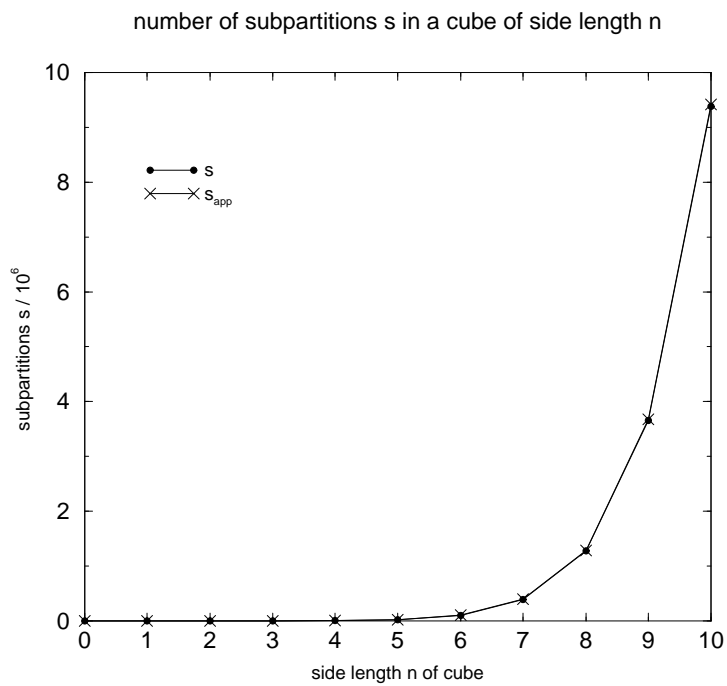
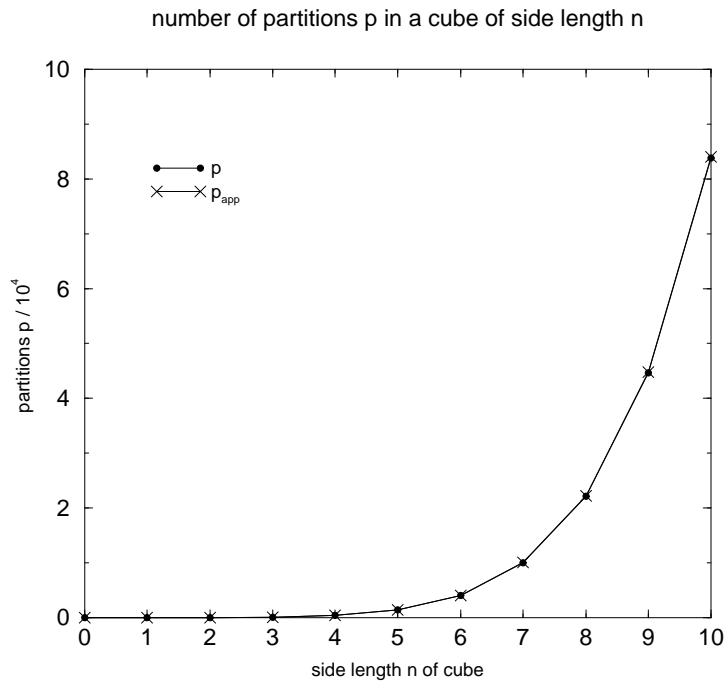
This sum is equivalent to

$$s = \begin{cases} \frac{1}{6} \left( 6 + 3n^d + 11 \lfloor \frac{n^d}{6} \rfloor + 15 \lfloor \frac{n^d}{6} \rfloor^2 + 4 \lfloor \frac{n^d}{6} \rfloor^3 + \right. \\ \left. + 5 \lfloor \frac{n^d-1}{3} \rfloor + 6 \lfloor \frac{n^d-1}{3} \rfloor^2 + \lfloor \frac{n^d-1}{3} \rfloor^3 \right) & \dots \text{ if } n \text{ is even} \\ \frac{1}{6} \left( 15 + 3n^d + 29 \lfloor \frac{n^d-3}{6} \rfloor + 21 \lfloor \frac{n^d-3}{6} \rfloor^2 + 4 \lfloor \frac{n^d-3}{6} \rfloor^3 + \right. \\ \left. + 5 \lfloor \frac{n^d-1}{3} \rfloor + 6 \lfloor \frac{n^d-1}{3} \rfloor^2 + \lfloor \frac{n^d-1}{3} \rfloor^3 \right) & \dots \text{ if } n \text{ is odd} \end{cases} \quad (9)$$

Omission of all floor brackets results in  $s_{omm}$ . The application of an overall floor function on  $s_{omm}$  for even  $n$  yields  $s_{app}$  as approximation for  $s$ .

$$s_{omm} = \begin{cases} \frac{6n^{3d}+105n^{2d}+570n^d+536}{648} & \dots \text{ if } n \text{ is even} \\ \frac{6n^{3d}+105n^{2d}+570n^d+536}{648} - \frac{1}{8} & \dots \text{ if } n \text{ is odd} \end{cases} \quad (10)$$

$$\Rightarrow s_{app} = \left\lfloor \frac{6n^{3d} + 105n^{2d} + 570n^d + 536}{648} \right\rfloor$$



**Figure 16:** Number of partitions  $p$  and subpartitions  $s$  for given side length  $n$  of the cube. The approximations are given by  $p_{app}$  and  $s_{app}$ .

In Figure 16  $p$  and  $p_{app}$  as well as  $s$  and  $s_{app}$  are compared. With larger  $n$  the relative error of the approximation functions becomes increasingly small.

### 2.1.3.2. Chain length

The chain length  $l$  equals  $b$  because at each position of the chain there is one block.

$$l = b \tag{11}$$

There is a restriction on  $b$  and therefore on  $l$  at given side length  $n$  of the square or cube.

Combination of equation (1) and (2) gives:

$$\left\lceil \frac{n^d}{3} \right\rceil \leq l \leq n^d \tag{12}$$

... where  $\lceil \cdot \rceil$  is the ceiling function. ( $\lceil x \rceil$ : smallest integer greater or equal than  $x$ )

Elimination of  $r_3$  in (1) and (2) yields:

$$r_2 = 3l - n^d - 2r_1 \tag{13}$$

Substitution of  $r_2$  in (1) using (13) gives:

$$r_1 - r_3 = 2l - n^d : \begin{cases} n : \text{even} & \text{then } r_1 \text{ and } r_3 \text{ are both even or both odd} \\ n : \text{odd} & \text{then } r_1 \text{ or (exclusive) } r_3 \text{ is even} \end{cases} \tag{14}$$

In a cube (square) of side length  $n$  the chain length has no influence on the case in (14). The left side of (14) is a constant for all partitions in a cube or square of side length  $n$  with equal chain length  $l$ .

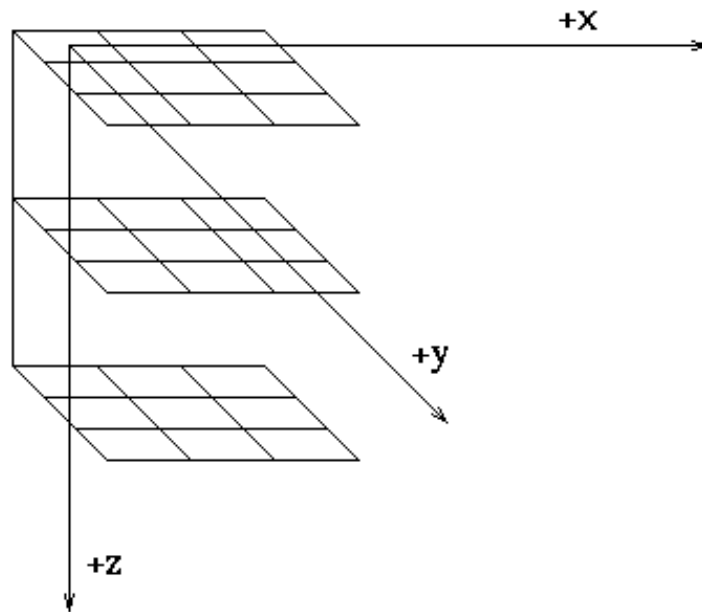
## 2.2. Coding of microconfigurations

### 2.2.1. Cell position

A microconfiguration (MC) consists of blocks of defined type and position. A block consists of different cells. To distinguish the positions of cells from each other the cube is fixed in a system of coordinates and the cells are numbered.

At this point a definition of the used system of coordinates is necessary.:

The origin of the used left handed system of coordinates lies in the center of the left upper cell (figure 17). The positive coordinates are parallel with the cube sides in such a way that the direction of the positive  $z$  coordinate points to the basis of the cube. Two neighboured cell centers have a distance of one.

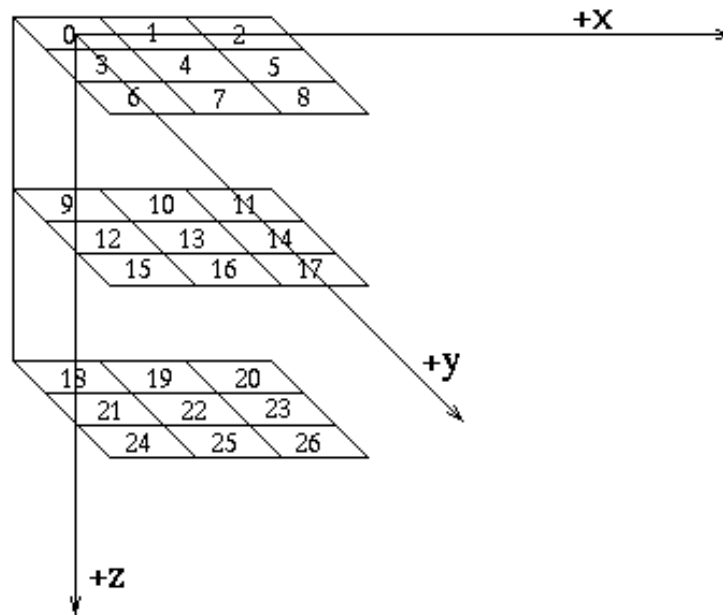


**Figure 17:** The applied coordinates

The numbering is done by the following steps:

- (i) The numbering starts with 0 at the origin of the described system of coordinates.
- (ii) Proceeding with numbering the direction of x has the highest priority. This means that if a neighbored cell in x-direction exists, this cell gets the next (natural) number. If there is no neighbored cell in the direction of x, numbering continues with a neighbored cell *of a cell of lowest number* in y-direction. Numbering in the direction of z has the lowest priority. If numbering in x-direction or y-direction is impossible the neighbored cell *of a cell of lowest number* in z-direction gets the next number.

Figure 18 gives the numbering of a 3x3x3 cube.



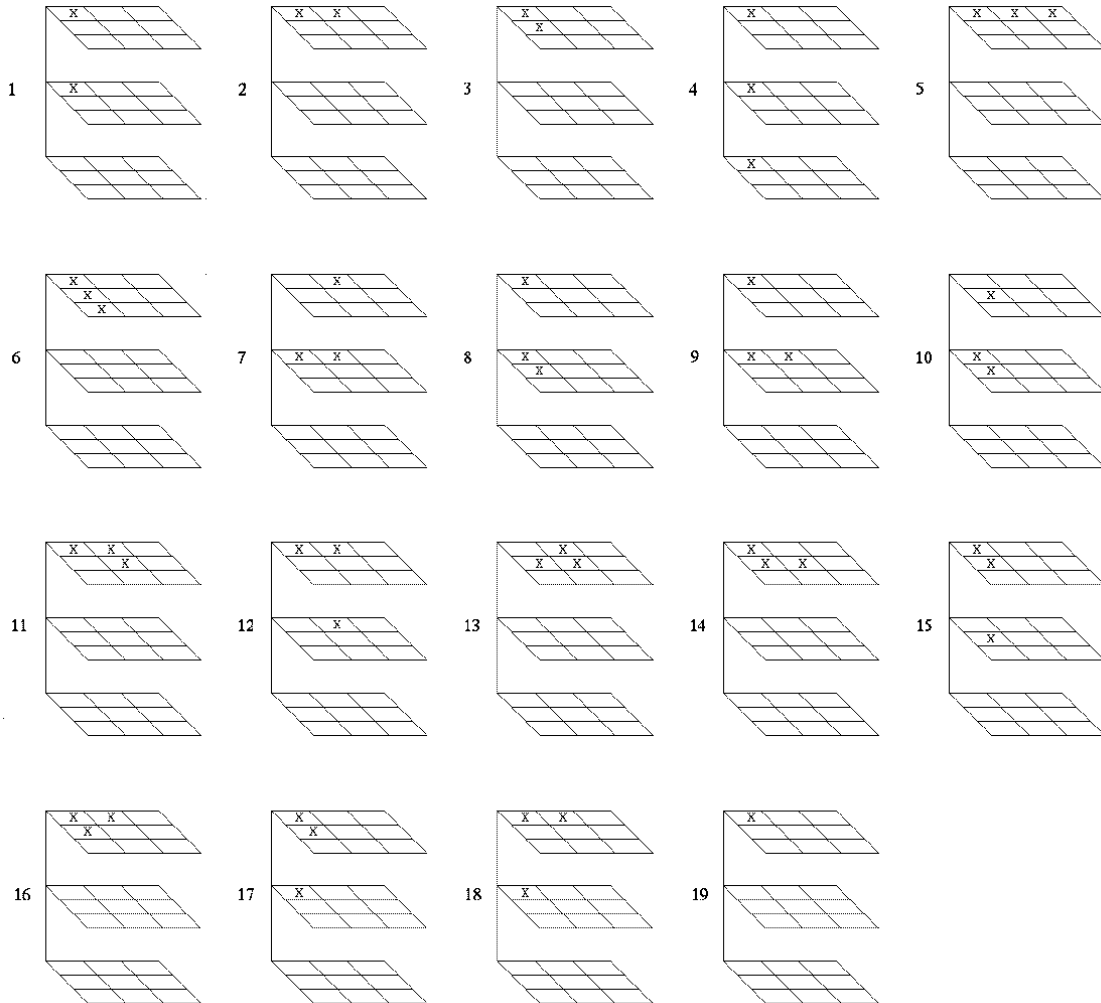
**Figure 18:** Cell positions

The resulting number that a cell gets by numbering is called its cell position  $p$ . It can be calculated from the coordinates of the cell center and the side length  $n$  of the numbered cube by:

$$p = n^2 z + ny + x$$

### 2.2.2. Block orientations

If a block of given type is randomly thrown into a cube and shifted without rotation as near to the origin as possible, the resulting block has after translation the same type and orientation than before. By this procedure one gains a reduced set of 19 blocks. In figure 19 each translated block of given size and orientation has a corresponding number. The number that results from a *hypothetical translation* of a given block is defined as the block orientation of this block.



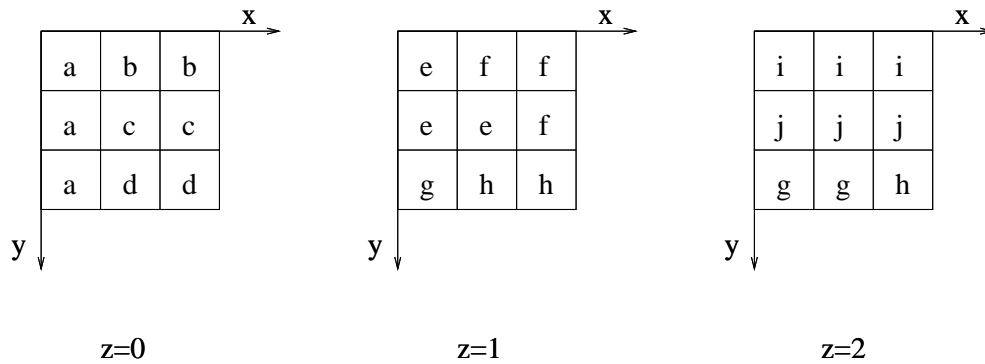
**Figure 19:** Table of all possible block orientation numbers for the used blocks

### 2.2.3. Code 1

If a MC is given one can visit all cells on a path  $P$  of increasing cell positions  $p_i$  with index  $i$  as number of steps along this path. Path  $P$  is defined to be:

$$P : p_i = \begin{cases} 0, & \text{for } i = 0; \\ p_{i-1} + 1, & \text{for } i > 0. \end{cases}$$

Each time a new block occurs the first time on path  $P$ , the block orientation of this block becomes the next number of code 1. Code 1 has as many numbers as the given MC has blocks because block orientations of blocks that occurred already before on  $P$  are ignored. This code was used for the exhaustive calculation procedure. Figure 20 shows a MC with corresponding code 1.



Cells with equal letters belong to the same block.

→ code 1 = (6, 2, 2, 2, 14, 11, 9, 12, 5, 5)

**Figure 20:** Code 1

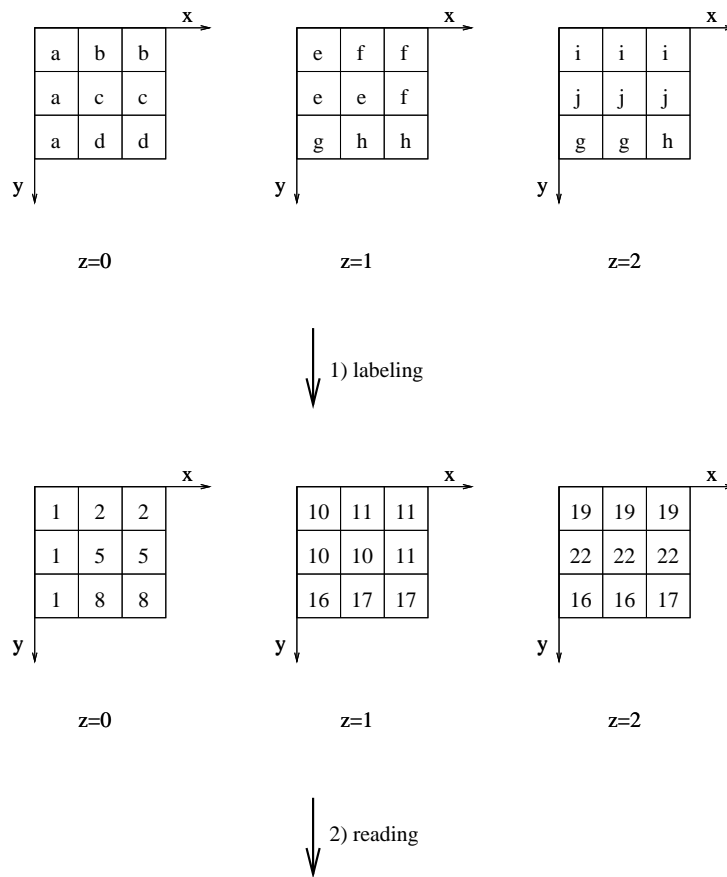
### 2.2.4. Code 2

In code 1 each number represents one block. To decode code 1 one has to know the block orientation table. In a given MC one can also choose the following procedure. Path  $P$ , as defined in chapter 2.2.3., is used. Like at the determination of code 1 the first occurrence of a block on  $P$  is important.

Now two steps, a labeling and reading step, are done.

- (i) labeling step: Only unlabeled blocks are labeled. All cells of the block with lowest position on  $P$  are labeled with 1. The cells of the second unlabeled block along  $P$  are all labeled with 2. This procedure is continued until all cells of the MC are labeled.
- (ii) reading step: Again  $P$  is traced. The number of the cells along  $P$  are the numbers of code 2.

Code 2 was applied in the microconfiguration space. It turned out to be very useful in comparing MCs. Common blocks in two MCs  $x$  and  $y$  have the same number on the same positions of their codes. Figure 21 shows the example of figure 20 with the corresponding code 2.



code 2 = (1, 2, 2, 1, 5, 5, 1, 8, 8, 10, 11, 11, 10, 10, 11, 16, 17, 17, 19, 19, 19, 22, 22, 22, 16, 16, 17)

**Figure 21:** Code 2



### 3. Exhaustive calculation of microconfigurations

#### 3.1. Basic concepts

Before the exhaustive calculation algorithm, or EC-algorithm for short, is described a short summary of necessary concepts is given.:

A block with one, two, three linear arranged, or three nonlinear arranged cells, is called a block of type 1, type 2, type 3a, or type 3b. In a MC  $r_1, r_2, r_{3a}$ , or  $r_{3b}$  are the numbers of blocks being of type 1, type 2, type 3a, or type 3b. All MCs having the same values for  $r_1, r_2, r_{3a}$ , and  $r_{3b}$  belong to the same subpartition  $\{r_1, r_2, r_{3a}, r_{3b}\}$ . A subpartition can therefore be imagined as blocks of specified type in a pool. The MC gives the specific way of how these blocks are put together to block out the final cubic space. As described in chapter 2.2.2. each block orientation have a corresponding, in the block orientation table predefined, number between 1 and 19. The block orientation of a block specifies the type of this block. The block start position of a block is the lowest cell position of the cells belonging to this block. Block orientation and block start position define type and exact position of a block in a cube.

The EC-algorithm treats the set of given blocks from a subpartition as block pool. An exhaustive calculation of all MCs that can be generated by these blocks is done. These MCs are calculated in one series of code 1 strings. The algorithm can roughly be described as construction of MCs with blocks from a given block pool. A block has to lie completely within the cube to be accepted and cells being occupied by previous blocks may not be part of a new block. If a block is accepted by the algorithm it is removed from the pool and inserted in the cube. Blocks that were previously accepted but do not lead to a valid or new MC are removed from the cube and added to the pool. The next section describes the algorithm more detailed. Cells of the cube that are not occupied by a block are called free cells.

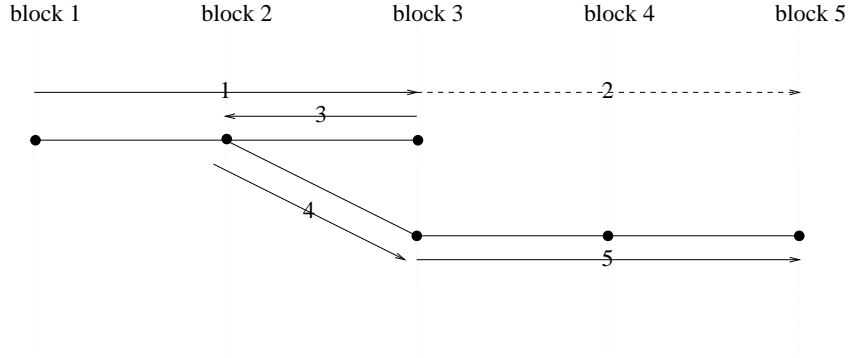
### 3.1.1. Algorithm

The EC-algorithm consists of the following steps.:

- (1) A free cell with lowest cell position has to be the block start position of the next block.
- (2) If the block orientation  $x$  is given, this block orientation is tested at the known block start position for acceptance. Otherwise  $x = 1$ . From the block orientation  $x$  the type of the block is specified. If no block of this type can be found in the pool or if it is not accepted in the cube the next higher block orientation  $x + 1$  is tested in step 2. If there is no next higher block orientation continue with step 3. If the block with block orientation  $x$  is accepted in the cube and a block of this type is contained in the pool, it is removed from the pool and inserted in the cube. Its block orientation is accepted as next number in code 1. If there are free cells left the algorithm continues with step 1. If the whole cube has been blocked out now code 1 is stored as solution and the algorithm continues with step 3.
- (3) The block orientation  $x$  of the last inserted block is stored as  $x_{old}$ . This block is removed from the cube and given back to the pool. The last number from code 1 is removed. If  $x_{old} < 19$  the algorithm stores  $x_{old} + 1$  as new block orientation  $x$  for step 2 and continues with step 1. If  $x = 19$  there are two cases:
  - (a) If the first number in code 1 has been removed in this step the algorithm ends with step 4.
  - (b) It was not the first number in code 1 that has been removed and step 3 is repeated.
- (4) All possible MC have been calculated.

In step 3 the EC-algorithm removes the last inserted block from the cube and adds a block of next higher block orientation (at the block start position of the removed block) to the cube. If no new block with higher block orientation exists the next “last inserted block” becomes removed.

Figure 22 shows that the EC-algorithm can be visualized by a block tree.



- steps:
- 1: block 1, 2, 3 fit in the cube
  - 2: there is no way to place a block of the pool at the lowest cell position
  - 3: block 3 gets removed from the cube
  - 4: next higher block orientation than that of block 3 is tried
  - 5: now block 4 and 5 fit in the cube

**Figure 22:** EC-algorithm visualized by a block tree

### 3.1.2. Resulting code 1

Code 1 of a MC  $m$  is a string of elements  $c_{m,i}$  with  $i$  as string number. The value of  $c_{m,i}$  gives the block orientation of the  $i^{\text{th}}$  block  $b_{m,i}$  along path  $P$  of rising block start positions.

If it is true for two MCs  $x$  and  $y$  that  $c_{y,i} > c_{x,i}$  at the first  $i$  with  $c_{y,i} \neq c_{x,i}$  the notation  $\text{code } 1(y) > \text{code } 1(x)$  is used.

The EC-algorithm generates  $y$  after  $x \Rightarrow \text{code } 1(y) > \text{code } 1(x)$

To proof this let MC  $x$  with code  $1(x)$  be generated earlier than MC  $y$  with code  $1(y)$  by the EC-algorithm. Blocks in  $x$  and  $y$  with the same block start position and block orientation are equal. The first string number  $i$  with  $c_{y,i} \neq c_{x,i}$  belongs to blocks  $b_{y,i}$  and  $b_{x,i}$ . These blocks have different block orientations  $c_{y,i}$  and  $c_{x,i}$  but the same block start position. All lower positions of  $x$  and  $y$  than this block start position have identical blocks. This means for the EC-algorithm that it has to remove  $b_{x,i}$  with block orientation  $c_{x,i}$  and to insert  $b_{y,i}$  with a block orientation  $c_{y,i} > c_{x,i}$  at the same block start position.

## 3.2. Symmetry of microconfigurations

The described EC-algorithm can be used together with two modules for rotation and reflection of microconfigurations.

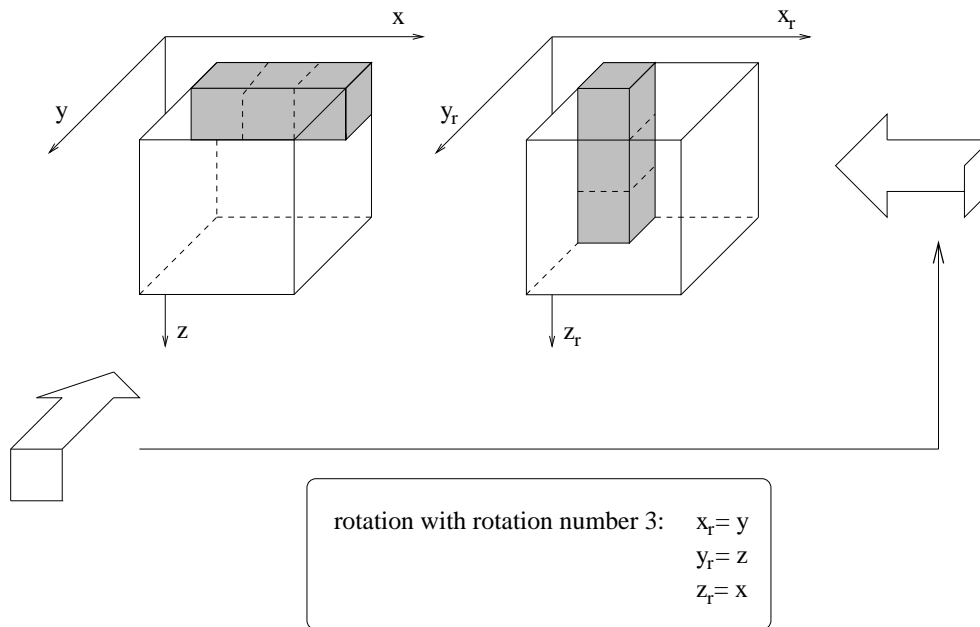
### 3.2.1. Rotation symmetry

The EC-algorithm finds the number of different MCs (= counts) for a given subpartition. MCs that generate each other by rotation are rotational equivalent and therefore rotamers. All rotamers of a MC are contained in a rotation class. A representative rotamer of a rotation class was defined in the following way.:

A MC  $a_i$  belonging to a rotation class  $A = \{a_1, a_2, \dots, a_n\}$  is called the representative rotamer of  $A$  iff

$$\text{code } 1(a_i) = \min(\text{code } 1(a_1), \text{code } 1(a_2), \dots, \text{code } 1(a_n))$$

Let MC  $b$  be a rotamer of MC  $a$ . MC  $b$  can be looked at in such a way that it looks exactly like MC  $a$ . This means that a transformation of cube coordinates  $\{x, y, z\}$  to the cube coordinates  $\{x_r, y_r, z_r\}$  can be done to describe rotation. Figure 23 shows an example for the effect of a rotation (rotation with rotation number 3 from table 1 (next chapter)) on a typ 3 block.



**Figure 23:** For rotation the cube coordinates has to be transformed.

## 3.2.1.1. Algorithm for rotation

The transformed system of coordinates may be centered at each of the eight corners of a cube. For each edge there are three rotational possibilities for a new system of coordinates. This short consideration shows that a microconfiguration can, in the absence of rotational symmetry, generate 24 different microconfigurations by rotation. To generate these rotations 24 rotational transformations of the system of coordinates were used (table 2).

rotation number	$\mathbf{x}_r$	$\mathbf{y}_r$	$\mathbf{z}_r$
1	x	y	z
2	z	x	y
3	y	z	x
4	y	n-x	z
5	x	n-z	y
6	z	n-y	x
7	n-x	n-y	z
8	n-z	n-x	y
9	n-y	n-z	x
10	n-y	x	z
11	n-x	z	y
12	n-z	y	x
13	y	x	n-z
14	x	z	n-y
15	z	y	n-x
16	x	n-y	n-z
17	z	n-x	n-y
18	y	n-z	n-x
19	n-y	n-x	n-z
20	n-x	n-z	n-y
21	n-z	n-y	n-x
22	n-x	y	n-z
23	n-z	x	n-y
24	n-y	z	n-x

table 2

The transformation of each cell  $(x, y, z)$  to  $(x_r, y_r, z_r)$  is accompanied with an one to one map of the cell positions  $p$  to  $p_r$ :

$$p = n^2e + ny + x \longrightarrow p_r = n^2e_r + ny_r + x_r$$

All cells of the same block in a given MC  $a$  are labeled with the same block number. Cube  $b$  with transformed coordinates  $\{x_r, y_r, z_r\}$  contains no blocks in the beginning of the procedure. From the block arrangement in  $a$ , given by the labeled blocks, the block arrangement in  $b$  has to be determined. Path  $P$ , introduced in chapter 2.2.3. as path through cells of stepwise increasing cell position, is followed in  $b$ . With the rotation number for each  $p_r$  on  $P$  the cell position  $p$  in  $a$  is determined and the block number of  $p$  is copied to  $p_r$ . This results in MC  $b$  as rotamer of  $a$ . Using table 3 the new block orientations in  $b$  result from the old block orientations in  $a$  and the rotation number.

rotation number	block orientation																		
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
2	3	1	2	6	4	5	10	14	15	13	9	8	7	12	11	18	16	17	19
3	2	3	1	5	6	4	13	12	11	7	15	14	10	8	9	17	18	16	19
4	1	3	2	4	6	5	10	7	8	9	13	15	14	16	18	11	12	17	19
5	3	2	1	6	5	4	13	10	14	15	7	11	12	18	17	9	8	16	19
6	2	1	3	5	4	6	7	13	12	11	10	9	8	17	16	15	14	18	19
7	1	2	3	4	5	6	9	10	7	8	14	18	16	11	17	13	15	12	19
8	3	1	2	6	4	5	15	13	10	14	12	17	18	9	16	7	11	8	19
9	2	3	1	5	6	4	11	7	13	12	8	16	17	15	18	10	9	14	19
10	1	3	2	4	6	5	8	9	10	7	16	17	11	13	12	14	18	15	19
11	3	2	1	6	5	4	14	15	13	10	18	16	9	7	8	12	17	11	19
12	2	1	3	5	4	6	12	11	7	13	17	18	15	10	14	8	16	9	19
13	1	3	2	4	6	5	15	18	17	12	14	10	13	11	7	16	9	8	19
14	3	2	1	6	5	4	11	17	16	8	12	13	7	9	10	18	15	14	19
15	2	1	3	5	4	6	9	16	18	14	8	7	10	15	13	17	11	12	19
16	1	2	3	4	5	6	12	15	18	17	13	7	11	16	8	14	10	9	19
17	3	1	2	6	4	5	8	11	17	16	7	10	9	18	14	12	13	15	19
18	2	3	1	5	6	4	14	9	16	18	10	13	15	17	12	8	7	11	19
19	1	3	2	4	6	5	17	12	15	18	11	8	16	14	9	13	7	10	19
20	3	2	1	6	5	4	16	8	11	17	9	14	18	12	15	7	10	13	19
21	2	1	3	5	4	6	18	14	9	16	15	12	17	8	11	10	13	7	19
22	1	2	3	4	5	6	18	17	12	15	16	9	14	13	10	11	8	7	19
23	3	1	2	6	4	5	17	16	8	11	18	15	12	7	13	9	14	10	19
24	2	3	1	5	6	4	16	18	14	9	17	11	8	10	7	15	12	13	19

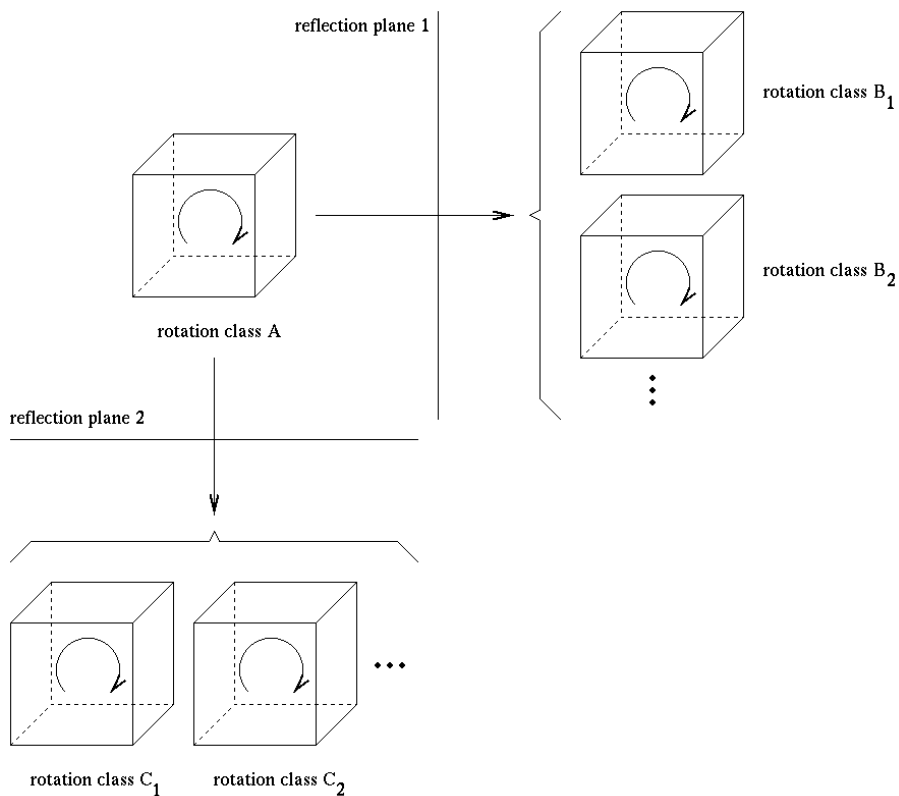
table 3

Because path  $P$  is followed in  $b$  the code 1 of  $b$  is generated automatically. If a block number occurs the first time on  $P$  in  $b$  the block orientation of the block is the next number in code 1 of  $b$ . The rotation modul calculates for every rotamer of a MC the corresponding code 1.

Together with the EC-algorithm all representative MCs of a subpartition can be calculated. First each MC is calculated by the EC-algorithm. If code 1 of the calculated MC is larger than code 1 of any of its rotamers (chapter 3.1.2.) it is ignored because it is no representative rotamer (chapter 3.2.1.).

### 3.2.2. Reflection symmetry

A set of rotamers belonging to rotation class  $A$  shall be reflected on several reflection planes. To how many rotation classes do the reflected rotamers belong? Figure 24 shows the general case for two reflection planes.



**Figure 24:** Reflection of rotamers belonging to the same rotation class on two reflection planes. The points indicate that more than two rotation classes could exist for reflection on both reflection planes.

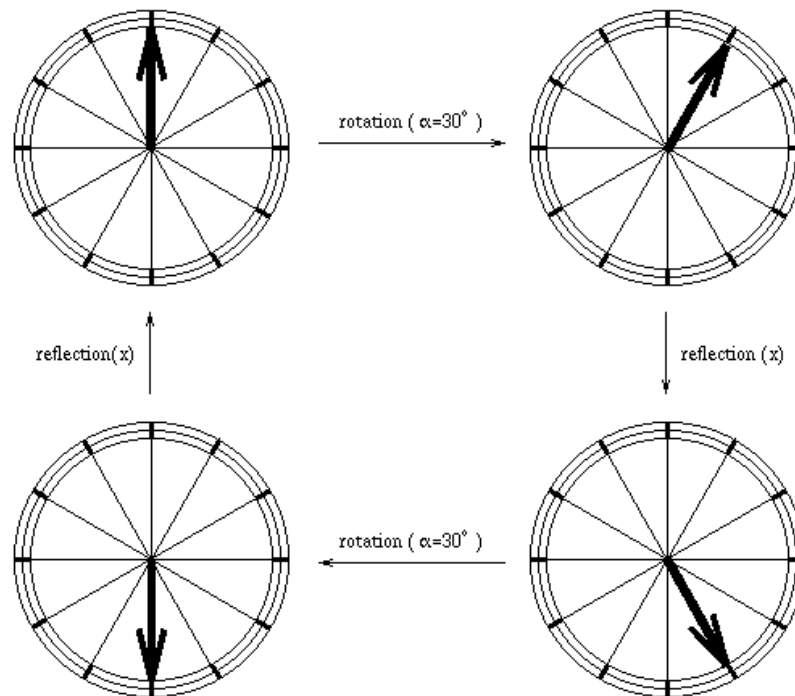
A closer look proves that the rotation classes  $C_1, C_2, \dots$  have to be pairwise identical with the rotation classes  $B_1, B_2, \dots$ . Consequently the problem is reduced to one reflection plane.

### 3.2.2.1. Reflection plane

To gain helpful equations a watch as a simple planar model is useful (figure 25). From this figure it can be seen that the following equation for the rotation operator  $O_{rotation}$  and the reflection operator  $O_{reflection}$  is true:

$$O_{rotation} \times O_{reflection} = O_{reflection}^{-1} \times O_{rotation}^{-1} = O_{reflection} \times O_{rotation}^{-1} \quad (1)$$

Rotation from 12 o'clock by  $+30^\circ$  followed by reflection on the x axis yields 5 o'clock. The same time can be reached by reflecting the watch on the x axis and doing an inverse rotation by  $-30^\circ$ . The planar case can be described by the z axis as rotation axis and a reflection plane containing the z vector.



**Figure 25:** Planar example



The three dimensional situation is not very different from the two dimensional one. The following consideration was done with the  $\{y, z\}$  reflection plane but there is no need for this choice. In the case of a microconfiguration there are three rotation axis (x, y, and z) that have to be considered. In the case of the y (or z) axis as rotation axis, the y (or z) vector is parallel to the  $\{y, z\}$  plane. Only for such cases equation 1 remains valid. With the x axis as rotation axis (like any other axis that is not parallel to the reflection plane) one have to be careful. Now rotation and reflection are independent operations and therefore another equation has to be applied.:

$$O_{rotation} \times O_{reflection} = O_{reflection} \times O_{rotation} \quad (2)$$

Table 4 gives an overview.

rotation axis	reflexion plane: $\{y, z\}$ plane
x	$O_{rotation} \times O_{reflection} = O_{reflection} \times O_{rotation}$
y	$O_{rotation} \times O_{reflection} = O_{reflection} \times O_{rotation}^{-1}$
z	$O_{rotation} \times O_{reflection} = O_{reflection} \times O_{rotation}^{-1}$

table 4

This can be generalized for a combined rotation operation:

$$(O_{rot,z} \times O_{rot,y} \times O_{rot,x}) \times O_{ref} = O_{ref} \times (O_{rot,z}^{-1} \times O_{rot,y}^{-1} \times O_{rot,x}) \quad (3)$$

If  $O_{rot,y}^{-1}$  and  $O_{rot,z}^{-1}$  are defined to be not inverse operators the corresponding operators on the left side of the equation become inverse. This yields:

$$(O_{rot,z}^{-1} \times O_{rot,y}^{-1} \times O_{rot,x}) \times O_{ref} = O_{ref} \times (O_{rot,z} \times O_{rot,y} \times O_{rot,x}) \quad (4)$$

or:

$$O_{ref} = (O_{rot,z}^{-1} \times O_{rot,y}^{-1} \times O_{rot,x})^{-1} \times O_{ref} \times (O_{rot,z} \times O_{rot,y} \times O_{rot,x}) \quad (5)$$

Equation 5 states that the reflexion of a  $MC_0$  on the  $\{y, z\}$  plane results in the same MC than the right rotation of any reflected rotamere of  $MC_0$ . All microconfigurations that are reflected on the  $\{y, z\}$  plane can be transformed into each other by rotation.

## 3.2.2.2. Algorithm for reflection

A MC has reflexion symmetry if the reflected form of this MC equals one of its rotameres. Such a MC can generate each of the reflected MCs by rotation, *independent* of the used reflection plane. This means:

In the case of the existence of reflection symmetry the reflected and not reflected microconfigurations belong to the same rotation class with one representative rotamere.

The used algorithm tests if the reflected MC equals one of its rotamers. As reflection plane the  $\{y, z\}$  plane is chosen. The coordinates and block orientations has to be transformed (table 5, table 6).

$X_{ref}$	n-x
$Y_{ref}$	y
$Z_{ref}$	z

table 5

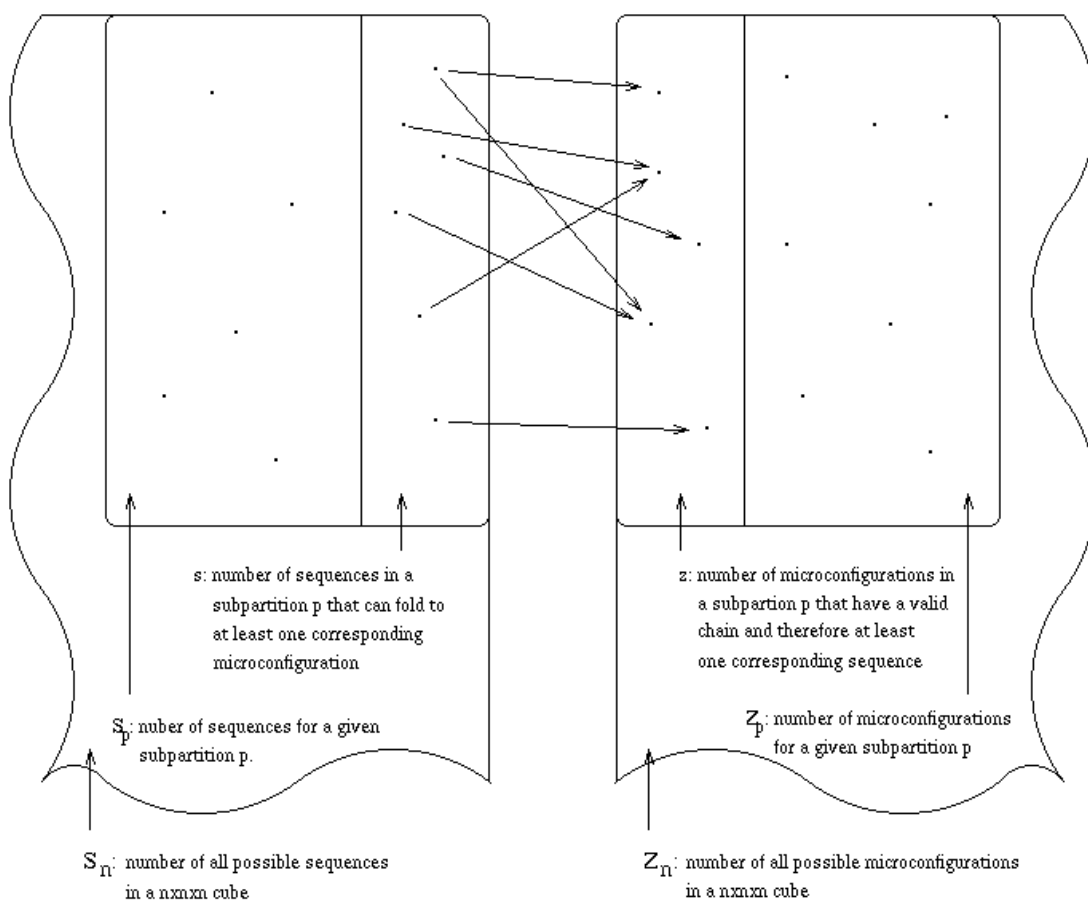
block orientation <sub>0</sub>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
block orientation <sub>reflected</sub>	1	2	3	4	5	6	9	8	7	10	16	18	14	13	15	11	17	12	19

table 6

## 4. Sequence Space And Microconfiguration Space

### 4.1. Sequence space

Figure 26 gives an overview over the correlation between sequence space and structure space. In general there is no 1 to 1 mapping between compact folding sequences and microconfigurations with chains.



**Figure 26:** Correlation between sequences and microconfigurations

### 4.1.1. Number of sequences in a given cube

#### 4.1.1.1. No $r_{3a}$ or $r_{3b}$ residues

The sequences shall be build by residues of typ  $r_1$  and  $r_2$ . Again the basic equations

$$r_1 + r_2 = r \tag{1}$$

$$r_1 + 2r_2 = n^d \tag{2}$$

are valid.  $n^d$  shall be written as number of cells  $c$ . From equation (2) it is clear, that

$$r_1 \in [0, c] \tag{3}$$

$$r_2 \in [0, \lfloor \frac{c}{2} \rfloor] \tag{4}$$

In contrast to  $r_2$  in (4),  $r_1$  cannot take on every value of the intervall in (3).

The total number of different sequences  $S$  from all different subpartitions is given by:

$$S_c = \sum_{r_2=0}^{\lfloor \frac{n^d}{2} \rfloor} S_{r_2} \tag{4}$$

For a given  $r_2$  (and  $r_1$ )  $S_{r_2}$  can be calculated:

$$S_{r_2} = \frac{r!}{r_1! \times r_2!} = \frac{r!}{(r - r_2)! \times r_2!} = \binom{r}{r_2} \tag{5}$$

Equations (1) and (2) give:

$$r = c - r_2 \tag{6}$$

The last equation combined with (4) and (5) results in the total number of sequences for a given cube.:

$$S_c = \sum_{r_2=0}^{\lfloor \frac{n^d}{2} \rfloor} \binom{c - r_2}{r_2} \tag{7}$$

#### 4.1.1.2. Use of all residues

Now the basic equations are:

$$r_1 + r_2 + r_{3a} + r_{3b} = r \quad (1)$$

$$r_1 + 2r_2 + 3r_{3a} + 3r_{3b} = c \quad (2)$$

As in section 1.1. the intervalls for  $r_{3b}, r_{3a}$ , and  $r_2$  can be given.:

$$r_{3b} \in [0, \lfloor \frac{c}{3} \rfloor] \quad (3)$$

$$r_{3a} \in [0, \lfloor \frac{c}{3} - r_{3b} \rfloor] \quad (4)$$

$$r_2 \in [0, \lfloor \frac{c - 3(r_{3a} + r_{3b})}{2} \rfloor] \quad (5)$$

After  $r_{3b}, r_{3a}, r_2$  has been choosen from the intervalls in (3), (4), (5), the remaining blocks are of typ  $r_1$ .

The total number of different sequences  $S$  from all different subpartitions is given by:

$$S_c = \sum_{r_2=0}^{\lfloor \frac{c-3(r_{3a}+r_{3b})}{2} \rfloor} \sum_{r_{3a}=0}^{\lfloor \frac{c}{3} \rfloor - r_{3b}} \sum_{r_{3b}=0}^{\lfloor \frac{c}{3} \rfloor} S_{r_2 r_{3a} r_{3b}} \quad (6)$$

$S_{r_2 r_{3a} r_{3b}}$  can be expressed as:

$$S_{r_2 r_{3a} r_{3b}} = \frac{r!}{r_1! \times r_2! \times r_{3a}! \times r_{3b}!} = \frac{r!}{r_2! \times r_{3a}! \times r_{3b}! \times (r - r_2 - r_{3a} - r_{3b})} \quad (7)$$

Equations (1) and (2) give:

$$r = c - r_2 - 2(r_{3a} + r_{3b}) \quad (8)$$

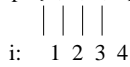
$$r - r_2 - r_{3a} - r_{3b} = r_1 = c - 2r_2 - 3(r_{3a} + r_{3b}) \quad (9)$$

These equations combined with (6) and (7) yield:

$$S_c = \sum_{r_2=0}^{\lfloor \frac{c-3(r_{3a}+r_{3b})}{2} \rfloor} \sum_{r_{3a}=0}^{\lfloor \frac{c}{3} \rfloor - r_{3b}} \sum_{r_{3b}=0}^{\lfloor \frac{c}{3} \rfloor} \frac{(c - r_2 - 2(r_{3a} + r_{3b}))!}{r_2! \times r_{3a}! \times r_{3b}! \times (c - 2r_2 - 3(r_{3a} + r_{3b}))!} \quad (10)$$

### 4.1.2. Sequences and microconfigurations

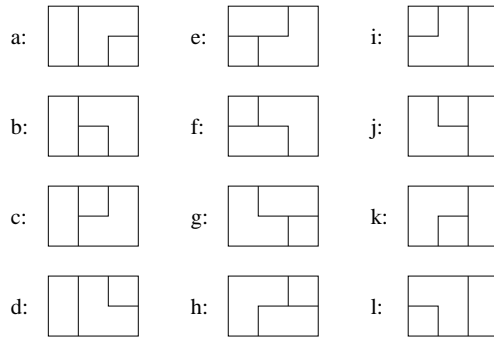
With  $p=\{1, 1, 0, 1\}$  the possible sequences  $s$  are:



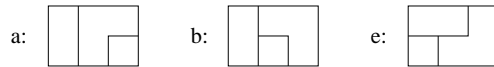
- $s$ : (1): 1, 2, 4  
 (2): 1, 4, 2  
 (3): 2, 1, 4  
 (4): 2, 4, 1 = (2)  
 (5): 4, 1, 2 = (3)  
 (6): 4, 2, 1 = (1)

(The numbers are the indices  $i$  of  $p$  and give the block types along the chain.)

there are 12 microconfigurations in a  $2 \times 3$  rectangle:



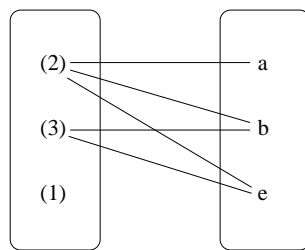
By rotation and reflexion of these microconfigurations one gets three not further reduceable microconfigurations. Such a set of microconfigurations is:



With  $C=\{\text{set of all possible chains through a, b, and e}\}$

Every foldable sequence is represented by a chain of the set  $C$ .

There is no one to one correlation between sequences and microconfigurations.:



One can see that there exist sequences like (1) that cannot fold as well as (in this special case) sequences like (2) that fold to every possible microconfiguration!

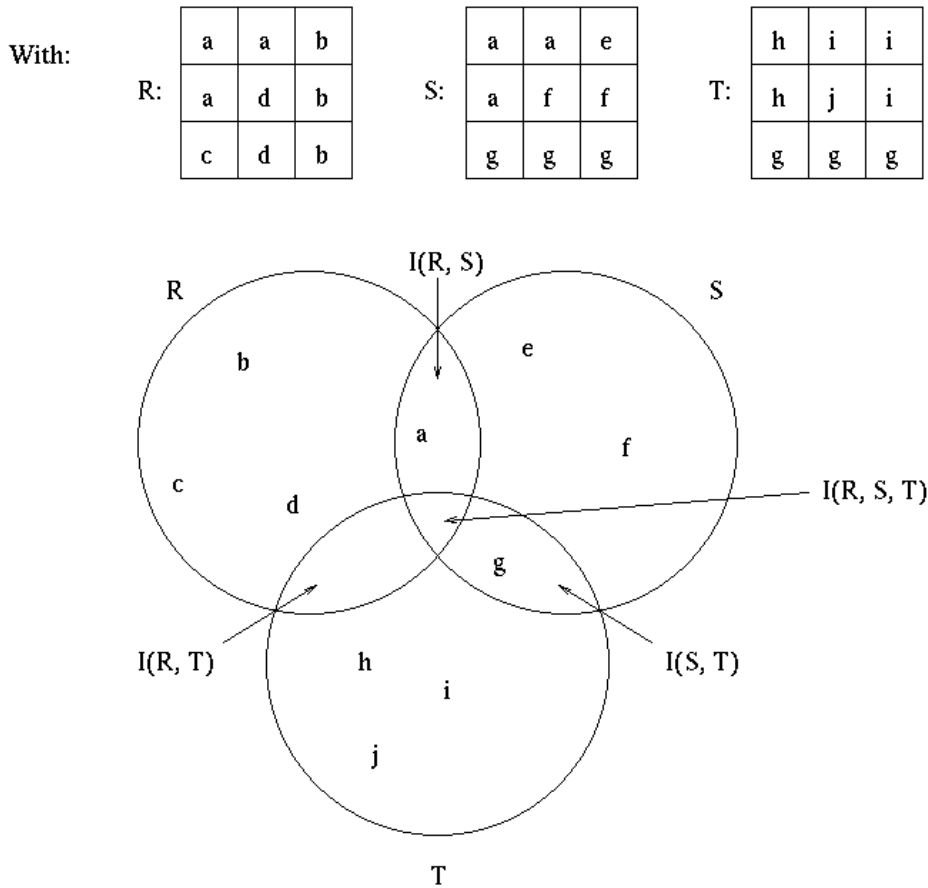
**Figure 27:** Example to visualize the kind of correlation between sequences and microconfigurations

## 4.2. Distance and microconfiguration space

To get a clearer idea of structural similarities between MCs of a given subpartition one have to compare them. For this purpose the microconfiguration space (= MCS) was developed. The MCs are represented by points in this finite space. Introducing the distance between two points as their number of different blocks the MCS is developed as finite metric space.

### 4.2.1. Distance $d(r, s)$ of two microconfigurations $r$ and $s$

Let the blocks of a MC  $x$  be elements of a blockset  $X$ . In a Venn diagram the blocks of a block set are represented by points in a circle. A Venn diagram of three block sets  $R, S$ , and  $T$  of MCs  $r, s$ , and  $t$  is shown in figure 28.



**Figure 28:** The block sets  $R, S$ , and  $T$  of MCs  $r, s$ , and  $t$  in a  $3 \times 3$  square belonging to the subpartition  $\{1, 1, 1, 1\}$  are shown. Different blocks are denoted by different letters.

MCs of a given subpartition have the same total number of blocks  $b$ . If  $|X|$  denotes the size of a block set  $X$  this means that

$$|R| = |S| = |T| = b \tag{1}$$

The term block is used very strictly. Two blocks are equal if they have the same block start position, the position of their first cell, and the same block orientation. Equal blocks of MCs  $r$  and  $s$  are elements of the intersection  $I(R, S)$  of the corresponding block sets  $R$  and  $S$  with

$$I(R, S) \doteq R \cap S \tag{2}$$

A space containing  $p_1, p_2, p_3$  as points has a metric  $d$  if:

- 1)  $d(p_1, p_2) \geq 0$  and  $d(p_1, p_2) = 0 \Leftrightarrow p_1 = p_2 \dots$  (definiteness)
- 2)  $d(p_1, p_2) = d(p_2, p_1) \dots$  (symmetry)
- 3)  $d(p_1, p_2) \leq d(p_1, p_3) + d(p_3, p_2) \dots$  (triangular inequality)

Lemma: The block distance  $d(.,.)$  is a metric on the microconfiguration space.

Proof:

(i) Definitness

The number of different blocks  $d(R, S)$  between two MCs  $r$  and  $s$  cannot be smaller than zero. If it is equal to zero the MCs  $r$  and  $s$  are equal (definitness).

(ii) Symmetry

The number of blocks of  $r$  that are not part of  $s$  ( $= d(R, S)$ ) shall be equal to the number of blocks in  $s$  that are not part of  $r$  ( $= d(S, R)$ ). It holds that

$$d(R, S) = b - |I(R, S)| \tag{3}$$

with  $I(R, S) = R \cap S = S \cap R = I(S, R)$  (equation (2)) one gets:

$$d(R, S) = b - |I(R, S)| = b - |I(S, R)| = d(S, R) \tag{4}$$



(iii) Triangular inequality

For MCs  $r, s$  and  $t$  the intersection of the block sets  $R, S$ , and  $T$  is  $I(R, S, T)$ . It holds that

$$I(R, S, T) \doteq I(S, R) \cap I(R, T) = I(R, S) \cap I(S, T) = I(R, T) \cap I(T, S) \quad (5)$$

Because an intersection of two sets is a subset of the sets that are intersected it is clear that

$$I(R, T) \cap I(T, S) = I(R, S, T) = I(R, S) \cap I(S, T) \in I(R, S) \quad (6)$$

If  $|X|$  denotes the size of set  $X$  equation (6) can be written:

$$|I(R, T) \cap I(T, S)| \leq |I(R, S)| \quad (7)$$

Addition of  $|I(R, T)| + |I(T, S)|$  on both sides of (7) gives:

$$|I(R, T)| + |I(T, S)| - |I(R, S)| \leq |I(R, T)| + |I(T, S)| - |I(R, T) \cap I(T, S)| \quad (8)$$

A look on the Venn diagram (figure 28) makes clear that the right side of equation (8) can be expressed easier:

$$|I(R, T)| + |I(T, S)| - |I(R, S)| \leq |I(R, T) \cup I(T, S)| \quad (9)$$

Because of

$$|I(T, X_1) \cup I(T, X_2) \dots \cup I(T, X_n)| \leq |T| \quad (10)$$

for any sets  $X_1, X_2 \dots X_n$  and  $T$  an upper boundary for the right side of equation 9 can be given. This results in

$$|I(R, T)| + |I(T, S)| - |I(R, S)| \leq |T| \quad (11)$$

Combination of equation (1) and (11) gives

$$|I(R, T)| + |I(T, S)| - |I(R, S)| \leq b \quad (12)$$

The use of equation (3) yields

$$(b - d(R, T)) + (b - d(T, S)) - (b - d(R, S)) \leq b \quad (13)$$

or

$$d(R, S) \leq d(R, T) + d(T, S)$$

q.e.d.

**4.2.2. Rotation reduced distance  $d(p, R_q)$**

In the last section (4.2.1.) it was shown that  $d(p, q)$  between two MCs  $p$  and  $q$  is a metric on the MCS. A second kind of distance is the distance between a MC and a set of MCs. Let  $R_q$  be the rotation class of  $q$  containing all MCs that are rotational equivalent to  $q$ . The distance  $d(p, R_q)$  between  $p$  and  $R_q$ , a distance between a point  $p$  and a not empty subset  $R_q$  of the MCS, is defined to be

$$d(p, R_q) = \inf\{d(p, q) : q \in R_q\}$$

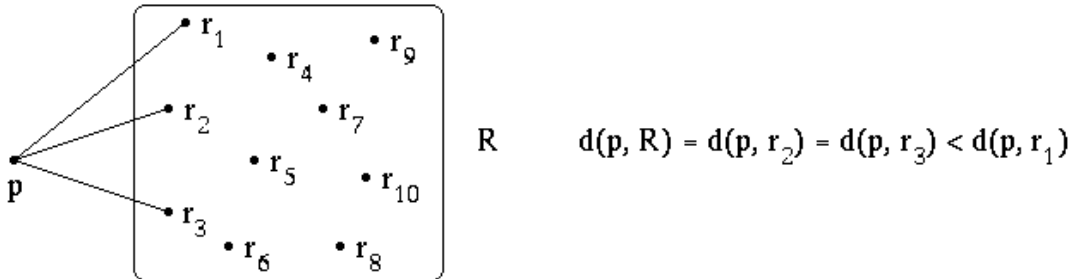
This means that  $d(p, R_q)$  is gained for  $p$  and  $q$  by rotation of  $q$  until its distance to  $p$  is at a minimum value  $d_{min}(p, q_{rot})$ . Of course  $q$  can already be at a minimum distance  $d(p, q)$  to  $p$  and

$$d(p, R_q) = d_{min}(p, q_{rot}) = d(p, q)$$

A MC  $q \in R_q$  was called neighbored to MC  $p$  if

$$d(p, R_q) = d(p, q)$$

Not neighbored MCs have a  $d(p, q) > d(p, R_q)$ . More than one neighbored MCs with the same minimum value  $d(p, R_q)$  can exist as shown in figure 29.



**Figure 29:** More than one MCs of the same rotation class can be neighbored to  $p$

## 5. Block permutations and graphs

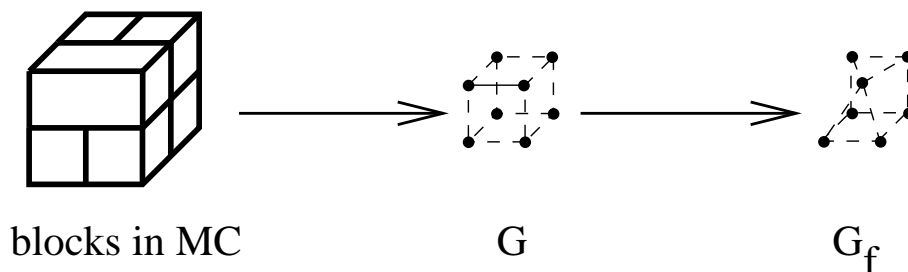
### 5.1. Representation of block sets

#### 5.1.1. Coarse grained representation of block sets

With a given set of blocks the position of block cells and the orientation of blocks have to be known. There exist coarse grained representations of block sets with certain features in common.

A uniform translation of all blocks in a block set  $S$  as near to the origin as possible results in a block set  $S_t$ . In  $S_t$  the information of block orientation and neighbourhood of cells in  $S$  is still given but knowledge on the position of cells in  $S$  is lost.  $G$  of a microconfiguration provides information on the neighbourhood of cells and with  $G_f$  the neighbourhood of blocks is known.  $G(S)$  and  $G_f(S)$  are induced subgraphs of  $G$  and  $G_f$  describing the blocks of a given block set  $S$ .  $G(S)$  and  $G_f(S)$  contain no information what cells or blocks are neighbored to each other.

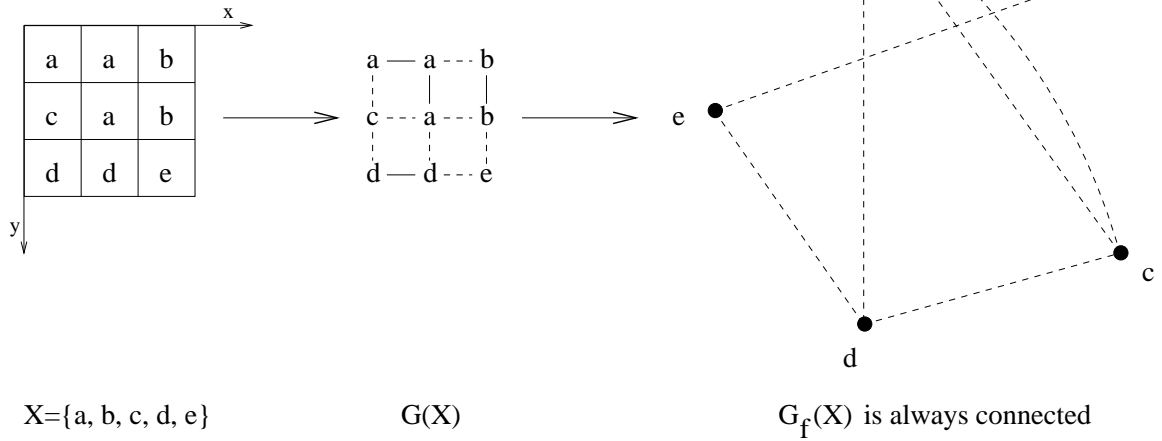
Figure 30 shows an example for  $G$  and  $G_f$ .



**Figure 30:** Generation of  $G$  and  $G_f$  from a MC in a  $2 \times 2 \times 2$  cube with one block consisting of two cells.

$G_f(S)$  is called connected if a path between every pair of vertices in  $G_f(S)$  exists. An example for a connected and not connected graph in a microconfiguration is given in figure 31.

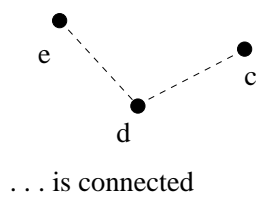
$G_f(X)$  with  $X$  as block set of all blocks in MC  $x$  :



$G_f(S)$  with  $S$  as subset of  $X$  :

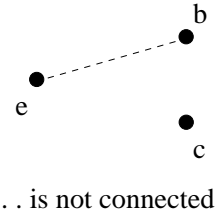
$S_1 = \{c, d, e\}$ :

$G_f(S_1)$ :



$S_2 = \{b, c, e\}$ :

$G_f(S_2)$ :



**Figure 31:**  $G_f(S)$  has not to be connected

If two blocks of set  $S$  are neighbours in  $x$  the corresponding two vertices in  $G_f(S)$  are linked by at least one edge.

Two graphs  $G = (V, E_1, E_2)$  and  $G' = (V', E'_1, E'_2)$  are equal if they are isomorphic.

### 5.1.2. Projection of block set representations

$S, S_t, G(S), G_f(S)$  are the block set representations of a block set  $S$ . They contain information on size, orientation, and neighbourhood of blocks.  $G(S_t)$  equals  $G(S)$  because block type and relative orientation of blocks in  $S_t$  and  $S$  is the same.

$S$ : size and orientation of blocks and position of their cells

$S_t$ : size and orientation of blocks and neighbourhood of their cells

$G(S_t)$ : size of blocks and neighbourhood of their cells

$G_f(S_t)$ : size and neighbourhood of blocks

The projections  $t$ ,  $g$ , and  $f$  are defined as:

$$t : S \mapsto S_t$$

$$g : S_t \mapsto G(S_t) = G(S)$$

$$f : G(S_t) \mapsto G_f(S_t) = G_f(S)$$

Let the different blocks of two MC  $x$  and  $x'$  belong to block set  $S$  and  $S'$ . The block set representations of  $S$  and  $S'$  are given by

$$S \xrightarrow{t} S_t \xrightarrow{g} G(S_t) \xrightarrow{f} G_f(S_t)$$

$$S' \xrightarrow{t} S'_t \xrightarrow{g} G(S'_t) \xrightarrow{f} G_f(S'_t)$$

The block sets  $(S, S')$  and their coarse grained representations  $(S_t, S'_t)$ ,  $(G(S_t), G(S'_t))$ , and  $(G_f(S_t), G_f(S'_t))$  are converted into each other by permutations. In the next section a method to change blocks of a block set  $S$  is developed.

### 5.2. Cell migrations as method to modify blocks in a block set

In a block permutation  $\pi$  the blocks of  $S$  and  $S'$  occupy the same cells of the cube. The blocks of  $S'$  have to be the rearranged blocks of  $S$ .

$$\pi : S \mapsto S'$$

In this chapter cell migrations are developed as method to change a set of blocks  $S$  into a any set of different blocks  $S'$  containing the same cells than  $S$ .

(a) Migration of a single cell

In the concept of cell migration, or migration for short, an initial unchanged block as donor block contains a migrating cell.

- (1) The migrating cell separates in a first step from its donor block. The cells of the donor block that do not migrate remain part of the donor block.
- (2) In a second step, the migration step, the free migrating cell fuses with an acceptor cell. The arrow starting at the migrating cell and heading to the acceptor cell is called the migration path of the migrating cell.

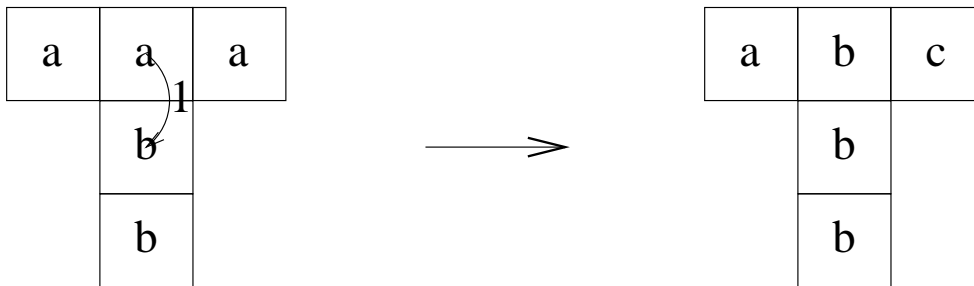
The acceptor cell as part of the acceptor block can be:

- (i) the free migrating cell to form a new singlecelled block. This kind of migration is named reflexive migration.



**Figure 32:** Reflexive migration

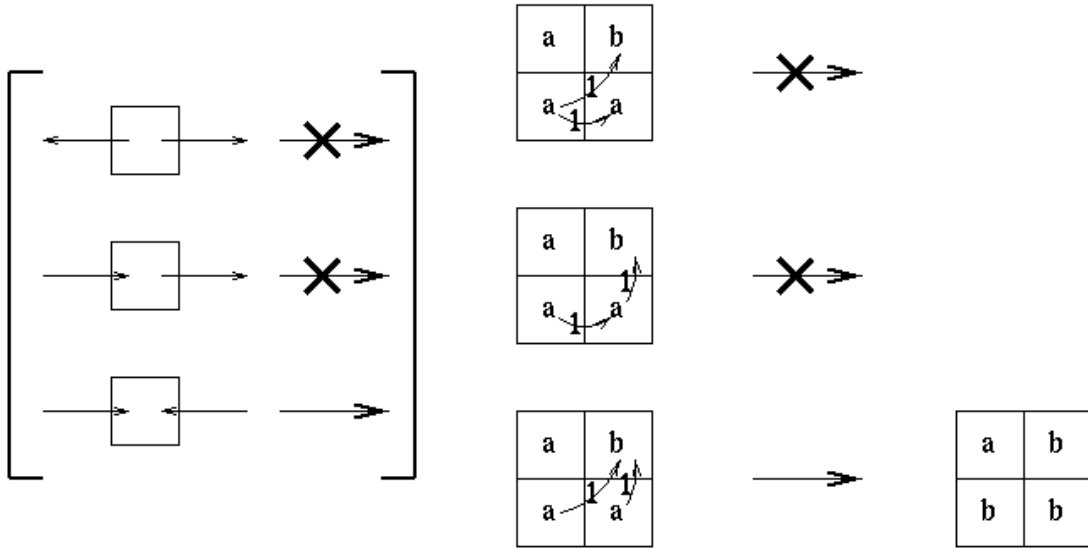
- (ii) a cell neighbouring the donor block. By the fusion of the migrating cell with the acceptor cell the migrating cell becomes part of the acceptor block. The resulting migration is referred to as nonreflexive cell migration.



**Figure 33:** Nonreflexive migration

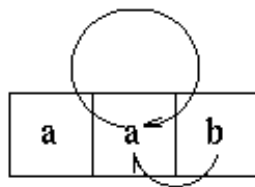
(b) Rules for combined migrations

- (i) A migrating cell has exactly one outgoing arrow. (An accepting cell may have more than one incoming arrow).
- (ii) The sequential concatenation of migrations is forbidden (because composition of sequential migration is not commutative).



**Figure 34:** Application of the two selection rules for migration

Application of these rules on migration results always in a unique result. As an example Figure 35 shows a combined reflexive and nonreflexive migration.



**Figure 35:** The reflexive migration has to be carried out first

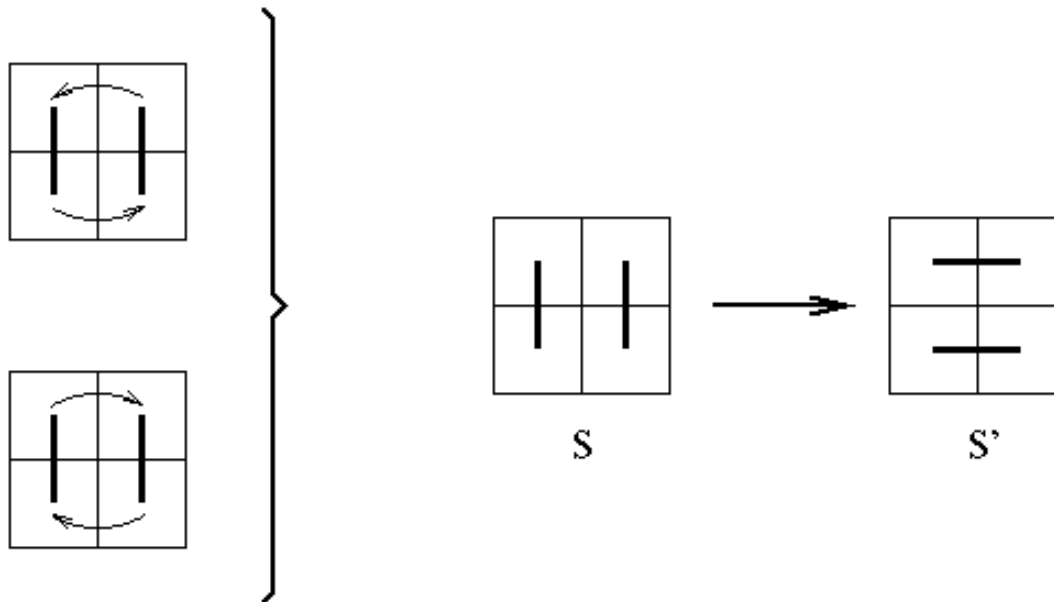
In example 1 there are two possible ways to apply the two migrations but only one of these ways is legal.

- (1) If the reflexive migration is carried out first, an outgoing arrow leaves the migrating cell  $c_1$  and in a second step an incoming arrow enters  $c_1$  as accepting cell of itself. Only the last action in  $c_1$ , described by the incoming arrow, is important for the next action of this cell. The second migration, fusion of the migrating cell  $c_2$  with  $c_1$  as acceptor cell, results in a second incoming arrow in  $c_1$ . A cell, like  $c_1$ , may be accepting cell of more than one cells. This combination of migrations is therefore legal.
- (2) If the nonreflexive migration is carried out first, the migrating cell  $c_2$  has  $c_1$  as acceptor cell and an incoming arrow enters  $c_1$ . Application of the reflexive migration would be started by an outgoing arrow. This would be a sequential concatenation of migrations, which is forbidden by rule (ii).

As a consequence of this consideration the reflexive migration has to be carried out first.

(c) Equivalence of combined migrations

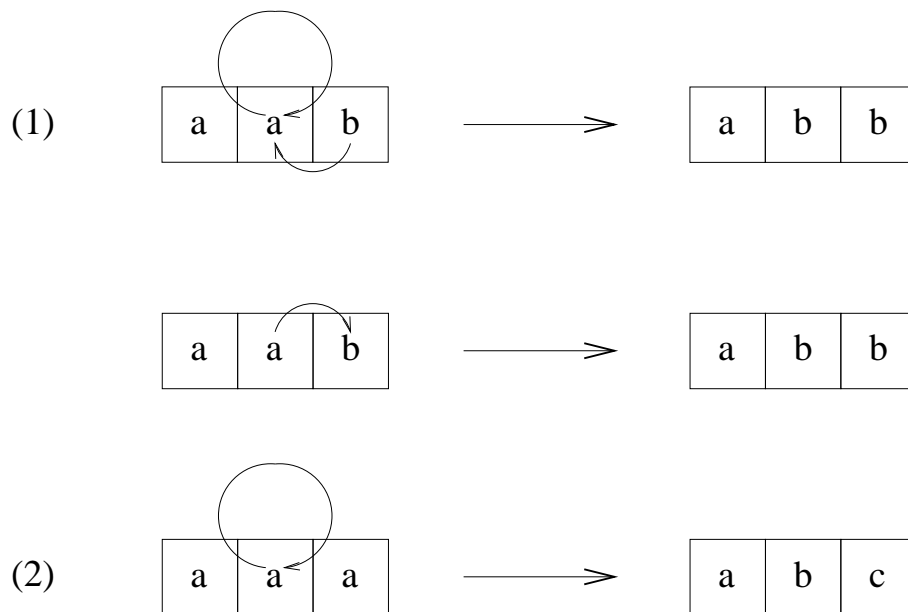
Combined migrations with the same block sets  $S, S'$  are equivalent (figure 36).



**Figure 36:** Equivalent combined migrations have the same effect on a given block set.



To reduce the number of equivalent migrations the use of reflexive migration is restricted to those cases where it cannot be avoided. The combined migration of figure 35 is legal but, as shown in example (1) of figure 37, can be substituted by one single nonreflexive migration.



**Figure 37:** The reflexive migration in (1) can be substituted. In (2) there is no equivalent nonreflexive migration.

If a cell after reflexive migration is an acceptor cell for cells of surrounding donor blocks, an equivalent nonreflexive migration is always possible. An isolated reflexive migration on the other hand increases the number of blocks and has no equivalent nonreflexive migration. To avoid complicated migration pattern reflexive migration is only used as isolated reflexive migration.

In a permutation the number of blocks in  $S$ , and  $S'$  may not differ. Consequently it holds that

- (1) A permutation of two blocks contains no reflexive migration.
- (2) One of three permuted blocks may have a reflexive migration. A more detailed discussion is given in section 5.3.1..

### 5.3. Permutation of block set representations

The permutations  $\pi, \pi', \gamma$ , and  $\varphi$  are defined as:

$$\left. \begin{array}{l} \pi : S \longmapsto S' \\ \pi' : S_t \longmapsto S'_t \\ \gamma : G(S_t) \longmapsto G(S'_t) \\ \varphi : G_f(S_t) \longmapsto G_f(S'_t) \end{array} \right\} \text{resulting in: } \begin{array}{cccc} S \xrightarrow{t} S_t \xrightarrow{g} G(S_t) \xrightarrow{f} G_f(S_t) & & & \\ \downarrow \pi & \downarrow \pi' & \downarrow \gamma & \downarrow \varphi \\ S' \xrightarrow{t} S'_t \xrightarrow{g} G(S'_t) \xrightarrow{f} G_f(S'_t) & & & \end{array}$$

The sets P, M, Q, and R contain all possible permutations  $\pi, \pi', \gamma$ , and  $\varphi$ :

$$\pi = (S, S') \in P$$

$$\pi' = (S_t, S'_t) \in M$$

$$\gamma = (G(S_t), G(S'_t)) \in Q$$

$$\varphi = (G_f(S_t), G_f(S'_t)) \in R$$

The size  $|P|$ , or number of elements, of set  $P$  is larger than than  $M$  because block sets being different from  $S$  can have the same  $S_t$ . The loss of specificity in  $S, S_t, G(S_t)$ , and  $G_f(S_t)$  results in

$$|P| \geq |M| \geq |Q| \geq |R|$$

In this section the permutations  $\pi, \pi', \gamma$ , and  $\varphi$  are described as migration processes.

(a) Block permutation  $\pi$

Let  $x$  and  $y$  be MCs belonging to the same subpartition. The sets  $X, Y, S, S'$  are defined as:

$$X \doteq \{ \text{blocks in } x \text{ as ordered pairs of (block startposition, block orientation)} \}$$

$$Y \doteq \{ \text{blocks in } y \text{ as ordered pairs of (block startposition, block orientation)} \}$$

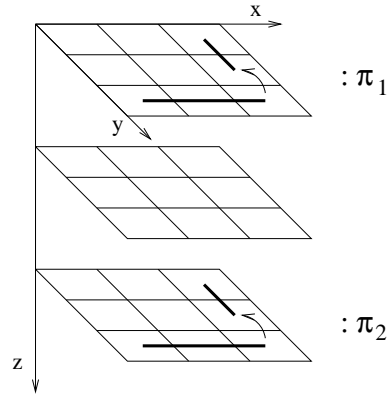
$$S \doteq X \setminus (X \cap Y)$$

$$S' \doteq Y \setminus (X \cap Y)$$

A block permutation  $\pi$  changes  $x$  into  $y$  and  $S \subset X$  into  $S' = \pi(S) \subset Y$ . It is defined as a restricted set of migrations, such that neither the number nor the type of blocks is changed.

$$\pi : S \longmapsto S'$$

Figure 38 shows two different block permutations  $\pi_1$  and  $\pi_2$ .



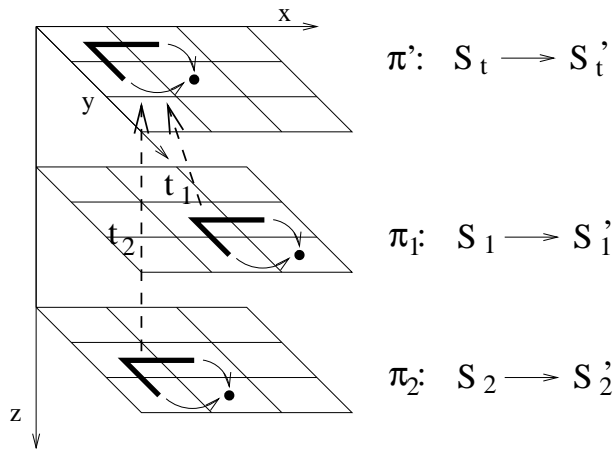
**Figure 38:** A permutation acts on a well defined set of blocks.

(b) Move  $\pi'$

The move  $\pi'$  is defined as:  $\pi' : S_t \mapsto S'_t$

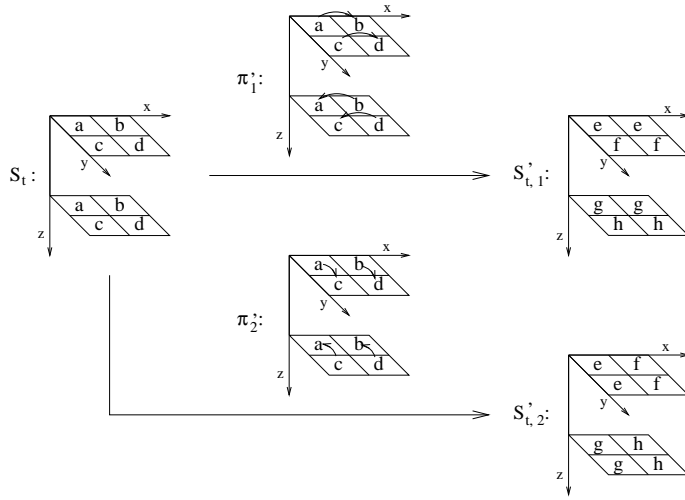
A translation  $t$  of permutation  $\pi = (S, S')$  as near to the center of the coordinate system as possible results in  $\pi' = (S_t, S'_t)$ .

- (i) Every cell of  $S$  is translated to the corresponding cell of  $S_t$  (figure 39).
- (ii) Corresponding, translational equivalent, cells of  $S$  and  $S_t$  migrate in the same way.



**Figure 39:** Function  $t$  translates the permuted blocks as near as possible to the origin.

The two different permutations  $\pi_1, \pi_2$  of figure 40 would have the same move if rotation of  $S$  and  $S'$  was permitted additionally to translation in  $\gamma = (S_t, S'_t)$ .



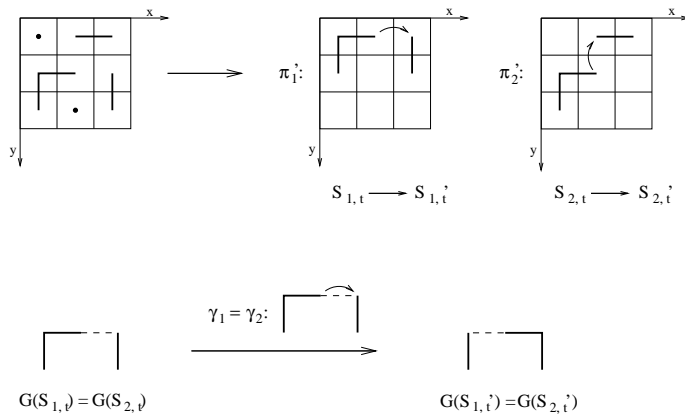
**Figure 40:**  $p_1$  and  $p_2$  are different permutations. If they would act in an asymmetric block surrounding, different MC's would be the result.

The move set M2 (M3, M4, ...) contains all moves that belong to permutations of 2 (3, 4, ...) blocks.

(c) Graph permutation  $\gamma$

The graph permutation  $\gamma$  is defined as:  $\gamma : G(S_t) \mapsto G(S'_t)$

Two different moves  $\pi'_1 = (S_{t,1}, S'_{t,1})$  and  $\pi'_2 = (S_{t,2}, S'_{t,2})$  with the same migration pattern have the same graph permutation  $\gamma$  if their graphs  $G(S_{1,t})$  and  $G(S_{2,t})$  are equal (figure 41).



**Figure 41:** Very similar moves have the same graph permutation.

(d) Move type  $\varphi$

The move type  $\varphi$  describes the migration pattern between blocks of certain size. It is defined as:

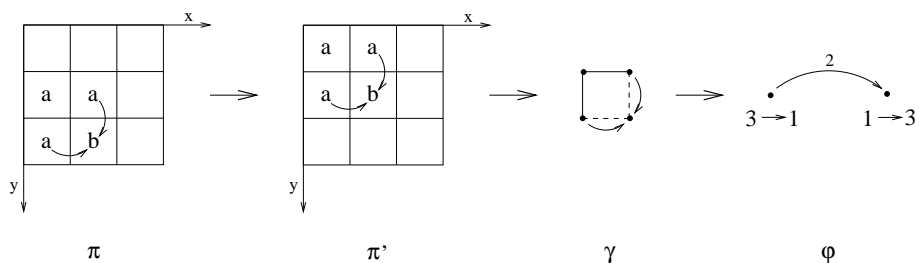
$$\varphi : G_f(S_t) \mapsto G_f(S'_t)$$

Each vertex in  $\varphi$  represents a block of  $S_t$  that is changed by migration into a different block of  $S'_t$ . The block size of these blocks is noted at each vertex. The edges of neighboured blocks in  $G_f(S_t)$  and  $G_f(S'_t)$  are omitted in  $\varphi$ .

The number of migrating cells is given on the migration paths because for fused migration paths it is larger than one.

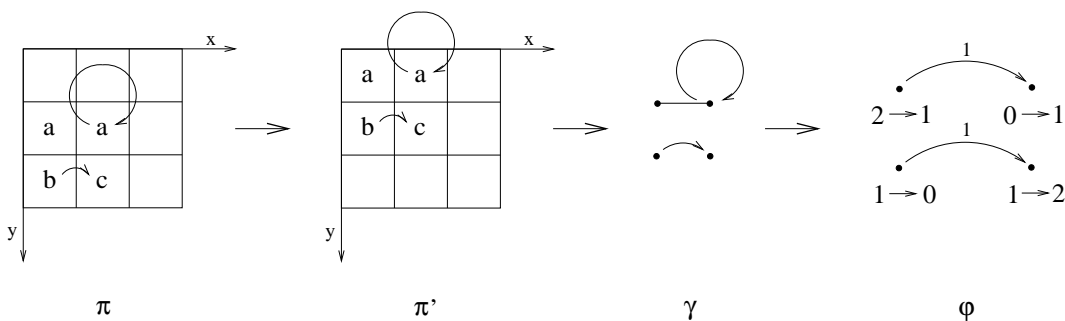
Fusion of migration paths is done in the following way:

- (i) Two nonreflexive migration paths of  $\gamma$  add in  $\varphi$  if their start vertices in  $G(S_t)$  are fused to one vertex in  $G_f(S_t)$  and their end vertices in  $G(S_t)$  are fused to one end vertex in  $G_f(S_t)$ .



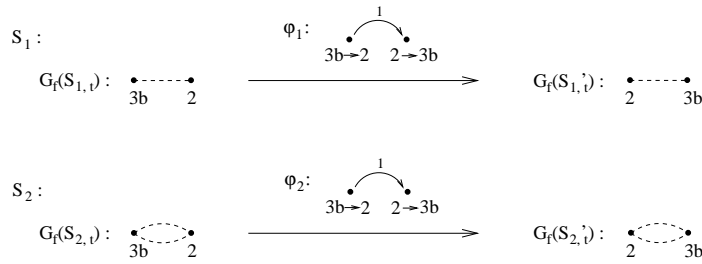
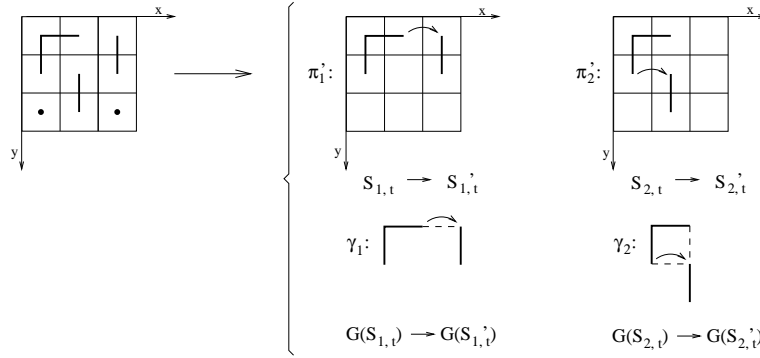
**Figure 42:** Addition of nonreflexive migrations in a move type

- (ii) Reflexive migration paths become migration paths ending at a acceptor block with size zero.



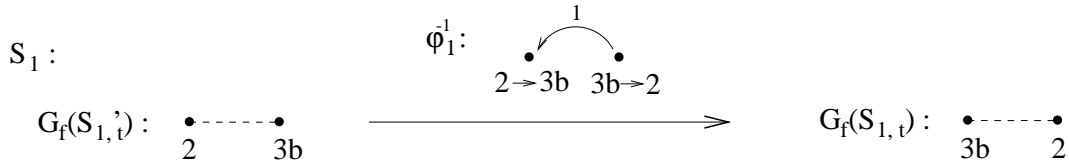
**Figure 43:** Generation of a new block is indicated by an acceptor block of size zero

The two moves of figure 44,  $\pi'_1 = (S_{t,1}, S'_{t,1})$  and  $\pi'_2 = (S_{t,2}, S'_{t,2})$ , have different graph permutations  $\gamma$  but equal move types  $\varphi$ .



**Figure 44:** Different graph permutations  $\gamma$  with equal move types  $\varphi$

For every move  $\pi' = (S_f, S'_f)$  with move type  $\varphi$  exists the inverse move  $(\pi')^{-1} = (S'_f, S_f)$  with move type  $\varphi^{-1}$ . In  $\varphi^{-1}$  all arrows of  $\varphi$  are inverted.  $\varphi^{-1}$  for the example  $\varphi_1$  of figure 45 is:



**Figure 45:** In  $\varphi_1^{-1}$  all arrows of  $\varphi^{-1}$  (figure 44) are simply inverted.  $\varphi_1^{-1}$  and  $\varphi^{-1}$  describe the same move type.

Because  $\varphi$  and  $\varphi^{-1}$  describe the same migration pattern with the same pairs of corresponding blocks in  $S_f$  and  $S'_f$  they are two equivalent descriptions of the same move type.

### 5.3.1. Developement of move type sets $R2$ and $R3$

#### 5.3.1.1. Considerations on move types in $R2$ and $R3$

The move type set  $R3$  contains a set of nonequivalent migration pattern of all possible permutations of three blocks. It contains  $R2$ . The following considerations on  $R3$  include therefore move types of  $R2$  as specialcase.

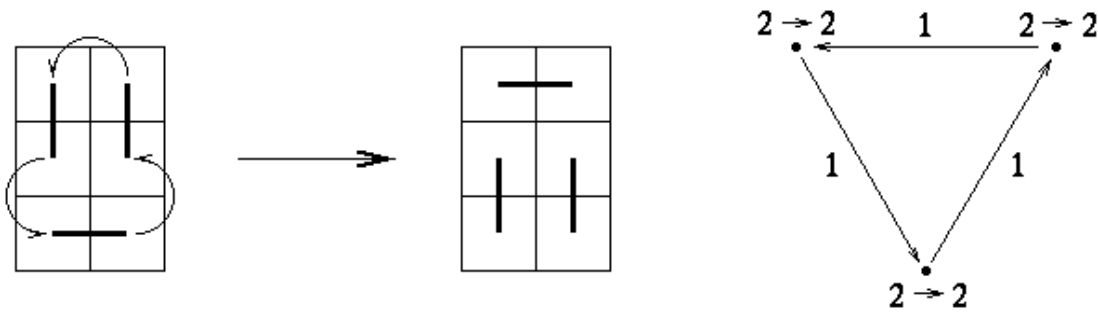
#### (A) Possible permutations

With  $bl$  blocks the number of possible permutations of block size is  $bl!$ . For three blocks  $bl! = 6$  and for two blocks  $bl! = 2$ .

$$\begin{pmatrix} a \\ b \\ c \end{pmatrix} \rightarrow \begin{pmatrix} a \\ b \\ c \end{pmatrix}, \begin{pmatrix} a \\ c \\ b \end{pmatrix}, \begin{pmatrix} b \\ a \\ c \end{pmatrix}, \begin{pmatrix} b \\ c \\ a \end{pmatrix}, \begin{pmatrix} c \\ a \\ b \end{pmatrix}, \begin{pmatrix} c \\ b \\ a \end{pmatrix}$$

$$\begin{pmatrix} a \\ b \end{pmatrix} \rightarrow \begin{pmatrix} a \\ b \end{pmatrix}, \begin{pmatrix} b \\ a \end{pmatrix}$$

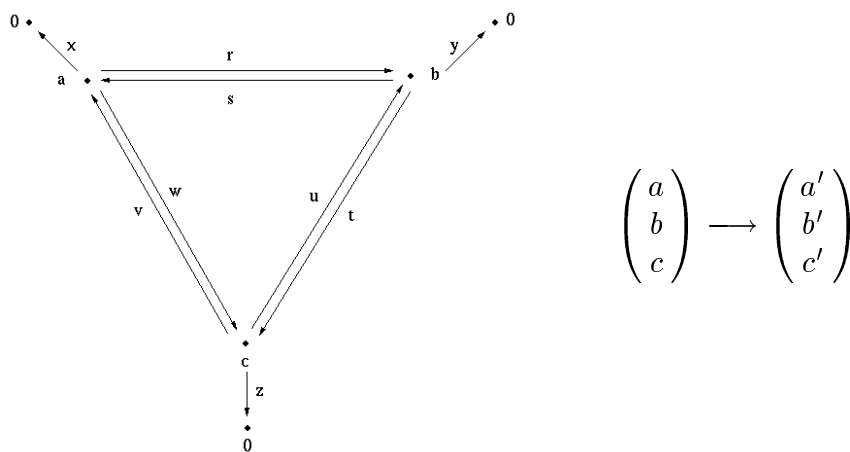
The size of all blocks may be unchanged (figure 46).



**Figure 46:** Example of a move where each block is changed into a block of equal size

(B) Schematic representation of permutations

The most general description of a  $R3$  move type is given by figure 47.



**Figure 47:** General description of a move type belonging to  $R3$

If the move of the move type has no reflexive migration paths  $x$ ,  $y$ , and  $z$  are zero. In this case a system of three equations is implied with  $|a|$  as size, the number of cells, in block  $a$ :

$$\begin{aligned} |a| + v + s - r - w &= |a'| \\ |b| + r - s + u - t &= |b'| \\ |c| + t - u + w - v &= |c'| \end{aligned}$$

Rearrangement of the first and last equation gives:

$$\begin{aligned} r &= s + (v - w) + (|a| - |a'|) \\ t &= u + (v - w) + (|c| - |c'|) \end{aligned}$$

The general linear move type with no parallel edges is a special case with:

$$v = w = 0; \quad s = 0; \quad u = 0$$

The general linear case is therefore:

$$a \xrightarrow{|a|-|a'|} b \xrightarrow{|c'|-|c|} c$$



(C) Use of selection rules

To find a legal move type between blocks of fixed size, several migration pattern can be tried (figure 48). A migration pattern has to obey the rules of migration and neither an old block nor a newly formed block may have a block size larger than three. Let  $i$  be the number of incoming migration paths and  $o$  be the number of outgoing migration paths. The difference  $\Delta = i - o$  is then the size change of the block. The minimal possible block size, the necessary number of cells in the block, is given by the use of the two selection rules. The number of incoming migration paths may have the same acceptor cell while each outgoing migration path has to start from a different cell. If there are only outgoing migration paths and no new block is formed by reflexive migration in the corresponding move, the block has to be one cell larger than  $o$ . It holds therefore that

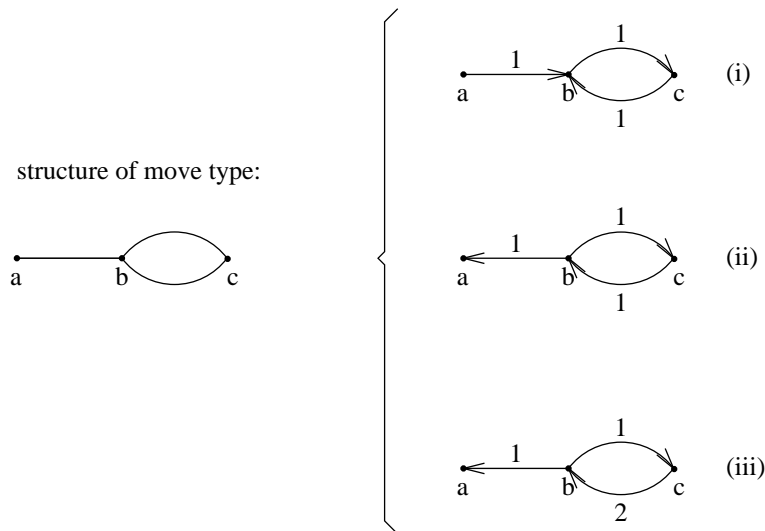
$$\Delta \doteq i - o$$

$$s_{min} \doteq \begin{cases} o & \text{if } i = 0 \text{ and at least one new blocks is formed} \\ & \text{or } o = 0 \text{ and at least one block is removed} \\ o + 1 & \text{otherwise} \end{cases}$$

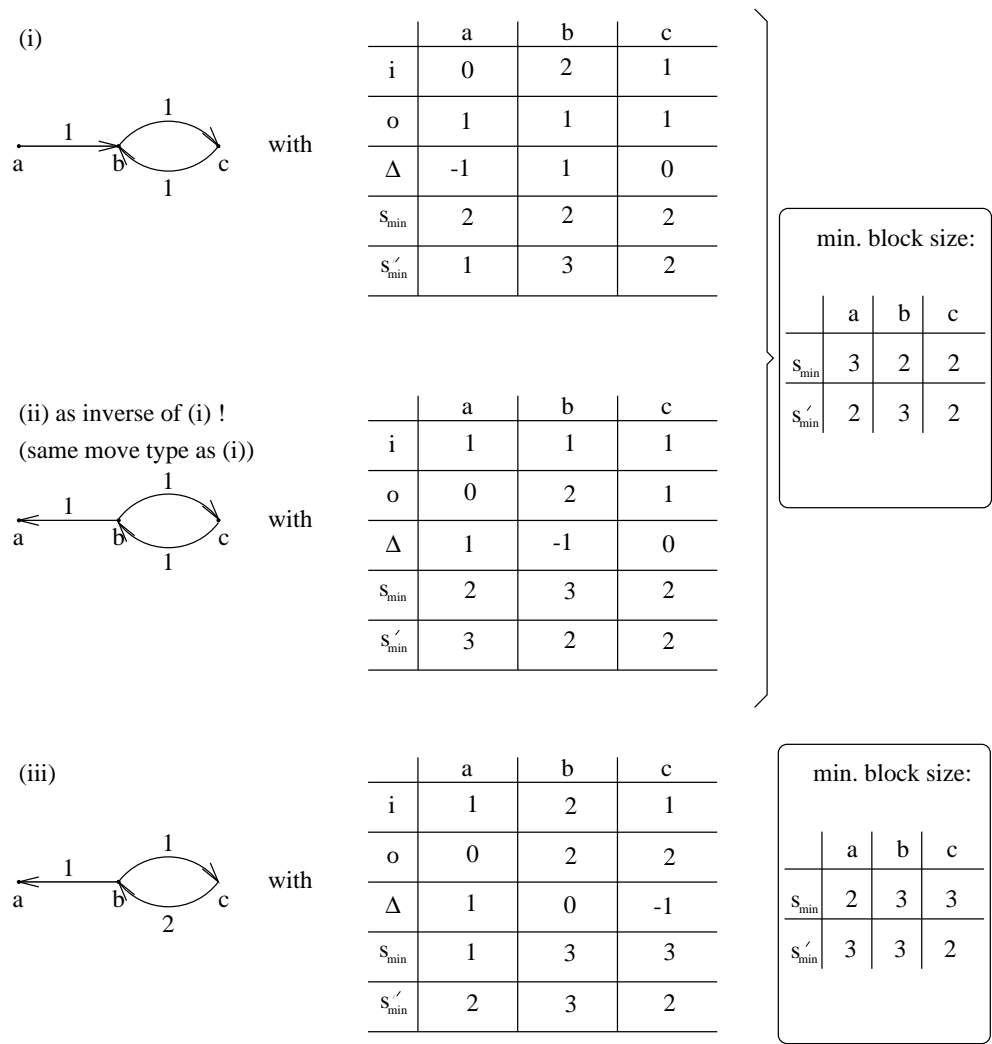
$$s'_{min} \doteq s_{min} + \Delta$$

$$\left. \begin{matrix} s_{min} \leq 3 \\ s'_{min} \leq 3 \end{matrix} \right\} \text{restriction on block size}$$

These equations distinguish between theoretically possible and impossible move types. If a move type is theoretically possible it has to describe at least one permutation. This procedure is demonstrated in figure 48, 49, and 50.

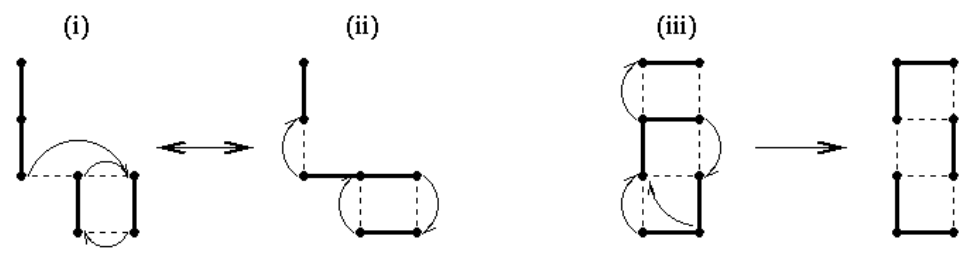


**Figure 48:** Three different move type candidates of given structure



**Figure 49:** The minimal block sizes before ( $s_{min}$ ) and after the move ( $s'_{min}$ ) have to be calculated to decide if a move type candidate for a given structure is legal.

For all three theoretically possible move types examples can be given. Figure 50 shows graph permutations as examples of the move types (i), (ii), and (iii).

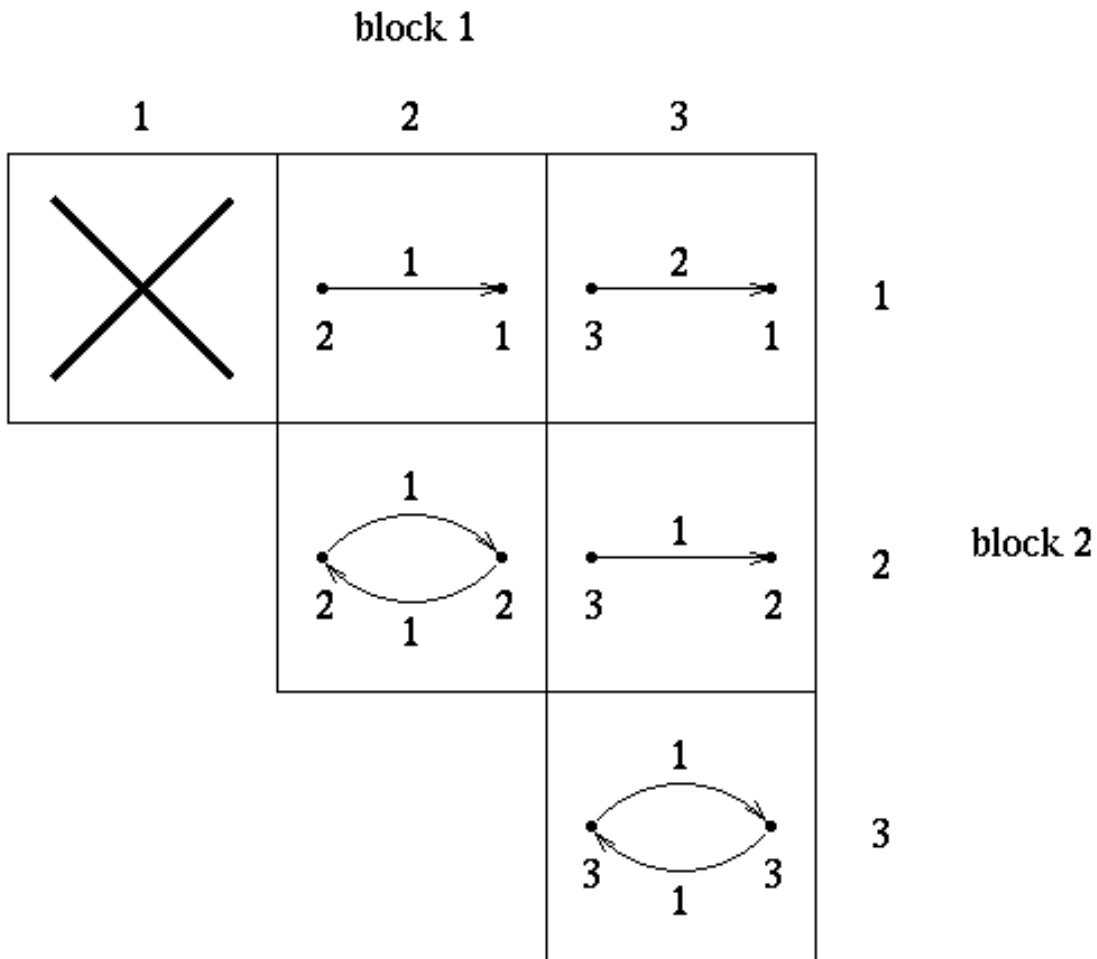


**Figure 50:** For a move type that passes the minimal block size test a permutation has to exist.

5.3.1.2. Move type sets  $R_2$  and  $R_3$ (A) Move types of  $R_2$ 

Figure 51 shows that  $R_2$  consists of five move types.

A move type was called linear if it had no parallel edges otherwise it was called cyclic.  $R_2$  consists of three linear move types and two cyclic ones. The linear move types describe moves on blocks with different size while the cyclic move types belong to moves on blocks with equal size.



**Figure 51:** Possible move types of  $R_2$

(B) Move types of  $R_3$

Move type set  $R_3$  contains 33 move types for permutation of three blocks.  $R_3$  contains all moves of  $R_2$ .

$R_3$  consists of:

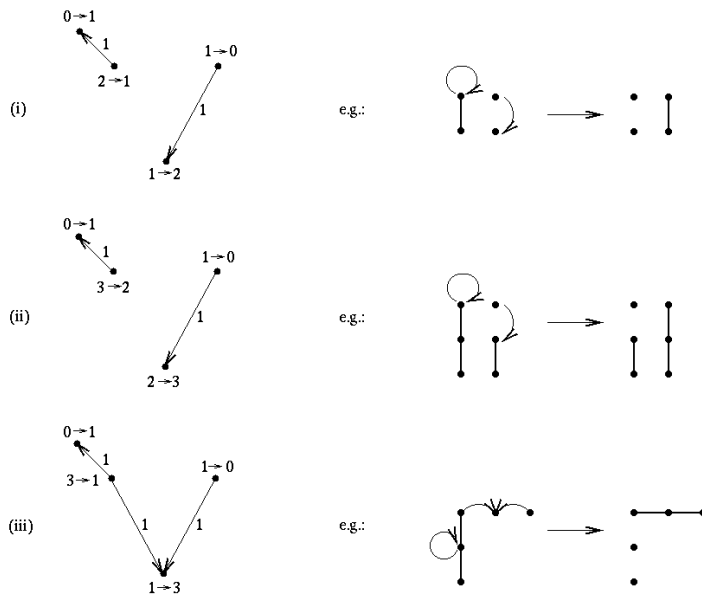
- (a) 5 move types of  $R_2$
- (b) 3 move types of moves with a reflexive move path
- (c) Move types of moves with no reflexive move path
  - (1) 11 linear move types
  - (2) 14 cyclic move types

(a) Move types of  $R_2$

The five move types of  $R_2$  belong to  $R_3$ .

(b) Move types of moves with a reflexive move path

Figure 52 shows the three move types of  $M_3$  with one reflexive move path.



**Figure 52:** The three move types of moves with a reflexive move path in  $R_3$  having no equivalent move without a reflexive move path

(c1) Linear move types

The general linear move type has the structure:

$$a \xrightarrow{|a|-|a'|} b \xrightarrow{|c'|-|c|} c$$

With  $M \notin M2$  all three blocks  $a$ ,  $b$ ,  $c$  get permuted. This means that  $|a| \neq |a'|$  and  $|c| \neq |c'|$ . The size of  $b$  may be unchanged. Three classes of move types are possible.:

$$(i) \quad \begin{pmatrix} a \\ b \\ c \end{pmatrix} \longrightarrow \begin{pmatrix} b \\ c \\ a \end{pmatrix} \text{ with } a \xrightarrow{|a|-|b|} b \xrightarrow{|a|-|c|} c; \quad |a| \neq |b|, |a| \neq |c|$$

$$(ii) \quad \begin{pmatrix} a \\ b \\ c \end{pmatrix} \longrightarrow \begin{pmatrix} c \\ a \\ b \end{pmatrix} \text{ with } a \xrightarrow{|a|-|c|} b \xrightarrow{|b|-|c|} c; \quad |a| \neq |c|, |b| \neq |c|$$

$$(iii) \quad \begin{pmatrix} a \\ b \\ c \end{pmatrix} \longrightarrow \begin{pmatrix} c \\ b \\ a \end{pmatrix} \text{ with } a \xrightarrow{|a|-|c|} b \xrightarrow{|a|-|c|} c; \quad |a| \neq |c|$$

Application of the operation of class (i) on the inverted sequence  $\star c, \star b, \star a$  equals an application of the operation of class (ii) on the sequence  $a, b, c$ .

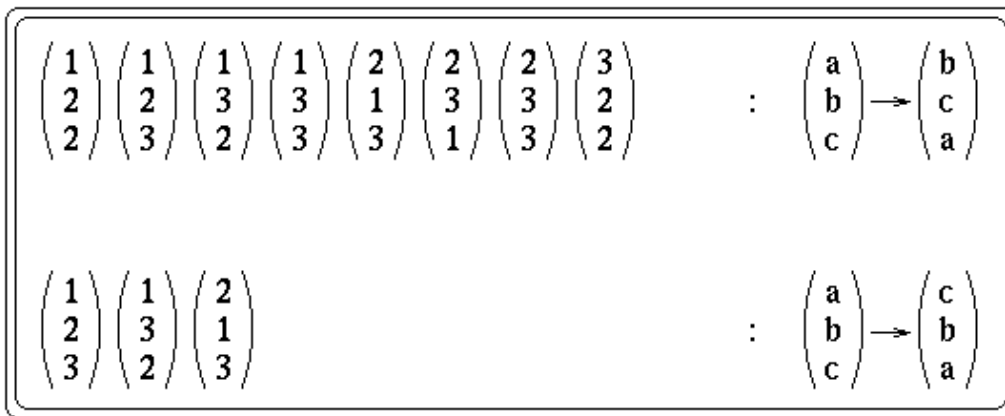
$$\begin{pmatrix} a \\ b \\ c \end{pmatrix} \longrightarrow \begin{pmatrix} b \\ c \\ a \end{pmatrix} \sim \begin{pmatrix} \star c \\ \star b \\ \star a \end{pmatrix} \longrightarrow \begin{pmatrix} \star b \\ \star a \\ \star c \end{pmatrix} = \begin{pmatrix} a \\ b \\ c \end{pmatrix} \longrightarrow \begin{pmatrix} c \\ a \\ b \end{pmatrix}$$

Classes (i) and (ii) are therefore identical. Application of the operation of class (iii) on  $\star c, \star b, \star a$  equals an application of the operation of class (iii) on the sequence  $a, b, c$ .

$$\begin{pmatrix} a \\ b \\ c \end{pmatrix} \longrightarrow \begin{pmatrix} c \\ b \\ a \end{pmatrix} \sim \begin{pmatrix} \star c \\ \star b \\ \star a \end{pmatrix} \longrightarrow \begin{pmatrix} \star a \\ \star b \\ \star c \end{pmatrix} = \begin{pmatrix} a \\ b \\ c \end{pmatrix} \longrightarrow \begin{pmatrix} c \\ b \\ a \end{pmatrix}$$

Class (iii) contains therefore pairwise identical move types.

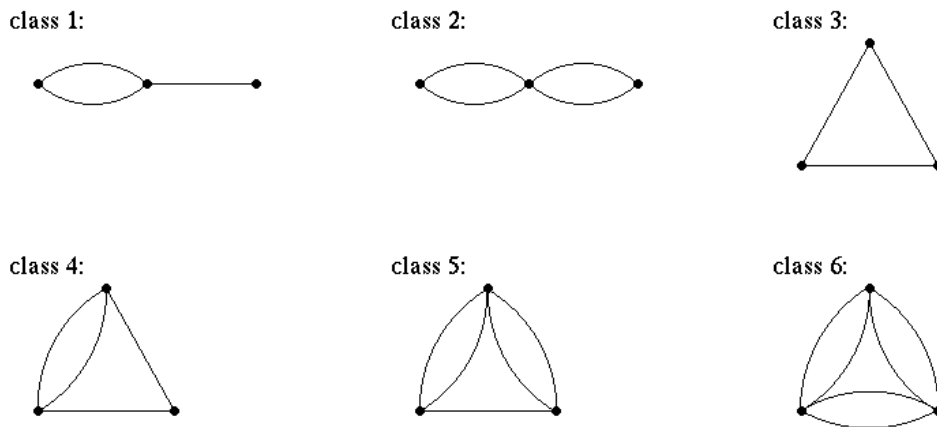
In (i) and (iii)  $|a|-|c|$  cells move to  $c$ . It holds therefore the restriction  $|b|+|c| > |a|$ . The resulting eleven linear (not equivalent) move types are given in figure 53.



**Figure 53:** The eleven linear move types of  $R3$

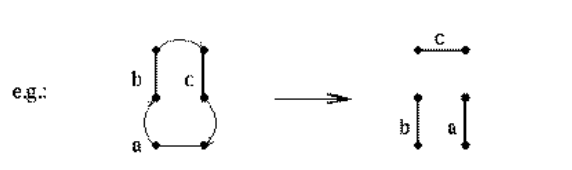
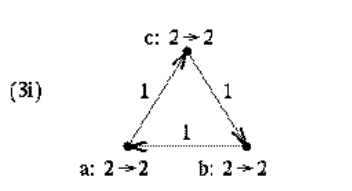
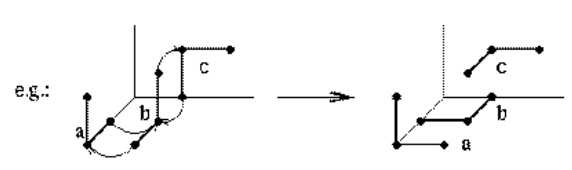
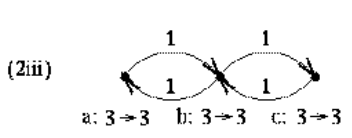
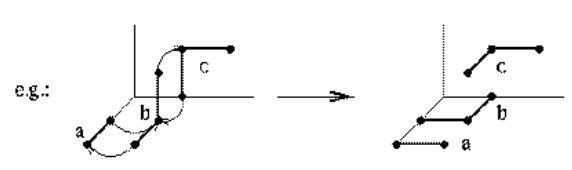
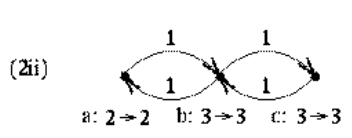
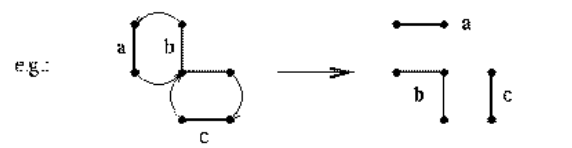
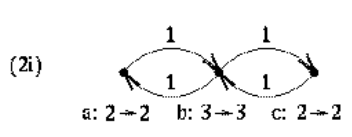
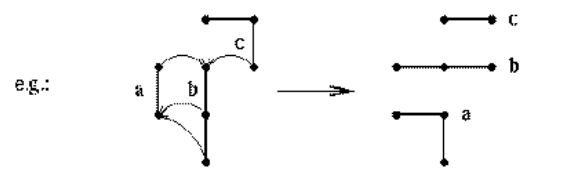
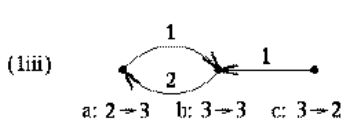
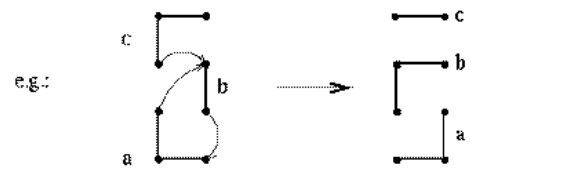
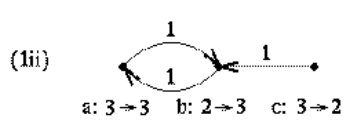
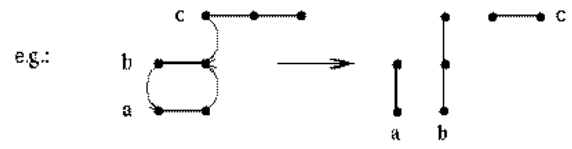
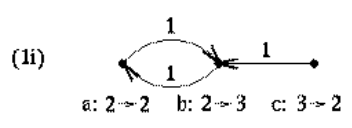
(c2) Cyclic move types

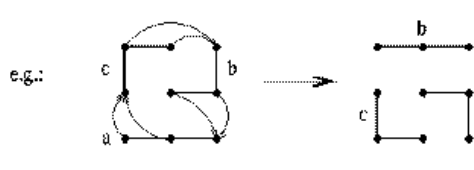
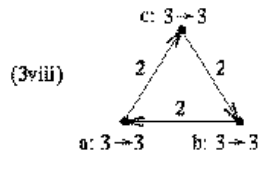
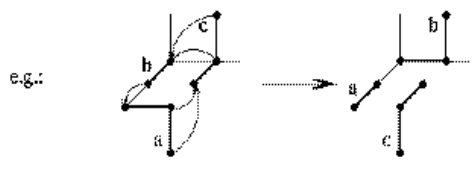
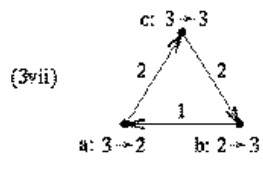
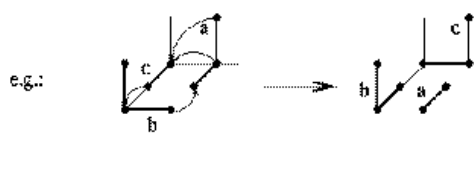
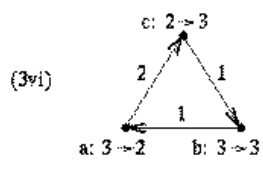
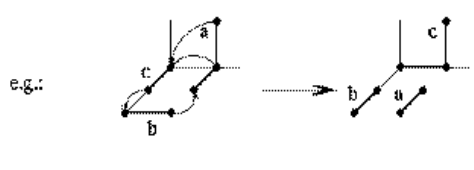
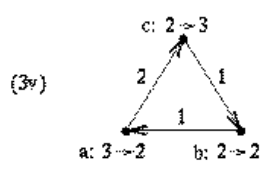
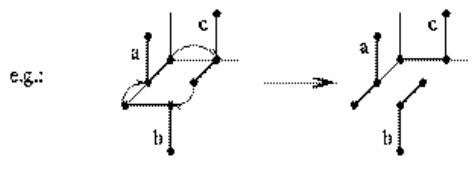
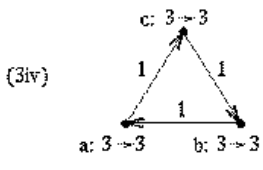
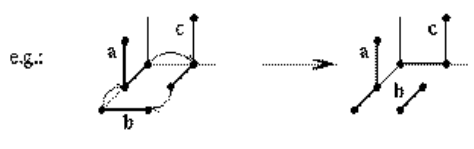
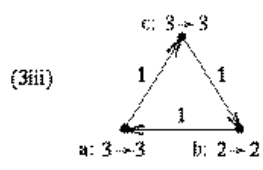
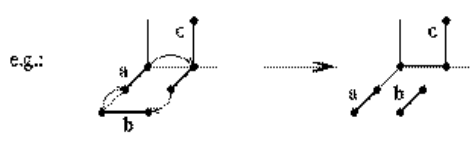
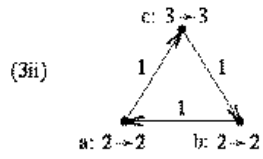
A nonlinear move type representing a move without reflexive migration path is called a cyclic move type. In this section a set of cyclic move types for  $R3$  is developed. A cyclic move type in  $R3$  has six structural possibilities (figure 54).



**Figure 54:** The six structural possibilities for a cyclic move type belonging to  $R3$  but not to  $R2$

Application of selection rules on move type candidates of these classes results in twentyone move types. These possible cyclic move types are given in figure 55 (next three pages).







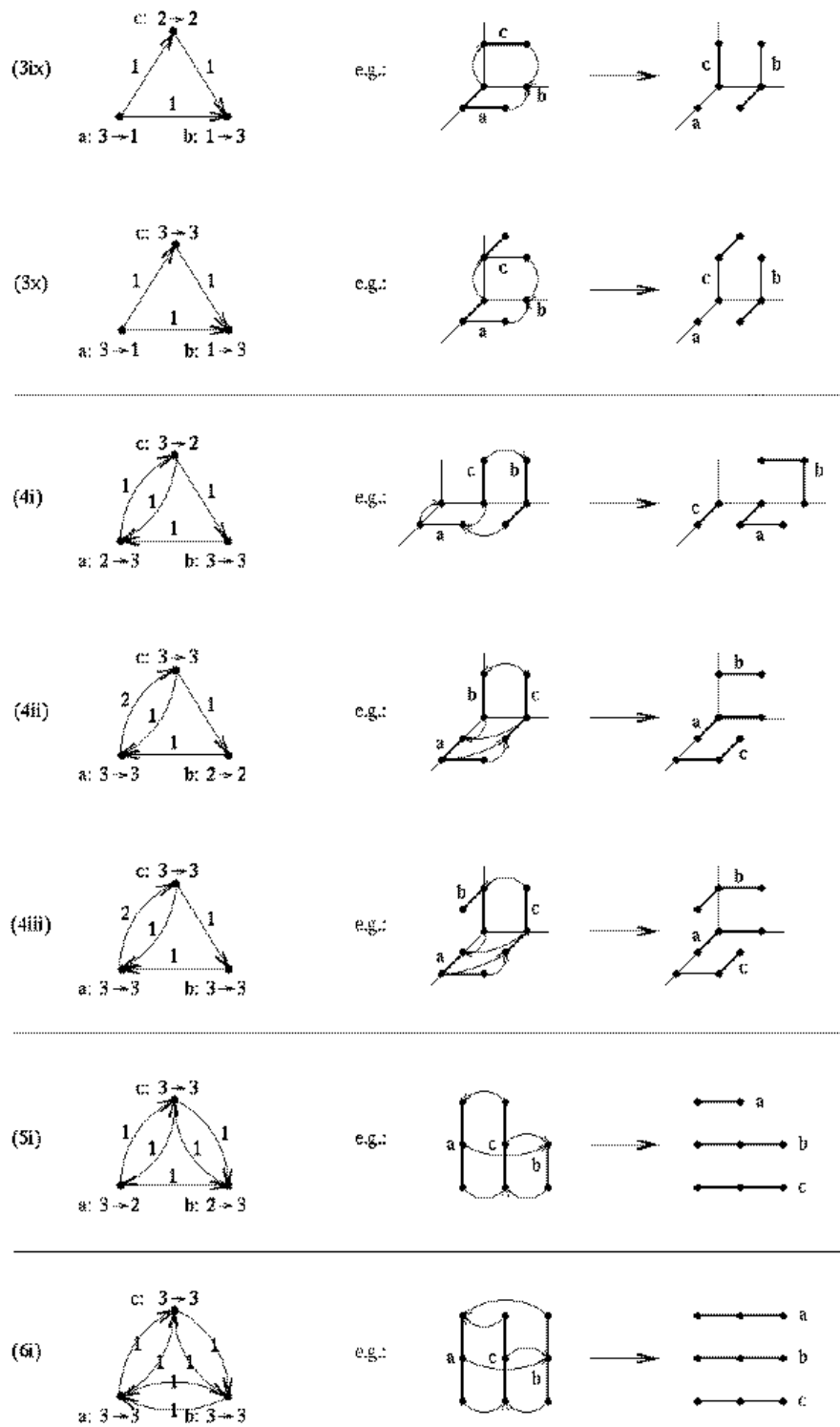


Figure 55: The twentyone possible cyclic move types of  $R_3$  (cyclic move types of  $R_2$  not included)

If a move belongs to a linear move type a migration of two cells from one to another block cannot be described by a move type with migration of only one cell between this pair of blocks. In a cyclic migration a move involving a migration of two cells from one to another block has always an equivalent move with migration of not more than one cell between each pair of blocks. These equivalent moves are represented by equivalent move types (describing the same permutations). The pairs of equivalent cyclic move types of  $R3$  are:  $((1iii), (1ii))$ ,  $((3v), (3ii))$ ,  $((3vi), (3iii))$ ,  $((3vii), (3iii))$ ,  $((3viii), (3iv))$ ,  $((4ii), (4i))$ ,  $((4iii), (2iii))$ .

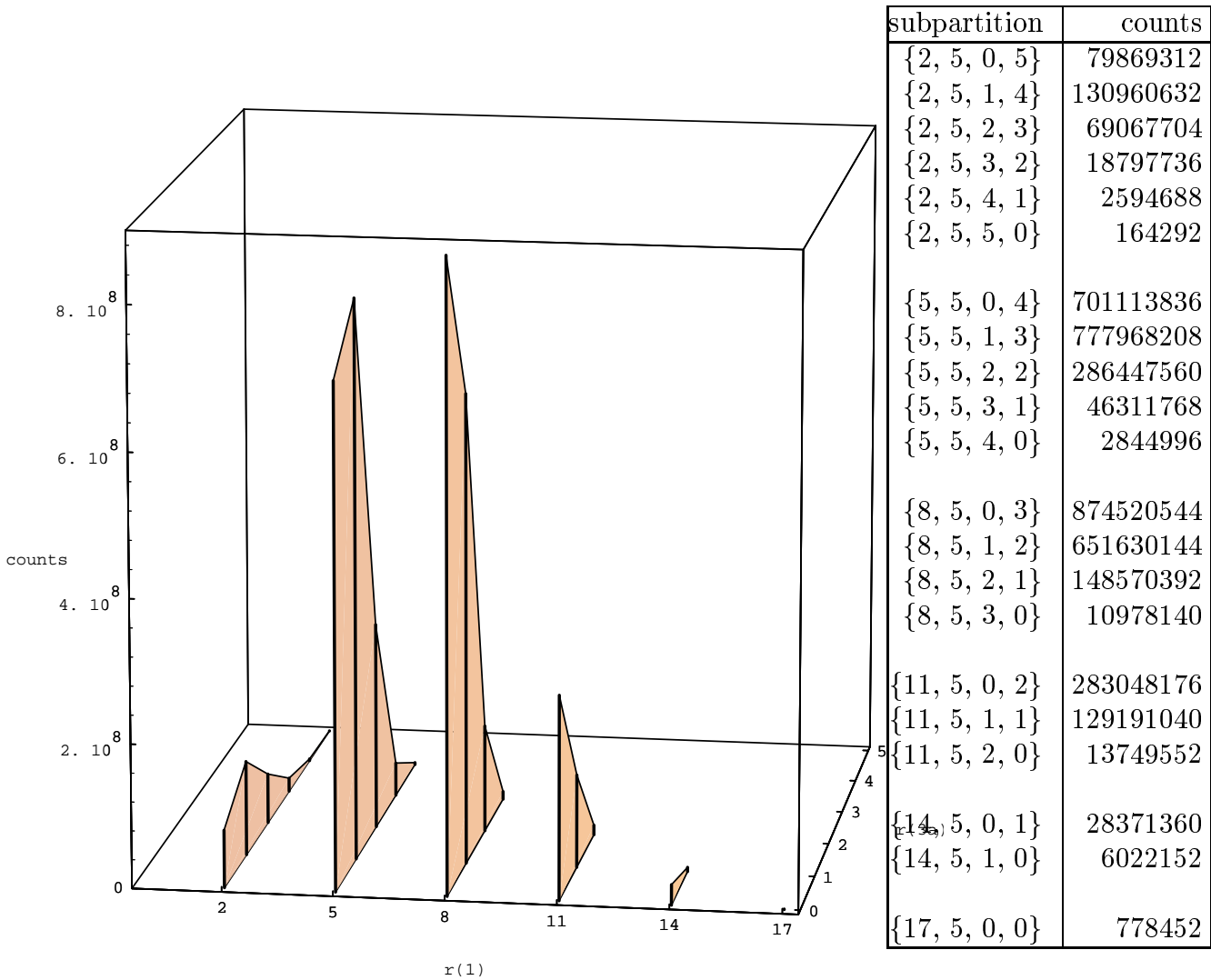
The 14 move types  $(1i)$ ,  $(1ii)$ ,  $(2i)$ ,  $(2ii)$ ,  $(2iii)$ ,  $(3i)$ ,  $(3ii)$ ,  $(3iii)$ ,  $(3iv)$ ,  $(3ix)$ ,  $(3x)$ ,  $(4i)$ ,  $(5i)$ , and  $(6i)$  are the not equivalent cyclic move types of  $R3$ .

## 6. Results

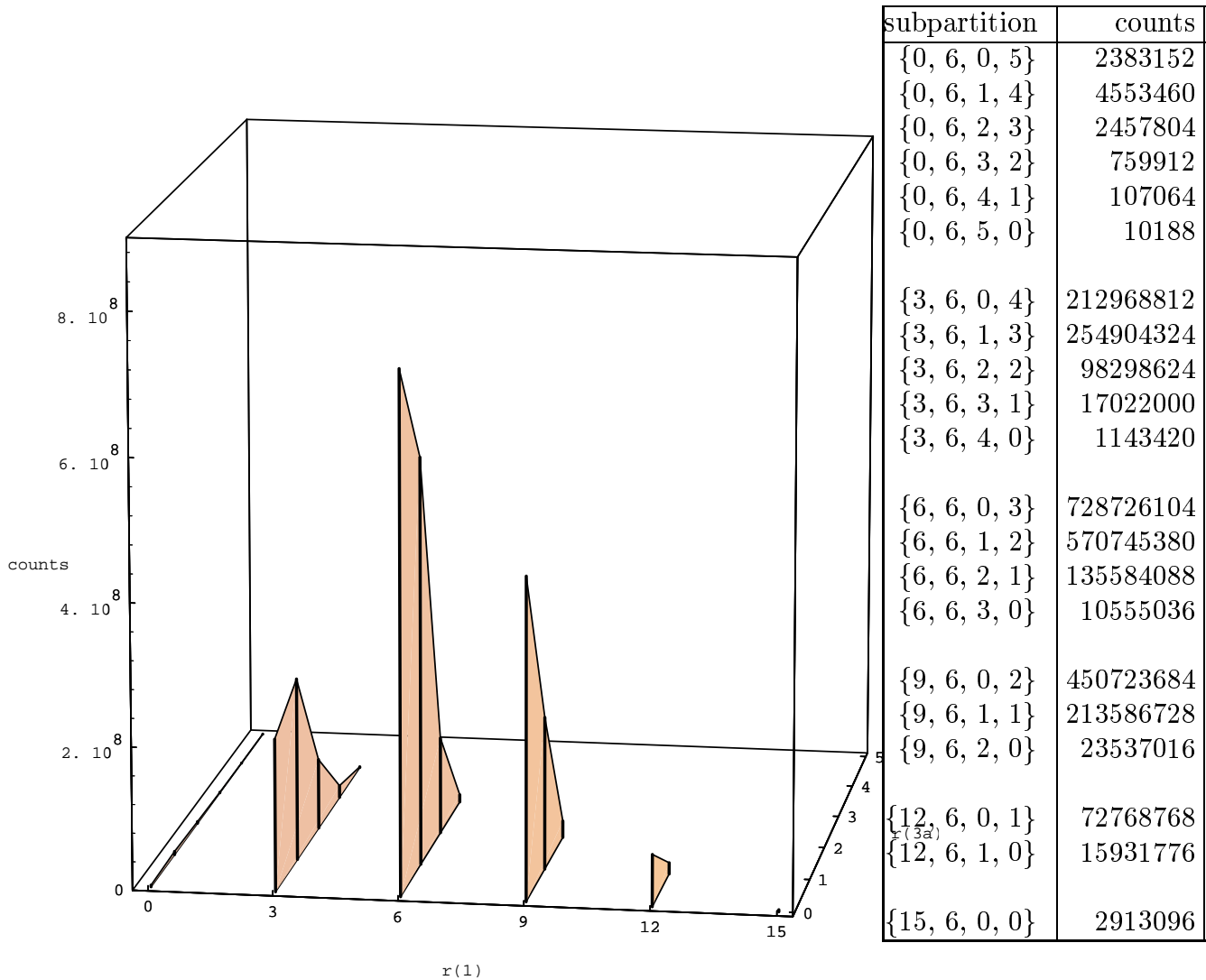
### 6.1. Exhaustive calculation of microconfigurations

#### 6.1.1. Counts

In a subpartition the number of blocks with one, two, three linear, and three nonlinear cells is given as  $\{r_1, r_2, r_{3a}, r_{rb}\}$ . For all subpartitions with  $r_2 = 5$  and  $r_2 = 6$  the number of MCs, or counts, is calculated using the EC-algorithm.



**Figure 56:** Counts in subpartitions with five two-celled blocks in a  $3 \times 3 \times 3$  cube.  $r(1)$  and  $r(3a)$  are the number of blocks with single and linear arranged cells



**Figure 57:** Counts in subpartitions with six two-celled blocks in a  $3 \times 3 \times 3$  cube.  $r(1)$  and  $r(3a)$  are the number of blocks with single and linear arranged cells

If the number of  $r_1$  and  $r_2$  blocks are constant the remaining cells have to belong to  $r_{3a}$  blocks, three linear arranged cells, or  $r_{3b}$  blocks, three nonlinear arranged cells. From figure and it can be seen that in general the variability of MCs is higher for subpartitions with higher amount of compact  $r_{3b}$  blocks. In subpartitions consisting only of  $r_{3b}$ , no  $r_{3a}$ , and a small number of other blocks ( $r_1$  and  $r_2$  blocks) a second effect can be seen. The exchange of one  $r_{3b}$  against a  $r_{3a}$  blocks increases the number of counts.

### 6.1.2. Symmetry

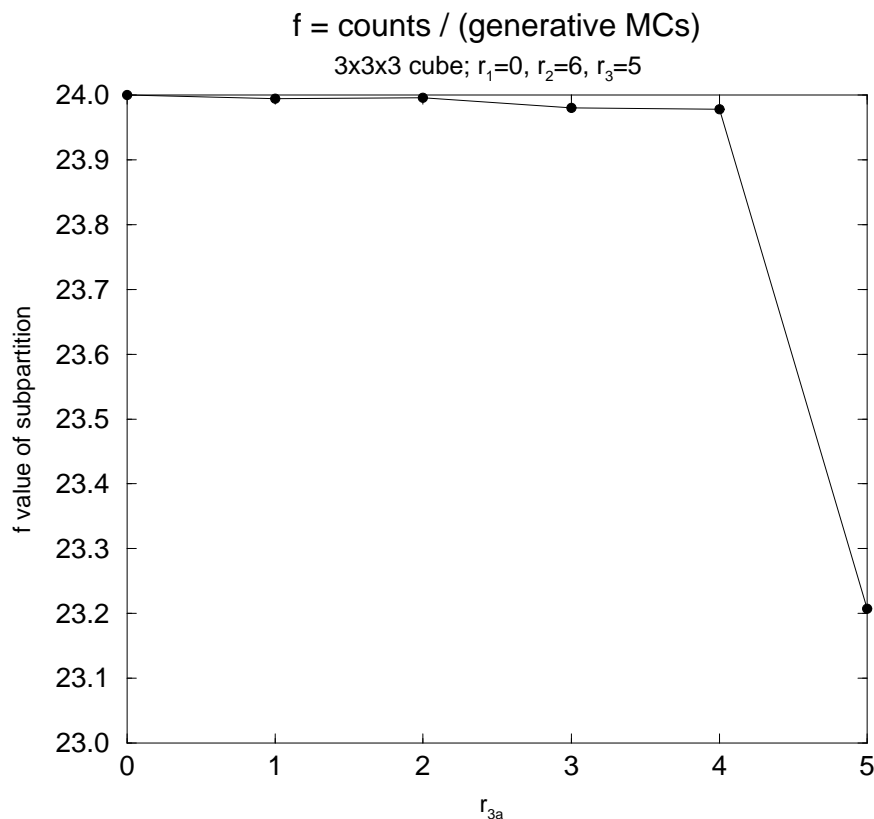
#### a) Rotation symmetry

MCs that are rotational equivalent belong to the same rotation class. One MC of a rotation class is generated first by EC-algorithm. This MC is defined as representative rotamere of the corresponding rotation class. Code 1 of a MC belongs to a representative rotamere if no rotated MC with a smaller code 1 exists. A representative rotamere generates 24 different MCs if it has no rotational symmetry.

The size of rotation class  $s_i$  given by  $|s_i|$  is the number of MCs belonging to this class. For a MC with rotational symmetry  $|s_i|$  will be smaller than 24. The ratio  $f$  of counts  $c$ , the total number of MCs calculated by EC-algorithm, and the number of representative MCs  $c_{rep}$  equals the mean rotation class size  $|s|$ .

$$f = \frac{c}{c_{rep}} = \frac{\sum_{i=1}^n |s_i|}{c_{rep}} = \langle |s| \rangle$$

In figure 58 the  $f$  values for the different subpartitions of partition  $\{0, 6, 5\}$  are plotted.



**Figure 58:** Mean rotation class size in partition  $\{0,6,5\}$ .  $r_{3a}$  is the number of blocks with three linear arranged cells.

At constant  $r_1, r_2$ , and  $r_3$  only the number of  $r_{3a}$  and  $r_{3b}$  blocks can be varied. If there are no MCs with rotational symmetry in a given subpartition  $f$  is 24. It will be lower than 24 if rotational symmetric MCs are present in the subpartition. A steady decrease of  $f$  with increasing  $r_{3a}$  blocks can be seen.

Table 7 gives all  $f$  values for  $r_2 = 5$  in a  $3 \times 3 \times 3$  cube. It can be seen that it is important for rotational symmetry if  $r_{3a}$  has an odd or even value. Subpartitions with odd  $r_{3b}$  have  $f = 24$  and therefore no rotational symmetry.

subpartition	counts	representative rotameres	f
{2, 5, 0, 5}	79869312	3327888	24,0000
{2, 5, 1, 4}	130960632	5456830	23,9994
{2, 5, 2, 3}	69067704	2877821	24,0000
{2, 5, 3, 2}	18797736	783353	23,9965
{2, 5, 4, 1}	2594688	108112	24,0000
{2, 5, 5, 0}	164292	6862	23,9423
{5, 5, 0, 4}	701113836	29213527	23,9996
{5, 5, 1, 3}	777968208	32415342	24,0000
{5, 5, 2, 2}	286447560	11935574	23,9995
{5, 5, 3, 1}	46311768	1929657	24,0000
{5, 5, 4, 0}	2844996	118588	23,9906
{8, 5, 0, 3}	874520544	36438356	24,0000
{8, 5, 1, 2}	651630144	27151600	23,9997
{8, 5, 2, 1}	148570392	6190433	24,0000
{8, 5, 3, 0}	10978140	457526	23,9946
{11, 5, 0, 2}	283048176	11794065	23,9992
{11, 5, 1, 1}	129191040	5382960	24,0000
{11, 5, 2, 0}	13749552	573009	23,9954
{14, 5, 0, 1}	28371360	1182140	24,0000
{14, 5, 1, 0}	6022152	250972	23,9953
{17, 5, 0, 0}	778452	32476	23,9701

table 7:  $3 \times 3 \times 3$  cube,  $r_2 = 5$

Subpartitions with  $r_2 = 6$  show a similar behavior.

subpartition	counts	representive rotameres	f
{0, 6, 0, 5}	2383152	99298	24,0000
{0, 6, 1, 4}	4553460	189769	23,9948
{0, 6, 2, 3}	2457804	102427	23,9957
{0, 6, 3, 2}	759912	31689	23,9803
{0, 6, 4, 1}	107064	4465	23,9785
{0, 6, 5, 0}	10188	439	23,2073
{3, 6, 0, 4}	212968812	8873840	23,9996
{3, 6, 1, 3}	254904324	10621273	23,9994
{3, 6, 2, 2}	98298624	4096091	23,9982
{3, 6, 3, 1}	17022000	709330	23,9973
{3, 6, 4, 0}	1143420	47730	23,9560
{6, 6, 0, 3}	728726104	30364121	23,9996
{6, 6, 1, 2}	570745380	23782124	23,9989
{6, 6, 2, 1}	135584088	5649563	23,9990
{6, 6, 3, 0}	10555036	440161	23,9799
{9, 6, 0, 2}	450723684	18780906	23,9990
{9, 6, 1, 1}	213586728	8899698	23,9993
{9, 6, 2, 0}	23537016	981140	23,9895
{12, 6, 0, 1}	72768768	3032262	23,9982
{12, 6, 1, 0}	15931776	664230	23,9853
{15, 6, 0, 0}	2913096	121557	23,9649

table 8:  $3 \times 3 \times 3$  cube,  $r_2 = 6$

#### b) Rotation and reflection symmetry

In chapter it is proofed, that MCs of the same rotation class  $R_1$  may belong after reflection again to  $R_1$  or may be part of a, from  $R_1$  different, rotation class  $R_2$ . In the latter case by the permission of rotation and reflection a new equivalence class  $R = R_1 \cup R_2$  is generated. This equivalence class has a representative MC, the reflective representative rotamere. Those reflective representative rotameres generate the set of rotation and reflection reduced counts.

The effect of pairwise equivalent rotation classes under the permission of reflection can be seen in partition  $\{0, 6, 5\}$  (table 9, figure 59). The number of representative MCs is approximately halved by permission of reflection.

$r_{3a}$	$r_{3b}$	count	representive rotameres	reflective representive rotameres
0	5	2383152	99298	49649
1	4	4553460	189769	94920
2	3	2457804	102427	51236
3	2	759912	31689	15880
4	1	107064	4465	2242
5	0	10188	439	238

table 9

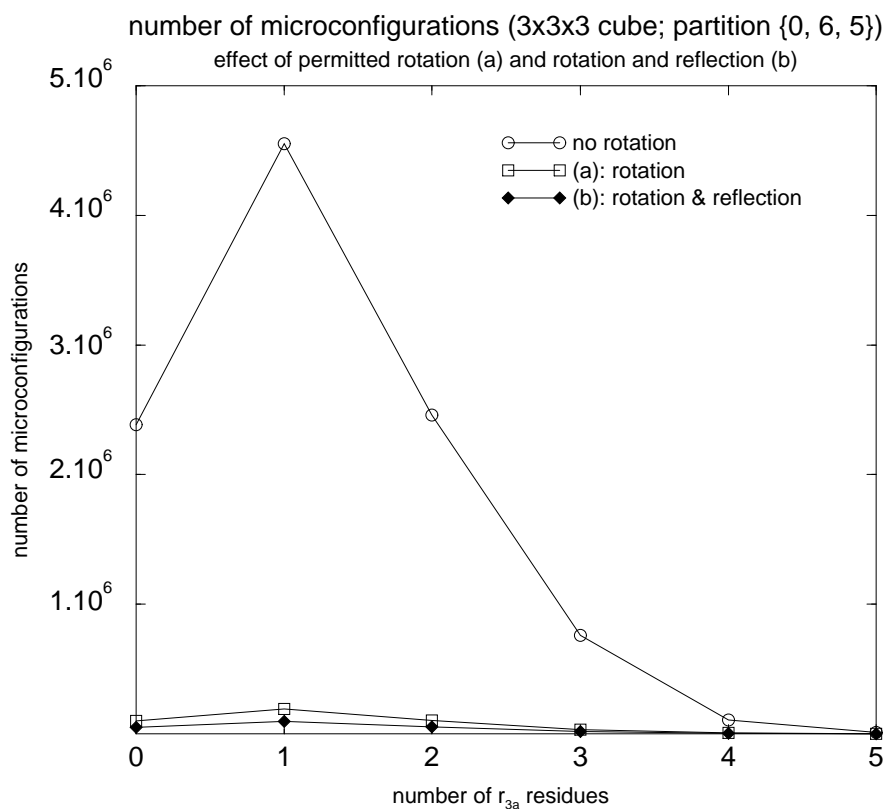


Figure 59: grafic to table 9

Table 10 shows that the average size of equivalence classes, the number of reflective representative rotameres, for subpartitions with  $r_2 = 6$  is approximately 48.



subpartition	counts	representive rotameres		$f_{\text{rot, ref}}$
		rotation	rotation and reflection	
{0, 6, 0, 5}	2383152	99298	49649	48,0000
{0, 6, 1, 4}	4553460	189769	94920	47,9716
{0, 6, 2, 3}	2457804	102427	51236	47,9703
{0, 6, 3, 2}	759912	31689	15880	47,8534
{0, 6, 4, 1}	107064	4465	2242	47,7538
{0, 6, 5, 0}	10188	439	238	42,8067
{3, 6, 0, 4}	212968812	8873840	4437180	47,9964
{3, 6, 1, 3}	254904324	10621273	5311070	47,9949
{3, 6, 2, 2}	98298624	4096091	2048420	47,9875
{3, 6, 3, 1}	17022000	709330	354889	47,9643
{3, 6, 4, 0}	1143420	47730	24059	47,5257
{6, 6, 0, 3}	728726104	30364121	15183100	47,9959
{6, 6, 1, 2}	570745380	23782124	11892200	47,9933
{6, 6, 2, 1}	135584088	5649563	2825430	47,9871
{6, 6, 3, 0}	10555036	440161	220791	47,8056
{9, 6, 0, 2}	450723684	18780906	9391570	47,9924
{9, 6, 1, 1}	213586728	8899698	4450540	47,9912
{9, 6, 2, 0}	23537016	981140	491440	47,8940
{12, 6, 0, 1}	72768768	3032262	1516730	47,9774
{12, 6, 1, 0}	15931776	664230	332757	47,8781
{15, 6, 0, 0}	2913096	121557	61008	47,7494

table 10

### 6.1.3. Distance

For two MCs  $x$  and  $y$  it is shown in section that  $d(x, y)$  is a metric on the MCS. To investigate the problem of shortest distance between two MCs if rotation of these MCs is possible, the class distance  $d_c(x, y)$  between MC  $x$  belonging to rotation class  $X$  and MC  $y$  belonging to rotation class  $Y$  is used. It is defined as

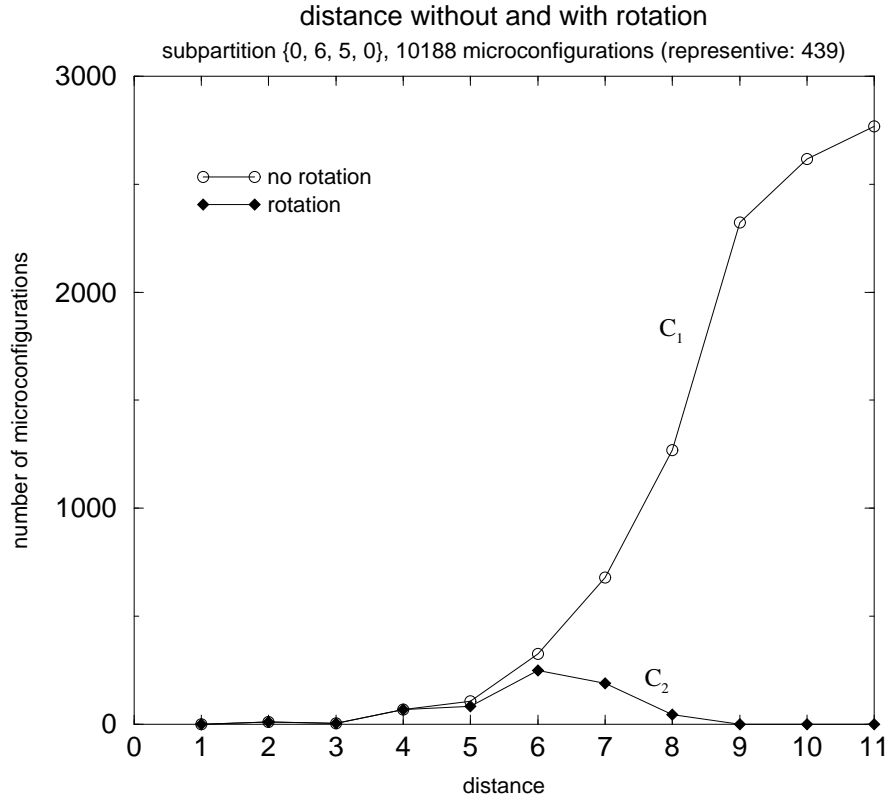
$$d_c(x, y) = \inf\{d(x, y) : x \in X, y \in Y\}.$$

This means that  $y$  is rotated until its distance to  $x$  is at a minimum value  $d_c(x, y)$ . It is possible that by rotation of  $y$  more than one MC is generated having the same minimum value  $d_c(x, y)$ . MCs  $x$  and  $y$  are called neighboured if

$$d_c(x, y) = d(x, y)$$

(a) Comparison of  $d(x, y)$  and  $d_c(x, y)$

Distances  $d(x, y)$  and class distances  $d_c(x, y)$  of subpartition  $\{0, 6, 5, 0\}$  to a fixed MC  $x$ , that was chosen to be the first EC-algorithm solution of the subpartition, are compared in figure 60.



**Figure 60:** Distance and class distance of MCs in  $\{0,6,5,0\}$  to the first EC-algorithm solution of  $\{0,6,5,0\}$ .

Curve  $C_1$  shows number of MCs at distance  $d(x, y)$ . Obviously  $d(x, y) \neq 1$  because two MCs cannot differ in only one block. Most MCs are completely different from MC  $x$ . This results in a sigmoid shape of  $C_1$  with a maximum at  $d(x, y) = 11$ . For the general case  $d(x, y) = b$  when  $b$  is the total number of blocks.

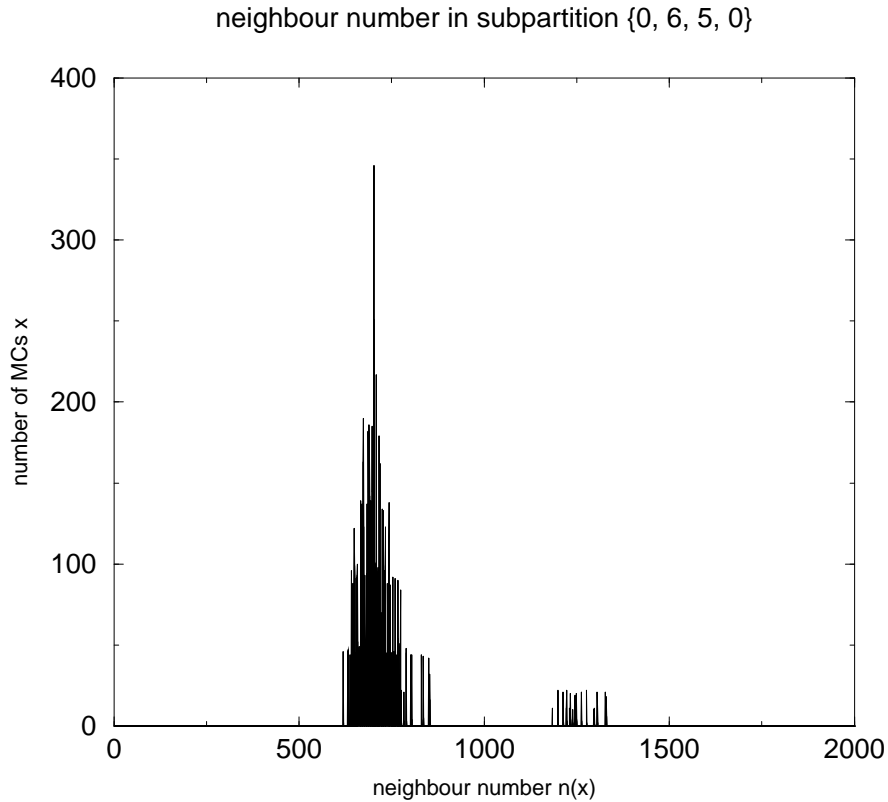
The number of neighbored MCs, those MCs with  $d(x, y) = d_c(x, y)$ , give curve  $C_2$ .  $C_1$  equals  $C_2$  for low values of  $d(x, y)$  because at low distance every  $y$  is neighbored to  $x$ .  $C_2$  has a maximum at a medium distance  $d(x, y)$ .

## (b) Neighbourhood MCs and rotation symmetry

The last section analysed the number of neighbored MCs having a certain distance  $d(x, y)$  from a fixed MC  $x$ . The sum of all neighbored MCs  $y$  over all  $d(x, y)$  is called neighbour number  $n(x)$ . With  $b$  as number of total blocks it is defined as

$$n(x) = \sum_{i=2}^{i=b} \{y \in MCS : d(x, y) = d_c(x, y) = i\}.$$

In figure 61 the number of all  $x$  with the same neighbour number  $n(x)$  is calculated for subpartition  $\{0, 6, 5, 0\}$ .



**Figure 61:** Number of MCs having a neighbour number  $n(x)$  in  $\{0, 6, 5, 0\}$

Two clearly separated classes of neighbour numbers are observed. A test on the rotational symmetry properties of all MCs gives the following result:

Class 1 at  $n(x) \simeq 700$  consists of all MCs with no rotational symmetry whereas all MCs of class 2 at  $n(x) \simeq 1300$  have rotational symmetry.

## 6.2. Random walk with move sets M2 and M3

For two MCs  $x$  and  $y$  the distance  $d(x, y)$  is a metric (chapter , p ). A step of a random walk changes a MC  $x$  to a different MC  $y \neq x$ . Steps with  $d(x, y) \leq 2$  and  $d(x, y) \leq 3$  were used.

### 6.2.1. Move set M2

#### 6.2.1.1. Diffusion

A random walk is a diffusion process from a given startpoint. To show that the used random walks are diffusion processes, the distance  $d(x, y)$  to a given start MC has to increase in a linear manner with  $\sqrt{\text{steps}}$ . The average distance from a start MC after a large number of steps has also to be in accordance with other results.

(a) Expectation for subpartition  $\{0, 6, 5, 0\}$

As the number of steps  $s$  is increased the average distance  $d_{av}$  of MCs is a measure for diffusion and is given by:

$$d_{av} = \frac{\sum_{i=0}^{i=s} d_i}{s}$$

The number of MCs having distance  $d$  from the start MC are the counts  $c_d$  at  $d$ . The largest possible distance between two MCs is the total number of blocks. The mean distance of all MCs from the subpartition to the start MC is:

$$\langle d \rangle = \frac{\sum_{d=0}^{d=b} c_d \times d}{\sum_{d=0}^{d=b} c_d} = \frac{\sum_{d=0}^{d=b} c_d \times d}{c}$$

In  $C_1$  of figure 12 in chapter 3.4 the  $c_d$  values were calculated in the same subpartition with the same reference MC. If one calculates the mean for this system the result is:

$$\langle d \rangle = 9.4$$

As the step number in a random walk experiment is increased the average distance  $d_{av}$  should converge against the mean distance  $\langle d \rangle$ .

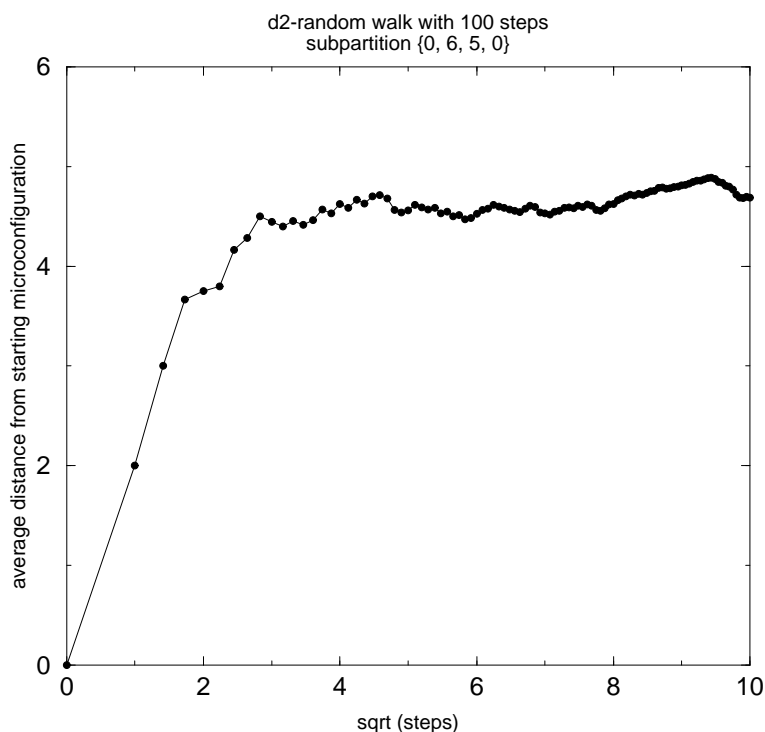
$$\lim_{s \rightarrow \infty} d_{av} = \langle d \rangle = 9.4$$

## (b) Result and interpretation

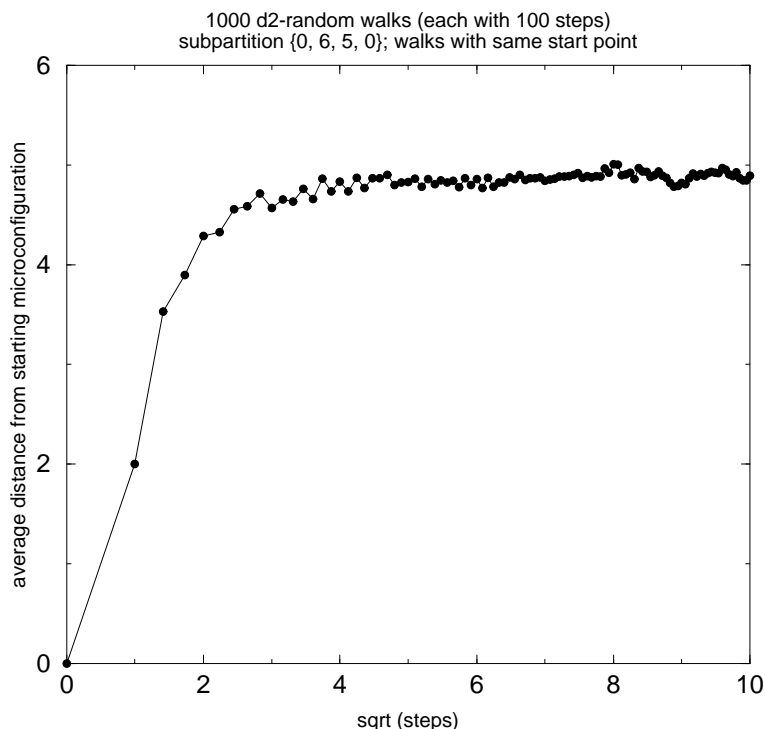
The two diagrams of figure 16 show how  $d_{av}$  behaves as the number of steps  $s$  is increased. The final step number was 100.

- 1) The first diagram shows that  $d_{av}$  of a single random walk converges against a much lower value than  $\langle d \rangle = 9.4$ .
- 2) In the second diagram the mean of  $d_{av}$  values from 1000 random walks at given step number was calculated. To check if  $d_{av}$  is at the beginning of the random walk proportional to  $\sqrt{s}$  the  $\sqrt{s}$  versus  $d_{av}$  is shown. In a cube of side length 3 after 4 or 5 steps a convergence behaviour of  $d_{av}$  can be seen. The cube is therefore too small to decide this question. Convergence against  $d_{av} = 4.8 \ll 9.4$  can be seen.

This result makes clear that obviously only a certain part of MCs is accessible to this random walk with step length = 2. If all MCs would have been reached convergence against  $\langle d \rangle = 9.4$  should have been occurred. This interpretation that not all MCs of MCS can be reached by a random walk with step length = 2 will be proofed in the next chapter.



**Figure 62:** Diffusion moving on a single random walk with permutation of two blocks ( $d_2$ -walk).



**Figure 63:** Average diffusion of 1000  $d_2$  random walks.

### 6.2.1.2. Accessible microconfigurations

In the diffusion experiment of chapter 4.2 the gap between expectation and result was explained by the hypothesis that not all MCs could be reached with steps of  $d=2$  (permutation of two blocks) and the first EC-algorithm solution as start MC in subpartition  $\{0, 6, 5, 0\}$ . The diffusion experiment had the following properties:

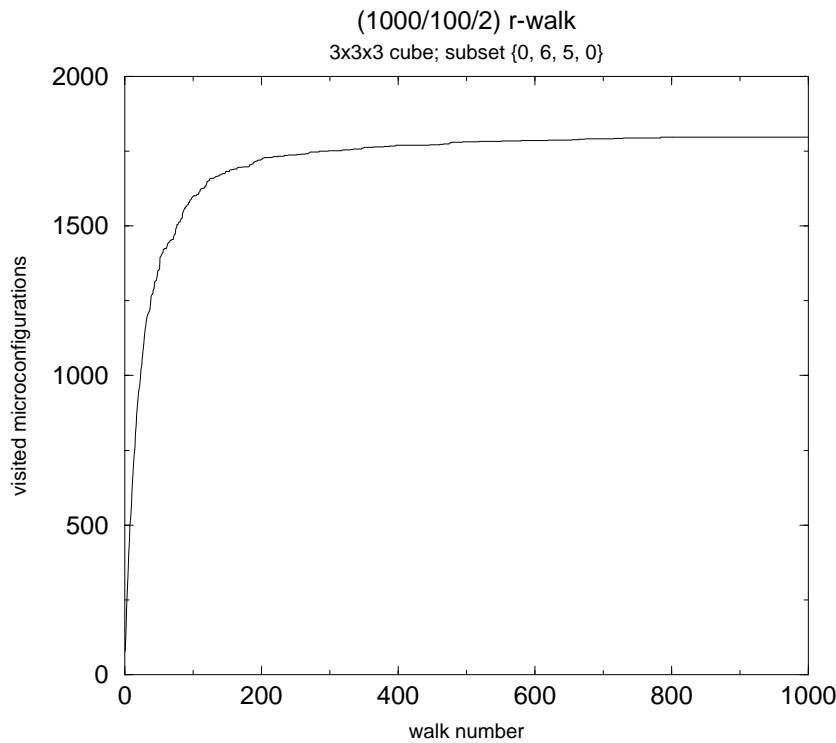
- The 1000 random walks with 100 steps/walk were calculated independent from each other but had the same start MC.
- The distance  $d$  between two legal steps was 2. The largest distance  $d_{max}$  between two MCs is their number of blocks  $b$  ( $=11$ ). 100 steps/walk are therefore enough to reach every MC.

The data of this calculation were  $1000 \times 100$  MCs given in code 2. In a new context these data were used now differently. The MCs of the first walk were given to an empty pool and the number of different MCs  $c_r$  (reached counts) in that pool was determined. Now the MCs of the second walk were added to the pool and  $c_r$  of that larger pool was determined.

The general algorithm is:

- 1) Take an empty pool.
- 2) The next of the 1000 calculated random walks is investigated. Add the new MCs of this walk to the MC pool.
- 3) Determine  $c_r$  of the resulting pool and go to step 2.

The result of this algorithm is shown in figure 64.



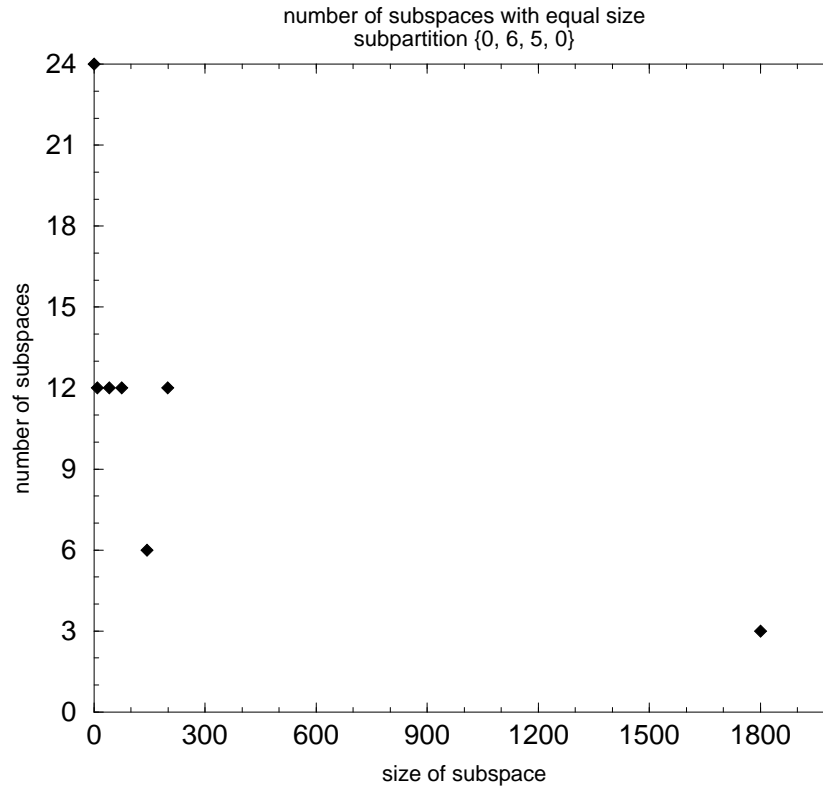
**Figure 64:** Only 1800 MCs are visited by 1000 random walks with 100 steps if two blocks are permuted in each step.

It can be seen that  $c_r$  converges against a value (1800) that is much lower than  $c = 10188$  (the total number of MCs). This convergence shows that new walks will visit no MCs that has not been occurred in former walks.

A random walk in subpartition  $\{0, 6, 5, 0\}$  with steps of  $d=2$  is not capable to reach all possible MCs. It is therefore not ergodic.

### 6.2.1.3. Subspaces

Subspace of a MCS are generated by the use of a certain move set like the permutation of two blocks. In figure 64 a subspace with 1800 MCs is described. There exist two other subspaces with 1800 MCs. The size and number of all subspaces in  $\{0, 6, 5, 0\}$  is given in the next figure.



**Figure 65:** Size and number of subspaces

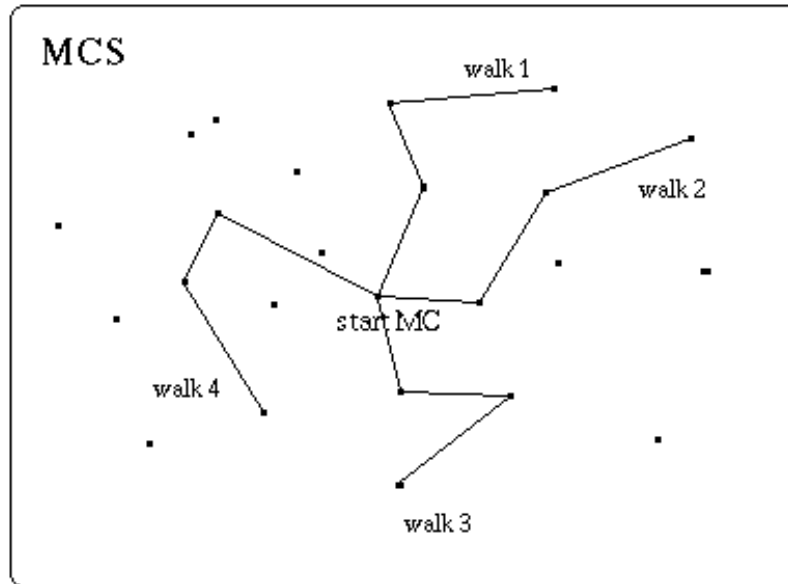
The next section will give the results of permutation of three blocks.



### 6.2.2. Move set M3

#### 6.2.2.1. Accessible microconfigurations

The algorithm used in chapter 4.3.1.1 with steps of  $d=2$  was now used for steps of  $d \leq 3$ . 10000 steps/walk were done. The start MC was the first EC-algorithm solution.



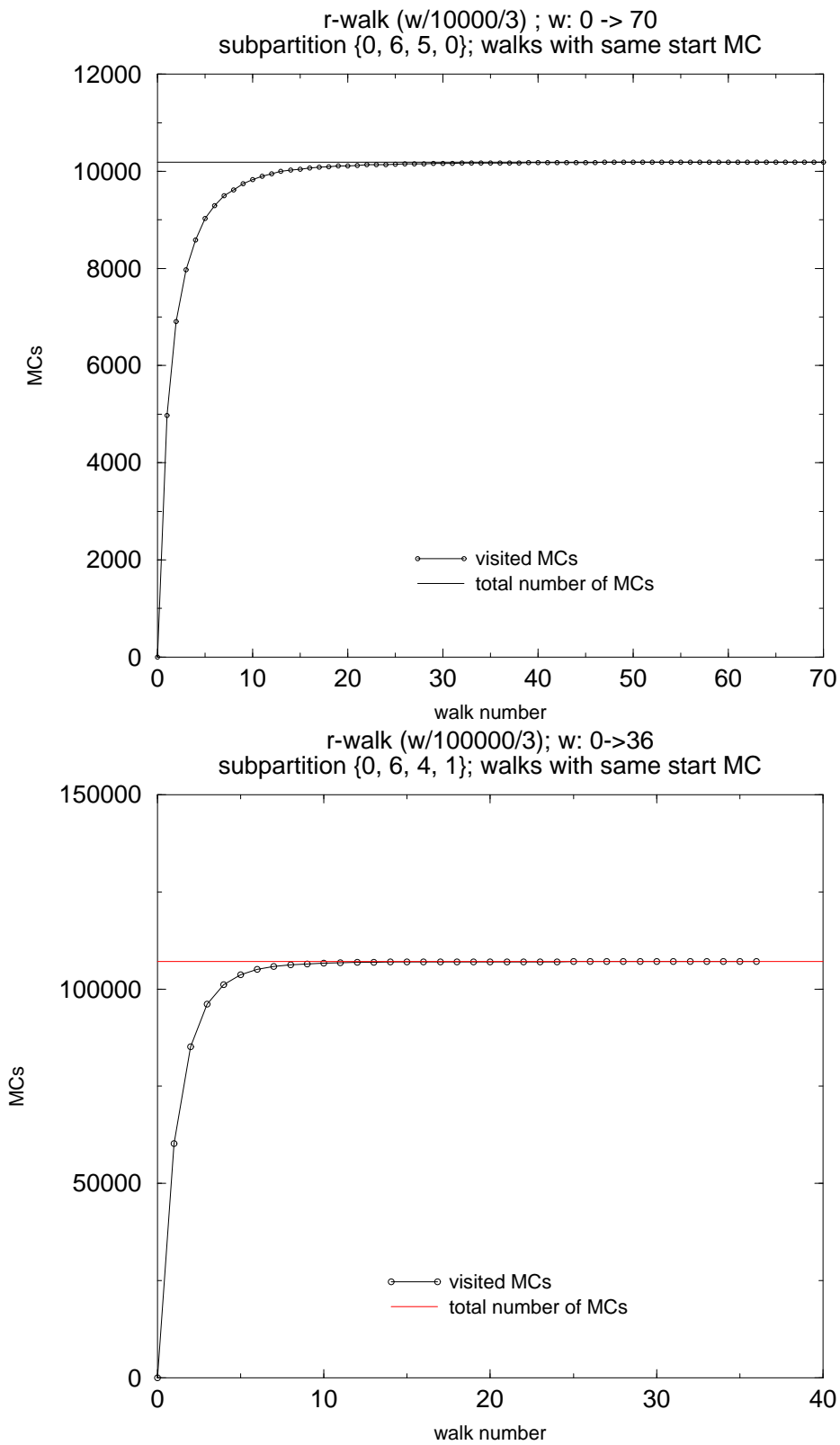
**Figure 66:** Ergodic random walk in microconfiguration space (MCS)

Figure 67: 70 random walks in subpartition  $\{0, 6, 5, 0\}$  were done. The pool was successively enlarged and the number of different MCs  $c_r$  (reached counts) were determined (see 4.3.1.1). The reached counts converge against the total counts of 10188 for this subpartition. The random walk with  $d \leq 3$  in subpartition  $\{0, 6, 5, 0\}$  is ergodic.

36 random walks in subpartition  $\{0, 6, 4, 1\}$  were done. The reached counts converge against the total counts of 107064 for this subpartition. The random walk with  $d \leq 3$  in subpartition  $\{0, 6, 4, 1\}$  is therefore ergodic.

Random walks in subpartition  $\{0, 6, 3, 2\}$  and other subpartitions are also ergodic (not shown).

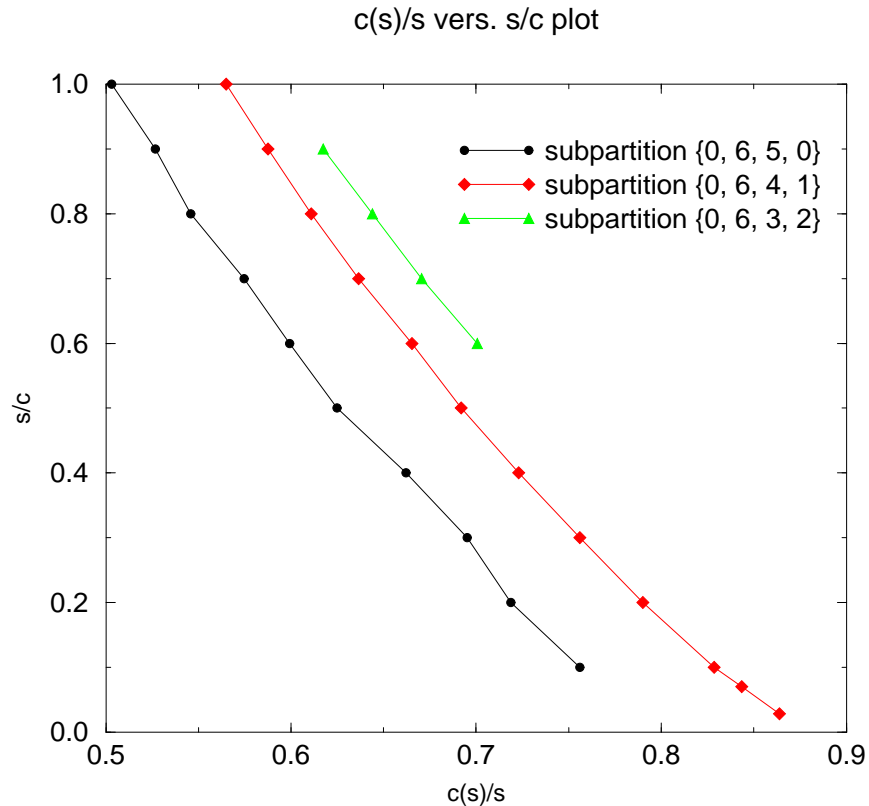
Until there are no other data the hypothesis was established that a random walk with  $d \leq 3$  is ergodic for any subpartition and any cube side length  $n$ .



**Figure 67:** Permutation of three blocks results in an ergodic random walk.

### 6.2.2.2. Random walks in similar subpartitions

Random walks in subpartitions  $\{0, 6, 5, 0\}$ ,  $\{0, 6, 4, 1\}$ ,  $\{0, 6, 3, 2\}$  with  $d \leq 3$  are ergodic. An experiment with no available interpretation till now is shown in figure 68. A single random walk was done in the three mentioned subpartitions. With steps  $s$ , total counts  $c$ , reached counts  $c_r$  the  $c_r/s$  versus  $s/c$  plot had a linear course as  $s$  came closer to  $c$  in all three cases. Moreover the slope was equal.



**Figure 68:** Random walks in similar subpartitions have similar walk properties

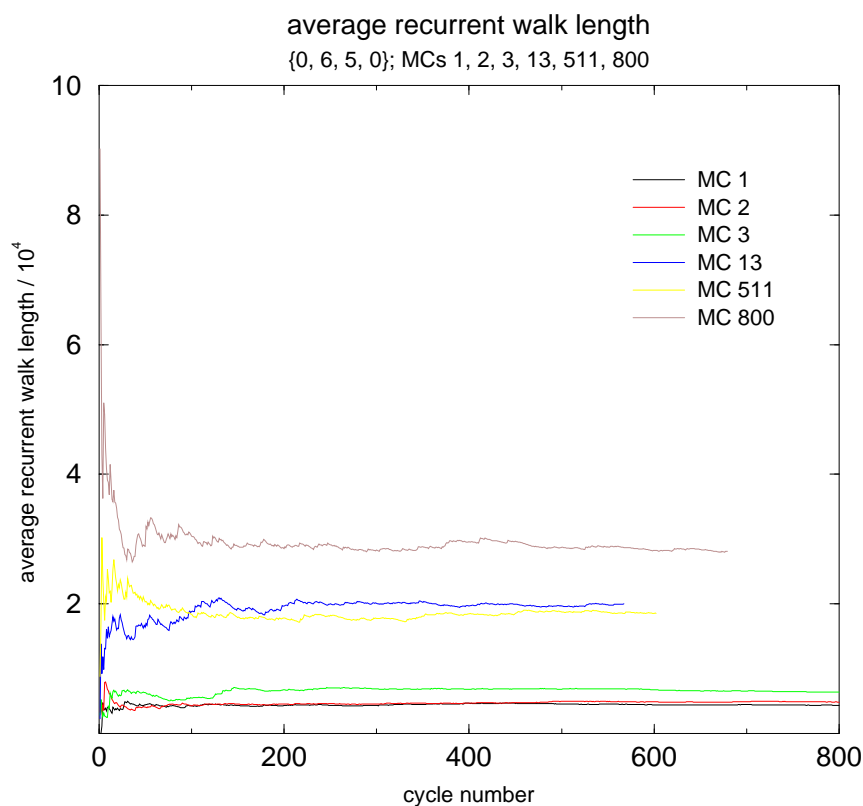
### 6.2.2.3. Probability $p$ of MCs in a random walk

In a random walk a step leads from MC  $x$  to  $y$ . Steps with  $d \leq 3$  in  $3 \times 3 \times 3$  cubes are ergodic. Accessibility of MCs is another important point. It will be shown how the structure of a MC determines its accessibility.

To get a measure for the accessibility of a MC an ergodic random walk is done. The number of steps between two successive visits of a MC gives one possible recurrent walk length. As the walk continues the mean of this recurrent walk length can be determined. It is clear that the ratio of the number of steps  $s$  and the hit number  $h$  ( $=$  number of visits) of MCs converges against the average recurrent walk length as the number of steps increases. On the other hand the ratio of the hit number  $h$  and steps  $s$  is the probability  $p$  that a MC is visited by a random walk. The average recurrent walk length ( $=$  ARW) is given by:

$$\text{ARW} = \lim_{s \rightarrow +\infty} \frac{s}{h} = \frac{1}{p}$$

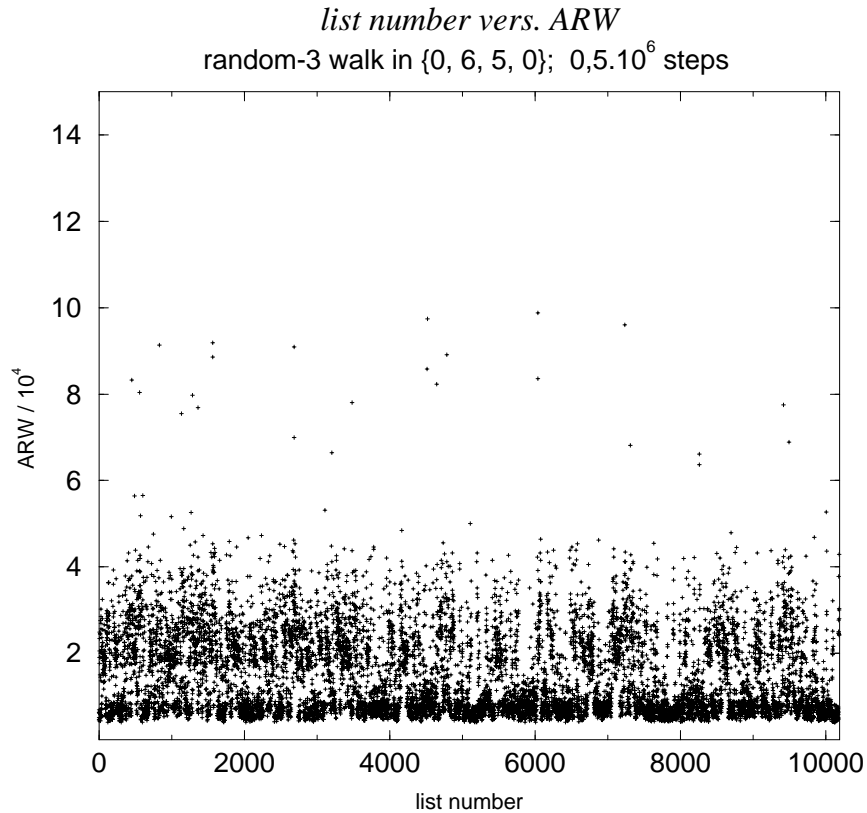
A first calculation on ARW values was done in subpartition  $\{0, 6, 5, 0\}$  for the MCs with list number 1, 2, 3, 13, 511, and 800. The result is shown in figure 69. The MCs 1, 2, 3 show that structural very similar MCs have very similar ARW values. The MCs 13, 511, 800 show that the ARW can exceed the sum of all MCs.



**Figure 69:** Different MCs can have different ARW

(a) ARW values and EC-algorithm solutions

The EC-algorithm generates a complete list of possible MCs in a given subpartition. The number of occurrence of a MC in this list, the list number, identifies the MC. The subpartition  $\{0, 6, 5, 0\}$  consists of 10188 MCs. The plot of list number versus ARW of MCs for this subpartition is given in figure 70.



**Figure 70:** ARW values of all MCs in  $\{0, 6, 5, 0\}$  in order of their calculation with the EC-algorithm.

The measure for the probability  $p_i$  of a MC with list number  $i$  in a random walk is  $ARW_i^{-1}$ . If two MCs with list numbers  $i$  and  $j$  are visited the probability  $p$  that at least one of them is visited will lie between a dependent, if the visit of the first MC implies very strongly the visit of the second MC, and independent probability.

$$p_{dep} \leq p \leq p_i + p_j = p_{ind}$$

From the ARW values the probability  $p_{ind}$  for a random walk visiting any MC in subpartition  $\{0, 6, 5, 0\}$  yields:

$$p_{ind} = \sum_{i=1}^{counts} \frac{1}{ARW_i} = 1,001 \simeq 1 = p$$

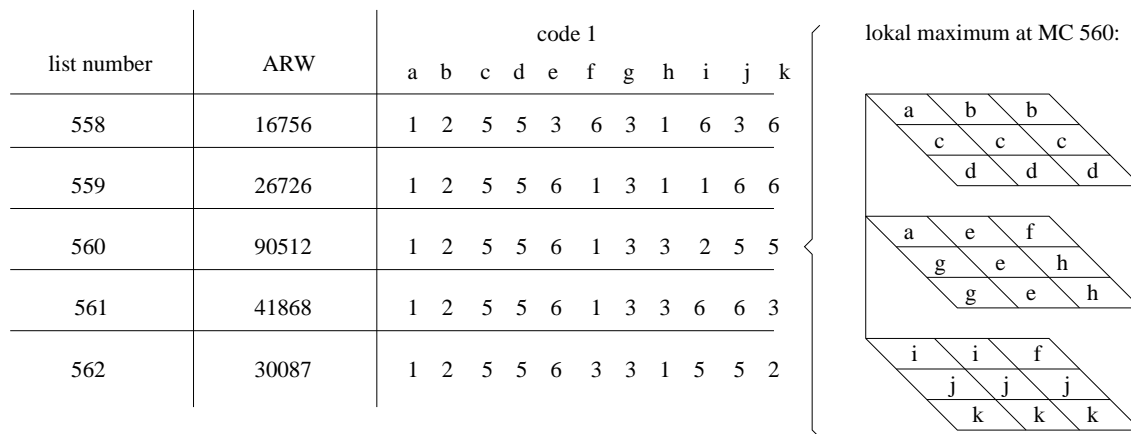
This demonstrates that for a large ensemble of MCs the assumption of independent MCs gives a good upper boundary for  $p$ .

With proceeding list number MCs have raising and falling ARW values. The lokal ARW maxima occure as ARW peaks. 24 lokal ARW maxima are larger than  $6 \cdot 10^4$ .

(b) High ARW values and ordered blocks

The ARW is defined as the average recurrent walk length over an infinit number of steps. If  $13, 14 \cdot 10^6$  instead of  $0, 5 \cdot 10^6$  steps are done the mentioned 24 MCs with ARW maxima remain the only lokal ARW maxima larger than  $6 \cdot 10^4$ . Their ARW values are of course improved (figure 71). A closer investigation reveals that the 24 MCs belong to two rotation classes, each containing 12 MCs. It is clear that rotameres of a MC converge against the same ARW.

The second high lokal maximum at list number 560 (MC 560), which is a rotamer of the first high maximum, is now regarded. The code 1 and ARW of MC 560 and the preceeding and succeeding MCs are listed in figure 71.

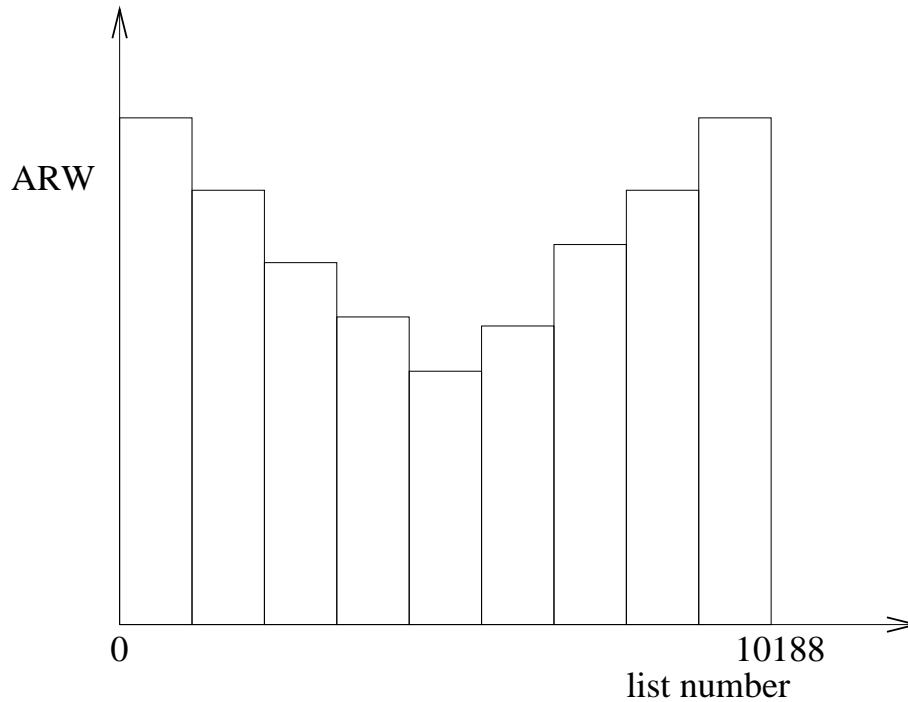


**Figure 71:** Structure of MC 560

Because MC 561 differs only in the three last blocks from MC 560 it has the next highest ARW of the nearer surrounding. From this and other possible examples it gets clear that two very similar MCs, like successive EC-algorithm solutions, have always a relatively similar ARW. MC 560 has very ordered blocks. So the question shows up if the order of blocks has an influence on high ARW values.

## (b1) Expectation

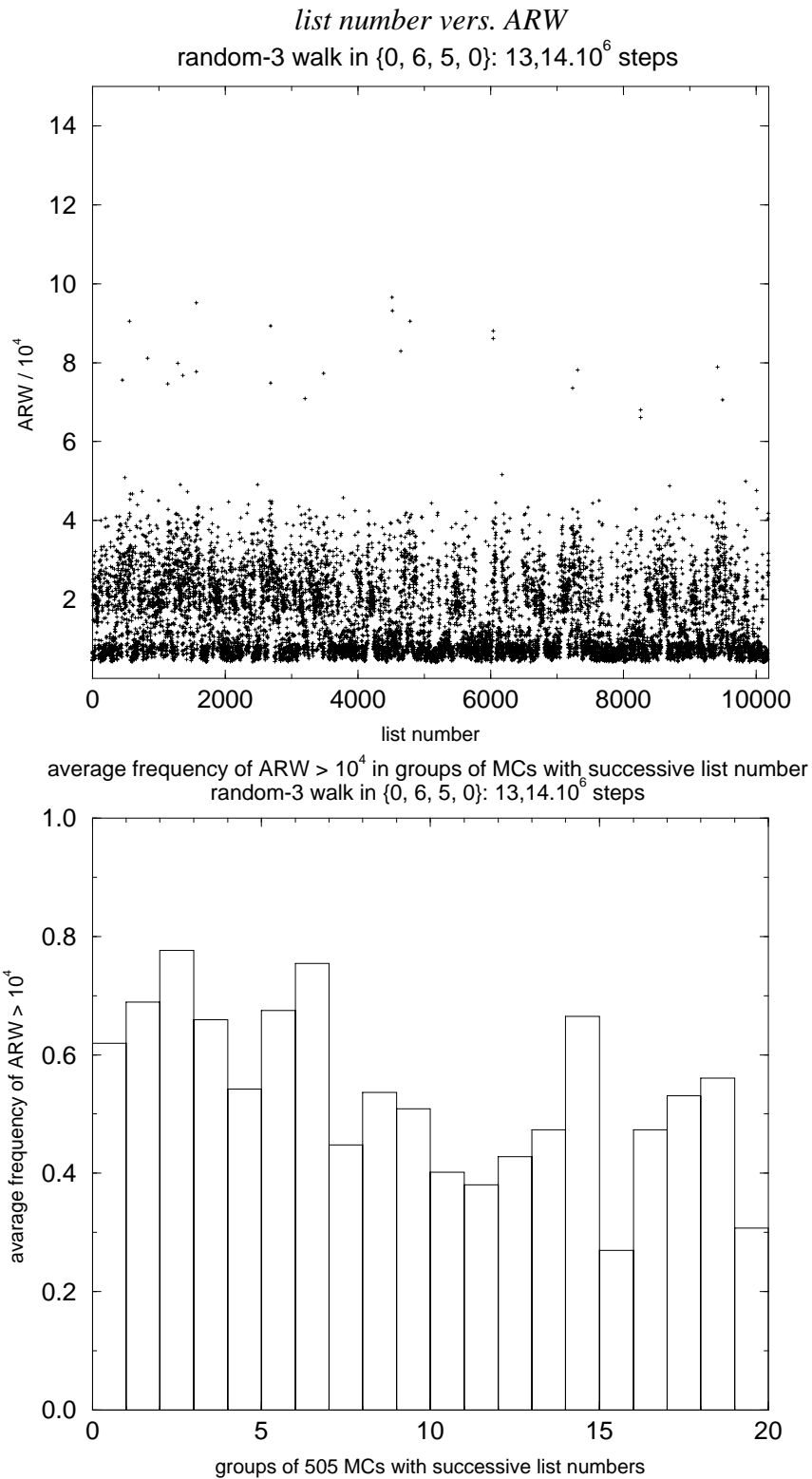
An EC-algorithm calculation starts with MCs that have blocks with lowest possible block orientation (block table) resulting in very ordered blocks. During the algorithm the blocks are packed in a more mixed orientation. At the end of the calculation all blocks are forced to be of highest possible block orientation resulting again in MCs with very ordered blocks. If the order of blocks determines the ARW of a MC a histogram like figure 72 should be observed.



**Figure 72:** Expected histogram

## (b2) Result

In figure 73 blocks of 505 succeeding MCs are analysed. In spite of the fact that the number of MCs with  $ARW > 10^4$  decreases with the list number, the histogram is not of the expected form. The order of blocks is therefore no good way to predict a high or low ARW of a MC.



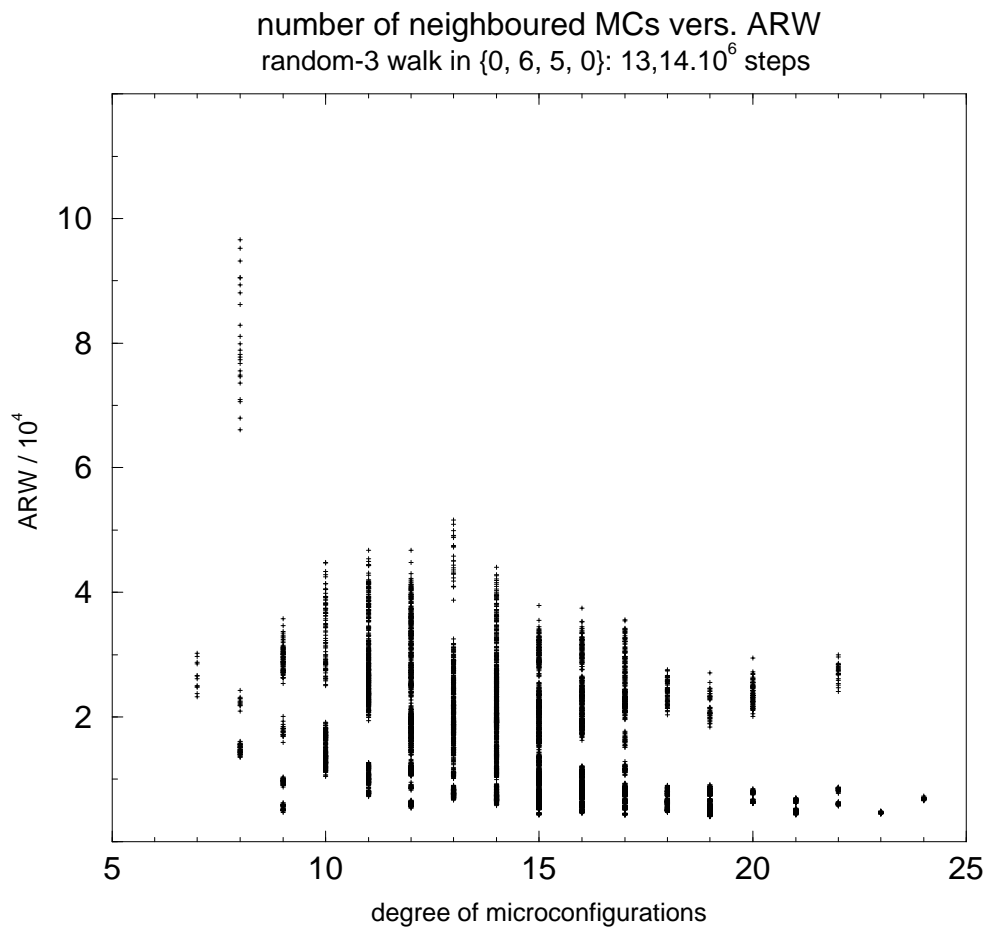
**Figure 73:** Average frequency of MCs with  $ARW > 10^4$  in groups of 505 successive MCs (EC-algorithm solutions)



## (c) Degree/ARW plot

Each MC in MCS is represented by a vertex of a graph  $G$ . If a move between two MCs is possible the two vertices of  $G$  are connected by an edge. The degree of a MC is the degree of its corresponding vertex on  $G$ .

The calculation of the degree and ARW of all MCs in subpartition  $\{0, 6, 5, 0\}$  is shown in figure 74. An ergodic random walk with steps of  $d \leq 3$  is used.



**Figure 74:** Degree and ARW of MCs in  $\{0, 6, 5, 0\}$

A clear negative correlation between the ARW and the degree can be seen. MCs of larger degree tend to be visited more frequently (= low ARW value) than MCs of lower degree. The two MCs and their rotamers with  $ARW > 10^6$  have a very low degree of 8.

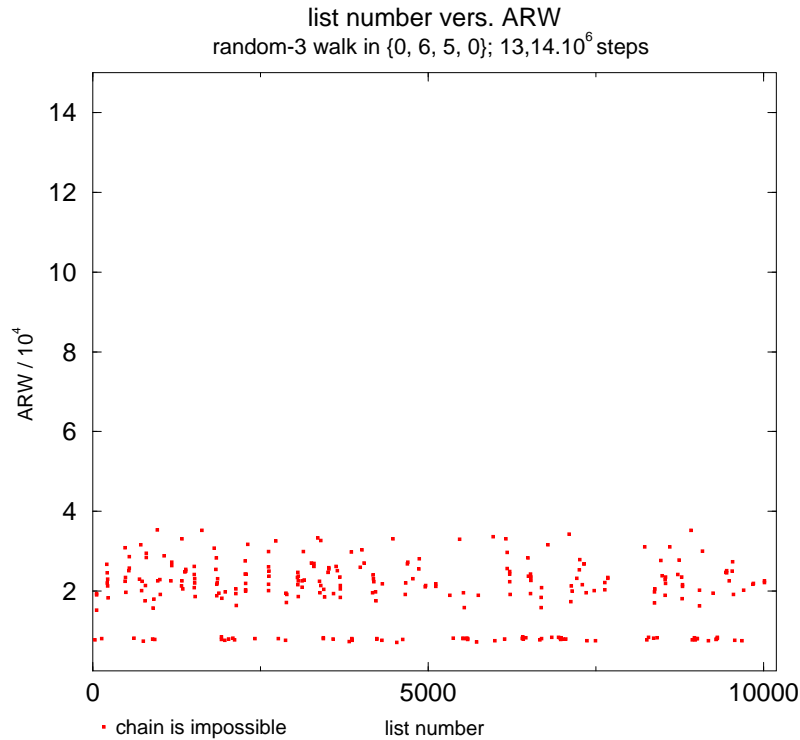
### 6.3. Chain in microconfiguration space and sequence space

With given subpartition a MCS is determined. The MCs that are generated by a folded chain are distinguished from the rest of MCs by the selection rule that the chain has to pass through each block exactly one time but may not pass the middle cell of blocks with three cells (linear or angled).

#### 6.3.1. Probability of MCs

(a) ARW values and EC-algorithm solutions

The subpartition  $\{0, 6, 5, 0\}$  has 10188 MCs with 276 MCs, distributed over all list numbers, where a chain is impossible (figure 75).



**Figure 75:** ARW of MCs where a chain is impossible

From several random walks (each more than  $10^6$  steps) the probability  $p_{exp}$  for a MC to have a chain was determined. Equivalence of  $p_{exp}$  with  $p_{ind}$ , which is calculated from the ARW values, is observed.

$$p_{exp} = 0,982 = p_{ind} \doteq \sum_{i=\text{MCs with chain}} \frac{1}{ARW_i}$$

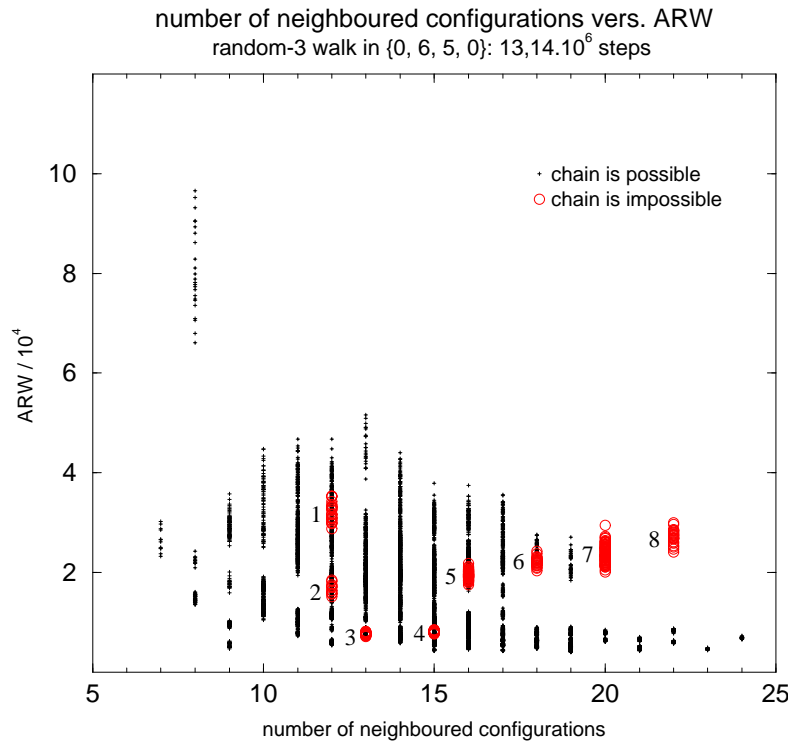
The value of 0,982 is larger than the ratio  $R = \frac{\text{MCs with chain}}{\text{counts}} = \frac{9912}{10188} = 0,973$ . The random walk visits MCs with chain therefore more often than MCs without chain.

(b) Degree/ARW plot

Some considerations on graph  $G$  are:

If MC  $x$  has a high degree and therefore a large number of possible moves to other MCs and if these surrounding MCs have also a high degree, the ARW of  $x$  is low. A second consequence of a high degree of a MC and its surrounding should be a higher combinatorial flexibility of the chain between the blocks of this MC. For MCs with high degree and high accessibility, which means low ARW, a chain should be more probable. It is expected that large and dense regions of  $G$  consist of MCs where a chain is more probable than in large regions of  $G$  that consist of MCs of low degree.

Again subpartition  $\{0, 6, 5, 0\}$  is regarded. In figure 77 those 276 MCs where a chain is impossible are distinguished from the other MCs.



**Figure 76:** Degree and ARW of MCs without chain. The eight different classes of MCs where no chain is possible are numbered.

Figure 76 is in accordance with the expectation. In all MCs of low ARW and high degree a chain is possible.

Those MCs where no chain is possible are localized at eight regions in the degree/ARW plot. If in a MC no chain is possible the same holds for the rotated and reflected forms of that MC. A closer investigation shows that they consist of 10 representative MCs and their rotated and reflected forms (figure 77).

class	number of MCs	description
1	24	$R_1$ : 24 MCs
2	12	$R_1$ : 12 MCs
3	24	$R_1$ : 24 MCs
4	48	$R_1$ : 24 MCs, $R_2$ : reflected MCs of $R_1$
5	48	$R_1$ : 24 MCs, $R_2$ : reflected MCs of $R_1$
6	24	$R_1$ : 24 MCs
7	72	$R_1$ : 12 MCs, $R_2$ : reflected MCs of $R_1$ $R_3$ : 12 MCs, $R_4$ : reflected MCs of $R_3$ $R_5$ : 24 MCs
8	24	$R_1$ : 24 MCs

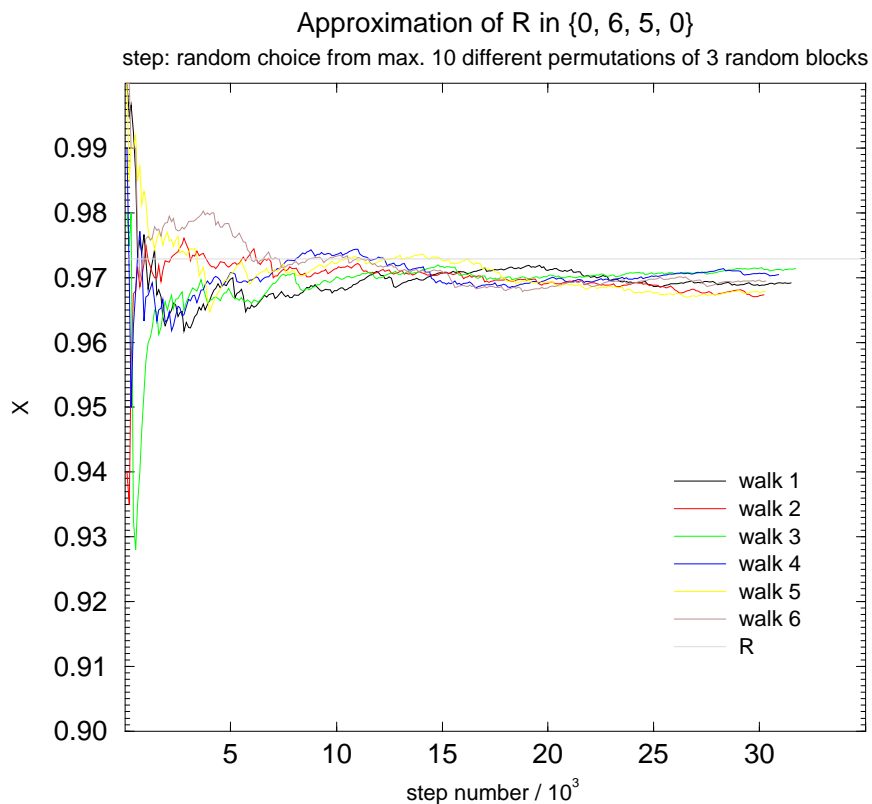
**Figure 77:** Analyse of MC where no chain is possible

All rotameres and enantiomeres of a MC have the same ARW. Class 1, 2, 3, 4, 5, 6, and 8 consist of rotameres and enantiomeres of one representative MC and their ARWs should be exactly the same. Obviously more than  $13,14 \cdot 10^6$  steps have to be done to reach an identity of ARW values. Class 7 contains three independent MCs.

### 6.3.2. Approximation of $p$ using walks with restricted degree

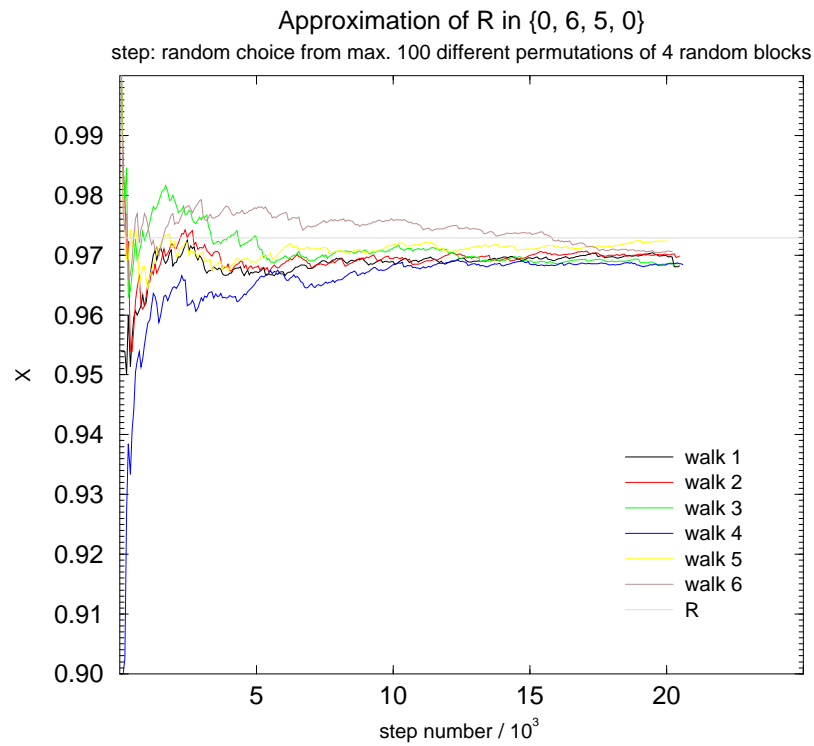
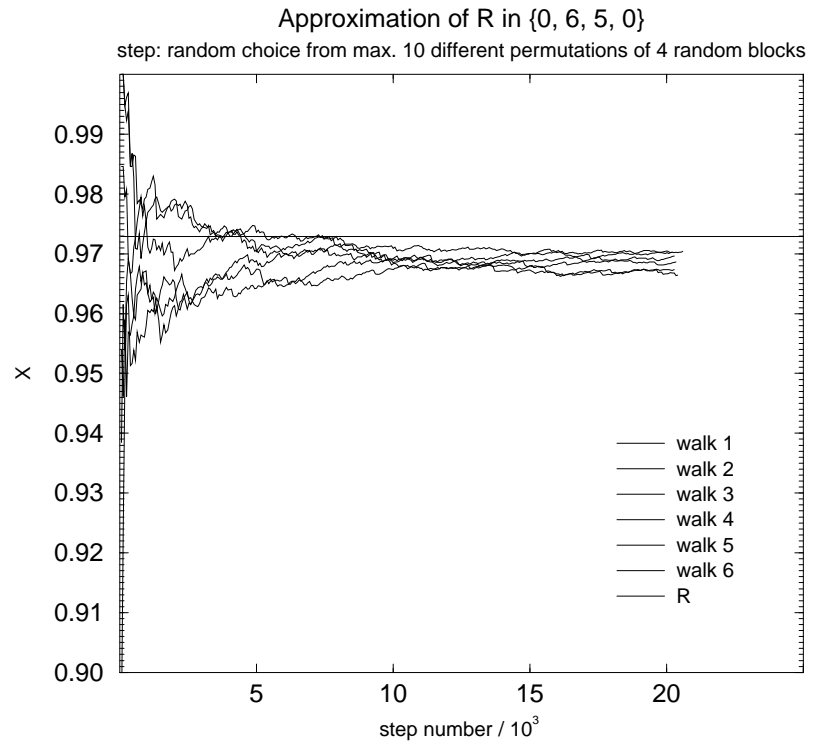
The degree of a MC is the number of different block permutations that lead to MCs within a certain distance ( $0 < d \leq 3$ ). To find the next step of a real random walk all possible different permutations are determined and one is randomly chosen as next step. This works for subpartitions having MCs with not too high degree in the  $3 \times 3 \times 3$  cube. For cubes of larger dimensions it is very time consuming to determine all possible permutations.

In MC  $\{0, 6, 5, 0\}$  the range of possible degrees  $deg$  is  $7 \leq deg \leq 24$ . In figure 78 not more than 10 different permutations are determined and one of them is chosen randomly.

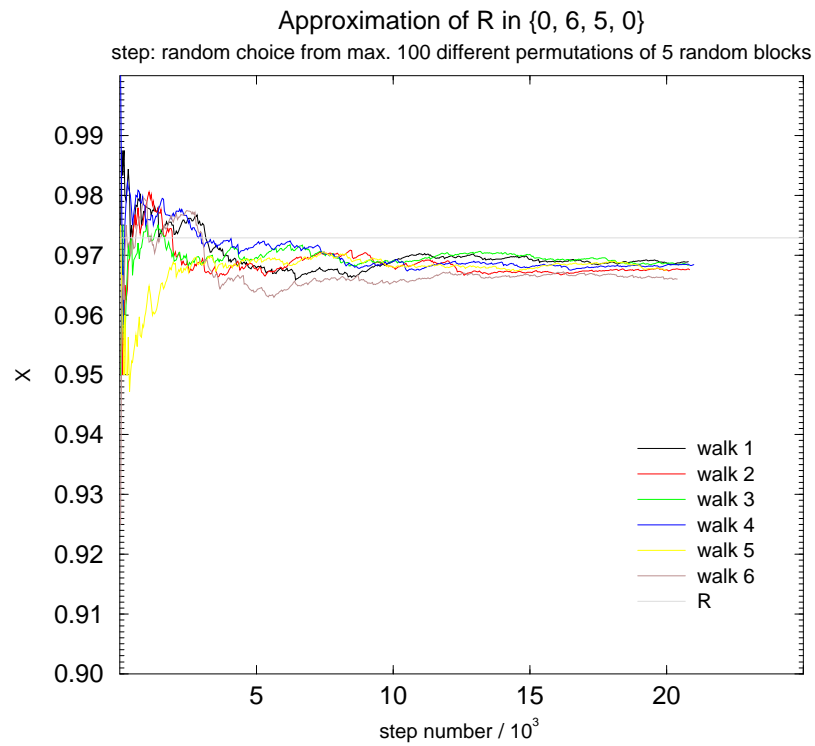
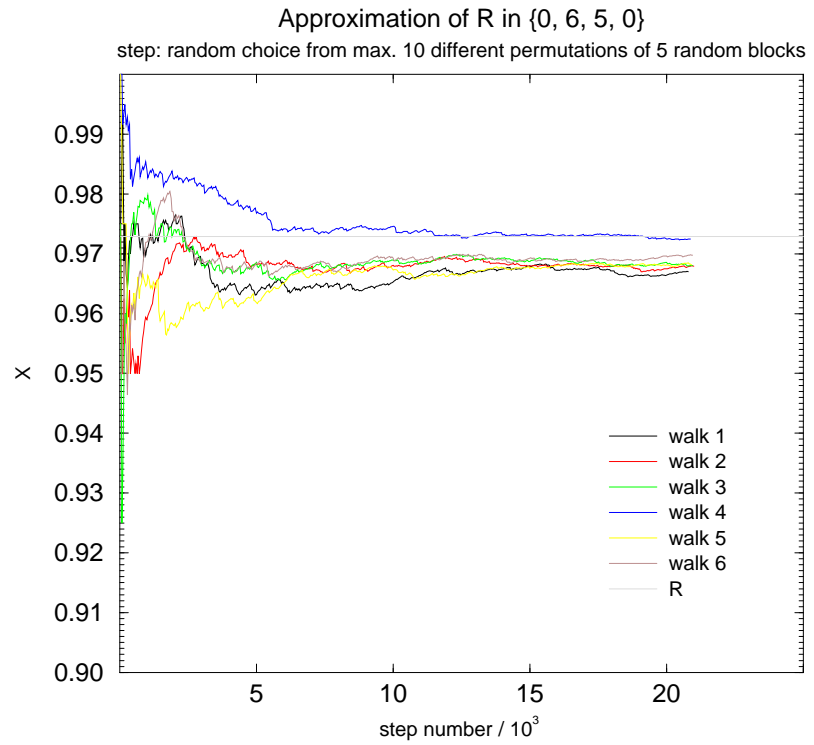


**Figure 78:** Random choice of the next MC out of subset  $S$ ,  $|S| \leq 10$ , of the set of possible permutations of three blocks

A convergence against a value (0,970) which lies below  $r$  (0,973) is observed. Compared to a random walk ( $p_{exp} = 0,982$ ) there is an error of 0,012. - This low difference to  $p_{exp}$  of a real random walk makes the idea attractive.



**Figure 79:** Random choice of the next MC out of subset  $S$  of the set of possible permutations of four blocks



**Figure 80:** Random choice of the next MC out of subset  $S$  of the set of possible permutations of five blocks

MCs with higher A.R.W. values in higher permutations

In section 6.2.2. it was described for the  $d \leq 3$  randomwalk that MCs with lower surrounding number had higher A.R.W. values. Now this reachability problem was investigated in the higher ergodic walks  $d \leq 4$  and  $d \leq 5$ .

### 6.3.3. Sequence space

If blocks of a given subpartition are lined up on a chain in general different block sequences are possible. With  $f_s$  as fraction of sequences in a subpartition that can fold to a cube subpartitions of the  $3 \times 3 \times 3$  and  $4 \times 4 \times 4$  cube were analyzed.

$3 \times 3$ subpartition	$\times 3$ sequences	cube $f_s$	$4 \times 4$ subpartition	$\times 4$ sequences	cube $f_s$
{27, 0, 0, 0}	1	1	{64, 0, 0, 0}	1	1
{1, 13, 0, 0}	14	1	{0, 32, 0, 0}	1	1
{0, 0, 9, 0}	1	1	{1, 0, 21, 0}	21	1
{0, 0, 0, 9}	1	1	{1, 0, 0, 21}	21	1
{0, 6, 5, 0}	254	0,55			
{0, 6, 0, 5}	254	0,77			
{6, 6, 1, 2}	1261260	0,68			
{13, 4, 1, 1}	813960	0,65			

table 11

On the one hand it can be seen that chains with a maximal number of blocks of the same type fold into a cube. If a mixture of blocks is used a chain sequence with flexible  $r_{3b}$  blocks has better chances to fold than a chain sequence with linear blocks.



## 7. Conclusion and outlook

The work presented in this thesis deals with an aspect of the inverse folding problem for proteins. Given a structure the task is to find amino acid sequences that form this structure under the specified folding conditions. A simplified model for protein folding is developed which mimicks amino acid residues by blocks of different shapes. The blocks are composed from elementary cubes. Shapes of the blocks resemble space filling models of real amino acid residues. The model is applied to the formation of compact hydrophobic cores. In accordance with the simplifications of the model the compact core is represented by a cube of  $n^3$  elementary cubes. The problem is to find compact “puzzle-like” foldings of sequences which can be arranged in the cube without leaving blocks unoccupied. Folding probabilities are computed by exhaustive enumeration of successful and unsuccessful trials.

A formula was obtained to enumerate the sequences for given  $n$ . A certain arrangement of blocks in a cube was called microconfiguration (MC), and all MCs form the corresponding microconfiguration space (MCS). It was proven that the number of different blocks between two MCs is a metric. Two MCs at distance two (three) can be transformed into each other by a permutation of the two (three) differing blocks.

To classify permutations the concept of a move type was developed. Every permutation of two (three) blocks belongs to one of five (thirtythree) corresponding move types.

- (i) For applied sets of blocks with not too many MCs an algorithm was developed to exhaustively enumerate the MCs. The rotational and mirror symmetry of MCs was analyzed.
- (ii) Random walk was tested in small MCSs and was proposed as a method to investigate large MCSs.

Moves involving the permutation of two blocks resulted in random walks being restricted to subspaces of the MCS. If steps permutating three blocks were used, ergodic random walks ensued. The probability to reach a MC in a random walk is the inverse of this walk length.

The random walk algorithm calculates for each step all possible permutations and chooses one randomly. To reduce computation time only a restricted number of permutations was calculated for each step. The error in the probability of MCs was observed to be small.

For some sequences with an applied set of blocks in the  $3 \times 3 \times 3$  and  $4 \times 4 \times 4$  cube it was possible to compute the folding probabilities. The available data show that chains with a maximal number of blocks of the same type can always fold into a cube.

The present study has shown that the simplified block model is suitable for the analysis of compact hydrophobic cores. In particular, probabilities of forming compact cores can be defined and computed. Investigations were restricted here to very small cubic arrangements of 27 and 64 blocks (with  $n = 3$  and 4, respectively). In order to study realistic proteins with hundred amino acid residues and more the current algorithm is not suitable because the computations increase too fast, presumably exponentially with the size of the core. Future work on the model presented in this thesis would therefore be dealing with the development of suitable statistical evaluation methods of folding probabilities. The model can be easily extended to handle larger amino acid residues resembling more closely the natural shapes. Such an extension would necessarily require larger cubes for representing hydrophobic cores.

## 8. References

- [1] R. L. Baldwin. Protein conformation. In *CIBA Foundation Symposium*, number 161, pages 22–24, January 1991.
- [2] H. S. Chan and K. A. Dill. Energy landscapes and the collapse dynamics of homopolymers. *J. Chem. Phys.*, 99:2116–2127, August 1993.
- [3] H. S. Chan and K. A. Dill. The protein folding problem. *Physics today*, pages 24–32, February 1993.
- [4] H. S. Chan and K. A. Dill. Transition states and folding dynamics of proteins and heteropolymers. *J. Chem. Phys.*, 100(12), 1994.
- [5] K. A. Dill. Dominant forces in protein folding. *Biochemistry*, 29(31):7133–7155, August 1990.
- [6] K. A. Dill and et al. Principles of protein folding—a perspective from simple exact models”. *Protein Science*, 4:561–602, January 1995.
- [7] K. A. Dill and et al. A test of lattice protein folding algorithms. *Proc. Natl. Acad. Sci. USA*, 92:325–329, January 1995.
- [8] D. A. Dolgikh, L. V. Abaturov, I. A. Bolotina, E. V. Brazhnikov, V. E. Bychkova, V. N. Bushuev, R. I. Gilmanshin, Y. O. Lebedev, G. V. Semisotnov, E. I. Tiktopulo, and O. B. Ptitsyn. *Eur. Biophys. J.*, 13:109–121, 1985.
- [9] D. A. Dolgikh and et al. *FEBS Letters*, 163:311–315, 1981.
- [10] D. A. Dolgikh, A. Kolomiets, I. A. Bolotina, and O. B. Ptitsyn. *FEBS Letters*, 1984.
- [11] G. A. Elöve, A. F. Chaffotte, H. Roder, and M. E. Goldberg. Early steps in cytochrome c folding probed by time-resolved circular dichroism and fluorescence spectroscopy. *Biochemistry*, 31:6876–6883, 1992.
- [12] K. M. Fiebig and K. A. Dill. Protein core assembly processes. *J. Chem. Phys.*, 98:3475–3487, February 1993.
- [13] A. Finkelstein. *Biopolymers*, 28:1681–1694, 1989.
- [14] A. Godzik, A. Kolinski, and J. Skolnick. Lattice representations of globular proteins: How good are they? *Journal of Computational Chemistry*, 14(10):1194–1202, 1993.

- [15] F. Hughson and et al. *Biochemistry*, 30:4113–4118, 1991.
- [16] F. M. Hughson and R. L. Baldwin. *Biochemistry*, 28:4415–4422, 1989.
- [17] M. Karplus and G. A. Petsko. Molecular dynamics simulations in biology. *Nature*, 347:631–639, October 1990.
- [18] K. Kuwajima, Y. Hiraoka, M. Ikeguchi, and S. Sugai. Comparison of the transient folding intermediates in lysozyme and alpha-lactalbumin. *Biochemistry*, 24:874–881, 1985.
- [19] V. I. Lim. Structural principles of the globular organization of protein chains. a stereochemical theory of globular protein secondary structure. *J.Mol.Biol.*, 88:857–872, 1974.
- [20] W. Pfeil and et al. *FEBS Letters*, 198:287–291, 1986.
- [21] O. Ptitsyn. How molten is the molten globule? *nature structural biology*, 3(6):488–490, June 1996.
- [22] O. B. Ptitsyn, R. H. Pain, G. V. Semisotnov, and O. I. Razgulyaev. *FEBS Letters*, 1990.
- [23] S. E. Radford, C. M. Dobson, and P. Evans. The folding of hen lysozyme involves partially structured intermediates and multiple pathways. *Nature*, 358:302–639, July 1992.
- [24] F. M. Richards. The interpretation of protein structures: Total volume, group volume distributions and packing density. *J. Mol. Biol.*, 82:1–14, 1974.
- [25] F. M. Richards. Areas, volumes, packing, and protein structure. *Ann. Rev. Biophys. Bioeng.*, 6:151–176, 1977.
- [26] N. A. Rodionova, G. V. Semisotnov, V. P. Kutysenko, V. N. Uversky, I. A. Bolotina, V. E. Bychkova, and O. B. Ptitsyn. *Mol. Biol.(USSR)*, 23:683–692, 1989.
- [27] R. T. Sauer. Protein folding from a combinatorial perspective. *Folding & Design*, (1):R27–R30, April 1996.
- [28] J. A. Schellman. The thermodynamic stability of proteins. *Ann. Rev. Biophys. Biophys. Chem.*, 16:115–137, 1987.
- [29] K. Yue and K. A. Dill. Sequence-structure relationships in proteins and copolymers. *The American Physical Society*, 48(3):2267–2278, 1993.
- [30] R. Zwanzig, A. Szabo, and B. Bagchi. Levingthal’s paradox. *Proc. Natl. Acad.Sci. USA*, 89:20–22, January 1992.

## 9. Curriculum Vitæ

- Name : Josef Erich Frömcke  
Geburtsdatum und -ort : 26. Februar 1967 in Wien  
Staatsbürgerschaft : Österreich
- Schulbildung :
- 1973 – 1977 : Volksschule in Wien  
1977 – 1986 : Naturwissenschaftliches Realgymnasium  
(1978-1979: Zeit wegen Krankheit verloren)  
26. Mai 1986 : Matura
- Studium :
- September 1986 : Immatrikulation an der Universität Wien  
(Studienrichtung Chemie; Studiengang Biochemie)  
3. Oktober 1989 : 1. Diplomprüfung  
Oktober 1992 : Beginn der Diplomarbeit am Institut für Biochemie und  
molekulare Zellbiologie bei Prof. Dr. Gerhard Wiche  
16. September 1994 : 2. Diplomprüfung  
Oktober 1994 : Sponsion zum Magister der Naturwissenschaften  
Beginn der Dissertation bei Prof. Dr. Peter Schuster

## Table of Contents

1. Introduction	3
1.1. Proteins	3
1.2. The structure of proteins	4
1.2.1. Elements of secondary structure	6
1.2.1.1. Helices	6
1.2.1.2. Reverse turns	7
1.2.1.3. Sheets	8
1.2.2. Elements of supersecondary structure	9
1.3. States in protein folding	10
1.3.1. General considerations	10
1.3.2. The native state as free energy minimum	10
1.3.3. The molten globule state	11
1.3.3.1. $\alpha$ -Lactalbumin	11
1.3.3.2. Cytochrome c	12
1.3.3.3. Lysozyme	12
1.3.3.4. General properties	13
1.4. Models for protein folding	15
1.4.1. Molecular dynamics simulation	15
1.4.2. Lattice models	15
1.4.2.1. Collaps models	15
1.5. Present Work	17
2. The Compact Block Model	19
2.1. Terminology	19
2.1.1. Blocks	19
2.1.2. Partition, subpartition, and microconfiguration	20
2.1.3. Considerations on arithmetical restrictions	21
2.1.3.1. Number of partitions and subpartitions	21
2.1.3.2. Chain length	24
2.2. Coding of microconfigurations	25
2.2.1. Cell position	25
2.2.2. Block orientations	27

2.2.3. Code 1	28
2.2.4. Code 2	28
3. Exhaustive calculation of microconfigurations	30
3.1. Basic concepts	30
3.1.1. Algorithm	31
3.1.2. Resulting code 1	32
3.2. Symmetry of microconfigurations	33
3.2.1. Rotation symmetry	33
3.2.1.1. Algorithm for rotation	34
3.2.2. Reflection symmetry	36
3.2.2.1. Reflection plane	37
3.2.2.2. Algorithm for reflection	39
4. Sequence Space And Microconfiguration Space	40
4.1. Sequence space	40
4.1.1. Number of sequences in a given cube	41
4.1.1.1. No $r_{3a}$ or $r_{3b}$ residues	41
4.1.1.2. Use of all residues	42
4.1.2. Sequences and microconfigurations	43
4.2. Distance and microconfiguration space	44
4.2.1. Distance $d(r, s)$ of two microconfigurations $r$ and $s$	44
4.2.2. Rotation reduced distance $d(p, R_q)$	47
5. Block permutations and graphs	48
5.1. Representation of block sets	48
5.1.1. Coarse grained representation of block sets	48
5.1.2. Projection of block set representations	50
5.2. Cell migrations as method to modify blocks in a block set	50
5.3. Permutation of block set representations	55
5.3.1. Developement of move type sets $R2$ and $R3$	60
5.3.1.1. Considerations on move types in $R2$ and $R3$	60
5.3.1.2. Move type sets $R2$ and $R3$	64

6. Results	72
6.1. Exhaustive calculation of microconfigurations	72
6.1.1. Counts	72
6.1.2. Symmetry	74
6.1.3. Distance	78
6.2. Random walk with move sets M2 and M3	81
6.2.1. Move set M2	81
6.2.1.1. Diffusion	81
6.2.1.2. Accessible microconfigurations	83
6.2.1.3. Subspaces	85
6.2.2. Move set M3	86
6.2.2.1. Accessible microconfigurations	86
6.2.2.2. Random walks in similar subpartitions	88
6.2.2.3. Probability $p$ of MCs in a random walk	88
6.3. Chain in microconfiguration space and sequence space	95
6.3.1. Probability of MCs	95
6.3.2. Approximation of $p$ using walks with restricted degree	98
6.3.3. Sequence space	101
7. Conclusion and outlook	102
8. References	104
9. Curriculum Vitæ	106