CROATICA CHEMICA ACTA CCACAA **68** (1) 91–98 (1995)

ISSN 0011-1643
CCA–2214 *Original Scientific Paper*

On the Complexity of Square-Cell Configurations*

Nenad Trinajstić

The Rugjer Bošković Institute, P.O.B. 1016, HR-41001 Zagreb, Croatia

Wolfgang R. Müller, Klaus Szymanski and Jan V. Knop

The Computer Centre, The Heinrich Heine University, D-40225 Düsseldorf, Germany

Received July 1, 1994; revised October 20, 1994; accepted October 20, 1944

The procedure is proposed for obtaining the complexity numbers of square-cell configurations. It is based on the concept of the canonical square-cell configuration. The complexity number of a square-cell configuration is then simply the minimal of edge-cuts by which this structure can be reduced to constituting canonical configurations.

INTRODUCTION

Recent interest in square-cell configurations in chemistry[1–3] prompted the present report. The name used for these configurations in discrete mathematics is square animals.[4] A square-cell configuration is made up of squares which are simply- or multiply-connected.[5] It starts with a single square and grows by adding squares, one at a time, in such a manner that the new square has at least one side in contact with a side of a square already present in the square-cell configuration. A square-cell configuration is simply-connected if it does not posses a hole, while the multiply-connected square-cell configuration is a configuration with hole(s). In Figure 1, a simply-connected square-cell configuration with nine squares is given as an example.

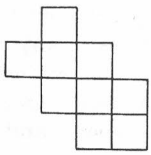


Figure 1. A graph representing a simply-connected square-cell configuration with nine cells.

* Reported in part at MATH/CHEM/COMP 1994, an international Course and Conference on the Interfaces between Mathematics, Chemistry and Computer Science, Dubrovnik, Croatia: June 26 – July 1, 1994.

In this work, we will be concerned only with the simply-connected square-cell configurations. However, our results can be extended to multiply-connected square-cell configurations without any difficulty.

Statistical properties of square-cell configurations (in this case usually referred to as lattice animals) and their embeddings in square lattices are important, for example, in modelling the thermodynamic properties of branched polymers in dilute solutions[6–10] and for characterizing shapes of two-dimensional solids and molecular aggregates on the surfaces of catalysts.[11] Additionally, analysis of the shapes of square-cell configurations is of interest in the design of new composite materials and in the study of interactions of enzymes with various molecules of biological relevance.[2] Square-cell configurations or lattice animals or polyominoes are also of interest in pure mathematics.[12–16]

THE NUMBERS OF SQUARE-CELL CONFIGURATIONS AND THE CORRESPONDING CANONICAL CONFIGURATIONS

We have recently developed an algorithm for the constructive enumeration of hexagonal-cell configurations (hexagonal animals, polyhexes)[17] which is based on the DAST (dualist angle-restricted spanning tree) code.[18] The DAST code was used with a slight modification to represent square-cell configurations. It was also used as the basis of a computer program for generating and enumerating all square-cell configurations with up to a given number of squares.[3]

The DAST code can be introduced in the following way. Let an entered polyomino (P,a,b) be a polyomino P together with a directed (»entrance«) edge (a,b) on its boundary. Then, the (»entrance«) square at that entrance edge may have up to three neighbours. – N_1, N_2, N_3 – beyond the other edges. This gives rise to a decomposition of the polyomino area into, at most, four disjoint areas:

P_0: the entrance square itself,

P_1: the edge-connectivity component of N_1 (if present) after elimination of P_0, N_2 and N_3,

P_2: the edge-connectivity component of N_2 (if present) after elimination of P_0, P_1 and N_3,

and

P_3: the edge-connectivity component of N_3 (if present) after elimination of P_0, P_1 and P_2.

Obviously, this decomposition will not be independent of the ordering of the neighbours, so there must be an arbitrary but then fixed convention about this sequel. To code the presence or emptiness of any of the components in the three bits of an octal digit, there must also be a convention on mapping. With the path (a,b,c,d,a) around the entrance square, we found it most convenient to look first between d and a with weight 4, then between c and d with weight 1 and lastly between b and c with weight 2 (see Figure 2). This gave us, at most, three smaller entered polyominoes (P_1,a,d), (P_2,d,c) and (P_3,c,b). Application of this decomposition recursively definitely terminates in the cases with a single square.

Now, we can define the »DAST (dualist angle-restricted spanning tree) tuple« of an entered polyomino (with respect to the above convention) recursively: We add the weights of the neighbours present to obtain a digit from 0 to 7 and append to it the

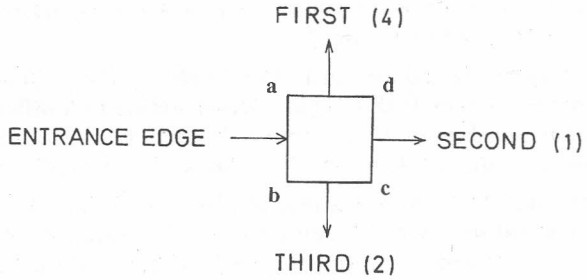


Figure 2. Selected directions relative to the entrance edge of the starting square and their weights.

DAST tuples of the up to 3 smaller components (if present) ordered by the convention. The same rule gives a DAST tuple of 0 for the one-square polyomino, which completes the recursion.

For a polyomino P, we define the »DAST code« as the lexicographic minimum of the DAST tuples of those up to 8 entered polyominoes (P,a,b) for which, after a suitable rotation and reflection, no vertex of the polyomino lies more to the west than or exactly north of a (a being a corner is necessary, but not a sufficient condition for this).

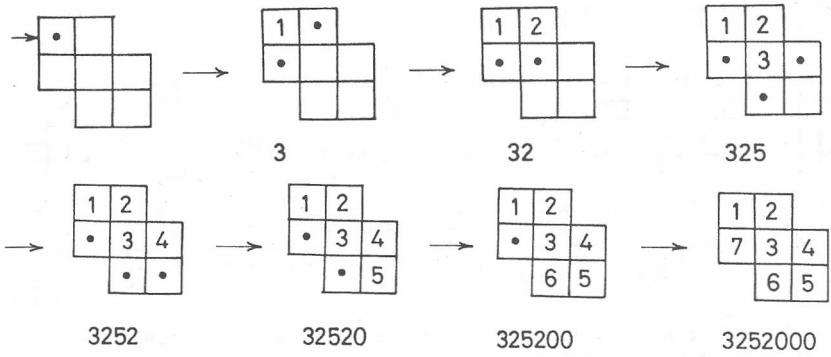
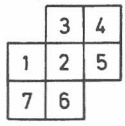


Figure 3. The step-by-step development of the DAST code for a square-cell configuration with seven squares. Black dots indicate square already reserved for later processing. The label in a square denotes the position in the DAST code corresponding to this square.



3720000

Figure 4. The DAST tuple for another orientation of the square-cell configuration from Figure 3.

The step-by-step development of the DAST code for a square-cell configuration with seven cells is presented in Figure 3.

This is lexicographically the smallest DAST code for the studied structure. We can easily write down another DAST tuple, corresponding to a different orientation of this configuration (see Figure 4), but this code cannot be taken into account because it is lexicographically greater than the code given in Figure 3.

The computer program, a block-diagram of which is given in Ref. 3, was adapted to compute the canonical square-cell congifurations.[1] A square-cell configuration that cannot be nontrivially reduced into a smaller one by the procedure called the squashing of a square animal in the manner decribed by Harary and Mezey[1] is called the irreducible or canonical configuration. In Figure 5, we give shapes of canonical square-cell configurations with up to seven cells.

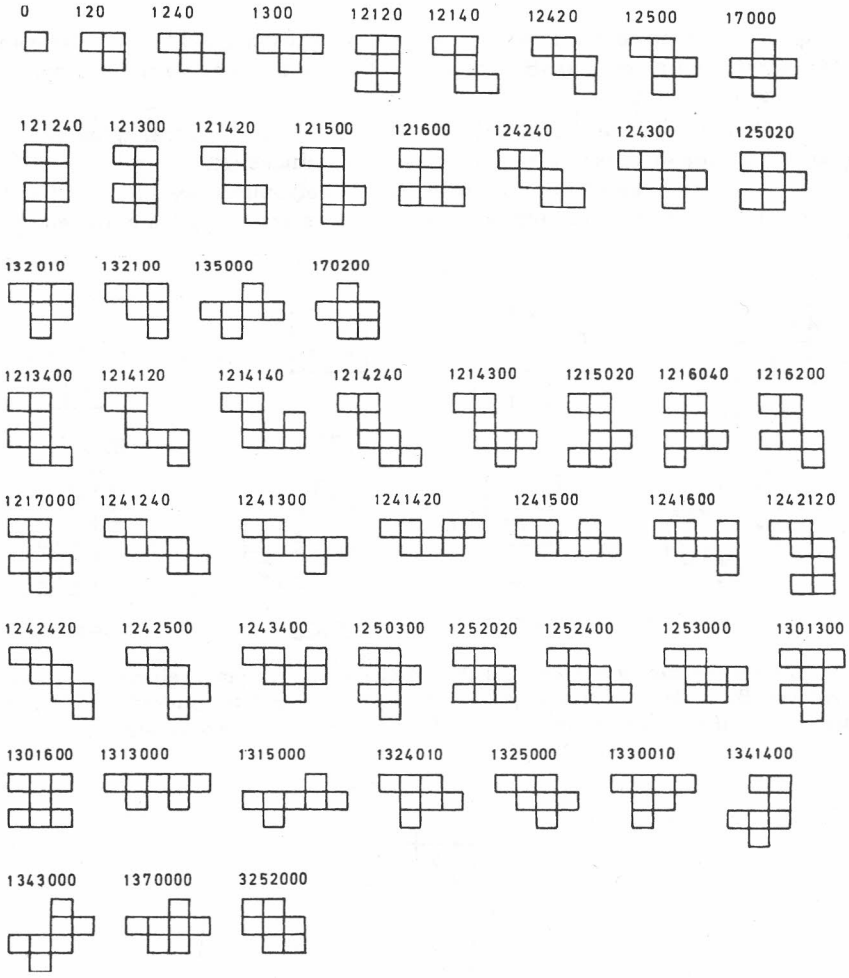


Figure 5. All canonical square-cell configurations with up to seven cells.

TABLE I

The numbers of square-cell configurations A_n and canonical square-cell configurations C_n with n cells

n	1	2	3	4	5	6	7	8	9
A_n	1	1	2	5	12	35	107	363	1248
C_n	1	0	1	2	5	12	33	95	300

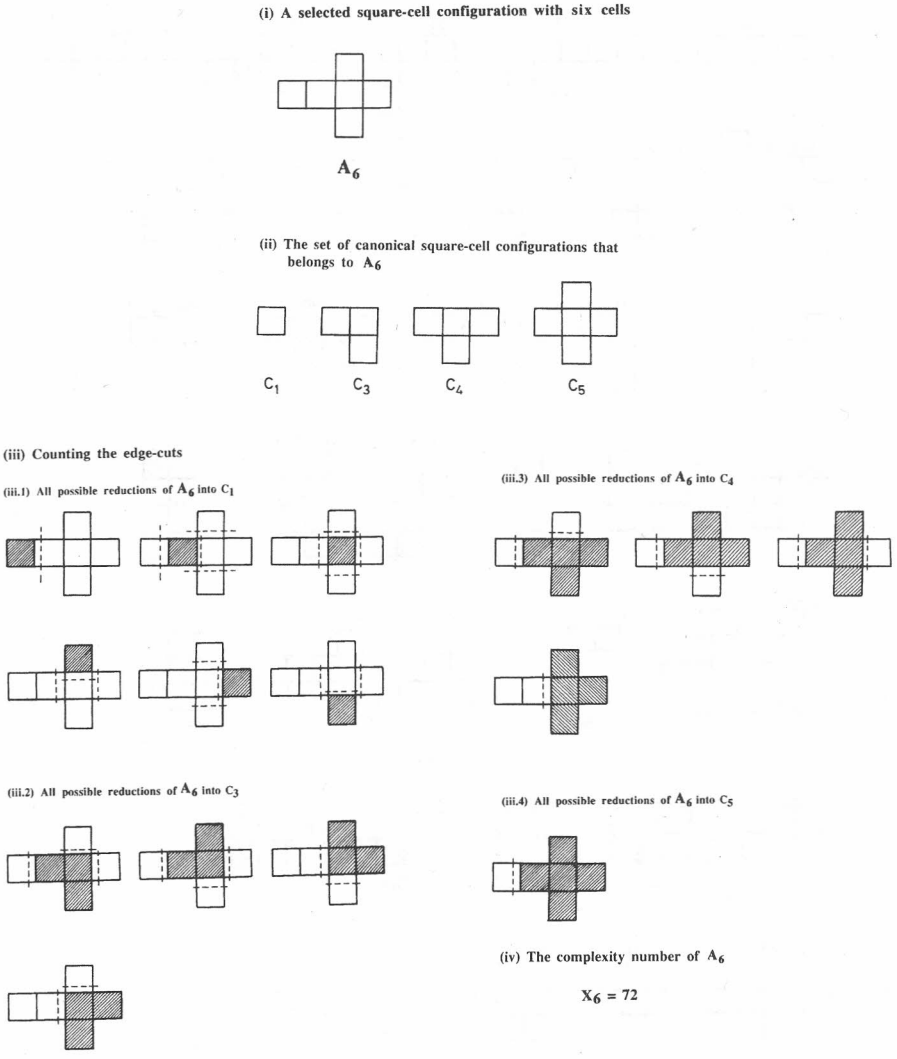


Figure 6. The step-by-step determination of the complexity number X_n for a square-cell configuration A_n with n=6.

In Table I, we give the numbers of square-cell configurations and the correspondigng canonical configurations with up to nine squares, computed using our program described earlier[3] to which we added a part related to the squashing of square animal according to the recipe given by Harary and Mezey.[1]

Computations have been carried out on a PC (386-AT, 40 MHz). The CPU time needed to complete computations reported in Table I was 6 min. 35 sec.

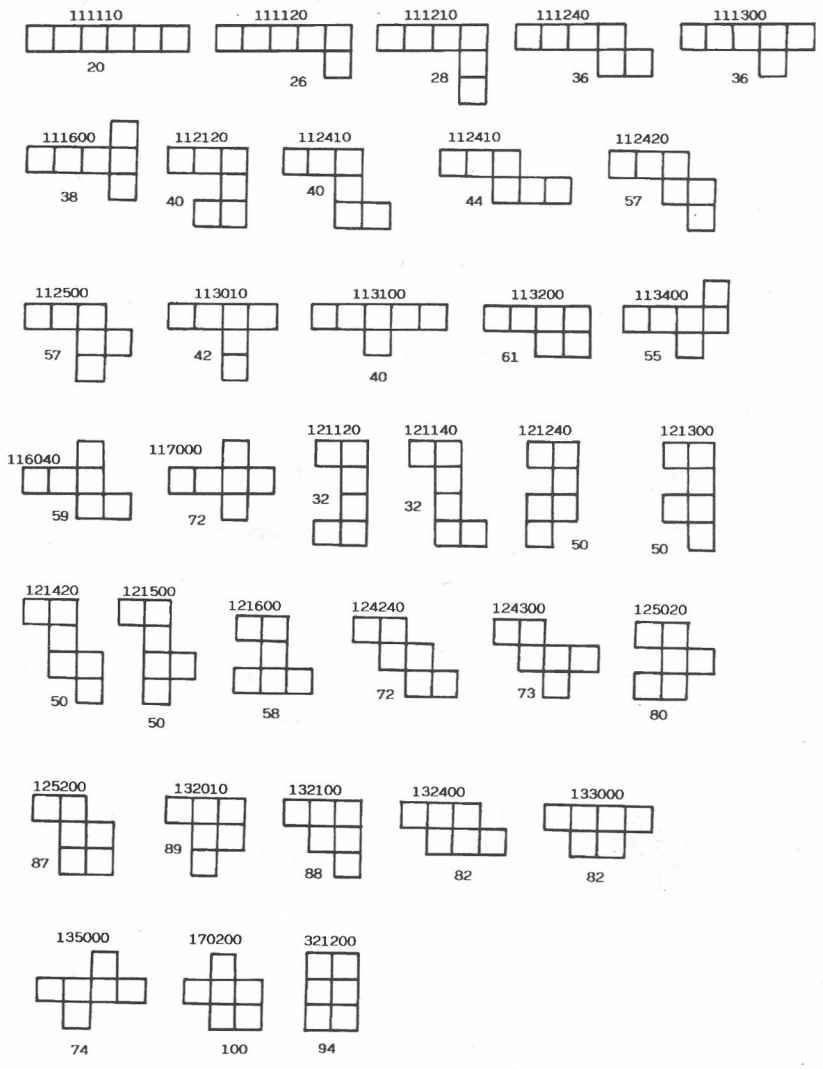


Figure 7. The DAST codes and complexity numbers for all square-cell configurations with six squares.

THE COMPLEXITY NUMBER OF SQUARE-CELL CONFIGURATIONS

We define the complexity number X_n of a square-cell configuration A_n with n squares as the sum of all minimum numbers of edge-cuts necessary to reduce A_n into the corresponding set of canonical configurations. The procedure for obtaining X_n consists of the following steps:

(i) Draw a square-cell configuration A_n with n cells;
(ii) Identify the set of canonical configurations C_k (k = 1,2,...n–1) that are subgraphs of A_n;
(iii) Count the edge-cuts by which A_n is reduced to a given C_k;
(iv) Sum the edge-cuts and this sum is the complexity number X_n of A_n.

An example, which illustrates the determination of the complexity number for a cross-like square-cell configuration with 6 squares, is presented in Figure 6.

In Figure 7, we give X_n numbers for all square-cell configurations with 6 cells. The results in Figure 7 reveal the following points:

(i) X_n increases with branching;
(ii) Isomers with a similar mode of branching possess the same X_n values;
(iii) Peri-condensed configurations posses higher values of X_n than the cata-condensed square-cell configurations.

Finally, we point out that the proposed complexity measure only very roughly follows the DAST code. Therefore, it really cannot be used to classify isomers; all it can say is that a certain isomer is more (or less or equally) complex with respect to the edge-cutting than the other. The DAST code still appears to be the most convenient way of classifying square animals. The only advantage of the proposed scheme over the DAST code is that the complexity is expressed as a number and not as a sequence of digits, like the DAST code. In some situations, this may be a more convenient piece of information. Further work on this topic is in progress.

Acknowledgement. – One of us (NT) was supported in part by the Ministry of Science and Technology of the Republic of Croatia *via* grant number 1-07-59. We thank the referees for their constructive comments.

REFERENCES

1. F. Harary and P. G. Mezey, *Theoret. Chim. Acta* **79** (1991) 379.
2. P. D. Walker and P. G. Mezey, *Int. J. Quantum Chem.* **43** (1992) 375.
3. W. R. Müller, K. Szymanski, J. V. Knop, and N. Trinajstić, *Theoret. Chim. Acta* **86** (1993) 269.
4. F. Harary, *Publ. Math. Inst. Hung.* **5** (1960) 63.
5. R. C. Read, *Can J. Math.* **14** (1962) 1.
6. S. G. Whittington, in: *MATH/CHEM/COMP 1987*, R. C. Lacher (Ed.), Elsevier, Amsterdam, 1987, p. 285.
7. C. E. Soteros and S. G. Whittington, *J. Phys.* **A21** (1988) 2187.
8. N. Madras, C. E. Soteros, and S. G. Whittington, *ibid.* **A21** (1988) 4617.
9. S. G. Whittington, C. E. Soteros, and N. Madras, *J. Math. Chem.* **7** (1991) 87.
10. R. Brak, A. J. Guttman, and S. G. Whittington, *ibid.* **8** (1991 255.

11. M. Silverberg and A. Ben-Shaul, *J. Chem. Phys.* **87** (1987) 3178.
12. M. P. Delest and G. Viennot, *Theor. Comput. Sci.* **34** (1984) 169.
13. M. P. Delest, *J. Math. Chem.* **8** (1991) 3.
14. M. P. Delest and S. Dulucq, *Croat. Chem. Acta* **66** (1993) 59.
15. S. Feretić, *Croat. Chem. Acta* **66** (1993) 81.
16. M. Delest and J. P. Dubernard, *Croat. Chem. Acta* **67** (1994) 45.
17. W. R. Müller, K. Szymanski, J. V. Knop, S. Nikolić, and N. Trinajstić, *J. Comput. Chem.* **11** (1990) 223.
18. S. Nikolić, N. Trinajstić, W. R. Müller, K. Szymanski, and J. V. Knop, *J. Math. Chem.* **4** (1990) 357.

SAŽETAK

O složenosti kvadratnih konfiguracija

Nenad Trinajstić, Wolfgang R. Müller, Klaus Szymanski and Jan V. Knop

Predložen je postupak za kvantificiranje (dobivanje broja) složenosti kvadratnih konfiguracija, koji se temelji na koncepciji kanonskih kvadratnih konfiguracija. Broj složenosti neke kvadratne konfiguracije jednak je minimalnom broju potrebnih rezanja bridova da se takva struktura reducira na jednu od svojih kanonskih konfiguracija.