One-sided Pentominoes in a 9×10 Rectangle

Preprint

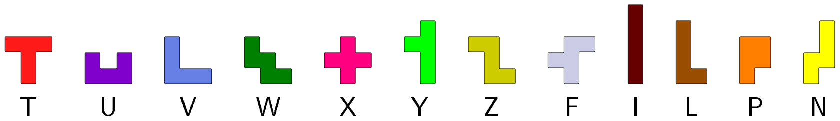
Alfred Wassermann, University of Bayreuth

May 26, 2000

Abstract

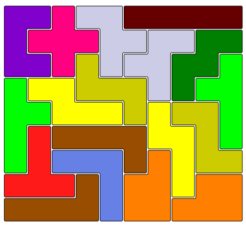
The total number of non-isomorphic coverings of a 9×10 board by the 18 one-sided pentominoes is equal to $10,440,433$. This number was determined by a parallel backtrack search. Also, the configurations with the highest symmetry have been determined.

Polyominoes are shapes made by connecting certain numbers of equal-sized squares, each joined together with at least one other square along an edge. For a positive integer n, an n-omino is a polyomino consisting of exactly n squares, 5-ominoes are called pentominoes. The definite source for polyminoes is the book of Golomb [2]. It is well known that there are 12 pentominoes, Golomb [2, p. 6] recommends to “name” them with the following letters:



If the figures are strictly confined to the plane and can not be turned over, i. e. if we distinguish the - and the -pentomino, Golomb introduced the term *one-sided n-ominoes*. In the case $n = 5$ we have 18 one-sided pentominoes.

O’Beirne [5] posed the question if the 18 one-sided pentominoes can be put into a 9×10 rectangle without allowing vacant squares or overlaps. It turns out that this is an relatively easy problem, already Golomb provided one solution in [2, p. 70]:



But the question, how many non-isomorphic arrangements for this problem exist, was still open.

These types of puzzles are well fitted for attacking them by computer search. In fact, one of the earliest application of backtrack programming was the covering of an 8×8 chessboard by the 12 pentominoes leaving the center four squares vacant [7].

Since in a solution each square of the board has to be covered by exactly one pentomino and each pentomino is placed in the rectangle exactly once, the problem can be formulated as *exact cover problem*:

> Find a 0/1 vector x such that $A \cdot x = b$, where A is a 0/1 matrix and b is the all-one vector.

The matrix A consists of two types of rows: There is a corresponding row of A for — i) every pentomino and — ii) for every square of the board, i. e. for the 18 one-sided pentominoes and a 9×10 board the matrix A has 108 rows. The columns of A correspond to all possible positions of all pentominoes. For every placement of each pentomino there exists a column of A where a "1" in one of the rows of type i) notifies which pentomino is to be placed and "1"s in five of the rows of type ii) notify which squares of the board are covered. The problem is to choose 18 columns such that all squares of the rectangle are covered exactly once and each pentomino is used exactly once. This is equivalent with multiplying A with a 0/1-vector x such that $A \cdot x$ equals the all-one vector.

Many problems from recreational mathematics can be formulated as exact cover problems. In his paper "dancing links" [4] Donald Knuth used a backtracking algorithm to attack some of these problems. Even if the strategy of this backtracking algorithm is quite obvious, there remain many possibilities for speed improvements in the software implementation. Donald Knuth used a simple, but powerful programming trick based on pointer arithmetics which goes back to Hirosi Hitotumatu and Kohei Noshita [3] and which enables a very fast managing of the backtracking steps. The matrix A is stored via doubly linked lists, the elementary operations — called *updates* — are deleting and restoring one entry of the matrix A. Doing this with pointers is extremely fast, because no additional arrays are necessary for storage of the backtrack information.

Nevertheless, after estimating the runtime of his algorithm on the 18 pentominoes problem, Donald Knuth decided not to spend his computing time on this problem [4]:

> "...If that estimate is correct, I could have the result in a few month; but I'd rather try for a new Mersenne prime."

The author implemented a parallel version of the algorithm of Donald Knuth, see also [8], based on PVM [1], and succeeded in enumerating all possible configurations by distributing the work on 16 Pentium computers.

To avoid isomorphic solutions through 180°-rotation and reflections around the horizontal and vertical axes, the problem was divided into two cases:

P_1: Restricting the center of the X-pentomino to the 12 positions $(1,1)$, ..., $(4,1)$, $(1,2)$, ..., $(4,3)$ in the lower left part of the rectangle.

P_2: Restricting the center of the X-pentomino to the left half of the vertical middle line of the rectangle (i. e. the positions $(1, 4)$, $(2, 4)$, $(3, 4)$, $(4, 4)$) and the V-pentomino to the two rotations  and  .

In problem P_1 the resulting Diophantine linear system has 108 equations with 3542 unknowns. This system has $8, 864, 824$ solutions which were found after $3, 743, 551, 091, 075$ updates (a Pentium computer running with 300 MHz performs approximately 4.5 million updates per second). In problem P_2 there are 3422 unknowns and $1, 575, 609$ solutions. The latter computation needed $906, 657, 749, 525$ updates. Altogether, there are

$$10, 440, 433 \text{ solutions after } 4, 650, 208, 840, 600 \text{ updates.}$$

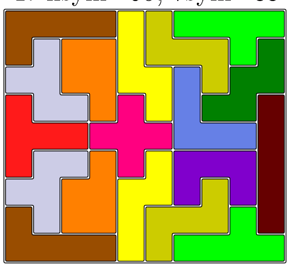
Meanwhile, Patric Östergård [6] confirmed the above number of arrangements, using a completely different method of parallelization of the program of Donald Knuth.

Beside the total number of non-isomorphic arrangements we may ask for the most symmetrical solutions. In [4] the *horizontal symmetry* (hsym) of a configuration is defined as the number of edges between pieces that are also edges between pieces in the left-right reflection of that arrangement. The *vertical symmetry* (vsym) of a configuration is defined similarly, but with respect to top-bottom reflection. Since the number of internal edges of the 9×10 rectangle is 161 and the 18 one-sided pentominoes contain 74 internal non-edges, a perfectly symmetrical arrangement would have a score of 87.

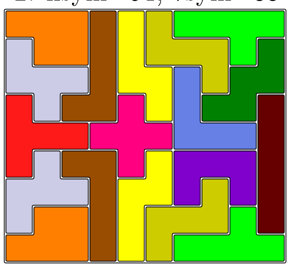
Configuration 1 has the highest (vertical) symmetry: vsym = 83 and hsym = 60. In the arrangements $1 - 7$ only 4 edges are missing to make them perfectly symmetrical. The highest horizontal symmetry is achieved by the arrangements 15, 16 and 17 (hsym = 75 and vsym = 55). The horizontal symmetry of these solutions (and of solution 12) supersedes the horizontal symmetry of the solution printed in “dancing links” [8], which is shown as solution 13. It has horizontal symmetry hsym = 74 and vertical symmetry vsym = 49 and Knuth conjectured it having the highest horizontal symmetry.

Solutions of P_1*:*

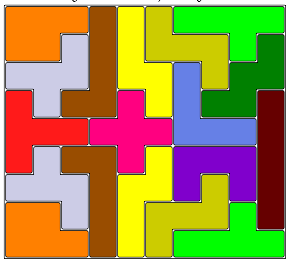
1: hsym=60, vsym=83



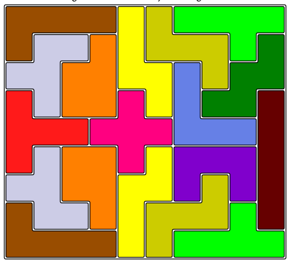
2: hsym=54, vsym=83



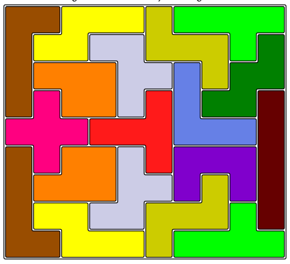
3: hsym=54, vsym=83



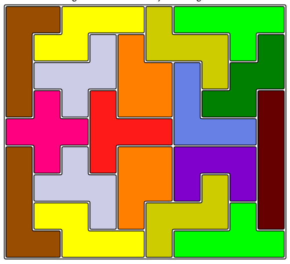
4: hsym=50, vsym=83



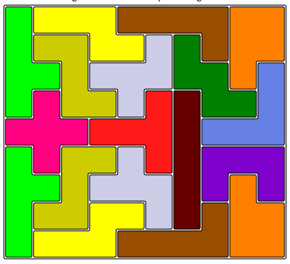
5: hsym=42, vsym=83



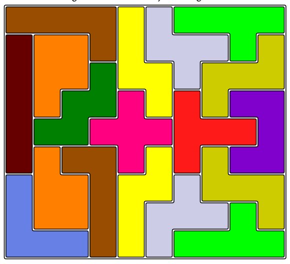
6: hsym=40, vsym=83



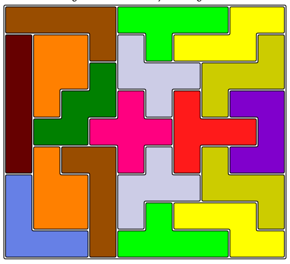
7: hsym=38, vsym=83



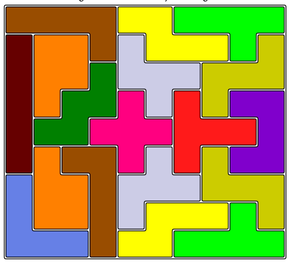
8: hsym=46, vsym=82



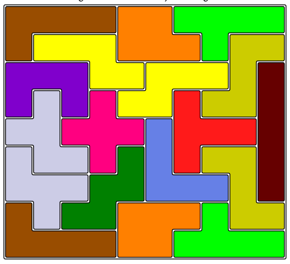
9: hsym=42, vsym=82



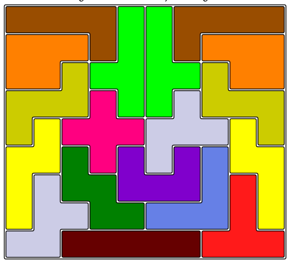
10: hsym=42, vsym=82



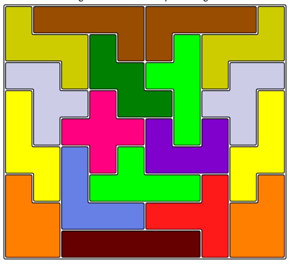
11: hsym=54, vsym=80



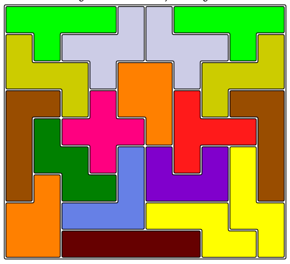
12: hsym=75, vsym=49



13: hsym=74, vsym=49

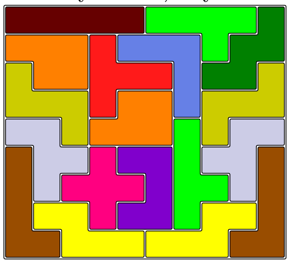


14: hsym=74, vsym=49

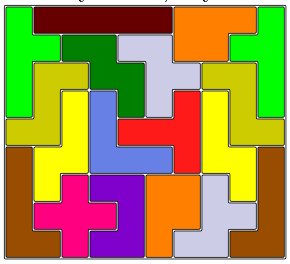


Solutions of P_2*:*

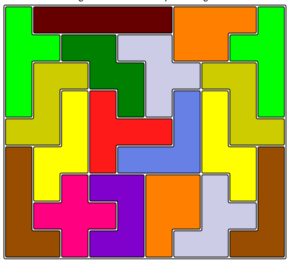
15: hsym=75, vsym=55



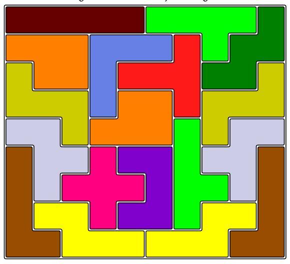
16: hsym=75, vsym=55



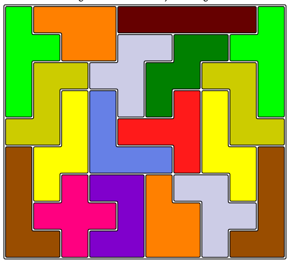
17: hsym=75, vsym=55



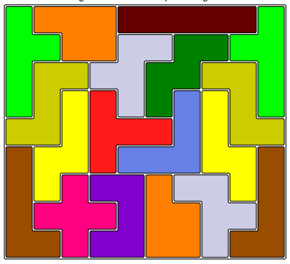
18: hsym=75, vsym=53



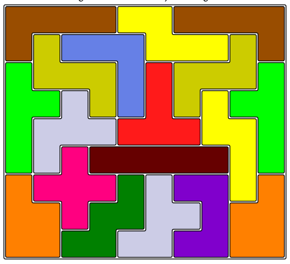
19: hsym=75, vsym=51



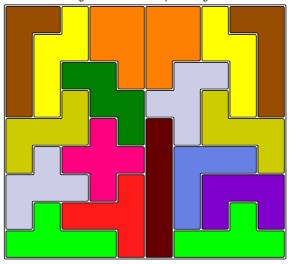
20: hsym=75, vsym=51



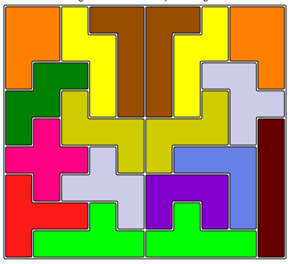
21: hsym=75, vsym=50



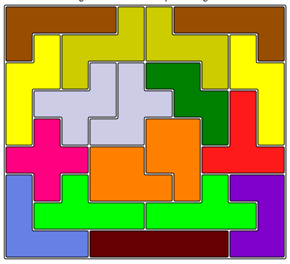
22: hsym=75, vsym=43



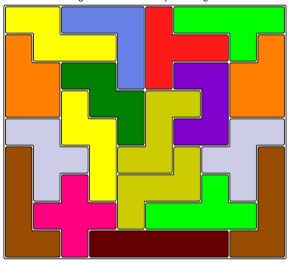
23: hsym=75, vsym=43



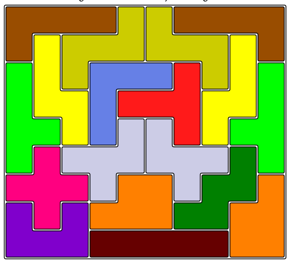
24: hsym=74, vsym=51



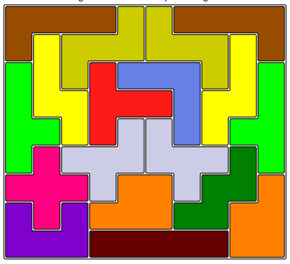
25: hsym=74, vsym=51



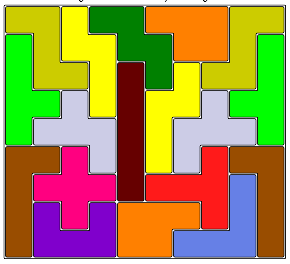
26: hsym=74, vsym=51



27: hsym=74, vsym=51



28: hsym=74, vsym=49



Altogether, there are 7 solutions with vsym = 83, 3 solutions with vsym = 82 and 1 solution with vsym = 80. Furthermore, there exist 10 solutions with hsym = 75 and 8 solutions with hsym = 74.

References

[1] Al Geist, Adam Beguelin, Jack Dongarra, Weicheng Jiang, Robert Manchek, Vaidy Sunderam, *PVM: Parallel Virtual Machine A Users' Guide and Tutorial for Networked Parallel Computing*, MIT Press, Scientific and Engineering Computation, Janusz Kowalik, Editor, (1994) Massachusetts Institute of Technology. The software is downloadable at http://www.epm.ornl.gov/pvm.

[2] Solomon W. Golomb, *Polyominoes*, second edition, Princeton University Press, Princeton, New Jersey, (1994).

[3] Hirosi Hitotumatu, Kohei Noshita, "A technique for implementing backtrack algorithms and its application," *Information Processing Letters* **8** (1979), 174–175.

[4] Donald E. Knuth, "Dancing Links," preprint, (1999), available from `http://www-cs-staff.Stanford.EDU/~knuth/preprints.html`.

[5] Thomas H. O'Beirne, "Puzzles and Paradoxes 44:": Pentominoes and hexiamonds," *New Scientist* **12** (1961), 316–317.

[6] Patric Östergård, personal communication.

[7] Dana S. Scott, "Programming a combinatorial puzzle," Technical Report No. 1 (Princeton, New Jersey: Princeton University Department of Electrical Engineering, 10 June 1958).

[8] Alfred Wassermann, "Covering the Aztec Diamond with One-sided Tetrasticks", (1999), to appear in the *Bulletin of the Institute of Combinatorics and its Applications*. An extended version is available from `http://did.mat.uni-bayreuth.de/wassermann/`.