GENETIC ALGORITHMS FOR A PARTICULAR COVERING PROBLEM

Illyés, László, Department of Mathematics and Informatics
Sapientia University – Miercurea-Ciuc, Romania

Abstract:

Let us consider "the optimal cover of an NxN sided "chessboard" with squares having different dimensions (the length of each square side being a multiple of a given unit)". Here the optimal cover means that the number of uncovered elementary squares has to be minimum.

The problem is complex, because it has many possible, acceptable solutions. I propose genetic algorithms to find theoretical and practical solutions.

We discuss about the 64x64 "chessboard".

Keywords: optimal cover, genetic algorithms, degree of freedom

This is a restriction-free cutting technology for a specific Two-dimensional Cutting Stock Problem with **one** Stock

Some specific notations – <u>the "chessboard" and the pieces are squares</u>
l_i - length of sides of the i^{th} square
N – length of side of the "chessboard"
S=NxN – The whole area of the "chessboard"
s_i – The area of square i – in our case $l_i^2=i^2$
w_p – waste proportion of material

We have two problems to solve:

1. Selection of pieces – The Area Problem
2. Putting on them on the pattern in a not overlaping form – The Covering Problem

The mathematical model for the problem 1 is:

$$\text{Max} \quad \sum_{i=1}^{m} s_i * x_i$$

$$\sum_{i=1}^{m} s_i * x_i \leq S$$

$x_i=\{0,1\}$ - $x_i=1$ when we use the **i**th square otherwise 0

The mathematical model for the problem 2 is the same with a new constraint:
- not overlaping form
We assume, that we try to found a non-trivial solution, better than that solution in that we put the (N-1) lenght sided square and the 1 square on board:

$(N-1)^2+1^2 \leq \sum_{i=1}^{m} s_i * x_i \leq N^2$ and in other terms

the waste proportion $w_p \leq \frac{2N-2}{N^2}$, in this case w_p=3,076% or 126 unit dimension squares.

The main thing in this kind of problems are the following:

A. We have to restrict the area of the possible solutions.
B. We have to calculate the complexity or upper bound value of lexicographical enumeration in this problem.

We try to solve The Area Problem with a genetic algorithm, then use the set resulted and resolve The Covering Problem with an another genetic algorithm.
We can consider the area-solution the **genotype** of the problem, and the cover problem the **phenotype**.

A. The Area Problem – hypothetical and practical aspects

We have to calculate the maximum number of squares that we can put on the „chessboard".

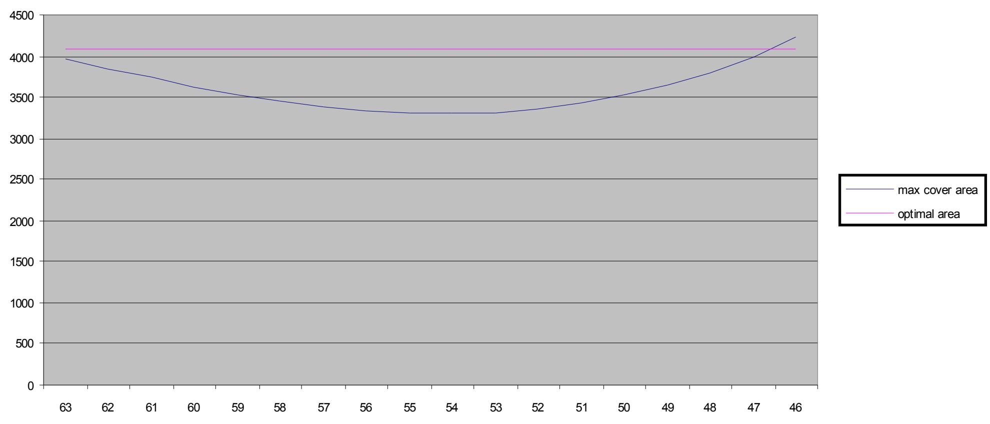
When we have a chessboard with 64x64 we observe that $1^2+2^2+...+21^2+28^2=4095$ <4096 (4096=64x64) and assume that <u>22 different squares are the maximum</u>, that we can put on the table in this case.
Considering, that only those squares can be situated side by side that satisfy the relation: $l_1+l_2<=N$ (where l_1 and l_2 are the sides of the squares), and calculating with these squares we have the next relations referring to their area:

$58^2+\sum_{i=1}^{6} i^2 = 3455<4096$ $\quad$ $54^2+\sum_{i=1}^{10} i^2 = 3301$ $\quad$ $53^2+\sum_{i=1}^{11} i^2 = 3315$

$47^2+\sum_{i=1}^{17} i^2 = 3994$ $\quad$ $46^2+\sum_{i=1}^{18} i^2 = 4225>4096$

$f(p)=p^2+\sum_{i=1}^{64-p} i^2 = p^2+\dfrac{(64-p).(63-p).(129-2p)}{6}$ where p is integer. [MZ04]

Possible covering area



There are obvious tendencys: the area of possible squares, that we can use decrease until we put the 54 unit length square on the chessboard, then the sum of area increases, and the sum of possible area overturn the 4096 value at the 46 unit length square, that means that the maximum square we can put on the board is 46.

We construct a **haploid** chromosome, we have a chromosome with 45 bit, where 1 represents the choosed square, and 0 represents the avoided square from the set. In this case we can see from the calculus, that the 46 unit square is not in our solutions.

Taking in consideration, that the sum of length of sides of 2 adiacent squares $lw_1+lw_2<=N$ means, that from 45 to 32 gene we have only one of 1 bit, we can calculate an upper bound value of lexicographical enumeration of the problem:

$2^{19}+2^{20}+2^{21}+...+2^{31}=2^{19}*(1+2+2^2+...+2^{12})=2^{19}*(2^{13}-1)=4.294.443.008$

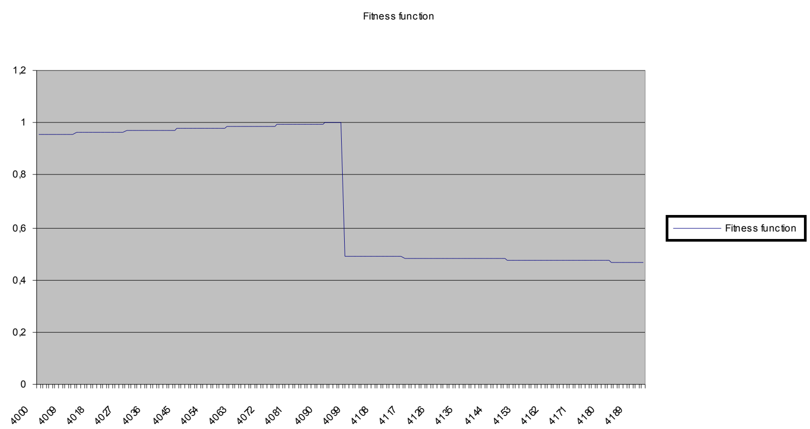
In practice we found few entrys upper than the 42 unit length square (1 piece of 44, 2 of 43).

We generate the chromosome of the first population using **4 randomized values**. The first number is between 0-13, reprezenting the pozition of the 1 in the higher part of the chromosome. We generate then 2 numbers between 0 and $2048=2^{11}$, and one number between 0 and $1024=2^{10}$, we transform decimal numbers in bits, and we have 13+2.11+10=45 bit chromosome.
Numbers generated are pseudo-random, but in time and in a longer cycle, where we generate the generations this is not an inconvenience.

There is no solution, when S>4096, because some parts of some squares are out of the „chessboard". When we perform operation on chromosomes we don't drop those solutions, because some part of the chromosome can be good, and we can use them when we make crossover operations, but we give them a **fewer fitness**, a fewer chance to be a choosed parent.

The fitness function of the chromosomes we write like this:

$$f=\begin{cases} 0,7*\dfrac{4096}{S^2} & \textbf{if } \mathbf{S>4096} \\ \dfrac{S^2}{4096} & \textbf{if } \mathbf{S<=4096} \end{cases}$$



We choose the quadratic function, to control the convergency of the population.

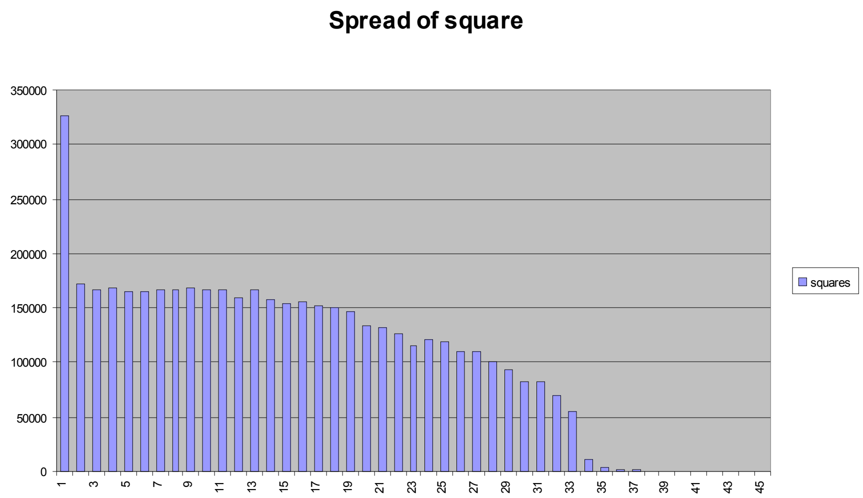
We calculate the average of the fitness function $\boldsymbol{ff}=\frac{\sum_{i=1}^{k} f_i}{k}$, where k is the number of individuals in a population. In the case of the **roulette wheel, fitness based selection**, the individuals of the populations are choosen to be parents according to this proportion: f/ff

Like *De Jong* proposed [AL02], every time, when we choose an individual to be a parent for the next population we decrease this probabilistic proportion with one, to avoid the early convergence of the population. If the proportion is 1, we don't decrease this.

We choose **two-point crossower operator**, and accept both of the children in the next generation. We can perform the operator over 45 bits or over the 32 last significant bit, to avoid the 2 bit of 1 in the significant part of the chromosome. This heuristical approach we discussed before for other reason.
We calculate with 100 individuals in a generation. We choose 200 as the number of generations. The resulted chromosomes converge to a chromosome, or have an oscillation between 2 chromosomes, that have many similar genes. This is the so-called „genetic draft**".**
We can avoid this with using a taboo list.

PRACTICAL RESULTS

1. Histogram from the till now obtained : 337,756 results from the algorithm used above. How many pieces have we from each kind.

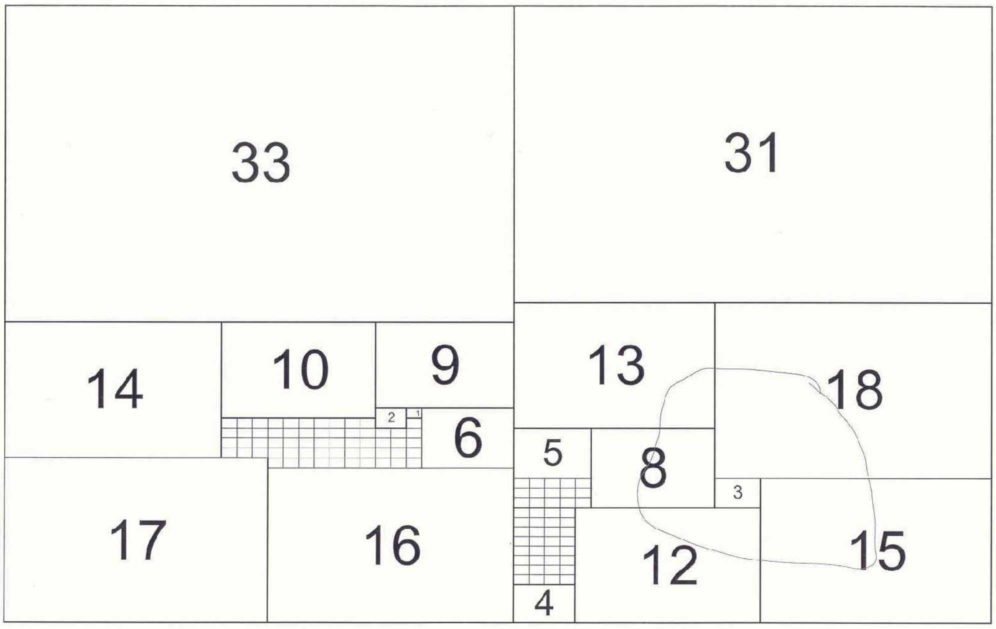


2. Results of analysing the influence of parameters used above
- Modifying the parameter of the fitness function from 0,7 to 1
  - we found that we have the best results with the **1** parameter, that means, that <u>diversity of chromosomes</u> is the most significant factor
- cardinality of population 100 – we don't examine
- cardinality of generations (100-200)
  - if the cardinality of generations are between 150 and 200 the population converge to a chromosome. We found, if we use 100 as the number of generations, the population convergency is not good, but we have only 10-15% fewer possible solution, and this bisect the running time. We use this to decrease running time.
- choosing the location of the crossover operation points –we don't examine

3. We can try to discuss about the adaptation of the parameters

B. The Covering Problem: -hypothetical and practical aspects

1. The board is symmetrical, we put the greatest square on the edge of the table. Why on the edge? If we move the square with 1 unit , we observe that *k*x1 square will be empty on the board, we can cover only one of those with the unit square we consider, that placing the greatest square elsewhere than in a corner decreases the **degree of freedom** for the other pieces.
2. Our intuition is, that sum of the length of side of the 2 biggest square let it be 64, that means that a compact part of the board, almost 50% will be covered. There is a human approach with 33+31=64 beginning.
3. A **structure – pattern** is coming on, something similar with the golden section of Fibonacci. The idea is shown below where I found a 107 uncover unit square result. I found the waste in percent 107/4096=2.61%, a good result in real applications.



I propose a data structure and a genome for the covering problem. The main hypothesis of the idea are from [BOL70], [FAB01], [FAB02] and [FAIL04] and was some aspects in [VKBK04].
The problem has specific aspects regarding the symmetry of the "chessboard" and squares.
Regarding to this kind of problems, where we know there are good solutions, we can formulate easier the problem and we can create better structure.

The covering problem with some generalizations will be discussed in next papers.

Generalization aspects:

A. Also with unit dimensional squares:
1. Consider the board **nxn** unit dimensional
2. Let us consider the board **nxm**-unit dimensional

B. Let the pieces be **rectangles**
1. proportion of their sides equal with 2
2. proportion of their sides equal with k

C. Let the pieces be triangles
1. With equal sides
2. With right angle and 2 equal sides

D. Let the pieces be circles, half-circle, or ¼ circle

Conclusions:
If we use mixed strategy we can combine good parts of some different strategy plans.
I find the **best chromosome length** for this solution 45.
I find a **better solution** of waste (smaller waste) than the N-1 and 1 sided squares on the board: our solution of 2,61% is better than 3,076%.
I find the **upper bound of lexicographical enumeration** using the 2 biggest square constraint.
I find, that in this case the **diversity of the chromosomes** is the most significant factor to obtain solutions.
I introduce the parallelism between a covering problem and genetics:

area-solution	**genotype**
covering problem solution	**phenotype**

Referencies:

[BOL70] Boldur Gh.
Unele aspecte ale problemei croirii elementelor dreptunghiulare,
SNCCECE (Studii de cercetari Economice si de cibernetica economica) Nr 6, 1970

[FAB01] Fabian,Csaba
Two-dimensional Cutting Stock Problems Using Evolutionary Algorithms,
IE-2001 Proceedings, Inforec, Bucharest, 2001, pag.513-517

[FAB02] Fabian, Csaba
Generalized Simple and Crossover Mutations for Evolutionary Algorithms,
International Conference on Economic Cybernetics Bucharest, 2002

[FAIL04] Fabian, Cs. B., Illyes, L.
Solving Fuzzy Cutting Stock Problems Using Fuzzified Genetic Algorithms
1'st ESICUP Meeting –Lutherstadt-Wittenberg-Germany 18-20 march 2004

[VKBK04] Vyatkina, K., Bukhalova, V.
On separating Boxes with Right-Angled Cuts
1'st ESICUP Meeting –Lutherstadt-Wittenberg-Germany 18-20 march 2004

[MZ04] Makó,Z. – private conversation 21 april 2004

[AL02] Álmos Attila et.al. Genetikus Algoritmusok Typotex 2002