IE 303

Modeling and Methods in Optimization

Project 1

Yiğit Ali Karadoğan

21902218

1.0 PROBLEM DEFINITION

According to the rules of the problem, Ali Baba has 7x7 chessboard and he needs to place gold and bronze coins on it. According to the deal between Ali Baba and the thieves, any 3x3 square of unit squares of the chessboard must contain an equal number of gold and bronze coins after all the coins have been placed. After the placement, Ali Baba can take away the gold coins that he have been placed to the board.

According to some inferences from the question, Ali Baba definitely thinks the ways of maximum amount of placement of gold coins in the border of the rules of the game. In this project, we will help Ali Baba to achieve this goal.

2.0 ANSWERS TO THE PROBLEM

To begin with, Ali Baba has the following chessboard in this game.

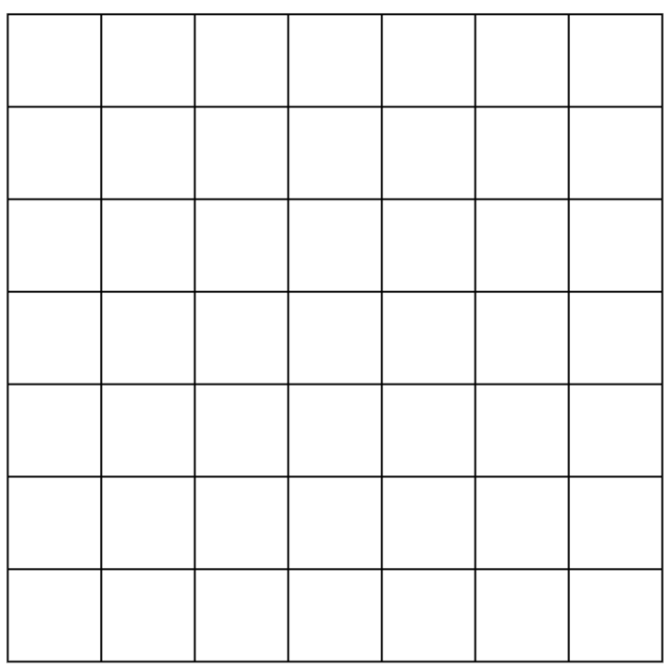

Fig. 1. 7x7 square chessboard given to Ali Baba

According to the rules of the game, he needs to properly place gold and silver coins given to him. Some examples are given.

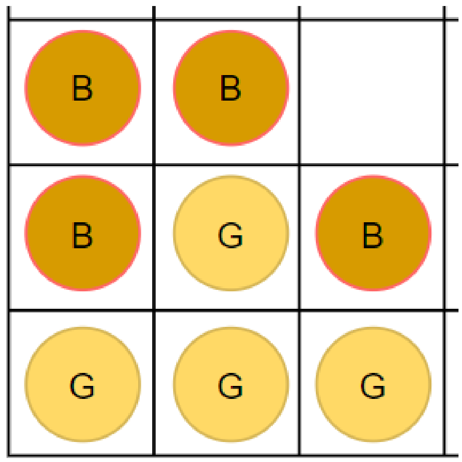

Fig. 2. Example 1 of a proper placement

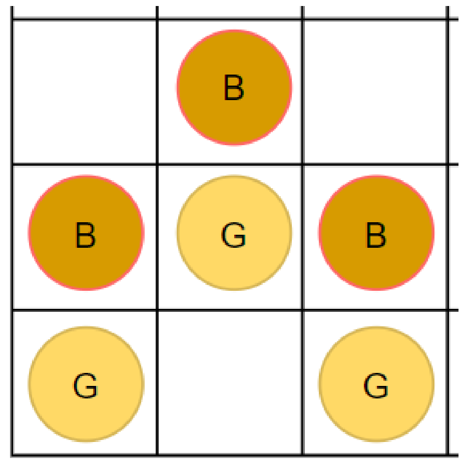

Fig. 3. Example 2 of a proper placement

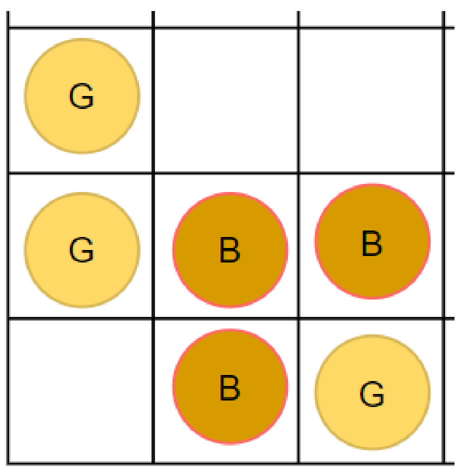

Fig. 4. Example 3 of a proper placement

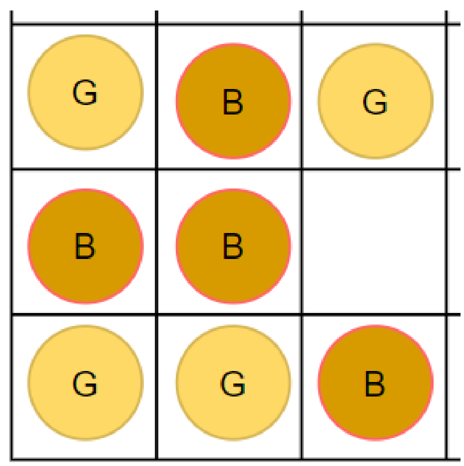

Fig. 5. Example 4 of a proper placement

2.1 IP-MODEL TO THE PROBLEM

I defined my binary decision variables according to placements of respective square, either gold or bronze (maybe neither of them). Then, my objective function is simply maximizing the gold placements to the 7x7 board. I ensured it by writing the following constraints: Each 3x3 square at the board (total 25 3x3 squares) has equal amount of bronze and gold coins. Also, I pushed my program to place at most one coin at single square in order to prevent double placement.

<u>*Decision Variables*:</u>

$$g_{(i,j)} = \left\{ \begin{array}{ll} 1 & \text{if there is a gold coin at square } (i,j) \\ 0 & o.w. \end{array} \right\}$$

$$b_{(i,j)} = \left\{ \begin{array}{ll} 1 & \text{if there is a bronze coin at square } (i,j) \\ 0 & o.w. \end{array} \right\}$$

<u>*Model*:</u>

$$\max \quad \sum_{i=1}^{7} \sum_{j=1}^{7} g_{(i,j)}$$

$s.t.$

$$\sum_{i=1}^{3} \sum_{j=1}^{3} g_{(i,j)} = \sum_{i=1}^{3} \sum_{j=1}^{3} b_{(i,j)}$$

$$\sum_{i=2}^{4} \sum_{j=1}^{3} g_{(i,j)} = \sum_{i=2}^{4} \sum_{j=1}^{3} b_{(i,j)}$$

$$\sum_{i=3}^{5} \sum_{j=1}^{3} g_{(i,j)} = \sum_{i=3}^{5} \sum_{j=1}^{3} b_{(i,j)}$$

$$\sum_{i=4}^{6} \sum_{j=1}^{3} g_{(i,j)} = \sum_{i=4}^{6} \sum_{j=1}^{3} b_{(i,j)}$$

$$\sum_{i=5}^{7} \sum_{j=1}^{3} g_{(i,j)} = \sum_{i=5}^{7} \sum_{j=1}^{3} b_{(i,j)}$$

$$\sum_{i=1}^{3}\sum_{j=2}^{4} g_{(i,j)} = \sum_{i=1}^{3}\sum_{j=2}^{4} b_{(i,j)}$$

$$\sum_{i=2}^{4}\sum_{j=2}^{4} g_{(i,j)} = \sum_{i=2}^{4}\sum_{j=2}^{4} b_{(i,j)}$$

$$\sum_{i=3}^{5}\sum_{j=2}^{4} g_{(i,j)} = \sum_{i=3}^{5}\sum_{j=2}^{4} b_{(i,j)}$$

$$\sum_{i=4}^{6}\sum_{j=2}^{4} g_{(i,j)} = \sum_{i=4}^{6}\sum_{j=2}^{4} b_{(i,j)}$$

$$\sum_{i=5}^{7}\sum_{j=2}^{4} g_{(i,j)} = \sum_{i=5}^{7}\sum_{j=2}^{4} b_{(i,j)}$$

$$\sum_{i=1}^{3}\sum_{j=3}^{5} g_{(i,j)} = \sum_{i=1}^{3}\sum_{j=3}^{5} b_{(i,j)}$$

$$\sum_{i=2}^{4}\sum_{j=3}^{5} g_{(i,j)} = \sum_{i=2}^{4}\sum_{j=3}^{5} b_{(i,j)}$$

$$\sum_{i=3}^{5}\sum_{j=3}^{5} g_{(i,j)} = \sum_{i=3}^{5}\sum_{j=3}^{5} b_{(i,j)}$$

$$\sum_{i=4}^{6}\sum_{j=3}^{5} g_{(i,j)} = \sum_{i=4}^{6}\sum_{j=3}^{5} b_{(i,j)}$$

$$\sum_{i=5}^{7}\sum_{j=3}^{5} g_{(i,j)} = \sum_{i=5}^{7}\sum_{j=3}^{5} b_{(i,j)}$$

$$\sum_{i=1}^{3}\sum_{j=4}^{6} g_{(i,j)} = \sum_{i=1}^{3}\sum_{j=4}^{6} b_{(i,j)}$$

$$\sum_{i=2}^{4}\sum_{j=4}^{6} g_{(i,j)} = \sum_{i=2}^{4}\sum_{j=4}^{6} b_{(i,j)}$$

$$\sum_{i=3}^{5}\sum_{j=4}^{6} g_{(i,j)} = \sum_{i=3}^{5}\sum_{j=4}^{6} b_{(i,j)}$$

$$\sum_{i=4}^{6}\sum_{j=4}^{6} g_{(i,j)} = \sum_{i=4}^{6}\sum_{j=4}^{6} b_{(i,j)}$$

$$\sum_{i=5}^{7} \sum_{j=4}^{6} g_{(i,j)} = \sum_{i=5}^{7} \sum_{j=4}^{6} b_{(i,j)}$$

$$\sum_{i=1}^{3} \sum_{j=5}^{7} g_{(i,j)} = \sum_{i=1}^{3} \sum_{j=5}^{7} b_{(i,j)}$$

$$\sum_{i=2}^{4} \sum_{j=5}^{7} g_{(i,j)} = \sum_{i=2}^{4} \sum_{j=5}^{7} b_{(i,j)}$$

$$\sum_{i=3}^{5} \sum_{j=5}^{7} g_{(i,j)} = \sum_{i=3}^{5} \sum_{j=5}^{7} b_{(i,j)}$$

$$\sum_{i=4}^{6} \sum_{j=5}^{7} g_{(i,j)} = \sum_{i=4}^{6} \sum_{j=5}^{7} b_{(i,j)}$$

$$\sum_{i=5}^{7} \sum_{j=5}^{7} g_{(i,j)} = \sum_{i=5}^{7} \sum_{j=5}^{7} b_{(i,j)}$$

$$g_{(i,j)} + b_{(i,j)} \leq 1 \qquad \forall i = 1,2,\cdots,7 \quad \forall j = 1,2,\cdots,7$$

$$g_{(i,j)}, b_{(i,j)} \,\epsilon\, \{0,1\} \qquad \forall i = 1,2,\cdots,7 \quad \forall j = 1,2,\cdots,7$$

2.2 SOLVING MODEL BY USING XPRESS-MP

Here, you can see some parts of the XPress-MP code of the mathematical model:

```
declarations
H = 1..7
V = 1..7
g:array(H,V) of mpvar
b:array(H,V) of mpvar

H1 = 1..3
H2 = 2..4
H3 = 3..5
H4 = 4..6
H5 = 5..7
V1 = 1..3
V2 = 2..4
V3 = 3..5
V4 = 4..6
V5 = 5..7
end-declarations

forall (i in H, j in V) g(i,j) is_binary
forall (i in H, j in V) b(i,j) is_binary

```

Fig. 6. Declarations part of the code

```
! Objective Function
Maxim := sum(i in H,j in V)(g(i,j))
```

Fig. 7. Objective Function of the model

```
! Each 3x3 square has equal amount of gold and bronze coins
sum(i in H1,j in V1)g(i,j) = sum(i in H1, j in V1)b(i,j)
sum(i in H2,j in V1)g(i,j) = sum(i in H2, j in V1)b(i,j)
sum(i in H3,j in V1)g(i,j) = sum(i in H3, j in V1)b(i,j)
sum(i in H4,j in V1)g(i,j) = sum(i in H4, j in V1)b(i,j)
sum(i in H5,j in V1)g(i,j) = sum(i in H5, j in V1)b(i,j)

sum(i in H1,j in V2)g(i,j) = sum(i in H1, j in V2)b(i,j)
sum(i in H2,j in V2)g(i,j) = sum(i in H2, j in V2)b(i,j)
sum(i in H3,j in V2)g(i,j) = sum(i in H3, j in V2)b(i,j)
sum(i in H4,j in V2)g(i,j) = sum(i in H4, j in V2)b(i,j)
sum(i in H5,j in V2)g(i,j) = sum(i in H5, j in V2)b(i,j)

sum(i in H1,j in V3)g(i,j) = sum(i in H1, j in V3)b(i,j)
sum(i in H2,j in V3)g(i,j) = sum(i in H2, j in V3)b(i,j)
sum(i in H3,j in V3)g(i,j) = sum(i in H3, j in V3)b(i,j)
sum(i in H4,j in V3)g(i,j) = sum(i in H4, j in V3)b(i,j)
sum(i in H5,j in V3)g(i,j) = sum(i in H5, j in V3)b(i,j)

sum(i in H1,j in V4)g(i,j) = sum(i in H1, j in V4)b(i,j)
sum(i in H2,j in V4)g(i,j) = sum(i in H2, j in V4)b(i,j)
sum(i in H3,j in V4)g(i,j) = sum(i in H3, j in V4)b(i,j)
sum(i in H4,j in V4)g(i,j) = sum(i in H4, j in V4)b(i,j)
sum(i in H5,j in V4)g(i,j) = sum(i in H5, j in V4)b(i,j)

sum(i in H1,j in V5)g(i,j) = sum(i in H1, j in V5)b(i,j)
sum(i in H2,j in V5)g(i,j) = sum(i in H2, j in V5)b(i,j)
sum(i in H3,j in V5)g(i,j) = sum(i in H3, j in V5)b(i,j)
sum(i in H4,j in V5)g(i,j) = sum(i in H4, j in V5)b(i,j)
sum(i in H5,j in V5)g(i,j) = sum(i in H5, j in V5)b(i,j)
```

Fig. 8. Gold and Bronze amounts in 3x3 are equal

```
! Each 1x1 square cannot have both bronze and gold coins
forall (i in H, j in V)
g(i,j) + b(i,j) <= 1

maximize(Maxim)
```

Fig. 9. 1x1 cannot contain more than a coin

I want to be sure about the correct answer. Therefore, I placed coins according to the solution of the problem. Here, you can see all the coins placed in the squares.

```
forall(i in H, j in V)
    writeln("g(",i,",",j,")=",getsol(g(i,j)),"   ","b(",i,",",j,")=",getsol(b(i,j)))
```

Fig. 10. Getting solutions of each square

```
Running model
g(1,1)=1   b(1,1)=0
g(1,2)=1   b(1,2)=0
g(1,3)=1   b(1,3)=0
g(1,4)=1   b(1,4)=0
g(1,5)=1   b(1,5)=0
g(1,6)=1   b(1,6)=0
g(1,7)=1   b(1,7)=0
g(2,1)=0   b(2,1)=0
g(2,2)=0   b(2,2)=1
g(2,3)=0   b(2,3)=1
g(2,4)=0   b(2,4)=0
g(2,5)=0   b(2,5)=1
g(2,6)=0   b(2,6)=1
g(2,7)=1   b(2,7)=0
g(3,1)=1   b(3,1)=0
g(3,2)=0   b(3,2)=1
g(3,3)=0   b(3,3)=1
g(3,4)=1   b(3,4)=0
g(3,5)=0   b(3,5)=1
g(3,6)=0   b(3,6)=1
g(3,7)=0   b(3,7)=0
g(4,1)=1   b(4,1)=0
g(4,2)=1   b(4,2)=0
g(4,3)=1   b(4,3)=0
g(4,4)=1   b(4,4)=0
g(4,5)=1   b(4,5)=0
g(4,6)=1   b(4,6)=0
g(4,7)=1   b(4,7)=0
```

```
g(5,1)=0   b(5,1)=0
g(5,2)=0   b(5,2)=1
g(5,3)=0   b(5,3)=1
g(5,4)=0   b(5,4)=0
g(5,5)=0   b(5,5)=1
g(5,6)=0   b(5,6)=1
g(5,7)=1   b(5,7)=0
g(6,1)=1   b(6,1)=0
g(6,2)=0   b(6,2)=1
g(6,3)=0   b(6,3)=1
g(6,4)=1   b(6,4)=0
g(6,5)=0   b(6,5)=1
g(6,6)=0   b(6,6)=1
g(6,7)=0   b(6,7)=0
g(7,1)=1   b(7,1)=0
g(7,2)=1   b(7,2)=0
g(7,3)=1   b(7,3)=0
g(7,4)=1   b(7,4)=0
g(7,5)=1   b(7,5)=0
g(7,6)=1   b(7,6)=0
g(7,7)=1   b(7,7)=0
Begin running model
27
End running model
```

Fig. 11. 12. Solutions of each square

According to the model, Ali Baba can take **maximum 27 gold coins** from the game of thieves. Followingly, you can see the representation of the final form of 7x7 chessboard.

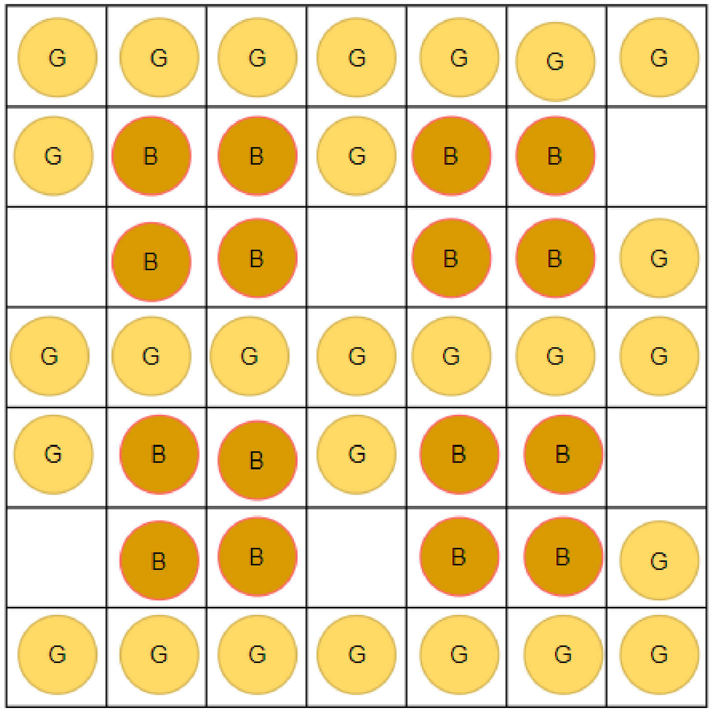

Fig. 13. Final form of 7x7 chessboard after placements

2.3 ANSWER OF 90000x90000 CHESSBOARD

If we observe the final form of placements clearly (fig. 13), we can see that there are **both horizontally and vertically patterns.** Each 3 steps of both horizontally and vertically, this pattern repeats itself. You can see the patterns in the following figures:

HORIZONTAL PATTERNS:

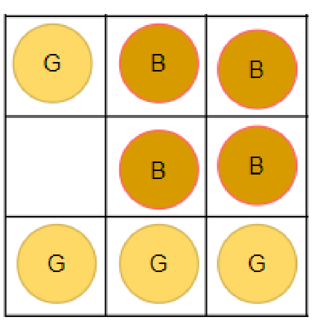

Fig. 14. H 1/3

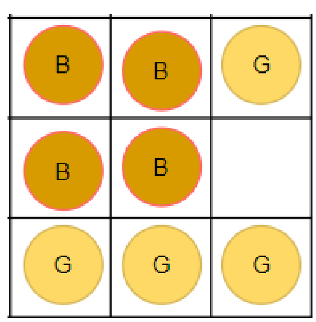

Fig. 15. H 2/3

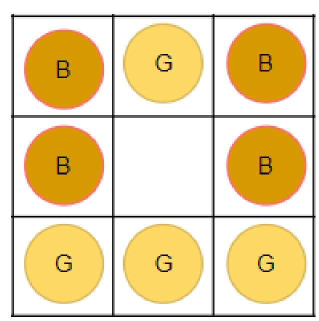

Fig. 16. H 3/3

VERTICAL PATTERNS:

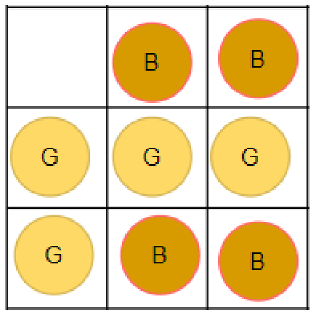

Fig. 19. V 3/3

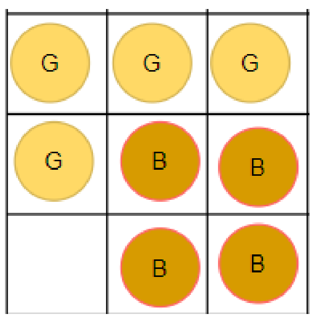

Fig. 18. V 2/3

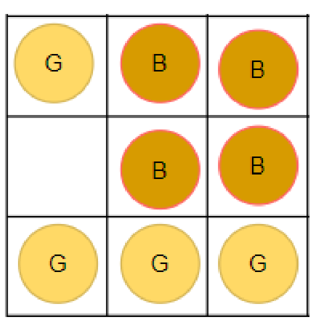

Fig. 17. V 1/3

These horizontal and vertical patterns repeat themselves in every three steps. Therefore, the optimal solution is nothing to do with the size of the chessboard (but at least 5x5). If thieves gave 5x5 or 100x100, or even 90000x90000 to Ali Baba, this pattern would repeat itself both horizontally and vertically in all of these boards mentioned. Hence, it will give an optimal solution in each case. These patterns are shown in the final form demonstration on the following page.

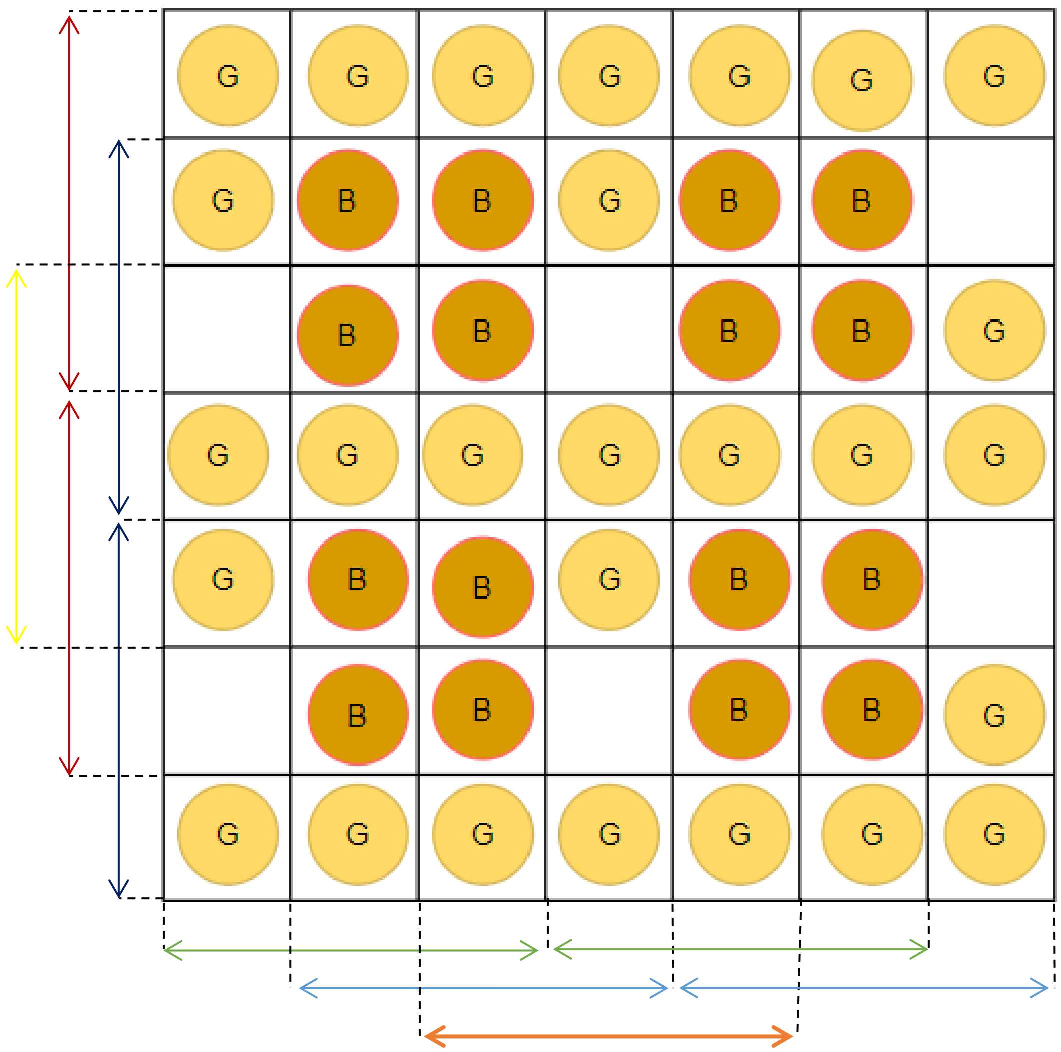

Fig. 20. Demonstration of Patterns at the Final Form