

# #5 Nonogram - Formulation

Authors: Éder Pinheiro, Aster Santana

August, 2021

This is the MIP formulation of the puzzle. Statement and solution implementation of all puzzles are available from the main page of the [Fun Puzzles](#) project, which is maintained by [Mip Master](#).

## Formulation

The purpose of the game is to paint some of the cells so that in every row and every column the painted cells form distinct strings with the lengths presented in correct order by the numbers on the left of the rows and on top of the columns.



## Input Data

- rows:

$I = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]$

- columns:

$J = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]$

- Row strings:

```
RS = {  
  1: {1: 3, 2: 3},  
  2: {1: 4, 2: 4},  
  3: {1: 2, 2: 4, 3: 2},  
  4: {1: 2, 2: 2, 3: 2},  
  5: {1: 2, 2: 1, 3: 2, 4: 2},  
  6: {1: 2, 2: 2, 3: 1, 4: 2},  
  7: {1: 2, 2: 2, 3: 2},  
  8: {1: 2, 2: 4, 3: 2},  
  9: {1: 4, 2: 4},  
 10: {1: 3, 2: 3}  
}
```

- Column strings:

```
CS = {  
  1: {1: 10},  
  2: {1: 10},  
  3: {1: 2, 2: 2},  
  4: {1: 2, 2: 2, 3: 2},  
  5: {1: 2, 2: 3},  
  6: {1: 3, 2: 2},  
  7: {1: 2, 2: 2, 3: 2},  
  8: {1: 2, 2: 2},  
  9: {1: 10},  
 10: {1: 10}  
}
```

Notice that  $RS[1][1]$  returns 4, which is the length of the first string of the first row. As another example,  $CS[5][2]$  returns 3, which is the length of the third string of the fifth column. We will use this notation in the formulation below.

## Decision Variables

The obvious decision we need to make is which cell get colored to form strings. For that, we define the following set of decision variables:

- $x_{ij}$  equals 1 if cell  $(i, j)$  gets colored, 0 otherwise.

With these variables only, it's hard to model the number of strings in each row/column, its length and position. So we define two more set of variables:

- $y_{ijk}$  equals 1 is the  $k$ -th string of row  $i$  begins in column  $j$ , 0 otherwise.
- $z_{ijk}$  equals 1 is the  $k$ -th string of column  $j$  begins in row  $i$ , 0 otherwise.

## Constraints

Let's start with the constraints that guarantee that every string begins exactly once. This may be obvious but must be modeled.

- *Each row string begins in exactly one column:*

$$\sum_j y_{ijk} = 1 \quad \forall i, k \in RS[i].$$

- *Each column string begins in exactly one row:*

$$\sum_i z_{ijk} = 1 \quad \forall j, k \in CS[j].$$

Next, we must ensure that every strings has the length it's supposed to have. For example, we know that the first string of the first row must have length 4. So if it starts at column 2, then cells  $(1, 2)$ ,  $(1, 3)$ ,  $(1, 4)$ , and  $(1, 5)$  must be filled. In terms of decision variables, this means that, if  $y_{121} = 1$ , then  $x_{12}, x_{13}, x_{14}, x_{15} = 1$ . This implication connects the  $y$  variables with the  $x$  variables and can be formulated as follows:

$$\begin{aligned} y_{121} &\leq x_{12} \\ y_{121} &\leq x_{13} \\ y_{121} &\leq x_{14} \\ y_{121} &\leq x_{15}. \end{aligned}$$

Alternatively, this could be formulated as a single constraint:

$$4y_{121} \leq x_{12} + x_{13} + x_{14} + x_{15}.$$

Although this second option is more compact and looks nicer, this is usually less computationally efficient because it leads to a weaker formulation, i.e., it's LP relaxation contains the LP relaxation of the first disaggregated formulation.

Next, we write this constrain in its generic form for rows and columns.

- *Row strings length:*

$$y_{ijk} \leq x_{i,j+t}, \quad \forall i, j, k, \forall t < RS[i][k].$$

- *Column strings length:*

$$z_{ijk} \leq x_{i+t,j}, \quad \forall i, j, k, \forall t < CS[j][k].$$

We are not done yet. Suppose the first string of the first row begins at column 2, like in our example above. Now, what happens if the second string begins at column 6? In this case, the first and second string would become a single string that would be 5 cells long. There is nothing in the formulation so far that would prevent that. So we need to ensure that there will exist at least one empty cell between any two subsequent strings. Here is one way we can accomplish this in our example. If the first string of the first row begins at column 2, i.e.,  $y_{121} = 1$ , then we want  $y_{1j2} = 0$  for all  $j \leq 2 + 4 + 1 = 6$ , where 2 is the starting column, 4 is the length of the string, and 1 is an empty cell to keep the first string disconnected from the second string. Notice that this also ensure that the second string will only come after the first string (not before, nor overlapping, nor connect, only after). This implication can be formulated as follows:

$$y_{1j2} \leq 1 - y_{121} \quad \forall j \leq 6.$$

- *Row strings disjunction and precedence:*

$$y_{i,j',k+1} \leq 1 - y_{ijk} \quad \forall i, j, k, \forall j' \leq j + RS[i][k] + 1.$$

- *Column strings disjunction and precedence:*

$$z_{i',j,k+1} \leq 1 - z_{ijk} \quad \forall i, j, k, \forall i' \leq i + CS[j][k] + 1.$$

## Objective Function

The objective function doesn't play any significant role in the formulation of most puzzles, because we are usually interested in finding a feasible solution only. This is the case with this puzzle as well, there is nothing to be optimized here. However, the objective functions plays a critical role in our formulation.

We set the objective to minimize the sum of the  $x$  variables. This is to guarantee that only cells composing a defined string will get filled. If the objective was something different, then the solution could have cells filled that don't belong to any string. For example, there is no constraint ensuring that the cell  $(1, 1)$  will remain empty if the first string of the first row starts in the second column.

$$\min \sum_{i,j} x_{ij}.$$

## Final Formulation

$$\begin{aligned}
& \min && \sum_{i,j} x_{ij} \\
& \text{s.t.} && \sum_j y_{ijk} = 1 \quad \forall i, k \in RS[i] \\
& && \sum_i z_{ijk} = 1 \quad \forall j, k \in CS[i] \\
& && y_{ijk} \leq x_{i,j+t}, \quad \forall i, j, k, \forall t < RS[i][k] \\
& && z_{ijk} \leq x_{i+t,j}, \quad \forall i, j, k, \forall t < CS[j][k] \\
& && y_{ijk} \leq 1 - y_{i,j',k+1} \quad \forall i, j, k, \forall j' \leq j + RS[i][k] + 1 \\
& && z_{ijk} \leq 1 - z_{i',j,k+1} \quad \forall i, j, k, \forall i' \leq i + CS[j][k] + 1 \\
& && x_{ij}, y_{ijk}, z_{ijk} \in \{0, 1\}, \quad \forall i, j, k.
\end{aligned}$$