Pizza model formulation

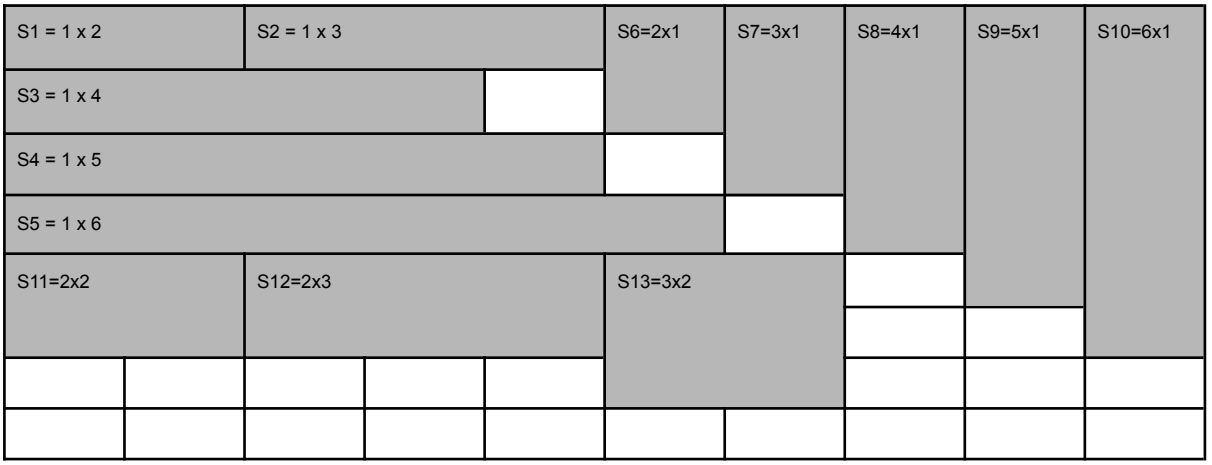
Problem statement can be accessed:
https://github.com/mmarouen/hascode/tree/master/pizza

Definitions:

r=0…R row index
c=0…C column index
Ingredients(r,c)=1 for mushroom, 0 for tomatoes
L: minimum number of each ingredient in each slice
H: maximum number of pizza cells in each slice

Approach

Given a max slice size **H** and a minimum size **2*L**, we can define the space of possible shapes S that satisfy a surface <= **H** and surface >= **2*L**.
Example, H = 6, L = 1 (demo file), there are in total **13** shapes (shown below):



In order to slice the pizza, some or all of the shapes need to be used at least once.
Lets denote by:
Shapes[s] = size of shape “s”

Decision variables

$occupancy_{r,c,s} = \{0, 1\}$: whether cell (r,c) is occupied by shape “s”

$x_{r,c,s} = \{0, 1\}$Whether cell (r,c) belongs to shape s

Objective function

$$maximize(\sum_{r,c} occ_{r,c})$$

Constraints

C1: Slice surface should not exceed total pizza size

Definition of the occupancy

C2: A cell belongs to one shape at most

$$\forall r, c \sum_{s} occupancy_{r,c,s} \leq 1$$

C3: For each slice, only the top left cell is active

$\forall r, c,\ x_{r,c,s} = 1 \rightarrow x_{r+i,c+j,s} = 0,\ occupancy_{r,c,s} = 1$i,j in shape s

$\forall r, c,\ x_{r,c,s} = 1 \rightarrow x_{r-i,c-j,s} = 0$: remove overlap possibility

C4: Border conditions

$\forall s, c, x_{R-r,c} = 0 for\ r < shape\ width$

$\forall s, r, x_{r,C-c} = 0 for\ c < shape\ height$

C5: Each slice must contain at least L cells from each ingredient

$$\forall r, c, s\ x_{r,c,s} = 1 \rightarrow \sum_{shape} Ingredients_{r+i,\ c+j,s} \geq L$$

$$\forall r, c, s\ x_{r,c,s} = 1 \rightarrow (shapes_{s} - \sum_{shape} Ingredients_{r+i,\ c+j,s}) \geq L$$