

Solving the lights game

This project aims at solving the carousel lights game (reference needed, Fede). The game is as follows.

A person goes to a carousel operator at the fair. The carousel operator tells him that anyone who can guess what happens to the 5 lights when he turns on all 5 switches gets free rides for life. The 5 switches do the following:

1. Turns on red when yellow is on
2. Red and green are never on together
3. Blue and green can only be on and off together
4. Blue and purple can not be off at the same time
5. If purple is on, blue and yellow have to be on

One can arrive to the with some logical ballet, and the answer is that blue and green are always on when the operator throws all 5 switches together.

Aim of the notebooks

The aim is to solve the game using a python code.

Logic

One way has already been found which is testing all possibilities against the logic.

Linear Program

This is where one can arrive at the answer without 'testing' anything. Here you 'solve' the logic to determine the outcome. The logic must be solved simultaneously, therefore it is envisioned as a linear program.

Let Y: Yellow , R: Red , G: Green , B: Blue , P: Purple. Assume a vlue of 1 as ON and 0 as OFF, i.e. all are binary.

Switch 1: $Y - R \leq 0$

Switch 2: $R + G \leq 1$

Switch 3: $B - G = 0$

Switch 4: $-B - P \leq -1$

Switch 5: $P - B - Y \leq -1$

$-Y - R - G - B - P \leq -1$ (all can not be off)

Looks like this will be a MILP, as all variables can only be 1 or 0.

The system of equations can be re-written in matrix form as

$$\begin{bmatrix} 0 & 0 & -1 & 1 & 0 \end{bmatrix} \begin{bmatrix} Y \\ R \\ G \\ B \\ P \end{bmatrix} = [0] \quad (1)$$

$$\begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & -1 \\ -1 & 0 & 0 & -1 & 1 \\ -1 & -1 & -1 & -1 & -1 \end{bmatrix} \begin{bmatrix} Y \\ R \\ G \\ B \\ P \end{bmatrix} \leq \begin{bmatrix} 0 \\ 1 \\ -1 \\ -1 \\ -1 \end{bmatrix} \quad (2)$$

After implementing in a cvxopt format and solving it, these equations successfully solve the problem.