Defender would respond by picking largest of of

$-y_c, -y_b, -2y_a$

Attacker would try to make these equal

$$y_a^* = \frac{1}{5}, \quad y_b^* = y_c^* = \frac{2}{5}$$

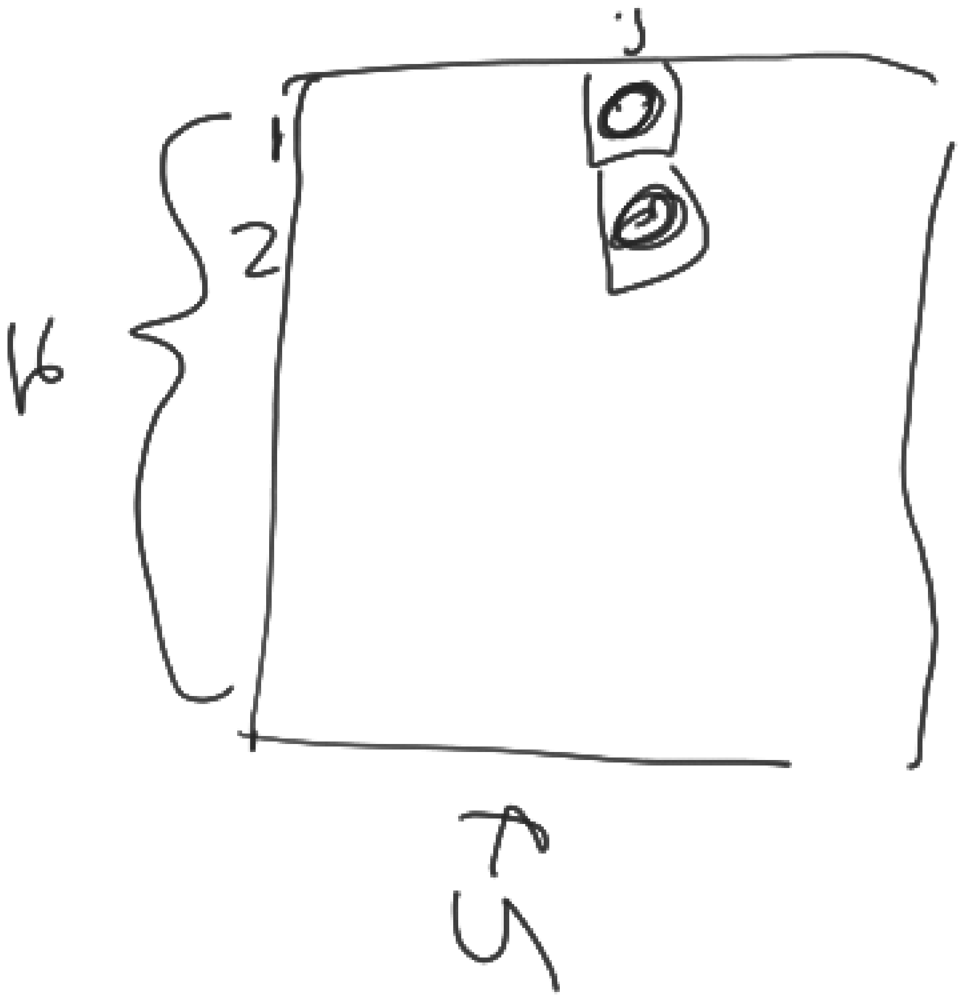
utility of defender is $-\frac{2}{5}$

Minimax Theorem: In a 2-player zero-sum game, the following scenarios lead to the same utility at equilibrium:

- Player 1 moves first (picking x^*)
- Player 2 moves first (picking y^*)
- players move simultaneously

Moreover, (x^*, y^*) is equilibrium of simultaneous game.

How to compute x^* and y^*?



Linear program

for x^*

$$\max \ \delta$$

s.t.

$$x_1 + x_2 + \dots + x_k = 1$$

$$x_1, \dots, x_k \geq 0$$

for every strategy j of opponent

$$x_1 u(1,j) + x_2 u(2,j) + \dots \ x_k u(k,j) \geq \delta$$

Note: maximizing P1's worst case utility over all possible responses j by P2

For scenario 2, similar linear program.

Security games are very successfull application of algorithmic game theory. In general

- Many targets
- Complex interdepences between targets
- Non-zero sum
- Huge matrix.

In most of these settings, there are clever algorithmic solutions.