Hard-To-Solve Bimatrix Games

Rahul Savani

Bernhard von Stengel

Department of Mathematics
London School of Economics

Nash equilibria of bimatrix games

$$A = \begin{bmatrix} 3 & 3 \\ 2 & 5 \\ 0 & 6 \end{bmatrix} \qquad B = \begin{bmatrix} 1 & 0 \\ 0 & 2 \\ 4 & 3 \end{bmatrix}$$

Nash equilibrium =

pair of strategies x , y with

x best response to y and
y best response to x.

Mixed equilibria

$$A = \begin{bmatrix} 3 & 3 \\ 2 & 5 \\ 0 & 6 \end{bmatrix} \qquad B = \begin{bmatrix} 1 & 0 \\ 0 & 2 \\ 4 & 3 \end{bmatrix}$$

$$x = \begin{bmatrix} 0 \\ 1/3 \\ 2/3 \end{bmatrix} \qquad y^T = \begin{bmatrix} 1/3 & 2/3 \end{bmatrix}$$

$$Ay = \begin{bmatrix} 3 \\ 4 \\ 4 \end{bmatrix} \qquad x^TB = \begin{bmatrix} 8/3 & 8/3 \end{bmatrix}$$

only **pure best responses** can have probability > 0

Best response polytope Q for player 2

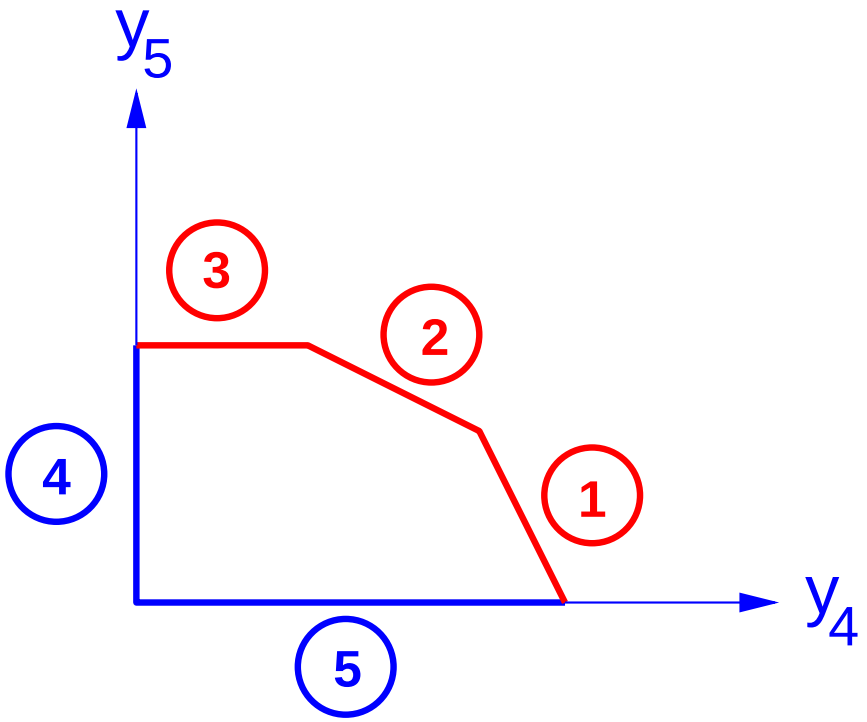
	y_4	y_5
①	3	3
②	2	5
③	0	6

$= A$

$$Q = \{ y \mid Ay \leq 1,\ y \geq 0 \}$$

$$Q = \{ (y_4, y_5) \mid$$

$$\begin{aligned}
① &: \quad 3y_4 + 3y_5 \leq 1 \\
② &: \quad 2y_4 + 5y_5 \leq 1 \\
③ &: \quad\quad 6y_5 \leq 1 \\
④ &: \quad y_4 \geq 0 \\
⑤ &: \quad\quad y_5 \geq 0 \ \}
\end{aligned}$$



Best response polytope Q for player 2

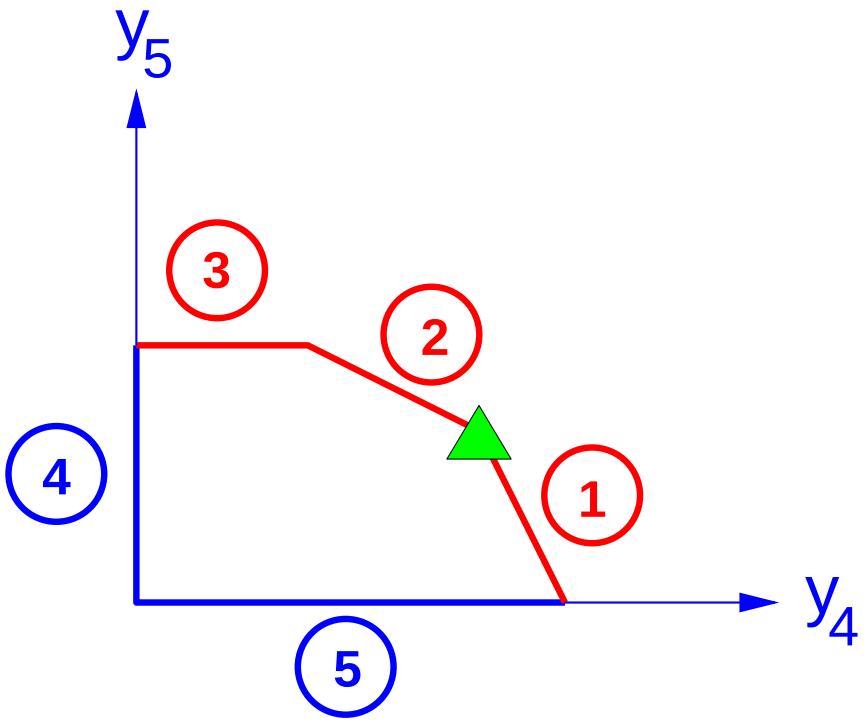
	y_4	y_5
①	3	3
②	2	5
③	0	6

$= A$

$$Q = \{ y \mid Ay \leq 1,\ y \geq 0 \}$$

$$Q = \{ (y_4, y_5) \mid$$

$$\begin{aligned}
① &: \quad 3y_4 + 3y_5 \leq 1 \\
② &: \quad 2y_4 + 5y_5 \leq 1 \\
③ &: \quad\quad 6y_5 \leq 1 \\
\\
④ &: \quad y_4 \geq 0 \\
⑤ &: \quad y_5 \geq 0 \ \}
\end{aligned}$$

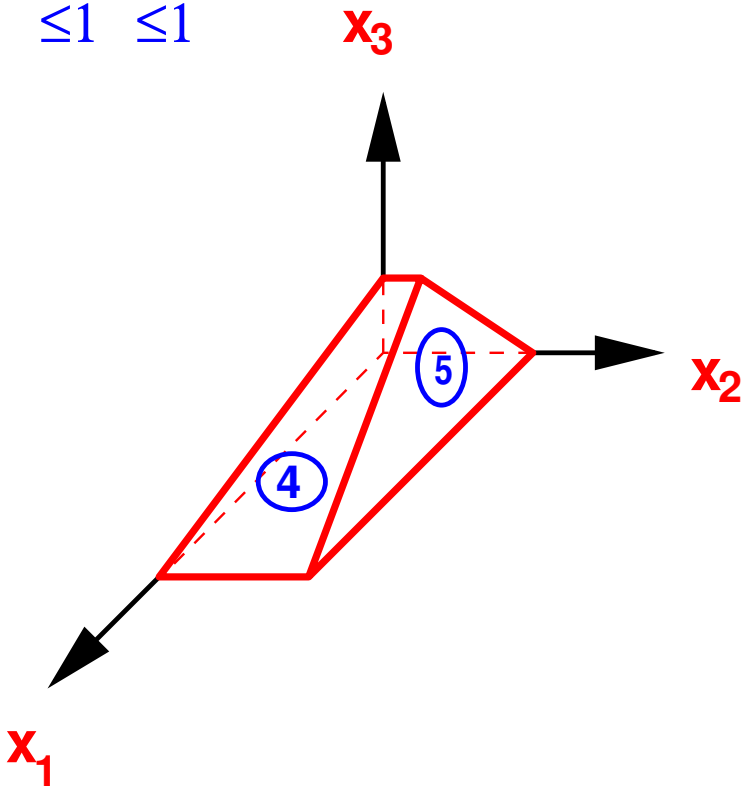


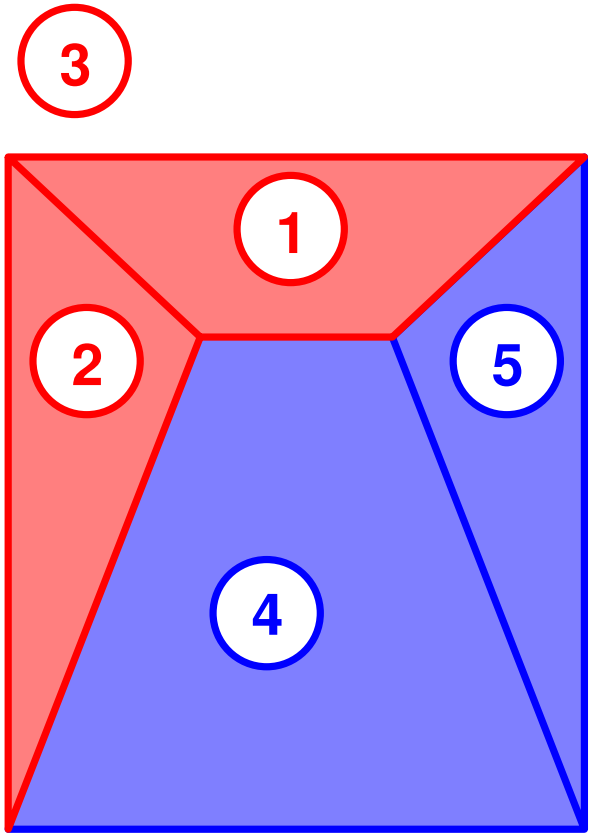
Best response polytope P for player 1

x_1	1	0
x_2	0	2
x_3	4	3
	≤1	≤1

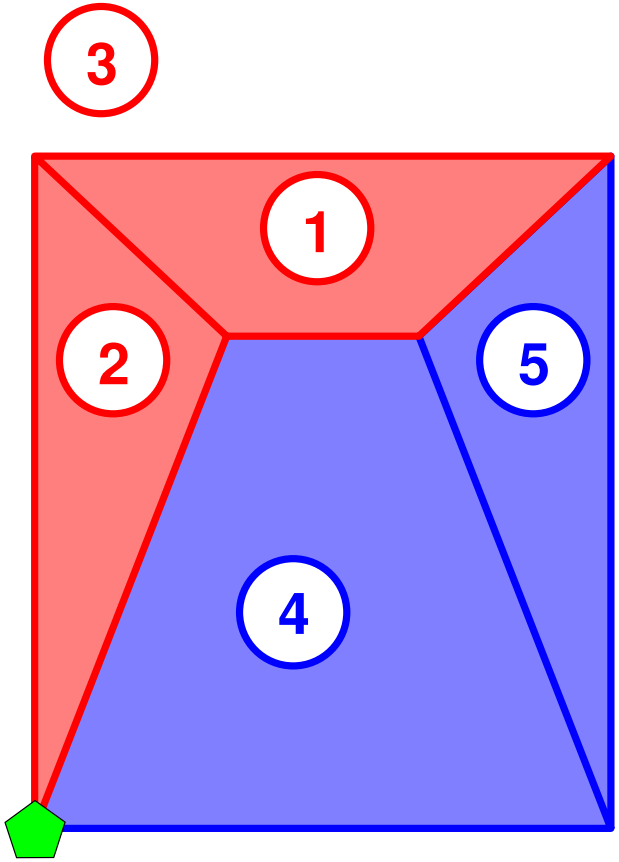
= B

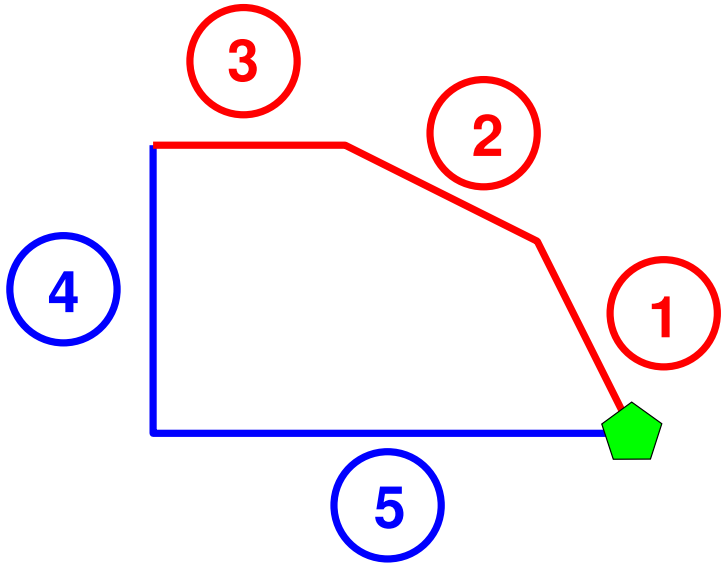
$$P = \{ x \mid x \geq 0,\ x^{\top} B \leq 1 \}$$





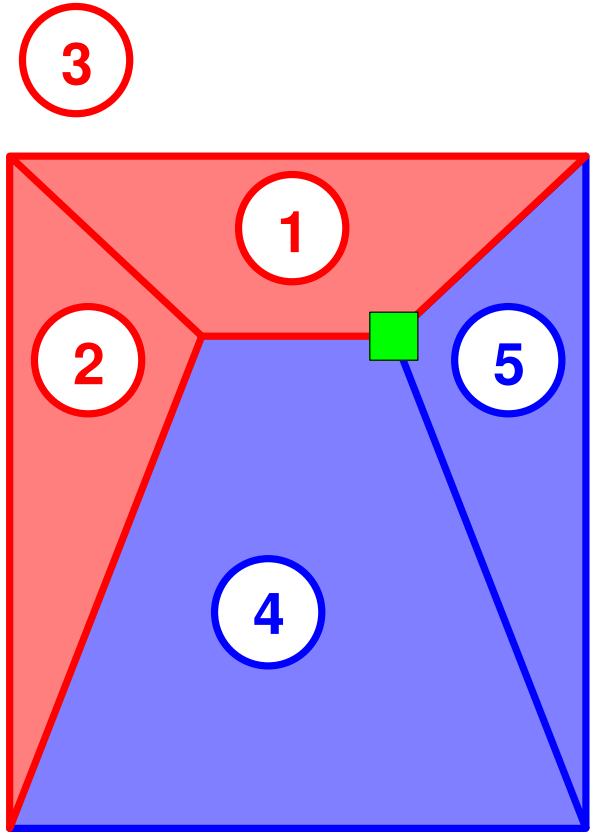
Equilibrium = completely labeled pair

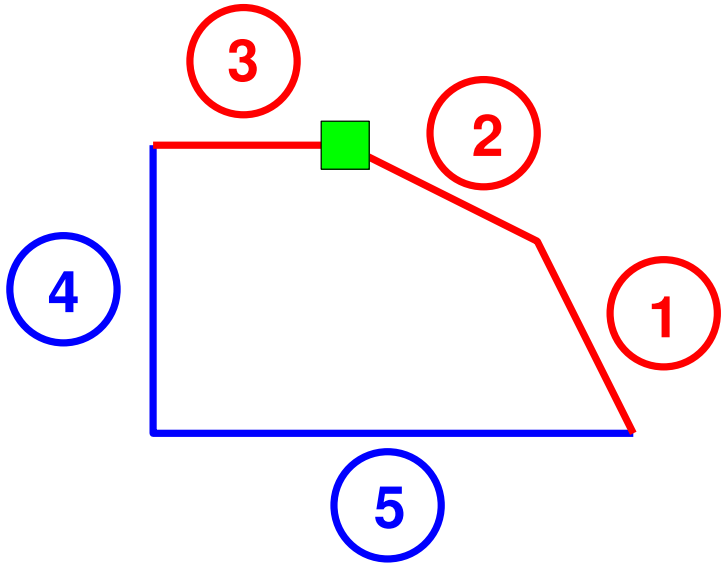




pure equilibrium

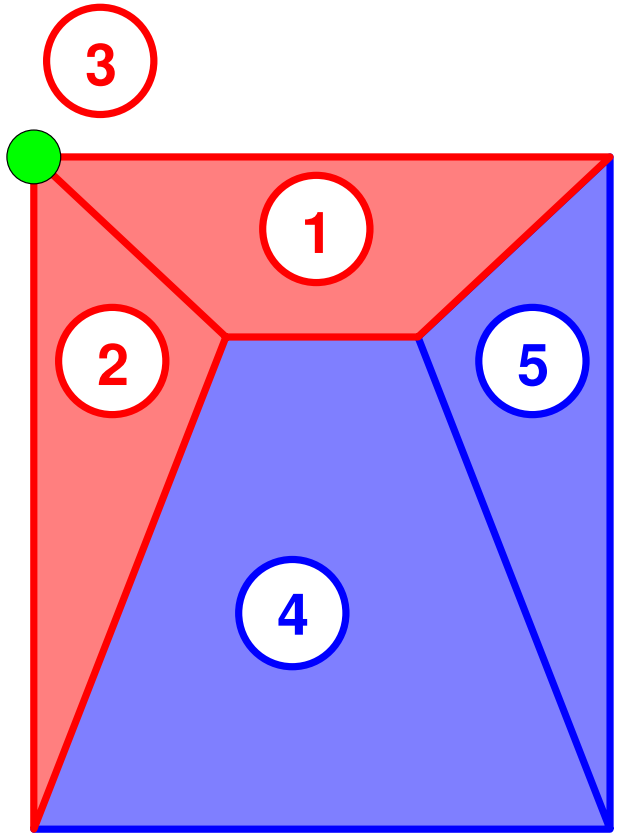
Equilibrium = completely labeled pair

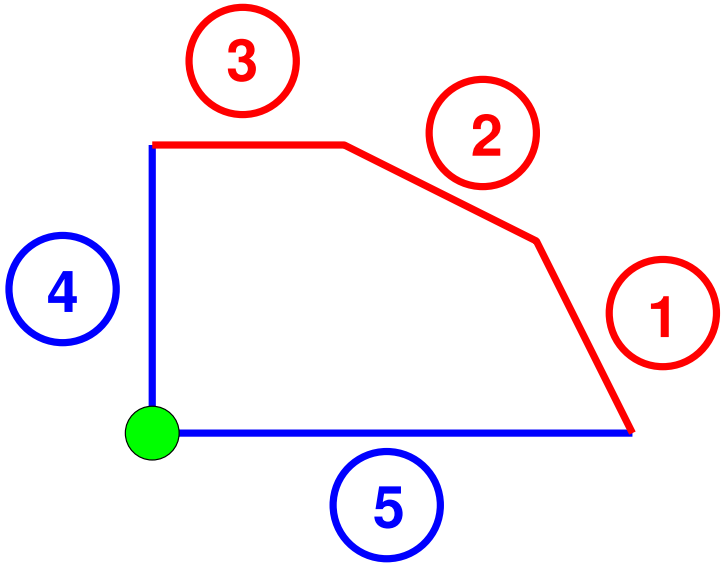




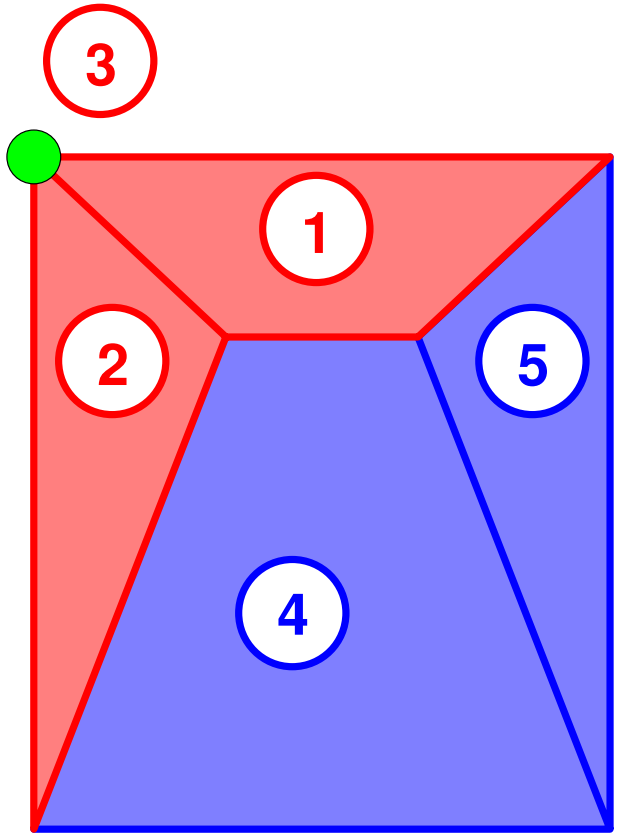
mixed equilibrium

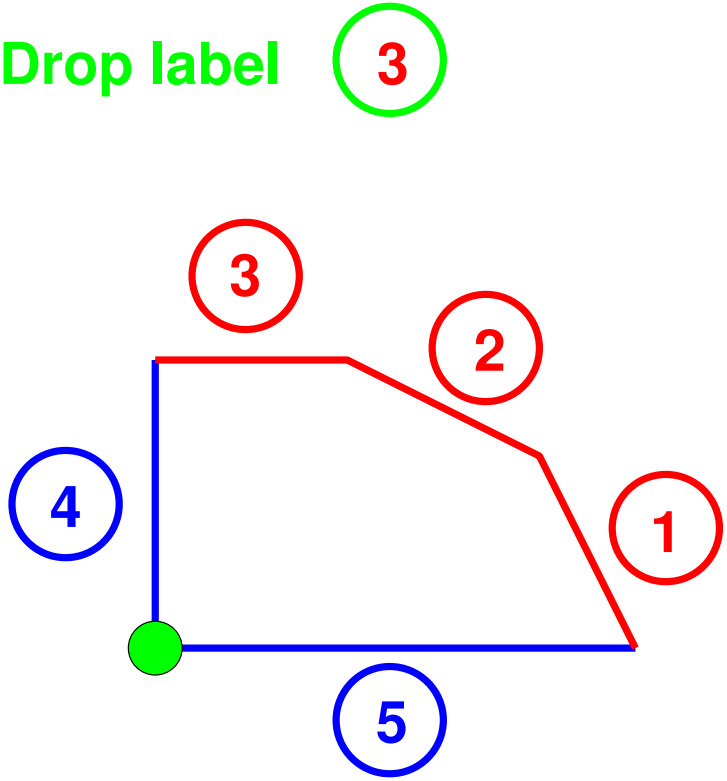
The Lemke–Howson algorithm



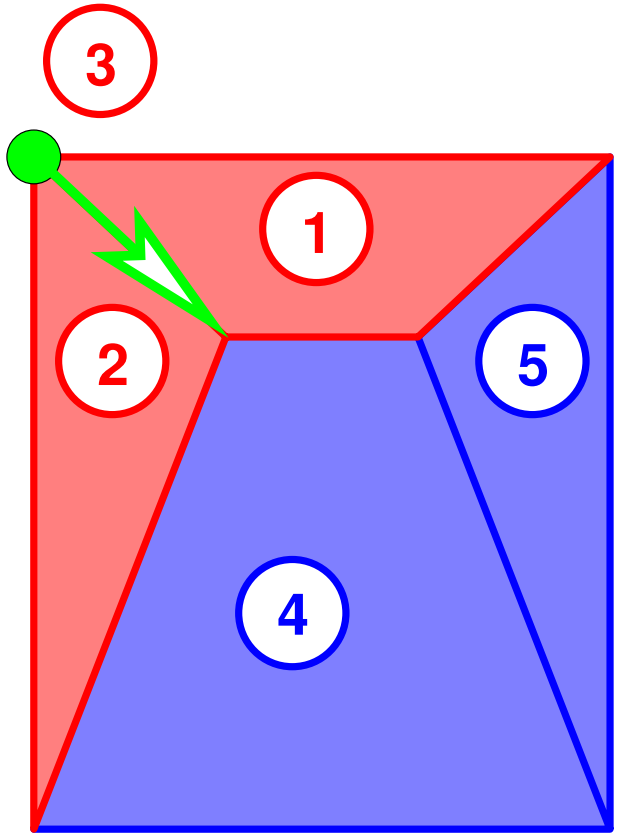


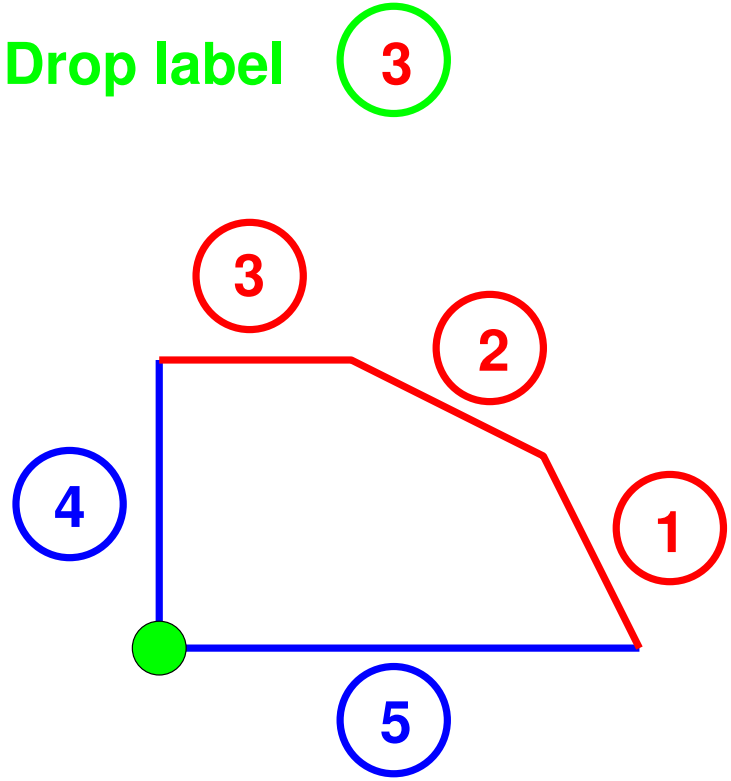
The Lemke–Howson algorithm



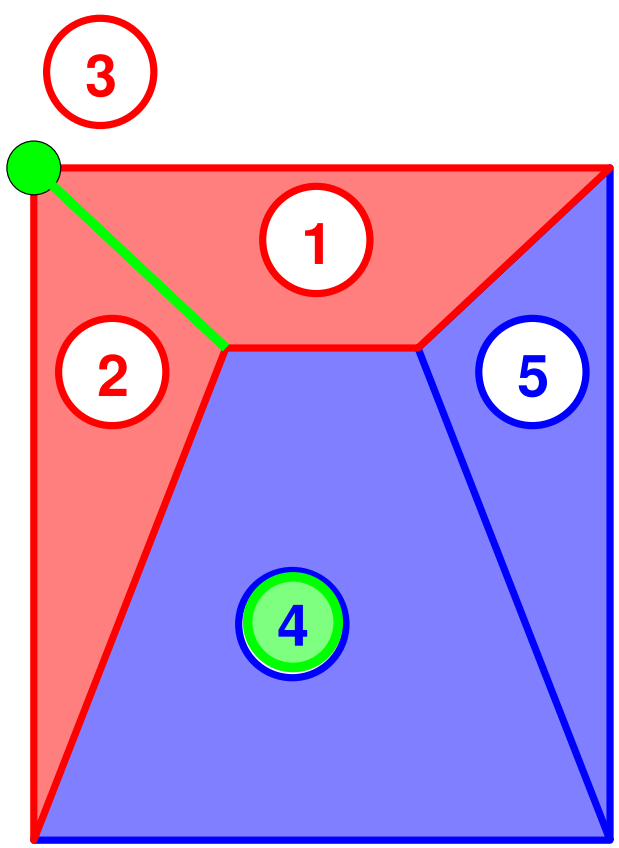


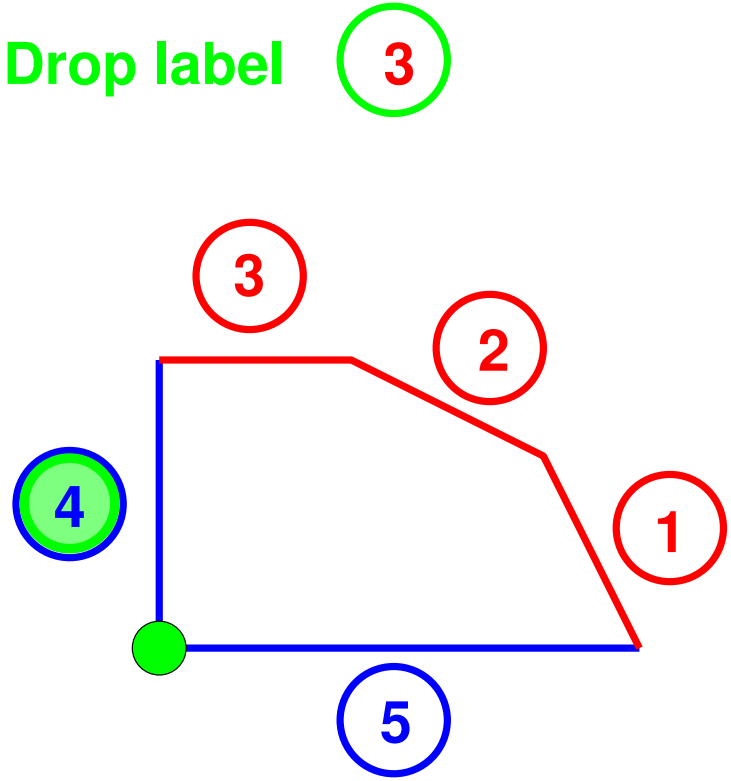
The Lemke–Howson algorithm



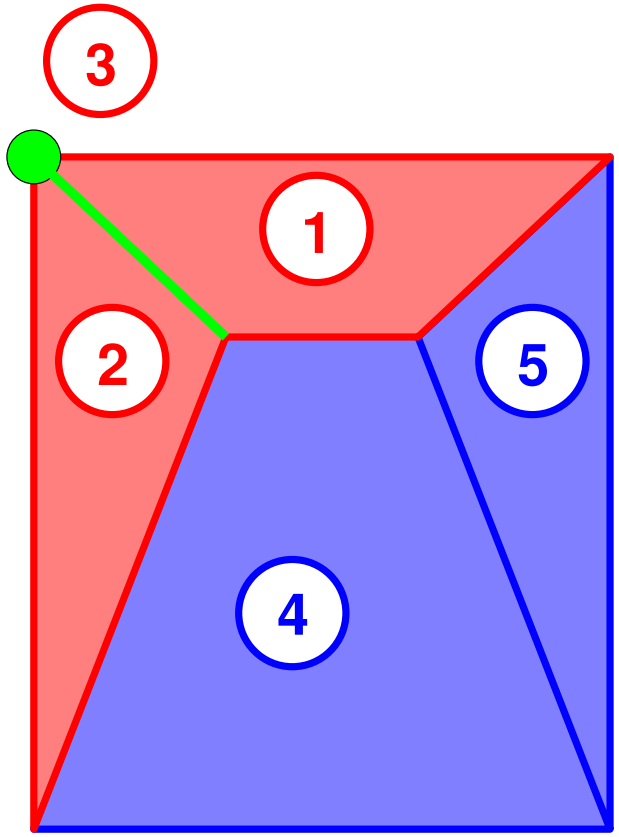


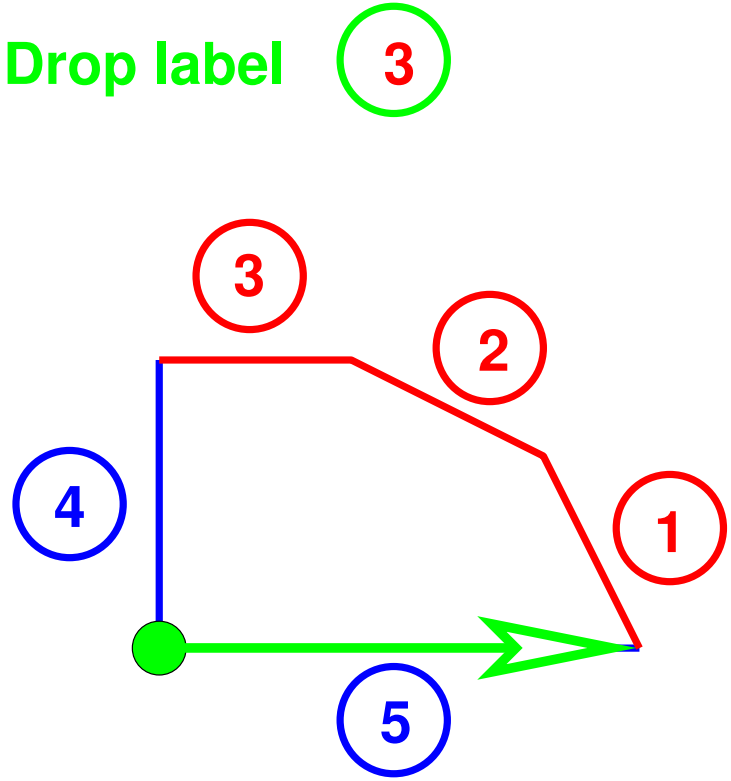
The Lemke–Howson algorithm



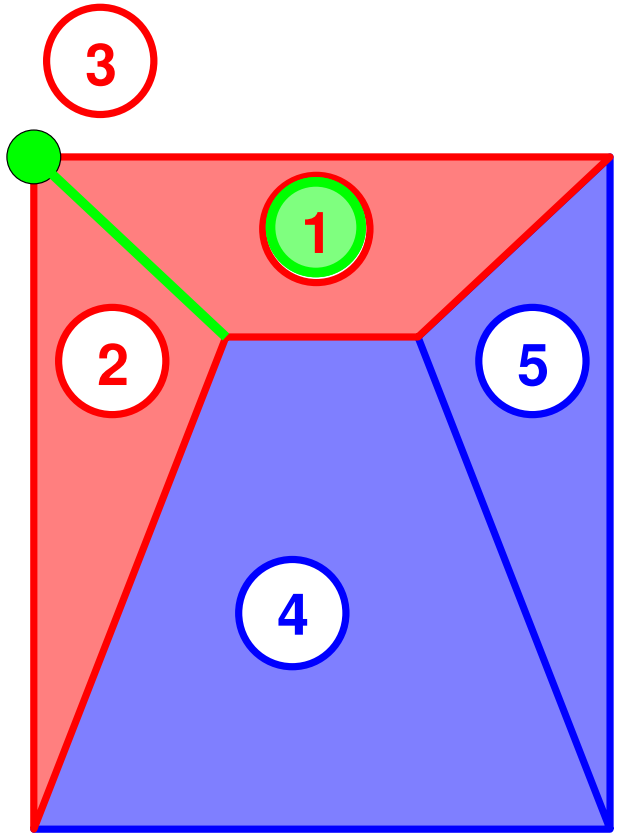


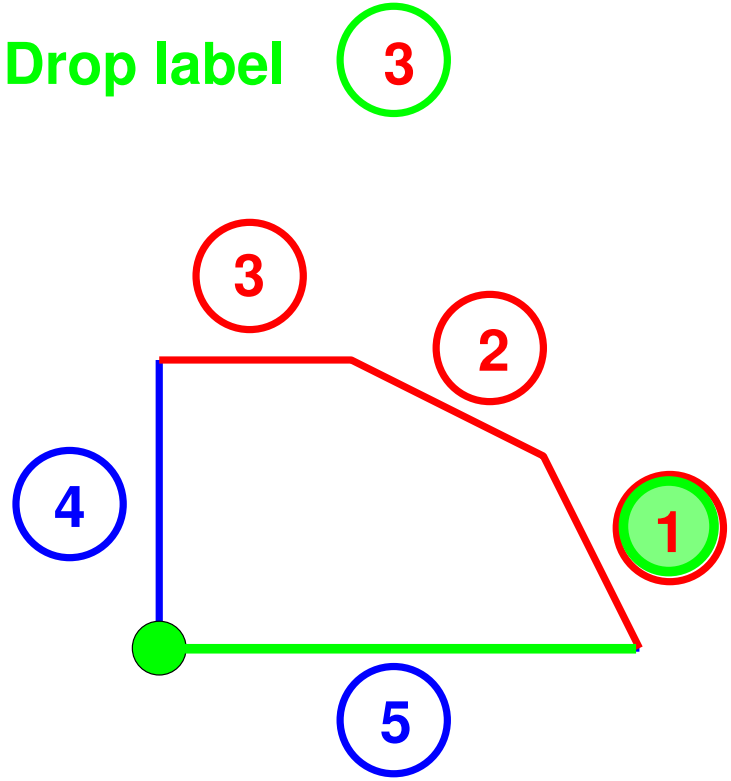
The Lemke–Howson algorithm



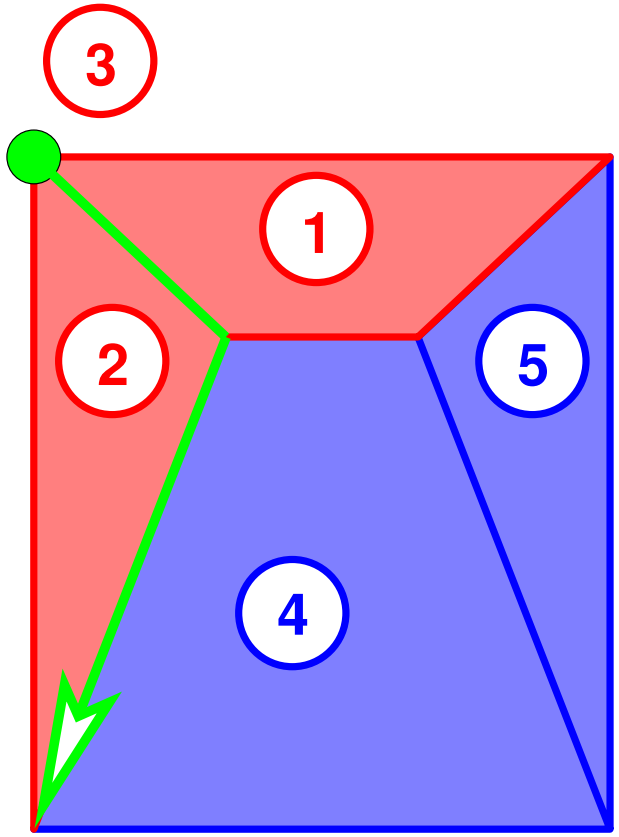


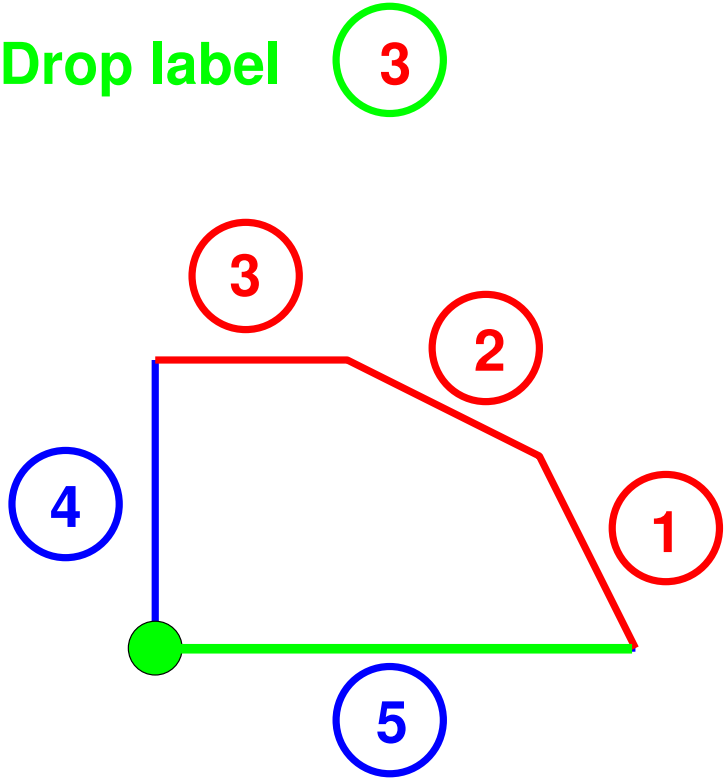
The Lemke–Howson algorithm



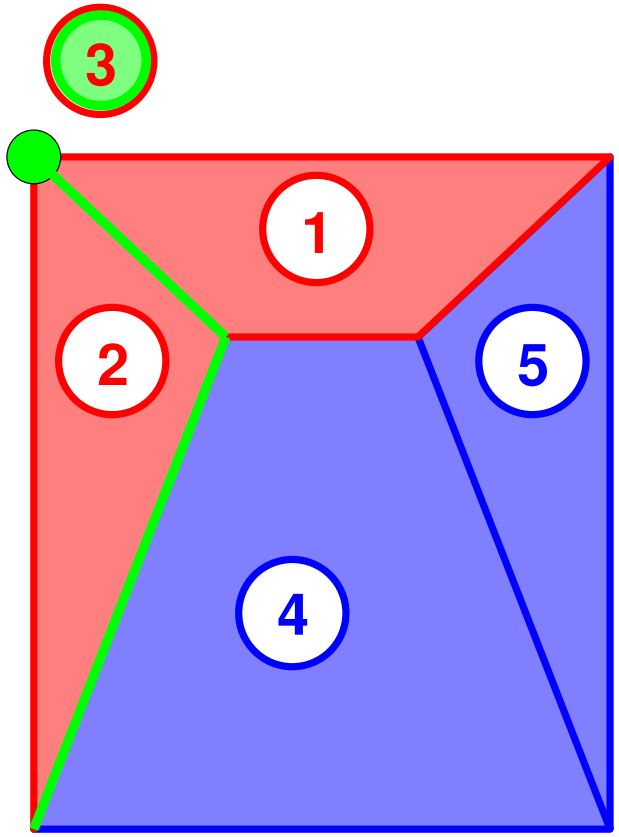


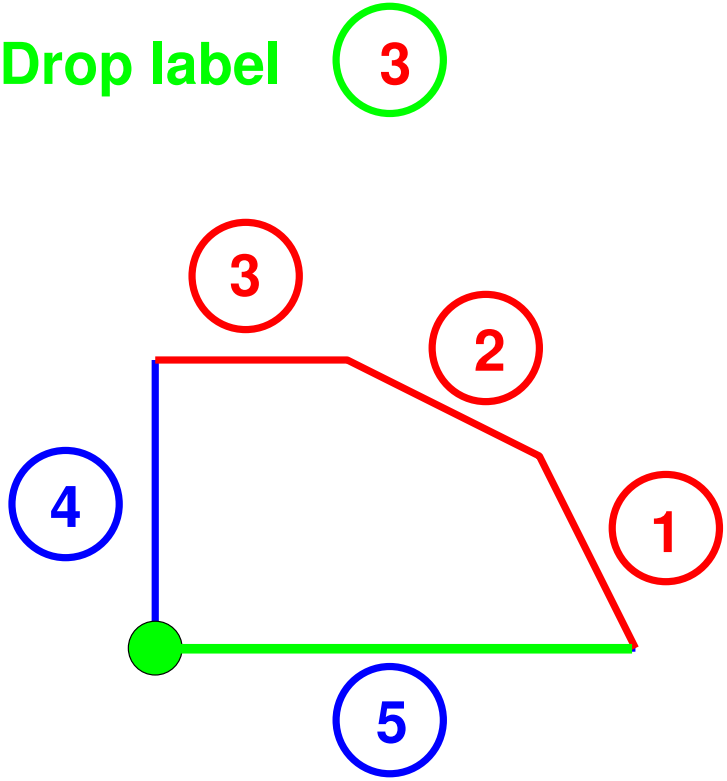
The Lemke–Howson algorithm



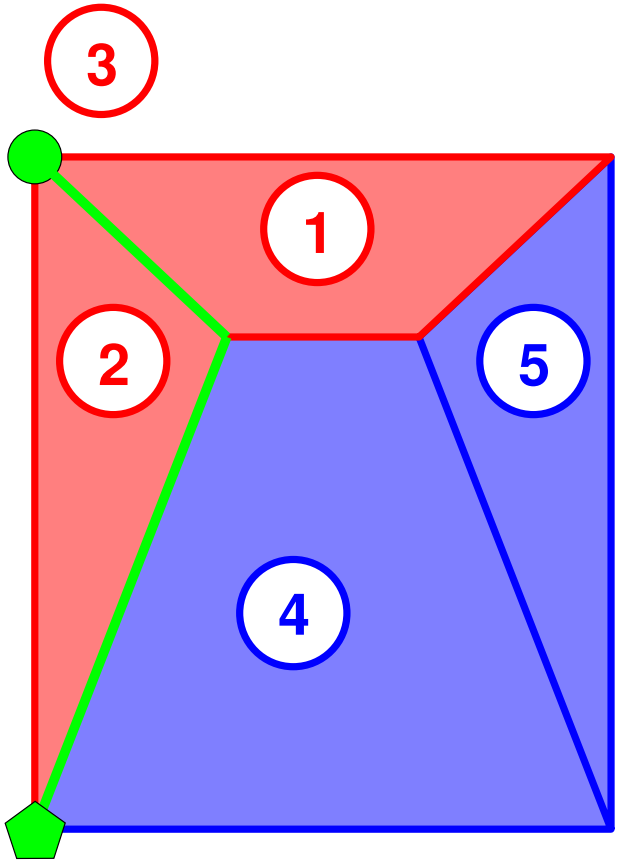


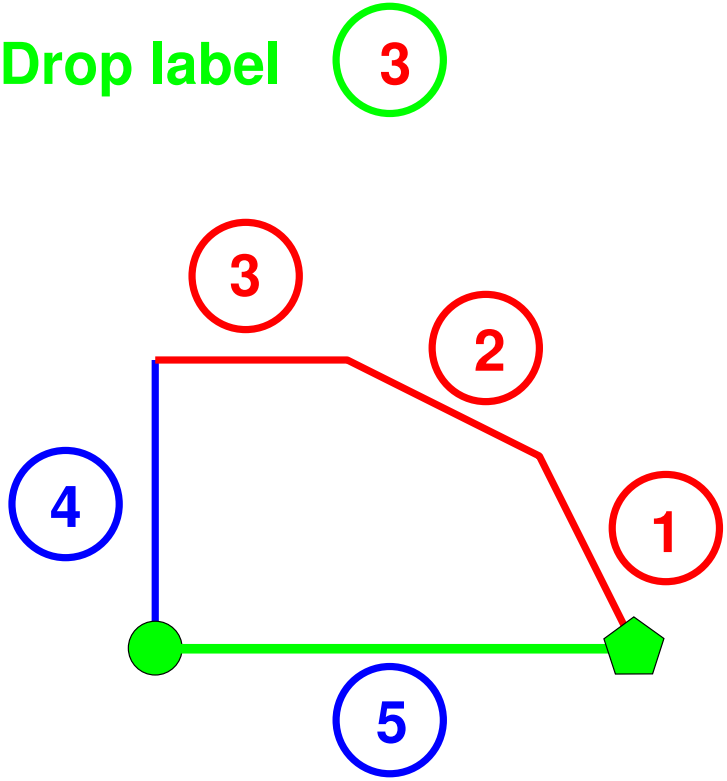
The Lemke–Howson algorithm





The Lemke–Howson algorithm





Why Lemke-Howson works

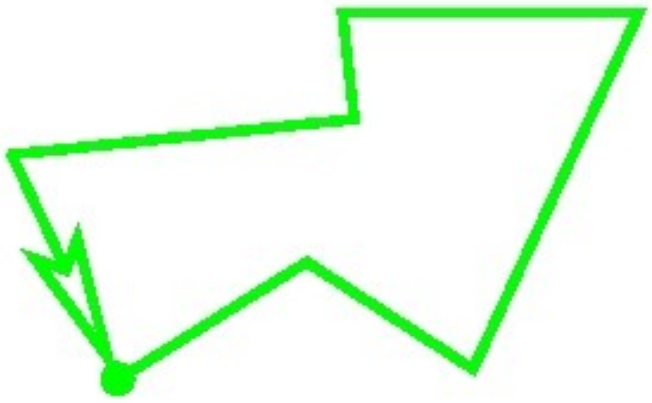
LH finds at least one Nash equilibrium because

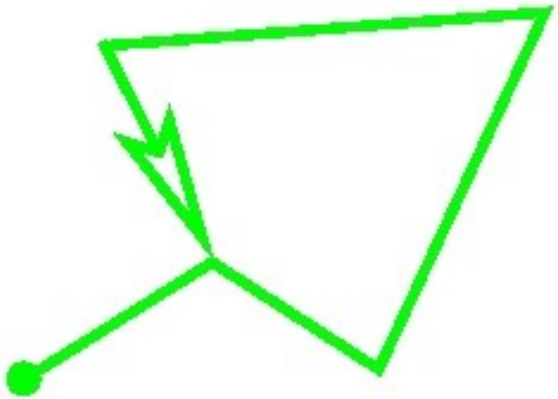
- finitely many "vertices"

for nondegenerate (generic) games:

- **unique** starting edge given missing label
- **unique** continuation

$\Rightarrow$ precludes "coming back" like here:





Complexity of Lemke-Howson

- finds at least one Nash equilibrium,
  pivots like Simplex algorithm for linear programming

- Simplex may be exponential [Klee-Minty cubes]

- exponentially many steps of Lemke-Howson
  for **any** dropped label?

- **Yes**! This is our result.

Our result

There are $d \times d$ games with exactly one Nash equilibrium, for which the Lemke-Howson algorithm takes $\geq \phi^{3d/4}$ many steps for **any dropped label** (with Golden Ratio $\phi = (\sqrt{5} + 1) / 2 = 1.618...$)

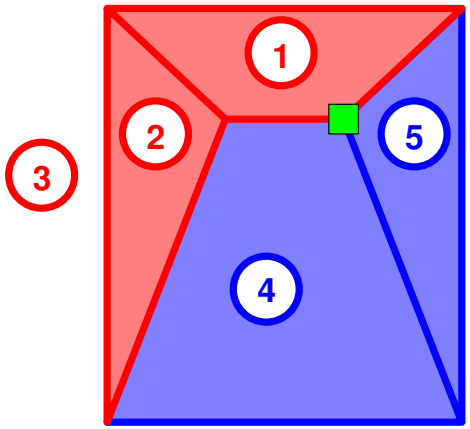
We will show this extending

[Morris 1994] - exponentially long Lemke paths (finds symmetric equilibria of symmetric games)

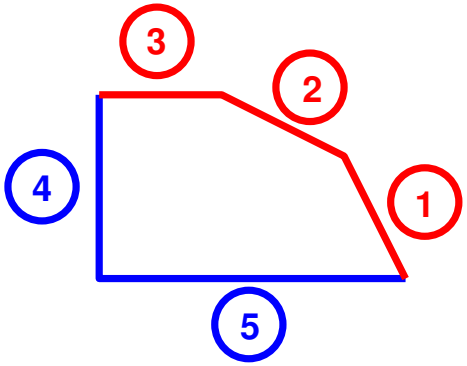
[von Stengel 1999] - games with many equilibria

using **dual cyclic polytopes**

Vertices as bit patterns



P

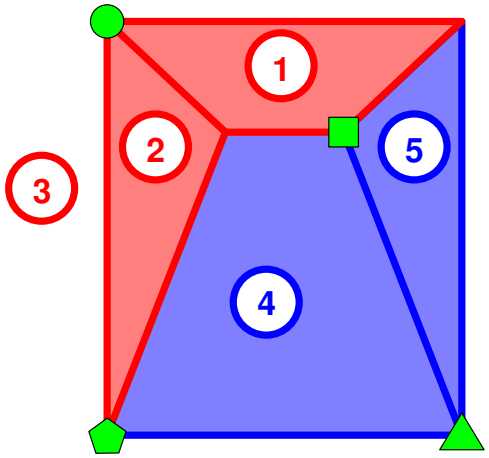


Q

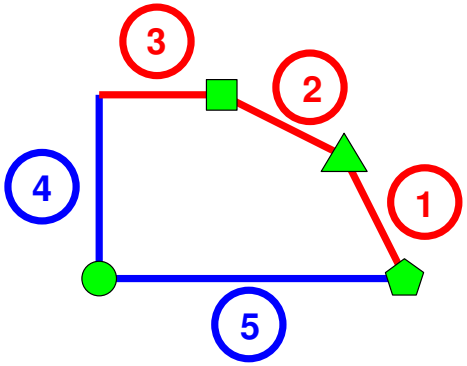
1	2	3	4	5
1	1	1	0	0
1	1	0	1	0
1	0	1	0	1
1	0	0	1	1
0	0	1	1	1
0	1	1	1	0

1	2	3	4	5
0	0	0	1	1
0	0	1	1	0
0	1	1	0	0
1	1	0	0	0
1	0	0	0	1

Vertices as bit patterns



P



Q

	1	2	3	4	5
●	1	1	1	0	0
	1	1	0	1	0
	1	0	1	0	1
■	1	0	0	1	1
▲	0	0	1	1	1
⬟	0	1	1	1	0

1	2	3	4	5
0	0	0	1	1
0	0	1	1	0
0	1	1	0	0
1	1	0	0	0
1	0	0	0	1

Dual cyclic polytopes

- vertices = strings of **N** bits with **d** bits "**1**",
- **no** odd substrings 010, 01110, 0111110, . . .
  [Gale **even**ness]

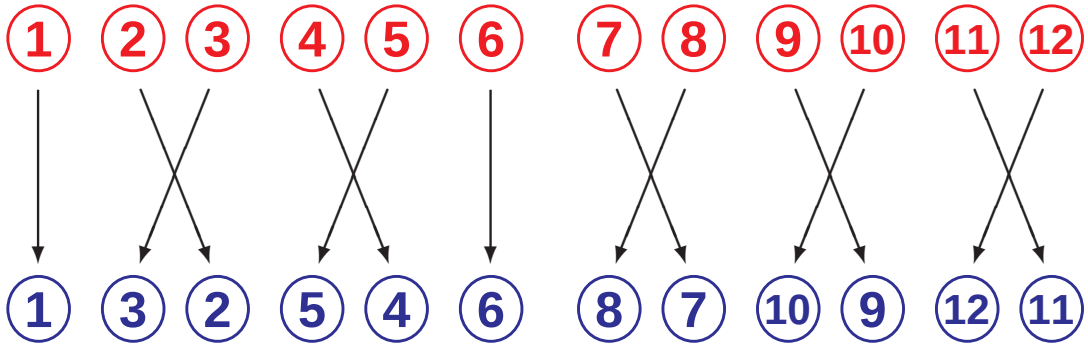
Example: d=4, N=6 d=2, N=6 (4×2 game)

d=4, N=6	d=2, N=6
111100	000011
111001	000110
110110	001100
110011	011000
101101	110000
100111	100001
011110	
011011	
001111	

Permuted labels

P = dual cyclic polytope in dimension **d** with **2d** facets

with facets labeled



Q = P

with facets labeled

$\Longrightarrow$ only **one** non-artificial equilibrium:

0 0 0 0 0 0 1 1 1 1 1 1

1 1 1 1 1 1 0 0 0 0 0 0

Lemke–Howson will take long to find it!

Lemke-Howson on dual cyclic polytopes

P								Q							
①	②	③	④	⑤	⑥	⑦	⑧	①	③	②	④	⑥	⑤	⑧	⑦
1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1

Lemke-Howson on dual cyclic polytopes

P								Q							
1	2	3	4	5	6	7	8	1	3	2	4	6	5	8	7
1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1
0	1	1	1	1	0	0	0								

Lemke-Howson on dual cyclic polytopes

P								Q							
①	②	③	④	⑤	⑥	⑦	⑧	①	③	②	④	⑥	⑤	⑧	⑦
1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1
0	1	1	1	1	0	0	0	0	0	0	1	1	0	1	1

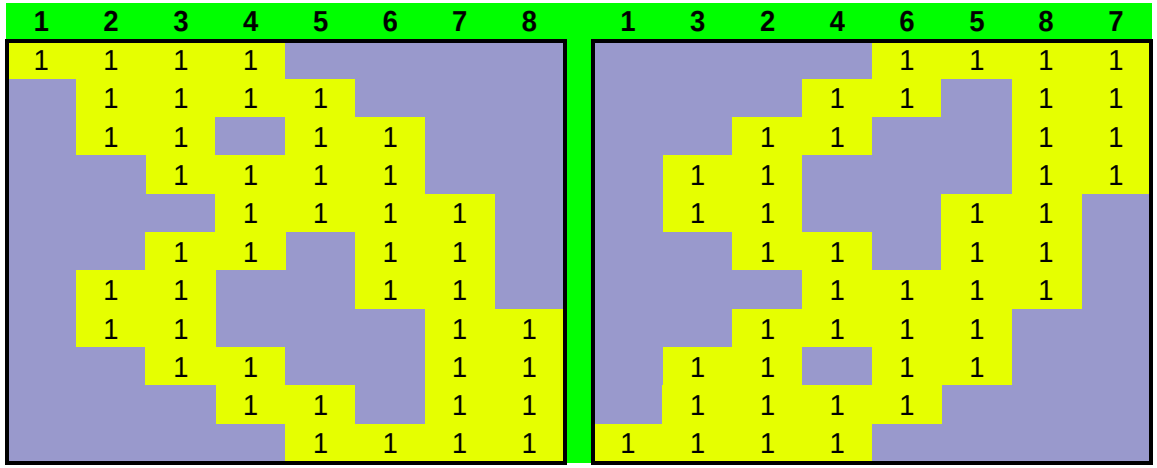
Lemke-Howson on dual cyclic polytopes

P								Q							
1	2	3	4	5	6	7	8	1	3	2	4	6	5	8	7
1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1
0	1	1	1	1	0	0	0	0	0	0	1	1	0	1	1
0	1	1	0	1	1	0	0								

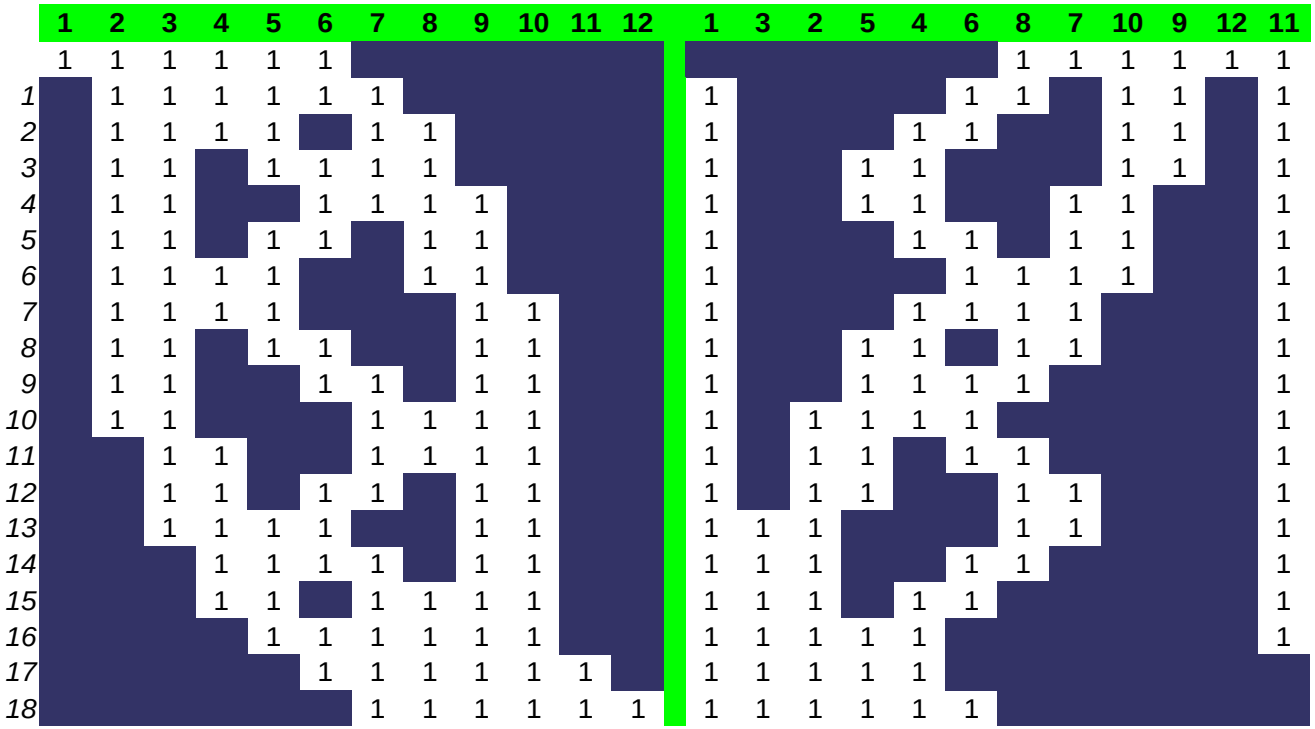
Lemke-Howson on dual cyclic polytopes

P								Q							
①	②	③	④	⑤	⑥	⑦	⑧	①	③	②	④	⑥	⑤	⑧	⑦
1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1
0	1	1	1	1	0	0	0	0	0	0	1	1	0	1	1
0	1	1	0	1	1	0	0	0	0	1	1	0	0	1	1
0	0	1	1	1	1	0	0	0	1	1	0	0	0	1	1
0	0	0	1	1	1	1	0	0	1	1	0	0	1	1	0
0	0	1	1	0	1	1	0	0	0	1	1	0	1	1	0
0	1	1	0	0	1	1	0	0	0	0	1	1	1	1	0
0	1	1	0	0	0	1	1	0	0	1	1	1	1	0	0
0	0	1	1	0	0	1	1	0	1	1	0	1	1	0	0
0	0	0	1	1	0	1	1	0	1	1	1	1	0	0	0
0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0

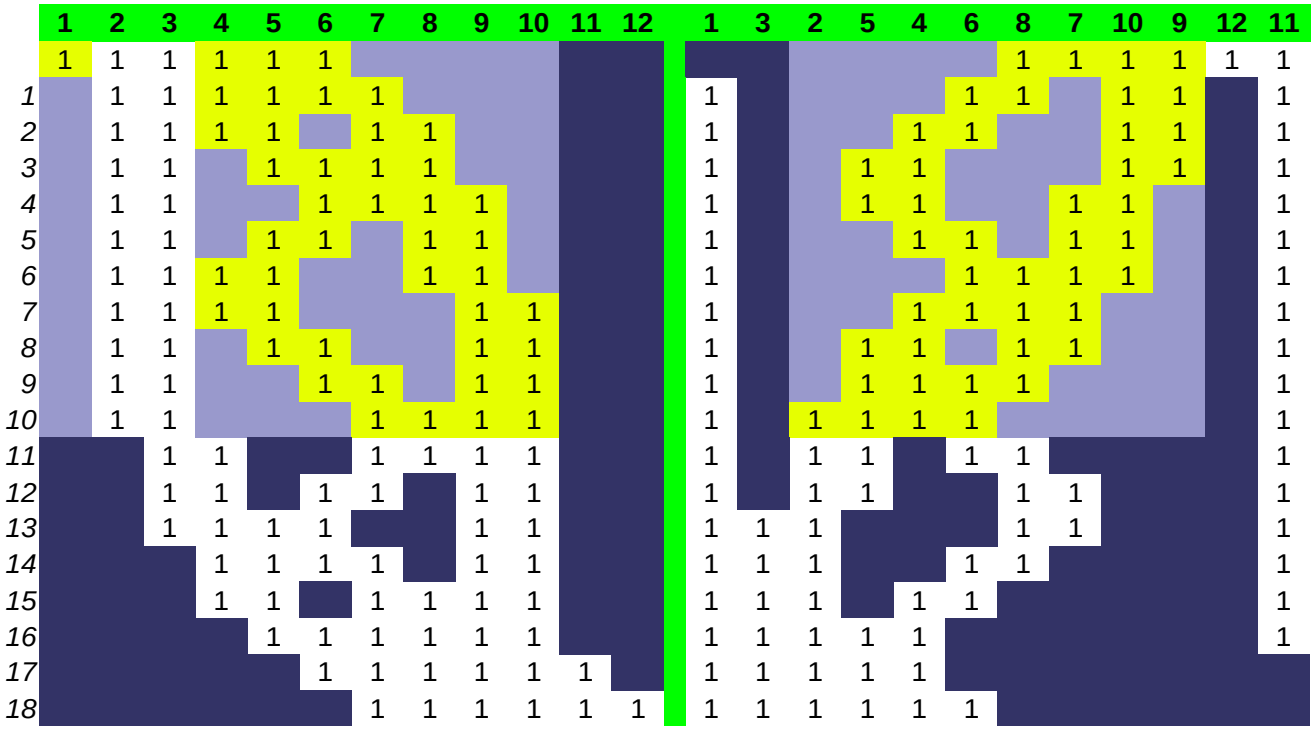
A(4) = path for d=4, label 1



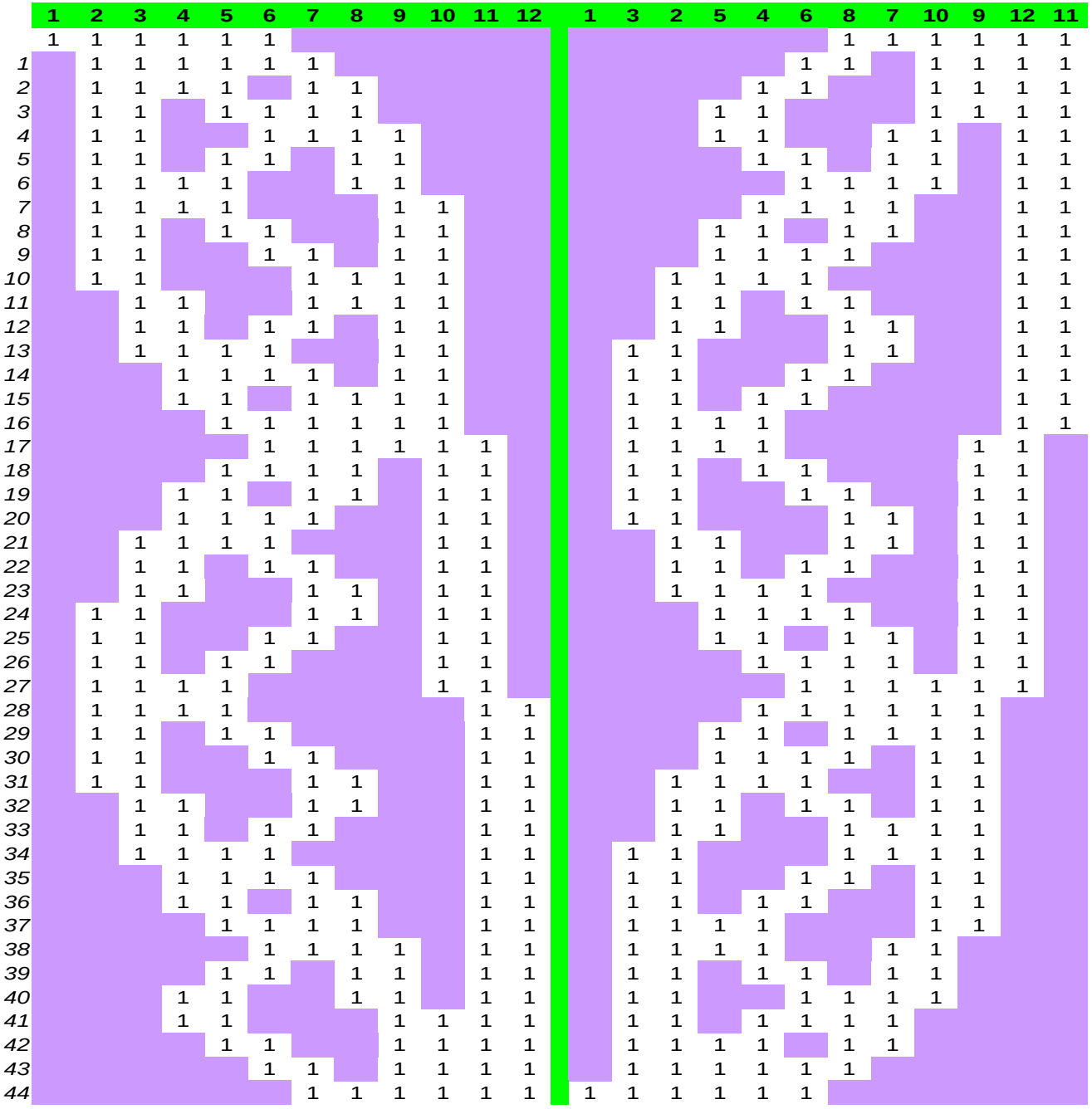
B(6) = path for d=6, label 12



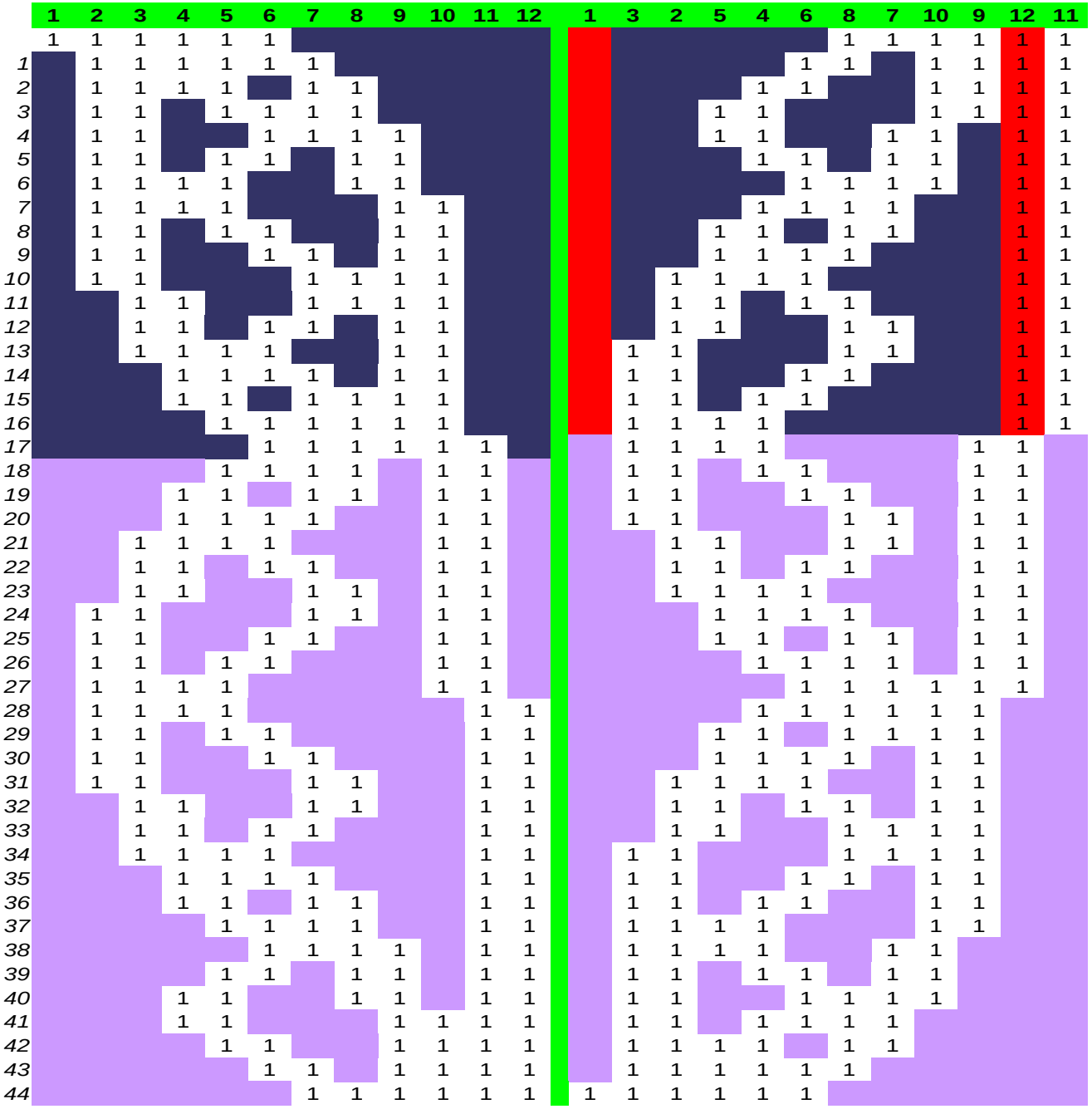
A(4) is prefix of B(6)



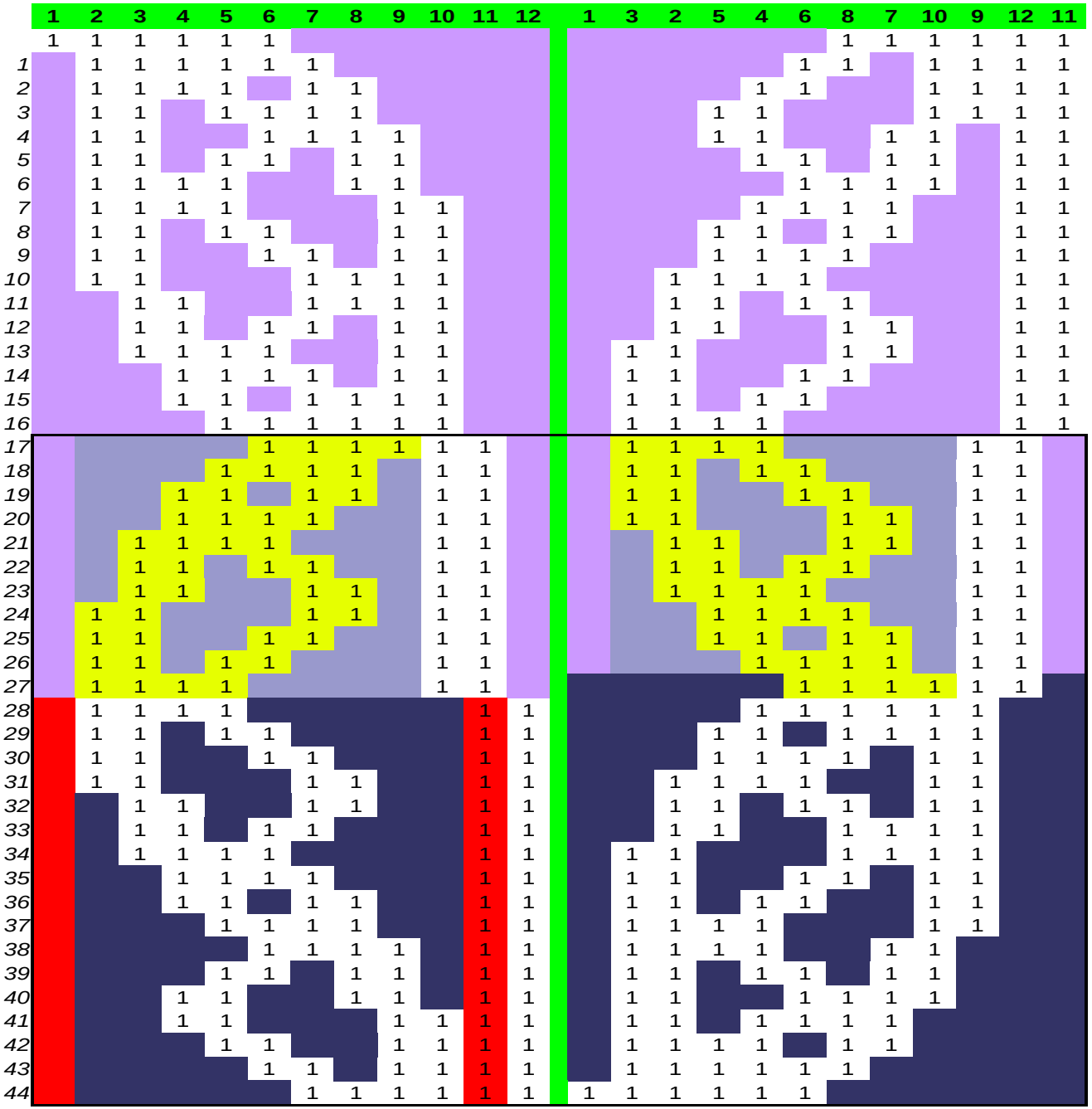
A(6) = path for d=6, label 1



B(6) is prefix of A(6)



Suffix of A(6) = C(6) = A(4)+B(6)



Recurrences for longest paths

A(d) = LH path dropping label 1 in dim d

B(d) = LH path dropping label 2d in dim d

C(d) = suffix of A(d)

lengths of

B(2) C(2) **A**(2) B(4) C(4) **A**(4) B(6) C(6) **A**(6) . . .

are the **Fibonacci** numbers

2 3 5 8 13 21 34 55 89 . . .

Exponential path lengths

longest paths: drop label 1 or 2d, paths A(d), B(d)

path length $\Omega(\phi^{3d/2})$

with Golden Ratio $\phi = (\sqrt{5} + 1) / 2 = 1.618...$

shortest paths: drop label 3d/2, path B(d/2)+B(d/2+1)

path length $\Omega(\phi^{3d/4}) = \Omega(1.434...^{d})$

So far:

- NE of a bimatrix game = combinatorial polytope problem
- label dual cyclic polytopes,
  equilibrium at end of **exponentially long** paths
- **but**: fully mixed equilibrium easily guessed
  by support enumeration algorithms

Extension to nonsquare games

- Extend to $d \times 2d$ games with hard-to-guess support (exponentially many guesses on average) **and** exponentially long paths

- Use dual cyclic polytopes with similar labeling as for $d \times d$ games.

Nonsquare games

P = dual cyclic polytope in dimension d with 3d facets

Q = dual cyclic polytope in dimension 2d with 3d facets

P's facets labeled 1 2 3 4 5 6 7 8 9 10 11 12

Q's facets labeled 1 3 2 4 6 5 8 7 10 9 12 11

equilibria:

0 0 0 0 1 1 1 1 0 0 0 0	1 1 1 1 0 0 0 0 1 1 1 1
0 0 0 0 1 1 0 0 1 1 0 0	1 1 1 1 0 0 1 1 0 0 1 1
0 0 0 0 0 1 1 0 0 1 1 0	1 1 1 1 0 1 1 0 0 1 1 0
0 0 0 0 0 0 1 1 0 1 1 0	1 1 1 1 1 1 0 0 0 1 1 0
. . .	

Equilibrium supports = exponentially small fraction of all supports