Constructing and computing equilibria for two-player games

Bernhard von Stengel

Department of Mathematics
London School of Economics

Nash equilibria of bimatrix games

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

Nash equilibrium =

pair of strategies x , y with

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

Mixed equilibria

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

$$x = \begin{bmatrix} 2/3 \\ 1/3 \\ 0 \end{bmatrix} \qquad x^T B = \begin{bmatrix} 5/3 & 5/3 \end{bmatrix}$$

$$Ay = \begin{bmatrix} 4 \\ 4 \\ 3 \end{bmatrix} \qquad y^T = \begin{bmatrix} 1/3 & 2/3 \end{bmatrix}$$

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

Best response condition

Let $\mathbf{x}$ and $\mathbf{y}$ be mixed strategies of player I and II, respectively. Then $\mathbf{x}$ is a best response to $\mathbf{y}$

$\iff$ for all pure strategies i of player I:

$$x_i > 0 \implies (\mathbf{Ay})_i = u = \max\{(\mathbf{Ay})_k \mid 1 \le k \le m\}.$$

Here, $(\mathbf{Ay})_i$ is the ith component of $\mathbf{Ay}$, which is the expected payoff to player I when playing row i.

Proof.

$$\mathbf{xAy} = \sum_{i=1}^{m} \mathbf{x}_i (\mathbf{Ay})_i = \sum_{i=1}^{m} \mathbf{x}_i (u - (u - (\mathbf{Ay})_i)$$

$$= \sum_{i=1}^{m} \mathbf{x}_i u - \sum_{i=1}^{m} \mathbf{x}_i (u - (\mathbf{Ay})_i) = u - \sum_{i=1}^{m} \mathbf{x}_i (u - (\mathbf{Ay})_i) \le u,$$

because $\mathbf{x}_i \ge 0$ and $u - (\mathbf{Ay})_i \ge 0$ for all i. Furthermore, $\mathbf{xAy} = u \iff \mathbf{x}_i > 0$ implies $(\mathbf{Ay})_i = u$, as claimed.

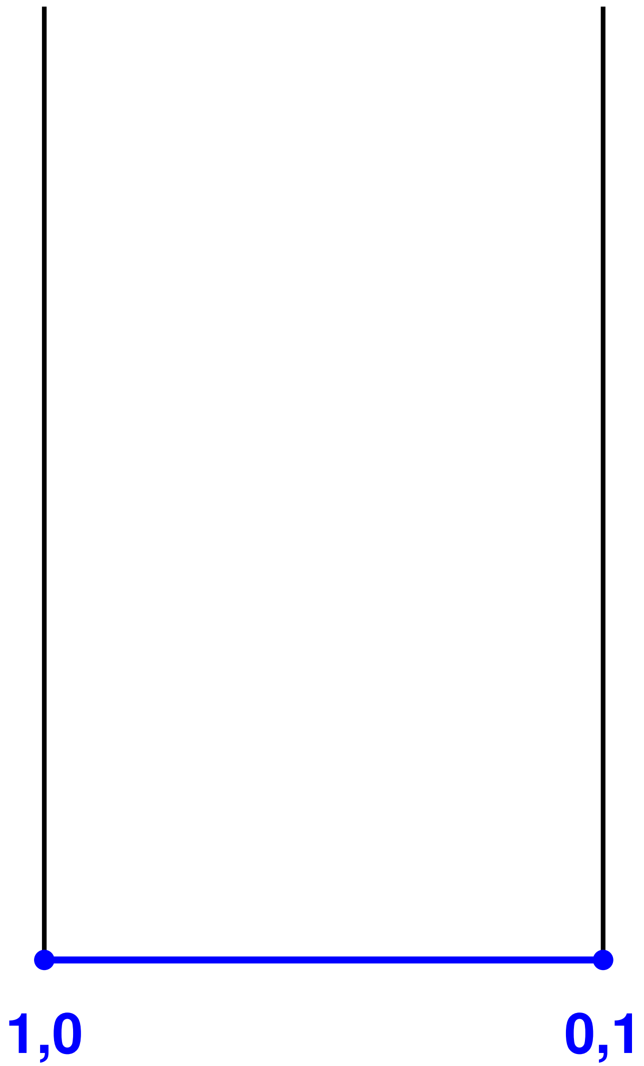
Best responses to mixed strategy of player 2

	4	5
1	0	6
2	2	5
3	3	3

= A

payoffs to
player I

Best responses to mixed strategy of player 2

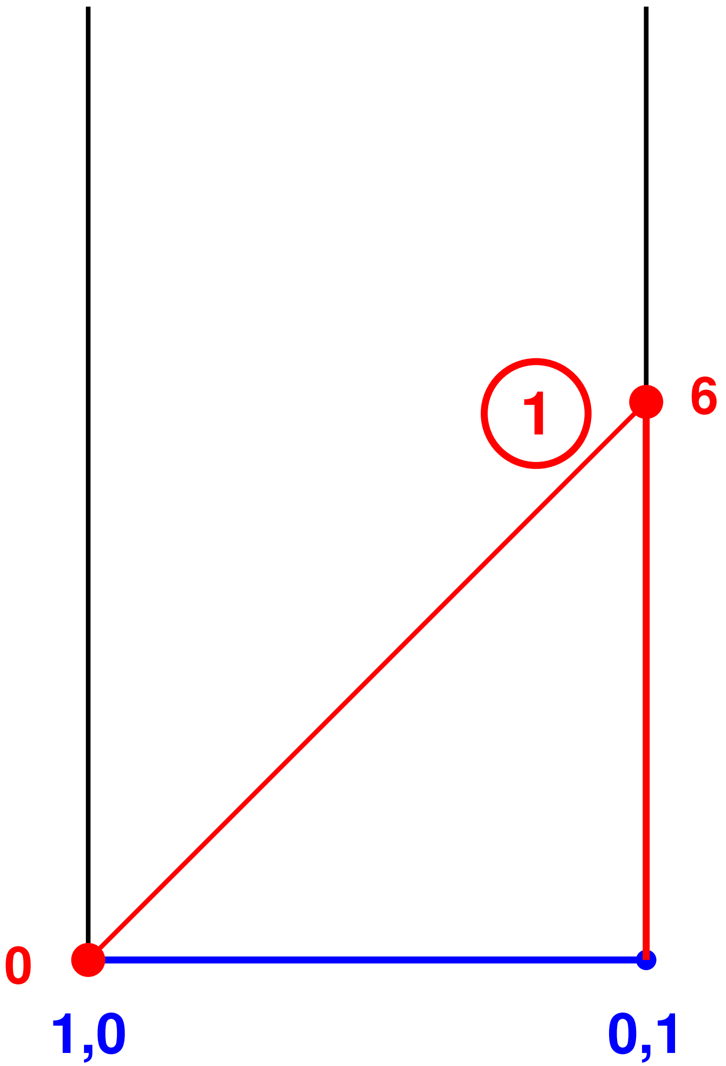


	4	5
1	0	6
2	2	5
3	3	3

= A

payoffs to
player I

Best responses to mixed strategy of player 2

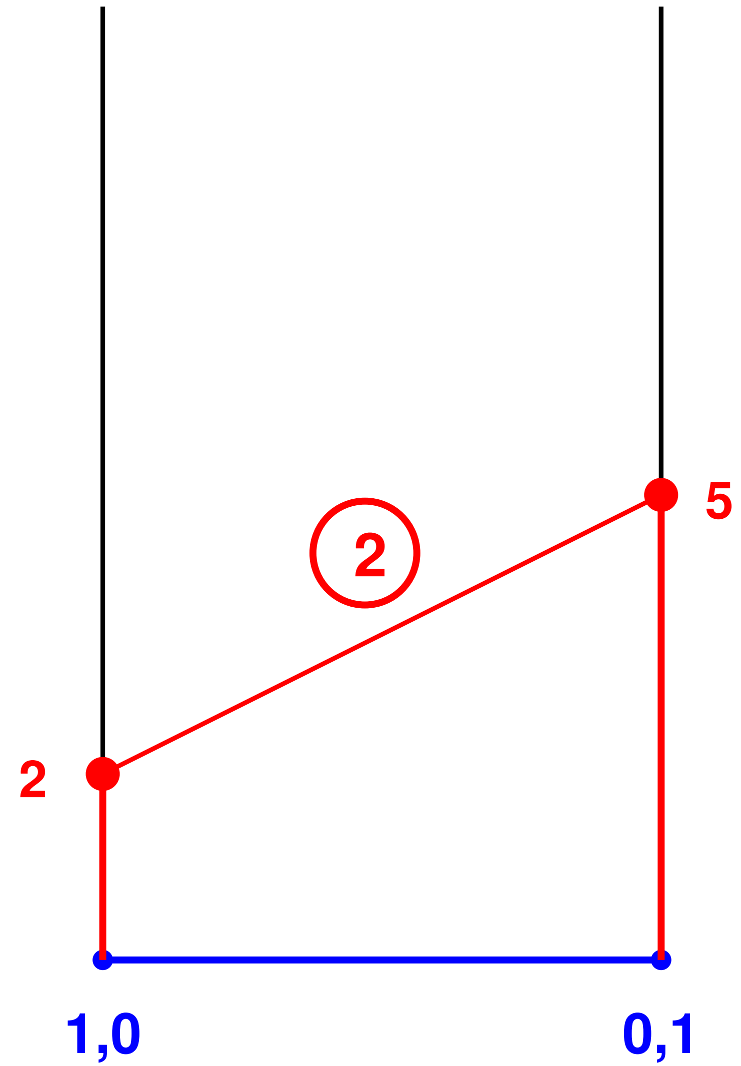


	4	5
1	0	6
2	2	5
3	3	3

= A

payoffs to player I

Best responses to mixed strategy of player 2

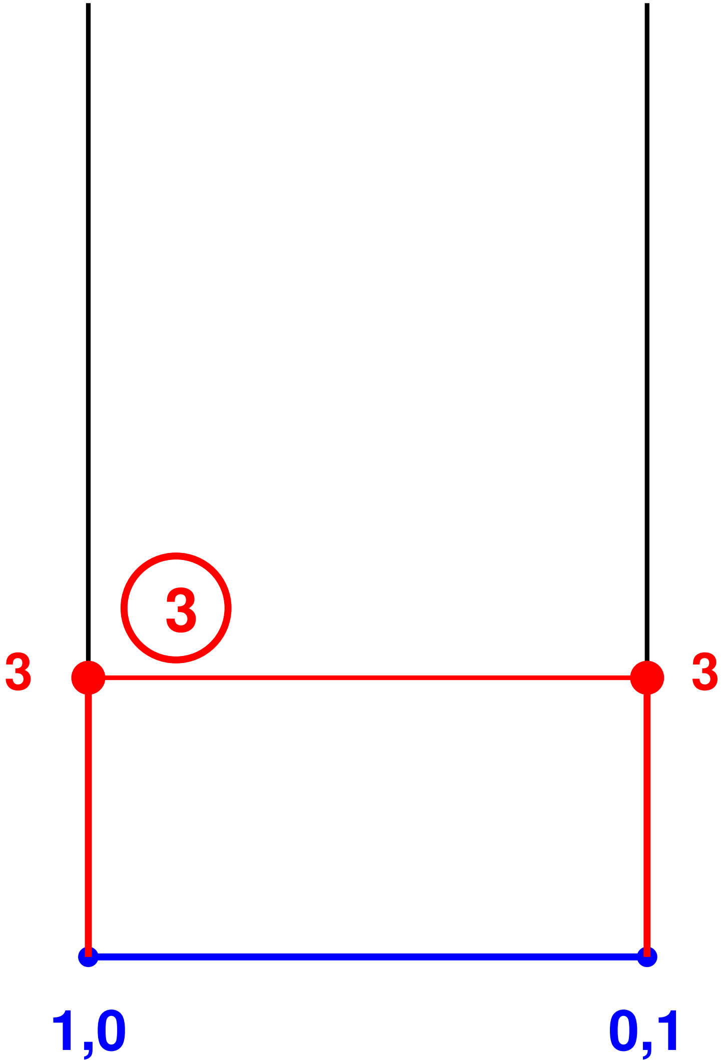


	4	5
1	0	6
2	2	5
3	3	3

= A

payoffs to player I

Best responses to mixed strategy of player 2

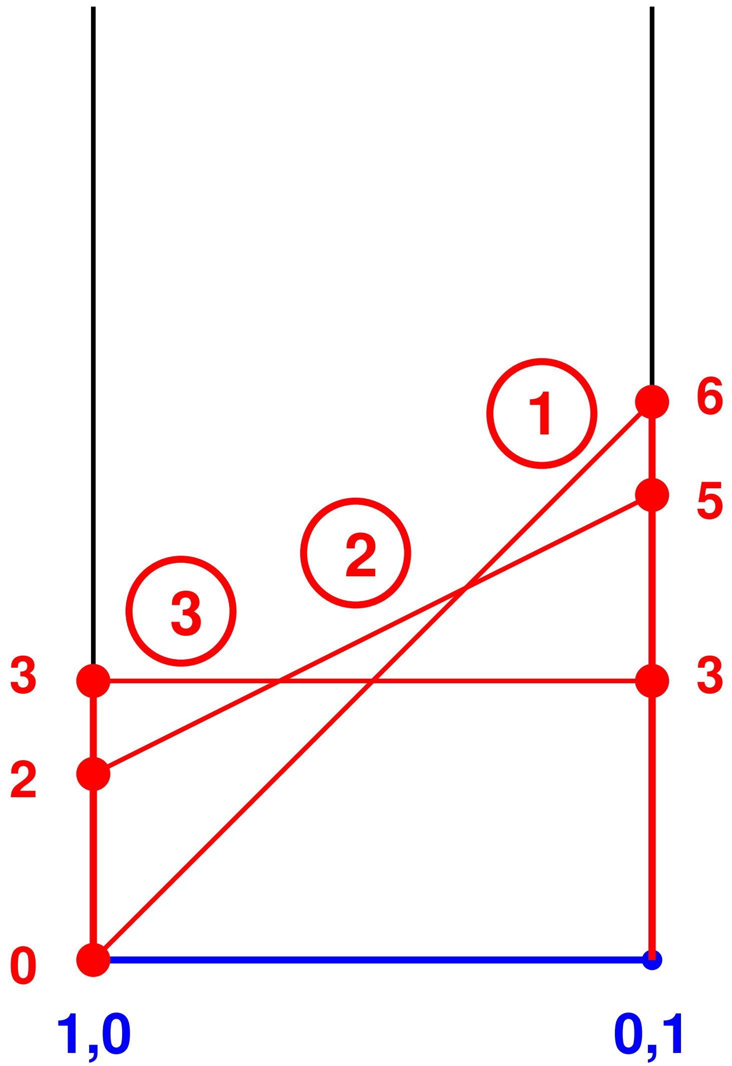


	4	5
1	0	6
2	2	5
3	3	3

= A

payoffs to
player I

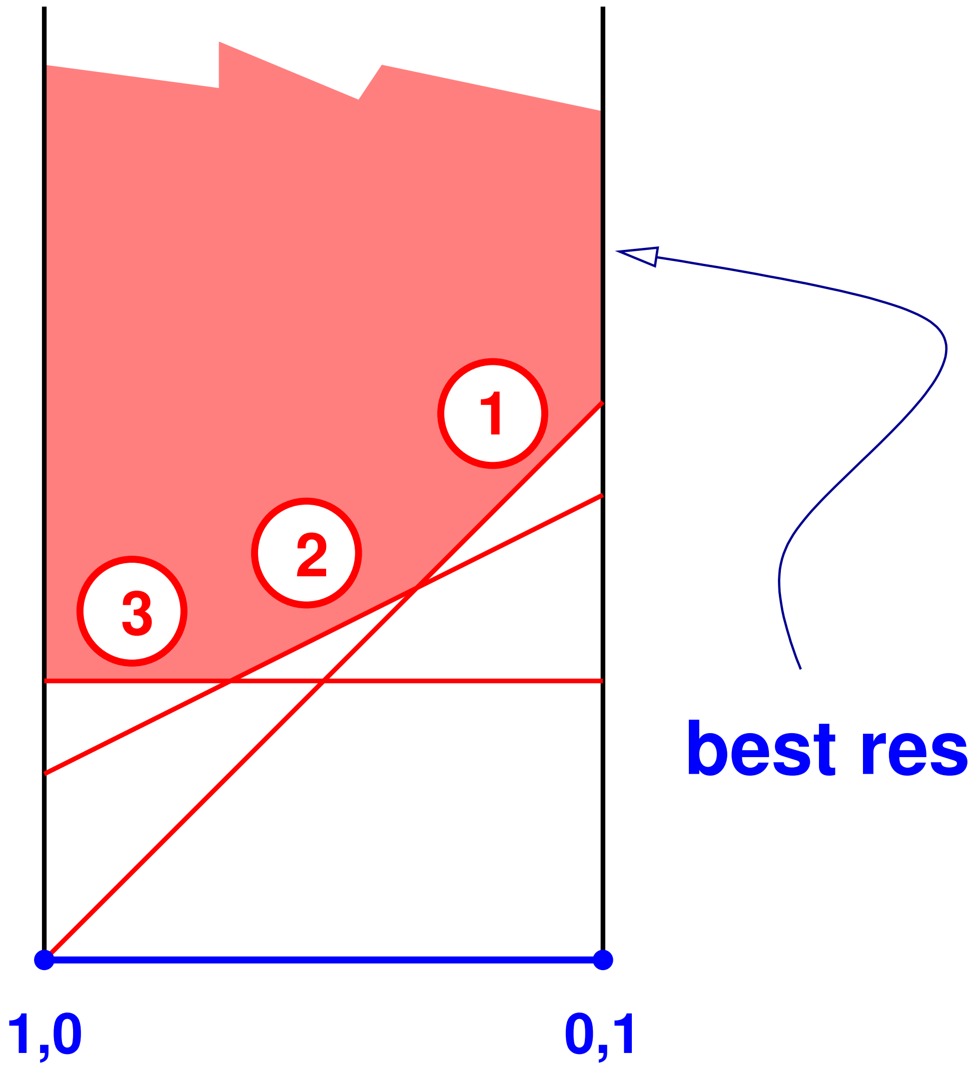
Best responses to mixed strategy of player 2



	4	5	
1	0	6	
2	2	5	= A
3	3	3	

payoffs to
player I

Best responses to mixed strategy of player 2

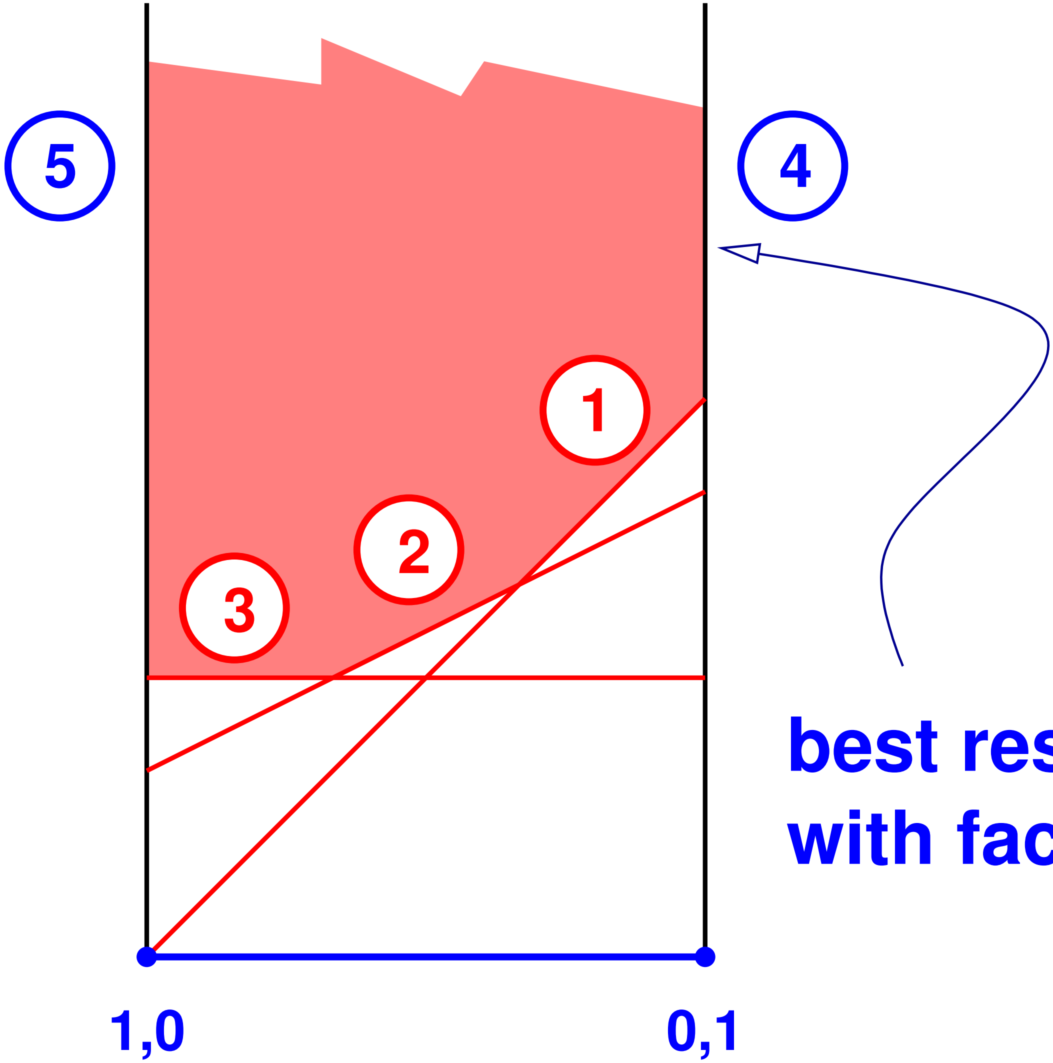


	4	5	
1	0	6	
2	2	5	= A
3	3	3	

payoffs to player I

best response polyhedron

Best responses to mixed strategy of player 2



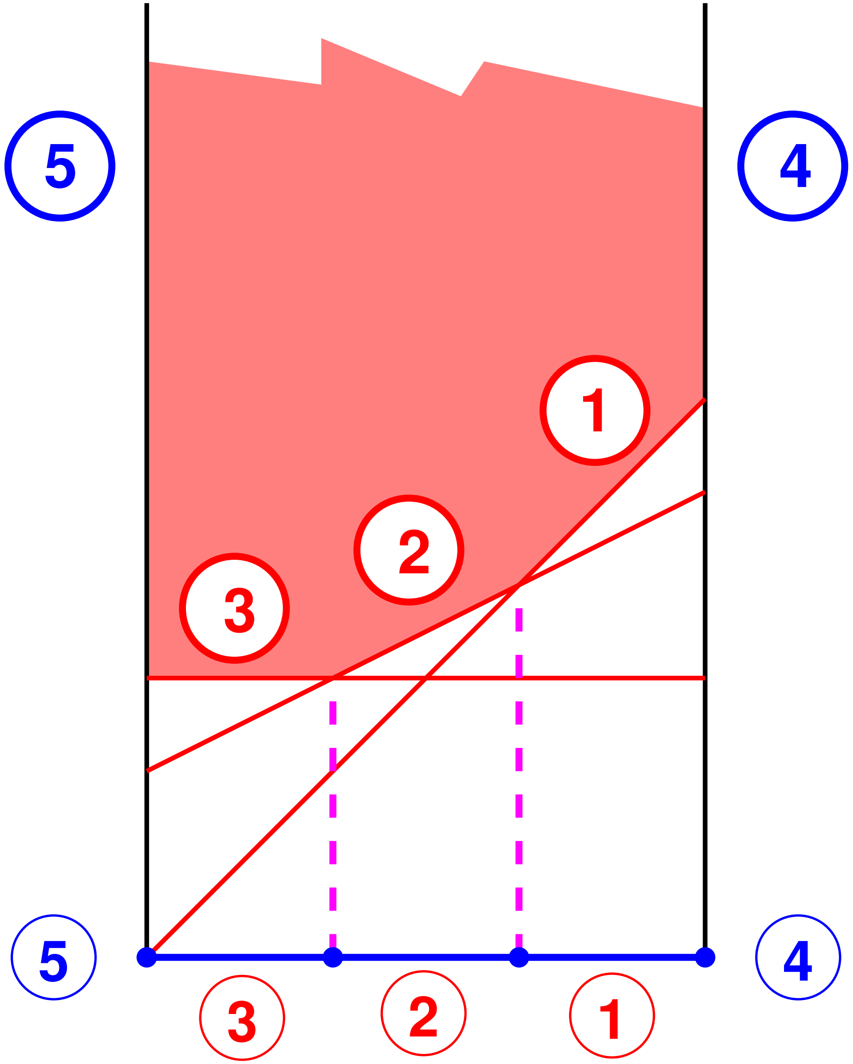
	4	5
1	0	6
2	2	5
3	3	3

= A

payoffs to player I

best response polyhedron with facet labels

Best responses to mixed strategy of player 2



	4	5	
1	0	6	
2	2	5	= A
3	3	3	

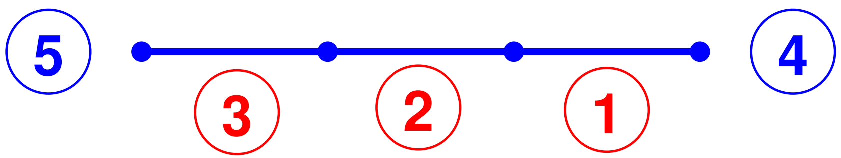
payoffs to
player I

Best responses to mixed strategy of player 2

	4	5
1	0	6
2	2	5
3	3	3

= A

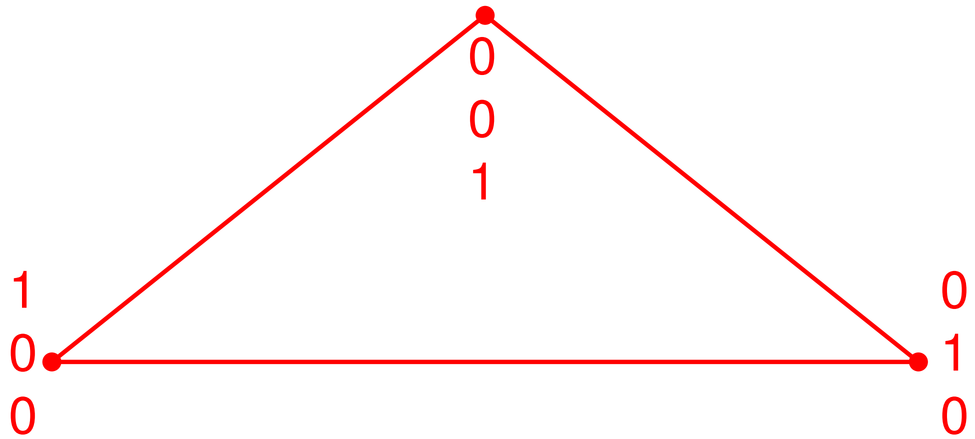
payoffs to
player I



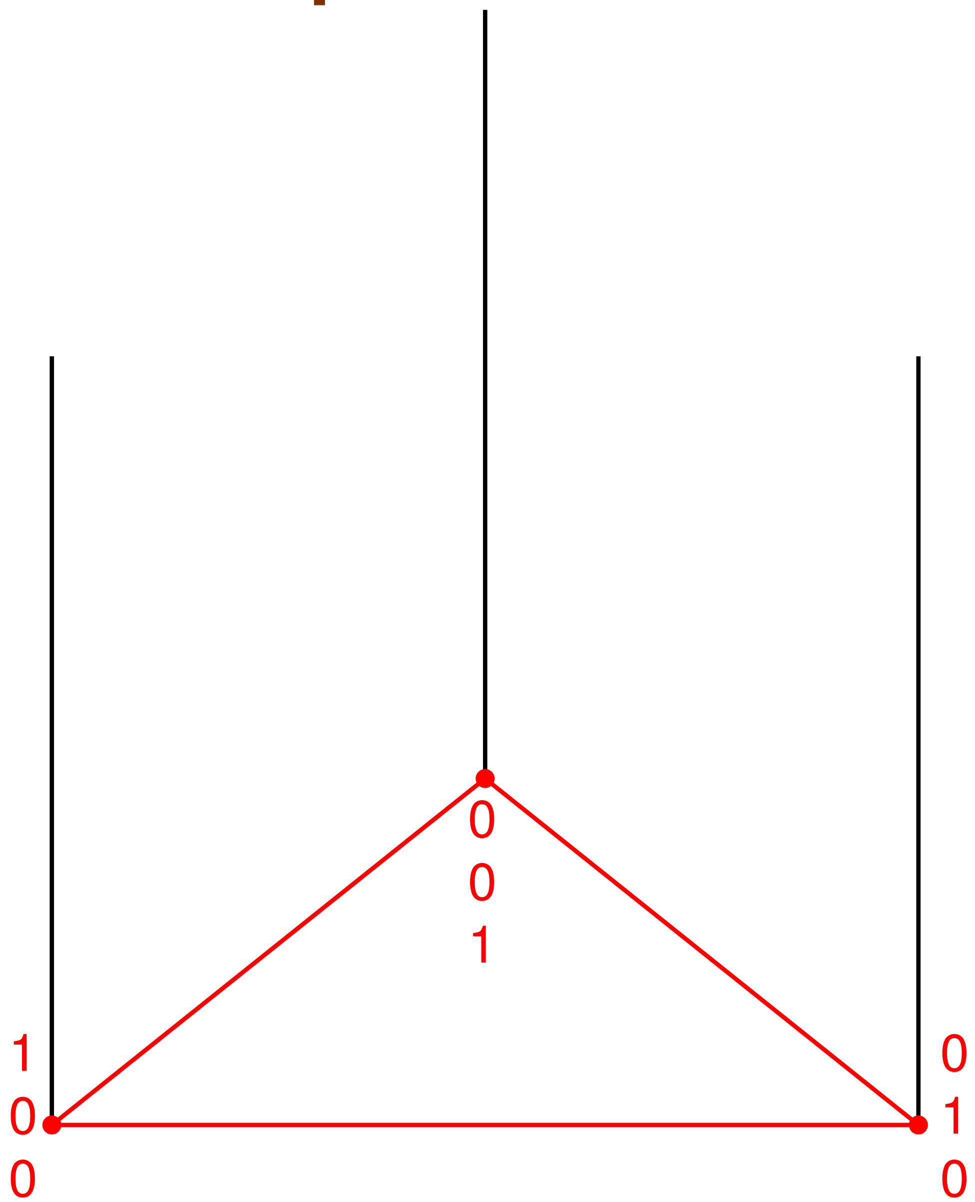
Best responses to mixed strategy of player 1

	4	5	
1	2	1	
2	1	3	= B
3	4	3	

payoffs to
player II



Best responses to mixed strategy of player 1

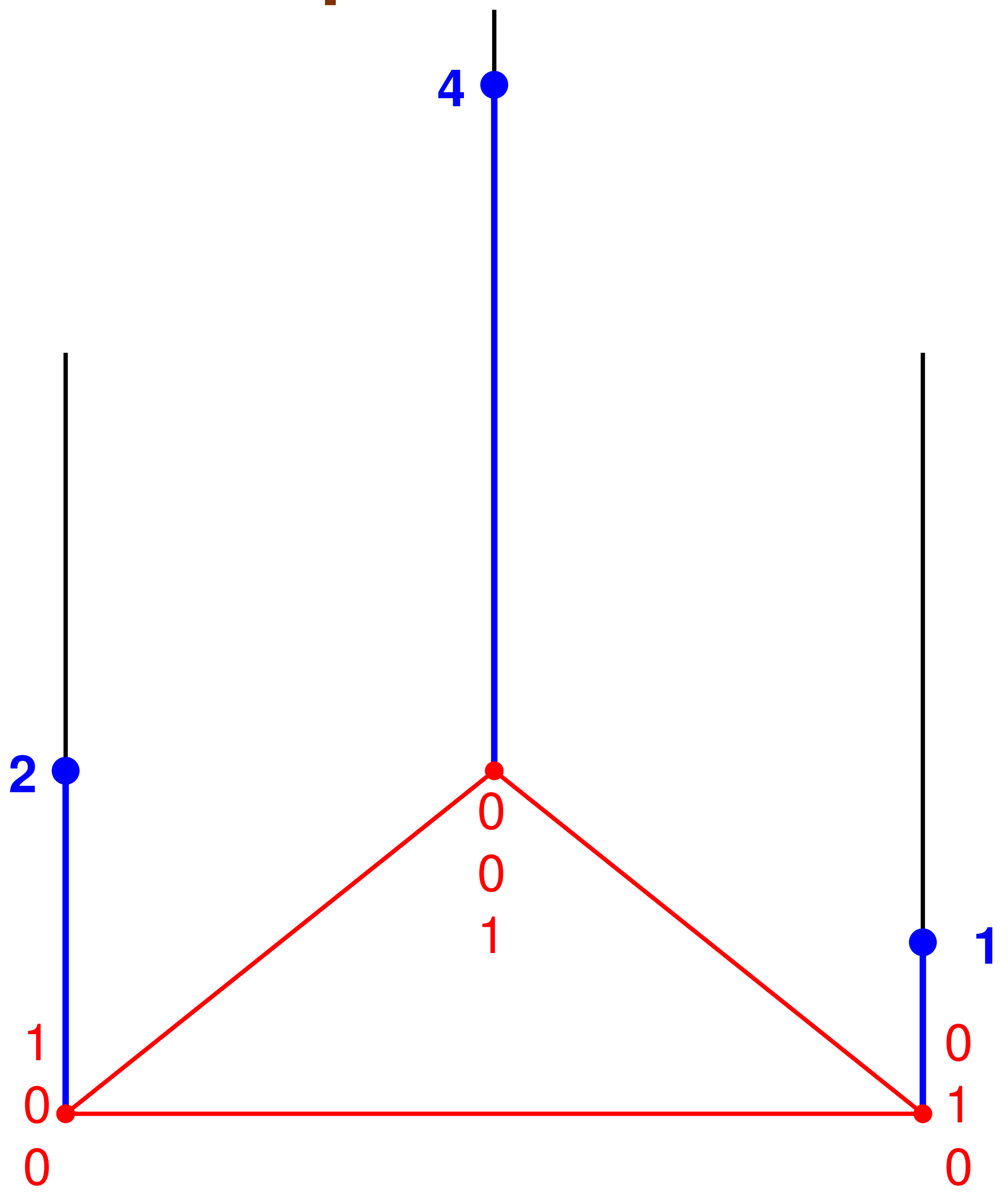


	4	5
1	2	1
2	1	3
3	4	3

= B

payoffs to
player II

Best responses to mixed strategy of player 1

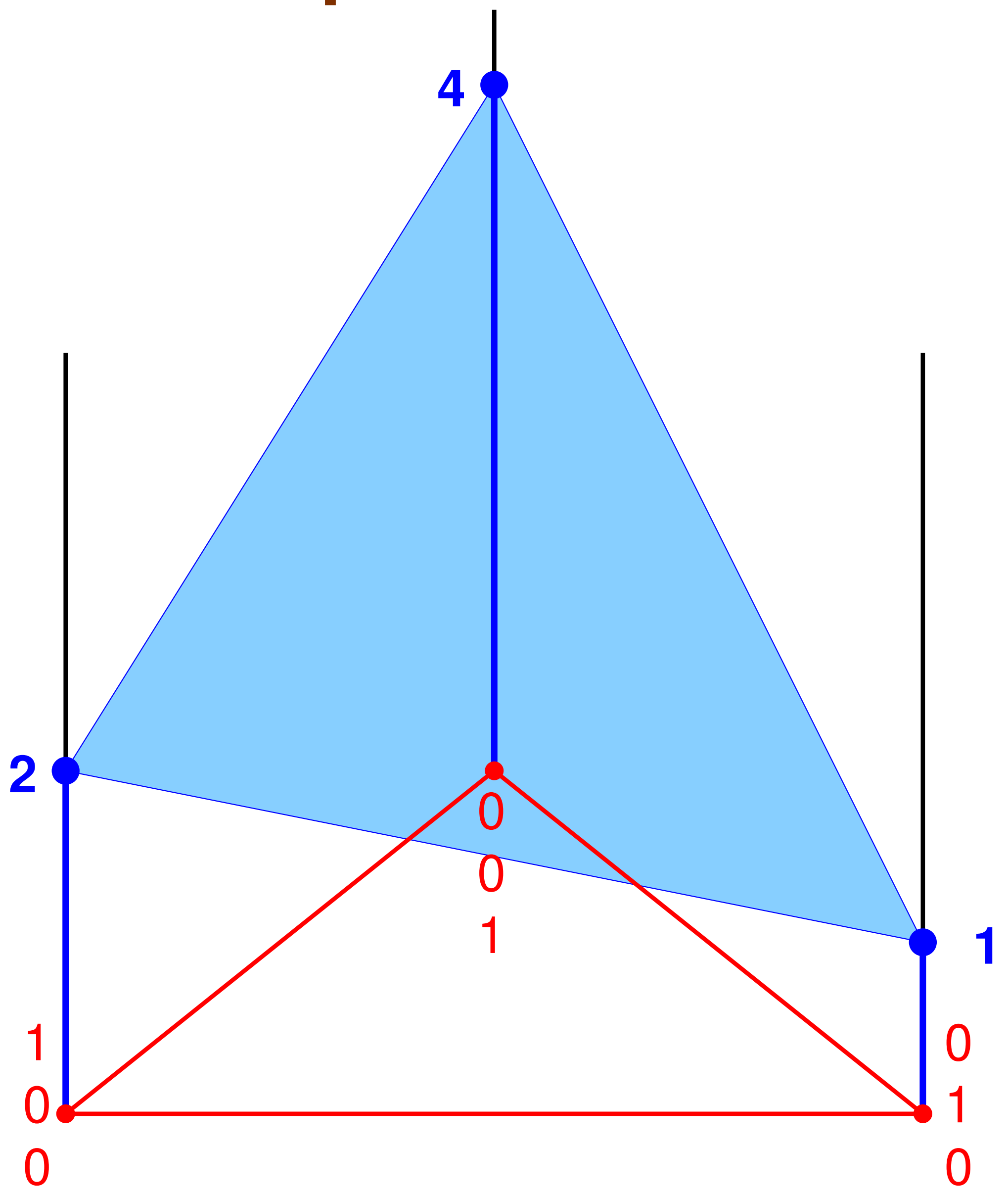


	4	5
1	2	1
2	1	3
3	4	3

= B

payoffs to
player II

Best responses to mixed strategy of player 1

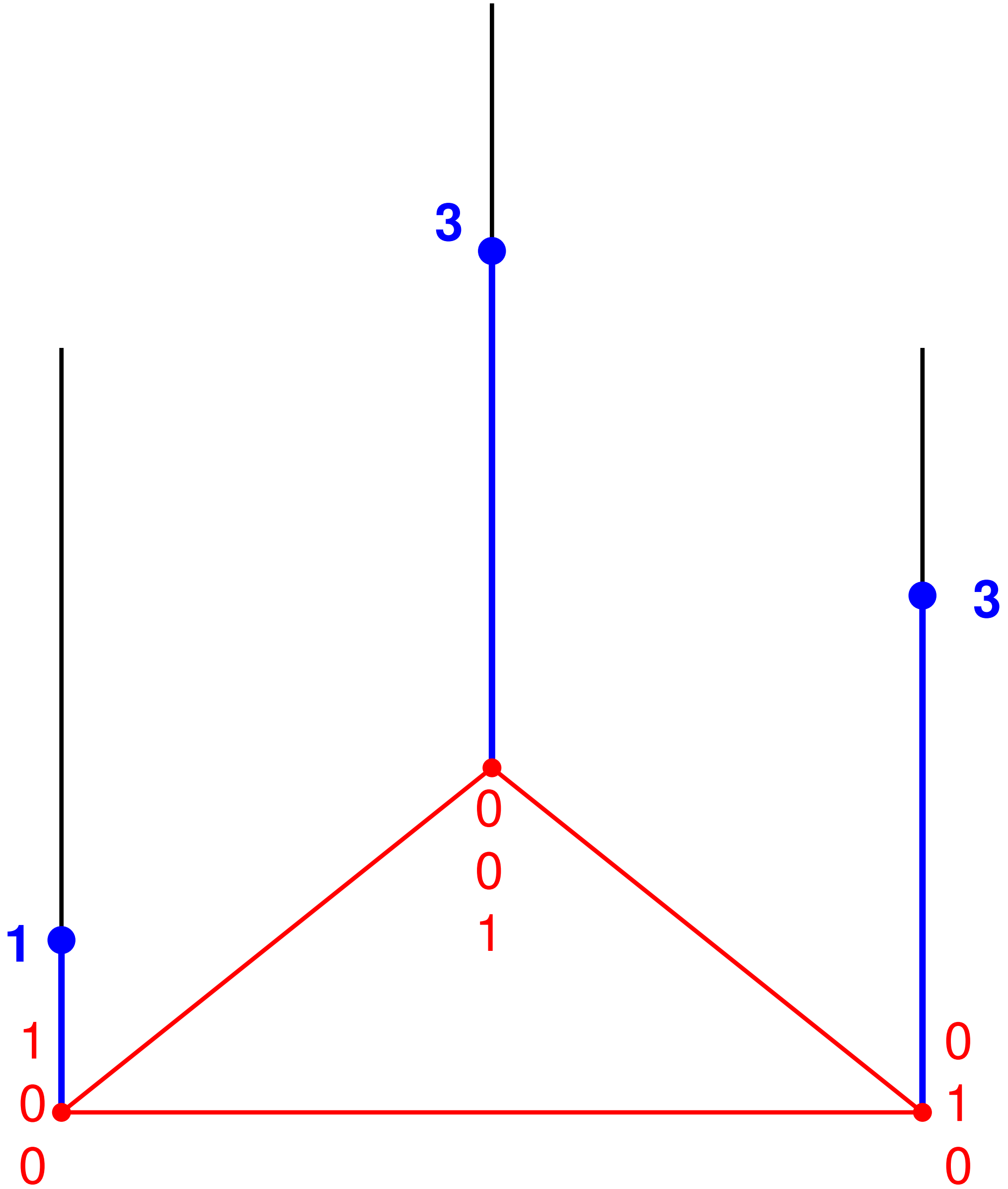


	4	5
1	2	1
2	1	3
3	4	3

= B

payoffs to
player II

Best responses to mixed strategy of player 1

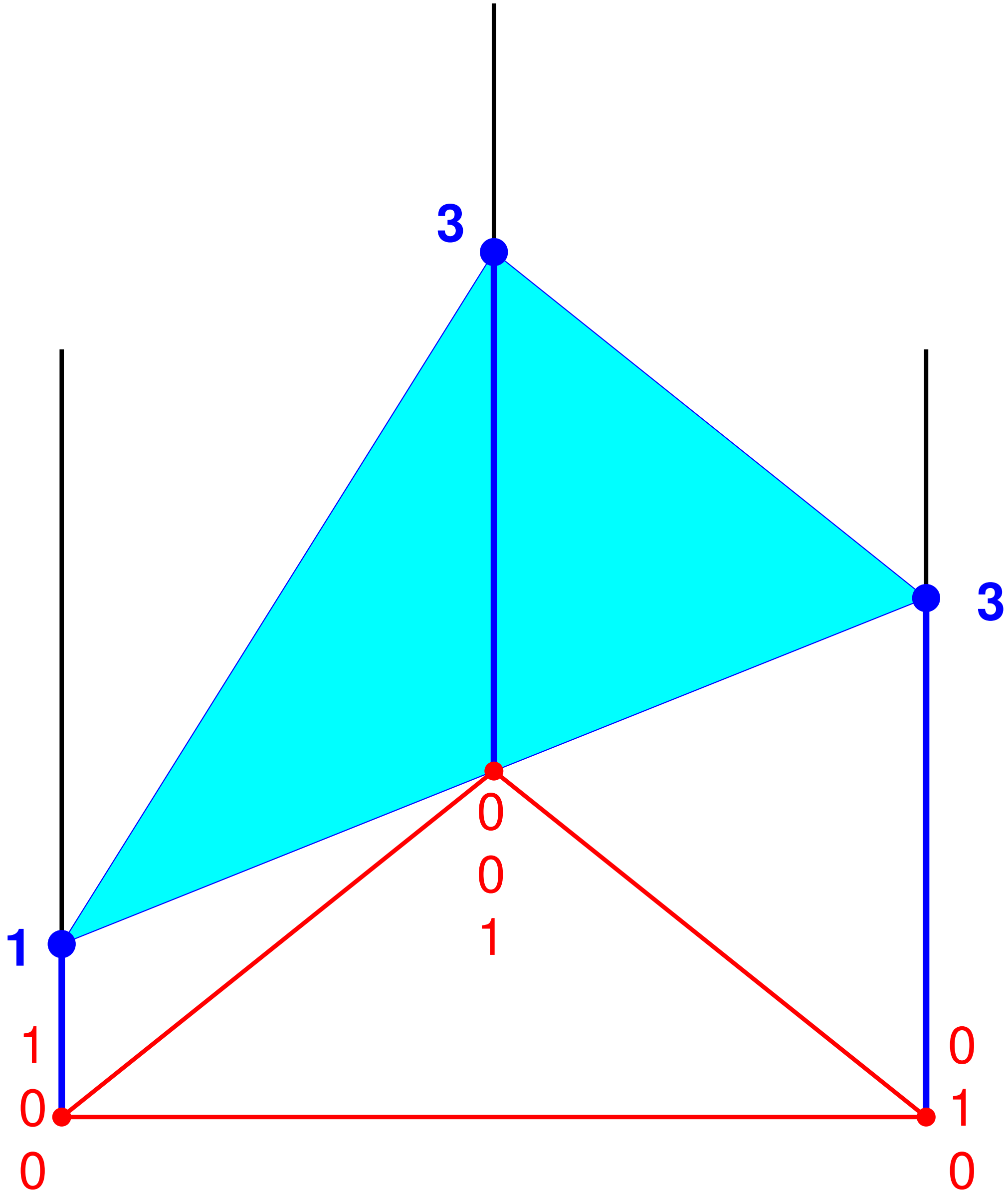


	4	5
1	2	1
2	1	3
3	4	3

= B

payoffs to player II

Best responses to mixed strategy of player 1

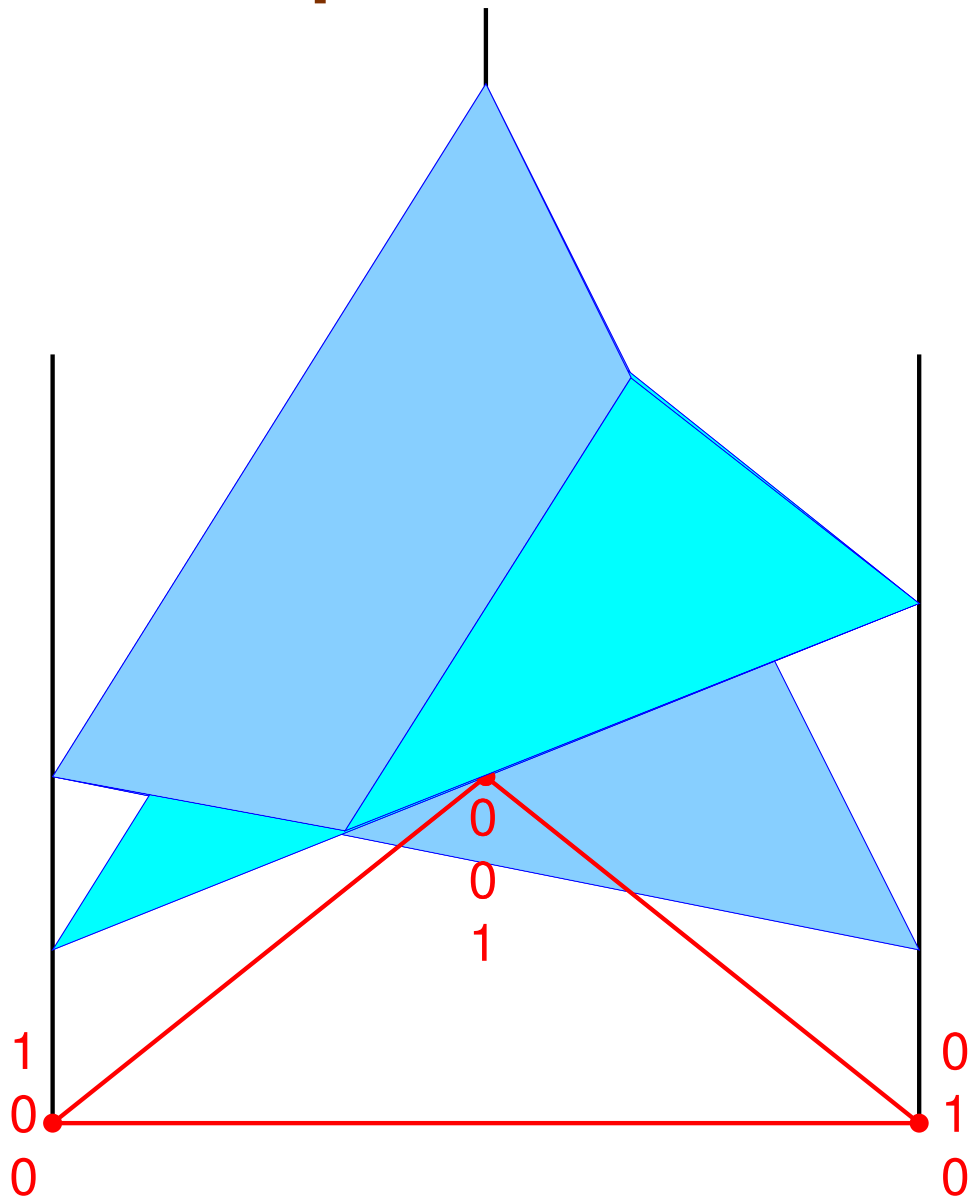


	4	5
1	2	1
2	1	3
3	4	3

= B

payoffs to
player II

Best responses to mixed strategy of player 1

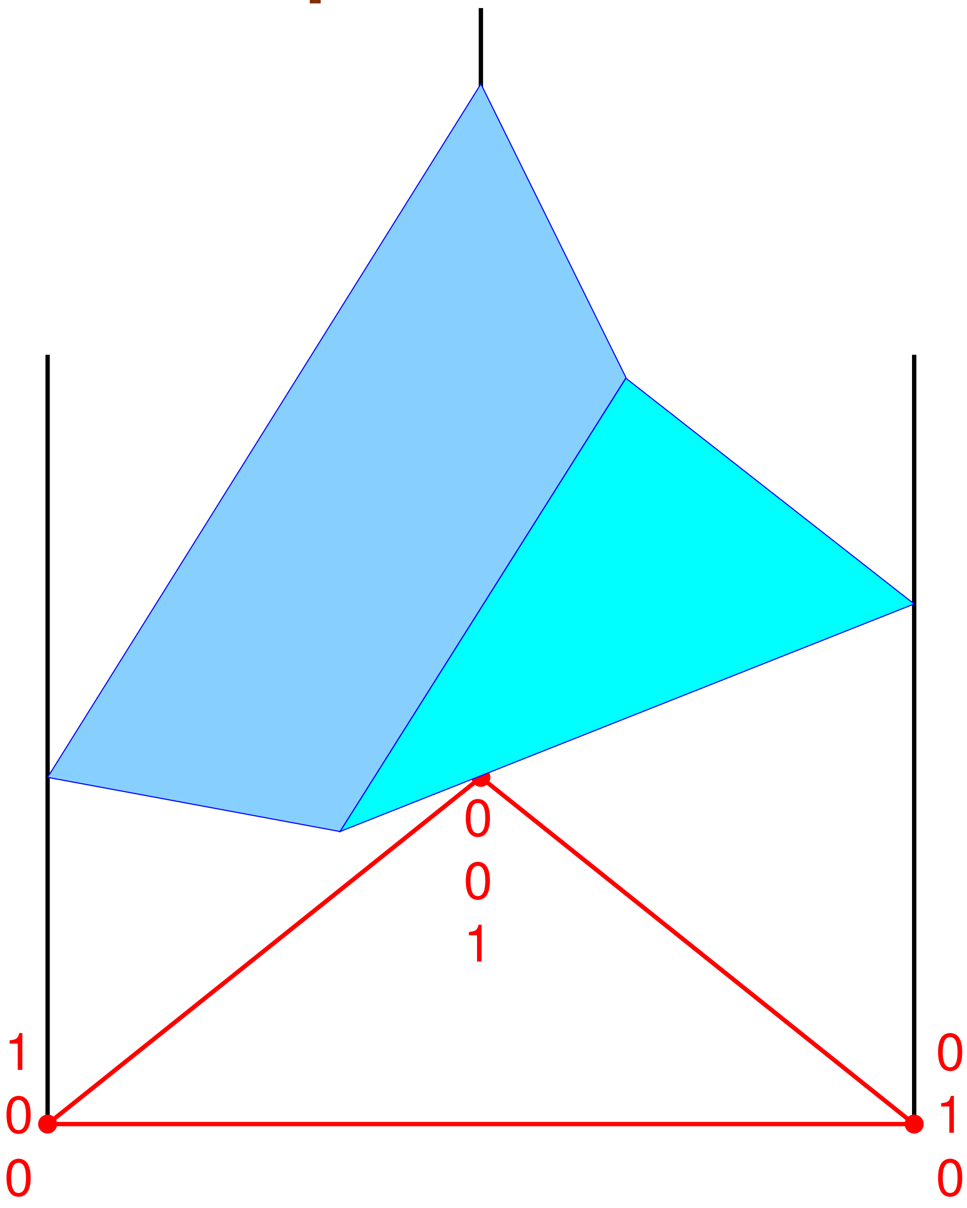


	4	5
1	2	1
2	1	3
3	4	3

= B

payoffs to
player II

Best responses to mixed strategy of player 1

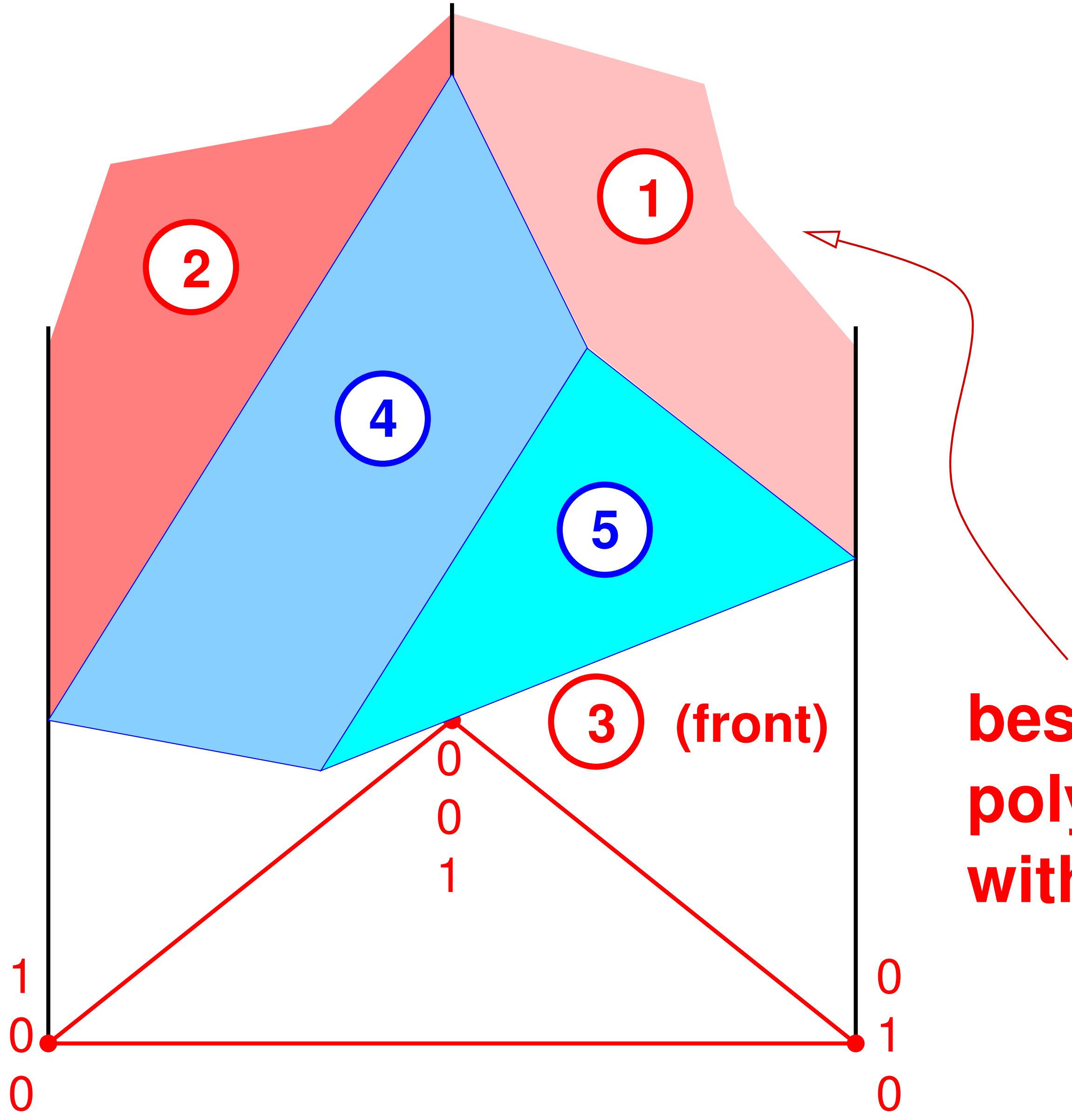


	4	5
1	2	1
2	1	3
3	4	3

= B

payoffs to
player II

Best responses to mixed strategy of player 1



	4	5
1	2	1
2	1	3
3	4	3

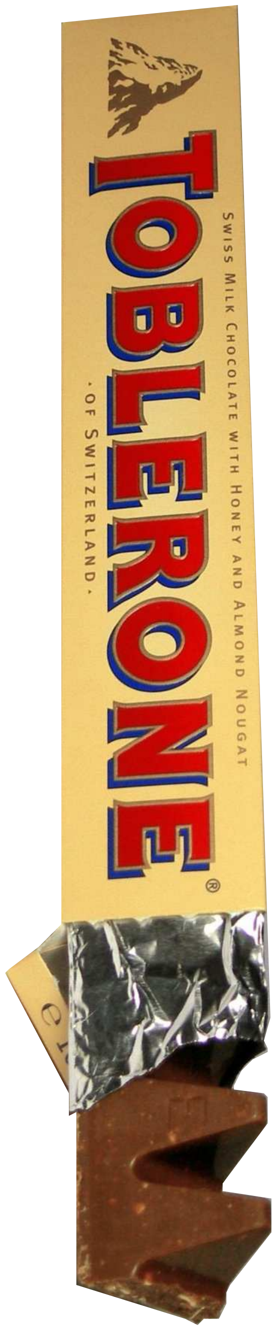
= B

payoffs to
player II

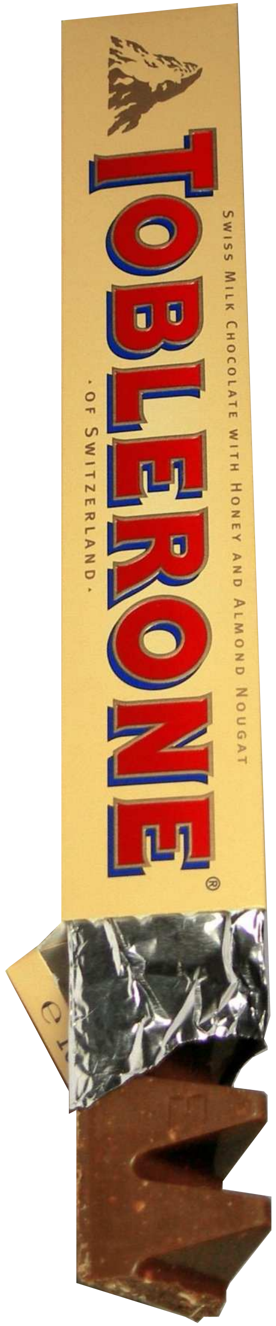
**best response
polyhedron
with facet labels**

Alternative view

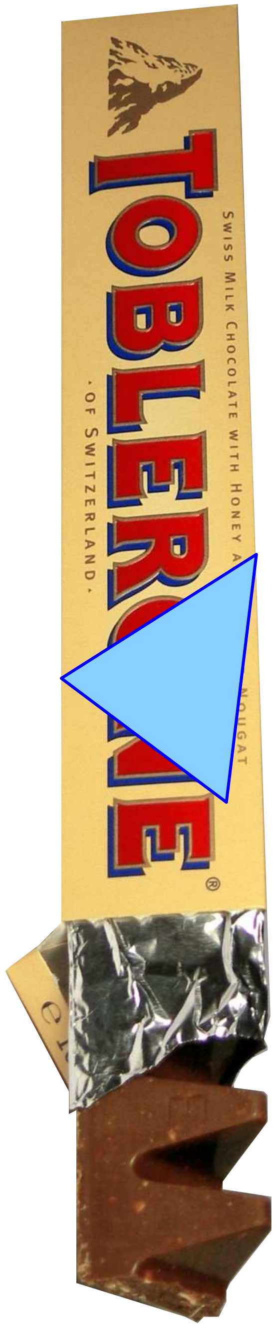
Chop off Toblerone prism



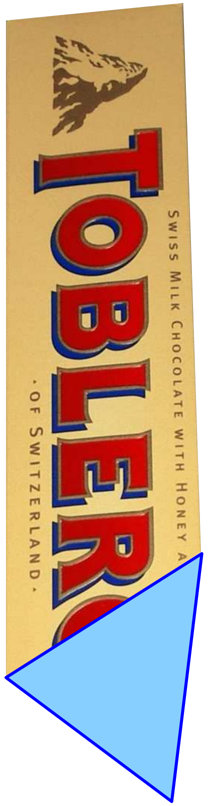
Chop off Toblerone prism



Chop off Toblerone prism

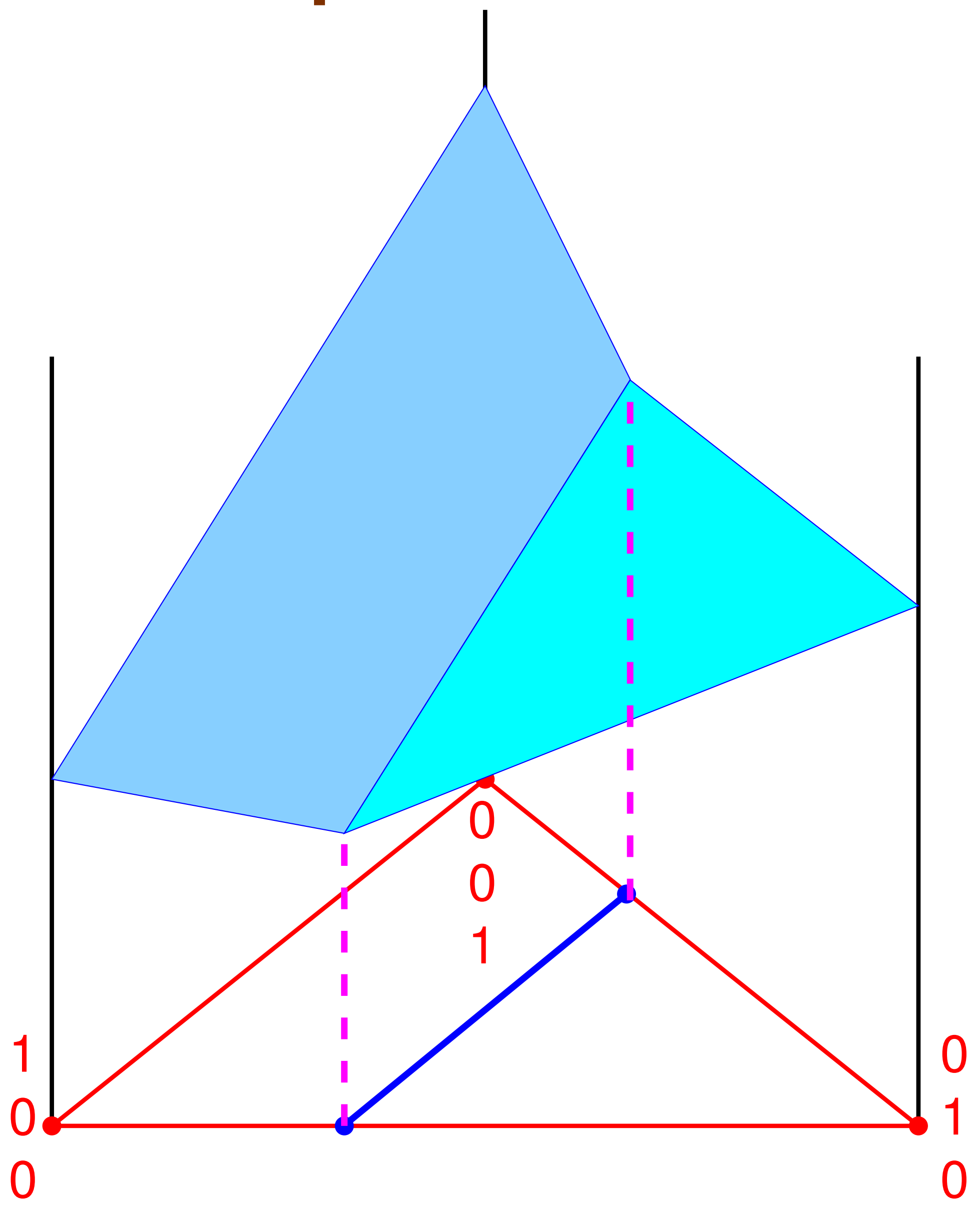


Chop off Toblerone prism



Chop off Toblerone prism

Best responses to mixed strategy of player 1



	4	5
1	2	1
2	1	3
3	4	3

= B

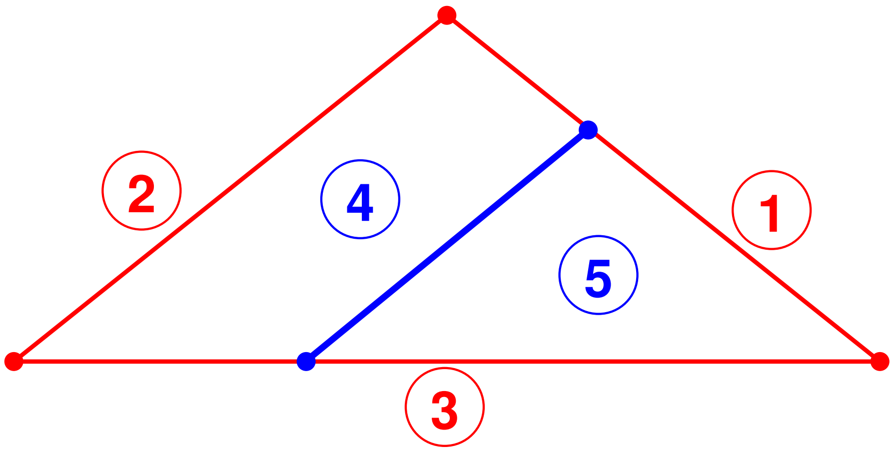
payoffs to player II

Best responses to mixed strategy of player 1

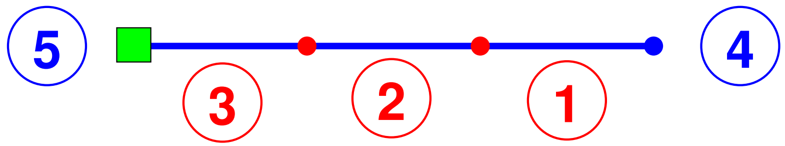
	4	5
1	2	1
2	1	3
3	4	3

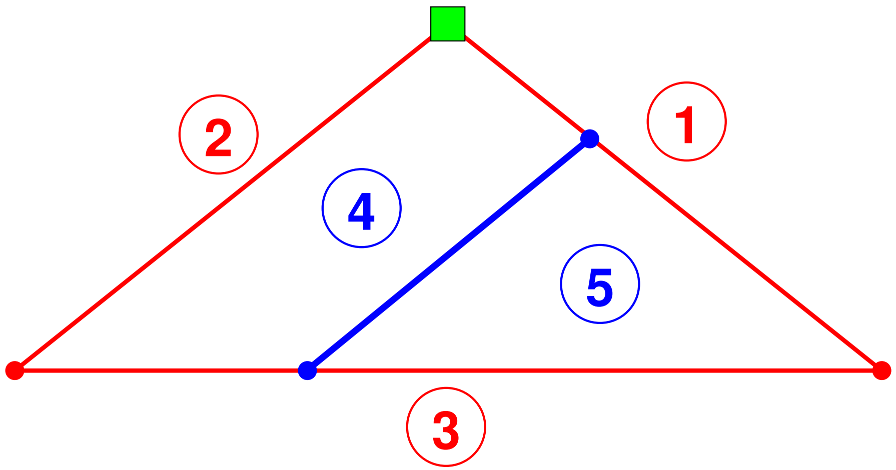
= B

payoffs to
player II

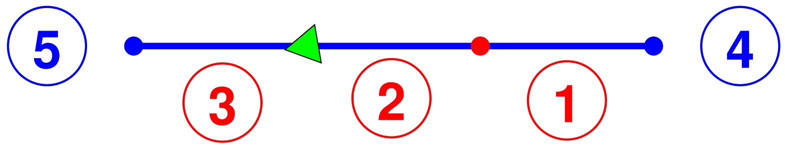


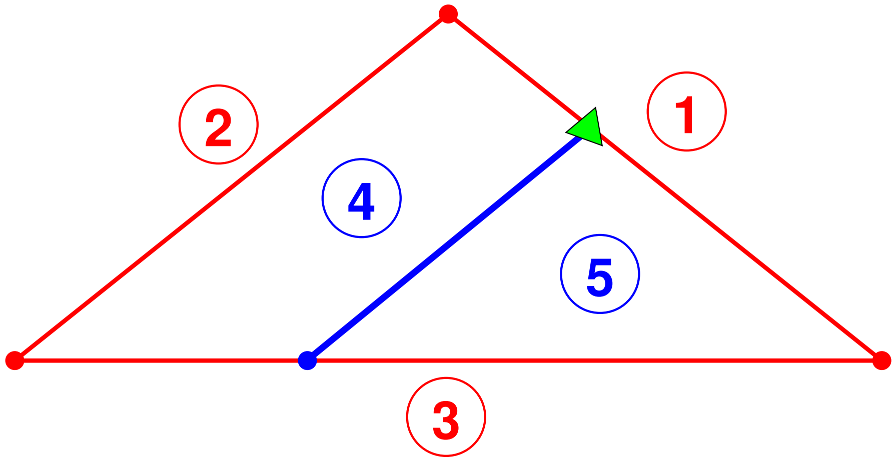
Equilibrium = completely labeled strategy pair



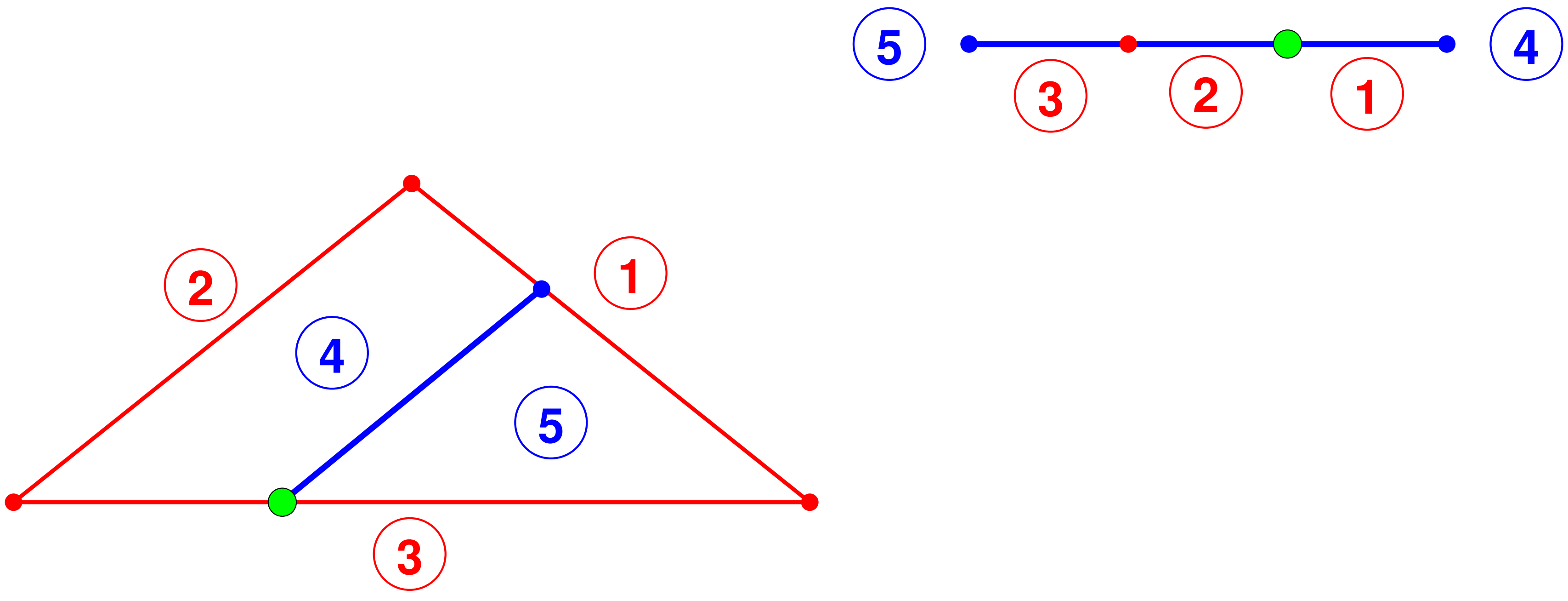


Equilibrium = completely labeled strategy pair





Equilibrium = completely labeled strategy pair



Constructing games using geometry

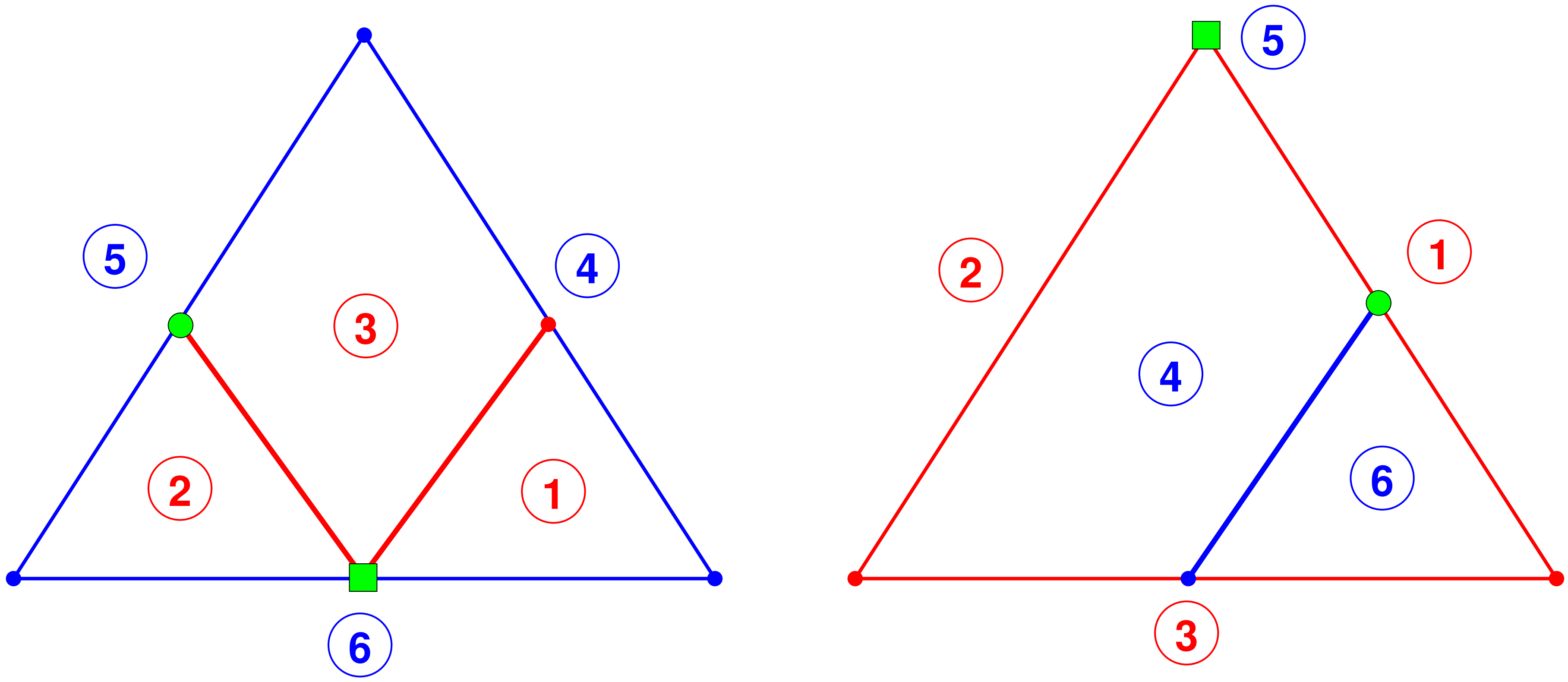
low dimension: 2, 3, (4) pure strategies:

subdivide mixed strategy simplex into
response regions, label suitably

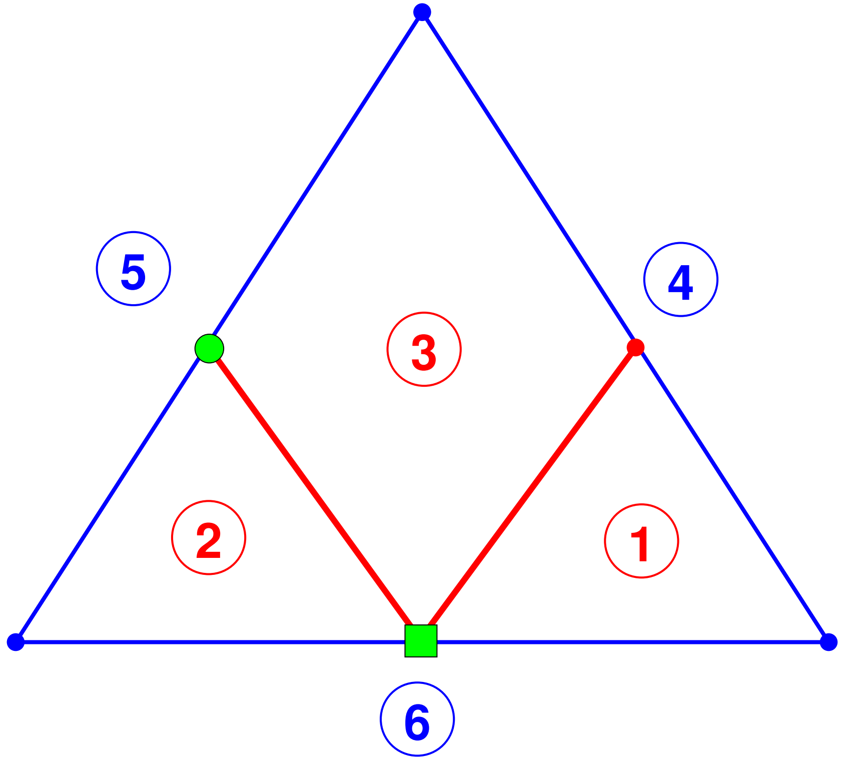
high dimension:

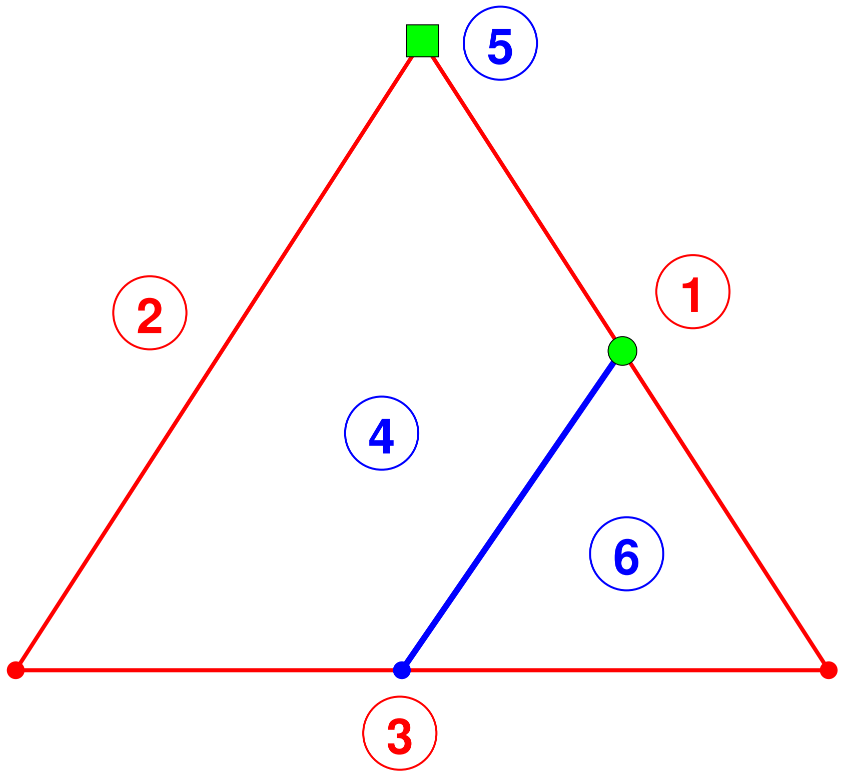
use polytopes with known combinatorial structure
e.g. for constructing games with many equilibria,
or long Lemke-Howson computations
[Savani & von Stengel, *FOCS 2004*,
Econometrica 2006]

Construct isolated non-quasi-strict equilibrium



Construct isolated non–quasi–strict equilibrium

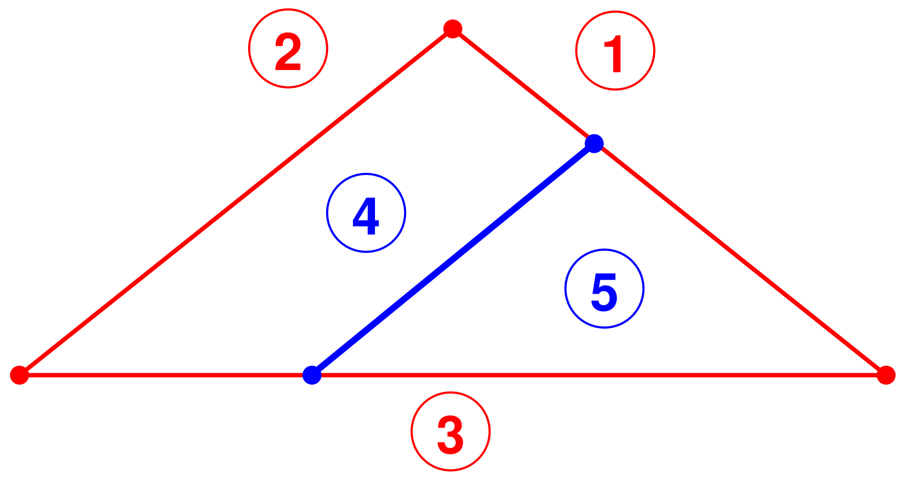


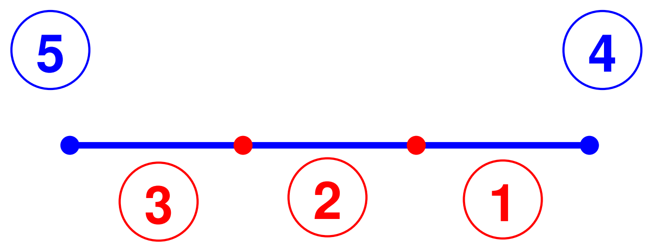


$$A = \begin{bmatrix} 0 & 2 & 0 \\ 2 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \begin{matrix} ① \\ ② \\ ③ \end{matrix}$$

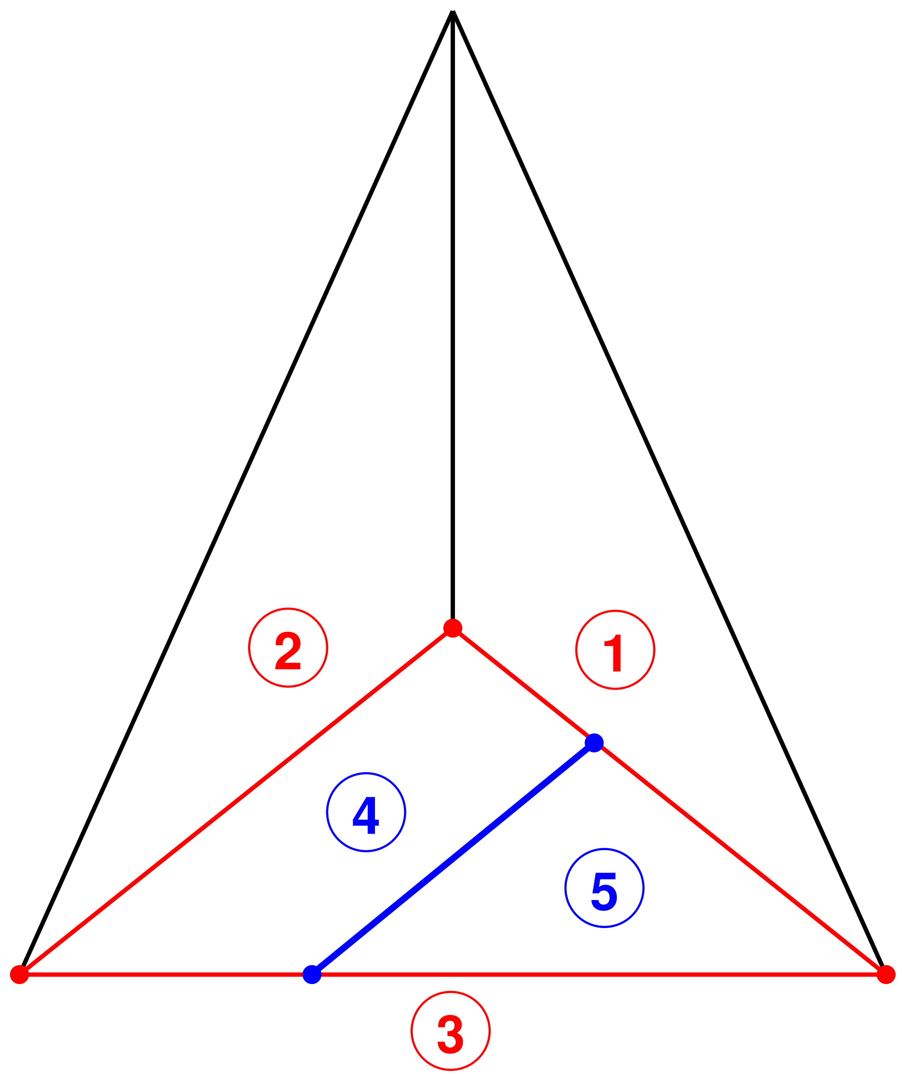
$$B = \begin{matrix} ④ & ⑤ & ⑥ \\ \end{matrix}$$
$$B = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 2 \\ 1 & 1 & 0 \end{bmatrix}$$

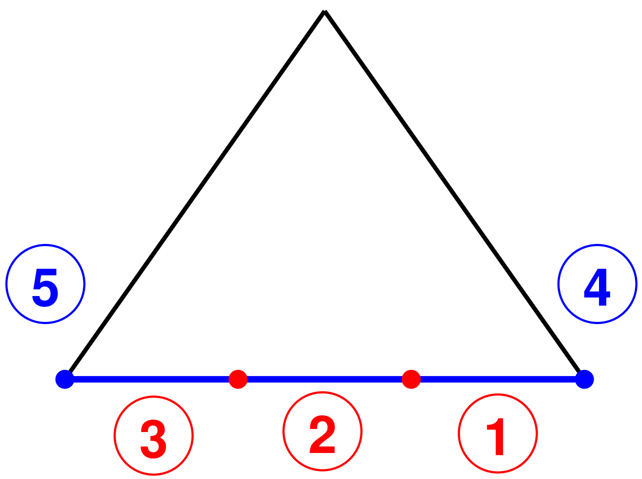
The Lemke–Howson algorithm



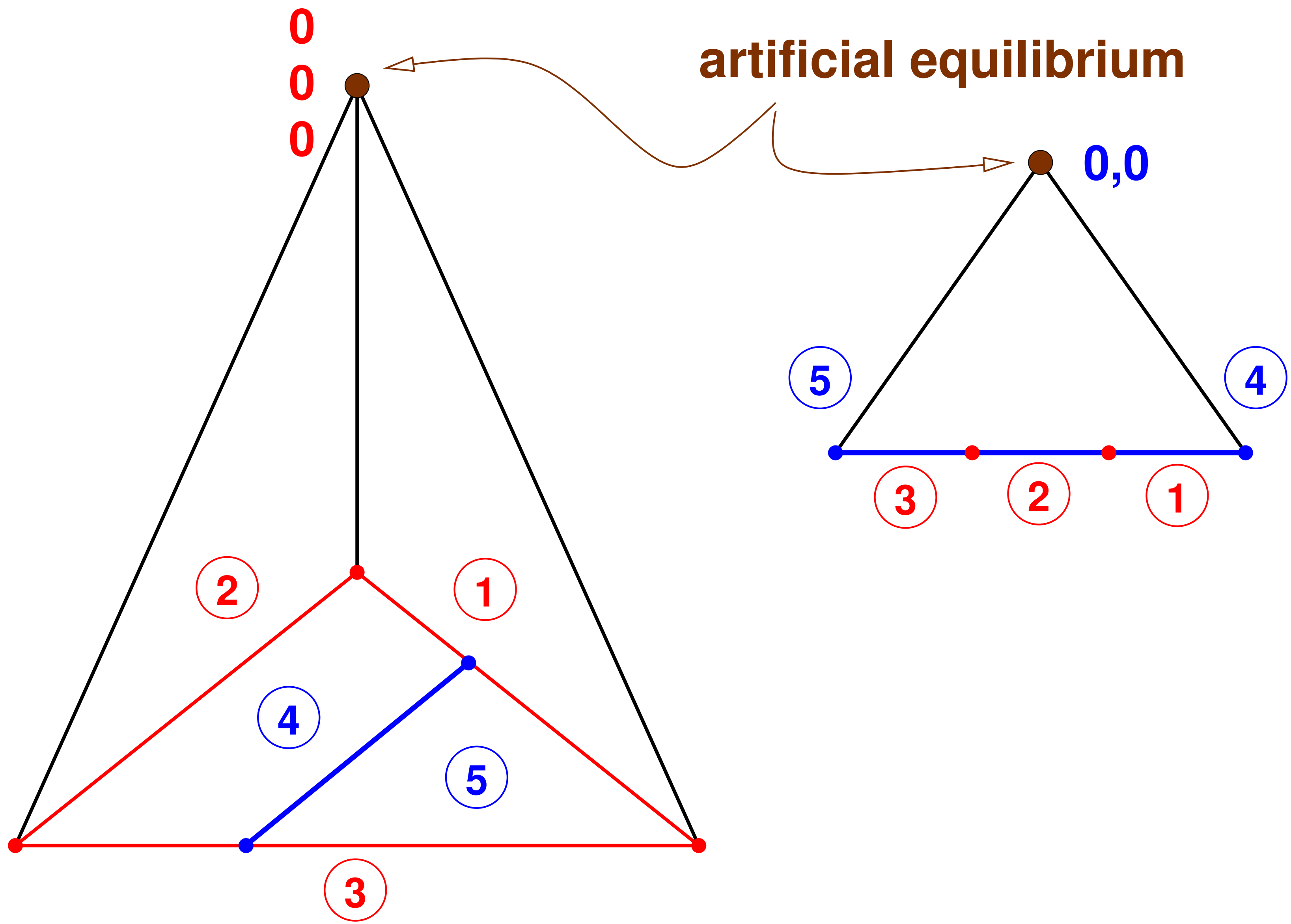


The Lemke–Howson algorithm

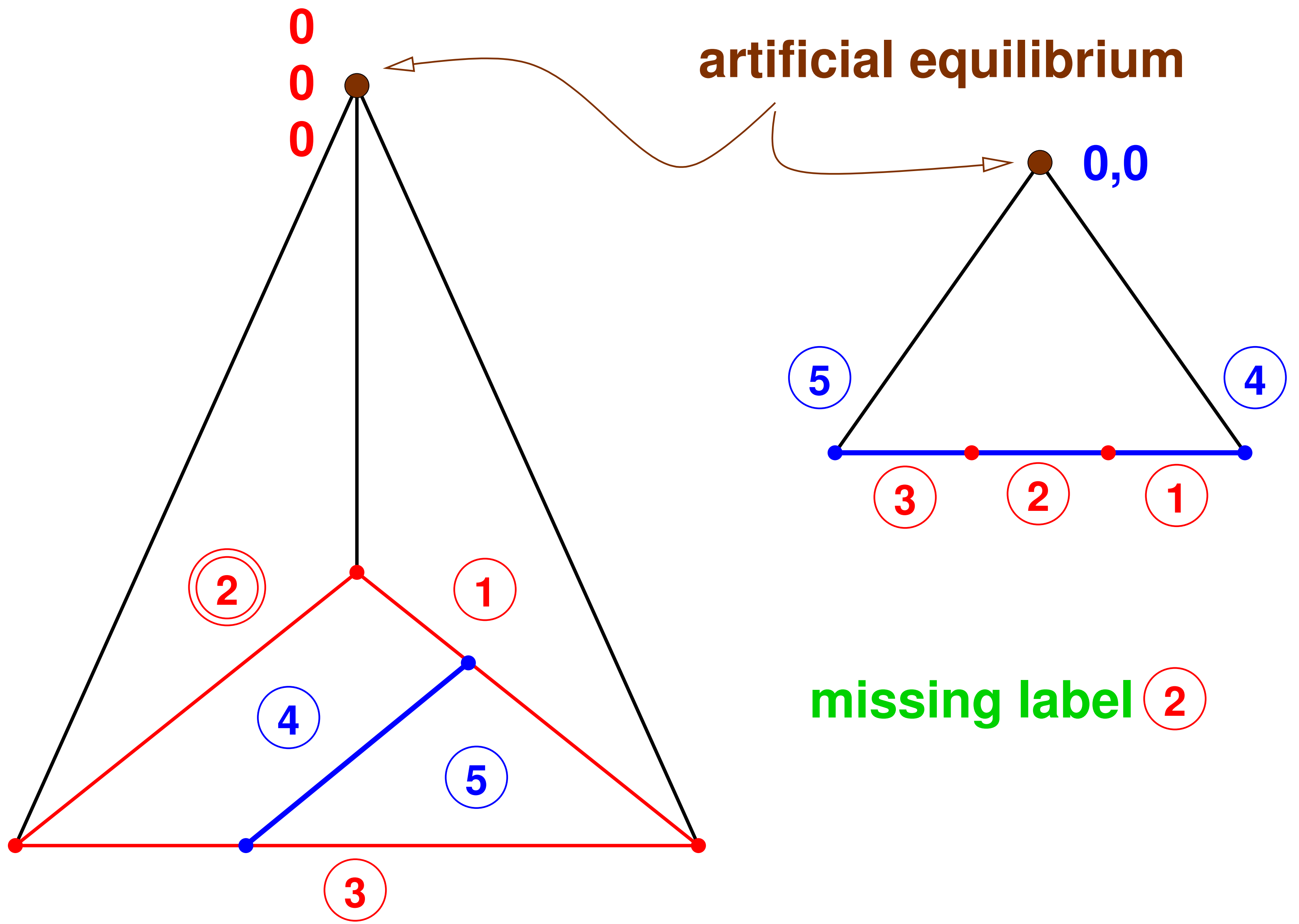




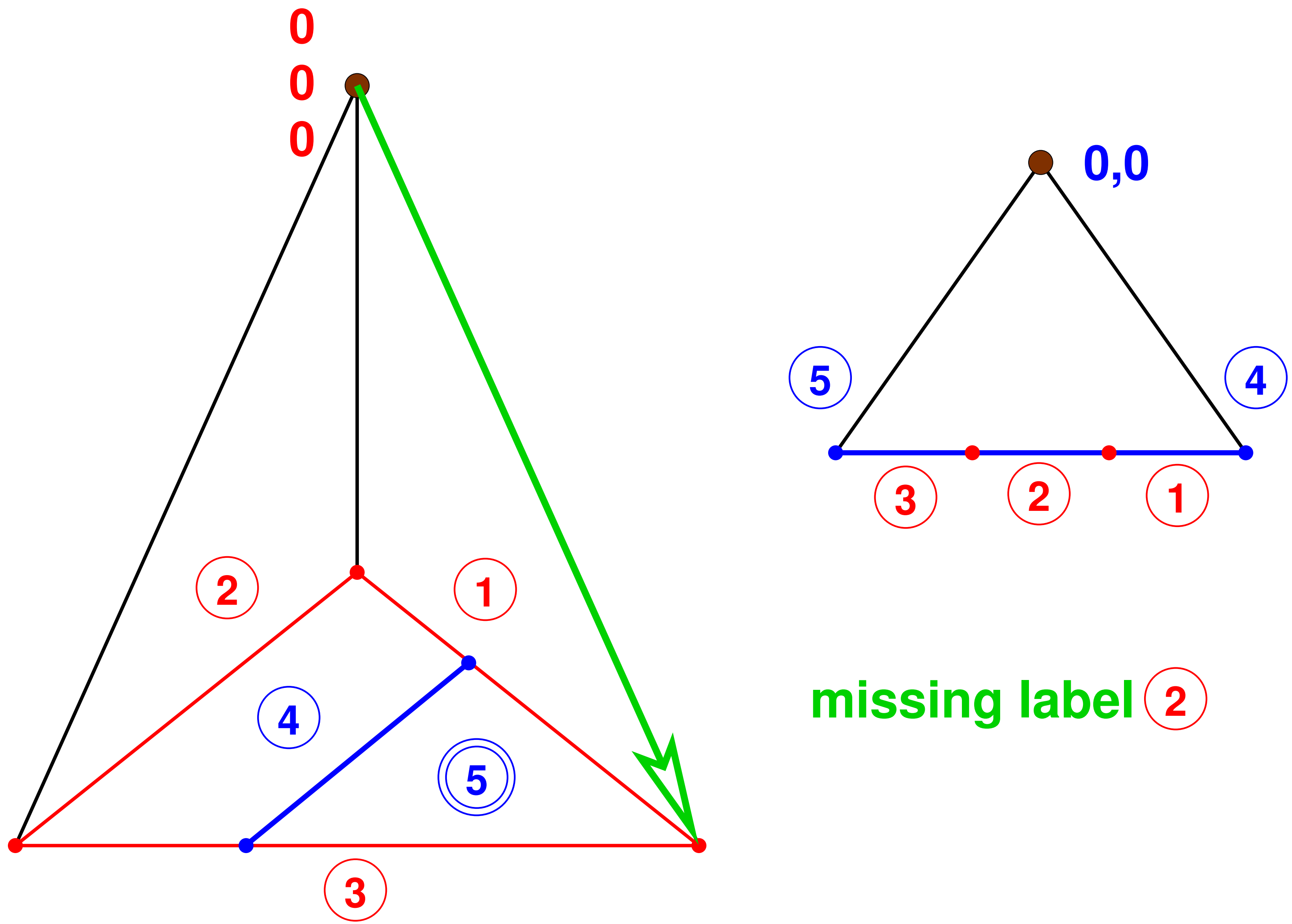
The Lemke–Howson algorithm



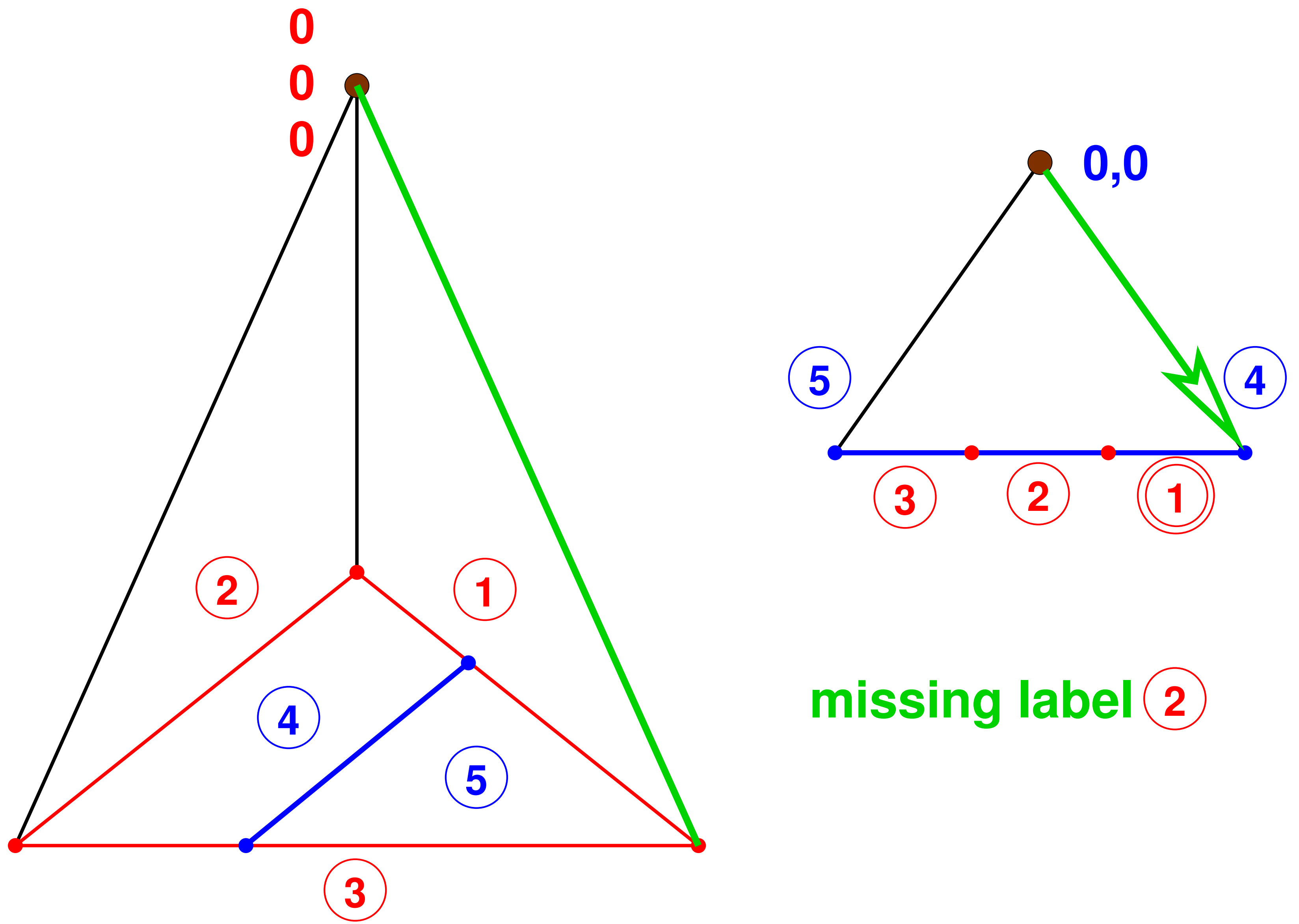
The Lemke–Howson algorithm



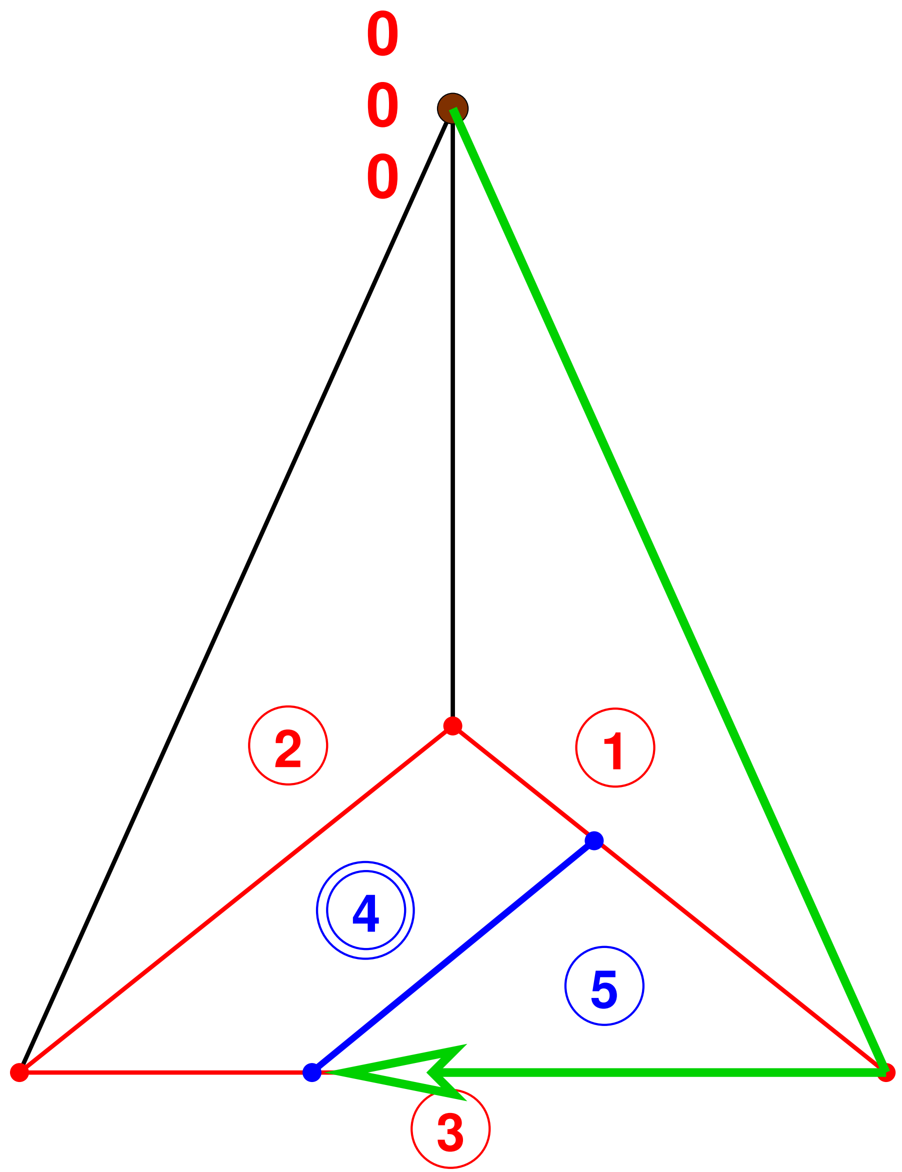
The Lemke–Howson algorithm

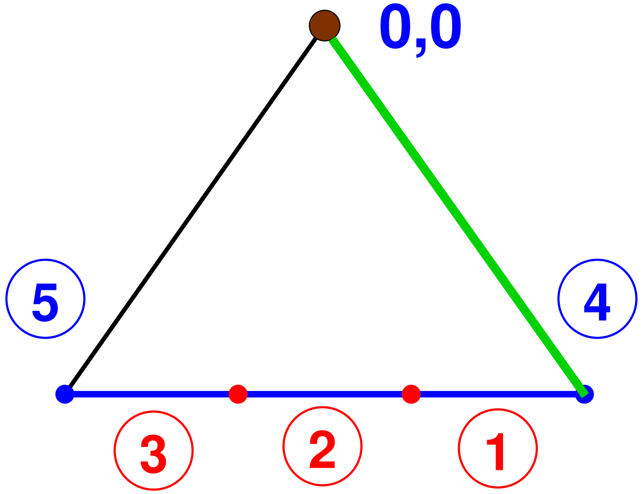


The Lemke–Howson algorithm



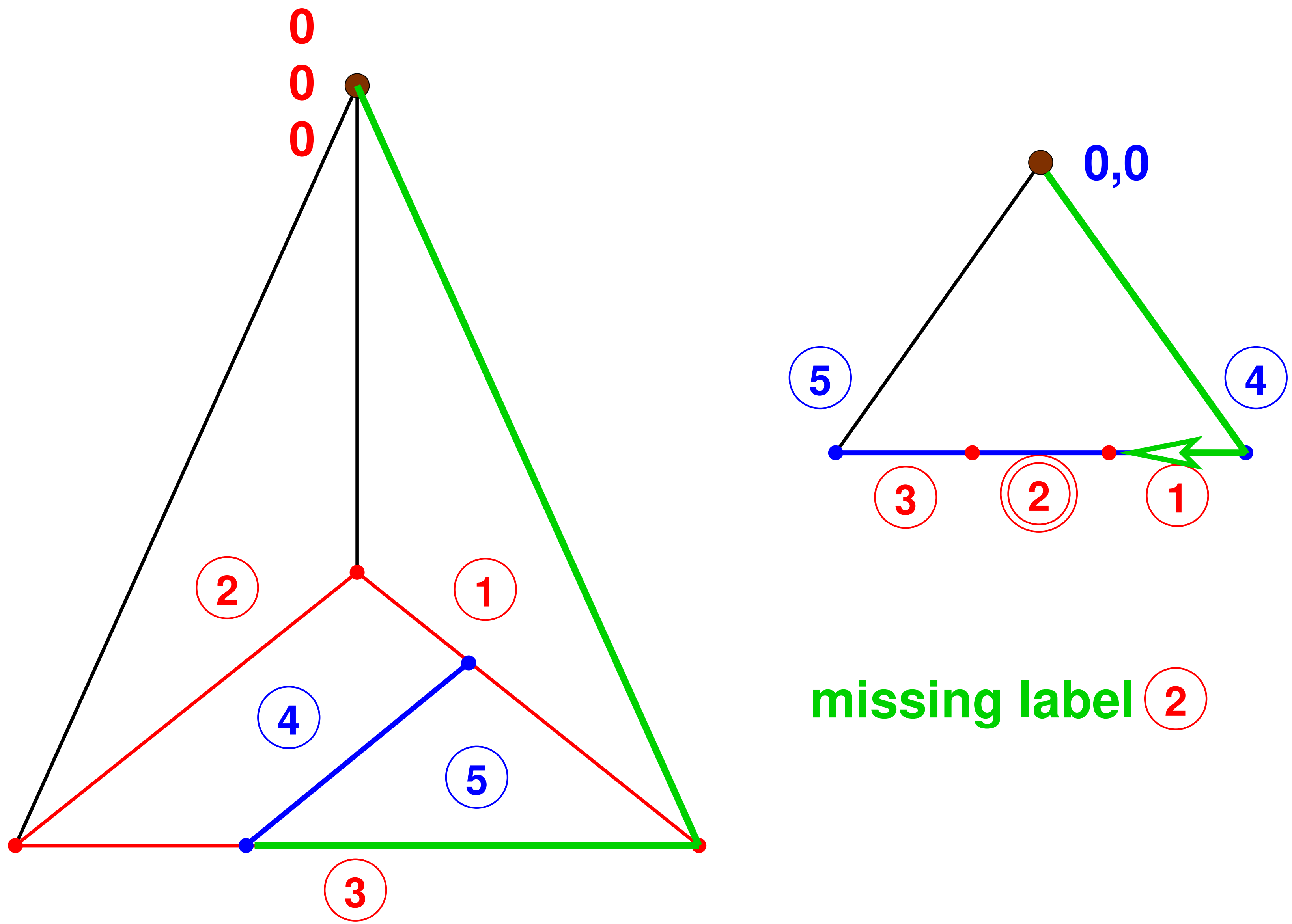
The Lemke–Howson algorithm





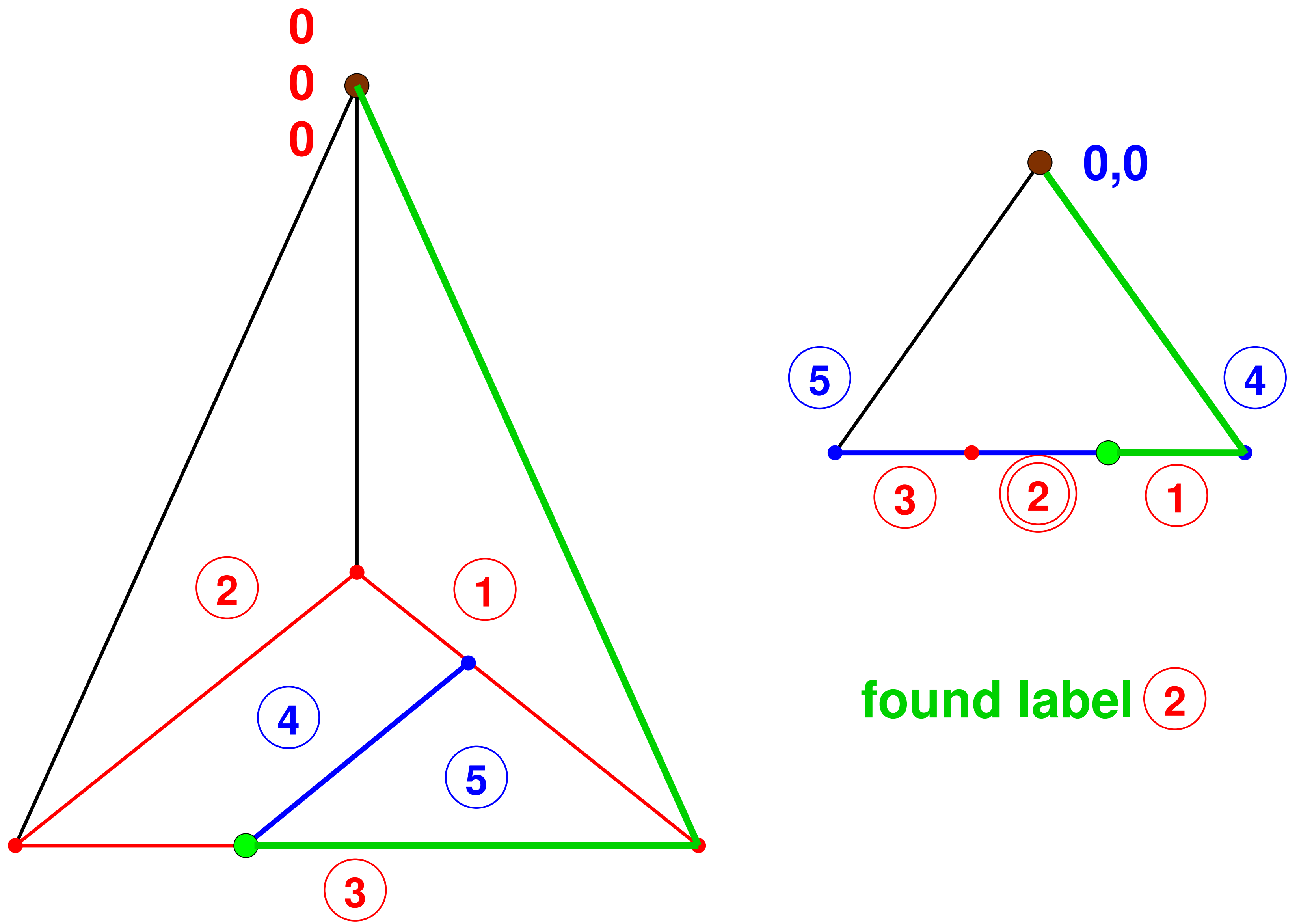
missing label 2

The Lemke–Howson algorithm



missing label 2

The Lemke–Howson algorithm



Why Lemke-Howson works

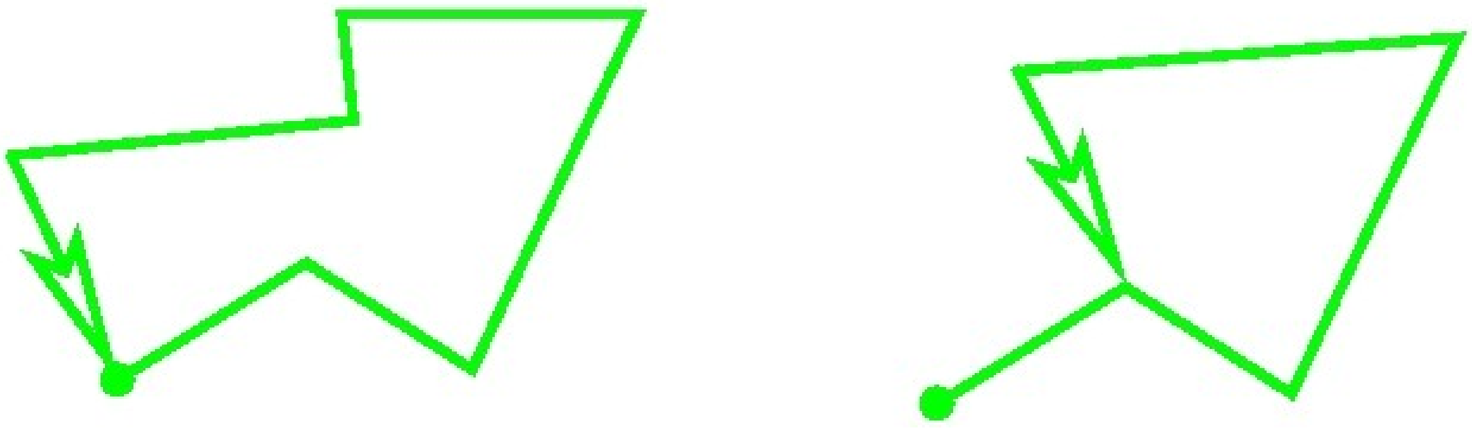
LH finds at least one Nash equilibrium because

- finitely many "vertices"

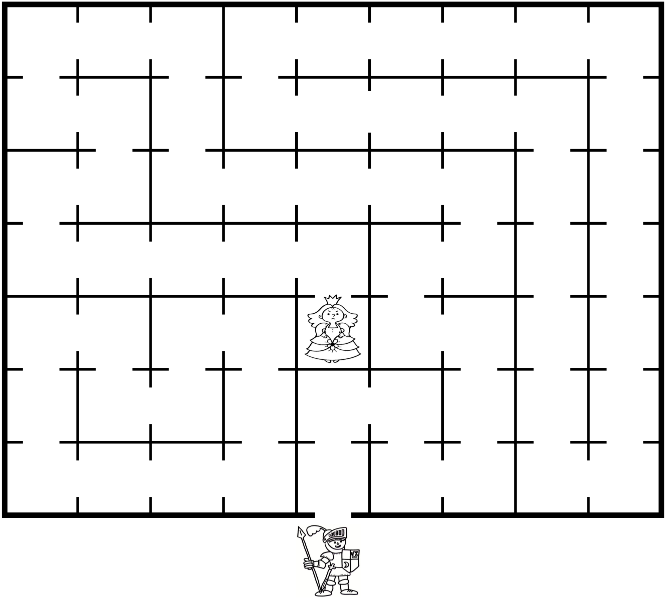
for nondegenerate (generic) games:

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

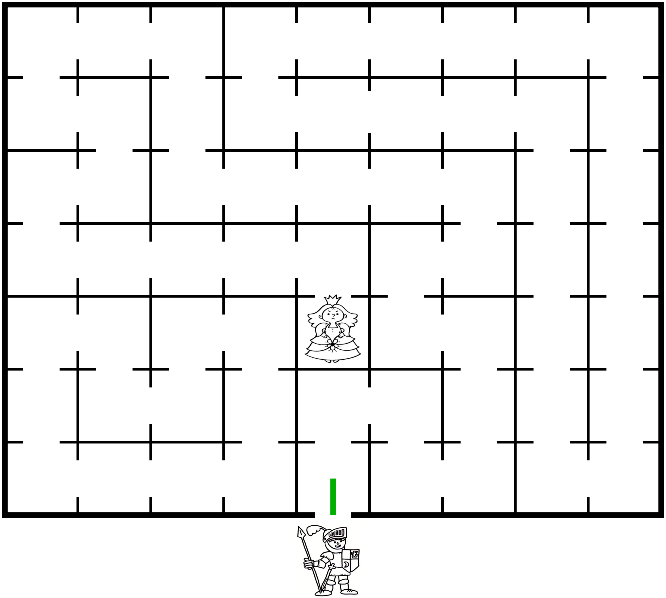
$\Rightarrow$ precludes "coming back" like here:



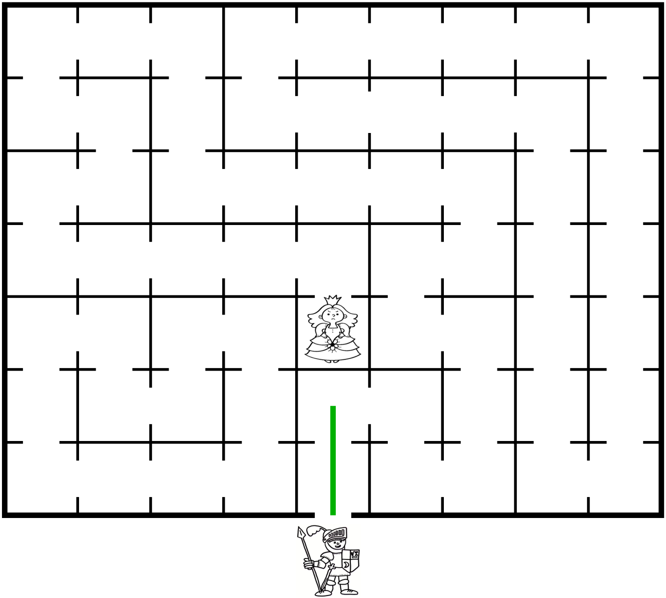
The castle where each room has at most two doors



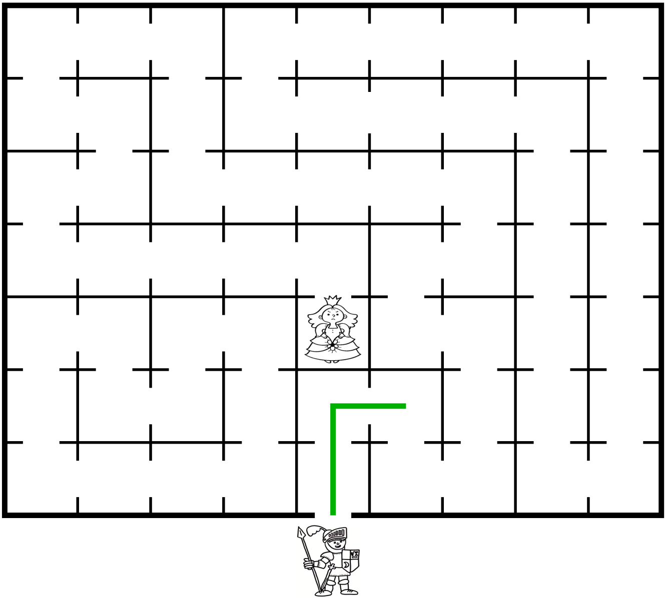
The castle where each room has at most two doors



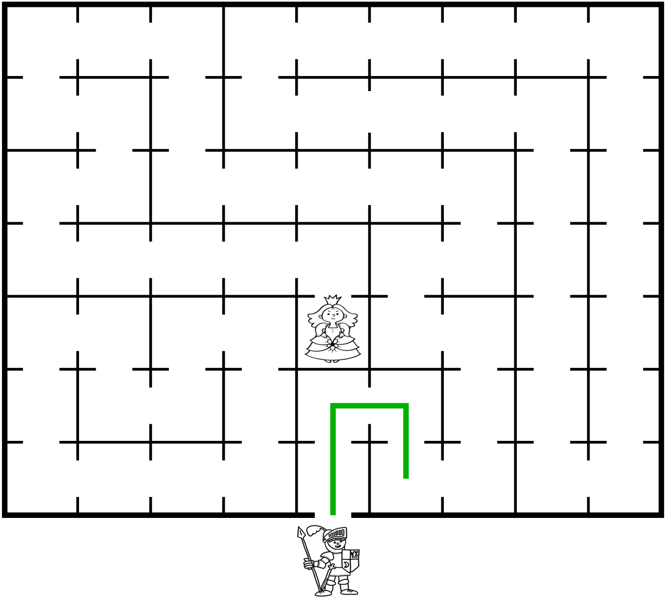
The castle where each room has at most two doors



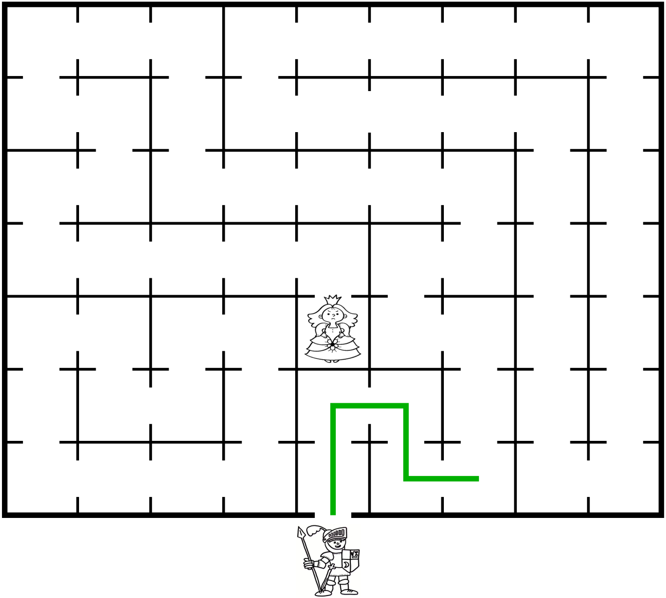
The castle where each room has at most two doors



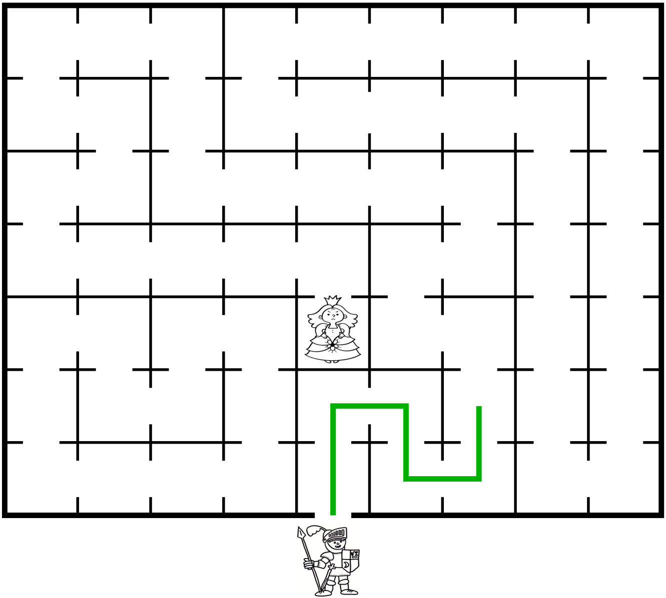
The castle where each room has at most two doors



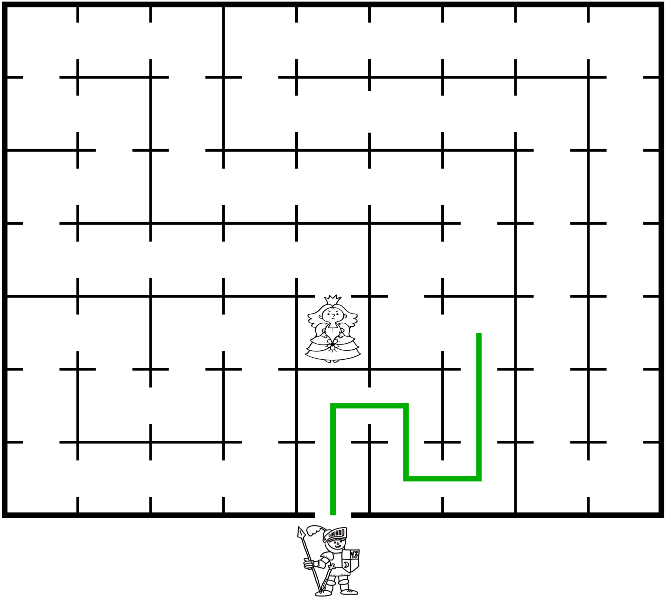
The castle where each room has at most two doors



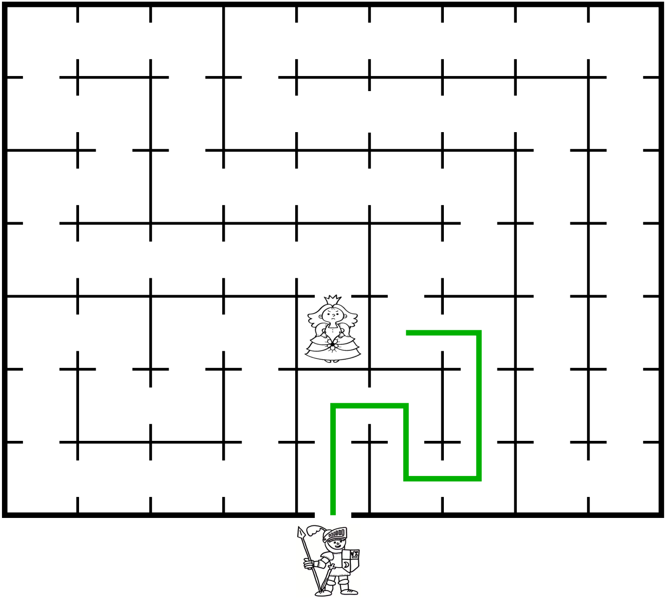
The castle where each room has at most two doors



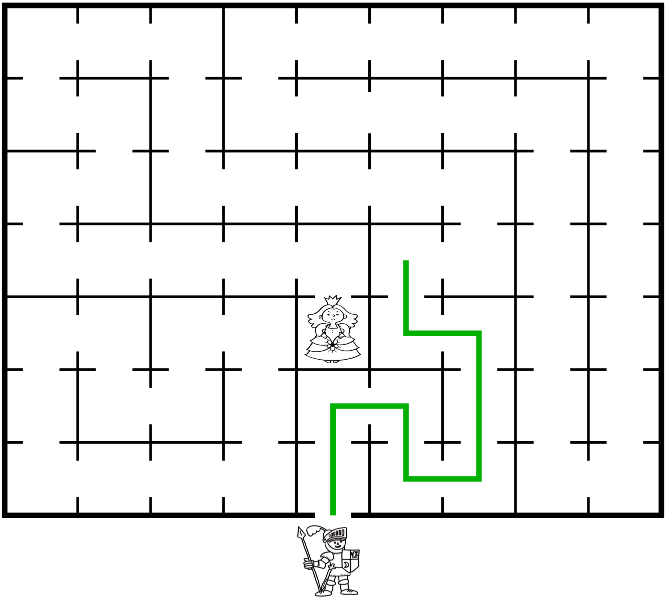
The castle where each room has at most two doors



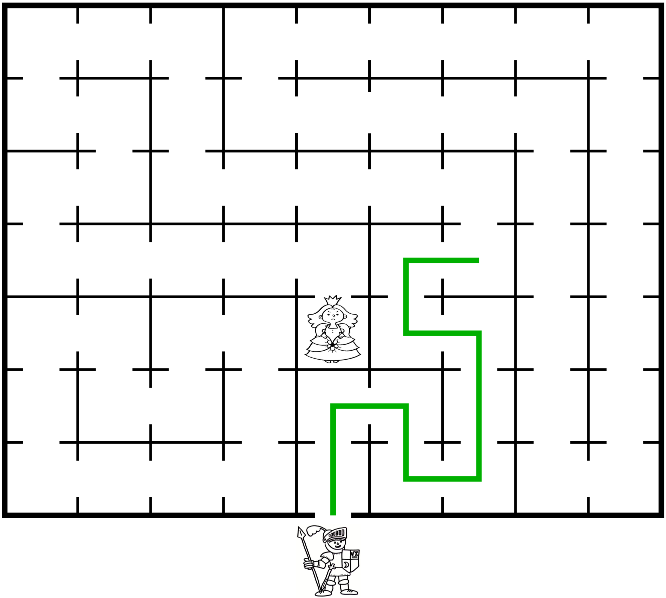
The castle where each room has at most two doors



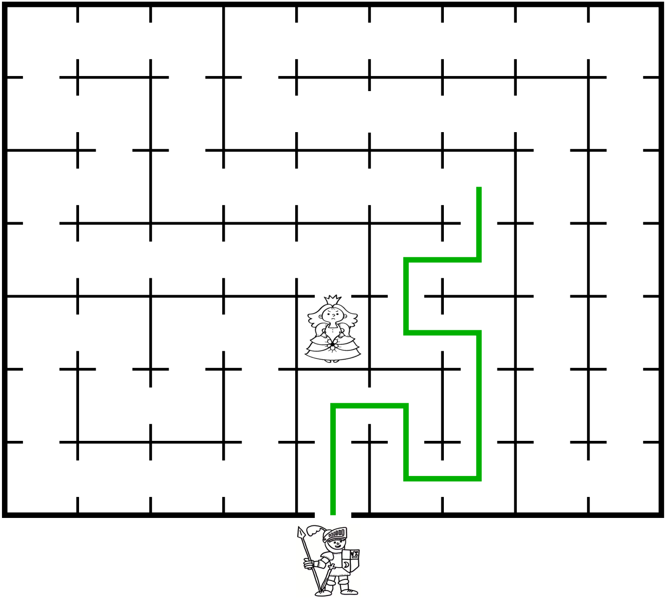
The castle where each room has at most two doors



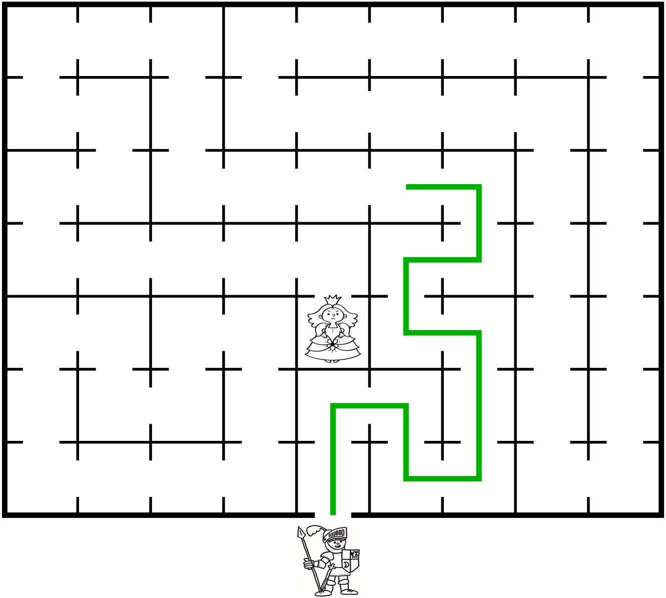
The castle where each room has at most two doors



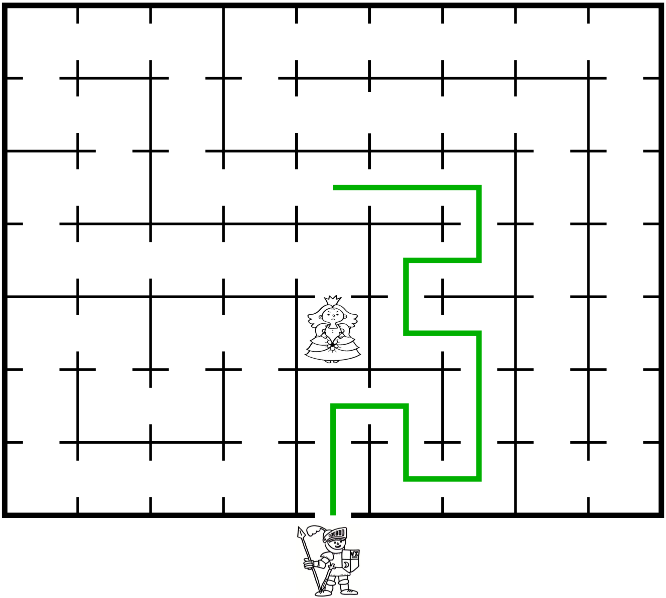
The castle where each room has at most two doors



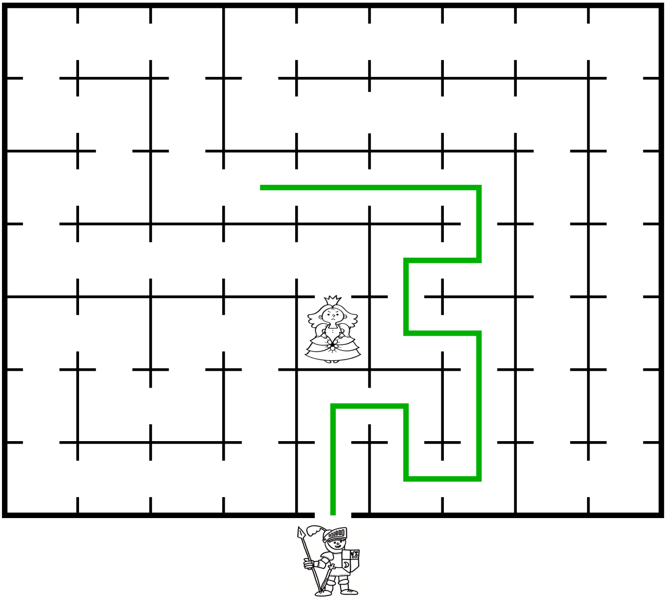
The castle where each room has at most two doors



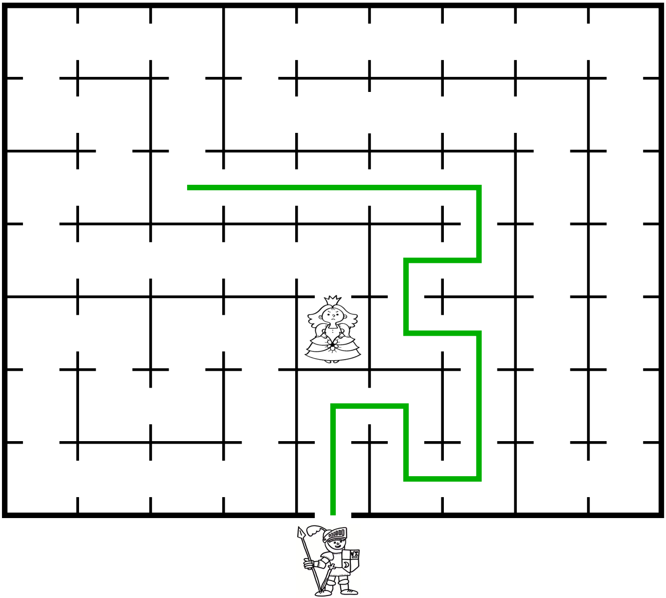
The castle where each room has at most two doors



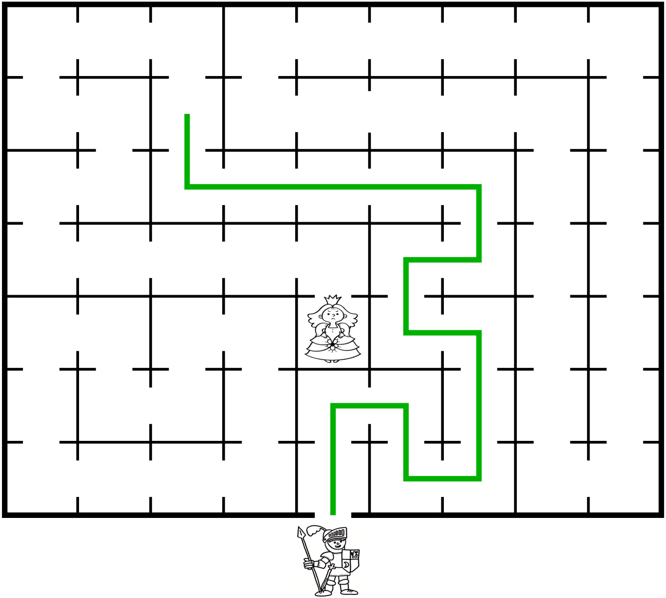
The castle where each room has at most two doors



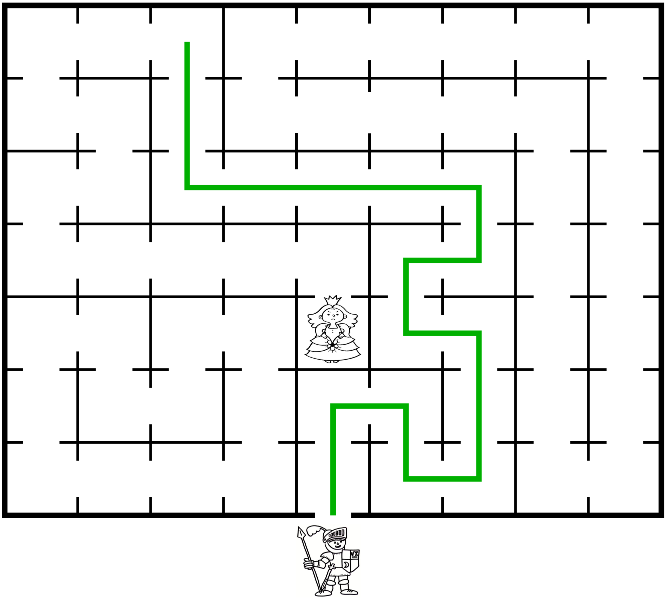
The castle where each room has at most two doors



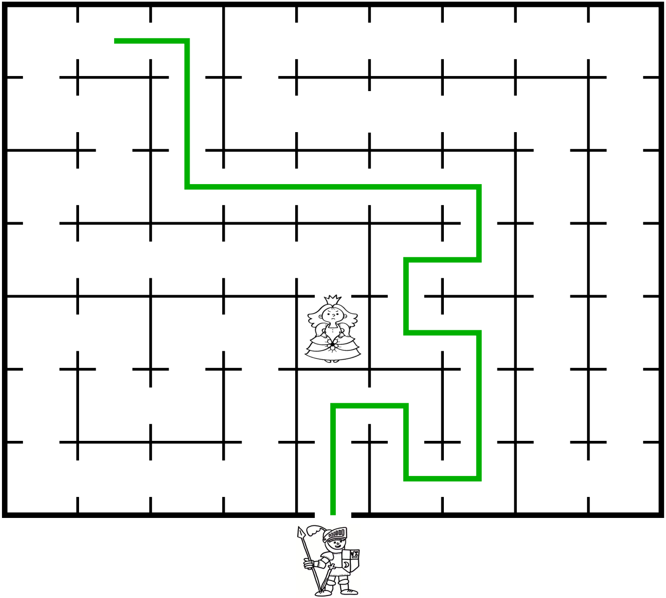
The castle where each room has at most two doors



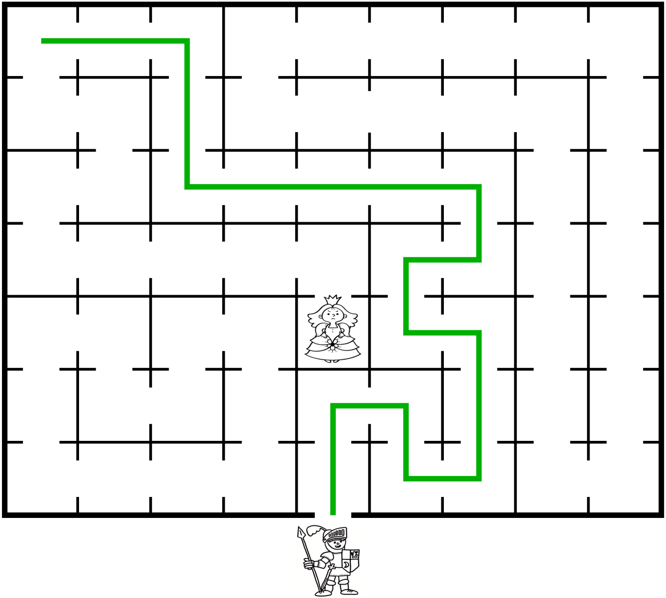
The castle where each room has at most two doors



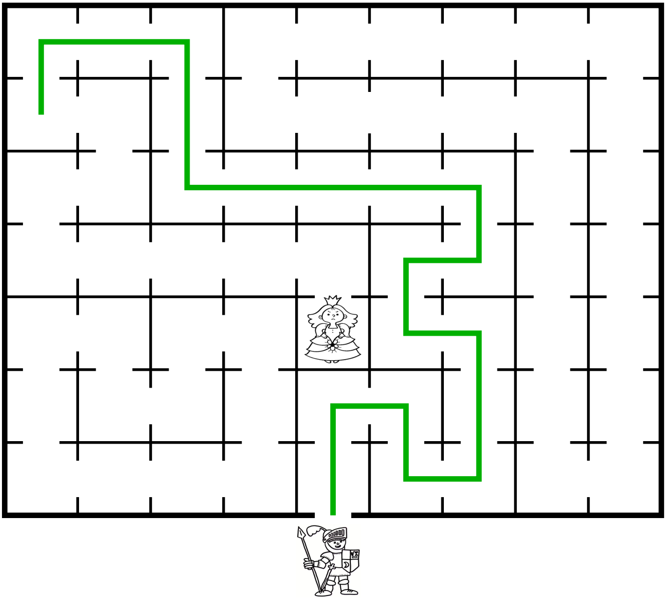
The castle where each room has at most two doors



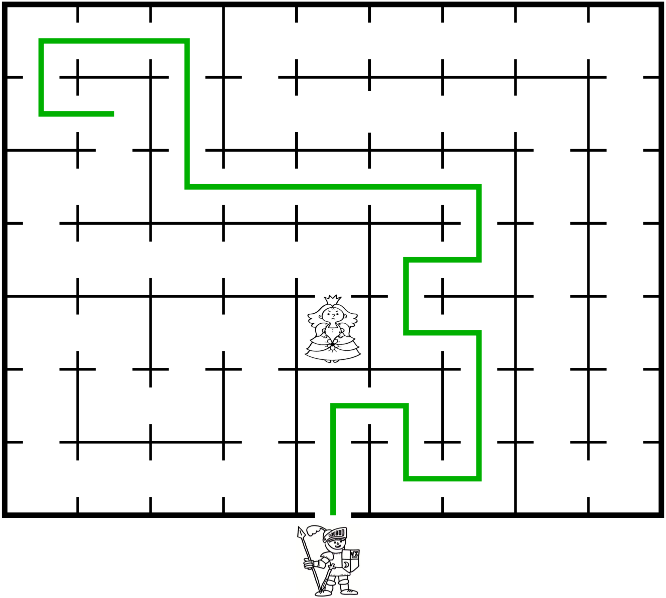
The castle where each room has at most two doors



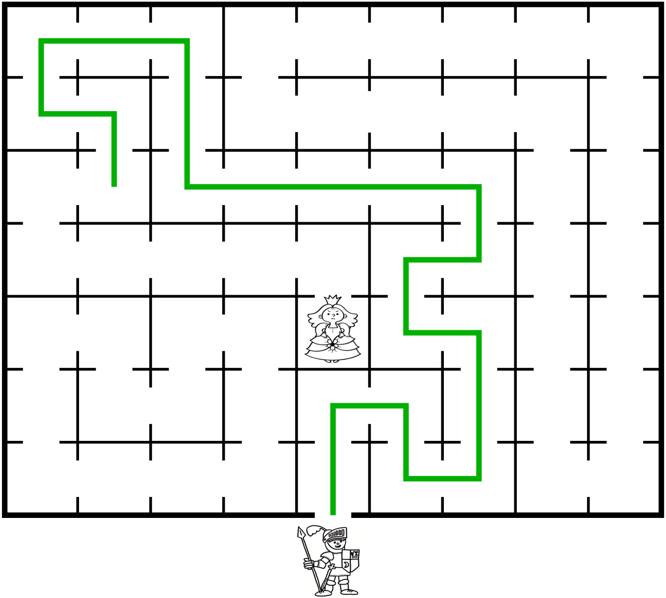
The castle where each room has at most two doors



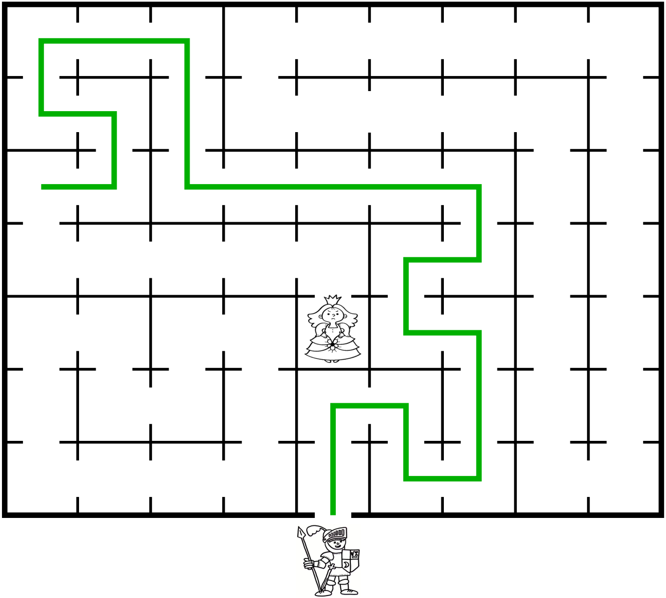
The castle where each room has at most two doors



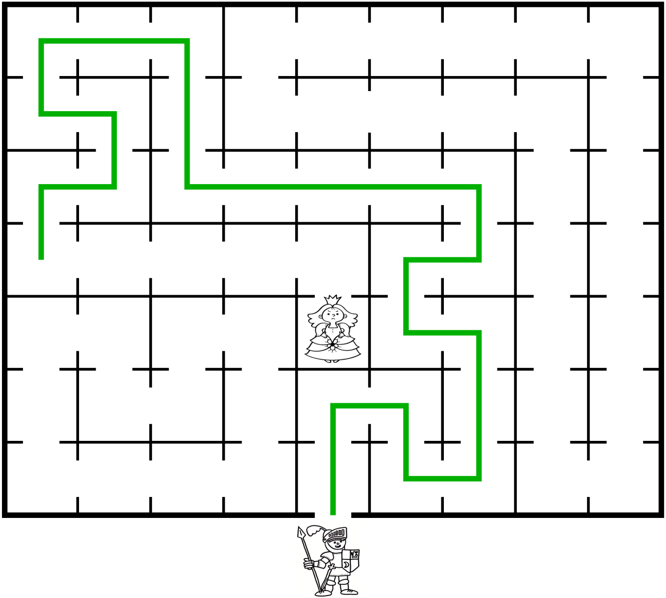
The castle where each room has at most two doors



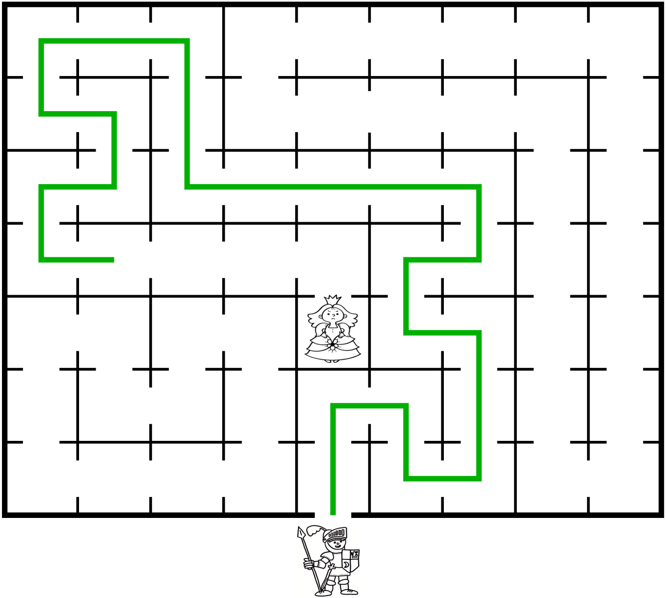
The castle where each room has at most two doors



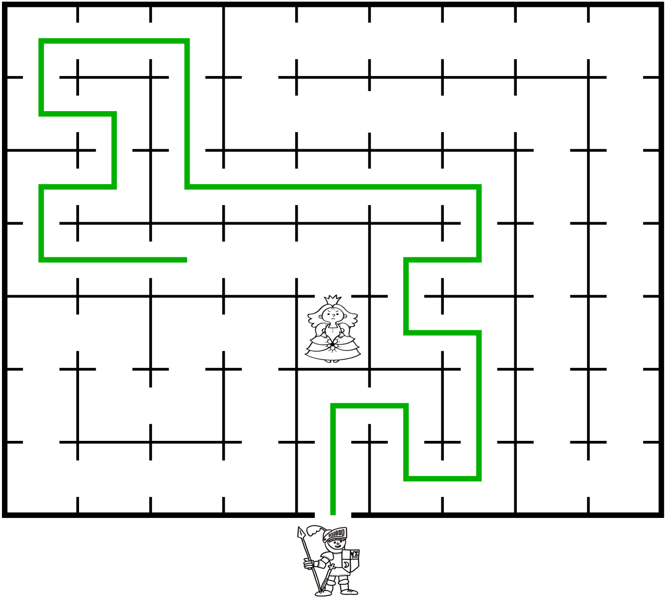
The castle where each room has at most two doors



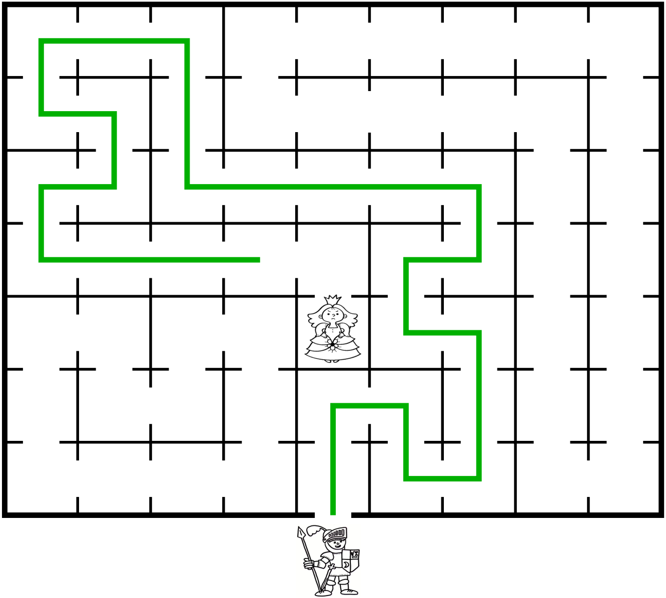
The castle where each room has at most two doors



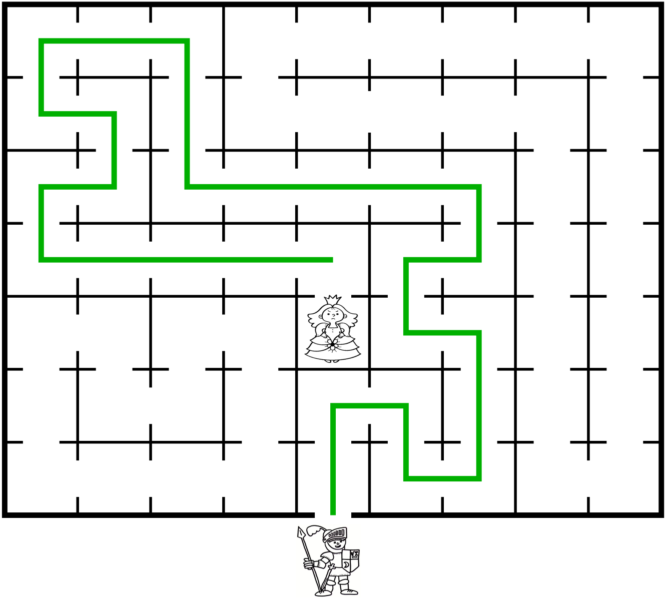
The castle where each room has at most two doors



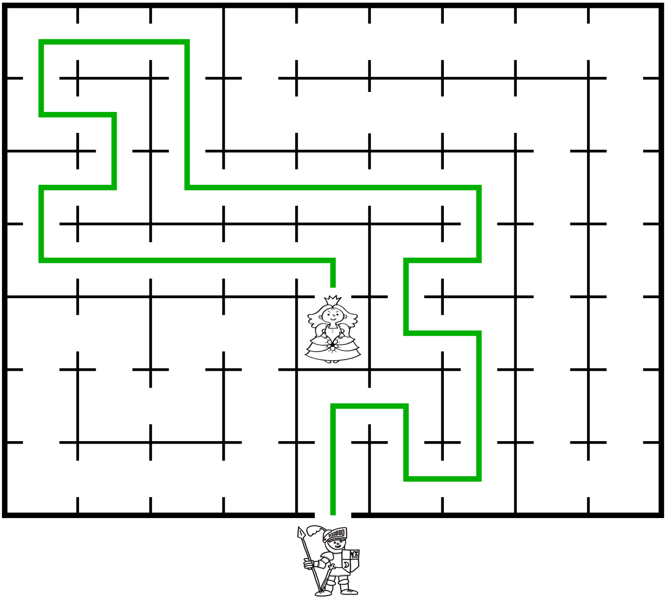
The castle where each room has at most two doors



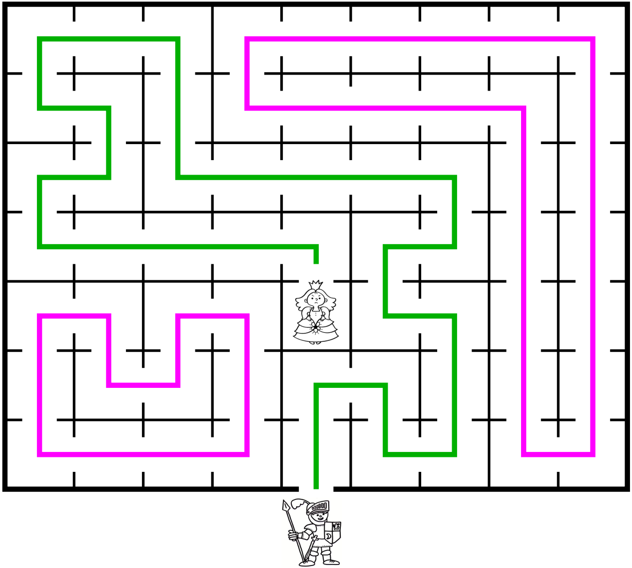
The castle where each room has at most two doors



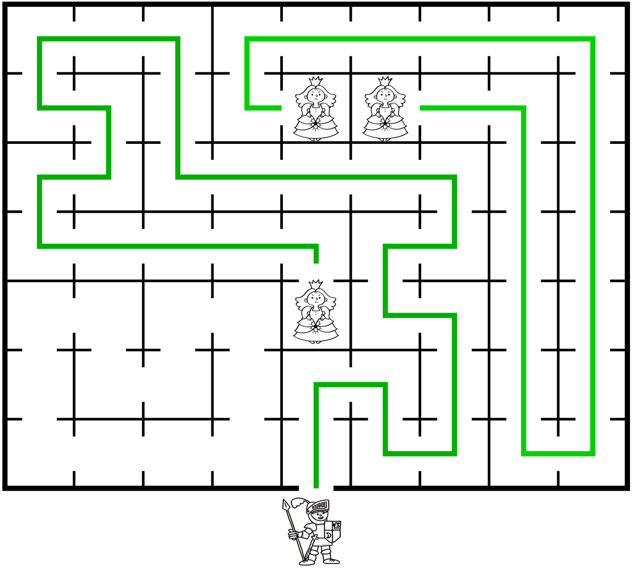
The castle where each room has at most two doors



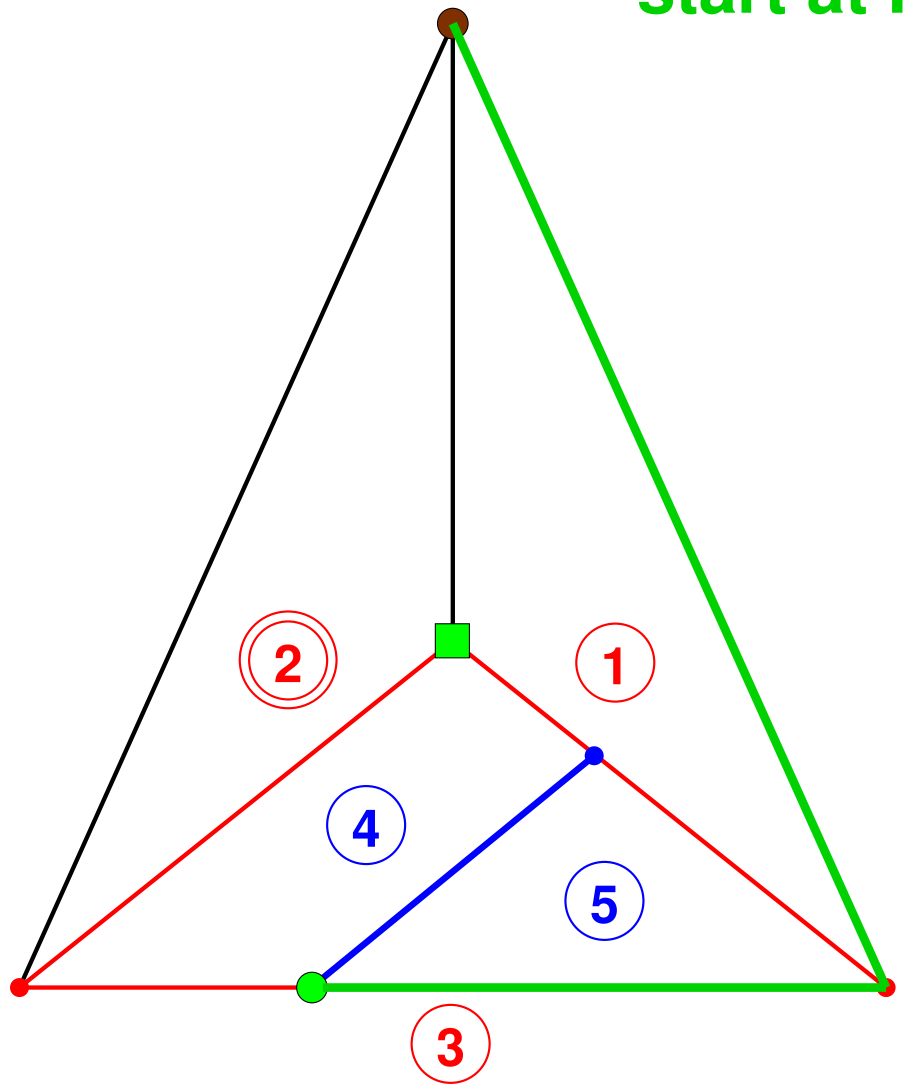
The castle where each room has at most two doors



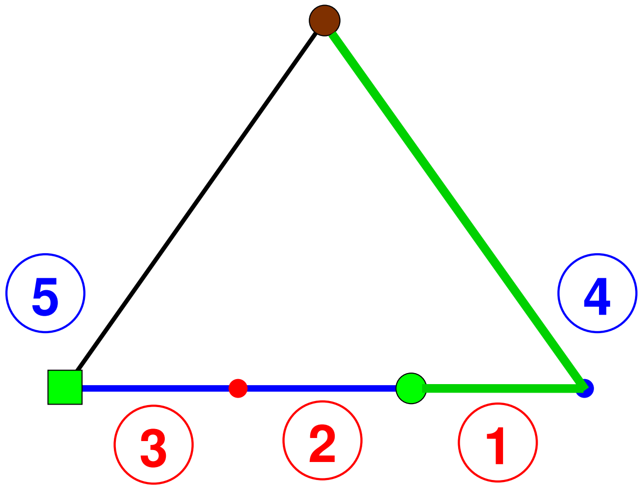
The castle where each room has at most two doors



The Lemke–Howson algorithm



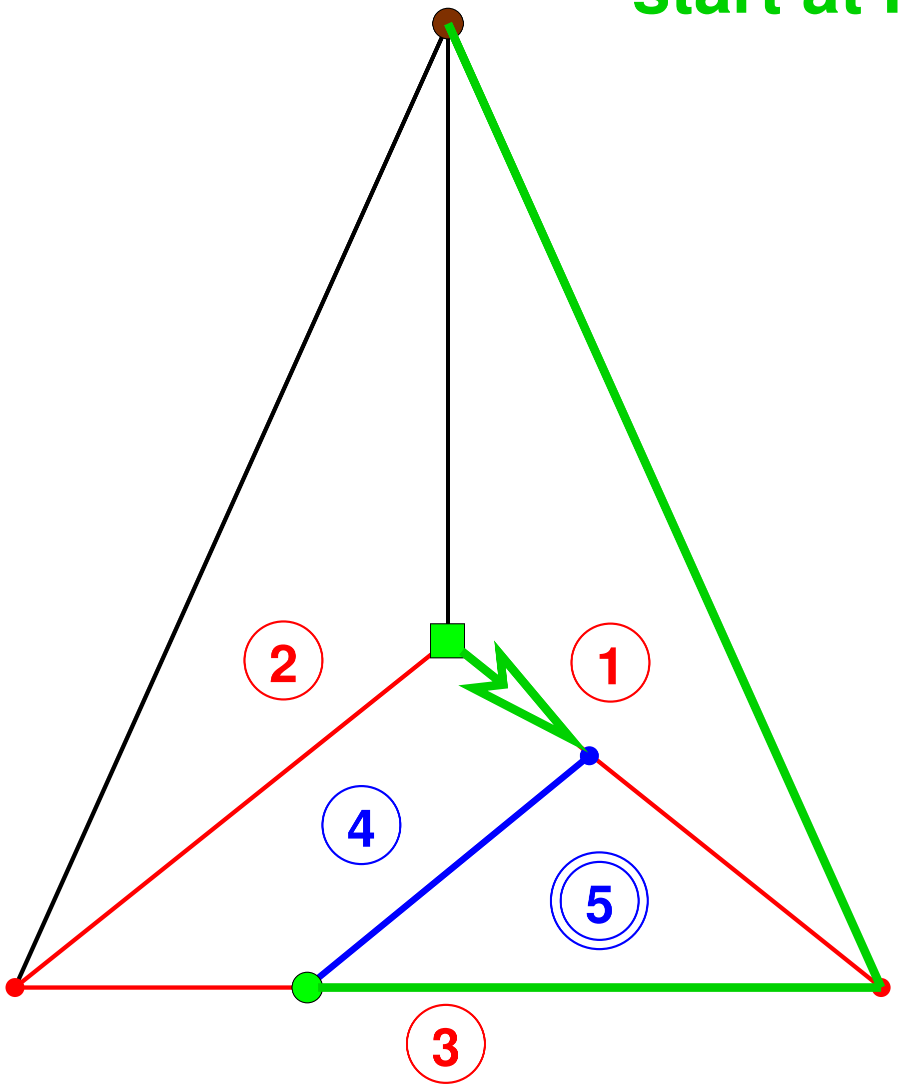
start at Nash equilibrium ■

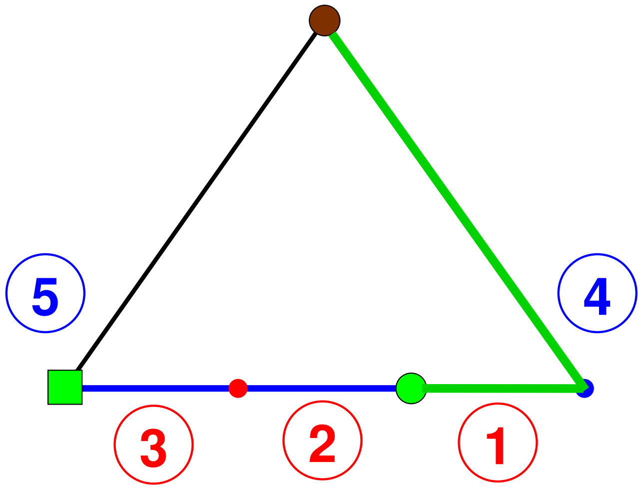


missing label 2

The Lemke–Howson algorithm

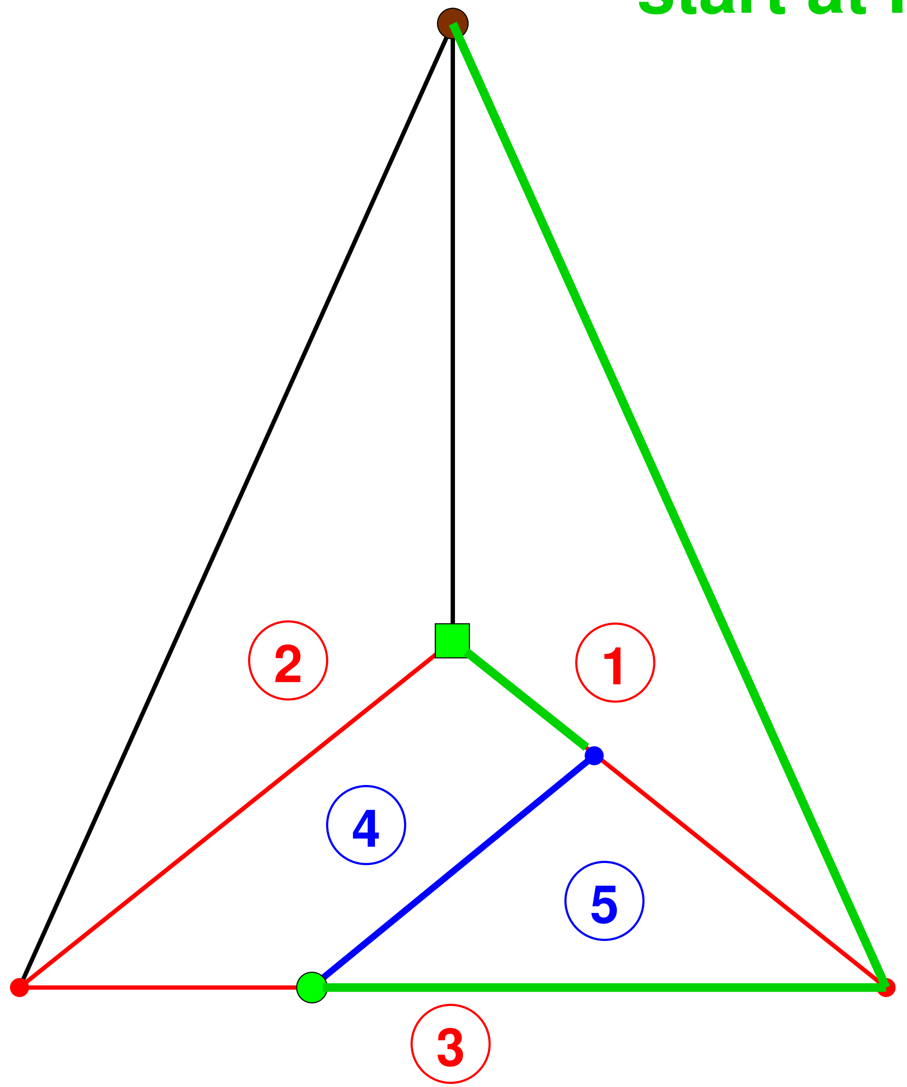
start at Nash equilibrium ■



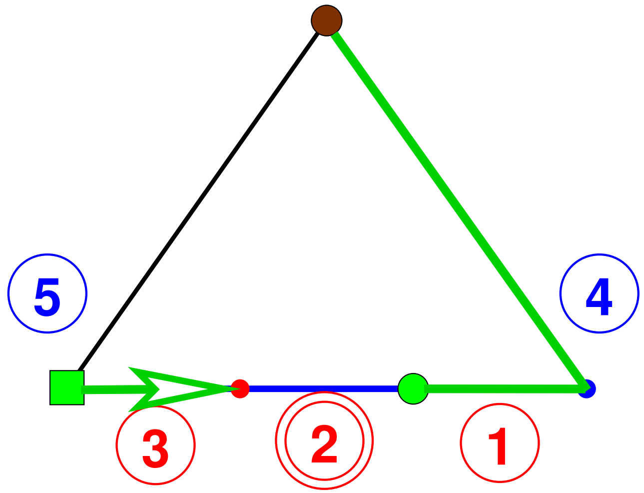


missing label 2

The Lemke–Howson algorithm



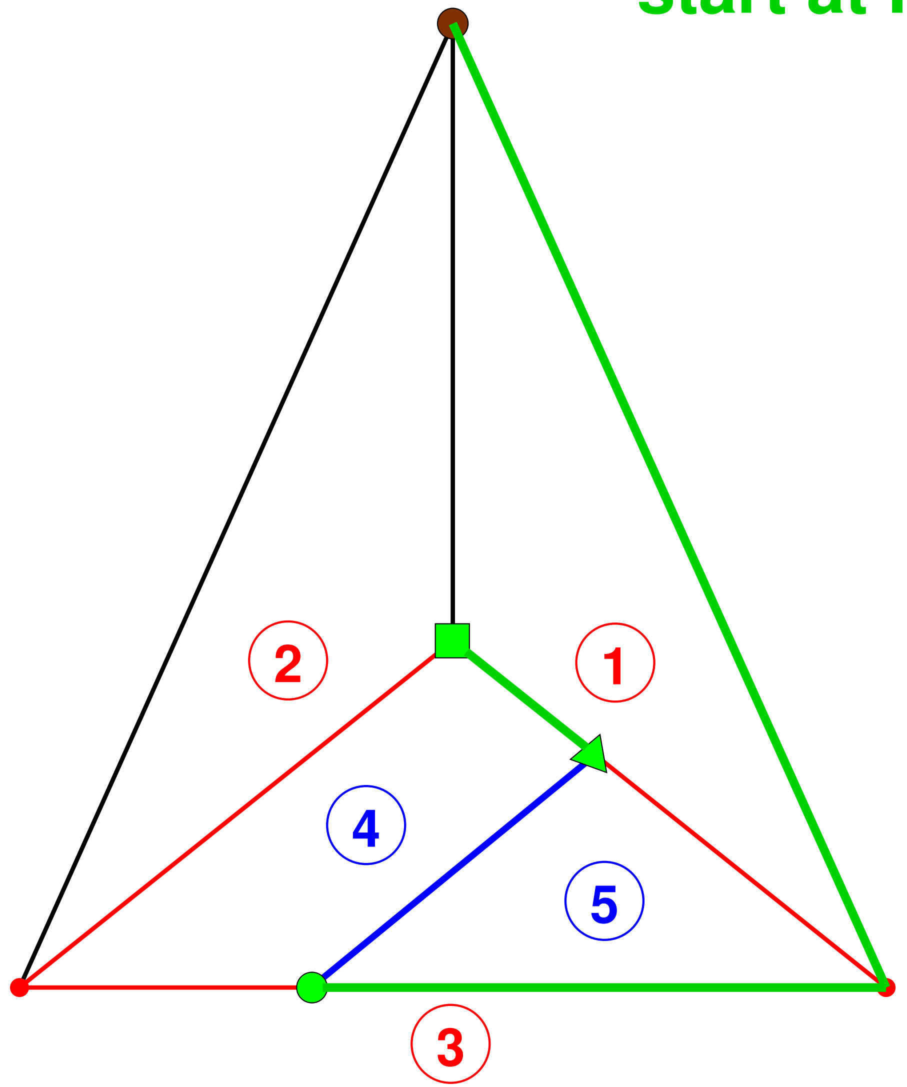
start at Nash equilibrium ■

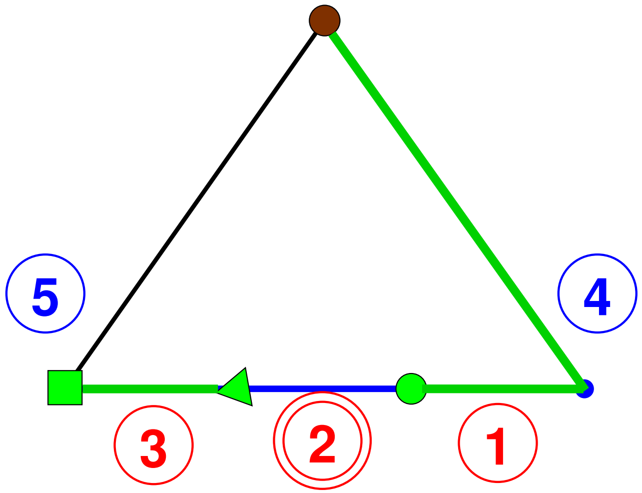


missing label 2

Odd number of Nash equilibria!

start at Nash equilibrium ■





found label 2

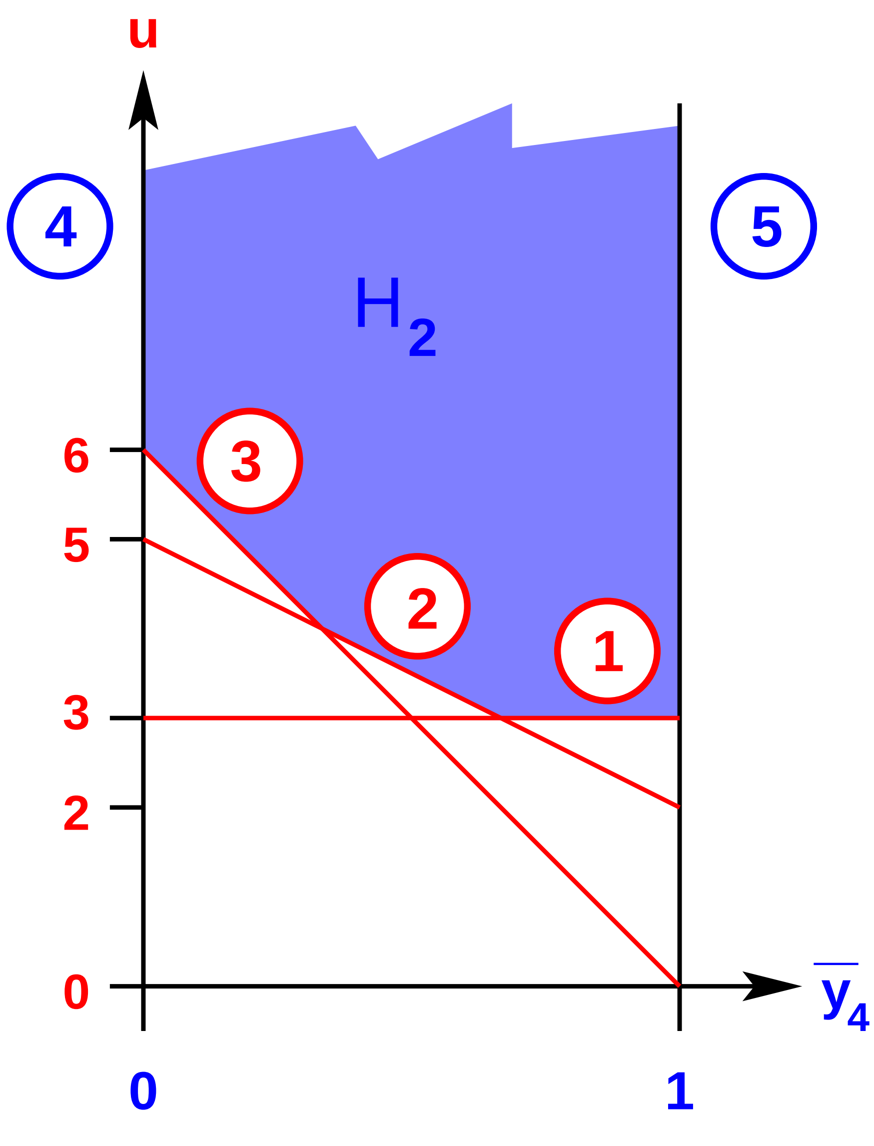
Best response polyhedron H_2 for player 2

	$\overline{y}_4$	$\overline{y}_5$
①	3	3
②	2	5
③	0	6

$= A$

$$H_2 = \{ (\overline{y}_4, \overline{y}_5, u) \mid$$

$$
\begin{array}{lrcl}
①: & 3\overline{y}_4 + 3\overline{y}_5 & \le & u \\
②: & 2\overline{y}_4 + 5\overline{y}_5 & \le & u \\
③: & 6\overline{y}_5 & \le & u \\
 & \overline{y}_4 + \overline{y}_5 & = & 1 \\
④: & \overline{y}_4 & \ge & 0 \\
⑤: & \overline{y}_5 & \ge & 0 \; \}
\end{array}
$$



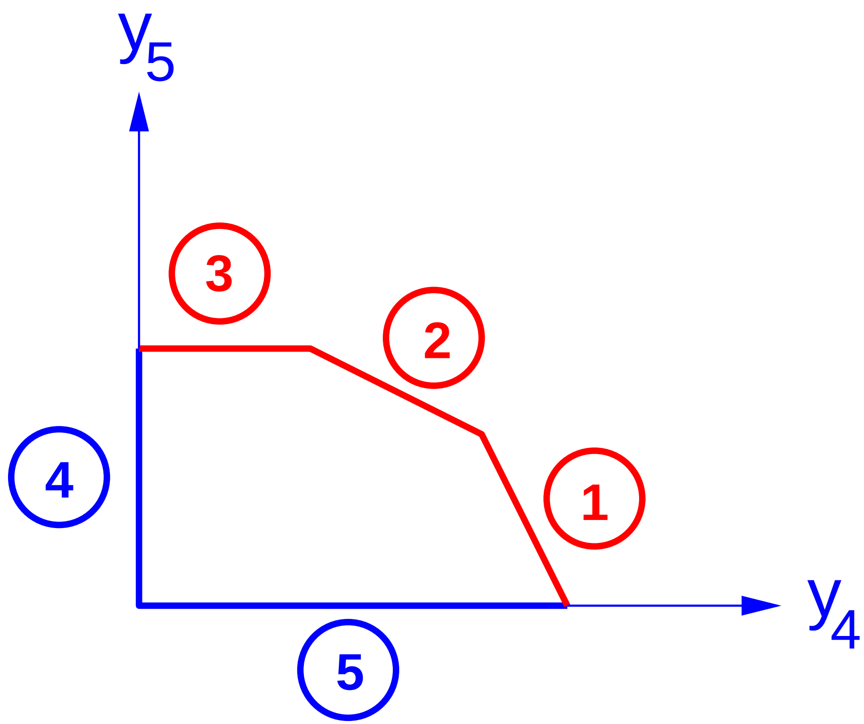
Best response polytope Q for player 2

$$\begin{array}{c|cc|} & y_4 & y_5 \\ \hline ① & 3 & 3 \\ ② & 2 & 5 \\ ③ & 0 & 6 \\ \hline \end{array} = A$$

$$Q = \{\, y \mid Ay \le 1,\ \ y \ge 0 \,\}$$

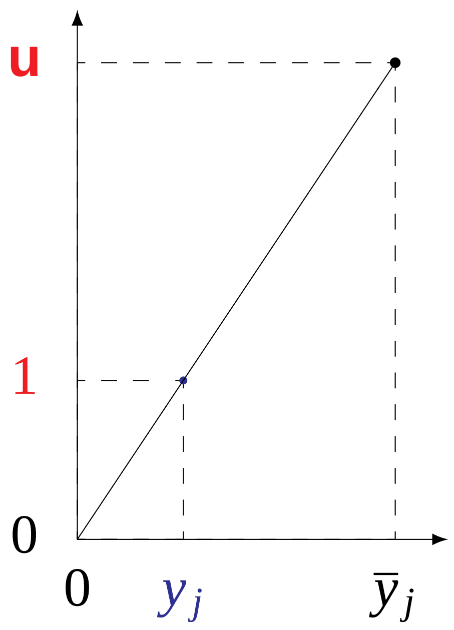
$$Q = \{ \ (y_4, y_5) \ |$$

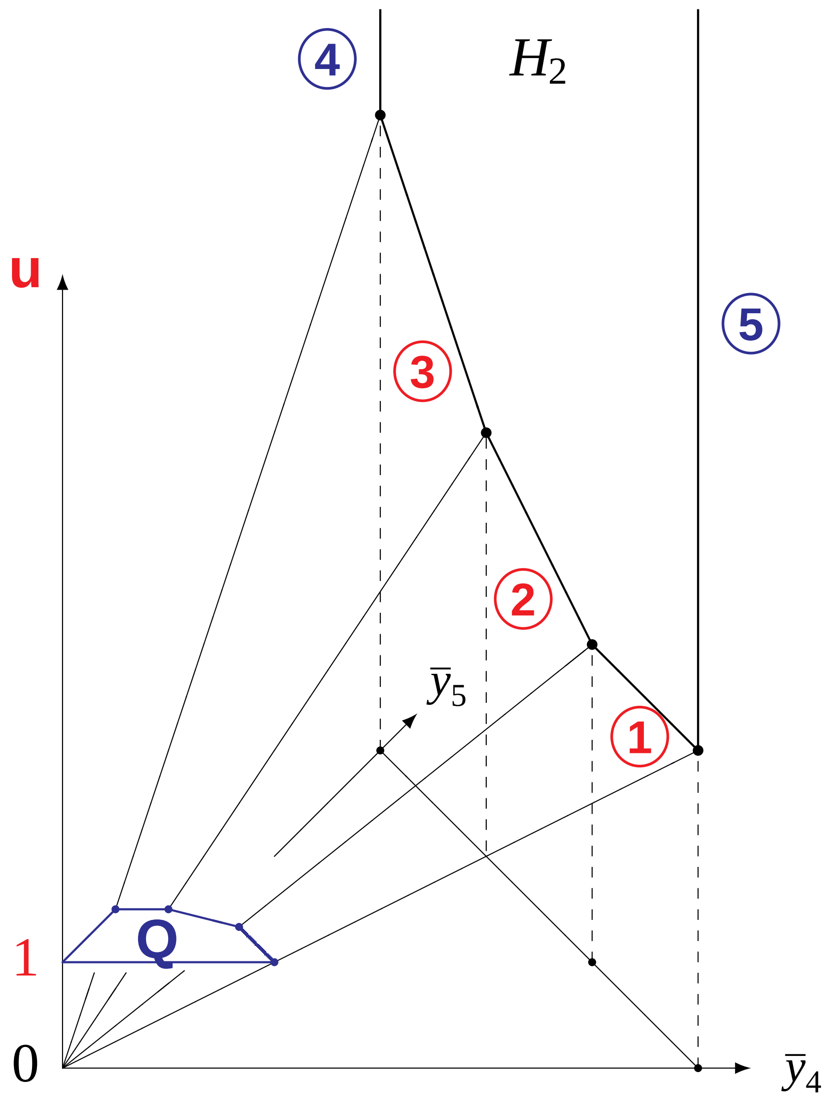
$$\begin{aligned}
①: &\quad 3y_4 + 3y_5 \le 1 \\
②: &\quad 2y_4 + 5y_5 \le 1 \\
③: &\quad \phantom{2y_4 + {}} 6y_5 \le 1 \\
\\
④: &\quad y_4 \ge 0 \\
⑤: &\quad y_5 \ge 0 \ \}
\end{aligned}$$



Projective transformation

H_2, **Q** same face incidences





Best response polytope Q for player 2

	y_4	y_5
①	3	3
②	2	5
③	0	6

$= A$

$$Q = \{\, y \mid Ay \le 1,\ y \ge 0 \,\}$$

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

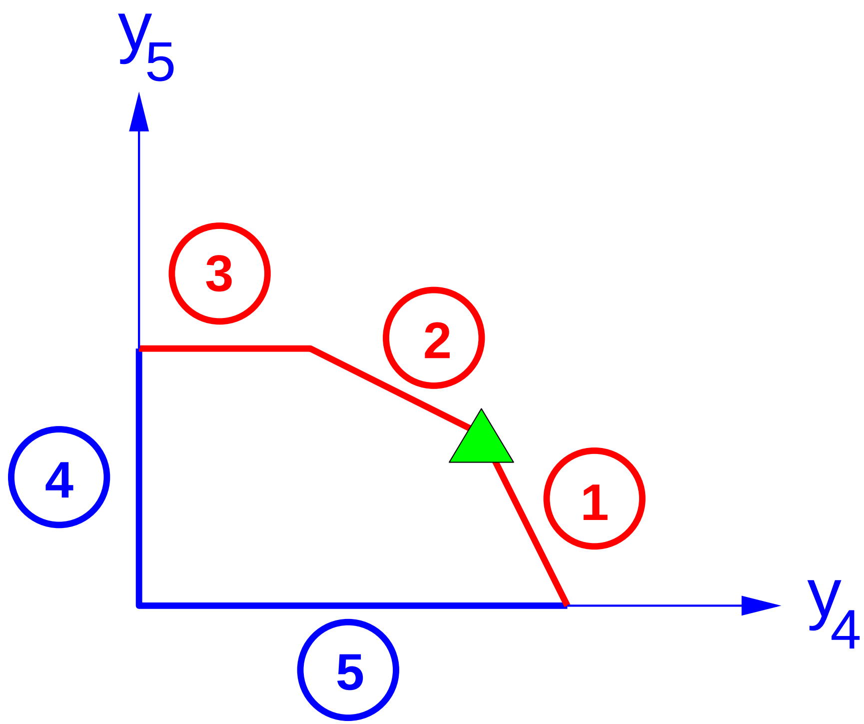
① : $3y_4 + 3y_5 \le 1$

② : $2y_4 + 5y_5 \le 1$

③ : $6y_5 \le 1$

④ : $y_4 \ge 0$

⑤ : $y_5 \ge 0\ \}$

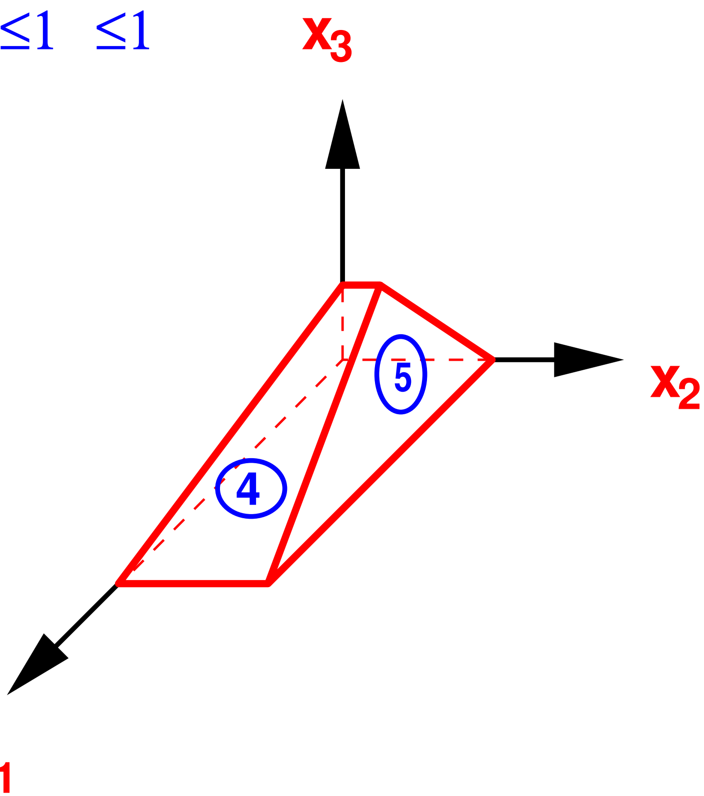


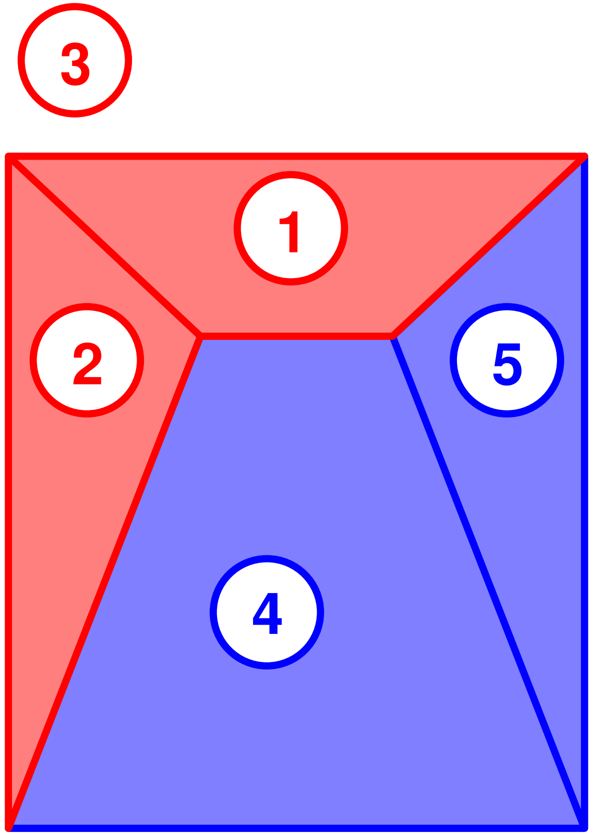
Best response polytope P for player 1

$$\begin{array}{c|cc|} x_1 & 1 & 0 \\ x_2 & 0 & 2 \\ x_3 & 4 & 3 \end{array} = B$$

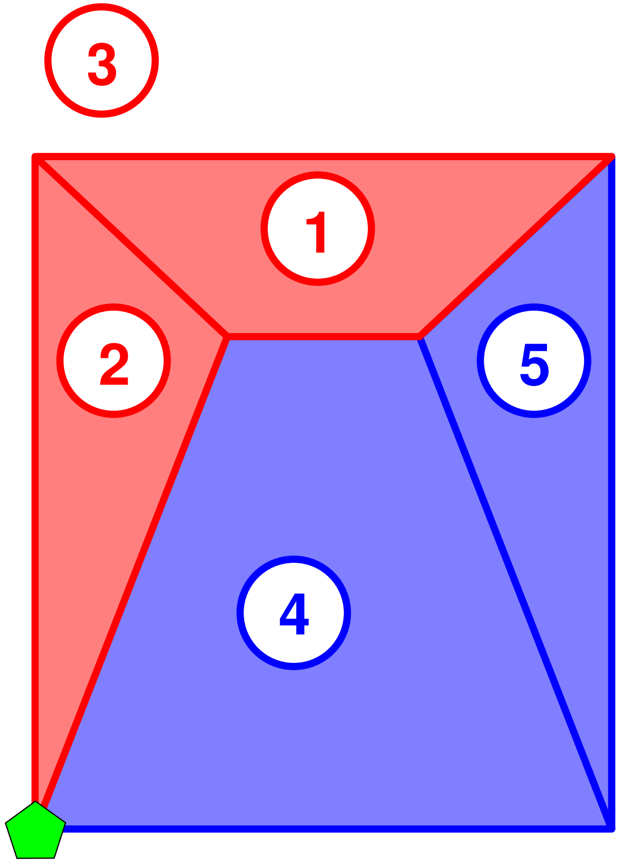
$\leq 1 \quad \leq 1$

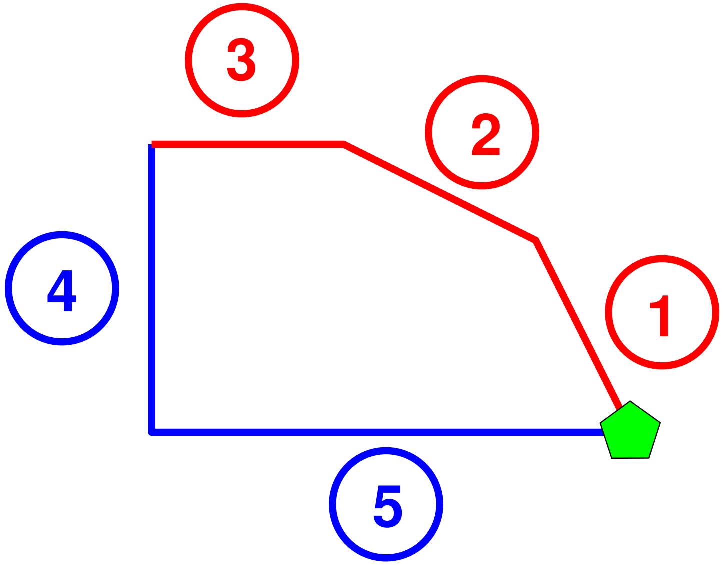
$$P = \{\, x \mid x \geq 0,\ x^{\top} B \leq 1 \,\}$$





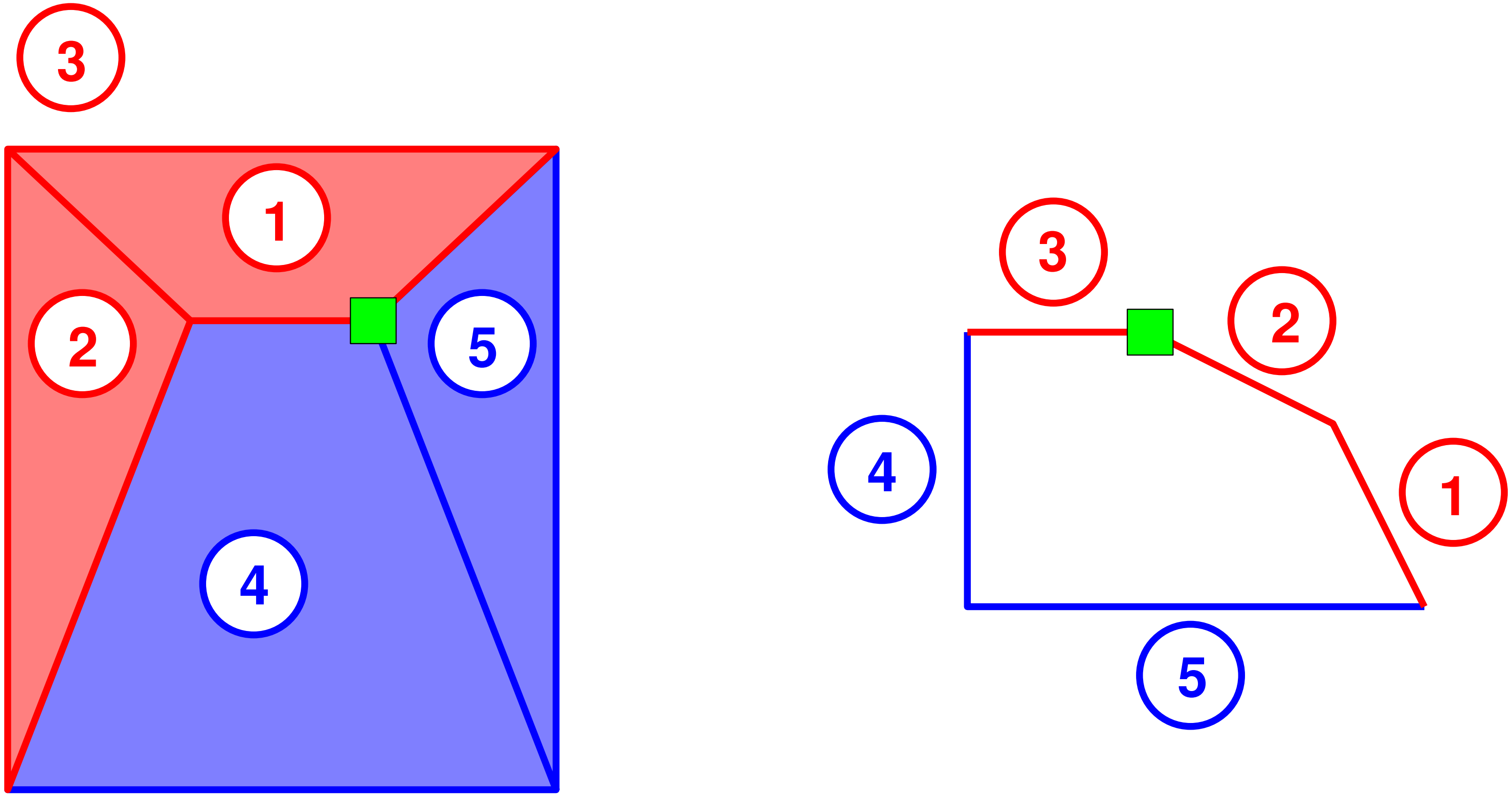
Equilibrium = completely labeled pair





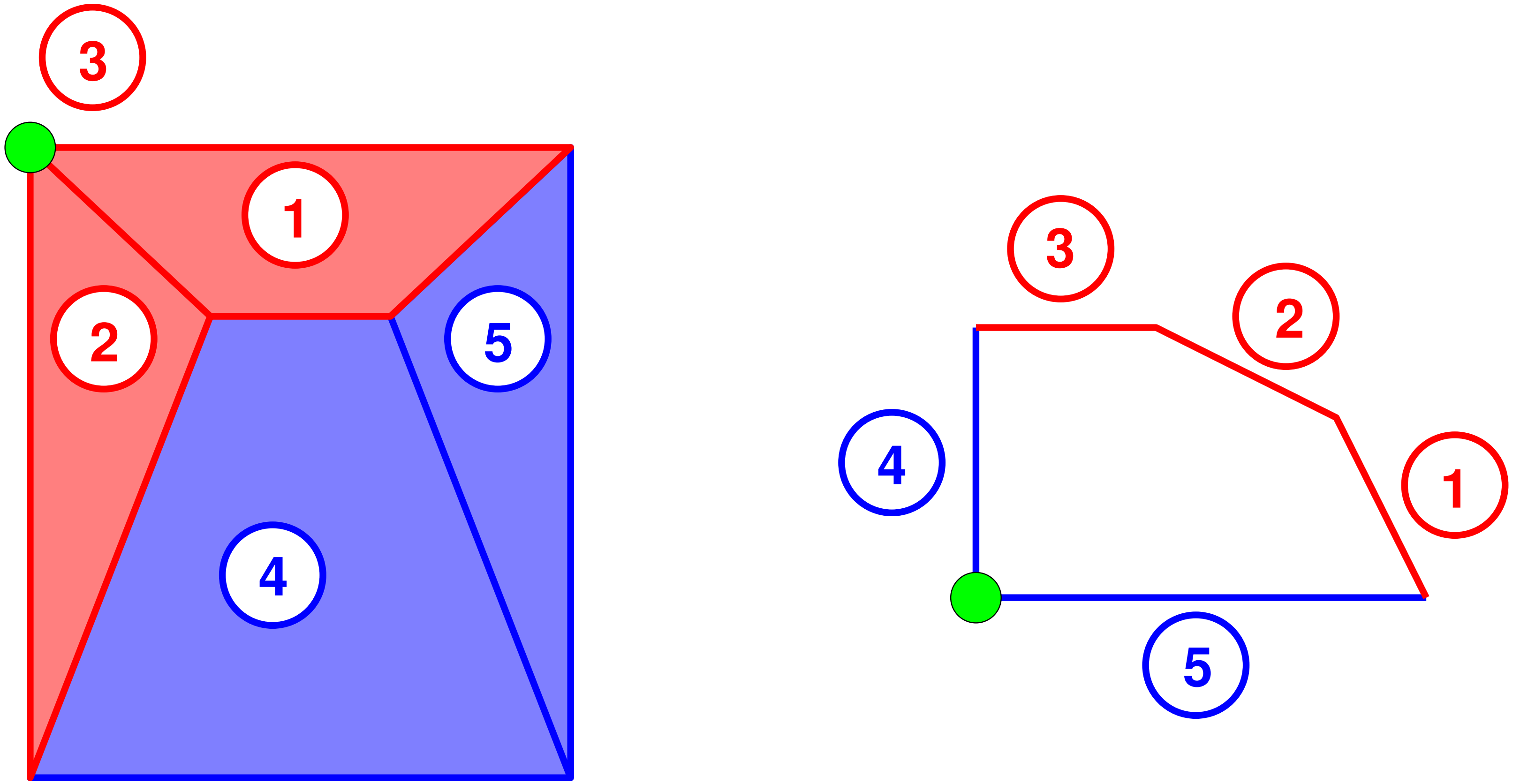
pure equilibrium

Equilibrium = completely labeled pair

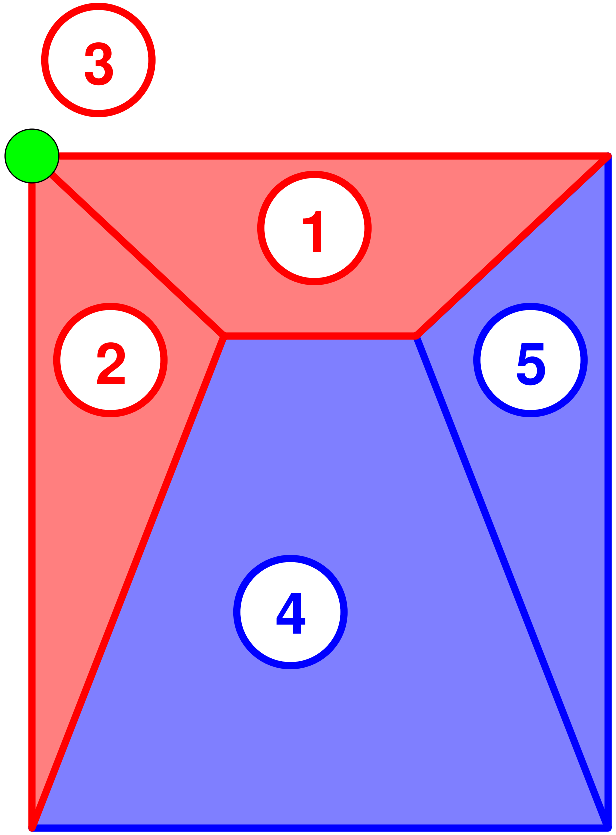


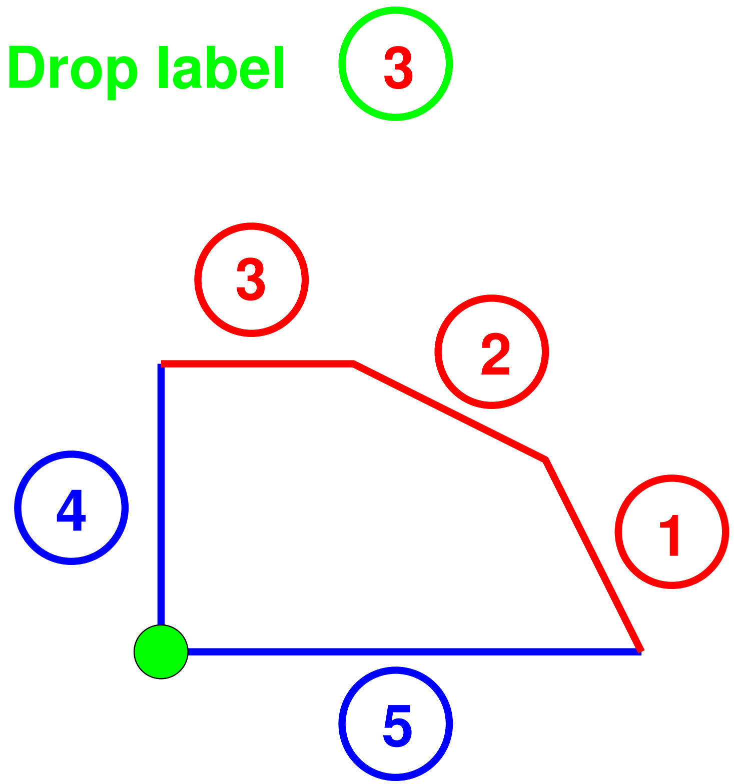
mixed equilibrium

The Lemke–Howson algorithm

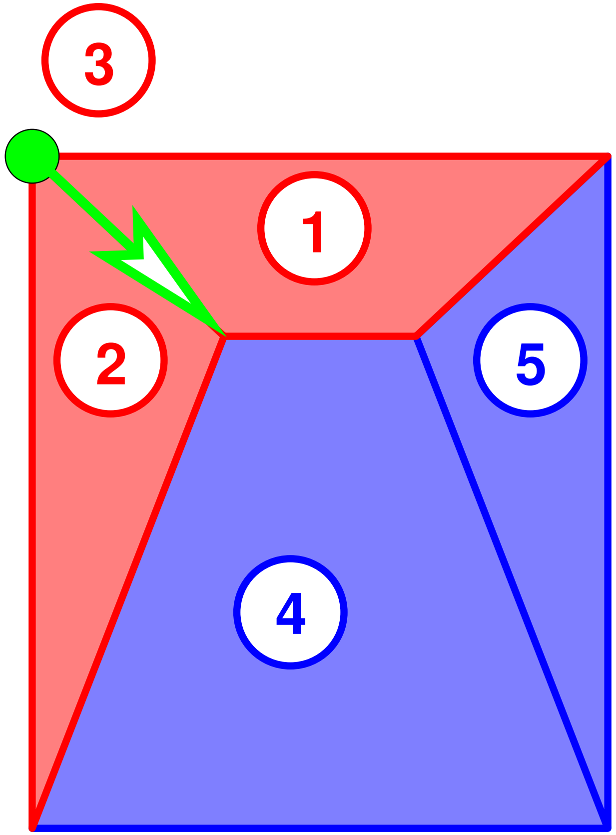


The Lemke–Howson algorithm

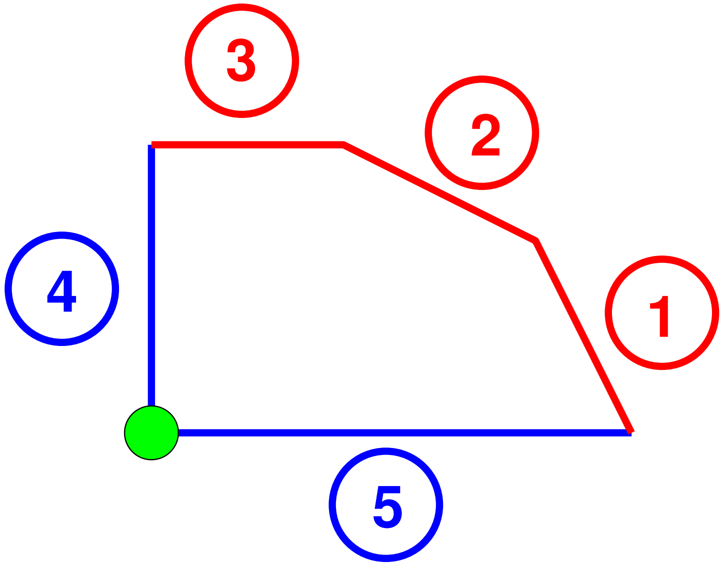




The Lemke–Howson algorithm

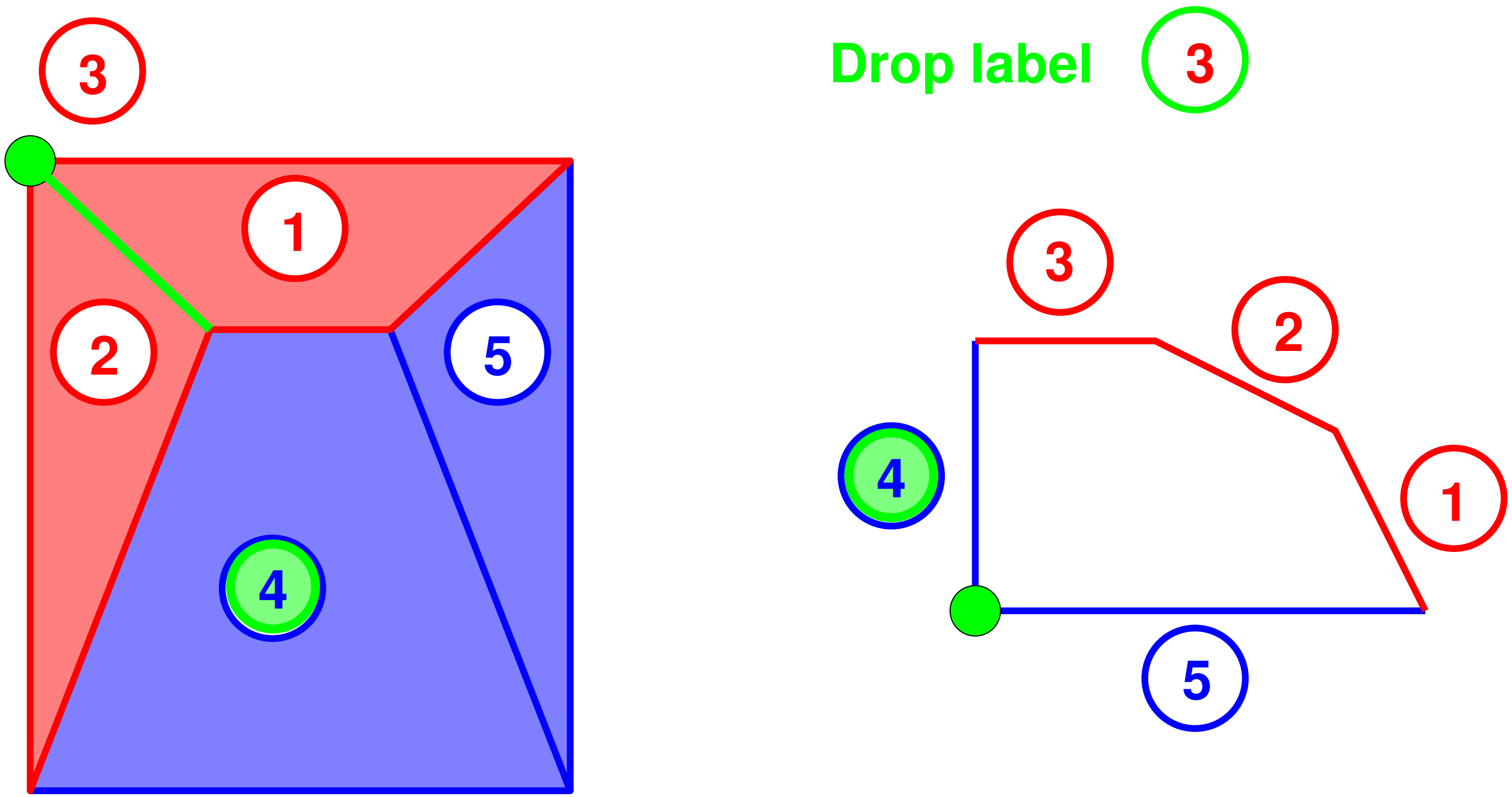


Drop label 3

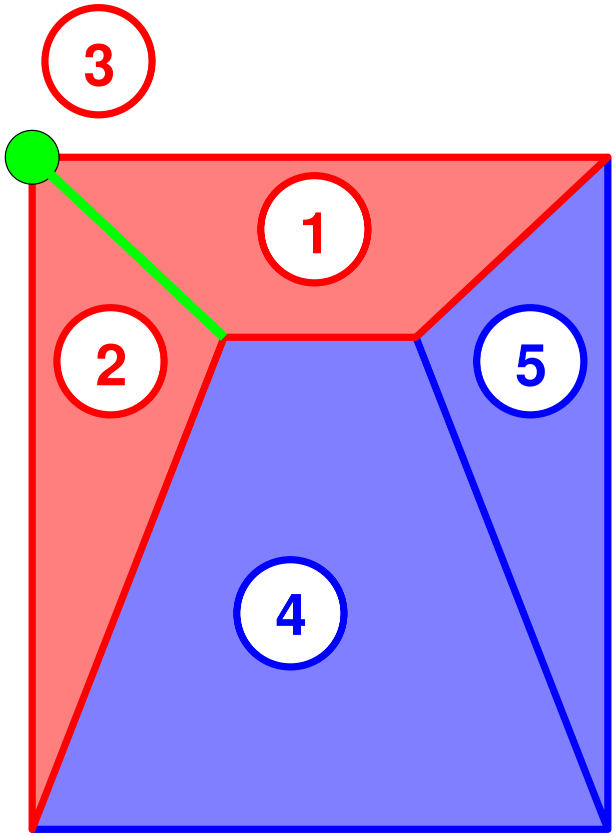


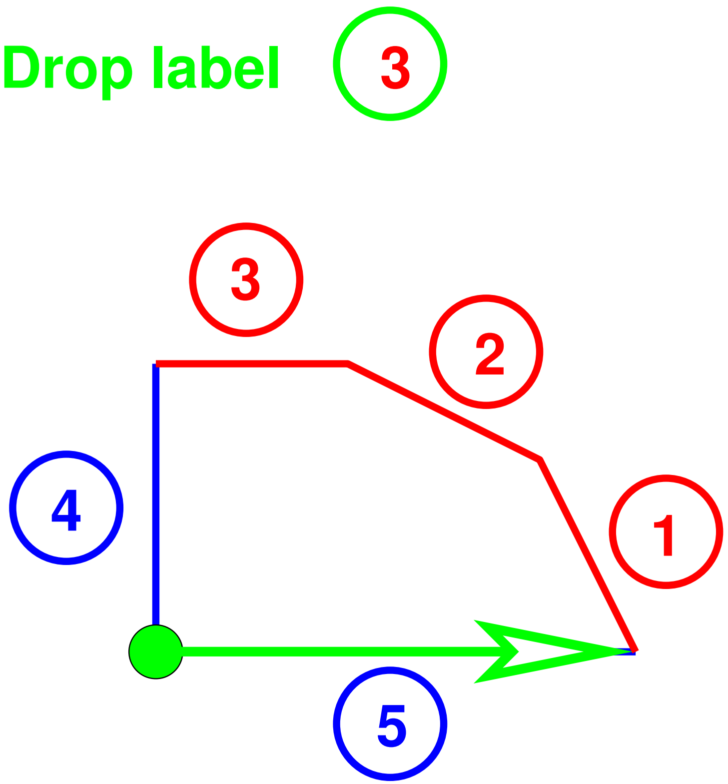
The Lemke–Howson algorithm

Drop label 3

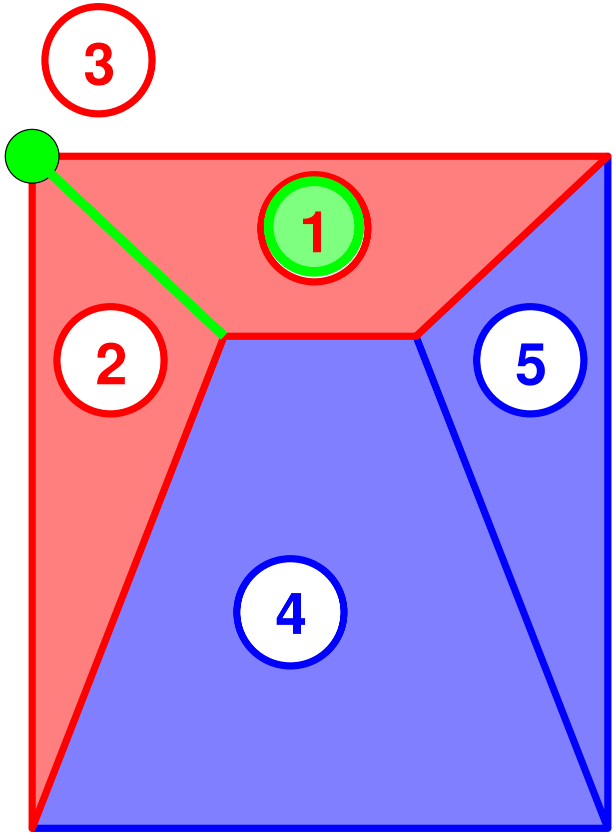


The Lemke–Howson algorithm

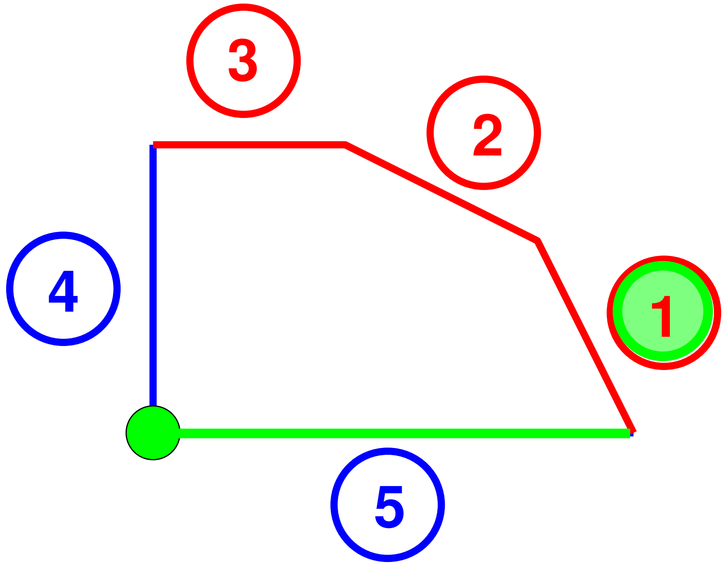




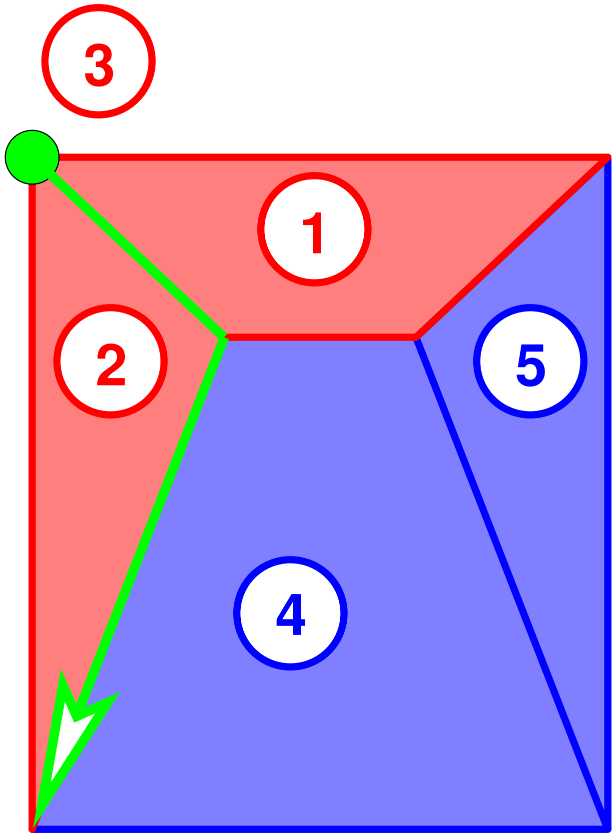
The Lemke–Howson algorithm



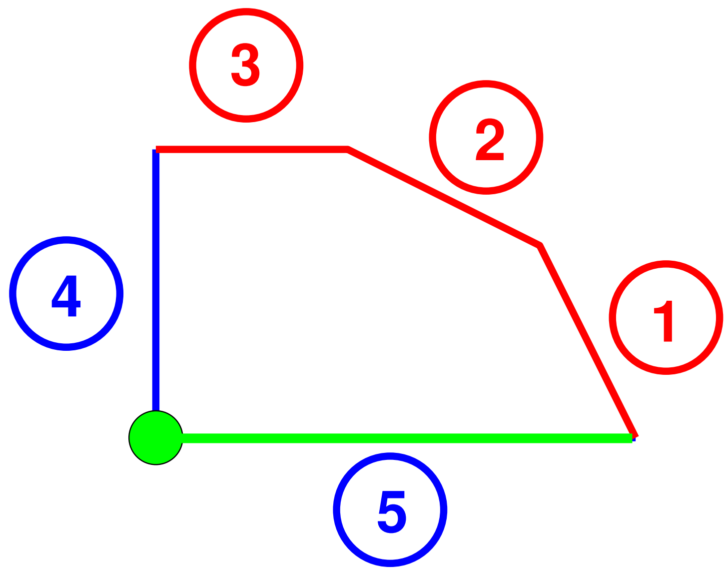
Drop label 3



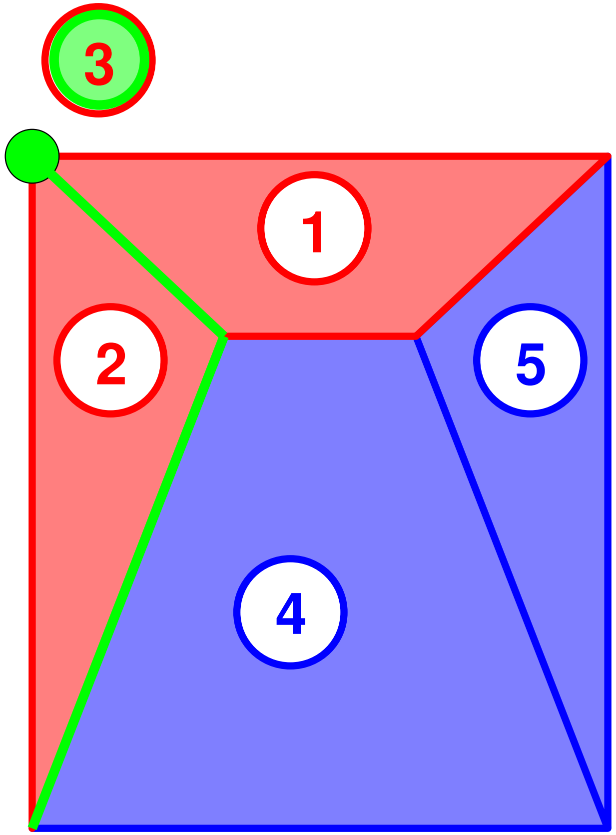
The Lemke–Howson algorithm



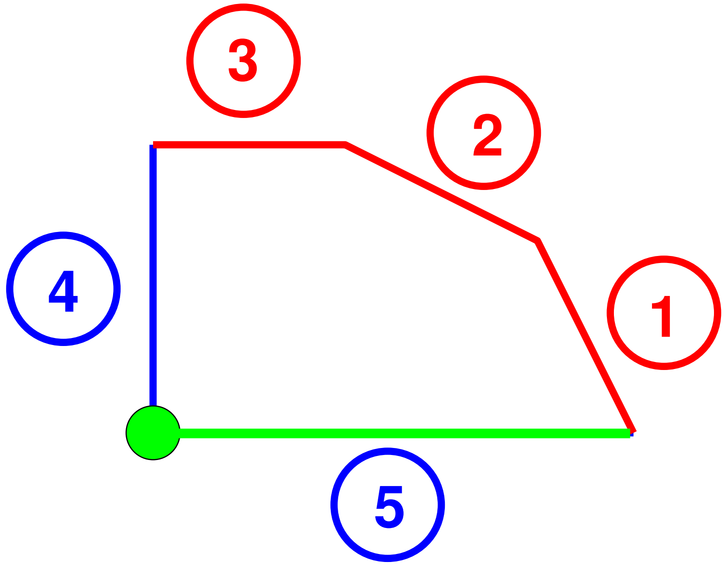
Drop label 3



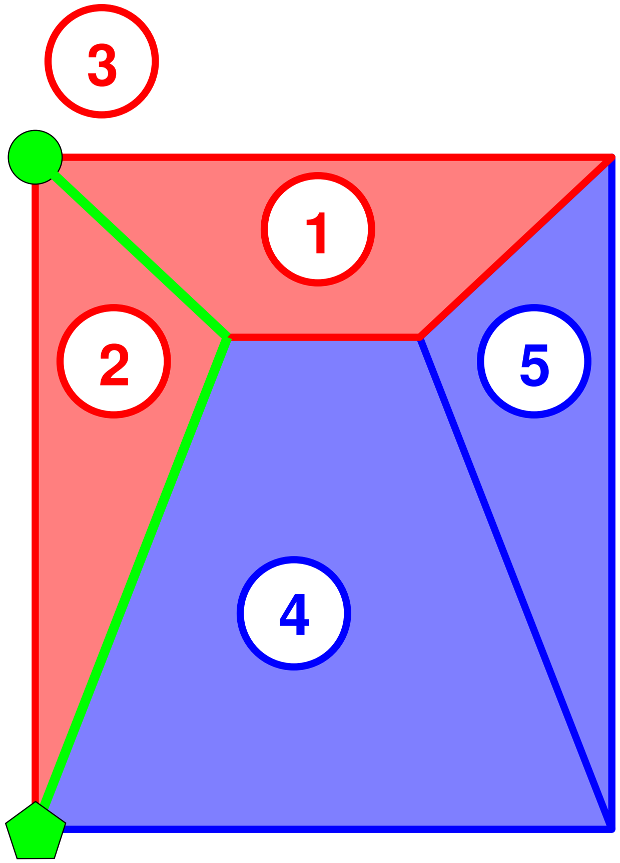
The Lemke–Howson algorithm



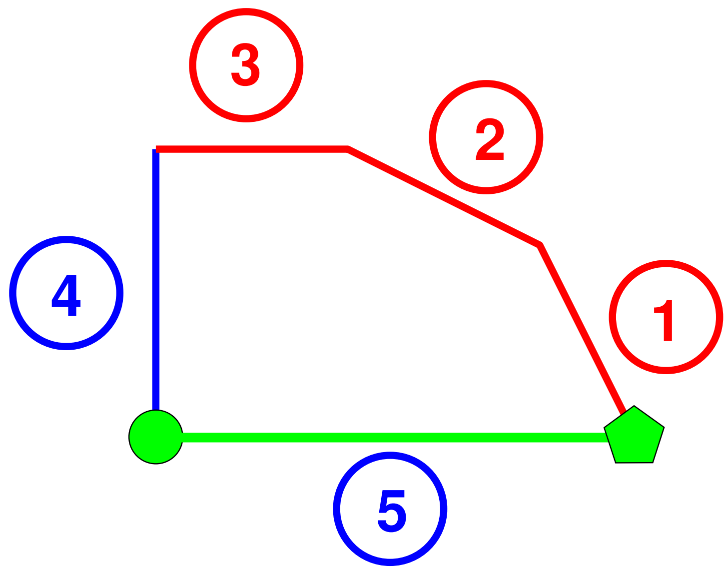
Drop label 3



The Lemke–Howson algorithm



Drop label 3



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

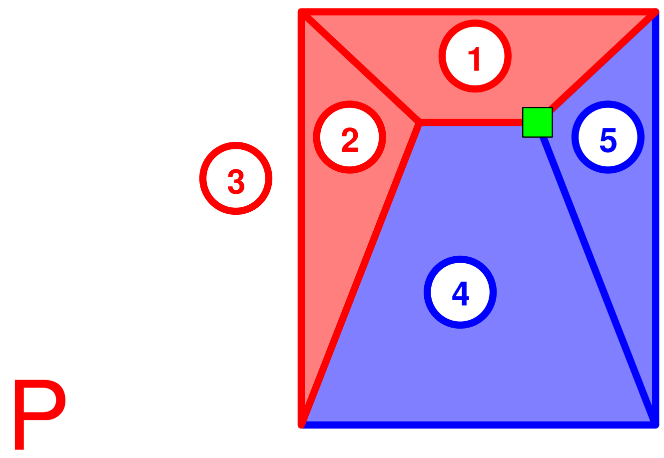
We will show this extending

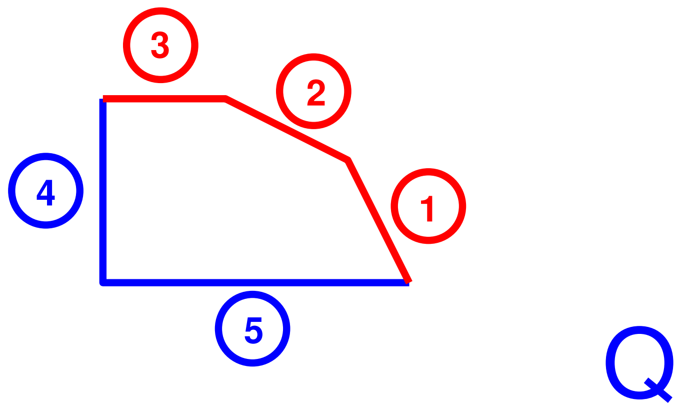
[Morris 1994] - exponentially long Lemke paths (finds symmetric equilibria of symmetric games)

[von Stengel 1999] - games with many equilibria

using **dual cyclic polytopes**

Vertices as bit patterns

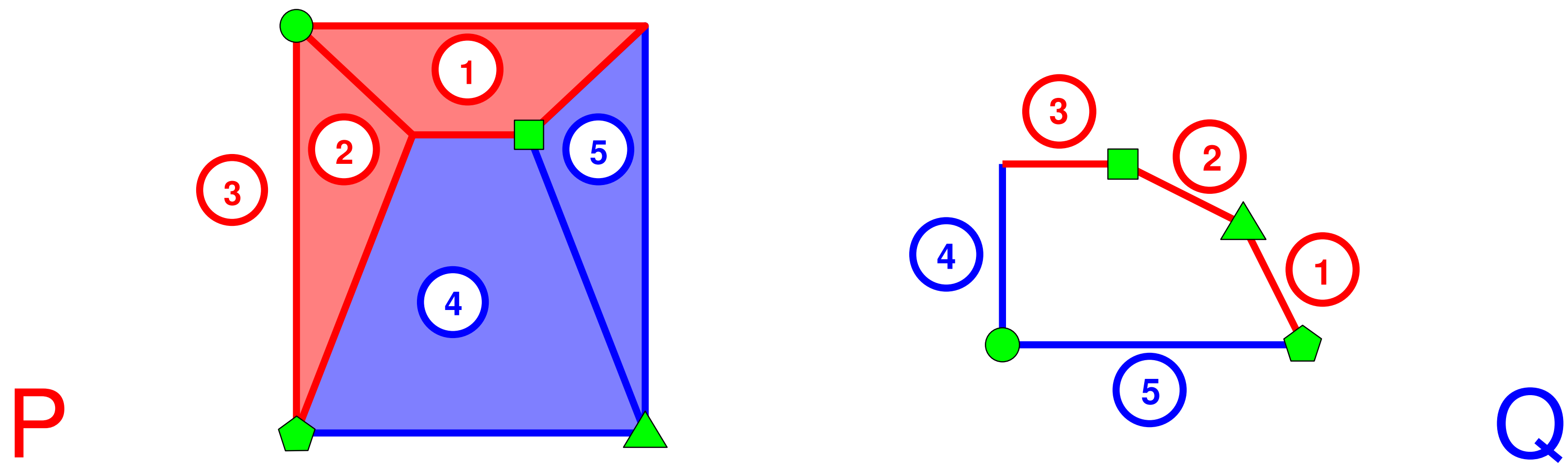




	1	2	3	4	5
	1	1	1	0	0
	1	1	0	1	0
	1	0	1	0	1
	1	0	0	1	1
	0	0	1	1	1
	0	1	1	1	0

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



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

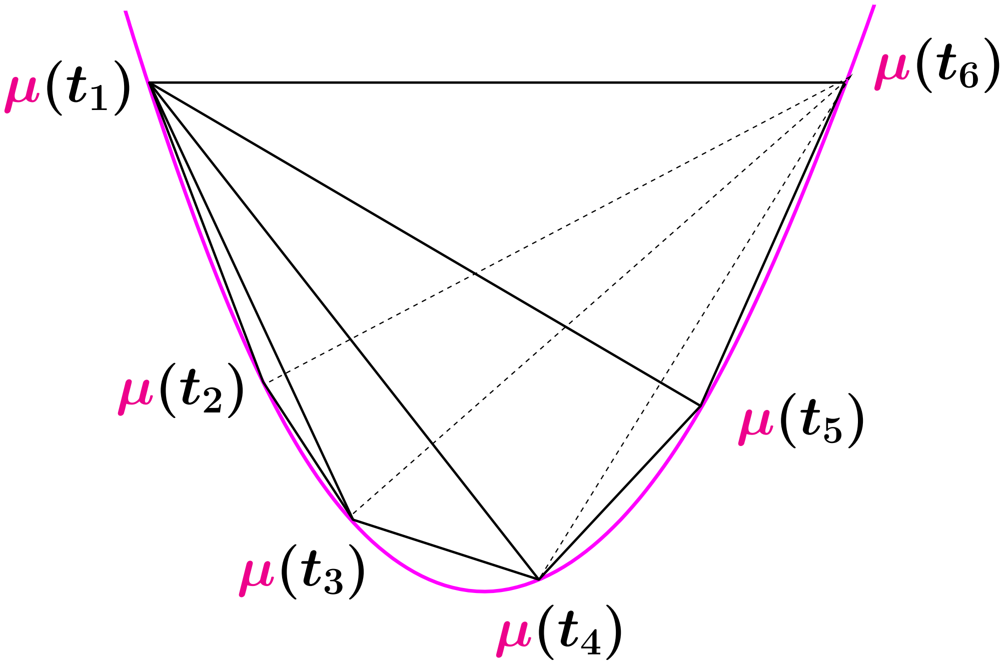
Cyclic polytopes

moment curve in $\mathbb{R}^d$

$$\mu : \mathbb{R} \longrightarrow \mathbb{R}^d \qquad t \mapsto \mu(t) = (t, t^2, \ldots, t^d)^\top.$$

cyclic polytope in dim d with N vertices: $\quad t_1 < t_2 < \cdots < t_N$

$$C_d(N) := \mathrm{conv}\{\mu(t_1), \ldots, \mu(t_N)\}$$

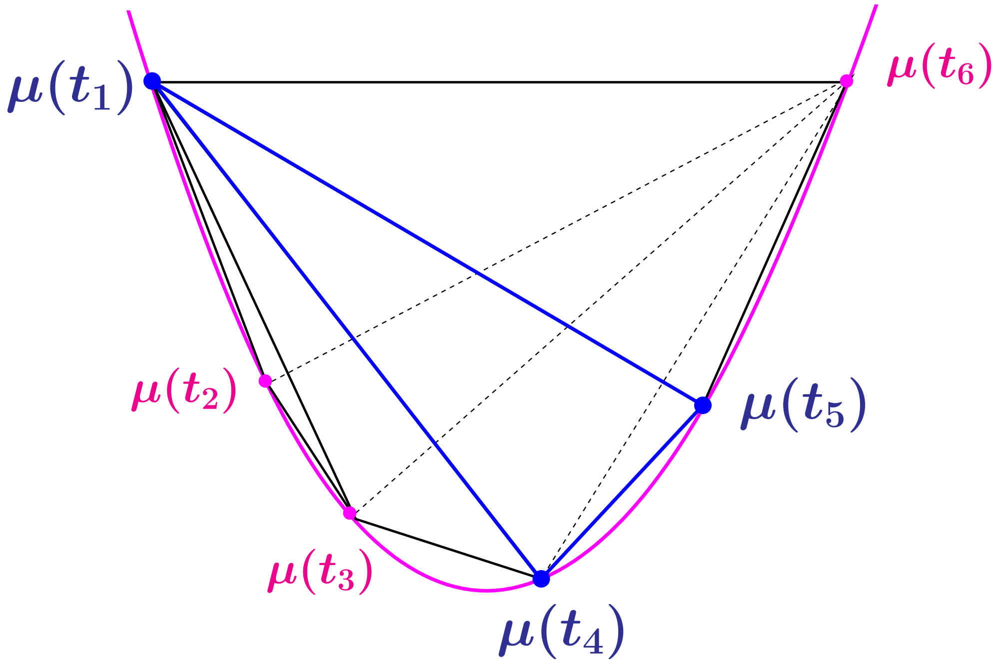


Facets of $C_d(N)$

Any d of the vertices $\mu(t_1), \ldots, \mu(t_N)$ define hyperplane F in $\mathbb{R}^d$.

F facet $\iff$ all **other** vertices are on one side of F

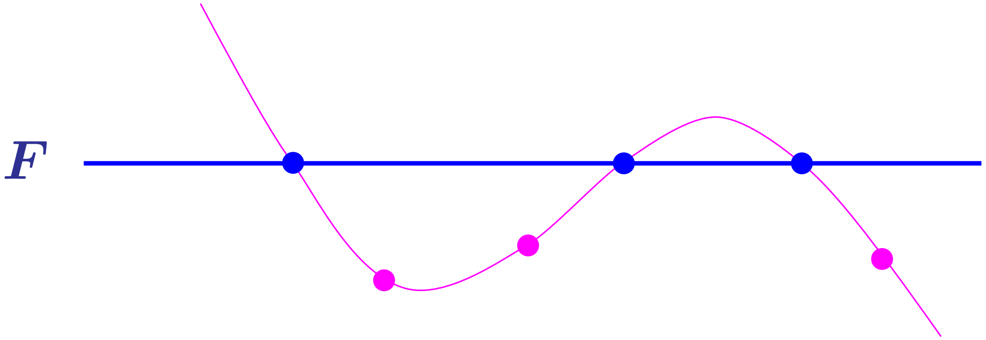
Example: $C_3(6)$, vertices 100110



Gale's Evenness condition

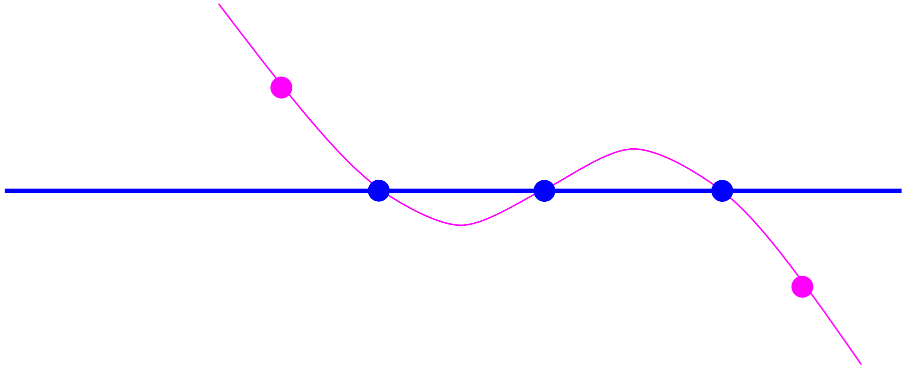
bitstring $s = s_1 s_2 \ldots s_N$, $s_i \in \{0, 1\}$ e.g. 100110

defines facet F = conv$\{\mu(t_i) \mid s_i = 1\}$ of $C_d(N)$



$\iff$ s has only even-length substrings 0110, 011110, 01111110,

forbidden: substrings 010, 01110, $\ldots$ of odd length.



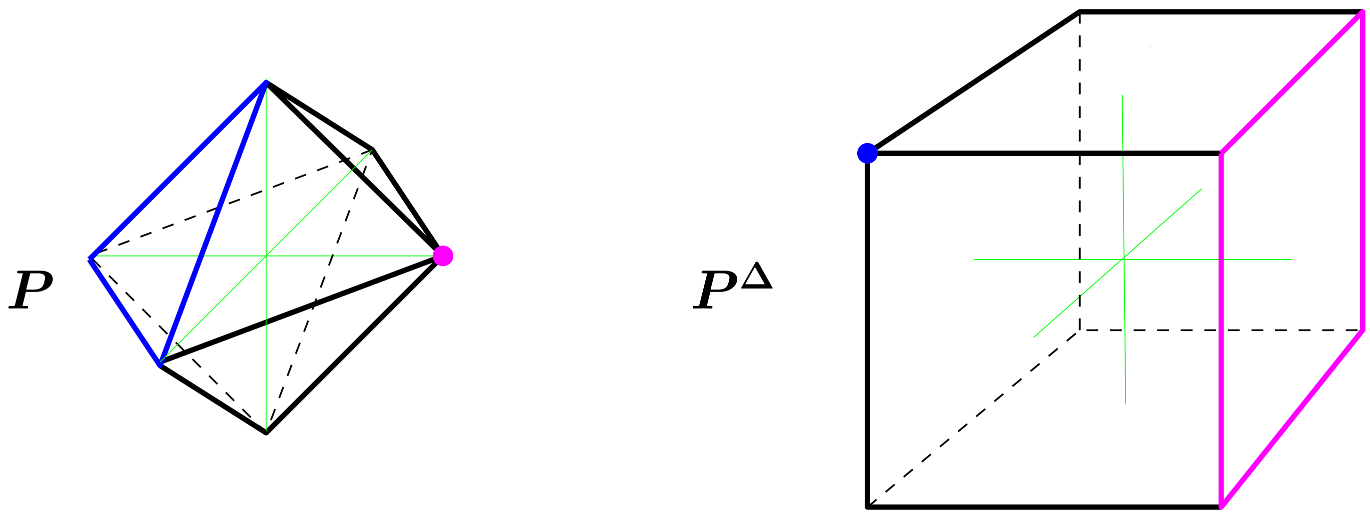
Polar polytopes

P = conv$\{c_1, \ldots, c_N\}$, $\quad 0 \in \mathrm{int}(P) \quad$ vertices c_i

polar polytope

$$P^\Delta = \{\, z \mid c_1{}^\top z \le 1, \ldots, \; c_N{}^\top z \le 1\}$$

facets $\{z \in P^\Delta \mid c_i{}^\top z = 1\}$



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	

Vertices of $C_d(2d)^\Delta$ and complementarity

$C_4(8)^\Delta$

vertex no.	defining facets	labels (example)
1	00001111	
2	00011011	
3	00011110	
4	00110011	
5	00110110	
6	00111100	
7	01100011	
8	01100110	② ③ ⑥ ⑦
9	01101100	
10	01111000	
11	10000111	
12	10001101	
13	10011001	① ④ ⑤ ⑧
14	10110001	
15	11000011	
16	11000110	
17	11001100	
18	11011000	
19	11100001	
20	11110000	

Permuted labels

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

with facets labeled 1 2 3 4 5 6 7 8 9 10 11 12

Q = P

with facets labeled 1 3 2 5 4 6 8 7 10 9 12 11

⟹ 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 ①	②	③	④	⑤	⑥	⑦	⑧	Q 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

1	2	3	4	5	6	7	8
1	1	1	1				
	1	1	1	1			
	1	1		1	1		
		1	1	1	1		
			1	1	1	1	
		1	1		1	1	
	1	1			1	1	
	1	1				1	1
		1	1			1	1
			1	1		1	1
				1	1	1	1

1	3	2	4	6	5	8	7
				1	1	1	1
			1	1		1	1
		1	1			1	1
	1	1				1	1
	1	1			1	1	
		1	1		1	1	
			1	1	1	1	
		1	1	1	1		
	1	1		1	1		
	1	1	1	1			
1	1	1	1				

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

	1	2	3	4	5	6	7	8	9	10	11	12	1	3	2	5	4	6	8	7	10	9	12	11
	1	1	1	1	1	1													1	1	1	1	1	1
1		1	1	1	1	1	1						1					1	1		1	1		1
2		1	1	1	1		1	1					1				1	1			1	1		1
3		1	1		1	1	1	1					1			1	1				1	1		1
4		1	1			1	1	1	1				1			1	1			1	1			1
5		1	1		1	1		1	1				1				1	1		1	1			1
6		1	1	1	1			1	1				1					1	1	1	1			1
7		1	1	1	1				1	1			1				1	1	1	1				1
8		1	1		1	1			1	1			1			1	1		1	1				1
9		1	1			1	1		1	1			1			1	1	1	1					1
10		1	1				1	1	1	1			1		1	1	1	1						1
11			1	1			1	1	1	1			1		1	1		1	1					1
12			1	1		1	1		1	1			1		1	1			1	1				1
13			1	1	1	1			1	1			1	1	1				1	1				1
14				1	1	1	1		1	1			1	1	1			1	1					1
15				1	1		1	1	1	1			1	1	1		1	1						1
16					1	1	1	1	1	1			1	1	1	1	1							1
17						1	1	1	1	1	1		1	1	1	1	1							
18							1	1	1	1	1	1	1	1	1	1	1	1						

A(4) is prefix of B(6)

	1	2	3	4	5	6	7	8	9	10	11	12	1	3	2	5	4	6	8	7	10	9	12	11
	1	1	1	1	1	1													1	1	1	1	1	1
1		1	1	1	1	1	1						1					1	1		1	1		1
2		1	1	1	1		1	1					1				1	1			1	1		1
3		1	1		1	1	1	1					1			1	1				1	1		1
4		1	1			1	1	1	1				1			1	1			1	1			1
5		1	1		1	1		1	1				1				1	1		1	1			1
6		1	1	1	1			1	1				1					1	1	1	1			1
7		1	1	1	1				1	1			1				1	1	1	1				1
8		1	1		1	1			1	1			1			1	1		1	1				1
9		1	1			1	1		1	1			1			1	1	1	1					1
10		1	1				1	1	1	1			1		1	1	1	1						1
11			1	1			1	1	1	1			1		1	1		1	1					1
12			1	1		1	1		1	1			1		1	1			1	1				1
13			1	1	1	1			1	1			1	1	1				1	1				1
14				1	1	1	1		1	1			1	1	1			1	1					1
15				1	1		1	1	1	1			1	1	1		1	1						1
16					1	1	1	1	1	1			1	1	1	1	1							1
17						1	1	1	1	1	1		1	1	1	1	1							
18							1	1	1	1	1	1	1	1	1	1	1	1						

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

	1	2	3	4	5	6	7	8	9	10	11	12	1	3	2	5	4	6	8	7	10	9	12	11
	1	1	1	1	1	1													1	1	1	1	1	1
1		1	1	1	1	1	1											1	1		1	1	1	1
2		1	1	1	1		1	1									1	1			1	1	1	1
3		1	1		1	1	1	1								1	1				1	1	1	1
4		1	1			1	1	1	1							1	1			1	1		1	1
5		1	1		1	1		1	1								1	1		1	1		1	1
6		1	1	1	1			1	1									1	1	1	1		1	1
7		1	1	1	1				1	1							1	1	1	1			1	1
8		1	1		1	1			1	1						1	1		1	1			1	1
9		1	1			1	1		1	1						1	1	1	1				1	1
10		1	1				1	1	1	1					1	1	1	1					1	1
11			1	1			1	1	1	1					1	1		1	1				1	1
12			1	1		1	1		1	1					1	1			1	1			1	1
13			1	1	1	1			1	1				1	1				1	1			1	1
14				1	1	1	1		1	1				1	1			1	1				1	1
15				1	1		1	1	1	1				1	1		1	1					1	1
16					1	1	1	1	1	1				1	1	1	1						1	1
17						1	1	1	1	1	1			1	1	1	1					1	1	
18					1	1	1	1		1	1			1	1		1	1				1	1	
19				1	1		1	1		1	1			1	1			1	1			1	1	
20				1	1	1	1			1	1			1	1				1	1		1	1	
21			1	1	1	1				1	1				1	1			1	1		1	1	
22			1	1		1	1			1	1				1	1		1	1			1	1	
23			1	1			1	1		1	1				1	1	1	1				1	1	
24		1	1				1	1		1	1					1	1	1	1			1	1	
25		1	1			1	1			1	1					1	1		1	1		1	1	
26		1	1		1	1				1	1						1	1	1	1		1	1	
27		1	1	1	1					1	1							1	1	1	1	1	1	
28		1	1	1	1						1	1					1	1	1	1	1	1		
29		1	1		1	1					1	1				1	1		1	1	1	1		
30		1	1			1	1				1	1				1	1	1	1		1	1		
31		1	1				1	1			1	1			1	1	1	1			1	1		
32			1	1			1	1			1	1			1	1		1	1		1	1		
33			1	1		1	1				1	1			1	1			1	1	1	1		
34			1	1	1	1					1	1		1	1				1	1	1	1		
35				1	1	1	1				1	1		1	1			1	1		1	1		
36				1	1		1	1			1	1		1	1		1	1			1	1		
37					1	1	1	1			1	1		1	1	1	1				1	1		
38						1	1	1	1		1	1		1	1	1	1			1	1			
39					1	1		1	1		1	1		1	1		1	1		1	1			
40				1	1			1	1		1	1		1	1			1	1	1	1			
41				1	1				1	1	1	1		1	1		1	1	1	1				
42					1	1			1	1	1	1		1	1	1	1		1	1				
43						1	1		1	1	1	1		1	1	1	1	1	1					
44							1	1	1	1	1	1	1	1	1	1	1	1						

B(6) is prefix of A(6)

	1	2	3	4	5	6	7	8	9	10	11	12	1	3	2	5	4	6	8	7	10	9	12	11
	1	1	1	1	1	1													1	1	1	1	1	1
1		1	1	1	1	1	1											1	1		1	1	1	1
2		1	1	1	1		1	1									1	1			1	1	1	1
3		1	1		1	1	1	1								1	1				1	1	1	1
4		1	1			1	1	1	1							1	1			1	1		1	1
5		1	1		1	1		1	1								1	1		1	1		1	1
6		1	1	1	1			1	1									1	1	1	1		1	1
7		1	1	1	1				1	1							1	1	1	1			1	1
8		1	1		1	1			1	1						1	1		1	1			1	1
9		1	1			1	1		1	1						1	1	1	1				1	1
10		1	1				1	1	1	1					1	1	1	1					1	1
11			1	1			1	1	1	1					1	1		1	1				1	1
12			1	1		1	1		1	1					1	1			1	1			1	1
13			1	1	1	1			1	1				1	1				1	1			1	1
14				1	1	1	1		1	1				1	1			1	1				1	1
15				1	1		1	1	1	1				1	1		1	1					1	1
16					1	1	1	1	1	1				1	1	1	1						1	1
17						1	1	1	1	1	1			1	1	1	1					1	1	
18					1	1	1	1		1	1			1	1		1	1				1	1	
19				1	1		1	1		1	1			1	1			1	1			1	1	
20				1	1	1	1			1	1			1	1				1	1		1	1	
21			1	1	1	1				1	1				1	1			1	1		1	1	
22			1	1		1	1			1	1				1	1		1	1			1	1	
23			1	1			1	1		1	1				1	1	1	1				1	1	
24		1	1				1	1		1	1					1	1	1	1			1	1	
25		1	1			1	1			1	1					1	1		1	1		1	1	
26		1	1		1	1				1	1						1	1	1	1		1	1	
27		1	1	1	1					1	1							1	1	1	1	1	1	
28		1	1	1	1						1	1					1	1	1	1	1	1		
29		1	1		1	1					1	1				1	1		1	1	1	1		
30		1	1			1	1				1	1				1	1	1	1		1	1		
31		1	1				1	1			1	1			1	1	1	1			1	1		
32			1	1			1	1			1	1			1	1		1	1		1	1		
33			1	1		1	1				1	1			1	1			1	1	1	1		
34			1	1	1	1					1	1		1	1				1	1	1	1		
35				1	1	1	1				1	1		1	1			1	1		1	1		
36				1	1		1	1			1	1		1	1		1	1			1	1		
37					1	1	1	1			1	1		1	1	1	1				1	1		
38						1	1	1	1		1	1		1	1	1	1			1	1			
39					1	1		1	1		1	1		1	1		1	1		1	1			
40				1	1			1	1		1	1		1	1			1	1	1	1			
41				1	1				1	1	1	1		1	1		1	1	1	1				
42					1	1			1	1	1	1		1	1	1	1		1	1				
43						1	1		1	1	1	1		1	1	1	1	1	1					
44							1	1	1	1	1	1	1	1	1	1	1	1						

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

	1	2	3	4	5	6	7	8	9	10	11	12	1	3	2	5	4	6	8	7	10	9	12	11
	1	1	1	1	1	1													1	1	1	1	1	1
1		1	1	1	1	1	1											1	1		1	1	1	1
2		1	1	1	1		1	1									1	1			1	1	1	1
3		1	1		1	1	1	1								1	1				1	1	1	1
4		1	1			1	1	1	1							1	1			1	1		1	1
5		1	1		1	1		1	1								1	1		1	1		1	1
6		1	1	1	1			1	1									1	1	1	1		1	1
7		1	1	1	1				1	1							1	1	1	1			1	1
8		1	1		1	1			1	1						1	1		1	1			1	1
9		1	1			1	1		1	1						1	1	1	1				1	1
10		1	1				1	1	1	1					1	1	1	1					1	1
11			1	1			1	1	1	1					1	1		1	1				1	1
12			1	1		1	1		1	1					1	1			1	1			1	1
13			1	1	1	1			1	1				1	1				1	1			1	1
14				1	1	1	1		1	1				1	1			1	1				1	1
15				1	1		1	1	1	1				1	1		1	1					1	1
16					1	1	1	1	1	1				1	1	1	1						1	1
17						1	1	1	1	1	1			1	1	1	1					1	1	
18					1	1	1	1		1	1			1	1		1	1				1	1	
19				1	1		1	1		1	1			1	1			1	1			1	1	
20				1	1	1	1			1	1			1	1				1	1		1	1	
21			1	1	1	1				1	1				1	1			1	1		1	1	
22			1	1		1	1			1	1				1	1		1	1			1	1	
23			1	1			1	1		1	1				1	1	1	1				1	1	
24		1	1				1	1		1	1					1	1	1	1			1	1	
25		1	1			1	1			1	1					1	1		1	1		1	1	
26		1	1		1	1				1	1						1	1	1	1		1	1	
27		1	1	1	1					1	1							1	1	1	1	1	1	
28		1	1	1	1						1	1					1	1	1	1	1	1		
29		1	1		1	1					1	1				1	1		1	1	1	1		
30		1	1			1	1				1	1				1	1	1	1		1	1		
31		1	1				1	1			1	1			1	1	1	1			1	1		
32			1	1			1	1			1	1			1	1		1	1		1	1		
33			1	1		1	1				1	1			1	1			1	1	1	1		
34			1	1	1	1					1	1		1	1				1	1	1	1		
35				1	1	1	1				1	1		1	1			1	1		1	1		
36				1	1		1	1			1	1		1	1		1	1			1	1		
37					1	1	1	1			1	1		1	1	1	1				1	1		
38						1	1	1	1		1	1		1	1	1	1			1	1			
39					1	1		1	1		1	1		1	1		1	1		1	1			
40				1	1			1	1		1	1		1	1			1	1	1	1			
41				1	1				1	1	1	1		1	1		1	1	1	1				
42					1	1			1	1	1	1		1	1	1	1		1	1				
43						1	1		1	1	1	1		1	1	1	1	1	1					
44							1	1	1	1	1	1	1	1	1	1	1	1						

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 . . .

Growth rate of Fibonacci numbers

n	F_n	F_{n+1}	F_{n+1}/F_n
1	1	1	1.0
2	1	2	2.0
3	2	3	1.5
4	3	5	1.66666666667
5	5	8	1.6
6	8	13	1.625
7	13	21	1.61538461538
8	21	34	1.61904761905
9	34	55	1.61764705882
10	55	89	1.61818181818
11	89	144	1.61797752809
12	144	233	1.61805555556
13	233	377	1.61802575107
14	377	610	1.61803713528

Growth rate of Fibonacci numbers

Successive values of F_{n+1}/F_n seem to "converge" to a certain number, about $\mathbf{1.618\ldots}$.

That is, for larger n the Fibonacci numbers seem to grow at this rate according to

$$F_{n+1} \approx \mathbf{1.618\ldots} \cdot F_n$$

Growth rate of Fibonacci numbers

Successive values of F_{n+1}/F_n seem to "converge" to a certain number, about $\mathbf{1.618\ldots}$.

That is, for larger n the Fibonacci numbers seem to grow at this rate according to

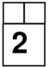
$$F_{n+1} \approx 1.618\ldots \cdot F_n$$

The following is a geometric plausibility argument (not a proof) for this specific growth rate.

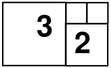
The Fibonacci spiral

The Fibonacci spiral

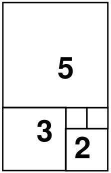
The Fibonacci spiral



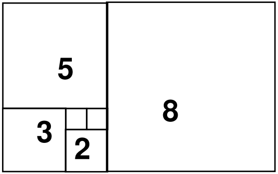
The Fibonacci spiral



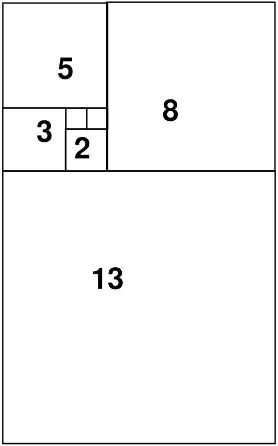
The Fibonacci spiral



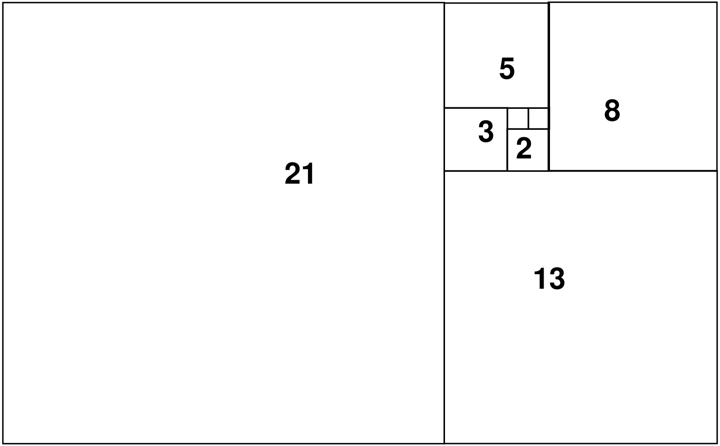
The Fibonacci spiral



The Fibonacci spiral



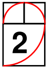
The Fibonacci spiral



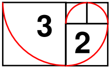
The Fibonacci spiral

The Fibonacci spiral

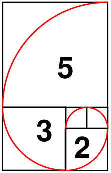
The Fibonacci spiral



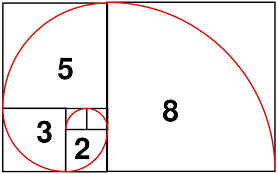
The Fibonacci spiral



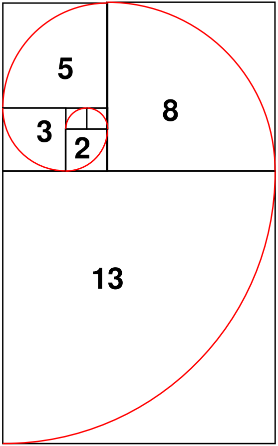
The Fibonacci spiral



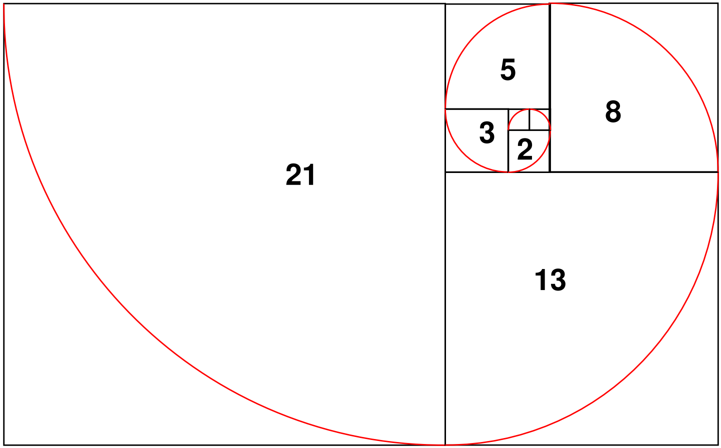
The Fibonacci spiral



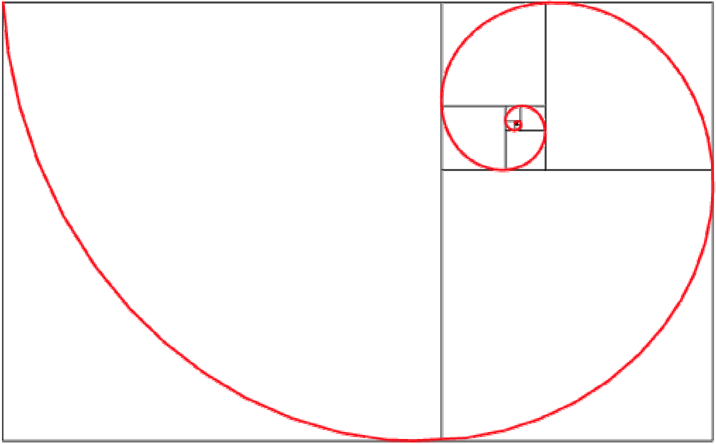
The Fibonacci spiral



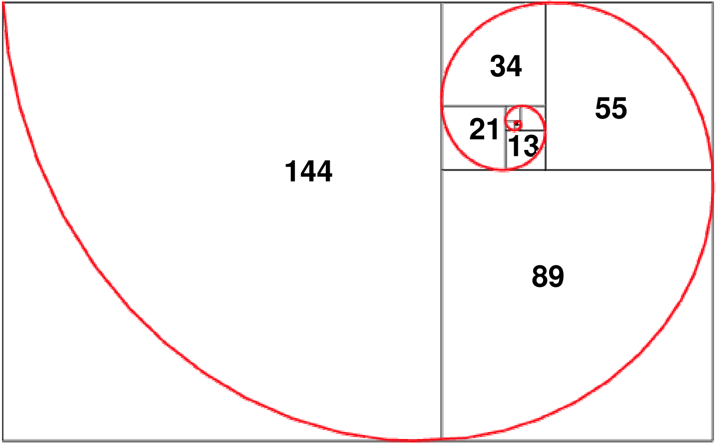
The Fibonacci spiral



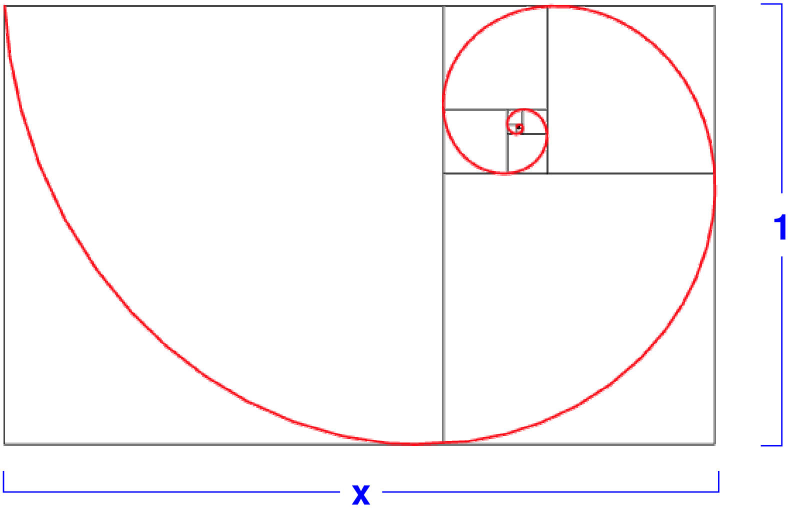
The Fibonacci spiral



The Fibonacci spiral

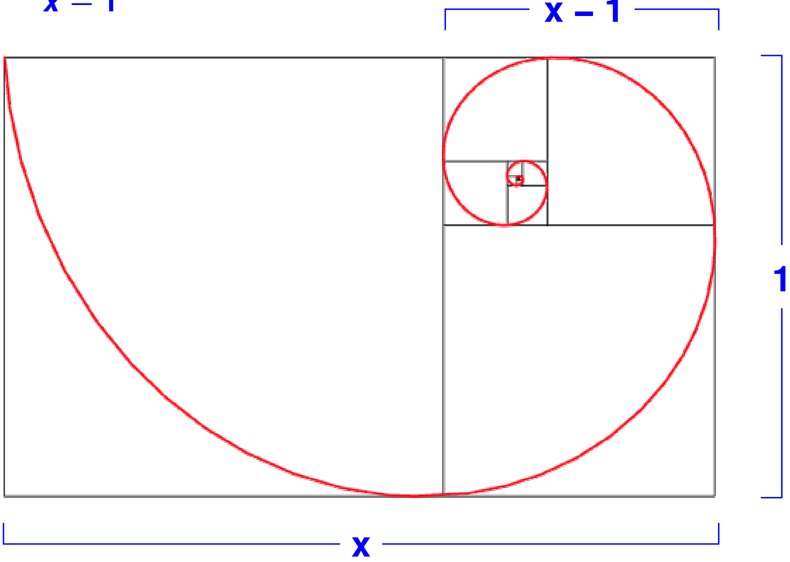


The Fibonacci spiral



The Fibonacci spiral

$\frac{x}{1} = \frac{1}{x-1}$, so $x \cdot (x-1) = 1$



The Golden Ratio

The solutions to $x \cdot (x - 1) = 1$, that is, $x^2 - x - 1 = 0$, are

$$x = \frac{1}{2} \pm \sqrt{\frac{1}{4} + 1} = \frac{1 \pm \sqrt{5}}{2} .$$

The Golden Ratio

The solutions to $x \cdot (x - 1) = 1$, that is, $x^2 - x - 1 = 0$, are

$$x = \frac{1}{2} \pm \sqrt{\frac{1}{4} + 1} = \frac{1 \pm \sqrt{5}}{2} .$$

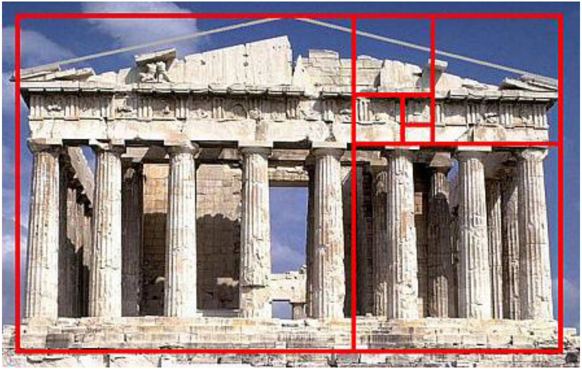
The number $(1 + \sqrt{5})/2 \approx 1.618033988$ is known as the **Golden Ratio** $\boxed{\phi}$.

The Golden Ratio

The solutions to $x \cdot (x - 1) = 1$, that is, $x^2 - x - 1 = 0$, are

$$x = \frac{1}{2} \pm \sqrt{\frac{1}{4} + 1} = \frac{1 \pm \sqrt{5}}{2} \, .$$

The number $(1 + \sqrt{5})/2 \approx 1.618033988$ is known as the **Golden Ratio** $\boxed{\phi}$.

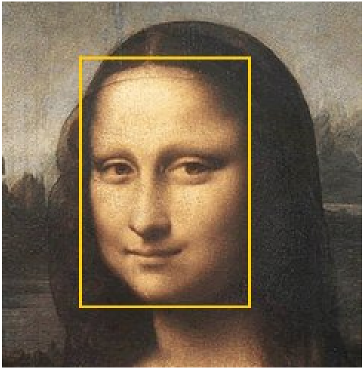


The Golden Ratio

The solutions to $x \cdot (x - 1) = 1$, that is, $x^2 - x - 1 = 0$, are

$$x = \frac{1}{2} \pm \sqrt{\frac{1}{4} + 1} = \frac{1 \pm \sqrt{5}}{2} .$$

The number $(1 + \sqrt{5})/2 \approx 1.618033988$ is known as the **Golden Ratio** $\boxed{\phi}$.



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+2)

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

Summary and extensions

– 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

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

The 1984 song „The longest time“ by Billy Joel was given the following „computer science“ version by Daniel Barrett, who wrote it as a graduate student at Johns Hopkins University, „on May 1, 1988, during a difficult Algorithms II final exam“, and subsequently recorded it.

Woh oh-oh-oh find the longest path
Woh oh-oh find the longest path.

If you say P is NP tonight
there would still be
papers left to write
I have a weakness
I'm addicted to completeness
and I keep searching
for the longest path.

The algorithm I would like to see
is of polynomial degree
but it's elusive
nobody has found conclusive
evidence that we can find
the longest path.

I have been
hard working for so long
I swear it's right
and he marks it wrong
somehow I feel
sorry when it's done
GPA 2.1
is more than I hope for

Garey, Johnson,
Karp and other men (and women, too)
try to make it order N log N
am I a mad fool
if I spend my life in grad school
forever following the longest path

Woh oh-oh-oh find the longest path
Woh oh-oh find the longest path
Woh oh-oh find the longest path.