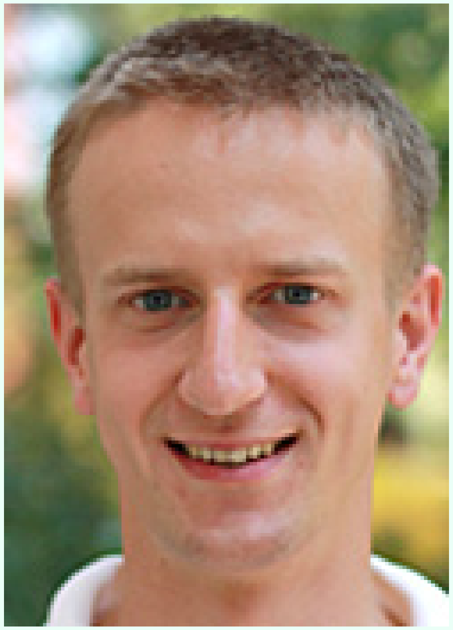
Computing Game-Theoretic Solutions for Security

Vincent Conitzer



Dmytro Korzhyk

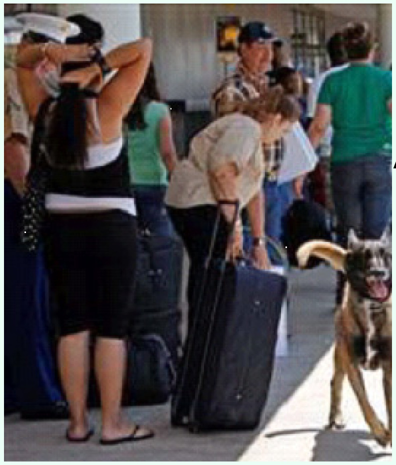
Joshua Letchford

Duke University

overview article: V. Conitzer. Computing Game-Theoretic Solutions and Applications to Security. *Proc. AAAI'12.*

Real-world security applications

Milind Tambe's TEAMCORE group (USC)



Airport security

- Where should checkpoints, canine units, etc. be deployed?

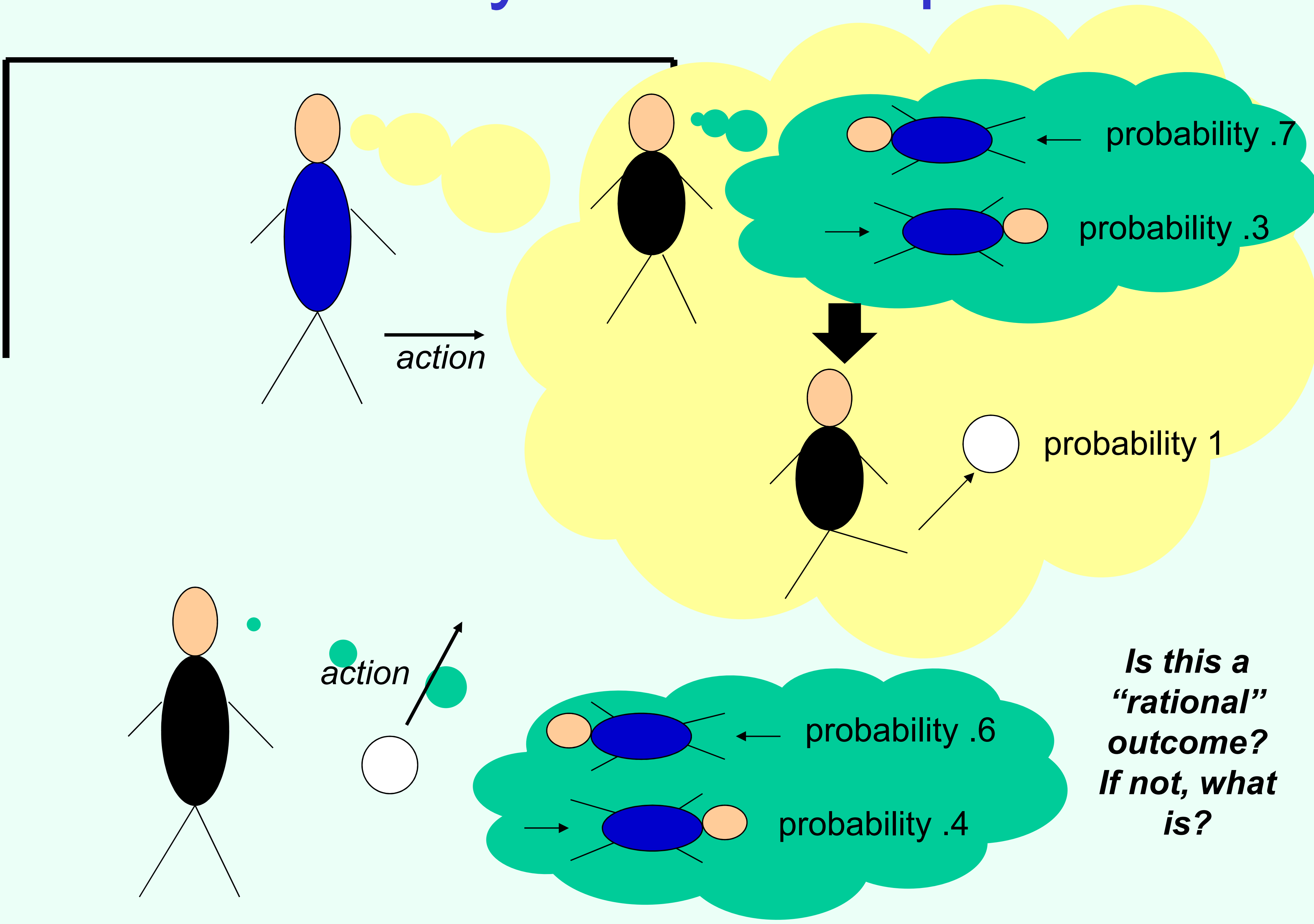
Federal Air Marshals

- Which flights get a FAM?

US Coast Guard

- Which patrol routes should be followed?

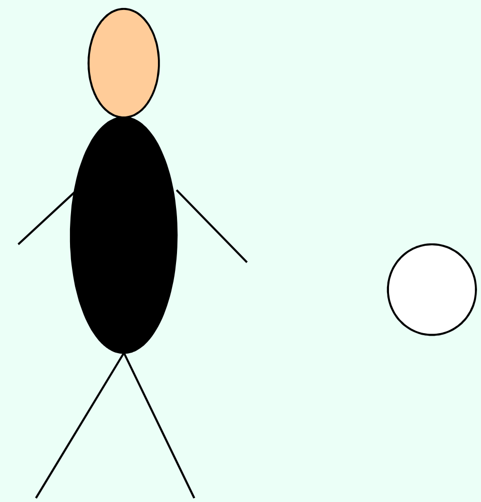
Penalty kick example

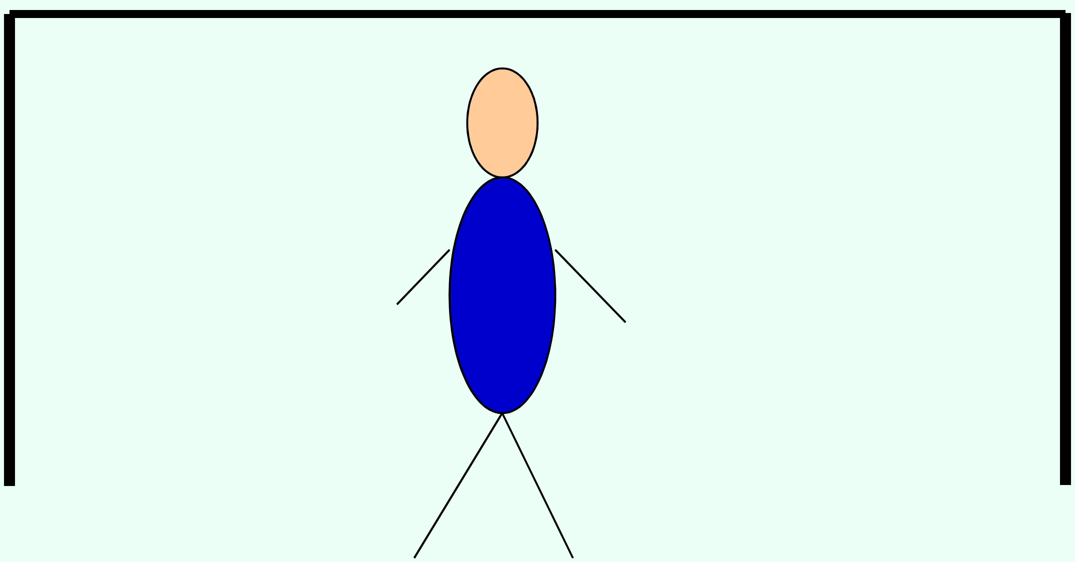


Is this a "rational" outcome? If not, what is?

Penalty kick

(also known as: matching pennies)

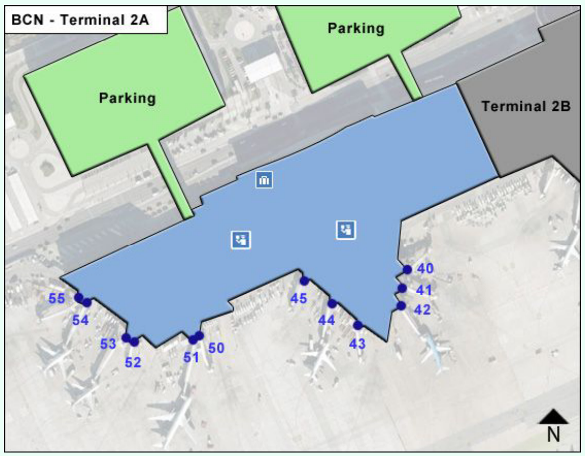


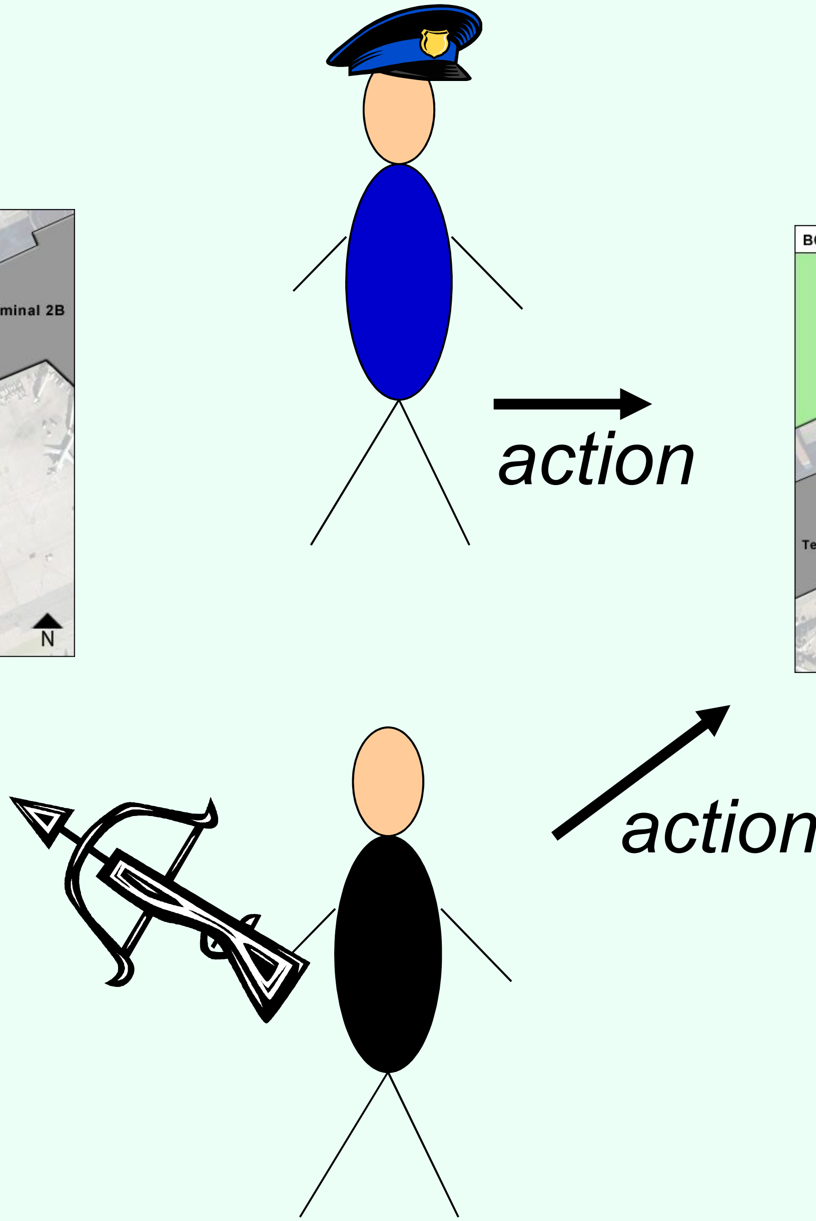


		.5	.5
		L	R
.5	L	0, 0	-1, 1
.5	R	-1, 1	0, 0

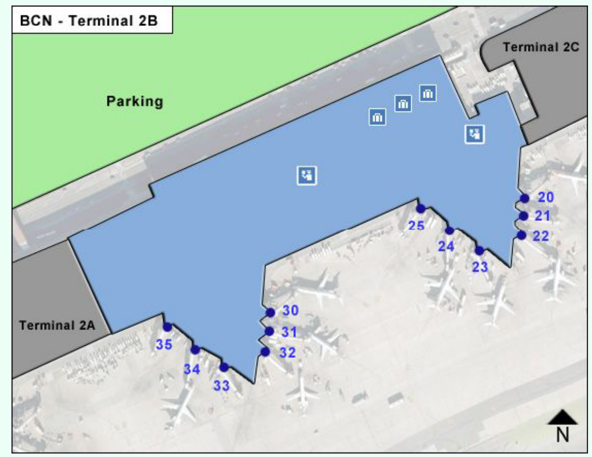
Security example

Terminal A

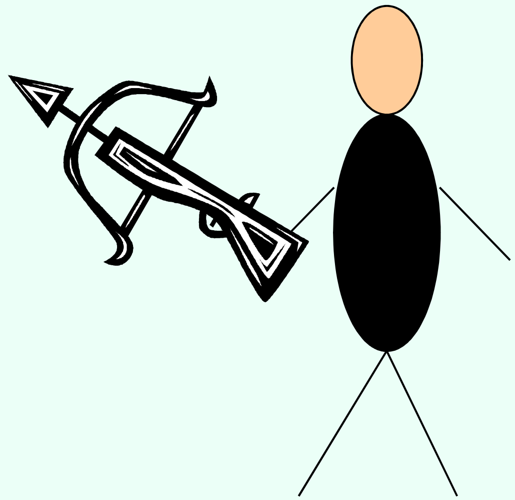


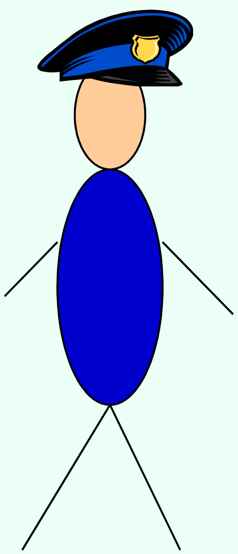


Terminal B



Security game





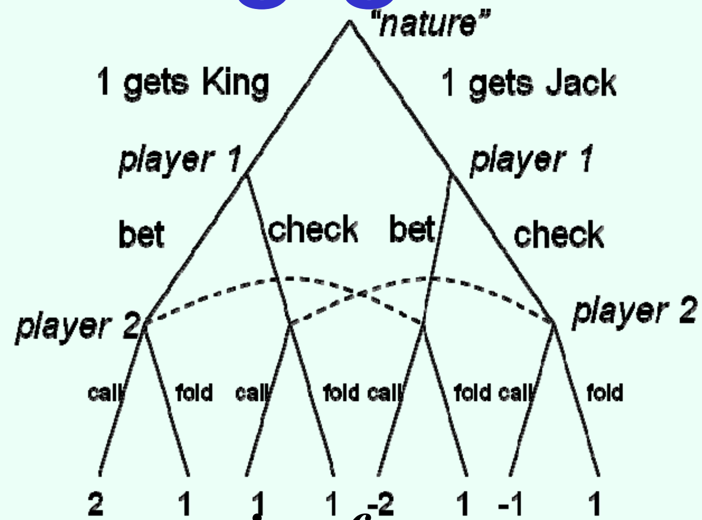
	A	B
A	0, 0	-1, 2
B	-1, 1	0, 0

Modeling and representing games

THIS TALK (unless specified otherwise)

2, 2	-1, 0
-7, -8	0, 0

normal-form games



extensive-form games

row player type 1 (prob. 0.5)

	L	R
U	4	6
D	2	4

column player type 1 (prob. 0.5)

	L	R
U	4	6
D	4	6

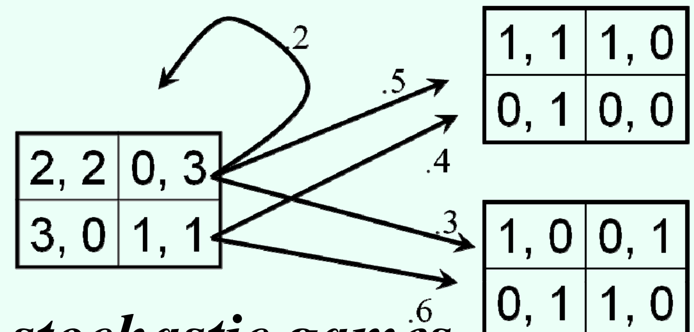
row player type 2 (prob. 0.5)

	L	R
U	2	4
D	4	2

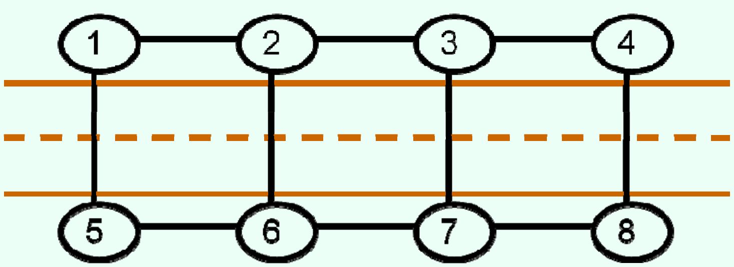
column player type 2 (prob. 0.5)

	L	R
U	2	2
D	4	2

Bayesian games

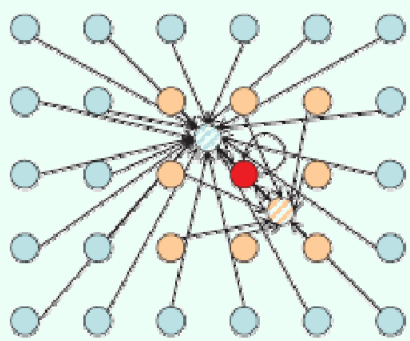


stochastic games



graphical games

[Kearns, Littman, Singh UAI'01]

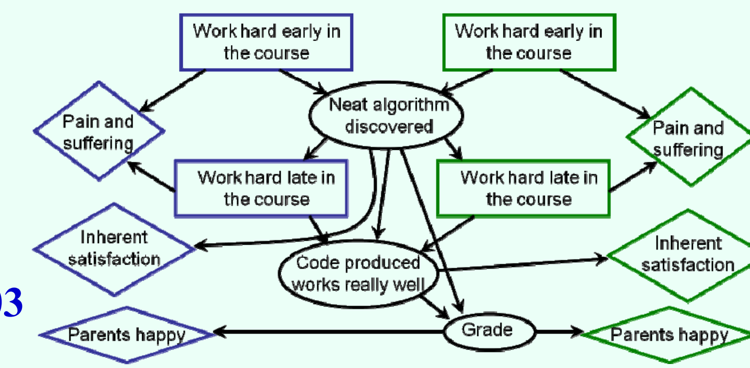


action-graph games

[Leyton-Brown & Tennenholtz IJCAI'03

[Bhat & Leyton-Brown, UAI'04]

[Jiang, Leyton-Brown, Bhat GEB'11]



MAIDs

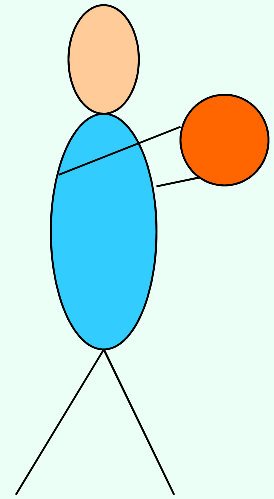
[Koller & Milch. IJCAI'01/GEB'03]

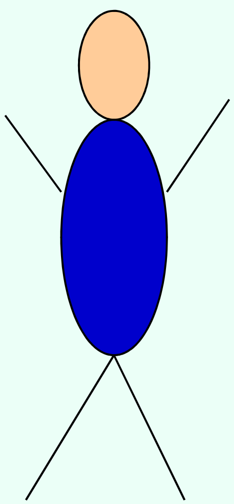
How to defend penalties

		Them L	*Them* R
Us	L	0, 0	-1, 1
Us	R	-1, 1	0, 0

- Assume opponent knows our strategy…
  - hopeless?
- … but we can use randomization
- If we play L 60%, R 40%...
- … opponent will play R…
- … we get .6*(-1) + .4*(0) = -.6
- Better: L 50%, R 50% guarantees -.5 (optimal)

A locally more popular sport





	go for 3	go for 2
defend the 3	0, 0	-2, 2
defend the 2	-3, 3	0, 0

Solving basketball

		Them 3	2
Us	3	0, 0	-2, 2
	2	-3, 3	0, 0

- If we 50% of the time defend the 3, opponent will shoot 3
  - We get .5*(-3) + .5*(0) = -1.5
- Should defend the 3 more often: 60% of the time
- Opponent has choice between
  - Go for 3: gives them .6*(0) + .4*(3) = 1.2
  - Go for 2: gives them .6*(2) + .4*(0) = 1.2
- We get -1.2 (the maximin value)

Let's change roles

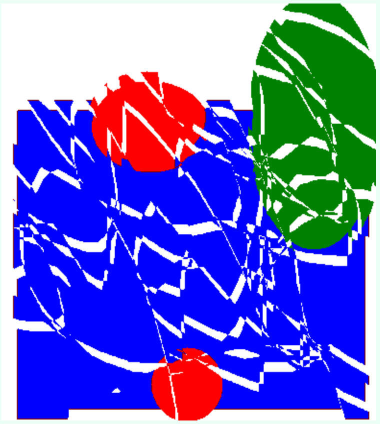
		Them 3	2
Us	3	0, 0	-2, 2
	2	-3, 3	0, 0

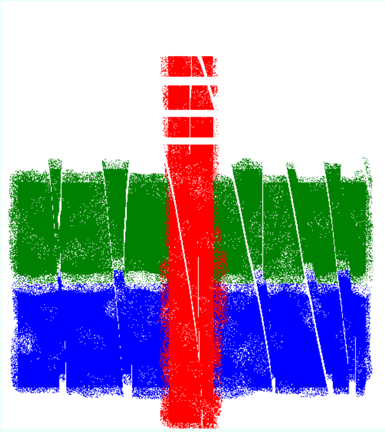
- Suppose **we** know **their** strategy
- If 50% of the time they go for 3, then we defend 3
  - We get .5*(0)+.5*(-2) = -1
- Optimal for them: 40% of the time go for 3
  - If we defend 3, we get .4*(0)+.6*(-2) = -1.2
  - If we defend 2, we get .4*(-3)+.6*(0) = -1.2
- This is the minimax value

von Neumann's minimax theorem [1928]: maximin value = minimax value
(~ linear programming duality)

Example linear program

- We make reproductions of two paintings





- Painting 1 sells for \$3, painting 2 sells for \$2
- Painting 1 requires 4 units of blue, 1 green, 1 red
- Painting 2 requires 2 blue, 2 green, 1 red
- We have 16 units blue, 8 green, 5 red

maximize $3x + 2y$

subject to

$4x + 2y \leq 16$

$x + 2y \leq 8$

$x + y \leq 5$

$x \geq 0$

$y \geq 0$

Solving the linear program graphically

maximize $3x + 2y$

subject to

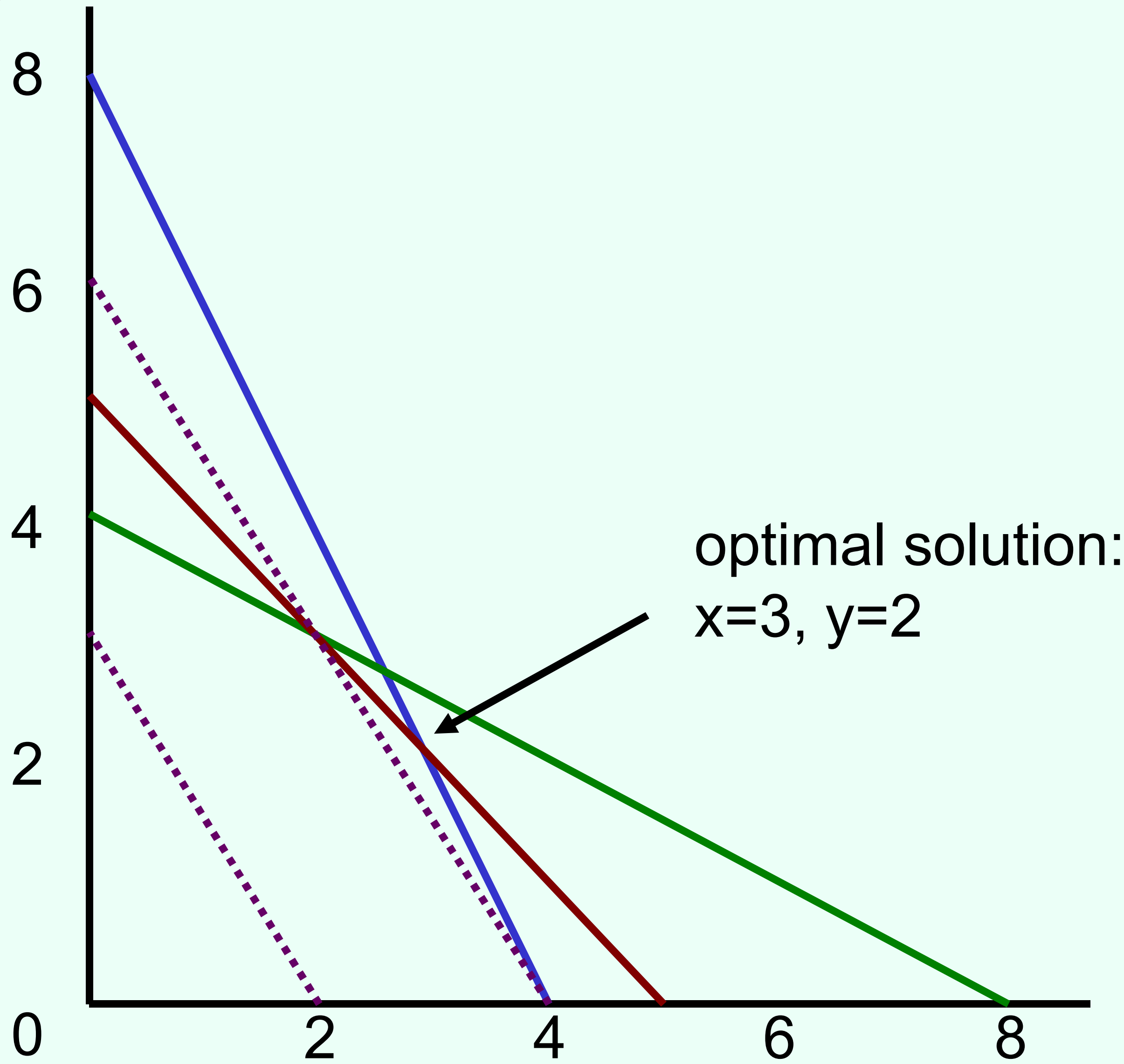
$4x + 2y \leq 16$

$x + 2y \leq 8$

$x + y \leq 5$

$x \geq 0$

$y \geq 0$



Solving for minimax strategies using linear programming

- maximize u
- subject to

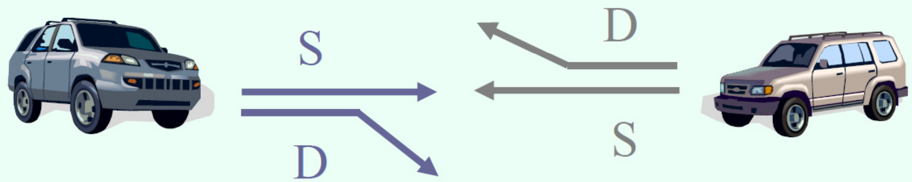
for any c, $\Sigma_r\, p_r\, u_R(r, c) \geq u$

$\Sigma_\mathrm{r}\, p_r = 1$

Can also convert linear programs to two-player zero-sum games, so they are equivalent

Some of the questions raised

- Equilibrium selection?



	D	S
D	0, 0	-1, 1
S	1, -1	-5, -5

- How should we model temporal / information structure?

2, 2	-1, 0
-7, -8	0, 0

row player type 1 (prob. 0.5)

	L	R
U	4	6
D	2	4

column player type 1 (prob. 0.5)

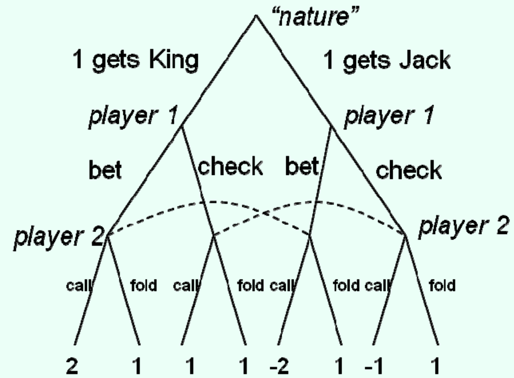
	L	R
U	4	6
D	4	6

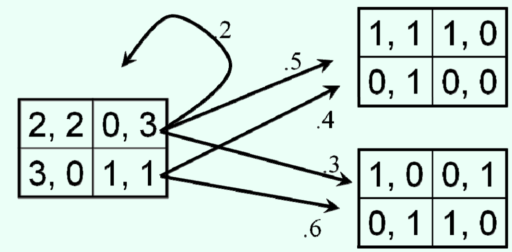
row player type 2 (prob. 0.5)

	L	R
U	2	4
D	4	2

column player type 2 (prob. 0.5)

	L	R
U	2	2
D	4	2

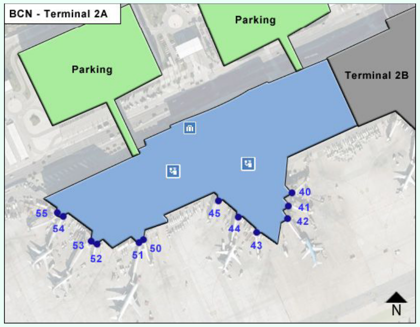




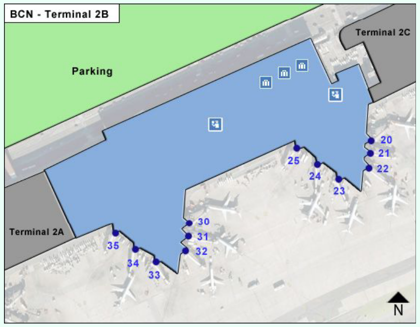
- What structure should utility functions have?
- Do our algorithms scale?

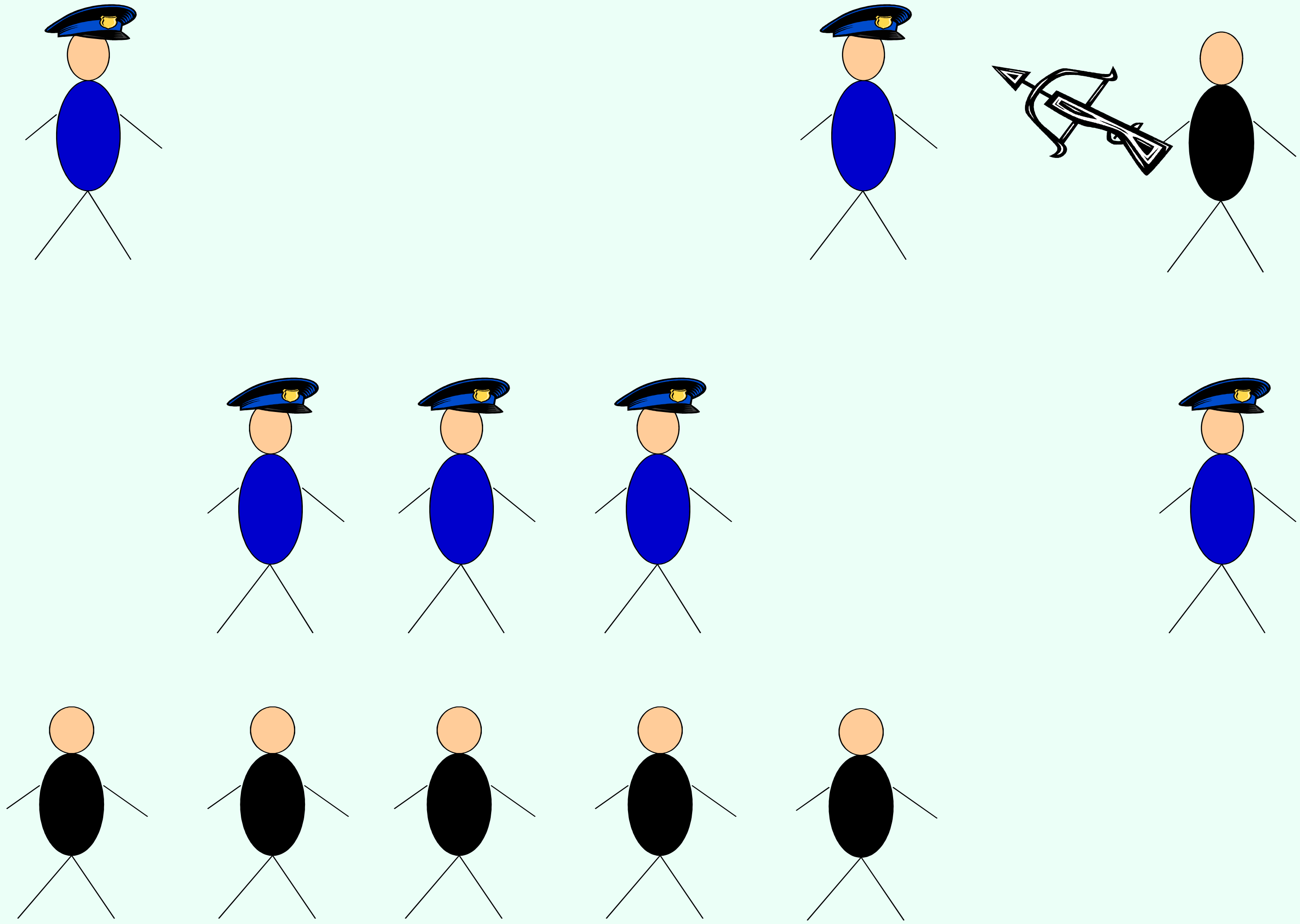
Observing the defender's distribution in security

Terminal A



Terminal B





observe

Mo Tu We Th Fr Sa

This model is not uncontroversial… [Pita, Jain, Tambe, Ordóñez, Kraus AIJ'10; Korzhyk, Yin, Kiekintveld, C., Tambe JAIR'11; Korzhyk, C., Parr AAMAS'11]

Commitment

Unique Nash equilibrium

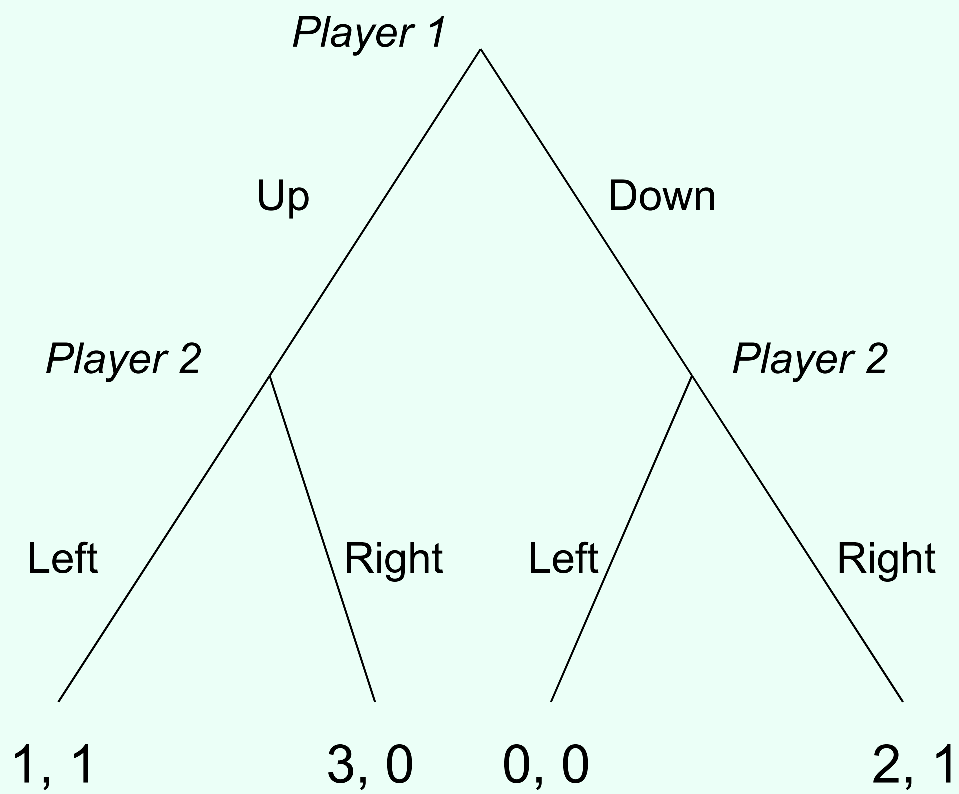
1, 1	3, 0
0, 0	2, 1

von Stackelberg

- Suppose the game is played as follows:
  - Player 1 commits to playing one of the rows,
  - Player 2 observes the commitment and then chooses a column
- Optimal strategy for player 1: commit to Down

Commitment as an extensive-form game

- For the case of committing to a pure strategy:



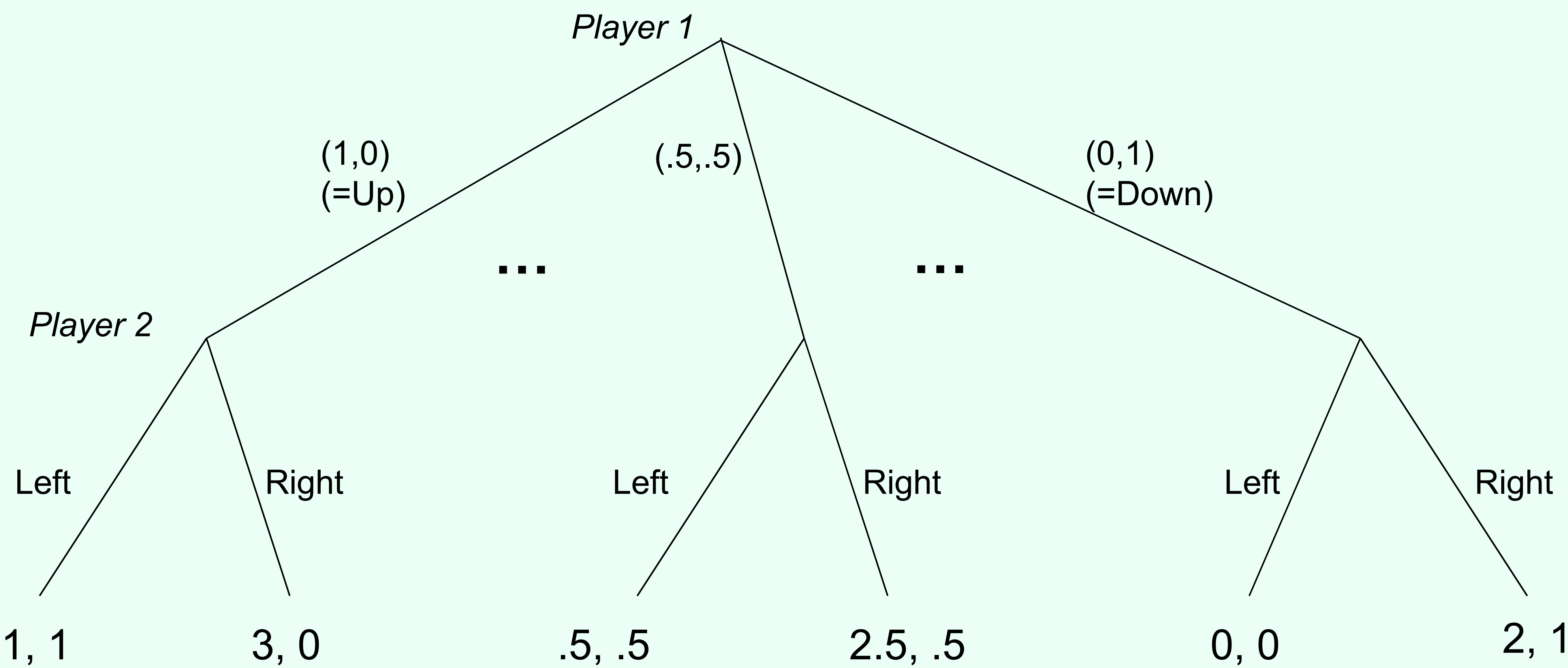
Commitment to mixed strategies

	0	1
.49	1, 1	3, 0
.51	0, 0	2, 1

– Sometimes also called a Stackelberg (mixed) strategy

Commitment as an extensive-form game…

- … for the case of committing to a mixed strategy:



- Economist: Just an extensive-form game, nothing new here
- Computer scientist: Infinite-size game! Representation matters

Computing the optimal mixed strategy to commit to

[C. & Sandholm EC'06, von Stengel & Zamir GEB'10]

- Separate LP for every column c^*:

maximize $\Sigma_r\, p_r\, u_R(r, c^*)$ leader utility

subject to

for all c, $\Sigma_r\, p_r\, u_C(r, c^*) \geq \Sigma_r\, p_r\, u_C(r, c)$ follower optimality

$\Sigma_r\, p_r = 1$ distributional constraint

… applied to the previous game

p	1, 1	3, 0
q	0, 0	2, 1

maximize $1p + 0q$

subject to

$1p + 0q \geq 0p + 1q$

$p + q = 1$

$p \geq 0$

$q \geq 0$

maximize $3p + 2q$

subject to

$0p + 1q \geq 1p + 0q$

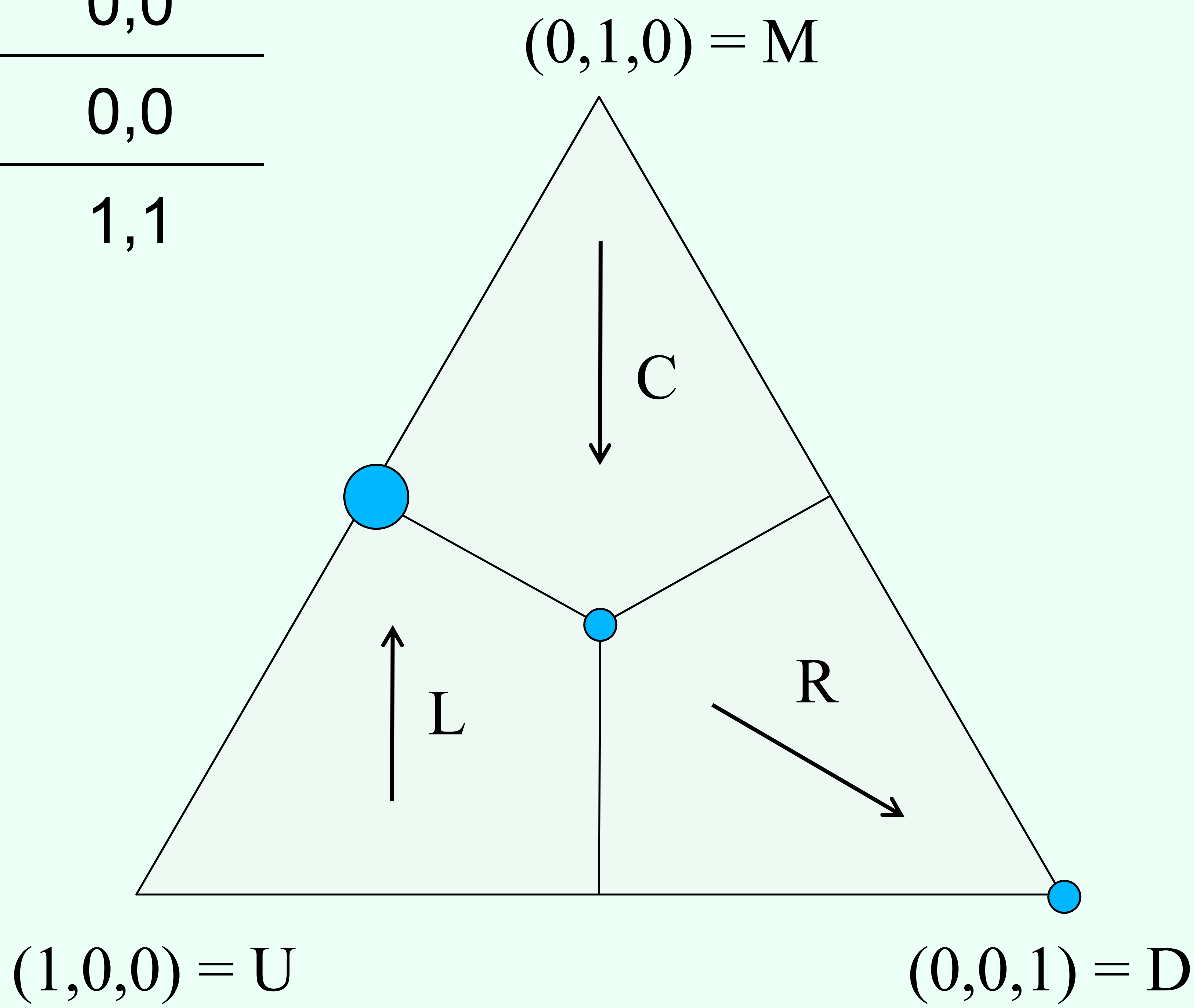
$p + q = 1$

$p \geq 0$

$q \geq 0$

Visualization

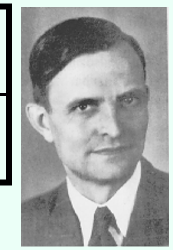
	L	C	R
U	0,1	1,0	0,0
M	4,0	0,1	0,0
D	0,0	1,0	1,1



Other nice properties of commitment to mixed strategies

- Agrees w. Nash in zero-sum games

0, 0	-1, 1
-1, 1	0, 0



- Leader's payoff at least as good as any Nash eq. or even correlated eq. (von Stengel & Zamir [GEB '10]; see also C. & Korzhyk [AAAI '11], Letchford, Korzhyk, C. [JAAMAS '14])

$\geq$

- No equilibrium selection problem

0, 0	-1, 1
1, -1	-5, -5

More discussion: V. Conitzer. On Stackelberg Mixed Strategies. [Synthese, to appear.]

Example security game

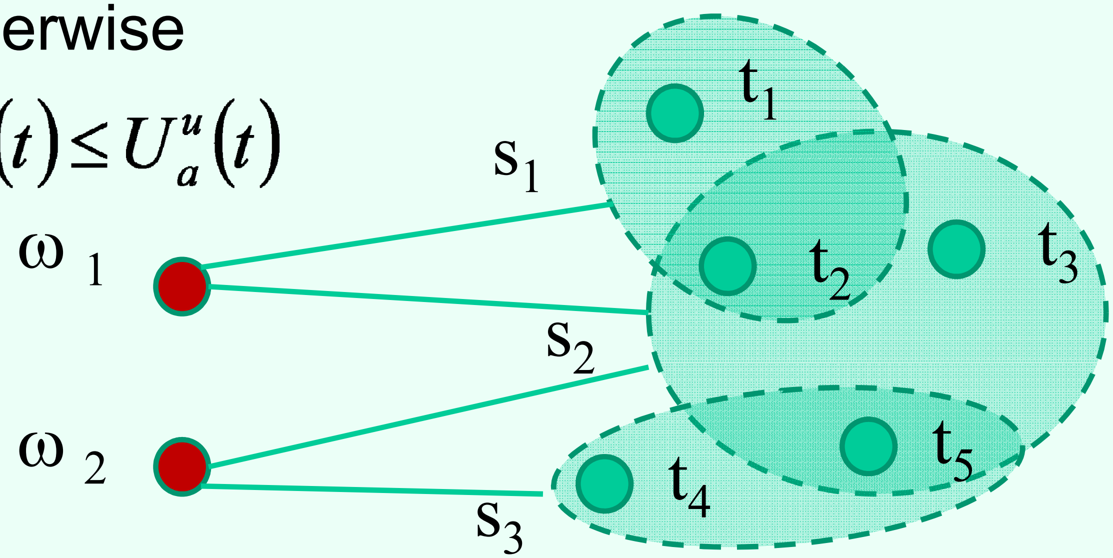
- 3 airport terminals to defend (A, B, C)
- Defender can place checkpoints at 2 of them
- Attacker can attack any 1 terminal

	A	B	C
{A, B}	0, -1	0, -1	-2, 3
{A, C}	0, -1	-1, 1	0, 0
{B, C}	-1, 1	0, -1	0, 0

Security resource allocation games

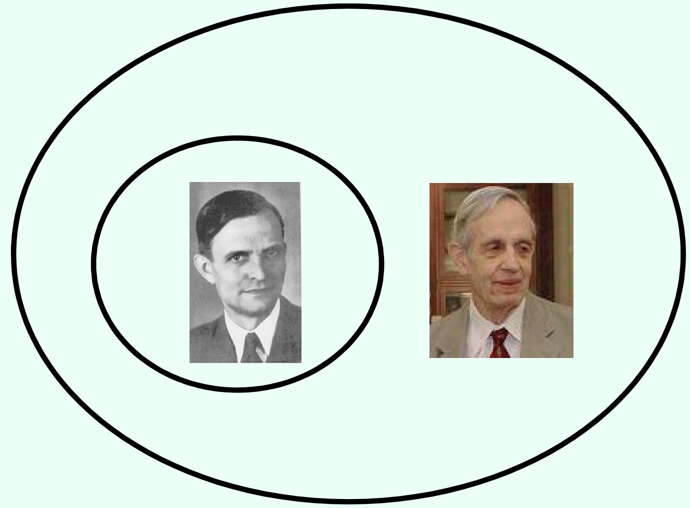
[Kiekintveld, Jain, Tsai, Pita, Ordóñez, Tambe AAMAS'09]

- Set of targets *T*
- Set of security resources Ω available to the defender (leader)
- Set of schedules $S \subseteq 2^T$
- Resource ω can be assigned to one of the schedules in $A(\omega) \subseteq S$
- Attacker (follower) chooses one target to attack
- Utilities: $U_d^c(t), U_a^c(t)$ if the attacked target is defended, $U_d^u(t), U_a^u(t)$ otherwise
- $U_d^c(t) \geq U_d^u(t); U_a^c(t) \leq U_a^u(t)$



Game-theoretic properties of security resource allocation games [Korzhyk, Yin, Kiekintveld, C., Tambe JAIR'11]

- For the defender: Stackelberg strategies are also Nash strategies
  - minor assumption needed
  - not true with multiple attacks



- Interchangeability property for Nash equilibria ("solvable")
  - no equilibrium selection problem
  - still true with multiple attacks [Korzhyk, C., Parr IJCAI'11]




1, 2	1, 0	2, 2
1, 1	1, 0	2, 1
0, 1	0, 0	0, 1

Compact LP

- Cf. ERASER-C algorithm by Kiekintveld et al. [2009]
- Separate LP for every possible t* attacked:

$$\text{Maximize } U_d^c(t^*)\sum_{\omega}\sum_{s:t^*\in s} c_{\omega,s} + U_d^u(t^*)\left(1-\sum_{\omega}\sum_{s:t^*\in s} c_{\omega,s}\right)$$

Defender utility

Marginal probability of t^* being defended (?)

Subject to

$$\forall \omega : \sum_{s} c_{\omega,s} \le 1$$

$$\forall t : \sum_{\omega}\sum_{s:t\in s} c_{\omega,s} \le 1$$

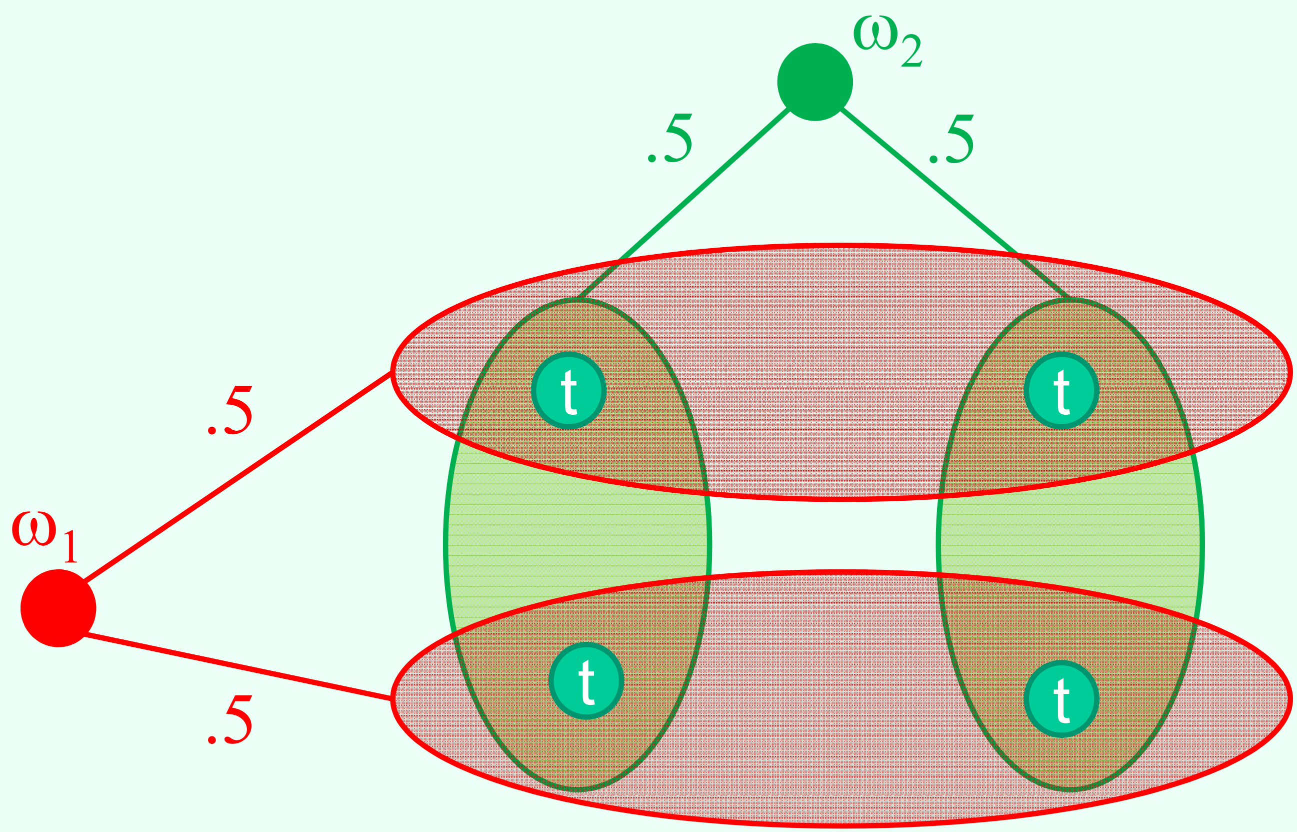
Distributional constraints

$$\forall t : U_a^c(t)\sum_{\omega}\sum_{s:t\in s} c_{\omega,s} + U_a^u(t)\left(1-\sum_{\omega}\sum_{s:t\in s} c_{\omega,s}\right)$$

$$\le U_a^c(t^*)\sum_{\omega}\sum_{s:t^*\in s} c_{\omega,s} + U_a^u(t^*)\left(1-\sum_{\omega}\sum_{s:t^*\in s} c_{\omega,s}\right)$$

Attacker optimality

Counter-example to the compact LP



- LP suggests that we can cover every target with probability 1…
- … but in fact we can cover at most 3 targets at a time

Birkhoff-von Neumann theorem

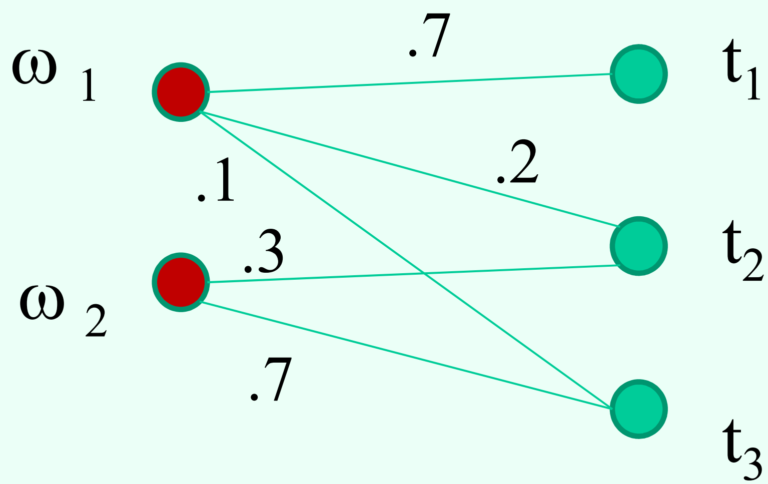
- Every *doubly stochastic n x n* matrix can be represented as a convex combination of *n x n* permutation matrices

.1	.4	.5
.3	.5	.2
.6	.1	.3

$$= .1 \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} + .1 \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} + .5 \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} + .3 \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

- Decomposition can be found in polynomial time $O(n^{4.5})$, and the size is $O(n^2)$ [Dulmage and Halperin, 1955]
- Can be extended to *rectangular* doubly *substochastic* matrices

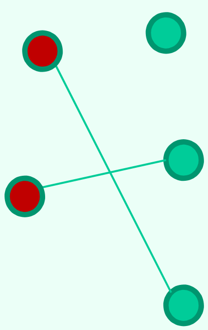
Schedules of size 1 using BvN



	t_1	t_2	t_3
ω_1	.7	.2	.1
ω_2	0	.3	.7

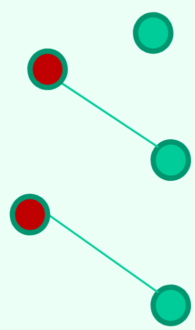
.1

0	0	1
0	1	0



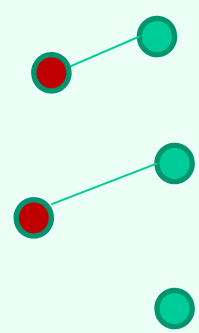
.2

0	1	0
0	0	1



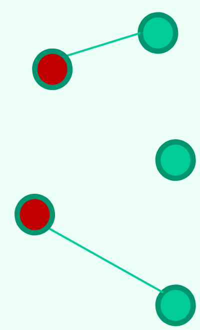
.2

1	0	0
0	1	0



.5

1	0	0
0	0	1



Algorithms & complexity

[Korzhyk, C., Parr AAAI'10]

		Homogeneous Resources	Heterogeneous resources
Schedules	Size 1	P	P (BvN theorem)
	Size ≤2, bipartite	P (BvN theorem)	NP-hard (SAT)
	Size ≤2	P (constraint generation)	NP-hard
	Size ≥3	NP-hard (3-COVER)	NP-hard

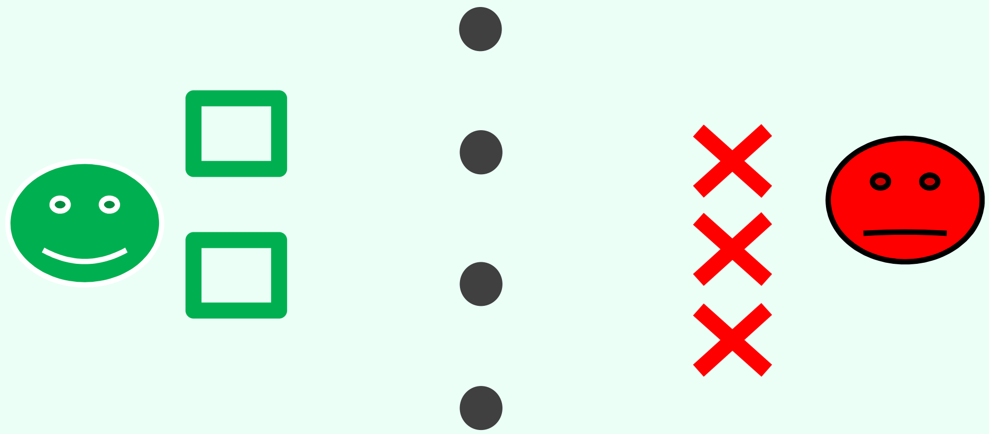
Also: security games on graphs

[Letchford, C. AAAI'13]

Security games with multiple attacks

[Korzhyk, Yin, Kiekintveld, C., Tambe JAIR’11]

- The attacker can choose multiple targets to attack



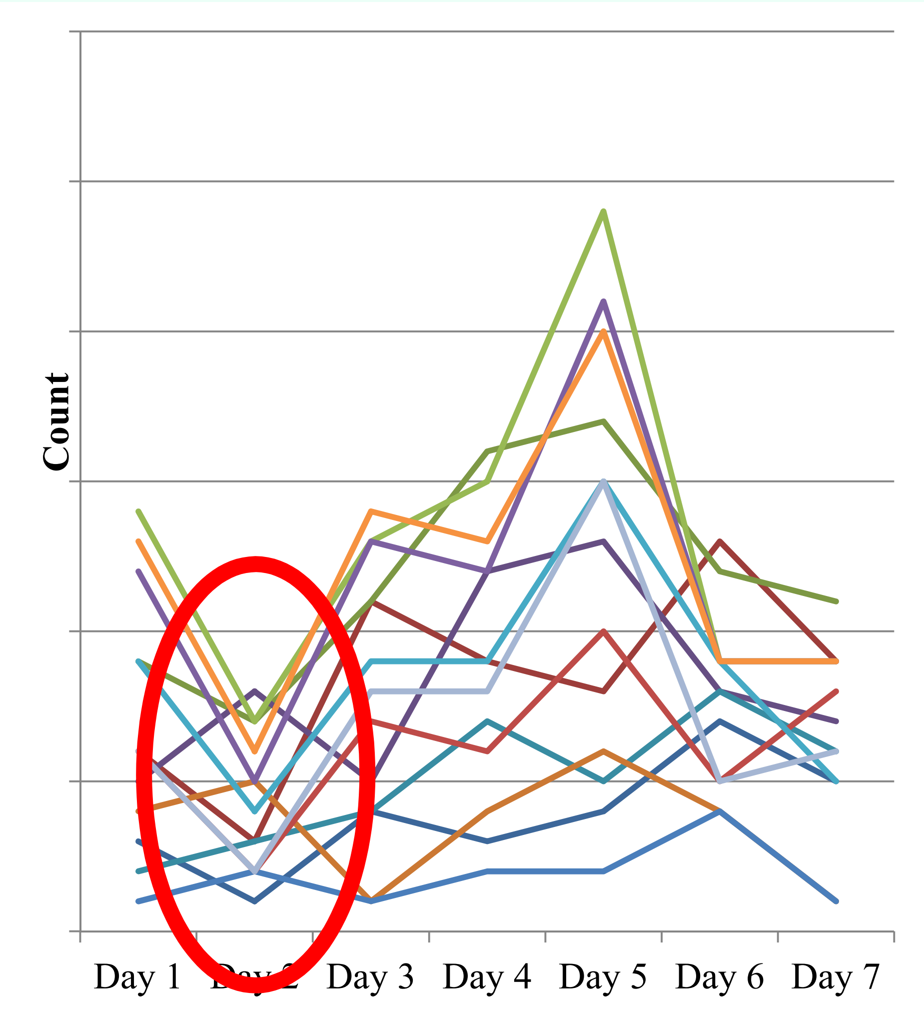
- The utilities are added over all attacked targets
- Stackelberg NP-hard; Nash polytime-solvable and interchangeable [Korzhyk, C., Parr IJCAI‘11]
  - Algorithm generalizes ORIGAMI algorithm for single attack [Kiekintveld, Jain, Tsai, Pita, Ordóñez, Tambe AAMAS’09]

Actual Security Schedules: Before vs. After

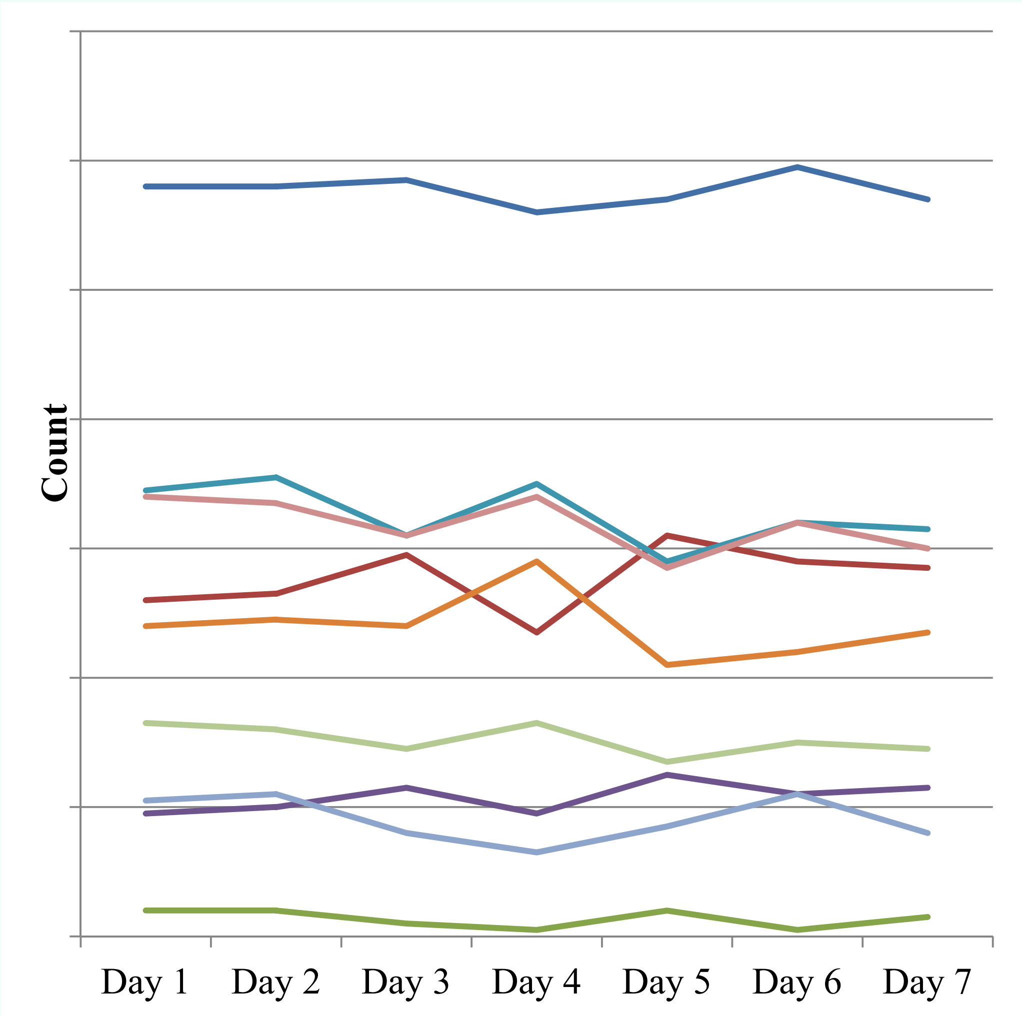
Boston, Coast Guard – “PROTECT” algorithm

slide courtesy of Milind Tambe

Before PROTECT



After PROTECT

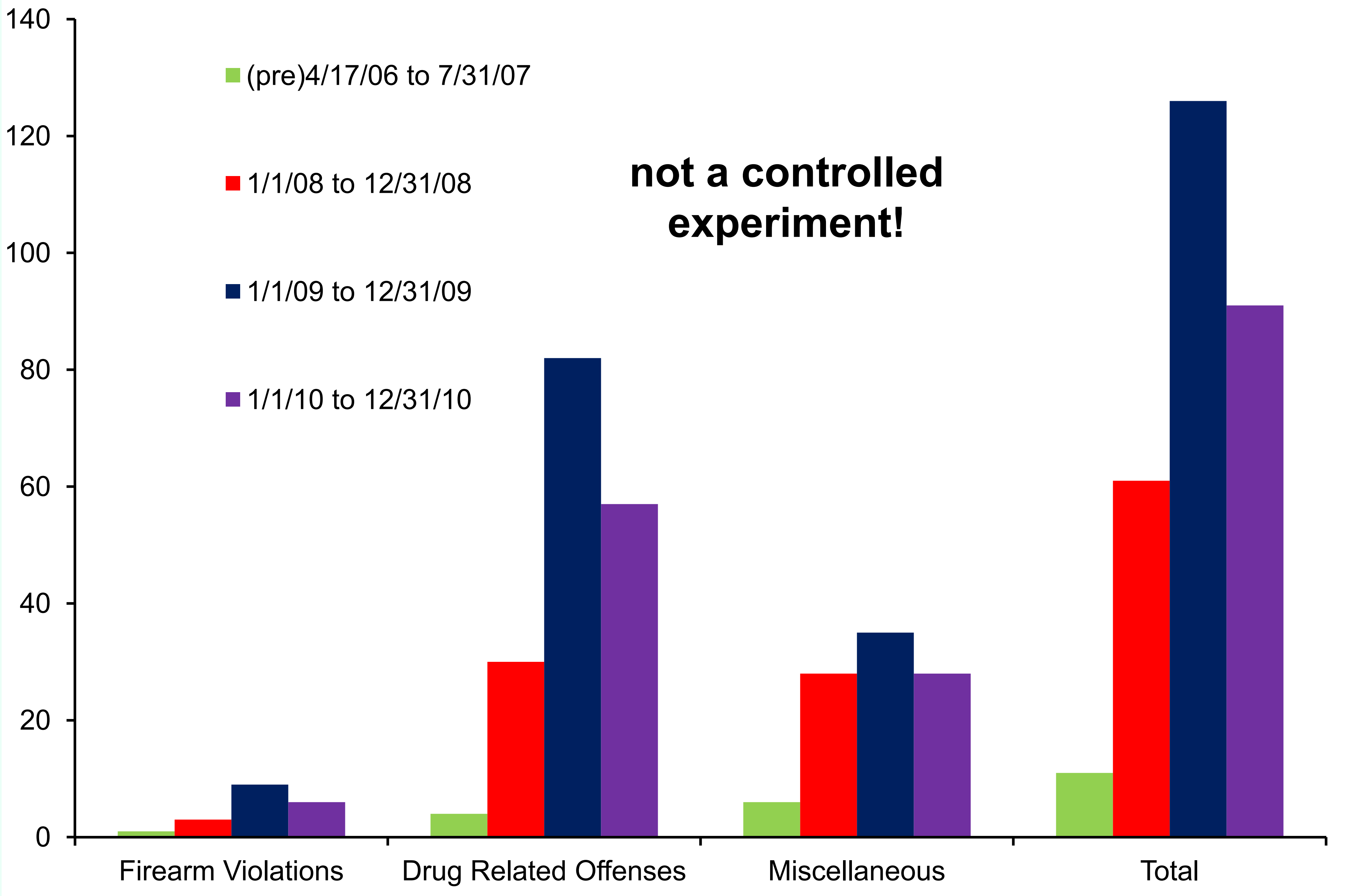


Industry port partners comment:

“The Coast Guard seems to be everywhere, all the time."

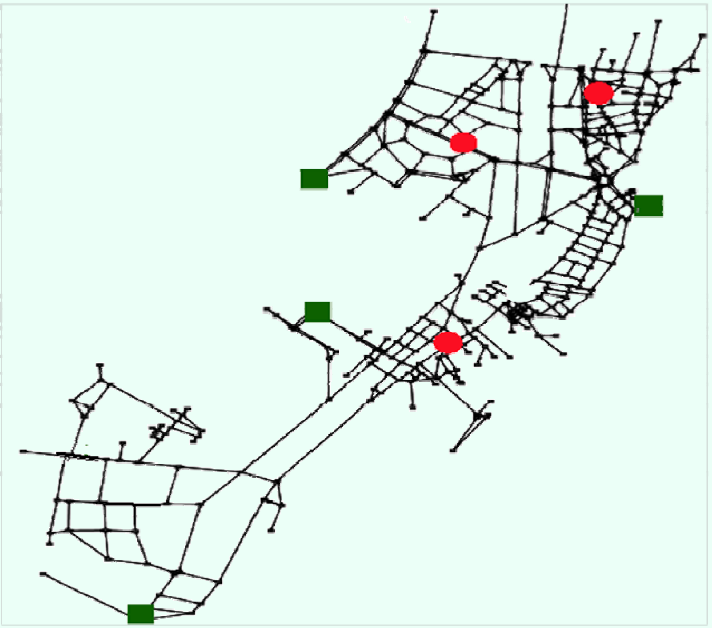
Data from LAX checkpoints before and after "ARMOR" algorithm

slide courtesy of Milind Tambe



Placing checkpoints in a city

[Tsai, Yin, Kwak, Kempe, Kiekintveld, Tambe AAAI'10; Jain, Korzhyk, Vaněk, C., Pěchouček, Tambe AAMAS'11; Jain, C., Tambe AAMAS'13]



In summary: CS pushing at some of the boundaries of game theory