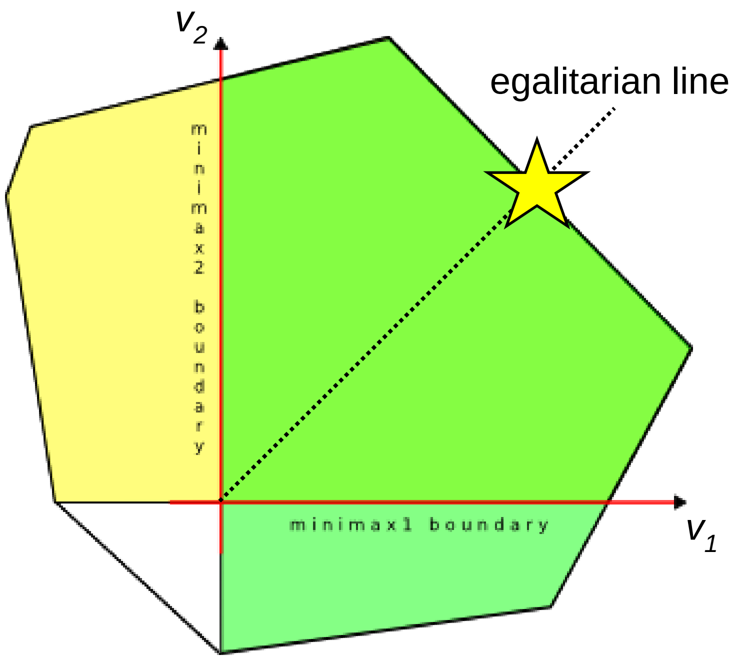
A Polynomial-time Nash Equilibrium Algorithm for Repeated Stochastic Games

Enrique Munoz de Cote
Michael L. Littman

Main Result

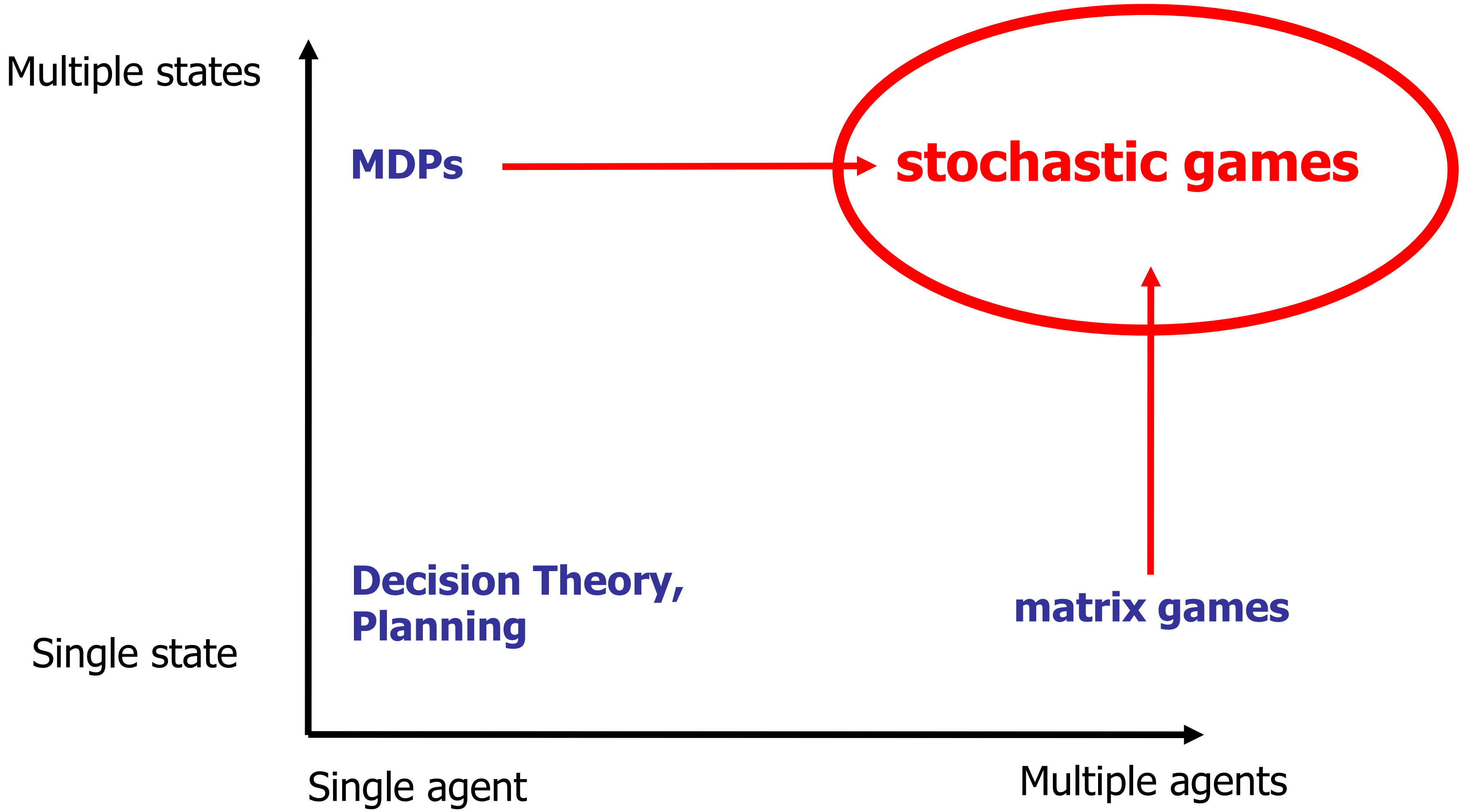
Concretely, we address the following computational problem:

- Given a repeated stochastic game, return a strategy profile that is a Nash equilibrium (specifically one whose payoffs match the egalitarian point) of the average payoff repeated stochastic game in polynomial time.



Convex hull of the average payoffs

Framework


Multiple states
MDPs
stochastic games
Decision Theory,
Planning
matrix games
Single state
Single agent
Multiple agents

Stochastic Games (SG)

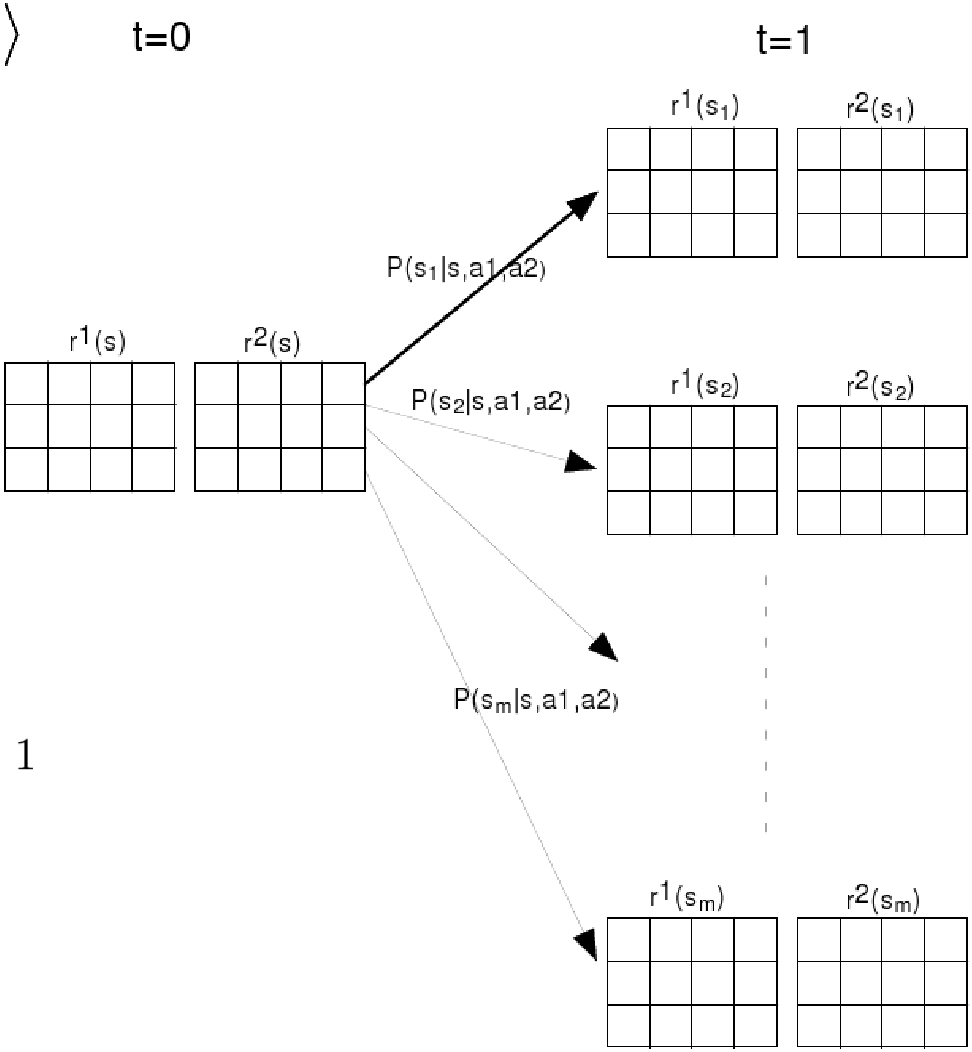
$$\langle \mathcal{N}, \mathcal{S}, \{\mathcal{A}_i(s)\}, \mathcal{T}, \{\mathcal{R}_i(s,a)\} \rangle$$

- Superset of MDPs & NFGs
- *S* is the set of states
- *T* is the transition function

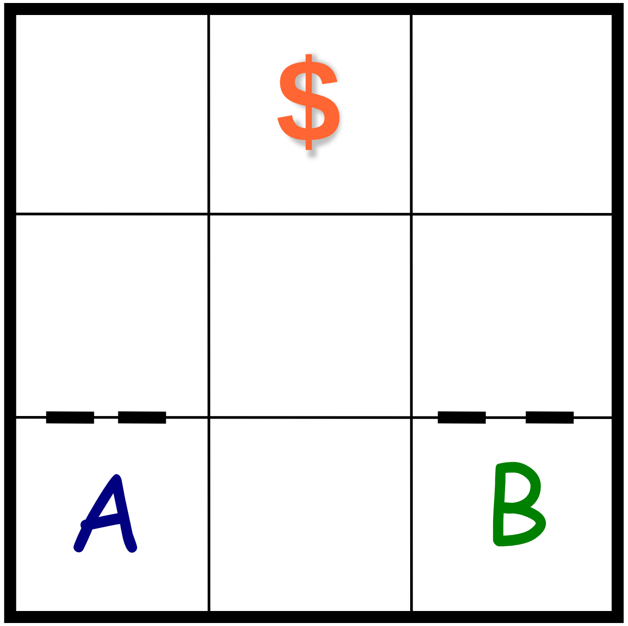
Such that:

$$\mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow [0,1]$$

$$\forall s \in \mathcal{S}, a \in \mathcal{A} \qquad \sum_{s' \in \mathcal{S}} \mathcal{T}(s,a,s') = 1$$



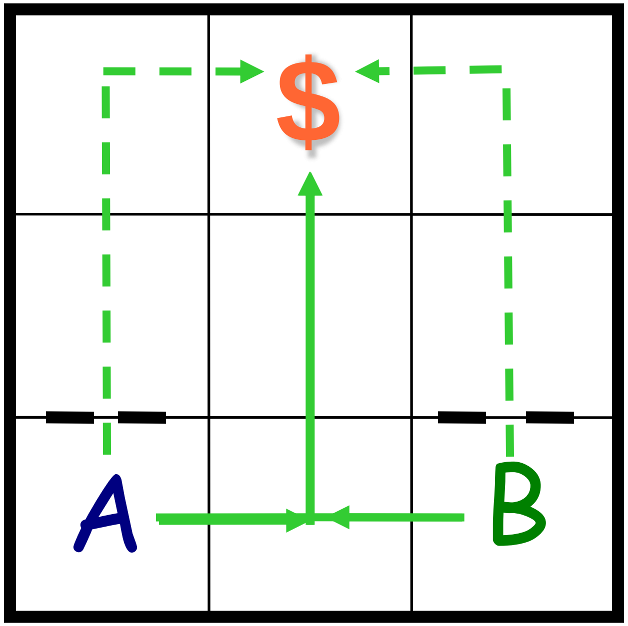
A Computational Example: SG version of chicken



SG of chicken [Hu & Wellman, 03]

- actions: U, D, R, L, X
- coin flip on collision
- Semiwalls (50%)
- collision = -5;
- step cost = -1;
- goal = +100;
- discount factor = 0.95;
- both can get goal.

Equilibrium values



Average total reward on equilibrium:

- Nash
  - (88.3,43.7) very imbalanced, inefficient
  - (43.7,88.3) very imbalanced, inefficient
  - (53.6,53.6) ½ mix, still inefficient
- Correlated
  - ([43.7,88.3],[43.7,88.3]);
- Minimax
  - (43.7,43.7);
- Friend
  - (38.7,38.7)

Nash: computationally difficult to find in general

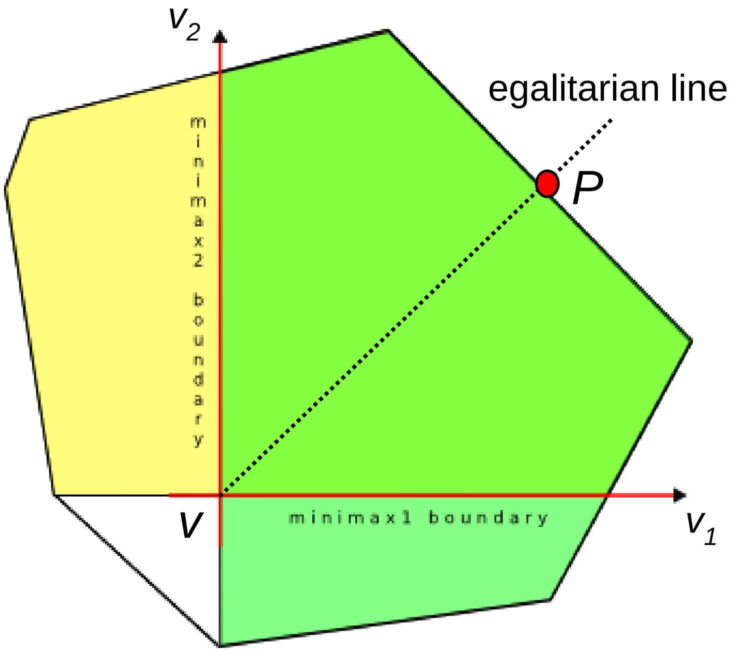
Egalitarian equilibrium point

Folk theorems conceptual drawback: infinitely many feasible and enforceable strategies

- Egalitarian line. line where payoffs are equally high above *v*

Egalitarian point. Maximizes the *minimum* advantage of the players' rewards

$$P = \arg\max_{x \in X} \min_{v}(x)$$



Convex hull of the average payoffs

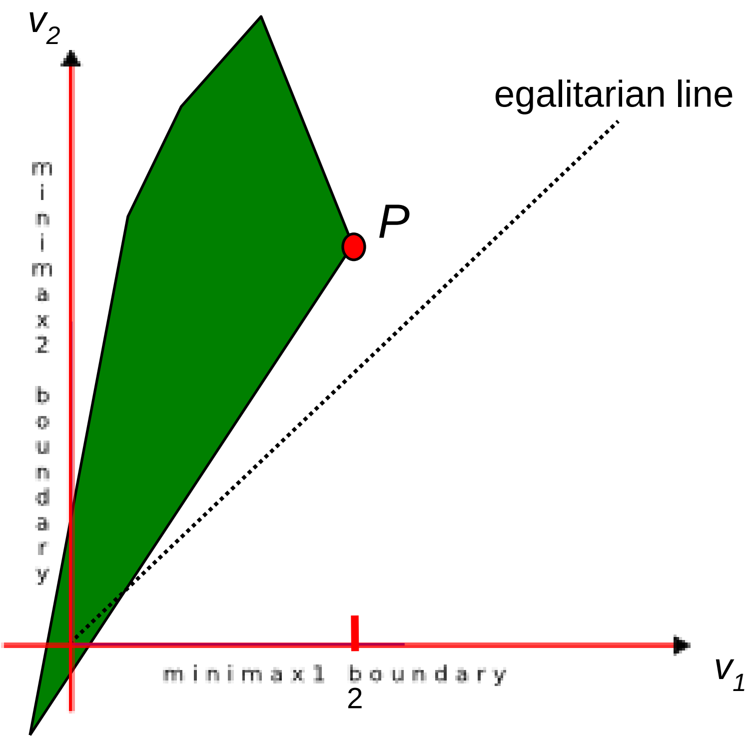
How? (the short story version)

- Compute `attack` and `defense` strategies.
  - Solve two linear programming problems.
- The algorithm searches for a point:

$$P = \arg\max_{x \in X} \min_v(x)$$

where

$$\begin{aligned} x &= (x_1, x_2) \\ \min_v(x) &= \min(x_1 - v_1, x_2 - v_2) \end{aligned}$$



Convex hull of a hypothetical SG

- P is the point with the highest egalitarian value.

Game representation

- Folk theorems can be interpreted computationally
  - Matrix form [Littman & Stone, 2005]
  - Stochastic game form [Munoz de Cote & Littman, 2008]
- Define a weighted combination value:

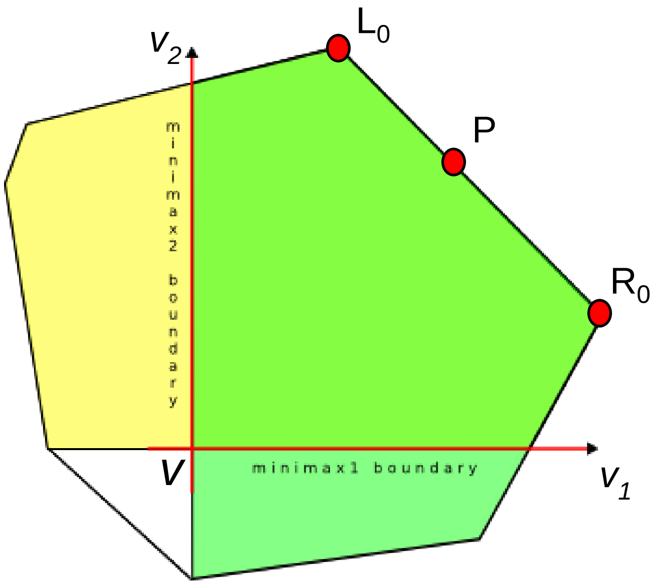
$$\sigma_w(p) = wp_1 + (1 - w)p_2$$

- A strategy profile (π) that achieves $\sigma_w(p^\pi)$ can be found by modeling an MDP

Markov Decision Processes

- We use MDPs to model 2 players as a *meta*-player
  - Return: joint strategy profile that maximizes a weighted combination of the players' payoffs

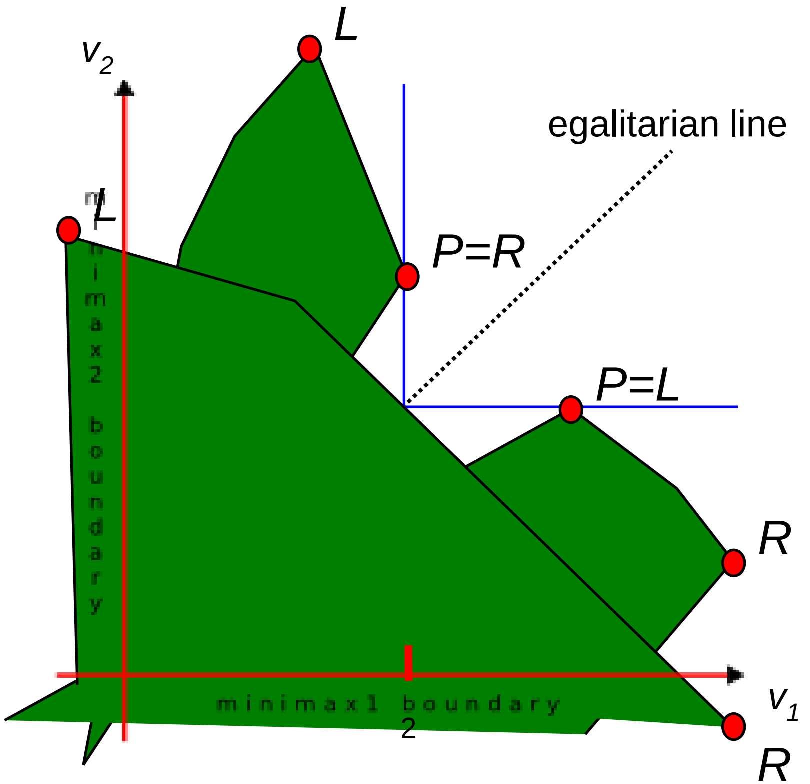
- Friend solutions:
  - $(R_0, \pi_1) = \text{MDP}(1)$,
  - $(L_0, \pi_2) = \text{MDP}(0)$,
- A weighted solution:
  - $(P, \pi) = \text{MDP}(w)$



The algorithm

`FolkEgal(U1,U2, ϵ)`

- Compute
  - $\texttt{attack}_1, \texttt{attack}_2$,
  - $\texttt{defense}_1, \texttt{defense}_2$ and
  - $\texttt{R=friend}_1, \texttt{L=friend}_2$
- Find egalitarian point and its strategy profilile
  - If `R` is left of egalitarian line: `P=R`
  - elseIf `L` is right of egalitarian line: `P = L`
  - Else `egalSearch(R,L,T)`



Convex hull of a hypothetical SG

The key subroutine

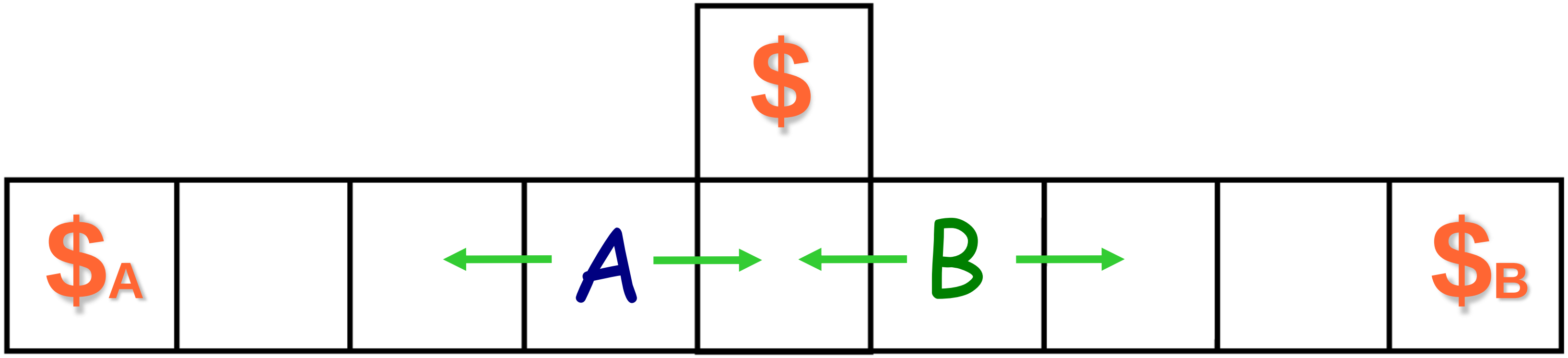
`EgalSearch(L,R,T)`

- Finds intersection between X and egalitarian line
- Close to a binary search
- Input:
  - Point L (to the left of egalitarian line)
  - Point R (to the right of egalitarian line)
  - A bound T on the number of iterations
- Return:
  - The egalitarian point P (with accuracy ϵ)
- Each iteration solves an MDP(w) by finding a solution to:

$$\sigma_w(L) = \sigma_w(R)$$

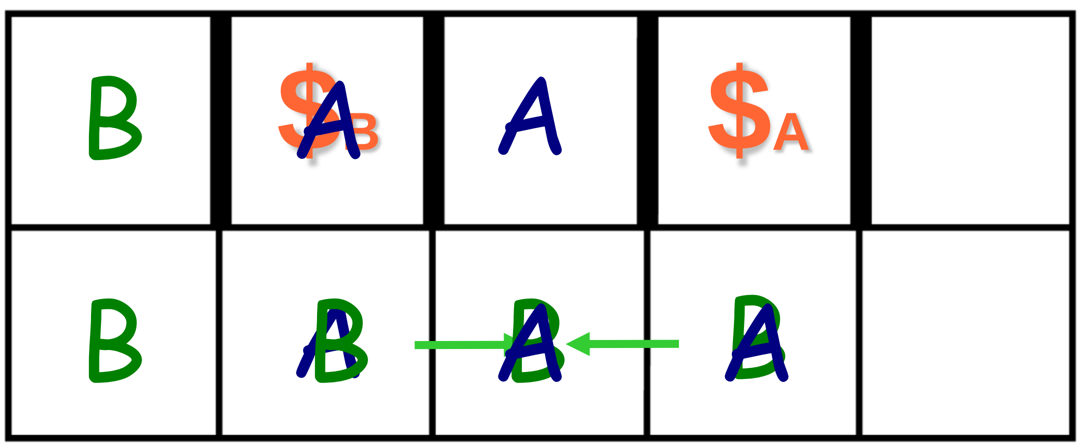
SG version of the PD game

Algorithm	Agent A	Agent B	
security-VI	46.5	46.5	mutual defection
friend-VI	46	46	mutual defection
CE-VI	46.5	46.5	mutual defection
folkEgal	88.8	88.8	mutual cooperation with threat of defection



Compromise game

Algorithm	Agent A	Agent B	
security-VI	0	0	attacker blocking goal
friend-VI	-20	-20	mutual defection
CE-VI	68.2	70.1	suboptimal waiting strategy
folkEgal	78.7	78.7	mutual cooperation (*w=0.5*) with treat of defection



Asymmetric game

Algorithm	Agent A	Agent B	
security-VI	0	0	attacker blocking goal
friend-VI	-200	-200	mutual defection
CE-VI	32.1	32.1	suboptimal mutual cooperation
folkEgal	37.2	37.2	mutual cooperation with threat of defection

Thanks for your attention!