Solving Stochastic Games

A tractable algorithm to find the optimal (game-theoretic with ε-error) solution to multi-agent reinforcement learning

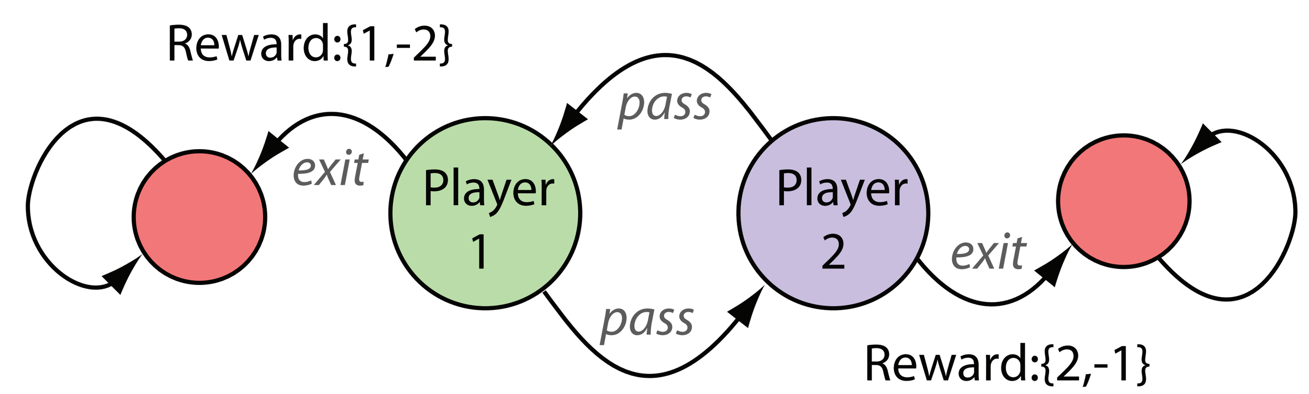
Liam Mac Dermed
Charles Isbell

Georgia Institute of Technology

Key Ideas:

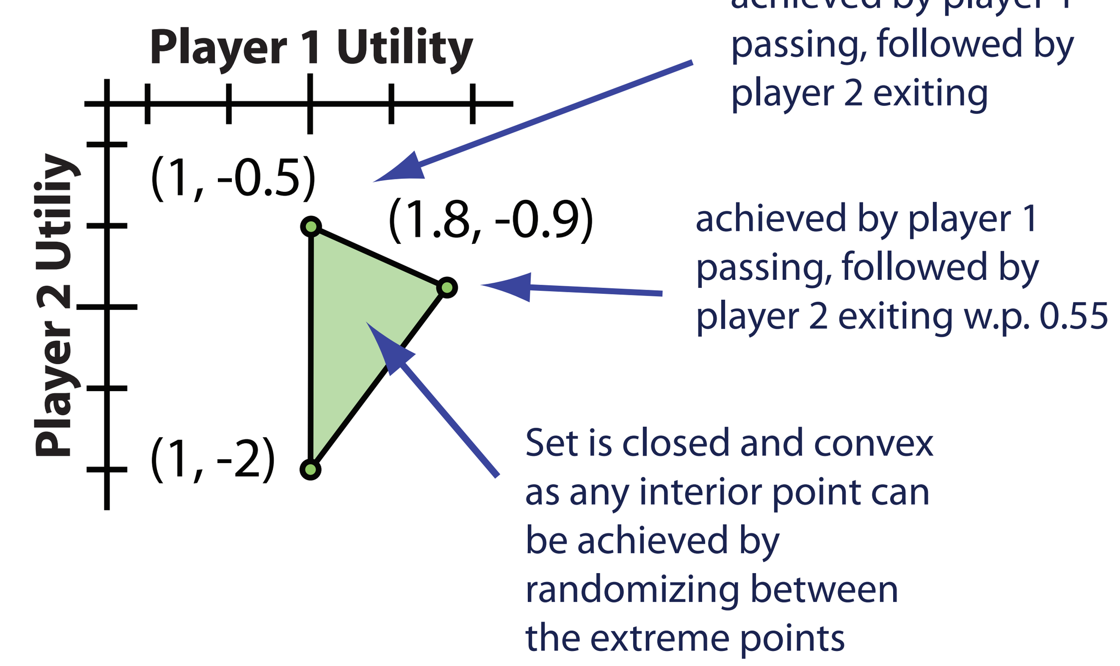
- Generalize the Bellman equation by using feasible-sets instead of values (Murray & Gordon 2007). A feasible-set captures the set of possible achievable utilities, instead of a single best utility (the value of a state).
- Approximate feasible-sets using a fixed collection of hyperplanes. Approximation scheme is chosen carefully to maintain convergence guarantees while bounding error.
- Use multi-objective linear programming to calculate backups.

An Example Game (The Breakup Game):



Circles represent states, outgoing arrows represent deterministic actions. Unspecified rewards are zero. Previous algorithms could not solve this game.

The final feasible-set for player 1's state



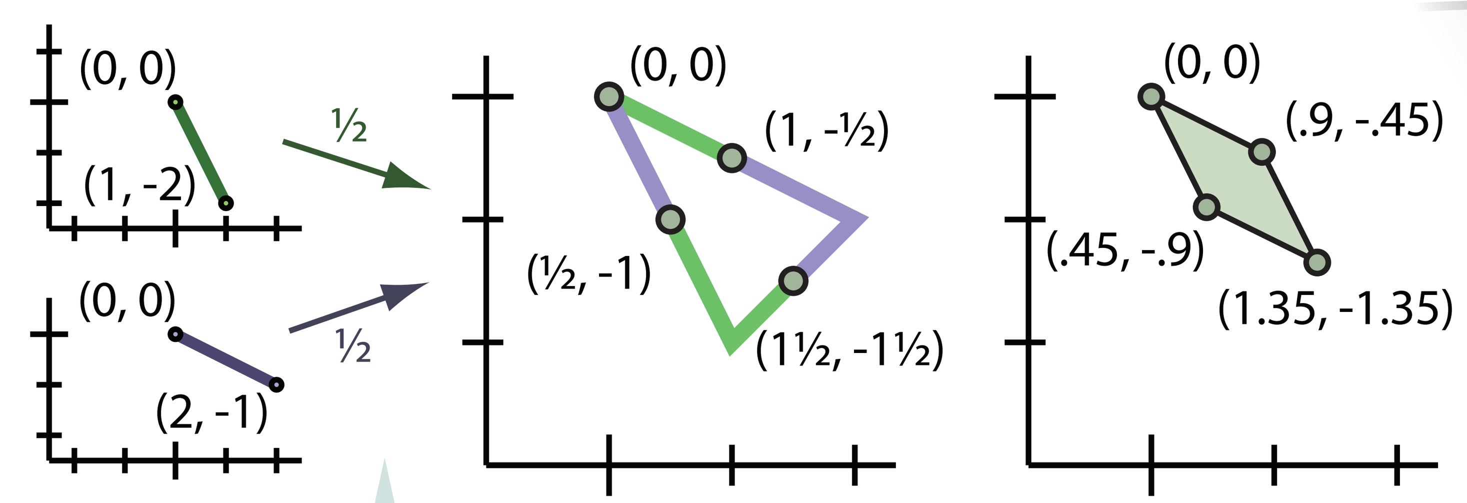
The tractable backup of feasible sets: (an iteration consists of a single backup for each state)

replaces 'max' in single agent RL

The state shown being calculated is an initial rock-paper-scissors game played to decide who goes first in the breakup game

$$\mathbf{Q}^*(s,\vec{a}) = \mathbf{R}(s,\vec{a}) + \gamma \sum_{s'} P(s'|s,\vec{a})\mathbf{V}^*(s')$$

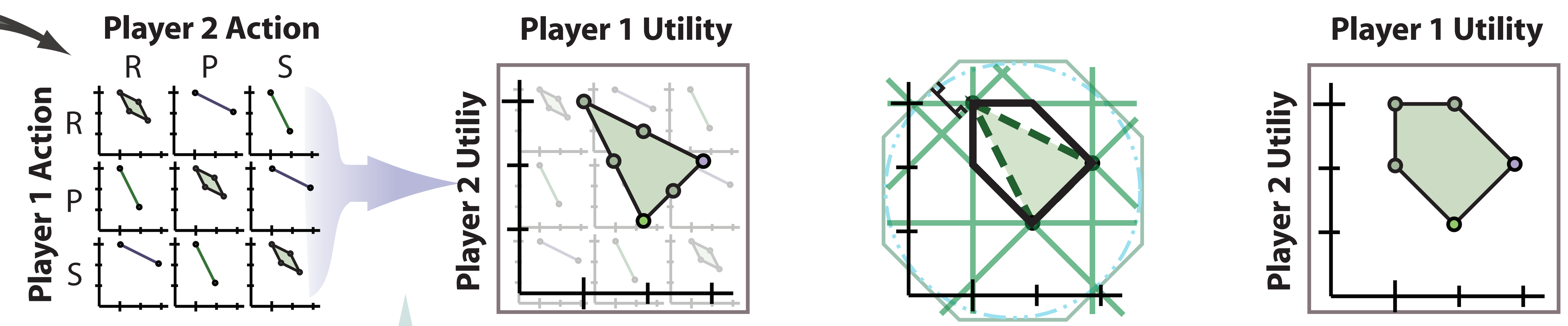
$$\mathbf{V}^*(s) = \underset{\vec{a}}{\text{equilibrium}}(\mathbf{Q}^*(s,\vec{a}))$$



Feasible sets of successor states V*(s′)

Expected values of all policies

Feasible set of expected values Q*(s,a)



Feasible sets of all joint actions {Q*(s,a)}

Exact Feasible set of initial state equilibrium({Q*(s,a)})

Approximation Using Set of Hyperplanes

Final Feasible Set of initial state V*(s)

Approximation guarantees an upper bound on the computational complexity of each backup

Computing the set of possible expectations:

Simultaneously maximize foreach player i from 1 to n: $\sum_{s'} \sum_{\vec{v} \in V(s')} v_i x_{s'\vec{v}}$

Subject to: for every state s' $\sum_{\vec{v} \in V(s')} x_{s'\vec{v}} = P(s'|s,\vec{a})$

Computing the set based correlated equilibrium:

Simultaneously maximize foreach player i from 1 to n: $\sum_{\vec{a}\vec{u}} u_i x_{\vec{a}\vec{u}}$

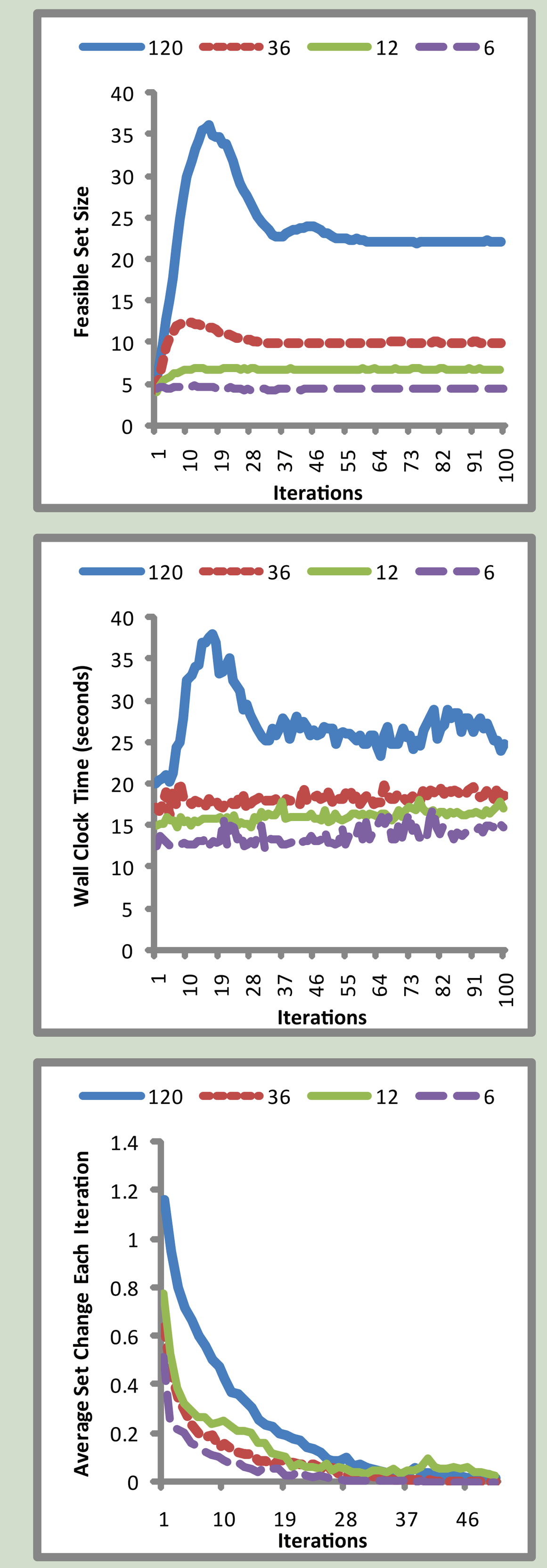
Subject to: probability constraints $\sum x_{\vec{a}\vec{u}} = 1$ and $x_{\vec{a}\vec{u}} \geq 0$

and foreach player i, actions $a_1, a_2 \in A_i$, $(a_2 \neq a_1)$

$\sum_{\vec{a}\vec{u}|a_i=a_1} u_i x_{\vec{a}\vec{u}} \geq \sum_{\vec{a}\vec{u}|a_i=a_2} F_{th}(s,\vec{a}) x_{\vec{a}\vec{u}}$

Results:

We provide the first approximation algorithm which solves stochastic games with cheap-talk to within ε absolute error of the optimal game-theoretic solution.



Statistics from a random game (100 states, 2 players, 2 actions each) run with different levels of approximation. The numbers shown (120, 36, 12, and 6) represent the number of predetermined hyperplanes used to approximate each Pareto frontier.