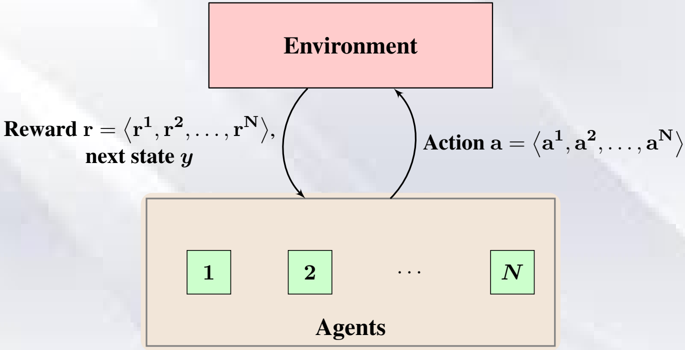
Two-Timescale Algorithms for Learning Nash Equilibria in General-Sum Stochastic Games

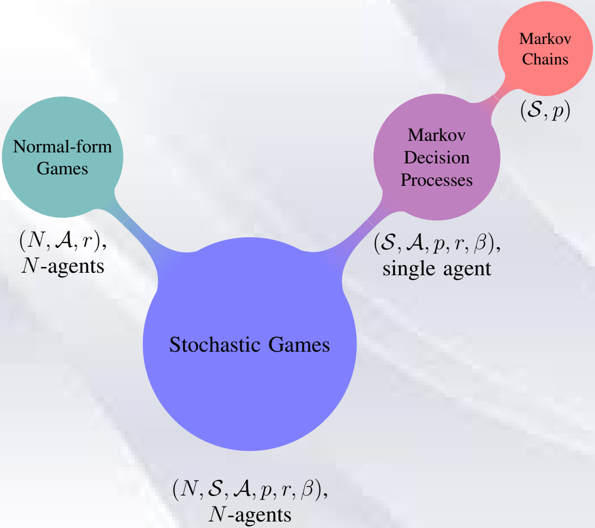
H.L. Prasad[†], Prashanth L.A.[♯] and Shalabh Bhatnagar[♯]

[†]Streamoid Technologies, Inc [♯]Indian Institute of Science

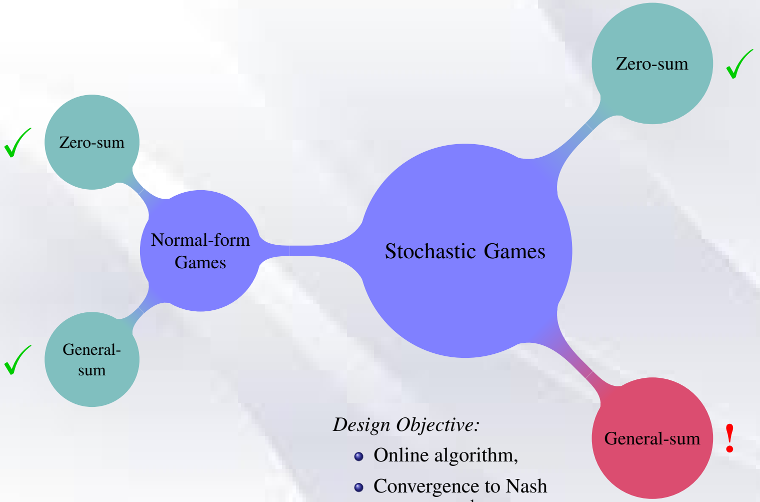
Multi-agent RL setting



Problem area



Problem area (revisited)



Design Objective:

- Online algorithm,
- Convergence to Nash equilibrium[1]

[1] If NE is a useful objective for learning in games, then we have a strong contribution!

A General Optimization Problem

Value function

$$v^\pi(s) \ = E\left[\sum_t \beta^t \sum_{a \in \mathcal{A}(x)} \ r(s_t, a) \ \ \pi(s_t, a) \ \mid s_0 = s\right]$$

Value function Reward Policy

A stationary Markov strategy $\pi^* = \langle \pi^{1*}, \pi^{2*}, \ldots, \pi^{N*} \rangle$ is said to be Nash if

$$v^i_{\pi^*}(s) \geq v^i_{\langle \pi^i, \pi^{-i*} \rangle}(s), \forall \pi^i, \forall i, \forall s \in \mathcal{S}$$

Value function

$$v^{\pi}(s) = E\left[\sum_{t} \beta^{t} \sum_{a \in \mathcal{A}(x)} r(s_t, a) \; \pi(s_t, a) \; \mid s_0 = s\right]$$

Value function

Reward

Policy

A stationary Markov strategy $\pi^* = \langle \pi^{1*}, \pi^{2*}, \ldots, \pi^{N*} \rangle$ is said to be Nash if

$$v^{i}_{\pi^*}(s) \geq v^{i}_{\langle \pi^{i}, \pi^{-i*} \rangle}(s), \forall \pi^{i}, \forall i, \forall s \in \mathcal{S}$$

Value function

$$v^{\pi}(s) \;= E\left[\sum_t \beta^t \sum_{a \in \mathcal{A}(x)} r(s_t, a)\;\; \pi(s_t, a)\;\; \middle|\; s_0 = s\right]$$

Value function

Reward

Policy

A stationary Markov strategy $\pi^* = \langle \pi^{1*}, \pi^{2*}, \ldots, \pi^{N*} \rangle$ is said to be Nash if

$$v^i_{\pi^*}(s) \geq v^i_{\langle \pi^i, \pi^{-i*} \rangle}(s), \forall \pi^i, \forall i, \forall s \in \mathcal{S}$$

Value function

$$v^{\pi}(s) = E\left[\sum_{t} \beta^{t} \sum_{a \in \mathcal{A}(x)} r(s_t, a)\ \pi(s_t, a) \mid s_0 = s\right]$$

Value function

Reward

Policy

A stationary Markov strategy $\pi^* = \langle \pi^{1*}, \pi^{2*}, \ldots, \pi^{N*} \rangle$ is said to be Nash if

$$v^{i}_{\pi^*}(s) \geq v^{i}_{\langle \pi^{i}, \pi^{-i*} \rangle}(s), \forall \pi^{i}, \forall i, \forall s \in \mathcal{S}$$

Value function

$$v^\pi(s) \;= E\Big[\sum_t \beta^t \sum_{a \in \mathcal{A}(x)} r(s_t,a)\;\; \pi(s_t,a)\;\; \mid s_0 = s\Big]$$

Value function Reward Policy

A stationary Markov strategy $\pi^* = \langle \pi^{1*}, \pi^{2*}, \ldots, \pi^{N*} \rangle$ is said to be Nash if

$$\boxed{v^i_{\pi^*}(s) \geq v^i_{\langle \pi^i, \pi^{-i*} \rangle}(s), \forall \pi^i, \forall i, \forall s \in \mathcal{S}}$$

Dynamic Programming Idea

$$v^i_{\pi^*}(x) = \max_{\pi^i(x) \in \Delta(\mathcal{A}^i(x))} \left\{ E_{\pi^i(x)} \; Q^i_{\pi^{-i*}}(x, a^i) \right\},$$

Optimal (Nash) Value

Marginal Value after fixing $a^i \sim \pi^i$

where Q-value is given by

$$Q^i_{\pi^{-i}}(x, a^i) = E_{\pi^{-i}(x)} \left[r^i(x, a) + \beta \sum_{y \in U(x)} p(y|x, a) v^i(y) \right]$$

Dynamic Programming Idea

$$v^i_{\pi^*}(x) = \max_{\pi^i(x) \in \Delta(\mathcal{A}^i(x))} \left\{ E_{\pi^i(x)} \; Q^i_{\pi^{-i*}}(x, a^i) \; \right\},$$

Optimal (Nash) Value

Marginal Value after fixing $a^i \sim \pi^i$

where Q-value is given by

$$Q^i_{\pi^{-i}}(x, a^i) = E_{\pi^{-i}(x)} \left[r^i(x,a) + \beta \sum_{y \in U(x)} p(y|x,a) v^i(y) \right]$$

Dynamic Programming Idea

$$v^i_{\pi^*}(x) = \max_{\pi^i(x)\in\Delta(\mathcal{A}^i(x))} \left\{ E_{\pi^i(x)}\ Q^i_{\pi^{-i*}}(x, a^i) \right\},$$

Optimal (Nash) Value

Marginal Value after fixing $a^i \sim \pi^i$

where Q-value is given by

$$Q^i_{\pi^{-i}}(x, a^i) = E_{\pi^{-i}(x)} \left[r^i(x, a) + \beta \sum_{y \in U(x)} p(y|x, a) v^i(y) \right]$$

Dynamic Programming Idea

$$v^i_{\pi^*}(x) = \max_{\pi^i(x)\in\Delta(\mathcal{A}^i(x))} \left\{ E_{\pi^i(x)}\ Q^i_{\pi^{-i*}}(x, a^i) \right\},$$

Optimal (Nash) Value

Marginal Value after fixing $a^i \sim \pi^i$

where Q-value is given by

$$Q^i_{\pi^{-i}}(x, a^i) = E_{\pi^{-i}(x)} \left[r^i(x,a) + \beta \sum_{y\in U(x)} p(y|x,a) v^i(y) \right]$$

Optimization problem - informal terms

Need to solve:

$$v^i_{\pi^*}(x) = \max_{\pi^i(x)\in\Delta(\mathcal{A}^i(x))} \left\{ E_{\pi^i(x)} Q^i_{\pi^{-i*}}(x, a^i) \right\} \tag{1}$$

Formulation:

Objective. minimize the Bellman error $v^i(x) - E_{\pi^i} Q^i_{\pi^{-i}}(x, a^i)$ in every state, for every agent

Constraint 1. ensure policy π is a distribution

Constraint 2. $Q^i_{\pi^{-i}}(x, a^i) \leq v^i_\pi(x) \longleftarrow$ a proxy for the max in (1)

Optimization problem in formal terms

$$\min_{v,\pi} f(v,\pi) = \sum_{i=1}^{N} \sum_{x \in \mathcal{S}} \left(v^i(x) - E_{\pi^i} Q^i_{\pi^{-i}}(x, a^i) \right)$$

subject to

$$\pi^i(x, a^i) \geq 0, \forall a^i \in \mathcal{A}^i(x), x \in \mathcal{S}, i = 1, 2, \ldots, N,$$

$$\sum_{i=1}^{N} \pi^i(x, a^i) = 1, \forall x \in \mathcal{S}, i = 1, 2, \ldots, N.$$

$$Q^i_{\pi^{-i}}(x, a^i) \leq v^i(x), \forall a^i \in \mathcal{A}^i(x), x \in \mathcal{S}, i = 1, 2, \ldots, N.$$

Solution approach

Usual approach: Apply KKT conditions to solve the general optimization problem

Caveat: Imposes a tricky linear independence requirement

Alternative: Use a simpler set of SG-SP conditions

A sufficient condition

SG-SP Point A point (v^*, π^*) is said to be an SG-SP point if it is feasible and for all $x \in \mathcal{X}$ and $i \in \{1, 2, \ldots, N\}$

$$\pi^{i*}(x, a^i) g^i_{x,a^i}(v^{i*}, \pi^{-i*}(x)) = 0, \qquad \forall a^i \in \mathcal{A}^i(x)$$

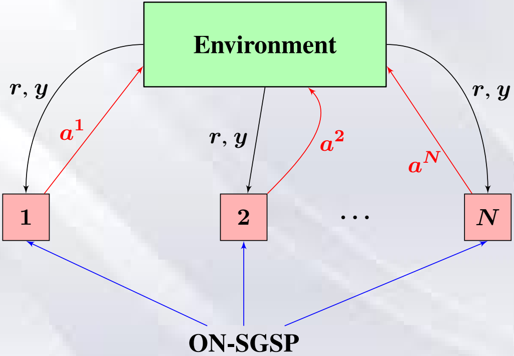
where $g^i_{x,a^i}(v^i, \pi^{-i}(x)) := Q^i_{\pi^{-i}}(x, a^i) - v^i(x)$.

Nash $\Leftrightarrow$ SG-SP:

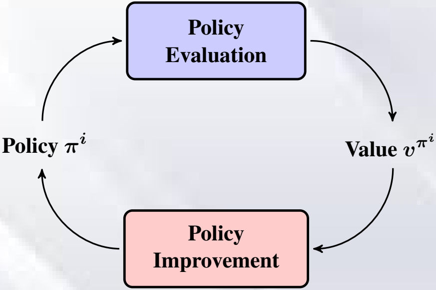
A strategy π^* is Nash if and only if (v^*, π^*) is an SG-SP point

An Online Algorithm: ON-SGSP

ON-SGSP's decentralized online learning model



ON-SGSP - operational flow

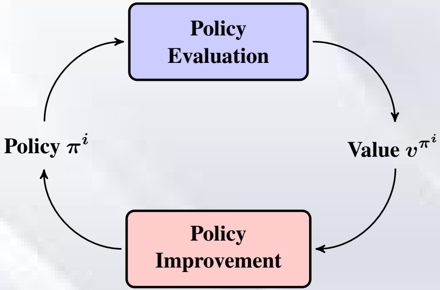


Policy evaluation: estimate the value function using temporal difference (TD) learning

Policy improvement: perform gradient descent for the policy using a descent direction

Descent direction ensures convergence to a global minimum of the optimization problem

ON-SGSP - operational flow



Policy evaluation: estimate the value function using temporal difference (TD) learning

Policy improvement: perform gradient descent for the policy using a descent direction

Descent direction ensures convergence to a global minimum of the optimization problem

More on the descent direction

Descend along

$$-\sqrt{\pi^i(x,a^i)}\left|g^i_{x,a^i}(v^i,\pi^{-i})\right| \times \overline{\mathrm{sgn}}\left(\frac{\partial f(v,\pi)}{\partial \pi^i}\right)$$

From Lagrange multiplier and slack variable theory

+

TD-learning for policy evaluation

Solution tracks an ODE with limit as an SG-SP point

[1] $\overline{\mathrm{sgn}}$ is a continuous version of sgn

Experiments

A single state non-generic 2-player game

Payoff Matrix

Player 2 → Player 1 ↓	a_1	a_2	a_3
a_1	$1, 0$	$0, 1$	$1, 0$
a_2	$0, 1$	$1, 0$	$1, 0$
a_3	$0, 1$	$0, 1$	$1, 1$

A single state non-generic 2-player game

Results from 100 simulation runs

	NashQ	FFQ (Friend Q)	ON-SGSP
Oscillate or converge to non-Nash strategy	**95%**	**40%**	0%
Converge to $(0.5, 0.5, 0)$	2%	0%	**99%**
Converge to $(0, 0, 1)$	3%	60%	1%

Stick-Together Game

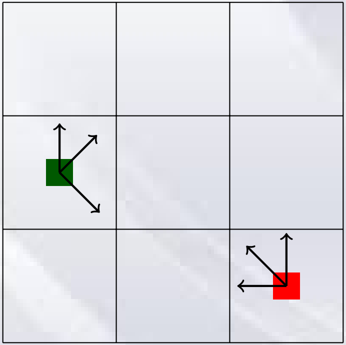
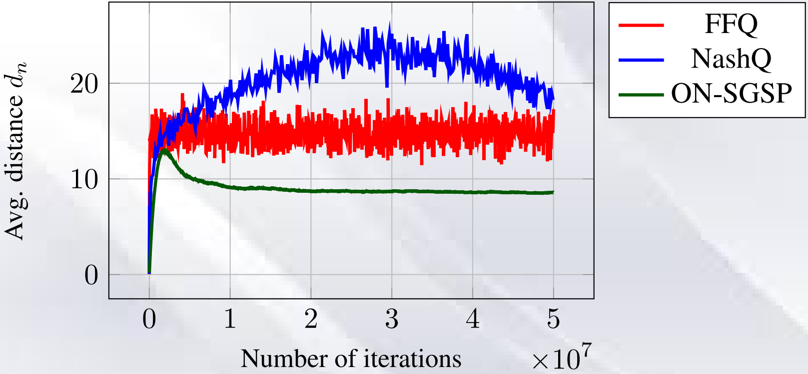


Figure: Stick Together Game for $M = 3$

For $M = 30$, STG has 810000 states!

Results for STG with $M = 30$



ON-SGSP takes agents to within a 4×4-grid, while NashQ/FFQ to a 8×8-grid.

Foe Q-learning/NashQ have higher per-iteration complexity than ON-SGSP

Thank You!