Adaptive Play *Q*-Learning with Initial Heuristic Approximation

Andriy Burkov and Brahim Chaib-draa

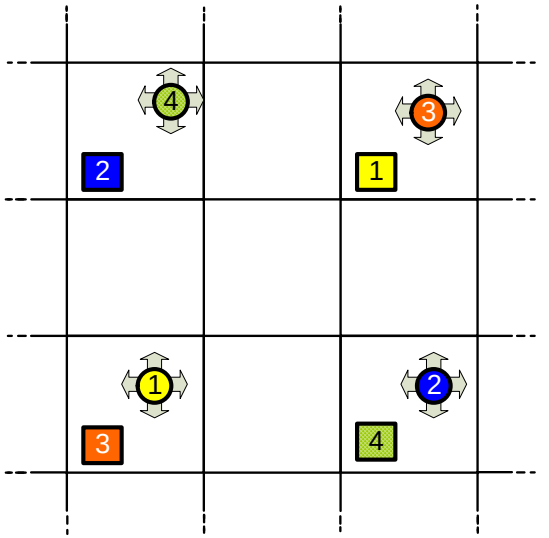
Laval University, Canada

DAMAS
Decision, Adaptation, Multi-AgentS
www.damas.ift.ulaval.ca

Motivations

- Use an optimal single-agent solution in order to speedup the learning in the multiagent stochastic game.
- Outline the limits of applicability of such an approach.
- Check experimentally the validity of the approach.

Example



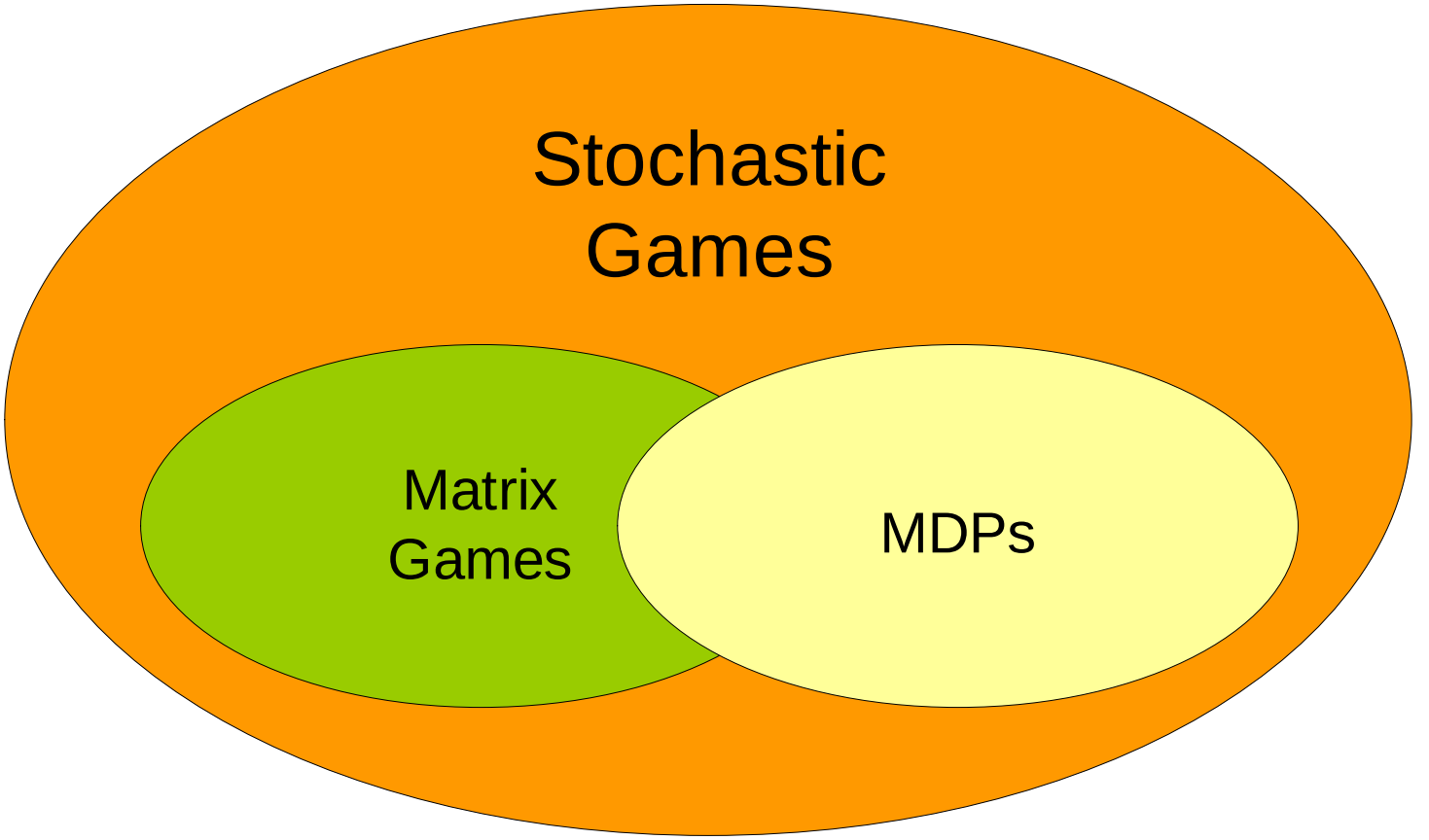
Four robot coordination game

- Four robots (circles 1, 2, 3 and 4) must learn to reach their respective goals
- The system's states are the joint positions of robots
- The actions are simultaneous and penalized
- The transitions are stochastic
- The collision additionally penalizes all affected robots and they keep their positions

Plan

- Motivations
- Stochastic Games
- Adaptive Play *Q*-Learning (APQ)
- *Q*-Value Initialization for APQ
- Results
- Conclusion

Stochastic Games



Stochastic Games

- MDP can model the problem of decision making in single-agent multi-state environments
- Matrix games can model the problem of decision making single-state multi-agent environments
- Stochastic games combine MDPs and Matrix games in order to model multi-state multi-agent decision problems

Stochastic games: model

- A stochastic game (SG) is a tuple $(n, \mathbf{S}, \mathcal{A}^{1\ldots n}, T, R^{1\ldots n})$
  - n is the number of agents (players)
  - $\mathcal{S}^j$ is the set of individual states of the agents, $j = 1 \ldots n$
  - $\mathbf{S}$ is the set of joint states $\mathbf{s} \in \mathbf{S} = \times_{j=1}^{n} \mathcal{S}^j$
  - $\mathcal{A}^j$ is the set of actions $a^j \in \mathcal{A}^j$ available for agent j
  - $\mathbf{A}$ is the joint action space, $\times_{j=1}^{n} \mathcal{A}^j$
  - T is the transition function: $\mathbf{S} \times \mathbf{A} \times \mathbf{S} \mapsto [0, 1]$
  - R^j is the reward fonction of agent j, $\mathbf{S} \times \mathbf{A} \mapsto \mathbb{R}$
  - $\mathbf{s}_0 \in \mathbf{S}$ est the initial joint state (where the agents are)

Adaptive Play for Matrix Games

- Adaptive Player (Young, 1993) j maintains a *history* $H_t^j = \{\mathbf{a}_{t-p}^{-j}, \ldots, \mathbf{a}_t^{-j}\}$ of last p opponents' joint actions.
- To select a strategy to play at time $t+1$, player j *samples* from H_t^j a subset, $\hat{H}_t^j = \{\mathbf{a}_{k_1}^{-j}, \ldots, \mathbf{a}_{k_l}^{-j}\}$, of length l
- Then it *estimate* the opponents' joint strategy $\hat{\Pi}^{-j}$ as follows: $\hat{\Pi}^{-j}(\mathbf{a}^{-j}) = \frac{C(\mathbf{a}^{-j}, \hat{H}_t^j)}{l}$
  - where $C(\mathbf{a}^{-j}, \hat{H}_t^j)$ is the number of times that the joint action $\mathbf{a}^{-j}$ was played by other players according to the set $\hat{H}_t^j$
- Given $\hat{\Pi}^{-j}$, player j *plays its best response* to this distribution with ϵ-exploration.

Adaptive Play Q-learning for Stochastic Games

- APQ (Gies & Chaib-draa, 2006) extends Adaptive Play to multi-state stochastic games.

function APQ($\mathbf{s}_0, t_{max}, p, l$)
returns: a policy.
Current state $\mathbf{s}_t \leftarrow \mathbf{s}_0$, time $t \leftarrow 1$.
while $t \leq t_{max}$ **do**
Select action a^j to play $a^j = \text{argmax}_{a^j} \sum_{\mathbf{a}^{-j}} \Pi^{-j}_{\mathbf{a}^{-j}}(\mathbf{s}_t)\hat{Q}(\mathbf{s}_t, \langle a^j, \mathbf{a}^{-j}\rangle)$
Play action a^j with some exploration
Observe new state $\mathbf{s}_{t+1}$, the joint action played $\mathbf{a}_t$, and the reward obtained R^j_t
Update $\hat{Q}^j(\mathbf{s}_t, \mathbf{a}_t) \leftarrow (1-\alpha)\hat{Q}^j(\mathbf{s}_t, \mathbf{a}_t) + \alpha\left[R^j_t + \gamma Value^j(\mathbf{s}_{t+1})\right]$ where
$Value^j(\mathbf{s}) = \max_{a^j} \sum_{\mathbf{a}^{-j}} \Pi^{-j}_{\mathbf{a}^{-j}}(\mathbf{s})\hat{Q}(\mathbf{s}, \mathbf{a})$ and $\mathbf{a} = \langle a^j, \mathbf{a}^{-j}\rangle$
Update history in $\mathbf{s}_t$: $H^j_t(\mathbf{s}_t) = H^j_t(\mathbf{s}_t) \cup \{\mathbf{a}^{-j}_t\} \setminus \{\mathbf{a}^{-j}_{t-p}\}$
Sample from $H^j_t(\mathbf{s}_t)$ a subset $\hat{H}^j_t(\mathbf{s}_t)$ of size l
Update $\Pi^{-j}_{\mathbf{a}^{-j}}(\mathbf{s}_t) = \frac{n_{\hat{H}^j_t(\mathbf{s}_t)}(\mathbf{a}^{-j})}{l}$, $\forall \mathbf{a}^{-j}$
Update current state $\mathbf{s}_t \leftarrow \mathbf{s}_{t+1}$ and increment the time $t \leftarrow t+1$
return current policy.

Q-value Initialization

- Complexity of the (single-agent) *Q*-learning (Koenig & Simmons, 1996)
  - Exponential in the number of states, $\mathcal{S}^j$
    - In the case of the stochastic games, the joint-state space is exponential in the number of agents ($\mathbf{S} = \times_{j=1}^{n} \mathcal{S}^j$)
  - Can be reduced if
    - A goal directed model is chosen
    - The reward function has action-penalty representation
    - The *Q*-values are initialized with an *admissible* and *monotonic* function

Q-Value Initialization in APQ

- Permits the considerable reducing of the explored part of the entire joint-state space
- Based of the following assumptions
  - The stochastic game is goal-oriented
  - The actions are penalized
  - Each agent is provided with its optimal single-agent policy $\overset{\star}{\pi}{}^{j}$ (assuming that all other agents do not exist)
  - In all states and for all j, there is no joint policy $\Pi \in \times_{i=1}^{n} \pi^{i}$ that could be better for j than the optimal single-agent policy $\overset{\star}{\pi}{}^{j}$

Q-Value Initialization in APQ

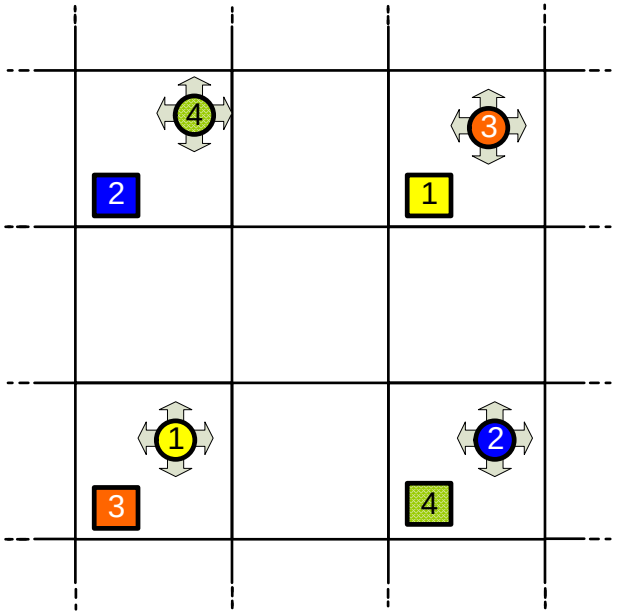
- We initialize the *Q*-values of the SG using the optimal single-agent values:

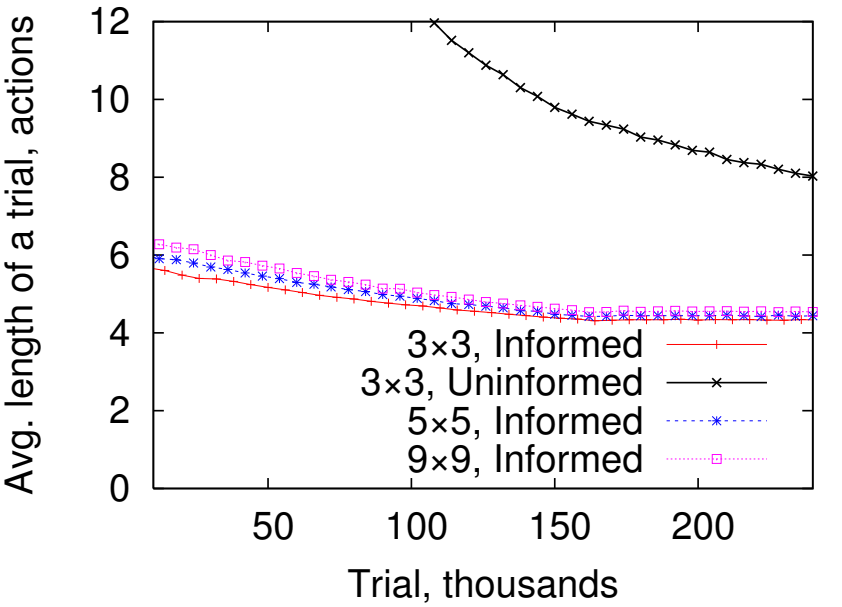
$$\hat{Q}^j(\langle s^j, \mathbf{s}^{-j}\rangle, \langle a^j, \mathbf{a}^{-j}\rangle) \leftarrow \overset{\star}{Q}{}^j(s^j, a^j), \quad \forall \mathbf{a}^{-j}, \forall \mathbf{s}^{-j}$$

  where $\overset{\star}{Q}{}^j(s^j, a^j)$ is the optimal single-agent *Q*-values of agent j

- These single-agent values can be calculated using any dynamic-programming method or the single-agent *Q*-learning (assuming that all other robots was removed from the environment)
- Thereafter, we use the APQ algorithm to make the learning in the SG (when all agents are present and initialized)

Results


4
3
2
1
1
2
3
4


Avg. length of a trial, actions
Trial, thousands
0
2
4
6
8
10
12
50
100
150
200
3×3, Informed
3×3, Uninformed
5×5, Informed
9×9, Informed

Conclusion

- We observed that the course of the initialized learning process is defined to a great extent by that initialization.
- The agents explore a (usually very small) subset of the joint-state space of the problem.
- The non-initialized (or, in other words, zero-initialized) algorithm rapidly explores all joint-states and is unable to find a solution in reasonable time, even in the very simple cases.

References

- (Gies & Chaib-draa, 2006) O. Gies and B. Chaib-draa, “Apprentissage de la coordination multiagent : une méthode basée sur le *Q*-learning par jeu adaptatif”, 2006.
- (Koenig & Simmons, 1996) S. Koenig and R. G. Simmons, “The effect of representation and knowledge on goal-directed exploration with reinforcement-learning
- (Young, 1993) H. P. Young, “The evolution of conventions”. “Econometrica”, 1993.

The End

Thank you!