Multi-Agent Model-Based Reinforcement Learning Experiments in the Pursuit Evasion Game

Bruno Bouzy
Marc Métivier
Outline

- Multi-Agent Learning (MAL)
- Pursuit-Evasion Game (PEG)
  - Previous works
  - Our PEG definition
  - Our purpose
- Centralized approach vs distributed approach
  - SetP and SumP
  - Rmax
  - Experiments
  - Results
- Conclusion and perspectives
The 5 MAL agendas (Shoham & al 2007)

- **Computational agenda**
  - Determine the properties of a multi-player game
    - Nash equilibria, correlated, Pareto-optimality

- **Normative agenda**
  - Describe and study equilibria arising between learning algorithms via game theory

- **Descriptive agenda**
  - Model and describe MAL for human people

- **Prescriptive agenda**
  - How an agent should play to maximise his cumulative reward?
  - Cooperative or not?
    - Cooperation (Nash 1950) = communication (MAS)
MAL experiments

- **Purpose**
  - Experimental comparison of a centralized approach and a distributed approach
  - In a stochastic game, not in a matrix game
  - With tools as optimal as possible

- **Domain selection**
  - The pursuit evasion game is a stochastic game
  - Classical testbed in multi-agent systems
  - Small grids
Pursuit Evasion Game (PEG)

Main features
- Played on a grid by preys and predators
- Simultaneous actions toward adjacent cells
- Cell neighbourhood: 4, 6 or 8 connected
- A predator kills a prey when going on its cell
- Prey: evasion goal or not?

Variations
- Size up to 100x100
- Environment rules: conflict management
- Team reward or individual rewards?
- Toroidal grid or not?
- Obstacles or not?
PEG, Previous works (1/2)

- (Brenda & al, 1985)
  - Pionneering work

- (Levy & Rosenschein 1992)
  - Game-theoretical work on an abstraction of PEG.

- (Korf 1992)
  - Distance heuristics

- (Tan 1993)
  - Specialised predators (hunters and scouts)
  - Communicating predators

- (Haynes & Sen 1996)
  - Limitation: the “straight forward” strategy
PEG, Previous works (2/2)

- **Critics**
  - No Centralized vs Distributed Assessment
  - Prey: No explicit goal, Random, No learning.
  - Model-free RL tools (Q learning)
  - Partial observation (large grids)
  - Communicating predators (Tan 1993)

- **Our approach**
  - Goal: Centralized vs Distributed Assessment
  - Prey: Has the evasion goal, learning.
  - Model-based RL tools (Rmax)
  - Complete observation (3x3 grids)
  - No communication between agents
PEG, 2 episodes

- The prey is escaping

1

\[ \rightarrow \]

2

\[ \rightarrow \]

3

\[ r = -1 \]

- The prey is killed

1

\[ \rightarrow \]

2

\[ r = 1 \]
« SetP » and « SumP »

- **SetP**
  - agent = set of predators
  - indifferenciated predators
  - action = joint action

- **SumP**
  - SumP = macro agent = set of predators
    - elementary agent = predator
    - elementary agent state = set state + agent position
    - action = elementary action
Counting the number of states

- **Raw representation**
  - 9 cells, 9 agents max, each agent has one position
  - \( \#\text{state} = 9^9 \)
  - Too high

- **Bitmap représentation**
  - Set of predators -> bitmap \( \#\text{state} < 512 \)
  - \( 1 + 4 + 64 + 256 = 325 \)
  - SetP: \( \#\text{state} = 9 \times 512 \)
  - SumP: \( \#\text{state} = 9 \times 512 \times 9 \)
  - \#state reduction
  - Not fair for SumP
Counting the number of actions

- SetP: legal or illegal joint actions
  - b, c, d, e, f: legal,
  - g: illégal

- SetP: bitmap representation
  - At most 8 prédateurs with 9 actions
  - $9^8$ actions: legality impossible to check on-line
  - Off-line generation of legal joint actions
  - -> At most 512 joint actions
**Rmax: a model-based RL algorithm**

- (Brafman & Tennenholtz 2002)
- Stochastic games
  - Action quality Q
  - Rmax states
- Policy: optimal action according Q
- Rmax state
  - #transitions to each following state
  - Reward for each transition
- Online Algorithm
  - For each transition,
    - #transitions++; Retour = retour observé
- “global” update
  - Value Iteration
Rmax in practice

- **Theoretically,**
  - After each modification,
    - optimal policy computation
    - Q values updates
  - Polynomial time convergence in \#state
  - Solution to the exploitation-exploration dilemma

- **In practice,**
  - Optimal policy computation: Value Iteration (VI)
  - Call VI every X timesteps
  - Memory space
  - Convergence time
    - beginning: wrong policies, wrong means
    - middle: correct policies, wrong means
    - end: correct policies, correct means
Problems to solve:

(B0)  
(B1)  
(B3)  
(B5)

(C1)  
(C2)  
(C3)
PEG, Experiments (2/2)

- #Episodes = 1,000,000
- maximal episode length = 10
- Episode end:
  - +1 kill
  - -1 evade
  - 0 otherwise
- Predators' set configuration
  - SetP or SumP
  - learning (Rmax)
- Prey configuration
  - random or learning (Rmax)
Against a...

... random prey:
- < 300,000 episodes
  - SumP > SetP
- > 300,000 episodes
  - SetP > SumP

... learning prey:
- SumP et SetP --> -1
Against a...

... random prey:
- < 200,000 episodes
  - SumP >> SetP
- > 200,000 episodes
  - SetP == SumP

... learning prey:
- SumP and SetP → 0.1
- SetP >= SumP
- SetP variance
Against a...

... random prey:
- < 200,000 episodes
  - SumP > SetP
- > 200,000 episodes
  - SetP == SumP

... learning prey:
- SumP and SetP --> 0.2
- SetP >= SumP
C1

- Against a...
- ... random prey:
  - SumP and SetP $\rightarrow 0.8$
  - SumP == SetP
- ... learning prey:
  - SumP and SetP $\rightarrow 0$
  - SumP == SetP
C2

- Against a...
- ... random prey:
  - SumP and SetP $\rightarrow 0.9$
  - SumP $\geq$ SetP
- ... learning prey:
  - SetP $\rightarrow 0.1$
  - SumP $\rightarrow 0$
C3

- Against a...
  - ... random prey:
    - SumP  --> 0.95
    - SetP  --> 0.8
    - SumP > SetP
  - ... learning prey:
    - SetP  --> 0.2
    - SumP  --> 0.1
    - SetP > SumP
PEG, Discussion

- In most cases: SetP >= SumP
- C3: SumP > SetP
- Optimal tactical sequences
- SumP: communicating predators?
PEG, Future works

- Continue the comparison: centralized vs distributed
- Rmax -> Q learning
- Increase the gridsize
  - Partial Observation
  - Q learning
Thank you for your attention...