Artificial Intelligence in Robotics

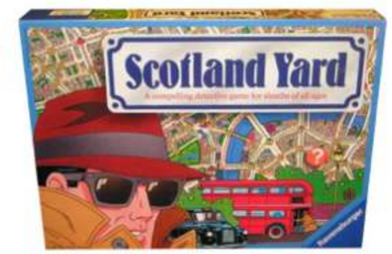
Lecture 9: GT in Robotics

Pavel Rytir
Artificial Intelligence Center
Department of Computer Science, Faculty of Electrical Engineering
Czech Technical University in Prague

Game Theory

- Mathematical framework studying strategies of players in situations where the outcomes of their actions critically depend on the actions performed by the other players.

- Desk games



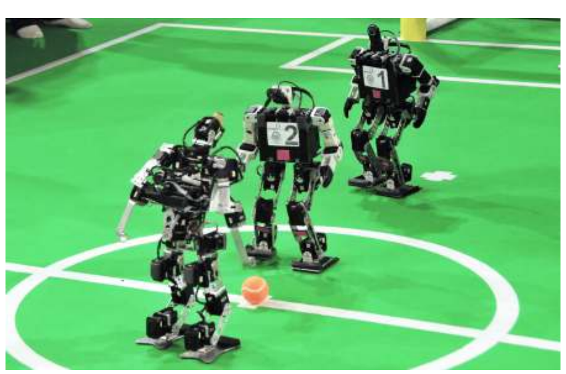
- Poker

- Cyber security

- Auctions

- Football

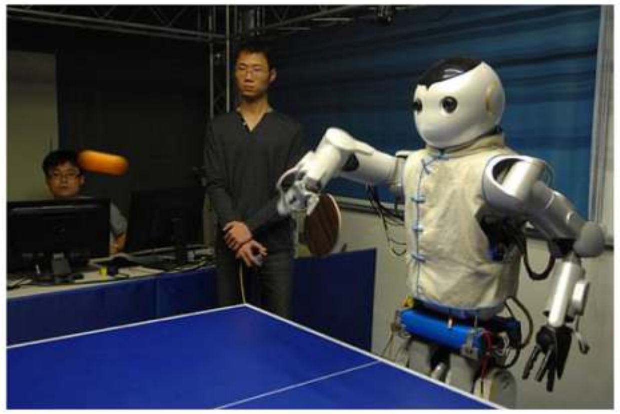
- Robotic football

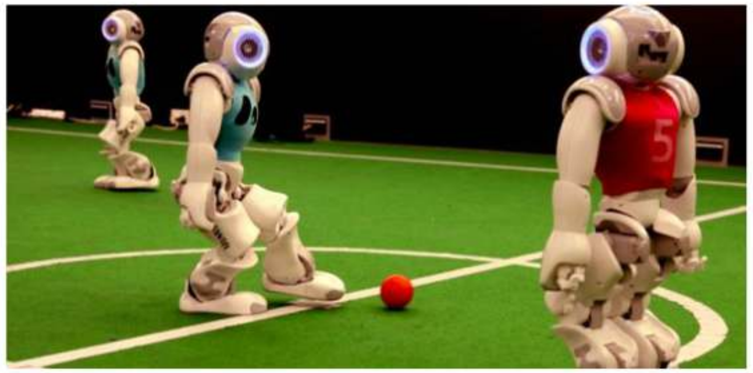


- Security

Game Theory applications in robotics

- Various application of game theory





Adversarial vs. Stochastic vs. Deterministic Environment

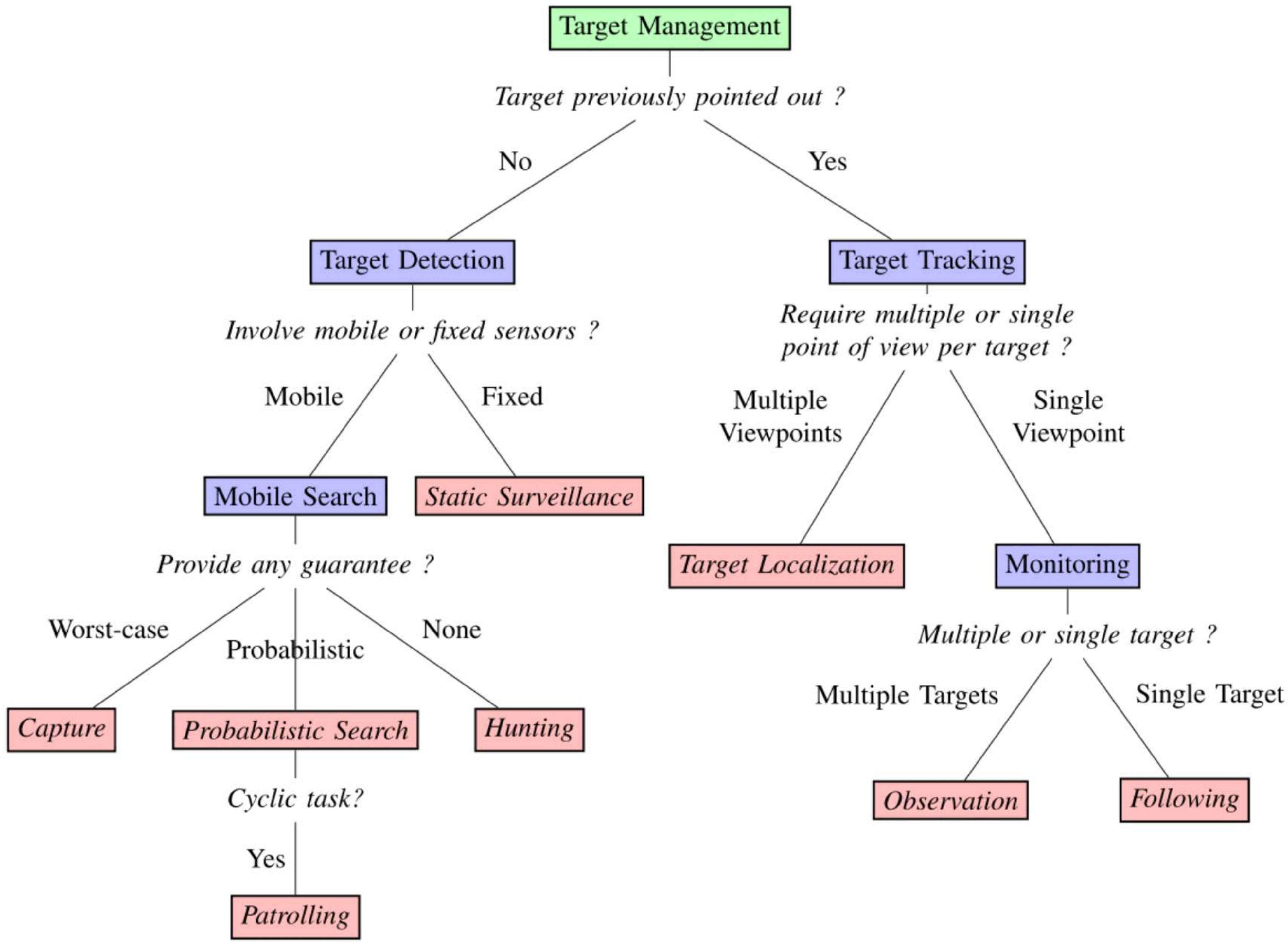
- Deterministic environment
    - The agent can predict exactly the next state of the environment
- Stochastic environment
    - Next state comes from a known distribution
- Adversarial environment
    - Next state comes from an unknown distribution (possibly non stationary)
- Game theory optimizes behavior in adversarial environment.

Game Theory and Robust Optimization

- We want to count with the worst case scenario.
  - The lost person in the woods moves to avoid detection.
  - The planned action depletes the battery the most it can.
- Game theory can be used for robust optimization without adversaries.

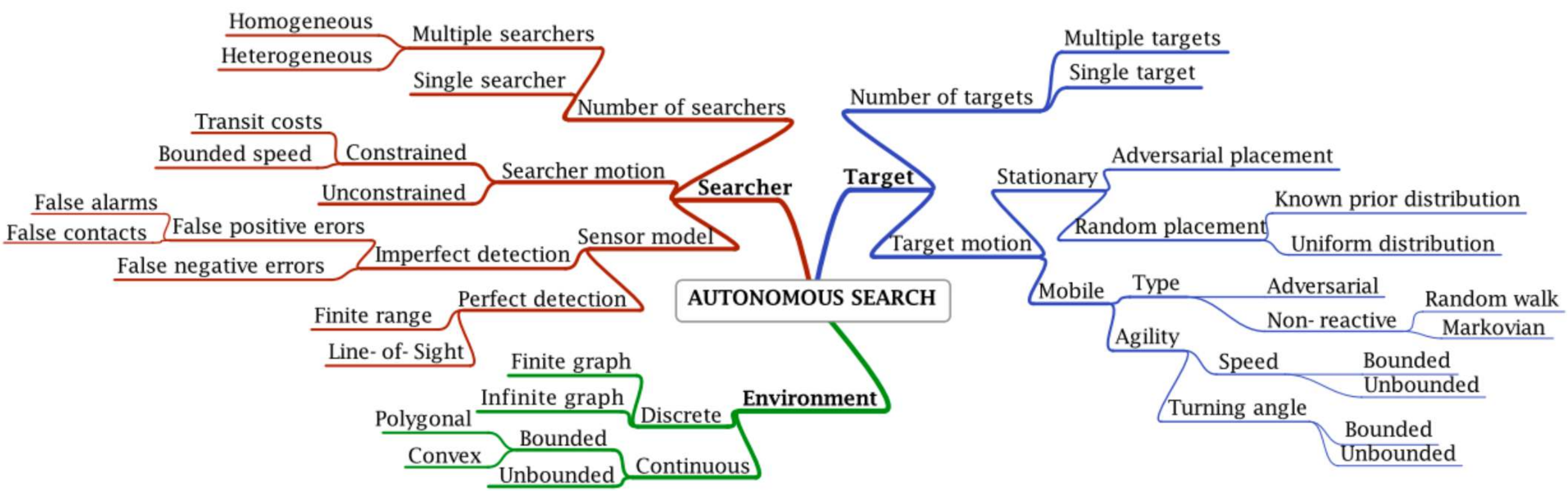
Pursuit-Evasion games

Pursuit-evasion task taxonomy



Robin, C., & Lacroix, S. (2016). Multi-robot target detection and tracking: taxonomy and survey. Autonomous Robots, 40(4), 729–760.

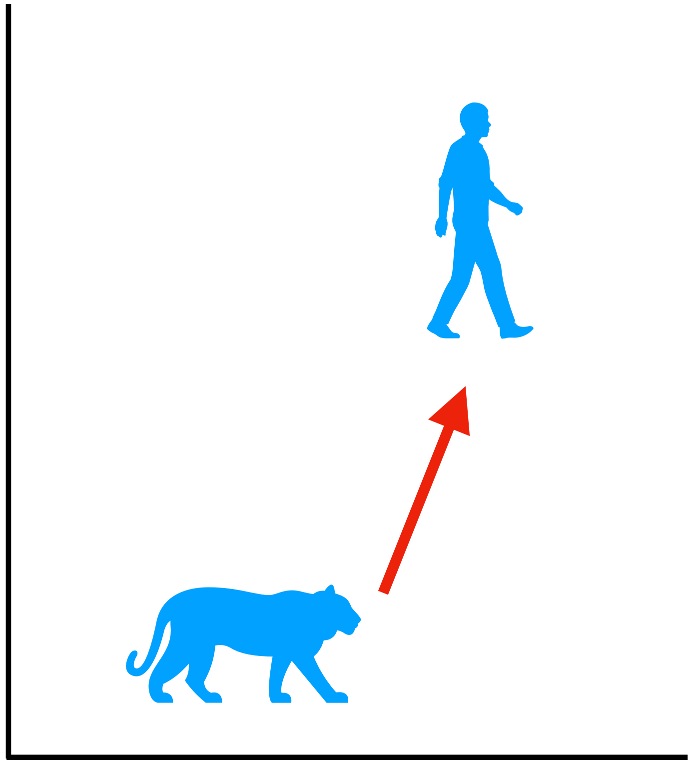
Pursuit evasion problem parameters



Chung, T. H., Hollinger, G. A., & Isler, V. (2011). Search and pursuit-evasion in mobile robotics: A survey. Autonomous Robots, 31(4), 299–316.

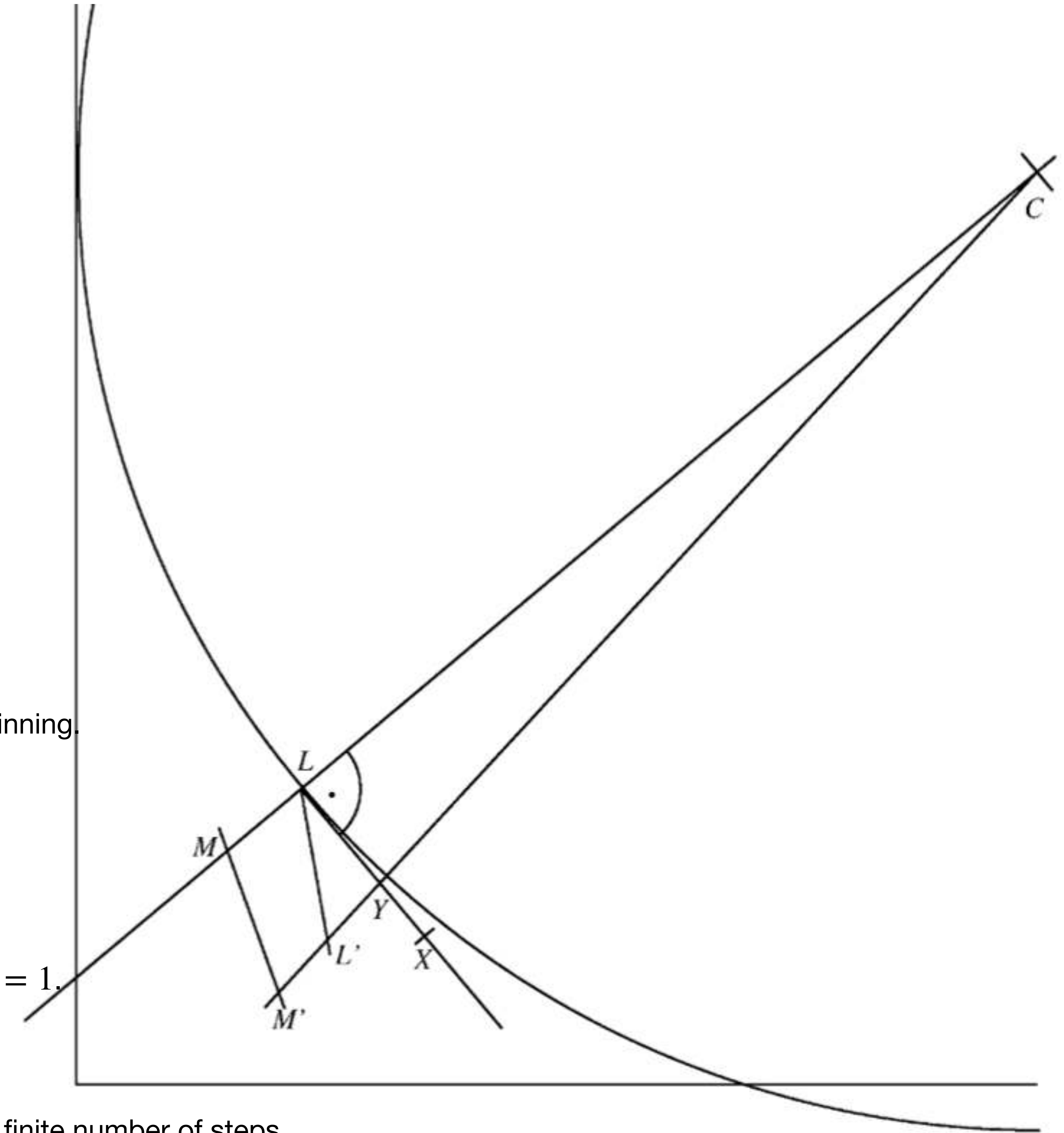
Lion and Man game

- Perfect information capture game
- **Rules:**
  - Arena is the non-negative quadrant of the plane.
  - Both man and lion have unit speed.
  - Alternating moves.
  - In each round man plays first.
  - Each make a move to any point in Eucl. Dist at most 1
  - from current position.
- Time is discreet. Space is Continuous.
- **Goal**: Lion wins if he captures man.
- Man wins if he can keep escaping for inf. time.



Lion and Man game

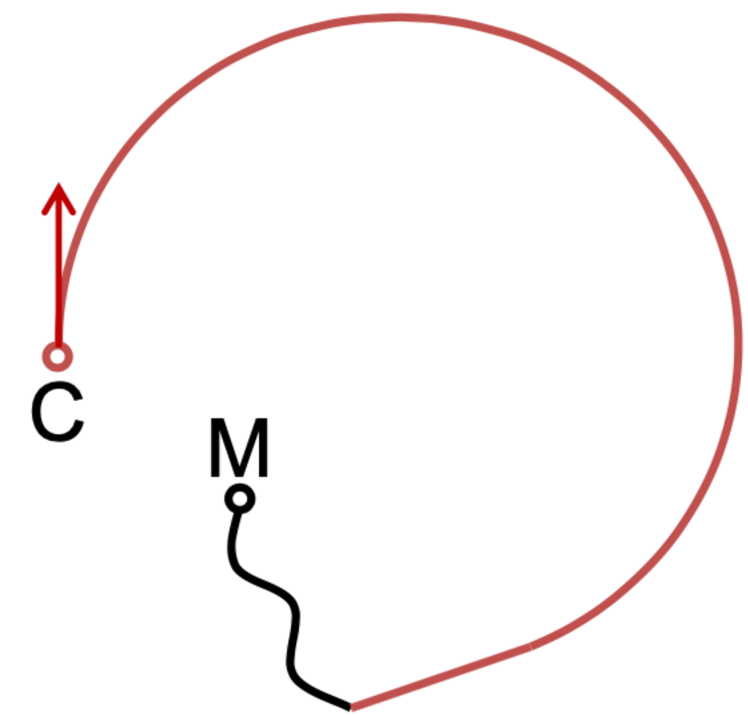
- Let $L_0 = [x_0, y_0]$ and $M_0 = [x'_0, y'_0]$ be initial positions.
- If either $x'_0 \geq x_0$ or $y'_0 \geq y_0$, then man wins.
- If both $x'_0 < x_0$ and $y'_0 < y_0$, the lion wins. Proof:
- Strategy for lion [Sgall 2001]
  - **INIT:** Find point C on line M_0L_0 such that:
  - L_0 is inside M_0C and the circle with center C
  - and radius $r = |CL_0|$ intersects both axes.
  - The point C remains the same during the game.
  - **IN EACH ROUND:** Let M and L denote positions at beginning.
  - Let M' denote point where man moves.
  - If $|M'L| \leq 1$, lion moves to M' and wins.
  - Else Lions moves to L' on the line $M'C$ such that $|L'L| = 1$.
  - The distance between the lion and man decreases.
  - Since the time is discreet, the algorithm converges after finite number of steps.



Lion and Man game

- Analysis [Sgall 2001]:
  - capture time with discrete steps $O(r^2)$
  - no capture in continuous time
  - the lion can get to distance c in time $O(rlog(r/c))$ [Alonso at al 1992]
  - single lion can capture the man in any polygon [Isler et al. 2005]

Homicidal chauffeur game



- [Isaacs 1951]; Added movement constraints
    - unconstrained space
    - pedestrian is slow, but highly maneuverable
    - car is faster, but less maneuverable (Dubin's car)
    - can the car run over the pedestrian?
- The constraints are described by the following differential equations:
- $x'_M = u_M,\ |u_M| \leq 1,\ x'_C = (v\,cos(\theta), v\,sin(\theta)),\ \theta' = u_c,\ u_c \in \{-1,0,1\}$
- It is a special case of Differential games described by the differential equations of the form:
- $x' = f(x, u_1(t), u_2(t)),\ L_i(u_1, u_2) = \int_{t=0}^{T} g_i(x(t), u_1(t), u_2(t))dt$
- These equations are generally analytically intractable.

Incremental sampling based method

- S. Karaman, E. Frazzoli: Incremental Sampling-Based Algorithms for a Class of Pursuit-Evasion Games, 2011.
- 1 evader, several pursuers
- Open-loop evader strategy (for simplicity)
- Stackelberg equilibrium
    - the evader picks and announces her trajectory
    - the pursuers select trajectory afterwards
- Heavily based on RRT* algorithm

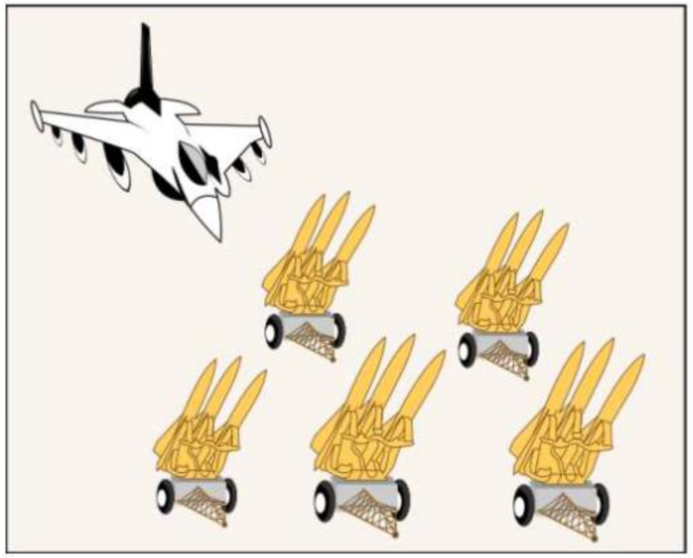


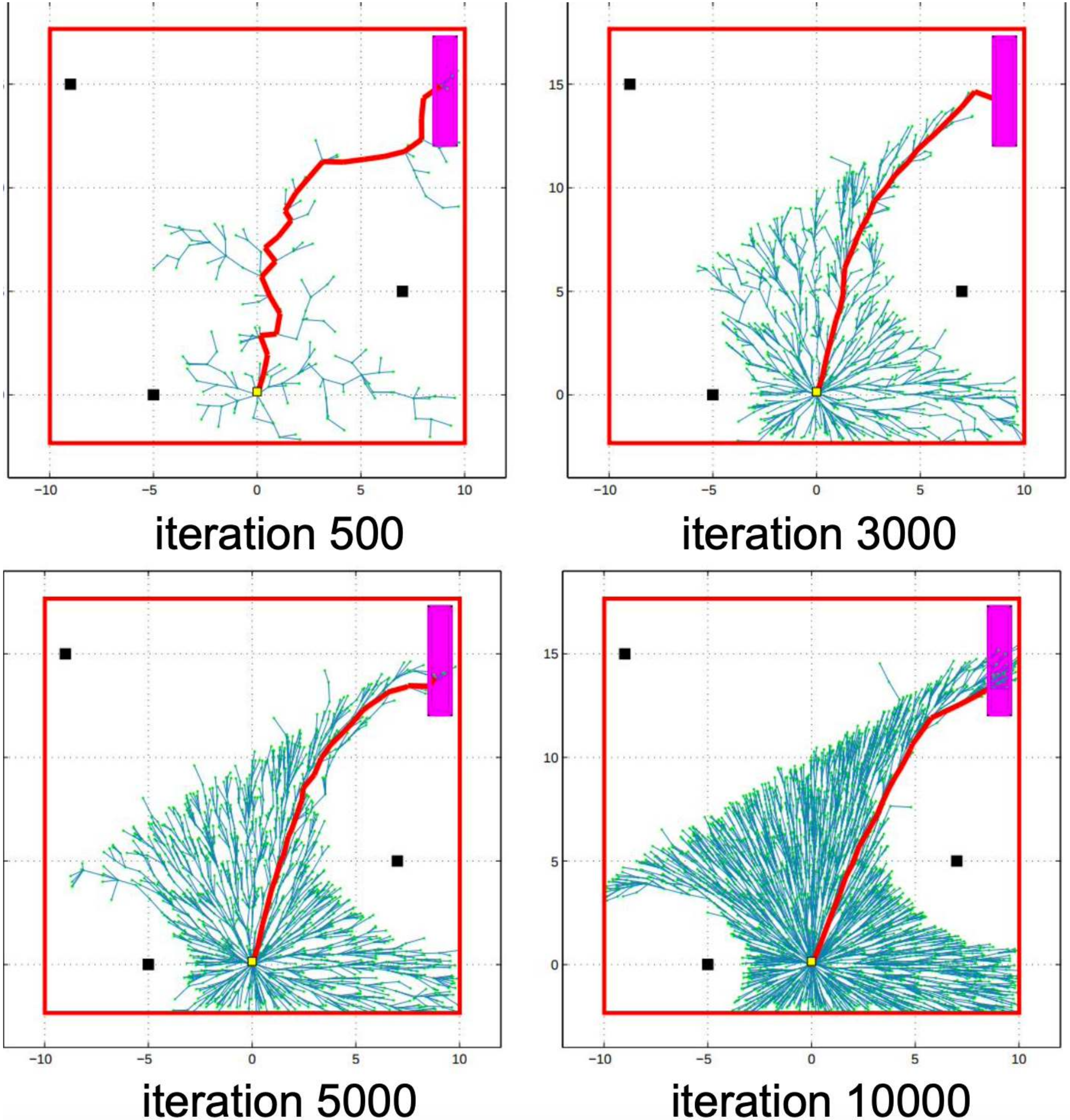
Image by MIT OpenCourseWare

Incremental sampling based method

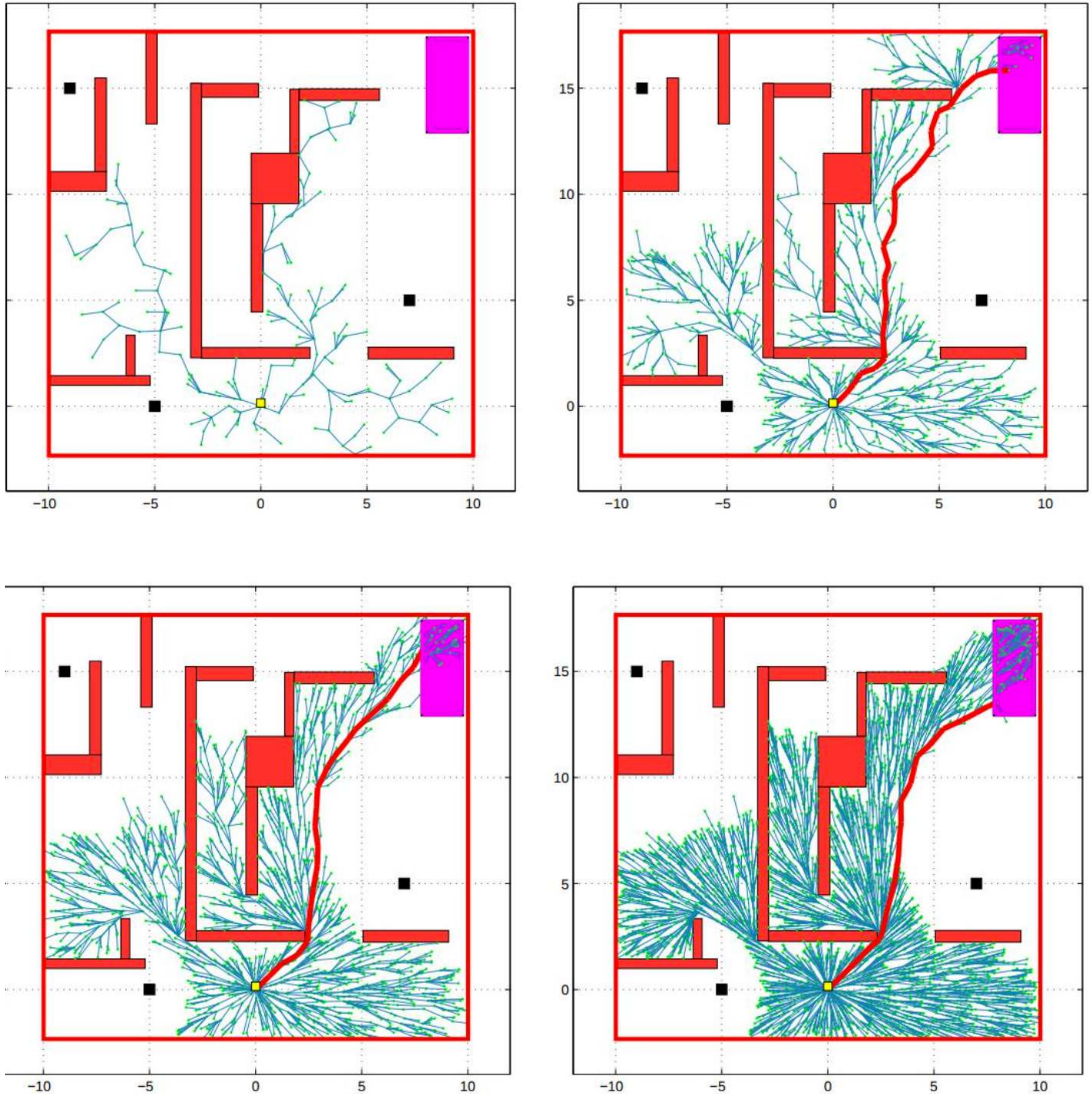
- Pursuit-Evasion Algorithm
- Initialize evader's and pursuer's trees T_e and T_p with starting vertex.
- For i = 1 to N do
  - $T_e, n_{e,new} \leftarrow Grow(T_e)$ [step from RRT*]
  - If $\{n_p \in T_p : dist(n_{e,new}, n_p) \leq f(i) \wedge time(n_p) \leq time(n_{e,ew})\} \neq \emptyset$
    - Then delete $n_{e,new}$ from T_e
  - $T_p, n_{p,new} \leftarrow Grow(T_p)$ [step from RRT*]
  - Let $C = \{n_e \in T_e : dist(n_e, n_{p,new}) \leq f(i) \wedge time(n_{p,new}) \leq time(n_e)\}$
  - Delete $C \cup descendants(C)$ from T_e

For efficiency pick

$$f(i) \approx \frac{log(|T_e|)}{|T_e|}$$



−10 −5 0 5 10
15 10 5 0
iteration 500
iteration 3000
iteration 5000
iteration 10000



15
10
5
0
−10
−5
0
5
10

Normal Form Games

- The normal form, also known as the strategic form, is the most familiar representation of strategic interactions in game theory.

- Most other game theoretic frameworks could be reduced to the normal form (of very big size).

- Definition: A (finite, n-person) normal form game is a tuple (N,A,u) where

- $N = (1, \ldots, n)$ is a finite set of players

- $A = A_1 \times \ldots \times A_n$, where A_i is a finite set of actions available to player i. Each vector $a = (a_1, \ldots, a_n) \in A$ is called an action profile.

- $u = (u_1, \ldots, u_n)$ where $u_i : A \mapsto \mathbb{R}$ is a real valued utility (payoff) function of player i.

- A natural way to represent games is an n-dimensional matrix(tensor).

Prisoner's Dilemma

- Two prisoners. Each can either cooperate (C) with other prisoner during an interrogation or defect (D)
- What is the optimal strategy for them?
- The best outcome is when both cooperate.
- But they will usually both defect.

	C	D
C	$-1, -1$	$-4, 0$
D	$0, -4$	$-3, -3$

Pareto optimality

- **Pareto domination.** Strategy profile s Pareto dominates strategy profile s' if for all $i \in N$, $u_i(s) \geq u_i(s')$, and there exists some $j \in N$ for which $u_j(s) > u_j(s')$.

- Strategy profile s is **Pareto optimal** (Pareto efficient), if there is no another strategy profile $s' \in S$ that Pareto dominates s.

	C	D
C	$-1, -1$	$-4, 0$
D	$0, -4$	$-3, -3$

Nash equilibrium

- **Best response.** Player i's best response to the strategy profile of other players s_{-i} is a strategy $s_i^* \in S_i$ such that $u_i(s_i^*, s_{-i}) \geq u_i(s_i, s_{-i})$ for all strategies $s_i \in S_i$.

P2 BR

P1

	C	D
C	$-1, -1$	$-4, 0$
D	$0, -4$	$-3, -3$

Nash equilibrium

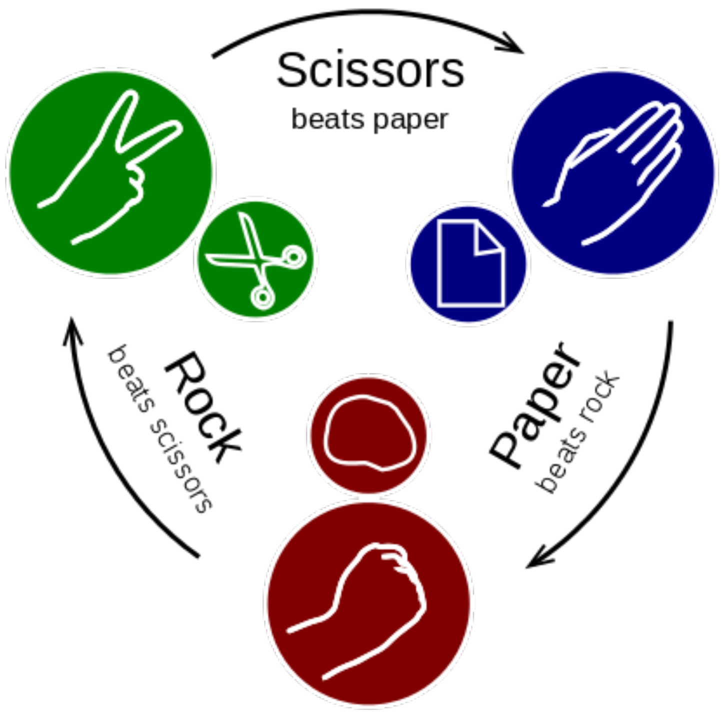
- **Nash equilibrium.** A strategy profile $s = (s_1, \ldots, s_n)$ is a Nash equilibrium if, for all players i, s_i is a best response to s_{-i}.

	C	D
C	$-1, -1$	$-4, 0$
D	$0, -4$	$-3, -3$

Mixed strategy

	Rock	Paper	Scissors
Rock	$0, 0$	$-1, 1$	$1, -1$
Paper	$1, -1$	$0, 0$	$-1, 1$
Scissors	$-1, 1$	$1, -1$	$0, 0$

Figure 3.7: Rock, Paper, Scissors game.



Mixed strategy

- **Mixed strategy**.
- Let X be a set. Let $\Pi(X)$ be the set of all probabilistic distributions over X.
- The set of all mixed strategies for player i is $S_i = \Pi(A_i)$.
- **Expected utility of a mixed strategy.**
- The expected utility u_i for player i of the mixed strategy profile $s = (s_1, \ldots, s_n)$ is defined as: $u_i(s) = \sum_{a \in A} u_i(a) \prod_{j=1}^{n} s_j(a_j)$

Finding Nash equilibria

- Theorem (Nash, 1951) Every game with a finite number of players and action profiles has at least one Nash equilibrium.

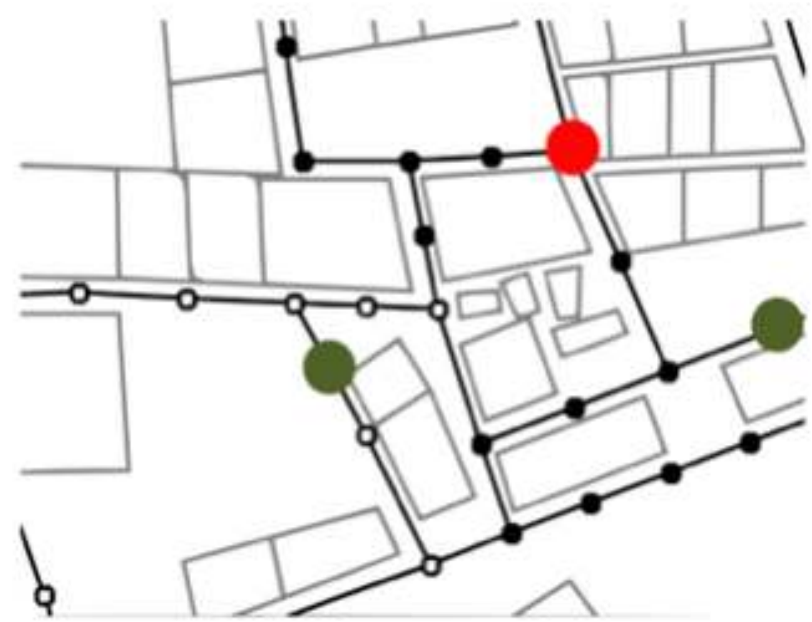
- A two players game is zero sum if for each strategy profile $a \in A_1 \times A_2$ it holds $u_1(a) + u_2(a) = 0$

- Nash equilibrium of two players zero sum game can be computed as a linear program

$$
\begin{aligned}
\text{minimize} \quad & U_1^* \\
\text{subject to} \quad & \sum_{k \in A_2} u_1(a_1^j, a_2^k) \cdot s_2^k \leq U_1^* && \forall j \in A_1 \\
& \sum_{k \in A_2} s_2^k = 1 \\
& s_2^k \geq 0 && \forall k \in A_2
\end{aligned}
$$

$u_1(\,.\,)$ are constants. s_2 and U_1^* are variables.

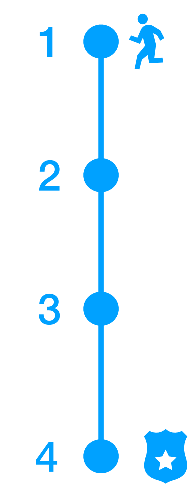
Cops and robbers game

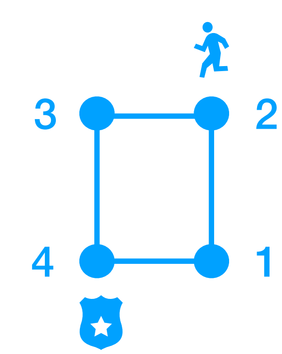


- Map is represented as a graph $G = (V, E)$
- Cops and robbers are in vertices.
- Alternating moves along edges.
- Perfect information game.
- Cops win if they step at the same vertex as the robber.
- Robbers win if they can keep escaping for infinite time.
- Cop number of a graph is the minimum number of cops to guarantee capture of the robber regardless of their initial positions.

Cops and robbers game

- Let v be a vertex. Neighborhood of v is: $N(v) = \{u \in V : (u, v) \in E\}$
- **Marking algorithm**.
- It determines who wins and provides strategy
- Single cop and robber
- 1. For all $v \in V$ mark state (v, v) [e.g. add tuple (v, v) into a hashset]
- 2. For all unmarked (c, r)
  - If $\forall r' \in N(r) \exists c' \in N(c)$ such that (c', r') is marked, then mark (c, r)
- 3. If there are new marks, go to 2.
- If there is an unmarked state, the robber wins.
- If there is none. The cop strategy follows from the marking order.

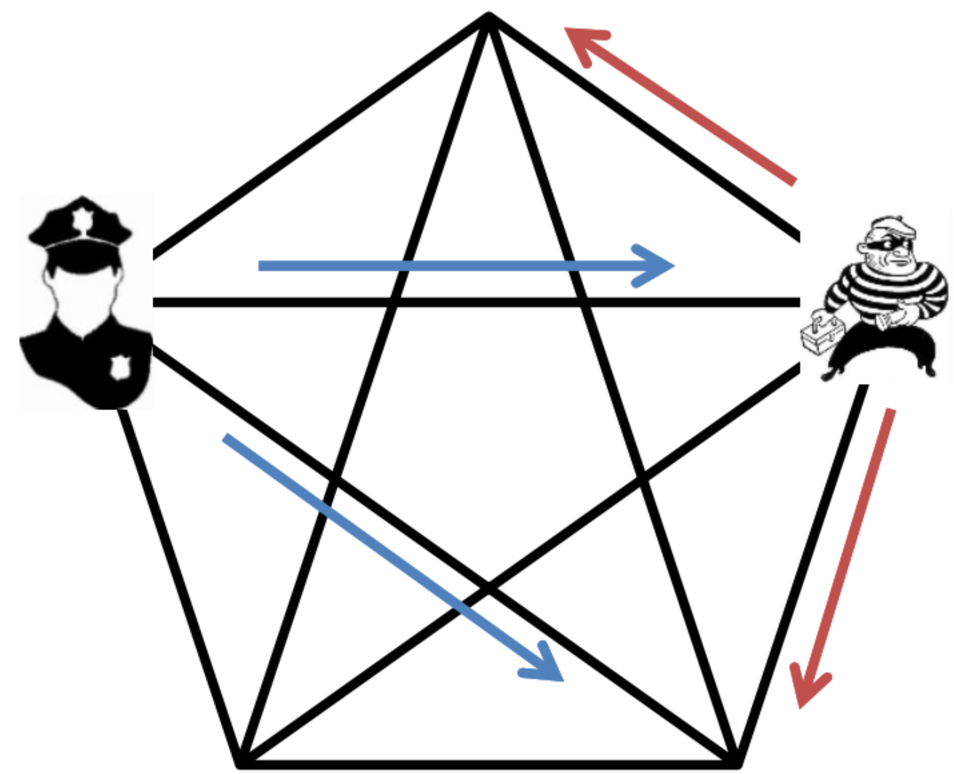




Cops and robbers game

- Marking algorithm can be generalized to k cops. It uses tuples $(c_1, \ldots, c_k, r)$.
- Time complexity of marking algorithm for k cops is $O(2^{n(k+1)})$.
- Determining whether k cops with a given locations can capture a robber on a given undirected graph is EXPTIME-complete [Goldstein and Reingold 1995].
- The cop number of trees and cliques is one.
- The cop number on planar graphs is at most three [Aigner and Fromme 1984].

Cops and robbers game



- Simultaneous moves
  - No deterministic strategy
  - Optimal strategy is randomized

Stochastic (Markov) games

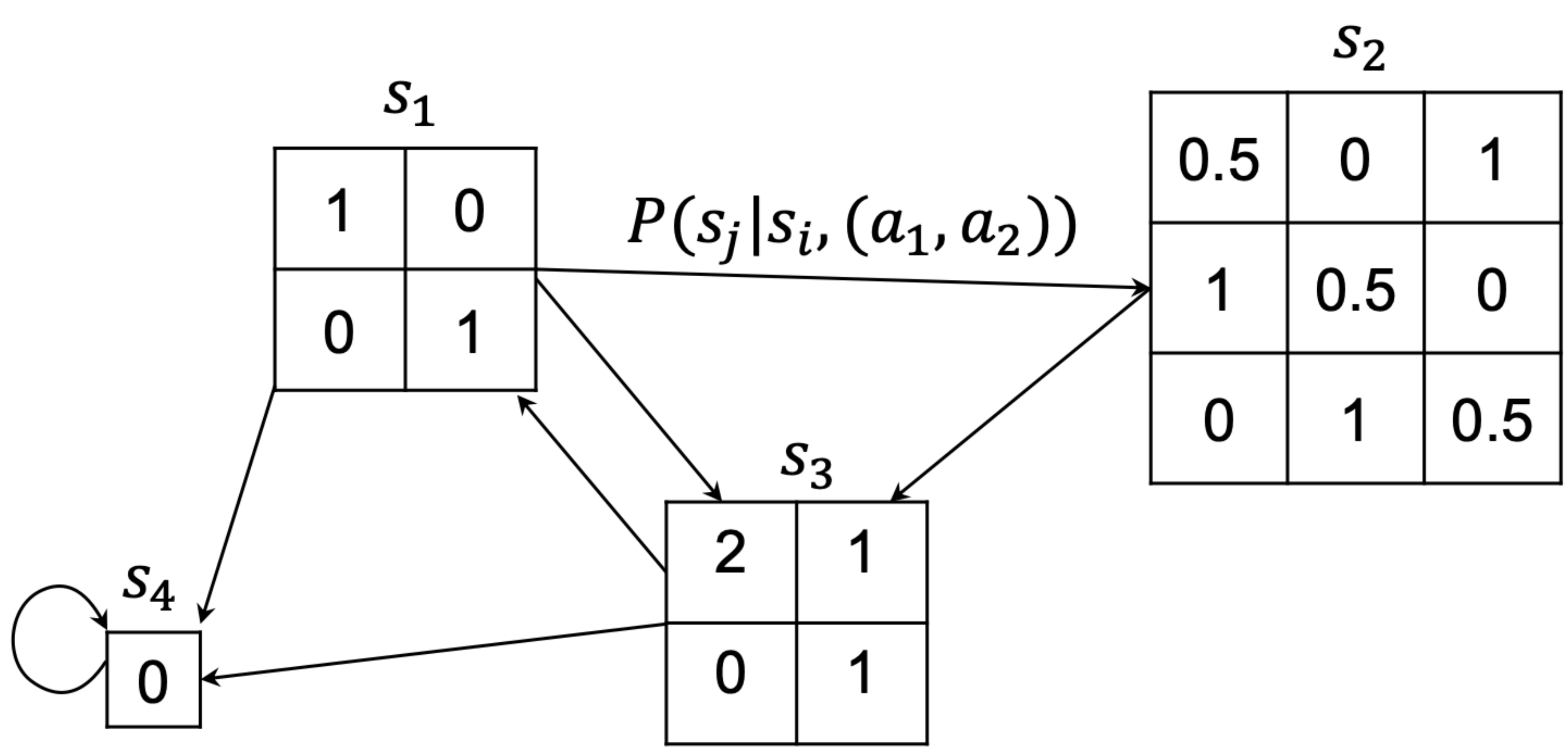
N is the set of players

S is the set of states (games)

$A = A_1 \times \cdots \times A_n$, where A_i is the set of actions of player i

$P: S \times A \times S \to [0,1]$ is the transition probability function

$R = r_1, \dots, r_n$, where $r_i: S \times A \to \mathbb{R}$ is immediate payoff for player i



Stochastic (Markov) games

Markovian policy: $\sigma_i : S \rightarrow \Delta(A)$

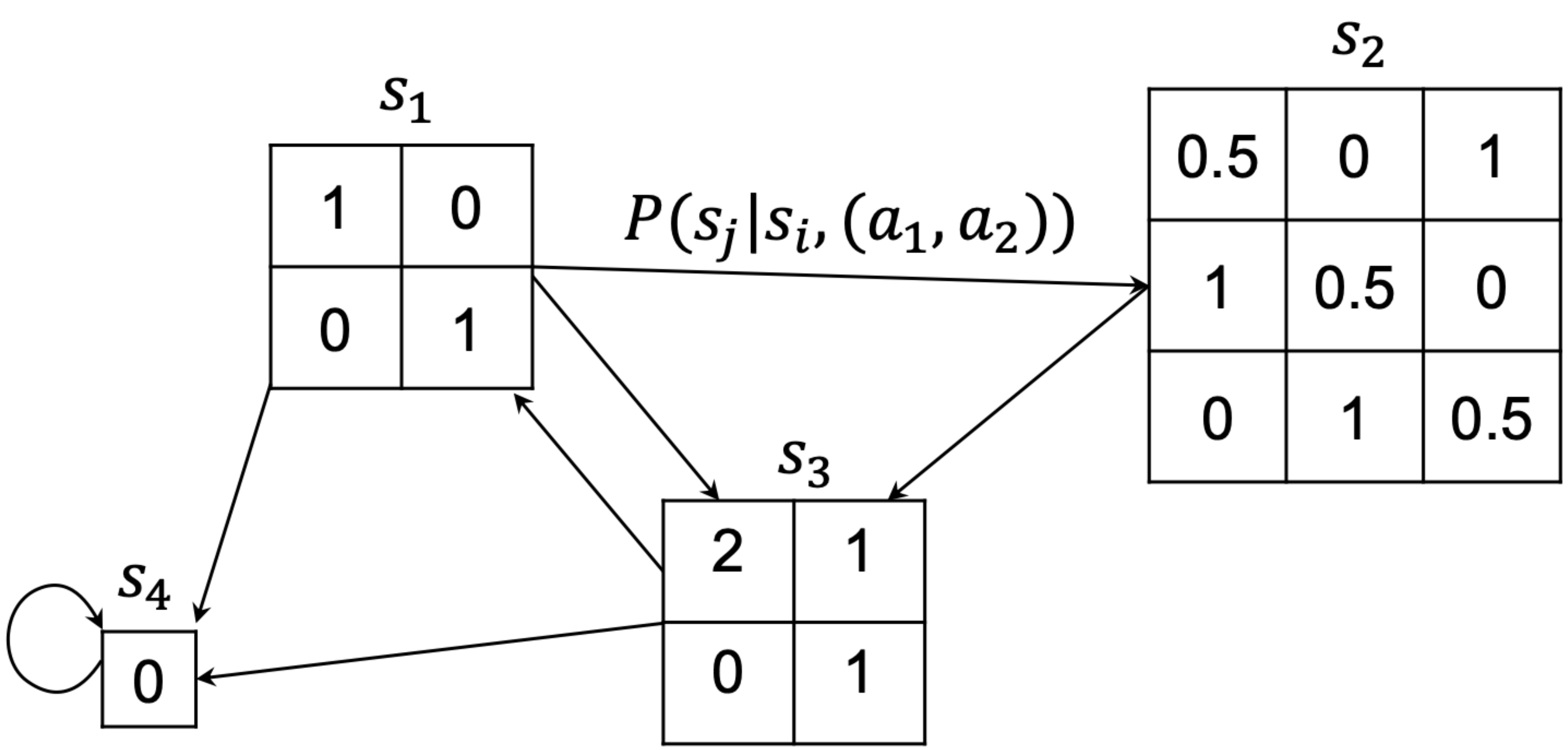
Objectives

Discounted payoff: $\sum_{t=0}^{\infty} \gamma^t r_i(s_t, a_t), \gamma \in [0,1)$

Mean payoff: $\lim_{T\rightarrow\infty} \frac{1}{T} \sum_{t=0}^{T} r_i(s_t, a_t)$

Reachability: $P\big(reach(G)\big),\ \ G \subseteq S$

Finite vs. infinite horizon



Value iteration in stochastic games

Adaptation of algorithm from Markov decision processes (MDP)

For zero-sum, discounted, infinite horizon stochastic games

- $\forall s \in S$ initialize $v(s)$ arbitrarily (e.g., $v(s) = 0$)
- until v converges
  - for all $s \in S$
    - for all $(a_1, a_2) \in A(s)$

$$Q(a_1, a_2) = \mathrm{r}(\mathrm{s}, \mathrm{a}_1, \mathrm{a}_2) + \gamma \sum_{s' \in S} P(s'|s, a_1, a_2) v(s')$$

$$v(s) = \max_x \min_y xQy$$ // solves the matrix game Q

Converges to optimum if each state is updated infinitely often

- the state to update can be selected (pseudo)randomly

Pursuit evasion as SG

$N = (e, p)$ is the set of players

$S = \left(v_e,\ v_{p_1}, \dots, v_{p_n}\right) \in V^{n+1} \cup T$ is the set of states

$A = A_e \times A_p$, where $A_e = E, A_p = E^n$ is the set of actions

$P: S \times A \times S \rightarrow [0,1]$ is deterministic movement along the edges

$R = r_e, r_p$, where $r_e = -r_p$ is one if the evader is captured

Summary

PEGs studied in various assumptions

Simplest cases can be solved analytically

More complex cases have problem-specific algorithms

Even more complex cases best handled by generic AI methods

Resources

Game theory basics

Yoav Shoham, Kevin Leyton-Brown: Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations. [Sections 3.2, 4.1, 6.3] http://www.masfoundations.org

Littman, M. L. (1994). Markov games as a framework for multi-agent reinforcement learning. Machine Learning Proceedings 1994, 157–163.

Pursuit-evasion games

Robin, C., & Lacroix, S. (2016). Multi-robot target detection and tracking: taxonomy and survey. Autonomous Robots, 40(4), 729–760.

Chung, T. H., Hollinger, G. A., & Isler, V. (2011). Search and pursuit-evasion in mobile robotics: A survey. Autonomous Robots, 31(4), 299–316.

Sgall J. (2001). Solution of David Gale's lion and man problem. Theoretical Computer Science. 259(1-2):663-70.

Homicidal chauffeur game: http://sector3.imm.uran.ru/poland2008patsko/index.html

S. Karaman, E. Frazzoli. Incremental Sampling-Based Algorithms for a Class of Pursuit-Evasion Games, 2011.