# Învățare automată - Laboratorul 4
# Procese Markov de Decizie

Mihai Trăscău
Laboratorul AIMAS
Facultatea de Automatică și Calculatoare

11 martie 2013

## 1 Introducere

În acest laborator veți aplica conceptul de *învățare prin recompensă* și veți implementa rezolvări ale problemelor descrise de *procese Markov de decizie*.

## 2 Procese Markov de Decizie

Procesele Markov de Decizie sunt definite prin tuplul $< S, A, T(\cdot,\cdot,\cdot), R(\cdot) >$, unde:

- $S$ reprezintă mulțimea stărilor
- $A$ reprezintă mulțimea acțiunilor
- $T(s, a, s')$ reprezintă probabilitatea ca acțiunea $a$ efectuată în starea $s$ să ducă în starea $s'$ (probabilitate de tranziție)
- $R(s)$ reprezintă recompensa imediată pentru ajungerea în starea $s$

Scopul este de a descrie un sistem folosind formalismul de mai sus și apoi de a găsi o politică optimă $\pi^*(s)$, care specifică ce acțiune trebuie aleasă în starea $s$ pentru a maximiza recompensa pe termen lung.

## 3 Metoda „Value Iteration"

Pentru a afla funcția $\pi(s)$ putem aplica metoda iterării valorii („value iteration") în care se calculează până la convergență formula

$$U^t(s) = R(s) + \delta \max_a \left\{ \sum_{s'} T(s, a, s') U^{t-1}(s') \right\} \tag{1}$$

unde $U^t(s)$ reprezintă utilitatea asociată stării $s$ la momentul $t$, iar $\delta$ reprezintă factorul de „discount". După terminarea calculelor, $\pi^*(s)$ se definește:

$$\pi^*(s) = arg \max_a \left\{ R(s) + \delta \sum_{s'} T(s, a, s') U^*(s') \right\} \tag{2}$$

## 4 Metoda „Policy Iteration”

Metoda iterării politicii („policy iteration”) se pornește de la o politică $\pi_0(s)$ arbitrară care va fi apoi îmbunătățită. Pornind de la $\pi_0(s)$ se va calcula utilitatea fiecărei stări până la convergență folosind formula:

$$U^t(s) = R(s, \pi_0(s)) + \delta \max_a \left\{ \sum_{s'} T(s, \pi_0(s), s') U^{t-1}(s') \right\} \tag{3}$$

Utilizând aceste utilități, se va determina $\pi_1(s)$:

$$\pi_1(s) = arg \max_a \left\{ R(s) + \delta \sum_{s'} T(s, a, s') U(s') \right\} \tag{4}$$

La pasul următor $\pi_1(s)$ devine $\pi_0(s)$. Cei doi pași se vor repeta până când politica nu se va mai modifica.

## 5 Cerințe

Un agent se mișcă într-un mediu sub forma unei matrice bidimensionale. Fiecare celulă din matrice are asociată o recompensă pe care agentul o primește în momentul când ajunge în celula respectivă. Mișcările agentului sunt stohastice și sunt definite după cum urmează:

- mișcare în sus - ajunge cu probabilitate:
  - 0.9 în celula de deasupra
  - 0.1 în celula din stânga
- mișcare în dreapta - ajunge cu probabilitate:
  - 1 în celula din dreapta
- mișcare în jos - ajunge cu probabilitate:
  - 0.6 în celula de jos
  - 0.2 în celula din stânga
  - 0.2 în celula din dreapta
- mișcare în stânga - ajunge cu probabilitate:
  - 0.8 în celula din stânga
  - 0.2 în celula de jos

Implementați „value iteration” pentru un astfel de agent aflat într-un mediu ale căror recompense sunt definite în fișierul `recompense.txt`.

## Bonus

Implementați „policy iteration” pentru aceeași problemă.