Methods in AI Research: Markov models for multi-agent learning

Part V: Markov reward processes

Gerard Vreeswijk

Intelligent Systems Group, Computer Science Department
Faculty of Sciences, Utrecht University, The Netherlands

October, 2016
Definition

A Markov reward process is a Markov process with rewards on transitions.
A Markov reward process is a Markov process with rewards on transitions.

**Definition (Markov reward process)**

A **Markov reward process** is a Markov process where transitions issue real-valued rewards.
A Markov reward process is a Markov process with rewards on transitions.

**Definition (Markov reward process)**

A *Markov reward process* is a Markov process where transitions issue real-valued *rewards*.

Markov reward processes are also known as *Markov reward chains*, or *valued Markov chains*.
A Markov reward process is a Markov process with rewards on transitions.

**Definition (Markov reward process)**

A *Markov reward process* is a Markov process where transitions issue real-valued *rewards*.

Markov reward processes are also known as *Markov reward chains*, or *valued Markov chains*.

Example

\[
P = \begin{pmatrix}
0.2 & 0.8 \\
0.3 & 0.7 \\
0.3 & 0.7
\end{pmatrix}, \text{ with reward matrix } R = \begin{pmatrix}
1 & 2 \\
0 & 3 \\
0 & 0
\end{pmatrix}.
\]
Example

\[ P = \begin{pmatrix} 0.2 & 0.8 \\ 0.3 & 0.7 \end{pmatrix}, \text{ with reward matrix } R = \begin{pmatrix} 1 & 2 \\ 0 & 3 \end{pmatrix}. \]
Example

\[ P = \begin{pmatrix} 0.2 & 0.8 \\ 0.3 & 0.7 \end{pmatrix}, \text{ with reward matrix } R = \begin{pmatrix} 1 & 2 \\ 0 & 3 \end{pmatrix}. \]

Transition graph with rewards:

\[ p = 0.2, r = 1 \]

\[ p = 0.8, r = 2 \]

\[ p = 0.7, r = 3 \]

\[ p = 0.3 \]
Let $\omega$ be a realisation of a Markov process:

$$\omega = s_0, s_1, s_2, \ldots,$$

where $s_0$ is a start state.
Let $\omega$ be a realisation of a Markov process:

$$\omega = s_0, s_1, s_2, \ldots,$$

where $s_0$ is a start state.

The rewards that accumulate are $r_1, r_2, r_3, \ldots$:

$$s_0 \xrightarrow{\text{reward } r_1} s_1 \xrightarrow{\text{reward } r_2} s_2 \xrightarrow{\text{reward } r_3} \ldots$$
Total reward

Let $\omega$ be a realisation of a Markov process:

$$\omega = s_0, s_1, s_2, \ldots,$$

where $s_0$ is a start state.

The rewards that accumulate are $r_1, r_2, r_3, \ldots$:

$$s_0 \rightarrow_{\text{reward } r_1} s_1 \rightarrow_{\text{reward } r_2} s_2 \rightarrow_{\text{reward } r_3} \ldots$$

**Definition (Value of a state)**

$$r_k \mid (s_0 = i) : \text{reward at } k\text{th step, given } s_0 = i,$$
Total reward

Let $\omega$ be a realisation of a Markov process:

$$\omega = s_0, s_1, s_2, \ldots,$$

where $s_0$ is a start state.

The rewards that accumulate are $r_1, r_2, r_3, \ldots$:

$$s_0 \xrightarrow{\text{reward } r_1} s_1 \xrightarrow{\text{reward } r_2} s_2 \xrightarrow{\text{reward } r_3} \ldots$$

**Definition (Value of a state)**

$$r_k \mid (s_0 = i) : \text{reward at } k\text{th step, given } s_0 = i,$$

$$R \mid (s_0 = i), \text{ the total reward} : (r_1 + r_2 + r_3 + \ldots) \mid s_0 = i,$$
Let $\omega$ be a realisation of a Markov process:

$$\omega = s_0, s_1, s_2, \ldots,$$

where $s_0$ is a start state.

The rewards that accumulate are $r_1, r_2, r_3, \ldots$:

$$s_0 \xrightarrow{\text{reward } r_1} s_1 \xrightarrow{\text{reward } r_2} s_2 \xrightarrow{\text{reward } r_3} \ldots$$

**Definition (Value of a state)**

- $r_k \mid (s_0 = i)$: reward at $k$th step, given $s_0 = i$,
- $R \mid (s_0 = i)$, the **total reward** : $(r_1 + r_2 + r_3 + \ldots) \mid s_0 = i$,
- $V(i)$, the **value** of state $i$ : $E[R \mid s_0 = i]$. 
The Bellman equation

\[ V(i) = \sum_{all \, j} p_{ij} [r_{ij} + V(j)], \]

for every state \( i \).
The Bellman equation

**Theorem (The Bellman equation)**

\[ V(i) = \sum_{all\ j} p_{ij} [r_{ij} + V(j)], \]

for every state \( i \).

This system of linear equations enables us to compute state values in two ways:
The Bellman equation

**Theorem (The Bellman equation)**

\[ V(i) = \sum_{all \ j} p_{ij} [r_{ij} + V(j)] , \]

for every state \( i \).

This system of linear equations enables us to compute state values in two ways:

1. Solve the equations *analytically* with the help of linear algebra to obtain an exact solution.
The Bellman equation

Theorem (The Bellman equation)

\[ V(i) = \sum_{all \ j} p_{ij} [r_{ij} + V(j)], \]

for every state \( i \).

This system of linear equations enables us to compute state values in two ways:

1. Solve the equations \textit{analytically} with the help of linear algebra to obtain an exact solution.
2. Turn the equation symbols “\( = \)” into an assignment operator “\( := \)” and keep \textit{iterating} the assignments. (Start values do not matter.)
Proof of the Bellman equation

\[ V(i) \]
Proof of the Bellman equation

\[ V(i) = E[R \mid s_0 = i] \]
Proof of the Bellman equation

\[ V(i) = E[R \mid s_0 = i] \]
\[ = E[r_1 + r_2 + r_3 + \cdots \mid s_0 = i] \]
Proof of the Bellman equation

\[ V(i) = E[R \mid s_0 = i] \]
\[ = E[r_1 + r_2 + r_3 + \cdots \mid s_0 = i] \]
\[ = \sum \limits_{\text{all } j} E[r_1 + r_2 + r_3 + \cdots \mid s_1 = j, s_0 = i] p_{ij} \]
Proof of the Bellman equation

\[ V(i) = E[R \mid s_0 = i] \]

\[ = E[r_1 + r_2 + r_3 + \cdots \mid s_0 = i] \]

\[ = \sum_{\text{all } j} E[r_1 + r_2 + r_3 + \cdots \mid s_1 = j, s_0 = i] p_{ij} \]

\[ = \sum_{\text{all } j} (E[r_1 \mid s_1 = j, s_0 = i] + E[r_2 + r_3 + \cdots \mid s_1 = j, s_0 = i]) p_{ij} \]
Proof of the Bellman equation

\[ V(i) = E[R \mid s_0 = i] \]
\[ = E[r_1 + r_2 + r_3 + \cdots \mid s_0 = i] \]
\[ = \sum_{\text{all } j} E[r_1 + r_2 + r_3 + \cdots \mid s_1 = j, s_0 = i] p_{ij} \]
\[ = \sum_{\text{all } j} (E[r_1 \mid s_1 = j, s_0 = i] + E[r_2 + r_3 + \cdots \mid s_1 = j, s_0 = i]) p_{ij} \]
\[ = \sum_{\text{all } j} (r_{ij} + E[r_2 + r_3 + \cdots \mid s_1 = j]) p_{ij} \]
Proof of the Bellman equation

\[ V(i) = E[R \mid s_0 = i] \]
\[ = E[r_1 + r_2 + r_3 + \cdots \mid s_0 = i] \]
\[ = \sum_{\text{all } j} E[r_1 + r_2 + r_3 + \cdots \mid s_1 = j, s_0 = i] p_{ij} \]
\[ = \sum_{\text{all } j} (E[r_1 \mid s_1 = j, s_0 = i] + E[r_2 + r_3 + \cdots \mid s_1 = j, s_0 = i]) p_{ij} \]
\[ = \sum_{\text{all } j} (r_{ij} + E[r_2 + r_3 + \cdots \mid s_1 = j]) p_{ij} \]
\[ = \sum_{\text{all } j} (r_{ij} + E[r'_1 + r'_2 + \cdots \mid s_0 = j]) p_{ij} \]
\[ V(i) = E[R \mid s_0 = i] \]
\[ = E[r_1 + r_2 + r_3 + \cdots \mid s_0 = i] \]
\[ = \sum_{\text{all } j} E[r_1 + r_2 + r_3 + \cdots \mid s_1 = j, s_0 = i] p_{ij} \]
\[ = \sum_{\text{all } j} \left( E[r_1 \mid s_1 = j, s_0 = i] + E[r_2 + r_3 + \cdots \mid s_1 = j, s_0 = i] \right) p_{ij} \]
\[ = \sum_{\text{all } j} \left( r_{ij} + E[r_2 + r_3 + \cdots \mid s_1 = j] \right) p_{ij} \]
\[ = \sum_{\text{all } j} \left( r_{ij} + E[r'_1 + r'_2 + \cdots \mid s_0 = j] \right) p_{ij} \]
\[ = \sum_{\text{all } j} \left( r_{ij} + E[R \mid s_0 = j] \right) p_{ij} \]
Proof of the Bellman equation

\[ V(i) = E[R \mid s_0 = i] \]
\[ = E[r_1 + r_2 + r_3 + \cdots \mid s_0 = i] \]
\[ = \sum_{\text{all } j} E[r_1 + r_2 + r_3 + \cdots \mid s_1 = j, s_0 = i] p_{ij} \]
\[ = \sum_{\text{all } j} (E[r_1 \mid s_1 = j, s_0 = i] + E[r_2 + r_3 + \cdots \mid s_1 = j, s_0 = i]) p_{ij} \]
\[ = \sum_{\text{all } j} (r_{ij} + E[r_2 + r_3 + \cdots \mid s_1 = j]) p_{ij} \]
\[ = \sum_{\text{all } j} (r_{ij} + E[r'_1 + r'_2 + \cdots \mid s_0 = j]) p_{ij} \]
\[ = \sum_{\text{all } j} (r_{ij} + E[R \mid s_0 = j]) p_{ij} \]
\[ = \sum_{\text{all } j} p_{ij} [r_{ij} + V(j)]. \]
For the above Markov reward process, with Bellman:

\[
\begin{align*}
V(a) &= 0.8(2 + V(b)) + 0.2(1 + V(a)), \\
V(b) &= 0.7(3 + V(c)) + 0.3V(a), \\
V(c) &= 1.0(0 + V(c))
\end{align*}
\]
For the above Markov reward process, with Bellman:

\[
\begin{align*}
V(a) &= 0.8(2 + V(b)) + 0.2(1 + V(a)), \\
V(b) &= 0.7(3 + V(c)) + 0.3V(a), \\
V(c) &= 1.0(0 + V(c))
\end{align*}
\]

or, with abuse of notation,

\[
\begin{align*}
a &= 0.8(2 + b) + 0.2(1 + a), \\
b &= 0.7(3 + c) + 0.3a, \\
c &= c.
\end{align*}
\]
For the above Markov reward process, with Bellman:

\[
\begin{align*}
V(a) &= 0.8(2 + V(b)) + 0.2(1 + V(a)), \\
V(b) &= 0.7(3 + V(c)) + 0.3V(a), \\
V(c) &= 1.0(0 + V(c))
\end{align*}
\]

or, with abuse of notation,

\[
\begin{align*}
a &= 0.8(2 + b) + 0.2(1 + a), \\
b &= 0.7(3 + c) + 0.3a, \\
c &= c.
\end{align*}
\]

Solving this set of linear equations, using our knowledge that \( c = V(c) \) must be 0, yields \( a \approx 6.21, \ b \approx 3.96, \ c = 0, \).
Computing total reward

For the above Markov reward process, with Bellman:

\[
\begin{align*}
V(a) &= 0.8(2 + V(b)) + 0.2(1 + V(a)), \\
V(b) &= 0.7(3 + V(c)) + 0.3V(a), \\
V(c) &= 1.0(0 + V(c))
\end{align*}
\]

or, with abuse of notation,

\[
\begin{align*}
a &= 0.8(2 + b) + 0.2(1 + a), \\
b &= 0.7(3 + c) + 0.3a, \\
c &= c.
\end{align*}
\]

Solving this set of linear equations, using our knowledge that \( c = V(c) \) must be 0,
yields \( a \approx 6.21, \ b \approx 3.96, \ c = 0, \)

ergo \( V(a) \approx 6.21, \ V(b) \approx 3.96, \ V(c) = 0. \)
Utility pump

A utility pump is a MRP in which at least one recurrent state possesses a non-zero reward.

Example

\[ P = \begin{bmatrix}
0.2 & 0.8 \\
0.3 & 0.7 \\
\end{bmatrix}, \quad R = \begin{bmatrix}
1 & 2 \\
0 & 3 \\
0 & 0.01 \\
\end{bmatrix}. \]

Transition graph with rewards:
A **utility pump** is a MRP in which at least one recurrent state possesses a non-zero reward.
A utility pump is a MRP in which at least one recurrent state possesses a non-zero reward. Example

$$P = \begin{pmatrix} 0.2 & 0.8 \\ 0.3 & 0.7 \end{pmatrix}, \text{ with reward matrix } R = \begin{pmatrix} 1 & 2 \\ 0 & 3 \end{pmatrix}.$$
A utility pump is a MRP in which at least one recurrent state possesses a non-zero reward. Example

\[ P = \begin{pmatrix}
0.2 & 0.8 \\
0.3 & 0.7
\end{pmatrix}, \text{ with reward matrix } R = \begin{pmatrix}
1 & 2 \\
0 & 3
\end{pmatrix}. \]

Transition graph with rewards:

\[ p = 0.2, r = 1 \]

\[ p = 0.8, r = 2 \]

\[ p = 0.7, r = 3 \]

\[ r = 0.01 \]
Computing total reward for a utility pump

For the above Markov reward process, with Bellman:

\[
\begin{align*}
V(a) &= 0.8(2 + V(b)) + 0.2(1 + V(a)), \\
V(b) &= 0.7(3 + V(c)) + 0.3V(a), \\
V(c) &= 1.0(0.01 + V(c))
\end{align*}
\]
Computing total reward for a utility pump

For the above Markov reward process, with Bellman:

\[
\begin{align*}
V(a) &= 0.8(2 + V(b)) + 0.2(1 + V(a)), \\
V(b) &= 0.7(3 + V(c)) + 0.3V(a), \\
V(c) &= 1.0(0.01 + V(c))
\end{align*}
\]

or, with abuse of notation,

\[
\begin{align*}
a &= 0.8(2 + b) + 0.2(1 + a), \\
b &= 0.7(3 + c) + 0.3a, \\
c &= 0.01 + c.
\end{align*}
\]
Computing total reward for a utility pump

For the above Markov reward process, with Bellman:

\[
\begin{align*}
V(a) &= 0.8(2 + V(b)) + 0.2(1 + V(a)), \\
V(b) &= 0.7(3 + V(c)) + 0.3V(a), \\
V(c) &= 1.0(0.01 + V(c))
\end{align*}
\]

or, with abuse of notation,

\[
\begin{align*}
a &= 0.8(2 + b) + 0.2(1 + a), \\
b &= 0.7(3 + c) + 0.3a, \\
c &= 0.01 + c.
\end{align*}
\]

Solving this set of linear equations yields

\[
\begin{align*}
a &= \infty, \\
b &= \infty, \\
c &= \infty.
\end{align*}
\]

Ergo,

\[
\begin{align*}
V(a) &= \infty, \\
V(b) &= \infty, \\
V(c) &= \infty.
\end{align*}
\]
A **discount factor** is a real number \( 0 \leq \gamma \leq 1 \). It represents the probability to survive from one state to the next.
A **discount factor** is a real number $0 \leq \gamma \leq 1$. It represents the probability to survive from one state to the next.

Remarks:

1. The discount factor is a constant.
2. The discount factor does not impose a fixed cutoff bound. It is a probabilistic concept.

Example:

$$
\begin{pmatrix}
0 & 0 \\
.2 & .8 \\
.3 & .7 \\
.6 & .4 \\
.6 & .4 \\
\end{pmatrix}, \quad \gamma = \frac{1}{2}
$$

Probability to arrive in the third state after 5 steps equals

$$
\gamma^4 p(5)_{13} = \left(\frac{1}{2}\right)^4 \cdot \frac{2679}{5000} = \frac{2679}{80000}
$$
Discount factor

**Definition (Discount factor)**

A *discount factor* is a real number $0 \leq \gamma \leq 1$. It represents the probability to survive from one state to the next.

Remarks:

1. The discount factor is a constant.
Discount factor

Definition (Discount factor)

A discount factor is a real number $0 \leq \gamma \leq 1$. It represents the probability to survive from one state to the next.

Remarks:

1. The discount factor is a constant.
2. The discount factor does not impose a fixed cutoff bound. It is a probabilistic concept.
Definition (Discount factor)

A **discount factor** is a real number $0 \leq \gamma \leq 1$. It represents the probability to survive from one state to the next.

Remarks:

1. The discount factor is a constant.
2. The discount factor does not impose a fixed cutoff bound. It is a probabilistic concept.

Example:

$$P = \begin{pmatrix} 0 & 0.2 & 0.8 \\ 0.3 & 0.1 & 0.6 \\ 0.6 & 0.4 & 0 \end{pmatrix}, \quad \gamma = \frac{1}{2}. $$
**Discount factor**

**Definition (Discount factor)**

A *discount factor* is a real number $0 \leq \gamma \leq 1$. It represents the probability to survive from one state to the next.

**Remarks:**

1. The discount factor is a constant.
2. The discount factor does not impose a fixed cutoff bound. It is a probabilistic concept.

**Example:**

$$P = \begin{pmatrix} 0 & 0.2 & 0.8 \\ 0.3 & 0.1 & 0.6 \\ 0.6 & 0.4 & 0 \end{pmatrix}, \quad \gamma = \frac{1}{2}.$$  

Probability to arrive in third state after 5 steps equals

$$\gamma^4 p_{13}^{(5)}$$
**Discount factor**

**Definition (Discount factor)**

A *discount factor* is a real number $0 \leq \gamma \leq 1$. It represents the probability to survive from one state to the next.

Remarks:

1. The discount factor is a constant.
2. The discount factor does not impose a fixed cutoff bound. It is a probabilistic concept.

Example:

$$P = \begin{pmatrix} 0 & 0.2 & 0.8 \\ 0.3 & 0.1 & 0.6 \\ 0.6 & 0.4 & 0 \end{pmatrix}, \quad \gamma = \frac{1}{2}.$$  

Probability to arrive in third state after 5 steps equals

$$\gamma^4 p_{13}^{(5)} = \left( \frac{1}{2} \right)^4 \frac{2679}{5000}.$$
Discount factor

Definition (Discount factor)

A *discount factor* is a real number $0 \leq \gamma \leq 1$. It represents the probability to survive from one state to the next.

Remarks:

1. The discount factor is a constant.
2. The discount factor does not impose a fixed cutoff bound. It is a probabilistic concept.

Example:

$$P = \begin{pmatrix}
0 & 0.2 & 0.8 \\
0.3 & 0.1 & 0.6 \\
0.6 & 0.4 & 0
\end{pmatrix}, \quad \gamma = \frac{1}{2}.$$  

Probability to arrive in third state after 5 steps equals

$$\gamma^4 p_{13}^{(5)} = \left( \frac{1}{2} \right)^4 \frac{2679}{5000} = \frac{1}{16} \times \frac{2679}{5000} = \frac{2679}{80000}. $$
Theorem (Discounted reward)

Suppose a discount factor $0 \leq \gamma < 1$. Then

$$V(i) = \sum_{\text{all } j} p_{ij} [r_{ij} + \gamma V(j)],$$

for every state $i$. 

*Gerard Vreeswijk*  
MAIR: Markov models for multi-agent learning
Bellman equations for discounted reward

Theorem (Discounted reward)

Suppose a discount factor $0 \leq \gamma < 1$. Then

$$V(i) = \sum_{all \ j} p_{ij} [r_{ij} + \gamma V(j)],$$

for every state $i$.

Definition (Valuation)

A \textit{valuation} is a vector

$$v = \text{Def} \begin{pmatrix} v(1) \\ \vdots \\ v(n) \end{pmatrix} \in \mathbb{R}^n.$$
The Bellman operator

**Definition (Expected immediate reward)**

The *expected immediate reward* at state $i$ is the by $p_{ij}$-probabilities weighted average of the $r_{ij}$-rewards:

$$C(i) = \text{Def} \sum_{\text{all } j} p_{ij} r_{ij}. \tag{2}$$

The (vertical) vector of expected immediate rewards is denoted by $C$. 

---

**Gerard Vreeswijk**

**MAIR: Markov models for multi-agent learning**
The Bellman operator

**Definition (Expected immediate reward)**

The *expected immediate reward* at state \( i \) is the by \( p_{ij} \)-probabilities weighted average of the \( r_{ij} \)-rewards:

\[
C(i) = \text{Def} \sum_{all \ j} p_{ij} r_{ij}. \tag{2}
\]

The (vertical) vector of expected immediate rewards is denoted by \( C \).

**Definition (Bellman operator)**

\[
B_P : \mathbb{R}^n \rightarrow \mathbb{R}^n : v \mapsto C + \gamma P v. \tag{3}
\]
Theorem (*Fixed point*)

The value-vector, \( v^* \), of a discounted Markov reward process is a fixed point of the \( B_P \)-operator:

\[
B_P(v^*) = v^*.
\]
Theorem (Fixed point)

The value-vector, $v^*$, of a discounted Markov reward process is a fixed point of the $B_P$-operator:

$$B_P(v^*) = v^*.$$ 

Proof: Amounts to showing that $B_P$ is a so-called contraction. See notes.

Theorem (*Fixed point*)

The value-vector, \(v^*\), of a discounted reward Markov process is a fixed point of the \(B_P\)-operator:

\[ B_P(v^*) = v^*. \]
The value-vector, $v^*$, of a discounted reward Markov process is a fixed point of the $B_P$-operator:

$$B_P(v^*) = v^*.$$
The value-vector, $v^*$, of a discounted reward Markov process is a fixed point of the $B_P$-operator:

$$B_P(v^*) = v^*.$$ 

$v_0 \rightarrow B_P(v_0) \rightarrow B_PB_P(v_0) \rightarrow B_PB_PB_P(v_0) \rightarrow \cdots \rightarrow B_P^n v_0 \rightarrow \cdots \rightarrow v^*$

Theorem (Policy evaluation)

Let $v_0$ be an arbitrary starting vector. The value-vector, $v^*$, of a discounted Markov reward process can be found by simply iterating the $B_P$-operator:

$$v^* = \lim_{n \to \infty} B_P^n v_0.$$
Consider the following Markov reward process. Suppose a discount factor $\gamma = 0.5$.

Consider the following Markov reward process. Suppose a discount factor $\gamma = 0.5$. 

\begin{align*}
\{ & s : (r = 1, p = 0.4); \\
& a : (r = 3, p = 0.3); \\
& b : (r = 2, p = 0.5); \\
& \} &
\end{align*}
Consider the following Markov reward process. Suppose a discount factor $\gamma = 0.5$. 

$$
\begin{cases}
  s : & s (r = 1, p = 0.4) ; a (r = 4, p = 0.6),
\end{cases}
$$
Consider the following Markov reward process. Suppose a discount factor $\gamma = 0.5$.

\[
\begin{align*}
  s : & \quad s (r = 1, p = 0.4) ; \quad a (r = 4, p = 0.6), \\
  a : & \quad s (r = 3, p = 0.3) ; \quad a (r = 8, p = 0.7),
\end{align*}
\]
Consider the following Markov reward process. Suppose a discount factor $\gamma = 0.5$.

\begin{align*}
  s &: s (r = 1, p = 0.4) ; a (r = 4, p = 0.6), \\
  a &: s (r = 3, p = 0.3) ; a (r = 8, p = 0.7), \\
  b &: s (r = 2, p = 0.5) ; b (r = 5, p = 0.5).
\end{align*}
Example

Consider the following Markov reward process. Suppose a discount factor \( \gamma = 0.5 \).

\[
\begin{align*}
  s : & \quad s (r = 1, p = 0.4) ; a (r = 4, p = 0.6), \\
  a : & \quad s (r = 3, p = 0.3) ; a (r = 8, p = 0.7), \\
  b : & \quad s (r = 2, p = 0.5) ; b (r = 5, p = 0.5).
\end{align*}
\]
Consider the following Markov reward process. Suppose a discount factor $\gamma = 0.5$.

\[
\begin{align*}
  s &: s (r = 1, p = 0.4); a (r = 4, p = 0.6), \\
  a &: s (r = 3, p = 0.3); a (r = 8, p = 0.7), \\
  b &: s (r = 2, p = 0.5); b (r = 5, p = 0.5).
\end{align*}
\]

1. Give a state transition diagram, a probability transition matrix, $P$, and an immediate reward matrix, $R$. 

Gerard Vreeswijk  
MAIR: Markov models for multi-agent learning
Consider the following Markov reward process. Suppose a discount factor $\gamma = 0.5$.

\[
\begin{align*}
  s: & \quad s (r = 1, p = 0.4) \ ; \ a (r = 4, p = 0.6), \\
  a: & \quad s (r = 3, p = 0.3) \ ; \ a (r = 8, p = 0.7), \\
  b: & \quad s (r = 2, p = 0.5) \ ; \ b (r = 5, p = 0.5).
\end{align*}
\]

1. Give a state transition diagram, a probability transition matrix, $P$, and an immediate reward matrix, $R$.

2. Express the vector of optimal values $v^* = (s, a, b)$ as a solution of a system of linear equations.
Consider the following Markov reward process. Suppose a discount factor $\gamma = 0.5$.

$$\begin{align*}
  s : & \quad s (r = 1, p = 0.4) ; \quad a (r = 4, p = 0.6), \\
  a : & \quad s (r = 3, p = 0.3) ; \quad a (r = 8, p = 0.7), \\
  b : & \quad s (r = 2, p = 0.5) ; \quad b (r = 5, p = 0.5).
\end{align*}$$

1. Give a state transition diagram, a probability transition matrix, $P$, and an immediate reward matrix, $R$.

2. Express the vector of optimal values $v^* = (s, a, b)$ as a solution of a system of linear equations.

3. Perform value evaluation
**Example**

**Problem**

Consider the following Markov reward process. Suppose a discount factor $\gamma = 0.5$.

\[
\begin{align*}
  s : & \quad s (r = 1, p = 0.4) \; ; \; a (r = 4, p = 0.6), \\
  a : & \quad s (r = 3, p = 0.3) \; ; \; a (r = 8, p = 0.7), \\
  b : & \quad s (r = 2, p = 0.5) \; ; \; b (r = 5, p = 0.5).
\end{align*}
\]

1. Give a state transition diagram, a probability transition matrix, $P$, and an immediate reward matrix, $R$.
2. Express the vector of optimal values $v^* = (s, a, b)$ as a solution of a system of linear equations.
3. Perform **value evaluation** (in MDP a.k.a. **policy iteration**).
Consider the following Markov reward process. Suppose a discount factor $\gamma = 0.5$.

$$\begin{align*}
    s : & \quad s (r = 1, p = 0.4) \quad ; \quad a (r = 4, p = 0.6), \\
    a : & \quad s (r = 3, p = 0.3) \quad ; \quad a (r = 8, p = 0.7), \\
    b : & \quad s (r = 2, p = 0.5) \quad ; \quad b (r = 5, p = 0.5).
\end{align*}$$

1. Give a state transition diagram, a probability transition matrix, $P$, and an immediate reward matrix, $R$.
2. Express the vector of optimal values $v^* = (s, a, b)$ as a solution of a system of linear equations.
3. Perform value evaluation (in MDP a.k.a. policy iteration) with convergence tolerance $\epsilon = 0.01$. 

Gerard Vreeswijk

MAIR: Markov models for multi-agent learning
State transitions, transition matrix, reward matrix

\[ P = \begin{bmatrix} 0.4 & 0.6 & 0 \\ 0.3 & 0.7 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad R = \begin{bmatrix} 1 & 4 & 0 \\ 3 & 8 & 0 \\ 2 & 0 & 5 \end{bmatrix}. \]

The order of the nodes is \( s, a, b \). So \( r_{00} = r_{s,s} \), \( r_{01} = r_{s,a} \), etc., and \( p_{00} = p_{s,s} \), \( p_{01} = p_{s,a} \), etc.
State transitions, transition matrix, reward matrix

\[ P = \begin{pmatrix} 0.4 & 0.6 & 0 \\ 0.3 & 0.7 & 0 \\ 0.5 & 0 & 0.5 \end{pmatrix} \]

The order of the nodes is \( s, a, b \). So \( p_{00} = r_{ss}, p_{01} = r_{sa}, \) etc., and \( p_{00} = p_{ss}, p_{01} = p_{sa}, \) etc.
The order of the nodes is $b, a, s$. So $r_{00} = r_{bs}$, $r_{01} = r_{sa}$, etc., and $p_{00} = p_{bs}$, $p_{01} = p_{sa}$, etc.

$$P = \begin{pmatrix} 0.4 & 0.6 & 0 \\ 0.3 & 0.7 & 0 \\ 0.5 & 0 & 0.5 \end{pmatrix}, \quad R = \begin{pmatrix} 1 & 4 & 0 \\ 3 & 8 & 0 \\ 2 & 0 & 5 \end{pmatrix}.$$
The order of the nodes is $s, a, b$. So $r_{00} = r_{s,s}$, $r_{01} = r_{s,a}$, etc., and $p_{00} = p_{s,s}$, $p_{01} = p_{s,a}$, etc.
Express the vector of optimal values $v^* = (s, a, b)$ as a solution of a system of linear equations.
Express the vector of optimal values $v^* = (s, a, b)$ as a solution of a system of linear equations.

**Solution:** Use

$$V(i) = \sum_{\text{all } j} p_{ij} [r_{ij} + \gamma V(j)] :$$
Express the vector of optimal values \( \mathbf{v}^* = (s, a, b) \) as a solution of a system of linear equations.

**Solution:** Use

\[
V(i) = \sum_{\text{all } j} p_{ij} [r_{ij} + \gamma V(j)]
\]

\[
\begin{align*}
V(s) &= 0.4(1 + \gamma s) + 0.6(4 + \gamma a) \\
V(a) &= 0.3(3 + \gamma s) + 0.7(8 + \gamma a) \\
V(b) &= 0.5(2 + \gamma s) + 0.5(5 + \gamma b)
\end{align*}
\]

\[
\begin{align*}
\iff \\
V(s) &= 2.8 + \gamma (0.4s + 0.6a) \\
V(a) &= 6.5 + \gamma (0.3s + 0.7a) \\
V(b) &= 3.5 + \gamma (0.5s + 0.5b)
\end{align*}
\]
Express the vector of optimal values $v^* = (s, a, b)$ as a solution of a system of linear equations.

**Solution:** Use

$$V(i) = \sum_{\text{all } j} p_{ij} [r_{ij} + \gamma V(j)] :$$

$$V(s) = 0.4(1 + \gamma V(s)) + 0.6(4 + \gamma V(a)) ,$$

$$V(a) = 0.3(3 + \gamma V(s)) + 0.7(8 + \gamma V(a)) ,$$

$$V(b) = 0.5(2 + \gamma V(s)) + 0.5(5 + \gamma V(b)) .$$
Express the vector of optimal values $v^* = (s, a, b)$ as a solution of a system of linear equations.

**Solution:** Use

$$V(i) = \sum_{all \; j} p_{ij} [r_{ij} + \gamma V(j)] :$$

$$\begin{cases} 
V(s) = 0.4(1 + \gamma V(s)) + 0.6(4 + \gamma V(a)) , \\
V(a) = 0.3(3 + \gamma V(s)) + 0.7(8 + \gamma V(a)) , \\
V(b) = 0.5(2 + \gamma V(s)) + 0.5(5 + \gamma V(b)).
\end{cases}$$
Express the vector of optimal values $v^* = (s, a, b)$ as a solution of a system of linear equations.

Solution: Use

$$V(i) = \sum_{all \ j} p_{ij} [r_{ij} + \gamma V(j)] :$$

$$\begin{align*}
V(s) &= 0.4(1 + \gamma V(s)) + 0.6(4 + \gamma V(a)) , \\
V(a) &= 0.3(3 + \gamma V(s)) + 0.7(8 + \gamma V(a)) , \\
V(b) &= 0.5(2 + \gamma V(s)) + 0.5(5 + \gamma V(b)) .
\end{align*}$$
Express the vector of optimal values $v^* = (s, a, b)$ as a solution of a system of linear equations.

**Solution:** Use

$$V(i) = \sum_{\text{all } j} p_{ij} [r_{ij} + \gamma V(j)] :$$

\[ \begin{align*}
V(s) &= 0.4(1 + \gamma V(s)) + 0.6(4 + \gamma V(a)) , \\
V(a) &= 0.3(3 + \gamma V(s)) + 0.7(8 + \gamma V(a)) , \\
V(b) &= 0.5(2 + \gamma V(s)) + 0.5(5 + \gamma V(b)) .
\end{align*} \]
Express the vector of optimal values $v^* = (s, a, b)$ as a solution of a system of linear equations.

**Solution:** Use

$$V(i) = \sum_{\text{all } j} p_{ij} [r_{ij} + \gamma V(j)] :$$

$$\begin{cases} 
V(s) = 0.4(1 + \gamma V(s)) + 0.6(4 + \gamma V(a)) , \\
V(a) = 0.3(3 + \gamma V(s)) + 0.7(8 + \gamma V(a)) , \\
V(b) = 0.5(2 + \gamma V(s)) + 0.5(5 + \gamma V(b)) .
\end{cases}$$

Shorter (abusing notation):
Express the vector of optimal values $\mathbf{v}^* = (s, a, b)$ as a solution of a system of linear equations.

**Solution:** Use

$$V(i) = \sum_{\text{all } j} p_{ij} [r_{ij} + \gamma V(j)] :$$

$$
\begin{align*}
V(s) &= 0.4(1 + \gamma V(s)) + 0.6(4 + \gamma V(a)), \\
V(a) &= 0.3(3 + \gamma V(s)) + 0.7(8 + \gamma V(a)), \\
V(b) &= 0.5(2 + \gamma V(s)) + 0.5(5 + \gamma V(b)).
\end{align*}
$$

Shorter (abusing notation):

$$
\begin{align*}
s &= 0.4(1 + \gamma s) + 0.6(4 + \gamma a), \\
a &= 0.3(3 + \gamma s) + 0.7(8 + \gamma a), \\
b &= 0.5(2 + \gamma s) + 0.5(5 + \gamma b).
\end{align*}
$$
Express the vector of optimal values \( \mathbf{v}^* = (s, a, b) \) as a solution of a system of linear equations.

**Solution:** Use

\[
V(i) = \sum_{\text{all } j} p_{ij}[r_{ij} + \gamma V(j)] :
\]

\[
\begin{align*}
V(s) &= 0.4(1 + \gamma V(s)) + 0.6(4 + \gamma V(a)) , \\
V(a) &= 0.3(3 + \gamma V(s)) + 0.7(8 + \gamma V(a)) , \\
V(b) &= 0.5(2 + \gamma V(s)) + 0.5(5 + \gamma V(b)) .
\end{align*}
\]

Shorter (abusing notation):

\[
\begin{align*}
s &= 0.4(1 + \gamma s) + 0.6(4 + \gamma a), \quad & s &= 2.8 + \gamma (0.4s + 0.6a), \\
a &= 0.3(3 + \gamma s) + 0.7(8 + \gamma a), \quad & a &= 6.5 + \gamma (0.3s + 0.7a), \\
b &= 0.5(2 + \gamma s) + 0.5(5 + \gamma b). \quad & b &= 3.5 + \gamma (0.5s + 0.5b).
\end{align*}
\]
Perform *value evaluation* (in MDP a.k.a. *policy iteration*)

Solution:
To perform value evaluation, iterate $V_i = \sum_{j} p_{ij} r_{ij} + \gamma V_j$ on all nodes:

<table>
<thead>
<tr>
<th>Node</th>
<th>$v_0$</th>
<th>$v_1$</th>
<th>$v_2$</th>
<th>$v_{10}$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.00</td>
<td>2.80</td>
<td>5.31</td>
<td>7.93</td>
</tr>
<tr>
<td></td>
<td>0.00</td>
<td>6.50</td>
<td>9.19</td>
<td>11.83</td>
</tr>
<tr>
<td></td>
<td>0.00</td>
<td>5.08</td>
<td>5.08</td>
<td>7.31</td>
</tr>
</tbody>
</table>

Value convergence $v^* = (V(s), V(a), V(b)) = (7.93, 11.83, 7.31)$ if $\gamma = 0.5.$
Perform \textit{value evaluation} (in MDP a.k.a. \textit{policy iteration}) with convergence tolerance $\epsilon = 0.01$. 

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|}
\hline
State & Value & Value convergence \\
\hline
0 & 0 & \ldots \\
1 & 0.80 & \\
2 & 0.5 & \\
10 & 0.93 & \\
\hline
\end{tabular}
\end{table}

So $v^* = (V(s), V(a), V(b)) = (0.93, 0.83, 0.31)$ if $\gamma = 0.5$. 

Gerard Vreeswijk

MAIR: Markov models for multi-agent learning
Perform value evaluation (in MDP a.k.a. policy iteration) with convergence tolerance $\epsilon = 0.01$.

**Solution:** To perform value evaluation, iterate

\[
V(i) := \sum_{\text{all } j} p_{ij} [r_{ij} + \gamma V(j)]
\]
Value evaluation

Perform \textit{value evaluation} (in MDP a.k.a. \textit{policy iteration}) with convergence tolerance \( \epsilon = 0.01 \).

\textbf{Solution:} To perform value evaluation, iterate

\[ V(i) := \sum_{\text{all } j} p_{ij} \left[ r_{ij} + \gamma V(j) \right] \quad \text{on all nodes :} \]

\begin{tabular}{|c|c|c|c|}
\hline
\textbf{Value evaluation} & \textbf{s} & \textbf{a} & \textbf{b} \\
\hline
Start values & \( v_0 \) & 0.00 & 0.00 & 0.00 \\
& \( v_1 \) & 2.80 & 6.50 & 3.50 \\
& \( v_2 \) & 5.31 & 9.19 & 5.08 \\
& ... & ... & ... & ... \\
& \( v_{10} \) & 7.93 & 11.82 & 7.30 \\
Value convergence & \( v_{11} \) & 7.93 & 11.83 & 7.31 \\
\hline
\end{tabular}

So \( v^* = (V(s), V(a), V(b)) = (7.93, 11.83, 7.31) \) if \( \gamma = 0.5 \).
Perform **value evaluation** (in MDP a.k.a. **policy iteration**) with convergence tolerance $\epsilon = 0.01$.

**Solution:** To perform value evaluation, iterate

$$V(i) := \sum_{\text{all } j} p_{ij} [r_{ij} + \gamma V(j)]$$

on all nodes:

<table>
<thead>
<tr>
<th>Value evaluation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
<tr>
<td><strong>Start values</strong></td>
</tr>
<tr>
<td>$v_0$</td>
</tr>
<tr>
<td>0.00</td>
</tr>
<tr>
<td>2.80</td>
</tr>
<tr>
<td>5.31</td>
</tr>
<tr>
<td>$\vdots$</td>
</tr>
<tr>
<td>7.93</td>
</tr>
<tr>
<td><strong>Value convergence</strong></td>
</tr>
<tr>
<td>$v_{10}$</td>
</tr>
<tr>
<td>$v_{11}$</td>
</tr>
</tbody>
</table>

So $v^* = (V(s), V(a), V(b)) = (7.93, 11.83, 7.31)$ if $\gamma = 0.5$. 
Problem

Given is the following Markov reward process with discount factor \( \gamma = 0.5 \).

\[
\begin{align*}
  &s: \quad a (r = 3, p = 0.1) ;
  &b (r = -5, p = 0.9), \\
  &a: \quad s (r = 1, p = 0.3) ;
  &b (r = -9, p = 0.7), \\
  &b: \quad s (r = 2, p = 0.4) ;
  &a (r = 4, p = 0.6).
\end{align*}
\]
Problem

Given is the following Markov reward process with discount factor $\gamma = 0.5$.

\[
\begin{align*}
    s &: a (r = 3, p = 0.1) ; b (r = -5, p = 0.9), \\
    a &: s (r = 1, p = 0.3) ; b (r = -9, p = 0.7), \\
    b &: s (r = 2, p = 0.4) ; a (r = 4, p = 0.6).
\end{align*}
\]

1. Give a state transition diagram.
Problem

Given is the following Markov reward process with discount factor \( \gamma = 0.5 \).

\[
\begin{align*}
\text{s:} & \quad a (r = 3, p = 0.1) ; \quad b (r = -5, p = 0.9), \\
\text{a:} & \quad s (r = 1, p = 0.3) ; \quad b (r = -9, p = 0.7), \\
\text{b:} & \quad s (r = 2, p = 0.4) ; \quad a (r = 4, p = 0.6).
\end{align*}
\]

1. **Give a state transition diagram.**

2. **Give the probability transition matrix and the immediate reward matrix.**
Problem

*Given is the following Markov reward process with discount factor $\gamma = 0.5$.*

\[
\begin{align*}
    s : & \ a \ (r = 3, p = 0.1) \ ; \ b \ (r = -5, p = 0.9), \\
    a : & \ s \ (r = 1, p = 0.3) \ ; \ b \ (r = -9, p = 0.7), \\
    b : & \ s \ (r = 2, p = 0.4) \ ; \ a \ (r = 4, p = 0.6).
\end{align*}
\]

1. **Give a state transition diagram.**
2. **Give the probability transition matrix and the immediate reward matrix.**
3. **Express the vector of optimal values as a solution of a system of linear equations.**
Problem

Given is the following Markov reward process with discount factor $\gamma = 0.5$.

\[
\begin{align*}
    s : & \quad a (r = 3, p = 0.1) ; b (r = -5, p = 0.9), \\
    a : & \quad s (r = 1, p = 0.3) ; b (r = -9, p = 0.7), \\
    b : & \quad s (r = 2, p = 0.4) ; a (r = 4, p = 0.6).
\end{align*}
\]

1. Give a state transition diagram.
2. Give the probability transition matrix and the immediate reward matrix.
3. Express the vector of optimal values as a solution of a system of linear equations.
4. Perform value evaluation with convergence tolerance $\epsilon = 0.01$. 

Gerard Vreeswijk

MAIR: Markov models for multi-agent learning
Give a state transition diagram, including probabilities and rewards.
Give a state transition diagram, including probabilities and rewards.

Solution:

- From state $s$ to state $a$: $p = 0.3, r = 1$
- From state $a$ to state $b$: $p = 0.7, r = -9$
- From state $a$ to state $s$: $p = 0.1, r = 3$
- From state $s$ to state $b$: $p = 0.4, r = 2$
- From state $b$ to state $s$: $p = 0.9, r = -5$
- From state $b$ to state $a$: $p = 0.6, r = 4$
Give $P$ and $R$. 

Solution:

$$P = \begin{bmatrix}
0 & 0 & 1 & 0 \\
0 & 0.9 & 0 & 0.3 \\
0 & 0.7 & 0 & 0.4 \\
0 & 0.6 & 0 & 0.5 \\
\end{bmatrix}, \\
R = \begin{bmatrix}
3 & -5 \\
1 & -9 \\
2 & 4 \\
\end{bmatrix}.$$ 

Express the vector of optimal values $v^* = (s, a, \ldots, b)$ as a solution of a system of linear equations. 

Solution:

$$\begin{align*}
s &= 0.1(3 + \gamma a) + 0.9(-5 + \gamma b) \\
a &= 0.3(1 + \gamma s) + 0.7(-9 + \gamma b) \\
b &= 0.4(2 + \gamma s) + 0.6(4 + \gamma a). \\
\end{align*}$$
Give $P$ and $R$.

**Solution:**

$$ P = \begin{pmatrix} 0 & 0.1 & 0.9 \\ 0.3 & 0 & 0.7 \\ 0.4 & 0.6 & 0 \end{pmatrix}, \quad R = \begin{pmatrix} 3 & -5 \\ 1 & -9 \\ 2 & 4 \end{pmatrix}. $$
Give $P$ and $R$.

**Solution:**

$$P = \begin{pmatrix} 0 & 0.1 & 0.9 \\ 0.3 & 0 & 0.7 \\ 0.4 & 0.6 & 0 \end{pmatrix}, \quad R = \begin{pmatrix} 3 & -5 \\ 1 & -9 \\ 2 & 4 \end{pmatrix}.$$ 

Express the vector of optimal values $v^* = (s, a, \ldots, b)$ as a solution of a system of linear equations.
Give $P$ and $R$.

**Solution:**

\[
P = \begin{pmatrix}
0 & 0.1 & 0.9 \\
0.3 & 0 & 0.7 \\
0.4 & 0.6 & 0
\end{pmatrix}, \quad R = \begin{pmatrix}
3 & -5 \\
1 & -9 \\
2 & 4
\end{pmatrix}.
\]

Express the vector of optimal values $\nu^* = (s, a, \ldots, b)$ as a solution of a system of linear equations.

**Solution:**

\[
\begin{cases}
s = 0.1(3 + \gamma a) + 0.9(-5 + \gamma b), \\
a = 0.3(1 + \gamma s) + 0.7(-9 + \gamma b), \\
b = 0.4(2 + \gamma s) + 0.6(2 + \gamma a).
\end{cases}
\]
Perform value evaluation with convergence tolerance $\epsilon = 0.01$. 

Solution:

Iterate the Bellman equation on all nodes:

\[ v_0 = 0.01 \]
\[ v_1 = -4.20 - 6.00 + 3.20 = -3.00 \]
\[ v_2 = -3.06 - 5.51 + 0.56 = -4.05 \]
\[ \vdots \]
\[ v_8 = -4.35 - 6.52 + 0.38 = -4.35 \]

Value convergence \[ v_9 = -4.35 - 6.52 + 0.38 = -4.35 \]

We have used a convergence tolerance of $\epsilon = 0.01$. 

Gerard Vreeswijk
MAIR: Markov models for multi-agent learning
Perform value evaluation with convergence tolerance $\epsilon = 0.01$.

**Solution:** Iterate the Bellman equation on all nodes:

<table>
<thead>
<tr>
<th>Value evaluation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start values $v_0$</td>
</tr>
<tr>
<td>$v_0$</td>
</tr>
<tr>
<td>$v_1$</td>
</tr>
<tr>
<td>$v_2$</td>
</tr>
<tr>
<td>$v_9$</td>
</tr>
<tr>
<td>Value convergence $v_9$</td>
</tr>
</tbody>
</table>
Perform value evaluation with convergence tolerance $\epsilon = 0.01$.

**Solution:** Iterate the Bellman equation on all nodes:

<table>
<thead>
<tr>
<th>Value evaluation</th>
<th>s</th>
<th>a</th>
<th>b</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start values</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$v_0$</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>$v_1$</td>
<td>$-4.20$</td>
<td>$-6.00$</td>
<td>3.20</td>
</tr>
<tr>
<td>$v_2$</td>
<td>$-3.06$</td>
<td>$-5.51$</td>
<td>0.56</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$v_8$</td>
<td>$-4.35$</td>
<td>$-6.51$</td>
<td>0.38</td>
</tr>
<tr>
<td>Value convergence</td>
<td>$v_9$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$v_9$</td>
<td>$-4.35$</td>
<td>$-6.52$</td>
<td>0.38</td>
</tr>
</tbody>
</table>

We have used a convergence tolerance of $\epsilon = 0.01$. 
Problems to work on

- Lecture notes: 
  http://www.cs.uu.nl/docs/vakken/mmair/

- Section 3.7 (MP).
  For example: 
  Problem nr. 7 (garbage robot).

- Section 5.9 (MRP).
  For example: 
  Problem nr. 4.