GMDP toolbox: a Matlab library for solving Graph-based Markov Decision Processes

Marie-Josée Cros, Nathalie Peyrard, Régis Sabbadin

UR MIAT, INRA Toulouse, France

JFRB, Clermont Ferrand, 27-28 juin 2016
System management in ecology and agriculture

Management is complex because several entities in interaction must be managed together with a long term objective with uncertain environment.

Integrated management:
- variety choice,
- cultural practices,
- soil management,
...

Finding an optimal (or at least a good) policy to govern these large systems is still a challenge in practice.
Markov Decision Processes (MDP) [Puterman 94, Sigaud et al. 10] provide a classical framework for modelling and solving problems of sequential decision under uncertainty.

A discrete-time stationary MDP is defined by a 4-tuple $< S, A, p, r >$:

- $S$ is the state space,
- $A$ is the action space
- $p(s'|s,a)$ is the transition probability function
- $r(s,a)$ is the reward function

For a given policy $\delta$, defines a stationary Markov chain over $S$, with transitions $p_\delta(s'|s) = p(s'|s, \delta(s))$. 

![Diagram of MDP](image-url)
• **A policy** is defined as a function $\delta : S \rightarrow A$. Let $v_\delta(s)$ is the value of the policy $\delta$.

For the infinite-horizon discounted reward criterion:

$$v_\delta(s) = E\left[\sum_{t=0}^{+\infty} \gamma^t r(s^t, \delta(s^t)) | s^0 = s\right], \forall s \in S.$$  

• Policies that maximizes $v_\delta$ can be computed in polynomial time in $|S|$ and $|A|$ using Dynamic Programming (Policy Iteration, Value Iteration...).

To address larger problems, several frameworks Factored MDP (FMDP) have been proposed for factored state or/and action spaces and policies [Guestrin et al. 01, Kim et al. 02]
Graph-based MDP (GMDP) framework [Sabbadin et al. 12] → states, actions spaces factorisation (sites in interaction).

For a given policy, the dynamic model is a Dynamic Bayesian Network.

Neighborhood relationship

Corresponding DBN
Graph-based MDP (GMDP) framework [Sabbadin et al. 12] → states, actions spaces factorisation (sites in interaction).

For a given policy, the dynamic model is a Dynamic Bayesian Network.
Graph-based MDP (GMDP) framework [Sabbadin et al. 12] → states, actions spaces factorisation (sites in interaction).

For a given policy, the dynamic model is a Dynamic Bayesian Network.

Neighborhood relationship

Corresponding DBN
GMDP - Policy finding

A discrete-time GMDP is defined by a 5-tuple $< S, A, N, p, r >$:

- $S = S_1 \times \cdots \times S_n$
- $A = A_1 \times \cdots \times A_n$
- $N = \{N_i, \forall i = 1, \ldots, n\}$ with $N_i \subset \{1, \ldots, n\}$
- $p = \{p_i(s' | s_{N_i}, a_i) \forall i = 1, \ldots, n\} \rightarrow p(s' | s, a) = \prod_{i=1}^{n} p_i(s' | s_{N_i}, a_i)$
- $r = \{r_i(s_{N_i}, a_i) \forall i = 1, \ldots, n\} \rightarrow r(s, a) = \sum_{i=1}^{n} r_i(s_{N_i}, a_i)$

Only local policies are considered: $\delta = (\delta_1, \ldots, \delta_n)$ where $\delta_i : S_{N_i} \rightarrow A_i$

Two algorithms, providing local policies by approximate resolution of a GMDP, have been defined by [Sabbadin et al 12].

- **MF-API**: Mean Field Approximate Policy Iteration
  Exploits the structure of the neighborhood relations of the GMDP and computes a *Mean-Field approximation* of the value function of a policy.

- **ALP**: Approximate Linear Programming
  Derived from the general class of ALP algorithms, for large size MDP. They usually find empirically good local policies.
GMDPtoolbox - An help to apprehend the framework and support studies

1. Describe the problem
   ▶ 3 functions available to define simple problems

2. Find a good policy
   ▶ 2 functions for MF-API and ALP
   ▶ 1 function to translate small problem in MDP format for MDPtoolbox\(^1\) [Chadès et al. 14]

3. Interpret the policy
   ▶ 5 functions to apprehend and visualize policies

4. Analyse policies
   ▶ 3 functions to evaluate the value function
   ▶ 4 functions to simulate policies

\(^1\) MDP toolbox: http://inra.fr/mia/T/MDPtoolbox
• 3 crop fields can be in two states: uninfected (1 ☑️) or infected (2 ❌).
• Each field is susceptible to contamination by a pathogen.
• Two actions for a field: a normal (1 ✗) or adapted (2 ✶) cultural mode.
• When a field is contaminated, the yield decreases.
• The problem is to optimize a long-term policy in terms of expected yield.
Quick start - Computing a policy

1. Describe the problem

>>> GMDP = gmdp_example_epidemio();

2. Find a good policy

>>> policyloc = gmdp_linear_programming(GMDP, discount);
>>> actions_repartition_state_site = gmdp_analyze_policy_neighbor(GMDP, policyloc);

Decision rule

IF field is infected THEN apply an adapted cultural mode
ELSE apply a normal cultural mode
Quick start - Evaluating a policy

3 Simulate policy application

```matlab
[ sim_state, sim_reward ] = gmdp_simulate_policy(GMDP, policyloc);
value_evolution1 = gmdp_eval_policy_value(discount, sim_reward);
state_time = gmdp_eval_policy_state_time(GMDP, sim_state);
```

Evaluate pest management

- Each site is non contaminated about 65% of the time.
- This could be compared with other policies.
Management of phoma pest on canola crop

A grid of 100 fields, for each field 11 states, $8(2^3)$ actions, 5 neighbors. Find a good strategy and simulate 4 policies from a given initial state.

![Expected rewards of policies](image)

Expected cumulative rewards

![mean state repartition in wheat state](image)

**Results**

- Put on the spotligth a new interesting strategy.
- Put in evidence contrasted long term effect of policies.
Conclusion

GMDPtoolbox: A free library\textsuperscript{2} that provides:

- a framework to set the problem,
- algorithms to find a good policy,
- tools to explore and analyze policies,

Perspectives

- A new version soon released providing more adapted analysis functions, and GNU Octave compatibility.
- An other solving algorithm [Cheng et al. 13] based on approximate Value Iteration which approximate the value function with a Belief Propagation algorithm.

Application

- GMDP framework yet used for: plant disease management [Peyrard et al. 07], human disease management [Choisy et al. 07], forest management [Forsell et al. 11] and invasive pest control [Nicol et al. 15].
- With current environment evolution (biodiversity decrease, climate change...), GMDPtoolbox could help adapting management in agriculture, epidemics control or ecology.

\textsuperscript{2}GMDPtoolbox: http://inra.fr/mia/T/GMDPtoolbox
MDPtoolbox: a multi-platform toolbox to solve stochastic dynamic programming problems.
Ecography, 37(9):916–920.

Variational planning for graph-based MDPs.

A probabilistic decision framework to optimize the dynamics of a network evolution: application to the control of childhood diseases.
In ECCS: European Conference on Complex Systems.

Management of the risk of wind damage in forestry: a graph-based Markov decision process approach.

Multiagent planning with factored MDPs.
In Proceedings of NIPS’01, pages 1523–1530.

Solving factored MDPs with large action space using algebraic decision diagrams.
In Proceedings of the 7th Pacific Rim International Conference on Artificial Intelligence, pages 80–89.

Scalable multiagent planning using probabilistic inference.
In Proceedings of the 22th International Joint Conference on Artificial Intelligence.


