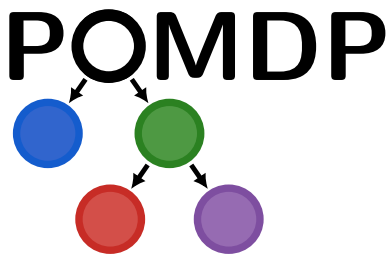
Julia Academy: POMDPs.jl

Decision Making Under Uncertainty

Robert Moss
CS PhD Student, Stanford University

mossr@stanford.edu
September, 2021



What is this course?

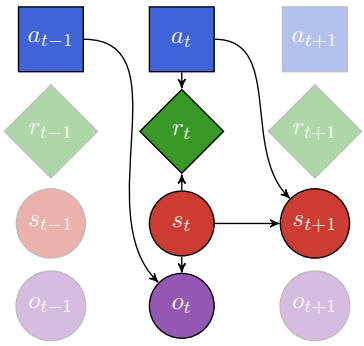


Figure: POMDP Sequence.

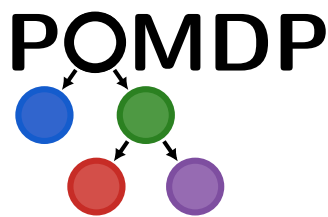
- A peek into the `POMDPs.jl` ecosystem of Julia packages
- "But what *are* POMDPs?"
  - POMDPs are a *problem formulation* that enable optimal[1] sequential decisions to be made in uncertain environments.
- Teaching *by example* using interactive `Pluto.jl` notebooks
  - No prior knowledge of MDPs/POMDPs necessary—all are welcome!
  - Can also be used as a refresher on *decision making under uncertainty*.
  - Target audience is wide, but familiarity with Julia is helpful.

[1] or *approximately* optimal.

Topics covered in this course

All topics highlight packages that adhere to the `POMDPs.jl` interface.

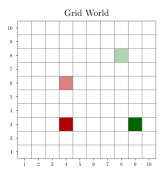
- **Sequential Decision Making**
  - *Markov decision processes* (MDPs)
  - *Partially observable Markov decision processes* (POMDPs)
- **Solution Methods**: Algorithms to solve MDPs/POMDPs
  - *Online* and *offline* solvers
  - *Value function approximation*
- **Simulations**
- **State Estimation using Particle Filters**
- **Reinforcement Learning**
- **Deep Reinforcement Learning**
- **Imitation Learning**
- **Black-Box Validation**

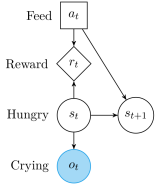


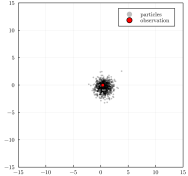
Example problems covered in this course

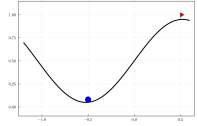
Common problems in the literature are used as running examples.

- (MDP) **Grid World**: Agent moving around a grid world, looking for rewards.
- (POMDP) **Crying Baby**: When to feed a baby, based on crying observations.
- (MDP) **1D Random Walk**: Agent moves around the number line.
- (POMDP) **2D Random Walk**: Estimating state of a moving agent based on observations.
- (MDP) **Mountain Car**: Reach a goal up a hill, starting in a valley.
- (MDP) **Swinging Pendulum**: Balance a swinging pendulum upright.









POMDPs.jl PACKAGE ECOSYSTEM

The POMDPs.jl package itself contains the interface to define problem definitions.

Other packages provide supporting tools that contain most of the functionality:[1]

- QuickPOMDPs.jl
- POMDPModelTools.jl
- POMDPPolicies.jl
- POMDPSimulators.jl
- POMDPModels.jl
- POMDPGallery.jl
- BeliefUpdaters.jl
- ParticleFilters.jl
- POMDPModelChecking.jl
- POMDPStressTesting.jl
- DiscreteValueIteration.jl
- LocalApproximationValueIteration.jl
- GlobalApproximationValueIteration.jl
- MCTS.jl
- TabularTDLearning.jl
- DeepQLearning.jl
- Crux.jl
- QMDP.jl
- FIB.jl
- BeliefGridValueIteration.jl
- SARSOP.jl
- BasicPOMCP.jl
- ARDESPOT.jl
- MCVI.jl
- POMDPSolve.jl
- IncrementalPruning.jl
- POMCPOW.jl
- AEMS.jl
- PointBasedValueIteration.jl

[1] *Key*: Tools, Extensions, MDP solvers, POMDP solvers.

Other resources

There are many *excellent* resources on MDPs/POMDPs and reinforcement learning:

- ***Algorithms for Decision Making*, Kochenderfer, Wheeler, & Wray**
  (https://algorithmsbook.com/)
- ***Reinforcement Learning: An Introduction*, Sutton & Barto**
  (http://incompleteideas.net/book/the-book.html)
- ***POMDPs.jl: A Framework for Sequential Decision Making under Uncertainty*, Egorov, Sunberg, et al., Journal of Machine Learning Research, 2017**
  (https://www.jmlr.org/papers/volume18/16-300/16-300.pdf)
- **Introduction to Reinforcement Learning with David Silver**
  (https://deepmind.com/learning-resources/-introduction-reinforcement-learning-david-silver)

Lecture breakdown

Each lecture has an associated `Pluto` notebook detailing the material.

1. **MDPs: Markov Decision Processes**
   - Includes: *planning, reinforcement learning, online/offline solvers, simulations*
2. **POMDPs: Partially Observable Markov Decision Processes**
3. **State Estimation using Particle Filtering**
4. **Approximate Methods for Continuous Spaces**
5. **Deep Reinforcement Learning**
6. **Imitation Learning: Learn from Demonstrations**
7. **Black-Box Validation**