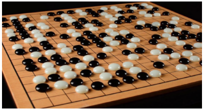
Reinforcement Learning in practice

Philippe Preux
philippe.preux@inria.fr
Scool, Lille, France

Jornadas Científicas Inria Chile

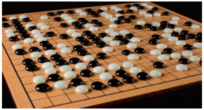
Reinforcement learning



Reinforcement learning

Ca. 1992.

Ca. 2013.



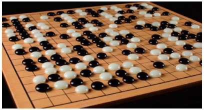
Ca. 2017.

Ca. 2022.

Reinforcement learning

Ca. 1992. Self-play learning, 1.5 10^6 games.
Trained 2 weeks on a few dozens Mb, 100 MHz CPU.
1 layer of 80 hidden neurons.

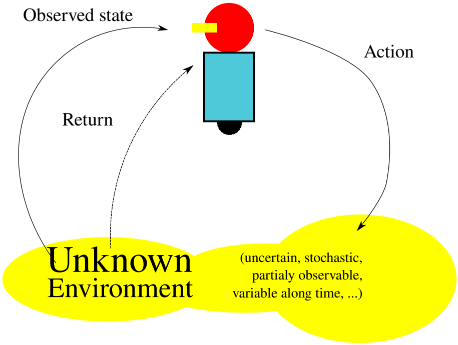
Ca. 2013.



Ca. 2017.

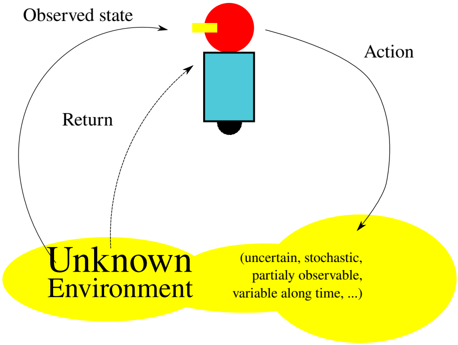
Ca. 2022.

The Reinforcement Learning Problem



Learn an optimal behavior.

The Reinforcement Learning Problem



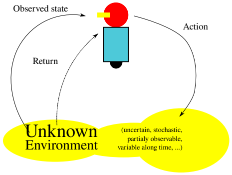
Learn an optimal behavior.

Unique feature of RL: by interacting with the environment.

The Reinforcement Learning Problem

Origins: theory of behavior adaptation.

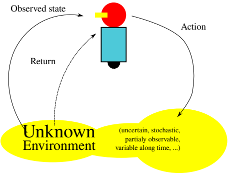
(Thorndike (1898), Skinner, and many others).



The Reinforcement Learning Problem

Origins: theory of behavior adaptation.

(Thorndike (1898), Skinner, and many others).



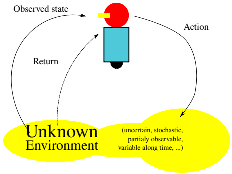
- “Operant conditioning”:
as species are selected along generations by their ability to survive, behaviors are selected by their consequences:

a behavior followed by “good” consequences is reinforced: its probability of being emitted increases.

The Reinforcement Learning Problem

Origins: theory of behavior adaptation.

(Thorndike (1898), Skinner, and many others).



- "Operant conditioning":
as species are selected along generations by their ability to survive, behaviors are selected by their consequences:

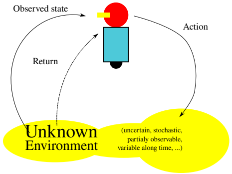
a behavior followed by "good" consequences is reinforced: its probability of being emitted increases.

- Until we are bored. An animal learns when it is surprised [Samuel 1959].

The Reinforcement Learning Problem

Origins: theory of behavior adaptation.

(Thorndike (1898), Skinner, and many others).



- "Operant conditioning": as species are selected along generations by their ability to survive, behaviors are selected by their consequences:

  a behavior followed by "good" consequences is reinforced: its probability of being emitted increases.

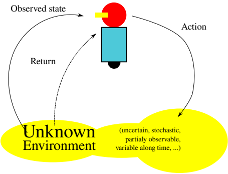
- Until we are bored. An animal learns when it is surprised [Samuel 1959].

- That's the basic idea of RL: the agent learns when it is surprised.

The Reinforcement Learning Problem

Origins: theory of behavior adaptation.

(Thorndike (1898), Skinner, and many others).



- "Operant conditioning":
as species are selected along generations by their ability to survive, behaviors are selected by their consequences:

a behavior followed by "good" consequences is reinforced: its probability of being emitted increases.

- Until we are bored. An animal learns when it is surprised [Samuel 1959].
- That's the basic idea of RL: the agent learns when it is surprised.
- First RL algorithms:
TD-Learning [Sutton, 1988], Q-Learning [Watkins, 1989], REINFORCE [Williams, 1992].

The Reinforcement Learning Problem

RL seems the answer to many problems.

The Reinforcement Learning Problem

RL seems the answer to many problems. But,

1. RL is (very) long to train.

The Reinforcement Learning Problem

RL seems the answer to many problems. But,

1. RL is (very) long to train.
2. No established metholodgy: artcraft.

The Reinforcement Learning Problem

RL seems the answer to many problems. But,

1. RL is (very) long to train.
2. No established metholodgy: artcraft.
3. Well-thought model of the task is crucial.

The Reinforcement Learning Problem

RL seems the answer to many problems. But,

1. RL is (very) long to train.
2. No established metholodgy: artcraft.
3. Well-thought model of the task is crucial.
4. Methodological issues: brittelness of experimental results.

Reconciling RL with more real applications

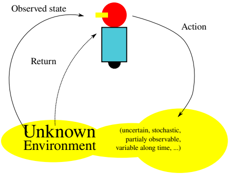
Reconciling RL with more real applications

The model

RL is based on Markov decision problems

- state space
- action space
- unknown dynamics
- return function: user defined to some extent.
- objective function

Solution: a *policy* that maps action to state.



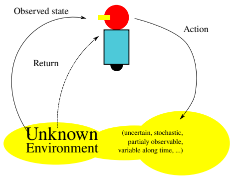
Reconciling RL with more real applications

The model

RL is based on Markov decision problems

- state space
- action space
- unknown dynamics
- return function: user defined to some extent.
- objective function

Solution: a *policy* that maps action to state.



Solving an RL problem is strictly equivalent to solving an LP:
$\max \mathbf{c}'\mathbf{x}, s.t. \mathbf{A}\mathbf{x} \leq \mathbf{b}$, and $\mathbf{x} \geq \mathbf{0}$,
under uncertainty: $\mathbf{A}$, $\mathbf{b}$, $\mathbf{c}$ unknown, often extremely large, or ∞.

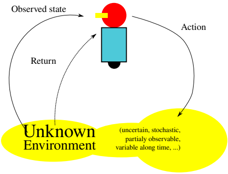
Reconciling RL with more real applications

The model

RL is based on Markov decision problems

- state space
- action space
- unknown dynamics
- return function: user defined to some extent.
- objective function

Solution: a *policy* that maps action to state.



Solving an RL problem is strictly equivalent to solving an LP:
$\max \mathbf{c}'\mathbf{x}, s.t. \mathbf{A}\mathbf{x} \leq \mathbf{b}$, and $\mathbf{x} \geq \mathbf{0}$,
under uncertainty: $\mathbf{A}$, $\mathbf{b}$, $\mathbf{c}$ unknown, often extremely large, or ∞.

$\mathbf{A}$, $\mathbf{b}$, $\mathbf{c}$ are related to the dynamics of the environment
$\rightsquigarrow$ sampling the environment $\rightarrow$ estimates.

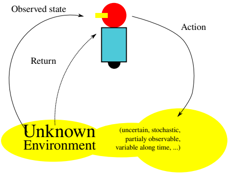
Reconciling RL with more real applications

The model

RL is based on Markov decision problems

- state space
- action space
- unknown dynamics
- return function: user defined to some extent.
- objective function

Solution: a *policy* that maps action to state.



- Basic principle: to learn, interact with the environment, and balance exploration and exploitation in a careful way.
- Basic idea: maintain expectations of the consequences of actions (their value). When the consequences do not match the expectations, update it.

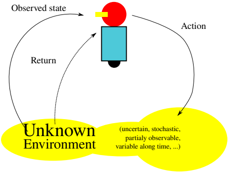
Reconciling RL with more real applications

The model

RL is based on Markov decision problems

- state space
- action space
- unknown dynamics
- return function: user defined to some extent.
- objective function

Solution: a *policy* that maps action to state.



The state must summarize the whole history of the agent.
The state determines the best action to perform.
The objective is to learn this mapping.

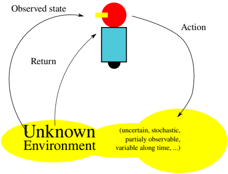
Reconciling RL with more real applications

The model

RL is based on Markov decision problems

- state space
- action space
- unknown dynamics
- return function: user defined to some extent.
- objective function

Solution: a *policy* that maps action to state.



The design of the return is crucial.

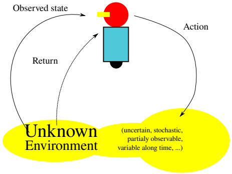
Reconciling RL with more real applications

The model

RL is based on Markov decision problems

- state space
- action space
- unknown dynamics
- return function: user defined to some extent.
- objective function

Solution: a *policy* that maps action to state.

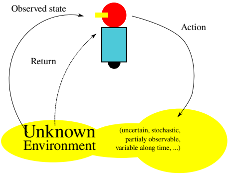


The consequences of actions are often diluted in the future $\rightsquigarrow$ credit assignment problem.

RL is based on Markov decision problems

- state space
- action space
- unknown dynamics
- return function: user defined to some extent.
- objective function

Solution: a *policy* that maps action to state.



Deterministic tasks are much simpler to solve than stochastic ones.

Reconciling RL with more real applications

Reconciling RL with more real applications

- We need an accurate simulator of the environment.

Reconciling RL with more real applications

- We need an accurate simulator of the environment.
- Many simulators out there in many scientific domains.

Reconciling RL with more real applications

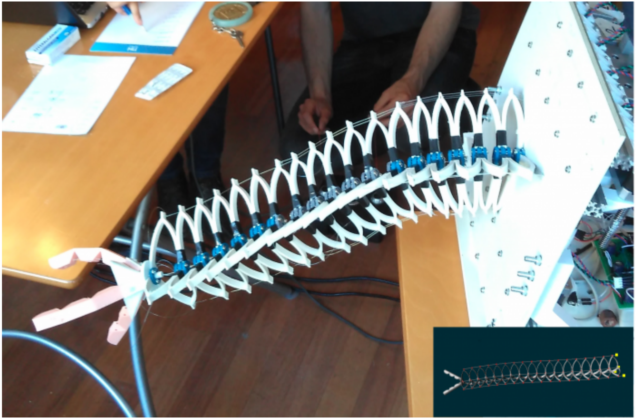
- We need an accurate simulator of the environment.
- Many simulators out there in many scientific domains.

- Often not designed to be used in an interaction loop:
  1) the user describes the simulation to be done in a configuration file,
  2) run the simulator on this configuration file, and
  3) the simulator outputs a result file.

Reconciling RL with more real applications

- We need an accurate simulator of the environment.
- Many simulators out there in many scientific domains.
- Often not designed to be used in an interaction loop:
  1) the user describes the simulation to be done in a configuration file,
  2) run the simulator on this configuration file, and
  3) the simulator outputs a result file.
- An RL compliant simulator requires sophisticated modifications.
  The simulator is usually a complex piece of software, resulting from years, decades sometimes, of work, usually written in either Fortran, or C, or C++.

RL meets soft robots



In collaboration with Ch. Duriez, Defrost @ Inria-Lille.

RL meets soft robots

- *Soft ($=$ deformable)* robot vs. rigid robot.

RL meets soft robots

- *Soft (= deformable)* robot vs. rigid robot.
- Infinite degrees of freedom.

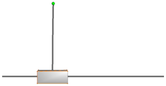
RL meets soft robots

- *Soft (= deformable)* robot vs. rigid robot.
- Infinite degrees of freedom.
- Simulation is much more complex than for rigid robots.

RL meets soft robots

- *Soft (= deformable)* robot vs. rigid robot.
- Infinite degrees of freedom.
- Simulation is much more complex than for rigid robots.
- Example: the famous cartpole turns into a cartStem.

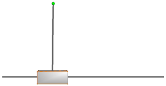
Rewarded when its tip is in the instable equilibrium position

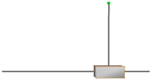


RL meets soft robots

- *Soft (= deformable)* robot vs. rigid robot.
- Infinite degrees of freedom.
- Simulation is much more complex than for rigid robots.
- Example: the famous cartpole turns into a cartStem.

  Rewarded when its tip is in the instable equilibrium position

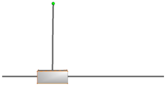
  

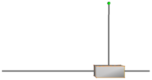
  can move to the right: 

RL meets soft robots

- *Soft (= deformable)* robot vs. rigid robot.
- Infinite degrees of freedom.
- Simulation is much more complex than for rigid robots.
- Example: the famous cartpole turns into a cartStem.

Rewarded when its tip is in the instable equilibrium position



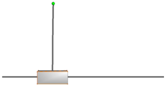
can move to the right: 

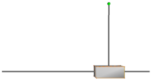
and decelerate: 

RL meets soft robots

- *Soft* (= *deformable*) robot vs. rigid robot.
- Infinite degrees of freedom.
- Simulation is much more complex than for rigid robots.
- Example: the famous cartpole turns into a cartStem.

  Rewarded when its tip is in the instable equilibrium position

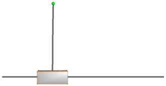
  can move to the right: 

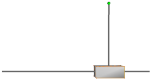
  and decelerate: 
- An action has a complex non instantaneous outcome,

RL meets soft robots

- *Soft (= deformable)* robot vs. rigid robot.
- Infinite degrees of freedom.
- Simulation is much more complex than for rigid robots.
- Example: the famous cartpole turns into a cartStem.

  Rewarded when its tip is in the instable equilibrium position

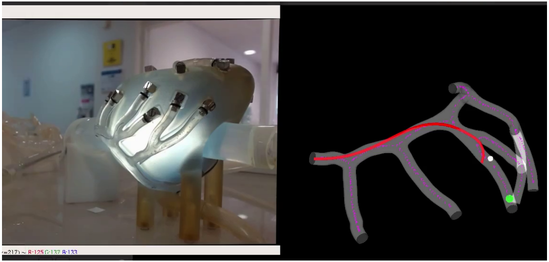
  can move to the right: 

  and decelerate: 
- An action has a complex non instantaneous outcome,
- There are many ways to design the MDP model.

RL meets soft robots

Playing with soft robots

- Some are used for serious matters:



Application to guiding a catheter in coronary arteries.

Controlled by model-based RL.

P. Schegg *et al.*, Automated planning for robotic guidewire navigation in the coronary arteries, *Proc. IEEE 5th International Conference on Soft Robotics*, April 2022, `hal-03778352`.

P. Schegg, *Autonomous Guidewire Navigation for Robotic Percutaneous Coronary Interventions*, Ph.D. dissertation, defended May 2022, Université de Lille, not publicly available.

P. Schegg *et al.*, SofaGym: An Open Platform for Reinforcement Learning Based on Soft Robot Simulations, Soft Robotics, **10**(2), Apr. 2023.

Learning to manage a crop field

In collaboration with Cirad, CGIAR, and BAU (India).

Learning to manage a crop field

- End goal: be able to recommend what to do next in the field to the farmer.
  In order to fullfill some objective: make money out of his harvest, feed his family, his animals, buy fertilizers, tools, *etc.*, while avoiding pollution, soil destruction, *etc.*
  Target: small farm holders in developing countries.

Learning to manage a crop field

- End goal: be able to recommend what to do next in the field to the farmer.
  In order to fullfill some objective: make money out of his harvest, feed his family, his animals, buy fertilizers, tools, *etc.*, while avoiding pollution, soil destruction, *etc.*
  Target: small farm holders in developing countries.

- Not enough training data available.

Learning to manage a crop field

- End goal: be able to recommend what to do next in the field to the farmer.
  In order to fullfill some objective: make money out of his harvest, feed his family, his animals, buy fertilizers, tools, *etc.*, while avoiding pollution, soil destruction, *etc.*
  Target: small farm holders in developing countries.

- Not enough training data available.
- Exploration? Very long to make field experiments to collect data.

Learning to manage a crop field

- End goal: be able to recommend what to do next in the field to the farmer.
  In order to fullfill some objective: make money out of his harvest, feed his family, his animals, buy fertilizers, tools, *etc.*, while avoiding pollution, soil destruction, *etc.*
  Target: small farm holders in developing countries.

- Not enough training data available.
- Exploration? Very long to make field experiments to collect data.
- Simulators exist.

Learning to manage a crop field

- Crop management has been formulated as a Markov Decision Problem since the 1950's.

Learning to manage a crop field

- Crop management has been formulated as a Markov Decision Problem since the 1950's.
- There exists very accurate crop growth simulators.

Learning to manage a crop field

- Crop management has been formulated as a Markov Decision Problem since the 1950's.
- There exists very accurate crop growth simulators.
- E.g. the "Decision Support System for Agrotechnology Transfer" (DSSAT):

Learning to manage a crop field

- Crop management has been formulated as a Markov Decision Problem since the 1950's.
- There exists very accurate crop growth simulators.
- E.g. the "Decision Support System for Agrotechnology Transfer" (DSSAT):
  - Developed for more than 30 years now, U. Florida, Gainsville.

Learning to manage a crop field

- Crop management has been formulated as a Markov Decision Problem since the 1950's.
- There exists very accurate crop growth simulators.
- E.g. the "Decision Support System for Agrotechnology Transfer" (DSSAT):
  - Developed for more than 30 years now, U. Florida, Gainsville.
  - 300 kloc of Fortran.

Learning to manage a crop field

- Crop management has been formulated as a Markov Decision Problem since the 1950's.
- There exists very accurate crop growth simulators.
- E.g. the "Decision Support System for Agrotechnology Transfer" (DSSAT):
  - Developed for more than 30 years now, U. Florida, Gainsville.
  - 300 kloc of Fortran.
  - Mechanistic crop model.

Learning to manage a crop field

- Crop management has been formulated as a Markov Decision Problem since the 1950's.
- There exists very accurate crop growth simulators.
- E.g. the "Decision Support System for Agrotechnology Transfer" (DSSAT):
  - Developed for more than 30 years now, U. Florida, Gainsville.
  - 300 kloc of Fortran.
  - Mechanistic crop model.
  - Simulates very accurately the growth of a plant based on the properties of the soil, the cultivar, the weather conditions, initial soil conditions (residue from previous year), ... interactions between the soil properties with roots then growth of the plant (PDE integration over time).
    + the actions made in the field: irrigation, fertilization, tillage, ... on a daily basis.

Learning to manage a crop field

From DSSAT to gym-DSSAT

- Crop management has been formulated as a Markov Decision Problem since the 1950's.
- There exists very accurate crop growth simulators.
- DSSAT.

Learning to manage a crop field

From DSSAT to gym-DSSAT

- Crop management has been formulated as a Markov Decision Problem since the 1950's.
- There exists very accurate crop growth simulators.
- DSSAT.

- gym interface to DSSAT.

Learning to manage a crop field

From DSSAT to gym-DSSAT

- Crop management has been formulated as a Markov Decision Problem since the 1950's.
- There exists very accurate crop growth simulators.
- DSSAT.

- gym interface to DSSAT.
- Freely available on `https://gitlab.inria.fr/rgautron/gym_dssat_pdi`.

- Problem: based on a maize field experiment [Morris *et al.*, 1982]

  How to manage irrigation or fertilization to maximize the yield of a certain cultivar of maize in a certain soil in certain weather conditions?

Learning to manage a crop field

Experiments

- Problem: based on a maize field experiment [Morris *et al.*, 1982]

  How to manage irrigation or fertilization to maximize the yield of a certain cultivar of maize in a certain soil in certain weather conditions?

- Action set: irrigate (volume), fertilize (amount), do-nothing

Learning to manage a crop field

Experiments

- Problem: based on a maize field experiment [Morris *et al.*, 1982]

  How to manage irrigation or fertilization to maximize the yield of a certain cultivar of maize in a certain soil in certain weather conditions?
- Action set: irrigate (volume), fertilize (amount), do-nothing
- Crop growth depends on action, soil nature, weather condition (stochastic), *etc.*

Learning to manage a crop field

Experiments

- Problem: based on a maize field experiment [Morris *et al.*, 1982]

  How to manage irrigation or fertilization to maximize the yield of a certain cultivar of maize in a certain soil in certain weather conditions?

- Action set: irrigate (volume), fertilize (amount), do-nothing
- Crop growth depends on action, soil nature, weather condition (stochastic), *etc.*
- Observation = Collection of measurements amenable to a real farmer

  $\rightsquigarrow$ partial observability.

Learning to manage a crop field

- Task: How to manage irrigation or fertilization to maximize the yield of a certain cultivar of maize in a certain soil in certain weather conditions?
  Let's focus on the fertilization problem.

Learning to manage a crop field

- Task: How to manage irrigation or fertilization to maximize the yield of a certain cultivar of maize in a certain soil in certain weather conditions?
  Let's focus on the fertilization problem.
- We look for a policy: $\forall$ day: (day, amount of fertilizer)
  which is efficient and effective:
  trades-off yield vs. pollution and cost.

Learning to manage a crop field

- Task: How to manage irrigation or fertilization to maximize the yield of a certain cultivar of maize in a certain soil in certain weather conditions?
  Let's focus on the fertilization problem.
- We look for a policy: $\forall$ day: (day, amount of fertilizer) which is efficient and effective: trades-off yield vs. pollution and cost.
- The daily return is defined by:

$$r(day) = \text{Nitrogen uptake}(day, day+1) - 0.5 \times \text{fertilizer quantity}(day)$$

Learning to manage a crop field

- Task: How to manage irrigation or fertilization to maximize the yield of a certain cultivar of maize in a certain soil in certain weather conditions?
  Let's focus on the fertilization problem.
- We look for a policy: $\forall$ day: (day, amount of fertilizer)
  which is efficient and effective:
  trades-off yield vs. pollution and cost.
- The daily return is defined by:

$$r(day) = \text{Nitrogen uptake}(day, day+1) - 0.5 \times \text{fertilizer quantity}(day)$$

- The goal is to maximize $\sum_{day=0}^{day=\text{harvest}} r(day)$.

Learning to manage a crop field

- The observation data:

	definition
istage	DSSAT maize growing stage
vstage	vegetative growth stage (number of leaves)
topwt	above the ground population biomass (kg/ha)
grnwt	grain weight dry matter (kg/ha)
swfac	index of plant water stress (unitless)
nstres	index of plant nitrogen stress (unitless)
xlai	plant population leaf area index (m^2 leaf/m^2 soil)
dtt	growing degree days for current day (°C/day)
dap	days after planting (day)
cumsumfert	cumulative nitrogen fertilizer applications (kg/ha)
rain	rainfall for the current day (L/m^2/day)
ep	actual plant transpiration rate (L/m^2/day)

- Action set: daily nitrogen fertilization amount $\in [0, 200]$ (kg/ha)

Learning to manage a crop field

Some results (1/3)

We compare:

1. A null policy which does not fertilize,
2. An expert policy used in the original 1982 field experiment,

DAP	quantity (kg N/ha)
40	27
45	35
80	54

3. A policy learned by RL (basic untuned PPO).

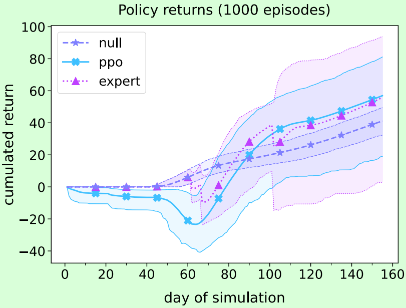
Remarks:

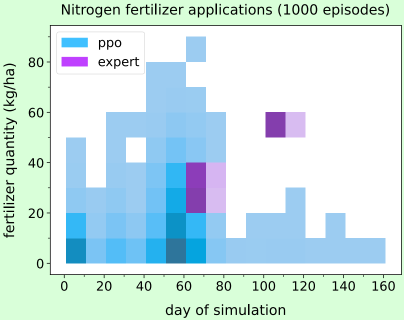
- Policies 1. and 2. are fixed and deterministic.
- Only the weather is stochastic.
- The seeding date depend on the weather, hence varies a bit from one simulation to an other.
- The expert policy depends on expert information that are not available to PPO.

Learning to manage a crop field
Some results (2/3)

Protocol:

- Null and expert policies are evaluated on 10^3 seasons.
- RL: Trained on 10^6 simulated seasons, then evaluated on 10^3 other seasons.





untuned PPO is best.
It shows less variability.

Learning to manage a crop field

Some results (3/3)

	null	expert	PPO
grain yield (kg/ha)	1141.1 (344.0)	**3686.5** (1841.0)	3463.1 (1628.4)
massic nitrogen in grains (%)	1.1 (0.1)	**1.7** (0.2)	1.5 (0.3)
total fertilization (kg/ha)	**0** (0)	115.8 (5.2)	82.8 (15.2)
number of applications	**0** (0)	3.0 (0.1)	5.7 (1.6)
nitrogen use efficiency (kg/kg)	n.a.	22.0 (14.1)	**28.3** (16.7)
nitrate leaching (kg/ha)	**15.9** (7.7)	18.0 (12.0)	18.3 (11.6)

Mean (std dev) computed on 10^3 seasons.

Learning to manage a crop field

Some results (3/3)

	null	expert	PPO
grain yield (kg/ha)	1141.1 (344.0)	**3686.5** (1841.0)	3463.1 (1628.4)
massic nitrogen in grains (%)	1.1 (0.1)	**1.7** (0.2)	1.5 (0.3)
total fertilization (kg/ha)	**0** (0)	115.8 (5.2)	82.8 (15.2)
number of applications	**0** (0)	3.0 (0.1)	5.7 (1.6)
nitrogen use efficiency (kg/kg)	n.a.	22.0 (14.1)	**28.3** (16.7)
nitrate leaching (kg/ha)	**15.9** (7.7)	18.0 (12.0)	18.3 (11.6)

Mean (std dev) computed on 10^3 seasons.

In short: an untuned PPO learns a very good policy that balances the different criteria.

Learning to manage a crop field

Some results (3/3)

	null	expert	PPO
grain yield (kg/ha)	1141.1 (344.0)	**3686.5** (1841.0)	3463.1 (1628.4)
massic nitrogen in grains (%)	1.1 (0.1)	**1.7** (0.2)	1.5 (0.3)
total fertilization (kg/ha)	**0** (0)	115.8 (5.2)	82.8 (15.2)
number of applications	**0** (0)	3.0 (0.1)	5.7 (1.6)
nitrogen use efficiency (kg/kg)	n.a.	22.0 (14.1)	**28.3** (16.7)
nitrate leaching (kg/ha)	**15.9** (7.7)	18.0 (12.0)	18.3 (11.6)

Mean (std dev) computed on 10^3 seasons.

In short: an untuned PPO learns a very good policy that balances the different criteria.

We obtain the same sort of results on the irrigation task.

R. Gautron *et al.*, *gym-DSSAT: a crop model turned into a Reinforcement Learning environment*, Inria Research Report 9460, June 2022, arxiv: 2207.03270.

R. Gautron, *Reinforcement learning for rop management support to smallholder farmers in countries of the South: towards risk management*, PhD dissertation, defended Dec. 2022, Université de Montpellier.

Learning to manage a crop field

A few last words about crop management

- Many topics remain to be studied.

Learning to manage a crop field

A few last words about crop management

- Many topics remain to be studied.
- Risk-aware policy.

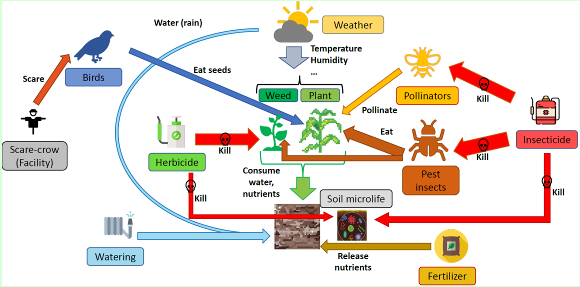
  See Baudry *et al.*, Optimal Thompson Sampling strategies for support-aware CVaR bandits, ICML 2021

Learning to manage a crop field

A few last words about crop management

- Many topics remain to be studied.
- Risk-aware policy.
- gym-DSSAT is great because it is very accurate, but it is hard to manage as a piece of software and limited to DSSAT features.

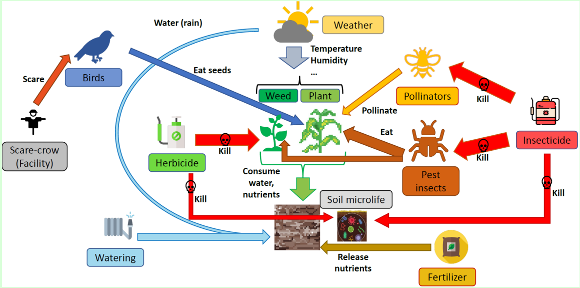
- Many topics remain to be studied.
- Risk-aware policy.
- gym-DSSAT is great because it is very accurate, but it is hard to manage as a piece of software and limited to DSSAT features.
- $\rightsquigarrow$ Farm-gym: toy farm management environment for RL:



Learning to manage a crop field

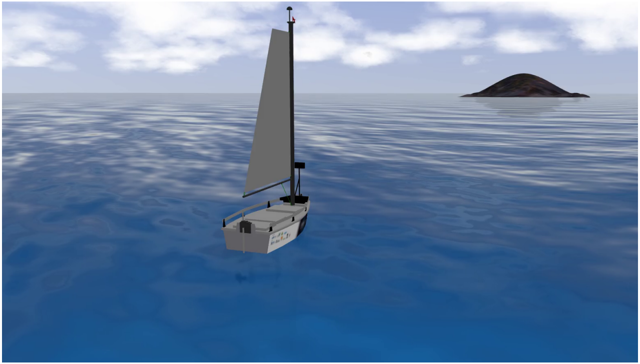
Farm-gym

- ⇝ Farm-gym: toy farm management environment for RL:



- Less accurate but much richer than gym-DSSAT.
- Meant to investigate new problem features: stochastic environment, several coupled feedback loops, cost of action, cost-benefit objective function, multi-objective objective function, *etc.*
- `https://github.com/farm-gym/farm-gym`

Learning to sail



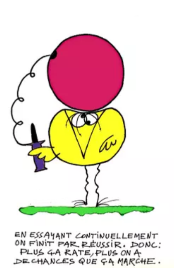
In collaboration with Inria-Chile (Luis, Nayat), UFF (Esteban Clua), Univ. Fed. do Rio Grande do Norte (Luis Gonçalvez).

Rooted on the Stic AmSud EMISTRAL project.

Work in progress.

Learning to sail

- Goal: create an autonomous sailing boat to complete biological missions.
- The environment (sea, wheather, etc) is unknown and non stationary $\rightsquigarrow$ learning and adaptation.
- Shadock approach: put a boat at sea, and let it learn.

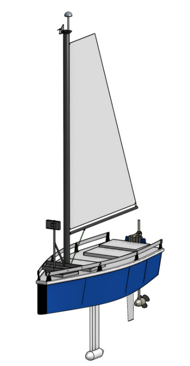


- Rational approach: design a digital twin, train it, transfer sim2real, have the real boat fine tuned and adapt its behavior to real conditions at sea.

Learning to sail

F-Boat

E-Boat

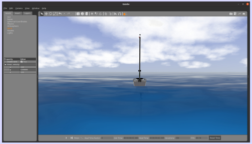


Length: 2.5 m; Width: 0.83 m; Mast Length: 3.13 m.
Electrical propeller, sail.
Actions: control the boom, the rudder, and propeller on/off.
2 cameras.

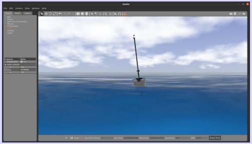
Learning to sail

- Simulator developed in Gazebo: sea, wind, interaction with the E-boat.
- Check that the E-Boat reacts in silico like the F-boat in the real.

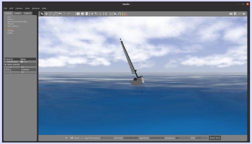
no wind



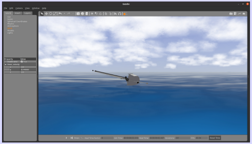
20 *knots*



25 *knots*

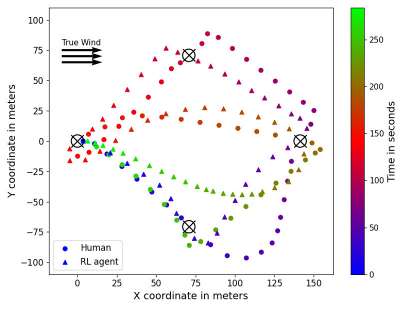


30 *knots*



Learning to sail

- Can E-Boat learn to sail?
- The electrical propeller is just a backup, to avoid losing the boat: we should use it as little as possible, possibly never, just using the sail.
- Learning to perform a regatta:



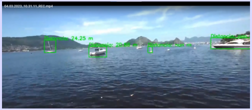
Learning to sail

On-going work

- image analysis 

Learning to sail

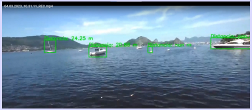
On-going work

- image analysis 
- learning to avoid obstacles 

Learning to sail

On-going work

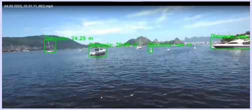
- image analysis 

- learning to avoid obstacles 
- improving the control of the E-Boat

Learning to sail

On-going work

- image analysis 

- learning to avoid obstacles 
- improving the control of the E-Boat
- sim2real

Vasconcellos *et al.*, RL robotic sailboats: simulator and preliminary results, *6th Robot Learning Workshop NeurIPS*, 2023.

Araújo *et al.*, General system architecture and COTS prototyping of an AIoT-enabled sailboat for autonomous aquatic ecosystem monitoring, *IEEE Trans on IoT*, to appear.

Araújo *et al.*, Vision of the Seas: Open Visual Perception Framework for Autonomous Sailing Vessels, *Proc. IWSSIP*, 2023.

Few last words

Few last words

- RL is picking-up.

Few last words

- RL is picking-up.
- RL can work.

Few last words

- RL is picking-up.
- RL can work.
- You don't need months of training of thousands on GPUs to obtain interesting and useful results.

Few last words

- RL is picking-up.
- RL can work.
- You don't need months of training of thousands on GPUs to obtain interesting and useful results.
- Yes, RL is not easy to tame.

Few last words

- RL is picking-up.
- RL can work.
- You don't need months of training of thousands on GPUs to obtain interesting and useful results.
- Yes, RL is not easy to tame.
- There are plenty of exciting fields of application for RL, with challenges for theoreticians and practitioners, fun applications, and serious ones.

Few last words

- RL is picking-up.
- RL can work.
- You don't need months of training of thousands on GPUs to obtain interesting and useful results.
- Yes, RL is not easy to tame.
- There are plenty of exciting fields of application for RL, with challenges for theoreticians and practitioners, fun applications, and serious ones.
- Experimental methodology issues: check-out our Adastop paper `hal-04132861` and code `https://github.com/TimotheeMathieu/adastop`.

Few last words

- RL is picking-up.
- RL can work.
- You don't need months of training of thousands on GPUs to obtain interesting and useful results.
- Yes, RL is not easy to tame.
- There are plenty of exciting fields of application for RL, with challenges for theoreticians and practitioners, fun applications, and serious ones.
- Experimental methodology issues: check-out our Adastop paper `hal-04132861` and code `https://github.com/TimotheeMathieu/adastop`.
- We make available open source software to challenge the RL community with real problems: feel free to use it!

Thank you for your attention.

Check out https://team.inria.fr/scool/.
Get in touch: philippe.preux@inria.fr.