Learning Crop Management by Reinforcement: gym-DSSAT

Romain Gautron, Cirad/CGIAR
Emilio J. Padrón, UDC–Computer Architecture Group & CITIC, Spain
Philippe Preux, Scool: Inria, Université de Lille, CNRS, France
Julien Bigot, Université Paris-Saclay, CEA, France
Odalric-Ambrym Maillard, Scool: Inria, Université de Lille, CNRS, France
Gerrit Hoogenboom, University of Florida Gainesville, USA
Julien Teigny, Scool: Inria, Université de Lille, CNRS, France

philippe.preux@inria.fr

Aim of gym-DSSAT

Simulate crop growth to train a sequential decision-making agent.

What is gym-DSSAT?

DSSAT: state-of-the-art crop simulator.

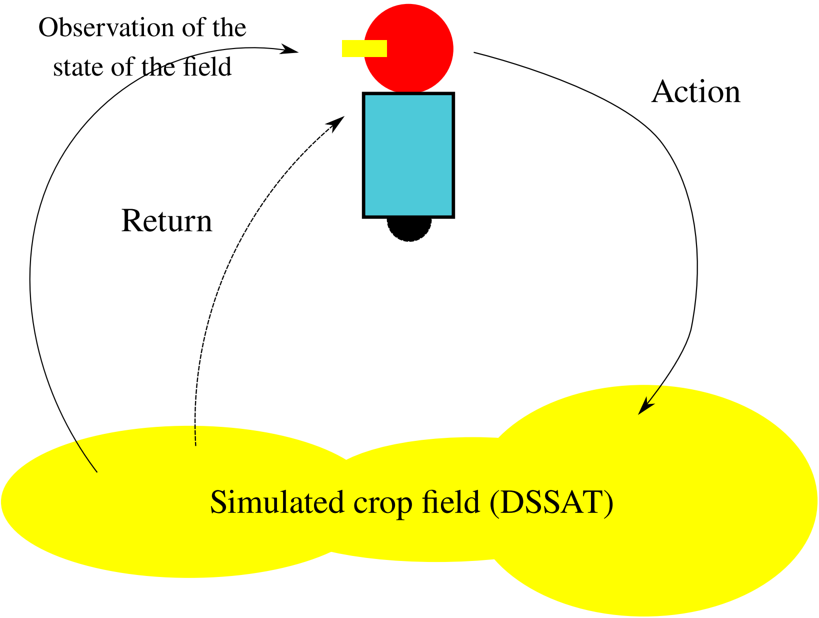
gym: standardized API to connect a reinforcement learning agent with a simulator of its environment.

- The very first version of gym-DSSAT dates back end of 2021.
- gym-DSSAT is an on-going effort.
- DSSAT offers a vast amount of possible simulations. gym-DSSAT currently handles some of them.

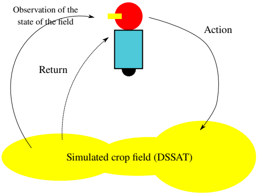
DSSAT https://dssat.net
Decision Support System for Agrotechnology Transfer

- developed for more than 30 years now, U. Florida, Gainsville.
- mechanistic model of crop.
- 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), ...
  + the actions made in the field: irrigation, fertilization, tillage, ... on a daily basis.
- simulates a unit of surface very finely: interactions between the soil properties with roots then growth of the plant (PDE integration over time).

RL agent


Observation of the
state of the field
Action
Return
Simulated crop field (DSSAT)

RL agent and gym



- create the environment `env`
- `env.reset ()`
- Iterate on: one day interaction:
  - choose the next action to perform
  - ```
    next_observation,
       return, finished?,
         d = env.step (action)
    ```
  - update the learning agent

In gym-DSSAT, the agent really gets an observation, not a full state.

Currently, actions consist in applying a certain amount of fertilizer and a certain amount of irrigation per day (per unit of surface).
    ```

gym-DSSAT features

- gym-DSSAT simulates a crop season.
- thousands different soils from all over the world.
- 42 potential crops (wheat, maize, rice, chickpea, ...): **only maize for RL problems so far**.
- Various cultivars for each crop.

  Soils and cultivars have been calibrated by agronomists using extensive, multi-year real field trials.
- Weather: recorded weather or weather generator (from all over the world).

- Observation = collection of measurements amenable to a real farmer.

  Observed features are defined in a config yaml file.
- Objective: may be customized, combining various performance indicators.

  The return function is defined in an easy to customize python file.

Out-of-the-box gym-DSSAT

- 3 built-in problems: 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?

  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 and cost.
- The daily return is defined by:
  $r(day) = \text{plant N uptake}(day, day+1) - 0.5 \times \text{fertilizer quantity}(day)$
- The goal is to maximize $\sum_{day=0}^{day=\text{harvest}} r(day)$.

A few preliminary results (1/3)

We compare:

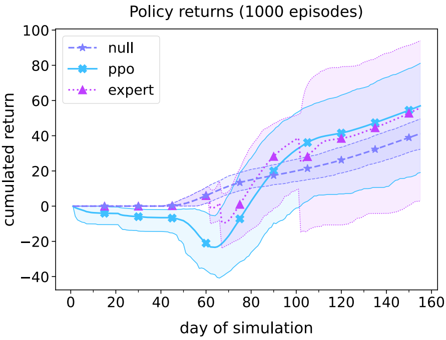
1. a null policy which does not fertilize,
2. an expert policy used in the original 1982 field experiment,
3. a policy learned by RL (basic untuned PPO).

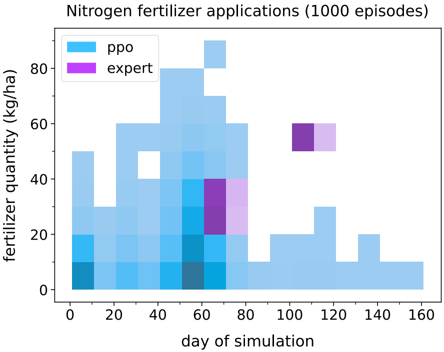
Policies 1. and 2. are fixed and deterministic.
Only the weather is stochastic.

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.

A few preliminary results (2/3)





A few preliminary 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)
application number	**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)

This table contains the mean (std dev) measured on 10^3 evaluation 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.

Under the hood

- DSSAT is a large software program written in Fortran (300 klocs, 450 files).
- DSSAT reads a set of configuration files, runs the simulation accordingly, and outputs result files.
  No notion of the interaction loop required by RL agents.
- Today, almost all RLers use Python as a scripting language and know nothing about Fortran.
- ⇒ a python/Fortran connection is necessary.

  Based on the PDI library (`https://pdi.dev/master`).
  PDI makes the interaction between python and the information processed in DSSAT much easier to configure.

How can I use gym-DSSAT?

Pre-requisite: you know the basics of RL, and how to code an RL agent.

1. Go to: `https://gitlab.inria.fr/rgautron/gym_dssat_pdi`
2. $\rightarrow$ Installation section.
3. $\rightarrow$ Tutorial section.

gym-DSSAT is open source software, released under a 3-Clause BSD licence.

The future of gym-DSSAT

Lots of things to do:

- Experiment with the existing out-of-the-box gym-DSSAT.

  E.g. studying the return function; fine tune RL algorithms able to generalize to various soil/weather/economical conditions; study the impact of global warming on learned policies.

- Study the multi-objective aspect of the problem.
- Extend the set of actions to those defined in DSSAT.
- Extending gym-DSSAT to the 41 other crops available in DSSAT.
- Extension to the management of more than 1 field.
- Keep up with DSSAT upgrades.

gym-DSSAT

For the RL community, gym-DSSAT is a great, original environment including a wide range of fundamental research problems still loosely investigated, potentially impactfull w.r.t. **sustainable development**, based on a state-of-the-art crop simulator.

For agronomists, a tool to investigate how reinforcement learning may be used to improve crop management.

If you want to get involved:

`gym-dssat@inria.fr`

https://gitlab.inria.fr/rgautron/gym_dssat_pdi

gym-dssat@inria.fr

Check out https://arxiv.org/abs/2207.03270 for further details.