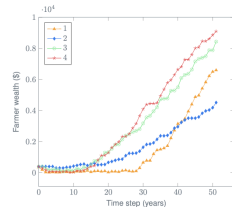
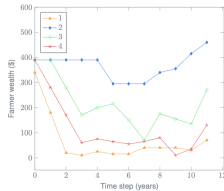
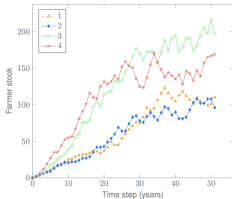




## 851-0101-86 S Complex Social Systems: Modeling Agents, Learning, and Games

### CropWar: Agent-based simulation of agricultural interactions

C.Golling, G.Mourouga, A.Moser, O.Schmidt, A.H.Keshavarzzadeh





## Introduction and motivation

## The CropWar model

### Deterministic versions

- Basic version: selling and stocking

- The Map class: expansion

- The Market model: dynamic pricing

### Reinforcement Learning versions

- Introduction

- Proximal Policy Optimization

- ML agent trained against introverts

- ML agent trained against traders

## Summary and Outlook

### Crop War in perspective

- Work Flow for model extension



CropWar aims at modelling **interactions between farmers** growing crops in a **geographic area** with a finite amount of **water resources** and selling them on a **market** to feed a population.

CropWar aims at modelling **interactions between farmers** growing crops in a **geographic area** with a finite amount of **water ressources** and selling them on a **market** to feed a population.

The model is organised as follows:

CropWar aims at modelling **interactions between farmers** growing crops in a **geographic area** with a finite amount of **water resources** and selling them on a **market** to feed a population.

The model is organised as follows:

- ▶ **v1.1** basic version: growing, selling, stocking

CropWar aims at modelling **interactions between farmers** growing crops in a **geographic area** with a finite amount of **water resources** and selling them on a **market** to feed a population.

The model is organised as follows:

- ▶ v1.1 basic version: growing, selling, stocking
- ▶ v1.2 implementation of the Map class: spatial expansion

CropWar aims at modelling **interactions between farmers** growing crops in a **geographic area** with a finite amount of **water resources** and selling them on a **market** to feed a population.

The model is organised as follows:

- ▶ v1.1 basic version: growing, selling, stocking
- ▶ v1.2 implementation of the Map class: spatial expansion
- ▶ v1.3 implementation of the Market model: dynamic pricing and agent personalities

CropWar aims at modelling **interactions between farmers** growing crops in a **geographic area** with a finite amount of **water resources** and selling them on a **market** to feed a population.

The model is organised as follows:

- ▶ v1.1 basic version: growing, selling, stocking
- ▶ v1.2 implementation of the Map class: spatial expansion
- ▶ v1.3 implementation of the Market model: dynamic pricing and agent personalities
- ▶ v2.1 reinforcement learning: identifying the optimal strategy against Introvert agents





CropWar aims at modelling **interactions between farmers** growing crops in a **geographic area** with a finite amount of **water ressources** and selling them on a **market** to feed a population.

The model is organised as follows:

- ▶ v1.1 basic version: growing, selling, stocking
- ▶ v1.2 implementation of the Map class: spatial expansion
- ▶ v1.3 implementation of the Market model: dynamic pricing and agent personalities
- ▶ v2.1 reinforcement learning: identifying the optimal strategy against Introvert agents
- ▶ v2.2 reinforcement learning: identifying the optimal strategy against Trader agents

# Deterministic versions

Basic version: selling and stocking

CropWar



3

**v1.1: Basic version**

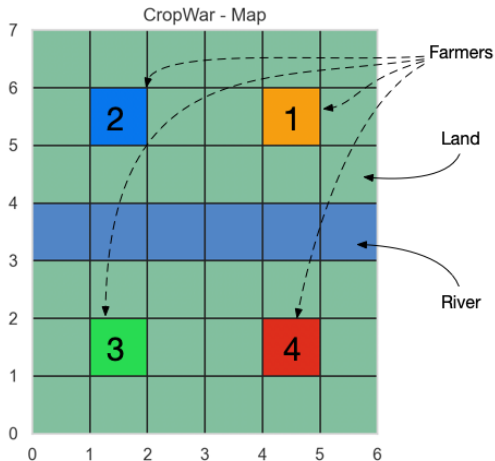
# Deterministic versions

Basic version: selling and stocking

CropWar



4



In **v1.1** farmers only have access to one cell on which they can choose to grow one of two crops (A or B)

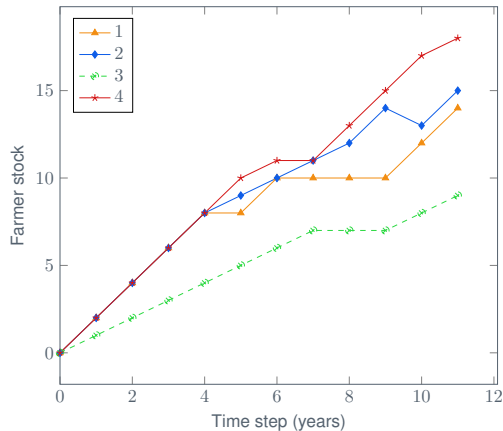
# Deterministic versions

Basic version: selling and stocking

CropWar



5



This plot shows the evolution of stock as a function of time:

Farmers 1,2 and 4 grow crop A

Farmer 3 grows crop B

At each time step, they chose to stock or sell their yield

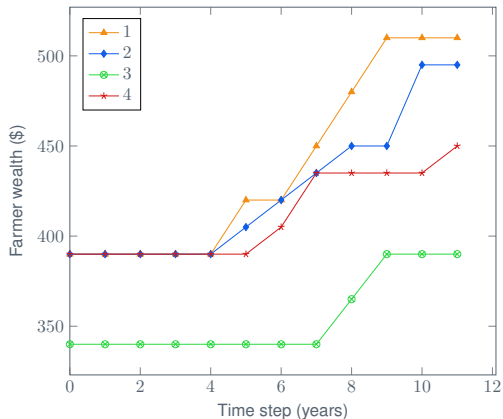
# Deterministic versions

Basic version: selling and stocking

CropWar



6



This plot shows the corresponding evolution of the farmer's wealth as a function of time. Selling corresponds to an increase in wealth, stocking to a constant value.

# Deterministic versions

The Map class: expansion



**v1.2: Spatial expansion**

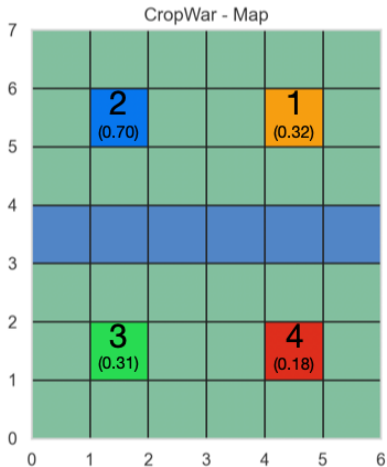
# Deterministic versions

The Map class: expansion

CropWar

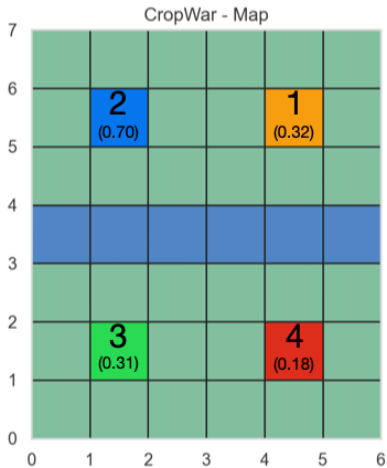


8



The `buy_threshold` (in brackets) indicates the tendency of farmers to expand their farms by buying neighbouring land in order to grow more crops

## The Map class: expansion





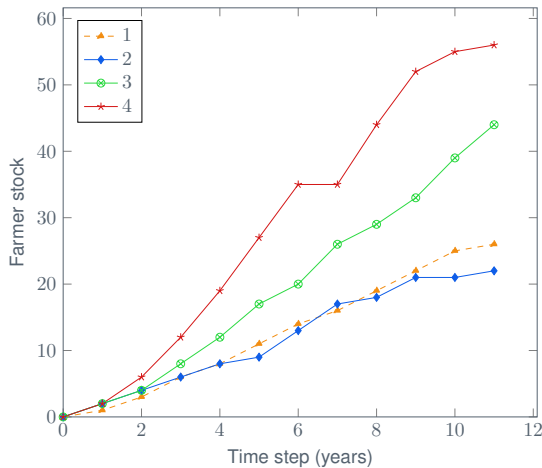
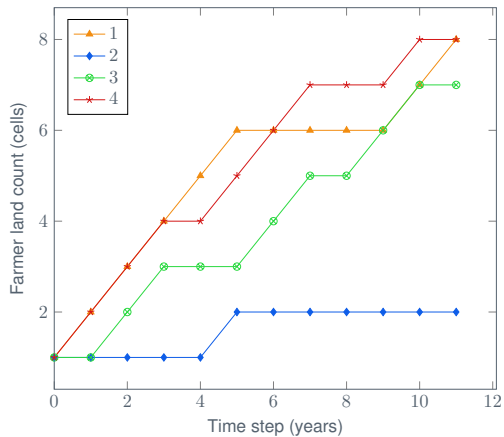
# Deterministic versions

The Map class: expansion

CropWar



10



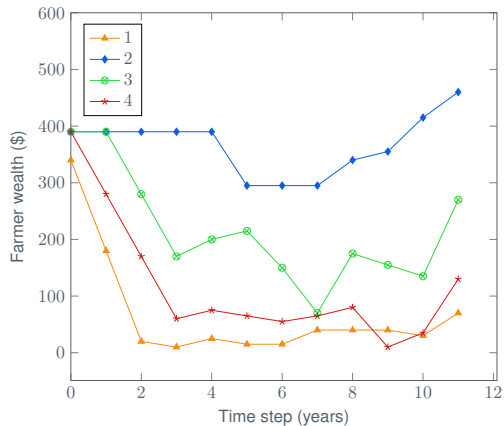
# Deterministic versions

The Map class: expansion

CropWar



11



Farmer 2 seems to remain wealthier than other farmers over 10 time steps, due to not spending money to expand

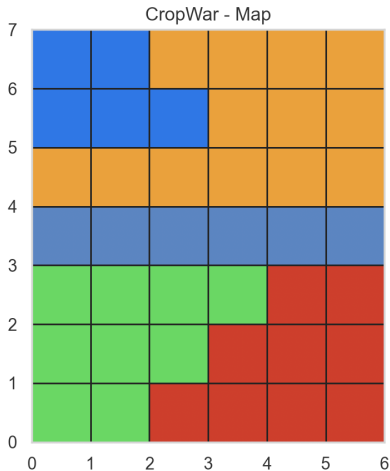
# Deterministic versions

The Map class: expansion

CropWar



12



If we extend the simulation to 50 time steps, all available land is bought by farmers

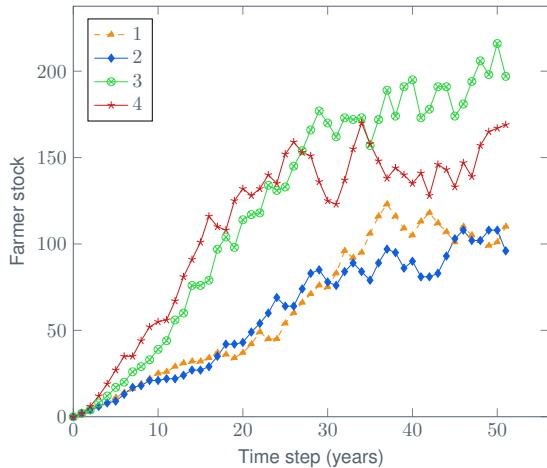
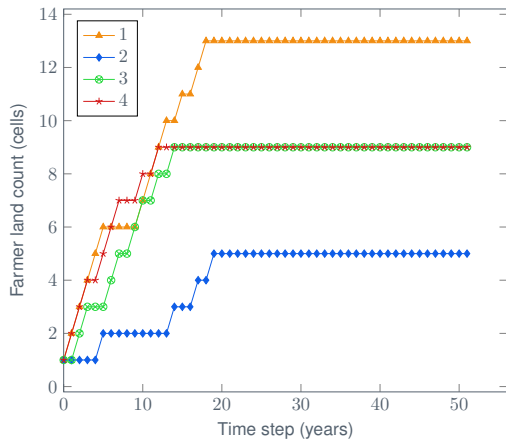
# Deterministic versions

The Map class: expansion

CropWar



13



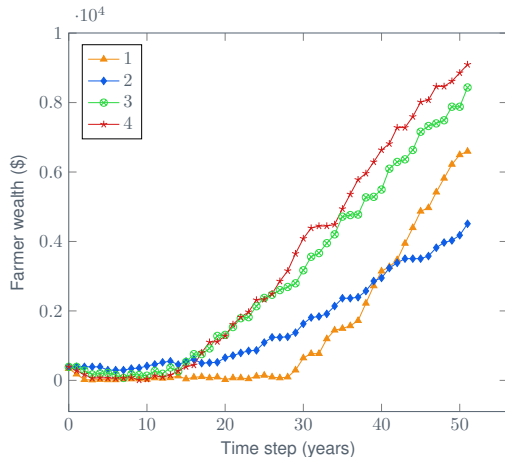
# Deterministic versions

The Map class: expansion

CropWar



14



On the long run, farmers with more land become wealthier, although the type of crop also seems to have an importance

# Deterministic versions

The Market model: dynamic pricing



## v1.3: Dynamic pricing of crops

# Deterministic versions

The Market model: dynamic pricing



In `v1.3` we implement a `Market` class, which updates the price of crops based on supply and demand.

# Deterministic versions

The Market model: dynamic pricing



In **v1.3** we implement a `Market` class, which updates the price of crops based on supply and demand.

1. Harvesting period: agents harvest according to their crop choice and add the harvest yield to the stock.



# Deterministic versions

The Market model: dynamic pricing



In **v1.3** we implement a `Market` class, which updates the price of crops based on supply and demand.

1. Harvesting period: agents harvest according to their crop choice and add the harvest yield to the stock.
2. Interaction period: market interaction takes place.

# Deterministic versions

The Market model: dynamic pricing



In **v1.3** we implement a `Market` class, which updates the price of crops based on supply and demand.

1. Harvesting period: agents harvest according to their crop choice and add the harvest yield to the stock.
2. Interaction period: market interaction takes place.
3. Strategy period: the Agent's personality influences its decision for the next time step.

# Deterministic versions

The Market model: dynamic pricing



In v1.3 we also implement agent personalities Traders and Introverts

# Deterministic versions

The Market model: dynamic pricing

CropWar



17

In **v1.3** we also implement agent personalities Traders and Introverts

- ▶ Traders agents react to market prices and may change the crops they were growing.

# Deterministic versions

The Market model: dynamic pricing

CropWar



17

In **v1.3** we also implement agent personalities Traders and Introverts

- ▶ Traders agents react to market prices and may change the crops they were growing.
- ▶ Introverts agents pick a crop and stick with it.

# Deterministic versions

The Market model: dynamic pricing

CropWar



17

In **v1.3** we also implement agent personalities `Traders` and `Introverts`

- ▶ `Traders` agents react to market prices and may change the crops they were growing.
- ▶ `Introverts` agents pick a crop and stick with it.

Some of these agents may also have different expanding strategies

# Deterministic versions

The Market model: dynamic pricing

CropWar



17

In **v1.3** we also implement agent personalities `Traders` and `Introverts`

- ▶ `Traders` agents react to market prices and may change the crops they were growing.
- ▶ `Introverts` agents pick a crop and stick with it.

Some of these agents may also have different expanding strategies

- ▶ `Stationary` agents remain on their initial cell

# Deterministic versions

The Market model: dynamic pricing

CropWar



17

In `v1.3` we also implement agent personalities `Traders` and `Introverts`

- ▶ `Traders` agents react to market prices and may change the crops they were growing.
- ▶ `Introverts` agents pick a crop and stick with it.

Some of these agents may also have different expanding strategies

- ▶ `Stationary` agents remain on their initial cell
- ▶ `Expanding` agents may buy neighbouring land



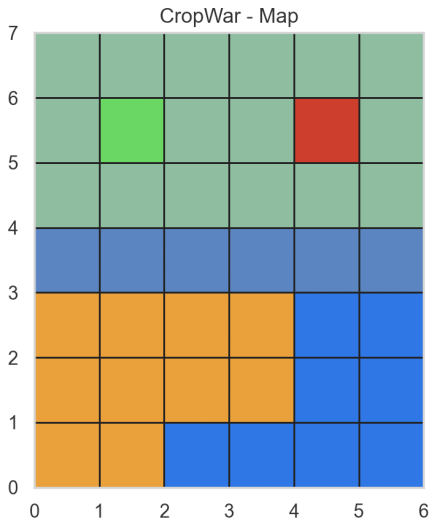
# Deterministic versions

The Market model: dynamic pricing

CropWar



18



Simulation with 2 stationary Introverts  
and 2 expanding Traders

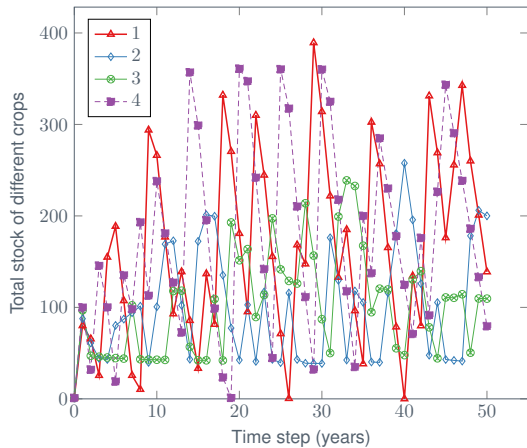
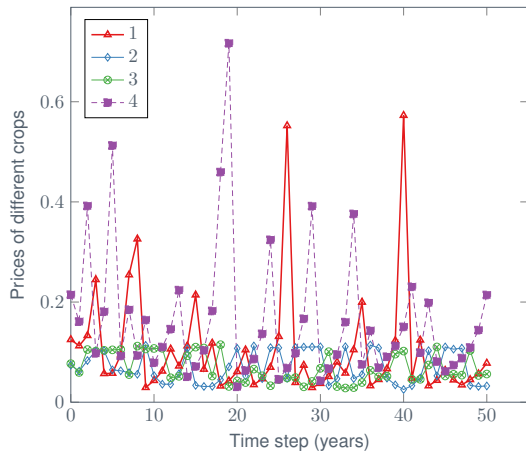
# Deterministic versions

The Market model: dynamic pricing

CropWar



19



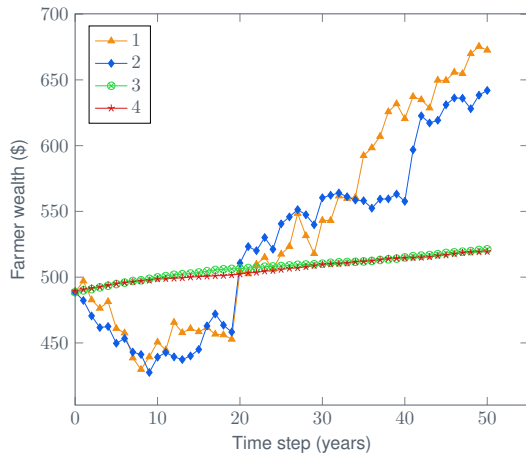
# Deterministic versions

The Market model: dynamic pricing

CropWar



20



Farmer wealth as a function of time: expanding  
Traders agents take more risks, which seem  
to pay off on the long term



## Reinforcement Learning

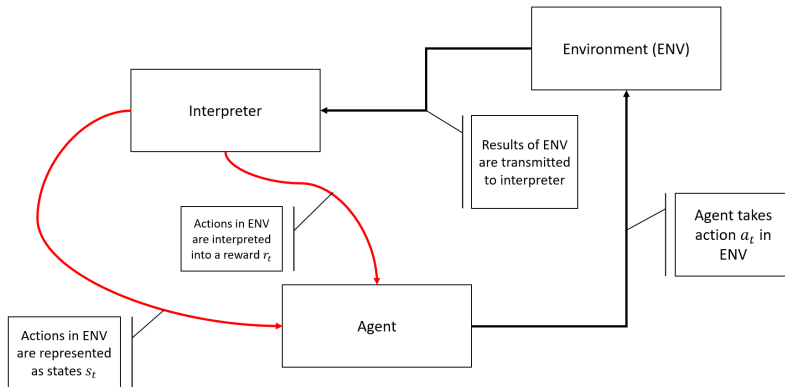


The preceding versions **v1.1 - v1.3** defined the deterministic agents and the model of the environment in which actions take place.

In order to see unpredictable, emerging behaviour, the agents would need to adjust to evolutions of the market and learn.

# Reinforcement Learning versions

## Introduction



Feedback loop in RL. An agent decides to do a certain action  $a_t$  based on his observation of the environment. This affects the environment, which yields a new state. The effect of the agents' action is then interpreted to update the strategy of the agent.

# Reinforcement Learning versions

## Proximal Policy Optimization



$$\underbrace{L(\theta)}_{\text{Policy loss function}} = \underbrace{\hat{\mathbf{E}}_t}_{\text{expectation at iteration } t} \left[ \underbrace{\log(\pi_{\theta}(a_t|s_t))}_{\text{stochastic policy}} \underbrace{\hat{A}_t}_{\text{estimator of advantage of current action}} \right] \quad (1)$$

# Reinforcement Learning versions

## Proximal Policy Optimization



$$\underbrace{L(\theta)}_{\text{Policy loss function}} = \underbrace{\hat{\mathbf{E}}_t}_{\text{expectation at iteration } t} \left[ \underbrace{\log(\pi_\theta(a_t|s_t))}_{\text{stochastic policy}} \underbrace{\hat{A}_t}_{\text{estimator of advantage of current action}} \right] \quad (1)$$

where:



# Reinforcement Learning versions

## Proximal Policy Optimization



$$\underbrace{L(\theta)}_{\text{Policy loss function}} = \underbrace{\hat{\mathbf{E}}_t}_{\text{expectation at iteration } t} \left[ \underbrace{\log(\pi_\theta(a_t|s_t))}_{\text{stochastic policy}} \underbrace{\hat{A}_t}_{\text{estimator of advantage of current action}} \right] \quad (1)$$

where:

- $\pi_\theta(a_t|s_t)$  is given as transition probabilities in the MC of taking action  $a_t$  in state  $s_t$ . It is a neural network, that suggests an action for a given state based on previous training experience.



$$\underbrace{L(\theta)}_{\text{Policy loss function}} = \underbrace{\hat{\mathbf{E}}_t}_{\text{expectation at iteration } t} \left[ \underbrace{\log(\pi_\theta(a_t|s_t))}_{\text{stochastic policy}} \underbrace{\hat{A}_t}_{\text{estimator of advantage of current action}} \right] \quad (1)$$

where:

- ▶  $\pi_\theta(a_t|s_t)$  is given as transition probabilities in the MC of taking action  $a_t$  in state  $s_t$ . It is a neural network, that suggests an action for a given state based on previous training experience.
- ▶  $\hat{A}_t$  is computed as a discounted reward. This neural net is updated with the experience (i.e. reward) that the agent collects in an environment.

# Reinforcement Learning versions

ML agent trained against introverts

CropWar



25

**v2.1: RL agent trained against Introvert farmers**

# Reinforcement Learning versions

ML agent trained against introverts

CropWar



26

The reward function that was used for training is given by

$$\frac{1}{3} \left( \underbrace{b^2}_{\text{Fraction of Global Budget}} + \underbrace{r^2}_{\text{Current Ranking}} + \underbrace{s^2}_{\text{Fraction of Global supply}} \right) \quad (2)$$

where  $r$  is 1 for first, 0.66 for second, 0.33 for third, and 0.0 for last agent.

# Reinforcement Learning versions

ML agent trained against introverts

CropWar



27

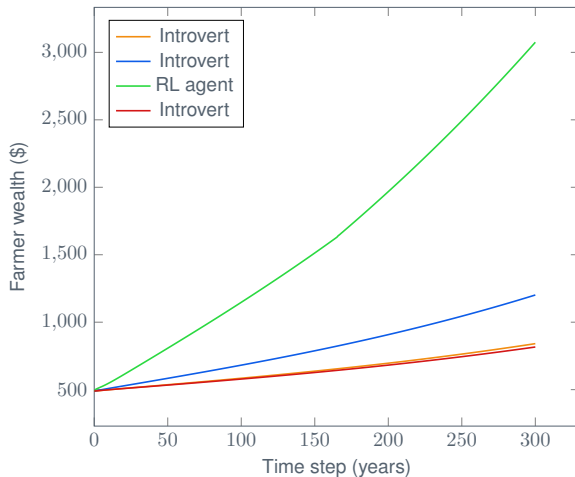


Figure: Budget evolution over 300 iterations

# Reinforcement Learning versions

ML agent trained against traders

CropWar



28

**v2.1: RL agent trained against Trader farmers**

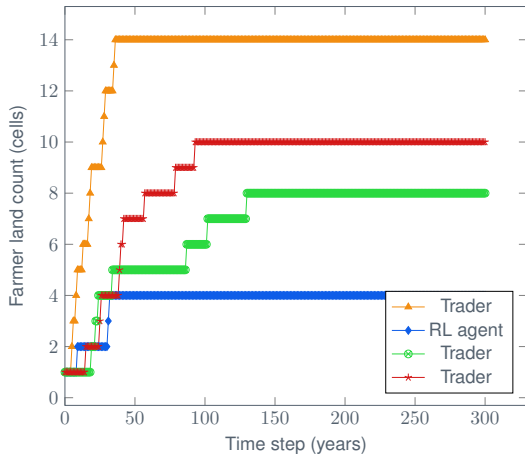
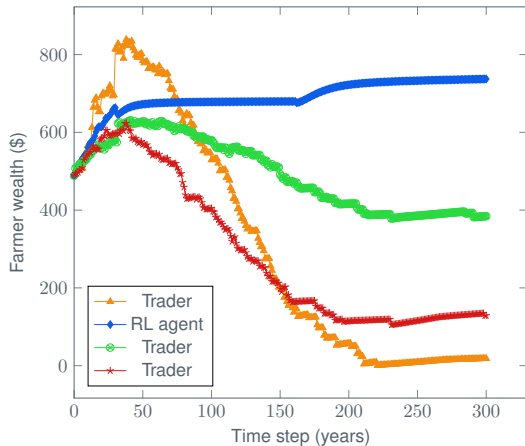
# Reinforcement Learning versions

ML agent trained against traders

CropWar



29



# Reinforcement Learning versions

ML agent trained against traders

CropWar



30

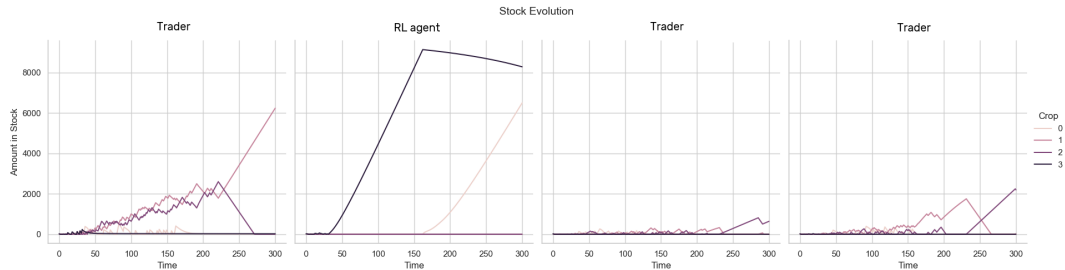


Figure: Stock evolution for the three Trader agents and the RL agent.



**Work Flow to improve the model**

# Crop War in perspective

Work Flow for model extension



We organised the model to allow for an easy code review and potential improvement by other people:

We organised the model to allow for an easy code review and potential improvement by other people:

- ▶ All relevant material is hosted in a Git repository on Github.

We organised the model to allow for an easy code review and potential improvement by other people:

- ▶ All relevant material is hosted in a Git repository on Github.
- ▶ A pdf report was created to offer a model overview.

We organised the model to allow for an easy code review and potential improvement by other people:

- ▶ All relevant material is hosted in a Git repository on Github.
- ▶ A pdf report was created to offer a model overview.
- ▶ The Python code can be edited to play around with different parameters and equations and a Documentation was created to allow for an easier code review.

Possible extensions include:

Possible extensions include:

- ▶ More realistic spending of ressources: equipment investments, maintenance costs, etc...

Possible extensions include:

- ▶ More realistic spending of resources: equipment investments, maintenance costs, etc...
- ▶ Bigger and more realistic maps



Possible extensions include:

- ▶ More realistic spending of resources: equipment investments, maintenance costs, etc...
- ▶ Bigger and more realistic maps
- ▶ Weather events

Possible extensions include:

- ▶ More realistic spending of resources: equipment investments, maintenance costs, etc...
- ▶ Bigger and more realistic maps
- ▶ Weather events
- ▶ Ecological impact of different crops