Environmental Data Science (2023), xx: 1–12
doi:10.1017/xxxx

Application Paper

Nitrogen management with Reinforcement Learning and crop growth models

Michiel G.J. Kallenberg*, Hiske Overweg, Ron van Bree and Ioannis N. Athanasiadis

Laboratory of Geo-information Science and Remote Sensing, Wageningen University & Research, The Netherlands
*Corresponding author. Email: michiel.kallenberg@wur.nl

Received xx xxx xxxx

Keywords: Reinforcement learning, smart farming, crop growth models, nitrogen, winter wheat

Abstract

The growing need for agricultural products and the challenges posed by environmental and economic factors have created a demand for enhanced agricultural systems management. Machine learning has increasingly been leveraged to tackle agricultural optimization problems, and especially Reinforcement Learning (RL), a subfield of machine learning, seems a promising tool for data driven discovery of future farm management policies. In this work we present the development of *CropGym*, an OpenAI Gym environment where a reinforcement learning agent can learn crop management policies using a variety of process-based crop growth models. As a use case we report on the discovery of strategies for nitrogen application in winter wheat. An RL agent is trained to decide weekly on applying a discrete amount of nitrogen fertilizer, with the aim of achieving a balance between maximizing yield and minimizing environmental impact. Results show that close to optimal strategies are learned, competitive with standard practices set by domain experts. In addition we evaluate, as an out-of-distribution test, whether the obtained policies are resilient against a change in climate conditions. We find that, when rainfall is sufficient, the RL agent remains close to the optimal policy. With *CropGym* we aim to facilitate collaboration between the RL and agronomy communities to address the challenges of future agricultural decision-making.

Impact Statement

This paper presents *CropGym*, an open simulation environment to conduct reinforcement learning research for discovering adaptive, data-driven policies for farm management using a variety of process-based crop growth models. With a use case on nitrogen management we demonstrate the potential of RL to learn sustainable policies that are competitive with standard practises set by domain experts.

1. Introduction

In recent years, smart farming technologies have been considered key-enablers to reduce the usage of chemicals (fertilizers and plant protection products) as well as to reduce greenhouse gas emissions to enable reaching the Green Deal targets [1]. A promising direction within smart farming technology research focuses on developing decision support systems (DSS). These human-computer systems aim at providing farmers with a list of advice for supporting their business or organizational decision-making activities to optimize returns on inputs while preserving resources, within (environmental) constraints. With the evolution of agriculture into Agriculture 4.0, thanks to the employment of current

© The Authors(s) 2020. Published by Cambridge University Press. This is an Open Access article, distributed under the terms of the Creative Commons Attribution licence (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted re-use, distribution, and reproduction in any medium, provided the original work is properly cited.

technologies like Internet of Things, Remote Sensing, Big Data, and Artificial Intelligence, DSSs of various kinds have found their way to agriculture. Examples include, but are not limited to, applications for agricultural mission planning, climate change adaptation, food waste control, plant protection, and resource management of water and nutrients [2].

The backbone of a DSS typically consists of a set of models that provide a representation of the environment and processes therein that are to be optimized. Especially for resource management a substantial share of DSSs are based on process-based crop growth models [3]. These models mathematically describe growth, development and yield of a crop for given environmental conditions, such as type of soil, weather, and availability of water and nutrients. The scientific community offers numerous crop growth models with different levels of sophistication, limitations and limits of applicability [4, 5]. Widely used frameworks are, amongst others, APSIM ([6]), DSSAT ([7]), and PCSE ([8]), which contains models such as LINTUL-3 ([9]) and WOFOST ([10]).

Generally speaking there are two major ways in which crop growth models are utilized in a DSS to derive crop management decisions [11]: (1) components in the model are exploited to provide estimates of crop yield limiting factors, such as (future) deficiencies of nutrients and water, and (2) the model is employed as a specialized simulator to assess the impact of a set of (predefined) crop management practices. For both cases it is not trivial to find the optimal set of actions, as decisions have to be made under uncertainty. For instance, driving factors for e.g. future nutrient uptake, such as future weather conditions, are uncertain at the time the model is asked for an advice on fertilizer application.

Finding an optimized sequence of (crop management) decisions under uncertainty is a challenging task for which machine learning has increasingly been leveraged. Especially reinforcement learning (RL), a subfield of machine learning, seems a relevant tool to tackle agricultural optimization problems [12]. RL seeks to train intelligent agents in a trial and error fashion to take actions in an environment based on a reward signal. In RL the environment is formally specified as a Markov decision process (MDP) $\{S, A, T, R\}$, with state space S, an available set of actions A, a transition function T, and a reward function R. In the context of e.g. crop management S may consist of (virtual) measurements on the state of the crop, A may be dose of fertilizer to apply, T may be represented by a simulation step of a crop growth model, and R may be defined as the (projected) amount of yield.

Recently a few research works have introduced RL for the management of agricultural systems. For instance RL has been used for climate control in a greenhouse [13], planting, and pruning in a polyculture garden [14], fertilizer [15] and/or water management [16–18], coverage path planning [19], and crop planning [20] in open-field agriculture. A comprehensive overview of reinforcement learning for crop management support is given in Gautron et al. (2022) [21].

As is common practice in RL research in a pioneering stage, practically all mentioned works used simulated environments. Some of these environments have been made publicly available as a software artifact. Examples that build on crop growth models include *CropGym* ([15]), an interface to the Python Crop Simulation Environment (PCSE) [8], *gym-DSSAT* ([22]), an integration of the DSSAT [23] crop models, *CropRL* ([24]), a wrapper around the SIMPLE crop model [25], *SWATGym* ([26]), a wrapper around SWAT [27], and *CyclesGym* ([20]), a wrapper around Cycles [28]. Mentioned examples are implemented with the OpenAI gym toolkit [29], which is a highly used framework for developing and comparing reinforcement learning algorithms. By providing standardized test beds, efforts like these are instrumental in further promoting and accelerating of RL research for agricultural problems.

In this work we present the development of *CropGym*, an OpenAI Gym environment where a reinforcement learning agent can learn farm management policies using a variety of process-based crop growth models. In particular we report on the discovery of strategies for nitrogen application in winter wheat and we evaluate the resiliency of the obtained policies against climate change. The focus on nitrogen is motivated by the fact that (in rain-fed winter wheat) nitrogen is a key driver for yield, yet, if supplied in excessive amount it has a detrimental effect on the environment, including eutrophication of freshwater, groundwater contamination, tropospheric pollution related to emissions of nitrogen oxides and ammonia gas, and accumulation of nitrous oxide, a potent greenhouse gas [30].

2. Methodology

2.1. CropGym

We developed *CropGym*, an OpenAI Gym environment for farm management policies, such as fertilization and irrigation, using process-based crop growth models. *CropGym* is built around the Python Crop Simulation Environment (PCSE), a well established open source framework that includes implementations of a variety of crop simulation models. The software is characterized by a high level of customizability. Input parameters, such as crop characteristics, are easily configurable. For deriving driving variables, such as weather information, a broad selection of sources is available. Furthermore dedicated routines facilitate the assimilation of observational data, such as field measurements. State parameters on crop growth and development, as well as carbon, water and nutrient balances are simulated and outputted at daily time steps. Farm management actions can be applied at the same resolution.

CropGym follows standard gym conventions and enables daily interactions between an RL agent and a crop model. The code is designed in a modular fashion, and allows users to flexibly and easily create custom environments. Users can, for example, base action and reward functions on crop state variables, such as water stress, nitrogen uptake, and biomass. As a backbone a variety of (components of) crop growth models can be selected, or combined. *CropGym* is shipped with a set of preconfigured environments that allow for readily conducting RL research for farm management practices. The source code and documentation is available at https://www.cropgym.ai.

2.2. Use case

In this work we present a use case on nitrogen management in rain-fed winter wheat. An agent was trained to decide weekly on applying a discrete amount of nitrogen fertilizer, with the goal of balancing the trade-off between yield and environmental impact.

In the following we outline the components that comprise the environment of our use case:

State space S consists of the current state of the crop and a multidimensional weather observation, as parameterized with the variables listed in table 1.
Action space A comprises three possible fertilizer application amounts, namely $\{0, 20, 40\}$ kg/ha.
Reward function R constitutes the balance between the gain in yield and the (environmental) costs associated with the application of nitrogen. R is formalized as follows:

$$r_t = (WSO_t^{\pi} - WSO_{t-1}^{\pi}) - (WSO_t^{0} - WSO_{t-1}^{0}) - \beta N_t, \tag{2.1}$$

with t the timestep, WSO the weight of the storage organ (g/m2), and N the amount of nitrogen (g/m2). The upper indices π and *0* refer to the agent's policy and a zero nitrogen policy, respectively. Parameter β determines the trade-off between increased yield and reduced environmental impact. Setting $\beta \approx 2.0$ corresponds to a reward that purely comprises the economic profitability, since a kg of fertilizer is twice as expensive as a kg of wheat [31, 32]. In this work we present results for $\beta = 10.0$ to emphasize the environmental costs.
Transitions are governed by the process-based crop model LINTUL-3 (light interception and utilization [9]), and the weather sequence. The model parameters have been calibrated to simulate winter wheat in the Netherlands [33, 34]. Weather data was obtained from the PowerNASA database for three locations in the Netherlands and one in France for the years 1990 to 2022. An episode runs until the crop has reached maturity, which differs between episodes because of weather conditions.

An RL agent was trained with Proximal Policy Optimization [35] as implemented in the Stable-Baselines3 library [36]. The environment was normalized with the VecNormalize environment wrapper, a normalized reward, and observation clipping set to 10. The discount factor γ was set to 1.0 as we aim to optimize the cumulative reward over the entire episode. To reduce redundancy among the input data we aggregated the timeseries data: The weather sequence, with size 3x7 (i.e. features x days), was

Table 1. *Crop growth and weather variables exposed in the state space S.*

Variable	Meaning	Unit
DVS	Development stage	-
TGROWTH	Total biomass growth (above and below ground)	g/m2
LAI	Leaf area index	-
NUPTT	Total nitrogen uptake	-
TRAN	Transpiration	mm/day
TNSOIL	Total soil inorganic nitrogen	gN/m2
TRAIN	Total rainfall	mm
TRANRF	Transpiration reduction factor	-
WSO	Weight storage organs	g/m2
IRRAD	Incoming global radiation	J/m2/day
TMIN	Minimum temperature	°C
RAIN	Precipitation	cm/day

processed with an average pooling layer, yielding a feature vector of size 3x1. The crop features, with size 9x7, were shrunk to 9x1, by taking the last entry for each feature. Both resulting feature vectors were concatenated and subsequently flattened to obtain a feature vector of size 12. The policy and the value network were a multilayer perceptron with two hidden layers, each of size 128, and activation function tanh . Weights were shared between both networks. Training was done on the odd years from 1990 to 2022 (the even years were reserved for validation), with weather data from (52,5.5), (51.5,5), and (52.5,6.0) (°N,°E). Training ran for 400,000 timesteps using default hyperparameters. We selected PPO as our choice of RL algorithm due to its consistent high performance in RL research and its robust nature [35]. We also explored training a Deep Q-Network (DQN) [37], which yielded similar results to those obtained with (see appendix A).

Two baseline agents were implemented as a reference for the RL agent:

The Standard Practice agent (SP) applies a fixed amount of nitrogen that is the same for all episodes. SP thereby reflects common practice, in which a predetermined amount of nitrogen is applied on three different dates during the season [33]. The static amount of nitrogen SP applies is determined by optimization[1] on the training set.

The Ceres agent applies an episode specific amount of nitrogen, that is optimized[1] for the episode it is evaluated on. Effectively Ceres has access to the weather data of the entire season, which contrasts with the RL agent that only has access to current and passed weather data. Ceres can thus base its actions on future weather conditions, and thereby reflects the upper bound of what any agent can achieve maximally.

[1] For training the baseline agents Ceres and SP we exploited a flaw in the nitrogen leaching component of LINTUL-3. In LINTUL-3 the nitrogen loss is computed as a fixed fraction of the amount of applied fertilizer (i.e. one minus fertilizer recovery fraction), regardless of timing and state dynamics, such as weather conditions. Any surplus of nitrogen is not leached, but remains available for uptake throughout the growing season. In principle, if we don not put constraints on the action space, we may apply all required nitrogen at once, right at the start of the season, thereby allowing for elimination of the timing dimension of the problem. In this setting, optimization of the fertilization policy is reduced to finding the right amount of fertilizer, which can be resolved with a simple optimizer. Policies obtained by this strategy effectively mimics practices in which fertilizer is always applied in a timely manner, since the crop never has to wait before the applied fertilizer becomes available.
Note that we can not employ mentioned optimization regime when the action space is constrained. This premise is violated for the RL agent, as its action space is limited to a discrete amount of fertilizer, with a maximum of 40 kg/ha per action. This prevents the RL agent from applying all fertilizer at once. Moreover unlike the Ceres agent, the RL agent does not have access to future weather conditions, and thus does not know the rewards of its actions in advance. As such, for the RL agent the timing dimension of the problem is preserved.

We evaluated the performance of the implemented agents in the even years from 1990 to 2022 with weather data from (52,5.5) (°N,°E). As performance metrics we computed the cumulative reward, the amount of nitrogen, and the yield, summarized as the median over the test years. For statistical analyses we performed 15 runs initialized with different seeds, and use bootstrapping to estimate the 95% confidence interval around the median.

As an out-of-distribution test we evaluated the resiliency of the policy against a change in climate conditions. For that, we deployed both the trained RL and the baseline agents in a more Southern climate. Practically this was implemented by taking weather data from 48°N, 0.0°E, located in France, as opposed to 52°N, 5.5°E, located in the Netherlands, used during training (see fig. 1).

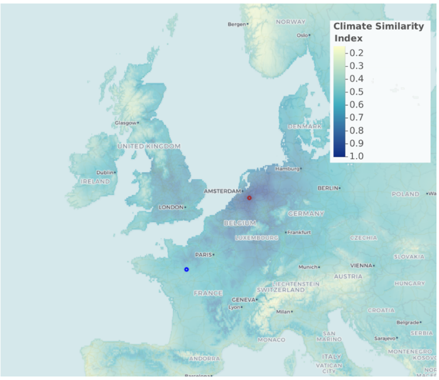


Figure 1. *Locations of the training (red) and out-of-distribution test (blue), with a CCAFS climate similarity index [38] of 1.0 (reference) and 0.573, respectively.*

To reestablish the upper bound Ceres was tuned on the weather data from the Southern climate; SP and RL were not retrained. Robustness of the RL (and SP) agents was evaluated by assessing how close the agents' performance remains to the optimum, as determined by Ceres.

3. Results

A reinforcement learning agent (RL) was trained to find the optimal policy for applying nitrogen that balances yield increase and (environmental) costs. Two baseline agents were implemented for comparison: (1) The *Standard Practice* agent (SP), applies a fixed amount of nitrogen that does not differ between episodes; (2) the *Ceres* agent applies an episode specific amount of nitrogen. The amount of nitrogen SP applies is determined by optimization on the training set. Ceres, on the other hand, applies an amount of nitrogen that is optimized for the episode it is evaluated on. Ceres thereby reflects the upper bound of what any agent can achieve maximally.

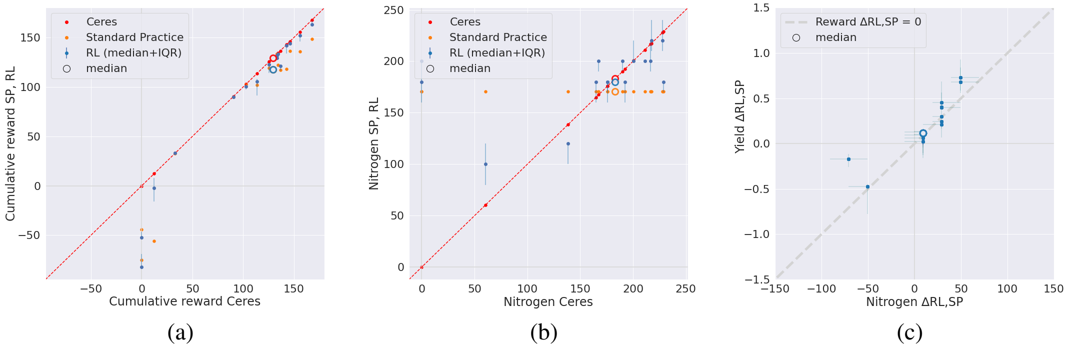
Table 2 reports the performance metrics for each of the three agents, summarized as the median over the test years, and its associated 95% confidence interval. The amount of nitrogen the RL agent applies, and the resulting yield and cumulative reward is close to the upper bound, as reflected by Ceres. Comparing the RL agent with the standard practice (SP), we see that RL applies more nitrogen, which results in a higher yield. The cumulative reward RL achieves is competitive with SP.

Figure 2 (left and middle) shows for each test year the reward obtained and amount of nitrogen applied by each of the three agents, as a function of the reward obtained and amount of nitrogen applied by Ceres. For most test years RL is closer to Ceres than SP, both in terms of obtained cumulative reward and applied nitrogen. For two test years (2006 and 2010) the optimal amount of nitrogen, as determined by Ceres, is zero. In these years, which are characterized by a low amount of rainfall, the extra yield obtained by applying nitrogen does not outweigh the costs. RL (and SP) fail(s) to limit the nitrogen

Table 2. *Cumulative reward, nitrogen, and yield (median and associated 95% CI).*

Agent	Cumulative reward		Nitrogen (kg/ha)		Yield (tonne/ha)	
Ceres	129.39	(73.34, 136.57)	183.0	(157.2, 211.1)	8.96	(8.41, 9.14)
SP	117.74	(67.47, 132.82)	170.7	(170.7, 170.7)	8.72	(8.13, 8.94)
RL	121.36	(67.96, 133.19)	180.0	(170.0, 200.0)	8.81	(8.24, 9.13)
$\Delta_{RL,SP}$	+3.33	(-1.94, 10.89) p=0.1057	+9.3	(-0.70, 29.3) p=0.0398	+0.13	(-0.01, 0.39) p=0.0290

application, however, resulting in negative cumulative rewards. Figure 2 (right) shows for each test year the difference in yield between the RL agent and the SP agent, as a function of the difference in invested nitrogen. The diagonal shows the break-even line, for which the difference in reward is zero. Most test years, as well as the median, are above the break-even line, demonstrating that the RL agent's decision to apply a different amount of nitrogen is adequate.



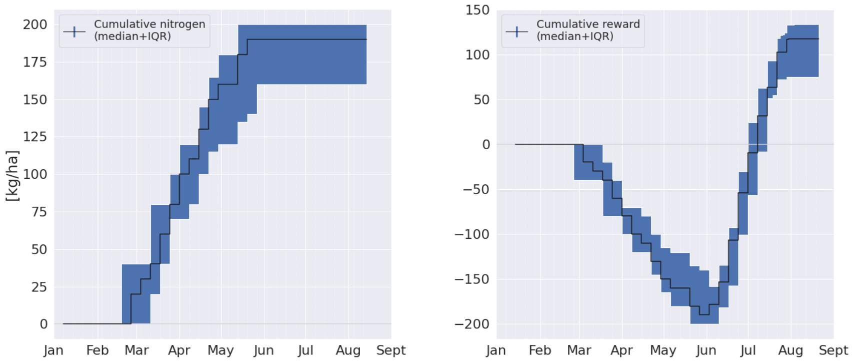
(a) (b) (c)

Figure 2. *(a): cumulative reward obtained, and (b): nitrogen applied by each of the three agents. Each dot depicts a test year (n=16). For most test years RL is closer to Ceres than SP. (c): the difference in yield between the RL agent and the SP agent as a function of the difference in amount of nitrogen applied. The dashed line indicates the break-even line, at which both agents achieve the same reward. Most test years are above the break-even line, demonstrating that the RL agent's choice of applying a different amount of nitrogen is adequate.*

Figure 3 shows the evolvement of the actions and rewards of the RL agent during the course of the growing season, as summarized by the median over the test years. Typically, the RL agent waits until spring for its first actions. The median number of fertilization events is 7.0 (95%CI 6.0-8.0). The median length of an episode is 208 days (95%CI 205-211).

The main driver for applying nitrogen is rainfall. Pearson correlation coefficient between the total amount of rainfall during the growing season and total amount of nitrogen applied is 0.69 (95%CI 0.58-0.76) for Ceres and 0.63 (95%CI 0.12-0.82) for RL (see fig. 4).

Climate resilience In order to assess the robustness of the learned policy against changing climate conditions we deployed both the trained RL and the baseline agents in a more Southern climate. With a climate similarity index of 0.573, as determined with the CCAFS method [38], using average temperature and precipitation as weather variables, the Southern climate differs substantially from the Northern climate. The Southern climate is characterized by a higher average temperature (10.4°C vs



(a)

(b)

Figure 3. *Policy visualization of the RL agent: (a) cumulative reward obtained and (b) nitrogen applied. Typically, the RL agent waits until spring for its first actions..*

9.8°C), less amount of average daily rainfall (1.92mm vs 2.12mm), and a shorter growing season (200 days vs 208 days).

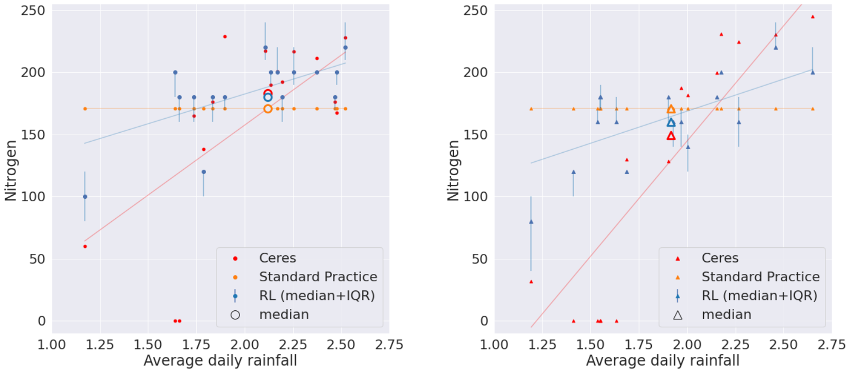
Table 3 reports the performance metrics for each of the three agents deployed in the Southern climate. The maximally achievable cumulative reward and yield, as represented by Ceres, is lower than what is obtained in the Northern climate. In six years, namely 1990, 1992, 1996, 2004, 2006, and 2010, yield is (partially) limited by a low amount of rainfall, resulting in low cumulative rewards. In these dry years the performance of the RL agent is suboptimal, as it does not limit its nitrogen application sufficiently. Yet the cumulative reward RL achieves is competitive with SP. In years with sufficient rainfall the RL agent remains close to the optimal policy, as is illustrated in Figure 4, just as we saw for the Northern climate.

Table 3. *Out-of-distribution results: cumulative reward, nitrogen, and yield (median and 95% CI) in Southern climate.*

Agent	Cumulative Reward		Nitrogen (kg/ha)		Yield (tonne/ha)	
Ceres	95.15	(0.0, 157.67)	149.6	(0.0, 202.99)	8.60	(4.69, 9.60)
SP	89.43	(-67.02, 140.23)	170.7	(170.7, 170.7)	8.50	(5.80, 9.13)
RL	78.17	(-49.92, 142.65)	160.0	(140.0, 180.0)	8.45	(5.80, 9.13)
$\Delta_{RL,SP}$	+4.52	(-4.85, 16.79) p=0.1792	-10.70	(-30.7, 9.3) p=0.7244	-0.04	(-0.17, 0.13) p=0.6459

4. Discussion

We presented *CropGym*, an OpenAI Gym environment to study policies for farm management, such as fertilization and irrigation, using process-based crop growth models. We developed a use case on nitrogen fertilization in rain-fed winter wheat. A reinforcement learning agent was trained to find the optimal timings and amounts for applying nitrogen that balance yield and environmental impact. The agent was found to learn close to optimal strategies, competitive with standard practices set by domain experts.



(a)

(b)

Figure 4. *Scatter plot with regression lines of the average daily rainfall and the total amount of nitrogen applied by all three agents for (a) the Northern and (b) the Southern climate. The optimal amount of nitrogen, as determined by Ceres, depends substantially on rainfall. Presumably, the RL agent has learned to adopt this general trend. In dry years, when lack of rainfall impairs yield and the optimal amount of nitrogen is (close to) zero, the RL agent does not limit its nitrogen application sufficiently, as it arguably sticks to the general trend. In years with sufficient rainfall the RL agent acts in line with the optimal policy. This effect is seen in both the Northern and Southern climate.*

As an out-of-distribution test we evaluated whether the obtained policies were resilient against a change in climate conditions, with sound results. Yet, in years where yield is limited by a shortage of rainfall, performance of the RL agent was suboptimal. The adoption of more dry weather data in the training through e.g. fine-tuning approaches, may improve these results. Other examples of out-of-distribution tests with practical impact include e.g. variations in soil characteristics, such as organic matter content.

Clearly, as is common in RL research, our experiments are done in silico, and it is an open question to what extent our results transfer into the real world. (Crop growth) models are by definition simplifications of reality, and thus policies derived from these models are inherently subject to a simulation-to-reality-gap. Narrowing this gap can be achieved by employing an ensemble of different crop growth models [39]. *CropGym* supports such a strategy by offering implementations of a variety of process-based crop growth models.

To further bridge the gap between simulation and reality, digital twin technology could be exploited [40]. A variety of sensors can be employed to synchronize digital representations of crops with their physical counterparts [41, 42]. Yet, acquisition of sensor data may come with high (monetary) costs. In this context *CropGym* could be utilized to train agents that are able to determine when and to what extent the environment should be measured [43]. In such a training the agent chooses between either relying on the simulated state of the crop, or paying the cost to measure the true state and update the crop growth model accordingly.

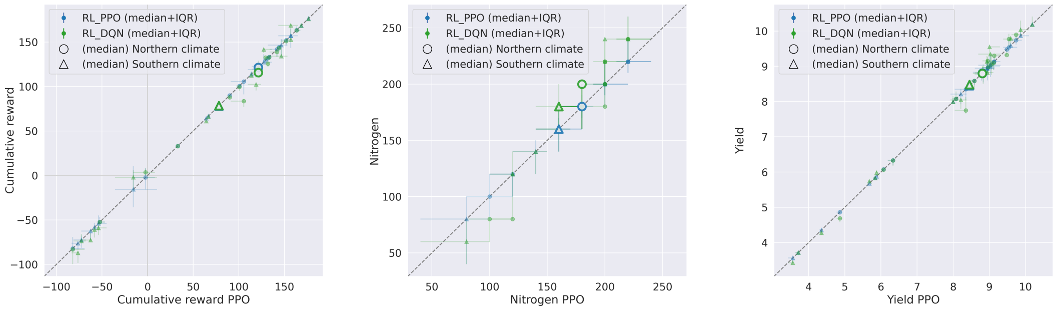
In this work we incentivize the RL agent to generate environmentally friendly policies by negotiating the environmental costs of nitrogen application in the reward function. An alternative approach would be to set hard constraints on the total amount of nitrogen applied. Such could be achieved by building on the works in the domain of (safety-) constrained RL [44], supported by e.g. OpenAI's dedicated Safety Gym benchmark suite [45]. Another constraint that could be considered is the number of fertilization events.

As an open simulation environment *CropGym* can be used to discover adaptive, data driven policies that perform well across a range of plausible scenarios for the future. With *CropGym* we aim to facilitate a joint research effort from the RL and agronomy communities to meet the challenges of future agricultural decision-making and to further match farmers' decision-making processes.

A. Appendix. Results DQN

In addition to training with PPO, we explored training a Deep Q-Network (DQN) [37]. Configurations were kept the same as with PPO. We used the default settings of the (hyper)parameters, as set by Stable Baselines, except for (1) the number of hidden units, which was set to 128x128, (2) the activation function, which was set to tanh, and (3) *exploration_final_eps*, which was set to 0.01. Training ran for 400,000 timesteps.

Below we report the key results, aggregated over 5 runs with different random seeds. Figure 5 demonstrates that for each test year (a) the cumulative reward, (b) the amount of fertilizer, and (c) yield obtained by the DQN agent closely resembles those of the PPO agent. Table 4 shows that, similar to RL_{PPO}, also RL_{DQN} achieves results competitive with SP.



(a) (b) (c)

Figure 5. *Scatter plot of PPO and DQN agent for (a): cumulative reward obtained, (b): nitrogen applied, and (c): yield obtained. Each point depicts a test year (n=32).*

Table 4. *Cumulative reward, nitrogen, and yield (median and associated 95% CI) for DQN.*

Agent	Cumulative Reward		Nitrogen (kg/ha)		Yield (tonne/ha)	
Northern climate						
RL$_{DQN}$	115.60	(59.73, 132.75)	200.0	(170.0, 210.0)	8.80	(8.18, 9.21)
$\Delta_{RL_{DQN},SP}$	+0.38	(-5.05, 9.87)	+29.30	(-0.70, 39.30)	+0.16	(-0.04, 0.41)
		p=0.4822		p=0.0355		p=0.0360
Southern climate						
RL$_{DQN}$	78.46	(-54.88, 144.71)	180.0	(150.0, 190.0)	8.48	(5.83, 9.33)
$\Delta_{RL_{DQN},SP}$	+1.34	(-6.85, 15.83)	+9.30	(-20.70, 19.30)	+0.02	(-0.18, 0.15)
		p=0.4043		p=0.4734		p=0.4726

Acknowledgments. We would like to thank Allard de Wit and Herman Berghuijs for discussion and revision of the employed crop growth model, and Lotte Woittiez for the constructive discussions during various stages of the research.

Funding Statement. This work has been partially supported by the European Union Horizon 2020 Research and Innovation program (Project code: 101070496, Smart Droplets), and the Wageningen University and Research Investment Programme "Digital Twins".

Competing Interests. None.

Data and Code Availability Statement. Data and replication code can be found at https://www.cropgym.ai.

Ethical Standards. The research meets all ethical guidelines, including adherence to the legal requirements of the study country.

Author Contributions. Conceptualization: M.K.; H.O.; I.A. Methodology: M.K.; I.A. Software: M.K. H.O.; R.B. Data visualisation: M.K. Validation: M.K.; I.A. Writing - original draft: M.K. Writing - review & editing: I.A; H.O.; R.B. All authors approved the final submitted draft.

Supplementary Material. No supplementary material has been provided with the submission.

References

1. Verónica Saiz-Rubio and Francisco Rovira-Más. From smart farming towards agriculture 5.0: A review on crop data management. *Agronomy*, 10(2), 2020. ISSN 2073-4395. doi: 10.3390/agronomy10020207. URL https://www.mdpi.com/2073-4395/10/2/207.
2. Zhaoyu Zhai, José Fernán Martínez, Victoria Beltran, and Néstor Lucas Martínez. Decision support systems for agriculture 4.0: Survey and challenges. *Computers and Electronics in Agriculture*, 170:105256, 2020. ISSN 0168-1699. doi: https://doi.org/10.1016/j.compag.2020.105256. URL https://www.sciencedirect.com/science/article/pii/S0168169919316497.
3. Simone Graeff, Johanna Link, Jochen Binder, and Wilhelm Claupein. Crop models as decision support systems in crop production. *Crop production technologies*, pages 3–28, 2012.
4. Arianna Di Paola, Riccardo Valentini, and Monia Santini. An overview of available crop growth and yield models for studies and assessments in agriculture. *Journal of the Science of Food and Agriculture*, 96(3):709–714, 2016. doi: https://doi.org/10.1002/jsfa.7359. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/jsfa.7359.
5. James W. Jones, John M. Antle, Bruno Basso, Kenneth J. Boote, Richard T. Conant, Ian Foster, H. Charles J. Godfray, Mario Herrero, Richard E. Howitt, Sander Janssen, Brian A. Keating, Rafael Munoz-Carpena, Cheryl H. Porter, Cynthia Rosenzweig, and Tim R. Wheeler. Brief history of agricultural systems modeling. *Agricultural Systems*, 155:240–254, 2017. ISSN 0308-521X. doi: https://doi.org/10.1016/j.agsy.2016.05.014. URL https://www.sciencedirect.com/science/article/pii/S0308521X16301585.
6. Dean P. Holzworth, Neil I. Huth, Peter G. deVoil, Eric J. Zurcher, Neville I. Herrmann, Greg McLean, Karine Chenu, Erik J. van Oosterom, Val Snow, Chris Murphy, Andrew D. Moore, Hamish Brown, Jeremy P.M. Whish, Shaun Verrall, Justin Fainges, Lindsay W. Bell, Allan S. Peake, Perry L. Poulton, Zvi Hochman, Peter J. Thorburn, Donald S. Gaydon, Neal P. Dalgliesh, Daniel Rodriguez, Howard Cox, Scott Chapman, Alastair Doherty, Edmar Teixeira, Joanna Sharp, Rogerio Cichota, Iris Vogeler, Frank Y. Li, Enli Wang, Graeme L. Hammer, Michael J. Robertson, John P. Dimes, Anthony M. Whitbread, James Hunt, Harm van Rees, Tim McClelland, Peter S. Carberry, John N.G. Hargreaves, Neil MacLeod, Cam McDonald, Justin Harsdorf, Sara Wedgwood, and Brian A. Keating. Apsim – evolution towards a new generation of agricultural systems simulation. *Environmental Modelling & Software*, 62:327–350, 2014. ISSN 1364-8152. doi: https://doi.org/10.1016/j.envsoft.2014.07.009. URL https://www.sciencedirect.com/science/article/pii/S1364815214002102.
7. J.W Jones, G Hoogenboom, C.H Porter, K.J Boote, W.D Batchelor, L.A Hunt, P.W Wilkens, U Singh, A.J Gijsman, and J.T Ritchie. The dssat cropping system model. *European Journal of Agronomy*, 18(3):235–265, 2003. ISSN 1161-0301. doi: https://doi.org/10.1016/S1161-0301(02)00107-7. URL https://www.sciencedirect.com/science/article/pii/S1161030102001077. Modelling Cropping Systems: Science, Software and Applications.
8. Allard de Wit. The python crop simulation environment, 2023. URL https://pcse.readthedocs.io/en/stable/.
9. Melvin Eldho Shibu, Peter A. Leffelaar, Herman Van Keulen, and Pramila Aggarwal. Lintul3, a simulation model for nitrogen-limited situations: Application to rice. *European Journal of Agronomy*, 32:255–271, 2010.
10. Allard de Wit, Hendrik Boogaard, Davide Fumagalli, Sander Janssen, Rob Knapen, Daniel van Kraalingen, Iwan Supit, Raymond van der Wijngaart, and Kees van Diepen. 25 years of the wofost cropping systems model. *Agricultural Systems*, 168:154–167, 2019. ISSN 0308-521X. doi: https://doi.org/10.1016/j.agsy.2018.06.018. URL https://www.sciencedirect.com/science/article/pii/S0308521X17310107.
11. Marisa Gallardo, Antonio Elia, and Rodney B. Thompson. Decision support systems and models for aiding irrigation and nutrient management of vegetable crops. *Agricultural Water Management*, 240:106209, 2020. ISSN 0378-3774. doi: https://doi.org/10.1016/j.agwat.2020.106209. URL https://www.sciencedirect.com/science/article/pii/S0378377420303267.
12. Jonathan Binas, Leonie Luginbuehl, and Yoshua Bengio. Reinforcement learning for sustainable agriculture. ICML 2019 Workshop Climate Change: How Can AI Help, 2019.
13. Lu Wang, Xiaofeng He, and Dijun Luo. Deep reinforcement learning for greenhouse climate control. In *2020 IEEE International Conference on Knowledge Graph (ICKG)*, pages 474–480. IEEE, 2020.

14. Yahav Avigal, William Wong, Mark Presten, Mark Theis, Shrey Aeron, Anna Deza, Satvik Sharma, Rishi Parikh, Sebastian Oehme, Stefano Carpin, Joshua H. Viers, Stavros Vougioukas, and Ken Y. Goldberg. Simulating polyculture farming to learn automation policies for plant diversity and precision irrigation. *IEEE Transactions on Automation Science and Engineering*, 19(3):1352–1364, 2022. doi: 10.1109/TASE.2021.3138995.
15. Hiske Overweg, Herman NC Berghuijs, and Ioannis N Athanasiadis. Cropgym: a reinforcement learning environment for crop management. *ICLR Workshop Modeling Oceans and Climate Change*, 2021.
16. Mengting Chen, Yuanlai Cui, Xiaonan Wang, Hengwang Xie, Fangping Liu, Tongyuan Luo, Shizong Zheng, and Yufeng Luo. A reinforcement learning approach to irrigation decision-making for rice using weather forecasts. *Agricultural Water Management*, 250:106838, 2021. ISSN 0378-3774. doi: https://doi.org/10.1016/j.agwat.2021.106838. URL https://www.sciencedirect.com/science/article/pii/S0378377421001037.
17. Ran Tao, Pan Zhao, Jing Wu, Nicolas F Martin, Matthew T Harrison, Carla Ferreira, Zahra Kalantari, and Naira Hovakimyan. Optimizing crop management with reinforcement learning and imitation learning. *arXiv preprint arXiv:2209.09991*, 2022.
18. Yuji Saikai, Allan Peake, and Karine Chenu. Deep reinforcement learning for irrigation scheduling using high-dimensional sensor feedback, 2023. URL https://arxiv.org/abs/2301.00899.
19. Ahmad Din, Muhammed Yousoof Ismail, Babar Shah, Mohammad Babar, Farman Ali, and Siddique Ullah Baig. A deep reinforcement learning-based multi-agent area coverage control for smart agriculture. *Computers and Electrical Engineering*, 101:108089, 2022. ISSN 0045-7906. doi: https://doi.org/10.1016/j.compeleceng.2022.108089. URL https://www.sciencedirect.com/science/article/pii/S0045790622003445.
20. Matteo Turchetta, Luca Corinzia, Scott Sussex, Amanda Burton, Juan Herrera, Ioannis N. Athanasiadis, Joachim M. Buhmann, and Andreas Krause. Learning long-term crop management strategies with cyclesgym. NeurIPS, 2022.
21. Romain Gautron, Odalric-Ambrym Maillard, Philippe Preux, Marc Corbeels, and Régis Sabbadin. Reinforcement learning for crop management support: Review, prospects and challenges. *Computers and Electronics in Agriculture*, 200:107182, 2022. ISSN 0168-1699. doi: https://doi.org/10.1016/j.compag.2022.107182. URL https://www.sciencedirect.com/science/article/pii/S0168169922004999.
22. Romain Gautron, Emilio José Padron Gonzalez, Philippe Preux, Julien Bigot, Odalric-Ambrym Maillard, and David Emukpere. *gym-DSSAT: a crop model turned into a Reinforcement Learning environment*. PhD thesis, Inria Lille, 2022.
23. Gerrit Hoogenboom, Cheryl H Porter, Kenneth J Boote, Vakhtang Shelia, Paul W Wilkens, Upendra Singh, Jeffrey W White, Senthold Asseng, Jon I Lizaso, L Patricia Moreno, et al. The dssat crop modeling ecosystem. In *Advances in crop modelling for a sustainable agriculture*, pages 173–216. Burleigh Dodds Science Publishing, 2019.
24. Chace Ashcraft and Kiran Karra. Machine learning aided crop yield optimization. *arXiv preprint arXiv:2111.00963*, 2021.
25. Chuang Zhao, Bing Liu, Liujun Xiao, Gerrit Hoogenboom, Kenneth J Boote, Belay T Kassie, Willingthon Pavan, Vakhtang Shelia, Kwang Soo Kim, Ixchel M Hernandez-Ochoa, et al. A simple crop model. *European Journal of Agronomy*, 104: 97–106, 2019.
26. Malvern Madondo, Muneeza Azmat, Kelsey DiPietro, Raya Horesh, Michael Jacobs, Arun Bawa, Raghavan Srinivasan, and Fearghal O'Donncha. A swat-based reinforcement learning framework for crop management. AAAI Conference on Artificial Intelligence, 2023.
27. JG Arnold, JR Kiniry, R Srinivasan, JR Williams, EB Haney, and SL Neitsch. Soil and water assessment tool input/output file documentation version 2009. Technical report, Texas Water Resources Institute, 2011.
28. Armen R. Kemanian, Charlie M. White, Yuning Shi, Claudio O. Stockle, and Lorne Leonard. Cycles: Agroecosystems model. https://plantscience.psu.edu/research/labs/kemanian/models-and-tools/cycles, 2022.
29. Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym, 2016.
30. Xin Zhang, Eric A Davidson, Denise L Mauzerall, Timothy D Searchinger, Patrice Dumas, and Ye Shen. Managing nitrogen for sustainable development. *Nature*, 528(7580):51–59, 2015.
31. Agri23a. agrimatie.nl. price development of seeds and grains. https://www.agrimatie.nl/ThemaResultaat.aspx?subpubID=2289&themaID=2263, 2023. Accessed: 2023-01-19.
32. Agri23b. agrimatie.nl. price development of fertilizer. https://www.agrimatie.nl/SectorResultaat.aspx?subpubID=2232&sectorID=2233&themaID=2263, 2023. Accessed: 2023-01-19.
33. Wiert Wiertsema. Obtaining winter wheat parameters for lintul from a field experiment. Master's thesis, Wageningen University, the Netherlands, 2015.
34. H.N.C. Berghuijs, J.V. Silva, H.C.A. Rijk, M.K. van Ittersum, F.K. van Evert, and P. Reidsma. Catching-up with genetic progress: Simulation of potential production for modern wheat cultivars in the netherlands. *Field Crops Research*, 296: 108891, 2023. ISSN 0378-4290. doi: https://doi.org/10.1016/j.fcr.2023.108891. URL https://www.sciencedirect.com/science/article/pii/S0378429023000849.
35. John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *CoRR*, abs/1707.06347, 2017. URL http://arxiv.org/abs/1707.06347.
36. Antonin Raffin, Ashley Hill, Adam Gleave, Anssi Kanervisto, Maximilian Ernestus, and Noah Dormann. Stable-baselines3: Reliable reinforcement learning implementations. *Journal of Machine Learning Research*, 22(268):1–8, 2021. URL http://jmlr.org/papers/v22/20-1364.html.
37. Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin A. Riedmiller. Playing atari with deep reinforcement learning. *CoRR*, abs/1312.5602, 2013. URL http://arxiv.org/abs/1312.

5602.

38. Julián Ramírez Villegas, Charlotte Lau, Ann-Kristin Köhler, Andy Jarvis, NP Arnell, Tom M Osborne, and Josh Hooker. Climate analogues: finding tomorrow's agriculture today. *CCAFS Working Paper*, 2011.
39. Daniel Wallach, Pierre Martre, Bing Liu, Senthold Asseng, Frank Ewert, Peter J. Thorburn, Martin van Ittersum, Pramod K. Aggarwal, Mukhtar Ahmed, Bruno Basso, Christian Biernath, Davide Cammarano, Andrew J. Challinor, Giacomo De Sanctis, Benjamin Dumont, Ehsan Eyshi Rezaei, Elias Fereres, Glenn J. Fitzgerald, Y. Gao, Margarita Garcia-Vila, Sebastian Gayler, Christine Girousse, Gerrit Hoogenboom, Heidi Horan, Roberto C. Izaurralde, Curtis D. Jones, Belay T. Kassie, Kurt C. Kersebaum, Christian Klein, Ann-Kristin Koehler, Andrea Maiorano, Sara Minoli, Christoph Müller, Soora Naresh Kumar, Claas Nendel, Garry J. O'Leary, Taru Palosuo, Eckart Priesack, Dominique Ripoche, Reimund P. Rötter, Mikhail A. Semenov, Claudio Stöckle, Pierre Stratonovitch, Thilo Streck, Iwan Supit, Fulu Tao, Joost Wolf, and Zhao Zhang. Multimodel ensembles improve predictions of crop–environment–management interactions. *Global Change Biology*, 24(11): 5072–5083, 2018. doi: https://doi.org/10.1111/gcb.14411. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/gcb.14411.
40. Christos Pylianidis, Sjoukje Osinga, and Ioannis Athanasiadis. Introducing digital twins to agriculture. *Computers and Electronics in Agriculture*, 184:105942, 05 2021. doi: 10.1016/j.compag.2020.105942.
41. Xiuliang Jin, Lalit Kumar, Zhenhai Li, Haikuan Feng, Xingang Xu, Guijun Yang, and Jihua Wang. A review of data assimilation of remote sensing and crop models. *European Journal of Agronomy*, 92:141–152, 2018. ISSN 1161-0301. doi: https://doi.org/10.1016/j.eja.2017.11.002. URL https://www.sciencedirect.com/science/article/pii/S1161030117301685.
42. Keiji Jindo, Osamu Kozan, and Allard de Wit. Data assimilation of remote sensing data into a crop growth model. In *Precision Agriculture: Modelling*, pages 185–197. Springer, 2023.
43. Colin Bellinger, Andriy Drozdyuk, Mark Crowley, and Isaac Tamblyn. Scientific discovery and the cost of measurement - balancing information and cost in reinforcement learning. *CoRR*, abs/2112.07535, 2021. URL https://arxiv.org/abs/2112.07535.
44. Yongshuai Liu, Avishai Halev, and Xin Liu. Policy learning with constraints in model-free reinforcement learning: A survey. In *IJCAI*, pages 4508–4515, 2021.
45. Alex Ray, Joshua Achiam, and Dario Amodei. Benchmarking safe exploration in deep reinforcement learning. *arXiv preprint arXiv:1910.01708*, 7:1, 2019.