THÈSE POUR OBTENIR LE GRADE DE DOCTEUR DE L'INSTITUT AGRO MONTPELLIER ET DE L'UNIVERSITÉ DE MONTPELLIER

En apprentissage automatique appliqué à l'agronomie

École doctorale GAIA – Biodiversité, Agriculture, Alimentation, Environnement, Terre, Eau
Portée par

Unité de recherche « agroécologie et intensification durable des cultures annuelles » (AIDA) du CIRAD

Apprentissage par renforcement pour l'aide à la conduite des cultures des petits agriculteurs des pays du Sud : vers la maîtrise des risques

Reinforcement learning for crop management support to smallholder farmers in countries of the South: towards risk management

Par Romain GAUTRON

Thèse débutée le 01 novembre 2019 et présentée le 09 décembre 2022

Sous la direction de Marc CORBEELS

Devant le jury composé de

Jean-Noël AUBERTOT, directeur de recherche, INRAE Toulouse	rapporteur
David MAKOWSKI, directeur de recherche, INRAE Paris-Saclay	rapporteur
Audrey DURAND, professeure adjointe, Université Laval	examinatrice
Ronan TREPOS, ingénieur de recherche, INRAE Toulouse	examinateur
Marc CORBEELS, chercheur, CIRAD	directeur de thèse
Philippe PREUX, professeur, Université de Lille	co-encadrant
Odalric-Ambrym MAILLARD, chercheur, Inria	co-encadrant

UNIVERSITÉ DE MONTPELLIER

L'INSTITUT agro Montpellier

Abstract

Crop management is the logical and ordered combination of agricultural operations applied to a field in order to obtain a particular crop production. Decisions about these operations are not straightforward as they occur in the face of uncertain events, such as weather events. After decades of development of computerized decision-making tools for crop management support, these specialized decision support systems (DSS) are still facing a poor adoption. DSS users deemed that information cannot directly be turned into actions, that farmers' natural decision-making processes are not adequately taken into account, that the sequential nature of decisions is poorly modeled or that risk management is lacking in the decision process.

Reinforcement learning (RL), a branch of machine learning, addresses the control of uncertain and unknown dynamical systems. RL inherently deals with sequences of decisions with uncertain consequences, and shares some similarities with how farmers are described to address crop management, e.g. learning by trial and errors. Yet, very few applications of RL for crop management support are found. RL generally requires millions of interactions to solve simple decision problems compared to crop management. In this thesis, we study how RL can improve the decision support of crop management, focusing on smallholder farmers of southern regions. In this context, crop management support is even more challenging because of the data scarcity and high yield variability in rainfed cropping systems.

We provide a generic method to turn crop models into standardized and easy to manipulate RL environments, which allow to extensively train RL agents at a negligible computational cost. In simulated conditions, we successfully learn sustainable crop practices with an RL algorithm. Yet, we show that for most applications, considering both a risk-neutral and risk-aware decision criterion, the statistical significance of the identification of best practices from model simulations to reality is unlikely to be supported by enough statistical evidences.

We then consider the collaborative identification of best management practices by a group of farmers performing on-farm trials. In a simulated exercise, we mimic the growing conditions of Southern Mali. We design an identification method based on a multi-armed bandit algorithm, a special case of RL, using a risk-aware decision criterion, with the constraint of minimizing farmers' crop yield losses occurring during this identification. By leveraging the expert knowledge to reduce the sample complexity of the decision problem, the identification method can be realistically employed in real conditions, and in most cases is better at reducing farmers' yield losses than equi-proportional field trials of each crop operation during a fixed number of years.

Résumé

Un itinéraire technique est défini comme la suite logique et ordonnée d'opérations culturales appliquées à une parcelle dans le but d'atteindre des objectifs de production donnés. Ces séquences de décisions d'opérations culturales ne sont pas triviales, du fait qu'elles font face à des évènements incertains, comme les évènements météorologiques. Après plusieurs décennies de développement de logiciels informatiques dédiés à l'aide à la décision pour les itinéraires techniques, ces logiciels (*decision support systems* en anglais) sont toujours peu adoptés en pratique. Les utilisateurs ont jugé que l'information ne peut pas être directement traduite en actions, que les processus cognitifs des agriculteurs ne sont pas bien pris en compte, que le caractère séquentiel des prises de décision n'est pas bien modélisé ou encore que la gestion du risque dans les décisions manque.

L'apprentissage par renforcement (AR) est un domaine de l'apprentissage automatique qui s'attache au contrôle des systèmes dynamiques, incertains et inconnus. L'AR traite de manière inhérente avec de séquences d'actions aux conséquences incertaines, et partage des similarités avec la manière dont les agriculteurs abordent la conduite des cultures, e.g. apprentissage par essai-erreur. Cependant, la littérature montre très peu d'applications de l'AR pour la conduite des cultures. L'AR requiert généralement des millions d'interactions pour résoudre des problèmes simples comparés à celui de la conduite des cultures. Nous étudions comment l'AR peut améliorer la prise de décision pour les itinéraires techniques, en particulier pour les petits agriculteurs des régions du Sud. Dans ce contexte, l'aide à la conduite des cultures est ardue, du fait de la faible disponibilité des données et de la grande variabilité des rendements dans les systèmes non irrigués.

Nous proposons une méthode générique pour convertir des modèles de culture en environnements d'apprentissage par renforcement faciles à manipuler et standardisés. Ces environnements permettent d'entraîner des agents d'AR avec un coût de calcul négligeable. En conditions simulées, à l'aide d'un algorithme d'AR, nous apprenons avec succès des pratiques durables de conduite des cultures. Cependant, nous montrons que, pour la plupart des applications, la signification statistique de l'identification d'une meilleure pratique pour les conditions réelles au champ en se basant sur les simulations est peu probablement appuyée par des preuves statistiques suffisantes. Nous avons considéré à la fois un critère de décision neutre face au risque et un critère avec aversion au risque.

Nous nous attachons enfin à l'identification collaborative des meilleures opérations culturales par un groupe d'agriculteurs conduisant des essais au champ. Dans un exercice simulé, nous reproduisons les conditions de culture de Sud du Mali. Nous concevons une méthode d'identification des meilleures opérations culturales à l'aide d'un algorithme de bandit à plusieurs bras, un cas particulier d'AR, avec un critère de décision avec aversion au risque. L'algorithme a la contrainte de minimiser les pertes accumulées par les agriculteurs durant le processus d'identification. En tirant parti des connaissances d'experts afin de réduire la complexité du problème de décision, nous montrons que la méthode d'identification avec l'algorithme de bandit pourrait être appliquée en conditions réelles. Par ailleurs, ladite méthode réduit davatange les pertes des agriculteurs dans la plupart des cas, comparé à la méthode classique qui consiste en des essais au champ équiproportionnels de chaque opération culturale durant un nombre fixe d'années.

Contents

General introduction **1**

1 Reinforcement learning for crop management support: review, prospects and challenges* **5**
1.1 Introduction . . . 6
1.2 Reinforcement learning . . . 9
1.2.1 Overview of reinforcement learning . . . 9
1.2.2 Formalization of a reinforcement learning problem . . . 10
1.2.3 A brief historical perspective of reinforcement learning . . . 12
1.2.4 Q-learning: a simple reinforcement learning algorithm . . . 13
1.2.5 Reinforcement learning today . . . 13
1.2.6 Multi-armed bandit . . . 15
1.3 Review of reinforcement learning for agriculture . . . 15
1.3.1 Early stirrings: farm decision-making under uncertainty . . . 15
1.3.2 Seminal works using reinforcement learning in agriculture . . . 16
1.3.3 Deep reinforcement learning applications . . . 17
1.3.4 Multi-armed bandits . . . 19
1.3.5 RL applications in other domains . . . 19
1.4 Prospects and challenges . . . 19
1.4.1 An RL-based crop management DSS . . . 20
1.4.2 Prospects . . . 22
1.4.3 Challenges . . . 23
1.5 Conclusions . . . 26

2 `gym-DSSAT`: a crop model turned into a reinforcement learning environment† **29**
2.1 Introduction . . . 30
2.2 Related work . . . 33
2.3 Formalization of RL decision problems . . . 34
2.3.1 From Markov decision processes to reinforcement learning . . . 34
2.3.2 Partially observable Markov decision process . . . 35
2.3.3 `gym` environments . . . 36
2.4 Decisions problems in `gym-DSSAT` . . . 37
2.4.1 Default crop management problems of `gym-DSSAT` . . . 37
2.4.2 Custom scenario definition . . . 40
2.5 Software architecture of the environment . . . 40
2.5.1 The PDI Data Interface . . . 40
2.5.2 Internals of `gym-DSSAT` . . . 41
2.6 Experimenting with `gym-DSSAT` . . . 43

*Article published in Computers and Electronics in Agriculture (Elsevier).
†Article published as an Inria Research Report and a short version was accepted for a poster and an oral presentation at the AAAI-23 conference, *AI for Agriculture and Food Systems* workshop.

2.6.1 Use case: learning an efficient maize fertilization . . . 43
2.6.2 Execution time and reproducibility . . . 48
2.7 Conclusion . . . 49

3 Quantifying the uncertainty of decisions based on crop model simulations[‡] 52
3.1 Introduction . . . 53
3.2 Methods . . . 54
3.2.1 Mean-variance: a risk-aware metric . . . 54
3.2.2 Confidence interval comparison . . . 55
3.2.3 Use case . . . 60
3.3 Results . . . 62
3.3.1 Model evaluation . . . 62
3.3.2 Hypothesis testing . . . 62
3.3.3 Uncertainty of simulated yield distributions . . . 64
3.3.4 Uncertainty of model error distribution . . . 65
3.3.5 Minimal risk level for decision . . . 65
3.4 Discussion . . . 66
3.4.1 Validity of the statistical analyses . . . 66
3.4.2 Comparison to existing quantification of crop model errors . . . 69
3.4.3 Implications for decisions made from model simulations . . . 70
3.5 Conclusion . . . 70

4 Towards an efficient and risk aware strategy to guide farmers in identifying best crop management[§] 74
4.1 Introduction . . . 75
4.2 Methods . . . 77
4.2.1 Virtual crop management identification problem . . . 77
4.2.2 Identification of the best fertilizer practices . . . 82
4.3 Results . . . 86
4.3.1 Simulated responses to nitrogen fertilizer practices . . . 86
4.3.2 Identification of best fertilizer practices . . . 87
4.4 Discussion . . . 91
4.4.1 Benefits from an adaptive identification strategy. . . . 91
4.4.2 Performances of fertilizer practices . . . 92
4.4.3 Definition of farmers' objective . . . 93
4.4.4 Limits and possible improvements . . . 93

General discussion 95

General conclusions 100

[‡]Article to be submitted to Environmental modeling & Software (Elsevier).
[§]Article to be submitted to Computers and Electronics in Agriculture (Elsevier).

Supplementary Materials A (corresponds to Chapter 2) **102**
A.1 Irrigation use case . . . 102
A.2 Fertilization use case complement . . . 104

Supplementary Materials B (corresponds to Chapter 3) **106**
B.1 Uncertainty vocabulary. . . . 106
B.2 Biased model error . . . 106
B.3 Confidence intervals . . . 107
B.3.1 Union bounds . . . 107
B.3.2 The case of Gaussian distributions . . . 108
B.3.3 The case of bounded distributions . . . 109
B.3.4 Second-order sub-Gaussian distributions . . . 113
B.4 Interval disjunction algorithmic search . . . 116
B.5 Cultivar parameters in model simulations. . . . 116
B.6 Hypothesis testing . . . 117
B.6.1 Simulated yields induced by weather generation . . . 117
B.6.2 Residuals . . . 118
B.7 Minimal risk level for interval disjunction . . . 122
B.7.1 Minimal risk level for confidence interval disjunction. . . . 122
B.7.2 Choice of the risk-aware metric . . . 124

Supplementary Materials C (corresponds to Chapter 4) **125**
C.1 Maize simulations . . . 125
C.2 Alternative performance measure of fertilization practices . . . 127
C.3 Algorithms . . . 127
C.3.1 Details about BCB . . . 127
C.3.2 Explore-Then-Commit (ETC) . . . 130
C.4 Experiment complements . . . 131
C.5 Theoretical Analysis . . . 131

Résumé opérationnel **138**

Publications and software **146**

Acknowledgements **147**

Glossary **151**

List of Figures **153**

List of Tables **158**

List of Algorithms **160**

Bibliography **161**

General introduction

It is well acknowledged that sustainable intensification (SI) of farming systems is required at the global level to meet an increasing food demand (FAO et al., 2017). SI encompasses several dimensions (Pretty et al., 2011): greater production outputs per unit of capital (e.g. land, labour), preservation and enhancement of ecosystemic services and increased resilience of the production systems in the face of perturbations (e.g. economic, or climatic). African smallholder farmers expect an increased farm productivity as an immediate benefit of SI (Vanlauwe et al., 2014). Improved crop management (e.g. the use of improved cultivars or fertilization practices) is a priority entry point to reduce yield variability, increase crop productivity (Tittonell and Giller, 2013), and to mitigate the negative effects of climate change (e.g. Adam et al., 2020, in the Sudano-Sahelian zone).

Data-driven decision-making tools have been used in agriculture, including for crop management support, for a long time. For instance, an early statistical analysis of error measures for agronomic field trials by Mercer and Hall (1911) can be traced back to 1911. In 1955, Tintner (1955) formalized a cropping plan decision as an advanced optimization problem, taking into account the uncertainty of the decision problem. Since then, agronomy research has experienced decades of a rich history of computerized decision-making tools (Jones et al., 2017), called decision support systems (DSS). A DSS generally aims at improving human decision making for unstructured, or semi-structured decision problems. Such problems have incomplete or uncertain information with possibly unforeseen events and complex trade-offs between different objectives (Arnott and Pervan, 2005; Power, 2008), as in the case for crop management. Crop management DSS are mostly found for fertilization, irrigation, pest and disease or weed management; the end users may be researchers, local advisers or farmers. With increasing amounts of real-time data, including on-farm data from field sensors, combined with analytic advances, so-called smart systems are expected to produce disruptive changes in the whole agricultural value chain and food systems (Tzachor et al., 2022; Wolfert et al., 2017). Machine learning (ML) models, i.e. computer programs designed to perform a task and able to self-improve with data or experience (Mitchell et al., 1997), are increasingly incorporated into agricultural DSS (e.g. Liakos et al., 2018; Zhai et al., 2020).

To date, agricultural DSS are still facing poor adoption (Evans et al., 2017; Hochman and Carberry, 2011; McCown, 2002a,b; Rose et al., 2016), and it is likely that future ML-based systems will face similar implementation barriers. For instance, DSS users assessed that DSS information cannot directly be turned into actions, that farmers' natural decision-making processes are not adequately taken into account, that the sequential nature of decisions is poorly modeled or that risk management is lacking in the decision process. Countries of the southern regions have additional constraints to agricultural decision support. Limited field data is available, for instance Africa lacks granular soil data for modeling purposes (Han et al., 2019). Besides, climate change is expected to heavily impact African agricultural production (Adhikari et al., 2015; Sultan et al., 2013), with already fragile farming systems. Farmers face many risks, and climatic risk is an important one (Huet et al., 2020). Indeed, most of crops are rainfed, and harvests are consequently greatly conditioned by the weather uncertainty (Mertz et al., 2011).

Research question

How reinforcement learning, a special branch of machine learning concerned with sequential decision-making under uncertainty, can improve the decision support of crop management in the case of smallholder farmers?

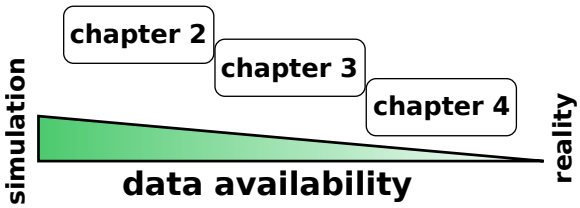


Figure i.1 Level of data availability that each chapter considers for the evaluation of crop operations. In chapter 2, the purely simulated conditions allow to explore crop operations millions of times. Chapter 3 addresses the quantification of the statistical guarantees of decisions, from crop model simulations to real-field conditions. Millions of simulated results of crop operations are possible, but the final results are also constrained by the availability of real-field data. In chapter 4, we target a realistic number of crop operation trials for the learning method to be applicable in real conditions.

In particular, we explore how reinforcement learning (RL) can better manage the risk in crop management decisions (e.g. yield loss avoidance), compared to existing decision methods. This study is limited to the design of *ad-hoc* RL-based decision models to support crop management decisions. We neither address the design of user interfaces, nor directly address the practical questions of DSS implementation.

Outline Chapter 1 introduces RL to non-specialists, provides a review of the applications of RL for crop management support, and a roadmap for the design of the next human-centered RL-based crop management DSS. The remaining chapters consider decreasing levels of data availability to explore the crop operations, from almost infinite data in purely simulated settings, to highly constrained data availability in realistic settings, see Figure i.1. In Chapter 2, we provide a generic method to turn Fortran/C/C^{++} crop models into standardized and easy to use crop management RL environments, and we show preliminary results of a successful RL-based learning of sustainable crop fertilization practices in simulated conditions. Then, in Chapter 3, for a best management practice being identified (using RL or other optimization methods) amongst a larger set of practices based on crop model simulations, we address the general question of the quantification of the statistical guarantees of this identification, from the crop model simulations to the real field conditions. We quantify the statistical guarantees of decisions for both risk-neutral and risk-aware decision criterion. We present a use case with a long-term maize experiment in Canada, a very favorable data context for the crop model calibration. We then discuss the implications for the less favorable data contexts as commonly found in countries of the South. In Chapter 4, we address the collaborative identification of the best maize nitrogen fertilization practices, supported by the field trials of a group of farmers. We introduce an identification strategy using a bandit algorithm, a special case of RL adapted to problems with limited data, for risk-aware decisions based on the work of Baudry et al. (2021a)[¶]. We test our approach with crop model simulations that mimic the growing conditions of southern Mali, using the RL environment introduced in Chapter 2. We use these simulations to test our identification method. Since, the method does not depend on simulations. The ultimate aim is to directly learn from real-field trials performed by the farmers. Finally, we discuss the results of this entire study, open up on perspectives, and conclude.

[¶]This publication was co-authored during this Ph.D.

Chapter 1 provides a literature review of studies that have applied RL for crop management support and defines a conceptual framework for future applications. This framework then guides the subsequent chapters of this thesis, which explore some of the research directions pointed out in Chapter 1.

Chapter 1

Reinforcement learning for crop management support: review, prospects and challenges*

Romain Gautron [† ‡ §] Odalric-Ambrym Maillard [¶] Philippe Preux [||]
Marc Corbeels [† ‡ **] Régis Sabbadin [††]

*Article published in Computers and Electronics in Agriculture (Elsevier).
†AIDA, Univ Montpellier, France.
‡CIRAD, Montpellier, France.
§CGIAR Platform for Big Data in Agriculture, Alliance of Bioversity International and CIAT, Km 17, Recta Cali-Palmira 763537, Colombia.
¶Université de Lille, Inria, CNRS, Centrale Lille UMR 9189 – CRIStAL, F-59000 Lille, France.
||Université de Lille, CNRS, Inria, F-59650 Villeneuve d'Ascq, France.
**International Institute of Tropical Agriculture, PO Box 30772, Nairobi, 00100, Kenya.
††Université de Toulouse, INRAE, UR MIAT, F-31320, Castanet-Tolosan, France.

Abstract

Reinforcement learning (RL), including multi-armed bandits, is a branch of machine learning that deals with the problem of sequential decision-making in uncertain and unknown environments through learning by practice. While best known for being the core of the artificial intelligence (AI) world's best Go game player, RL has a vast range of potential applications. RL may help to address some of the criticisms leveled against crop management decision support systems (DSS): it is an interactive, geared towards action, contextual tool to evaluate series of crop operations faced with uncertainties. A review of RL use for crop management DSS reveals a limited number of contributions. We profile key prospects for a human-centered, real-world, interactive RL-based system to face tomorrow's agricultural decisions, and theoretical and ongoing practical challenges that may explain its current low uptake. We argue that a joint research effort from the RL and agronomy communities is necessary to explore RL's full potential.

1.1 Introduction

Reinforcement learning (RL), a branch of machine learning and more generally artificial intelligence (AI), addresses the control of uncertain and unknown dynamical systems. Although information about recent research in RL is widely available, it is too specialized and abstract to be easily understandable (Lapan, 2018, preface). RL is potentially a well suited paradigm to support crop management decisions, but few applications are found in the literature. This paper aims to help the RL and agronomy communities to gain mutual understanding, identify promising research directions and current bottlenecks to foster future joint research to support the design of the next human-centered and data-driven crop management decision support tools. We first define the crop management decision problem as an element of farm decision-making, and describe the dedicated decision support systems. Section 1.2 introduces the RL paradigm. Section 1.3 provides a review focused on RL applied to crop management. Finally, Section 1.4 explores research opportunities and challenges for the use of RL to support crop operation decisions.

Crop management. Crop management is the logical and ordered combination of agricultural practices or operations applied to a field in order to obtain a particular crop production (Sebillotte, 1974, 1978). A field plot is the site of complex interactions happening between biotic (all living organisms) and abiotic components (soil and atmosphere as supports for living organisms) and crop management through physical, biological and chemical processes, as demonstrated by Husson et al. (2021) with soil Eh-pH dynamics. Consequently, decisions about these operations occur in the face of uncertain events (e.g. climatic events), and within a dynamical system that is only partially known. We consistently use the adjective uncertain for events with unsure realizations.

Through crop management, farmers aim to obtain a production result that matches as closely as possible the targets they defined at the beginning of the cultivation period, such as a minimum yield level and certain quality criteria. Typically, at the start of the cropping season, a crop management plan is defined, as illustrated in Figure 1.1. This plan follows a logical structure, but is an uncertain procedure that requires adaptations to the events occurring during the growing season. Each crop operation is parameterized by multiple factors which determine its outcome and success, further conditioning the remaining crop cycle and future crop operations (Boiffin et al., 2001). For instance,

once a cultivar is chosen, the planting operation is defined by a planting date, planting density, sowing depth, possible chemical seed treatment and the choice of machinery (with its own parameters, such as sowing speed) in a mechanized context.

Operational observations during the cultivation period may reveal issues farmers cannot predict with certainty, such as an outbreak of pests and/or diseases, and this will require adaptive operations. Based on the severity of an unforeseen event, the objective defined before cultivation such as a minimum yield might be revised to compensate for these changes (Cerf and Sebillotte, 1988; Papy, 1998). For instance, if a drought occurs after planting, a farmer may not provide a second fertilizer dose to maize as the application cost is not likely to be rewarded by a yield increase. Consequently, the farmer may reduce the yield target.

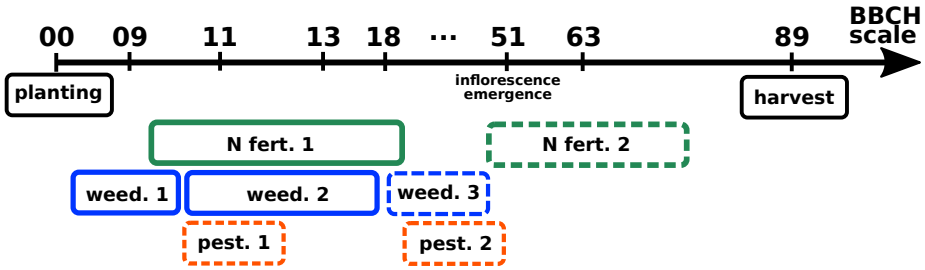


Figure 1.1 A simplified example of maize management plan. The BBCH scale follows the successive maize growth stages as found in Meier (1997), where the first and second digits respectively correspond to the primary and secondary growth stage codes. A dashed box indicates that the operation requirement is uncertain. All operations are made within a time window where the exact date of occurrence is uncertain. 'N fert.' stands for nitrogen fertilization ; 'weed.' for weeding ; 'pest.' for pest and disease control.

Farm decision-making levels. Farm decision-making encompasses multiple nested levels over different time and spatial scales (Chatelin et al., 1993; Papy, 1998). For instance, a cropping system refers to an ensemble of plots equally treated with the same crop rotation (an ensemble of crop types in a given successive order) and crop management (see Boiffin et al., 2001; Sebillotte, 1974). While long-term decisions on a structural production system level are made on an annual to multi-year time line, such as investments in land or machinery, perennial crop implementation or annual cropping systems, crop management decisions are made on a monthly to daily basis. Levels of decision-making may strongly interact. Indeed, the strategic and tactical[⌗⌗] levels may be affected by operational events, as a recurrent operational issue may motivate a change in machinery or crop rotation.

Decision support systems. Decision support systems (DSS) are computer-based solutions designed to assist decision makers in addressing unstructured or semi-structured problems (Arnott and Pervan, 2005; Power, 2008). Structured problems have unambiguous solutions which can be found with an automatic routine. In contrast, semi-structured or unstructured problems have incomplete or uncertain information with possibly unforeseen events and complex trade-offs between different objectives. DSS provide distilled information as evidence to facilitate and improve human decision-making.

[⌗⌗]We define the *strategic* level as long term, covering more than a few years; the *tactical* level as intermediate, ranging from a few years to a few months; and the *operational* level ranging from a few months to a daily basis.

DSS are used in a broad range of domains. For instance, DSS are commonly used in railway track maintenance scheduling to avoid derailments (e.g. Ferreira and Murray, 1997; Guler, 2013), for medical diagnostics (Miller, 2016), or operation planning. As an example, da Silva et al. (2006) designed a DSS to optimize the number of workers, overtime hours and the level of outsourcing in order to meet trade-offs between economic returns to maximize profits while maintaining client and worker satisfaction. DSS can be geared towards a single user from an operator to an executive, to a group that shares decision-making responsibility, or be used to support negotiation between different parties. DSS are not meant to provide off-the-shelf solutions to decision makers to solve a given problem but, rather, to provide a human-machine dialogue, as pointed out by Arnott and Pervan (2005).

Crop Management DSS. Commonly found DSS supporting crop management deal with fertilization, irrigation, pest and disease or weed management; the end users may be researchers, local advisers or farmers. Crop management DSS come in various forms, from advanced user-oriented complex crop models, to easy to use graphical user interface software or even spread sheets (examples can be found in Cerf and Meynard, 2006; Evans et al., 2017; Jones et al., 2017; Le Gal et al., 2010; Manos et al., 2004). In general, they intended to support decisions taken under great uncertainty. For instance, decisions on pest and disease control are usually based on the assessment of the imminence or intensity of crop damage (Gent et al., 2011). They depend on complex interactions of uncertain biotic factors, such as the crowding effect and host-plant response, and a-biotic factors, such as temperature and humidity (Khaliq et al., 2014).

Crop management DSS are based on underlying formal models of various complexity which predict the consequences of actions. These models can take many different formalisms, sometimes combined: a simple set of equations such as soil nitrogen balances (Hébert, 1969; Stanford, 1973), knowledge bases for expert systems (e.g. Lemmon, 1986; Sønderskov et al., 2016), mechanistic models explicitly simulating the processes at stake with crop growth using differential equations (e.g. Brisson et al., 2003; Hoogenboom et al., 2019; McCown et al., 1995) or machine learning models (e.g. Barbosa et al., 2020; Ip et al., 2018; Navarro-Hellín et al., 2016; Sabzi et al., 2018; Saikai et al., 2020; Waghmare et al., 2016). The modeling part is usually done offline, based on prior data. The exploration of candidate crop operations can be made by manual expert guided search (e.g. He et al., 2012; Thorburn et al., 2011), an inference engine for knowledge bases (e.g. Lemmon, 1986), or by using numerical optimization techniques (e.g. Bergez et al., 2001; Epperson et al., 1993; Royce et al., 2001; Saikai et al., 2020).

Despite the existence of numerous applications, the level of crop management DSS use among farmers remains low, as shown by Evans et al. (2017); Gent et al. (2011); Hochman and Carberry (2011); McCown (2002a,b); Rose et al. (2016). The use of DSS in family farming depends on the user's willingness and interest, and is directly related to potential learning through DSS, as emphasised by Evans et al. (2017); McCown (2002a). Thorburn et al. (2011) provide an example of a group comprising sugarcane farmers and local industry representatives who, supported by scientists, learned through a DSS. Based on simulations, the group jointly explored and discussed the environmental benefits of splitting nitrogen applications. While the simulations did not show clear benefits in splitting the applications, the authors concluded that there was an improved understanding of nitrogen dynamics among participants, and thereby a better understanding of the consequences of nitrogen fertilizer management at the individual level. Agricultural DSS have a life cycle where dis-adoption

may occur after users have learned and internalized the assessment of risk in decisions, without being a sign of failure (Evans et al., 2017; Gent et al., 2011; Thorburn et al., 2011).

Several critiques and guidelines for the use of DSS in crop management can be found in the literature. In particular, users have deemed that DSS information cannot directly be turned into actions, that farmers' natural decision-making processes are not adequately taken into account, that the sequential nature of decisions is poorly modeled or that risk management is lacking in the decision process (Cerf and Meynard, 2006; Evans et al., 2017; Hochman and Carberry, 2011; McCown, 2002a,b). Ideas of a "discussion support software" from Nelson et al. (2002), or an "information and advice system" from Cerf and Meynard (2006) or Hochman and Carberry (2011) describe DSS that take advantage of the social tissue in which farmers evolve. A DSS should integrate information fluxes at different scales –from plot to regional– and from various actors involved in multi-level decisions such as local suppliers, pest control advisers and environmental protection bodies.

1.2 Reinforcement learning

In this section, we shall introduce the ideas behind reinforcement learning (RL). In Section 1.2.1, we informally present the elements of RL. Section 1.2.2 then formalizes an RL problem. In Section 1.2.3, we provide a short historical perspective of RL. Section 1.2.4 presents the famous Q-learning RL algorithm. In Section 1.2.5 we describe the main RL algorithm categories. Finally, Section 1.2.6 is dedicated to bandit algorithms, a particular case of RL adapted to small-sample settings.

1.2.1 Overview of reinforcement learning

Machine learning (ML) is the study of computer programs designed to perform a task and able to self-improve with data or experience (Mitchell et al., 1997). Machine learning comprises three subfields: Unsupervised learning, supervised learning, and reinforcement learning. Unsupervised learning deals with learning a representation of data, for instance with clustering tasks. Supervised learning is about learning to label new data based on a set of labelled data (examples) with classification and regression tasks (Mitchell et al., 1997). Reinforcement learning is about learning to control a dynamical system. After a ML model has been trained to perform a given task based on training situations, its performance is measured as its ability to perform the same task in situations that have not been met during the training phase. Overfitting is a recurrent issue in ML, which occurs when, after being trained, a model performs well in training situations but performs poorly in unseen situations.

A reinforcement learning problem is a sequential decision-making problem in which a decision maker iteratively interacts with an **environment** which is an unknown and uncertain dynamical system. The decision maker, called the **agent**, learns the task of controlling the evolution of the environment by taking **actions**. A **policy** corresponds to a set of decision rules which determines which action the agent takes, generally depending on an **observation** of the environment. The learning process proceeds through a loop of interactions between the agent and its environment. Each time the agent performs an action according to its policy, the action affects the environment and the agent receives a return. A **return** is a scalar value which indicates how the agent performs with regard to the task to be completed. This process is repeated until a decision sequence eventually ends. The goal of the agent is to compute a policy which maximizes a utility function of the returns it receives

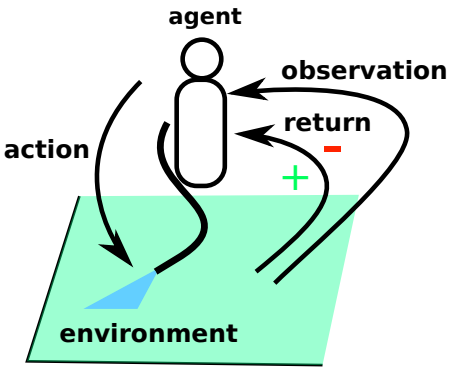


Figure 1.2 The reinforcement learning loop. A decision maker, called the agent, interacts with its environment. The agent task is to control the environment evolution. Sequentially, the agent takes an action based on an observation of the environment. The action impacts the environments, and the agent receives a return that indicates how it performs regarding the task to be completed. This loop repeats until the decision sequence eventually ends.

during a sequence of decisions, called an **objective function**. To do so, the agent adjusts its policy based on the returns it has collected through its experience. The RL loop is summarized in Figure 1.2. RL algorithms are inherently online methods, geared towards action, which react to the ongoing uncertain changes in a system and learn to perform a task by trial and error.

1.2.2 Formalization of a reinforcement learning problem

Markov decision processes. The canonical RL problem formulation models the environment as a **Markov decision process** (MDP, Puterman, 1994). At any moment, the environment is described by its **state** $s \in \mathcal{S}$. $\mathcal{S}$ is the state space, i.e. the set of possible states, known to the learner. Sequentially, at each moment $t \in \{0, 1, \cdots, T-1\}$ the agent chooses an action $a_t \in \mathcal{A}$ depending on the current state of the environment s_t. $\mathcal{A}$ is the action space, i.e. the set of possible actions, known to the learner. T is the **horizon** which may be known or not, and be finite or not. Performing an action affects the environment which transits to its next state $s_{t+1} \in \mathcal{S}$ according to the MDP transition function $\mathbf{p} : \mathcal{S} \times \mathcal{A} \to \mathcal{P}(\mathcal{S})$, where $\mathcal{P}(\mathcal{S})$ denotes the set of probability distributions over states. $\mathbf{p}(s'|s, a)$ is the probability of reaching $s' \in \mathcal{S}$ after action a has been performed in the state s. A random return r accompanies each transition of the environment from a state s to a state s' after taking an action a. We define the return function $\mathbf{r} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ as $\mathbf{r}(s, a, s') = \mathbb{E}[r|s, a, s']$.

In an MDP, the Markov property holds: the probability law of s_{t+1} is fully specified by the knowledge of (s_t, a_t) ; all anterior states and actions can be ignored. The quadruplet $\langle \mathcal{S}, \mathcal{A}, \mathbf{p}, \mathbf{r} \rangle$ is fixed: the environment is **stationary**. For instance, the probability of transiting from one state s to a next state s' after taking an action a is always the same. Figure 1.3 illustrates the elements forming an MDP. In Figure 1.4, we model a simplistic irrigation problem as an MDP.

Markov decision problems A Markov Decision Problem is the combination of a Markov Decision Process and an objective function to be optimized which is usually defined as the expectation $\mathbb{E}\big[R(t)\big]$

A Markov decision process (MDP) $\mathfrak{M}$ is defined by: $\mathfrak{M} = \langle \mathcal{S}, \mathcal{A}, \mathbf{p}, \mathbf{r} \rangle$

- $\mathcal{S}$ the state space,
- $\mathcal{A}$ the action space,
- $\mathbf{p}(s'|s, a)$ is the transition function which give the probability that the environment transits to state s' after action a is performed in state s,
- $\mathbf{r}(s, a, s')$ is the return function, that is the average return after the agent performed action a in state s resulting in a transition to s'.

Figure 1.3 The four elements of a Markov decision process. An MDP models the environment in reinforcement learning problems.

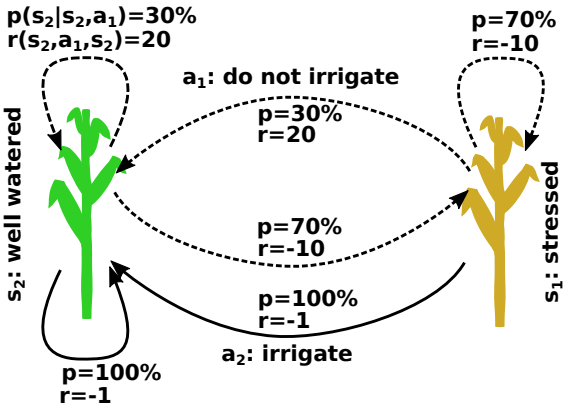


Figure 1.4 A simplistic irrigation problem modeled as a Markov decision process (MDP). Two states are possible: a stressed crop (s_1) or a well watered crop (s_2). Each arrow between two states is a transition which ends in the state pointed by the arrow head. Watering the crop (a_2) always leads to a well watered state, but it has a cost, hence the negative return. If no irrigation is provided (a_1), 30% of the time rainfall occurs and the crop will be well watered for free, hence the great return. But, 70% of the time, no rainfall occurs and the crop gets stressed, which is highly penalized by the return.

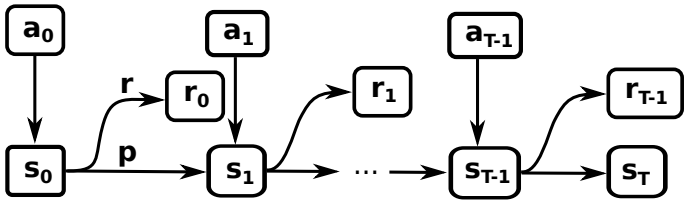


Figure 1.5 The representation of a sequence of decisions is called an episode. In a canonical reinforcement learning problem, starting with the environment in an initial state s_0, at each discrete decision step t, depending on the environment current state s_t the agent decides on an action a_t thanks to its policy. After the agent takes the action a_t, the environment transits towards its uncertain next state s_{t+1}, given by the transition function $\mathbf{p}$. The return function $\mathbf{r}$ provides a return r_t which indicates to the agent how it performs regarding the task to be completed.

of the **discounted return** $R(t)$ collected by the agent (Puterman, 1994, p. 80):

$$R(t) = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \gamma^3 r_{t+4} + \cdots \tag{1.1}$$

$$= \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \tag{1.2}$$

where $\gamma \in [0, 1)$ is the discount factor. The use of γ can be interpreted as with discounted cash flows: future returns are less valuable than immediate returns. A sequence of interactions from an initial state to a given horizon is called a trajectory, or **episode**, which is illustrated in Figure 1.5.

A policy $\pi : \mathcal{S} \to \mathcal{P}(\mathcal{A})$ maps a state to probability distributions over actions. The objective of the agent is to find an optimal policy π^* that maximizes the objective function. To measure the performance of a policy π, we define the **Value function** function (V: $\mathcal{S} \to \mathbb{R}$) and the **Quality function** function (Q: $\mathcal{S} \times \mathcal{A} \to \mathbb{R}$). Acting according to policy π, the value of a state s is the expected return starting from state s, denoted $\mathbb{E}_\pi[R(t)|s_0 = s]$; the quality of an action a in state s is defined as the value of first taking action a starting from state s and then following the policy π:

$$V_\pi(s) = \mathbb{E}_\pi\Big[R(t)|s_0 = s\Big], \forall s \in \mathcal{S} \tag{1.3}$$

$$Q_\pi(s, a) = \mathbb{E}_\pi\Big[R(t)|s_0 = s, a_0 = a\Big], \forall s \in \mathcal{S}, \forall a \in \mathcal{A} \tag{1.4}$$

Denoting Π the set of possible policies, there exists an optimal policy π^* such that:

$$Q_{\pi^*} \geq Q_\pi, \ \forall \pi \in \Pi \tag{1.5}$$

We have:

$$Q_{\pi^*}(s, a) = \max_\pi Q_\pi(s, a), \forall s \in \mathcal{S}, \forall a \in \mathcal{A} \tag{1.6}$$

$$\pi^\star(s) = \operatorname*{argmax}_{a \in \mathcal{A}} Q_{\pi^*}(s, a), \forall s \in \mathcal{S} \tag{1.7}$$

1.2.3 A brief historical perspective of reinforcement learning

We say that an MDP is known, or fully specified, when we have access to the MDP transition probabilities given by the transition function **p** and reward function **r**, see Section 1.2.2. Historically, (Stochastic) Optimal Control (SOC, Kushner, 1967) addresses the control of systems with known MDP. The RL came from the merging of (S)OC and animal psychology to address the problem of controlling a system with an unknown MDP through trial and error: the environment is seen as a black box. (S)OC emphasizes stability analysis, frequency analysis of the controlled systems whereas RL emphasizes the learning process of controlling an unknown dynamical system. (S)OC deals with continuous time and actions while canonical RL problems deal with discrete time, states, and actions. Later, (S)OC and RL converged by addressing decision problems historically belonging to each other's fields, for instance continuous time, states, and actions in RL (e.g. Munos, 1996) and the discrete case in (S)OC (e.g. Bertsekas and Shreve, 1996). Figure 1.6 summarizes the main difference between SOC and RL.

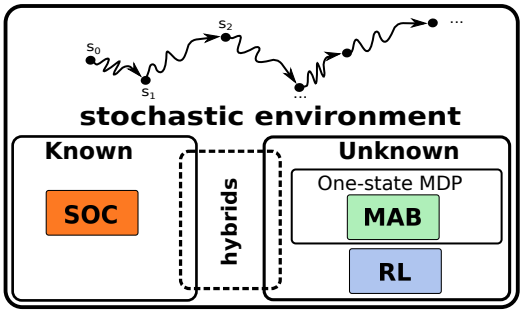


Figure 1.6 Both stochastic optimal control (SOC) and reinforcement learning (RL) address the problem of controlling a system with uncertain dynamics. The main historical difference is that SOC supposes the dynamics of the system to be known while RL does not. Recently, hybrid algorithms have been developed, combining RL and SOC. The multi-armed bandit (MAB) is a simplified case of RL with a one-state MDP, see Section 1.2.6.

1.2.4 Q-learning: a simple reinforcement learning algorithm

Q-Learning (Watkins, 1989) is one of the simplest RL algorithms. It consists of estimating Q_{π^*}, defined in Equation 1.6. We present its pseudo-code with algorithm 1. Q-learning leverages Bellman's optimality equation which makes explicit a recursive relation between the qualities of states for an optimal policy (Bellman, 1957):

$$Q_{\pi^*}(s,a) = \sum_{s'} \underbrace{\mathbf{p}(s'|s,a)}_{\text{weighing}} \Big[\underbrace{\mathbf{r}(s,a,s') + \gamma \times \overbrace{\max_{a' \in \mathcal{A}} Q(s',a')}^{\text{future optimal returns from } s'}}_{\text{discounted optimal returns } R(t) \text{ transiting from } s \text{ to } s'} \Big] \tag{1.8}$$

At each time step $t \in \{1, \cdots, T\}$, after the algorithm takes an action a_t depending on s_t and consequently observes return r_t and next state s_{t+1}, it updates:

$$\underbrace{Q(s_t,a_t)}_{\text{new prediction}} \leftarrow \underbrace{Q(s_t,a_t)}_{\text{current prediction}} + \alpha(s_t,a_t) \times \underbrace{\Big(\overbrace{r_t + \gamma \times \max_{a' \in \mathcal{A}} Q(s_{t+1},a')}^{\text{prediction target}} - Q(s_t,a_t) \Big)}_{\text{prediction error}} \tag{1.9}$$

for instance with learning rate $\alpha(s_t,a_t) = 1/\sqrt{N_{s_t,a_t}+1}$ where N_{s_t,a_t} is the number of times the action a_t has been taken in state s_t. Assuming a proper learning rate and all (state, action) pairs are asymptotically visited an infinite number of times, the Q-value function which the Q-Learning algorithm learns is guaranteed to converge to Q_{π^*} (Bertsekas and Tsitsiklis, 1996).

1.2.5 Reinforcement learning today

Modern RL algorithms stemmed from three archetypal methods shown in Figure 1.7: the **Critic**, **Actor**, and **Planning** methods. Planning methods focus on deriving a policy by interacting with a simulator of the true environment. Planning methods can be used when a simulator of the environment is available to the agent, or when the agent explicitly learns the transition and return functions of the MDP (i.e. a model of the environment) and learns an optimal policy at the same time. Because the potential number of trajectories to be explored is very large, the solutions must be explored efficiently.

Algorithm 1 Q-Learning algorithm

Input: $\varepsilon \in (0, 1]$ `// the greediness parameter`
Initialize Q-values for all state–action pairs with arbitrate values
for *episode* $\in \{0, \ldots, N-1\}$ **do**
  for $t \in \{0, \ldots, T-1\}$ **do**
    observe environment state s_t
    with a probability $1 - \varepsilon$ choose the action a_t as $a^* = \arg\max_a Q(s_t, a)$, else randomly choose $a_t \in \mathcal{A}_t \setminus \{a^*\}$
    observe environment next state s_{t+1} and return r_t
    update $Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha(s_t, a_t) \times \big(r_t + \gamma \times \max_{a' \in \mathcal{A}} Q(s_{t+1}, a') - Q(s_t, a_t)\big)$
  end
end
return *Q-values*

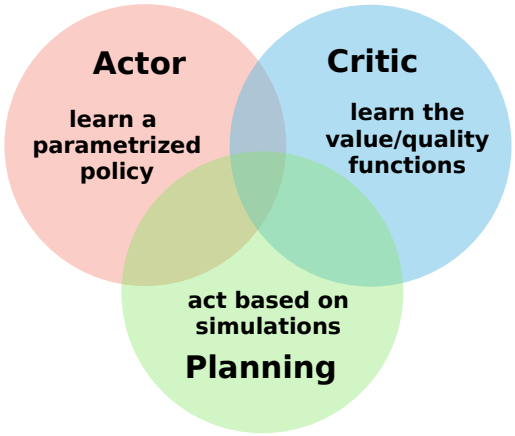


Figure 1.7 Modern reinforcement learning methods are hybrids of three problem solving methods: critic, actor and planning methods, see Section 1.2.5.

A celebrated planning algorithm is Monte Carlo Tree Search (Coulom, 2006; Kocsis and Szepesvári, 2006) which explores the most rewarding simulated trajectories to decide on an agent's action in a given state. The second class of algorithms are critic methods which consists in learning a value function V, or Q. One example is the Q-learning (Watkins, 1989) introduced in Section 1.2.4. Finally, actor methods directly learn an optimal policy in a parameterized fashion (a policy is modeled as a function of a set of parameters) without representing the V or Q functions. For instance REINFORCE (Williams, 1987) searches for an optimal policy using a gradient descent approach in the space of possible policies.

Most of the recent methods are hybrids of the three stems presented in Figure 1.7, combined with the use of neural networks (NN). An NN is made of a set of interconnected units structured in successive layers. Each unit is called a neuron. It computes a function made of simple arithmetic operations from multiple input values and outputs its result. NN are widely used due to the fact that they can approximate any bounded continuous function (Cybenko, 1989). Deep learning is dedicated to the study of the deep neural networks which are neural networks made of multiple layers. Deep neural networks are a powerful way to represent functions when the number of state-action pairs is too large to represent with finite tables. An early achievement of an RL algorithm using NN is

Tesauro's TD-Gammon program (Tesauro, 1995) which learned to play the game of backgammon through self-play, succeeding in challenging expert human players. Mnih et al. (2015) reached human performance playing Atari games using a combination of Q-Learning and a neural network (the Deep Q-Network algorithm, DQN). The Alpha-Go program (Silver et al., 2017), the world's best Go player, is a combination of actor, critic and planning methods using NN to deal with the 10^{170} states and 400 actions.

1.2.6 Multi-armed bandit

The multi-armed bandit (MAB) problem (Lattimore and Szepesvári, 2020), originally introduced for drug allocation by Thompson (1933), can be seen as a special case of RL problem with a one-state MDP. For each time step $t \in \{0, 1, \cdots, T-1\}$, the agent sequentially chooses a single action a among a fixed set of possible actions $\mathcal{A}$. Each time the agent selects an action $a \in \mathcal{A}$, it observes a return r drawn from a fixed distribution of returns of mean value $\mathbf{r}(a) = \mathbb{E}[r|s, a, s]$, and a transition back to the same single state s. In the most common setting, named cumulated regret minimization (Robbins, 1952), the agent objective is to maximize the expectation of the undiscounted sum of rewards it has collected after time T, that is $\mathbb{E}[\sum_{t=1}^{T} r_t]$. This objective is equivalent to minimizing the expected regret, which is a measure of the expected total loss from sub-optimal action taking up to time T. To correctly identify optimal action(s), the agent must try all actions a sufficient, but *a priori* unknown, number of times –which implies choosing sub-optimal actions-. This is an example of the exploration-exploitation dilemma. For various families of algorithms, the bandit theory focuses on providing strong statistical guarantees for the expected regret.

The simpler problem formulation in MAB makes it possible to reduce the sample complexity of the decision problems –that is to say the number of samples required to solve a problem– compared to the general RL setting. MAB algorithms address a rich range of extension settings (Lattimore and Szepesvári, 2020). For instance, risk aware bandits (Cassel et al., 2018) evaluate actions with a risk measure. Considering a random variable X, the mean $\mathbb{E}[X]$ is said to be risk neutral as it equally weighs all possible outcomes whereas risk metrics typically stress bad possible outcomes. To exemplify this, the conditional value-at-risk (CVaR) at level $\alpha \in (0, 1]$ (Mandelbrot, 1997) can be defined as $\text{CVaR}_\alpha(X) := \mathbb{E}[X|X \leq \text{VaR}_\alpha(X)]$ where $\text{VaR}_\alpha(X)$ is the quantile of probability α of X. When $\alpha \to 0^+$, CVaR_α tends to the worst case analysis and with $\alpha = 1$ it recovers the usual mean. Contextual bandits (Lattimore and Szepesvári, 2020, ch. 5) leverage extra information about the context of a decision, such as demographic data for online advertisements.

1.3 Review of reinforcement learning for agriculture

The following review reveals that while stochastic optimal control (see Section 1.2.3) has been widely used to support farm level decisions, attempts to use RL for crop-management purposes are scarce and applications only considered simulated environments.

1.3.1 Early stirrings: farm decision-making under uncertainty

The inclusion of uncertainty and risk to support farm decision-making is not new. Early examples are Tintner (1955) and Freund (1956): stochastic linear programming was used to maximize a utility function for crop allocation under uncertainty and resource constraints at the farm level. The

utility function depended on a farmer's net revenue and degree of risk aversion. Hildreth (1957) discussed the use of game theory (Osborne et al., 2004) to make a decision on crop production plans when the environment dynamics are unknown. Risk treatment assumed that the worst possible scenario occurred. Burt and Allison (1963) later defined decision-making around the choice of crop rotations explicitly as a Markov Decision Problem (see Section 1.2.2) and addressed it using dynamic programming and Bellman's equation (Bellman, 1957), which are the foundations of modern RL.

Approaches using stochastic linear or dynamic programming and their derivatives are part of stochastic optimal control (SOC). There are numerous examples in which (stochastic) optimal control has been applied to farm level decision-making. These can be found in Dury et al. (2012); Glen (1987); Kennedy (1986); Lowe and Preckel (2004); Norton and Hazell (1986) and Weintraub and Romero (2006). Most of these applications were defined at the farm level addressing cropping plans or farm resource allocation, while this article focuses on crop management at the field level, see Section 1.1 for the distinction. As a recent example of an application of SOC, Boyabatlı et al. (2019) formalized a farmer's cropping plan decision problem as a finite horizon stochastic dynamic programming problem, to maximize in expectation an uncertain gross margin due to uncertain yield and selling price. They provided a decision heuristic which was nearly optimal and outperformed the ones provided by the literature.

1.3.2 Seminal works using reinforcement learning in agriculture

The seminal works which applied RL to crop-management are summarized in Table 1.1. Garcia (1999); Trépos et al. (2014) used the R_H-Learning algorithm from Garcia and Ndiaye (1998) which introduced adaptations of Q-learning (Watkins, 1989), see Section 1.2.4, and R-learning (Schwartz, 1993) –a variant of Q-learning with undiscounted returns i.e. $\gamma = 1$ in Equation 1.1– to the case of non-stationary finite-horizon MDP. While Garcia (1999) considered continuous actions, Bergez et al. (2001); Trépos et al. (2014) considered discrete actions. Bergez et al. (2001); Garcia (1999); Trépos et al. (2014) all considered continuous state variables.

In all of these works, the use of RL relies on a crop model to simulate real field conditions. Crop models have their own limits: the policies obtained by RL were inherently limited by the simulation biases. The algorithms are not envisioned as using feedback from farmers to continuously improve the policy learned from the simulator. While Garcia (1999) focused on wheat yield maximization under strong limitations on nitrogen pollution of drinking water supplies, Bergez et al. (2001); Trépos et al. (2014) maximized the gross margin which induces *de facto* a great non-stationarity. Fossil-fuels are required to produce nitrogen fertilizer or to pump irrigation water: their price is known to be highly volatile and consequently an optimal management strategy is likely to be different every year. Such non-stationarity is not problematic in a simulated setting: many simulations can be run before each season to train an agent to maximize the gross-margin. Nonetheless, for *in situ* field-trial based learning, this non-stationarity will dramatically increase the sample complexity which is already highly challenging. As shown in Table 1.1, the number of episodes to train agents ranges from 500,000 to 1,000,000, where one episode corresponds to one year in the real world: this clearly precludes any straight application of the learning procedure in real conditions.

In Trépos et al. (2014), for each episode of the learning process of the algorithm, a sample has randomly been chosen from 41 annual weather records to generate weather uncertainty. This limited number of weather records is likely to have induced overfitting. Because Trépos et al. (2014)

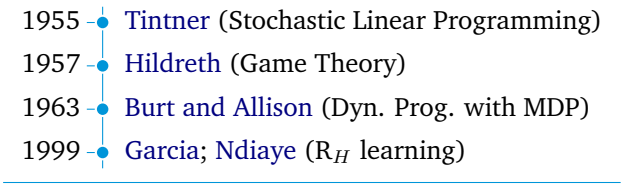


Figure 1.8 Key contributions towards reinforcement learning (RL) use in agriculture. Only Garcia (1999) is categorized as modern RL. Earlier work are based on paradigms that are the historical parents of RL.

evaluated their algorithm on the same weather records as the ones used during the training stage, the performance they measured was likely to be overly optimistic for unseen weather conditions. The use of a stochastic weather generator in Bergez et al. (2001); Garcia (1999) guaranteed more robust results with respect to weather uncertainty. Interestingly, after agent learning Garcia (1999) used an *ad hoc* automatic method of rule extraction to express an optimized policy in a naturalistic fashion "if this is observed then do this . . . ", i.e. as a set of simple decision rules that fit farmers' habits (Evans et al., 2017; Papy, 1998). Key contributions towards RL-supported crop management are summarized in Figure 1.8.

1.3.3 Deep reinforcement learning applications

Recently, Deep RL techniques have been suggested for crop management support. The internet of things (IoT) refers to networks of uniquely identified physical devices (sensors and/or actuators) which can autonomously communicate between themselves or with humans, and process data (Rose et al., 2015). Bu and Wang (2019) have proposed a general IoT architecture for smart decision-making in agriculture based on Deep Q-Learning which combines Deep Neural Networks and Q-learning (see Section 1.2), to directly learn from field trials. The authors discuss the use of improved efficiency algorithms using Transfer Learning (see Taylor and Stone, 2009; Weiss et al., 2016), which is discussed in Section 1.4, and relatively multitask learning (Zhang and Yang, 2021). In a foresight study, Binas et al. (2019) also see potential in combining RL with Deep Learning for sustainable agriculture and propose similar solutions to overcome learning process limitations, such as the use of crop simulators to pre-train algorithms and the use of short-cycle plants for *in situ* learning.

Several works have recently applied (Deep) RL techniques to support crop-management in simulated environments. Wang et al. (2020) used Deep RL with Transfer Learning to control the CO_2 concentration and humidity in a simulated greenhouse to maximize cucumber cumulative weight. Sun et al. (2017) applied RL and Chen et al. (2021); Wang et al. (2020); Yang et al. (2020) applied Deep RL to control the irrigation at the field level, based on atmospheric, soil and plant state features; Chen et al. (2021) included seven day forecasts in the state. The objective functions of Sun et al. (2017) and Yang et al. (2020) were related to the gross margin at crop harvest; in Chen et al. (2021) the return is a score related to rainfall use efficiency and yield. (Chen et al., 2021; Sun et al., 2017; Wang et al., 2020; Yang et al., 2020) compared the performances of their RL algorithms to already existing decision models based on expert knowledge or machine learning. They measured superior performances of their RL algorithms.

However, we should mention that these recent applications share a common caveat in the method of evaluation of their performances. The authors evaluated their algorithms with a single year of

Table 1.1 Principal works which have applied reinforcement learning algorithms to crop management. (c) indicates a continuous variable; (integer) indicates the number of discrete elements; (y/n) indicates a binary feature. In all works, decisions are made during a single growing season.

Reference	Number of decisions	State variables	Actions	Return	Algorithm	Number of episodes	Weather generator	Baseline	Results
Garcia (1999)	3	• planting date • tillering date • plant density (c) • N in soil (c) • date of start the stem elongation • aerial biomass (c)	• seed rate (c) • cultivar (2) • basal N date (2) • basal N rate (c) • top N date (2) • top N rate (c)	yield thresholded to 0 if post-harvest nitrogen in soil greater than 30 kg/ha at crop harvest.	R_H-Learning (Garcia and Ndiaye, 1998)	800,000	yes	experts' policy	The algorithm learned strategies for wheat management under strong nitrogen pollution constraint which performed close to the experts' policy without outperforming them.
Bergez et al. (2001)	daily	• soil water deficit (c) • accumulated thermal units (c)	• irrigate (y/n)	gross margin at crop harvest.	Q-Learning (Watkins, 1989)	1,000,000	yes	policy obtained by dynamic programming (DP) solving	reinforcement learning solutions were better than DP with less than 100,000 learning steps which then exhibited similar performances.
Trépos et al. (2014)	4	• N in soil (c) • water in soil (c) • aerial biomass (c) • plant nutrition (c) • planting date • past fertilization • past herbicide applications	• planting date (3) • first fertilization (2) • herbicide application (y/n) • second fertilization (6)	gross margin at crop harvest	R_H-Learning (Garcia and Ndiaye, 1998)	500,000	no	fixed crop management plan obtained by exhaustive search	a 18% margin increase compared to the optimal fixed crop management plan.

the weather time series and/or with weather time series used during the training phase. Because of the enhanced flexibility of Deep RL techniques compared to more basic RL algorithms, they are more prone to overfitting. The evaluations of the authors are likely to be over-optimistic. A proper evaluation should ideally be done with a great number of weather time series, unused during the training phase. Machine learning results should be presented with a measure of their uncertainty, and the experiments to be reproducible (Pineau et al., 2021).

1.3.4 Multi-armed bandits

Currently, the use of the MAB framework to support crop management remains anecdotal. Kirschner and Krause (2019) tailored a contextual bandit algorithm, see Section 1.2.6, for cultivar choice to maximize the yield under uncertain weather forecasts. A decision context was defined as the union of climatic suitability factors (Holzkämper et al., 2013) and the cultivation site. The authors evaluated their algorithm thanks to a regression model of wheat yield trained on multiyear field trials. Their algorithm was substantially outperformed by the exact knowledge of future weather conditions prior to the decision, but showed better performances for other decision problems.

Baudry et al. (2021a) provide a MAB example of a risk-aware bandit for crop management. They evaluated their algorithm for maize planting date decision-making using the `DSSAT` crop simulator (Hoogenboom et al., 2019) to maximize the CVaR at level α of grain yield, see Section 1.2.6, where α models a farmer's risk aversion. For each decision made, the weather used by DSSAT during the growing season was stochastically generated using the WGEN (Richardson and Wright, 1984) weather generator. The algorithm of Baudry et al. (2021a) proved to be state-of-art for this decision problem. For practical use, ongoing work addresses the adaptation of the algorithm of Baudry et al. (2021a) to batch recommendations, i.e. recommendation to a group of farmers each year to increase the number of samples, the original algorithm being purely sequential (one observation per year).

1.3.5 RL applications in other domains

Li (2019) presents some examples of RL real-world applications, including recommender systems, computer systems, energy, finance, robotics and transportation. Nevertheless, the practical use of RL remains sporadic in industry at the time this article is being written. Over the past few years, research efforts in the field of RL *sensu lato* have focused on other challenging application domains, such as personalized adaptive treatments in health care. As a particularly interesting *in vivo* bandit application, Durand et al. (2018) designed a contextual MAB for sequential drug administration to maximize the information collected from mouse experiments.

1.4 Prospects and challenges

In Section 1.4.1, we first present what conceptually could be an on-farm, human-centered RL-based crop management DSS. Section 1.4.2 prospects how RL problem solving could help to address the challenges of future agricultural decision-making and to further match farmers' decision-making processes. Section 1.4.3 details the specific learning challenges associated with learning from interactions in true conditions. Figure 1.10 wraps up the elements orbiting around a ground-learning RL DSS that we discuss in this Section.

1.4.1 An RL-based crop management DSS

We start by introducing what could be an on-farm RL-based crop management DSS, learning from on-the-ground experiences. A trained RL agent is viewed as an assistant for a human-centered system in the vein of Evans et al. (2017). For instance, an agent's task is to learn to maximize yield under a pollution constraint, as found in Garcia (1999). We suppose that at any time during the growing season, a farmer can query the RL agent. The agent has access to a snapshot describing the field characteristics at the moment it takes a decision, such as: past fortnight meteorological features, current plant nitrogen and water stress status from leaf inspection (after leaf emergence) and the crop growth stage. Depending on the farmer's settings –such as risk aversion level–, defining its constraints and objectives, and plot field state, the agent provides tailored recommendations.

A farmer may first query a planting date choice at the beginning of the growing season. Once a decision has been made by the farmer, the RL agent is provided with the farmer's decision and a time step later, the field parameters are measured again to evaluate the effect of the action that has been taken. The user may request the next time step to evaluate fertilization in the same fashion with the RL agent's support. This time, nitrogen stress would probably increase pest control needs in the area, thus suggesting a minimal fertilization level requirement. The whole interactive process is eventually repeated until the end of crop cultivation by an ensemble of farmers every season. Such an approach would consequently be a dynamic, interactive system between farmers, fields and agent(s) as illustrated in Figure 1.9. As an on-farm real-world RL system would learn from an ensemble of individual experiences on the ground, it is *de facto* a cooperative system supported by a community of farmers.

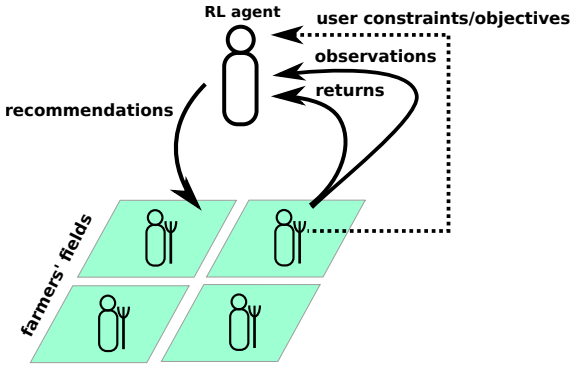


Figure 1.9 An RL-based decision support system for a community of farmers. At any moment, a farmer can query the agent to explore tailored crop management recommendations based on farmer's constraints and objectives. Data should be interactively and iteratively exchanged between farmers and the agent in order to collectively improve the policy for crop management decision problems.

Data collection. An RL on-farm solution would learn from a substantial number of interactions on the ground to evaluate the actions taken. The new data collection techniques and computing frameworks summarized in Table 1.2 could make this interactive learning possible. With such a system, field data (state measurements) must be collected such as human observations (e.g. pest and disease inspection), field sensors (e.g. soil moisture sensors) or remote sensing (e.g. to derive plant stress). Action recommendations must be communicated to the user or additional observations may be

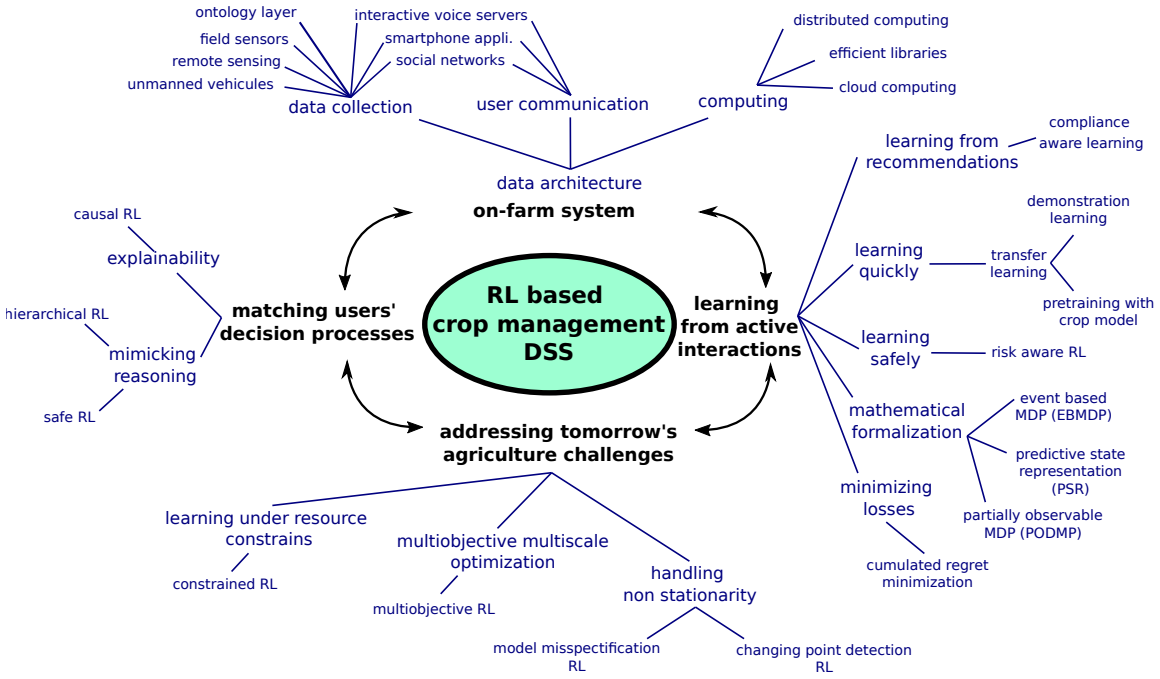


Figure 1.10 Challenging features and respective prospects for RL-based crop management decision support systems. The inner circle represents the desirable features for an RL based crop management DSS. All of these features inter-relate. The outer circle represents the potential technical or theoretical solutions to reach the corresponding features of the inner circle.

requested. Once the user has taken an action, which is not necessarily the recommended one, it should be communicated to the system. The use of field sensors requires the determination of the minimum density for optimal coverage and the minimum frequency of data capture for it to be efficient. More generally, each field observation has a cost which is likely to depend on its precision. A semantic layer is necessary to ensure data harmonization and relevant annotations: digital fieldbooks are an example of such efforts (Shrestha et al., 2010). Crowd sourcing requires specific data management, including *ad-hoc* data quality assessment. Field data traceability is another desirable feature (Quinton et al., 2019).

Data architecture. An overall RL data architecture is necessary to handle recurrent communications between farmers and agent(s) at each decision-making stage. Producing relevant recommendations assumes the storage of past interactions and an *ad-hoc* back-end system to learn from the data. Cloud computing (Hayes, 2008) and distributed computation (Attiya and Welch, 2004) combined with optimized software libraries would be basic tools. Providing personalized recommendations to approximate individual constraints and objectives requires the storage of user-specific information in the data architecture. This consequently raises the common question of data privacy in agriculture (Sykuta, 2016).

Table 1.2 Technological opportunities for Reinforcement Learning (RL) applications. The interactive communication between a virtual agent and the ground reality with farmers, as shown in Figure 1.9, require an *ad hoc* data architecture to allow the RL loop. The back end system is dedicated to agent's computational requirements. The data collection elements essentially captures fields states. Finally, the communication elements allow the human-machine dialog.

Technology	Back-end System	Data Collection	Communication
High-level machine Learning libraries	✖		
Distributed systems	✖		
Cloud computing	✖		
Remote sensing		✖	
Unmanned aerial systems		✖	
Field sensors		✖	
Social network platforms		✖	✖
Smartphone applications		✖	✖
Interactive voice response servers		✖	✖

1.4.2 Prospects

RL appears to be a promising paradigm for meeting the challenges of future agricultural decision-making and to further match farmers' decision-making processes.

Tackling tomorrow's challenges

Faced with increasing decision-making complexity and processes that are too complex/uncertain to be jointly modeled, directly learning through the experience thanks to RL provides interesting perspectives. In particular, sharing farmers' tacit individual experiences, as explored in Evans et al. (2017). As Goulet et al. (2008) point out, farmers also innovate and this knowledge should be leveraged.

Researchers usually employ crop model to elaborate crop management in the context of changing climates. As an example, based on simulations, Adam et al. (2020) showed that in the Sudano-Sahelian zone, in most cases, the yield increase due to improved sorhgum management practices positively compensated the yield decrease due to projected climate change. Nevertheless, Falconnier et al. (2020) pointed out that the effects of nitrogen fertilization and an elevated CO_2 concentration or nitrogen mineralization combined with high temperatures were modeled with large uncertainty (often larger than the simulated impact on yield) for low-nitrogen-input cropping systems in sub-Saharan Africa. Agroecology is a promising paradigm for change-resilient agriculture (Altieri et al., 2015). Agroecological systems are highly complex, and modeling has been limited. For instance, simulations of pest and disease dynamics are limited (Donatelli et al., 2017); intercropping modeling is still in its early stage and so uncertain (Chimonyo et al., 2015). Even under well simulated processes, climatic projections still remain uncertain, for instance with the impact of climate change on droughts (Cook et al., 2018).

Special RL adaptations have been developed for changing environments, named non-stationary, such as a region under climate change. Change point detection in MAB algorithms (see Hartland et al.,

2006; Liu et al., 2018; Mellor and Shapiro, 2013) addresses non-stationary situations and may be extended to MDP (e.g. Padakandla et al., 2020); the Model Misspecification framework also addresses non-stationarity in MDP (e.g. Mankowitz et al., 2020).

Matching users' decision processes

Hochman and Carberry (2011) write "decision support systems need to better match farmers' naturalistic decision-making processes [...]". RL appears to be close to the description of farmers' decision processes. Cerf and Meynard (2006); Evans et al. (2017) point out that farmers usually use small-scale tests and learn by trial and error, repeating experiments under different conditions over the years, given the cyclical nature of crop management. McCown (2002a) uses the expression "learning-in-action". Papy (1998); Sebillotte and Soler (1988) describe how farmers refine crop operations based on successive intermediary crop state checkpoints, as RL does. The use of small-scale tests also directly refers to the exploration-exploitation dilemma introduced in Section 1.2.6: farmers seek to learn potentially better options, but also want to limit potential losses that may occur due to a change in practices. The cumulated regret minimization is largely present in the bandit literature and increasingly found for the general RL setting, for instance with the UCRL algorithm (Auer et al., 2008; Auer and Ortner, 2006). To our knowledge, currently no data-driven crop management support model enjoys such properties.

Learning safely

Farmers have been described to be primarily interested in support for highly uncertain decisions and risk to be a central stressful decision-making determinant (see Cerf and Sebillotte, 1997; Evans et al., 2017; Hochman and Carberry, 2011; McCown, 2002a). The Safe RL (Garcıa and Fernández, 2015) is the generalisation of the risk-aware bandit setting introduced in Section 1.2.6. In Safe RL or equivalently risk-aware RL, the learner has the constraint of avoiding catastrophic failures while learning, e.g. Leurent (2020) with autonomous vehicles, which is of prime interest for subsistence agriculture and food security issues. The use of a risk-aware objective for crop operation evaluation currently remains limited. For instance, Taylor et al. (1999) used the coefficient of variation and Baudry et al. (2021a) used the CVaR (see Section 1.2.6) to compare yield distributions.

1.4.3 Challenges

Crop management has domain-specific constraints for the *in situ* learning process that we detailed in Section 1.4.1. Each constraint introduces specific challenges for the RL community that must be addressed.

Learning is costly

RL involves active data collection, where actions and their consequences are explored while learning; this is unconventional in agriculture. Experiments in agriculture are expensive, with the duration of a crop cycle allowing only a limited number of experiments. Plausible confounding factors may turn unclear research results on the effects of crop management practices and subsequently require meta-analysis, as exemplified by Giller et al. (2009) in conservation agriculture. During a season, the effect of actions can exhibit long delays, for instance an uneven sowing depth for maize

is likely to result in infertile plants due to uneven growth and therefore competition which leads to reduced grain yields. While having shown great progress recently, the learning efficiency and statistical guarantees of RL are still limited (excepted for bandit algorithms). In other words, the amount of data required is generally impracticable for real-world like problems, and the results are uncertain (Dulac-Arnold et al., 2019; Hester et al., 2018).

To speed up an agent learning, transfer learning (see Taylor and Stone, 2009; Weiss et al., 2016) consists of leveraging prior available knowledge for the task to be learned. For instance, in the field of robotics, one does not want to damage the robot while it learns. Therefore, training may first be performed *in silico*, i.e. in simulated conditions, and then transferred to the real-world, though such an approach is not straightforward (Golemo et al., 2018). With demonstration learning (Ravichandar et al., 2020), an expert shows an RL agent how to act before the agent learns on its own. Recently, it has been successfully applied in healthcare to perform complex tasks such as myoelectric prosthesis control (Vasan and Pilarski, 2017), and for ophthalmic microsurgery (Keller et al., 2020).

A need for testbeds. In RL, the first step to address real world problems is generally to create simulated environments to explore the use of candidate algorithms. Despite numerous crop models, very few Open Source RL environments for crop management tasks can be found. More crop models should be turned into RL environments to provide a wide range of crop management learning tasks. The OpenAI gym toolkit is a popular Python encapsulation of complex pre-parameterized underlying models turned into easy to manipulate RL environments with a unified interface. Overweg et al. (2021) introduced an OpenAI gym environment, called `CropGym` which is an interface to the Python Crop Simulation Environment (PCSE) `LINTUL3` (Shibu et al., 2010) wheat crop model and features fertilization tasks. Gautron et al. (2022b) turned the DSSAT (Hoogenboom et al., 2019) Fortran crop model in a Python OpenAI gym environment, named `gym-DSSAT`, for both maize nitrogen fertilization and irrigation tasks. In contrast to `CropGym`, `gym-DSSAT` features a stochastic weather generator which is DSSAT default one (Richardson and Wright, 1984).

Actions are only suggestions

In usual RL problems, the agent has direct control over actions made in the environment. Because recommendations are not authoritative instructions there is no guarantee that an agent choice of action will be consistent with a farmer's decision, which differs from the usual RL problems. As a consequence, an agent cannot freely explore uncertain action effects and cannot directly evaluate its policy. These kinds of settings, known as Compliance Aware Learning, need to be explicitly considered for practical applications. Examples are found in recommender systems or healthcare applications, e.g. Della Penna et al. (2016); Swaminathan and Joachims (2015) with bandit problems, and Sunehag et al. (2015) in an MDP context.

Substantive rationality and utility in RL

In economics, an agent with a substantive rational behavior, as defined by Simon (1976), performs an algorithmic optimization in order to maximize a specific criterion, such as the economic return, under a set of constraints. However, human decision makers tend to use procedural rationality, rather than substantive rationality. With limited information, farmers seek sub-optimal pragmatic solutions that they can implement, thereby meeting the minimum requirements that were set, such as

a minimum yield (Hochman and Carberry, 2011). Farmer's practices are also influenced by social, cultural and economic conditions (Milleville, 1987) and farmer's health (Edwards-Jones, 2006). Deffontaines and Petit (1985) observed that farmers are often not able to provide a clear definition of their own objectives. In contrast, RL intimately relates to the optimization of an explicit utility function which defines the agent's goal. Practitioners should therefore be careful in the inherent limits for characterizing users' decision determinants and bear in mind that any utility function is a proxy (Hochman and Carberry, 2011).

Mathematical formalization

In practice, real world systems are unlikely to strictly follow the stringent assumptions of an MDP (Section 1.2.2). All the parameters describing a field plot are not accessible. Some of them may not be directly or precisely measurable, or are even currently not studied in the literature. Overall, they are too numerous to be jointly measured and they continuously and autonomously evolve with time. Garcia (1999) observed that their crop management problem did not strictly follow the Markov property. To model a field plot in an RL problem, several extensions relax the assumptions of the canonical MDP. As an example, in a partially observable MDP (POMDP, Åström, 1965) the agent does not fully observe the environment state, but still knows the state space. The agent only accesses observations of the environment that it can use as proxies of the real states (e.g. noisy sensor data). With Predictive State Representation (PSR, Littman et al., 2001) the agent does not fully observe the environment state, and nor knows the state space. As an alternative modeling, event-based MDP (EBMDP, Cao, 2008) focus action taking on a limited number of transition events (subsets of state transitions) rather than considering the whole state space. These extensions are still active areas of research. Finally, other research communities addressing sequential decision-making under uncertainty have also developed approaches of potential interest for agriculture. In particular, Ding et al. (2018) dedicated a review to the applications of Predictive Model Control, a sub-field of Optimal Control (see Section 1.2.3), for agricultural decision-making.

Policy explainability

It seems natural that a decision maker would like to know why one crop management action is preferable to another. DSS require user trust (Evans et al., 2017; Rose et al., 2016). As pointed out by Garcia (1999), RL-learned policies are often not directly usable in practice by agronomists or farmers. Causability is a desirable feature of solutions based on AI as a measure of the quality of explanations (Holzinger et al., 2019). A novel and promising RL research trend is Causal RL (Dasgupta et al., 2019; Madumal et al., 2020). While learning to act, Causal RL makes it possible to discover and take advantage of cause to effect models at a symbolic level, allowing better generalization capabilities between learning problems and counterfactual reasoning (Roese, 1997). In a perspective of practicability, an RL agent's crop management policy should be provided with some high probability future action-taking and expected results (such as expected yields). This seems necessary to allow farmers to compare alternatives and plan real-world actions such as anticipating fertilizer purchases.

A need for multi-scale, multi-objective, resource-constrained RL

Agroecology requires thinking about taking actions at larger temporal and spatial scales than the typical plot and crop-cycle scales because the sustainability of agricultural practices requires

multicriteria evaluations (Duru et al., 2015). As examples, crops from surrounding fields may impact local pollinators and/or pest dynamics (Vasseur et al., 2013). So far, most RL algorithms deal with a single, real-valued objective. Based on expert knowledge, practitioners commonly handcraft the MDP return function to express a desirable tradeoff between multiple objectives, and provide localized advice to the agent (Laud, 2004). Multi-objective RL (MORL, Liu et al., 2014) formally addresses the simultaneous optimization of multiple criteria, and is of increasing interest as it relates to many real-world problems. Crop operations are subject to resource constraints (for example, labor, land or input availability) and feasibility conditions (for example, for the soil to have enough load-bearing capacity to use machinery). Resource arbitration at the farm level should ideally also be taken into account.

1.5 Conclusions

Reinforcement learning (RL) deals with the problem of sequential decision making under uncertainty, which appears to fit the purpose of supporting crop management. RL is a contextual, geared toward action tool, which seems to share some similarities with how farmers have been described to deal with crop management while considering inherent uncertainty and evaluating joint action sequences. We have envisioned RL as the core of a human-centered support for learning from real experiments at the community level. RL appears to have great potential for agriculture's future challenges, in particular climate change, in a context of increasingly abundant in-field data, computational resources and theoretical advances. However, a joint research effort by the RL and agronomy communities, supported by ergonomists, is required to turn concepts into practicable tools.

A review of RL applied to crop management has revealed that efforts to apply RL in the agronomy community have so far been limited. A probable explanation is that crop management presents a set of domain-specific practical and theoretical challenges. Decision support cannot be reduced to an algorithmic optimization procedure, user objectives and constraints should be carefully taken into account. Furthermore, data is scarce and costly, and taking the wrong action can be deleterious, especially from a food security perspective. We identified as theoretical challenges how to efficiently learn; how to model crop management decision problems; how to learn explainable crop management policies; how to learn problems with multiple objectives under resource constraints. The multi-armed bandit framework appears one of the most suitable RL approaches for *in situ* learning due to its limited sample complexity and the versatility of the settings found in the literature.

Acknowledgments

This work has been supported by:

- The French Agricultural Research Centre for International Development (CIRAD).
- The Consultative Group for International Agricultural Research (CGIAR) Platform for Big Data in Agriculture. Special thanks to Brian King.
- The French Ministry of Higher Education and Research, Hauts-de-France region, Inria within the Scool team project and MEL.

The authors would like to thank Marianne Cerf, Ronan Trépos, Eric Penot and Mathieu Seurin for their comments that helped to improve the manuscript. We also thank Andrew Lewer for proofreading.

In Chapter 1, we have pointed out the lack of standardized RL environments for crop management learning tasks, and the fact that existing crop models should be turned into RL environments. However, no standard method exists for such conversions, due to the difference in crop model programming languages (usually, crop models are programmed in Fortran/C/C $^{++}$, and RL software are programmed in Python) and because of the usual internal execution of the crop models, which were not designed to be convertible into RL environments. In Chapter 2, we address this methodological gap, and apply the novel method on a well-recognized crop model, namely the Decision Support System for Agrotechnology Transfer (DSSAT, Hoogenboom et al., 2019). Furthermore, we provide preliminary results for learning sustainable crop management practices with RL, in simulated conditions.

Chapter 2

gym-DSSAT: a crop model turned into a reinforcement learning environment*

Romain Gautron [†] [‡] [§] Emilio J. Padrón [¶] Philippe Preux [||] Julien Bigot [**]
Odalric-Ambrym Maillard [††] David Emukpere [‡‡]

*Article published as an Inria Research Report and a short version was accepted for a poster and an oral presentation at the AAAI-23 conference, *AI for Agriculture and Food Systems* workshop.
[†]AIDA, Univ Montpellier, France.
[‡]CIRAD, Montpellier, France.
[§]CGIAR Platform for Big Data in Agriculture, Alliance of Bioversity International and CIAT, Km 17, Recta Cali-Palmira 763537, Colombia.
[¶]UDC–Computer Architecture Group & CITIC (Center for ICT Research) & Edif. Área Científica, Campus Elviña S/N 15071, A Coruña, Spain.
[||]Université de Lille, CNRS, Inria, F-59650 Villeneuve d'Ascq, France.
[**]Université Paris-Saclay, UVSQ, CNRS, CEA, Maison de la Simulation, 91191, Gif-sur-Yvette, France.
[††]Université de Lille, Inria, CNRS, Centrale Lille UMR 9189 – CRIStAL, F-59000 Lille, France.
[‡‡]Work done at Inria, F-59650 Villeneuve d'Ascq, France.

Abstract

Addressing a real world sequential decision problem with Reinforcement learning (RL) usually starts with the use of a simulated environment that mimics real conditions. We present a novel open source RL environment for realistic crop management tasks. `gym-DSSAT` is a `gym` interface to the Decision Support System for Agrotechnology Transfer (`DSSAT`), a high fidelity crop simulator. `DSSAT` has been developped over the last 30 years and is widely recognized by agronomists. `gym-DSSAT` comes with predefined simulations based on real world maize experiments. The environment is as easy to use as any `gym` environment. We provide performance baselines using basic RL algorithms. We also briefly outline how the monolithic DSSAT simulator written in Fortran has been turned into a Python RL environment. Our methodology is generic and may be applied to similar simulators. We report on very preliminary experimental results which suggest that RL can help researchers to improve sustainability of fertilization and irrigation practices.

Software availability

`gym-DSSAT` [https://gitlab.inria.fr/rgautron/gym_dssat_pdi] is an open source software, released under a 3-Clause BSD licence. A complete documentation is available [https://rgautron.gitlabpages.inria.fr/gym-dssat-docs/]. `gym-DSSAT` uses a modification of the Decision Support System for Agrotechnology Transfer (`DSSAT`) software (https://dssat.net/) and the PDI Data Interface (`PDI`) library (https://pdi.dev/master/). Both `DSSAT` and `PDI` are open source software, released under a 3-Clause BSD licence. In this work, we used `gym-DSSAT 0.0.7`.

2.1 Introduction

During a growing season, farmers perform series of crop operations in their fields in order to reach production objectives. They make these decisions under uncertainty, for instance weather uncertainty. We consistently use the adjective uncertain for events with unsure realizations. Reinforcement learning (RL) addresses such problems where an agent learns to control the evolution of an unknown and uncertain dynamical system, in order to perform a given task. In RL, addressing a complex real-world problem usually starts with the use of a high-fidelity simulator which mimics real learning conditions. We present `gym-DSSAT`, an RL environment based on a celebrated crop model, the Decision Support System for Agrotechnology Transfer (`DSSAT`, Hoogenboom et al., 2019) cropping system model. In this introduction, we define the concepts of crop management, mechanistic models and RL, and show how `gym-DSSAT` ties together these notions as an RL environment for crop management tasks.

Crop management is the series of crop operations a farmer performs in a field in order to reach production objectives (Sebillotte, 1974, 1978), such as reaching at least minimum yield and grain protein content. In a field, complex physical, chemical and biological dynamical processes interact (Husson et al., 2021). Uncertain factors, such as weather events, drive the evolution of this dynamical system. In rainfed cropping systems, i.e. non-irrigated cropping systems, rainfall is a major determinant of maize yield besides nutrient availability (Kadam et al., 2014; Li et al., 2019; Mueller et al., 2012). Water stress occurring during maize flowering period may greatly reduce final grain yield (Kamara et al., 2003). Weather forecasts remain highly uncertain beyond 1-month lead time (Hao et al., 2018).

Consequently, at the beginning of the growing season, harvest is highly uncertain in rainfed cropping systems.

Learning sustainable crop management practices is not a trivial task. Nitrogen fertilization requires future minimum rainfall and temperature following the application for the fertilized nitrogen to become available to plants. For an efficient nitrogen fertilizer management, available nitrogen in soil must match plant uptake, both in time and quantity (Meisinger and Delgado, 2002). Indigenous soil nitrogen supply, i.e. nitrogen supply which does not come from fertilizer applications during the current growing season, is often the first crop nitrogen supplier (Cassman et al., 2002). If total nitrogen supply is greater than total plant uptake, the excess of nitrogen will be a source of water pollution, especially with excessive rainfall. If total nitrogen supply is less than total plant nitrogen uptake, then crops may suffer nitrogen deficiency. Maize nitrogen uptake depends on growth stage, and is greater during silking (Hanway, 1963). Early and severe maize nitrogen deficiencies require earlier nitrogen supply compared to situations without such early nitrogen deficiencies (Binder et al., 2000). Thereby, designing an optimal fertilization policy is a complex task. At the time a farmer makes a decision on fertilization, future plant nitrogen uptake, temperature, rainfall and other important factors that determine nitrogen plant nutrition are uncertain and so are the consequences of nitrogen applications (Morris et al., 2018).

In order to address complex crop management decisions, such as designing fertilization or irrigation policies, scientists have developed specialized simulators. Mechanistic models are based on the laws of nature and implemented with expert knowledge to simulate physical, chemical, and/or biological processes with high fidelity (Sokolowski and Banks, 2012). These models have often evolved into complex software over decades of research and collaborative development. Crop models, often called process-based crop models, are mechanistic models which a user uses to simulate crop growth, generally at the plot scale. They model interactions among crops, soil, atmosphere, and crop operations (e.g. planting, fertilizing: see Wallach et al., 2018). As an example, the Decision Support System for Agrotechnology Transfer[§§] (`DSSAT`, Hoogenboom et al., 2019) software is a high-fidelity crop model developed over the past three decades. `DSSAT` is widely recognized by agronomists for crop simulations. It is based on the daily integration of a set of partial differential equations describing the various processes at stake. For instance, nitrogen dynamics partially depend on soil dynamics (e.g. mineralization processes or soil water flows) and plant uptake (itself partially determined by physiological processes such as carbohydrate allocation in plant, depending on growth stages). Crop models can be used as exploratory tools to find best management practices. For instance He et al. (2012) identified best sweetcorn irrigation and fertilization practices in Florida, USA, based on simulations.

Reinforcement learning (RL, Sutton and Barto, 2018) is a domain of machine learning (ML) and more generally artificial intelligence (AI) that addresses sequential decision problems under uncertainty. A decision maker, called an agent, interacts with a dynamical system called the environment which dynamics may be stochastic. The goal of the agent is to control the evolution of the environment in order to perform a given task. Along a series of decisions, named an episode, the agent sequentially interacts with its environment until the decision sequence eventually ends. At each time step of an episode, the agent observes its environment, decides on an action and performs it. After the agent has taken an action, the action impacts the environment, and the agent receives a return from the environment. In general, the return is a scalar value, which indicates how the agent is performing

[§§] https://dssat.net

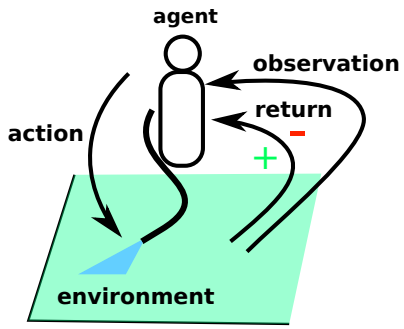


Figure 2.1 The Reinforcement learning loop. The goal of the decision maker, called the agent, is to control the evolution of a dynamical system called the environment, in order to perform a given task. Sequentially, the agent observes the environment and takes an action based on this observation. The action affects the environment, and the agent receives a return that indicates how it is performing regarding the task to perform. The process repeats until the decision series eventually ends.

regarding his task. The agent task is to maximize, in expectation, the total reward it has collected during an episode. To do so, the agent learns from multiple episodes in a trial and error fashion. Figure 2.1 illustrates the interaction loop occurring during an episode. One can think of the “hot and cold" kid game where the hunter’s goal is to find a hidden object in a room. Each time after the hunter has moved, if he gets closer to its target, the other kids indicate “hotter", else “colder". Based on trial and error, the hunter will try to refine its position to maximize the temperature. This process repeats until the hunter finally finds the object, and the episode ends. This simple example illustrates the concepts of RL, where the hunter is the agent, the environment is the room with the position of the hunter and the hidden object, and finally the temperature is the return. RL generalizes these concepts to the stochastic case where after each action, the environment evolution and returns are drawn from probability distributions. RL seems an relevant tool to solve crop management problems, and in particular, to address sustainable agriculture challenges (Binas et al., 2019; Gautron et al., 2022a).

In the vast majority of RL applications, researchers only experiment with simulated RL environments. Nonetheless, RL algorithms ultimately intend to directly learn from real-world interactions (Sutton and Barto, 2018, Chapter 17, Section 6). Still, real-world RL applications generally begin with the use of a simulator of the environment as testbed for candidate algorithms, and/or used to facilitate real-world learning with the help of prior knowledge learned from simulated interactions. In the latter case, such knowledge transfer from imperfect simulations to reality is still challenging in practice (e.g. Golemo et al., 2018). The simulation of real conditions require complex models that accurately mimic the evolution of the environment. These simulators embed state-of-the-art and continuously evolving knowledge. Crop models are consequently of great interest to address real world crop management problems with RL.

Crop modellers historically belong to scientific communities that are generally far from the ML/RL communities. Crop models were not designed to fit into an RL interaction loop. Most of widely used crop models (see examples in Camargo and Kemanian, 2016) internally work on a daily state update during the growing season but do not allow daily interactions with the user (be it human or virtual). A user first parameterizes a simulation, which often requires substantial domain specific knowledge. Then the simulator runs until reaching a final state which is generally crop maturity. After completion of the simulation, the user accesses partial in-season intermediate and final field states that have

been internally stored during the execution. Moreover, crop modelers usually have implemented these models using Fortran, C, or C++ programming languages whereas RL researchers tend to favor Python nowadays. It follows that turning a crop simulator into a proper RL environment –without the burden of simulation setting requiring advanced expert knowledge– is challenging. To turn the monolithic `DSSAT` Fortran crop model into an RL environment, we introduce the use of the PDI Data Interface[¶¶] (`PDI`) which allows loose coupling between C/C++/Fortran code and Python code. Beyond `DSSAT`, this approach may be used to turn other C/C++/Fortran monolithic mechanistic models into RL environments. We think this approach could reveal the value of many existing simulators as RL environments.

Section 2.2 presents similar works which turned crop models into RL environments. Section 2.3 briefly introduces mathematical and practical formalization of RL problems. Section 2.4 describes `gym-DSSAT` features and decision problems. In Section 2.5, we show the internals of `gym-DSSAT` in a nutshell. Section 2.6 provides an example of how to address the problem of maize nitrogen fertilization in `gym-DSSAT` as a use case, and discusses execution time and reproduciblity of experiments using `gym-DSSAT`. Finally, in Section 2.7 we open on limits of our current crop management environment and discuss future improvements.

2.2 Related work

Early seminal works addressed agricultural decision-making under uncertainty at the farm scale (Freund, 1956; Tintner, 1955). The first case of an RL agent interacting with a crop simulator in order to learn crop management is found in Garcia (1999). The author used a modification of the `Déciblé` crop model (Chatelin et al., 2005). The RL agent learned wheat sowing and nitrogen fertilization under pollution constraints. During simulations, weather series were stochastically generated. The modified version of `Déciblé` is not available anymore. In Garcia (1999), the RL agent did not manage to outperform the crop management policy of an expert. Opportunities modern RL techniques bring for learning sustainable crop intensification have been prospected by Binas et al. (2019); Gautron et al. (2022a). Recently, several works directly used crop models or surrogate models as RL environments (e.g. Chen et al., 2021; Sun et al., 2017; Wang et al., 2020). However, none of these works has provided an open source and standardized crop management RL environment.

Overweg et al. (2021) proposed `CropGym`, a `gym` interface to train an agent to perform wheat nitrogen fertilization. The environment uses the Python Crop Simulation Environment (PCSE) `LINTUL3` (Shibu et al., 2010) wheat crop model. Fertilization is treated as a weekly choice of a discrete amount of fertilizer to apply. The authors successfully trained an RL algorithm to address nitrogen fertilization in their RL environment. The agent performed better than the two expert fertilization policies they considered. In the aforementioned RL environment, there is no built-in stochastic weather generation. Overfitting describes the fact an algorithm, after being trained, performs poorly in unseen situations, despite having shown good performance in training situations. In `CropGym`, simulations use a limited set of historical weather records, which may favor overfitting due to limited randomness, especially for data intensive algorithms used in deep RL (see Section 2.3.1).

Contribution `gym-DSSAT` provides both maize fertilization and irrigation RL problems. Our RL environment features a built-in stochastic weather generator. We designed `gym-DSSAT` to allow

[¶¶] https://pdi.dev

researchers to easily customize realistic crop simulations of one of the most celebrated crop simulator, the DSSAT crop model. DSSAT datasets being abundant in the literature, gym-DSSAT allows to mimic a wide range of real-world growing conditions. Our Python RL crop management environment provides to the user a simple standardized interface, and still results in a lightweight software. Our technical approach is generalizable to any of the 41 other crops DSSAT simulates, and more broadly to other C/C++/Fortan mechanistic models.

2.3 Formalization of RL decision problems

Sections 2.3.1 and 2.3.2 present most common mathematical formalization of RL decision problems. Section 2.3.3 presents gym, a practical pythonic interface to RL environments.

2.3.1 From Markov decision processes to reinforcement learning

Though RL paradigm may address a wide range of sequential decision problems, RL is usually employed to solve Markov decision problems (MDP). We introduce minimal materials on MDP, for the reader to get an appropriate understanding of this paper. For an in-depth presentation of MDP, see Puterman (1994).

Markov decision process A Markov Decision (MD) process describes the evolution of a dynamical system over discrete time. The system evolution is impacted by the actions an agent can perform. An MD process $\mathfrak{M}$ is defined by a tuple $\mathfrak{M} = \langle \mathcal{S}, \mathcal{A}, \mathbf{p}, \mathbf{r} \rangle$. At each decision step $t \in \{1, 2, 3, \cdots\}$, an agent observes the state of the environment $s_t \in \mathcal{S}$ and takes an action $a_t \in \mathcal{A}$, where $\mathcal{S}$ is the state space, i.e. the set of all possible states and $\mathcal{A}$ is the action space, i.e. the set of all possible actions. Each action $a \in \mathcal{A}$ leads to a stochastic transition from current state s_t to next state s_{t+1}. $\mathbf{p}$, the transition function, defines the transition dynamics: $\mathbf{p}(s, a, s')$ is the probability the environment transits to state s' if action a has been performed in state s. After performing an action, the agent receives a return, or reward, from the environment. Returns are given by the real function $\mathbf{r}$, named return function. $\mathbf{r}(s, a, s')$ is the expected return when action a is performed in state s leading to next state s'. The interaction between an agent and an MD process generates a sequence $s_0, a_0, r_0, s_1, a_1, r_1, s_2, a_2, r_2, etc.$, called an episode, as Figure 2.2 illustrates. An MD process verifies the Markov property: the probability law of s_{t+1} is fully specified by the knowledge of the current state s_t and the action performed in this state a_t at time t (and $\mathcal{M}$). There may exist a subset of states $\mathcal{S}_{\text{final}} \subset \mathcal{S}$, called the set of final states, such that when the agent reaches a state $s \in \mathcal{S}_{\text{final}}$, the episode ends.

Markov decision problem A Markov decision problem (MDP) is a Markov decision process in which the agent has to optimize a given objective function. Let us consider an MDP in which the agent performs a given number $T < \infty$ of interactions and let us define the objective function J as:

$$J(T) = \sum_{t=0}^{T-1} r_t, \tag{2.1}$$

where r_t is the return collected by the agent at time step t. The state reached at time T is a final state. The agent goal is to maximize $J(T)$. A policy $\pi : \mathcal{S} \to \mathcal{P}(\mathcal{A})$ maps each state to a distribution over the set of actions $\mathcal{P}(\mathcal{A})$. A policy specifies which action the agent performs in any state. The objective

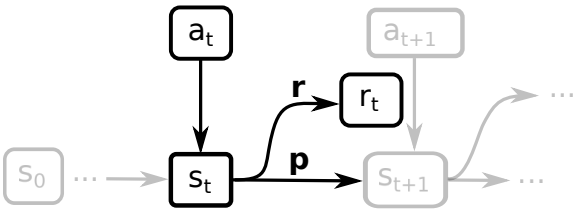


Figure 2.2 In reinforcement learning, a Markov decision process models the environment. At each time step t, the agent observes the environment current state s_t. Depending on s_t, the agent takes an action a_t according to its policy. As a consequence of taking action a_t, the environment transits to next state s_{t+1}, depending on the transition function $\mathbf{p}$, and the agent observes the return r_t which depends on the return function $\mathbf{r}$. This process repeats until the episode eventually ends.

function $J(T)$ depends on the returns the agent has collected between $t = 0$ and $t = T - 1$. Collected returns depend on the agent policy, consequently, $J(T)$ is a function of the agent policy. The more a policy maximizes $J(T)$, the better the policy is. Considering a policy π, we define the value of a state s as the expectation of the objective function when the agent follows policy π starting from state s:

$$V_\pi(s) = \mathbb{E}_\pi\left[\sum_{t=0}^{T-1} r_t \middle| s_0 = s\right], \forall s \in \mathcal{S}. \tag{2.2}$$

The Q-value of state s and action a is defined as the expectation of the objective function when the agent performs a in s and then follows π:

$$Q_\pi(s,a) = \mathbb{E}_\pi\left[\sum_{t=0}^{T-1} r_t \middle| s_0 = s, a_0 = a\right], \forall (s,a) \in \mathcal{S} \times \mathcal{A}. \tag{2.3}$$

The goal of the agent is to learn an optimal policy $\pi^\star$ that maximizes the value in all states. For the MDP we consider in this paper, it can be proven that at least one optimal policy exists (Puterman, 1994). When an MDP is fully defined, i.e. $\mathcal{S}$, $\mathcal{A}$, $\mathbf{p}, \mathbf{r},$ and T are known to the agent, finding an optimal policy is an optimization problem where all necessary quantities to compute a solution are available. For instance, dynamic programming can be employed to approximate an optimal policy. When $\mathbf{p}, \mathbf{r}$ (and T) are unknown, then RL can be employed. In the latter case, in general $\mathbf{p}$ and $\mathbf{r}$ can only be sampled through interactions of the agent with the environment. Most of RL algorithms belong to one of the three following families (Sutton and Barto, 2018): (1) *critic* methods which are algorithms that learn a value function (e.g. Q-Learning, FQI, DQN) and then derive an optimal policy from it; (2) *actor* methods which are algorithms that directly learn an optimal policy (e.g. REINFORCE); (3) *actor-critic* methods which simultaneously combine actor and critic methods (e.g. A2C, PPO, SAC). In order to deal with potentially very large state and/or action spaces, RL algorithms generally use function approximators, to compactly represent value and/or policy functions. Deep RL is a special case of RL where function approximators are neural networks (Lapan, 2018).

2.3.2 Partially observable Markov decision process

An MDP is an idealized model of a real-world system because real systems are unlikely to verify the properties associated to MDP, in particular the Markov property. A field plot involves many interleaved

dynamical processes, and parameters which are still partially discovered/measurable and the study of these dynamics are active areas of research (e.g. Husson et al., 2021). In an MDP, each state is supposed to contain all necessary information for the agent to be able to decide which action is the best to perform in order to optimize the objective function. Except from synthetic problems like games with complete information, such as the game of Go, for most systems, the exact environment state is unknown to the agent. In contrast, with real-world systems, the agent is likely to only access uncertain or incomplete observations of states. Such problems can be formalized as partially observable Markov decision Problem (POMDP, Åström, 1965). POMDP are a specific topic of study in the RL literature, and require *ad-hoc* algorithms to solve them (e.g. Spaan, 2012).

2.3.3 `gym` environments

OpenAI gym[***] is an open source toolkit initially developed by the Open AI company, that provides light RL environments with a standardized Application Programming Interface (API). `gym` API has become a reference in the RL community to create standardized RL environments in order to compare performances of RL algorithms. Many environments are available with `gym`, for instance with simulated games or physical dynamical systems, including robots. Typically, `gym` environments are straightforward to use: all simulated dynamics are pre-parameterized and hidden. The user instantiates an environment as simply as:

```
import gym
env = gym.make("CartPole-v0") # create an instance of the environment CartPole-v0
```

As Figure 2.3 shows, `gym` is a wrapper that gives access to a more complex simulator. `gym` environments come with default attributes which specify action and observation spaces. For instance in the case of the `CartPole-v0` environment, the user gets the specifications of a four-dimensional state space and a set of two possible actions:

```
>>> env.observation_space # outputs observation lower bound, upper bound, shape, data
                                                type
Box(-3.4028234663852886e+38, 3.4028234663852886e+38, (4,), float32)
>>> env.action_space # if Discrete class, outputs the number of possible values
Discrete(2)
```

The user interacts with the environment through standardized methods. `gym` is independent of the implementation of the agent policy. The agent interacts with the environment by calling the `step()` method with an argument a_t specifying the action to take, in order to receive the transition and reward generated by $\mathbf{p}$ and $\mathbf{r}$. The objective function is neither part of `gym` implementation.

To illustrate the simplicity of interactions, we exhibit a basic RL loop:

```
observation = env.reset() # reset the environment and get initial observation
# >>> observation
# array([-0.03325944, -0.02851367,  0.00086817, -0.00618905])

done = False # True when the episode is ended
while not done:
    action = policy(observation)  # get action depending on agent policy
    observation, reward, done, info = env.step(action) # perform the action
    # update the policy
```

[***] https://gym.openai.com/

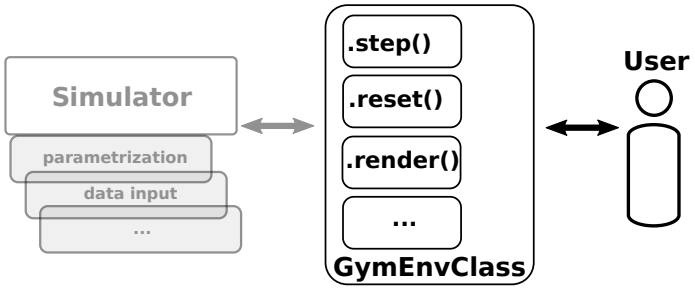


Figure 2.3 From a user's perspective, gym environments are simplified interfaces to simulators, through standardized methods.

```
env.render() # graphical representation of environment state
env.close()  # gracefully exits the environment
```

observation corresponds to a possibly incomplete MDP state s_t, reward corresponds to r_t, done is True if the episode has ended, i.e. if the agent has reached a final state, and finally info provides optional extra information about the environment. We refer to the documentation available at https://gym.openai.com/ for further details.

2.4 Decisions problems in gym-DSSAT

In Section 2.4.1 we introduce the default crop management problems gym-DSSAT provides and Section 2.4.2 outlines how a user can create customized crop management problems.

2.4.1 Default crop management problems of gym-DSSAT

By default, gym-DSSAT sequential decision problems simulate a maize experiment which has been carried out in 1982 in the experimental farm of the University of Florida, Gainesville, USA (Bennett et al., 1989, UFGA8201 experiment). An episode is a simulation of a growing season. A simulation starts prior to planting and ends at crop harvest which is automatically defined as the crop maturity date. Crop maturity, a final state in gym-DSSAT, depends on crop growth, which depends itself on crop management and weather events, and the time to reach it is stochastic. Note that other final states exist in gym-DSSAT. For instance, improper crop management or too stressing weather conditions may lead to early crop failure, which is also a final state. During the whole growing season (about 160 days on average), an RL agent daily decides on the crop management action(s) to perform: fertilize and/or irrigate. By default, for each episode, the weather is generated by the WGEN stochastic weather simulator (Richardson, 1985; Soltani and Hoogenboom, 2003). WGEN has been parameterized based on historical weather records of the location of the original experiment. The duration between the starting date of the simulation and the planting date, which lasts about one month, induces stochastic soil conditions at the time of planting (e.g. soil nitrate, or soil water content), as a result of stochastic weather events.

The number of measurable attributes in a field is extremely large. In contrast, farmers have been described to make crop management decisions based on a limited practicable number of field observations (Papy, 1998). For this reason, in gym-DSSAT, the RL agent only accesses a restricted

subset of DSSAT state variables which constitutes the observation space of the environment. Based on agronomic knowledge, we selected this subset with the constraint of the variables to be realistically measurable/proxiable in real conditions. These observation variables are mixed, and take either continuous or discrete values. We documented all observation and action variables in the gym-DSSAT YAML configuration file[†††]. This file includes description of variables type, range, and agronomic meaning.

In DSSAT, the WGEN stochastic weather simulator is implemented as a first-order Markov chain, but all other processes are deterministic. Therefore, gym-DSSAT decision problems are Markovian. Because the agent only accesses a subset of all DSSAT internal variables, a gym-DSSAT problem is a POMDP, similarly to the real problems faced by farmers. From an RL perspective, one can rigorously address a gym-DSSAT decision problem as a POMDP, or follow the common pragmatic approach which treats a POMDP as an MDP. In contrast with many toy RL environments, the environment is autonomous: it evolves by itself and not only because an action has been performed by the agent. Indeed, if on a given day a farmer does not fertilize/irrigate, its field plot still evolves. A do-nothing action is always available at each time step, which corresponds to the spontaneous field evolution.

DSSAT simulates dynamics at the plot level; likewise, the agent performs actions on the whole field plot. Growing conditions such as soil characteristics and other crop operations such as soil tillage, cultivar choice are fixed. We defined default return functions based on agronomic knowledge following the reward shaping principle (Ng et al., 1999; Randløv and Alstrøm, 1998), such that rewards were as much frequent and as much informative as possible regarding the desired behaviour of the agent. Reward shaping aims both at facilitating an agent learning and to steer policies towards desirable trade-offs such as maximizing grain yield and minimizing induced pollution. We define return functions in a standalone Python file, and users can find admissible values of actions in the environment YAML configuration file, or in gym-DSSAT action space attribute.

By default, gym-dssat provides three RL problems:

1. A **fertilization problem** in which the agent can apply every day a real valued quantity of nitrogen, as indicated in Table 2.1. Crops are rainfed, and no irrigation is applied during the growing season, excepted a single one before planting. DSSAT automatically performs planting operation when soil temperature and humidity lie in favorable ranges. Denoting $\texttt{trnu}(t, t+1)$ the plant nitrogen uptake (kg/ha) from its environment between days t and $t+1$; and $\texttt{anfer}(t)$ the nitrogen fertilizer application (kg/ha) on day t, we crafted the default fertilization return function as:

$$r(t) = \underbrace{\texttt{trnu}(t, t+1)}_{\substack{\text{plant nitrogen} \\ \text{uptake}}} - \underbrace{0.5}_{\substack{\text{penalty} \\ \text{factor}}} \times \underbrace{\texttt{anfer}(t)}_{\substack{\text{fertilizer} \\ \text{quantity}}} \tag{2.4}$$

   The return is the daily population nitrogen uptake (to be maximized) which we penalize if the agent has fertilized the previous day. We defined the penalty factor based on expert knowledge such that the return corresponds to a desirable trade-off between agronomic, economical and environmental potentially conflicting objectives. Table 2.2 details the observation space.
2. An **irrigation problem** in which the agent can provide every day a real valued quantity of water to irrigate, as indicated in Table 2.1. Independently of agent actions, this problem features at the same time a deterministic low input nitrogen fertilization (see Table 2.3). Planting date is

[†††] https://gitlab.inria.fr/rgautron/gym_dssat_pdi/-/blob/stable/gym-dssat-pdi/gym_dssat_pdi/envs/configs/env_config.yml

action	description	range
fertilization	daily nitrogen fertilization amount ($\mathrm{kg/ha}$)	[0,200]
irrigation	daily irrigation amount ($\mathrm{L/m^2}$)	[0,50]

Table 2.1 gym-DSSAT available actions

	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 ($\mathrm{m^2}$ leaf/$\mathrm{m^2}$ soil)
dtt	growing degree days for current day ($^\circ\mathrm{C}$.day ; base temp. 6.2 $^\circ\mathrm{C}$)
dap	days after planting (day)
cumsumfert	cumulative nitrogen fertilizer applications (kg/ha)
rain	rainfall for the current day ($\mathrm{L/m^2}$/day)
ep	actual plant transpiration rate ($\mathrm{L/m^2}$/day)

Table 2.2 Default observation space for the fertilization task.

fixed, about one month after the beginning of simulation. The daily-based return is the daily change in above the ground population biomass (to be maximized), which we penalize if the agent has irrigated the previous day, similarly to the fertilization problem. We provide default reward function in Appendix Figure A.1 and observation space in Appendix Table A.3.

3 A mixed **fertilization and irrigation problem** which combines both previous decision problems: every day, the agent can fertilize and/or irrigate. Planting date is fixed, about one month after the beginning of simulation. In this case, the return has two components, one for each sub-problem: this is a multi-objective problem (e.g. Hayes et al., 2021). The default observation space is the union of the observation spaces of the fertilization and irrigation problems.

We did not define returns of decision problems as economic returns to avoid issues due to cost variations over time (e.g. petrochemicals). Fossil fuel necessary to produce artificial nitrogen fertilizers (see the Haber process Modak, 2002) or to pump water are highly variable over time, making these decision problems non-stationary. Consequently, optimal solutions are likely to change through time. This is why we chose an arbitrary penalization of actions as a proxy of a notion of cost with sound agronomic trade-off, as shown in Equation 2.4. Despite their apparent simplicity, from an agronomic perspective, the three aforementioned decision problems are non-trivial. These problems can be made

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

Table 2.3 Expert fertilization policy. 'DAP' stands for Day After Planting.

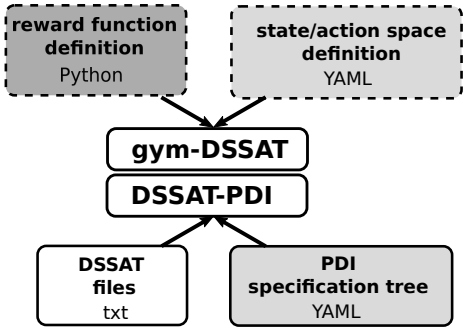


Figure 2.4 Configuration files used by the crop management reinforcement learning environment. At the top of the figure, files in dashed boxes define the reward function and state and action spaces of the Markov decision process. Dashed boxes indicate straightforward to customize configuration files. At the bottom of the figure, `DSSAT` files parameterize simulations, and the `PDI` specification tree is a technical file which manages the communication between `DSSAT-PDI` and `gym-DSSAT`.

harder by providing a more restricted and/or noisy observation space to the agent, see the discussion of the fertilization use case (Section 2.6.1).

2.4.2 Custom scenario definition

A user can customize `gym-DSSAT` problems, with an ease that depends on the features to be modified, see Figure 2.4. An observation is a subset of `DSSAT` internal state variables. Figure 2.5 shows the technical files which define the subset of variables constituting an observation. A user can straightforwardly modify the observation space in the YAML configuration file. In the same way, the definition of the return functions can be easily modified by the user by editing a standalone Python file[‡‡‡]. Built-in `DSSAT` features can be directly leveraged, such as environmental modifications with changes in atmospheric CO_2 concentration or meteorological features, to mimic the effects of climate change. Including other state variables, actions, crops, soil or weather generation parameterizations requires a deeper understanding of how `gym-DSSAT` works and some agronomic knowledge. This goes beyond the scope of this report; additional information is available in `gym-DSSAT` GitLab page.

2.5 Software architecture of the environment

In contrast with the simplicity of use of `gym-DSSAT`, we had to modify the original `DSSAT` simulator in a non-trivial manner to enable daily interactions with an agent and to interface the modified `DSSAT` Fortran program with Python. `DSSAT` was not designed to be used in an interaction loop. In this section, we detail how we have technically proceeded.

2.5.1 The PDI Data Interface

The PDI Data Interface (`PDI`, Roussel et al., 2017) was the key element in `gym-DSSAT` which turned the original monolithic `DSSAT` simulator implemented in Fortran into an interactive Python RL environment. `PDI` is a library designed to decouple C/C++/Fortran codes, typically high-performance

[‡‡‡] https://gitlab.inria.fr/rgautron/gym_dssat_pdi/-/blob/stable/gym-dssat-pdi/gym_dssat_pdi/envs/configs/rewards.py

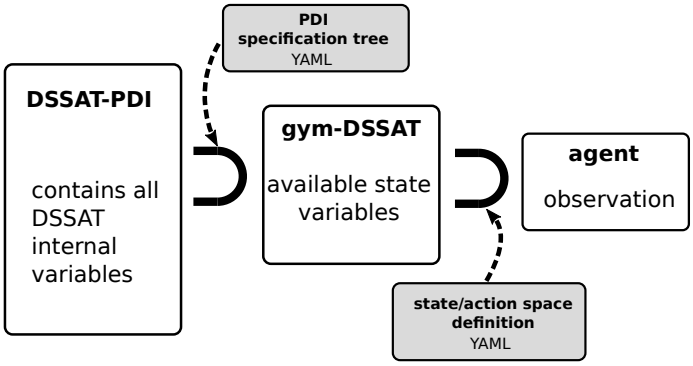


Figure 2.5 Successive subsets of DSSAT state variables until agent observations. Boxes filled with grey indicate files defining state variable subsets.

numerical simulations, from Input/Output (I/O) concerns. It offers a declarative low-invasive API to instrument the simulation source code, enabling the exposition of selected memory buffers used in the simulation to be read/written from/to PDI, and the notification to PDI of significant steps of the simulation. By itself, PDI does not provide any tool for the manipulation of data, instead it offers an event-driven plugin system to ease interfacing external tools with the simulation.

PDI moves most of the logic for the I/O interface away from the code: specifically, a YAML file is used to describe data structures and to specify when and which actions (provided by the different PDI plugins) to trigger on the selected data. The exposed data is selected by adding a few PDI calls in the source code with a very simple syntax. Other I/O libraries in the High Performance Computing field follow a similar declarative approach, such as ADIOS-II (Godoy et al., 2020), Damaris (Dorier et al., 2016) or XIOS (Meurdesoif et al., 2013). However, most of these alternatives are mainly focused on providing high-level abstractions of high-performace I/O operations, working with some domain-specific assumptions and providing additional features on top of parallel I/O streams, such as burst buffering or compression. PDI design has a general and global approach, aiming at more versatile scenarios, with a plugin system that enables substantially different possibilities and I/O strategies, such as the interaction with external Python code. As a result, PDI makes possible the implementation of gym-DSSAT: an external software (gym), directly interacts with a modification of a stand-alone, monolithic simulator (DSSAT).

Figure 2.6 shows a simplified example of PDI use in gym-DSSAT, for the fertilization problem. Figure 2.6a lists chunks of the YAML file with declarations of exposed variables in the simulation code and definitions of events to be triggered. This YAML file corresponds to the PDI specification tree file in Figure 2.4. Figure 2.6b shows a snippet of the instrumented Fortran source code of DSSAT, with PDI initialization and three exposed simulation variables: two are read by PDI and will be available as observation variables, the third one is written by PDI, and corresponds to the action decided by the agent regarding crop fertilization for the current day. The whole instrumented code corresponds to DSSAT-PDI, see Section 2.5.2.

2.5.2 Internals of gym-DSSAT

We present a generic procedure which is an important methodological contribution of this work. gym-DSSAT is made of two communicating processes, as shows Figure 2.7a:

```yaml
data:
### action
  ANFER: float
#[...]
### state
  DAP: int
  ISTAGE: int
#[...]

on_event:
  fertilize:
    with: {anfer: $ANFER}
    exec:
      anfer.itemset(action['anfer'])
#[...]
```

(a) PDI YAML file.

```
PROGRAM CSM

! [...]
CALL pc_parse_path("dssat-pdi.yml", conf)
CALL pdi_init(conf)
! [...]

! read DSSAT internal state variables
CALL pdi_expose("DAP", DAP, pdi_out)
CALL pdi_expose("ISTAGE", ISTAGE, pdi_out)
! [...]

IF (IFERI == 'L') THEN
   ! specify fertilization for today
   CALL pdi_expose("ANFER", ANFER(I), pdi_in)

   IF (ANFER(I) > 0) THEN
      FERTILIZE_TODAY = .TRUE.
   ENDIF
ENDIF
!     [...]

CALL pdi_finalize()
CALL PC_tree_destroy(conf)
END PROGRAM CSM
```

(b) DSSAT code instrumented with PDI calls.

Figure 2.6 Simplified example of PDI use in gym-DSSAT for the fertilization decision problem. The left-hand side corresponds to the PDI specification tree (Figure 2.4), and the right-hand side to the Fortran code of DSSAT-PDI (Section 2.5.2).

(i) DSSAT-PDI which is the compiled Fortran code of a modification of the original DSSAT crop model, using the PDI library.
(ii) gym-DSSAT which, from a user perspective, is the usual gym interface to the RL environment.

DSSAT-PDI The modification of the original DSSAT software, named DSSAT-PDI, allows an agent to daily interact with the crop simulator during a growing season. This interaction loop consists in repeatedly pausing DSSAT, reading DSSAT internal variables, providing these internal variables to the agent, specifying the action(s) of the agent to DSSAT and finally resuming DSSAT execution. DSSAT being in continuous development, the goal was to modify as little as possible the original source code for easy updates. Minimal interventions on DSSAT code have been facilitated by the PDI library. PDI manages data communication with a Python process, through the PDI `pycall` plugin. Figure 2.7b illustrates how DSSAT-PDI works. During the execution of the internal daily loop of DSSAT, PDI code snippets allow data coupling: accessing, writing in memory variables and triggering events. DSSAT-PDI execution starts with an initialization event, which provides all necessary elements for PDI, DSSAT-PDI and gym-DSSAT to start. Then, DSSAT-PDI enters its daily loop which consists in all successive daily updates of the crop simulator state during a growing season. While the daily loop executes, when the `get state` event occurs, PDI stores the values of a subset of DSSAT internal state variables in the PDI Store. After then, the PDI `pycall plugin` accesses these values, executes a Python script corresponding to the interaction with the agent, and stores back in the PDI `Store` the action(s) taken by the agent. Then, when the `set action` event occurs, PDI writes the variables corresponding to the agent action(s) into DSSAT memory and releases DSSAT daily loop execution. Finally, at the end of the simulation, a `finalization event` occurs to gracefully terminate the whole process. For the

same parametrization and input, DSSAT-PDI and the vanilla DSSAT both consistently provide the same output.

gym-DSSAT From a user's perspective, the gym-DSSAT environment is a simple interface to DSSAT-PDI, but from a technical point of view, gym-DSSAT handles all the execution of DSSAT-PDI. gym-DSSAT provides the necessary data input to DSSAT-PDI, including parametrization and weather data; it manages data communication and translation between the RL agent and DSSAT-PDI without extra effort. Finally, gym-DSSAT is responsible for the graceful termination of DSSAT-PDI.

Messaging between DSSAT-PDI and gym-DSSAT. During the execution of a block of Python code, the PDI `pycall plugin` accesses DSSAT-PDI state variables, which have been previously stored in the `PDI Store`. Nevertheless, the data available in this Python process still requires to be communicated to gym-DSSAT, another Python process which is independent of the DSSAT-PDI process. As shown in Figure 2.7a, the communication between gym-DSSAT and DSSAT-PDI is powered by ZeroMQ (Hintjens, 2013) Python sockets, with the PyZMQ package. Python sockets exchange data as JSON files, encoded as strings. Every transaction is a blocking event such that DSSAT-PDI daily loop is resumed only after DSSAT-PDI has received agent action(s).

2.6 Experimenting with gym-DSSAT

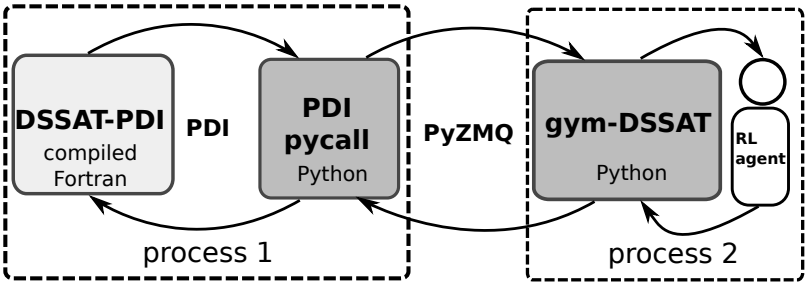
In this section, we provide an RL use case for the maize fertilization problem using gym-DSSAT. We also discuss execution time and reproducibility issues using gym-DSSAT.

2.6.1 Use case: learning an efficient maize fertilization

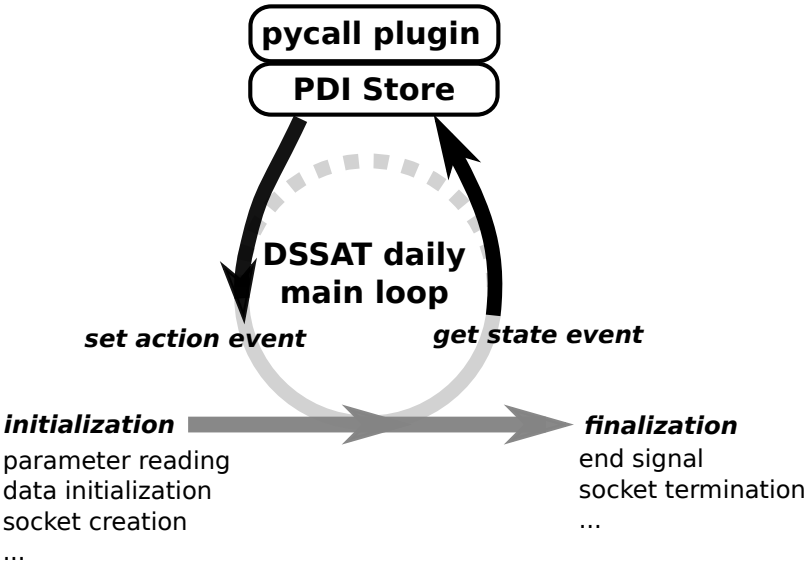
As a simple use case, we present an example of how to address the fertilization task. We provide the irrigation use case in Appendix A.1. The source code of these experiments is available in gym-DSSAT GitLab page.

Methods We consider the nitrogen fertilization task, as introduced in Section 2.4.1. The decision problem being on a finite horizon, i.e. each episode lasted during a growing season, we defined the objective function of the agent as the undiscounted sum of returns, see Equation 2.1. Table 2.2 presents the subset of DSSAT internal variables we have selected to define the observation space provided to the agent. These observation variables were selected as they could be realistically measured on farm. As a common practice, we pragmatically addressed this decision problem as an MDP, even though it is a POMDP (Section 2.4). We used the Proximal Policy Optimization algorithm (PPO, Schulman et al., 2017), as implemented in `Stable-Baslines3 1.4.0` (Hill et al., 2018). PPO belongs to the family of deep RL actor-critic methods (see Section 2.3.1) and uses gradient descent to search for a good policy. PPO generally performs well on a wide range of problems and has been adopted as a standard baseline by the RL community. It is versatile as it can deal with both continuous and/or discrete actions and observation variables. In this experiment, we considered three policies:

- We first considered the most trivial fertilization policy: the "null" policy that never fertilized. As there still remains nitrogen in soil before cultivation (Morris et al., 2018), without mineral fertilization, the reference experiment, or *control*, is usually the null policy. Agronomists then



(a) The reinforcement learning environment consists of two interacting processes. (i) the core modification of the DSSAT simulator, DSSAT-PDI, with its PDI module to execute Python code (pycall plugin); (ii) the gym Python interface gym-DSSAT. PyZMQ handles messaging between (i) and (ii) through Python sockets.



(b) Simplified PDI data coupling and program execution of DSSAT-PDI which is the instrumented code of DSSAT. PDI handles the oftware initialization, data exchange with gym-DSSAT during the whole simulated growing season through the pycall plugin, and finally software graceful termination. The execution of the Python code by PDI pycall plugin is a blocking transaction.

Figure 2.7 The elements of gym-DSSAT.

measure the effect of a nitrogen fertilization policy as a performance gain compared to the null policy, in order to decouple the effect of nitrogen fertilizer from the effect of already available nitrogen in soil (Vanlauwe et al., 2011).

- The second baseline is the 'expert' policy, which is the fertilization policy of the original maize field experiment (Bennett et al., 1989, `UFGA8201` experiment number #1), see Section 2.4.1. As Table 2.3 shows, this policy consists in three deterministic nitrogen fertilizer applications, which only depend on the number of days after planting.
- Finally, the policy learned by PPO. As our goal was not to obtain the best performance with an RL algorithm, but to simply establish a baseline, we used PPO default hyper-parameters as set in `Stable-Baselines3 1.4.0`. It is most likely better PPO hyper-parameters may be found. We trained PPO during 10^6 episodes, with stochastic weather generation. The training procedure was light in terms of computation: it was possible to complete the 10^6 episodes in about 1.5 hour of computation with a standard 8 core laptop. During training, the performance of PPO was evaluated on a validation environment every 10^3 episodes. We seeded the validation environment with a different seed than for the training environment. Consequently, the validation environment generated a different sequence of weather series compared to the training environment. The model with the best validation performance was saved as the result of the training.

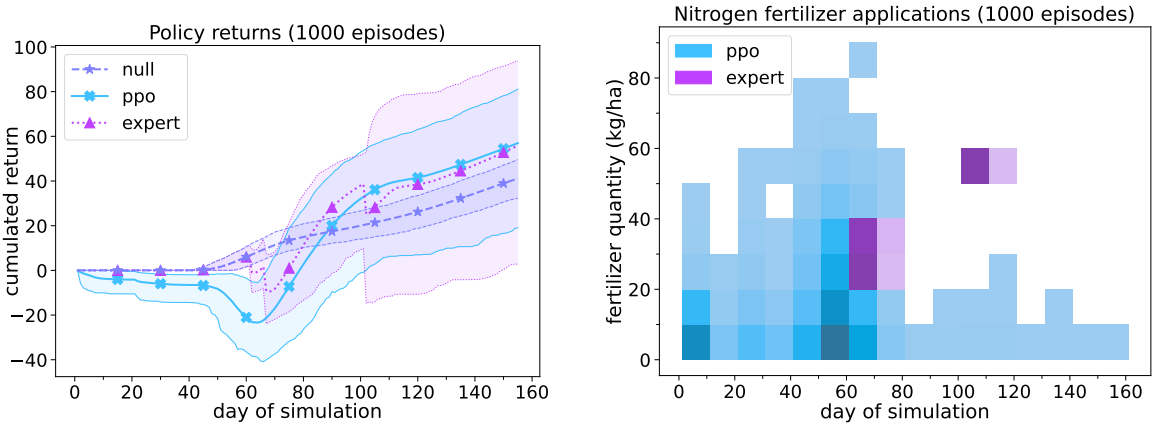
In order to compare fertilization policies, we measured their performance with 10^3 episodes in a test environment. Test environment also featured stochastic weather generation, but with isolated seeds i.e. different from the ones used in training and evaluation environments of PPO. This guaranteed that while testing policies, none of the stochastic weather series have been met by PPO during training or evaluation phases, in order to avoid over-optimistic performance measures (Stone, 1974). In the performance analysis of policies, the evolution of returns r_t provides information about the learning process from an RL perspective, but returns are still not directly interpretable from an agronomic perspective. Performance analysis of crop management options require multiple evaluation criteria (Doré et al., 2006; Duru et al., 2015). To remedy this problem, we used a subset of `DSSAT` internal state variables, provided in Table 2.4, as performance indicators. Note that these variables are not necessarily contained in the observation space of the fertilization problem (Table 2.2) because we used them for another purpose than algorithm training. Each of these performance criteria covariates with the other ones. For instance, increasing the total fertilizer amount is likely to increase the grain yield, but also likely to increase the pollution induced by nitrate leaching. The agronomic nitrogen use efficiency (ANE, Equation 1, Vanlauwe et al., 2011) is a common indicator of fertilization sustainability. For a fertilization policy π, denoting $\texttt{grnwt}^\pi$ the dry matter grain yield of the policy π (kg/ha), $\texttt{grnwt}^0$ the dry matter grain yield with no fertilization (kg/ha), and $\texttt{cumsumfert}^\pi$ the total fertilizer quantity applied with policy π (kg/ha), we have:

$$\text{ANE}^\pi(t) = \frac{\texttt{grnwt}^\pi(t) - \texttt{grnwt}^0(t)}{\texttt{cumsumfert}^\pi(t)} \tag{2.5}$$

The ANE indicates the grain yield response with respect to the null policy provided by each unit of nitrogen fertilizer. Maximizing the ANE relates to economic and environmental aspects, leading to an efficient use of fertilizer which limits risks of pollution. Performance indicators presented in Table 2.4 express a complex trade-off between conflicting objectives.

variable	definition	comment
grnwt	grain yield (kg/ha)	quantitative objective to be maximized
pcngrn	massic fraction of nitrogen in grains	qualitative objective to be maximized
cumsumfert	total fertilization (kg/ha)	cost to be minimized
–	application number	cost to be minimized
–	nitrogen use efficiency (kg/kg)	agronomic criteria to be maximized
cleach	nitrate leaching (kg/ha)	loss/pollution to be minimized

Table 2.4 Performance indicators for fertilization policies. An hyphen means gym-DSSAT does not directly provide the variable, but it can be easily derived.



(a) Mean cumulated return of each of the 3 policies against the day of the simulation. Shaded area displays the $[0.05, 0.95]$ quantile range for each policy.

(b) 2D histogram of fertilizer applications (the darker the more frequent).

Figure 2.8 Undiscounted cumulated returns and applications for the fertilization problem.

Results Figure 2.8a displays the evolution of undiscounted cumulated rewards (Equation 2.1) of policies, against the day of simulation. PPO ended with the highest mean cumulated return compared to the null and expert policies. PPO cumulated returns were less variable than with the expert policy, as can be seen from the reduced range of values between upper and lower quantiles. Figure 2.8b provides a 2D histogram of fertilizer applications, against the day of simulation. The darker a cell, the more frequent the fertilizer application. PPO fertilizer applications were more frequent at the beginning of the growing season and around day of simulation 60. The latter application date corresponds to the beginning of the floral initiation stage, see Table A.6 in Appendix. Nevertheless, the variability of rates and application dates of PPO indicated that PPO policy did not only depend on days after planting as the expert policy did, but also depended on more factors. Note that while the expert policy was deterministic, the day of simulation of applications showed slight variations. This was because in simulations, the planting date was automatically determined within a time window, depending on soil conditions, depending itself on (stochastic) weather events. Because the expert policy specified fertilizer application dates in days after planting, and not in days of simulation, a shift in planting dates consistently induced a shift in the corresponding day of simulation of fertilizer applications.

Table 2.5 shows statistics of the performance indicators detailed in Table 2.4. As expected, there was no policy that was optimal for all performance criteria. PPO policy exhibited performance trade-offs between the expert and the null policies we deemed satisfying. Grain yield and nitrogen content

	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)

Table 2.5 Mean (st. dev.) fertilization baselines performances computed using 1000 episodes. For each criterion, bold numbers indicate the best performing policy.

in grains (a nutritional criteria) were close to the ones of expert policy. On average, PPO policy consumed about 28% less nitrogen than the expert policy. Consistently, PPO ANE (Equation 2.5) –a key metric of sustainable fertilization– was about 29% greater than for the expert policy. From a practical perspective, a good fertilization policy consists in a limited number of applications during an episode, as the expert policy suggests. Indeed, each nitrogen application costs both in terms of fertilizer (as a product of natural gas) and application costs (e.g. mechanized nitrogen broadcasting). The mean number of applications of PPO (about 6) was higher than for the expert policy (3), but still practicable. Finally, PPO policy showed a slighlty lower nitrate leaching than the expert policy, which means less nitrate pollution induced by nitrogen fertilization.

Discussion We have shown that with an off-the-shelf `Stable-Baselines3` PPO implementation, we have been able to learn a relevant fertilization policy that slightly outperforms the expert fertilization policy regarding the objective function. From an agronomic perspective, PPO policy reached superior nitrogen use efficiency, with a substantially reduced nitrogen fertilizer consumption compared to the expert policy, while still yielding maize grain with satisfying quantity and quality. PPO focused nitrogen fertilizer applications at the beginning of the floral initiation stage, where maize nitrogen needs are the greatest and most crucial (Hanway, 1963). The performance of PPO is likely to increase with a proper tuning. Nevertheless:

(1) The fertilization policy an agent has learned still requires *explainability*. For instance, discovering which are the most important observation variables that determine a fertilizer application, how their values impact fertilization, and if these results are consistent with the agronomic knowledge is a requirement. For crop management decision support systems, user trust is essential (Evans et al., 2017; Rose et al., 2016). As an example, Garcia (1999) translated an RL agent policy into a set of simple decision rules (e.g. “if condition 1 or condition 2, then do ...") which were easily interpretable and usable by farmers and/or agronomists.

(2) In real conditions, each field observation costs. As an example, the growth stage (`istage`, Table 2.2), which is an observation variable, would only require a periodic visual inspection of the field. Growth stage is consequently a realistic and inexpensive observation. In contrast, the measure of the daily population nitrogen uptake, necessary to compute the return (`trnu`, Equation 2.4), would require destructive plant sampling and extensive laboratory analysis. In case the agent is trained with real field trials, then computing rewards becomes necessary, and might be problematic. Consequently, in latter case, an alternative reward function could be employed. The cost of measuring each observation variable –related to the precision of

measurement– and the frequency of these observations should be minimized for practical applications.

(3) Learning a relevant fertilization policy from scratch required 10^6 episodes. The stochastic weather time series gym-DSSAT used being sampled from independent and identically distributions, 10^6 episodes means 10^6 cultivation cycles under different weather conditions. If the objective of the experiment is to design *in silico* fertilization policies, then learning efficiency and field measure costs (2) are not problematic, but remark (1) still applies. If gym-DSSAT is used to mimic real-world conditions and the objective is to design an RL algorithm able to learn/improve from real interactions, then the learning efficiency of the off-the-shelf PPO clearly precludes any straight application in real conditions. Thereby, researchers must reduce the sample complexity of the decision problem, i.e. simplify the problem to reduce the number of samples required to solve this problem, and/or researchers must use/design other RL algorithms with improved learning efficiency (e.g. using demonstration learning Taylor and Stone, 2009, to leverage existing expert policies).

2.6.2 Execution time and reproducibility

In this section, we now briefly highlight that gym-DSSAT is a lightweight RL environment and discuss reproducibility issues.

Execution time We performed all measures of gym-DSSAT execution time for the fertilization task. The mean duration of an episode was 156 ± 7 days (1 time step was 1 simulated day), averaged over 1000 episodes. We measured the following time executions averaged over 1000 episodes, each episode lasted until 100 time steps. In practice, we insured that all episodes did not end between step 1 and step 100, so environments had to update their state for the 100 time steps. During an episode, actions were randomly sampled from the action space. On a standard 8 core laptop, the mean running time to simulate one day in gym-DSSAT i.e. taking a single step in the environment was 2.56 ± 0.22 ms. In comparison, the mean running time of taking a step in gym default MuJoCo (version 2.1.0) environment HumanoidStandup-v2 was 0.61 ± 0.21 ms. While gym-DSSAT is more expensive in time than typical gym environments, the simulation is still responsive enough for typical usage in RL experiments.

Reproducibility According to the Association For Computing Machinery (ACM), a computational experiment is said *reproducible* if an "[...] independent group can obtain the same result using the author's own artifacts"[§§§], summarized as "different team, same experimental setup". Based on our tests, we successfully reproduced the results of the present study on the same platform i.e. on the same hardware and software layers. This means that both results of gym-DSSAT and Stable-Baselines3 PPO were reproducible on the same platform. Nevertheless, as a more general reproducibility issue, we cannot guarantee the cross-platform reproducibility. Reaching cross-platform reproducibility is a generally hard issue, even for deterministic software, due to the multiple factors at stake. As an example, compiling DSSAT Fortran code with two different compilers may not result in the same exact DSSAT outputs. This is because the order of multiple arithmetic operations, despite being

[§§§] Artifact Review and Badging Version 1.1 - August 24, 2020, https://www.acm.org/publications/policies/artifact-review-and-badging-current

mathematically commutative, may not follow the same order in practice and the final result might be different because of floating point number rounding effects. To enhance reproducibility, we provide Docker containers for various Linux distributions for `gym-DSSAT` (see installation instructions[¶¶¶]).

2.7 Conclusion

In this paper, we briefly presented `gym-DSSAT`, a Reinforcement Learning (RL) environment for crop management, and exposed uses cases for fertilization and irrigation decision problems. `gym-DSSAT` is based on `DSSAT`, a celebrated crop simulator used by worldwide agronomists. To turn the original Fortran `DSSAT` software into a Python `gym` environment, we used a recently introduced library, named `PDI`. Currently, only maize fertilization and irrigation problems are available. `gym-DSSAT` can be extended to any of the 41 other crops `DSSAT` currently simulates, such as wheat or sorghum and/or to other crop operations. Further predefined scenarios will be defined to reflect a diversity of soil and climate combinations. Weather forecasts being of major interest for crop management (Hoogenboom, 2000), short time weather predictions of stochastically generated weather will be provided in the state space. `gym-DSSAT` will be connected to `Ray rllib` (Liang et al., 2017) to enhance environment scalability. For both irrigation and fertilization use cases, we showed that an untuned RL algorithm was able to learn more sustainable practices than the expert policies we considered. Beyond the use cases we have provided, further work is still required to tailor RL algorithms to the idiosyncracies of crop management problems. The performance baselines of each decision problem can be iteratively refined, for instance using the expert policy with Transfer Learning (Taylor and Stone, 2009) or extra exploration such as with Random Network Distillation (Burda et al., 2018). With a limited software development effort, `PDI` can be used to turn other existing mechanistic models into `gym` environments, hence opening the doors of a potentially large number of mechanistic models to the RL community. We hope the whole approach we used to create `gym-DSSAT` will be replicated to other complementary C, C++ or Fortran based crop models, such as STICS (Brisson et al., 2003) and other mechanistic models.

Acknowledgments

Emilio J. Padrón's work was partially supported through the research projects PID2019-104184RB-I00 funded by MCIN/AEI/10.13039/501100011033, and ED431C 2021/30 and ED431G 2019/01 funded by Xunta de Galicia. The authors acknowledge the PDI team, in particular Karol Sierocinski, for their help. The authors also acknowledge Bruno Raffin for his support and Essam Morsi for his initial help. Thanks to the `DSSAT` team, especially Gerrit Hoogenboom and Cheryl Porter for their continuous support. We thank Jacob van Etten for his remarks. We acknowledge the Consultative Group for International Agricultural Research (CGIAR) Platform for Big Data in Agriculture and we especially thank Brian King. Ph. Preux, O-A. Maillard and D. Emukpere acknowledge the support of the Métropole Européenne de Lille (MEL), ANR, Inria, Université de Lille, through the AI chair Apprenf number R-PILOTE-19-004-APPRENF. We acknowledge the AIDA team of the French Agricultural Research Centre for International Development (CIRAD) and the outstanding working environment provided by Inria and the Scool research group.

[¶¶¶] https://rgautron.gitlabpages.inria.fr/gym-dssat-docs/Installation/index.html

In Chapter 2, we showed that a widely used and already implemented RL algorithm could be successfully used to identify sustainable crop management practices, in a context of abundant data under simulated conditions. In Chapter 3, we consider the general case of a best management practice being identified amongst a larger set of practices, based on model simulations –using RL or other optimization methods–. We quantify the statistical guarantees of this identification, from crop model simulations to real field conditions. The statistical method we provide is derived from concentration inequalities (e.g. Maillard, 2019), which are widely used for bandit algorithms (e.g. Auer, 2002).

Chapter 3

Quantifying the uncertainty of decisions based on crop model simulations*

Romain Gautron [† ‡ §] Patrick Saux [¶] Odalric-Ambrym Maillard [¶]
Marc Corbeels [† ‡ ||] Chandra A. Madramootoo [**] Nitin Joshi [††]

*Article to be submitted to Environmental modeling & Software (Elsevier).
†AIDA, Univ Montpellier, France.
‡CIRAD, Montpellier, France.
§CGIAR Platform for Big Data in Agriculture, Alliance of Bioversity International and CIAT, Km 17, Recta Cali-Palmira 763537, Colombia.
¶Universit[e] de Lille, Inria, CNRS, Centrale Lille UMR 9189 – CRIStAL, F-59000 Lille, France.
||International Institute of Tropical Agriculture, PO Box 30772, Nairobi, 00100, Kenya.
**Bioresource Engineering Department, McGill University, 21111 Lakeshore Road, Ste. Anne de Bellevue QC H9X3V9, Canada.
††Department of Civil Engineering, Indian Institute of Technology Jammu, India.

Abstract

Process-based crop models are predictive models that can be used to support real-world crop management decisions. For a calibrated crop model at the field level, we provide a method to assess the minimum attainable risk level – similarly to p-values – to make a decision on a crop operation, i.e. crop planting date, for real conditions based on model simulations. We evaluate planting dates in the face of the weather uncertainty, with extensive inter-annual statistics using a weather generator. We provide both a risk-neutral (mean) and a risk-aware (mean-variance) criterion. Using the DSSAT model calibrated with a 23-year maize experiment for planting date decision, the simulated yield responses could not be assumed to be normally distributed and model errors had a non-negligible effect on conclusions with respect to best planting date of maize. The mean of the simulated yield response induced by weather exhibited high uncertainty when estimated on less than 30 years of weather data. For crop models calibrated based on a few years of field data, the identification of best practices based on model simulations is unlikely to be grounded with respect to real conditions.

Software availability

All the numerical experiments in this paper are meant to be as reproducible as possible, and the code is open source. The Python code with the necessary packages, instructions and experimental data are provided in the following public GitHub repository: https://github.com/rgautron/cropModelUncertainty. The experiments used `concentration_lib`, a jointly developed Python package available through the `pip` Python package installer: https://pypi.org/project/concentration-lib/. Crop growth simulations were made with the Decision Support System for Agrotechnology Transfer (DSSAT) software: https://dssat.net/. DSSAT is an Open Source software, and its source code available in the following Github repository: https://github.com/DSSAT/dssat-csm-os/tree/master.

3.1 Introduction

Computerized process-based crop models simulate crop growth as affected by interacting dynamics of biotic and abiotic growth factors. They are generally made up of three main compartments: soil, atmosphere and crop, that interact with crop management through time (Wallach et al., 2014). Input variables to crop growth models are typically daily meteorological variables, crop cultivar and soil properties, and crop management operations. Crop models generally predict multiple crop and soil variables, such as crop leaf area index, grain yield, total biomass, and soil water and nitrogen contents (e.g. Corbeels et al., 2016; Hoogenboom et al., 2019). Process-based crop models can be used as generative tools to provide sample responses from unknown, underlying distributions of e.g. grain yields in the case of cereals. For instance, to evaluate a crop response to a given crop management, one can characterize the inter-annual crop yield distribution induced by the weather uncertainty. More generally, stochastic weather generators coupled with crop growth simulation models can be used to extensively sample simulated responses for uncertainty analysis with respect to weather uncertainty, as a substitute or complement to historical weather records (e.g. Falconnier et al., 2019; Semenov and Porter, 1995; Soltani and Hoogenboom, 2007). A crop growth model can then be employed as an

exploratory tool to analyze 'what-if' scenarios, exploring and comparing crop management options (e.g. Adam et al., 2020; Boote et al., 1996).

Uncertainty[⌗⌗] in crop model predictions became a central research topic (e.g. Asseng et al., 2013; Chapagain et al., 2022), but methodological approaches are still limited. Wallach and Thorburn (2017) distinguished three sources of uncertainty in crop modeling: *structural uncertainty*, *input uncertainty* and *parametrization uncertainty*. First, model structural uncertainty refers to the uncertainty about how crop and soil processes are modeled. For instance, Hoogland et al. (1981) considered plant water uptake to linearly decrease with soil-rooting depth, while Li et al. (1999) used an exponentially decreasing function of plant water uptake with soil depth. Many other examples exist; such uncertainty happens because, in practice, a range of (sub)models can fit experimental data. Crop model ensembles can quantify this type of uncertainty (Falconnier et al., 2019; Maiorano et al., 2017; Opitz and Maclin, 1999). Second, model input uncertainty refers to imprecisely measured initial values of model variables (i.e. initial soil water content measured with an error) or inherently stochastic features (e.g. precipitation), which both entail a distribution of model inputs. Finally, model parametrization uncertainty refers to the uncertainty about the values of adjustable model parameters (e.g. genetic coefficients of cultivars or soil parameters). Bayesian methods for crop model parametrization can address such uncertainty (e.g. Jones et al., 2011).

In this study, we consider the special case of model prediction uncertainty for a single crop model with fixed parametrization and fixed model inputs, apart from the weather input data. We provide a decision rule for crop management that is based on the comparison of the confidence intervals for the true crop responses, for both risk-neutral (mean) and risk-averse (mean-variance) criteria. The primary outcome of our method is a quantification of the risk level of a decision on a best crop management option, similarly to a p-value. We illustrate this approach with an application to the decision on planting dates for maize in the humid continental region of Canada as a use case, and address the following three questions: (1) how many years are necessary to approximate the true mean of the weather induced uncertainty of crop model responses (in this case, maize grain yield); (2) how do model errors impact the identification of best crop management practices, and (3) how confident can one be about the identification of the best management practices for real field conditions inferred from crop model simulations.

3.2 Methods

3.2.1 Mean-variance: a risk-aware metric

Risk can broadly be defined as an event whose instance is uncertain, and which can potentially cause damages (Hoc and Rogalski, 1992). Hence risk with respect to e.g. a set of management options inherently refers to a distribution of possible responses whose realizations will be more or less favor able to the management decision maker. More specifically, when evaluating a crop management option in the face of weather variability, crop yield responses to management can be seen as uncertain realizations of a set of random variables with unknown underlying joint distributions caused by the weather uncertainty (e.g. Semenov and Porter, 1995).

[⌗⌗]According JCGM et al. (2008), the uncertainty is a "parameter, associated with the result of a measurement, that characterizes the dispersion of the values that could reasonably be attributed to the measurand". Additional vocabulary is defined in Appendix Section B.1.

Decision makers usually compare uncertain outcomes from management options with a metric that summarizes the underlying outcome distribution, the most common metric being the mean value. Using the mean criterion as a metric is considered as risk-neutral because it equally weights positive and negative outcomes of the same magnitude. In contrast, a risk-aware metric emphasizes the adverse impact of negative outcomes of random variables (Dowd, 2007). Risk-aware metrics are for instance of interest for evaluating crop management options in the context of food security, when one wants to avoid total crop failures, sometimes at the cost of overall lower mean crop yields. A well-known risk metric is the mean-variance (MV) criterion (Markowitz, 1952) a decision maker wants to maximize. For a random variable X of mean μ_X, standard deviation σ_X and an arbitrary risk-aversion factor $\rho > 0$, MV can be defined as:

$$\mathrm{MV}_X^{\rho} := \mu_X - \rho\sigma_X \,. \tag{3.1}$$

When $\rho \to 0^+$, the MV criterion equals the mean criterion. On the other hand, when $\rho \to \infty$, MV selects the smallest standard deviation irrespective of the mean, and thus is more risk averse. The MV criterion is similar to the Sharpe Ratio (Sharpe, 1966), or its inverse, the coefficient of variation, but with an easier statistical analysis due to its linearity. In this study, we use MV as a risk-aware criterion for the identification of best crop management practices, i.e. planting date of maize, based on crop growth model simulations.

3.2.2 Confidence interval comparison

In this section, we provide the statistical methods we designed for taking a crop management decision from crop growth model simulations to real field conditions with statistical guarantees.

Overview of the statistical methods

We consider the case where a decision maker wants to take a decision on crop management (i.e. best planting date for maize) for a specific region with the help of a crop growth simulation model that was calibrated for the specific growing conditions of the region. A decision is made based on an ensemble of simulated crop responses, induced by an ensemble of model inputs (i.e. weather series). We consider a single calibrated crop model f, with fixed parameters θ, which yields a prediction (or response) y^i_{sim} from an input X_i:

$$\underbrace{y^i_{\text{sim}}}_{\textbf{response}} = \underbrace{f}_{\textbf{model}}\Big(\overbrace{X_i}^{\textbf{input}} \,|\, \underbrace{\theta}_{\textbf{parametrization}}\Big) \,. \tag{3.2}$$

The calibrated crop growth model is evaluated by a prediction error ensemble $\mathcal{E}$ derived from field observations. Then, an ensemble $\mathcal{X}$ of model inputs are stochastically generated (in case, weather series) to produce I ensembles of independent and identically distributed simulated crop responses (in case, maize grain yield predictions) denoted $(\mathcal{Y}_{\text{sim}}{}^i)_{i\in\mathcal{I}}$ for the I crop management options (in case planting dates).

To determine the overall uncertainty of a decision on crop management from model simulations to real field conditions, we distinguished two terms contributing to the uncertainty of a decision (Section 3.2.2), both related to the limited number of samples drawn from an underlying, hidden distribution: (i) the uncertainty related to the model error distribution, and (ii) the uncertainty related

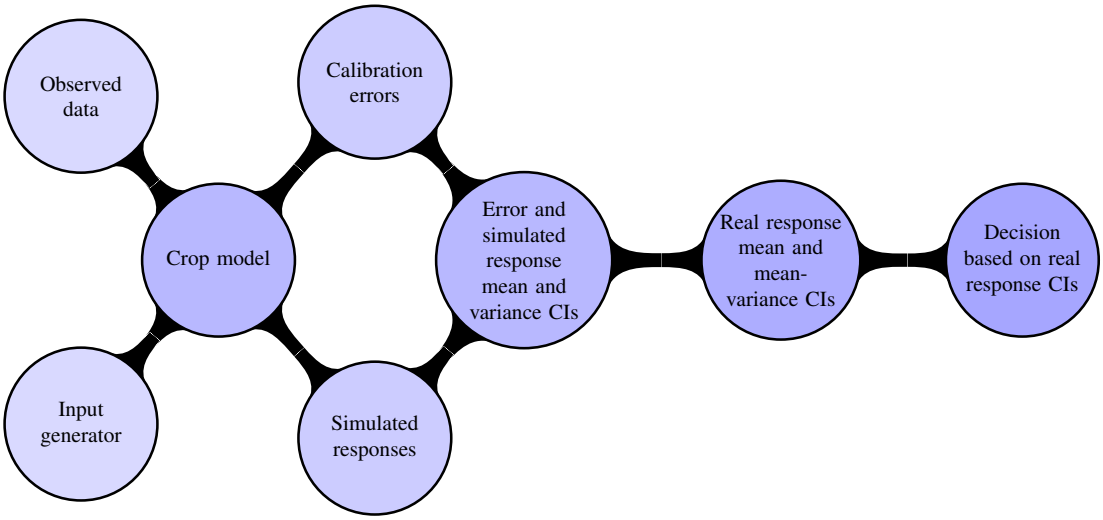


Figure 3.1 Methodological steps for crop management decision making from crop model simulations to real field conditions with statistical guarantees. CI stands for confidence interval.

to the simulated crop response distributions. The methodological steps of the management decision process are depicted in Figure 3.1. The first step is to establish valid confidence intervals for the mean and variance, respectively for the model errors and simulated response (i.e. grain yield) distributions. This is treated in Section 3.2.2. Then, depending on these confidence intervals, in Section 3.2.2, we quantify the mean and mean-variance confidence intervals for the true crop yield responses. Finally, in Section 3.2.2 we provide risk-neutral (mean based) and risk-aware (mean-variance based) decision rules of crop management, based on the confidence intervals of the true crop responses.

Uncertainty of simulated response and error distributions

We quantify the uncertainty in estimating a parameter of a distribution, based on a limited number of samples, as a confidence interval for the true value of this parameter. Formally, for a parameter ϕ_X related to the distribution of a continuous random variable X, we quantify the uncertainty in its estimation by providing a confidence interval $[\underline{\phi_X}(\delta/2),\ \overline{\phi_X}(\delta/2)]$ at risk level $\delta \in (0, 0.5)$ such that:

$$\Pr\left(\underline{\phi_X}(\delta/2) \leq \phi_X \leq \overline{\phi_X}(\delta/2)\right) \geq 1 - \delta\,. \tag{3.3}$$

Uncertainty of simulated response distributions We consider an ensemble of n multidimensional input vectors $\mathcal{X} = \{\vec{X_1}, \cdots, \vec{X_n}\}$ such that $\vec{X_i} \overset{\text{i.i.d.}}{\sim} \mathsf{G} \in \mathcal{P}(\mathfrak{I})$ where G is a stochastic input distribution and $\mathcal{P}(\mathfrak{I})$ the space of probability distributions over the input space $\mathfrak{I}$. We define the simulation model as a deterministic function $f : \mathfrak{I} \to \mathbb{R}$. We denote by $\mathcal{Y}_{\text{sim}}$ the set of n simulations generated from $\mathcal{X}$, which are realizations of the random variable Y_{sim} of known support $[0, y_{\max}]$:

$$\mathcal{Y}_{\text{sim}} = \{f(\vec{X_1}), \cdots, f(\vec{X_n})\} \tag{3.4}$$

$$= \{{\mathrm{y}_{\text{sim}}}^1, \cdots, {\mathrm{y}_{\text{sim}}}^n\}\,, \tag{3.5}$$

$${\mathrm{y}_{\text{sim}}}^j \overset{\text{i.i.d.}}{\sim} \mathrm{Y}_{\text{sim}},\ \text{and } 0 \leq \mathrm{Y}_{\text{sim}} \leq y_{\max} \text{ with probability 1.} \tag{3.6}$$

The value $y_{\max}$ represents the maximum value the random variable $\mathrm{Y_{sim}}$ can take. For instance, $y_{\max}$ can be the yield potential if $\mathrm{Y_{sim}}$ is the weather-induced crop yield distribution for a fixed crop management. We denote by $\mathbb{E}[\mathrm{Y_{sim}}] = \mu_{\mathrm{Y_{sim}}}$ its mean and by $\mathbb{V}[\mathrm{Y_{sim}}] = {\sigma^2}_{\mathrm{Y_{sim}}}$ its variance. The uncertainty in the estimation of the mean of the simulated response distribution is defined by $\underline{\mu_{\mathrm{Y_{sim}}}}(\delta/2)$, $\overline{\mu_{\mathrm{Y_{sim}}}}(\delta/2)$ such that with probability $1-\delta$:

$$\mu_{\mathrm{Y_{sim}}} \in [\underline{\mu_{\mathrm{Y_{sim}}}}(\delta/2),\ \overline{\mu_{\mathrm{Y_{sim}}}}(\delta/2)]\,. \tag{3.7}$$

Likewise, the uncertainty in the estimation of the variance of the response distribution is defined by $\underline{\sigma_{\mathrm{Y_{sim}}}}(\delta)$, $\overline{\sigma_{\mathrm{Y_{sim}}}}(\delta)$ such that with probability $1-\delta$:

$${\sigma^2}_{\mathrm{Y_{sim}}} \in [\underline{{\sigma^2}_{\mathrm{Y_{sim}}}}(\delta/2),\ \overline{{\sigma^2}_{\mathrm{Y_{sim}}}}(\delta/2)]\,. \tag{3.8}$$

Uncertainty of model error distribution Since a model is an imperfect representation of the real field conditions, each model prediction was considered to have an error with respect to its true value, such as simulated grain yields vis-à-vis observed grain yields. Furthermore, we assumed true-responses to be perfectly measured (e.g. precise field measures of yield). For an observation ${\mathrm{y_{true}}}^i$ and a simulated value ${\mathrm{y_{sim}}}^i$, we define a *model error* (or residual) e^i as:

$$e^i := {\mathrm{y_{true}}}^i - {\mathrm{y_{sim}}}^i\ . \tag{3.9}$$

Typically, at the model calibration stage, a set of model errors are computed to evaluate the model performance. We denote $\mathcal{E}$ the set of m model errors computed at the model calibration stage. Errors are supposed to belong to the same underlying error distribution E of support $[-e_{\max}, e_{\max}]$[§§]:

$$\mathcal{E} = \{e^1, \cdots, e^m\}\,, \tag{3.10}$$

$$e^j \overset{\text{i.i.d.}}{\sim} E, \text{ and } -e_{\max} \leq E \leq e_{\max} \text{ with probability 1.} \tag{3.11}$$

The model calibration procedure commonly ensures that model errors are centered[¶¶]. Such errors are known as random errors (see Supplementary Materials Section B.1). Denoting $\mathbb{E}[E] = \mu_E$, we consequently assumed :

$$\mu_E = 0\,. \tag{3.12}$$

We denote by $\mathbb{V}[E] = \sigma_E^2$ the variance of the model error distribution. The variance uncertainty for the model error distribution is given by $\underline{\sigma_E}(\delta/2)$, $\overline{\sigma_E}(\delta/2)$ such that with probability $1-\delta$:

$$\sigma_E^2 \in [\underline{\sigma_E^2}(\delta/2),\ \overline{\sigma_E^2}(\delta/2)]\,. \tag{3.13}$$

Mean and variance confidence interval computation

Here, we provide generic statistical methods to compute the confidence intervals introduced in Section 3.2.2, depending on the assumptions made about the underlying distributions. The detailed confidence interval calculations are provided in Supplementary Materials Section B.3 and a technical decision flow for the choice of confidence intervals is available in Supplementary Materials Figure B.1.

[§§] If errors are assumed to be symmetric, a natural assumption is to set $e_{\max} = y_{\max}$.
[¶¶] In Supplementary Materials Section B.2, we treat the case where $\mathbb{E}[E] \neq 0$.

Assumption 1: random variable with a Gaussian distribution If the random variable is assumed to be Gaussian, the confidence interval for the mean with unknown variance can be computed through a Student confidence interval, and the confidence interval for the variance through a chi-squared confidence interval (see Casella and Berger, 2021, and Supplementary Materials Section B.3.2).

Assumption 2: bounded random variable Here the random variable is assumed to be bounded with known support, i.e. within a known range, without further assumption. This is a basic hypothesis for physically bounded quantities, such as the maize grain yield that can take values between 0 kg/ha and some yield potential under the given growing conditions. We further describe these methods in Supplementary Materials, Section B.3.3, including refined confidence intervals for the variance which we developed.

Assumption 3: second ordered sub-Gaussian centered random variable In case a distribution is assumed to be centered but not Gaussian, we introduce a novel assumption called *second-order sub-Gaussian*, which relaxes the stringent requirement of normality while providing a tight confidence interval. Sub-Gaussian distributions are probability distributions that have tail probabilities that are upper bounded by Gaussian tails. This assumption allows for a larger class of distributions, including Gaussian distributions but also uniform or symmetric triangular distributions, amongst others. We refer to Supplementary Materials Section B.3.4, for a detailed description of such distributions and corresponding mean and variance confidence intervals.

True uncertainty calculation

We name *true distributions* the distributions which can be statistically inferred from model simulations to field observations. In order to take a crop management decision from crop model simulations to the real-field conditions, with statistical significance, our methodology relies on building valid confidence intervals for the mean and mean-variance of the true response distributions. We denote by Y_{true} the random variable corresponding to the distribution of observed responses, such as observed grain yields. Given the definition from Equation 3.9, we define Y_{true} as:

$$\mathrm{Y}_{\text{true}} = \mathrm{Y}_{\text{sim}} + E\,. \tag{3.14}$$

We denote by $\mathbb{E}\left[\mathrm{Y}_{\text{true}}\right] = \mu_{\mathrm{Y}_{\text{true}}}$ its mean, and by $\mathbb{V}\left[\mathrm{Y}_{\text{true}}\right] = {\sigma^2}_{\mathrm{Y}_{\text{true}}}$ its variance. It follows:

$$\mu_{\mathrm{Y}_{\text{true}}} = \mu_{\mathrm{Y}_{\text{sim}}} + \underbrace{\mu_E}_{0}\,. \tag{3.15}$$

Furthermore, we consider that Y_{sim} and E are uncorrelated, which gives:

$${\sigma^2}_{\mathrm{Y}_{\text{true}}} = {\sigma^2}_{\mathrm{Y}_{\text{sim}}} + \sigma_E^2 + \underbrace{2\mathrm{Cov}(\mathrm{Y}_{\text{sim}}, E)}_{0}\,. \tag{3.16}$$

Risk-neutral criterion Following Equation 3.15, the true mean uncertainty is defined by $\underline{\mu_{\mathrm{Y_{true}}}}(\delta/2),\ \overline{\mu_{\mathrm{Y_{true}}}}(\delta/2)$ such that, with probability $1-\delta$:

$$\mu_{\mathrm{Y_{true}}} \in [\underline{\mu_{\mathrm{Y_{true}}}}(\delta/2),\ \overline{\mu_{\mathrm{Y_{true}}}}(\delta/2)] \tag{3.17}$$

$$\Leftrightarrow \mu_{\mathrm{Y_{true}}} \in [\underline{\mu_{\mathrm{Y_{sim}}}}(\delta/2),\ \overline{\mu_{\mathrm{Y_{sim}}}}(\delta/2)]\,. \tag{3.18}$$

In other words, the true mean uncertainty is equivalent to the simulated mean uncertainty.

Risk-aware criterion Following Equations 3.15, 3.16, and using a union bound on events (see Supplementary Materials, Section B.3.1), the true mean-variance uncertainty is defined by $\underline{\mathrm{MV}}_{\mathrm{Y_{true}}}(\delta/2),\ \overline{\mathrm{MV}}_{\mathrm{Y_{true}}}(\delta/2)$ such that, with probability $1-\delta$:

$$\mathrm{MV}_{\mathrm{Y_{true}}} \in [\underline{\mathrm{MV}}_{\mathrm{Y_{true}}}(\delta/2),\ \overline{\mathrm{MV}}_{\mathrm{Y_{true}}}(\delta/2)]\,,\ \text{with} \tag{3.19}$$

$$\underline{\mathrm{MV}}_{\mathrm{Y_{true}}}(\delta/2) = \underline{\mu_{\mathrm{Y_{true}}}}(\delta/6) - \rho\sqrt{\overline{\sigma^2{}_{\mathrm{Y_{sim}}}}(\delta/6) + \overline{\sigma^2_E}(\delta/6)}\,, \tag{3.20}$$

$$\overline{\mathrm{MV}}_{\mathrm{Y_{true}}}(\delta/2) = \overline{\mu_{\mathrm{Y_{true}}}}(\delta/6) - \rho\sqrt{\underline{\sigma^2{}_{\mathrm{Y_{sim}}}}(\delta/6) + \underline{\sigma^2_E}(\delta/6)}\,. \tag{3.21}$$

We chose $\rho = 1$, i.e. $\mathrm{MV}_Y^{\rho=1} = \mu_Y - \sigma_Y$ as this value is commonly used in practice.

Risk for multiple option comparisons When a number I of crop management options are simultaneously compared at a total risk δ, the individual risk δ' of each confidence interval should be set to $\delta' = \delta/I$, following the union bound principle outlined in Supplementary Materials Section B.3.1.

Decision-making criteria

As a general property, all confidence intervals become tighter when the risk level δ increases (less conservative) or sample size increases (more observations). We consider a number m of error measures (e.g. from observed grain yields) and a user-defined number n of model simulations, as the cost of accessing the simulator is typically negligible compared to that of collecting more field observations. For both the risk-neutral and risk-aware criterion, the approach consists of performing a large number of model simulations, to minimize the uncertainty of simulated responses, and then to search for the minimal risk level δ such that the confidence interval of the best management option does not overlap with those of other (sub-optimal) options, a procedure subsequently called *confidence interval disjunction search*. An example of algorithm for confidence interval disjunction search is presented in Supplementary Materials, Section B.4.

Interpretation If a global risk level $\delta \in (0, 0.5)$ is found such that the confidence interval of the best option (denoted $A^\star$) is disjoint from all others, the result can be interpreted as "at risk δ, option $A^\star$ can be defined as better than all others considering the risk-neutral/risk-aware criterion". The interpretation of the disjunction risk level δ can be thought of as an alternative to p-values: for instance, $\delta < 5\%$ means that we reject with a significance level of 5% the hypothesis that A^* is not better than all other options.

3.2.3 Use case

In this Section, we describe the use case of our study to which the statistical methods provided in 3.2.2 were applied.

Field experiment and model parametrization

We simulated with the Decision Support System for Agrotechnology Transfer (DSSAT) crop model version `4.7.5.12` (Hoogenboom et al., 2019) the 23-year maize field experiment on the research farm of McGill University in Sainte-Anne-de-Bellevue, Québec, Canada (Joshi et al., 2017). Average annual temperature at the site is 7°C and total rainfall is 1033 mm (averaged from 1991 to 2013). Precipitation is well distributed during the whole year. During the growing season, which goes from May to October, average temperature and precipitation were respectively 546 mm and 16.5°C. The site has an average slope inferior to 1%. The elevation of the site is 36 m. The 2-meter deep soil is a Dystric Gleysol (FAO soil classification) with hydromorphic properties but with good overall fertility. The experimental plots had a water drainage system, and crop management was mechanized. The maize crop was rainfed. In this study, we specifically considered the treatment under conventional tillage (using a moldboard plow at 20cm soil depth at fall after harvest and a disk harrow at spring before sowing) without crop residue incorporation. Plant density was 7.6 plants/m^2. Planting was done each year between 125 to 146 days of the year (DOY). The maize crop was fertilized each year with 180 kg/ha of nitrogen, split into two fractions, 40 kg N/ha at sowing and 140 kg N/ha, on average, 2 to 6 weeks after sowing. Each year, the phosphorus fertilization consisted of 70 to 100 kg P/ha, applied at seeding, and potassium fertilization consisted of 69 to 150 kg K/ha, top dressed at the same time than nitrogen fertilization.

The DSSAT model was parameterized based on measured crop biomass and grain yield, leaf area index, soil moisture and soil nitrogen contents following standard approaches for DSSAT (e.g. as found in Gijsman et al., 2007). The different maize cultivars used in the experiment over the 23 years were classified and parameterized as three maize cultivars (indexed as cultivar 1 to 3) in DSSAT for the respective periods: 1991 to 2000 (cultivar 1), 2001 to 2007 (cultivar 2), and 2008 to 2013 (cultivar 3). We provide the parameters for each cultivar in DSSAT in Supplementary Materials, Section B.5. We considered the DSSAT model calibration for all three cultivars to be equivalent. Consequently, all error measures could be aggregated without further distinction between cultivars. Based on expert knowledge, we considered a maize grain yield potential of 20000 kg/ha for all model simulations.

Model evaluation

We performed independent maize growth simulations using DSSAT for each experimental year, from soil tillage to harvest date, with the same model calibration, except for the cultivars over the years (see Section 3.2.3) and historical weather records from the site. The model was evaluated against observed data of maize grain yields. Field observations comprised 23 years, with 21 grain yield measurements (two missing years), leading to 21 yield error measures. In order to apply the statistical methods we provided in Section 3.2.2, we evaluated the following hypotheses based on all 21 error measures:

(H1) $e^j \overset{\text{i.i.d.}}{\sim} E, j \in \{1, \cdots, m\}$,

(H2) E is normal and centered,

(H3) E is homoscedastic ,

(H4) $\mathrm{Y}_{\mathrm{sim}}$ and E are uncorrelated .

We also assumed that 'observed and simulated situations are not too different' when the calibrated crop model is used as an exploratory tool in scenario simulations (e.g. to explore the effect of planting dates). This means that the errors for observed maize grain yield data and for yield predictions in the scenario simulations share the same error distribution.

Yield distributions induced by weather uncertainty

For the scenario simulations, we used the calibrated DSSAT model coupled with the WGEN stochastic weather generator (see Richardson and Wright, 1984; Soltani and Hoogenboom, 2003). The weather generator was calibrated following Soltani and Hoogenboom (2003), based on the 23-year historical weather records of Sainte-Anne-de-Bellevue, Québec, Canada, as used in Joshi et al. (2017). To limit the complexity of this study, we did not consider the model uncertainty related to the weather generator. In the scenario simulations, we produced maize grain yield response distributions $\mathrm{Y}_{\mathrm{sim}135}$ and $\mathrm{Y}_{\mathrm{sim}165}$, for the planting dates 135 and 165 DOY. Apart from the weather variables (radiation, rainfall, and temperature), values for all other model input variables were kept constant (e.g. soil, cultivar, crop management, except the planting date). In all scenario simulations, we used the cultivar 1 from Joshi et al. (2017). The two corresponding yield distributions were approximated from 10^5 samples, i.e. simulation runs with different random states for the weather generator. The empirical means from these 10^5 samples were considered as the exact simulated means due to the large number of samples. Based on visual inspection and statistical tests, we evaluated whether simulated yield distributions could be considered as Gaussian or not.

Minimal risk level for confidence interval disjunction

Risk-neutral decision criterion As mean confidence intervals are usually tight, we only generated $n = 500$ samples, or model simulation runs, and then followed the procedure described in Section 3.2.2 to compute the minimal risk level for true mean interval disjunction. We did not consider a variable number of errors (see below) as the confidence intervals for the true mean response did not depend on σ_E.

Risk-aware decision criterion Here, we determined how reaching the confidence interval disjunction for true mean-variance was influenced by the number of centered model error measures. Thus, we run model simulations to determine how the duration of field experiments used for the model calibration impacts the minimal risk level of a decision on a best crop management option, in the face of the weather uncertainty. In order to limit the uncertainty from a limited number of samples from the simulated distributions, we generated $n = 10000$ samples for planting dates DOY 135 and 165, respectively. Then, we computed the minimal risk level for the true mean-variance yield confidence interval disjunction for the two planting dates (Section 3.2.2), as a function of the number of errors, ranging from $m = 2$ to $m = 21$. This number of errors corresponded to a hypothetical number of observed grain yields for the model calibration and was considered as a hypothetical number of years of the field experiment (originally 21 error measures corresponding to 21 different years). For simplicity, we did not recalibrate the crop model for each m. Instead, we computed the minimal risk

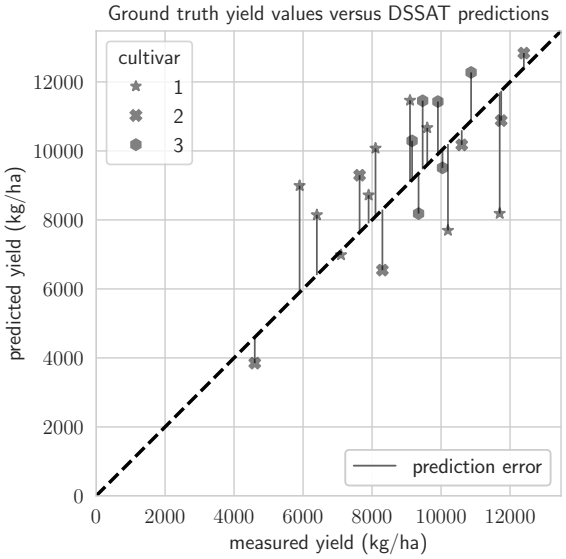


Figure 3.2 Observed values (symbols) versus DSSAT model predictions (lines) of maize grain yield for 21 years at Sainte-Anne-de-Bellevue, Québec, Canada. Observed data are from a long-term maize experiment (Joshi et al., 2017).

level for confidence interval disjunction between the two planting dates, for 1000 randomly ordered error sequences and for all $m \in \{2, \cdots, 21\}$, based on the original sequence of 21 yield errors.

3.3 Results

3.3.1 Model evaluation

The DSSAT model calibration is presented in Figure 3.2 for the three successive maize cultivars cultivars 1, 2 and 3 (see Supplementary Materials, Section B.5), over the 21 experimental years that had observed maize grain yields. The usual assumption of centered error ($\mathbb{E}[E] = 0$), referred to as random errors in Section 3.1, was supported by visual inspection of the residuals (see Figure B.5 in Supplementary Materials). Assuming the errors are centered, independent and identically distributed, the empirical error standard deviation $\hat{\sigma}_E = \sqrt{\sum_{j=1}^{m}(e^j)^2/m}$ is then equivalent to the root mean squared error. For the 21 error measures, corresponding to the 21 years of the experiment with observed grain yields, we obtained $\hat{\sigma}_E \approx 1750$ kg/ha.

3.3.2 Hypothesis testing

As stated in Section 3.2.2, the uncertainty of a decision on crop management from model simulations to real field conditions depend on the underlying statistical hypotheses made about the model error and simulated crop yield response distributions. We describe the results of hypothesis testing for both the distribution of simulated yields and the distribution of model errors.

Simulated yield distributions. Figure 3.3 and Table 3.1 show the simulated maize grain yield distributions estimated from 10^5 samples and the corresponding statistics. The difference between

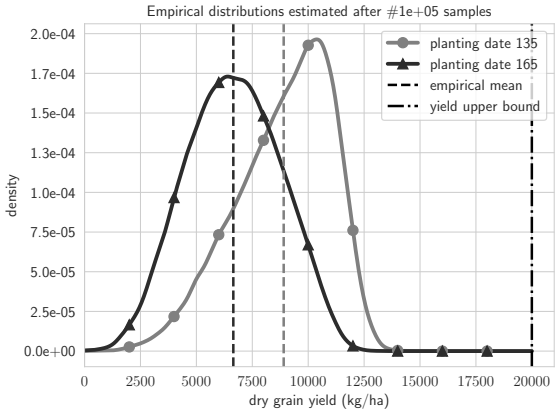


Figure 3.3 Simulated maize grain yield distributions induced by weather uncertainty for planting at days of the year (DOY) 135 and 165 (10^5 samples) at Sainte-Anne-de-Bellevue, Québec, Canada. The DSSAT model (Hoogenboom et al., 2019) was used for the simulations.

Table 3.1 Statistics of simulated maize grain yield responses for planting dates DOY 135 and 165 at Sainte-Anne-de-Bellevue, Québec, Canada, computed from 10^5 samples. The DSSAT model (Hoogenboom et al., 2019) was used for the simulations. Due to the high number of samples, these statistics are considered as exact.

	$Y_{sim\,135}$	$Y_{sim\,165}$
μ (kg/ha)	8918	6669
σ (kg/ha)	2157	2107

mean simulated crop yields of the two planting dates is about $\mu_{Y_{sim\,135}} - \mu_{Y_{sim\,165}} \approx 2250$ kg/ha. A first visual inspection of Figure 3.3 reveals some skewness in the case of planting at DOY 135. The mean and mode are distinct (which would not be the case with normally distributed data). This indicates that the simulated yield distributions induced by the weather uncertainty can be subject to departures from normal distributions. A more formal, quantified approach using standard normality tests rejected the Gaussian hypothesis for Y_{sim} (Supplementary Materials, Section B.6.1). Furthermore, we empirically showed that the confidence intervals under the Gaussian center error distribution assumption (assumption 1 in Section 3.2.2) were not appropriate as they empirically exceeded their supposed theoretical risk levels, confirming a substantial deviation from a normal distribution (Figures B.3b and B.4 in Supplementary Materials, Section B.6.1). As a consequence, Y_{sim} was considered to support the sole boundedness assumption (assumption 2) that grain yields are contained within the bounded interval $[0, 20000]$ kg/ha (20000 kg/ha being the yield potential).

Model error distribution Drawing on the statistical tests of the residuals (Supplementary Materials, Section B.6), all hypotheses in Section 3.2.3 regarding model error distributions were supported, at a risk level of 5%: the model errors had a centered Gaussian distribution E, error measures were independent and identically distributed, E was homoscedastic and uncorrelated with Y_{sim}.

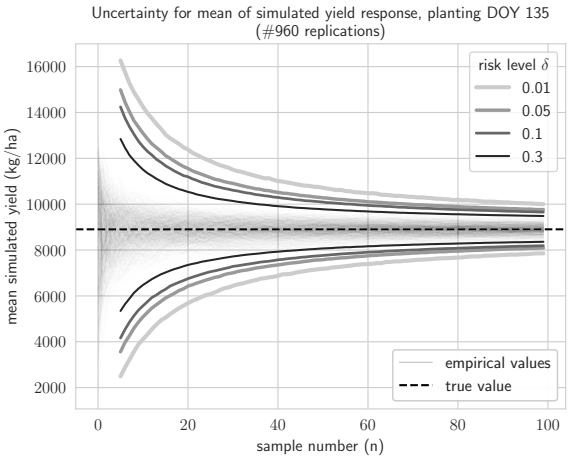


Figure 3.4 Uncertainty for the mean of the simulated maize grain yield at Sainte-Anne-de-Bellevue, Québec, Canada, as a function of the sample size for weather-induced yield distributions (planting date at day of year 135), at different risk levels. Results are computed under assumption 2 (see Section 3.2.2, eq. B.15). The true mean is the dashed horizontal line. 960 replications were performed using the DSSAT crop model (Hoogenboom et al., 2019).

Errors for scenario simulations We do not have numerical results to support the hypothesis that errors for observed grain yield and the yield scenario simulations are of the same order of magnitude. We support this assumption by the subjective judgement that the DSSAT model showed a coherent behavior, given that all hypotheses about model errors were supported (see above), and that simulated yield distributions contain values in the expected range of maize grain yields for the considered growing conditions at the experimental site (Figure 3.3).

3.3.3 Uncertainty of simulated yield distributions

Because all confidence intervals provided in Section 3.2.2 are random variables, we computed them based on 960 replications (12 replications by core, on an 80-core machine), to show their variability.

Uncertainty of the mean for simulated yield distributions Under the sole boundedness hypothesis (assumption 2, grain yields are contained within the bounded interval $[0, 20000]$ kg/ha), Figure 3.4 presents confidence intervals for the value of $\mu_{Y_{sim}}$ (see eq. B.15 in Supplementary Materials, Section B.3) at different risk levels, using the method by Phan et al. (2021). With less than 30 years of simulations (1 simulation corresponding to 1 cropping year), for all considered risk levels, the uncertainty in estimating the mean maize grain yield is larger than 2000 kg/ha.

Uncertainty of the variance for simulated yield distributions Under assumption 2 of grain yields being contained within the bounded interval $[0, 20000]$ kg/ha, Figure 3.5 depicts confidence intervals for the value of $\sigma_{Y_{sim}}$ (see equation B.18 in Supplementary Materials, Section B.3) at different risk levels δ. Compared to the mean confidence interval for $\mu_{Y_{sim}}$, about 10 times more samples are required to achieve the same confidence interval width than for $\sigma_{Y_{sim}}$.

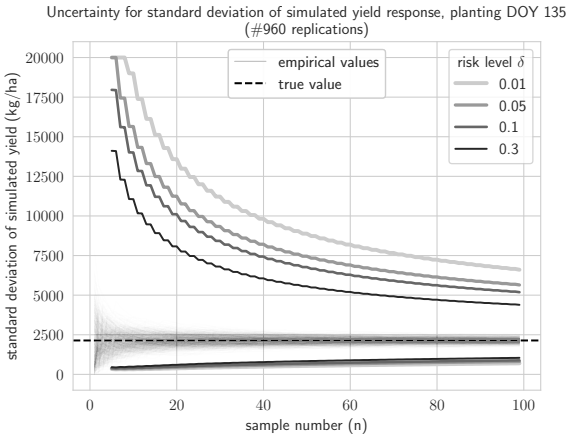


Figure 3.5 Uncertainty for the standard deviation of the simulated yield response as a function of the sample size for weather-induced yield distribution (planting date at the day of the year (DOY) 135), under different risk levels. Results are computed under assumption 2 (see Section 3.2.2), eq. B.18. The true standard deviation is the dashed horizontal line. 960 replications were performed using the DSSAT crop model (Hoogenboom et al., 2019).

3.3.4 Uncertainty of model error distribution

Figure 3.6 shows the confidence intervals of σ_{E} for the 21 observed model errors at different risk levels, respectively for the centered Gaussian (assumption 1), sole boundedness (assumption 2) and second-order sub-Gaussian (assumption 3) hypotheses. The error confidence intervals with the second order sub-Gaussian hypothesis were relatively close to the strictly Gaussian ones. On the other hand, the confidence intervals derived under the sole boundedness hypotheses proved to be very wide.

3.3.5 Minimal risk level for decision

In assessing the minimal risk levels for decision, $\mathrm{Y}_{\mathrm{sim}}$ is supposed to be solely bounded in $[0, 20000]$ kg/ha (assumption 2). We describe the results, for risk-neutral and risk-aware criteria, depending on the hypotheses about the model error distribution.

Risk neutral metric

After searching for the minimal risk level for interval disjunction (see Section 3.2.2), only assuming the model error distribution to be centered and considering the mean (risk-neutral) criterion we found that at risk level $\delta = 0.002\%$, planting at DOY 135 performs better than planting at DOY 165. Empirical mean maize grain yields were 8843 kg/ha for DOY 135 against 6699 kg/ha for DOY 165. The corresponding confidence intervals are presented in Figure B.8, Supplementary Materials, Section B.7.

Risk-aware metric

Considering all 21 error measures After searching for the minimal risk level for interval disjunction, under the centered Gaussian error hypothesis (assumption 1), we found that at risk level $\delta = 0.065\%$, planting at DOY 135 performs better than planting at DOY 165 considering the mean-variance criterion.

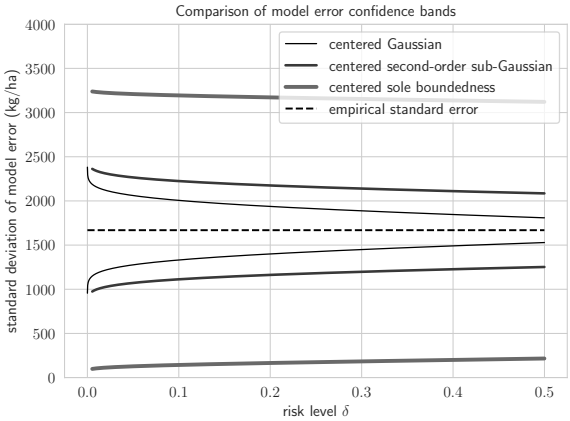


Figure 3.6 Uncertainty comparison of model (DSSAT model Hoogenboom et al., 2019) error standard deviation for 21 error measures (maize field experiment in Sainte-Anne-de-Bellevue, Québec, Canada, see Joshi et al., 2017). The Gaussian hypothesis refers to assumption 1 in Section 3.2.2 (Eq. B.12), the centered second order sub-Gaussian hypothesis to assumption 2 (Eq. B.29) and the sole boundedness hypothesis to assumption 3 (Eq. B.18).

Empirical mean-variance yields were 6165 kg/ha for DOY 135 against 3955 kg/ha for DOY 165. On the other hand, the centered second-order sub-Gaussian error hypothesis (assumption 3) resulted in planting at DOY 135 performing better than planting at DOY 165 at risk level $\delta = 6.878\%$. Confidence intervals are illustrated in Supplementary Materials, Section B.7, in Figures B.9 and B.10 respectively. No solutions were found assuming the sole boundedness hypothesis for model error distribution.

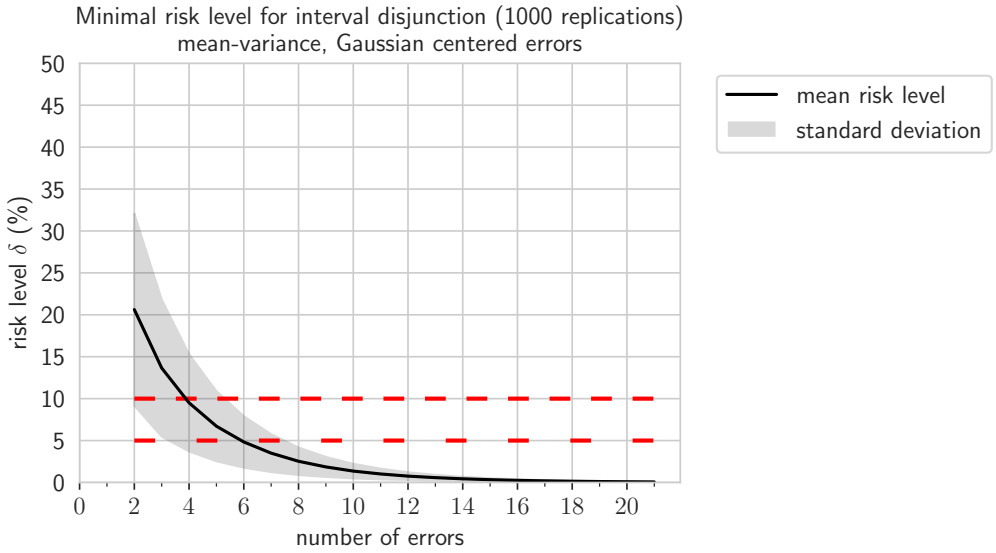
Considering an increasing number of error measures Figure 3.7 shows the minimal risk levels for confidence interval disjunction between planting date DOY 135 and 165, depending on the hypothesis made about model error distributions, in the case of the risk-averse (mean-variance) criterion. When model errors were assumed to be centered and Gaussian, the typical 5% and 10% risk level were on average reached respectively after 4 and 6 error measures corresponding to the number of years of the field experiment (Figure 3.7a). When model errors were assumed to be centered and second-order sub-Gaussian, the typical 5% risk level was, on average, reached after 19 errors measures. On the other hand, the 10% risk level was still not reached after 21 errors measures (Figure 3.7b).

3.4 Discussion

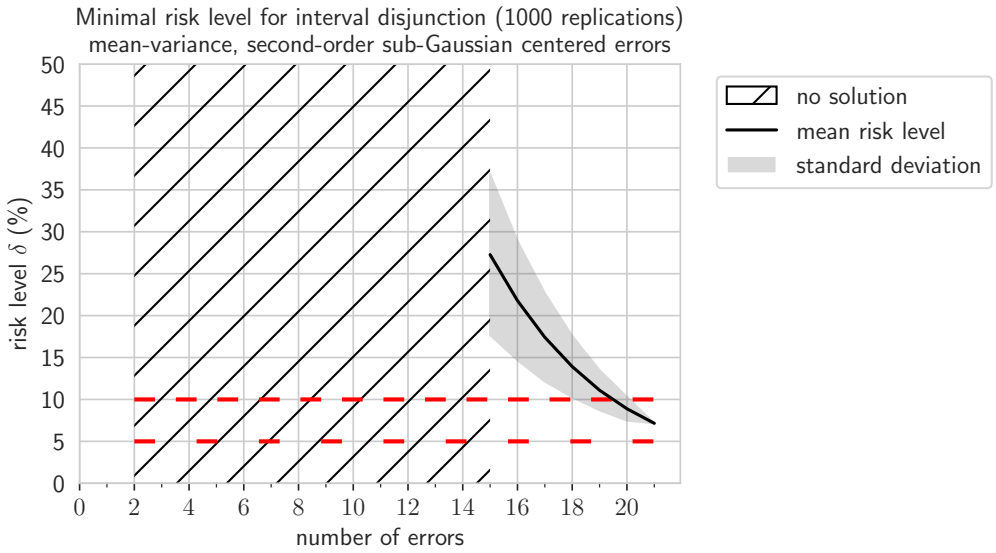
3.4.1 Validity of the statistical analyses

Statistical guarantees for a crop management decision from crop growth model simulations to real field conditions are largely determined by the model calibration. Assuming a centered model error distribution and considering the risk-neutral decision criterion (the mean), statistical guarantees for a decision can be obtained with an arbitrate risk level on condition of enough model simulations (the number is not known in advance). However, with the risk-aware decision criterion, i.e. mean-variance[***], error distribution and variance had a great impact on the risk level of a decision.

[***]The choice of the mean-variance is discussed in Supplementary Materials, Section B.7.2.



(a) Mean-variance with centered Gaussian error hypothesis.



(b) Mean-variance with centered second-order sub-Gaussian error hypothesis.

Figure 3.7 Minimal risk level for confidence interval disjunction between maize planting date DOY 135 and 165 at Sainte-Anne-de-Bellevue, Québec, Canada, as function of the number of errors for (a) the centered Gaussian error hypothesis, and (b) the centered second-order sub-Gaussian error hypothesis. 1000 replications were performed using the DSSAT crop model (Hoogenboom et al., 2019). The red lines indicate the typical 5% and 10% risk levels.

Computing tight confidence intervals on the standard deviation of a centered distribution required more samples in case of the second-order sub-Gaussian hypothesis than the Gaussian hypothesis. In our example, even if the mean simulated difference of maize yield between two defined planting date was larger than 2000 kg/ha, relaxing the centered Gaussian hypothesis by the centered second-order

sub-Gaussian error largely affected the conclusions. Confidence interval disjunction then required 19 experimental years to reach the 10% risk level for decision making on planting dates of maize, instead of 6 years for the centered Gaussian hypothesis.

Subjectivity of hypotheses about model error distribution In many cases, the Gaussian distribution for model errors is assumed, arguing that the underlying complexity of the variable described by the true response distribution Y_{true} may be accurately captured by the model, resulting in normal residuals. Nevertheless, such a strong assumption should be supported by statistical evidences. With small sample sizes (e.g. $m < 20$), normality tests are likely to show decreased power (Öztuna et al., 2006). Visual inspection is of prime interest to support the Gaussian error distribution hypothesis (Öztuna et al., 2006), but the resulting conclusions remain subjective. The same remarks apply to the centered model error hypothesis. With very few error measures (e.g. $m < 10$), supporting the hypotheses described in Section 3.2.3 is likely to be speculative. Furthermore, the confidence intervals of the error standard deviation are likely to be wide due to the small sample size.

Subjective hypothesis for exploratory analysis We assumed that the distribution of model errors was the same for the crop management options to be explored (i.e. new planting dates for which the crop model was not calibrated) as for the model calibration. Using model errors from the data set used for model calibration is always an optimistic error estimation for not observed scenarios (commonly called resubstitution error by the practitioners). A proper unbiased error estimation would require cross-validation (Stone, 1974), or (better) nested-cross-validation (Cawley and Talbot, 2010), but these procedures are too data intensive to be meaningful with limited field observations, as mostly found with crop model applications. In cases where the model calibration is performed with an automatic method, a cross-validation is required as the risk of overfitting[†††] is increased considering the extensive parameter optimization at stake. On the other hand, when the model calibration is made by hand using expert knowledge, supported by interpretable intermediary model outputs that are critically evaluated along the calibration process, the risk of overfitting can be considered limited. The present case study falls into the latter category.

Hypotheses about simulated distributions As a consequence of the central limit theorem (Casella and Berger, 2021), one may argue that a confidence interval for the mean of a distribution, such as the simulated crop yield distribution of Y_{sim}, can be asymptotically estimated as if the samples were Gaussian, provided the sample size is sufficiently large. Nevertheless, 'sufficiently large' remains unclear in practice, as distributions are not *a priori* known and their shapes are potentially of a wide range. Although the bounded random variable hypothesis is more sample-intensive than its Gaussian counterpart, the former is a more general, non-parametric hypothesis which encompasses a larger class of distributions, with concentration properties around the mean and variance similar to those of Gaussian distributions (bounded distributions are in particular sub-Gaussian, see Boucheron et al. (2013) and Supplementary Materials, Section B.3). Given the fact that the user can freely choose the number n of simulations, we advocate for relaxing the strict Gaussian hypothesis and using confidence intervals related to any bounded random variable.

[†††] A model that is calibrated and predicts well the field observations used at the model calibration stage, but which predicts poorly unseen observations.

Confidence interval disjunction risk level as an alternative to p-values Assessing the better performance of one crop management option over possible alternatives with respect to the mean criterion is classically done in the framework of statistical hypothesis testing, using e.g. a *t*-test or a one-way ANOVA (see Casella and Berger, 2021). The main outcome of these procedures is a p-value, which is compared to a prescribed significance level, typically 5%, to decide whether to accept or reject the null hypothesis that all options have the same mean response. However, p-values have recently come under scrutiny due to their frequent misuse and erroneous interpretation (Wasserstein and Lazar, 2016), leading to a so-called replication crisis in empirical sciences (e.g. Pashler and Wagenmakers, 2012). Among the theoretical shortcomings of p-values are the stringent distributional assumptions (e.g. the *t*-test assumes Gaussian samples), which are only valid for asymptotically large sample sizes [‡‡‡]. In addition, such tests are often designed for the mean criterion and are not directly applicable to risk-averse settings. In contrast, the confidence interval disjunction risk level is founded on finite-sample concentration statistics, and is thus valid even for small samples. It applies to a broad range of distributional assumptions (from Gaussian to nonparametric bounded) and is flexible enough to handle the mean-variance criterion. Other alternatives to p-values make use of finite-sample concentration, such as *E-values* (Grünwald et al., 2020; Vovk and Wang, 2021).

3.4.2 Comparison to existing quantification of crop model errors

To our knowledge, no study has provided a method to quantify the statistical guarantees that, based on model simulations, a given crop management option is better than other alternative management options in real field conditions. Few studies provided methodological contributions for the case of a single crop model, with fixed parametrization and uncertain model inputs. Yang et al. (2014) discussed the use of various statistical tools, including statistical tests with a centered model error as a null hypothesis, to evaluate crop models calibrated with field observations. However, they did not discuss the uncertainty of the estimates of model error standard deviations, due to the small number of error measures.

On the other hand, Willmott et al. (1985) proposed the use of bootstrap sampling (Efron and Gong, 1983), a non-parametric method, i.e. that does not assume data to be sampled from a given probability distribution (e.g. Gaussian distribution), to provide confidence intervals for the statistics of a model evaluation. They designed their methods for meteorological models, but these methods are applicable to a larger class of mechanistic process-based models, including crop growth models. Similarly, Roux et al. (2014) introduced a method to build approximate confidence intervals over the simulated responses, considering both model input and model error uncertainties. Their method is also based on bootstrapping. Bootstrapping methods are convenient because they are non-parametric, but require a large number of samples to be valid. A common criterion states that at least 1000 samples are required for bootstrap confidence intervals to be reliable, and this minimum sample size remains unclear in practice as it is likely to be different for every underlying distribution. With few samples, especially for the model error distributions, bootstrap confidence intervals will be over-optimistically tight. In contrast, our (parametric) statistical method is valid for any finite number of observations.

[‡‡‡]These distributional assumptions can be partially lifted by the use of nonparametric tests (e.g. Mann-Whitney, Kruskall-Wallis), but at the cost of lower statistical power.

3.4.3 Implications for decisions made from model simulations

In this study, we provided statistical guarantees for decision-making on the planting date of maize at a risk level lower than 10%, even in the case of a relaxed centered Gaussian error distribution hypothesis. This was possible because of a widely tested crop growth model, DSSAT, that was calibrated on a 23-year maize field experiment. Such long-term experiments are rather rare (e.g. Berti et al., 2016). Most calibrations of crop models face high uncertainty in model inputs (e.g. soil (initial) properties and crop variety parameters), and most of the time rely on partially available in-season observations in the field experiments (such as crop leaf area index, aboveground crop biomass or soil water measures) for calibration (e.g. in sub-Saharan Africa Falconnier et al., 2019).

Most studies concluding on planting date decisions in the face of weather uncertainty are based on simulations with crop models that are calibrated with only based few years of field data (see Anapalli et al., 2005; Egli and Bruening, 1992; Soler et al., 2007; Tang et al., 2018, for various growing conditions). The decision criterion is generally the mean simulated crop yield, averaged over historical (weather) records, often comprising more than 20 years. These studies are unlikely to provide strong statistical evidences which support the decisions (see examples above, and Section 3.4.1). In the case multiple planting dates are tested on the same site and year, the model errors for yield prediction are likely to be correlated (White et al., 2007), and these correlations should be taken into account during the crop model evaluation to avoid an over-optimistic estimate of model prediction performance with respect to the weather uncertainty.

However, even with limited of statistical significance, this does not mean that these model-based decisions are deemed irrelevant. Model evaluation involves usually a multi-criteria assessment (e.g. soil water and nitrogen dynamics, aboveground crop biomass, or the final grain yield), follows a structured procedure (White et al., 2005), and multiple performance statistics are used (Yang et al., 2014). Following the recommendations of Wasserstein and Lazar (2016), a single statistical criterion, such as the risk level of a decision, should not be used as a mechanical "bright-line" which states the trueness or falseness of a decision. Rather, such criterion should used as an element of a larger decision context which includes the assessment of the methods of data collection or data generation. For instance, in the case of the use of crop model, practitioners assess the consistency of multi-variable model responses with the expected agronomic behaviour in the field (Hochman et al., 2009). We generally advocate the recommendations made from model simulations to be considered uncertain and cautiously confirmed with agronomic expertise, and if possible, additional field trials.

3.5 Conclusion

We addressed the quantification of a minimal risk level of crop management decisions made for real field conditions, based on simulated crop yield results using a crop growth model (DSSAT). We specifically considered inter-annual weather statistics using a stochastic weather generator (WGEN), addressing weather uncertainty, and used both risk-neutral (mean) and risk-aware (mean-variance) decision criteria. We illustrated the approach for decision making on the planting date of maize using a 23-year maize experiment (21 model error observations) in the humid continental region of Canada. The use case revealed that, even in a favorable data context, the ability to conclude on the better yield performance of one planting date over another is highly constrained by model errors. Simulated maize

grain yields could not be assumed to be normally distributed, but required instead a less stringent statistical hypothesis which only assumed that the simulated yields are bounded in a known range.

In the use case, estimating the mean simulated maize yield with less than 30 years of historical weather records showed an uncertainty greater than 2000 kg/ha, for all the risk levels we considered. To obtain comparable confidence intervals for the simulated variance as for the mean, about tenfold increase in samples is required. Although the two selected planting dates (DOY 135 and 165) had a difference of simulated yield greater than 2000 kg/ha, four and 19 years of field experiment were required to reach the typical 10% risk level considering the risk-aware decision criterion, when model errors were respectively assumed to be centered and Gaussian, or centered and second-order sub-Gaussian. Making assumptions on the model error distribution is a partially subjective but necessary step for quantifying uncertainty before making a management decision based on crop model simulations. In most applications of crop models, the very small number of error measures do not allow to make management decisions that are statistically grounded. This does not mean that these decisions are deemed meaningless, as crop models are calibrated following a structured procedure, based on multiple evaluation criteria, and the model behaviours are assessed against the expected agronomic behavior of crop management. Therefore, we recommend crop management decisions made from model simulations to be confirmed by sound agronomic expertise and, if possible, additional field experiments.

Acknowledgments

The authors acknowledge the Consultative Group for International Agricultural Research (CGIAR) Platform for Big Data in Agriculture for partially funding this research, and address special thanks to Brian King. We also acknowledge the French Agricultural Research Centre for International Development (CIRAD) for it contribution in funding this work. This work has been further supported by the French Ministry of Higher Education and Research, Hauts-de-France region, through INRIA within the team-project Scool and the MEL. Finally, the authors ac- knowledge the funding by the I-Site ULNE through the projects R-PILOTE-19-004-APPRENF and Bandits For Health (B4H), as well as through the project A.Ex. SR4SG. The simulation experiments presented in this paper were carried out using the Grid5000 testbed, supported by a scientific interest group hosted by INRIA and including CNRS, RENATER and several universities as well as other organizations (see https://www.grid5000.fr). We acknowledge Gatien Falconnier and Myriam Adam for their guidance on the calibration of DSSAT. The authors acknowledge Dr Ajay K. Singh, Mr Peter Kirby and several other students at McGill University who collected field data over 23 years, for sharing their work and experimental data used in Joshi et al. (2017). We acknowledge the numerous funding agencies supporting McGill University, particularly Fonds de recherche du Québec – Nature et technologies FRQ-NT), Natural Sciences and Engineering Research Council (NSERC) and the Canadian Foundation of Innovation (CFI) for financing the field research. We gratefully acknowledge McGill University's Macdonald Campus Research Farm for providing the logistical support for the field experiments. The authors acknowledge the DSSAT team (https://dssat.net/) for their hard work making their tools Open Source.

In Chapter 3, we showed that for most applications, the statistical guarantees of the identification of best crop operations –be it with RL or other optimization methods–, from crop model simulations to real-field conditions, are highly constrained by the crop model prediction errors. Furthermore, exploring new crop operations through crop model simulations is inherently limited by the processes the crop models simulate. In Chapter 4, based on a bandit algorithm, we design an RL-based identification method of best crop management options, targeting a direct identification from field trials. With this objective, crop model simulations are still useful to mimic real-world problems, and to compare numerical methods of best crop management identification. Yet, the main challenge is to design an RL-based identification method which requires a practicable number of trials of crop operations (e.g. hundreds to thousands), as opposed to the millions of trials used in Chapter 2.

Chapter 4

Towards an efficient and risk aware strategy to guide farmers in identifying best crop management*

Romain Gautron [† ‡ §] Dorian Baudry [¶] Myriam Adam [|| ** ††]
Gatien Falconnier [† ‡ ‡‡] Marc Corbeels [† ‡ §§]

*Article to be submitted to Computers and Electronics in Agriculture (Elsevier).
†AIDA, Univ Montpellier, France.
‡CIRAD, Montpellier, France.
§CGIAR Platform for Big Data in Agriculture, Alliance of Bioversity International and CIAT, Km 17, Recta Cali-Palmira 763537, Colombia.
¶Univ. Lille, CNRS, Inria, Centrale Lille, UMR 9198-CRIStAL, F-59000 Lille, France.
||CIRAD, UMR AGAP Institut, Bobo-Dioulasso 01, Burkina Faso.
**UMR AGAP Institut, Univ Montpellier, CIRAD, INRAE, Institut Agro, Montpellier, France.
††Institut National de l'Environnement et de Recherches Agricoles (INERA), Burkina Faso.
‡‡International Maize and Wheat Improvement Centre (CIMMYT)-Zimbabwe, 12.5 km Peg Mazowe Road, Harare, Zimbabwe.
§§International Institute of Tropical Agriculture, PO Box 30772, Nairobi, 00100, Kenya.

Abstract

Identification of best performing fertilizer practices among a set of contrasting practices with field trials is challenging as crop losses are costly for farmers. To identify best management practices, an "intuitive strategy" would be to set multi-year field trials with equal proportion of each practice to test. Our objective was to provide an identification strategy using a bandit algorithm that was better at minimizing farmers' losses occurring during the identification, compared with the "intuitive strategy". We used a modification of the Decision Support Systems for Agro-Technological Transfer (DSSAT) crop model to mimic field trial responses, with a case-study in Southern Mali. We compared fertilizer practices using a risk-aware measure, the Conditional Value-at-Risk (CVaR), and a novel agronomic metric, the Yield Excess (YE). YE accounts for both grain yield and agronomic nitrogen use efficiency. The bandit-algorithm performed better than the intuitive strategy: it increased, in most cases, farmers' protection against worst outcomes. This study is a methodological step which opens up new horizons for risk-aware ensemble identification of the performance of contrasting crop management practices in real conditions.

Software availability

All the numerical experiments in this paper are meant to be as reproducible as possible, and the code is open source. The Python code with the necessary packages, instructions and experimental data are provided in the following public GitLab repository: https://gitlab.inria.fr/rgautron/batch-cvts/-/tree/master. The simulations are performed with `gym-dssat` (https://gitlab.inria.fr/rgautron/gym_dssat_pdi), a modified version of the Decision Support System for Agrotechnology Transfer (DSSAT) software (https://dssat.net/).

4.1 Introduction

Identifying site-specific best-performing crop management is crucial for farmers to increase their income from crop production, but also for minimizing the negative environmental impact of cropping activities (Tilman et al., 2002). However, due to weather variability, the identification of these practices can be challenging, in particular with rainfed farming: what worked best in a wet year, might not work in the next season, when rainfall is less (Affholder, 1995). In fact, the performance of crop management at a given site has an underlying "hidden" distribution due to inter-annual weather variability, thus creating great uncertainty (Fosu-Mensah et al., 2012). Because crop management decisions are recurrent, i.e. they are repeated for each new crop growing season, the identification of optimal crop management falls into the category of sequential decision making under uncertainty (Gautron et al., 2022a). Computer-based decision support tools can allow farmers to make more informed (less uncertain) decisions about their cropping practices from one year to the next, and can facilitate farmers' risk management in the face of seasonal weather variability (Hochman and Carberry, 2011). There exist numerous decision support tools of widely ranging complexity for crop management, introduced to farmers with varying degrees of success (Gautron et al., 2022a).

Machine learning (ML) and more generally artificial intelligence (AI) can help address sequential decision making under uncertainty. In particular, the bandit algorithm paradigm (Lattimore and

Szepesvári, 2020) considers a decision-maker, called agent, who repeatedly faces a choice between contending actions, and has to iteratively improve its decisions with trials. The canonical bandit problem originates from clinical trials with sequential drug allocation (Thompson, 1933). At each time step, the agent chooses one action (i.e., one drug for a patient) amongst a set of possible actions. Each action provides a reward (i.e.; tumor cell reduction after taking the drug), drawn from a corresponding unknown reward distribution (i.e., the distribution of tumor cell reduction for the drug). The optimal action has the reward distribution with the highest mean reward (i.e., the highest mean tumor cell reduction). The objective of the agent is to sequentially choose actions such that the expected sum of rewards is maximized. Maximizing the total expected rewards is equivalent to minimizing the regret, which is a measure of the total losses that occur with sub-optimal actions (Robbins, 1952).

Iteratively, the agent refines his next decision based on all previous results. To know how a given action performs, a sufficient number of (possibly poor) rewards is required: this is the exploration phase. To maximize the expected sum of rewards, the previous actions that provided good results so far must be selected more frequently; this is the exploitation phase. Bandit algorithms aim at finding the right balance between exploration and exploitation. This *exploration-exploitation dilemma* is a reality for farmers when implementing crop management. Farmers typically want to minimize overall crop yield losses and typically explore the performance of promising new crop management practices on small test plots (Cerf and Meynard, 2006; Evans et al., 2017). They avoid potentially large crop yield losses from new management by managing a gradual transition between the current management and the promising new one(s), based on the results they obtain on the small test plots.

The objective of this paper is to develop a novel strategy to identify best crop management. We set as baseline an "intuitive strategy" which consists in identifying the best crop management through multi-year field trials in which a set of crop management practices is tested in an equiproportional way. We compare this "intuitive strategy" to a novel crop management identification strategy, based on a bandit algorithm. This novel identification strategy aims to minimize farmers' yield losses occurring during the identification process, compared to the intuitive strategy. Thus, we test the hypothesis that bandit algorithm can help farmers to better identify the best crop management for their context, while further minimizing crop yield losses related to sub-optimal choices in new crop management.

Our case study considers the rainfed maize production in southern Mali, and we compare the performance of both crop management identification strategies based on maize growth simulations using a calibrated crop model in order to mimic real-world performance of crop management. The novel identification strategy does, however, not depend on model simulations, and ultimately aims at being applied in real field conditions. As for crop management, we focus on nitrogen fertilization. Tailoring nitrogen fertilizer recommendations to farmers' contexts is known to be challenging. Indigenous soil nitrogen supply, depending to a large extent on past-season events, is not accurately known to farmers, whilst in-season nitrogen mineralization depends largely on weather events(Morris et al., 2018), themselves uncertain. Crop nitrogen requirements, such as with maize, are related to specific crop growth stages (Hanway, 1963) and excessive mineral nitrogen supply can induce nitrate leaching, especially in wet conditions (Meisinger and Delgado, 2002). Therefore, there are *a priori* no upfront optimal nitrogen fertilizer practices.

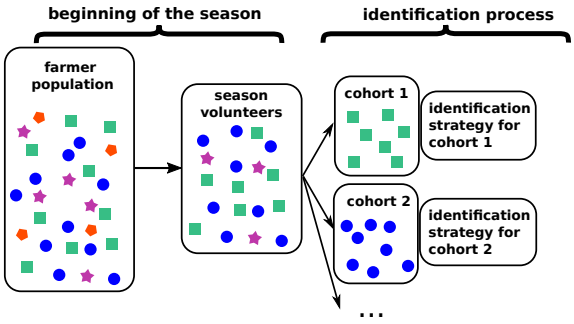


Figure 4.1 Yearly process to generate nitrogen fertilizer recommendations: at the beginning of the crop ping season. Individuals from the overall farmer population volunteered to test a fertilizer practice. Similar symbols represent a cohort, i.e., a group of farmers having fields with the same soil type. The group of volunteer farmers was broken down by cohort and researchers independently generated fertilizer recommendations for each cohort. Researchers did not control the number of volunteers from the respective cohorts In this example, only three of the four possible cohorts are found in the volunteer group.

4.2 Methods

4.2.1 Virtual crop management identification problem

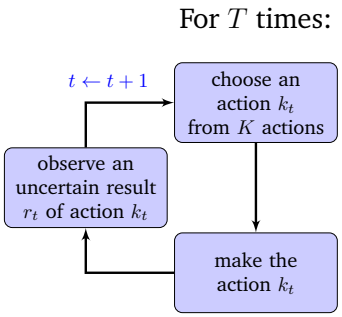
In our virtual crop management identification problem, a population or ensemble of farmers joined a participatory experiment to identify the best nitrogen fertilizer practices for maize production in their region, Koutiala in southern Mali. A total population of 500 farmers was considered. The distribution of soil types of the fields associated with the group of farmers was representative of the region (Table 4.1). A total population of 500 farmers was considered. Each farmer belonged to a cohort that corresponded to an ensemble of farmers growing maize on the same soil type. For each cohort, we wanted to identify the best nitrogen fertilizer practice from a set of recommended practices (see Table 4.3 and Section 4.2.1 for the performance metrics we considered). The research team set the additional objective to limit the crop yield losses of individual farmers that could arise from poor nitrogen fertilizer practice recommendations during the identification process.

At the beginning of each crop growing season, we assumed that a random number of farmers (uniformly obtained between 250 and 350) of the population of 500 farmers volunteered to apply the recommended fertilizer applications provided by the research team. Each year, the group of volunteers was variable in size and in the representation of cohorts, as could occur in reality (Figure 4.1). Thus, researchers did not control the composition of the group of volunteers. Each farmer indicated the fields and corresponding soils on which she/he planned to grow maize. Researchers then provided a fertilizer recommendation (Table 4.3) to each farmer for the ongoing season, depending on her/his soil i.e. cohort. At the end of the season, volunteer farmers shared their results in terms of maize grain yields with the research team, allowing to refine the recommendations for the next season. The whole process was repeated during 20 consecutive years following the same process (Figure 4.2a).

Nitrogen fertilizer practices. Ten nitrogen fertilizer practices were considered as recommendations in the virtual modeling experiment (see Table 4.2). Practices 0 to 7 explored the following set of split applications for a total amount of 135 kg N/ha applied:



(a) Diagram of the ensemble best fertilizer identification process. Each year, a group of volunteer farmers test fertilizer practices recommended by experts and contribute to identifying the best fertilizer practices for the region. At the end of each season, the farmers share their results with experts. The experts will use these results to improve their recommendations for the next growing season. The process repeats for a total number of T years.



(b) Canonical bandit problem. For T times, an agent sequentially makes a decision on an action k_t from the set $\{1, \cdots, K\}$ of possible actions. After making the action k_t, the agent observes an uncertain result r_t. This result is sampled from a fixed distribution, unknown to the agent, which corresponds to the effect of action k_t.

Figure 4.2 Schematic representation of the ensemble best fertilization identification process and the canonical bandit problem.

- Two split applications (practice 0): 15 days after planting (DAP) and 30 DAP.
- Three split applications (practice 4) :15 DAP, 30 DAP and 45 DAP.
- Split applications according to the rainfall amount (practices 2, 3 and 6, 7): 2nd and 3rd top-dressing applications only if the cumulated rainfall amount from the start of the season to 30 DAP exceeds the 30th percentile of historical rainfall i.e. 200 mm.
- Split applications according to plant nitrogen content (practices 1, 3 and 5, 7): 2nd and 3rd top-dressing applications only if the simulated nitrogen stress factor (`NSTRES` in DSSAT, see below) exceeds 0.2 (0 no stress, 1 maximal stress), hereby mimicking the use of a portable chlorophyll meter to monitor plant nitrogen content (e.g. Kalaji et al., 2017).

Practice 8 corresponded to the optimal fertilization for maize (70 kg N/ha) in the study area based on simulations (Huet et al., 2022) , i.e. the average of the N fertilizer rates that were observed to result in maximum positive return on fertilizer investment (Getnet et al., 2016). Finally, practice 9 (180 kg N/ha) corresponded to a nitrogen fertilizer practice that is likely excessive. For all these practices, the nitrogen fertilizer applied was assumed to be ammonium nitrate broadcasted on the soil surface.

Maize growth simulations. In order to get a proxy for real-world performances of the maize nitrogen fertilizer practices, we simulated maize growth responses to fertilization under the growing conditions of Koutiala in southern Mali using `gym-DSSAT` (Gautron and Padrón González, 2022). `gym-DSSAT` is a modification of the DSSAT crop simulator (Hoogenboom et al., 2019) to allow a user to read DSSAT

Table 4.1 : Main properties of the soil types of the fields of farmers growing maize in Koutiala, Mali (Adam et al., 2020). 'SLOC.' stands for soil organic matter (g C/ 100 g soil, mean value for the 0-30 cm topsoil); 'SLDR' stands for soil drainage rate (fraction/day); 'SLDP' stands for soil depth (cm); 'Prop' stands for the percentage of each soil type present in the study area.

Soil name	Texture	SLDR	SLOC	SLDP	Prop.
ITML840101	clay loam	0.60	0.20	110	7%
ITML840102	loam	0.60	0.45	100	9%
ITML840103	silty loam	0.60	0.27	160	21%
ITML840104	silty clay loam	0.25	0.70	105	4%
ITML840105	silty clay loam	0.40	0.35	120	24%
ITML840106	loam	0.60	0.30	110	27%
ITML840107	silty clay loam	0.25	0.60	105	8%

Table 4.2 Maize nitrogen fertilizer recommendations for maize in Koutiala, Southern Mali, that were considered in the virtual experiment. Whether or not rainfall and plant nitrogen stress were considered as factors for the fertilizer recommendation is indicated by Yes or No. 'NSTRES' stands for plant nitrogen stress and 'DAP' for days after planting.

index	max amount applied (kgN/ha)	max applications	rainfall threshold	NSTRES threshold	15 DAP N (kgN/ha)	30 DAP N (kgN/ha)	45 DAP N (kgN/ha)
0	135	2	No	No	15	120	0
1	135	2	No	Yes	15	120	0
2	135	2	Yes	No	15	120	0
3	135	2	Yes	Yes	15	120	0
4	135	3	No	No	15	60	60
5	135	3	No	Yes	15	60	60
6	135	3	Yes	No	15	60	60
7	135	3	Yes	Yes	15	60	60
8	70	2	No	No	23	0	47
9	180	3	No	No	60	60	60

internal states and take daily fertilization decisions during the simulations (e.g. based on DSSAT internal states). For each soil type in Table 4.1 that was parameterized in DSSAT using the data from Adam et al. (2020), each simulated maize grain yield value is a sample of the response distribution for the considered fertilizer practice. This response distribution is the result of weather variability, generated in our study by the stochastic weather generator WGEN (Richardson and Wright, 1984; Soltani and Hoogenboom, 2003), which was calibrated using the 47-year-long weather records from N'tarla, about 30 km from Koutiala (Ripoche et al., 2015). The 'sotubaka' maize cultivar (from the DSSAT default cultivar list) was used for all model simulations as a representative of maize variety in southern Mali. Water and nitrogen stresses were simulated, but yield reduction through pests and diseases were not considered, neither was weed competition. In the model simulations, a different weather time series was generated for each growing season and for each recommendation using WGEN, inducing independent simulated maize yield responses to nitrogen fertilization. Section C.1 of Supplementary Materials gives further details of the simulation settings.

We simulated 10^5 times the maize grain yield responses to a given fertilizer practice for the differet soil types, which corresponds to 10^5 hypothetical growing seasons. These samples were used i) to ensure that simulated maize yield responses were in realistic expected ranges, ii) to qualitatively evaluate the complexity of the decision problem, and iii) to determine best nitrogen fertilizer practices whilst analyzing the performance of the crop management identification strategies. The samples were not provided to the algorithms prior to their application (i.e. no prior knowledge of the problem).

Performance indicators of fertilizer practices

A criterion to evaluate both the economic and environmental performance of a fertilizer practice π is Agronomic Nitrogen use Efficiency (ANE), as defined in Vanlauwe et al. (2011):

$$\mathrm{ANE}^{\pi} := \frac{\mathrm{Y}^{\pi} - \mathrm{Y}^{0}}{\mathrm{N}^{\pi}} \tag{4.1}$$

where Y^{π} is the crop yield obtained with the nitrogen fertilizer practice π which required a quantity N^{π} of nitrogen and Y^{0} is the yield of the control obtained in the same conditions without nitrogen fertilization. Maximising ANE is a proxy of minimizing the quantity of nitrogen losses, e.g. through nitrate leaching.

However, ANE has some limitations: for example, an ANE value of 25 kg grain/kg N can be achieved with a fertilizer input of 20 kg N/ha yielding a total yield gain of 500 kg/ha, or with an input of 60 kg N/ha yielding a total gain of 1500 kg/ha. For the same ANE, a farmer is likely to prefer the fertilizer practice that provides the greatest crop yield gain, i.e. with 60 kg N/ha. Similarly, choosing fertilizer practices only based on the associated crop yield gains is not satisfying. A similar yield gain can be achieved with different nitrogen fertilizer input rates which result in fairly different ANE: the practice with the highest efficiency must be preferred as it required less nitrogen fertilizer to achieve the same yield gain.

We built the Yield Excess (YE) indicator that favors the nitrogen fertilizer practice with the highest yield gain for those practices sharing the same ANE, and favors the practice with the highest efficiency for those practices sharing the same yield gain. YE of a nitrogen fertilizer practice π with respect to the reference practice π_{ref} of constant efficiency $\mathrm{ANE}_{\mathrm{ref}}$ using the same quantity of nitrogen fertilizer

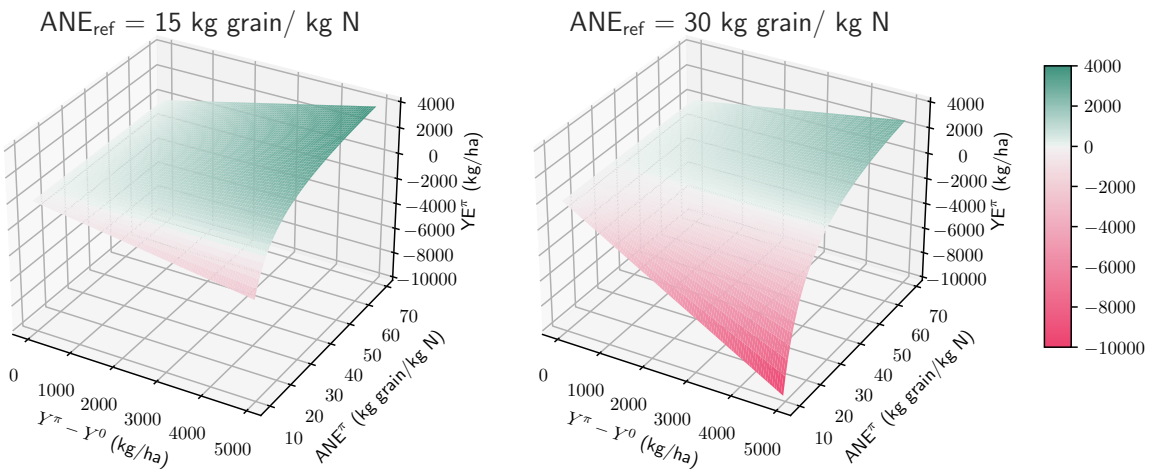


Figure 4.3 Yield Excess (YE^{π}, Equation 4.5) for $ANE_{ref} = 15$ kg grain /kg N and $ANE_{ref} = 30$ kg grain /kg N. Y^{π} is the maize grain yield obtained with nitrogen fertilizer practice π, Y^0 is the yield obtained with no nitrogen fertilization (control). ANE^{π} is the Agronomic Nitrogen use Efficiency of the nitrogen fertilizer practice π (Equation 4.1).

as practice π, denoted N^{π}, is computed as follows:

$$YE^{\pi} := Y^{\pi} - Y^{\pi_{ref}} \tag{4.2}$$

$$= \underbrace{Y^{\pi} - Y^{0}}_{\substack{\text{yield gain of } \pi \\ \text{w.r.t. control}}} - \underbrace{\left(Y^{\pi_{ref}} - Y^{0}\right)}_{\substack{\text{yield gain of } \pi_{ref} \\ \text{w.r.t. control}}} \tag{4.3}$$

$$= Y^{\pi} - Y^{0} - N^{\pi} \times ANE_{ref} \tag{4.4}$$

$$= \left(Y^{\pi} - Y^{0}\right) \times \underbrace{\left(1 - \frac{ANE_{ref}}{ANE^{\pi}}\right)}_{\text{penalization factor}} \tag{4.5}$$

The YE of practice π with respect to the reference practice π_{ref} corresponds to the yield difference between the practice π and a reference practice that has a constant ANE equal to ANE_{ref} and which uses the same quantity N^{π} of nitrogen fertilizer as π. YE^{π} increases with ANE^{π} (Figure 4.3). YE^{π} is negative and decreases with $Y^{\pi} - Y^{0}$ when $ANE^{\pi} < ANE_{ref}$ and is positive and increases with $Y^{\pi} - Y^{0}$ when $ANE^{\pi} \geq ANE_{ref}$. The YE of fertilizer practices with efficiency below ANE_{ref} are negatively affected by this metric. We chose $ANE_{ref} = 15$ kg grain/kg N for our model simulation experiments, the average ANE currently achieved by farmers across sub-Saharan Africa (Ten Berge et al., 2019; Vanlauwe et al., 2011).

Because farmers are usually risk averse (e.g. Cerf and Sebillotte, 1997; Jourdain et al., 2020; Menapace et al., 2013), they are likely to prefer, for example, a stable maize grain yield of 3000 kg/ha rather than a yield of 5000 kg/ha in half of the years, and of 1000 kg/ha in the other half of the years, while both distributions share the same expectation. To account for risk aversion, we computed the Conditional-Value-at-Risk (CVaR, Acerbi and Tasche, 2002; Mandelbrot, 1997), a risk-aware metric that originated from finance. The CVaR focuses on the lower tail of the distribution[¶¶]. For a

[¶¶]Two definitions of the CVaR coexist in the literature, depending if an outcome is considered as a gain or a cost (Dowd, 2007). We adopted the gain point of view.

(a) High risk aversion ($\alpha \approx 20\%$) (b) Low risk aversion ($\alpha \approx 80\%$)

Figure 4.4 The Conditional Value-at-Risk (CVaR) of level α is the mean value of the blue area of the distribution of probability $0 < \alpha \leq 1$. VaR_α stands for Value-at-Risk of level α and is the quantile of probability α of the distribution. The more $\alpha \to 1$, the more risk neutral is the CVaR. μ represents the mean value of the distribution which equivalent to the CVaR of level $\alpha = 100\%$.

(continuous) random variable X with cumulative distribution function F_X, we call Value-at-Risk (VaR) of level α the quantile of probability $\alpha \in (0, 1]$ of X, defined as:

$$\text{VaR}_\alpha(X) := \inf \{x \in \mathbb{R} : F_X(x) > \alpha\} \tag{4.6}$$

Then the CVaR of X of level $\alpha \in (0, 1]$ is the mean value of the left tail of X of probability α, defined as:

$$\text{CVaR}_\alpha(X) := \mathbb{E}[X | X \leq \text{VaR}_\alpha(X)] \tag{4.7}$$

A decision maker would choose the option with the highest CVaR for the considered level α. The more $\alpha \to 0^+$, the more the metric focuses on the worst observable yields. On the contrary, the more $\alpha \to 1$, the less risk averse is the measure. When $\alpha = 1$, the CVaR equals the usual expectation $\mathbb{E}[X]$, which is risk neutral (Figure 4.4). In our model simulation experiments, we chose $\alpha = 30\%$. The $\text{CVaR}_{30\%}$ represents the mean crop yield of the 30% worst observable years.

4.2.2 Identification of the best fertilizer practices

The canonical and batch bandit problems The ensemble identification of the best crop management practices with the constraint of minimizing farmers' crop yield losses occurring during the identification process (Section 4.2.1) can be modeled as a special type of bandit problems. The canonical bandit problem, which is the cumulated regret minimization (see Introduction), assumes that at each time step, a single trial is made and is followed by a single observation of a result, in a purely sequential mode. In contrast, the batch bandit setting (Perchet et al., 2015) assumes that at each time step an ensemble of trials are conducted in parallel, followed by the observation of an ensemble of results. Figure 4.2 illustrates on the one hand the ensemble identification process of best crop fertilizer practices (Figure 4.2a), modeled as a batch-bandit problem, and the on other hand the canonical bandit problem (Figure 4.2b).

In the canonical bandit problem, the agent goal is to maximize the expectation of the sum of rewards that were collected since the first decision. The agent objective can be formalized as maximizing $\mathbb{E}\left[\sum_{t=1}^{T} r_t\right]$ for any time horizon $T \geq 1$, with r_t the reward the agent has collected at time t. On the other hand, bandits that are *risk-aware* (Cassel et al., 2018), the agent maximizes a risk-aware measure of the collected rewards, such as the CVaR (Section 4.2.1), instead of the expectation of rewards. Our ensemble fertilizer decision problem can be described as a *risk-aware batch-bandit* decision problem.

The ensemble identification problem of best fertilizer practices In our virtual modeling experiment, for $t \in \{1, 2, \cdots, T\}$, at each season t, researchers assigned each n_t volunteer farmers for season t with a nitrogen fertilizer practice $\pi \in \{1, 2, \cdots, K\}$. Each farmer belonged to a cohort

$c \in \{1, 2, \cdots, C\}$. At the end of season t, researchers assemble rewards $Y_t = \{y_t^1, \ldots, y_t^{n_t}\}$ as a result of the fertilizer practices of all farmers for season t. For each cohort $c \in \{1, \cdots, C\}$, rewards are independently and identically distributed from unknown stationary distributions $\{\nu_1^c, \cdots, \nu_K^c\}$. These reward distributions are the YE with $\text{ANE}_{\text{ref}} = 15$ kg grain/kg N associated to each of the ten recommended nitrogen fertilizer practices, for a given soil type. We denote $\mathcal{Y}_T = \bigcup_{t=1}^{T} Y_t$ the set of all rewards observed by all farmers between $t = 1$ and $t = T$. The objective of an identification strategy is to maximize, for a given CVaR level α and any time horizon $T \geq 1$:

$$\mathbb{E}[\text{CVaR}_\alpha(\mathcal{Y}_T)] \tag{4.8}$$

For each cohort $c \in \{1, \cdots, C\}$, an optimal nitrogen fertilizer practice π_*^c is given by:

$$\pi_*^c = \underset{k}{\text{argmax}}\ \text{CVaR}_\alpha(\nu_k^c) \tag{4.9}$$

Consequently, an optimal identification strategy always assigns nitrogen fertilizer practice π_*^c to all farmers belonging to cohort c.

Identification strategies

We expected fertilizer practices to perform differently within each cohort, i.e. each soil. For example, the optimal nitrogen fertilizer practices were expected to be different between a cohort growing maize on a shallow sandy soil and a cohort growing maize on a deep clayey soil. Consequently, the results of one cohort were not supposed to be directly relevant for another cohort. Each soil was considered as an independent identification problem, i.e. had its own independent identification strategy which did not share information with the identification strategies of other soils.

For a given soil, from one season to another, the identification strategy kept memory of all results observed during past seasons, for the same soil. In model simulation experiments, we considered two types of identification strategies: either the standard `ETC` (Explore-Then-Commit) strategy, previously referred as the "intuitive strategy", or `BCB`, the bandit-algorithm based identification strategy. For the seven soils in Table 4.1, the identification strategy types were either all `ETC`, or all `BCB`, but not a mix of both.

Intuitive identification strategy `ETC` provides a simple and intuitive solution to the exploration-exploitation dilemma. During an initial exploration phase of an arbitrary number of years, `ETC` equiproportionally test all nitrogen fertilizer strategies. Thereafter, the exploitation phase starts and `ETC` chooses for the remaining time the fertilizer strategy that has shown best performance during the exploration phase. In Section C.3.2 of Supplementary Materials, we provide a simple adaptation of `ETC` to the batch setting (see Section 4.2.1) using the CVaR of rewards rather than the classical expectation. We considered `ETC-3` and `ETC-5`, with respectively 3 and 5 years of exploration phases. During the exploration phase, fertilizer practices are randomly assigned in equal proportions to the farmers within the cohort.

Bandit based identification strategy `BCB` is a risk-aware bandit algorithm (Cassel et al., 2018) which uses the CVaR of rewards as decision criterion, in the batch bandit setting. `BCB` derives from the the work of Baudry et al. (2021a). We provide the pseudo-code of `BCB` and detail how it works in

Supplementary Materials Section C.3.1. The general idea of the bandit algorithm is, for each season, to leverage the information acquired during all past seasons, such that the algorithm adapts to optimally manage the exploration-exploitation dilemma.

We provide a quick overview of the execution of BCB with algorithm 2. Considering the YE with $\text{ANE}_{\text{ref}} = 15$ kg grain/kg N as results, we set its maximum observable result to 4000 kg/ha for all fertilizer practices as required for the execution of BCB (see first execution step of algorithm 2), based on Figure 4.3. As an additional feature, BCB provides a fair distribution of risky option trials amongst farmers at the cohort level. The bandit algorithm ranks each fertilizer practice according to its observed performance in the previous year. The algorithm then recommends first the practices that appear to yield best results to the farmers that have experienced worst results so far.

Algorithm 2 Simplified pseudo-code of BCB.

```
for fertilizer practice k ∈ {1, ··· , K} do
    Add maximum observable value to the results of fertilizer practice k   // prior to any experiments
end
for season t ∈ {1, ··· , T} do
    for farmer f ∈ {1, ··· , n} do
        for fertilizer practice k ∈ {1, ··· , K} do
            Re-weight the rewards of the fertilizer practice k with random weights sampled from a Dirichlet distribution (Everitt and Skrondal, 2002)
            Score practice k with a noisy empirical measure of the CVaR at level α of practice k from the re-weighted rewards
        end
        Recommend to the farmer f the fertilizer practice with the maximum score
    end
    Collect and store all results of the season for all fertilizer practices
end
```

Direct measure of performance of an identification strategy

We denote $\widehat{C}_\alpha$ the expression of the empirical CVaR of level $\alpha \in (0, 1]$. The empirical CVaR is an estimate of the true CVaR as defined in Equation 4.7 –just as an average value is an estimate of the true mean of a distribution–. Assuming a sample $\mathcal{Y}$ of rewards sorted in an increasing order i.e. $\mathcal{Y} = \{y_1, \cdots, y_n\}$ such that $y_i \leq y_{i+1}$, and defining $q = y_{\lceil \alpha n \rceil}$ the empirical quantile of level α, we have:

$$\widehat{C}_\alpha(\mathcal{Y}) := q - \frac{1}{n\alpha} \sum_{i=1}^{n} \max(q - y_i, 0) \tag{4.10}$$

In a simulated problem, the quantity in Equation 4.8 can be estimated by repeatedly applying R times an identification strategy during T years, and then concatenating all results of all farmers from time $t = 1$ to time $t = T$ for all replications, and finally computing the empirical CVaR of the resulting set. In order to approximate all expectations, for all experiments, in practice we consider $R = 960$ (12 executions in parallel on an 80 core machine; for each one of the 960 experiments, the weather generator had a different random state). We denote $r \in \{1, \cdots, R\}$ the repetition index. We define

$\mathfrak{Y}_T = \bigcup_{r=1}^{R} \mathcal{Y}_T^r$ i.e. the results of all farmers until year T for all replications. Then:

$$\mathbb{E}\left[\text{CVaR}_\alpha(\mathcal{Y}_T)\right] \widehat{=} \widehat{C}_\alpha(\mathfrak{Y}_T) \tag{4.11}$$

The resulting quantity is an average measure of the results of the group. The more an identification strategy maximizes this quantity, the better it is. In a real-world problem, only one realization of $\text{CVaR}_\alpha(\mathcal{Y}_T)$ is computable.

Proxy measure of performance of identification strategy

While the quantity in Equation 4.8 can be estimated with Equation 4.11, it is intricate to analyze and derive statistical guarantees for this estimator. This is why, in the following, we introduce a proxy of this quantity called the cumulated CVaR regret, which is a central element behind the theoretical performance guarantees of bandit algorithms. The cumulated regret is also a convenient statistic to represent the performance of an algorithm, with little noise.

Mean cumulated regret of the farmer population Considering a single cohort c, we suppose that we sequentially repeat T times the choice of one option k from an ensemble of K possible options. Here k is the index of the fertilizer practice. We denote $\text{CVaR}_\alpha(\nu_k^c)$ the CVaR of level α associated with the option k and cohort c, and $\text{CVaR}_\alpha(\nu_*^c) = \max_{k\in\{1,\cdots,K\}} \text{CVaR}_\alpha(\nu_k^c)$ the highest CVaR at level α of all options for cohort c i.e. the CVaR of the best option for cohort c. In expectation, for a farmer belonging to cohort c and following T years the recommendations of a given identification strategy selecting a fertilizer practice $k(t)$ each year $t \in \{1, \cdots, T\}$, we define the cumulated regret for the CVaR as in Tamkin et al. (2020):

$$\underbrace{R_\alpha^c(T)}_{\substack{\text{loss of the}\\ \text{strategy}}} := \underbrace{T \times \text{CVaR}_\alpha(\nu_*^c)}_{\substack{\text{score of the best}\\ \text{possible strategy}}} - \underbrace{\mathbb{E}\left[\sum_{t=1}^{T} \text{CVaR}_\alpha(\nu_{k(t)}^c)\right]}_{\substack{\text{score of the actual}\\ \text{strategy}}} \tag{4.12}$$

$$= \sum_{k=1}^{K} \underbrace{\left(\text{CVaR}_\alpha(\nu_*^c) - \text{CVaR}_\alpha(\nu_k^c)\right)}_{\substack{\text{loss between the best option}\\ \text{and the option } k \text{ for cohort } c}} \times \underbrace{\mathbb{E}\left[N_k^c(T)\right]}_{\substack{\text{expected number of times}\\ \text{option } k \text{ is chosen for cohort } c\\ \text{during the } T \text{ years}}} \tag{4.13}$$

For cohort c, the cumulated regret $R_\alpha^c(T)$ can be seen as a loss occurred with the considered strategy with respect to the best possible strategy –the one that always chooses the fertilizer practice with the best CVaR–. Equivalently, it can be interpreted as a measure of the expected total error due to sub-optimal actions made during a series of T decisions: the more the best option is chosen within the T decisions, the smaller the cumulated regret is. The mean cumulated regret of the total farmer population is given by the cumulated regret of each cohort, weighted by the probability of an individual to belong to this cohort:

$$R_\alpha(T) = \sum_{c=1}^{C} R_\alpha^c(T) \times \Pr(c), \text{ with } \sum_{c=1}^{C} \Pr(c) = 1 \tag{4.14}$$

When extensively testing an identification strategy on a simulated problem, the CVaR of the different options can be approximated with a large enough number of samples or analytically computed, irrespective of the identification strategy. For each cohort, this corresponds to the left-hand side of Equation 4.13, and is thus supposed to be known. Note that, for a real-world problem, these quantities are unknown –else the decision problem would have been solved–. On the right hand side of Equation 4.13, the quantity $\mathbb{E}\left[N_k^c(T)\right]$ can be empirically approximated by repeatedly performing experiments with the identification strategy, and averaging the number of times each fertilizer practice has been chosen since time step T for each cohort. Finally, in Equation 4.14, the proportion of each soil, i.e. cohort, can be found in Table 4.1. Minimizing the cumulated regret maximizes the quantity in Equation 4.8, as shown by Cassel et al. (2018). For a given identification strategy, the smaller and less variable the mean cumulated regret of population (Equation 4.14), the more farmers are guaranteed to maximize their CVaR of YE.

Distribution of the cumulated regret of individual farmers The mean cumulated regret of the population given in Equation 4.14 does not indicate the distribution of individual farmer regrets. For each farmer f belonging to cohort c, the individual regret after T years for the CVaR of level $\alpha \in (0, 1]$ is computed as:

$$\tilde{R}_\alpha^{f,c}(T) := \sum_{k=1}^{K} \underbrace{\left(\text{CVaR}_\alpha(\nu_*^c) - \text{CVaR}_\alpha(\nu_{k(t)}^c)\right)}_{\substack{\text{loss between the best option} \\ \text{and the option } k}} \times \underbrace{N_k^{f,c}(T)}_{\substack{\text{number of times option k} \\ \text{is chosen during T years} \\ \text{for farmer } f}} \tag{4.15}$$

For each cohort c, the distribution of $\tilde{R}_\alpha^{f,c}(T)$ indicates how the potential losses due to bad recommendations are distributed amongst farmers.

4.3 Results

4.3.1 Simulated responses to nitrogen fertilizer practices

Table 4.3 provides the statistics of the optimal nitrogen fertilizer practices for each soil type (Table 4.1), i.e. for each cohort, and Figure C.1 in Supplementary Materials shows the distribution of grain yield, ANE and YE responses. All responses showed values within the expected ranges for the considered growing conditions, with an average grain yield varying from 3125 kg/ha for a sandy soil with low fertility (`ITML84105`) up to 3945 kg/ha for a loamy soil (`ITML84106`). When a applying the most promising fertilization strategies, on average the YE (i.e. yield gain compared to the reference) for farmers ranged from 1200 kg/ha to 1800 kg/ha, and the $\text{CVaR}_{30\%}(\text{YE})$ (i.e. the mean crop YE of the 30% worst observable years) from 500 kg/ha to 1032 kg/ha.

There was no simple parametric assumption that could be made about YE, such as its probability distribution to be Gaussian (e.g. practice 5 in Figure C.1e). The thicker left tails for e.g. fertilizer practices 4 and 0 or the bi-modality of YE for practices 6 and 7 (Figure C.1e), further supported the use of the CVaR as a relevant risk measure. Indeed, the CVaR is most relevant for asymmetric and irregularly shaped distributions, such as thick-tailed or multi-modal distributions. For all soils, the optimal nitrogen fertilizer practices were either nitrogen fertilizer practice 0 or 8 i.e. nitrogen

Table 4.3 Statistics of the optimal nitrogen fertilizer practices for each of the soil types presented in Table 4.1. For the corresponding optimal nitrogen fertilizer practice π^*, we define N^{π^*}: quantity of nitrogen fertilizer applied; $CVaR_{30\%}(X)$: conditional Value-at-Risk of X of level 30% (Section 4.2.1); $\bar{X}$: mean value of X; Y^{π^*}: maize grain yield; ANE^{π^*}: Agronomic Nitrogen use Efficiency; YE^{π^*}: Yield Excess (Section 4.2.1); parentheses indicate standard deviations.

soil	π^*	$\bar{N}^{\pi^*}$ (kg/ha)	$CVaR_{30\%}(Y^{\pi^*})$ (kg/ha)	$\bar{Y}^{\pi^*}$ (kg/ha)	$\bar{ANE}^{\pi^*}$ (kg/kg)	$CVaR_{30\%}(YE^{\pi^*})$ (kg/ha)	$\bar{YE}^{\pi^*}$ (kg/ha)
ITML840101	**0**	120.0 (1.0)	3091	3874 (666)	30.0 (5.4)	**1032**	1795 (651)
ITML840102	**8**	69.8 (4.0)	2391	3150 (653)	33.2 (7.5)	**652**	1270 (529)
ITML840103	**8**	70.0 (0.4)	2539	3152 (526)	34.4 (6.8)	**808**	1356 (475)
ITML840104	**8**	69.9 (2.7)	2533	3339 (682)	31.7 (8.1)	**500**	1169 (565)
ITML840105	**8**	70.0 (1.2)	2467	3127 (570)	34.2 (7.3)	**757**	1346 (508)
ITML840106	**0**	120.0 (1.2)	3132	3945 (695)	28.9 (5.5)	**900**	1667 (660)
ITML840107	**8**	69.9 (2.7)	2472	3247 (659)	32.5 (8.0)	**565**	1226 (559)

practices without threshold dependent top-dressing, and with a single nitrogen top-dressing application (Table 4.3).

The nitrogen fertilizer practices had different responses for the different soil types in terms of the grain yield and ANE (and consequently YE), and ranking of the practices was inconsistent across the soil types (Figure C.1). For instance, for the soil ITML840104 (silt clay loam of medium fertility), fertilizer practices 0 to 4 had similar YE (Figure C.1e). For the soil ITML840105 (silt clay loam of low fertility), practices 0, 1 and 4 were substantially better than practices 2 and 3 (Figure C.1f).

Threshold-based fertilizer practices behaved inconsistently across the soil types. As an example, for the bi-modal YE distribution of the fertilizer practice 1, most of the probability density was concentrated around 0 kg/ha for the soil ITML840104 (Figure C.1e) and around 1800 kg/ha for the soil ITML840105 (Figure C.1f). For the soil ITML840104 and practice 1, YE were mostly found around 0 kg/ha because most of the seasons, the nitrogen-stress threshold of 0.2 was not reached, and consequently no top-dressing occurred (Table 4.2). In such cases, only a basal-dressing of 15 kg N/ha was applied, instead of a total of 135 kg N/ha when the top-dressing was triggered. Consistently, for the same soil and fertilizer practice, the probability density of grain yield was concentrated around the low value of 1000 kg/ha (Figure C.1a). On the other hand, with the soil ITML840105, most of the seasons, the nitrogen-stress threshold of 0.2 was reached and practice 1 applied both basal and top-dressing. This corresponded to YE mostly found around 1800 kg/ha (Figure C.1f), and the corresponding grain yields were mostly found around 4000 kg/ha (Figure C.1b).

4.3.2 Identification of best fertilizer practices

In Section 4.3.2 and 4.3.2, we present respectively a direct measure of empirical performances of the nitrogen fertilizer practice identification strategies (see Section 4.2.2), and the regret as a proxy measure, both for the farmer population average and the individual farmer regret distribution (see Section 4.2.2). Section 4.3.2 provides a visual comparison of nitrogen fertilizer recommendations following respectively the BCB and ETC-5 identification strategies.

Sampling visualization

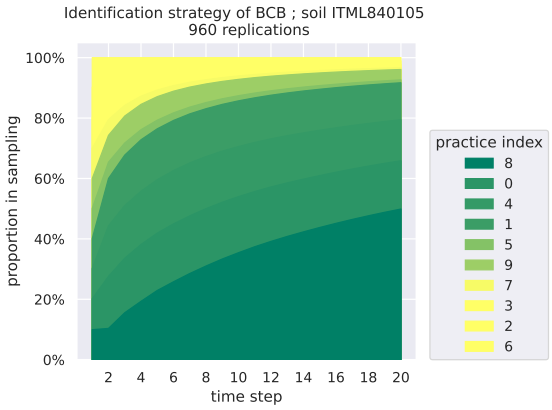
Figures 4.5 provides the average frequency with which the fertilizer practices were selected by the identification strategies, from the beginning of the experiment to time T, for soils ITML840105 and ITML840101. For the soil ITML840105, respectively for the BCB and ETC-5 strategies. After 20 years, BCB had selected the fertilizer practice 8, which was the optimal one (see Table 4.3), with an average proportion of 50%. The proportions of the optimal practice continuously increased from year 2 onwards (Figure 4.5a). During the first 5 years, ETC-5 uniformly sampled all fertilizer practices (Figure 4.5b), thus inducing potentially high losses for farmers. The proportion of the optimal practice started to increase from year 5 onwards. After year 20, ETC-5 sampled the optimal practice with an average proportion of 31%. For soil ITML840101, results are more contrasted. After year 20, both BCB has sampled the optimal strategy, which was fertilizer practice 0 (see Table 4.3) with an average proportion of 27% (Figure 4.5c) and ETC-5 (Figure 4.5d) with an average proportion of 26%. Note that in Figures 4.5c and 4.5d, the color differences are almost not perceptible for nitrogen fertilizer practices 0, 1 and 4, because all three practices showed similar performances. In Sections 4.3.2 and 4.3.2, we provide the results of statistics that account for all cohorts, i.e. soils.

Direct measure of performances of identification strategies

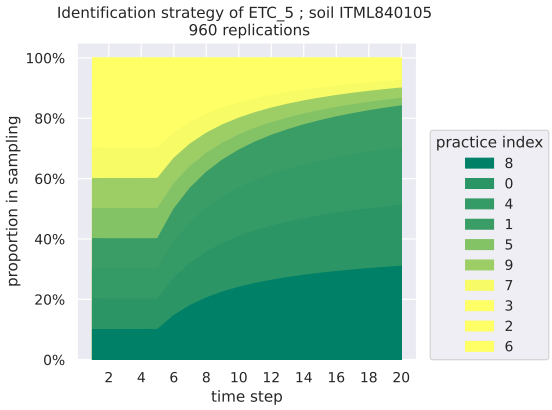
Figure 4.7 represents the evolution of the $\text{CVaR}_{30\%}(\text{YE})$ for all cohorts trough the years (Equation 4.11). On average, farmers following the nitrogen fertilizer recommendations based on the BCB strategy had higher empirical CVaR at 30% of YE than farmers following those from ETC strategies, from the second year of the experiment onwards (Figure 4.6). The difference in performance between BCB and ETC is high during the initial years. For instance, at year 4, farmers following recommendations from the BCB identification strategy had a CVaR at 30% of YE of 318 kg/ha, compared to 168 kg/ha (47% less than BCB) and 74 kg/ha (77% less than BCB) for farmers following the recommendations respectively from the ETC-3 and the ETC-5 identification strategies. BCB allowed to identify faster the optimal fertilizer practices and consequently further avoided low crop yield outcomes compared to ETC strategies. ETC strategies were adversely affected by their exploration phases during which all fertilizer practices were equiproportionally tested. In contrast, BCB had a continuously increasing empirical CVaR, for the whole duration of the experiment.

Regret

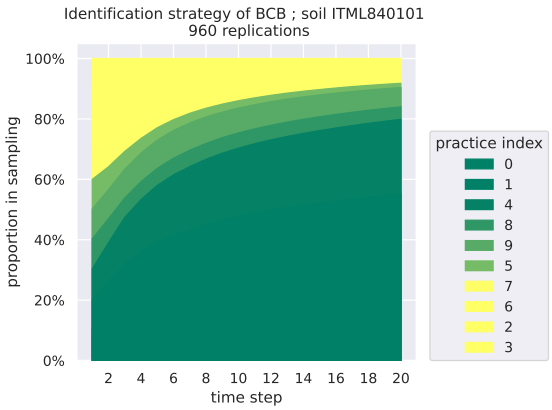
Mean cumulated regret of the farmer population Figure 4.7 represents the evolution of the mean regret for all cohorts trough the years (Equation 4.14). For $\alpha = 30\%$, BCB identification strategy outperformed ETC strategies, regardless of the number of years during which the strategy was applied. The difference in performance between BCB and ETC increases for the whole duration of the experiments. After 20 years, farmers following recommendations from BCB identification strategy experiences a mean cumulated regret of 2400 kg/ha, compared to 3385 kg/ha (41% more than BCB) and 3701 kg/ha (54% more than BCB) for farmers following the recommendations respectively from the ETC-3 and ETC-5 strategies. Consequently, farmers following BCB recommendations accumulated less regret compared to farmers following ETC recommendations. Furthermore, the variance of the cumulated regret (due to all different weather series in the experiments, for each season and each field trial, and the variability in cohorts each year) was smaller for BCB than for ETC, confirming that BCB strategy was more robust (see quantile ranges in Figure 4.7) for this decision problem.



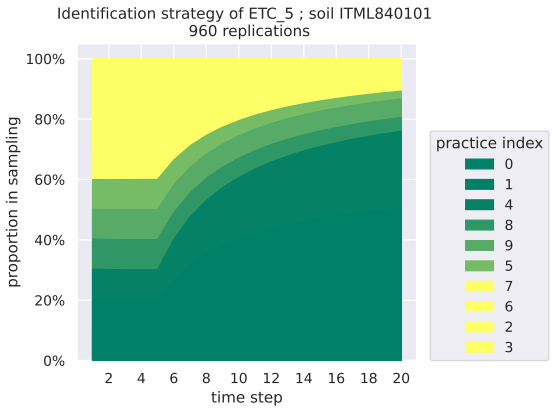
(a) `BCB` sampling proportions for soil `ITML840105`.



(b) `ETC-5` sampling proportions for soil `ITML840105`.



(c) `BCB` sampling proportions for soil `ITML840101`.



(d) `ETC-5` sampling proportions for soil `ITML840101`.

Figure 4.5 Averaged sampling proportions for soils `ITML840105` and `ITML840101`, $T = 20$ years. 960 replications of the whole experiment were done. The fertilizer practices are ordered according to the true Conditional Value-at-Risk at level 30% (CVaR) of their Yield Excess (YE) with ANE_{ref}=15 kg grain/kg N ; the greener the color, the better a fertilizer practice is.

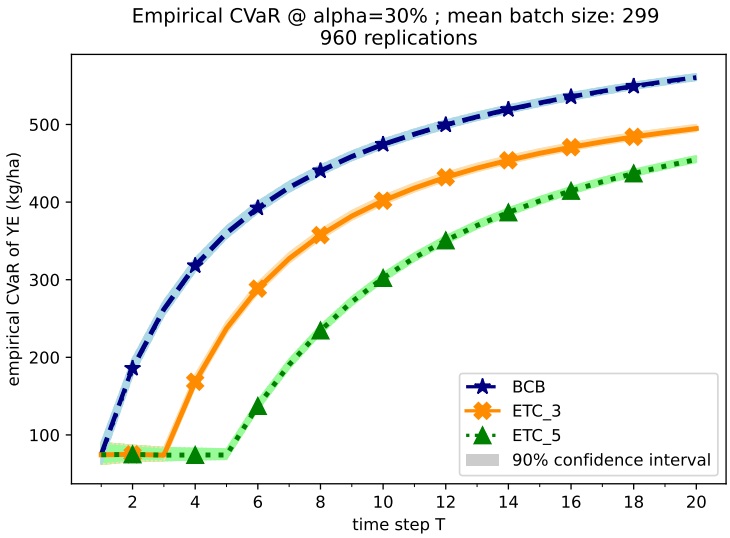


Figure 4.6 Empirical conditional Value-at-Risk (CVaR) at level 30% (CVaR) of maize yield excesses (YE) between $T = 0$ and the considered T ; $\text{ANE}_{\text{ref}} = 15$ kg grain/kg N. 960 replications of the whole experiment were done. One time step T is one year ; 'mean batch size' is the number of farmers who have volunteered to participate at the trials, averaged over all years and all replications. Confidence intervals were computed following Thomas and Learned-Miller (2019).

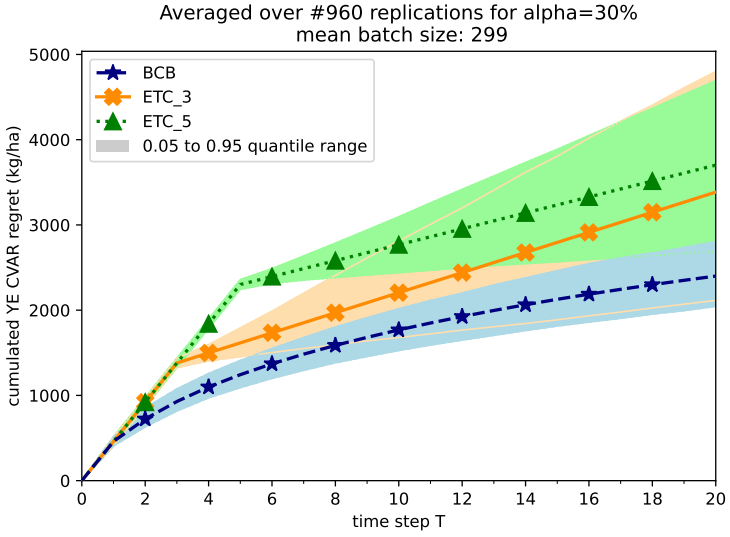


Figure 4.7 Mean cumulated regret of population, for the Conditional Value-at-Risk (CVaR) at level 30% of Yield Excess (YE); $\text{ANE}_{\text{ref}} = 15$ kg grain/kg N. The cumulated cumulated regret is averaged over the farmers' population, between $T = 0$ and the considered T. 960 replications of the whole experiment were done. One time step T is one year, 'mean batch size' is the number of farmers who have volunteered to participate in the trials, averaged over all years and all replicates.

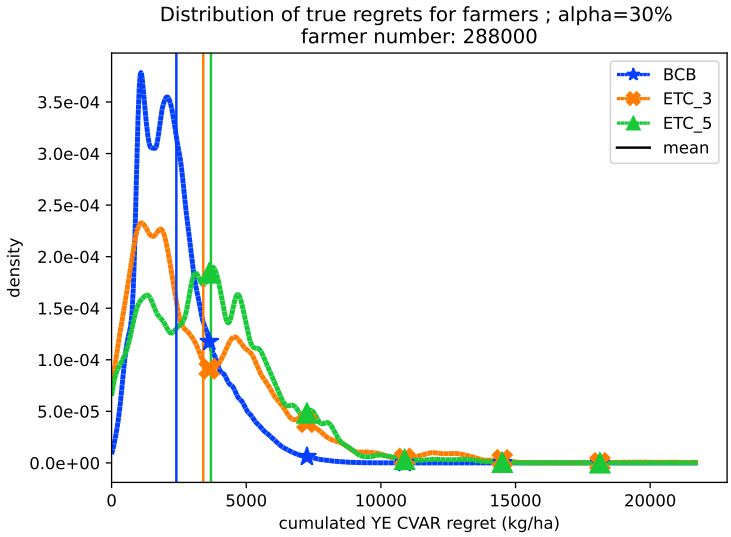


Figure 4.8 Distribution of individual cumulated regret after $T = 20$ years for Conditional Value-at-Risk at level 30% (CVaR) of the yield excess (YE) ; $\text{ANE}_{\text{ref}} = 15$ kg grain/kg N. The total number of farmers corresponds to a group of 300 farmers, with 960 replications of the whole experiment.

Individual cumulated regret distribution BCB prevented farmers from accumulating large individual cumulated regret during the participatory identification of the group (Figure 4.8): individual cumulated regrets for BCB were distributed towards lower values than for ETC strategies. With BCB, almost no individual cumulated regret was greater than 7.5 t/ha after 20 years, as opposed to ETC strategies. Consequently, BCB allowed a fairer sharing of identification mistakes in the population of farmers than ETC strategies.

Sensitivity analysis

In Section C.4 of Supplementary Materials, we present the same results than Sections 4.3.2 and 4.3.2 for higher CVaR levels of $\alpha = 50\%$ and $\alpha = 100\%$. The CVaR with the latter level recovers the usual expectation. For $\alpha = 50\%$, BCB showed similar performance than for $\alpha = 30\%$. For $\alpha = 100\%$, ETC-3 was the best performer, BCB and ETC-5 performed similarly. Nonetheless, BCB showed a smaller variance than both ETC-3 and ETC-5. The theoretical performance guarantee is presented in Section C.5 of Supplementary Materials.

4.4 Discussion

4.4.1 Benefits from an adaptive identification strategy.

Practical perspective In multi-year multi-location on-farm trials, participating farmers simultaneously conduct field experiments with crops over multiple seasons to compare e.g. crop management practices (e.g. Baudron et al., 2012; Falconnier et al., 2016; Naudin et al., 2010). After a given number of years, results (often in terms of crop yields) are typically analyzed using mixed linear models (Laird and Ware, 1982), to take into account the design of an experiment with repeated measures, such as

random effects associated with fields and farms. Best crop management practices are then identified based on this statistical analysis. In our simulated nitrogen fertilizer practice decision problem, we approximated multi-year on-farm trials with the ETC intuitive identification strategy. Both replicated on-farm trials and ETC consist of an exploration phase of a fixed duration (data collection), followed by an exploitation phase (application of the best identified practice after analysis of collected data). Consequently, both replicated on-farm trials and ETC can be considered as non-adaptive identification strategies: before the end of the exploration phase, the intermediary results are not exploited to gradually refine the experimental setup. In contrast, BCB refines its the recommendations every year, based on the results observed so far. The better a crop management option, the more its proportion in recommendations should increase with time. From a farmer's perspective, this mean that the probability of highly sub-optimal recommendation decreases with time, as opposed to non-adaptive identification strategies during the exploration phase, which equi-proportionally recommend all crop management practices. Because with the bandit-algorithm-based identification strategy yield losses are reduced in most cases, compared to the non-adaptive identification strategies, the cost of the identification of best management practices is likely to be reduced for the farmers. Another common method to generate crop management recommendation consists in the use of calibrated crop models and scenario analyses (e.g. Huet et al., 2022). This method can be complementary to the bandit-based approach. Candidate crop management practices can first be determined based on crop modeling results for the considered conditions, and then best crop management can be identified with the bandit algorithm.

Theoretical perspective ETC is theoretically proven to be a sub-optimal identification strategy without a calibration of the duration of the exploration phase that requires unavailable strong prior knowledge on the complexity of the decision problem (Lattimore and Szepesvári, 2020, Chapter 6). In numerical experiments, for $\alpha = 100\%$, ETC-3 best performed, probably because with these particular YE distributions and size of farmer group, 3 years of exploration was an optimal number. A slight change in the decision problem may induce that 3 or 5 years of exploration phase are no longer optimal (e.g. changing α to 30% or 50%). More generally, prior to an experiment, there is no guarantee than an arbitrate number of years of exploration of ETC will be optimal, and consequently there are no guarantees about the performance ETC, as opposed to BCB (see theoretical results in Section C.5). The main benefit of BCB over ETC is that it does not require a choice on parameters that require prior knowledge that is *a priori* not available. BCB neither requires strong assumptions about probability laws of reward distributions, as opposed to other common bandit algorithms. The only requisite for BCB is the knowledge of the maximum observable reward. In agronomy, such knowledge is easily available with expert knowledge: for instance, considering yield as reward, an expert can easily estimate a yield potential in the given crop growing conditions, for instance through modeling (Affholder et al., 2013).

4.4.2 Performances of fertilizer practices

For all soils, no optimal nitrogen fertilizer practice was threshold-based, nor shown split top-dressing. This does not discard a potential benefit from the threshold-based fertilizer practices, or split top dressing. Model simulations revealed that the effect of the nitrogen-stress or rainfall threshold depended on each soil, and the effect of the thresholds was not easy to anticipate. Consequently, the definition of the set of candidate fertilizer practices to explore must be carefully selected within the

vast possible combination of practice parameters, e.g. application timing, rates, thresholds or number of split. In the experiments, the optimal values of practice parameters were not adjusted, because our focus was on designing a better generic identification method, rather than on designing refined parameterized fertilizer practices. For an application in real field conditions, we recommend these parameters to be adjusted based on simulations (see Section 4.4.1) and/or on prior small test plots. More generally, the design of fertiliser practices must include experts, local extensionists and farmers themselves (Cerf and Meynard, 2006; Hochman and Carberry, 2011). For instance, the maximum quantity of nitrogen fertilizer a farmer can apply may depend on the availability of fertilizer in the local market, and on the economic situation of each farmer.

4.4.3 Definition of farmers' objective

We defined the farmers' objective as maximizing the CVaR at level $\alpha = 30$ of the YE with $\text{ANE}_{\text{ref}} = 15$ kg grain/kg N. This quantity is interpretable as it represents a yield gain compared to a reference fertilizer practice, and it is easily calculable. The value of α allows to adjust the risk aversion level for a cohort of farmers. The value of ANE_{ref} defines an invariant economical and environmental trade-off which penalizes more or less the use of nitrogen fertilizer. Losses were defined as the expected performance difference between the best available nitrogen fertilizer practice, and the sub-optimal nitrogen fertilizer practices, in the face of the seasonal uncertainty.

However, we did not directly evaluate fertilizer practices by their economic return. Despite market risks being a reality, the economic return of maize nitrogen fertilization depends on many parameters changing through time, such as fertilizer subsidies, fertilizer market price, application costs, or harvest selling price. Because each year, the optimal nitrogen fertilizer practice is likely to change, such setting dramatically increases the complexity of the identification problem, and so does the required amount of data to identify best practices (we provide more details in Supplementary Materials, Section C.2). In any case, modelers should bear in mind the inherent limitations of the modeling of a farmer's objective, which always remains a proxy (McCown, 2002a).

4.4.4 Limits and possible improvements

In our simulated crop management decision problem, we largely simplified the experimental structure of multi-year replicated field trials. First, for all simulations, weather time series were independent and identically distributed. Such assumption is unlikely to be true in the real world. During the same year, weather spatial correlations can be high, for instance in case of extreme weather events (Tack and Holt, 2016). Second, within the same cohort, all farmers were supposed to have exactly the same soil and cultivar, and to implement closely the fertilizer practice they were assigned. For real application, a farmer's soil, site, year and other potential random effects should be properly considered. The bandit identification strategy we introduced should be extended to account for experimental structure and multiple factors at stake. For instance, contextual bandits (Lattimore and Szepesvári, 2020), which would allow to share information between decision contexts (here, the cohorts), might offer solutions.

As another limit, in simulations, we considered climate to be the same during the 20 years of the experiment. Such hypothesis is improbable in real conditions (e.g. Traore et al., 2017). Nevertheless, as Adam et al. (2020) has shown based on simulations, in Mali, improving current crop management, in particular nitrogen fertilization, may compensate the long-term effects of climate change, while

addressing the urgent necessity of closing yield gaps. For a decision problem perspective, if climate changes through time, then optimal practices are likely to change with time. Such problem can be formalized as a non-stationary bandit problem (Lattimore and Szepesvári, 2020). To handle non-stationary, BCB can be equipped with a sliding window (Baudry et al., 2021b; Garivier and Moulines, 2011). This mechanism forces the bandit algorithm to overlook observations older than a given number of years, which consequently must regularly re-evaluate all fertilizer practices. Such approach reiterates the recommendations formulated by Adam et al. (2020): the bandit algorithm would handle climate change by regularly trying to improve current fertilizer practices.

General discussion

In the same way as past agricultural DSS, in Chapter 1, we envisioned a human-centered RL-based system as an "assistance to farmers in solving their own problems in their terms" (quoted from McCown and Parton, 2006). Two dimensions make, however, the design of such RL-based DSS difficult: (i) the highly complex and uncertain nature of the dynamical bio-physical system that a crop field is, and (ii) the necessity need to provide meaningful information to human decision makers. The canonical RL problems, where an explicit utility function is optimized (i.e. MDP) in a context of abundant data, are relatively simple compared to the crop management decision problems, with limited data and multiple and possibly contrasting objectives. Even for the canonical RL decision problems, the sample efficiency of RL algorithms is generally poor, which often require millions of interactions before the agent is able to solve a task (e.g. as discussed in Dulac-Arnold et al., 2019). Moreover, in agriculture, many confounding factors are found, in many possible crop growing conditions. Identifying best crop management practices is generally challenging, and may require the results from many field experiments, such as with meta-analysis (e.g. Giller et al., 2009, see Chapter 1). Field trials are often expensive multi-year experiments, hard to conduct, especially in the South. In the following, we discuss the research directions we took to start addressing the gap between the canonical RL applications and the reality of crop management support. We also highlight a few promising research directions for future work.

Addressing point (i): crop management is a complex decision problem

A first contribution of this work was to explore how RL could address complex crop management decision problems with limited data. We discuss the use of crop growth models, and how expert knowledge can be leveraged, to facilitate the learning of crop management.

Opportunities and limitations of crop models for RL. Despite that limited data is generally available at fine granularity in countries in the South, crop models, once calibrated, allow to generate a very large number of simulated cropping cycles at a negligible computational cost. With simulated decision problems, the use of RL algorithms is thus not constrained by the data availability. Crop models can be turned into RL environments (e.g. Garcia and Ndiaye, 1998, as an early example), and RL algorithms can be successfully applied to learn sustainable crop management with accurate crop growth simulations (Chapter 2). Yet, in the context of a crop model calibrated based on limited field data, the statistical significance of the identification, be it with RL or other optimization methods, of best crop management options from crop model simulations to real field conditions is unlikely to be supported by enough statistical evidences (Chapter 3). This led to the question whether RL-based models are able to learn directly from field experiments.

Going beyond model simulations. As identifying best crop management options from model simulations to real field conditions is inherently limited by the accurateness of the crop growth

simulations, a central question is: what are the available levers to directly identify these best options from real-field experiments? A first step is to limit the sample complexity of the decision problem. The agronomy benefits from a considerable amount of expert knowledge that should be exploited as much as possible. It is unlikely for a decision problem to be completely unknown to decision makers. Researchers, agricultural extensionists or farmers can all leverage their expert knowledge, be it technical or theoretical (including model simulations), to jointly formulate priors about the solutions to explore for a given problem (e.g. the collaborative *what-if* analysis in Thorburn et al., 2011). Considering the task of sustainable maize nitrogen fertilization, Chapter 4 provides such an example. We leveraged expert knowledge to reduce a high-dimensional sequential decision problem, i.e. the choice of a continuous fertilizer quantity everyday of the growing season depending on the field state comprising continuous and discrete variables (as in the use case of Chapter 2), to a single decision at the beginning of the growing season, i.e. a choice of one rule-based state-dependent fertilization practice predetermined by experts. We also used expert knowledge to group the farmers and their fields in cohorts with similar crop growing conditions. Following the reward shaping principle (Ng et al., 1999; Randløv and Alstrøm, 1998), expert knowledge can also be employed, as in Chapter 2 and 4, to define a real-valued objective function, such that maximizing this function leads to desirable agronomic, economic and environmental trade-offs.

Considering a given decision problem, an ensemble of field trials can be performed in parallel at each time step, called batch learning (Perchet et al., 2015), instead of one trial at each time step, which we call a purely sequential setting (Lattimore and Szepesvári, 2020). The former setting allows, for a same period, to increase the total number of interactions with an RL model compared to the latter. For instance, in Chapter 4, we considered a collaborative identification of best nitrogen fertilization practices, supported by a group of farmers who simultaneously carried out field trials each growing season. Such experiments can be found with on-farm trials (e.g. Baudron et al., 2012; Falconnier et al., 2016; Naudin et al., 2010). As a result of this redefinition of the decision problem, a bandit-based algorithm was able to efficiently identify the best nitrogen fertilization practices after a few years. Nevertheless, in real conditions, an ensemble of field trials carried out during the same year induces a structure with correlated groups. For instance, for a given year, nearby farmers are expected to experience similar weather. These correlations still need to be addressed for the methods introduced in Chapter 4, for instance by adapting contextual bandits (Lattimore and Szepesvári, 2020) with concepts from mixed linear models (Laird and Ware, 1982).

Addressing point (ii): designing relevant formal models for farmers

Bio-economical formal models should primarily target their usefulness to practitioners (Charlton and Street, 1975, p. 263-265). Farmers' natural decision-making processes should be taken into account in the design of agricultural DSS. These systems should allow a user to explore pragmatic solutions rather than delivering optimized solutions, as highlighted by Hochman and Carberry (2011). We discuss how we accounted for some of these aspects in the design of RL-based models.

Accounting for the uncertainty and risk in decisions. Decision support should be targeted on the characterization of the uncertainty of farm decisions (McCown et al., 2006; McCown and Parton, 2006). Beyond the fact that RL inherently addresses uncertain sequential decision problems, we explored the use of risk-aware decision criteria with mean-variance (MV, Markowitz, 1952) and the

conditional value-at-risk (CVaR, Mandelbrot, 1997), respectively in Chapter 3 and 4. Such metrics are appealing for food security questions where major crop failures must be avoided. We did, however, not address how to objectively choose a relevant value of parameter for the MV or CVaR (respectively $\rho > 0$ and $\alpha \in (0, 1]$, see Equation 3.1 and Equation 4.7) for each farmer. Choosing the right risk parameter for a decision maker is not a trivial choice, as pointed out by Freund (1956). Techniques for risk preference elicitation can be found in relevant literature (e.g. Iyer et al., 2020, in European context), and a proper translation of risk preferences into the parameters of MV or CVaR should be studied.

Exploration-exploitation dilemma. In Chapter 4, we explored the cumulated regret minimization (CRM) which addresses the exploitation-exploration dilemma, combined with a risk-aware decision criterion. Minimizing the crop yield losses while exploring new crop management options is a documented behavior of farmers (e.g. Cerf and Meynard, 2006; Evans et al., 2017). In the experiments of Chapter 4, by minimizing the yield losses, we showed that the CRM allowed to reduce the cost of the identification of the best nitrogen fertilization practice from field trials performed by a group of virtual farmers in the conditions of Southern Mali, compared to the conventional approach. We argue that the cumulated regret minimization framework for bandits, or its extensions to the general RL case (see Chapter 1), is a novel approach that should be further explored for crop management support.

Future work

We detail some of the research directions we deem the most stimulating amongst the many-ones possible we have extensively discussed in Chapter 1. As a general guideline, we recommend the studies on RL applied to crop management support to follow a multi-disciplinary approach. Agricultural DSS are, by nature, at the confluence of many disciplines, including the agronomy, computer science, sociology, economics, cognitive science and ergonomics. Multi-disciplinary understanding of agricultural DSS proved to be a requirement for their relevance (Cerf and Meynard, 2006; McCown et al., 2006; McCown and Parton, 2006). We also recommend that all expert knowledge, including the practical knowledge of farmers, should be exploited for pragmatic solutions to very complex decision problems in a *satisficing* way (see Hochman and Carberry, 2011), rather than an absolute search of optimal solutions from a machine-learning perspective.

6-month time horizon

Contextual batch bandits The combination of the batch bandit approach introduced in Chapter 4 and contextual bandits should be explored. Contextual bandits allow to exploit the *context* of decisions, i.e. additional information. Contexts are somewhat the equivalent of states in MDP. For instance, in Chapter 4, contexts were defined by experts as a discrete set of cohorts of farmers and their fields, in which the crop growing conditions and thus the optimal fertilizer practices were expected to be similar. A central question with contextual bandits is to define the class of functions that map decision contexts to the results of actions and/or a distance between decision contexts. Kernel methods (Hofmann et al., 2008) allow to enhance a linear model, in order to fit a large class of non-linear functions. Kernel bandits (e.g. Krause and Ong, 2011) combine contextual bandits and kernel methods into versatile contextual bandit algorithms. With the case of *in-vivo* bandit-based sequential treatments for mice,

Durand et al. (2018) provides highly relevant materials that would be adapted for bandit-based crop management support. However, it is *a-priori* unclear if contextual bandits would allow a sufficient sample efficiency to learn crop management tasks from a limited number of interactions, as we pursued in Chapter 4. Indeed, decision contexts are likely to contain a large number of continuous and discrete features. As an example, the soils in crop models are usually defined by dozens of discrete and continuous variables. Expert knowledge can help transforming decision contexts into more compact representations, for instance using the soil fertility capability classification (FCC, Sanchez et al., 2003) which summarizes a large number of parameters into a finite number of soil classes with distinct crop growing conditions.

Accounting for observational cost. In the canonical RL problems, at each time step, the agent observes its environment at no cost. In agriculture, this assumption is particularly wrong. The cost of observations, depending on the measurement precision (e.g. detailed field sample analysis against remote sensing) and on the measurement frequency (e.g. avoiding redundant measurements) should be minimized. The same remarks hold for agent returns. Few studies address this question (e.g. Bellinger et al., 2020; Krueger et al., 2020). Minimizing the observational cost could also be envisioned by only updating most important observation features before making new decisions.

Addressing climate change. In the context of a changing climate, the optimal crop operations are likely to change through time. Climate change consequently turns stationary decision problems into non-stationary decision problems. RL offers specific solutions to address non-stationary problems (see Chapter 1). Because RL provides an active, gradual adaptation of actions as changes arise, it appears as a unique complementary tool to the common assessment of climate change impact and its mitigation through crop-model simulations (White et al., 2011). The crop management RL environment `gym-DSSAT` (Chapter 2) features built-in changing temperature, rainfall and CO_2 concentration, easy to use for simulated RL experiments.

5-year time horizon

Real-world application of bandits At a 5-year time horizon, the first exploitable results from real field experiments to identify best crop management practices using a bandit-based strategy (as developed in Chapter 4) should be available. We discussed a few elements for its practical implementation in Chapter 1. Nevertheless, the evaluation of a bandit-based identification method might face counter-intuitive aspects for stakeholders, due to the inherent uncertainty of these identification problems. First, for a single application of the identification method, after a few years of results, the uncertainty in the inter-annual statistics, such as an average result, might still be high (depending on the number of observations). This uncertainty will be larger for risk-aware inter-annual statistics, such as the MV or the CVaR, compared to the mean (risk-neutral statistic). Second, the statistical guarantees of an identification method of best crop management options hold in the face of *many decision problems*. To exemplify, one cannot support that climate change occurs because of a single observation of an outlier. For instance if a single mean temperature of a given month is observed above the 90^{th} percentile of all recorded previous temperatures. One can support that climate change occurs if such events are repeatedly observed. In the same way, a single application of an identification method is not meaningful, but rather the method should be evaluated by multiple applications in

different contexts (e.g. multi-year experiments in different locations), to conclude on its performance. Yet, such large-scale experiments might be difficult to conduct, hence the importance of prior extensive preliminary model simulations to support the validity of an identification method.

RL-oriented data collection. Offline RL, i.e. RL algorithms able to learn from already collected data sets, rather than from active interactions, is an active research area with great perspectives to address real-world decision problems (Levine et al., 2020). Despite that many data sets of agricultural trials are digitally accessible[***], a small fraction of all data sets corresponds to repeated measures of the evolution of a field state through time, as affected by all crop management operations, as RL requires. The data used for crop model calibration may be the closest example. For instance, crop models are usually calibrated using daily measures of plant leaf area index, soil water and nitrogen content, and crop phenological phases. Such data sets could be collected at the opportunistic occasion of other field experiments. Finally, all these data sets could be compiled in a dedicated platform to promote the use of data and RL for crop management support.

High sample efficiency RL algorithms. The general RL framework allows to successively consider sequences of decisions during the same episode, as opposed to the bandit framework. Using a compact state representation, model-based RL could have high learning efficiency in real-world agricultural applications (e.g. Deisenroth and Rasmussen, 2011)[†††]. Model-based RL could be combined with a special kind of transfer learning which relies on the pre-training of an RL agent in simulated conditions before it learns to act in real conditions (e.g. Golemo et al., 2018). Offline data, as mentioned above, could also be used for transfer learning. In the same was as in Chapter 4, POMDP with increasing state and action spaces should be considered, to limit the sample complexity of decision problems and thus, to limit the volume of required interactions to solve these problems. Collaborations could be envisioned with RL researchers working on robotics or healthcare, as all real applications in these fields are expensive, with highly constrained (possibly noisy) small data, and with a prominent risk of bad decisions.

10-year time horizon

Towards a human-machine dialog. Although we provided formal elements to account for some of the decision determinants of farmers, we did not directly address how an RL-based model could allow the human-machine dialog which DSS are supposed to provide (Arnott and Pervan, 2005; Power, 2008). Whereas being in its early age, causal RL (Dasgupta et al., 2019; Gasse et al., 2021; Madumal et al., 2020) may provide features for such human-machine dialog. Indeed, causal RL allows counterfactual reasoning, which has been a successful way for decision makers to interact with formal decision models in agriculture (see 'what-if analysis' in McCown, 2012). Furthermore, because causal RL leverages causal models at the symbolic level, it then provides RL policies in an intelligible form to decision makers for human-machine interactions (Madumal et al., 2020), which is an interesting feature for users' trust (Chapter 1).

[***] For instance, using https://gardian.bigdata.cgiar.org/.
[†††] For a demonstration of PILCO, see https://www.youtube.com/watch?v=XiigTGKZfks.

General conclusions

In this thesis, we explored how reinforcement learning (RL) could improve the decision support of crop management in the case of smallholder farmers. Based on the criticism on agricultural decision support systems (DSS), we first carried out a literature review and an exploratory exercise to identify promising research directions and, to define a conceptual framework for the application of RL. RL has a great potential for crop-management support. It inherently deals with sequences of decisions to control an unknown uncertain dynamical system. It is *de facto* geared towards action-making. Furthermore, RL shares some similarities with how farmers have been described to address crop management. In particular, RL learns a task by trial and error, and action-making is determined by observing the uncertain evolution of the system (in the case of crop management, the field crop). Nevertheless, RL also faces many challenges to its application to crop management, as demonstrated by the limited amount of relevant literature. The reality of crop management is far more challenging than the canonical RL decision problems. The context of countries in the South makes the use of RL for crop management support even more challenging, in particular because of data scarcity.

In our study, we considered decreasing numbers of possible interactions between an RL agent and its environment: from an almost infinite number to a few thousands. We provided a generic method to turn crop growth models into standardized and easy to manipulate RL environments. Crop models turned into RL environments allow to extensively train RL agents at a negligible computational cost. In an RL crop management environment, we were able to learn and explore sustainable fertilizer (and irrigation) practices with a commonly used RL algorithm. We also provided a method to quantify the statistical significance of a decision on a best management practice (whether it be identified by RL or other optimization methods), from crop model simulations to real field conditions. We considered both risk-neutral and risk-aware decision criteria. We took a decision on planting dates for maize in Canada, based on the crop model simulations of a long-term field experiment as a use case. We showed the value of such risk-aware decision criterion in the face of the seasonal weather uncertainty, especially in the context of food security where major crop yield losses should be avoided. However, in the case of countries in the South, we concluded that for most applications, the statistical significance of this identification of the best crop management practices from crop model simulations to reality was unlikely to be supported by enough statistical evidences.

The limits of simulated environments led us to study how RL could be employed to directly learn in real conditions, with a practicable number of interactions. We considered the collaborative identification of the crop best management practices by a group of farmers performing field trials. In a simulated exercise, we mimicked the crop growing conditions of Southern Mali. We designed an identification method based on a bandit algorithm using a risk-aware decision criterion, with the constraint of minimizing farmers' crop yield losses occurring during the identification process. Simulations showed that, through exploiting expert knowledge to reduce the sample complexity of the decision problem, the identification method could be realistically employed in real field conditions. In most cases, the bandit algorithm was better at reducing farmers' yield losses than equi-proportional field trials during a fixed number of years.

RL may give a new breath to crop management DSS, and is of interest for the contexts of countries in the South. Its application requires the design of *ad-hoc* algorithms, able to deal with the many constraints and objectives of crop-management support. A field and its crop management are elements of a larger agricultural system which comprises the whole farm, and its ecosystem *sensu lato*, including the social dimensions. Consequently, the design of RL-based agricultural DSS goes far beyond a simple numerical problem. By addressing some key aspects of crop management support, we showed that such adaptations from the canonical RL problems to the reality of crop management, were possible. The arising of novel promising technologies should not distract us from exploiting all the knowledge and tools already developed in the history of DSS, involving multiple disciplines, including economics, sociology, cognitive sciences and ergonomics. An application of RL will most likely be successful if the tradition of a multi-disciplinary approach is maintained, including at the stage of the design of formal RL-based decision models.

Supplementary Materials A

(corresponds to Chapter 2)

A.1 Irrigation use case

We provide a simple baseline for the irrigation problem, as introduced in Section 2.4.1.

Methods Overall, the irrigation use case follows the same methods than the fertilization use case (Section 2.6.1). It only differs from the fertilization use case in the observation space and return function. Table A.3 details the default observation space for the irrigation problem. Denoting $\texttt{topwt}(t, t+1)$ the above the ground population biomass change between t and $t+1$ (kg/ha); and $\texttt{amir}(t)$ the irrigated water on day t (L/m^2), the default irrigation return function was defined as:

$$r(t) = \underbrace{\texttt{topwt}(t, t+1)}_{\substack{\text{change in above} \\ \text{the ground biomass}}} - \underbrace{15}_{\substack{\text{penalty} \\ \text{factor}}} \times \underbrace{\texttt{amir}(t)}_{\substack{\text{irrigated water} \\ \text{quantity}}} \tag{A.1}$$

We considered 3 different policies:

- The 'null' policy that never irrigated, which corresponded to rainfed crops. Agronomists may measure the effect of an irrigation policy as a performance gain compared to the null policy, in order to decouple the effect of irrigation from the effect of rainfall (Howell, 2003).
- The second baseline was the "expert" policy, which was an approximation of the irrigation policy of the original maize field experiment (Bennett et al., 1989, `UFGA8201` experiment number #3), see Section 2.4.1. As Table A.1 shows, this policy consisted in sixteen deterministic water applications, which only depended on the number of days after planting. In contrast with the fertilization expert policy (Table 2.3), this irrigation expert policy was a simplistic approximation of the true expert policy of the original field experiment. Indeed, the true expert policy, unavailable, was likely to depend on more factors (e.g. soil moisture, or days without effective rainfall in a given growth stage) rather than only on days after planting. Nevertheless, the irrigation policy in Table A.1 was still a convenient baseline for this experiment.
- The policy learned by PPO.

For an irrigation policy π, denoting $\texttt{grnwt}^{\pi}$ the dry matter grain yield of the policy π (kg/ha) ; $\texttt{grnwt}^{0}$ the dry matter grain yield with no fertilization (kg/ha); and $\texttt{totir}^{\pi}$ the total irrigated water with

DAP	quantity (L/m^2)
6	13
20	10
37	10
50	13
54	18
65	25
69	25
72	13
75	15
77	19
80	20
84	20
91	15
101	19
104	4
105	25

Table A.1 Expert irrigation policy. 'DAP' stands for Day After Planting.

variable	definition	comment
`grnwt`	grain yield (kg/ha)	quantitative objective to be maximized
`totir`	total irrigation (L/m^2)	cost to be minimized
–	application number	cost to be minimized
–	water use efficiency (kg/m^3)	agronomic criteria to be maximized
`runoff`	running-off water (L/m^2)	loss to be minimized
`cleach`	nitrate leaching (kg/ha)	loss/pollution to be minimized

Table A.2 Performance indicators for irrigation policies. An hyphen means `gym-DSSAT` does not directly provide the variable, but it can be easily derived.

policy π (L/m^2), we define the water use efficiency (WUE, Equation 15, Howell, 2003) as:

$$\text{WUE}^{\pi}(t) = 10 \times \frac{\texttt{grnwt}^{\pi}(t) - \texttt{grnwt}^{0}(t)}{\texttt{totir}^{\pi}(t)} \tag{A.2}$$

Similarly to the fertilization use case, Table A.2 shows the performance indicators we considered for the irrigation problem. In particular, for excessive irrigation, nitrate leaching may increase (Meisinger and Delgado, 2002). Thus, nitrate leaching is a pollution performance indicator of irrigation.

Results Regarding the maximization of the undiscounted objective function, PPO showed the best mean performance and slightly outperformed the expert policy, but had an increase variance than the latter, see the wider range of values between upper and lower quantiles in Figure A.1a. PPO water applications were more frequently found between days 80 and 120 of the simulation, which mostly corresponds to the grain filling stage, see Table A.5 in Appendix. During this period, in most cases, PPO irrigated less water than the expert policy, see Figure A.1b. As indicated in Table A.4, PPO irrigation policy consumed in average about 49% less water than the expert policy, while maintaining a maize grain yield close to the one of the expert policy. Consistently, the water use efficiency (Equation A.2) of PPO policy was 54% higher than for the expert policy. Total nitrate leaching for PPO policy was

	definition
istage	DSSAT maize growing stage
vstage	vegetative growth stage (number of leaves)
grnwt	grain weight dry matter (kg/ha)
topwt	above the ground population biomass (kg/ha)
xlai	plant population leaf area index (m^2 leaf/m^2 soil)
tmax	maximum temperature for current day °C
srad	solar radiation during the current day (MJ/m^2/day)
dtt	growing degree days for current day (°C.day ; base temp. 6.2 °C)
dap	duration after planting (day)
sw	volumetric soil water content in soil layers (cm^3 [water] / cm^3 [soil])
ep	actual plant transpiration rate (L/m^2/day)
wtdep	depth to water table (cm)
rtdep	root depth (cm)
totir	total irrigated water (L/m^2)

Table A.3 Default observation space for the irrigation task.

	null	expert	PPO
grain yield (kg/ha)	3734.8 (1852.2)	**8306.6** (562.0)	7082.2 (1455.7)
total irrigation (L/m^2)	**0** (0)	264.0 (0)	133.8 (40.3)
application number	**0** (0)	16.0 (0.0)	16.2 (3.7)
water use efficiency (kg/m^3)	n.a.	17.3 (7.1)	**26.3** (13.6)
runoff (L/m^2)	**0.4** (3.5)	**0.4** (3.5)	**0.4** (3.5)
nitrate leaching (kg/ha)	**18.5** (12.6)	24.6 (9.0)	18.7 (9.6)

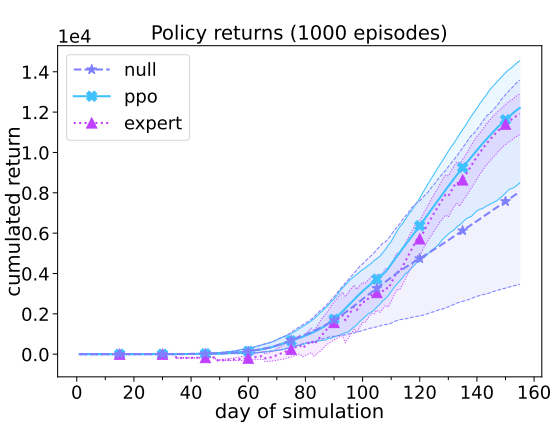
Table A.4 Mean (st. dev.) irrigation baselines performances computed using 1000 episodes. For each criterion, bold numbers indicate the best performing policy.

very close to the null policy, and was about 24% less than for the expert policy. The number of water applications were similar for both expert and PPO policies. Null, expert, and PPO policies had similar water runoff, indicating no water loss due to excessive irrigation of expert or PPO policies.

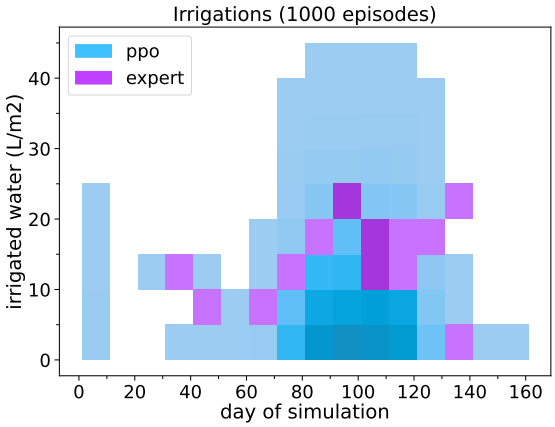
Discussion PPO showed a great efficiency advantage over the expert policy, while maintaining a comparable average grain yield. Water applications of PPO were most frequently focused during maize anthesis period, where maize water needs are the greatest and most crucial with respect to grain yield (NeSmith and Ritchie, 1992). Because the expert irrigation policy was likely to be a poor simplification of the real expert irrigation strategy, the advantage PPO irrigation strategy showed might be overly optimistic. However, because PPO has shown largely reduced irrigated water and nitrate leaching, we still deem these results interesting. An alternative baseline could be to reproduce the built-in automatic irrigation policy implemented in DSSAT (Hoogenboom et al., 2019), and compare its performance to the irrigation policy of PPO.

A.2 Fertilization use case complement

Table A.6 provides statistics about the growth stages for the three policies of the fertilization use case.



(a) Mean cumulated return of each of the 3 policies against the day of the simulation. Shaded area displays the $[0.05, 0.95]$ quantile range for each policy.



(b) 2D histogram of irrigations (the darker the more frequent).

Figure A.1 Undiscounted cumulated returns and applications for the irrigation problem.

`istage`	meaning	**null**	**expert**	**ppo**
8	50% of plants germinated	28 (0)	28 (0)	28 (0)
9	50% of plants with some part visible at soil surface	29 (0)	29 (0)	29 (0)
1	end of juvenile stage	40 (3)	40 (3)	40 (3)
2	50% of plants completed floral initiation	64 (4)	64 (4)	64 (4)
3	50% of plants with some silks visible outside husks	69 (4)	69 (4)	69 (4)
4	beginning of grain filling	110 (4)	110 (4)	110 (4)
5	end of grain filling	120 (4)	120 (4)	120 (4)
6	50% of plants at harvest maturity	158 (4)	158 (4)	158 (4)

Table A.5 Mean (st. dev.) days of simulation to reach growth stages for the irrigation problem (1000 episodes).

`istage`	meaning	**null**	**expert**	**ppo**
8	50% of plants germinated	22 (1)	22 (1)	22 (1)
9	50% of plants with some part visible at soil surface	23 (1)	23 (1)	23 (1)
1	end of juvenile stage	34 (3)	34 (3)	34 (3)
2	50% of plants completed floral initiation	60 (5)	60 (5)	60 (5)
3	50% of plants with some silks visible outside husks	65 (5)	65 (5)	65 (5)
4	beginning of grain filling	107 (4)	107 (4)	107 (4)
5	end of grain filling	117 (4)	117 (4)	117 (4)
6	50% of plants at harvest maturity	155 (5)	155 (5)	155 (5)

Table A.6 Mean (st. dev.) days of simulation to reach growth stages for the fertilization problem (1000 episodes).

Supplementary Materials B

(corresponds to Chapter 3)

B.1 Uncertainty vocabulary.

Dealing with uncertainty requires clearly defined semantics, as word use can sometimes be inconsistent in the literature. Out of preciseness, we here follow the definitions from The Guide to the Expression of Uncertainty in Measurement (JCGM et al., 2008). **True value** is defined as "a value that would be obtained by a perfect measurement" ; **error** as "result of a measurement minus a true value of the measurand" ; and finally **uncertainty** as a "parameter, associated with the result of a measurement, that characterizes the dispersion of the values that could reasonably be attributed to the measurand", for instance a confidence interval. The error definition is refined with the concept of **random error**: "result of a measurement minus the mean that would result from an infinite number of measurements of the same measurand carried out under repeatability conditions" ; and of **systematic error**: "mean that would result from an infinite number of measurements of the same measurand carried out under repeatability conditions minus a true value of the measurand".

B.2 Biased model error

In this section, we provide the concentration inequalities for the case where the model error distribution E is assumed biased: $\mathbb{E}[E] \neq 0$. The resulting confidence intervals are wider than with the centered error assumption (for an equal number of observations) and the second order sub-Gaussian concentration inequalities for a centered random variable provided in Section B.3.4 do not hold anymore.

Ground truth distribution Without assuming the error distribution to be centered but the simulated and error distributions still to be uncorrelated, and following the same conventions as in Section 3.2:

$$\mathrm{Y}_{\text{true}} = \mathrm{Y}_{\text{sim}} + E\,, \tag{B.1}$$

$$\mu_{\mathrm{Y}_{\text{true}}} = \mu_{\mathrm{Y}_{\text{sim}}} + \mu_E\,, \tag{B.2}$$

$$\sigma^2{}_{\mathrm{Y}_{\text{true}}} = \sigma^2{}_{\mathrm{Y}_{\text{sim}}} + \sigma_E^2 + \underbrace{2\mathrm{Cov}(\mathrm{Y}_{\text{sim}}, E)}_{0}\,. \tag{B.3}$$

Ground truth mean uncertainty The ground truth mean uncertainty is defined by the interval $[\underline{\mu_{\mathrm{Y_{true}}}}(\delta/2), \overline{\mu_{\mathrm{Y_{true}}}}(\delta/2)]$ such that for $\delta \in (0, 0.5)$ with probability $1-\delta$:

$$\mu_{\mathrm{Y_{true}}} \in [\underline{\mu_{\mathrm{Y_{true}}}}(\delta/2), \overline{\mu_{\mathrm{Y_{true}}}}(\delta/2)]\,, \tag{B.4}$$

$$\Leftrightarrow \mu_{\mathrm{Y_{true}}} \in [\underline{\mu_{\mathrm{Y_{sim}}}}(\delta/4) + \underline{\mu_E}(\delta/4), \overline{\mu_{\mathrm{Y_{sim}}}}(\delta/4) + \overline{\mu_E}(\delta/4)]\,. \tag{B.5}$$

Ground truth mean-variance uncertainty The ground truth mean variance uncertainty is given by $\underline{\mathrm{MV}}_{\mathrm{Y_{true}}}(\delta/2), \overline{\mathrm{MV}}_{\mathrm{Y_{true}}}(\delta/2)$ such that for $\delta \in (0, 0.5)$ with probability $1-\delta$:

$$\mathrm{MV}_{\mathrm{Y_{true}}} \in [\underline{\mathrm{MV}}_{\mathrm{Y_{true}}}(\delta/2), \overline{\mathrm{MV}}_{\mathrm{Y_{true}}}(\delta/2)]\,, \tag{B.6}$$

$$\underline{\mathrm{MV}}_{\mathrm{Y_{true}}}(\delta/2) = \underline{\mu_{\mathrm{Y_{true}}}}(\delta/8) + \underline{\mu_E}(\delta/8) - \rho\sqrt{\overline{\sigma^2}_{\mathrm{Y_{sim}}}(\delta/8) + \overline{\sigma^2_E}(\delta/8)}\,, \tag{B.7}$$

$$\overline{\mathrm{MV}}_{\mathrm{Y_{true}}}(\delta/2) = \overline{\mu_{\mathrm{Y_{true}}}}(\delta/8) + \overline{\mu_E}(\delta/8) - \rho\sqrt{\underline{\sigma^2}_{\mathrm{Y_{sim}}}(\delta/8) + \underline{\sigma^2_E}(\delta/8)}\,. \tag{B.8}$$

B.3 Confidence intervals

In this section, we detail some standard tools to build confidence intervals, discuss their shortcomings for small samples of unspecified distributions, and introduce alternative confidence bounds motivated by the problem of crop yield estimation. We summarize the various confidence intervals and their use cases in Figure B.1. Our implementation is available as part of the `concentration_lib` library*.

B.3.1 Union bounds

The *union bound* argument is a generic method to control the probability that multiple events simultaneously hold from their individual probabilities. If A, B are two measurable events, then $\Pr(A \cup B) \leq \Pr(A) + \Pr(B)$, with equality if and only if A and B are almost surely disjoint ($\Pr(A \cap B) = 0$).

In the context of confidence estimation for a quantity ϕ_X, events are typically of the form $A = \{\phi_X \leq \overline{\phi_X}(\delta)\}$ and $B = \{\phi_X \geq \underline{\phi_X}(\delta)\}$, with probability $\Pr(A) \geq 1-\delta$, $\Pr(B) \geq 1-\delta$. The corresponding two-sided confidence set is given by the intersection $A \cap B$. Using the fact that the complement of the intersection of two sets is the union of their complements, we get:

$$\Pr(A \cap B) = 1 - \Pr\left(\overline{A} \cup \overline{B}\right) \geq 1 - \Pr\left(\overline{A}\right) - \Pr\left(\overline{B}\right) \geq 1 - 2\delta\,. \tag{B.9}$$

Therefore, two one-sided confidence sets at level $1-\delta$ can be combined into a two-sided set at level $1-2\delta$, or equivalently:

$$\begin{cases} \Pr\left(\phi_X \leq \overline{\phi_X}(\delta/2)\right) & \geq 1-\delta/2 \\ \Pr\left(\phi_X \geq \underline{\phi_X}(\delta/2)\right) & \geq 1-\delta/2 \end{cases} \implies \Pr\left(\underline{\phi_X}(\delta/2) \leq \phi_X \leq \overline{\phi_X}(\delta/2)\right) \geq 1-\delta\,. \tag{B.10}$$

*https://github.com/sauxpa/concentration_lib

The same method holds for the union of more than two events, leading to the following principle, ubiquitous in the present work: to build a confidence set on I simultaneous events at level $1-\delta$, it is enough to have confidence set at level $1-\frac{\delta}{I}$ for each individual event.

B.3.2 The case of Gaussian distributions

We recall here standard confidence intervals for the mean and standard deviation of normally distributed random variables. By virtue of the central limit theorem, the results below hold for any square-integrable random variable, not necessarily Gaussian, in the limit of large sample size $n \to +\infty$. As we are interested in finite. possibly small samples, we will later establish confidence sets under milder assumptions.

Confidence interval with known variance

We recall that a real-valued random variable Y is said to be Gaussian with mean μ and variance σ^2 if for any range $[a,b] \subset \mathbb{R}$, the probability that Y lies in $[a,b]$ is $\int_a^b \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(y-\mu)^2}{2\sigma^2}} dy$. Moreover, if Z is a standard Gaussian variable, that is of mean zero and unit variance, then the distribution of Y is equal to that of $\mu + \sigma Z$. We denote by $\Phi(a) = \int_{-\infty}^a \frac{1}{\sqrt{2\pi}} e^{-\frac{y^2}{2}} dy$ the probability that $Z < a$. The function Φ, known as the cumulative distribution function (cdf) of Z, as well as its inverse Φ^{-1}, the quantile function, can be easily tabulated and are available up to arbitrary precision in most statistics libraries.

If one observes n i.i.d samples $Y_1, \ldots, Y_n$ distributed as Y, then their empirical average $\widehat{\mu}_n = \frac{1}{n}\sum_{i=1}^n Y_i$ follows a Gaussian distribution of same mean μ and variance $\frac{\sigma^2}{n}$. Assuming the variance σ^2 is known, the *exact* confidence interval at level $\delta \in (0,1)$ for μ is then:

$$\Pr\left(\widehat{\mu}_n - \frac{\sigma}{\sqrt{n}}\Phi^{-1}(1-\delta/2) \leq \mu \leq \widehat{\mu}_n - \frac{\sigma}{\sqrt{n}}\Phi^{-1}(\delta/2)\right) = 1-\delta\,. \tag{B.11}$$

Confidence interval for the standard deviation

The natural unbiased estimator for the variance is $\widehat{\sigma}_n^2 = \frac{1}{n(n-1)}\sum_{1\leq i<j\leq n}(X_i - X_j)^2$. We recall that a sum of n i.i.d standard Gaussian variables follows the $\chi^2(n)$ distribution, and its cdf and quantile function $q_{\chi^2(n)}$ are also tabulated and available at arbitrary numerical precision. The quantity $\frac{n-1}{\sigma^2}\widehat{\sigma}_n^2$ is known to follow a $\chi^2(n-1)$ distribution. We thus obtain the following confidence interval.

Chi2 standard deviation bound

$$\Pr\left(\widehat{\sigma}_n\sqrt{\frac{(n-1)}{q_{\chi^2(n-1)}(1-\delta/2)}} \leq \sigma \leq \widehat{\sigma}_n\sqrt{\frac{(n-1)}{q_{\chi^2(n-1)}(\delta/2)}}\right) = 1-\delta\,. \tag{B.12}$$

Confidence interval for the mean with unknown variance

In many applications where μ is unknown and requires estimation, so is σ, thus precluding the use of Equation B.11. Replacing σ with its empirical estimator $\widehat{\sigma}_n$ leads to a tractable confidence interval, again based on a tabulated quantile function. More precisely, under the Gaussian hypothesis,

$T_n = \frac{\sqrt{n}(\widehat{\mu}_n - \mu)}{\widehat{\sigma}_n}$ follows the Student $t(n-1)$-distribution, the quantile function of which we denote by $q_{t(n-1)}$. The resulting *empirical* confidence interval reads:

Student

$$\Pr\left(\widehat{\mu}_n - \frac{\widehat{\sigma}_n}{\sqrt{n}} q_{t(n-1)}(1-\delta/2) \leq \mu \leq \widehat{\mu}_n - \frac{\widehat{\sigma}_n}{\sqrt{n}} q_{t(n-1)}(\delta/2)\right) = 1-\delta\,. \tag{B.13}$$

Note that this is a reformulation of the standard Student's *t*-test.

B.3.3 The case of bounded distributions

The above confidence intervals derived under the Gaussian hypothesis are the tightest possible because they are based on the knowledge of the *exact* distributions of $\widehat{\mu}_n$, $\widehat{\sigma}_n$ and T_n. However, as shown in Appendix B.6, this hypothesis is rejected for the distribution of Y_{sim} considered in our use-case, and we observe too few calibration errors to confidently conclude on the distribution of E. A more natural assumption is that the studied distribution is supported in an interval $[\underline{B}, B]$, for instance $\underline{B} = 0$ and $B = y_{\text{max}}$ for Y_{sim}, where y_{max} is provided by expert knowledge (e.g the maximum possible yield per hectare is limited by physical constraints and the chosen crop management policy). We detail below the statistical concentration properties required to derive confidence intervals based solely on the boundedness, without further prior knowledge.

Small samples concentration for the mean

Phan et al. (2021)[†] recently proposed new confidence sets for bounded distributions based on random histogram resampling. We recall below the main steps of their method.

Let $y_1, \ldots, y_n$ be i.i.d realizations of Y, and let $y_{(1)} \leq \cdots \leq y_{(n)}$ denote the corresponding sorted sequence (in increasing order). For the sake of simplicity, we will consider the random variable

$$X = (Y - \underline{B}) \,/\, (B - \underline{B})\,, \tag{B.14}$$

so that the sequence of sorted observations $x = \left(x^{(i)}\right)_{i=1,\ldots,n}$ lies in $[0,1]$ ($y_{(i)}$ can be deduced from $x_{(i)}$ with an order-preserving linear transform and vice-versa so a confidence interval for one immediately implies one for the other).

The empirical mean of the random variable X can be expressed as $\widehat{\mu_{x_n}}(u) = \frac{1}{n}\sum_{i=1}^{n} x^{(i)} = \sum_{i=1}^{n+1} x^{(i)}\left(u_{(i)} - u_{(i-1)}\right)$, where $u = (1, 1/n, \ldots, (n-1)/n, 1)$ is the sequence of n cumulative steps of size $1/n$. For ease of notation, we have by convention $u_{(0)} = 0$ and $x_{(n+1)} = u_{(n+1)} = 1$. To study the possible upper dispersion of $\widehat{\mu_{x_n}}$, Phan et al. (2021) replace the sequence u with n sorted observations of a uniform random variable U over $[0,1]$, an approach akin to bootstrapping. Their upper $(1-\delta)$-confidence bound is then the $1-\delta$ quantile of the maximal empirical mean obtained in this manner, where the randomization comes from the variable U. More precisely, we have

[†]See https://github.com/myphan9/small_sample_mean_bounds for the initial implementation.

Small sample PTLM

$$\begin{aligned} &\Pr\left(\mu_X \leq q_M(1-\delta)\right) \geq 1-\delta\,, \\ &M = \sup_{y \preccurlyeq x} \widehat{\mu_{y_n}}\left(U\right)\,, \\ &U \sim \text{Unif}\left([0,1]\right)\,, \end{aligned} \tag{B.15}$$

where $\preccurlyeq$ is a (weak) ordering on the set of sequences in $[0,1]^n$. An example of such ordering, which we will use from now on, is the ordering induced by the ℓ^2 norm, i.e $y \preccurlyeq x \iff \sum_{i=1}^n y_i^2 \leq \sum_{i=1}^n x_i^2$.

As an intricate function of the uniform distribution of U, the law of M cannot be computed explicitly, and in particular the $1-\delta$-quantile needs to be estimated. The authors advocate the use of the *Monte Carlo* method, which consists in sampling multiple independent realizations of U, say $U_1, \ldots, U_m$ for m large enough, and replacing $q_M(1-\delta)$ with the $1-\delta$-quantile of the sample $\left(q_{M_j}(1-\delta)\right)_{j=1,\ldots,m}$, where $M_j = \sup_{y \preccurlyeq x} \widehat{\mu_{y_n}}\left(U_j\right)$ for $j = 1, \ldots, m$ (for instance this quantile can be chosen to be $M_{(\lfloor (1-\delta)m \rfloor)}$, where $\left(M_{(j)}\right)_{j=1,\ldots,m}$ is the sequence $\left(M_j\right)_{j=1,\ldots,m}$ sorted in increasing order). Note that the number of simulations m is a free parameter that is *not* related to the sample size of x; in particular m can be made arbitrarily large so that the quantile $q_M(1-\delta)$ can be considered well estimated up to any given precision.

From Equation B.15 and the scaling transform B.14 provide an upper confidence bound on μ. Applying the same method to $-Y$ and negating the resulting bound gives the corresponding lower confidence bound. Finally, a two-sided interval around μ can be obtained thanks to the union bound argument.

Hedged capital concentration for the mean

While it guarantees rather sharp confidence intervals, the small samples method of Phan et al. (2021) is computationally costly for larger samples due to the high number of Monte Carlo simulations involved at each step. Other recent works on the concentration of bounded distributions include that of Waudby-Smith and Ramdas (2020), which we briefly detail below. Assuming X again denotes the variable rescaled in $[0,1]$, we define the following quantities, following the recommendations of the authors:

$$
\begin{aligned}
&\mathcal{K}_n(m) = \max(\mathcal{K}_n^+, \mathcal{K}_n^-(m))\,,\\
&\mathcal{K}_n^\pm(m) = \prod_{i=1}^{n} \left(1 \pm \lambda_i^\pm(m)\,(X_i - m)\right)\,,\\
&\lambda_i^+(m) = \min\left(|\lambda_i|, \frac{1/2}{m}\right), \lambda_i^-(m) = \min\left(|\lambda_i|, \frac{1/2}{1-m}\right)\,,\\
&\lambda_i = \sqrt{\frac{2\log(2/\delta)}{n\hat{\sigma}_{i-1}^2}}\,,\\
&\hat{\sigma}_i^2 = \frac{1/4 + \sum_{k=1}^{i}(X_k - \hat{\mu}_k)^2}{i+1}\,,\\
&\hat{\mu}_i = \frac{1/2 + \sum_{k=1}^{i} X_k}{i+1}\,.
\end{aligned}
\tag{B.16}
$$

Then, the following confidence result holds.

Hedged capital

$$
\{m \in [0,1] \colon \mathcal{K}_n(m) < 1/\delta\} \text{ is a confidence set of level } 1-\delta \text{ around } \mathbb{E}[X]\,. \tag{B.17}
$$

It is shown in Waudby-Smith and Ramdas (2020) that this set is indeed an interval, the lower and upper bounds of which can be found by numerically tracking the minimum and maximum $m \in [0,1]$ that satisfy $\mathcal{K}_n(m) < 1/\delta$ respectively.

Empirically, this method produces confidence bounds similar those of Phan et al. (2021) for sample sizes larger than 100, while being much faster to compute.

Confidence interval on the standard deviation via Bentkus-Pinelis concentration

A classical tool to control the concentration of standard deviation for bounded distributions is Maurer-Pontil inequality (Theorem 10 in Maurer and Pontil (2009)). However, confidence intervals derived from this approach proved too loose to provide meaningful guarantees in our case for small to medium sample sizes. We use instead a construction inspired by Bentkus-Pinelis concentration (see Lemma F.1 in Kuchibhotla and Zheng (2021) for a similar upper bound; we detail below the derivation of the two-sided interval).

Bentkus-Pinelis standard deviation bound

$$
\begin{aligned}
&\Pr\left(-q(\delta/2) + \sqrt{q(\delta/2)^2 + \widehat{\sigma}_n^2} \le \sigma \le q(\delta/2) + \sqrt{q(\delta/2)^2 + \widehat{\sigma}_n^2}\right) \ge 1-\delta\,,\\
&q(\delta/2) = \Phi^{-1}\left(1 - \frac{\delta}{2c}\right)\frac{(B-\underline{B})}{2\sqrt{2\lfloor n/2 \rfloor}}\,,\\
&c = \frac{e^2}{2} \approx 3.7\,.
\end{aligned}
\tag{B.18}
$$

U-statistics Estimating confidence bounds for standard deviation or variance offers an additional technical challenge compared to the mean. Indeed, the natural unbiased estimator $\widehat{\sigma}_n^2 = \frac{1}{n(n-1)} \sum_{1 \le i < j \le n} (X_i - X_j)^2$ is *not* a sum of i.i.d observations (the same X_i is repeated multiple times in the $n(n-1)/2$ terms of the sum). Following Hoeffding (1948), the sample variance can be rewritten as a sum of so-called quadratic U-statistics:

$$\begin{aligned} \widehat{\sigma}_n^2 &= \frac{1}{n!} \sum_{\tau \in \mathbb{S}_n} V_{\tau,n} \,, \\ V_{\tau,n} &= \frac{1}{2} \sum_{i=1}^{\lfloor n/2 \rfloor} \left(X_{\tau(2i)} - X_{\tau(2i-1)} \right)^2 \,, \\ \tau &\in \mathbb{S}_n \quad \text{(set of permutations of } \{1, \dots, n\}) \,. \end{aligned} \tag{B.19}$$

Note that for a given permutation $\tau \in \mathbb{S}_n$, $V_{\tau,n}$ is the sum of $\lfloor n/2 \rfloor$ i.i.d random variables.

Bentkus-Pinelis concentration The deviation probability of $\widehat{\sigma}_n^2$ can be controlled by the above U-statistics. Let $u > 0$, then $\Pr\left(\widehat{\sigma}_n^2 \ge \sigma^2 + u^2\right) = \mathbb{E}\left[\mathbb{1}_{\widehat{\sigma}_n^2 - \sigma^2 - u^2 \ge 0}\right] \le \mathbb{E}\left[\left(1 + \lambda\left(\widehat{\sigma}_n^2 - \sigma^2 - u^2\right)/2\right)_+^2\right]$ since $\mathbb{1}_{x \ge 0} \le (1 + \lambda x/2)_+^2$ for $\lambda > 0$. Jensen's inequality and the fact that $V_{\tau,n}$ and $V_{I,n}$ have the same law (I is the identity permutation) then yields

$$\Pr\left(\widehat{\sigma}_n^2 \ge \sigma^2 + u^2\right) \le \frac{1}{n!} \sum_{\tau \in \mathbb{S}_n} \mathbb{E}\left[\left(1 + \lambda\left(V_{\tau,n} - \sigma^2 - u^2\right)/2\right)_+^2\right] = \mathbb{E}\left[\left(1 + \lambda\left(V_{I,n} - \sigma^2 - u^2\right)/2\right)_+^2\right] . \tag{B.20}$$

Now let us consider the change of variable $x = u^2 - 2/\lambda \in (-\infty, u^2)$. Note that since λ is a free parameter, so is x. Therefore optimizing in x yields:

$$\Pr\left(\widehat{\sigma}_n^2 \ge \sigma^2 + u^2\right) \le \inf_{x < u^2} \frac{\mathbb{E}\left[\left(V_{I,n} - \sigma^2 - u^2\right)_+^2\right]}{\left(u^2 - x\right)_+^2} \,. \tag{B.21}$$

Let $b = \frac{(B - \underline{B})^2}{2}$ and $v = \frac{(B - \underline{B})^2}{2}\sigma^2$. Straightforward calculations show that $\frac{1}{2}\left(X_{2i} - X_{2i-1}\right)^2$ is almost surely bounded by b and its variance by v^2. Theorem 1.1 as well as an asymptotic argument in the spirit of Theorem 1.3 in Bentkus (2004) shows that:

$$\inf_{x < u^2} \frac{\mathbb{E}\left[\left(V_{I,n} - \sigma^2 - u^2\right)_+^2\right]}{\left(u^2 - x\right)_+^2} \le c\left(1 - \Phi\left(\frac{\sqrt{2\lfloor n/2 \rfloor}u}{(B - \underline{B})\,\sigma}\right)\right) \,, \tag{B.22}$$

where $c = \frac{e^2}{2} \approx 3.7$ and Φ the cumulative distribution function of the standard Gaussian distribution.

Self-bounding inequality on σ Combining the above results gives

$$\Pr\left(\widehat{\sigma}_t^2 - \sigma^2 \ge u^2\right) \le c\left(1 - \Phi\left(\frac{\sqrt{2\lfloor n/2 \rfloor}u}{(B - \underline{B})\sigma}\right)\right) \,. \tag{B.23}$$

Let $\delta \in (0,1)$ and $q = \Phi^{-1}\left(1 - \frac{\delta}{c}\right) \frac{(B-\underline{B})}{2\sqrt{2\lfloor n/2 \rfloor}}$. Equating the right-hand side to δ yields $\Pr\left(\widehat{\sigma}_n^2 - \sigma^2 \geq 2q\sigma\right) \leq \delta$. The discriminant of the polynomial $P(X) = X^2 + 2qX - \widehat{\sigma}_t^2$ is $\Delta = 4q^2 + 4\widehat{\sigma}_n^2 > 0$. Standard deviation being nonnegative, we deduce that

$$\Pr\left(\sigma \leq -q + \sqrt{q^2 + \widehat{\sigma}_n^2}\right) \leq \delta\,. \tag{B.24}$$

Finally, the reverse inequality is obtained by considering the quantity $\sigma^2 - V_{I,n}$. After similar computations, this yields $\Pr\left(\sigma^2 - \widehat{\sigma}_n^2 \geq 2q\sigma\right) \leq \delta$.

Practical considerations for mean confidence intervals

For calculating the mean confidence intervals, we recommend the use of the recent method from Phan et al. (2021) for samples smaller than 100 points, else the hedged capital method from Waudby-Smith and Ramdas (2020) for computation speed.

B.3.4 Second-order sub-Gaussian distributions

A natural relaxation of the strict Gaussian hypothesis is to consider distributions that concentrate *at least* as fast as a normal distribution around their means. This can be expressed in terms of the *moment-generating function* of Y, motivating the definition of *R-sub-Gaussian* distributions:

$$\forall \lambda \in \mathbb{R},\ \mathbb{E}\left[e^{\lambda(Y-\mu)}\right] \leq e^{\frac{\lambda^2 R^2}{2}}\,, \tag{B.25}$$

with equality if and only if $Y \sim \mathcal{N}\left(\mu, R^2\right)$. This popular control of exponential moments is indeed related to the concentration of measure phenomenon via the Cramér-Chernoff inequality:

$$\Pr\left(\widehat{\mu}_n - R\sqrt{\frac{2}{n}\log 2/\delta} \leq \mu \leq \widehat{\mu}_n + R\sqrt{\frac{2}{n}\log 2/\delta}\right) \geq 1 - \delta\,, \tag{B.26}$$

which holds as soon as Y is R-sub-Gaussian. The parameter R is related to the standard deviation σ of Y by the inequality $\sigma \leq R$.

An interesting feature of this approach is that it does not assume the boundedness of Y. In particular, even if Y is indeed bounded in $[\underline{B}, B]$, one does not need to know the exact value of $\underline{B}$ and B, which may be unknown or crudely estimated. Instead, the Cramér-Chernoff method relies on the proxy variance R^2, inducing a tighter control if Y concentrates on a narrow region of its support. However, the knowledge of a tight parameter R is often inaccessible to the practitioner, similar to the Gaussian case where σ is unknown.

In order to replace R with an empirical estimate, in the spirit of the Student's bound (Equation B.13), we introduce the *second-order sub-Gaussian* hypothesis, which in essence assumes that Y^2 concentrates at least as fast as the square of a normal random variable:

$$\forall \lambda < \frac{1}{2R^2},\ \mathbb{E}\left[e^{\lambda(Y-\mu)^2}\right] \leq \frac{1}{\sqrt{1 - 2R^2\lambda}}\,, \tag{B.27}$$

with equality if and only if $\left(\frac{Y-\mu}{R}\right)^2$ follows a $\chi^2(1)$ distribution. As shown in Anonymous (2021), this assumption results in the following confidence interval for R:

$$\Pr\left(\frac{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(Y_i-\mu)^2}}{1+\sqrt{\frac{2}{n}\log 2/\delta}} \leq R \leq \frac{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(Y_i-\mu)^2}}{1-\sqrt{\frac{1}{n}\log 2/\delta}}\right) \geq 1-\delta\,. \tag{B.28}$$

In addition, under the assumptions that *(i)* Y is centered ($\mu = 0$) and *(ii)* Y is second-order sub-Gaussian with parameter equal to its standard deviation ($R = \sigma$), this confidence bound simplifies to

Second-order sub-Gaussian standard deviation bound

$$\Pr\left(\frac{\widetilde{\sigma}_n}{1+\sqrt{\frac{2}{n}\log 2/\delta}} \leq \sigma \leq \frac{\widetilde{\sigma}_n}{1-\sqrt{\frac{1}{n}\log 2/\delta}}\right) \geq 1-\delta\,, \tag{B.29}$$

where $\widetilde{\sigma}_n^2 = \frac{1}{n}\sum_{i=1}^{n} Y_i^2$ is the empirical estimator of $\sigma^2 = \mathbb{E}\left[Y^2\right]$. We interpret this bound as an analogue of Equation B.12 where the χ^2 quantile has been relaxed to explicit terms thanks to the condition B.27.

Examples of probability laws that satisfy the above assumption with $\mu = 0$ and $R = \sigma$ include the Gaussian distributions (by definition of $\chi^2(1)$) but also other centered, symmetric distributions such as uniform and triangular distributions (see Anonymous (2021)). Therefore, the second-order sub-Gaussian assumption extends the Gaussian hypothesis to a broader class of distributions while keeping sound statistical concentration properties. The confidence interval provided by Equation B.29 is slightly looser than in the Gaussian case (Equation B.12), but less prone to model misspecification when Y is not Gaussian.

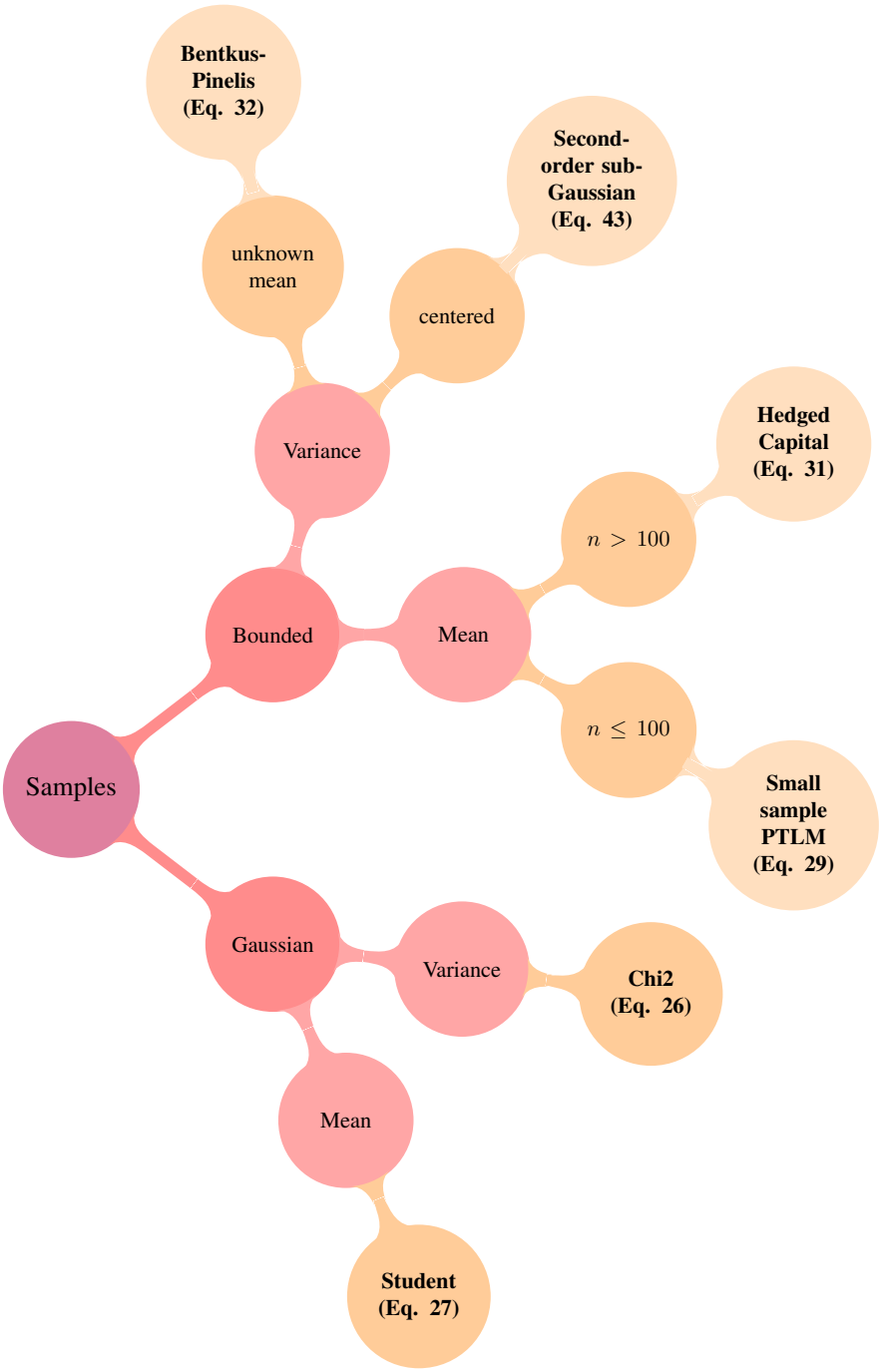


Figure B.1 Confidence interval decision flow, based on the target metric (mean, variance) and hypotheses (Gaussian or bounded, unknown mean or centered, sample size).

B.4 Interval disjunction algorithmic search

In this section, we present a complement to Section 3.2 for searching for a minimal risk level for confidence interval disjunction between the best crop management option and all other competing options for the mean-variance case. The approach provided is recommended after a great number of simulations (e.g. $n > 1000$) to minimize as much as possible the sampling error. As the total uncertainty strictly decreases with risk, a minimal value δ meeting the interval disjunction can be found with the bisection method, as presented in algorithm 3 to compare I crop-management options.

Algorithm 3 Bisection method for confidence interval disjunction risk level search.

Input: ε

/* the error tolerance to choose small, e.g. $\varepsilon = 10^{-6}$ */

Output: $A^\star, \delta$

/* the best crop management option, the risk level */

Data: $\mathcal{E}, \{\mathcal{Y}^i_{\text{sim}}\}_{i\in\mathcal{I}}$

/* the model error set and the simulated samples for the I options to compare */

$A^\star \leftarrow \emptyset$

$\underline{\delta} \leftarrow \varepsilon$ /* lower bound of the interval where searching δ */

$\overline{\delta} \leftarrow 0.5 - \varepsilon$ /* upper bound of the interval where searching δ */

while $\overline{\delta} - \underline{\delta} > \varepsilon$ **do**

  $\delta = (\overline{\delta} + \underline{\delta})/2$

  for $i \in \mathcal{I}$ **do**

    $\overline{\text{mv}}_{\text{Y}_{\text{true}}}{}^i \leftarrow \overline{\text{MV}}_{\text{Y}_{\text{true}}}(\frac{\delta}{2I}, \mathcal{E}, \mathcal{Y}^i_{\text{sim}})$

    $\underline{\text{mv}}_{\text{Y}_{\text{true}}}{}^i \leftarrow \underline{\text{MV}}_{\text{Y}_{\text{true}}}(\frac{\delta}{2I}, \mathcal{E}, \mathcal{Y}^i_{\text{sim}})$

  end

  /* see Section 3.2 */

  $j = \underset{\left(\underline{\text{mv}}_{\text{Y}_{\text{true}}}{}^i\right)_{i\in\mathcal{I}}}{\text{argmax}}$

  if $\underline{mv}_{Y_{true}}{}^j > \left(\overline{mv}_{Y_{true}}{}^i\right)_{i\in\mathcal{I}\setminus\{j\}}$ **then**

    $\overline{\delta} \leftarrow \delta$

    $A^\star \leftarrow j$

  else

    $\underline{\delta} \leftarrow \delta$

  end

end

return $A^\star, \delta$

B.5 Cultivar parameters in model simulations.

Table B.1 presents the parameters in DSSAT of the three cultivars used in (Joshi et al., 2017), respectively for periods: 1991 to 2000, 2001 to 2007, and 2008 to 2013.

Table B.1 Maize cultivar parametrization in DSSAT based on Joshi et al. (2017). See (Hoogenboom et al., 2019) for the detailed meaning of these coefficients.

index	ecotype	P1	P2	P5	G2	G3	PHINT
1	IB0001	165.0	0.660	740.0	800.0	9.20	40.00
2	IB0001	155.0	0.660	830.0	940.0	9.40	42.00
3	IB0001	155.0	0.650	750.0	930.0	9.40	42.00

B.6 Hypothesis testing

In this section, we apply various statistical tests on simulated yields and calibration residuals to assess the validity of our assumptions and demonstrate the interest of moving beyond the strict Gaussian hypothesis.

B.6.1 Simulated yields induced by weather generation

We first analyze the statistical properties of the simulated yield induced by weather generation, namely samples of the random variable Y_{sim}. In the interest of conciseness, we only report results for the distribution corresponding to planting date DOY 140.

Visual diagnostic

First, we perform an informal diagnostic of normality using a QQ plot, a visual comparison of the empirical quantiles of the standardized samples (centered and rescaled by the empirical mean and standard deviation respectively) against the quantile of the $\mathcal{N}(0,1)$ law. The results are reported in Figure B.2 and reveal a deviation to normality in the form of a lighter right tail.

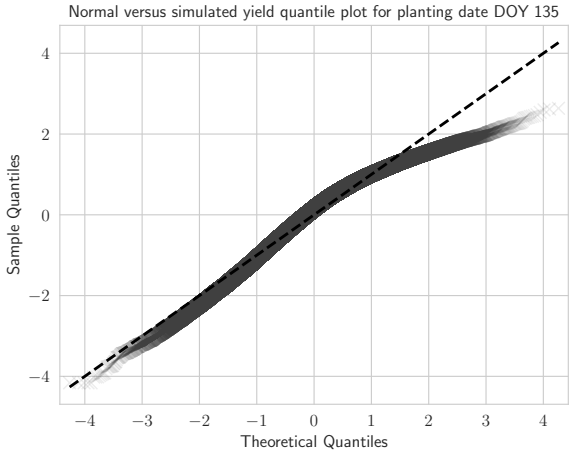


Figure B.2 Normal quantile/normalized empirical quantile plot for 10^5 yield simulations (planting date DOY 135).

Table B.2 Statistical tests for normality of simulated distribution of planting date DOY 135. #1: d'Agostino and Pearson (1973); d'Agostino (1971) ; #2: Jarque and Bera (1980) ; #3: Shapiro and Wilk (1965).

Test	Statistic value	pvalue	Conclusion
#1	328.5	0.0	Reject normality.
#2	340.7	0.0	Reject normality.
#3	0.98	0.0	Reject normality.

Statistical tests

We perform three standard tests of normality: Jarque-Bera (Jarque and Bera (1980)), Shapiro-Wilk (Shapiro and Wilk (1965)) and the d'Agostino omnibus test (d'Agostino and Pearson (1973)). The null hypothesis is that the sample's distribution is normal. As we have many samples ($n > 2000$), these tests are deemed efficient. We test samples from planting date DOY 135 with $n = 10000$. We refer to Table B.2 for the statistical and p-values of these tests. In accordance with the visual diagnostic of the QQ plot (see Figure B.2), all tests reject the normality hypothesis.

We stress that the result of these tests is somewhat obvious after the visual inspection of the QQ plot and we perform them as routine, standard assessments. We also note that in most applications, normality is not an end in itself but merely a means to justify the use of the subsequent method, such as using Student's statistics to build confidence intervals. In the following, we investigate whether such intervals hold nonetheless, if only numerically.

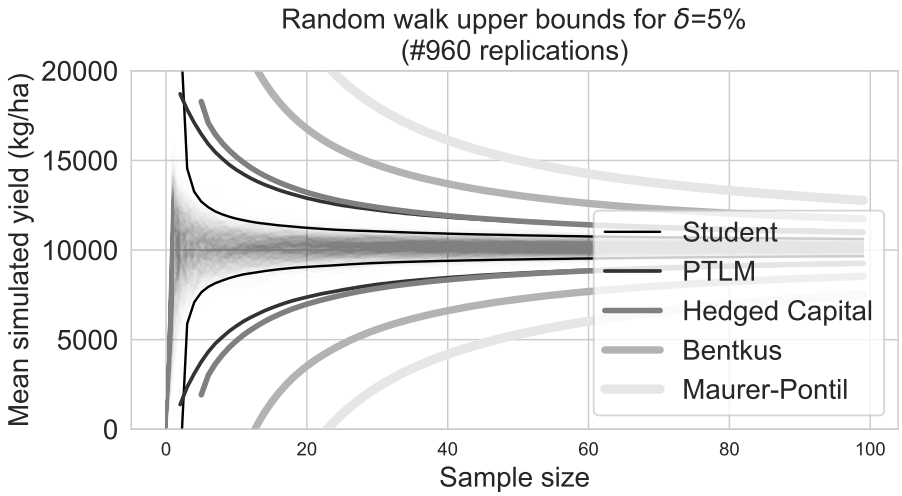
Boundary crossing

We compute Student's bound (see Appendix B.3.2), which is the tightest possible for Gaussian distributions, and compare it with standard confidence bounds for bounded distributions across multiple independent simulations (empirical Bernstein (Maurer and Pontil (2009)), empirical Bentkus (Kuchibhotla and Zheng (2021)), hedged capital (Waudby-Smith and Ramdas (2020)), small samples PTLM (Phan et al. (2021))). We show the results in Figure B.3a. By definition, if $\left[\underline{\mu_{Y_{\text{sim}}}}(\delta/2),\, \overline{\mu_{Y_{\text{sim}}}}(\delta/2)\right]$ is a confidence upper bound on $\mu_{Y_{\text{sim}}}$ at level δ, then $\mu_{Y_{\text{sim}}} > \overline{\mu_{Y_{\text{sim}}}}(\delta/2)$ or $\mu_{Y_{\text{sim}}} < \underline{\mu_{Y_{\text{sim}}}}(\delta/2)$ should not happen more than a fraction δ of the time. Figure B.3b shows that Student's bound regularly breaches the 5% threshold, thus empirically refuting the assumption that Gaussian confidence could apply to Y_{sim}. In other words, the Gaussian assumption is overly optimistic when applied to the outcome of the simulator.

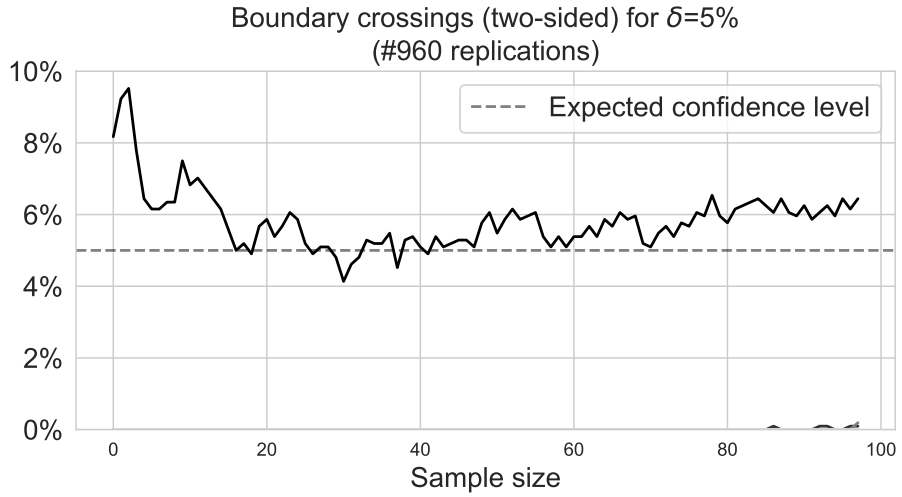
To go one step further, note that the union bound principle (see Appendix B.3.1) on which these two-sided intervals rely actually implies that lower and upper crossing should not occur individually more than a fraction $\delta/2$ of the time. We report these one-sided boundary crossing frequencies in Figure B.4. In accordance with the visual inspection, the simulated distribution is asymmetric with a lighter right tail, which translates to overly frequent crossings of the lower confidence bound.

B.6.2 Residuals

We now apply the same methodology to the calibration residuals, i.e samples of the error variable E. We recall that contrary to simulated yields, for which we can generate arbitrary large samples, these



(a) Standard confidence intervals $\left[\underline{\mu_{Y_{\text{sim}}}}(\delta/2),\ \overline{\mu_{Y_{\text{sim}}}}(\delta/2)\right]$ on $\mu_{Y_{\text{sim}}}$ for planting date DOY 135, $\delta = 5\%$. Student assumes Y_{sim} is normally distributed, while Maurer-Pontil, Bentkus, hedged capital and PTLM only assume $Y_{\text{sim}} \in [0, y_{\text{max}}]$. The dashed line represents the true mean $\mu_{Y_{\text{sim}}}$. Bounds are averaged over 1000 simulations.



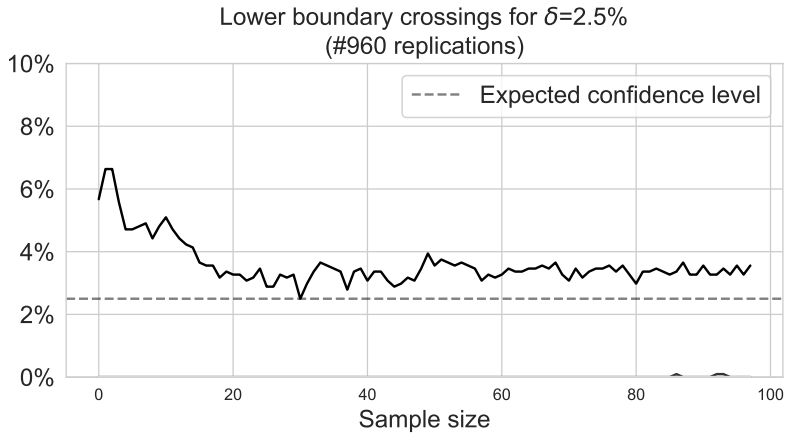
(b) Frequency of boundary crossings for planting date DOY 135 under Gaussian hypothesis, defined as the fraction of times $\mu_{Y_{\text{sim}}} > \overline{\mu_{Y_{\text{sim}}}}(\delta/2)$ or $\mu_{Y_{\text{sim}}} < \underline{\mu_{Y_{\text{sim}}}}(\delta/2)$ over 1000 simulations.

Figure B.3 Confidence intervals for planting date DOY 135.

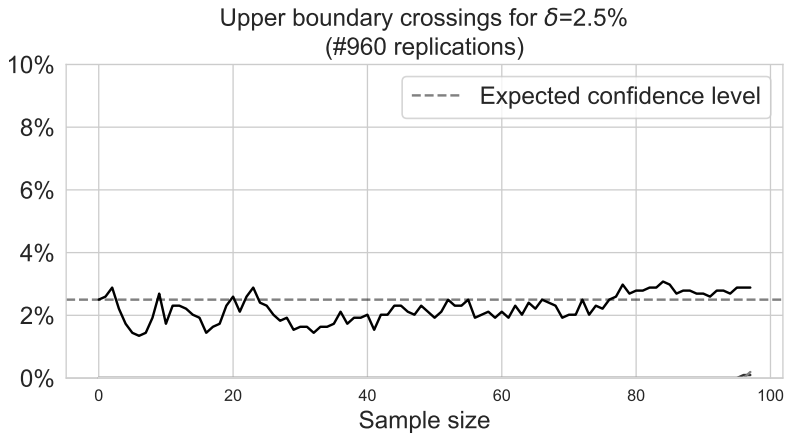
residuals are based on ground observations and are therefore available in limited number, typically around 20 or less points.

Visual diagnostic

Based on Figure B.5, error distribution seems to be centered and homoscedastic. Figure B.6 suggests non-normality, although the sample size is too small to conclude with confidence.



(a) Frequency of lower boundary crossing



(b) Frequency of upper boundary crossing

Figure B.4 Frequency of lower and upper boundary crossings for planting date DOY 135, defined as the fraction of times $\mu_{Y_{\text{sim}}} < \underline{\mu_{Y_{\text{sim}}}}(\delta/2)$ and $\mu_{Y_{\text{sim}}} > \overline{\mu_{Y_{\text{sim}}}}(\delta/2)$ respectively over 1000 simulations.

Statistical tests

For all tests, a total of $m = 21$ residuals are tested. The results are presented in Table B.3. If we consider a total risk level $\delta = 5\%$ for the five tests to hold simultaneously, a simple correction to allow multiple testing is to require each individual test to hold with a risk level $\delta' = \delta/5$ (union bound) that is to say $\delta' = 1\%$. With that consideration, we do not have enough evidence to reject the hypothesis that Y_{sim} and E are uncorrelated. However, note that at significance level $\delta = 5\%$, the same hypothesis would be rejected, although by a narrow margin (p-value is 0.0487), and indeed Figure B.7 hints at a slight negative correlation pattern. Given the small sample size, we consider this correlation pattern statistically indecisive and neglect it.

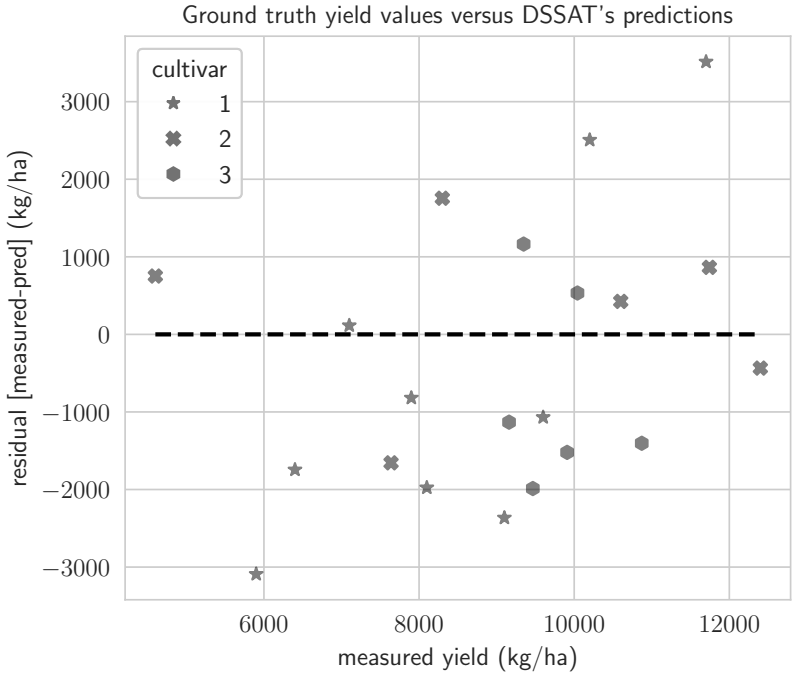


Figure B.5 Residual plot for 21 error observations.

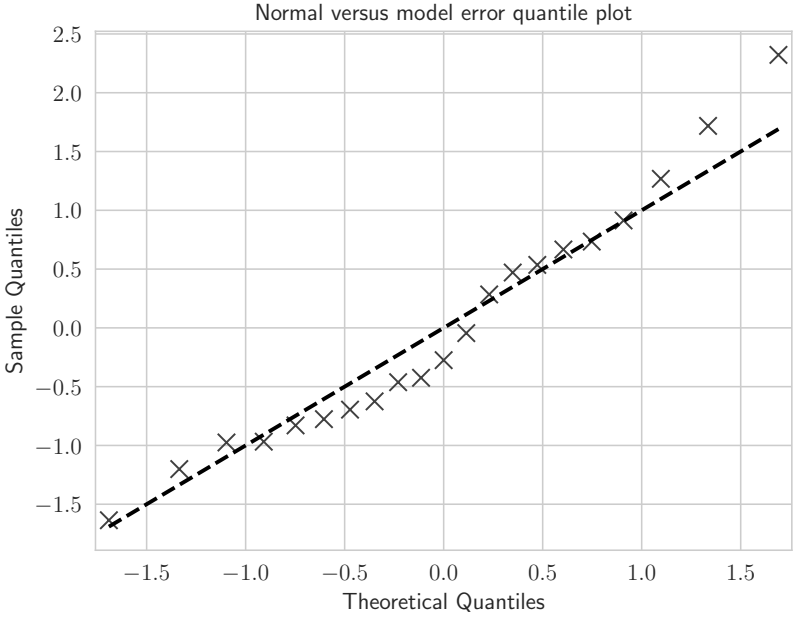


Figure B.6 Normal quantile/normalized empirical quantile plot for 21 error observations.

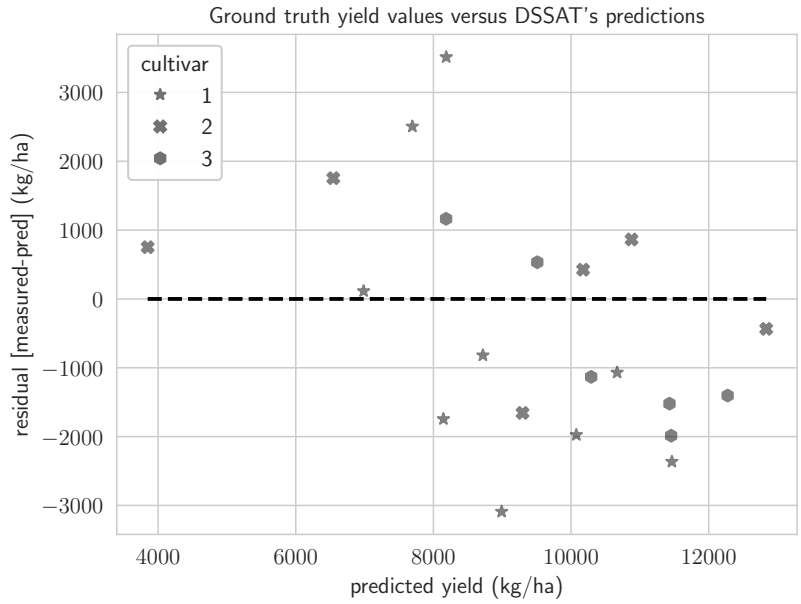


Figure B.7 Residuals against predicted yield values, 21 observations.

Table B.3 Statistical tests for model error distribution (significance level for multiple testing $\delta' = 1\%$). #1: Ljung and Box (1978), #2: Casella and Berger (2021) (Z-test), #3: Snedecor and Cochran (1989), #4: Kendall (1938).

Test	Null hypothesis (H_0)	Statistic value	p-value	Conclusion
#1	errors are serially uncorrelated	0.0826	0.7738	H_0 accepted
#2	$E \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma_E)$	-0.9665	0.3338	H_0 accepted
#3	(normal) errors are homoscedastic	0.1808	0.6707	H_0 accepted
#4	Y_{sim} and E are uncorrelated	-0.3142	0.0487	H_0 accepted

B.7 Minimal risk level for interval disjunction

B.7.1 Minimal risk level for confidence interval disjunction.

Risk-neutral decision criterion

Figure B.8 illustrates confidence intervals for minimal risk level $\delta = 0.02\%$, for the true mean yield criterion, between DOY 135 and 165. Y_{sim} was only assumed to be bounded in $[0, 20000]$ kg/ha.

Risk-aware decision criterion

This Section provides minimal risk level for interval disjunction between planting date DOY 135 and DOY 165, for the mean-variance criterion, considering all 21 error measures.

Centered Gaussian error hypothesis Figure B.8 illustrates confidences intervals for minimal risk level $\delta = 0.065\%$, for the true mean-variance yield criterion, between DOY 135 and 165, under centered Gaussian error hypothesis.

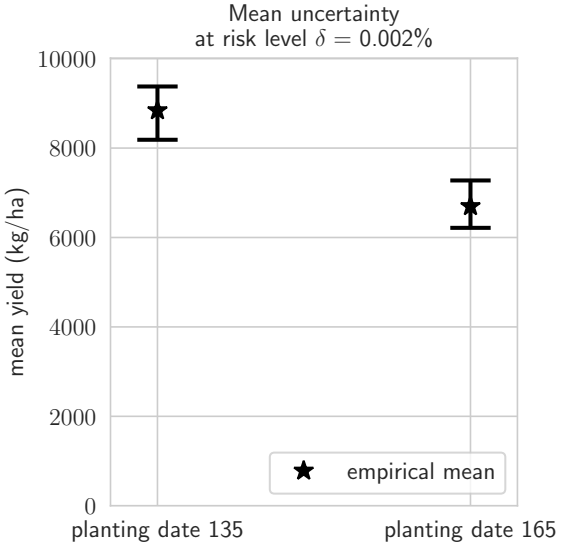


Figure B.8 Uncertainty of the mean criteria for ground truth distributions of planting date DOY 135 and 165 ($n = 500$ simulations).

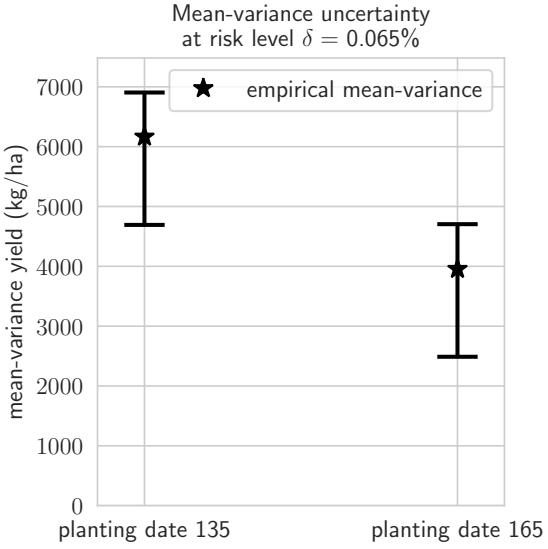


Figure B.9 Uncertainty for ground truth mean-variance ($\rho = 1$) for planting date DOY 135 and 165 ($n = 10000$ simulations). Gaussian error hypothesis

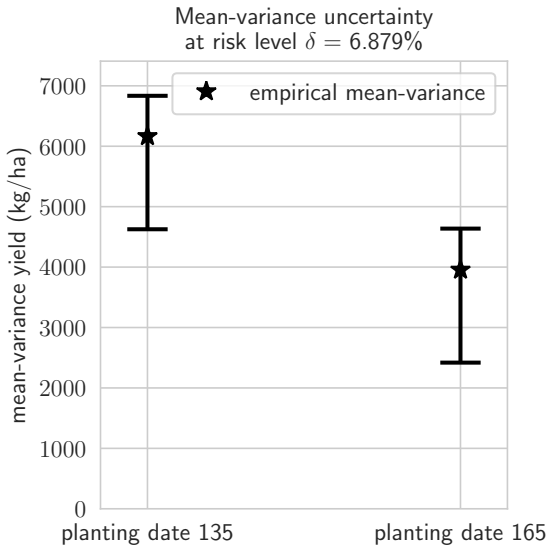


Figure B.10 Uncertainty for ground truth mean-variance ($\rho=1$) for planting date DOY 135 and 165 ($n=10000$ simulations). Second-order sub-Gaussian error hypothesis.

Centered second-order sub-Gaussian error hypothesis , Figure B.10 illustrates confidences intervals for minimal risk level $\delta = 6.878\%$, for the true mean-variance yield criterion, between DOY 135 and 165, under centered second-order sub-Gaussian error hypothesis.

B.7.2 Choice of the risk-aware metric

Despite the simplicity of its calculation, the MV criteron is not straightforward to interpret and parameterize (choice of ρ, see Eq. 3.1). An alternative risk aware metric is the conditional Value-at-Risk at level α denoted CVaR_α (Mandelbrot, 1997), that satisfies a number of desirable mathematical properties. For a (continuous) random variable X and $\alpha \in (0, 1]$:

$$\text{CVaR}_\alpha(X) := \mathbb{E}[X | X \leq \text{VaR}_\alpha(X)]\,, \tag{B.30}$$

where $\text{VaR}_\alpha(X)$ is the quantile of level α of X, also known as *Value-at-Risk*. The CVaR_α can be interpreted as the averaged worst α proportion of observable outcomes. As an example, for a grain yield distribution, the CVaR at level 20% would be the average of the 20% worst observable yields. When $\alpha \to 0^+$, only the worst realizations are considered in the average, which therefore emphasizes the most adverse outcomes; for $\alpha = 1$, the metric recovers the mean criterion. An application of the CVaR in an agricultural decision making context is presented in Baudry et al. (2021a). While being of interest, the sub-additivity of the CVaR rather than its additivity (Rockafellar et al., 2000) did not lend itself to the kind of analysis we envisioned in this study. For certain distributions however, $\text{CVaR}_\alpha(X)$ and MV_X^ρ capture similar tail risk behavior; for instance when X is normally distributed with mean μ_X and variance σ_X^2, $\text{CVaR}_\alpha(X) = \mu_X - \rho\sigma_X$, $\rho = \frac{1}{\alpha\sqrt{2\pi}} \exp\left(-\Phi^{-1}(\alpha)^2/2\right)$, where $\Phi^{-1}(\alpha)$ is the α-quantile of the standard normal distribution. Alternative risk measures could have been considered as well. A complete list is provided in Cassel et al. (2018), together with a discussion about their respective mathematical properties.

Supplementary Materials C

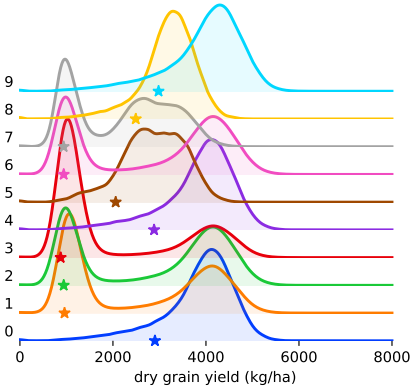
(corresponds to Chapter 4)

C.1 Maize simulations

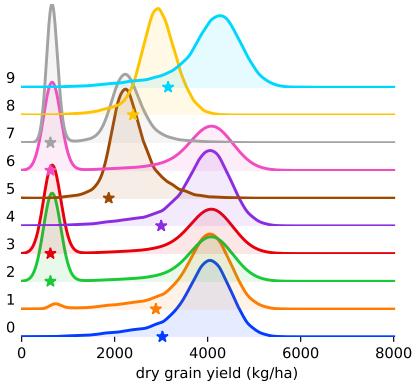
The cultivation scenarios were based on the the conditions found in Southern Mali. The soils came from Adam et al. (2020) who compiled and supplemented with survey data the soils found in the literature for the location of Koutiala, Mali. The data of Adam et al. (2020) included soils' depth, texture, water capacity, bulk density, organic matter content, pH and initial mineral nitrogen content. Soil characteristics and proportions in the population were summarized in Table 4.1, based on Adam et al. (2020). During the simulations, the weather times series were generated using the WGEN weather model (see Richardson and Wright, 1984; Soltani and Hoogenboom, 2003). WGEN had been calibrated on 40 year long historical daily weather records from a weather station located in N'Tarla found in Ripoche et al. (2015), which was located about 20 km from Koutiala ; these historical weather records were the best available. The cultivars used in the simulation and its parametrization in DSSAT are presented in Table C.1 ; this cultivars comes with DSSAT default data and was representative of the cultivars used in Mali. The cultivars were already calibrated based on experiments carried out in Mali. The simulations were initiated on Day Of Year (DOY) 140 and the planting is automatically performed in a window ranging from DOY 155 to 185 ; we specified the parameters of the automatic planting with Table C.2. For each soil, the initial soil nitrogen content was set according to the values found in Adam et al. (2020). The soil water content was set to crop lower limit, as a result of the end of the dry season at the usual planting dates. Because the simulations were initiated prior to planting date and because the weather was stochastically generated, the soil nitrogen mineral and water contents were uncertain at planting time. Each simulation was performed independently from the previous ones. At the beginning of the experiment, all the soils described in Table 4.1 were randomly distributed amongst the initial group of farmers following the proportions provided in Table 4.1. Figure C.1 shows the simulated yield distributions for `ITML840104` and `ITML840105` soils.

Table C.1 Maize cultivar parametrization in DSSAT

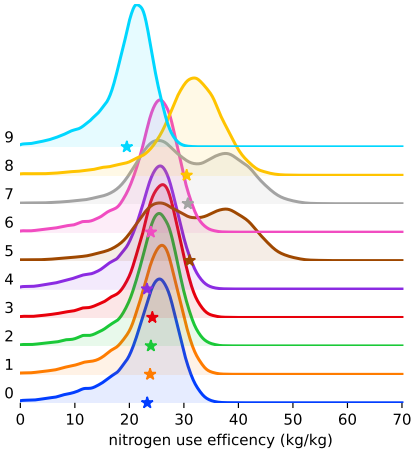
name	ecotype	P1	P2	P5	G2	G3	PHINT
Sotubaka	`IB0001`	300.0	0.520	930.0	500.0	6.00	38.90



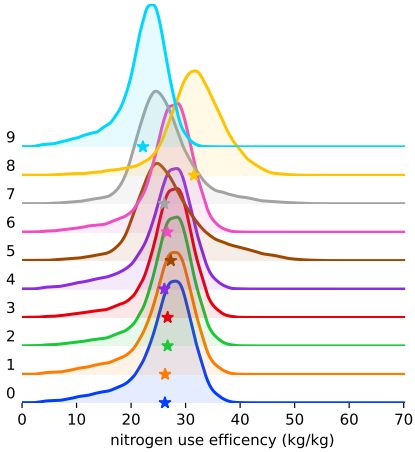
(a) Yield distributions for soil ITML840104. Stars represent the CVaR at level 30%.



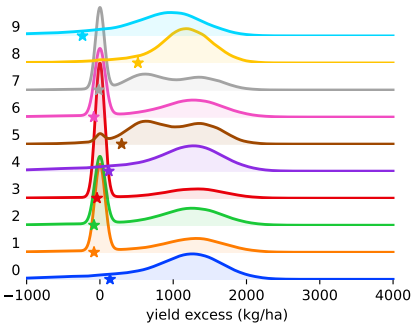
(b) Yield distributions for soil ITML840105. Stars represent the CVaR at level 30%.



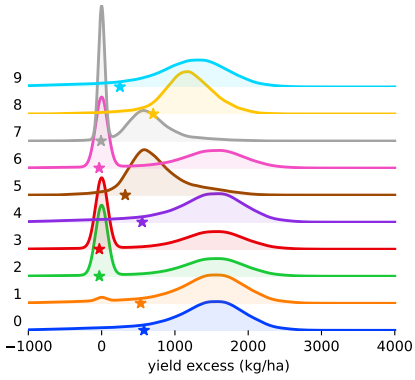
(c) Agronomic Nitrogen Efficiency (ANE) distributions for soil ITML840104. Stars represent the mean value.



(d) Agronomic Nitrogen Efficiency (ANE) distributions for soil ITML840105. Stars represent the mean value.



(e) Yield Excess (YE) distributions for soil ITML840104 with ANE_{ref}=15 kg grain/kg N. Stars represent the CVaR at level 30%.



(f) Yield Excess (YE) distributions for soil ITML840105 with ANE_{ref}=15 kg grain/kg N. Stars represent the CVaR at level 30%.

Figure C.1 Simulated impact of maize fertilizer practices on grain yield, Agronomic Nitrogen use Efficiency (ANE), Yield Excess (YE) for 10^5 hypothetical years using a weather generator. Maize cultivar was the same for all simulations. Practices indexes are indicated on the left-hand side of each sub-figure.

Table C.2 Automatic planting parametrization in DSSAT. PFRST: Starting date of the planting window; PLAST: End date of the planting window; PH2OL: Lower limit on soil moisture for automatic planting; PH2OU: Upper limit on soil moisture for automatic planting; PH2OD: Depth to which average soil moisture is determined for automatic planting; PSTMX: Maximum temperature of planting; PSTMN: Minimum temperature of planting.

PFRST (DOY)	155
PLAST (DOY)	185
PH2OL (%)	40
PH2OU (%)	100
PH2OD (cm)	30
PSTMX (°C)	40
PSTMN (°C)	10

C.2 Alternative performance measure of fertilization practices

We briefly discuss economical criteria we considered as performance indicators of fertilizer practices. A first indicator we considered was the gross margin. The cost of production of nitrogen fertilizer being indexed on the price of natural gas, it is subject to high volatility. As a consequence, an optimal practice is likely to be different each year and thus the decision problem would turn to be highly non-stationary. Such setting dramatically increases the complexity of the decision problem, and the chance of observing good identification performances are lowered.

Another economic measure could be the value:cost ratio (VCR), which is given for a fertilizer practice π as:

$$\text{VCR}^{\pi} = \frac{p_{\text{maize}}}{p_{\text{N}}} \times \frac{Y^{\pi} - Y^{0}}{\text{N}^{\pi}} \tag{C.1}$$

$$= \frac{p_{\text{maize}}}{p_{\text{N}}} \times \text{ANE}^{\pi} \tag{C.2}$$

where p_{N} is fertilizer unitary cost and p_{maize} unitary maize grain selling price. Remarking that each given year the ratio $\frac{p_{\text{maize}}}{p_{\text{N}}}$ is shared by all fertilizer practices. We neglect a possible quality consideration that could motivate a different maize selling price between the fertilizer practices, for instance a difference of protein content in maize grains. Then the decision problem is perfectly equivalent to choosing the fertilizer practice which maximizes the ANE. Thereby, the use of the cost:value ratio suffers from the same drawbacks as the ANE.

C.3 Algorithms

C.3.1 Details about `BCB`

In algorithm 4, we provide the detailed pseudo-code of `BCB` (`BCB`). As shown by Figure C.2, the higher the number of collected rewards, the less the weights sampled from Dirichlet distributions

exhibit variance. This variance directly relates to the noise introduced in the computation of the score of the different available actions.

Algorithm 4 BCB: identification strategy at cohort level (detailed)

Input: Level α, horizon T, K options, upper bounds $B_1, \ldots, B_K$, $\mathcal{F}^c$ the set of all farmers in the cohort

Init.: $\forall k \in \{1, ..., K\}$: $\mathcal{X}_k = \{B_k\}$, $N_k = 0$; $\mathcal{F}_1^c = \{f_1, \cdots, f_{n_1}\}$; $t = 1$; $\mathcal{A}_1 = \{\emptyset\}$

// Beginning of first season

for $f \in \mathcal{F}_1^c$ **do**

  Randomly assign a crop management option $a \in \{1, \ldots, K\}$ to the farmer f

  $\mathcal{A}_1 = \mathcal{A}_1 \cup \{a\}$

end

// End of first season

for $(a, f) \in (\mathcal{A}_1, \mathcal{F}_1^c)$ **do**

  Receive the result of the option a from farmer f: $r_{f,a}$

  Update $\mathcal{X}_a = \mathcal{X}_a \cup \{r_{f,a}\}, N_a = N_a + 1$

end

for $t \in \{2, \ldots, T\}$ **do**

  // Beginning of season t

  Get $\mathcal{F}_t^c = \{f_1, \cdots, f_{n_t}\}$; // the set of farmers of the same cohort to provide recommendations

  for $k \in \{1, \ldots, K\}$ **do**

    Update the empirical CVaR of action k: $\hat{c}_{k,t-1} = \widehat{C}_\alpha(\mathcal{X}_k)$

  end

  for $f \in \mathcal{F}_t^c$ **do**

    Update the empirical regret of farmer f: $l_{f,t-1} = \widehat{R}_f^\alpha(t-1)$

  end

  $\mathcal{A}_t = \{\emptyset\}$; // the set of recommendations to provide to the farmers

  for $f \in \mathcal{F}_t^c$ **do**

    for $k \in \{1, \ldots, K\}$ **do**

      Draw $\omega_k = \{w_1, \cdots, w_{N_k}\} \sim \mathcal{D}_{N_k}$; // Dirichlet of concentration parameter $\underbrace{(1, \cdots, 1)}_{N_k \text{ times}}$

      Search j the maximum index such that $\sum_{i=1}^{j} w_i \leq \alpha$

      Sort $\mathcal{X}_k$ in increasing order

      Compute $\tilde{c}_k = x_j - \frac{1}{\alpha} \sum_{i=1}^{N_k} w_i \max(x_j - x_i, 0)$; // assign a score to action k

    end

    $a = \text{argmax}_{k \in \{1, \ldots, K\}} \tilde{c}_k$

    $\mathcal{A}_t = \mathcal{A}_t \cup \{a\}$

  end

  Sort the set of farmers $\mathcal{F}_t^c$ according their increasing empirical regrets $l_{f,t-1}$

  Sort the set of actions $\mathcal{A}_t$ according their increasing empirical CVaR $\hat{c}_{k,t-1}$

  for $(a, f) \in (\mathcal{A}_t, \mathcal{F}_t^c)$ **do**

    Assign action a to farmer f ; // fair exploration

  end

  // End of season t

  for $(a, f) \in (\mathcal{A}_t, \mathcal{F}_t^c)$ **do**

    Receive result of action a from farmer f: $r_{f,a}$

    Update $\mathcal{X}_a = \mathcal{X}_a \cup \{r_{f,a}\}, N_a = N_a + 1$

  end

end

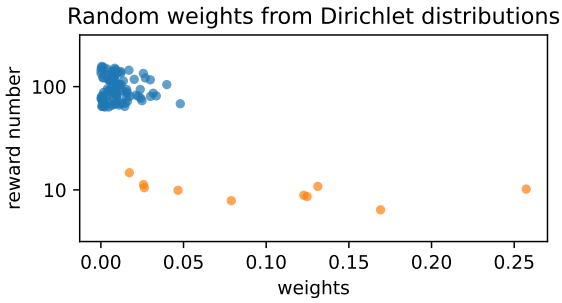


Figure C.2 Examples of weights sampled from Dirichlet distributions during BCB execution, respectively for 10 and 100 rewards. The greater the number of rewards, the less variance the weights show. The variance of weights is related to the noise level in the computation of the empirical CVaR of BCB.

Remark C.3.1 (First season)**.** *Algorithm 4 is well defined for the first season as without data all CVaRs will be equal to the maximum observable result, making the algorithm choose each option arbitrarily at random. On average, each option will be equally explored. Note that we could replace this step by an equi-proportional exploration step (similar to Explore-Then-Commit, see C.3.2) without changing the theoretical properties of our algorithm. Furthermore, the decision maker could also include any additional results collected before the experiment (if the practices has already been tested for some time) in the initialization of the algorithm.*

C.3.2 Explore-Then-Commit (ETC)

We provide the pseudo-code of the Explore-Then-Commit (ETC) strategy with algorithm 5. The noise introduced by random weights and the presence of the maximum observable results in the histories manage the exploration/exploitation dilemma. BCB will favor fertilizer practices with higher CVaR compared to the others. But, the algorithm will still prevent the under-exploration of fertilizer practices by choosing them with a proper probability, even if e.g. poor YE have been observed due to rare unfavorable weather events. Indeed, with the extra randomness introduced by the random weighting of rewards, poor rewards may be re-weighted by smaller weights compared to higher rewards, yielding a good score. The amount of noise introduced by the random weights sampled from the Dirichlet distribution is related to variance of these random weights. The greater the number of rewards, the lesser the variance and consequently the lesser the noise (Figure C.2). Thereby, the more a fertilizer practice was tried by the algorithm, the closer its score gets to the true CVaR of rewards. The presence of the maximum observable YE acts as an “optimistic bonus" in the computation of the scores, encouraging exploration even for sub-optimal practices, as it raises up their initial values when few rewards have been observed.

Algorithm 5 ETC: identification strategy at cohort level

Input: Level α, horizon T, K options, $\mathcal{F}^c$ the set of all farmers in the cohort, t_{trials} the number of years of trials

Init.: $\forall k \in \{1, \cdots, K\} : N_k = 0$

// Do trials during t_{trials} years

for $t \in \{1, \cdots, t_{trials}\}$ **do**

  // Beginning of the season t

  Get $\mathcal{F}_t^c = \{f_1, \cdots, f_{n_t}\}$; // get the farmers willing to participate

  $\mathcal{A}_t = \{\emptyset\}$

    Fill $\mathcal{A}_t$ by uniformly distributing the K options to the farmers in $\mathcal{F}_t^c$

    // End of the season t

  for $(a, f) \in (\mathcal{A}_t, \mathcal{F}_t^c)$ **do**

    Receive the result of the option a from farmer f: $r_{f,a}$

    Update $\mathcal{X}_a = \mathcal{X}_a \cup \{r_{f,a}\}, N_a = N_a + 1$

  end

end

for $k \in \{1, \ldots, K\}$ **do**

  Compute the empirical CVaR of action k: $\hat{c}_{k,t-1} = C_\alpha(\mathcal{X}_k)$

end

$a_{\max} = \text{argmax}_{k \in \{1,\ldots,K\}} \hat{c}_k$; // get the action that best performed during trials

// After trial phase, always recommend the action that best performed during trials

for $t \in \{t_{trials} + 1, \cdots, T\}$ **do**

  // Beginning of the season t

  Get $\mathcal{F}_t^c = \{f_1, \cdots, f_{n_t}\}$

  for $f \in \mathcal{F}_1^c$ **do**

    Assign option $a_{\max}$ to the farmer f

  end

  // End of the season t

  for $f \in \mathcal{F}_t^c$ **do**

    Receive the result of the option $a_{\max}$ from farmer f: $r_{f,a_{\max}}$

    Update $\mathcal{X}_{a_{\max}} = \mathcal{X}_{a_{\max}} \cup \{r_{f,a_{\max}}\}, N_{a_{\max}} = N_{a_{\max}} + 1$

  end

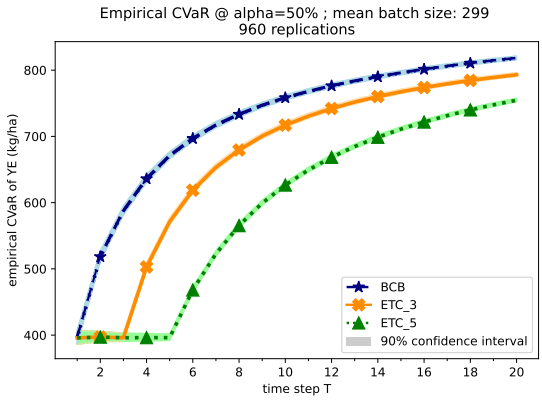
end

C.4 Experiment complements

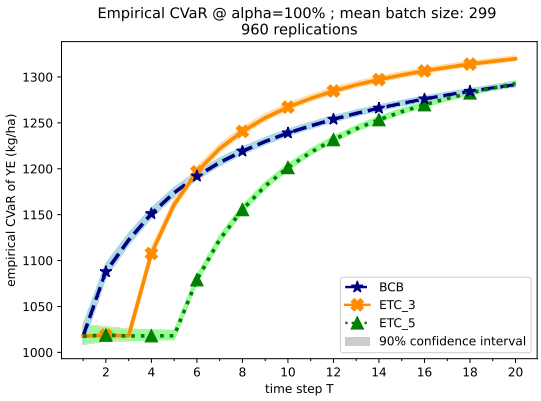
Following methods of Section 4.2 of the main text, we provide identification performances of identification strategies for CVaR levels $\alpha = 50\%$ and $\alpha = 100\%$ with Figures C.3, C.4 and C.5. For both CVaR levels, the YE is defined with $\text{ANE}_{\text{ref}} = 15$ kg N/kg grain.

C.5 Theoretical Analysis

This section is devoted to the theoretical analysis of the BCB algorithm. We will mostly adapt the analysis of Baudry et al. (2021a), and show that the problem of learning with batched data of finite

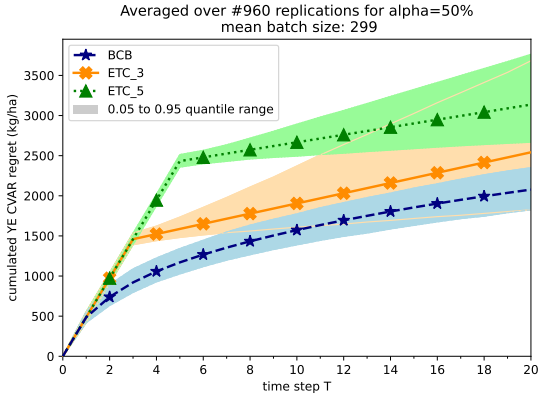


(a) $\alpha = 50\%$

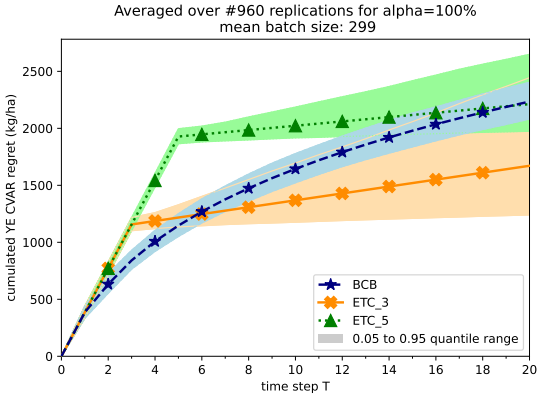


(b) $\alpha = 100\%$

Figure C.3 Farmers' empirical CVaR at level of all YE received between $T = 0$ and the considered T.

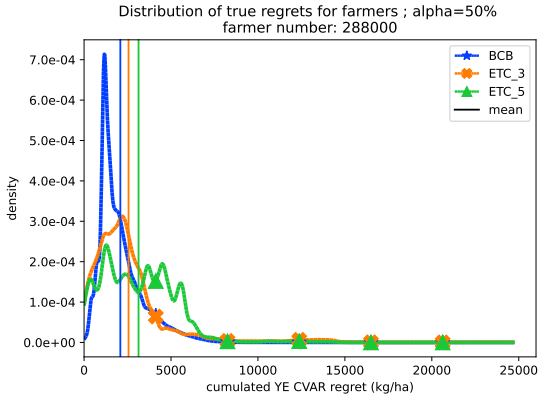


(a) $\alpha = 50\%$

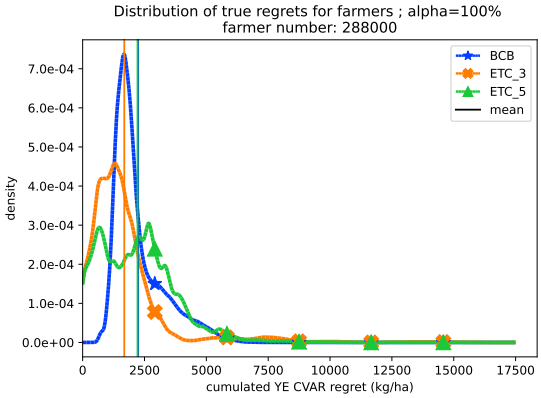


(b) $\alpha = 100\%$

Figure C.4 Cumulated regret averaged over the population for the CVaR at level of YE.



(a) $\alpha = 50\%$



(b) $\alpha = 100\%$

Figure C.5 Distribution of individual cumulated regret after $T = 20$.

upper bounded size is no harder than the pure online learning problem considered in the original paper.

Theorem C.5.1 (α-CVaR Regret of BCB)**.** *Consider a bandit problem $(F_1, \dots, F_K) \in \mathcal{F}^K$, with respective CVaR$_\alpha$ denoted by $(c_1, \dots, c_K)$ with $c_1 = argmax_{k=1,\dots,K} c_k$. Assume that BCB runs for T seasons, and that at each season the size of the batch is $n_T \leq F \in \mathbb{N}$. Then, for any $\varepsilon > 0$ small enough there exists some $\varepsilon_1 > 0, \varepsilon_2 > 0$ such that the regret of BCB satisfies*

$$\mathcal{R}_T^\alpha \leq \sum_{k=2}^{K} \Delta_k^\alpha \left(m_T^k + F + 2F \frac{e^{-2m_T^k \varepsilon_1^2}}{1 - e^{-2\varepsilon_1^1}} + C_{1,\varepsilon_2}^\alpha \right) ,$$

where $m_T^k = \frac{\log(T)+\log(F)}{\mathcal{K}_{\inf}^{\alpha,\mathcal{D}}(F_k,c_1)-\varepsilon}$ and C_{1,ε_2} is a constant depending only on the distribution F_1, the family $\mathcal{F}$ and ε_2.

It is interesting to compare this regret upper bound to the one obtained in the purely sequential setting, that we recall in Theorem C.5.2.

Theorem C.5.2 (α-CVaR Regret of B-CVTS with time horizon S_T (adapted from Theorem 3 in Baudry et al. (2021a)))**.** *Consider a bandit problem $(F_1, \dots, F_K) \in \mathcal{F}^K$, with respective CVaR$_\alpha$ denoted by $(c_1, \dots, c_K)$ with $c_1 = argmax_K c_k$. Consider a number of data collected S_T. Then, for any $\varepsilon > 0$ small enough there exists some $\varepsilon_1 > 0, \varepsilon_2 > 0$ such that the CVaR-regret of B-CVTS satisfies*

$$\mathcal{R}_T^\alpha \leq \sum_{k=2}^{K} \Delta_k^\alpha \left(n_{S_T}^k + 2 \frac{e^{-2n_{S_T}^k \varepsilon_1^2}}{1 - e^{-2\varepsilon_1^1}} + C_{1,\varepsilon_2}^\alpha \right) ,$$

where $m_{S_T}^k = \frac{\log(S_T)}{\mathcal{K}_{\inf}^{\alpha,\mathcal{D}}(F_k,c_1)-\varepsilon}$ and C_{1,ε_2} is a constant depending only on the distribution F_1, the family $\mathcal{F}$ and ε_2.

First, we see that if F is indeed a constant (i.e do not depend on the time) then when T is large enough then F has not impact on the scaling of the regret. In our proof the main impact of the batch setting is an additive term F for each arm, hence the regret becomes close to the one of the sequential setting once $m_T^k \gg F$. Finally, if the number of farmers in each batch is exactly F at each step then $S_T = FT$ and, $m_T^k = n_{S_T}^K$, hence the asymptotically dominant (logarithmic) term is the same in the two settings.

These theoretical results show that learning with batch feedback does not introduce theoretical limitations in our setting, and so the BCB algorithm is theoretically grounded.

Proof of Theorem C.5.1. As in the proof of Baudry et al. (2021a) we will decompose the expected number of pulls of each sub-optimal arm inside the cohort according to several possible events, corresponding to "good" scenarios (the empirical distributions accurately reflect the true distributions) and "bad" ones (the empirical distributions give a wrong idea of the true performance of some arms) for the trajectory of the bandit algorithms. We denote by T the number of seasons in the experiments and n_t the number of farmers at each season t for this cohort, and by F the total number of farmers available for the experiment. Then, the expected number of pulls of arm k during the total duration of

the experiment inside the cohort is

$$\mathbb{E}[N_k(T)] = \mathbb{E}\left[\sum_{t=1}^{T}\sum_{f=1}^{n_t}\mathbb{1}(A_{t,f}=k)\right] ,$$

where $A_{t,f}$ denotes the recommendation to farmer f at season t.

The first step of the proof consists in considering the number of pulls of k when its sample size is larger (resp. smaller) than some fixed threshold m_T, that we will specify later.

$$\begin{aligned}\mathbb{E}[N_k(T)] &= \mathbb{E}\left[\sum_{t=1}^{T}\sum_{f=1}^{n_t}\mathbb{1}(A_{t,f}=k)\right] \\ &\leq \mathbb{E}\left[\sum_{t=1}^{T}\sum_{f=1}^{n_t}\mathbb{1}(A_{t,f}=k, N_k(t-1)\leq m_T)\right] + \mathbb{E}\left[\sum_{t=1}^{T}\sum_{f=1}^{n_T}\mathbb{1}(A_{t,f}=k, N_k(t-1)\geq m_T)\right]\end{aligned}$$

We now consider the first term and introduce the random variable $\tau = \{\sup_{t\leq T} : N_k(t-1)\leq m_T\}$. By construction, τ is the last season for which the total number of observations for arm k is smaller than m_T. Using the basic properties of τ we obtain that

$$\begin{aligned}\sum_{t=1}^{T}\sum_{f=1}^{n_t}\mathbb{1}(A_{t,f}=k, N_k(t-1)\leq m_T) &\leq \sum_{t=1}^{\tau}\sum_{f=1}^{n_t}\mathbb{1}(A_{t,f}=k, N_k(t-1)\leq m_T) + \sum_{t=\tau+1}^{T}\sum_{f=1}^{n_t}\mathbb{1}(A_{t,f}=k, N_k(t-1)\leq m_T) \\ &\leq N_k(\tau) + \sum_{f=1}^{n_{\tau+1}}\mathbb{1}(A_{\tau,f}=k) \\ &\leq m_T + F\end{aligned}$$

As this result does not depend on the value of τ, we can then obtain

$$\mathbb{E}[N_k(T)] \leq m_T + F + \underbrace{\mathbb{E}\left[\sum_{t=1}^{T}\sum_{f=1}^{n_t}\mathbb{1}(A_{t,f}=k, N_k(t-1)\geq m_T)\right]}_{A} .$$

At this step, the only difference with the purely sequential bandit problem is the additional F. We now consider the term A, that we further analyze according to three events: (1) the empirical distribution of arm k is not close to its true distribution, (2) the empirical distribution of arm k is close to its true distribution but the "noisy" CVaR computed for arm k over-estimates its true CVaR, and (3) the "noisy" CVaR computed for the optimal arm 1 under-estimates its true CVaR. Classically in bandit analysis, we decompose the number of pulls of arm k according to these three events, as at least one of them must be true when $A_{t,f}=k$ holds, that is

$$\{A_t = k\} \subset \{F_{k,t-1}\notin\mathcal{B}_{\varepsilon_1}(F_k)\}\cup\{F_{k,t-1}\in\mathcal{B}_{\varepsilon_1}(F_k), \widetilde{c}_{k,t,f}\geq c_1-\varepsilon_2\}\cup\{\widetilde{c}_{1,t,f}\leq c_1-\varepsilon_2\} ,$$

where $\mathcal{B}_{\varepsilon_1}(F_k)$ is an ε_1-Levy ball around F_k, and $\varepsilon_1, \varepsilon_2$ are two small positive constants. This leads to

$$
\begin{aligned}
A \leq & \underbrace{\mathbb{E}\left[\sum_{t=1}^{T}\sum_{f=1}^{n_t} \mathbb{1}(A_{t,f}=k, N_k(t-1)\geq m_T, F_{k,t-1}\notin \mathcal{B}_{\varepsilon_1}(F_k))\right]}_{A_1} \\
& + \underbrace{\mathbb{E}\left[\sum_{t=1}^{T}\sum_{f=1}^{n_t} \mathbb{1}(A_{t,f}=k, N_k(t-1)\geq m_T, F_{k,t-1}\in \mathcal{B}_{\varepsilon_1}(F_k), \widetilde{c}_{k,t,f}\geq c_1-\varepsilon_2)\right]}_{A_2} \\
& + \underbrace{\mathbb{E}\left[\sum_{t=1}^{T}\sum_{f=1}^{n_t} \mathbb{1}(A_{t,f}=k, N_k(t-1)\geq m_T, \widetilde{c}_{1,t,f}\leq c_1-\varepsilon_2)\right]}_{A_3} .
\end{aligned}
$$

Upper bounding A_2 Denoting by $\widehat{F}_{k,n}$ the empirical distribution of arm k after a total number of *pulls* n (instead of after season t), we obtain

$$
\begin{aligned}
A_1 &\coloneqq \mathbb{E}\left[\sum_{t=1}^{T}\sum_{f=1}^{n_t} \mathbb{1}(A_{t,f}=k, N_k(t-1)\geq m_T, F_{k,t-1}\notin \mathcal{B}_{\varepsilon_1}(F_k))\right] \\
&\leq \mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}(N_k(t-1)\geq m_T, F_{k,t-1}\notin \mathcal{B}_{\varepsilon_1}(F_k))\sum_{f=1}^{n_t}\mathbb{1}(A_{t,f}=k)\right] \\
&\leq \mathbb{E}\left[\sum_{t=1}^{T}\sum_{n=m_T}^{T} \mathbb{1}(N_k(t-1)=n, F_{k,t-1}\notin \mathcal{B}_{\varepsilon_1}(F_k))\sum_{f=1}^{n_t}\mathbb{1}(A_{t,f}=k)\right] ,
\end{aligned}
$$

with a union bound on the number of pulls. Under $N_k(t-1)=n$ it holds that $F_{k,t-1}=\widehat{F}_{k,n}$, and so we can further write that

$$
\begin{aligned}
A_1 &\leq \mathbb{E}\left[\sum_{t=1}^{T}\sum_{n=m_T}^{T} \mathbb{1}(N_k(t-1)=n, \widehat{F}_{k,n} \notin \mathcal{B}_{\varepsilon_1}(F_k)) \sum_{f=1}^{n_t} \mathbb{1}(A_{t,f}=k)\right] \\
&\leq \mathbb{E}\left[\sum_{n=m_T}^{T} \mathbb{1}(\widehat{F}_{k,n} \notin \mathcal{B}_{\varepsilon_1}(F_k)) \sum_{t=1}^{T}\sum_{f=1}^{n_t} \mathbb{1}(A_{t,f}=k, N_k(t-1)=n)\right] \\
&\leq F\mathbb{E}\left[\sum_{n=m_T}^{T} \mathbb{1}(\widehat{F}_{k,n} \notin \mathcal{B}_{\varepsilon_1}(F_k))\right] \\
&= F\sum_{n=m_T}^{+\infty} \Pr(F_{k,n} \notin \mathcal{B}_{\varepsilon_1}(F_k))
\end{aligned}
$$

Finally, using the Dvoretzky–Kiefer–Wolfowitz inequality (Massart, 1990) we obtain

$$
\begin{aligned}
&\leq F\sum_{n=m_T}^{+\infty} 2e^{-2n\,\varepsilon_1^2} \\
&\leq \frac{2Fe^{-2m_T\,\varepsilon_1^2}}{1-e^{-2\,\varepsilon_1^2}}\,.
\end{aligned}
$$

This upper bound holds for any choice of m_T, ε_1, and we remark that if $m_T \to +\infty$ then $A_1 \to 0$.

Upper bounding A_2 The term A_2 is then handled with similar tricks, and the arguments used in Baudry et al. (2021a).

$$
\begin{aligned}
A_2 &\coloneqq \mathbb{E}\left[\sum_{t=1}^{T}\sum_{f=1}^{n_t} \mathbb{1}(A_{t,f}=k, N_k(t-1)\geq m_T, F_{k,t-1}\in\mathcal{B}_{\varepsilon_1}(F_k), \widetilde{c}_{k,t,f}\geq c_1-\varepsilon_2)\right] \\
&\leq \mathbb{E}\left[\sum_{t=1}^{T}\sum_{f=1}^{F} \mathbb{1}(N_k(t-1)\geq m_T, F_{k,t-1}\in\mathcal{B}_{\varepsilon_1}(F_k)) \times \Pr\left(\widetilde{c}_{k,t,f}\geq c_1-\varepsilon_2\,|\mathcal{F}_t\right)\right]\,,
\end{aligned}
$$

where $\mathcal{F}_t$ is the canonical filtration, so the probability is obtained conditioning on the data observed before the beginning of the round. Using the the continuity of $\mathcal{K}_{\inf}^{\alpha,\mathcal{D}}$ in its two arguments as proved in Agrawal et al. (2021), we obtain that for any $\varepsilon > 0$ small enough there exist some $\varepsilon_1, \varepsilon_2$ such that

$$
\begin{aligned}
A_2 &\leq \mathbb{E}\left[\sum_{t=1}^{T}\sum_{f=1}^{F} \mathbb{1}(A_{t,f}=k, N_k(t-1)=n, F_{k,t-1}\in\mathcal{B}_{\varepsilon_1}(F_k)) e^{-m_T\left(\mathcal{K}_{\inf}^{\alpha,\mathcal{D}}(F_k,c_1)-\varepsilon\right)}\right] \\
&\leq F\times T\times e^{-m_T\left(\mathcal{K}_{\inf}^{\alpha,\mathcal{D}}(F_k,c_1)-\varepsilon\right)}\,.
\end{aligned}
$$

As we did not specify the choice of $\varepsilon_1, \varepsilon_2$ already we simply require them to be small enough to satisfy this condition. Then, we can calibrate m_T as

$$m_T = \frac{\log(T) + \log(F)}{\mathcal{K}_{\inf}^{\alpha,\mathcal{D}}(F_k, c_1) - \varepsilon} ,$$

Furthermore, with this choice m_T will become the main term in the regret upper bound when T becomes large enough.

Upper bounding A_3 The final term is the one that leading to the most complicated part of the analysis in Baudry et al. (2021a). Fortunately, the batch setting will have no impact on this part, so we can directly reuse the results provided in this paper.

Indeed, we can re-write A_3 to make it equivalent to the corresponding term in the purely sequential problem:

$$A_3 = \mathbb{E}\left[\sum_{t=1}^{T}\sum_{f=1}^{n_t} \mathbb{1}(\widetilde{c}_{1,t,f} \leq c_1 - \varepsilon_2)\right] = \mathbb{E}\left[\sum_{r=1}^{S_T} \mathbb{1}(\widetilde{c}_1(r) \leq c_1 - \varepsilon_2)\right] ,$$

where in the second term we count the number of recommendations provided by the algorithm, assigning those in the same batch an arbitrary order, $\widetilde{c}_1(r)$ is then the noisy CVaR computed for arm 1 for this specific round. Furthermore, we write $S_T = \sum_{t=1}^{T} n_t \leq FT$. In Baudry et al. (2021a), the authors obtain a constant upper bound for this term, depending only on ε_2 (and the upper bound of the support), and in particular not depending on the exact number of plays. We conclude that there exists some constant C_{1,ε_2} satisfying

$$A_3 \leq C_{1,\varepsilon_2} .$$

This result concludes our proof, and we refer the interested reader to the original paper for a complete proof and a detailed expression for C_{1,ε_2}. We further remark that contrarily to the previous terms, the upper bound of A_3 does not depend on F at all. □

Résumé opérationnel

Introduction

L'apprentissage par renforcement (AR) est une méthode d'apprentissage automatique, et plus largement d'intelligence artificielle, dans laquelle un agent apprend à contrôler un système dynamique et incertain (Sutton and Barto, 2018). Séquentiellement, l'agent effectue des actions et cherche à amener le système vers des états qui sont favorables étant donnée la tâche dudit agent. Chaque action effectuée sur le système conduit celui-ci vers un nouvel état qui est incertain. Un itinéraire technique est défini comme la suite logique et ordonnée d'opérations culturales effectuées sur une parcelle, dans le but d'obtenir une production qui réponde à des objectifs pré-définis (Sebillotte, 1974, 1978). Les deux concepts d'itinéraire technique et de problème d'apprentissage par renforcement peuvent être aisément rapprochés. Par exemple, un agriculteur, i.e. l'agent, effectue un ensemble d'actions (par exemple, semer, ou fertiliser) sur une parcelle. Le résultat de chaque opération est incertain. Par exemple, s'il ne pleut pas suffisamment après une fertilisation minérale azotée, celle-ci ne sera pas disponible pour les plantes. Or, au moment où l'agriculteur fertilise la culture, il n'a pas connaissance avec certitude des précipitations qui vont se produire dans les prochaines semaines : l'efficacité de la fertilisation azotée n'est donc pas garantie. Le but de l'agriculteur est d'amener le peuplement végétal vers des états favorables (par exemple, un peuplement sans carence azotée) qui vont maximiser ses objectifs (par exemple, un agriculteur peut vouloir maximiser le rendement avec une contrainte sur l'utilisation d'engrais azotés).

Les logiciels informatiques d'aide à la prise de décision (*decision support systems* en anglais) visent à faciliter la prise de décision des décideurs humains, pour des problèmes peu ou non-structurés, avec une information incomplète et des objectifs multiples et potentiellement conflictuels (Arnott and Pervan, 2005; Power, 2008). L'aide à la prise de décision dans les itinéraires techniques par de tels logiciels existe depuis plusieurs décennies (Jones et al., 2017). Ces logiciels traitent le plus souvent de la fertilisation, l'irrigation, et de la gestion des maladies et ravageurs ou bien des adventices. Les utilisateurs peuvent être les chercheurs, techniciens ou les agriculteurs. Cependant, à l'heure actuelle, ceux-ci sont toujours peu adoptés en pratique. Notamment, les utilisateurs de ces logiciels ont jugé que l'information ne pouvait être directement traduite en actions, que ces logiciels ne correspondaient pas au processus de décisions qu'utilisent les agriculteurs, que le caractère séquentiel de la prise de décision n'était pas bien pris en compte, ou que les outils de gestion du risque manquent.

L'aide à la prise de décision pour les itinéraires techniques dans les pays du Sud ajoute des contraintes supplémentaires pour de tels logiciels. Les données de terrain sont peu disponibles, comme par exemple le manque de données granulaires pour les sols d'Afrique (Han et al., 2019). Les effets du changement climatique anticipés sont importants pour les productions agricoles en Afrique (Adhikari et al., 2015; Sultan et al., 2013), avec des systèmes de production déjà fragiles. Les agriculteurs doivent faire face à de nombreux risques (Huet, 2022) et le risque climatique est l'un d'entre eux. En effet, la plupart des cultures sont non-irriguées, et par conséquence les récoltes sont grandement conditionnées par l'incertitude météorologique (Mertz et al., 2011).

Question de recherche

Dans quelle mesure l'apprentissage par renforcement peut-il améliorer l'aide à la prise de décisions dans les itinéraires techniques, dans le cas des petits agriculteurs du Sud ?

En particulier, nous explorons comment l'AR peut aider à améliorer la gestion du risque dans les décisions de conduite des cultures (e.g. éviter les pertes importantes de rendement), par comparaison avec les méthodes existantes. Cette étude se limite à la conception d'algorithmes d'AR *ad hoc* pour l'aide à la prise de décision dans les itinéraires techniques. Nous ne nous intéressons pas à la conception d'interfaces utilisateur, nous ne répondons directement non plus aux questions pratiques d'implémentation.

Principaux résultats

Chapitre 1 Ce chapitre donne un cadre conceptuel pour l'application de l'AR pour l'aide à la prise de décision pour la conduite des cultures. L'AR apparaît comme un outil pertinent. L'AR est tourné vers l'action, s'adapte à chaque contexte de décision, montre des similarités avec la manière dont les agriculteurs appréhendent les décisions culturales (dilemme exploration-exploitation*, apprentissage par essai-erreur, adaptation successive des opérations culturales en fonction de l'évolution du peuplement), tout en considerant des séquences d'opérations culturales (et non seulement des opérations de manière indépendante).

Toutefois, l'application de l'AR pour l'aide à la conduite des cultures présente un ensemble de défis théoriques et pratiques, qui peut expliquer le faible nombre de publications jusqu'à présent. Résoudre ces défis requiert un travail joint entre les chercheurs en AR et en agronomie (au sens large, agronomie en tant que confluence de plusieurs disciplines). L'aide à la décision dans la conduite des cultures ne peut être réduite à une optimisation algorithmique ; les objectifs et contraintes des utilisateurs doivent être soigneusement pris en compte. Les données de terrain sont rares et coûteuses, et choisir une mauvaise opération culturale peut fortement impacter un agriculteur (et par extension son noyau familial), en particulier dans une perspective de sécurité alimentaire. Nous avons identifié comme défis théoriques comment résoudre les problèmes de décisions avec peu d'essais ; comment formaliser les problèmes de décisions dans les itinéraires de cultures ; comment traduire les solutions d'AR en une forme interprétable ; comment appliquer l'AR pour des problèmes multi-objectifs et avec des ressources contraintes. L'utilisation des algorithmes de bandit, une forme simplifiée d'AR, semble une voie prometteuse pour appliquer l'AR aux itinéraires techniques en conditions réelles.

Chapitre 2 Ce chapitre présente une nouvelle méthode pour convertir les modèles de cultures existants en Fortran/C/C^{++} en environnements d'AR faciles à manipuler et standardisés. Une telle conversion n'est pas aisée car il faut faire interagir un programme en Python (le langage le plus utilisé en AR) avec un simulateur de culture qui est un programme complexe dans un langage de programmation compilé. De plus, la plupart des simulateurs n'ont pas été conçus pour une interaction journalière avec l'utilisateur, comme le requiert l'AR.

La méthode de conversion proposée est basée sur une librairie spécialisée (*PDI data interface*) qui permet de mettre en pause le programme compilé du simulateur, de lire ses variables internes et d'en

*Voir ci-dessous.

modifier leurs valeurs, et également d'interfacer ce même programme avec Python. Au-delà même des simulateurs de culture, cette avancée méthodologique ouvre la porte à la conversion d'autres simulateurs de haute fidelité en modèles d'apprentissage par renforcement. Comme exemple, le simulateur *Decision Support System for Agrotechnology Transfer* (DSSAT, Hoogenboom et al., 2019), programmé en Fortran, a été converti en modèle d'apprentissage par renforcement en Python. A l'aide d'un algorithme courant d'AR (Proximal Policy Optimization, Schulman et al., 2017), un agent a appris avec succès une pratique durable, respectivement de fertilisation azotée et d'irrigation du maïs. La politique de fertilisation apprise par l'agent a montré des performances de rendement similaires à une politique déterminée par un expert tout en consommant, en moyenne, 28% moins d'engrais azoté. Dans le cas de la politique d'irrigation apprise par l'agent, les performances en rendement sont une nouvelle fois comparables à la politique déterminée par un expert, avec cette fois une réduction en moyenne de 49% de la consommation totale d'eau à la fin de la saison.

Chapitre 3 Ce chapitre propose une nouvelle méthode statistique pour quantifier le risque statistique (similairement à une valeur p ou *p-value* en anglais) de l'identification d'une meilleure pratique culturale parmi d'autres, pour les conditions réelles au champ en se basant sur des résultats simulés. La méthode permet d'évaluer une opération culturale par son résultat moyen (pas d'aversion au risque) ou bien par la moyenne-variance de son résultat (aversion au risque). La détermination de l'ensemble d'opérations culturales à comparer peut être faite par exemple grâce à l'apprentissage par renforcement (comme dans le Chapitre 2 par exemple), ou bien avec une exploration manuelle par un expert, ou encore d'autres méthodes d'optimisation.

La méthode statistique a été appliquée pour l'identification d'une meilleure date de semis, pour une expérimentation de long terme (23 ans) avec du maïs dans la région continentale humide du Canada (Joshi et al., 2017), à l'aide du simulateur de culture DSSAT (Hoogenboom et al., 2019). Les dates de semis ont été évaluées en se basant sur les distributions de leurs rendements grain, à l'aide d'un générateur de séries météorologiques. L'application a révélé que, même dans un contexte prédictif favorable, les possibilités de conclure sur la supériorité d'une date de semis comparée aux autres sont limitées par la distribution des erreurs de prédiction du modèle. Les distributions simulées de rendement grain n'ont pas pu être supposées gaussiennes. Estimer la moyenne du rendement grain avec moins de 30 ans de données météorologiques a montré une incertitude de plus de 2000 kg/ha. L'estimation de la variance du rendement a requis environ dix fois plus d'années que pour la moyenne. Pour deux dates de semis montrant une différence moyenne de rendement supérieure à 2000 kg/ha, il a fallu 4 et 19 ans d'expérimentations au champ pour montrer la supériorité d'une date de semis par rapport à l'autre avec un risque typique de 10% en considérant respectivement la distribution d'erreurs du modèle comme centrée gaussienne, et centrée sous-gaussienne de second ordre. Formuler des hypothèses à propos de la distribution des erreurs du modèle de culture est une étape subjective mais nécessaire à l'inférence. Dans la plupart des applications, le nombre très petit d'erreurs observées (en général, moins de 5 ans d'aléa climatique) ne permet pas de conclure sur une décision avec des garanties statistiques suffisantes. Par conséquent, il est nécessaire de bien confronter les résultats des simulations à l'expertise agronomique, et si possible, de confirmer ceux-ci par de nouvelles expérimentations au champ.

Chapitre 4 Ce chapitre considère le problème simulé d'un groupe d'agriculteurs du Sud du Mali (d'environ 300 individus chaque année), qui accompagné d'experts, identifient ensemble les meilleures

pratiques de fertilisation azotée du maïs pour leurs conditions de culture. Durant plusieurs années, chaque année, les experts assignent une pratique de fertilisation azotée à chaque agriculteur, et à la fin de la saison, les résultats sont collectés. Les experts doivent minimiser les pertes des agriculteurs au cours de ce processus d'identification des meilleures pratiques de fertilisation azotée.

Les simulations reproduisent de manière réaliste les conditions du Sud Mali (voir Adam et al., 2020), le simulateur ayant été calibré sur des données réelles. Bien que les expériences soient simulées, l'identification ne doit pas reposer sur les simulations, dans le sens où ladite méthode doit être applicable directement en conditions réelles. La performance des pratiques de fertilisation est évaluée par une statistique avec aversion au risque (la valeur conditionnelle au risque ou *conditional value-at-risk* en anglais) d'une mesure nouvellement introduite (l'excès de rendement) qui prend en compte à la fois l'efficience de la fertilisation azotée ansi que le rendement grain. Basé sur les travaux de Baudry et al. (2021a)[†], un algorithme de bandit *ad hoc* est proposé. Celui-ci est comparé à une méthode classique qui consiste à tester dans les mêmes proportions chaque pratique durant un nombre pré-déterminé d'années, appelé phase d'exploration, puis à choisir pour les années restantes la pratique ayant eu les meilleurs résultats durant ladite phase d'exploration. En moyenne, la méthode classique (avec 5 ans de phase d'exploration) a montré une performance (voir métrique sus-mentionnée) jusqu'à 77% moins grande que la méthode utilisant un algorithme de bandit, après 4 années. Pour l'ensemble de pratiques de fertilisation azotée considérées par les experts, les simulations n'ont pas montré d'avantages à fractionner les apports d'azote succédant à l'apport de fond, ni à déclencher ceux-ci en fonction de la valeur d'une variable seuil (par exemple, si un total minimum de pluviométrie a été observé depuis le début de la saison).

Discussion

A la manière des logiciels d'aide à la décision existants, nous avons envisagé une aide à la décision, qui, basée sur de l'AR, est centrée sur l'humain : "une assistance aux agriculteurs pour résoudre leurs propres problèmes dans leurs propres termes" (cité et traduit de McCown et al., 2006). Deux dimensions rendent difficile la conception d'un tel système : (i) la grande complexité et la nature incertaine du système dynamique et bio-physique qu'est une parcelle (ii) rendre ces modèles formels d'aide à la décision utiles aux preneurs de décision humains. Les problèmes canoniques d'AR avec l'optimisation d'une fonction d'utilité explicite (i.e. processus de décision de Markov) dans un contexte de données abondantes, sont relativement simples comparés aux problèmes de conduite des cultures, avec peu de données et des objectifs multiples et possiblement conflictuels. Même dans le cas des problèmes canoniques d'AR, l'efficacité des algorithmes d'AR est généralement faible : des millions d'interactions sont nécessaires pour que l'agent puisse résoudre une tâche (voir comme discuté par Dulac-Arnold et al., 2019). D'un autre côté, en agriculture, de nombreux facteurs de confusion sont présents, et de nombreuses conditions pédoclimatiques existent. Identifier les meilleures opérations culturales est généralement difficile, et requiert parfois des méta-analyses (e.g. Giller et al., 2009, voir Chapitre 1). Les essais au champ sont généralement coûteux, et difficiles à conduire durant plusieurs années, en particulier dans les pays du Sud. Dans la suite, nous discutons des directions de recherche que nous avons prises afin de commencer à combler les différences entre les problèmes canoniques de RL et la réalité des problèmes de conduite des cultures.

[†]Publication en co-autorat durant ce doctorat.

Réponses au point (i) : la conduite des cultures est un problème de décision complexe

Une première contribution de ce travail a été d'explorer comment l'AR peut résoudre des problèmes de conduite des cultures avec peu de données. Nous discutons ici l'utilisation des modèles de culture, et de l'incorportation du savoir expert pour faciliter l'apprentissage des tâches de conduite des cultures.

Opportunités et limites des modèles de culture pour l'AR. En dépit du fait qu'en général peu de données granulaires sont disponibles dans les pays du Sud, une fois calibrés, les modèles de culture permettent de générer un grand nombre de cycles de cultures avec un coût de calcul négligeable. En simulant les problèmes de décision, l'utilisation d'algorithmes d'AR n'est pas contrainte par la disponibilité de ces données simulées. Les simulateurs de cultures peuvent être transformés en environnements d'AR (e.g. Garcia, 1999, pour un exemple séminal), et des algorithme d'AR peuvent être utilisés avec succès pour apprendre des pratiques culturales durables avec des simulations de haute fidélité (Chapitre 2). Cependant, dans le contexte des modèles de culture calibrés à partir de quelques années d'expérimentations au champ, la signification statistique de l'identification (soit par AR, ou soit par une autre méthode d'optimisation) de la meilleure opération culturale pour les conditions réelles depuis les simulations, sera peu probablement appuyée par des preuves statistiques suffisantes (Chapitre 3). Ceci nous a mené à savoir si des modèles d'AR pouvaient apprendre directement d'expérimentations au champ.

Dépasser les simulations. Dû au fait que l'identification des meilleures opérations culturales à travers des simulations de modèles de culture est limitée de manière inhérente par l'exactitude des prédictions de ces modèles, quels sont les leviers disponibles pour identifier ces meilleures options directement depuis les expérimentations au champ ? Une première étape est de limiter la complexité du problème de décision. L'agronomie bénéficie d'un volume considérable de savoir expert dont il faut tirer parti le plus possible. Il est peu probable qu'un problème soit totalement étranger aux preneurs de décision. Les chercheurs, techniciens ou agriculteurs peuvent tous contribuer à apporter du savoir expert, soit-il technique ou théorique (par exemple, à travers des simulations), pour formuler ensemble des *a priori* sur les solutions à explorer pour un problème donné (e.g. l'exploration collaborative d'hypothèses de Thorburn et al., 2011). Pour la fertilisation azotée du maïs, le Chapitre 4 donne un tel exemple. Nous avons utilisé le savoir expert pour réduire un problème de décision séquentiel de grande dimensionnalité, i.e. le choix d'une quantité continue d'engrais azoté chaque jour durant la période de culture en fonction de l'état de la parcelle, à un choix unique d'une pratique de fertilisation au début de saison (qui dépend de l'état de la parcelle au cours de la saison, mais qui a été pré-déterminée par les experts). Nous avons aussi utilisé le savoir expert pour regrouper les agriculteurs en cohortes qui partagent des conditions de culture similaires. En suivant le principe de modelage des récompenses (Ng et al., 1999; Randløv and Alstrøm, 1998) à l'aide du savoir expert, dans les chapitres 2 et 4 nous avons défini des fonctions de récompense réelles, qui une fois maximisées, mènent à des compromis agronomiques, économiques et environnementaux souhaitables.

Pour un problème de décision donné, un ensemble d'essais peuvent être conduits en parallèle à chaque pas de temps, appelé apprentissage par lots (Perchet et al., 2015), au lieu d'un seul essai dans la configuration canonique purement séquentielle (Lattimore and Szepesvári, 2020). L'apprentissage par lots permet donc, pour une même durée, de conduire plus d'essais que la configuration purement

séquentielle. Par exemple, dans le Chapitre 4, nous avons considéré l'identification collaborative de la meilleure opération culturale à l'aide d'un groupe d'agriculteurs conduisant simultanément des essais au champ chaque saison. De telles configurations existent avec les essais en ferme (e.g. Baudron et al., 2012; Falconnier et al., 2016; Naudin et al., 2010). Grâce à cette redéfinition du problème de décision, un algorithme de bandit a été capable d'identifier les meilleures opérations culturales après quelques années. Cependant, en conditions réelles, une telle configuration induit des corrélations entre groupes. Par exemple, pour une même année, des agriculteurs à proximité sont susceptibles d'observer des séries météorologiques similaires. Ces corrélations doivent être correctement prises en compte par les méthodes introduites au Chapitre 4, par exemple en utilisant des bandits contextuels (Lattimore and Szepesvári, 2020) avec les concepts des modèles linéaires mixtes (Laird and Ware, 1982).

Réponses au point (ii) : concevoir des modèles formels utiles aux agriculteurs

Les modèles bio-économiques formels devraient d'abord viser leur utilité pour les praticiens (Charlton and Street, 1975, p. 263-265). Les processus cognitifs des agriculteurs doivent être pris en compte dans la conception des logiciels d'aide à la prise de décision. Ces logiciels doivent permettre aux utilisateurs d'explorer des solutions pragmatiques, plutôt que d'indiquer des solutions optimisées, comme souligné par Hochman and Carberry (2011). Dans cette partie, nous discutons de la manière dont nous avons pris en compte certains de ces aspects dans la conception de logiciels d'aide à la décision basés sur le RL.

Rendre compte de l'incertitude et du risque liés aux décisions L'aide à la prise de décisions des agriculteurs doit être ciblée sur la caractérisation de l'incertitude des décisions (McCown et al., 2006; McCown and Parton, 2006). Au-delà du fait que l'AR traite par essence des séquences de décisions dans un environnement incertain, nous avons exploré particulièrement l'usage de statistiques avec aversion au risque avec la moyenne-variance (MV, Markowitz, 1952), et la valeur condtionnelle au risque (*conditional value-at-risk* en anglais ou CVaR, Mandelbrot, 1997), respectivement dans les Chapitres 3 et 4. De telles statistiques sont d'intérêt pour les problématiques de sécurité alimentaire où les échecs de cultures doivent être évités. Cependant, nous n'avons pas répondu à la question de savoir comment choisir de manière objective les paramètres de la moyenne-variance, ou de la valeur conditionnelle au risque, pour chaque agriculteur. Des techniques d'élicitation existent dans la litératture (par exemple Iyer et al., 2020, dans le contexte européen). Une traduction adéquate des préférences de risque des agriculteurs en lesdits paramètres doit être étudiée.

Le dilemme exploration-exploitation Dans le Chapitre 4, nous avons exploré l'usage de la minimisation du regret cumulé (MRC) qui répond au dilemme d'exploration-exploitation. Ce dilemme peut se résumer au fait que, pour déterminer quelle est la meilleure action parmi un ensemble d'actions, il faut tester suffisamment chaque action (du fait de l'aléa). Cependant, chaque action sous-optimale fait accumuler (en espérance) une perte pour le preneur de décision. Or, le preneur de décision veut identifier au plus vite la meilleure action, tout en évitant les pertes. Le MRC répond directement à cette problématique. Ce type de comportement a été rapporté chez les agriculteurs (e.g. Cerf and Meynard, 2006; Evans et al., 2017). Dans les expérience simulées du Chapitre 4, en minimisant les pertes, nous avons montré que le MRC permettait de réduire le coût associé à l'identification des meilleures pratiques culturales pour les agriculteurs, par rapport à une approche conventionnelle.

Nous recommandons que la minimisation du regret cumulé, qui est une approche novatrice, continue d'être explorée dans le futur pour l'aide à la conduite des cultures.

Conclusion

Dans cette thèse, nous avons exploré dans quelle mesure l'apprentissage par renforcement (AR) pouvait améliorer les logiciels existants d'aide à la décision pour la conduite des cultures, dans le cas particulier des petits agriculteurs des pays du Sud. En se basant sur les critiques faites par les utilisateurs de ces logiciels, nous avons d'abord conduit un exercice exploratoire pour identifier les directions de recherche prometteuses, afin de définir un cadre conceptuel pour l'application de l'AR pour l'aide à la conduite des cultures. L'AR est apparu comme un outil *ad hoc* avec un grand potentiel comme élément de prise de décision dans les itinéraires techniques. L'AR traite de manière inhérente avec des séquences de décisions afin de contrôler un système dynamique, incertain et inconnu. Il est, de fait, tourné vers l'action. De plus, l'AR partage des similitudes avec la manière dont les agriculteurs abordent la conduite des cultures. En particulier, l'AR apprend une tâche par essai-erreur, et les actions sont déterminées selon l'évolution incertaine du système (dans le cas des itinéraires de culture, l'évolution de la parcelle). Cependant, l'application de l'AR fait aussi face à de nombreux défis, d'où la cause probable d'une littérature limitée sur le sujet. La réalité de la conduite des cultures est éloignée des problèmes canoniques d'AR. Le contexte des pays du Sud rend son application à l'aide à la conduite des cultures encore plus ardue, du fait principalement du manque de données de terrain.

Dans cette thèse, nous avons considéré des niveaux décroissants du nombre possible d'interactions entre l'agent d'AR et son environnement : depuis un nombre presque infini, jusqu'à quelques milliers d'interactions. Nous avons proposé une méthode générique pour convertir des modèles de culture en environnements d'apprentissage par renforcement standardisés et faciles à manipuler. Les modèles de cultures permettent d'entraîner des agents d'AR avec un coût de calcul négligeable. Dans un environnement d'AR, nous avons entraîné avec succès un agent à la fertilisation (et l'irrigation) durable du maïs grâce à un algorithme commun d'AR. Nous avons aussi proposé une méthode statistique pour quantifier la signification statistique de l'identification des meilleures pratiques culturales pour les conditions réelles au champ, en se basant sur des simulations. Nous avons considéré à la fois un critère de décision neutre au risque et un critère avec aversion au risque. Nous avons pris une décision sur une date de semis pour le maïs au Canada, basé les simulations d'une expérimentation de long terme. Nous avons montré l'intérêt des statistiques avec aversion au risque face à l'incertitude saisonnière pour les problématiques de sécurité alimentaires où les échec de culture doivent être évités. Cependant, dans le cas des pays du Sud, nous concluons sur le fait que pour la plupart des applications des modèles de culture, il est peu probable que l'identification d'une meilleure opération culturale soit appuyée par assez de preuves statistiques.

Les limites des environnements simulés nous ont conduits à considérer une application de l'AR pour apprendre directement en conditions réelles, avec un nombre réaliste d'interactions. Nous avons abordé l'identification collaborative des meilleurs pratiques culturales par un groupe d'agriculteurs conduisant les essais au champ. Dans un exercice simulé, nous avons reproduit les conditions de culture du Sud du Mali. Nous avons conçu une méthode d'identification utilisant un algorithme de bandit avec un critère de décision avec aversion au risque, et avec la contrainte de minimiser les pertes accumulées par les agriculteurs durant le processus d'identification. Les simulations ont montré que, en tirant parti du savoir expert pour réduire la complexité du problème de décision, l'identification des

meilleures pratiques culturales à l'aide d'un algorithme de bandit pouvait être envisagée en conditions réelles. Dans la plupart des cas, l'algorithme de bandit a été capable de réduire davantage les pertes des agriculteurs comparé à la méthode qui consiste à essayer chaque opération culturale de manière equi-proportionnelle durant un nombre fixe d'années.

En conclusion, l'AR peut donner au nouveau souffle aux logiciels d'aide à la prise de décision pour les itinéraires techniques, et est d'intérêt dans le cas des pays du Sud. Son application requiert la conception d'algorithmes *ad hoc*, capables de répondre aux nombreuses contraintes et objectifs de la gestion des cultures. Une parcelle et sa conduite sont des éléments d'un système plus large qui est celui de la ferme, et son écosystème *sensu lato*, en incluant sa dimension sociale. Par conséquent, la conception de logiciels d'aide à la prise de décision dans les itinéraires techniques va bien au-delà d'un problème purement numérique. Nous avons montré que l'adaptation de l'AR pour la conduite des cultures est toutefois possible, et son application est d'intérêt dans le contexte des petits agriculteurs des pays du Sud. L'arrivée de nouvelles technologies et leurs promesses ne doivent pas nous dispenser de tirer profit de la riche expérience déjà accumulée avec les logiciels d'aide à la décision pour l'agriculture. Cette expérience a impliqué de multiples disciplines comme l'économie, la sociologie, les sciences cognitives ou encore l'ergonomie. Une application de l'AR sera probablement un succès à la condition qu'une approche multi-disciplinaire soit poursuivie, et ce depuis la conception des modèles formels de décision.

Publications and software

Publications presented in this document

Gautron et al. (2022a) Gautron, R., Maillard, O.-A., Preux, P., Corbeels, M., and Sabbadin, R. (2022a). Reinforcement learning for crop management support: Review, prospects and challenges. *Computers and Electronics in Agriculture*, 200:107182
Chapter 1 is dedicated.

Baudry et al. (2021a) Baudry, D., Gautron, R., Kaufmann, E., and Maillard, O. (2021a). Optimal thompson sampling strategies for support-aware cvar bandits. In *International Conference on Machine Learning*, pages 716–726. PMLR
Used in Chapter 4.

Other publications

Mitton et al. (2022) Mitton, N., Brossard, L., Bouadi, T., Garcia, F., Gautron, R., Hilgert, N., Ienco, D., Largouët, C., Lutton, E., Masson, V., et al. (2022). Foundations and state of play

Software

Gautron (2021, unmaintained) Gautron, R. (2021). A bandit geared gym encapsulation of the DSSAT crop simulator. https://github.com/rgautron/DssatBanditEnv
Used in Chapter 3.

Gautron (2022) Gautron, R. (2022). gym-DSSAT: an easy to manipulate crop environment for reinforcement learning. https://gitlab.inria.fr/rgautron/gym_dssat_pdi — a short article describing `gym-DSSAT` was accepted for a poster and an oral presentation at the AAAI-23 conference, *AI for Agriculture and Food Systems* workshop.
Chapter 2 is dedicated. Used in Chapter 4.

Acknowledgements

Immense thanks to Brian King, who made this Ph.D. possible. I am also incredibly grateful to Marc Corbeels, Philippe Preux and Odalric-Ambrym Maillard for their outstanding supervision, dedication, support and kindness. I am honored by the wholehearted support of Emilio Padrón, who have dedicated a considerable amount of time to help developing `gym-DSSAT`. Thanks to whoever directly or indirectly contributed to this Ph.D. I acknowledge all the co-authors of the articles presented in this Ph.D. Special thanks to my Ph.D. fellows Dorian Baudry, and Patrick Saux who have contributed in many aspects. Thanks to SequeL (Inria Lille) and AIDA (CIRAD) team. In particular, thanks to Edouard Leurent, Mathieu Seurin and Emilie Kaufmann for their kind support. Thanks to Nathalie Mitton (head of Inria FUN team), Bruno Raffin (head of DataMove), Julien Bigot (original author of PDI Data Interface) for their time. Thanks to Krishna Naudin (head of CIRAD AIDA team) for having facilitated this Ph.D. Thanks to the CGIAR Big Data Platform for agriculture, to the CIRAD and Inria Lille for having founded and supported this Ph.D. Thanks to Gerrit Hoogenboom and Cheryl Porter for their active support with the DSSAT crop model. I am grateful to Marie-Hélène Jeuffroy and Eric Justes for their remarkable help as members of my annual supervision committee. I am also grateful to all the members of the jury for their careful examination, remarks and improvements of this manuscript. Last but not least, I thank all my relatives, far beyond the accomplishment of this Ph.D.

Nomenclature

Roman Symbols

a	action
a^*	optimal action
$\mathcal{A}$	action space
$\hat{C}$	empirical conditional value-at-risk
e	error measure
$\mathbb{E}[X]$	expectation of the random variable X
$J(T)$	objective function of undiscounted returns until hortizon T
$\mathfrak{M}$	Markov decision process
$N_{(s,a)}$	number of times the action a has been taken in state s
$\mathcal{N}(\mu, \sigma^2)$	normal distribution of mean μ and variance σ^2
$\mathcal{P}(X)$	the set of probability distributions over set X
$Q(s, a)$	quality of state s and action a
r	return sample in a Markov decision process
$\mathbf{r}$	the return function of a Markov decision process
s	state
$\mathcal{S}$	state space
T	horizon of a Markov decision problem
t	time step
$\mathbf{p}$	the transition function of a Markov decision process
$V(s)$	value of state s
$\mathbb{V}[X]$	variance of the random variable X

Greek Symbols

α	parameters for the conditional value-at-risk $\big(\alpha \in (0, 1)\big)$
$\alpha(s, a)$	learning rate for the Q-learning algorithm for state s and action a
δ	risk level $\big(\delta \in (0, 5)\big)$

γ	discount factor of the objective function $\big(\gamma \in [0, 1)\big)$
μ_X	mean of the random variable X
ν_X	distribution of the random variable X
π	policy
π^*	optimal policy
ρ	parameters for the mean-variance $\rho > 0$
σ_X	standard deviation of the random variable X

Other Symbols

argmax	arguments of the maxima
$X \times Y$	if X and Y are sets, Cartesian product between X and Y, else, multiplication operator
$:=$	definition
$X \widehat{=} Y$	Y is an estimator of X
$\hat{X}$	empirical estimate of the random variable X
$\overset{\text{i.i.d.}}{\sim}$	Independent and identically distributed
$x \in X$	element x in set X
inf(X)	infimum of X
$\Pr(X)$	probability of X
$\{x, y\}$	set of elements x and y
$\langle X, Y \rangle$	tuple of elements X and Y
Unif(a, b)	Uniform distribution with support $[a, b]$

Acronyms / Abbreviations

CoV	co-variance
CVaR	conditional value-at-risk
VaR	value-at-risk
MV	mean-variance

Acronyms

A2C advantage actor-critic

AI artificial intelligence

ANE agronomic nitrogen use efficiency

CRM cumulated risk minimization

DP dynamic programming

DQN deep Q-learning

DSS decision support system

DSSAT decision support system for agrotechnology transfer

EBMDP event based markov decision problem

ETC explore-then-commit

FQI fitted Q-iteration

IoT internet of things

MAB multi-armed bandit

MDP markov decision process

ML machine learning

NN neural network

PDI PDI data interface

PODMP partially observable markov decision problem

PPO proximal policy optimization

PSR predictive state representation

RL reinforcement learning

SAC soft actor-critic

SARSA state–action–reward–state–action

SI sustainable intensification

SOC stochastic optimal control

TD temporal differences

WUE water use efficiency

YE yield-excess

Glossary

action How the environment dynamics are controlled by the agent.

action space The set of possible actions.

agent The entity that acts on the environment in order to optimize the objective function.

deep neural network Neural network with several layers. In RL, this number of layers is limited (from a few to say a dozen layers) whereas in machine learning, there may be hundreds and even thousands of layers.

environment The object with which the agent interacts.

episode A single sequence of interactions of the agent with the environment, from a given initial state.

exploration-exploitation dilemma The situation in which an agent has the choice between performing an action with consequences which are known (exploitation) and an action with consequences which are unknown (exploration).

horizon Maximum number of time steps of an episode.

in silico A virtual experience.

internet of things (IoT) Networks of uniquely identified physical devices which can autonomously communicate between themselves or with humans, and process data.

Markov decision process Mathematical formalization of the environment in a Reinforcement Learning problem, see Figure 1.5.

neural networks A neural network is made up of a set of layers of simple computation units, called neurons. Each neuron receives data as input and outputs one or more labels (usually either symbolic, or numeric). Mathematically speaking, a neural network is a function.

objective function The function that the agent optimizes by controlling the environment.

observation In an MDP, a snapshot of the environment state. In the general case, there is no assumption that an optimal action may be determined using an observation.

overfitting A machine learning model that has been trained and performs well in training situations but performs poorly in unseen situations.

policy A function that indicates how the agent acts depending on the environment state.

quality function The expected value of the objective function when the environment is in a given state and the agent first performs a given action and then follows a given policy.

return A positive or negative stimulus provided by the environment to the agent which indicates if the past actions have been beneficial to the agent with regards to its objective.

sample complexity Number of samples required to solve a problem. The higher the sample complexity, the harder the problem.

state A set of descriptors of the environment that is sufficient to decide on an optimal action.

state space The set of possible states.

stationary A random process in which distributions do not change over time.

value function The expected value of the objective function when the environment is in a given state and the agent follows a given policy.

List of Figures

i.1 Level of data availability that each chapter considers for the evaluation of crop operations. In chapter 2, the purely simulated conditions allow to explore crop operations millions of times. Chapter 3 addresses the quantification of the statistical guarantees of decisions, from crop model simulations to real-field conditions. Millions of simulated results of crop operations are possible, but the final results are also constrained by the availability of real-field data. In chapter 4, we target a realistic number of crop operation trials for the learning method to be applicable in real conditions. 2

1.1 A simplified example of maize management plan. The BBCH scale follows the successive maize growth stages as found in Meier (1997), where the first and second digits respectively correspond to the primary and secondary growth stage codes. A dashed box indicates that the operation requirement is uncertain. All operations are made within a time window where the exact date of occurrence is uncertain. 'N fert.' stands for nitrogen fertilization ; 'weed.' for weeding ; 'pest.' for pest and disease control. . . 7

1.2 The reinforcement learning loop. A decision maker, called the agent, interacts with its environment. The agent task is to control the environment evolution. Sequentially, the agent takes an action based on an observation of the environment. The action impacts the environments, and the agent receives a return that indicates how it performs regarding the task to be completed. This loop repeats until the decision sequence eventually ends. 10

1.3 The four elements of a Markov decision process. An MDP models the environment in reinforcement learning problems. 11

1.4 A simplistic irrigation problem modeled as a Markov decision process (MDP). Two states are possible: a stressed crop (s_1) or a well watered crop (s_2). Each arrow between two states is a transition which ends in the state pointed by the arrow head. Watering the crop (a_2) always leads to a well watered state, but it has a cost, hence the negative return. If no irrigation is provided (a_1), 30% of the time rainfall occurs and the crop will be well watered for free, hence the great return. But, 70% of the time, no rainfall occurs and the crop gets stressed, which is highly penalized by the return. 11

1.5 The representation of a sequence of decisions is called an episode. In a canonical reinforcement learning problem, starting with the environment in an initial state s_0, at each discrete decision step t, depending on the environment current state s_t the agent decides on an action a_t thanks to its policy. After the agent takes the action a_t, the environment transits towards its uncertain next state s_{t+1}, given by the transition function **p**. The return function **r** provides a return r_t which indicates to the agent how it performs regarding the task to be completed. 11

1.6 Both stochastic optimal control (SOC) and reinforcement learning (RL) address the problem of controlling a system with uncertain dynamics. The main historical difference is that SOC supposes the dynamics of the system to be known while RL does not. Recently, hybrid algorithms have been developed, combining RL and SOC. The multi-armed bandit (MAB) is a simplified case of RL with a one-state MDP, see Section 1.2.6. 13
1.7 Modern reinforcement learning methods are hybrids of three problem solving methods: critic, actor and planning methods, see Section 1.2.5. 14
1.8 Key contributions towards reinforcement learning (RL) use in agriculture. Only Garcia (1999) is categorized as modern RL. Earlier work are based on paradigms that are the historical parents of RL. 17
1.9 An RL-based decision support system for a community of farmers. At any moment, a farmer can query the agent to explore tailored crop management recommendations based on farmer's constraints and objectives. Data should be interactively and iteratively exchanged between farmers and the agent in order to collectively improve the policy for crop management decision problems. 20
1.10 Challenging features and respective prospects for RL-based crop management decision support systems. The inner circle represents the desirable features for an RL based crop management DSS. All of these features inter-relate. The outer circle represents the potential technical or theoretical solutions to reach the corresponding features of the inner circle. 21

2.1 The Reinforcement learning loop. The goal of the decision maker, called the agent, is to control the evolution of a dynamical system called the environment, in order to perform a given task. Sequentially, the agent observes the environment and takes an action based on this observation. The action affects the environment, and the agent receives a return that indicates how it is performing regarding the task to perform. The process repeats until the decision series eventually ends. 32
2.2 In reinforcement learning, a Markov decision process models the environment. At each time step t, the agent observes the environment current state s_t. Depending on s_t, the agent takes an action a_t according to its policy. As a consequence of taking action a_t, the environment transits to next state s_{t+1}, depending on the transition function $\mathbf{p}$, and the agent observes the return r_t which depends on the return function $\mathbf{r}$. This process repeats until the episode eventually ends. 35
2.3 From a user's perspective, `gym` environments are simplified interfaces to simulators, through standardized methods. 37
2.4 Configuration files used by the crop management reinforcement learning environment. At the top of the figure, files in dashed boxes define the reward function and state and action spaces of the Markov decision process. Dashed boxes indicate straightforward to customize configuration files. At the bottom of the figure, `DSSAT` files parameterize simulations, and the `PDI` specification tree is a technical file which manages the communication between `DSSAT-PDI` and `gym-DSSAT`. 40
2.5 Successive subsets of `DSSAT` state variables until agent observations. Boxes filled with grey indicate files defining state variable subsets. 41

2.6 Simplified example of PDI use in gym-DSSAT for the fertilization decision problem. The left-hand side corresponds to the PDI specification tree (Figure 2.4), and the right-hand side to the Fortran code of DSSAT-PDI (Section 2.5.2). 42
2.7 The elements of gym-DSSAT. 44
2.8 Undiscounted cumulated returns and applications for the fertilization problem. 46

3.1 Methodological steps for crop management decision making from crop model simulations to real field conditions with statistical guarantees. CI stands for confidence interval. 56
3.2 Observed values (symbols) versus DSSAT model predictions (lines) of maize grain yield for 21 years at Sainte-Anne-de-Bellevue, Québec, Canada. Observed data are from a long-term maize experiment (Joshi et al., 2017). 62
3.3 Simulated maize grain yield distributions induced by weather uncertainty for planting at days of the year (DOY) 135 and 165 (10^5 samples) at Sainte-Anne-de-Bellevue, Québec, Canada. The DSSAT model (Hoogenboom et al., 2019) was used for the simulations. . 63
3.4 Uncertainty for the mean of the simulated maize grain yield at Sainte-Anne-de-Bellevue, Québec, Canada, as a function of the sample size for weather-induced yield distributions (planting date at day of year 135), at different risk levels. Results are computed under assumption 2 (see Section 3.2.2, eq. B.15). The true mean is the dashed horizontal line. 960 replications were performed using the DSSAT crop model (Hoogenboom et al., 2019). 64
3.5 Uncertainty for the standard deviation of the simulated yield response as a function of the sample size for weather-induced yield distribution (planting date at the day of the year (DOY) 135), under different risk levels. Results are computed under assumption 2 (see Section 3.2.2), eq. B.18. The true standard deviation is the dashed horizontal line. 960 replications were performed using the DSSAT crop model (Hoogenboom et al., 2019). 65
3.6 Uncertainty comparison of model (DSSAT model Hoogenboom et al., 2019) error standard deviation for 21 error measures (maize field experiment in Sainte-Anne-de-Bellevue, Québec, Canada, see Joshi et al., 2017). The Gaussian hypothesis refers to assumption 1 in Section 3.2.2 (Eq. B.12), the centered second order sub-Gaussian hypothesis to assumption 2 (Eq. B.29) and the sole boundedness hypothesis to assumption 3 (Eq. B.18). 66
3.7 Minimal risk level for confidence interval disjunction between maize planting date DOY 135 and 165 at Sainte-Anne-de-Bellevue, Québec, Canada, as function of the number of errors for (a) the centered Gaussian error hypothesis, and (b) the centered second-order sub-Gaussian error hypothesis. 1000 replications were performed using the DSSAT crop model (Hoogenboom et al., 2019). The red lines indicate the typical 5% and 10% risk levels. 67

4.1 Yearly process to generate nitrogen fertilizer recommendations: at the beginning of the crop ping season. Individuals from the overall farmer population volunteered to test a fertilizer practice. Similar symbols represent a cohort, i.e., a group of farmers having fields with the same soil type. The group of volunteer farmers was broken down by cohort and researchers independently generated fertilizer recommendations for each cohort. Researchers did not control the number of volunteers from the respective cohorts In this example, only three of the four possible cohorts are found in the volunteer group. 77
4.2 Schematic representation of the ensemble best fertilization identification process and the canonical bandit problem. 78
4.3 Yield Excess (YE^{π}, Equation 4.5) for $ANE_{ref} = 15$ kg grain /kg N and $ANE_{ref} = 30$ kg grain /kg N. Y^{π} is the maize grain yield obtained with nitrogen fertilizer practice π, Y^{0} is the yield obtained with no nitrogen fertilization (control). ANE^{π} is the Agronomic Nitrogen use Efficiency of the nitrogen fertilizer practice π (Equation 4.1). 81
4.4 The Conditional Value-at-Risk (CVaR) of level α is the mean value of the blue area of the distribution of probability $0 < \alpha \leq 1$. VaR_{α} stands for Value-at-Risk of level α and is the quantile of probability α of the distribution. The more $\alpha \to 1$, the more risk neutral is the CVaR. μ represents the mean value of the distribution which equivalent to the CVaR of level $\alpha = 100\%$. 82
4.5 Averaged sampling proportions for soils `ITML840105` and `ITML840101`, $T = 20$ years. 960 replications of the whole experiment were done. The fertilizer practices are ordered according to the true Conditional Value-at-Risk at level 30% (CVaR) of their Yield Excess (YE) with ANE_{ref}=15 kg grain/kg N ; the greener the color, the better a fertilizer practice is. 89
4.6 Empirical conditional Value-at-Risk (CVaR) at level 30% (CVaR) of maize yield excesses (YE) between $T = 0$ and the considered T ; $ANE_{ref} = 15$ kg grain/kg N. 960 replications of the whole experiment were done. One time step T is one year ; 'mean batch size' is the number of farmers who have volunteered to participate at the trials, averaged over all years and all replications. Confidence intervals were computed following Thomas and Learned-Miller (2019). 90
4.7 Mean cumulated regret of population, for the Conditional Value-at-Risk (CVaR) at level 30% of Yield Excess (YE); $ANE_{ref} = 15$ kg grain/kg N. The cumulated cumulated regret is averaged over the farmers' population, between $T = 0$ and the considered T. 960 replications of the whole experiment were done. One time step T is one year, 'mean batch size' is the number of farmers who have volunteered to participate in the trials, averaged over all years and all replicates. 90
4.8 Distribution of individual cumulated regret after $T = 20$ years for Conditional Value-at-Risk at level 30% (CVaR) of the yield excess (YE) ; $ANE_{ref} = 15$ kg grain/kg N. The total number of farmers corresponds to a group of 300 farmers, with 960 replications of the whole experiment. 91

A.1 Undiscounted cumulated returns and applications for the irrigation problem. 105

B.1 Confidence interval decision flow, based on the target metric (mean, variance) and hypotheses (Gaussian or bounded, unknown mean or centered, sample size). 115
B.2 Normal quantile/normalized empirical quantile plot for 10^5 yield simulations (planting date DOY 135). 117
B.3 Confidence intervals for planting date DOY 135. 119
B.4 Frequency of lower and upper boundary crossings for planting date DOY 135, defined as the fraction of times $\mu_{Y_{\text{sim}}} < \underline{\mu_{Y_{\text{sim}}}}(\delta/2)$ and $\mu_{Y_{\text{sim}}} > \overline{\mu_{Y_{\text{sim}}}}(\delta/2)$ respectively over 1000 simulations. 120
B.5 Residual plot for 21 error observations. 121
B.6 Normal quantile/normalized empirical quantile plot for 21 error observations. 121
B.7 Residuals against predicted yield values, 21 observations. 122
B.8 Uncertainty of the mean criteria for ground truth distributions of planting date DOY 135 and 165 ($n=500$ simulations). 123
B.9 Uncertainty for ground truth mean-variance ($\rho=1$) for planting date DOY 135 and 165 ($n=10000$ simulations). Gaussian error hypothesis 123
B.10 Uncertainty for ground truth mean-variance ($\rho=1$) for planting date DOY 135 and 165 ($n=10000$ simulations). Second-order sub-Gaussian error hypothesis. 124

C.1 Simulated impact of maize fertilizer practices on grain yield, Agronomic Nitrogen use Efficiency (ANE), Yield Excess (YE) for 10^5 hypothetical years using a weather generator. Maize cultivar was the same for all simulations. Practices indexes are indicated on the left-hand side of each sub-figure. 126
C.2 Examples of weights sampled from Dirichlet distributions during BCB execution, respectively for 10 and 100 rewards. The greater the number of rewards, the less variance the weights show. The variance of weights is related to the noise level in the computation of the empirical CVaR of BCB. 130
C.3 Farmers' empirical CVaR at level of all YE received between $T = 0$ and the considered T. 132
C.4 Cumulated regret averaged over the population for the CVaR at level of YE. 132
C.5 Distribution of individual cumulated regret after $T = 20$. 132

List of Tables

1.1 Principal works which have applied reinforcement learning algorithms to crop management. (c) indicates a continuous variable; (integer) indicates the number of discrete elements; (y/n) indicates a binary feature. In all works, decisions are made during a single growing season. 18

1.2 Technological opportunities for Reinforcement Learning (RL) applications. The interactive communication between a virtual agent and the ground reality with farmers, as shown in Figure 1.9, require an *ad hoc* data architecture to allow the RL loop. The back end system is dedicated to agent's computational requirements. The data collection elements essentially captures fields states. Finally, the communication elements allow the human-machine dialog. 22

2.1 `gym-DSSAT` available actions 39

2.2 Default observation space for the fertilization task. 39

2.3 Expert fertilization policy. 'DAP' stands for Day After Planting. 39

2.4 Performance indicators for fertilization policies. An hyphen means `gym-DSSAT` does not directly provide the variable, but it can be easily derived. 46

2.5 Mean (st. dev.) fertilization baselines performances computed using 1000 episodes. For each criterion, bold numbers indicate the best performing policy. 47

3.1 Statistics of simulated maize grain yield responses for planting dates DOY 135 and 165 at Sainte-Anne-de-Bellevue, Québec, Canada, computed from 10^5 samples. The DSSAT model (Hoogenboom et al., 2019) was used for the simulations. Due to the high number of samples, these statistics are considered as exact. 63

4.1 : Main properties of the soil types of the fields of farmers growing maize in Koutiala, Mali (Adam et al., 2020). 'SLOC.' stands for soil organic matter (g C/ 100 g soil, mean value for the 0-30 cm topsoil); 'SLDR' stands for soil drainage rate (fraction/day); 'SLDP' stands for soil depth (cm); 'Prop' stands for the percentage of each soil type present in the study area. 79

4.2 Maize nitrogen fertilizer recommendations for maize in Koutiala, Southern Mali, that were considered in the virtual experiment. Whether or not rainfall and plant nitrogen stress were considered as factors for the fertilizer recommendation is indicated by Yes or No. 'NSTRES' stands for plant nitrogen stress and 'DAP' for days after planting. . . . 79

4.3 Statistics of the optimal nitrogen fertilizer practices for each of the soil types presented in Table 4.1. For the corresponding optimal nitrogen fertilizer practice π^*, we define N^{π^*}: quantity of nitrogen fertilizer applied; $\text{CVaR}_{30\%}(X)$: conditional Value-at-Risk of X of level 30% (Section 4.2.1); $\bar{X}$: mean value of X; Y^{π^*}: maize grain yield; ANE^{π^*}: Agronomic Nitrogen use Efficiency; YE^{π^*}: Yield Excess (Section 4.2.1); parentheses indicate standard deviations. 87

A.1 Expert irrigation policy. 'DAP' stands for Day After Planting. 103
A.2 Performance indicators for irrigation policies. An hyphen means `gym-DSSAT` does not directly provide the variable, but it can be easily derived. 103
A.3 Default observation space for the irrigation task. 104
A.4 Mean (st. dev.) irrigation baselines performances computed using 1000 episodes. For each criterion, bold numbers indicate the best performing policy. 104
A.5 Mean (st. dev.) days of simulation to reach growth stages for the irrigation problem (1000 episodes). 105
A.6 Mean (st. dev.) days of simulation to reach growth stages for the fertilization problem (1000 episodes). 105

B.1 Maize cultivar parametrization in DSSAT based on Joshi et al. (2017). See (Hoogenboom et al., 2019) for the detailed meaning of these coefficients. 117
B.2 Statistical tests for normality of simulated distribution of planting date DOY 135. #1: d'Agostino and Pearson (1973); d'Agostino (1971) ; #2: Jarque and Bera (1980) ; #3: Shapiro and Wilk (1965). 118
B.3 Statistical tests for model error distribution (significance level for multiple testing $\delta' = 1\%$). #1: Ljung and Box (1978), #2: Casella and Berger (2021) (Z-test), #3: Snedecor and Cochran (1989), #4: Kendall (1938). 122

C.1 Maize cultivar parametrization in DSSAT 125
C.2 Automatic planting parametrization in DSSAT. PFRST: Starting date of the planting window; PLAST: End date of the planting window; PH2OL: Lower limit on soil moisture for automatic planting; PH2OU: Upper limit on soil moisture for automatic planting; PH2OD: Depth to which average soil moisture is determined for automatic planting; PSTMX: Maximum temperature of planting; PSTMN: Minimum temperature of planting. 127

List of Algorithms

1 Q-Learning algorithm 14

2 Simplified pseudo-code of BCB. 84

3 Bisection method for confidence interval disjunction risk level search. 116

4 BCB: identification strategy at cohort level (detailed) 129

5 ETC: identification strategy at cohort level 131

Bibliography

Acerbi, C. and Tasche, D. (2002). On the coherence of expected shortfall. *Journal of Banking & Finance*, 26:1487–1503.

Adam, M., MacCarthy, D. S., Traoré, P. C. S., Nenkam, A., Freduah, B. S., Ly, M., and Adiku, S. G. (2020). Which is more important to sorghum production systems in the sudano-sahelian zone of west africa: Climate change or improved management practices? *Agricultural Systems*, 185:102920.

Adhikari, U., Nejadhashemi, A. P., and Woznicki, S. A. (2015). Climate change and eastern africa: a review of impact on major crops. *Food and Energy Security*, 4(2):110–132.

Affholder, F. (1995). Effect of organic matter input on the water balance and yield of millet under tropical dryland condition. *Field Crops Research*, 41(2):109–121.

Affholder, F., Poeydebat, C., Corbeels, M., Scopel, E., and Tittonell, P. (2013). The yield gap of major food crops in family agriculture in the tropics: Assessment and analysis through field surveys and modeling. *Field Crops Research*, 143:106–118.

Agrawal, S., Koolen, W. M., and Juneja, S. (2021). Optimal best-arm identification methods for tail-risk measures. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*.

Altieri, M. A., Nicholls, C. I., Henao, A., and Lana, M. A. (2015). Agroecology and the design of climate change-resilient farming systems. *Agronomy for sustainable development*, 35(3):869–890.

Anapalli, S. S., Ma, L., Nielsen, D., Vigil, M., and Ahuja, L. (2005). Simulating planting date effects on corn production using rzwqm and ceres-maize models. *Agronomy Journal*, 97(1):58–71.

Anonymous (2021). Empirical chernoff concentration: beyond bounded distributions. Manuscript under review by NEURIPS 2022.

Arnott, D. and Pervan, G. (2005). A critical analysis of decision support systems research. *Journal of information technology*, 20(2):67–87.

Asseng, S., Ewert, F., Rosenzweig, C., Jones, J. W., Hatfield, J. L., Ruane, A. C., Boote, K. J., Thorburn, P. J., Rötter, R. P., Cammarano, D., et al. (2013). Uncertainty in simulating wheat yields under climate change. *Nature climate change*, 3(9):827–832.

Åström, K. J. (1965). Optimal control of markov processes with incomplete state information. *Journal of Mathematical Analysis and Applications*, 10(1):174–205.

Attiya, H. and Welch, J. (2004). *Distributed computing: fundamentals, simulations, and advanced topics*, volume 19. John Wiley & Sons.

Auer, P. (2002). Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, 3(Nov):397–422.

Auer, P., Jaksch, T., and Ortner, R. (2008). Near-optimal regret bounds for reinforcement learning. *Advances in neural information processing systems*, 21.

Auer, P. and Ortner, R. (2006). Logarithmic online regret bounds for undiscounted reinforcement learning. *Advances in neural information processing systems*, 19.

Barbosa, A., Trevisan, R., Hovakimyan, N., and Martin, N. F. (2020). Modeling yield response to crop management using convolutional neural networks. *Computers and Electronics in Agriculture*, 170:105197.

Baudron, F., Tittonell, P., Corbeels, M., Letourmy, P., and Giller, K. E. (2012). Comparative performance of conservation agriculture and current smallholder farming practices in semi-arid zimbabwe. *Field crops research*, 132:117–128.

Baudry, D., Gautron, R., Kaufmann, E., and Maillard, O. (2021a). Optimal thompson sampling strategies for support-aware cvar bandits. In *International Conference on Machine Learning*, pages 716–726. PMLR.

Baudry, D., Russac, Y., and Cappé, O. (2021b). On limited-memory subsampling strategies for bandits. In *International Conference on Machine Learning*, pages 727–737. PMLR.

Bellinger, C., Coles, R., Crowley, M., and Tamblyn, I. (2020). Active measure reinforcement learning for observation cost minimization. *arXiv preprint arXiv:2005.12697*.

Bellman, R. (1957). Dynamic programming. *Princeton, USA: Princeton University Press*, 1(2):3.

Bennett, J., Mutti, L., Rao, P., and Jones, J. (1989). Interactive effects of nitrogen and water stresses on biomass accumulation, nitrogen uptake, and seed yield of maize. *Field Crops Research*, 19(4):297–311.

Bentkus, V. (2004). On Hoeffding's inequalities. *The Annals of Probability*, 32(2):1650 – 1673.

Bergez, J., Eigenraam, M., and Garcia, F. (2001). Comparison between dynamic programming and reinforcement learning: A case study on maize irrigation management. In *Proceedings of the 3rd European Conference on Information Technology in Agriculture (EFITA01), Montpellier (FR) pp*, pages 343–348. Citeseer.

Berti, A., Marta, A. D., Mazzoncini, M., and Tei, F. (2016). An overview on long-term agro-ecosystem experiments: Present situation and future potential. *European Journal of Agronomy*, 77:236–241.

Bertsekas, D. P. and Shreve, S. E. (1996). *Stochastic optimal control: the discrete-time case*, volume 5. Athena Scientific.

Bertsekas, D. P. and Tsitsiklis, J. N. (1996). *Neuro-dynamic programming*. Athena Scientific.

Binas, J., Luginbuehl, L., and Bengio, Y. (2019). Reinforcement learning for sustainable agriculture. In *ICML Workshop Climate Change: How Can AI Help?*

Binder, D. L., Sander, D. H., and Walters, D. T. (2000). Maize response to time of nitrogen application as affected by level of nitrogen deficiency. *Agronomy Journal*, 92(6):1228–1236.

Boiffin, I., Malezieuxz, E., and Picard, D. (2001). Cropping systems for the 1 6 future. *Crop science: Progress and prospects*, page 261.

Boote, K. J., Jones, J. W., and Pickering, N. B. (1996). Potential uses and limitations of crop models. *Agronomy journal*, 88(5):704–716.

Boucheron, S., Lugosi, G., and Massart, P. (2013). *Concentration inequalities: A nonasymptotic theory of independence*. Oxford university press.

Boyabatlı, O., Nasiry, J., and Zhou, Y. (2019). Crop planning in sustainable agriculture: Dynamic farmland allocation in the presence of crop rotation benefits. *Management Science*, 65(5):2060–2076.

Brisson, N., Gary, C., Justes, E., Roche, R., Mary, B., Ripoche, D., Zimmer, D., Sierra, J., Bertuzzi, P., Burger, P., et al. (2003). An overview of the crop model stics. *European Journal of agronomy*, 18(3-4):309–332.

Bu, F. and Wang, X. (2019). A smart agriculture iot system based on deep reinforcement learning. *Future Generation Computer Systems*, 99:500–507.

Burda, Y., Edwards, H., Storkey, A., and Klimov, O. (2018). Exploration by random network distillation. *arXiv preprint arXiv:1810.12894*.

Burt, O. R. and Allison, J. R. (1963). Farm management decisions with dynamic programming. *Journal of Farm Economics*, 45(1):121–136.

Camargo, G. G. and Kemanian, A. R. (2016). Six crop models differ in their simulation of water uptake. *Agricultural and forest meteorology*, 220:116–129.

Cao, X.-R. (2008). Limitation of markov models and event-based learning and optimization. In *2008 Chinese Control and Decision Conference*, pages 14–17. IEEE.

Casella, G. and Berger, R. L. (2021). *Statistical inference*. Cengage Learning.

Cassel, A., Mannor, S., and Zeevi, A. (2018). A general approach to multi-armed bandits under risk criteria. In *Conference On Learning Theory*, pages 1295–1306. PMLR.

Cassman, K. G., Dobermann, A., and Walters, D. T. (2002). Agroecosystems, nitrogen-use efficiency, and nitrogen management. *AMBIO: A Journal of the Human Environment*, 31(2):132–140.

Cawley, G. C. and Talbot, N. L. (2010). On over-fitting in model selection and subsequent selection bias in performance evaluation. *The Journal of Machine Learning Research*, 11:2079–2107.

Cerf, M. and Meynard, J.-M. (2006). Les outils de pilotage des cultures: diversité de leurs usages et enseignements pour leur conception. *Natures Sciences Sociétés*, 14(1):19–29.

Cerf, M. and Sebillotte, M. (1988). Le concept de modele general et la prise de decision dans la conduite d'une culture. *Comptes Rendus de l'Académie d'Agriculture de France 4 (74), 71-80.(1988)*.

Cerf, M. and Sebillotte, M. (1997). Approche cognitive des décisions de production dans l'exploitation agricole [confrontation aux théories de la décision]. *Economie rurale*, 239(1):11–18.

Chapagain, R., Remenyi, T. A., Harris, R. M., Mohammed, C. L., Huth, N., Wallach, D., Rezaei, E. E., and Ojeda, J. J. (2022). Decomposing crop model uncertainty: A systematic review. *Field Crops Research*, 279:108448.

Charlton, P. and Street, P. (1975). practical application of bioeconomic models. *Study of Agricultural Systems. GE Dalton, ed*.

Chatelin, M.-H., Aubry, C., Leroy, P., Papy, F., and Poussin, J.-C. (1993). Pilotage de la production et aide à la décision stratégique: le cas des exploitations en grande culture. *Cahiers d'Economie et de Sociologie Rurales (CESR)*, 28(905-2016-70228):119–138.

Chatelin, M.-H., Aubry, C., Poussin, J.-C., Meynard, J.-M., Massé, J., Verjux, N., Gate, P., and Le Bris, X. (2005). Déciblé, a software package for wheat crop management simulation. *Agricultural Systems*, 83(1):77–99.

Chen, M., Cui, Y., Wang, X., Xie, H., Liu, F., Luo, T., Zheng, S., and Luo, Y. (2021). A reinforcement learning approach to irrigation decision-making for rice using weather forecasts. *Agricultural Water Management*, 250:106838.

Chimonyo, V. G. P., Modi, A. T., and Mabhaudhi, T. (2015). Perspective on crop modeling in the management of intercropping systems. *Archives of Agronomy and Soil Science*, 61(11):1511–1529.

Cook, B. I., Mankin, J. S., and Anchukaitis, K. J. (2018). Climate change and drought: From past to future. *Current Climate Change Reports*, 4(2):164–179.

Corbeels, M., Chirat, G., Messad, S., and Thierfelder, C. (2016). Performance and sensitivity of the dssat crop growth model in simulating maize yield under conservation agriculture. *European journal of agronomy*, 76:41–53.

Coulom, R. (2006). Efficient selectivity and backup operators in monte-carlo tree search. In *International conference on computers and games*, pages 72–83. Springer.

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, 2(4):303–314.

da Silva, C. G., Figueira, J., Lisboa, J., and Barman, S. (2006). An interactive decision support system for an aggregate production planning model based on multiple criteria mixed integer linear programming. *Omega*, 34(2):167–177.

d'Agostino, R. and Pearson, E. S. (1973). Tests for departure from normality. empirical results for the distributions of b^2 and $\sqrt{b}$. *Biometrika*, 60(3):613–622.

d'Agostino, R. B. (1971). An omnibus test of normality for moderate and large size samples. *Biometrika*, 58(2):341–348.

Dasgupta, I., Wang, J., Chiappa, S., Mitrovic, J., Ortega, P., Raposo, D., Hughes, E., Battaglia, P., Botvinick, M., and Kurth-Nelson, Z. (2019). Causal reasoning from meta-reinforcement learning. *arXiv preprint arXiv:1901.08162*.

Deffontaines, J.-P. and Petit, M. (1985). *Comment étudier les exploitations agricoles d'une région: présentation d'un ensemble méthodologique*. INRA.

Deisenroth, M. and Rasmussen, C. E. (2011). Pilco: A model-based and data-efficient approach to policy search. In *Proceedings of the 28th International Conference on machine learning (ICML-11)*, pages 465–472.

Della Penna, N., Reid, M. D., and Balduzzi, D. (2016). Compliance-aware bandits. *arXiv preprint arXiv:1602.02852*.

Ding, Y., Wang, L., Li, Y., and Li, D. (2018). Model predictive control and its application in agriculture: A review. *Computers and Electronics in Agriculture*, 151:104–117.

Donatelli, M., Magarey, R. D., Bregaglio, S., Willocquet, L., Whish, J. P., and Savary, S. (2017). modeling the impacts of pests and diseases on agricultural systems. *Agricultural systems*, 155:213–224.

Doré, T., Martin, P., Le Bail, M., Ney, B., and Roger-Estrade, J. (2006). *L'agronomie aujourd'hui*. Editions Quae.

Dorier, M., Antoniu, G., Cappello, F., Snir, M., Sisneros, R., Yildiz, O., Ibrahim, S., Peterka, T., and Orf, L. (2016). Damaris: Addressing performance variability in data management for post-petascale simulations. *ACM Transactions on Parallel Computing (TOPC)*, 3(3):1–43.

Dowd, K. (2007). *Measuring market risk*. John Wiley & Sons.

Dulac-Arnold, G., Mankowitz, D., and Hester, T. (2019). Challenges of real-world reinforcement learning. *arXiv preprint arXiv:1904.12901*.

Durand, A., Achilleos, C., Iacovides, D., Strati, K., Mitsis, G. D., and Pineau, J. (2018). Contextual bandits for adapting treatment in a mouse model of de novo carcinogenesis. In *Machine learning for healthcare conference*, pages 67–82. PMLR.

Duru, M., Therond, O., Martin, G., Martin-Clouaire, R., Magne, M.-A., Justes, E., Journet, E.-P., Aubertot, J.-N., Savary, S., Bergez, J.-E., et al. (2015). How to implement biodiversity-based agriculture to enhance ecosystem services: a review. *Agronomy for sustainable development*, 35(4):1259–1281.

Dury, J., Schaller, N., Garcia, F., Reynaud, A., and Bergez, J. E. (2012). Models to support cropping plan and crop rotation decisions. a review. *Agronomy for sustainable development*, 32(2):567–580.

Edwards-Jones, G. (2006). modeling farmer decision-making: concepts, progress and challenges. *Animal science*, 82(6):783–790.

Efron, B. and Gong, G. (1983). A leisurely look at the bootstrap, the jackknife, and cross-validation. *The American Statistician*, 37(1):36–48.

Egli, D. and Bruening, W. (1992). Planting date and soybean yield: evaluation of environmental effects with a crop simulation model: Soygro. *Agricultural and Forest Meteorology*, 62(1-2):19–29.

Epperson, J. E., Hook, J. E., and Mustafa, Y. R. (1993). Dynamic programming for improving irrigation scheduling strategies of maize. *Agricultural Systems*, 42(1-2):85–101.

Evans, K. J., Terhorst, A., and Kang, B. H. (2017). From data to decisions: helping crop producers build their actionable knowledge. *Critical reviews in plant sciences*, 36(2):71–88.

Everitt, B. and Skrondal, A. (2002). *The Cambridge dictionary of statistics*, volume 106. Cambridge University Press Cambridge.

Falconnier, G., Corbeels, M., Boote, K., Adam, M., Basso, B., and Ruane, A. C. (2019). Model intercomparison of maize response to climate change in low-input smallholder cropping systems.[p179]. Elsevier.

Falconnier, G. N., Corbeels, M., Boote, K. J., Affholder, F., Adam, M., MacCarthy, D. S., Ruane, A. C., Nendel, C., Whitbread, A. M., Justes, É., et al. (2020). modeling climate change impacts on maize yields under low nitrogen input conditions in sub-saharan africa. *Global change biology*, 26(10):5942–5964.

Falconnier, G. N., Descheemaeker, K., Van Mourik, T. A., and Giller, K. E. (2016). Unravelling the causes of variability in crop yields and treatment responses for better tailoring of options for sustainable intensification in southern mali. *Field Crops Research*, 187:113–126.

FAO, F. et al. (2017). The future of food and agriculture–trends and challenges. *Annual Report*, 296:1–180.

Ferreira, L. and Murray, M. H. (1997). modeling rail track deterioration and maintenance: current practices and future needs. *Transport Reviews*, 17(3):207–221.

Fosu-Mensah, B., MacCarthy, D., Vlek, P., and Safo, E. (2012). Simulating impact of seasonal climatic variation on the response of maize (zea mays l.) to inorganic fertilizer in sub-humid ghana. *Nutrient cycling in agroecosystems*, 94(2):255–271.

Freund, R. J. (1956). The introduction of risk into a programming model. *Econometrica: Journal of the econometric society*, pages 253–263.

Garcia, F. (1999). Use of reinforcement learning and simulation to optimize wheat crop technical management. In *Proceedings of the International Congress on modeling and Simulation (MODSIM'99) Hamilton, New-Zealand*, pages 801–806.

Garcia, F. and Ndiaye, S. M. (1998). A learning rate analysis of reinforcement learning algorithms in finite-horizon. In *Proceedings of the 15th International Conference on Machine Learning (ML-98*. Citeseer.

Garcıa, J. and Fernández, F. (2015). A comprehensive survey on safe reinforcement learning. *Journal of Machine Learning Research*, 16(1):1437–1480.

Garivier, A. and Moulines, E. (2011). On upper-confidence bound policies for switching bandit problems. In *International Conference on Algorithmic Learning Theory*, pages 174–188. Springer.

Gasse, M., Grasset, D., Gaudron, G., and Oudeyer, P.-Y. (2021). Causal reinforcement learning using observational and interventional data. *arXiv preprint arXiv:2106.14421*.

Gautron, R. (2021). A bandit geared gym encapsulation of the DSSAT crop simulator. https://github.com/rgautron/DssatBanditEnv.

Gautron, R. (2022). gym-DSSAT: an easy to manipulate crop environment for reinforcement learning. https://gitlab.inria.fr/rgautron/gym_dssat_pdi.

Gautron, R., Maillard, O.-A., Preux, P., Corbeels, M., and Sabbadin, R. (2022a). Reinforcement learning for crop management support: Review, prospects and challenges. *Computers and Electronics in Agriculture*, 200:107182.

Gautron, R., Padrón, E. J., Preux, P., Bigot, J., Maillard, O.-A., and Emukpere, D. (2022b). gym-DSSAT: a crop model turned into a Reinforcement Learning environment. Research Report RR-9460, Inria Lille.

Gautron, R. and Padrón González, E. J. (2022). gym-DSSAT - A crop model turned into a Reinforcement Learning environment.

Gent, D. H., De Wolf, E., and Pethybridge, S. J. (2011). Perceptions of risk, risk aversion, and barriers to adoption of decision support systems and integrated pest management: an introduction. *Phytopathology*, 101(6):640–643.

Getnet, M., Van Ittersum, M., Hengsdijk, H., and Descheemaeker, K. (2016). Yield gaps and resource use across farming zones in the central rift valley of ethiopia. *Experimental Agriculture*, 52(4):493–517.

Gijsman, A. J., Thornton, P. K., and Hoogenboom, G. (2007). Using the wise database to parameterize soil inputs for crop simulation models. *Computers and Electronics in Agriculture*, 56(2):85–100.

Giller, K. E., Witter, E., Corbeels, M., and Tittonell, P. (2009). Conservation agriculture and smallholder farming in africa: the heretics' view. *Field crops research*, 114(1):23–34.

Glen, J. J. (1987). Mathematical models in farm planning: A survey. *Operations Research*, 35(5):641–666.

Godoy, W. F., Podhorszki, N., Wang, R., Atkins, C., Eisenhauer, G., Gu, J., Davis, P., Choi, J., Germaschewski, K., Huck, K., et al. (2020). Adios 2: The adaptable input output system. a framework for high-performance data management. *SoftwareX*, 12:100561.

Golemo, F., Taiga, A. A., Courville, A., and Oudeyer, P.-Y. (2018). Sim-to-real transfer with neural-augmented robot simulation. In *Conference on Robot Learning*, pages 817–828. PMLR.

Goulet, F., Pervanchon, F., Conteau, C., and Cerf, M. (2008). Les agriculteurs innovent par eux-mêmes pour leurs systèmes de culture. *R. Reau et T. Doré, Systèmes de culture innovant et durables. Dijon, educagri éditions*, pages 53–69.

Grünwald, P., de Heide, R., and Koolen, W. M. (2020). Safe testing. In *2020 Information Theory and Applications Workshop (ITA)*, pages 1–54. IEEE.

Guler, H. (2013). Decision support system for railway track maintenance and renewal management. *Journal of Computing in Civil Engineering*, 27(3):292–306.

Han, E., Ines, A. V., and Koo, J. (2019). Development of a 10-km resolution global soil profile dataset for crop modeling applications. *Environmental modeling & software*, 119:70–83.

Hanway, J. (1963). Growth stages of corn (zea mays, l.) 1. *Agronomy Journal*, 55(5):487–492.

Hao, Z., Singh, V. P., and Xia, Y. (2018). Seasonal drought prediction: advances, challenges, and future prospects. *Reviews of Geophysics*, 56(1):108–141.

Hartland, C., Gelly, S., Baskiotis, N., Teytaud, O., and Sebag, M. (2006). Multi-armed bandit, dynamic environments and meta-bandits.

Hayes, B. (2008). Cloud computing.

Hayes, C. F., Rădulescu, R., Bargiacchi, E., Källström, J., Macfarlane, M., Reymond, M., Verstraeten, T., Zintgraf, L. M., Dazeley, R., Heintz, F., et al. (2021). A practical guide to multi-objective reinforcement learning and planning. *arXiv preprint arXiv:2103.09568*.

He, J., Dukes, M. D., Hochmuth, G. J., Jones, J. W., and Graham, W. D. (2012). Identifying irrigation and nitrogen best management practices for sweet corn production on sandy soils using ceres-maize model. *Agricultural Water Management*, 109:61–70.

Hébert, J. (1969). La fumure azotée du blé tendre d'hiver. *Bull Tech Inf*, 244:755–766.

Hester, T., Vecerik, M., Pietquin, O., Lanctot, M., Schaul, T., Piot, B., Horgan, D., Quan, J., Sendonaris, A., Osband, I., et al. (2018). Deep q-learning from demonstrations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Hildreth, C. (1957). Problems of uncertainty in farm planning. *Journal of Farm Economics*, 39(5):1430–1441.

Hill, A., Raffin, A., Ernestus, M., Gleave, A., Kanervisto, A., Traore, R., Dhariwal, P., Hesse, C., Klimov, O., Nichol, A., Plappert, M., Radford, A., Schulman, J., Sidor, S., and Wu, Y. (2018). Stable baselines. https://github.com/hill-a/stable-baselines.

Hintjens, P. (2013). *ZeroMQ: messaging for many applications*. O'Reilly Media, Inc.

Hoc, J. and Rogalski, J. (1992). Régulation des activités cognitives et gestion du risque par l'opérateur humain. *Les rationalisations de la production. Toulouse: CEPADUES*, pages 147–168.

Hochman, Z. and Carberry, P. (2011). Emerging consensus on desirable characteristics of tools to support farmers' management of climate risk in australia. *Agricultural Systems*, 104(6):441–450.

Hochman, Z., Van Rees, H., Carberry, P., Hunt, J., McCown, R., Gartmann, A., Holzworth, D., Van Rees, S., Dalgliesh, N., Long, W., et al. (2009). Re-inventing model-based decision support with australian dryland farmers. 4. yield prophet® helps farmers monitor and manage crops in a variable climate. *Crop and Pasture Science*, 60(11):1057–1070.

Hoeffding, W. (1948). A Class of Statistics with Asymptotically Normal Distribution. *The Annals of Mathematical Statistics*, 19(3):293 – 325.

Hofmann, T., Schölkopf, B., and Smola, A. J. (2008). Kernel methods in machine learning. *The annals of statistics*, 36(3):1171–1220.

Holzinger, A., Langs, G., Denk, H., Zatloukal, K., and Müller, H. (2019). Causability and explainability of artificial intelligence in medicine. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 9(4):e1312.

Holzkämper, A., Calanca, P., and Fuhrer, J. (2013). Identifying climatic limitations to grain maize yield potentials using a suitability evaluation approach. *Agricultural and Forest Meteorology*, 168:149–159.

Hoogenboom, G. (2000). Contribution of agrometeorology to the simulation of crop production and its applications. *Agricultural and forest meteorology*, 103(1-2):137–157.

Hoogenboom, G., Porter, C., Boote, K., Shelia, V., Wilkens, P., Singh, U., White, J., Asseng, S., Lizaso, J., Moreno, L., et al. (2019). The dssat crop modeling ecosystem. *Advances in crop modeling for a sustainable agriculture*, pages 173–216.

Hoogland, J., Feddes, R. A., and Belmans, C. (1981). Root water uptake model depending on soil water pressure head and maximum extraction rate. In *III International Symposium on Water supply and Irrigation in the open and under Protected Cultivation 119*, pages 123–136.

Howell, T. A. (2003). Irrigation efficiency. *Encyclopedia of water science*, 467:500.

Huet, E. (2022). No title. *Forthcoming*.

Huet, E., Adam, M., Giller, K. E., and Descheemaeker, K. (2020). Diversity in perception and management of farming risks in southern mali. *Agricultural Systems*, 184:102905.

Huet, E., Adam, M., Traore, B., Giller, K., and Descheemaeker, K. (2022). Coping with cereal production risks due to the vagaries of weather, labour shortages and input markets through management in southern mali. *European Journal of Agronomy*, 140:126587.

Husson, O., Sarthou, J.-P., Bousset, L., Ratnadass, A., Schmidt, H.-P., Kempf, J., Husson, B., Tingry, S., Aubertot, J.-N., Deguine, J.-P., Goebel, F.-R., and Lamichhane, J. R. (2021). Soil and plant health in relation to dynamic sustainment of eh and ph homeostasis: A review. *Plant and Soil*.

Ip, R. H., Ang, L.-M., Seng, K. P., Broster, J., and Pratley, J. (2018). Big data and machine learning for crop protection. *Computers and Electronics in Agriculture*, 151:376–383.

Iyer, P., Bozzola, M., Hirsch, S., Meraner, M., and Finger, R. (2020). Measuring farmer risk preferences in europe: a systematic review. *Journal of Agricultural Economics*, 71(1):3–26.

Jarque, C. M. and Bera, A. K. (1980). Efficient tests for normality, homoscedasticity and serial independence of regression residuals. *Economics letters*, 6(3):255–259.

JCGM et al. (2008). Evaluation of measurement data—guide to the expression of uncertainty in measurement. *Int. Organ. Stand. Geneva ISBN*, 50:134.

Jones, J. W., Antle, J. M., Basso, B., Boote, K. J., Conant, R. T., Foster, I., Godfray, H. C. J., Herrero, M., Howitt, R. E., Janssen, S., et al. (2017). Brief history of agricultural systems modeling. *Agricultural systems*, 155:240–254.

Jones, J. W., He, J., Boote, K. J., Wilkens, P., Porter, C., and Hu, Z. (2011). Estimating dssat cropping system cultivar-specific parameters using bayesian techniques. *Methods of introducing system models into agricultural research*, 2:365–393.

Joshi, N., Singh, A. K., and Madramootoo, C. A. (2017). Application of dssat model to simulate corn yield under long-term tillage and residue practices. *Transactions of the ASABE*, 60(1):67–83.

Jourdain, D., Lairez, J., Striffler, B., and Affholder, F. (2020). Farmers' preference for cropping systems and the development of sustainable intensification: a choice experiment approach. *Review of Agricultural, Food and Environmental Studies*, 101(4):417–437.

Kadam, N. N., Xiao, G., Melgar, R. J., Bahuguna, R. N., Quinones, C., Tamilselvan, A., Prasad, P. V. V., and Jagadish, K. S. (2014). Agronomic and physiological responses to high temperature, drought, and elevated co2 interactions in cereals. *Advances in agronomy*, 127:111–156.

Kalaji, H. M., Dabrowski, P., Cetner, M. D., Samborska, I. A., Lukasik, I., Brestic, M., Zivcak, M., Tomasz, H., Mojski, J., Kociel, H., et al. (2017). A comparison between different chlorophyll content meters under nutrient deficiency conditions. *Journal of Plant Nutrition*, 40(7):1024–1034.

Kamara, A., Menkir, A., Badu-Apraku, B., and Ibikunle, O. (2003). The influence of drought stress on growth, yield and yield components of selected maize genotypes. *The journal of agricultural science*, 141(1):43–50.

Keller, B., Draelos, M., Zhou, K., Qian, R., Kuo, A. N., Konidaris, G., Hauser, K., and Izatt, J. A. (2020). Optical coherence tomography-guided robotic ophthalmic microsurgery via reinforcement learning from demonstration. *IEEE Transactions on Robotics*, 36(4):1207–1218.

Kendall, M. G. (1938). A new measure of rank correlation. *Biometrika*, 30(1/2):81–93.

Kennedy, J. O. (1986). *Dynamic programming: applications to agriculture and natural resources*. Springer Science & Business Media.

Khaliq, A., Javed, M., Sohail, M., and Sagheer, M. (2014). Environmental effects on insects and their population dynamics. *Journal of Entomology and Zoology studies*, 2(2):1–7.

Kirschner, J. and Krause, A. (2019). Stochastic bandits with context distributions. *arXiv preprint arXiv:1906.02685*.

Kocsis, L. and Szepesvári, C. (2006). Bandit based monte-carlo planning. In *European conference on machine learning*, pages 282–293. Springer.

Krause, A. and Ong, C. (2011). Contextual gaussian process bandit optimization. *Advances in neural information processing systems*, 24.

Krueger, D., Leike, J., Evans, O., and Salvatier, J. (2020). Active reinforcement learning: Observing rewards at a cost. *arXiv preprint arXiv:2011.06709*.

Kuchibhotla, A. K. and Zheng, Q. (2021). Near-optimal confidence sequences for bounded random variables. In Meila, M. and Zhang, T., editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 5827–5837. PMLR.

Kushner, H. J. (1967). Stochastic stability and control. Technical report, Brown Univ Providence RI.

Laird, N. M. and Ware, J. H. (1982). Random-effects models for longitudinal data. *Biometrics*, pages 963–974.

Lapan, M. (2018). *Deep Reinforcement Learning Hands-On: Apply modern RL methods, with deep Q-networks, value iteration, policy gradients, TRPO, AlphaGo Zero and more*. Packt Publishing Ltd.

Lattimore, T. and Szepesvári, C. (2020). *Bandit algorithms*. Cambridge University Press.

Laud, A. D. (2004). *Theory and application of reward shaping in reinforcement learning*. University of Illinois at Urbana-Champaign.

Le Gal, P.-Y., Merot, A., Moulin, C.-H., Navarrete, M., and Wery, J. (2010). A modeling framework to support farmers in designing agricultural production systems. *Environmental modeling & Software*, 25(2):258–268.

Lemmon, H. (1986). Comax: An expert system for cotton crop management. *Science*, 233(4759):29–33.

Leurent, E. (2020). *Safe and Efficient Reinforcement Learning for Behavioural Planning in Autonomous Driving*. PhD thesis, Université de Lille.

Levine, S., Kumar, A., Tucker, G., and Fu, J. (2020). Offline reinforcement learning: Tutorial, review, and perspectives on open problems. *arXiv preprint arXiv:2005.01643*.

Li, K., Boisvert, J., and Jong, R. D. (1999). An exponential root-water-uptake model. *Canadian Journal of soil science*, 79(2):333–343.

Li, Y. (2019). Reinforcement learning applications. *arXiv preprint arXiv:1908.06973*.

Li, Y., Guan, K., Schnitkey, G. D., DeLucia, E., and Peng, B. (2019). Excessive rainfall leads to maize yield loss of a comparable magnitude to extreme drought in the united states. *Global change biology*, 25(7):2325–2337.

Liakos, K. G., Busato, P., Moshou, D., Pearson, S., and Bochtis, D. (2018). Machine learning in agriculture: A review. *Sensors*, 18(8):2674.

Liang, E., Liaw, R., Nishihara, R., Moritz, P., Fox, R., Gonzalez, J., Goldberg, K., and Stoica, I. (2017). Ray rllib: A composable and scalable reinforcement learning library. *arXiv preprint arXiv:1712.09381*, page 85.

Littman, M. L., Sutton, R. S., and Singh, S. P. (2001). Predictive representations of state. In *NIPS*, volume 14, page 30.

Liu, C., Xu, X., and Hu, D. (2014). Multiobjective reinforcement learning: A comprehensive overview. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 45(3):385–398.

Liu, F., Lee, J., and Shroff, N. (2018). A change-detection based framework for piecewise-stationary multi-armed bandit problem. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Ljung, G. M. and Box, G. E. (1978). On a measure of lack of fit in time series models. *Biometrika*, 65(2):297–303.

Lowe, T. J. and Preckel, P. V. (2004). Decision technologies for agribusiness problems: A brief review of selected literature and a call for research. *Manufacturing & Service Operations Management*, 6(3):201–208.

Madumal, P., Miller, T., Sonenberg, L., and Vetere, F. (2020). Explainable reinforcement learning through a causal lens. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 2493–2500.

Maillard, O.-A. (2019). *Mathematics of Statistical Sequential Decision Making*. Habilitation à diriger des recherches, Université de Lille Nord de France.

Maiorano, A., Martre, P., Asseng, S., Ewert, F., Müller, C., Rötter, R. P., Ruane, A. C., Semenov, M. A., Wallach, D., Wang, E., et al. (2017). Crop model improvement reduces the uncertainty of the response to temperature of multi-model ensembles. *Field Crops Research*, 202:5–20.

Mandelbrot, B. B. (1997). The variation of certain speculative prices. In *Fractals and scaling in finance*, pages 371–418. Springer.

Mankowitz, D. J., Calian, D. A., Jeong, R., Paduraru, C., Heess, N., Dathathri, S., Riedmiller, M., and Mann, T. (2020). Robust constrained reinforcement learning for continuous control with model misspecification. *arXiv preprint arXiv:2010.10644*.

Manos, B. D., Ciani, A., Bournaris, T., Vassiliadou, I., and Papathanasiou, J. (2004). A taxonomy survey of decision support systems in agriculture. *Agricultural Economics Review*, 5(389-2016-23416):80–94.

Markowitz, H. (1952). March 1952. portfolio selection. *Journal of finance*, 7(1):77–91.

Massart, P. (1990). The tight constant in the dvoretzky-kiefer-wolfowitz inequality. *Annals of Probability*, 18.

Maurer, A. and Pontil, M. (2009). Empirical bernstein bounds and sample variance penalization. *arXiv preprint arXiv:0907.3740*.

McCown, R. (2012). A cognitive systems framework to inform delivery of analytic support for farmers' intuitive management under seasonal climatic variability. *Agricultural Systems*, 105(1):7–20.

McCown, R., Brennan, L., and Parton, K. (2006). Learning from the historical failure of farm management models to aid management practice. part 1. the rise and demise of theoretical models of farm economics. *Australian Journal of Agricultural Research*, 57(2):143–156.

McCown, R., Hammer, G., Hargreaves, J., Holzworth, D., and Huth, N. (1995). Apsim: an agricultural production system simulation model for operational research. *Mathematics and computers in simulation*, 39(3-4):225–231.

McCown, R. and Parton, K. (2006). Learning from the historical failure of farm management models to aid management practice. part 2. three systems approaches. *Australian Journal of Agricultural Research*, 57(2):157–172.

McCown, R. L. (2002a). Changing systems for supporting farmers' decisions: problems, paradigms, and prospects. *Agricultural systems*, 74(1):179–220.

McCown, R. L. (2002b). Locating agricultural decision support systems in the troubled past and socio-technical complexity of 'models for management'. *Agricultural systems*, 74(1):11–25.

Meier, U. (1997). *Growth stages of mono-and dicotyledonous plants*. Blackwell Wissenschafts-Verlag.

Meisinger, J. J. and Delgado, J. A. (2002). Principles for managing nitrogen leaching. *Journal of soil and water conservation*, 57(6):485–498.

Mellor, J. and Shapiro, J. (2013). Thompson sampling in switching environments with bayesian online change point detection. *arXiv preprint arXiv:1302.3721*.

Menapace, L., Colson, G., and Raffaelli, R. (2013). Risk aversion, subjective beliefs, and farmer risk management strategies. *American Journal of Agricultural Economics*, 95(2):384–389.

Mercer, W. and Hall, A. (1911). The experimental error of field trials. *The Journal of Agricultural Science*, 4(2):107–132.

Mertz, O., Mbow, C., Reenberg, A., Genesio, L., Lambin, E. F., D'haen, S., Zorom, M., Rasmussen, K., Diallo, D., Barbier, B., et al. (2011). Adaptation strategies and climate vulnerability in the sudano-sahelian region of west africa. *Atmospheric Science Letters*, 12(1):104–108.

Meurdesoif, Y., Ozdoba, H., Caubel, A., and Marti, O. (2013). Xios. In *Second Workshop on Coupling Technologies for Earth System Models (CW2013), NCAR, Boulder, CO, USA, available at: http://forge.ipsl.jussieu.fr/ioserver/raw-attachment/wiki/WikiStart/XIOS-BOULDER.pdf (last access: 5 August 2021)*.

Miller, R. A. (2016). Diagnostic decision support systems. In *Clinical decision support systems*, pages 181–208. Springer.

Milleville, P. (1987). Recherches sur les pratiques des agriculteurs. *Les cahiers de la Recherche Développement*, 16:3–7.

Mitchell, T. M. et al. (1997). Machine learning. 1997. *Burr Ridge, IL: McGraw Hill*, 45(37):870–877.

Mitton, N., Brossard, L., Bouadi, T., Garcia, F., Gautron, R., Hilgert, N., Ienco, D., Largouët, C., Lutton, E., Masson, V., et al. (2022). Foundations and state of play.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. (2015). Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533.

Modak, J. M. (2002). Haber process for ammonia synthesis. *Resonance*, 7(9):69–77.

Morris, T. F., Murrell, T. S., Beegle, D. B., Camberato, J. J., Ferguson, R. B., Grove, J., Ketterings, Q., Kyveryga, P. M., Laboski, C. A., McGrath, J. M., et al. (2018). Strengths and limitations of nitrogen rate recommendations for corn and opportunities for improvement. *Agronomy Journal*, 110(1):1.

Mueller, N. D., Gerber, J. S., Johnston, M., Ray, D. K., Ramankutty, N., and Foley, J. A. (2012). Closing yield gaps through nutrient and water management. *Nature*, 490(7419):254–257.

Munos, R. (1996). A convergent reinforcement learning algorithm in the continuous case : the finite-element reinforcement learning. In *International Conference on Machine Learning*. Morgan Kaufmann.

Naudin, K., Gozé, E., Balarabe, O., Giller, K. E., and Scopel, E. (2010). Impact of no tillage and mulching practices on cotton production in north cameroon: a multi-locational on-farm assessment. *Soil and Tillage Research*, 108(1-2):68–76.

Navarro-Hellín, H., Martinez-del Rincon, J., Domingo-Miguel, R., Soto-Valles, F., and Torres-Sánchez, R. (2016). A decision support system for managing irrigation in agriculture. *Computers and Electronics in Agriculture*, 124:121–131.

Ndiaye, S. M. (1999). *Apprentissage par renforcement en horizon fini: application à la génération de règles pour la Conduite de Culture*. PhD thesis, Toulouse 3.

Nelson, R., Holzworth, D., Hammer, G., and Hayman, P. (2002). Infusing the use of seasonal climate forecasting into crop management practice in north east australia using discussion support software. *Agricultural Systems*, 74(3):393–414.

NeSmith, D. and Ritchie, J. (1992). Short-and long-term responses of corn to a pre-anthesis soil water deficit. *Agronomy journal*, 84(1):107–113.

Ng, A. Y., Harada, D., and Russell, S. (1999). Policy invariance under reward transformations: Theory and application to reward shaping. In *Icml*, volume 99, pages 278–287.

Norton, R. D. and Hazell, P. B. (1986). *Mathematical programming for economic analysis in agriculture*. Macmillan New York, NY, USA.

Opitz, D. and Maclin, R. (1999). Popular ensemble methods: An empirical study. *Journal of artificial intelligence research*, 11:169–198.

Osborne, M. J. et al. (2004). *An introduction to game theory*, volume 3. Oxford university press New York.

Overweg, H., Berghuijs, H. N., and Athanasiadis, I. N. (2021). Cropgym: a reinforcement learning environment for crop management. *arXiv preprint arXiv:2104.04326*.

Öztuna, D., Elhan, A. H., and Tüccar, E. (2006). Investigation of four different normality tests in terms of type 1 error rate and power under different distributions. *Turkish Journal of Medical Sciences*, 36(3):171–176.

Padakandla, S., Prabuchandran, K., and Bhatnagar, S. (2020). Reinforcement learning algorithm for non-stationary environments. *Applied Intelligence*, 50(11):3590–3606.

Papy, F. (1998). Savoir pratique sur les systèmes techniques et aide à la décision. *La conduite du champ cultivé. Points de vue d'agronomes. IRD*, pages 245–259.

Pashler, H. and Wagenmakers, E.-J. (2012). Editors' introduction to the special section on replicability in psychological science: A crisis of confidence? *Perspectives on psychological science*, 7(6):528–530.

Perchet, V., Rigollet, P., Chassang, S., and Snowberg, E. (2015). Batched bandit problems. In Grünwald, P., Hazan, E., and Kale, S., editors, *Proceedings of The 28th Conference on Learning Theory, COLT 2015, Paris, France, July 3-6, 2015*, volume 40 of *JMLR Workshop and Conference Proceedings*, page 1456. JMLR.org.

Phan, M., Thomas, P., and Learned-Miller, E. (2021). Towards practical mean bounds for small samples. In Meila, M. and Zhang, T., editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8567–8576. PMLR.

Pineau, J., Vincent-Lamarre, P., Sinha, K., Lariviere, V., Beygelzimer, A., d'Alche Buc, F., Fox, E., and Larochelle, H. (2021). Improving reproducibility in machine learning research (a report from the neurips 2019 reproducibility program). *Journal of Machine Learning Research*, 22(164):1–20.

Power, D. J. (2008). Decision support systems: a historical overview. In *Handbook on decision support systems 1*, pages 121–140. Springer.

Pretty, J., Toulmin, C., and Williams, S. (2011). Sustainable intensification in african agriculture. *International journal of agricultural sustainability*, 9(1):5–24.

Puterman, M. L. (1994). *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons.

Quinton, E., Pignol, C., Linyer, H., Ancelin, J., Cipière, S., Heintz, W., Rouan, M., Damy, S., Bretagnolle, V., et al. (2019). Towards better traceability of field sampling data. *Computers & Geosciences*, 129:82–91.

Randløv, J. and Alstrøm, P. (1998). Learning to drive a bicycle using reinforcement learning and shaping. In *ICML*, volume 98, pages 463–471. Citeseer.

Ravichandar, H., Polydoros, A. S., Chernova, S., and Billard, A. (2020). Recent advances in robot learning from demonstration. *Annual Review of Control, Robotics, and Autonomous Systems*, 3:297–330.

Richardson, C. (1985). Weather simulation for crop management models. *Transactions of the ASAE*, 28(5):1602–1606.

Richardson, C. W. and Wright, D. A. (1984). Wgen: A model for generating daily weather variables. *ARS (USA)*.

Ripoche, A., Crétenet, M., Corbeels, M., Affholder, F., Naudin, K., Sissoko, F., Douzet, J.-M., and Tittonell, P. (2015). Cotton as an entry point for soil fertility maintenance and food crop productivity in savannah agroecosystems–evidence from a long-term experiment in southern mali. *Field crops research*, 177:37–48.

Robbins, H. (1952). Some aspects of the sequential design of experiments. *Bulletin of the American Mathematical Society*, 58(5):527–535.

Rockafellar, R. T., Uryasev, S., et al. (2000). Optimization of conditional value-at-risk. *Journal of risk*, 2:21–42.

Roese, N. J. (1997). Counterfactual thinking. *Psychological bulletin*, 121(1):133.

Rose, D. C., Sutherland, W. J., Parker, C., Lobley, M., Winter, M., Morris, C., Twining, S., Ffoulkes, C., Amano, T., and Dicks, L. V. (2016). Decision support tools for agriculture: Towards effective design and delivery. *Agricultural systems*, 149:165–174.

Rose, K., Eldridge, S., and Chapin, L. (2015). The internet of things: An overview. *The internet society (ISOC)*, 80:1–50.

Roussel, C., Keller, K., Gaalich, M., Gomez, L. B., and Bigot, J. (2017). PDI, an approach to decouple I/O concerns from high-performance simulation codes. Working paper.

Roux, S., Brun, F., and Wallach, D. (2014). Combining input uncertainty and residual error in crop model predictions: A case study on vineyards. *European Journal of Agronomy*, 52:191–197.

Royce, F., Jones, J., and Hansen, J. (2001). Model–based optimization of crop management for climate forecast applications. *Transactions of the ASAE*, 44(5):1319.

Sabzi, S., Abbaspour-Gilandeh, Y., and Garcia-Mateos, G. (2018). A fast and accurate expert system for weed identification in potato crops using metaheuristic algorithms. *Computers in Industry*, 98:80–89.

Saikai, Y., Patel, V., and Mitchell, P. D. (2020). Machine learning for optimizing complex site-specific management. *Computers and Electronics in Agriculture*, 174:105381.

Sanchez, P. A., Palm, C. A., and Buol, S. W. (2003). Fertility capability soil classification: a tool to help assess soil quality in the tropics. *Geoderma*, 114(3-4):157–185.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. (2017). Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Schwartz, A. (1993). A reinforcement learning method for maximizing undiscounted rewards. In *Proceedings of the tenth international conference on machine learning*, volume 298, pages 298–305.

Sebillotte, M. (1974). Agronomie et agriculture. essai d'analyse des tâches de l'agronome. *Cahiers Orstom, série biologie*, 24:3–25.

Sebillotte, M. (1978). Itinéraires techniques et évolution de la pensée agronomique. *CR Acad. Agric. Fr*, 64(11):906–914.

Sebillotte, M. and Soler, L. G. (1988). Le concept de modele general et la comprehension du comportement de l'agriculteur.

Semenov, M. A. and Porter, J. (1995). Climatic variability and the modeling of crop yields. *Agricultural and forest meteorology*, 73(3-4):265–283.

Shapiro, S. S. and Wilk, M. B. (1965). An analysis of variance test for normality (complete samples). *Biometrika*, 52(3/4):591–611.

Sharpe, W. F. (1966). Mutual fund performance. *The Journal of business*, 39(1):119–138.

Shibu, M. E., Leffelaar, P. A., Van Keulen, H., and Aggarwal, P. K. (2010). Lintul3, a simulation model for nitrogen-limited situations: Application to rice. *European Journal of Agronomy*, 32(4):255–271.

Shrestha, R., Arnaud, E., Mauleon, R., Senger, M., Davenport, G. F., Hancock, D., Morrison, N., Bruskiewich, R., and McLaren, G. (2010). Multifunctional crop trait ontology for breeders' data: field book, annotation, data discovery and semantic enrichment of the literature. *AoB plants*, 2010.

Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., Hubert, T., Baker, L., Lai, M., Bolton, A., et al. (2017). Mastering the game of go without human knowledge. *Nature*, 550(7676):354.

Simon, H. A. (1976). From substantive to procedural rationality. In *25 years of economic theory*, pages 65–86. Springer.

Snedecor, G. W. and Cochran, W. G. (1989). Statistical methods, eight edition. *Iowa state University press, Ames, Iowa*, 1191.

Sokolowski, J. A. and Banks, C. M. (2012). *Handbook of real-world applications in modeling and simulation*, volume 2. John Wiley & Sons.

Soler, C. M. T., Sentelhas, P. C., and Hoogenboom, G. (2007). Application of the csm-ceres-maize model for planting date evaluation and yield forecasting for maize grown off-season in a subtropical environment. *European Journal of Agronomy*, 27(2-4):165–177.

Soltani, A. and Hoogenboom, G. (2003). A statistical comparison of the stochastic weather generators wgen and simmeteo. *Climate Research*, 24(3):215–230.

Soltani, A. and Hoogenboom, G. (2007). Assessing crop management options with crop simulation models based on generated weather data. *Field Crops Research*, 103(3):198–207.

Sønderskov, M., Rydahl, P., Bøjer, O. M., Jensen, J. E., and Kudsk, P. (2016). Crop protection online—weeds: a case study for agricultural decision support systems. In *Real-World Decision Support Systems*, pages 303–320. Springer.

Spaan, M. T. (2012). Partially observable markov decision processes. In *Reinforcement Learning*, pages 387–414. Springer.

Stanford, G. (1973). Rationale for optimum nitrogen fertilization in corn production. *Journal of Environmental Quality*, 2(2):159–166.

Stone, M. (1974). Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society: Series B (Methodological)*, 36(2):111–133.

Sultan, B., Roudier, P., Quirion, P., Alhassane, A., Muller, B., Dingkuhn, M., Ciais, P., Guimberteau, M., Traore, S., and Baron, C. (2013). Assessing climate change impacts on sorghum and millet yields in the sudanian and sahelian savannas of west africa. *Environmental Research Letters*, 8(1):014040.

Sun, L., Yang, Y., Hu, J., Porter, D., Marek, T., and Hillyer, C. (2017). Reinforcement learning control for water-efficient agricultural irrigation. In *2017 IEEE International Symposium on Parallel and Distributed Processing with Applications and 2017 IEEE International Conference on Ubiquitous Computing and Communications (ISPA/IUCC)*, pages 1334–1341. IEEE.

Sunehag, P., Evans, R., Dulac-Arnold, G., Zwols, Y., Visentin, D., and Coppin, B. (2015). Deep reinforcement learning with attention for slate markov decision processes with high-dimensional states and actions. *arXiv preprint arXiv:1512.01124*.

Sutton, R. S. and Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.

Swaminathan, A. and Joachims, T. (2015). Counterfactual risk minimization: Learning from logged bandit feedback. In *International Conference on Machine Learning*, pages 814–823. PMLR.

Sykuta, M. E. (2016). Big data in agriculture: property rights, privacy and competition in ag data services. *International Food and Agribusiness Management Review*, 19(1030-2016-83141):57–74.

Tack, J. B. and Holt, M. T. (2016). The influence of weather extremes on the spatial correlation of corn yields. *Climatic Change*, 134(1-2):299–309.

Tamkin, A., Keramati, R., Dann, C., and Brunskill, E. (2020). Distributionally-aware exploration for cvar bandits. In *NeurIPS 2019 Workshop on Safety and Robustness in Decision Making; RLDM 2019*.

Tang, J., Wang, J., Fang, Q., Wang, E., Yin, H., and Pan, X. (2018). Optimizing planting date and supplemental irrigation for potato across the agro-pastoral ecotone in north china. *European Journal of Agronomy*, 98:82–94.

Taylor, M. E. and Stone, P. (2009). Transfer learning for reinforcement learning domains: A survey. *Journal of Machine Learning Research*, 10(7).

Taylor, S., Payton, M., and Raun, W. (1999). Relationship between mean yield, coefficient of variation, mean square error, and plot size in wheat field experiments. *Communications in Soil Science and Plant Analysis*, 30(9-10):1439–1447.

Ten Berge, H. F., Hijbeek, R., Van Loon, M., Rurinda, J., Tesfaye, K., Zingore, S., Craufurd, P., van Heerwaarden, J., Brentrup, F., Schröder, J. J., et al. (2019). Maize crop nutrient input requirements for food security in sub-saharan africa. *Global Food Security*, 23:9–21.

Tesauro, G. (1995). Temporal difference learning and td-gammon. *Communications of the ACM*, 38(3):58–68.

Thomas, P. and Learned-Miller, E. (2019). Concentration inequalities for conditional value at risk. In *International Conference on Machine Learning*, pages 6225–6233. PMLR.

Thompson, W. R. (1933). On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294.

Thorburn, P., Jakku, E., Webster, A., and Everingham, Y. (2011). Agricultural decision support systems facilitating co-learning: a case study on environmental impacts of sugarcane production. *International Journal of Agricultural Sustainability*, 9(2):322–333.

Tilman, D., Cassman, K. G., Matson, P. A., Naylor, R., and Polasky, S. (2002). Agricultural sustainability and intensive production practices. *Nature*, 418(6898):671–677.

Tintner, G. (1955). Stochastic linear programming with applications to agricultural economics. In *Proceedings of the Second Symposium in Linear Programming*, volume 1, pages 197–228. National Bureau of Standards Washington, DC.

Tittonell, P. and Giller, K. E. (2013). When yield gaps are poverty traps: The paradigm of ecological intensification in african smallholder agriculture. *Field Crops Research*, 143:76–90.

Traore, B., Descheemaeker, K., Van Wijk, M. T., Corbeels, M., Supit, I., and Giller, K. E. (2017). modeling cereal crops to assess future climate risk for family food self-sufficiency in southern mali. *Field Crops Research*, 201:133–145.

Trépos, R., Lemarié, S., Raynal, H., Morison, M., Couture, S., and Garcia, F. (2014). Apprentissage par renforcement pour l'optimisation de la conduite de culture du colza. In *JFPDA'14. Journées Francophones Planification, Décision, Apprentissage pour la conduite de système.*, pages 1–13.

Tzachor, A., Devare, M., King, B., Avin, S., and Ó hÉigeartaigh, S. (2022). Responsible artificial intelligence in agriculture requires systemic understanding of risks and externalities. *Nature Machine Intelligence*, 4(2):104–109.

Vanlauwe, B., Coyne, D., Gockowski, J., Hauser, S., Huising, J., Masso, C., Nziguheba, G., Schut, M., and Van Asten, P. (2014). Sustainable intensification and the african smallholder farmer. *Current Opinion in Environmental Sustainability*, 8(0):15–22.

Vanlauwe, B., Kihara, J., Chivenge, P., Pypers, P., Coe, R., and Six, J. (2011). Agronomic use efficiency of n fertilizer in maize-based systems in sub-saharan africa within the context of integrated soil fertility management. *Plant and soil*, 339(1):35–50.

Vasan, G. and Pilarski, P. M. (2017). Learning from demonstration: Teaching a myoelectric prosthesis with an intact limb via reinforcement learning. In *2017 International Conference on Rehabilitation Robotics (ICORR)*, pages 1457–1464. IEEE.

Vasseur, C., Joannon, A., Aviron, S., Burel, F., Meynard, J.-M., and Baudry, J. (2013). The cropping systems mosaic: how does the hidden heterogeneity of agricultural landscapes drive arthropod populations? *Agriculture, ecosystems & environment*, 166:3–14.

Vovk, V. and Wang, R. (2021). E-values: Calibration, combination and applications. *The Annals of Statistics*, 49(3):1736–1754.

Waghmare, H., Kokare, R., and Dandawate, Y. (2016). Detection and classification of diseases of grape plant using opposite colour local binary pattern feature and machine learning for automated decision support system. In *2016 3rd international conference on signal processing and integrated networks (SPIN)*, pages 513–518. IEEE.

Wallach, D., Makowski, D., Jones, J. W., and Brun, F. (2018). *Working with dynamic crop models: methods, tools and examples for agriculture and environment*. Academic Press.

Wallach, D., Makowski, D., Jones, J. W., Brun, F., and Jones, J. (2014). Working with dynamic crop models. *Methods, tools and examples for agriculture and environment*.

Wallach, D. and Thorburn, P. J. (2017). Estimating uncertainty in crop model predictions: Current situation and future prospects.

Wang, L., He, X., and Luo, D. (2020). Deep reinforcement learning for greenhouse climate control. In *2020 IEEE International Conference on Knowledge Graph (ICKG)*, pages 474–480. IEEE.

Wasserstein, R. L. and Lazar, N. A. (2016). The asa statement on p-values: context, process, and purpose.

Watkins, C. J. C. H. (1989). Learning from delayed rewards.

Waudby-Smith, I. and Ramdas, A. (2020). Estimating means of bounded random variables by betting. *arXiv preprint arXiv:2010.09686*.

Weintraub, A. and Romero, C. (2006). Operations research models and the management of agricultural and forestry resources: a review and comparison. *Interfaces*, 36(5):446–457.

Weiss, K., Khoshgoftaar, T. M., and Wang, D. (2016). A survey of transfer learning. *Journal of Big data*, 3(1):1–40.

White, J., Hoogenboom, G., and Hunt, L. (2005). A structured procedure for assessing how crop models respond to temperature. *Agronomy Journal*, 97(2):426–439.

White, J. W., Boote, K. J., Hoogenboom, G., and Jones, P. G. (2007). Regression-based evaluation of ecophysiological models. *Agronomy Journal*, 99(2):419–427.

White, J. W., Hoogenboom, G., Kimball, B. A., and Wall, G. W. (2011). Methodologies for simulating impacts of climate change on crop production. *Field Crops Research*, 124(3):357–368.

Williams, R. J. (1987). *Reinforcement-learning connectionist systems*. College of Computer Science, Northeastern University.

Willmott, C. J., Ackleson, S. G., Davis, R. E., Feddema, J. J., Klink, K. M., Legates, D. R., O'donnell, J., and Rowe, C. M. (1985). Statistics for the evaluation and comparison of models. *Journal of Geophysical Research: Oceans*, 90(C5):8995–9005.

Wolfert, S., Ge, L., Verdouw, C., and Bogaardt, M.-J. (2017). Big data in smart farming–a review. *Agricultural Systems*, 153:69–80.

Yang, J., Yang, J.-Y., Liu, S., and Hoogenboom, G. (2014). An evaluation of the statistical methods for testing the performance of crop models with observed data. *Agricultural Systems*, 127:81–89.

Yang, Y., Hu, J., Porter, D., Marek, T., Heflin, K., and Kong, H. (2020). Deep reinforcement learning-based irrigation scheduling. *Transactions of the ASABE*, 63(3):549–556.

Zhai, Z., Martínez, J. F., Beltran, V., and Martínez, N. L. (2020). Decision support systems for agriculture 4.0: Survey and challenges. *Computers and Electronics in Agriculture*, 170:105256.

Zhang, Y. and Yang, Q. (2021). A survey on multi-task learning. *IEEE Transactions on Knowledge and Data Engineering*.