Making and Keeping Probabilistic Commitments for Trustworthy Multiagent Coordination

by

Qi Zhang

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Computer Science and Engineering)
in The University of Michigan
2020

Doctoral Committee:

Professor Satinder Singh Baveja, Co-Chair
Professor Edmund H. Durfee, Co-Chair
Professor Richard L. Lewis
Assistant Professor Arunesh Sinha

Qi Zhang
qizhg@umich.edu
ORCID iD: 0000-0002-8562-5987

ACKNOWLEDGEMENTS

First and foremost I would like to give my sincerest and warmest thanks to my co-advisors, Edmund Durfee and Satinder Singh, for their wisdom and patience in training me to become a researcher. They have also been extraordinarily supportive of my academic career after PhD. I was extremely lucky to be your student.

I would like to give thanks to Rick Lewis, who served on my doctoral committee, for advising me on an early project that did not end up in this thesis but I am particularly proud of. I would like to thank Arunesh Sinha, who also served on my committee, for giving thoughtful feedback in my dissertation defense and writing.

I would like to express my gratitude to Nan Jiang, Xiaoxiao Guo, Junhyuk Oh, Shun Zhang, Aditya Modi, Janarthanan Rajendran, Vivek Veeriah, John Holler, Max Smith, Zeyu Zheng, Ethan Brooks, Chris Grimm, Wilka Carvalho, Risto Vuorio, and other members in the RL lab, with whom I spent a wonderful time as a PhD student. In particular, I am thankful to Xiaoxiao who introduced me to IBM Research, where Murray Campbell hosted me as an intern. I truly enjoyed discussions and conversations with Murray, Gerald Tesauro, Miao Liu, Ashwin Kalyan, and fellow interns at IBM.

I would like to give my heartiest thanks to my parents, my twin brother, and my grandparents for their love and support. Finally, I would like to thank Xiaoyu, my wife, for everything you have brought to me. As I am writing down these words, we are celebrating your birthday. Happy birthday to my little sweetheart.

TABLE OF CONTENTS

LIST OF FIGURES

Figure

LIST OF TABLES

Table

ABSTRACT

In a large number of real world domains, such as the control of autonomous vehicles, team sports, medical diagnosis and treatment, and many others, multiple autonomous agents need to take actions based on local observations, and are interdependent in the sense that they rely on each other to accomplish tasks. Thus, achieving desired outcomes in these domains requires interagent coordination. The form of coordination this thesis focuses on is *commitments*, where an agent, referred to as the commitment *provider*, specifies guarantees about its behavior to another, referred to as the commitment *recipient*, so that the recipient can plan and execute accordingly without taking into account the details of the provider's behavior. This thesis grounds the concept of commitments into decision-theoretic settings where the provider's guarantees might have to be probabilistic when its actions have stochastic outcomes and it expects to reduce its uncertainty about the environment during execution.

More concretely, this thesis presents a set of contributions that address three core issues for commitment-based coordination: probabilistic commitment adherence, interpretation, and formulation. The first contribution is a principled semantics for the provider to exercise maximal autonomy that responds to evolving knowledge about the environment without violating its probabilistic commitment, along with a family of algorithms for the provider to construct policies that provably respect the semantics and make explicit tradeoffs between computation cost and plan quality. The second contribution consists of theoretical analyses and empirical studies that improve our understanding of the recipient's interpretation of the partial information specified in a probabilistic commitment; the thesis shows that it is inherently easier for the recipient to robustly model a probabilistic commitment where the provider promises to enable preconditions that the recipient requires than where the provider instead promises to avoid changing already-enabled preconditions. The third contribution focuses on the problem of formulating probabilistic commitments for the fully cooperative provider and recipient; the thesis proves structural properties of the agents' values as functions of the parameters of the commitment specification that can be exploited to achieve orders of magnitude less computation for 1) formulating optimal

commitments in a centralized manner, and 2) formulating (approximately) optimal queries that induce (approximately) optimal commitments for the decentralized setting in which information relevant to optimization is distributed among the agents.

CHAPTER I

Introduction

The capability of making sequences of decisions to accomplish complex tasks is a fundamental characteristic of both humans and artificial intelligent systems. Thus, developing sequential decision makers, or *agents*, that are capable of accomplishing tasks in uncertain, complex environments is a core research area in Artificial Intelligence (AI). Agent-based systems, which are designed to support intelligent decision making, address many important AI applications, such as household robots, medical diagnosis and treatment, online recommendation systems, video game AI, etc.

In multiagent systems, multiple agents make decisions in a distributed manner, in the sense that each agent's decisions are based on its local information, with limited or even no communication with others. When agents are purely selfish with potential conflict of interests, each agent aims to find its optimal strategy given others' strategies. This is the scenario that is commonly studied in game theory. This thesis focuses on a different situation in which agents' interests are (at least partially) aligned, so that they have incentives to cooperate on shared goals that require collective efforts. Successful cooperation often requires coordination among agents, especially when agents are interdependent in the sense that one agent's actions will not yield desired outcomes unless other agents act in concert. Coordination can be best achieved by a centralized decision maker that can directly control all agents. However, the distributed nature of multiagent systems often precludes that, making multiagent coordination a challenging problem.

This thesis focuses on the two-agent coordination problem where one agent depends on the other to achieve goals. In such a dependency, successful coordination requires the agent that is depended on to provide some guarantee on the outcomes of its actions, so that the other agent can plan its own actions accordingly. When a centralized decision maker is precluded, how can the two agents be coordinated in a distributed manner? *Commitments* in a multiagent system capture relationships

between two agents, and thus can be used to achieve successful coordination for such dependencies. By making a commitment, the depended-on agent probabilistically guarantees to bring about outcomes that the other agent needs. In this thesis, we refer to the agent that makes a commitment as the commitment *provider*, and to the other agent as the commitment *recipient*. A commitment *decouples* the planning between the provider and the recipient. The provider can freely exercise its individual autonomy as long as its actions are in accordance with the commitment, and the recipient autonomously plans its own actions with the expectation that the commitment will be realized. Other than the commitment, neither agent needs to take into account the details of the other agent. This decoupling effectively divides the coordination problem into two independent subproblems, making commitments a flexible and scalable framework for multiagent coordination.

Due to their efficacy, commitments are pervasive both in human society and among artificial intelligent agents. Drivers use turn signals as commitments to changing lanes shortly, so that other drivers can react safely. When a doctor is treating a patient, the (perhaps implicit) commitment by the doctor is to cure the patient. As an instance in artificial intelligence systems, consider a rover sent by a spacecraft to collect rocks on the surface of Mars. After collection, the rover should deliver the rocks to a base station, where the spacecraft will pick up the rocks and send them back to Earth. The rover can make a commitment that specifies the time it will arrive at the base station, so that, instead of staying idle, the spacecraft can be occupied with other missions before the time the rover commits to. Commitments can also exist between humans and artificial agents. Consider another example, in which a household robot is washing dishes while its human user has asked it to make a cup of coffee in 5 minutes. The request from the human can be modeled by the robot as a commitment it has made. If the robot can finish the ongoing dishwashing task quickly enough, it might want to finish it first before making the coffee; otherwise, the robot should pause and fulfill its commitment first. In a more complicated scenario, if there is no clean cup at the moment, the robot might need to pause immediately and clean a cup first. By modelling and reasoning with commitments it has made, the robot can efficiently and reliably meet the human user's requests.

1.1 Problems and Contributions

Although a commitment framework is a general notion for multiagent coordination, to make it useful, computational models of commitments are needed to address

challenges that arise from inherent uncertainty in the agents' environment. This section introduces the core problems of interest in the commitment framework, and summarize the contributions of this thesis that solve these problems.

The outcome of actions might be uncertain, immediately giving rise to a challenge for the commitment provider. We say that a commitment is *realized* if the provider has successfully brought about the outcome specified in the commitment. Due to the uncertainty in the outcome of actions, the provider might still fail to realize the commitment despite its best effort. Therefore, such uncertainty might preclude the perfect guarantee that a commitment can be surely realized. In the doctor-patient commitment, for instance, the prescribed treatment might not be effective for the patient, despite that it is effective for most other people. In the rover-spacecraft commitment, the extreme weather on Mars might be so impeding that the rover can be delayed on its way to the base station. Under the uncertainty about the outcome of the provider's actions, how can the provider be trusted by the recipient if the commitment cannot be surely realized, and how should we specify such uncertain commitments to facilitate coordination between the agents?

To answer such questions, people have framed the uncertainty of actions' outcomes using probability models, such as the Markov Decision Process (MDP), which is going to be reviewed in Section 2.1, and have developed the notion of *probabilistic commitments* this thesis adopts [XL00, WD07, BLG10]. In MDPs, we assume that the possible outcomes of actions are known, and the likelihood of each outcome is quantified with some probability. Thus, a commitment can be associated with a probability that quantifies the likelihood of the commitment being realized, which defines a probabilistic commitment. *The provider can be trusted to adhere to a probabilistic commitment if it takes a course of action that would have realized the commitment with sufficient likelihood, even if in a particular instance the specified outcome was not realized.* In the doctor-patient commitment, for instance, the doctor could be deemed to adhere to the commitment in a trustworthy manner if the prescribed treatment is appropriate for the patient's condition, even if the condition may not end up being improved. Moreover, by using a probability to quantify the provider's action outcome uncertainty, the commitment gains predictive value for coordination with the recipient, because the recipient can predict the likelihood of possible outcomes, plan to exploit the outcome of the commitment being realized, and at the same time prepare against the opposite outcome.

Contribution 1 - Trustworthy Adherence to Probabilistic Commitments

After making a probabilistic commitment, the provider exercises its own autonomy to maximize its own utility as long as it takes a course of action that realizes the commitment with a probability that is at least what was promised. The provider can solve this commitment-constrained planning problem offline if it has a model of its environment, which computes the probability of realizing the commitment for any course of action, along with the utility associated with the course of action. In general, however, the provider only has partial knowledge about its environment at the time of making a commitment, and thus it is uncertain about the model it needs for planning; as the provider interacts with its environment after making the commitment, it can obtain information to refine the model. The refined model might reveal that the commitment, already agreed to, is more/less preferable or more/less possible for the provider to realize. When the Mars rover makes the commitment, for instance, it might be uncertain at the beginning about how the weather on Mars can change over time, and it is also uncertain about which areas on Mars have more valuable rocks to collect. The rover's uncertainty will be reduced when it actually explores Mars after the commitment is already made: it could be that the weather suddenly turns worse, increasing its difficulty to move to the base station, or the rover could discover that a remote area has more valuable rocks, thus adjusting its plan to visit that area. The uncertainty about the model gives rise to a problem for the commitment provider. What is the semantics of a probabilistic commitment, if at the time of making it the provider is uncertain about how likely its course of action would be to realize it? How can the provider respond to its evolving knowledge about the environment model to maximize its utility without violating its probabilistic commitment?

The first contribution of this thesis, based on joint work with Edmund Durfee and Satinder Singh [ZDS+16, ZSD17, ZDS20b], generalizes the framework of probabilistic commitments to situations where the provider has Bayesian uncertainty about its preferences and/or the environment dynamics at the time it makes a probabilistic commitment. Crucially, in such a setting the provider expects to learn information in the midst of execution that improves its knowledge about how preferable and/or possible it is to realize the commitment. This thesis develops a formal semantics that builds on the novel perspective that the provider, under the Bayesian model uncertainty, should fulfill the commitment's probabilistic guarantee with respect to its Bayesian prior. This semantics equips the provider with the flexibility to respond to its evolving uncertainty while still preserving the provider's trustworthiness that secures effective coordination with the recipient. *It is the first prescriptive commitment*

semantics under decision-theoretic model uncertainty. In an illustrative domain, we compare it to several alternative semantics in related work and show how our prescriptive probabilistic commitment semantics allows agents to achieve better coordination than these alternatives.

Further, this thesis develops *a family of methods for the provider to construct policies that provably respect the semantics.* These methods use novel techniques that implement parametrized lookahead and online iterations to make it practical for the provider to plan with its evolving posterior in a careful way that provably fulfills the commitment's probabilistic guarantee. In several classic planning domains under model uncertainty, we show that these methods are able to strike a balance between computation cost and plan quality. The techniques developed in the methods are well suited for a wide range of settings where an agent is learning about the environment to maximize its value while its behavior is regulated by predefined constraints, including but not restricted to probabilistic commitments.

Contribution 2 - Robust Interpretation of Probabilistic Commitments

Another major challenge for the probabilistic commitment framework comes from the other end of a commitment, which is the recipient. Conditioning on the provider's adherence to the commitment, the recipient aims to find its own action selection policy that is best aligned with its own interests. As the provider's guarantee is in general not perfect (the probability of realizing the commitment is less than one), the recipient's action selection policy should exploit the likely outcome that the commitment will be realized, while at the same time prepare against the possibility that the commitment will not. Moreover, even when the provider realizes the commitment, the provider might be unable to specify the exact timing of the realization, as it responds to its evolving knowledge about the environment.

For the second contribution, this thesis formally analyzes alternative strategies the recipient can use to plan its action selection policy for the commitment's uncertain outcomes. Specifically, given a probabilistic commitment, there is a set of candidate behaviors of the provider that respect the commitment semantics, including various timings of realizing the commitment, and the possibility that the commitment will not be realized, and the recipient is uncertain about which of these behaviors the provider will eventually follow. To deal with such uncertainty for the recipient, this thesis considers alternative strategies to model a candidate behavior of the provider, which is then used for the recipient's planning. To compare the performance between alternative strategies, this thesis develops a novel notion of suboptimality that quan-

tifies loss in plan quality against the provider's possible behaviors in the candidate set.

To better analyze the recipient's alternative strategies, this thesis further makes a clear distinction between two types of commitment by formally defining them in the probabilistic commitment framework. Specifically, we are concerned with commitments of achievement, where the provider commits to enabling a precondition needed by the recipient, and also commitments of maintenance, where the provider's commitment is instead to avoid changing a precondition that is already enabled for the recipient. This thesis is the first to give formal definitions of the two types of commitment in the probabilistic commitment framework.

With the novel notion of suboptimality and the formal definitions, this thesis presents results showing that, *despite strong superficial similarities between the two types of commitment, there is an inexpensive strategy with low suboptimality for modelling commitments of achievement, while no such strategy exists for maintenance.* The results are obtained from a combination of theoretical analyses on worst-case suboptimality in an exemplar domain for the recipient, and empirical studies on the suboptimality for rational commitments in that domain. These results assure us that the recipient can robustly interpret achievement commitments, and thus the agents can reliably coordinate well with them, while leaving the question open of how to improve the representation and reasoning for coordination with maintenance commitments. This is the second contribution of this thesis, based on joint work with Edmund Durfee and Satinder Singh [ZDS18, ZSD20].

Contribution 3 - Efficient Formulation of Cooperative Commitments

With the prior contributions that apply to an arbitrary probabilistic commitment, the third and final contribution of this thesis focuses on the question of what probabilistic commitment the agents should agree upon. This thesis takes on this question in the fully cooperative case where the objective is for the agents to agree on a commitment that induces behavior that optimizes their joint performance. The joint performance is measured by the sum of the provider's value and the recipient's value of the commitment they have agreed on. As the third contribution, this thesis solves the problem of how the agents can efficiently determine a commitment of high joint performance in the setting where the information of the two agents is precisely known to a centralized coordinator, and the setting where the information is distributed among the agents. This contribution is based on joint work with Edmund Durfee and Satinder Singh [ZDS20a].

Specifically, this thesis reveals structural properties of the agents' values as functions of the parameters of the commitment specification, where these properties can be provably exploited to efficiently formulate optimal cooperative commitments in a centralized manner. For the provider, this thesis proves the structural properties of its commitment value function by analyzing the mathematical program that solves its planning problem, where the commitment parameters appear as the program's constraints. This novel angle of analyzing the commitment value allows us to show the regularity of the provider's value as a function the commitment parameters. For the recipient, the structural properties of its commitment value function are proved by establishing that the recipient's robust interpretation defines its commitment value as a linear function of probability. Excitingly, *the structural properties for the two agents are compatible in a way that the optimal cooperative commitment can be efficiently formulated with a binary search procedure in the centralized setting.*

Further, this thesis considers the decentralized setting in which information relevant to optimization is distributed among the agents. It turns out that *the structural properties define a small number of commitments with potentially high summed value of the two agents, which can be exploited to develop an efficient querying approach for the agents to exchange information to converge on (approximately) optimal cooperative commitments.* Even on problem instances with randomly-generated MDPs that have minimal structural assumptions, the method empirically proves to be orders of magnitude more computationally efficient than several alternatives that are ignorant of the structural properties. The properties reveal the structure of the commitment space, and thus they not only lead to efficient solutions to the problems in this thesis, but also provide insights to any problems involving optimization over the commitment space.

1.2 Thesis Structure

Chapter II presents the technical background for this thesis, including the formal notions of Markov Decision Processes (MDPs), Bayesian model uncertainty, and probabilistic commitments. With the technical background, Chapter III formulates the three problems of interest in this thesis, which we have introduced in Section 1.1, in the context of related work. The three problems are solved in Chapters IV, V, and VI, respectively, with the three core contributions unrolled in detail. Chapter VII concludes this thesis with the emphasis of how the thesis work contributes to the broader research community, and suggests future research directions.

CHAPTER II

Background

In this chapter, we introduce the computational models this thesis uses for probabilistic commitments and for the decision-making problems of both the provider and the recipient.

2.1 Markov Decision Processes

We first provide background on the Markov Decision Process (MDP) [Put14], which describes the interactions between a single agent and its environment. A finite MDP is formally defined by a tuple $M = (\mathcal{S}, \mathcal{A}, P, R, H)$ where

- State space $\mathcal{S}$ is a finite set of states that the agent might encounter.
- Action space $\mathcal{A}$ is a finite set of actions that are available for the agent.
- $P : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$, where $\Delta(\mathcal{S})$ denotes the set of all probability distributions over $\mathcal{S}$, is the transition function. $P(s_{t+1}|s_t, a_t)$ specifies the probability of transitioning into state s_{t+1} upon taking action a_t in state s_t.
- $R : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the reward function. $R(s_t, a_t)$ is the immediate reward upon taking action a_t in state s_t. Note that we assume that the reward only depends on s_t and a_t deterministically. In general, the reward may also depend on s_{t+1} and may even be stochastic. With respect to expected cumulative reward, which we ultimately care about, this general setup can be reduced to our setup by defining $R(s_t, a_t) = E[r_{t+1}|s_t, a_t]$, where r_{t+1} is the immediate reward upon (s_t, a_t) that can be dependent on s_{t+1} and stochastic.
- H is the decision horizon. This thesis considers the finite horizon case, in which the agent interacts with the environment over a finite number, H, of transitions.

With the horizon being finite, the state space is partitioned into disjoint sets by the time step, $\mathcal{S} = \bigcup_{t=0}^{H} \mathcal{S}_t$, and the agent starts in an initial state $s_0 \in \mathcal{S}_0$. In this thesis, the notation of state s implicitly specifies its time step (i.e. $s \in \mathcal{S}_t$ for some t), and s_t explicitly specifies its time step. At each time step $t = 0, 1, ..., H-1$, the agent takes an action $a_t \in \mathcal{A}$ in state $s_t \in \mathcal{S}_t$, obtains a reward $r_{t+1} = R(s_t, a_t)$, and transits to a new state $s_{t+1} \in \mathcal{S}_{t+1}$ stochastically drawn from $P(\cdot|s_t, a_t)$, or $s_{t+1} \sim P(\cdot|s_t, a_t)$. In this thesis, we consider the case in which the initial state is fixed, i.e. $\mathcal{S}_0 = \{s_0\}$.

2.1.1 MDP Planning

A (stochastic) policy $\pi : \mathcal{S} \to \Delta(\mathcal{A})$ specifies a decision-making strategy for which the agent chooses actions based on the current state s, i.e. $a \sim \pi(\cdot|s)$. Starting in initial state s_0, a sequence of transitions $(s_0, a_0, r_1, s_1, ..., s_{H-1}, a_{H-1}, r_H, s_H)$ is generated, which records the entire history up to horizon H. The record

$$h_t = (s_0, a_0, r_1, s_1, ..., s_{t-1}, a_{t-1}, r_t, s_t)$$

up to time t is referred to as history h_t. The value function of π is $V_M^\pi(s) = E[\sum_{t'=t+1}^{H} r_{t'}|\pi, s_t = s]$ where t is such that $s \in \mathcal{S}_t$. There always exists an optimal policy, denoted as π_M^*, that maximizes V_M^π for all $s \in \mathcal{S}$. Planning refers to the problem of computing an optimal policy with the MDP specification $M = (\mathcal{S}, \mathcal{A}, P, R, H)$ given.

There are several planning algorithms. Here we summarize one based on linear programming (LP) [Put14]. Each policy π has a corresponding occupancy measure $x^\pi : \mathcal{S} \times \mathcal{A} \to [0, 1]$, where $x^\pi(s, a)$ is the expected number of times action a will be taken in state s over horizon H, starting in initial state s_0:

$$x^\pi(s, a) = E\left[1_{\{s_t=s, a_t=a\}}|s_0, \pi\right],$$

where t is such that $s \in \mathcal{S}_t$, and 1_E is the indicator function that takes value one if event E occurs and zero otherwise. We will use shorthand notation x in place of x^π when policy π is clear from the context. Policy π can be recovered from its occupancy measure via

$$\pi(a|s) = \frac{x(s, a)}{\sum_{a'} x(s, a')}.$$

Figure 2.1 is the linear program that solves an MDP M. It introduces the occupancy measure as decision variables, and the policy is constructed from the program's op-

$$\max_{x} \sum_{s,a} x(s,a)R(s,a) \tag{2.1a}$$

$$\text{subject to } \forall s,a \quad x(s,a) \geq 0; \tag{2.1b}$$

$$\forall s' \quad \sum_{a'} x(s',a') = \sum_{s,a} x(s,a)P(s'|s,a) + \delta(s',s_0). \tag{2.1c}$$

Figure 2.1: The linear program for MDP planning.

timal solution. Constraints (2.1b) and (2.1c) guarantee that x is a valid occupancy measure, where $\delta(s', s_0)$ is the Kronecker delta that returns 1 when $s' = s_0$ and 0 otherwise. The expected cumulative reward can be expressed using x in the objective function (2.1a).

2.2 Bayesian Model Uncertainty

For a single agent, it can solve its planning problem and find the optimal policy if it knows precisely its MDP tuple M. In this thesis, we will also consider a more realistic scenario where the agent has uncertainty about the transition and reward functions of its MDP. This type of uncertainty is referred to as model uncertainty. Formally, we will consider the Bayesian setting in which the agent's true MDP is one out of K possible MDPs drawn from a known prior distribution μ_0, where all MDPs share identical state and action spaces but possibly different transition and reward functions, and the state and the reward are fully observable during execution. Thus, the environment with Bayesian model uncertainty is formally defined by the tuple $(\mathcal{S}, \mathcal{A}, \{P_k, R_k\}_{k=1}^{K}, s_0, \mu_0, H)$. The agent's objective under Bayesian model uncertainty is to maximize its initial state value with respect to the prior. For policy π, its value for initial state s_0 under Bayesian model uncertainty is defined as

$$V_{\mu_0}^{\pi}(s_0) = E_{M_k \sim \mu_0}\left[V_{M_k}^{\pi}(s_0)\right]$$

where M_k is the k-th candidate MDP, and the expectation is with respect to prior μ_0. During execution, the agent can use the knowledge provided by the history so far to infer which MDP is more/less likely to be the true MDP it is facing. Therefore, to maximize value $V_{\mu_0}^{\pi}(s_0)$, the agent's policy should choose actions depending on

the history (instead of only on the current state). We use $\pi(a_t|h_t)$ to denote the probability of choosing action a_t given history h_t up to time t when following a history-dependent policy π starting from s_0. We refer to a policy that chooses actions only depending on the current state, as those discussed in Section 2.1, as a Markov policy. With no model uncertainty, there is no loss of optimality by restricting to Markov policies.

2.3 Probabilistic Commitments

Using MDPs, this section formulates the dependency between the provider and the recipient established by a probabilistic commitment.

2.3.1 Provider-Recipient Decision-Making Model

This thesis uses MDPs to model the decision-making problems for both the provider and the recipient, denoted by superscripts p and r, respectively. Thus, the provider's MDP is $M^{\mathrm{p}} = (\mathcal{S}^{\mathrm{p}}, \mathcal{A}^{\mathrm{p}}, P^{\mathrm{p}}, R^{\mathrm{p}}, H^{\mathrm{p}})$ with initial state s_0^{p}, and the recipient's MDP is $M^{\mathrm{r}} = (\mathcal{S}^{\mathrm{r}}, \mathcal{A}^{\mathrm{r}}, P^{\mathrm{r}}, R^{\mathrm{r}}, H^{\mathrm{r}})$ with initial state s_0^{r}. We assume the two MDPs share the same horizon $H = H^{\mathrm{p}} = H^{\mathrm{r}}$. Intuitively, the provider's actions not only determine the transitions in its own MDP but also influence the transitions in the recipient's MDP, and therefore, by making a commitment, the provider can promise to bring about transitions desired by the recipient with some probability.

To formulate such a coupling between the two MDPs, this thesis adopts the Transition-Decoupled Partially Observable MDP (TD-POMDP) model [WD07, WD10], which assumes that states in each MDP can be factored into features, and models the coupling as the dependency between shared state features. Specifically, both the provider's state s^{p} and the recipient's state s^{r} can be factored into state features. The provider can fully control its state, in the sense that the next provider state s_{t+1}^{p} is entirely dependent on the current provider's state and action $(s_t^{\mathrm{p}}, a_t^{\mathrm{p}})$, but not on the recipient's state or action. The recipient's state can be factored as $s^{\mathrm{r}} = (l^{\mathrm{r}}, u)$, where l^{r} is the set of all the recipient's state features locally controlled by the recipient, and u is the set of state features uncontrollable by the recipient but shared with the provider, i.e. $u = s^{\mathrm{p}} \cap s^{\mathrm{r}}$. Formally, the recipient's transition function is factored as $P^{\mathrm{r}} = (P_l^{\mathrm{r}}, P_u^{\mathrm{r}})$:

$$\begin{aligned} P^{\mathrm{r}}\left(s_{t+1}^{\mathrm{r}}|s_t^{\mathrm{r}}, a_t^{\mathrm{r}}\right) =& P^{\mathrm{r}}\left((l_{t+1}^{\mathrm{r}}, u_{t+1})|(l_t^{\mathrm{r}}, u_t), a_t^{\mathrm{r}}\right) \\ =& P_u^{\mathrm{r}}(u_{t+1}|u_t) P_l^{\mathrm{r}}\left(l_{t+1}^{\mathrm{r}}|(l_t^{\mathrm{r}}, u_t), a_t^{\mathrm{r}}\right), \end{aligned}$$

where the transition dynamics of u, P_u^{r}, is determined only by the provider's policy (i.e., it is not a function of a_t^{r}). We refer to P_u^{r} as the true *influence* that the provider exerts on the recipient's MDP.

2.3.2 Predictive Commitment Semantics

A probabilistic commitment is concerned with state features u that are shared by both agents but only controllable by the provider. Intuitively, a probabilistic commitment partially specifies how the provider will influence u's dynamics P_u^r, and therefore can be exploited by the recipient to plan accordingly. Definition II.1 formally gives the definition of a probabilistic commitment regarding the provider's MDP M^{p} and the recipient's MDP M^{r}.

Definition II.1. Regarding M^{p} and M^{r}, a probabilistic commitment is formally defined as a tuple $c = (u_c, T_c, p_c)$:

- u_c is the commitment value for features u.
- T_c is the commitment time.
- p_c is the commitment probability.

As a predictive semantics, the commitment probability p_c gives a lower bound on how likely the shared state features u will be taking the value of u_c at time T_c, i.e. $\Pr(u_{t=T_c} = u_c) \geq p_c$, based on whatever policy the provider is following.

With its predictive semantics, the probabilistic commitment quantifies the possibility that the commitment can be unrealized due to action outcome uncertainty. For example, actions might stochastically have irreversible outcomes from which the commitment value is unrealizable.

2.3.3 Commitment-based Multiagent Coordination

Since the commitment is concerned with the shared state features, it can be used to achieve coordination among the two agents. With the commitment's predictive semantics, the recipient can make useful predictions about the provider's influence and plan accordingly, and the provider can plan its policy to improve its own value as long as it meets the commitment's probabilistic guarantee. Thus, high-quality coordination can be achieved if the two agents agree on an appropriate commitment. In the next chapter, we will formulate problems solved in this thesis that arise in the commitment-based multiagent coordination framework.

Before moving on to the next chapter, we here review alternative frameworks for multiagent coordination by listing below the strengths of commitment-based coordination compared with these alternatives.

Improved scalability over centralized multiagent planning. Researcher have developed a variety of decision-making models that describe the interaction between multiple agents via shared state features, such as the Multiagent Markov Decision Process (MMDP) introduced in [Bou96], and the Decentralized MDP (Dec-MDP) introduced in [BGIZ02], which can be viewed as a variation and a generalization of the TD-POMDP model this thesis adopts, respectively. For these models, several solution methods have been proposed that either solve multiagent planning exactly or approximately. Optimal solution methods include extensions of dynamic programming [HBZ04], heuristic search [SCZ05], and iterative policy optimization [BAHZ09]. Due to the intractable complexity of these models [BGIZ02], the scalability of these optimal methods is limited. To a great extent, the successes in scaling multiagent planning to more than two or three agents are achieved by approximate solution methods that rely on the use of *decoupled* solution methods that reason about and optimize each agent's individual policy locally, in contrast to centralized methods that optimize all agent's policies in combination. For example, [NTY+03] develops Joint Equilibrium-based Search for Policies (JESP) that converges to a set of equilibrium local policies in which each policy is the best response given the others. Commitment-based coordination is one of such decoupled methods because both the provider and the recipient only optimize their local policies upon an agreed commitment, and thus enjoys improved scalability over centralized multiagent planning.

Reduced complexity by interaction abstraction. In many decoupled solution methods, like JESP, each agent needs to have others' local models and/or policies in detail in order to optimize its own decisions. In contrast, for the provider-recipient interaction, either agent only needs to reason about the shared state feature for coordination, and not other details, and this is exactly what the commitment is used for. Thus, compared with alternative decoupled solution methods, the commitment provides an abstraction of the multiagent interaction in a way that further reduces the complexity of coordination.

As a quick summary, the probabilistic commitment abstracts the interaction between the two agents by partially specifying the provider's influence on the shared state features, and thus decouples the planning of the two agents and further reduces

the complexity of coordination. Crucially, the commitment abstraction specifies the provider's influence only at a single time step T_c. How is it compared to a more detailed specification that specifies the influence at multiple time steps? For example, prior work has proposed to instead fully specify the provider's influence on the shared state features at every time step [WD10, OWK12]. We next compare the commitment abstraction with more detailed specifications. There are two clear advantages of using a more detailed specification over the single time step commitment abstraction:

- First, as the recipient gets more information about the influence at multiple time steps, it can predict the provider's behavior with more certainty, and plan with more confidence. In contrast, for the single time step commitment abstraction, the recipient can only estimate the values of influence at those unspecified time steps, and inaccurate estimation can sometimes yield a policy of lower quality.

- Second, a specification that specifies the influence at multiple time steps can be viewed as an generalization of the single time step commitment abstraction, which can specify only one time step and leave other time steps underspecified. Thus, advantages of the commitment abstraction can in principle be preserved in more detailed specifications.

Despite the two advantages of more detailed specifications, in many cases the commitment abstraction is still preferred due to the following reasons.

Computational efficiency for optimization. It is relatively easy to determine a commitment that best coordinates the provider and the recipient, by identifying a single time step as the commitment time, along with a probability. In contrast, it can be computationally challenging to optimize a multiple time step specification due to the combinatorics. When optimizing for coordination, the agents might prefer to use the commitment abstraction simply for computational efficiency.

Flexibility for the provider under model uncertainty. As we will formally describe later in Chapter III, in this thesis we consider the setting in which the provider has model uncertainty as defined in Section 2.2, and thus would need flexibility to respond to its evolving knowledge about the model by adopting a history-dependent policy that equivalently can shift from one Markov policy to another. For a detailed specification, like the specification that fully specifies the influence at every time step, it is possible but not convenient for the provider to do Markov policy shifting without violating the detailed specification. In Chapter IV, we will see empirical evidence that

shows how the provider's value is greatly limited without such flexibility. In contrast, the provider can easily gain more flexibility with the less detailed commitment abstraction, and Chapter IV develops methods for the provider to exploit the flexibility that comes with the commitment. Thus, the provider under model uncertainty would generally prefer the commitment abstraction.

Being efficiently and robustly modelled by the recipient. While the commitment abstraction gives the provider flexibility, it imposes uncertainty on the recipient about the exact specification of the influence, including the exact probability of the commitment being realized at both the commitment time and other time steps. In order to plan, the recipient's estimation on the influence with such uncertainty can yield low-quality policies. Chapter III formulates the problem that arises from such uncertainty on the recipient, and Chapter V shows that, in many cases, it is possible for the recipient to estimate the influence in a efficient and robust manner to yield a high-quality policy no matter what influence the provider ultimately exerts, and therefore the recipient's inefficiency induced by the commitment's partial specification is nonetheless outweighed by the provider's flexibility gained from the commitment's partial specification.

Although this thesis focuses entirely on the single-time step commitment abstraction, many of the contributions, especially in Chapters IV and VI, can be extended to multiple time step abstractions. Chapters IV develops a family of methods based on mathematical programming that solve the provider's planning problem for a given commitment. The mathematical programming techniques can be straightforwardly extended to multiple time step abstractions by incorporating the specification at additional time steps as additional constraints to the mathematical programs. Chapters VI develops efficient algorithms for optimizing a commitment to induce optimal cooperation between the two agents. These algorithms can be used as subprocedures for optimizing multiple time step abstractions, if the optimization is performed on a single time step at a time and alternates between time steps. These extensions to multiple time step commitments are left for future work.

CHAPTER III

Problem Formulation

With the provider's and the recipient's decision-making problem described using two coupled MDPs as in Chapter II, probabilistic commitments that are concerned with the shared state features that couple the MDPs provide a framework for coordination. The commitment predictive semantics suggests a two-phase procedure for coordination, which we describe now. In the first phase of commitment formulation, the two agents agree on a probabilistic commitment, which is determined by a centralized coordinator or as an outcome of communication between the agents in a decentralized manner. In the second phase of commitment execution, the provider and the recipient separately compute and follow their plans with respect to the commitment, and in this thesis we assume there is no communication in this phase. For the commitment execution phase, specifically, the provider should adhere to the commitment by computing and following a policy that will realize the commitment with at least the promised probability, as described in Section 2.3, because this allows the recipient to plan its policy accordingly by predicting the provider's influence on the shared state features. In this chapter, we formulate the three problems relating to the three major contributions of this thesis, all of which arise from the two-phase procedure of the commitment-based coordination.

The first problem is regarding the provider's commitment execution for a given probabilistic commitment, which is formulated in Section 3.1 and solved in Chapter IV. The second problem is regarding the recipient in the commitment execution phase, which is formulated in Section 3.2 and solved in Chapter V. The third problem is the commitment formulation, which is formulated in Section 3.3 and solved in Chapter VI.

3.1 The Provider's Adherence under Model Uncertainty

The predictive commitment semantics from prior work defined in Section 2.3 can fail to match commonsense notions of what making a commitment means, since it does not in any way impede a provider from unilaterally changing its commitment whenever it chooses to alter its policy. Therefore, in this thesis, we define and adopt a *prescriptive* semantics for a probabilistic commitment. Definition III.1 formally gives the prescriptive commitment semantics for the provider that knows precisely its MDP.

Definition III.1. The *prescriptive probabilistic commitment semantics* for probabilistic commitment $c = (u_c, T_c, p_c)$ requires that the commitment provider, knowing its MDP, is constrained to follow a Markov policy π^{p}, such that

$$\Pr(u_c \in s^{\mathrm{p}}_{T_c} | s^{\mathrm{p}}_0; \pi^{\mathrm{p}}) \geq p_c. \tag{3.1}$$

By Equation (3.1), our prescriptive probabilistic commitment semantics is clear: the provider is constrained to follow a policy, such that, starting at the initial state s^{p}_0, the probability of being in a state with commitment value u_c at the commitment time T_c is at least the commitment probability p_c. If the provider follows such a policy, then by Definition III.1 we say it adheres to its commitment in a trustworthy manner. Thus, adhering to a commitment is entirely under the provider's control, despite the fact that the commitment might be unrealized due to action outcome uncertainty.

To satisfy the prescriptive semantics, the provider should only agree to a commitment if it can find a policy with a sufficiently high probability ($\geq p_c$) of realizing the commitment. Formally, for a commitment c, let Π^{p}_c be the set of all possible provider's policies that satisfy the commitment constraint (3.1), i.e.

$$\Pi^{\mathrm{p}}_c = \{\text{Markov policy } \pi^{\mathrm{p}} : \pi^{\mathrm{p}} \text{ satisfies Equation (3.1)}\}.$$

We say commitment c is feasible if and only if Π^{p}_c is non-empty. Note that since commitment constraint (3.1) is an inequality, for a given commitment time T_c, there is a maximum feasible probability $\overline{p}(T_c)$ such that the commitment is feasible if and only if $p_c \in [0, \overline{p}(T_c)]$. This maximum feasible commitment probability $\overline{p}(T_c)$ can be computed by solving the provider's MDP with the reward function replaced with a simple reward function that gives +1 reward for states where the commitment is realized at T_c and 0 otherwise. This is because maximizing this reward is equivalent to maximizing the probability of realizing the commitment at time step T_c, i.e., $u_c \in s^{\mathrm{p}}_{T_c}$,

and thus the optimal initial state value is the maximum feasible probability $\overline{p}(T_c)$.

For trustworthy adherence to a given probabilistic commitment c, the provider can choose any policy in the commitment-constrained policy set Π_c^{p} to respect the commitment's prescriptive semantics. The provider would choose the optimal policy in Π_c^{p} that maximizes its own value of the initial state, i.e.

$$v^{\mathrm{p}}(c) = \max_{\pi^{\mathrm{p}} \in \Pi_c^{\mathrm{p}}} V_{M^{\mathrm{p}}}^{\pi^{\mathrm{p}}}(s_0^{\mathrm{p}}). \tag{3.2}$$

We refer to $v^{\mathrm{p}}(c)$ as the provider's commitment value function. This commitment-constrained optimization problem is illustrated in Figure 3.1.

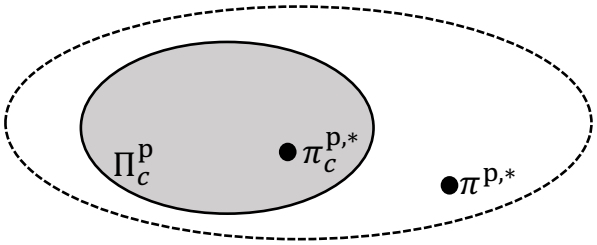


Figure 3.1: Illustration of the provider's commitment-constrained policy optimization problem. The dashed circle denotes the provider's policy set, in which the optimal (unconstrained) policy is denoted as $\pi^{\mathrm{p},*}$. The solid circle denotes the commitment-constrained policy set Π_c for a commitment c, in which the optimal commitment-constrained policy is denoted as $\pi_c^{\mathrm{p},*} \in \Pi_c$ as the solution of problem (3.2).

As the provider knows precisely its MDP M^{p}, in principle, it can enumerate policies in Π_c^{p}, compute their values, and choose the one that yields the most value. In practice, prior work has developed more efficient methods for this commitment-constrained policy optimization. Specifically, the provider's planning problem (3.2) can be solved with the linear program in Figure 3.2 [Alt99, WD07], which is adapted from the program in Figure 2.1 that solves (unconstrained) MDP planning. Specifically, the decision variable x^{p} is the provider's occupancy measure; objective (3.3a) and constraints (3.3b), (3.3c) are counterparts of (2.1a), (2.1b), and (2.1c), respectively; and constraint (3.3d) expresses the commitment constraint of Equation (3.1).

In this thesis, we consider the scenario in which the provider, at the time of making a probabilistic commitment, has the Bayesian model uncertainty with prior μ_0 over K candidate MDPs, as described in Section 2.2, and thus it is not able to directly perform the commitment-constrained policy optimization in Equation (3.2). An immediate question arises due to the provider's model uncertainty, and especially due to its uncertainty about the transition function: What does it mean for the provider

$$\max_{x^{\mathrm{p}}} \sum_{s^{\mathrm{p}},a^{\mathrm{p}}} x^{\mathrm{p}}(s^{\mathrm{p}},a^{\mathrm{p}})R^{\mathrm{p}}(s^{\mathrm{p}},a^{\mathrm{p}}) \tag{3.3a}$$

$$\text{subject to } \forall s^{\mathrm{p}},a^{\mathrm{p}} \quad x^{\mathrm{p}}(s^{\mathrm{p}},a^{\mathrm{p}}) \geq 0; \tag{3.3b}$$

$$\forall s^{\mathrm{p}\prime} \quad \sum_{a^{\mathrm{p}\prime}} x^{\mathrm{p}}(s^{\mathrm{p}\prime},a^{\mathrm{p}\prime}) = \sum_{s^{\mathrm{p}},a^{\mathrm{p}}} x^{\mathrm{p}}(s^{\mathrm{p}},a^{\mathrm{p}})P^{\mathrm{p}}(s^{\mathrm{p}\prime}|s^{\mathrm{p}},a^{\mathrm{p}}) + \delta(s^{\mathrm{p}\prime},s_0^{\mathrm{p}}); \tag{3.3c}$$

$$\sum_{s_{T_c}^{\mathrm{p}} \ni u_c} \sum_{a^{\mathrm{p}}} x^{\mathrm{p}}(s_{T_c}^{\mathrm{p}},a^{\mathrm{p}}) \geq p_c. \tag{3.3d}$$

Figure 3.2: The linear program for the provider's planning.

to adhere to its probabilistic commitment in a trustworthy manner, if the provider cannot be certain about the probability of its policy realizing the commitment at the time of making it? This question urges us to develop prescriptive semantics for probabilistic commitments that can be generalized to model uncertainty. This thesis develops such a prescriptive semantics, formally defined in Definition III.2, that generalizes the semantics in Definition III.1 to Bayesian model uncertainty with two key modifications: the provider is allowed to follow a general history-dependent policy (instead of only a Markov policy), and it incorporates the Bayesian prior into its guarantee on probabilistically realizing the commitment.

Definition III.2. After making commitment $c = (u_c, T_c, p_c)$ under Bayesian model uncertainty with prior μ_0, the provider is constrained to follow a (in general) history-dependent policy π^{p}, such that

$$\Pr_{M_k^{\mathrm{p}} \sim \mu_0} (u_c \in s_{T_c}^{\mathrm{p}} | s_0^{\mathrm{p}}, M_k^{\mathrm{p}}; \pi^{\mathrm{p}}) \geq p_c. \tag{3.4}$$

where M_k^{p} is the provider's k-th candidate MDP.

In words, knowing that it is facing an MDP drawn from prior μ_0 over possible MDPs ($M_k^{\mathrm{p}} \sim \mu_0$), the provider is constrained to follow a (in general) history-dependent policy π^{p}, such that, starting at the initial state s_0^{p}, the probability of realizing the commitment is at least the commitment probability p_c. The problem formulated here of adhering to a probabilistic commitment under model uncertainty is novel, and this thesis contributes the first prescriptive semantics for probabilistic commitments under model uncertainty.

With the novel commitment semantics (3.4) for Bayesian model uncertainty, the set of commitment-constrained policies can be enlarged to include history-dependent policies. With slight abuse of notation, we use

$$\Pi_c^{\mathrm{p}} = \{\text{History-dependent policy } \pi^{\mathrm{p}} : \pi^{\mathrm{p}} \text{ satisfies Equation (3.4)}\}.$$

to denote these commitment-constrained policies, and we say commitment c is feasible (under model uncertainty) if and only if Π_c^{p} defined as above is non-empty.

As we have discussed in Section 2.2, including history-dependent policies enables the provider to respond to its evolving knowledge about the environment, thus improving its value. We are interested in finding a policy that maximizes the initial state value with respect to the prior, while satisfying the constraint of a given feasible probabilistic commitment, which is formally formulated as the following problem:

$$\arg\max_{\pi^{\mathrm{p}} \in \Pi_c^{\mathrm{p}}} V_{\mu_0}^{\pi^{\mathrm{p}}}(s_0^{\mathrm{p}}).$$

Solving this problem involves two main challenges. First, it is non-trivial to characterize Π_c^{p} in a computationally-efficient manner that eases the policy optimization step. Second, under model uncertainty, finding the optimal policy (even without the constraint prescribed by the commitment semantics) requires planning with histories. This imposes additional computational difficulty, since the space of histories grows exponentially with the time horizon.

Related Work

There are alternative computational methods to the probabilistic commitment framework to model commitments among agents. A comprehensive overview of research into using formal (temporal and modal) logic to characterize and operationalize commitments has appeared [Sin12], and is based on literature in this field (e.g., [CL90, Cas95, Sin99, MH03, CMMT13, ASBSEM14]). These representations support important objectives like the provable pursuit of mutually agreed-upon goals, and codifying conventions and protocols for managing uncertainty (e.g., [Jen93, XS01, Win06]). As an example of a convention, an agent that determines that it will not keep a commitment might be obligated to inform dependent agents [Jen93].

Some of the logical representations above (e.g., [Jen93]) enumerate conditions where an agent is allowed to abandon its local component of a mutual goal, where

in general these conditions are either: (1) when the agent believes it is impossible to achieve its local component; (2) when the agent believes the mutual goal is not worth pursuing anymore; or (3) when the agent believes one or more of the other agents participating in the mutual goal have abandoned their local components of it. These conditions are logically reasonable, but fail to impose a prescriptive semantics for the agent to use in making local decisions. For example, to satisfy the first condition, is an agent never allowed to take an action that has even a small chance of rendering its local component unachievable? What if all of its actions have such a chance? For the second condition, if an agent can unilaterally drop a commitment whenever its preferred goal changes, then has it really committed in the first place?

To make an agent more predictable, a commitment can be paired with conditions under which it is guaranteed to hold [Raf82, Sin12, VKP09, AGJ07]. In transactional settings, for example, an agent could commit to providing a good or service on the condition that it first receives payment. However, if conditions can be over anything, then they can make commitments worthless because a commitment might be conditioned simply on no better option coming along. Sandholm and Lesser [SL01] recognized the general impracticality of enumerating all the conditions that might affect commitment adherence, and, even if the conditions could be specified, in verifying they hold in a distributed setting. Their solution was a contracting framework where a decommitment penalty is associated with each commitment, so as to accommodate uncertainty but discourage frivolous decommitment. However, even though the recipient will know it will be compensated if the commitment is abandoned, it in general will be unable to know how likely that will be, since it cannot look inside the provider to discern how likely it is that its actions to achieve the commitment will fail, or that it will decide that other goals should take priority.

Therefore, an alternative to a decommitment penalty is for the commitment provider to summarize the likelihood that its commitment's various conditions will jointly hold (e.g., a factory's suppliers will meet deadlines, its workers will not strike, its shippers will fulfill orders, etc.) into a summary probability. Hence, a probabilistic commitment [XL00, BLG10, WD09] is a form of conditional commitment where the details of the conditions have been replaced by an estimate of the probability that they will hold. Xuan and Lesser [XL00] have explained how probabilistic commitments improve joint planning by allowing agents to find policies that are responsive to possible contingencies, including even unlikely ones, and computing appropriate alternative courses of action as the probabilities for commitments being met change. A more myopic (and more tractable) variation of this approach was developed for

the DARPA Coordinators program [MSB+08], where instead of anticipating ways that probabilities might change, the recipient would revise its plans only when the commitment provider would send an updated probability of the commitment being satisfied. These prior approaches however only treat commitment probabilities as predictions about how the provider's plan will affect recipients. In contrast, our goal is that probabilistic commitments not only provide such predictive information to the recipient, but also impose prescriptive semantics on the provider to influence its behavior into a good faith effort towards making those predictions come true.

Our work, summarized in this thesis, is the first to develop prescriptive commitment semantics under decision-theoretic model uncertainty, along with algorithms that operationalize this semantics for faithful commitment pursuit. The model uncertainty that we consider in this thesis is a form of partial observability, and thus the algorithms we develop can be viewed as extensions of existing techniques for solving (unconstrained) partially-observable Markov decision problems [SS73, KLC98, Han99]. Our commitment semantics of Equation (3.4) prescribes additional constraints to the original planning problem, and we develop algorithms that exactly meet the commitment constraints under partial observability. Existing work has developed methods for constrained decision-theoretic planning without model uncertainty [Alt99], or has solved the constraints only approximately [PMP+15, STW16]. Others have also developed planning approaches for given commitments formulated using formal logic, which mainly rely on techniques of heuristic search (e.g., [TMS13, MTYS15, MMS+18]). These approaches usually amount to enumerating courses of action in search for conditions that ensure the feasibility of the commitments. For example, Meneguzzi et al. [MMS+18] develop a depth-first search algorithm to generate realizable enactments of the commitment. These logic-based planning techniques deal with the provider's uncertainty about the outcomes of its actions, while we also consider the provider's uncertainty over the rewards and dynamics of its environment.

3.2 The Recipient's Robust Interpretation

As we have discussed in Section 2.3, the commitment specification (u_c, T_c, p_c) provides partial, and also the only, information the recipient has about the provider's influence P_u^{r}. As elaborated in Section 2.3.3, while specifying just a single time-probability constraint for the provider gives it more flexibility than a more detailed specification, doing so also increases the uncertainty for the recipient. This thesis considers a popular procedure in the literature [WD07, WD10, ZDS+16] for the re-

cipient to exploit such partial information to plan, which we describe next. The recipient adopts a strategy, denoted as $\widehat{P}^{\mathrm{r}}_u(\cdot)$, that maps a given probabilistic commitment c to $\widehat{P}^{\mathrm{r}}_u(c)$ as an approximation of the provider's true influence P^{r}_u, which is then used for planning [WD07, ZDS+16]. Formally, given $\widehat{P}^{\mathrm{r}}_u(c)$, let $\widehat{M}^{\mathrm{r}}(c) = (\mathcal{S}^{\mathrm{r}}, \mathcal{A}^{\mathrm{r}}, \widehat{P}^{\mathrm{r}}(c), R^{\mathrm{r}}, H^{\mathrm{r}})$ be the recipient's approximate model that differs from M^{r} only in terms of the dynamics of u, i.e. $\widehat{P}^{\mathrm{r}}(c) = (P^{\mathrm{r}}_l, \widehat{P}^{\mathrm{r}}_u(c))$. The recipient then plans in $\widehat{M}^{\mathrm{r}}(c)$:

$$v^{\mathrm{r}}(c) = \max_{\pi^{\mathrm{r}}} V^{\pi^{\mathrm{r}}}_{\widehat{M}^{\mathrm{r}}(c)}(s^{\mathrm{r}}_0). \tag{3.5}$$

We refer to $v^{\mathrm{r}}(c)$ as the recipient's commitment value function. For the remainder of this section, we will abbreviate $\widehat{P}^{\mathrm{r}}_u(c)$, $\widehat{P}^{\mathrm{r}}(c)$, and $\widehat{M}^{\mathrm{r}}(c)$ as $\widehat{P}^{\mathrm{r}}_u$, $\widehat{P}^{\mathrm{r}}$, and $\widehat{M}^{\mathrm{r}}$ whenever the dependency on commitment c is clear from context.

We are interested in the quality of the policy computed from approximate influence $\widehat{P}^{\mathrm{r}}_u$, i.e. $\pi^*_{\widehat{M}^{\mathrm{r}}}$ as the solution to Equation (3.5), when evaluated in M^{r} with the (true) influence P^{r}_u. Formally, the gap between the optimal value for M^{r} and the value of policy $\pi^*_{\widehat{M}^{\mathrm{r}}}$ evaluated in M^{r} is defined as the suboptimality of the approximate influence, i.e.

$$\text{Suboptimality}\left(\widehat{P}^{\mathrm{r}}_u(\cdot); P^{\mathrm{r}}_u, c\right) = V^*_{M^{\mathrm{r}}}(s^{\mathrm{r}}_0) - V^{\pi^*_{\widehat{M}^{\mathrm{r}}}}_{M^{\mathrm{r}}}(s^{\mathrm{r}}_0).$$

Since the recipient is uncertain about P^{r}_u because it is entirely determined by the provider, the recipient should adopt a strategy $\widehat{P}^{\mathrm{r}}_u(\cdot)$ that robustly induces low suboptimality for an arbitrary (P^{r}_u, c) pair. This problem is referred to as the recipient's robust interpretation of the commitment. Specifically, this thesis studies the following three notions of robustness:

1. Worst-case suboptimality. We are interested in finding a strategy $\widehat{P}^{\mathrm{r}}_u(\cdot)$ that induces low suboptimality for a worst (P^{r}_u, c) pair, i.e.

$$\min_{\widehat{P}^{\mathrm{r}}_u(\cdot)} \max_{P^{\mathrm{r}}_u, c} \text{Suboptimality}\left(\widehat{P}^{\mathrm{r}}_u(\cdot); P^{\mathrm{r}}_u, c\right).$$

2. Suboptimality for a general (P^{r}_u, c) pair. We are interested in finding a strategy $\widehat{P}^{\mathrm{r}}_u(\cdot)$ that induces low suboptimality for a general (P^{r}_u, c) pair which is drawn from an underlying distribution, i.e.

$$\min_{\widehat{P}^{\mathrm{r}}_u(\cdot)} E_{P^{\mathrm{r}}_u, c}\left[\text{Suboptimality}\left(\widehat{P}^{\mathrm{r}}_u(\cdot); P^{\mathrm{r}}_u, c\right)\right].$$

3. Suboptimality for a rational (P_u^{r}, c) pair. We are interested in finding a strategy $\widehat{P}_u^{\mathrm{r}}(\cdot)$ that induces low suboptimality when the true influence P_u^{r} is determined by the provider's optimal policy, and the commitment c is chosen to be either a local or a joint value maximizer, i.e.

$$\min_{\widehat{P}_u^{\mathrm{r}}(\cdot)} E_{M^{\mathrm{p}}, M^{\mathrm{r}}} \left[\text{Suboptimality} \left(\widehat{P}_u^{\mathrm{r}}(\cdot); P_u^{\mathrm{r}}, c \right) \right],$$

where P_u^{r} is determined by $\pi^*_{M^{\mathrm{p}}}$, and c maximizes $v^{\mathrm{p}}(c)$, $v^{\mathrm{r}}(c)$, or $v^{\mathrm{p}}(c) + v^{\mathrm{r}}(c)$.

In Chapter V, this thesis focuses on these notions of robustness for two types of probabilistic commitment, achievement and maintenance, that are commonly studied in the literature. In an achievement commitment, the provider promises to change the shared state features of the state in a way desired by the recipient. In a maintenance commitment, the provider instead promises not to change features that are already the way the recipient wants them maintained. The chapter presents theoretical analyses and empirical results showing that, perhaps surprisingly, despite strong similarities in the provider's modeling of the two types of commitment, there is an inexpensive strategy for achievement that satisfies all the three notions of robustness for the recipient's interpretation, while no such strategy exists for maintenance.

Related Work

As we have discussed in Section 3.1, others have adopted alternative frameworks, such as conditional commitments and contracting frameworks, for managing the uncertainty when the commitment is being pursued. In this vein, there has been substantial work for developing protocols for agents who are modeling and communicating about commitments. The focus is on the lifecycle of a commitment, from its initial proposed creation, to the mutual agreement to adopt it, to determining whether it has been fulfilled, to whether it is time to abandon it. Over the lifecycle, it is important that interacting agents engage in a communication protocol that ensures their beliefs about the status of a shared commitment are aligned. In this thesis, we adopt the probabilistic commitment framework to study both achievement and maintenance commitments, and focus just on the "detached" stage of the commitment lifecycle where an agreed-upon commitment is being actively pursued, and the pursuit requires a sequence of actions, where some might not have desired outcomes, or an agent's priorities could change in the midst of executing the sequence.

Even though the probabilistic commitment representations of, and reasoning methods for, achievement and maintenance are nearly identical, prior work has found it

much harder to successfully coordinate for maintenance than achievement [CS08, GMDB08, Hia09]. In the past, it has been assumed that the difficulty lies on the provider's side—that it might be inherently harder for a provider to find good policies that maintain a feature than to change it. However, in this thesis we claim and justify that instead the challenge actually lies on the recipient's side: that a maintenance commitment is fundamentally harder for the recipient to interpret in a robust manner than an achievement commitment is. In Chapter V, we substantiate this claim theoretically and empirically. We begin by analyzing an intuitive and straightforward strategy, adopted in previous work [WD07, WD10, ZDS+16], where the recipient models an achievement commitment pessimistically by assuming the feature will not (probabilistically) attain its desired value any earlier than the commitment's promised time. We show analytically that the worst-case suboptimality induced by such pessimism can be bounded fairly tightly. For the maintenance counterpart, however, we show that no comparable pessimistic model, and hence no bound on suboptimality, exists.

3.3 Efficient Formulation of Cooperative Commitments

In the previous two sections, we have formally defined the problems of the provider's adherence under model uncertainty and the recipient's robust interpretation for an arbitrary commitment, which arise from the commitment execution phase. In this section, we turn to the earlier phase that formulates a commitment the two agents agree on.

As will be formally stated in Chapter VI, the provider would prefer a weaker commitment (e.g., lower commitment probability) to increase its value because the prescriptive commitment semantics constrains its policy choice; on the other hand, the recipient would prefer a stronger commitment (e.g., higher commitment probability) since the outcome specified by the commitment is desired by the recipient. In this thesis, we consider the scenario in which the two agents are cooperative and would agree on the commitment that maximizes their summed commitment value:

$$\max_{c} v^{\mathrm{p}}(c) + v^{\mathrm{r}}(c).$$

Specifically, this thesis considers cooperative commitment formulation in both centralized and decentralized settings. In the centralized formulation process, there exists a centralized coordinator that knows precisely the specifications of both agents' MDPs,

and aims to compute the optimal cooperative commitment that maximizes the joint commitment value. Such a coordinator does not exist in the decentralized formulation process, where neither agent has full knowledge about the other's MDP.

The cooperative commitment formulation is an optimization problem over the space of the commitment tuple $c = (u_c, T_c, p_c)$. This thesis focuses on the optimization over the joint space of $(T_c, p_c) \in [H] \times [0, 1]$ with fixed u_c, where $[H] = \{1, 2, ..., H\}$. Formally, in the centralized setting, we aim to solve problem

$$\max_{c=(T_c,p_c)\in[H]\times[0,1]} v^{\mathrm{p}}(c) + v^{\mathrm{r}}(c)$$

with the provider's MDP M^{p} and the recipient's MDP M^{r} fully known to the centralized coordinator, where we use abbreviation $c = (T_c, p_c)$ since u_c is fixed. A naïve strategy for solving the problem is to discretize the commitment probability space $[0, 1]$, and evaluate every commitment probability in the discretized probability space for every commitment time. The finer the discretization is, the more commitments are considered and the better the solution will be. At the same time, the finer the discretization, the larger the computational cost of evaluating every commitment in the discretized commitment space. In Chapter VI, we prove several structural properties of the joint space of $[H] \times [0, 1]$, which enable us to develop a centralized algorithm that efficiently searches for the optimal commitment.

In the decentralized setting, we assume each agent fully knows its own MDP but only partially knows the other's MDP, and they aim to find a jointly-preferred commitment via communication. As a communication scheme, we consider a querying approach where one agent poses a query consisting of information about a set of feasible commitments, and the other responds by selecting the commitment from the set that has the highest joint commitment value. To limit costs for communication and computation, the set of commitments in the query is small. A query poser thus should optimize its choices of commitments to include, and the responder's choice should reflect joint value. In general, either the provider or recipient could be responsible for posing the query. In this thesis, though, we always assign the provider to be the query poser and the recipient to be the responder, because the set of feasible commitments is known only to the provider. Since our aim in this thesis is for the agents to successfully converge quickly, after just a single query-response round, the responder's selected commitment must be feasible, which means the poser must only offer feasible commitments. Only the provider can do this. In Chapter VI, we solve the provider's querying problem of formulating a high-quality commitment

query, such that the two agents will be able to agree on an approximately optimal commitment after querying.

Related Work

In this work, we assume agents are coordinating through the commitment framework. The commitment framework is general, in that it can support both self-interested agents and cooperative agents. Much of the literature considers the self-interested case, focusing on issues of the commitment lifecycle [DNS08, GLZ16, POM17], and on managing reputation and establishing trust among agents [SS02, CBG02, RHJ04, HJS06, PSM13, GBL+15].

When agents are cooperative, the focus of agreeing on a commitment shifts from strategic reasoning to joint optimization. The agents want to form a commitment to maximize the combined rewards of their plans. Of course, there is considerable literature on cooperative multiagent planning focused exactly on the problem of maximizing joint reward [LR00, KK02, BGIZ02, NTY+03, PL05]. Of that literature, the closest prior work to our problem of centralized commitment formulation is that of Witwicki et al. [WD07, WD10, OWK12], whose approach exploited the asymmetric influence relationship between an agent (in our terminology, the provider) that affects the state of another, and an agent (the recipient) that relies upon the state changes. That work showed how the agents could improve the efficiency when maximizing joint reward by abstracting their policies into subsets.

In contrast to that work, and the larger literature on cooperative multiagent planning, cooperative agents that can only coordinate through commitments will generally achieve lower joint reward, because the commitment specification contains less information than the specifications used in cooperative planning systems, as we have discussed in Section 2.3.

When the commitment formulation process is decentralized, it will involve message passing. The literature on message-passing search techniques is large (e.g., [Dec87, Dur99, GKP02, Rob04, BDAMY13]). The message passing between our decision-theoretic agents serves the purpose of preference elicitation, which is typically framed in terms of an agent querying another about which from among a set of choices it most prefers [CKP00, Bou02, BPPS06, VB10, RD11]. Thus, we adopt a querying protocol. In particular, we draw on recent work that uses value-of-information concepts to formulate multiple-choice queries [Bou02, VB10, CSD14, ZDS17], but as we will explain we augment prior approaches by annotating offered choices with the preferences of the agent posing the query.

CHAPTER IV

Trustworthy Adherence to Probabilistic Commitments

As we have discussed in Chapter III, if the provider knows precisely the model of its environment (i.e. the specifications of its MDP), it is able to compute policies (i.e. mapping from states to actions) that satisfy the commitment's probabilistic guarantee, and then chooses from them the one that yields the most value. This chapter considers the problem formulated in Section 3.1, where the provider, at the time of making a probabilistic commitment, has uncertainty about the model, or specifically, about the transition function and reward function of its MDP, and thus it is not able to directly perform that computation. An immediate question arises due to model uncertainty: What does it mean for the provider to adhere to its probabilistic commitment in a trustworthy manner, if the provider is uncertain about the probability and the value of its policy realizing the commitment at the time of making it? In Section 3.1, we answered this question for the setting in which the provider's model uncertainty is Bayesian: a trustworthy provider is required to follow a (in general history-dependent) policy that realizes the commitment with sufficient probability with respect to its Bayesian prior at the time of making the commitment. This semantics, as formally defined in Definition III.2, preserves the commitment's predictive value for coordination with the recipient, and moreover, since the provider is allowed to adopt a history-dependent policy that chooses the next action based on the previous experience, it can respond to evolving knowledge about its model to improve its utility without undermining its trustworthiness. In this chapter, we develop tractable methods for the provider to construct policies that respect the commitment semantics.

4.1 Problem Statement Recapitulation

We begin by revisiting the provider's commitment semantics under Bayesian model uncertainty and its policy optimization problem, as formulated in Section 3.1. As the discussion will be focused on the provider only, we will drop superscripts p for the notations in this chapter. We consider the setting in which the provider's true sequential decision-making problem is one out of K possible MDPs drawn from a known prior distribution μ_0, where all MDPs share identical state and action spaces but possibly different transition and reward functions, and the state and the reward are fully observable during execution. Formally, the environment is defined by the tuple $(\mathcal{S}, \mathcal{A}, \{P_k, R_k\}_{k=1}^{K}, s_0, \mu_0, H)$, and the MDP that the provider is in is drawn from the known prior distribution, i.e. $M_k \sim \mu_0$. For such model uncertainty, as we have discussed in Section 2.2, we consider history-dependent stochastic policies that map the history up to time step t,

$$h_t = (s_0, a_0, r_1, s_1, ..., s_{t-1}, a_{t-1}, r_t, s_t),$$

to a probability distribution over the next action. Specifically, we use $\pi(a|h)$ to denote the probability of choosing action a given history h when following policy π.

For the provider operating under such Bayesian model uncertainty, Definition III.2 in Section 3.1 formally gives the semantics of a probabilistic commitment: knowing that it is facing an MDP drawn from the prior distribution μ_0 over possible MDPs in the environment ($M_k \sim \mu_0$), the provider is constrained to follow a (in general history-dependent) policy π, such that, starting at the initial state s_0, the probability of reaching a state with commitment feature value u_c at the commitment time T_c is at least the commitment probability p_c. This semantics is prescriptive by constraining the provider's choice of its policy, and thus it secures the commitment's predictive value for coordination with the recipient. Knowing that the provider follows a policy respecting the semantics, the recipient can plan accordingly by predicting how likely the shared state feature u will take value u_c at time T_c. As previously said, this is the first prescriptive commitment semantics under decision-theoretic model uncertainty.

As defined in Section 3.1, we let Π_c be the set of all history-dependent policies satisfying the constraint of commitment c and say that commitment c is feasible if and only if Π_c is not empty. We are interested in finding a policy that maximizes the initial state value with respect to the prior, while satisfying the constraint of a given feasible probabilistic commitment, which is formally formulated as the following

problem:

$$\arg\max_{\pi \in \Pi_c} V^{\pi}_{\mu_0}(s_0). \tag{4.1}$$

For the remainder of this chapter, we develop tractable solutions to this problem.

4.2 Methods

This section describes several methods for constructing policies with different tradeoffs between solution quality and computational cost, while all the constructed policies are guaranteed to be in Π_c to respect the semantics of a given commitment c. In order to achieve high expected cumulative reward, the provider has to plan not only with fully observable states but also with the most recent knowledge about the true MDP it is in. Our first method, Commitment Constrained Full Lookahead (CCFL), finds the optimal policy in set Π_c by generating beforehand all possible posterior distributions over possible MDPs up to the finite time horizon. As a downside, since the number of posterior distributions generally grows exponentially as the time horizon grows, planning with all possible posterior distributions can make CCFL computationally infeasible. To this end, our Commitment Constrained Lookahead (CCL) method, generalizes CCFL by taking as input an integer parameter, L, as the number of time steps for posterior lookahead. Our Commitment Constrained No-Lookahead (CCNL) method can be treated as a special case of CCL, in which $L = 0$, and therefore actions are chosen only based on the initial conditions and ignoring posterior distributions. A small L often saves a lot of computational time compared to full lookahead, but by being more myopic decreases the expected cumulative reward. To partially mitigate this shortcoming of CCL (at the cost of a more modest increase in computation), we create an iterative version of it, referred to as Commitment Constrained Iterative Lookahead (CCIL), which reapplies the CCL method in the midst of execution, where the posterior lookahead of successive applications of CCL reach closer to the time horizon.

Commitment Constrained Full Lookahead

During execution, the provider can use the knowledge provided by the history so far to infer which MDP is more/less likely to be the true MDP it is facing. Formally, one can summarize current history h into a belief, $b := \langle s, \mu \rangle$, where s is the provider's current physical state, and μ is the posterior distribution over all possible MDPs

given h. We use b_t to denote the belief given history h_t. The provider can find the optimal history-dependent policy by planning in the belief MDP defined as the tuple $\langle \mathcal{B}, \mathcal{A}, b_0, \tilde{P}, \tilde{R} \rangle$, where $\mathcal{B}$ is the set of all beliefs reachable from initial belief $b_0 = \langle s_0, \mu_0 \rangle$, which is finite because every possible true MDP k is finite and the time horizon is finite. $\tilde{P}$ and $\tilde{R}$ are belief transition and reward functions, respectively. Specifically, if we let $b|(a, r, s')$ be the belief after taking action a in belief state b, receiving reward r and transiting to state s', then the probability of transiting to any belief $b' \in \mathcal{B}$ after taking action a in belief state b can be expressed as

$$\tilde{P}(b'|b, a) = \sum\nolimits_{\{r, s': b|(a,r,s')=b'\}} \Pr(r, s'|b, a),$$

where $\Pr(r, s'|b, a)$ is the probability of receiving reward r and transiting to state s' after taking action a in belief b and can be expressed using $\{P_k, R_k\}_{k=1}^K$ as

$$\Pr(r, s'|b, a) = \Pr(r, s'|\langle s, \mu \rangle, a) = \sum\nolimits_{k=1}^{K} \mu_k P_k(s'|s, a) 1_{\{r=R_k(s,a)\}}.$$

In words, given any belief $b' \in \mathcal{B}$, $\tilde{P}(b'|b, a)$ sums up probabilities over transitions (r, s') which update the belief to b'. Similarly, the belief reward function can be defined as

$$\tilde{R}(b, a) = \tilde{R}(\langle s, \mu \rangle, a) = \sum\nolimits_{k=1}^{K} \mu_k R_k(s, a).$$

Our Commitment Constrained Full Lookahead (CCFL) method finds an optimal policy in Π_c among all belief-based policies, i.e., policies that choose actions as a function of the current belief, while satisfying the commitment constraint. Note since a belief is a function of the history, then a belief-based policy also gives action probabilities as a function of the history. For MDP k, each policy π has a corresponding occupancy measure y_k^π for the expected number of times action a will be taken in belief-state b over the time horizon H:

$$y_k^\pi(b, a) = E\left[1_{\{b_t=b, a_t=a\}} | b_0; k, \pi\right]$$

where t is such that $s \in \mathcal{S}_t$ for $b = \langle s, \mu \rangle$. We will use shorthand notation y_k in place of y_k^π when policy π is clear from the context. If π is a belief-based policy, it can be recovered from its belief-action occupancy measure in any MDP k via

$$\pi(a|b) = \frac{y_k(b, a)}{\sum_{a'} y_k(b, a')}. \tag{4.2}$$

$$\max_{y} \sum_{b,a} y(b,a)\tilde{R}(b,a) \tag{4.3a}$$

$$\text{subject to } \forall b,a \quad y(b,a) \geq 0; \tag{4.3b}$$

$$\forall b' \quad \sum_{a'} y(b',a') = \sum_{b,a} y(b,a)\tilde{P}(b'|b,a) + \delta(b',b_0); \tag{4.3c}$$

$$\sum_{b_{T_c}:u_c \in s_{T_c}} \sum_{a} y(b_{T_c},a) \geq p_c. \tag{4.3d}$$

Figure 4.1: CCFL program.

CCFL solves the mathematical program shown in Figure 4.1, which introduces as decision variables the belief-action occupancy measure for all possible MDPs, and constructs the policy via Equation (4.2) using the program's optimal solution. The CCFL program is a straightforward adaptation of the linear program in Figure 3.2 that solves an MDP. Constraints (4.3b) and (4.3c), which are the counterparts of constraints (3.3b) and (3.3c) in Figure 3.2, guarantee that y is a valid occupancy measure with the initial belief being b_0 and the transition function being $\tilde{P}$. The expected cumulative reward is expressed using y in the objective function (4.3a), which is the counterpart of objective (3.3a). The commitment semantics of Equation (3.4) imposes an additional constraint (4.3d), which is the counterpart of objective (3.3d).

Because the belief is a sufficient statistic (i.e. it provides as much information for predicting the future as the history does), the CCFL program is feasible if the commitment is feasible, and the policy constructed by CCFL is optimal among all history-dependent policies satisfying the commitment constraint, as formally stated in Theorem IV.1.

Theorem IV.1. *If commitment c is feasible, meaning $\Pi_c \neq \emptyset$, then the CCFL program in Figure 4.1 is also feasible. Let y^* be an optimal solution to the CCFL program. The policy constructed via Equation* (4.2) *using y^* is optimal with respect to the problem in Equation* (4.1)*.*

The proofs of theorems in this section are presented at the end of this section.

Commitment Constrained No-Lookahead

Planning with all possible posterior distributions can make CCFL computationally infeasible. To counter this, we now consider policies that ignore this posterior

knowledge and only depend on the current state to choose actions. We refer to them as Markov policies and let Π_0 be the set of all Markov policies. If commitment c is feasible for Markov policies, i.e., $\Pi_c \cap \Pi_0 \neq \emptyset$, our Commitment Constrained No-Lookahead (CCNL) method will find an optimal Markov policy that maximizes expected cumulative reward satisfying the commitment constraint, which is a solution to the following problem:

$$\arg\max_{\pi \in \Pi_c \cap \Pi_0} V^{\pi}_{\mu_0}(s_0). \tag{4.4}$$

Note that Π_0 is a subset of all history-dependent policies. When, as would generally be the case, Π_0 is a much smaller policy set, the computational cost of CCNL would be much less than that of CCFL, but the solution policy of CCNL is only an approximation of the optimal commitment constraint-satisfying policy yielded by CCFL.

Similar to the belief-action occupancy measure, for MDP k, any policy π has a corresponding occupancy measure x^{π}_k of state-action pairs:

$$x^{\pi}_k(s,a) = E\left[1_{\{s_t=s, a_t=a\}} | s_0; k, \pi\right]$$

where t is such that $s \in \mathcal{S}_t$. We will use shorthand notation x_k in place of x^{π}_k when policy π is clear from the context. If π is a Markov policy, it can be recovered from its state-action occupancy measure in any MDP k via

$$\pi(a|s) = \frac{x_k(s,a)}{\sum_{a'} x_k(s,a')}. \tag{4.5}$$

CCNL constructs the policy by solving the mathematical program shown in Figure 4.2. It introduces as decision variables the state-action occupancy measure for all possible MDPs. Constraints (4.6b) and (4.6c), as counterparts of constraints (3.3b) and (3.3c), guarantee that x_k is a valid occupancy measure with the initial state being s_0 and the transition function being P_k. The commitment semantics of Equation (3.4) is explicitly expressed in constraint (4.6e), which is the counter counterpart of constraint (3.3d). The expected cumulative reward is expressed using x in the objective function (4.6a), where $\mu_{0,k}$ is the probability that the true MDP is k according to μ_0. The corresponding Markov policy can be derived via Equation (4.5). Unlike CCFL, the CCNL program is no longer a straightforward adaptation of the linear program in Figure 3.2 because a challenging problem here is to ensure that these K sets of occupancy measures all derive the same Markov policy. To this end, we use

$$\max_{x} \sum_k \mu_{0,k} \Big(\sum_{s,a} x_k(s,a) R_k(s,a) \Big) \tag{4.6a}$$

$$\text{subject to } \forall k, s, a \quad x_k(s,a) \geq 0; \tag{4.6b}$$

$$\forall k, s' \quad \sum_{a'} x_k(s',a') = \sum_{s,a} x_k(s,a) P_k(s'|s,a) + \delta(s', s_0); \tag{4.6c}$$

$$\forall k, k', s, a \quad \frac{x_k(s,a)}{\sum_{a'} x_k(s,a')} = \frac{x_{k'}(s,a)}{\sum_{a'} x_{k'}(s,a')}; \tag{4.6d}$$

$$\sum_{s_{T_c} \ni u_c} \sum_a \Big(\sum_k \mu_{0,k} x_k(s_{T_c}, a) \Big) \geq p_c. \tag{4.6e}$$

Figure 4.2: CCNL program.

constraint (4.6d) to enforce alignment across all K sets of occupancy measures. The constraints in Figure 4.2 are feasible if and only if $\Pi_c \cap \Pi_0 \neq \emptyset$.

Commitment Constrained Lookahead

CCFL pre-plans for every possible revision to the provider's posterior knowledge about the true MDP it might be in, which guarantees optimality but possibly at a huge computational cost. At the other extreme, CCNL only considers Markov policies that ignore this evolving posterior knowledge. Here we consider the general case where the provider plans its first $L \in [0, H]$ actions as a function of the evolving belief, and thereafter plans actions based on the evolving state but with the belief (including both the state and the posterior distribution) the provider was in at time L. We refer to this parameter, L, as the belief-update lookahead boundary, which tells the planner how far beyond the current time to look ahead about states and posterior distributions. The resulting *L-updates* policy takes the form:

$$\pi(a|h_t) = \begin{cases} \pi(a|b_t) & t < L \\ \pi(a|s_t, b_L) & t \geq L \end{cases}$$

where b_t is the belief consistent with h_t, and b_L is the belief consistent with h_L when $t \geq L$. Note that a 0-update policy is the same as a Markov policy and an H-update policy is a full width belief-based policy. Therefore, belief-update lookahead boundary L defines a continuum between CCNL and CCFL.

Given a specific value of L, let Π_L be the set of all L-updates policies. If com-

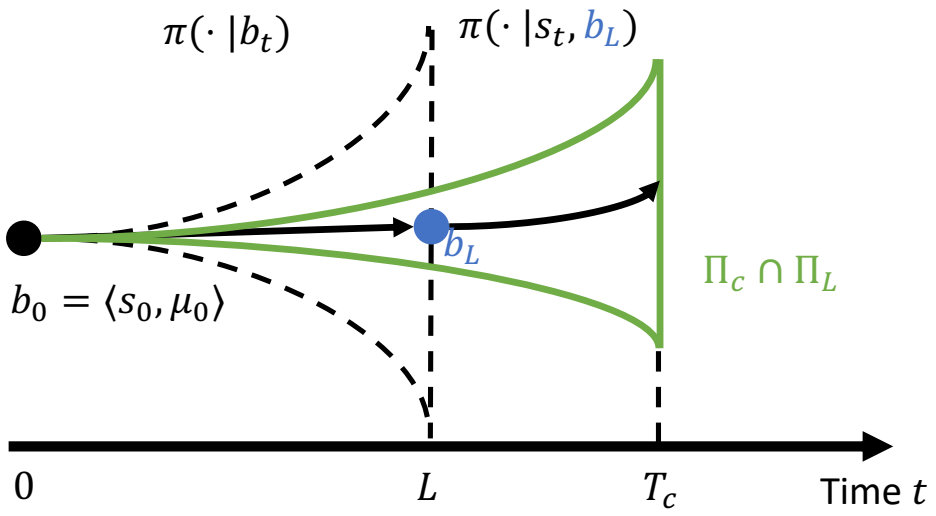


Figure 4.3: Illustration of CCL. The dashed area denotes the set of reachable beliefs after executing L from initial belief $b_0 = \langle s_0, \mu_0 \rangle$. The solid area (in green) denotes the set of commitment constrained L-updates policies, i.e. $\Pi_c \cap \Pi_L$, and the solid arrow denotes a specific history derived from such a policy. An L-update policy selects the first L actions based on the evolving belief, i.e. $\pi(\cdot|b_t)$ for $t < L$, and thereafter based on the evolving state and the belief the provider was in at time L, i.e. $\pi(\cdot|s_t, b_L)$ for $t \geq L$.

mitment c is feasible for belief-update lookahead boundary L, i.e., $\Pi_c \cap \Pi_L \neq \emptyset$, our Commitment Constrained Lookahead (CCL) method will find an optimal L-updates policy that maximizes expected cumulative reward satisfying the commitment constraint, which is a solution to the following problem:

$$\arg\max_{\pi \in \Pi_c \cap \Pi_L} V^{\pi}_{\mu_0}(s_0). \tag{4.7}$$

Figure 4.3 illustrates the CCL's construction of the commitment constrained L-updates policies, with the dashed area denoting the set of reachable beliefs after executing L from initial belief b_0, the solid area (in green) denoting the set of commitment constrained L-updates policies $\Pi_c \cap \Pi_L$, and the solid arrow denoting a specific history derived from such a policy. CCL constructs the policy by solving the mathematical program shown in Figure 4.4, which is a novel and carefully-crafted combination of the techniques in CCFL and CCNL. The program introduces as decision variables y and x, where y is the belief-action occupancy measure (as defined for CCFL) for those beliefs reachable within the first L time steps of the plan, and x is the state-action occupancy measures (as defined for CCNL) for the remaining time steps to the horizon. We use $\mathcal{B}_l^b$ to denote the set of reachable beliefs after executing

exactly l actions from belief b, and $\mathcal{B}^b_{\leq l} = \bigcup_{l=0}^{l'} \mathcal{B}^b_{l'}$ to denote the set of reachable beliefs from b by executing at most l actions starting from b. Because time is a state feature, $\mathcal{B}^b_l$ and $\mathcal{B}^b_{l'}$ are disjoint if $l \neq l'$. CCL generates beforehand all reachable beliefs from initial belief $b_{(t=)0}$ within L actions, $\mathcal{B}^{b_{(t=)0}}_{\leq L}$, as illustrated as the dashed area in Figure 4.3. The belief-action and state-action measures enable us to express the expected cumulative reward very conveniently in the objective (4.9a) where the first term sums up the reward of the first L time steps, and the second term the remaining time steps to the horizon. The occupancy measures also enable us to express commitment semantics conveniently: if the lookahead does not reach the commitment time T_c, then the commitment semantics can be expressed in terms of the belief-action occupancy measure via constraint (4.9h); otherwise, the commitment constraint can be expressed in terms of those state-action occupancy measures via constraint (4.9i). Constraints (4.9b) and (4.9c) on y are the counterparts of (4.3b) and (4.3c) in the CCFL program of Figure 4.1. Similarly, constraints (4.9e), (4.9f), and (4.9g) on x are the counterparts of (4.6b), (4.6c), and (4.6d) in the CCNL program of Figure 4.2, which means the CCL program is considerably more sophisticated than the original linear program of Figure 3.2. These constraints are feasible if and only if $\Pi_c \cap \Pi_L \neq \emptyset$. Any L-updates policy π_L that respects the commitment semantics can be derived from a feasible solution to the program in Figure 4.4 via:

$$\pi_L(a|h_t) = \begin{cases} \pi_L(a|b_t) = \dfrac{y(b_t, a)}{\sum_{a'} y(b_t, a')} & t < L \\ \pi_L(a|s_t, b_L) = \dfrac{x_{b_L,k}(s_t, a)}{\sum_{a'} x_{b_L,k}(s_t, a')} & t \geq L \end{cases}. \tag{4.8}$$

Theorem IV.2 states that CCL using belief-update lookahead boundary L finds an optimal policy in $\Pi_c \cap \Pi_L$.

Theorem IV.2. *If $\Pi_c \cap \Pi_L \neq \emptyset$ holds for commitment c, then the program in Figure 4.4 is feasible. Let x^*, y^* be its optimal solution, then the policy derived via Equation (4.8) with x^*, y^* is the optimal policy in $\Pi_c \cap \Pi_L$.*

Intuitively, a belief-update lookahead boundary greater than zero enables the provider to plan actions not only based on the states it will visit, but also based on how its actions can provide information to improve its posteriors about what its true MDP is. Sacrifices in short-term reward may ultimately improve long-term performance. Theorem IV.3 says the expected cumulative reward of the policy derived by CCL using any $L > 0$ is lower bounded by that of the policy derived by CCNL. This is because, by definition, for any L and any Markov policy, there exists an L-updates

$$\max_{x,y} \sum\nolimits_{b \in \mathcal{B}^{b_0}_{\leq L-1}, a} y(b,a)\tilde{R}(b,a) + \sum\nolimits_{b_L \in \mathcal{B}^{b_0}_{L}, k, s, a} x_{b_L,k}(s,a) R_k(s,a) \tag{4.9a}$$

subject to

$$\forall b \in \mathcal{B}^{b_0}_{\leq L}, a \quad y(b,a) \geq 0; \tag{4.9b}$$

$$\forall b' \in \mathcal{B}^{b_0}_{\leq L} \quad \sum\nolimits_{a'} y(b',a') = \sum\nolimits_{b,a} y(b,a)\tilde{P}(b'|b,a) + \delta(b',b_0); \tag{4.9c}$$

$$\forall b_L \in \mathcal{B}^{b_0}_{L} \quad y_{b_L} = \sum\nolimits_{a} y(b_L,a); \tag{4.9d}$$

$$\forall b_L \in \mathcal{B}^{b_0}_{L}, k, s, a \quad x_{b_L,k}(s,a) \geq 0; \tag{4.9e}$$

$$\forall b_L = \langle s_L, \mu_L \rangle \in \mathcal{B}^{b_0}_{L}, k, s'$$

$$\sum\nolimits_{a'} x_{b_L,k}(s',a') = \sum\nolimits_{s,a} x_{b_L,k}(s,a) P_k(s'|s,a) + \mu_{L,k} y_{b_L} \delta(s', s_L); \tag{4.9f}$$

$$\forall b_L \in \mathcal{B}^{b_0}_{L}, k, k', s, a \quad \frac{x_{b_L,k}(s,a)}{\sum_{a'} x_{b_L,k}(s,a')} = \frac{x_{b_L,k'}(s,a)}{\sum_{a'} x_{b_L,k'}(s,a')}; \tag{4.9g}$$

$$\sum\nolimits_{b_{T_c} \in \mathcal{B}^{b_0}_{T_c} : s_{T_c} \ni u_c, a} y(b_{T_c}, a) \geq p_c, \text{ if } T_c < L; \tag{4.9h}$$

$$\sum\nolimits_{b_L \in \mathcal{B}^{b_0}_{L}, k, s_{T_c} \ni u_c, a} x_{b_L,k}(s_{T_c}, a) \geq p_c, \text{ if } T_c \geq L. \tag{4.9i}$$

Figure 4.4: CCL program.

policy that behaves exactly the same as the Markov policy, i.e. $\Pi_0 \subseteq \Pi_L$.

Theorem IV.3. *If* $\Pi_c \cap \Pi_0 \neq \emptyset$ *holds for commitment c, then for any integer* $L \in [0, H]$ *the CCL program in Figure 4.4 is feasible, and we have*

$$V_{\mu_0}^{\pi_L^*}(s_0) \geq V_{\mu_0}^{\pi_0^*}(s_0)$$

where π_L^* *and* π_0^* *are the policies derived by CCL using belief-update lookahead boundary L and zero, respectively.*

However, one has to be careful in using deeper boundaries because the performance of CCL is guaranteed to be monotonically non-decreasing in L only when MDPs vary solely in reward functions, but this monotonicity cannot be guaranteed in general, as stated in Theorem IV.4 and Theorem IV.5.

Theorem IV.4. *If MDPs vary in reward functions and not in transition dynamics, i.e.* $\forall k, k', P_k = P_{k'}$, *and* $\Pi_c \cap \Pi_L \neq \emptyset$ *for boundary L, then for any* $L' > L$ *we have* $\Pi_c \cap \Pi_{L'} \neq \emptyset$, *and*

$$V_{\mu_0}^{\pi_L^*}(s_0) \leq V_{\mu_0}^{\pi_{L'}^*}(s_0)$$

where π_L^* *and* $\pi_{L'}^*$ *are the policies derived by CCL using boundaries L and* L', *respectively.*

Theorem IV.5. *There exists an environment, a commitment c, and boundaries* $0 < L < L' < H$ *satisfying* $\Pi_c \cap \Pi_L \neq \emptyset$ *and* $\Pi_c \cap \Pi_{L'} \neq \emptyset$, *such that*

$$V_{\mu_0}^{\pi_L^*}(s_0) > V_{\mu_0}^{\pi_{L'}^*}(s_0)$$

where π_L^* *and* $\pi_{L'}^*$ *are the policies derived by CCL using belief-updates boundaries L and* L', *respectively.*

These theoretical results provide some insights when choosing L. If the transition dynamics do not vary across MDPs, as suggested by Theorem IV.4, Π_L is monotonically increasing in L. One should use the largest affordable L because a larger L is likely to include more policies in Π_c and improve the value. A commitment that is infeasible for a smaller L could be feasible for a larger L. In general, though, the transition dynamics can vary across MDPs, and Π_L is not guaranteed to be monotonically increasing in L. One should use CCFL if it is affordable. CCFL considers all policies in Π_c if it is non-empty and therefore it yields optimal value. When CCFL is

not affordable, then as suggested by Theorem IV.3 we can check the feasibility of a commitment with CCNL because a commitment feasible to CCNL (i.e. $\Pi_c \cap \Pi_0 \neq \emptyset$) is also feasible for any L. For the empirical results in Section 4.3, we experiment with several candidate values of L. Our experience suggests that L can best be chosen with problem-specific knowledge.

Commitment Constrained Iterative Lookahead

At each time step during execution, the provider observes the state transition that occurs and reward received to update its posterior μ about the true MDP it is in. One might think it would be a good idea for the provider to construct and follow an updated policy from its current state, substituting its updated belief state for the initial belief. However, the provider cannot shift from one policy to another without considering its commitment. Clearly, if the provider can find a plan that achieves the original commitment probability conditioned on the current belief, then shifting to such a plan will certainly respect the commitment semantics. Observation IV.1 says this re-planning is not always feasible.

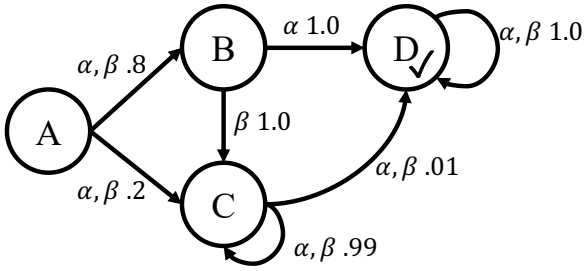
Observation IV.1. There exists an environment, a feasible commitment c, a policy $\pi \in \Pi_c$, and a history h_t induced by π, such that

$$\forall \pi' \quad \Pr_{k \sim \mu_t}(u_c \in s_{T_c} | s_t, k; \pi') < p_c,$$

where $\langle s_t, \mu_t \rangle$ is the belief consistent with h_t.

The example shown in Figure 4.5 verifies Observation IV.1. Starting in state A, the provider can feasibly commit to reaching the absorbing state D at time step 2 with at least probability .8. If the provider stochastically reached state C at time step 1, there is no plan that reaches state D from state C with probability at least .8, and this verifies Observation IV.1.

Our Commitment Constrained Iterative Lookahead (CCIL) method instead updates the commitment probability in a way that guarantees feasible re-planning, and iteratively applies CCL with that updated commitment probability during execution. The idea is that, when re-planning, satisfying the commitment constraint does not require meeting the original probabilistic commitment, but instead to *fulfill the commitment probability that had originally been associated with the physical-state history traversed so far*. Here we formally describe CCIL's first iterative application of CCL after having executed one or more actions. Suppose the provider now has belief



	R_1	R_2
α	0	1
β	1	0

Figure 4.5: The example that verifies Observation IV.1. There are two possible reward functions R_1 and R_2 shown above with 50-50 prior. In both reward functions, the reward only depends on the action. There are two actions, α and β, and the transition dynamics is shown in the annotations of the edges. Starting in A, the provider commits to reaching the absorbing location D at time step two with at least probability .8. If the provider happens to be in C at time step one, there is no plan that reaches D from C with probability at least .8 (verifying Observation IV.1). Even though re-planning from C does not yield a plan that leads to D with probability 0.8, the new plan will nonetheless yield more reward because at time step one we will know which reward function applies and can therefore choose the more rewarding action in C.

$b_t = \langle s_t, \mu_t \rangle$ at time step $t \leq L$ after following policy π_L^* derived from the initial optimal solution to the CCL program with belief-update lookahead boundary L. Now the provider re-plans from s_t using its updated posterior μ_t, with the commitment probability that its previous policy π_L^* ascribed to meeting the commitment if state s_t were reached:

$$p_{c,t} = \Pr_{k \sim \mu_t}(u_c \in s_{T_c} | s_t, k; \pi_L^*). \tag{4.10}$$

Specifically, the provider constructs and follows a new L-updates policy, beginning from the current belief, by reusing the CCL program in Figure 4.4 with the following modifications:

1. Start from current belief $b_t = \langle s_t, \mu_t \rangle$ instead of $b_0 = \langle s_0, \mu_0 \rangle$.

2. Let $L \leftarrow \min(L, H - t)$ to ensure that the lookahead from the current time step is bounded by the time horizon, i.e. $t + L \leq H$.

3. If the provider has not reached the commitment time, i.e. $t < T$, plan with the updated commitment probability by replacing p_c with $p_{c,t}$ calculated as in Equation (4.10) in constraint (4.9h) if $T < t + L$ or in constraint (4.9i) if $T \geq t + L$; otherwise, discard constraints (4.9h) and (4.9i) (e.g., let $p_{c,t} = 0$).

Revisiting the example in Figure 4.5, the initial policy could meet the commitment probability (0.8) by committing to take action α with probability 1 if B is reached at time 1, and otherwise the provider is unconstrained. After taking action α (or β) at time 0, then at time 1 the provider is either in B or C, and from the reward it just received knows the true reward function. Using CCIL, the provider re-plans. If it is in B, then since the original policy attributed probability 1 to meeting the commitment down this path, its new policy is constrained to take action α (whatever the true reward is), and afterwards take the better action. If it is in C, the updated commitment probability is zero (the original policy did not count at all on possibly meeting the commitment down this path), so the new policy can optimize reward without constraints.

In principle, the provider can iteratively apply the above procedure at any time during execution. For example, the provider can apply the procedure whenever the posterior undergoes a substantial change. We will evaluate empirically a simpler version of CCIL that takes as input a pair of integers, (L, I), such that it iteratively uses L as the belief-update lookahead boundary to update the policy every $I \leq L$ steps. This procedure is outlined in Algorithm 1, and Figure 4.6 illustrates CCIL's first iteration with parameter (L, I). Theorem IV.6 proves that CCIL respects our commitment semantics.

Theorem IV.6. *Let π_{IL} be the history-dependent policy defined as in Algorithm 1. We have $\pi_{IL} \in \Pi_c$.*

Dealing with the Quadratic Equality Constraint

The CCFL program in Figure 4.1 is a linear program straightforwardly adapted from the program in Figure 3.2 and thus can be solved by standard linear programming algorithms. The CCL program in Figure 4.4, however, is no longer a straightforward adaptation of Figure 3.2 because it introduces a quadratic equality constraint (4.9g) to ensure that the action selection rules derived from occupancy measures in all possible MDPs are identical. Similarly, the CCNL program in Figure 4.2 also introduces such a quadratic equality constraint (4.6d). These quadratic constraints make the mathematical programs non-convex and hard to solve. In practice, many math-programming solvers are unable to handle programs with quadratic equality constraints (e.g., [CPL, Gur]). Although some solvers can deal with such programs (e.g., [MAT, OPT]), they often need to take as input a feasible solution as the starting point, but finding an initial feasible solution by itself might be difficult, and the

Algorithm 1: Commitment Constrained Iterative Lookahead (L, I)

Input: Environment tuple $(\mathcal{S}, \mathcal{A}, \{P_k, R_k\}_{k=1}^{K}, s_0, \mu_0)$,
commitment $c = \langle u_c, T_c, p_c \rangle$,
integers $L \in [0, H], I \in (0, H]$ such that $\Pi_c \cap \Pi_L \neq \emptyset$ and $I \leq L$;

$b_0 \leftarrow \langle s_0, \mu_0 \rangle$;
$\pi_0 \leftarrow L$-updates policy derived by solving the program in Figure 4.4;
$t \leftarrow 0$;
while $t < H$ **do**
  for $i = 1, 2, ..., I$ **do**
    Take action $a_t \sim \pi_t$ and observe reward-state transition $(s_t, a_t, r_{t+1}, s_{t+1})$;
    Update belief as $b_{t+1} = \langle s_{t+1}, \mu_{t+1} \rangle$;
    $\pi_{t+1} \leftarrow \pi_t$;
    $t \leftarrow t + 1$;
    if $t == H$ **then**
      Break the while loop;
    end
  end
  if $t < T$ **then**
    $p_{c,t} = \Pr_{k \sim \mu_t}(u_c \in s_{T_c} | s_t, k; \pi_t)$;
  end
  else
    $p_{c,t} = 0$;
  end
  $\pi_t \leftarrow$ Policy derived by solving a modified version of the program in Figure 4.4: let $L \leftarrow \min(L, H - t)$; replace every b_0 with b_t; replace p_c with $p_{c,t}$ in constraint (4.9h) if $T < t + L$ or in constraint (4.9i) if $T \geq t + L$;
end

final solutions are usually sensitive to starting points. Here we introduce two variant formulations of the CCL program in Figure 4.4 that avoid quadratic equality constraints.

Deterministic CCL. The policy derived from the program in Figure 4.4 via Equation (4.8) is in general stochastic. To enforce deterministic policies, Dolgov and Durfee [DD04, DD05] introduced binary indicators in the linear programs for solving MDPs. Inspired by their work, we propose a novel formulation that avoids quadratic equality constraints by introducing binary indicators that force the action selection to be deterministic *after* belief-update lookahead boundary L. Specifically, we introduce indicators Δ as additional decision variables into the CCL program in Figure 4.4 with

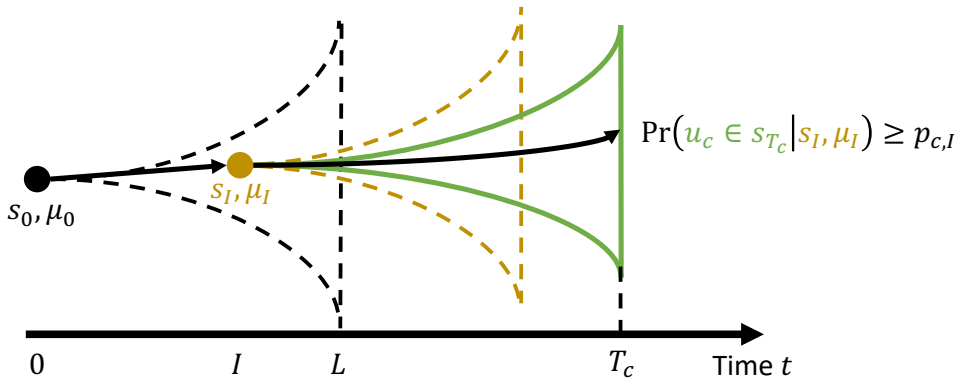


Figure 4.6: Illustration of CCIL's first iteration with lookahead boundary L and iteration parameter I. Following policy π_L^* computed from CCL, the provider executes the first I actions in the dashed area (in black) starting from initial belief $\langle s_0, \mu_0 \rangle$, and arrives in belief $\langle s_I, \mu_I \rangle$. The provider then re-plans from $\langle s_I, \mu_I \rangle$ using $p_{c,I}$, which is defined in Equation (4.10) with $t = I$, with another L time steps of lookahead denoted as the other dashed area (in gold).

the following constraints replacing the quadratic equality constraint (4.9g):

$$\begin{aligned}
&\forall b_L \in \mathcal{B}_L^{b_0}, s, a \quad \Delta_{b_L}(s,a) \in \{0,1\};\\
&\forall b_L \in \mathcal{B}_L^{b_0}, s \quad \sum\nolimits_a \Delta_{b_L}(s,a) \leq 1;\\
&\forall b_L \in \mathcal{B}_L^{b_0}, k, s, a \quad x_{b_L,k}(s,a) \leq \Delta_{b_L}(s,a).
\end{aligned}$$

This reformulation yields a Mixed Integer Linear Program (MILP) which is well studied with many available solvers (e.g., [CPL, Gur, MAT, OPT]). Any feasible solution with the above constraints replacing constraints (4.9g) of the program in Figure 4.4 yields a policy with deterministic action selection at time steps after belief-update lookahead boundary L via Equation (4.8), which can be alternatively expressed using the indicator variables:

$$\pi_L(a|h_t) = \begin{cases} \pi_L(a|b_t) = \dfrac{y(b_t,a)}{\sum_{a'} y(b_t,a')} & t < L \\ \pi_L(a|s_t,b_L) = 1_{\{\Delta_{b_L}(s_t,a)=1\}} & t \geq L \end{cases}. \tag{4.11}$$

Reward uncertainty only. Quadratic equality constraint (4.9g) can be avoided when the transition dynamics do not vary across possible MDPs, i.e. $\forall k, k', P_k = P_{k'}$. In this case, for the action selection at time step $t \in [H-L, H]$, without loss of optimality, the provider needs only to plan for the Bayes-optimal Markov policy

$$
\begin{aligned}
\max_{x,y} \ & \sum\nolimits_{b \in \mathcal{B}^{b_0}_{\leq L-1}, a} y(b,a)\tilde{R}(b,a) + \sum\nolimits_{b_L \in \mathcal{B}^{b_0}_{L}, s, a} x_{b_L}(s,a) R_{\mu_L}(s,a) \\
\text{subject to } & \forall b \in \mathcal{B}^{b_0}_{\leq L}, a \quad y(b,a) \geq 0; \\
& \forall b' \in \mathcal{B}^{b_0}_{\leq L} \sum\nolimits_{a'} y(b',a') = \sum\nolimits_{b,a} y(b,a)\tilde{P}(b'|b,a) + \delta(b', b_0); \\
& \forall b_L \in \mathcal{B}^{b_0}_{L} \quad y_{b_L} = \sum\nolimits_{a} y(b_L, a); \\
& \forall b_L \in \mathcal{B}^{b_0}_{L}, s, a \quad x_{b_L}(s,a) \geq 0; \\
& \forall b_L = \langle s_L, \mu_L \rangle \in \mathcal{B}^{b_0}_{L}, s' \\
& \quad \sum\nolimits_{a'} x_{b_L}(s',a') = \sum\nolimits_{s,a} x_{b_L}(s,a) P(s'|s,a) + y_{b_L}\delta(s', s_L); \\
& \sum\nolimits_{b_L \in \mathcal{B}^{b_0}_{L}, s_{T_c} \ni u_c, a} x_{b_L}(s_{T_c}, a) \geq p_c, \text{ if } L \leq T; \\
& \sum\nolimits_{b_{T_c} \in \mathcal{B}^{b_0}_{T_c} : s_{T_c} \ni u_c, a} y(b_{T_c}, a) \geq p_c, \text{ if } L > T;
\end{aligned}
$$

Figure 4.7: CCL program in the reward uncertainty only case, i.e. $\forall k, k' \ P = P_k = P_{k'}$.

w.r.t. the mean reward R_{μ_L} according to the belief it ended up in at time step L:

$$R_{\mu_L}(s,a) = \sum\nolimits_k \mu_{L,k} R_k(s,a)$$

The resulting mathematical program is shown in Figure 4.7. The main difference from the original CCL program in Figure 4.4 is that it only introduces one occupancy measure x_{b_L} for each reachable belief b_L at time step L, instead of K sets of occupancy measures $\{x_{b_L,k}\}_{k=1}^K$ in the original CCL program. The derived policy can be expressed via:

$$\pi_L(a|h_t) = \begin{cases} \pi_L(a|b_t) = \dfrac{y(b_t,a)}{\sum_{a'} y(b_t,a')} & t < L \\ \pi_L(a|s_t,b_L) = \dfrac{x_{b_L}(s_t,a)}{\sum_{a'} x_{b_L}(s_t,a')} & t \geq L \end{cases}$$

Proofs

Here we present all the technical proofs of the theorems in this chapter.

Proof of Theorem IV.1. Note that the belief is a sufficient statistic: given history h_t at time step t and the corresponding belief b_t consistent with h_t, one does not

need any other information in h_t besides b_t to predict the future state transitions and rewards after time step t. Therefore, solving problem (4.1) is equivalent to solving a constrained MDP, where the MDP is the belief MDP defined as the tuple $\langle \mathcal{B}, \mathcal{A}, b_0, \tilde{P}, \tilde{R} \rangle$ with finite state space of beliefs, and the constraint comes from the semantics of commitment c. Our CCFL method can be viewed as a standard linear programming approach to solving a finite state constrained MDP. □

Proof of Theorem IV.2. It is sufficient to show (1) any policy in $\Pi_c \cap \Pi_L$ can be derived from a feasible solution to the program in Fig. 4.4, and (2) any feasible solution to the program derives a policy in $\Pi_c \cap \Pi_L$.

To show (1), for any policy $\pi \in \Pi_c \cap \Pi_L$, we are going to define vectors m^π and n^π such that with m^π treated as x and n^π treated as y, m^π and n^π satisfy the constraints of the program in Fig. 4.4, and the L-updates policy π can be derived via Equation (4.8). Specifically, given any policy $\pi \in \Pi_c \cap \Pi_L$, let n^π be its belief-action occupancy measure for beliefs in $\mathcal{B}^{b_0}_{\leq L}$, and m^π be its state-action occupancy measure for states from time step L on:

$$\forall b \in \mathcal{B}^{b_0}_{\leq L}, a \quad n^\pi(b, a) = \Pr(b_t = b, a_t = a | b_0; \pi)$$

where t is the time of belief b, and

$$\forall s, a \quad m^\pi_{b_L,k}(s, a) = \begin{cases} \Pr(s_t = s, a_t = a, b_L, k | b_0; \pi) & t \geq L \\ 0 & t < L \end{cases}$$

where t is the time of state s. Then, with m^π treated as x and n^π treated as y, m^π and n^π satisfy the constraints of the program in Fig. 4.4, and the L-updates policy π can be derived via Equation (4.8).

To show (2), given a feasible solution x, y to the program, let policy π be the derived policy via (4.8). Then π is in Π_L by definition. Further we have $m^\pi_{b_L,k}(s, a) = x_{b_L,k}(s, a), n^\pi(b, a) = y(b, a)$, where m^π and n^π are defined as above. Therefore π is also in Π_c because x satisfies commitment constraints (4.9i), (4.9h). □

Proof of Theorem IV.3. By Theorem IV.2, CCL with boundary L finds the optimal policy in $\Pi_c \cap \Pi_L$. Therefore, it is sufficient to show

$$\forall L > 0, \Pi_0 \subseteq \Pi_L.$$

This holds because given any Markov policy $\pi_0 \in \Pi_0$ we can define an L-updates

policy $\pi_L \in \Pi_L$ that is equivalent to π_0:

$$\pi_L(a|h_t) = \begin{cases} \pi_L(a|b_t) = \pi_0(a|s_t) & t < L \\ \pi_L(a|s_t, b_L) = \pi_0(a|s_t) & t \geq L \end{cases}.$$

Thus, we know that $\pi_0 \in \Pi_L$. □

Proof of Theorem IV.4. It is sufficient to show that the statement holds when $L' = L+1$. We next show that when $P_k = P_{k'}\ \forall k, k'$, given any policy $\pi_L \in \Pi_L$, there exists an $(L+1)$-updates policy, π_{L+1}, that mimics π_L , and therefore $V_{\mu_0}^{\pi_L^*}(s_0) \leq V_{\mu_0}^{\pi_{L+1}^*}(s_0)$.

For the first L actions, an $(L+1)$-updates policy can map the current belief to a distribution of the next actions identical to π_L, and the action that is going to be taken at time step L by π_L can also be recovered by an $(L+1)$-updates policy, which gives

$$\pi_{L+1}(a|h_t) = \begin{cases} \pi_{L+1}(a|b_t) = \pi_L(a|b_t) & t < L \\ \pi_{L+1}(a|b_L) = \pi_L(a|s_L, b_L) & t = L \end{cases}.$$

Under any L-updates policy π_L, and conditioned on being in belief b_{L+1} at time step $L+1$, the provider thereafter selects actions according to $\pi_L(\cdot|s_t, b_L)$ with probability that the provider was in belief b_L at time step L: $\Pr(b_L|b_{L+1}; \pi_L)$. If the transition dynamics does not vary across MDPs in the environment, it is well known [Put14] that a Markov policy $\pi_{b_{L+1}}(\cdot|s_t), t \geq L+1$ is sufficient to recover the state occupancy measure of π_L starting at belief b_{L+1}. Then π_{L+1} can also recover π_L for $t \geq L+1$ by demonstrating that $\pi_{b_{L+1}}$ satisfies

$$\pi_{L+1}(a|h_t) = \pi_{L+1}(a|s_t, b_{L+1}) = \pi_{b_{L+1}}(a|s_t) \qquad \text{for } t \geq L+1.$$

This guarantees that the optimal L-updates policy can be represented by an $(L+1)$-updates policy, and thus the statement of the theorem holds for $L' = L+1$. □

Proof of Theorem IV.5. In the proof of Theorem IV.4, we have shown that for any L-updates policy π_L there exists an $(L+1)$-update policy that is able to mimic π_L up to time step $L+1$. Provided that $P_k = P_{k'}\ \forall k, k'$, one can find a Markov policy that mimics π_L starting at any belief at time step $L+1$. When $P_k = P_{k'}\ \forall k, k'$ does not hold, however, this Markov policy in general does not exist, and therefore no $(L+1)$-update policy is able to mimic π_L. Inspired by this, we next give an example as a formal constructive proof.

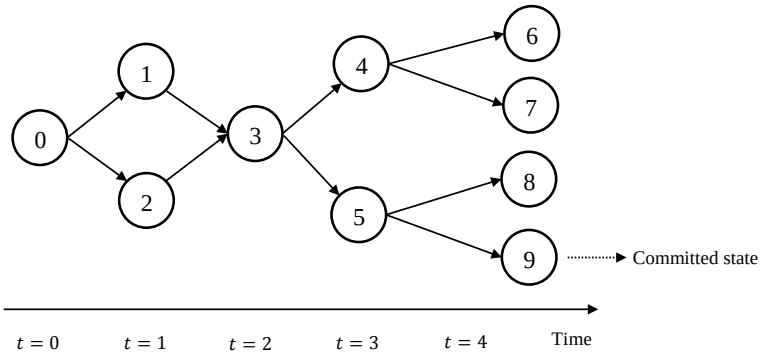


Figure 4.8: Example as a proof of Theorem IV.5.

Consider the example shown in Fig. 4.8. The environment has 10 locations {0,1,...,9}, action space $\{up, down\}$, time horizon $T = 4$, and $K = 2$ possible MDPs. The agent starts in location 0 at time step $t = 0$ with a prior probability of 0.8 for MDP $k = 1$ and a prior probability of 0.2 for MDP $k = 2$. In MDP $k = 1$, no matter which action the provider takes, it transits to location 1 or 2 uniformly at random at time step $t = 1$, and then to location 3 with probability one at time step $t = 2$. Starting from location 3, on taking action *up* (*down*) the provider transits to the upper (lower) location to the right. The transition dynamics of MDP $k = 2$ is the same as MDP $k = 1$ until the provider reaches location 3, and thereafter the transition is flipped: starting from location 3, on taking action *up* (*down*) the provider transits to the lower (upper) location to the right. In both MDPs, the provider will receive large negative reward ($-\infty$) in locations 7 and 8. In MDP $k = 1$, the provider will receive $+1$ reward if it reaches location 6. There is no reward elsewhere. The agent commits to reaching location 9 with probability 0.5. Consider the following ($L =$)1-updates policy: if the provider was in location 1 at time step $t = 1$, always choose action *up*; if the provider was in location 2 at time step $t = 1$, always choose action *down*. Under this ($L =$)1-updates policy the probability of reaching the commitment location 9 is 0.5 and the expected reward is $0.8 \times 0.5 \times 1 = 0.4$. Now consider ($L =$)2-updates policies. Because the provider is in location 3 with probability one at time step $t = 2$, a ($L =$)2-updates policy amounts to a Markov policy for time steps $t \geq 2$. Further the provider should minimize the probability of reaching locations 7 and 8 that yield large negative reward. One can verify that the only Markov policy for time steps $t \geq 2$ that avoids reaching locations 7 and 8 while satisfying the commitment constraint is to always choose action *down*, whose expected reward is 0, smaller than that of the ($L =$)1-updates policy. □

Proof of Theorem IV.6. We need to show π_{IL} satisfies Equation (3.4), i.e.,

$$\Pr_{k\sim\mu_0}(u_c \in s_{T_c}|s_0, k; \pi_{IL}) \geq p_c.$$

Let π_L be the CCL L-updates policy derived from the program in Fig. 4.4. The above inequality holds because:

$$\begin{aligned}
&\Pr_{k\sim\mu_0}(u_c \in s_{T_c}|s_0, k; \pi_{IL}) \\
=&\sum\nolimits_{b_I\in\mathcal{B}_I^{b_0}} \Pr_{k\sim\mu_0}(b_I|s_0, k; \pi_{IL}) \Pr(u_c \in s_{T_c}|b_I; \pi_{IL}) && \text{(law of total probability)} \\
=&\sum\nolimits_{b_I\in\mathcal{B}_I^{b_0}} \Pr_{k\sim\mu_0}(b_I|s_0, k; \pi_{L}) \Pr(u_c \in s_{T_c}|b_I; \pi_{IL}) \\
& && (\pi_L \text{ and } \pi_{IL} \text{ are identical in the first } I \text{ steps}) \\
\geq&\sum\nolimits_{b_I\in\mathcal{B}_I^{b_0}} \Pr_{k\sim\mu_0}(b_I|s_0, k; \pi_{L}) \Pr(u_c \in s_{T_c}|b_I; \pi_{L}) \\
=&\Pr_{k\sim\mu_0}(u_c \in s_{T_c}|s_0, k; \pi_{L}) && \text{(law of total probability)} \\
\geq&p_c && (\pi_L \in \Pi_c)
\end{aligned}$$

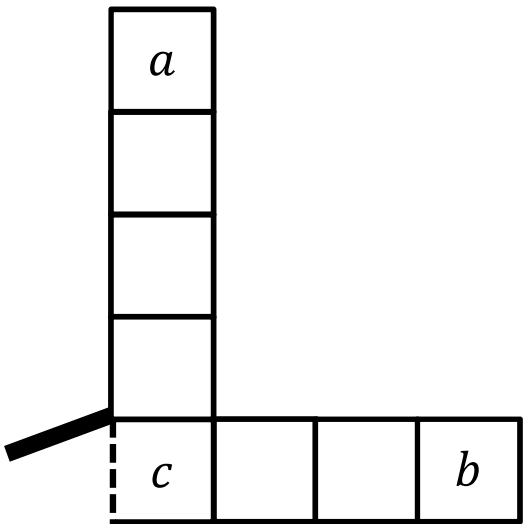
The first inequality holds because CCIL iteratively applies L-step lookahead with the commitment probability achieved by the policy of the previous iteration. This concludes the proof. □

4.3 Empirical Study

Overview

As summarized in Section 4.1, we are the first to define a prescriptive semantics for probabilistic commitments under model uncertainty, and develop algorithms that respect the semantics. Hence, in the empirical studies that follow, we predominantly focus on developing a deeper understanding of the strengths and limitations of different flavors of our algorithms. However, in an effort to illustrate empirically the difference between our approach and prior work, in our first study in the illustrative Windy L-Maze domain, we compare to the closest related work we could identify: a non-prescriptive semantics for probabilistic commitments, and a prescriptive semantics for non-probabilistic commitments. We show how our prescriptive probabilistic commitment semantics allows agents to outperform either of these others because with it agents can balance selfish and unselfish behavior.

We next use a small size Food-or-Fire domain to show how our CCL performs in an



d	0	1	2	3	4
R_1^p	0	0	0	0	0.1
R_2^p	-4	-3	-2	-1	0.1
R_3^p	4	3	2	1	0.1
R^r	3	0	0	0.1	0

Figure 4.9: Windy L-Maze. The provider starts in the cell labeled a and can only move in the vertical corridor, and the recipient starts in the cell labeled b and can only move in the horizontal corridor. It is admissible that both agents occupy the cell labeled c at the same time step. The table on the right specifies the reward functions, where d is the distance, measured by number of cells, between cell c and the provider/the recipient. For the provider, there are three possible reward functions $\{R_k^p\}_{k=1}^3$. The recipient's reward, R^r (bottom row), is known for certain.

environment with both transition and reward uncertainty, and under various choices of belief-update lookahead boundary. In the subsequent two domains of RockSample and Change Detection, the number of possible posterior distributions can grow so quickly with the time horizon that CCFL becomes computationally infeasible. In RockSample, we show how the iterative version of CCL, CCIL, is able to improve performance over CCL with modest additional computational cost. In Change Detection, we perform a detailed case study on the effects of the belief-update lookahead boundary and how it should be chosen with domain-specific knowledge, along with results reconfirming the improvement of CCIL over CCL.

Windy L-Maze

The purpose of the experiments in this domain is to illustrate how our prescriptive probabilistic commitment semantics can improve multi-agent planning compared to alternative semantics. The domain consists of an L-maze occupied by a commitment provider and a recipient, as shown in Figure 4.9.

The provider starts in the cell labeled a and can only move in the vertical corridor, and the recipient starts in the cell labeled b and can only move in the horizontal corridor. It is admissible that both agents occupy the cell labeled c at the same time

step. Let d^p, d^r be the distance, measured by number of cells, between cell c and the provider, the recipient, respectively. For the provider, there are three possible reward functions as functions of d^p, $\{R_k^p\}_{k=1}^3$, with a uniform prior:

$$\text{for } d^p = 4, R_1^p(d^p) = R_2^p(d^p) = R_3^p(d^p) = 0.1$$
$$\text{for } d^p < 4, R_1^p(d^p) = 0, R_2^p(d^p) = -R_3^p(d^p) = d^p - 4$$

The recipient's reward, R^r, is known as a function of d^r: $R^r(d^r) = 0.1$ if $d^r = 3$; $R^r(d^r) = 3$ if $d^r = 0$; $R^r(d^r) = 0$ for other values of d^r. The provider can move up, down, or stay in the current cell, and its moves succeed with probability one. The recipient can move left, right, or stay in the current cell. Initially, a door located in cell c is open with a strong wind blowing in such that the recipient's moves to the left only succeed with probability 0.1, and its other moves succeed with probability one. By occupying cell c, the provider can permanently close the door, in which case the wind stops and all the recipient's moves succeed with probability one. The two agents aim to maximize the joint expected reward up to the time horizon $H = 10$.

Because the recipient will get a significantly larger reward in cell c than in cell b, it is beneficial for the recipient if the provider could move to cell c to close the door. However, under reward functions R_1^p and R_2^p, traveling down the corridor to cell c will yield less reward for the provider than staying in the starting cell a. Therefore, effective coordination between the two agents is crucial to achieving high expected joint reward, where (as we shall see) the uncertain rewards of the provider make an "all-or-nothing" commitment suboptimal compared to a probabilistic commitment.

We compare the following three commitment semantics:

Non-Prescriptive Probabilistic Semantics: In this case, a probabilistic commitment only represents a prediction of the provider's behavior [XL00, MSB+08], rather than a prescription for how it will act. The provider computes and follows its history-dependent policy maximizing just its own local reward. It informs the recipient of the probability, $\underline{p_c}$, that the door will be closed at time step $T \geq 4$ under the provider's policy, and the recipient then computes and follows its own locally-optimal policy with respect to $\underline{p_c}$ by standard methods of solving MDPs. We refer to this semantics as *selfish* and *no-commitment* because the provider makes no effort to consider the preferences of the recipient when computing and executing its policy.

Prescriptive Non-Probabilistic Semantics: This semantics is the logic-based

Table 4.1: Evaluation of Non-Prescriptive Semantics, Prescriptive Non-Probabilistic Semantics, and Prescriptive Probabilistic Commitment on the Windy L-maze domain. The columns represent the cumulative rewards for the provider individually, the recipient individually, and both agents jointly.

Semantics	Provider	Recipient	Provider + Recipient
Non-Prescriptive Probabilistic ($\underline{p_c} = 1/3$)	9.17	4.33	13.50
Prescriptive Non-Probabilistic ($\overline{p_c} = 1.0$)	4.90	10.61	15.51
Prescriptive Probabilistic ($p_c = 0.6$)	9.06	6.84	15.90
Prescriptive Probabilistic ($p_c = 0.7$)	8.62	7.79	16.41
Prescriptive Probabilistic ($p_c = 0.8$)	7.38	8.73	15.61

semantics alluded to in work on detecting commitment abandonment [POM17], where a commitment provider will drop all else and single-mindedly pursue a commitment. In this case, the provider computes and follows its history-dependent policy that achieves the highest probability, $\overline{p_c}$, of closing the door at the earliest possible time step which is $T = 4$. The recipient uses $\overline{p_c}$ to compute and follow its optimal policy assuming maximum help from the provider. We refer to this semantics as *unselfish* and *full-commitment* because the provider prioritizes satisfying the preferences of the recipient over its own rewards.

Prescriptive Probabilistic Commitment: This is the semantics we advocate in this thesis. The provider makes a probabilistic commitment: it commits to closing the door at time step $T = 4$ with at least probability p_c. It uses the CCFL algorithm to compute and follow its locally-optimal policy that respects the commitment semantics. The recipient trusts this commitment, and computes and follows its optimal policy assuming the door will be closed at time step $T \geq 4$ with probability p_c.

The performance of each of the three different semantics (with a few choices of p_c for our prescriptive probabilistic semantics) is shown in Table 4.1. Notice that even when the provider is acting entirely selfishly (the non-prescriptive probabilistic case), it predicts that it will nevertheless close the door with probability $\underline{p_c} = 1/3$. This is because its optimal policy is to move down the corridor one step, observe the reward signal to know exactly what the true reward function is, and then either go immediately back to a, or, with probability 1/3, it will learn that the reward function is R_3^p and continue on to c. Following the prescriptive non-probabilistic semantics, the unselfish provider will follow a policy guaranteed to close the door ($\overline{p_c} = 1.0$), because its moves succeed with certainty. With the prescriptive probabilistic commitment

semantics, the providers can choose a probability of closing the door $p_c \in [0, 1]$ that balances selfishness and unselfishness in the provider to attain a higher joint reward. As p_c increases, the provider's value monotonically decreases and the recipient's value monotonically increases. As shown in Table 4.1, both $p_c = 0.6$ and $p_c = 0.8$ achieve higher joint reward than $\underline{p_c}$ and $\overline{p_c}$, and $p_c = 0.7$ is even better than $p_c = 0.6$ and $p_c = 0.8$.

These results confirm that our semantics for probabilistic commitments, coupled with algorithms for agent decision-making that respect the semantics, can lead to better joint performance than treating commitments either as inflexible logical constraints on the provider's plan (such that it must provably satisfy the commitment) or as non-binding predictions about the likelihood the provider's plan will happen to satisfy the commitment. Our semantics enable agents to strike a compromise between these extremes.

Food-or-Fire

The purpose of the experiment in this domain is twofold: 1) it is used to simply illustrate that CCL works well in an environment with both transition and reward uncertainty to construct policies satisfying the constraint of a given probabilistic commitment, and 2) it is small enough that we can show the effect of the belief-update lookahead boundary by experimenting with all possible choices for the boundary from zero to the time horizon.

The environment is a simple two by three grid maze with $K = 3$ possible scenarios, as shown in Figure 4.10, where solid black lines indicate impassable walls. The prior over the three scenarios is a uniform distribution. In the "empty" scenario, the provider can move freely in four directions within the maze, and no reward signal occurs. In the "food" scenario, there are two sections of impassable wall, and food associated with a reward of +1 exists in the mid-left cell between the walls. The "fire" scenario is the same as the second except that food is replaced with fire associated with a reward of -1. The agent, starting in the bottom left cell, commits to reach the top left cell (Exit) at the time horizon, i.e. $T = H$, with at least probability p_c. The agent can fully observe its current location but can only detect a wall by trying (and failing) to move between two adjacent cells.

Because the transition dynamics vary across the three scenarios, we only implemented deterministic CCL. Figure 4.11 plots the expected cumulative reward against all possible belief-update boundaries using deterministic CCL under various choices of T_c and p_c. According to Theorem IV.5, the monotonic performance in belief-update

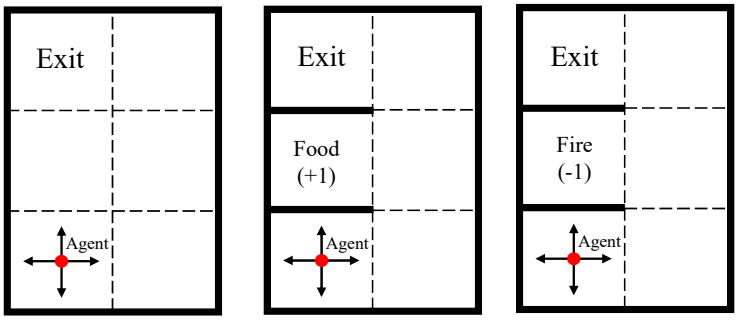


Figure 4.10: Food-or-Fire. *Left:* the "empty" scenario. *Middle:* the "food" scenario. *Right:* the "fire" scenario.

lookahead boundary L cannot be guaranteed, but it turns out the expected cumulative reward using deterministic CCL is monotonically non-decreasing with L for all choices of T_c and p_c we tried. Thus, anecdotally, it is not hard to find cases in which a larger L yields higher value, even though by Theorem IV.5 it is not guaranteed. Moreover, when L increases from two to three, we observe that the expected cumulative reward increases significantly for most choices of T_c and p_c. This is because a belief-update lookahead boundary L of three is just sufficient to identify which scenario the provider is actually facing by moving to the middle-left cell using three actions and reasoning about the observed reward signal of food, fire, or neither. Not surprisingly, with lower commitment probabilities, the provider is able to achieve higher expected reward. An interesting observation is that, compared with $p_c = 0.8$, we see the the expected cumulative reward is more like a step function at $L = 3$ for $p_c = 0.5$ and $p_c = 1.0$. When $p_c = 1.0$, the provider has to reach the Exit at time T_c in all three scenarios, so it suffices to determine the optimal behavior as soon as the provider figures out at time $L = 3$ which scenario it is facing. When $p_c = 0.5$, the provider would certainly reach the Exit in the "empty" scenario and the "fire" scenario. With the uniform prior, these two scenarios already contribute to $2/3 \geq p_c = 0.5$ probability of fulfilling the commitment, and therefore in the "food" scenario the provider would stay in the cell with food for the $+1$ reward and never exit. To achieve this behavior when $p_c = 0.5$, it suffices to use $L = 3$. For $p_c = 0.8$, it is more complicated in the sense that the provider also needs to reach the Exit with some positive probability in the second (food) scenario, and our results show that, with deterministic CCL, using L larger than 3 is able to improve the value.

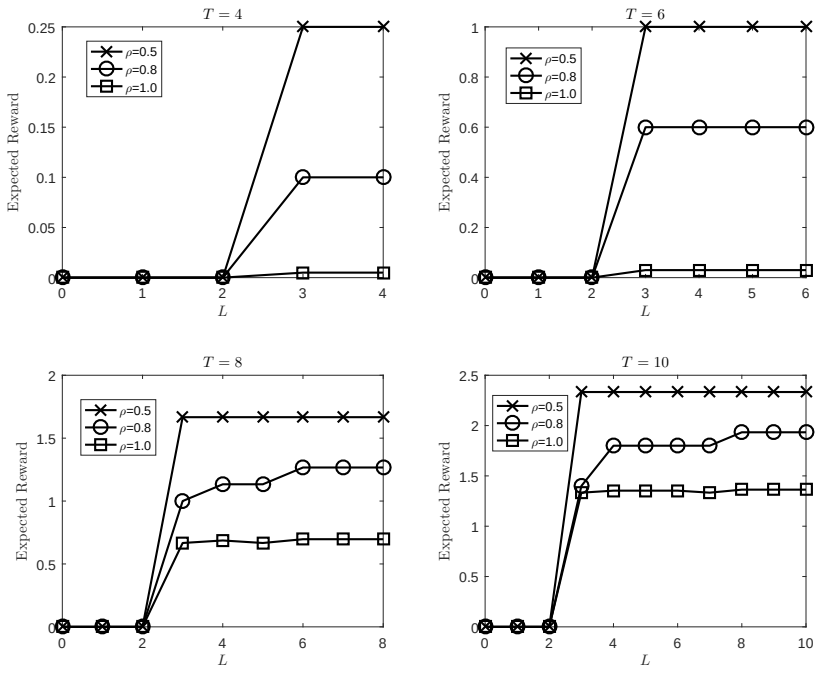


Figure 4.11: Expected cumulative reward in Food-or-Fire domain as a function of the commitment and the belief-update lookahead boundary.

RockSample

The size of the Food-or-Fire domain is small enough for us to afford computing belief-update boundaries up to the time horizon. In this RockSample domain and the following Change Detection domain, the number of posterior distributions grows so quickly as the time horizon grows that CCFL becomes computationally infeasible. Our results show that using the iterative version of CCL, CCIL, can improve the performance significantly with moderate additional computational cost.

RockSample [SS04] is a classic POMDP problem that models a rover exploring an unknown environment. In an instance of RockSample(n, s), the rover can move in an $n \times n$ grid containing s rocks. When n and s become large, a large belief-update lookahead boundary becomes computationally infeasible. The locations of the rocks are known. Only some of the rocks have scientific value and are of type *Good*; the others are of type *Bad*. The type of each rock is uniformly random. The task is to determine which rocks are valuable, approach and take samples of valuable rocks, and leave the map by moving off the right-hand edge of the map. Each time step, the rover can select from $s + 5$ actions: {$North$, $East$, $South$, $West$, $Sample$, $Check_1$,...,$Check_s$}. Each $Check_i$ action directs the rover's sensor to rock i, returning a noisy observation from {*Good*, *Bad*}. The noise in the observations received by

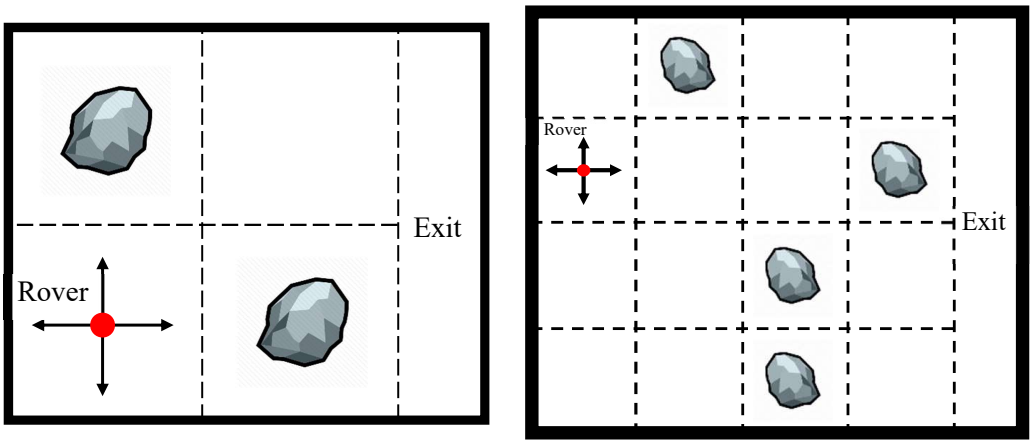


Figure 4.12: RockSample instances. *Left:* RockSample(2,2). *Right:* RockSample(4,4).

executing each $Check_i$ action is determined by the Manhattan distance between the rover and the rock being checked: the probability of receiving a correct observation is 0.9, 0.7, and 0.5 when the the Manhattan distance is 0, 1, and at least 2, respectively. In an instance of RockSample(n, s), s rocks could have 2^s possible combinations of type assignments. We treat them as $K = 2^s$ possible MDPs that only differ in reward, and solve the program in Figure 4.7 to construct CCL and CCIL policies. During execution, the observations from $Check_i$ actions are model-informative, suggesting which MDP is more likely.

In the original RockSample problem, the rover chooses actions to execute until it moves off the map and receives a positive reward. We adapted it to incorporate the probabilistic commitment: the rover does not receive any reward by moving off the map, but it has to move off the map by the time horizon, i.e. $T = H$, with at least the commitment probability p_c. We scale the reward to the range of $[-1, 1]$: the rover receives a reward of 1.0 for sampling a rock of type *Good*, a reward of -1.0 for sampling a rock of type *Bad*, and no reward occurs for re-sampling the same rock.

We evaluated CCL and CCIL on instances of RockSample(2, 2) and RockSample(4, 4) (Figure 4.12). Table 4.2 contains the results of expected reward and run time in RockSample(2, 2) for commitment time $T = 10$ and commitment probability $p_c = 1.0$ with various choices of L and I. The run time for CCIL is the sum of the CPU times for each iteration. Note that because 1) the rover can get pretty accurate observations since it is always close to the rocks, 2) the types of rocks are uniformly random, and 3) time horizon 10 is large enough, the optimal behavior can collect in expectation one good rock, yielding an expected cumulative reward close

to 1.0. For CCL, the results in Table 4.2 indicate that a larger belief-update lookahead boundary indeed improves the expected reward, but the computational time also increases dramatically. We can see that CCIL can achieve comparable expected reward with much less computational time than CCL. Although CCIL($L = 3, I = 1$), CCIL($L = 4, I = 4$), and CCL($L = 8$) all achieve near-optimal expected reward, CCIL($L = 3, I = 1$) and CCIL($L = 4, I = 4$) use much less computational time than CCL($L = 8$).

Table 4.2: Results on RockSample(2,2), $|\mathcal{S}| = 177, |\mathcal{A}| = 7, |\mathcal{O}| = 4$ with $T = 10$, $p_c = 1.0$. 1000s run time limit.

L	I	Expected Reward	Time(s)
0	n.a.	0.00	0.30
1	n.a.	0.20	0.54
2	n.a.	0.40	1.07
3	n.a.	0.60	3.05
4	n.a.	0.64	7.53
6	n.a.	0.82	45
8	n.a.	0.90	710
10	n.a.	n.a.	>1000
1	1	0.53±0.02	4.83±0.28
3	1	1.01±0.02	33.89±1.67
3	3	0.81±0.02	7.73±0.13
4	1	0.97±0.02	133.11±10.67
4	4	0.92±0.02	17.55±0.30

Table 4.3 contains the results in RockSample(4, 4) for commitment time $T = 15$ and probability $p_c = 1.0$. With $T = 15$, the time is just enough for the rover to correctly detect 3 rocks, sample the good rocks, and move off the map. Since a rock is good with probability .5, the expected cumulative reward of the optimal behavior is close to 1.5. For RockSample(4, 4), we can see that CCL can only scale to relatively small belief-update boundaries. The computational time grows dramatically, and we run out of memory when $L = 5$. CCL achieves an expected cumulative reward of 0.9 for $L = 4$, which means that a larger L is needed to find the near-optimal behavior. CCIL performs much better than CCL because it iteratively re-plans during the execution. The performance of CCIL($L = 1, I = 1$) is between that of CCL($L = 3$) and CCL($L = 4$). CCIL($L = 2, I = 2$), CCIL($L = 2, I = 1$), and CCIL($L = 3, I = 3$) all achieve behavior with expected cumulative reward close to 1.3, which cannot be achieved by CCL using a moderate amount of computational time. These three choices of (L, I) achieve comparable expected reward (no statistically significant

Table 4.3: Results on RockSample(4,4), $|\mathcal{S}| = 4097, |\mathcal{A}| = 9, |\mathcal{O}| = 8$, with $T = 15$, $p_c = 1.0$. 1000s run time limit.

L	I	Expected Reward	Time(s)
0	n.a.	0.00	4.33
1	n.a.	0.30	5.11
2	n.a.	0.30	8.71
3	n.a.	0.60	23.36
4	n.a.	0.90	113
5	n.a.	Out of memory	n.a.
1	1	0.74±0.02	83.06±0.55
2	1	1.32±0.02	482.30±31.53
2	2	1.31±0.02	132.17±3.73
3	1	n.a.	>1000
3	3	1.34±0.02	634.27±67.37

difference), with CCIL($L = 2, I = 2$) being the fastest because its iterative lookahead is less frequent than CCIL($L = 2, I = 1$) and shallower than CCIL($L = 3, I = 3$).

Change Detection

In Change Detection, we perform a detailed case study on the effects of the belief-update lookahead boundary, where time horizon H is short enough so that we can experiment with every $L \leq H$ for CCL. We also experiment with a larger H for which CCFL is computationally infeasible, to develop further intuitions about balancing lookahead with iteration to achieve good performance with reasonable computation.

Change Detection is a classic constrained POMDP problem [Shi63]. The agent can partially observe the environment, and at some point the environment will transit into a state where the alarm should be sounded by the agent. The agent aims to minimize the delay in alerting (sounding the alarm) after the transition, and the probability of a false alarm should be lower than a given threshold which is referred to as the false alarm (F.A.) tolerance. Formally, the state space and action space are $\mathcal{S}$=$\{PreChange, PostChange, PostAlarm, FalseAlarm\}$, $\mathcal{A}$=$\{NoAlarm, Alarm\}$, respectively. The environment starts in $PreChange$, and transits to $PostChange$ at a random time step if the provider has not performed action $Alarm$. Specifically, the problem has a geometric change time parameter η, such that at every time step, if the state is still $PreChange$, it will transit to $PostChange$ with probability η. Once the provider performs action $Alarm$, the state transits to $PostAlarm$ from $PostChange$ with a positive reward, or to $FalseAlarm$ from $PreChange$ with no reward. The commitment is to *not* reach $FalseAlarm$ with at least a given probability. To en-

courage early detection, the provider receives a reward of +1.0 if it executes action *Alarm* immediately after transiting to *PostChange*, with the reward discounted each subsequent time step. The states are not fully observable. Instead, the provider makes an observation o every time step from the observation space $\mathcal{O}$, suggesting if the environment has changed or not. The probability of making a specific observation is determined by probability mass functions $f_0, f_1 : \mathcal{O} \mapsto [0, 1]$ when the environment is in *PreChange*, and *PostChange*, respectively. In our experiments, the provider can make an observation every time step from a set of size $|\mathcal{O}| = 3$. The reward discount factor is set to $\gamma = 0.8$. The *PreChange* and *PostChange* observation distributions are

$$f_0(o_1) = 0.6, f_0(o_2) = 0.3, f_0(o_3) = 0.1,$$
$$f_1(o_1) = 0.2, f_1(o_2) = 0.4, f_1(o_3) = 0.4.$$

Parameter η provides the provider with the prior distribution of the change time. After making observations, the provider can use Bayes' rule to calculate the posterior distributions.

We consider the finite horizon decision problem, with the commitment time $T = H$ being equal to the time horizon, and define the state of the Change Detection problem as $s = \langle t, Alarmed \rangle$ where *Alarmed* is a Boolean that takes the value of true when the provider executed action *Alarmed* in any time step before t, or false otherwise. The current time step t and Boolean *Alarmed* are both fully observable to the provider. We define belief as $b = \langle s, \mu \rangle$, where state s is augmented by probability mass function μ that gives the probability of all possible change times up to the horizon.

Figure 4.13 contains the results when experimenting with CCL on a Change Detection instance with horizon $H = T = 10$, where CCFL is computationally feasible. We have experimented with two choices of the geometric change time parameter, $\eta = 0.1, 0.2$, and four choices of the false alarm (F.A.) tolerance. When F.A. tolerance is 0.0, the provider is forbidden to execute *Alarm* actions if there is any possiblity of false alarm, and therefore the expected cumulative reward is 0 for any choice of the belief-update lookahead boundary L. Otherwise, the expected cumulative reward is monotonically increasing with L. Moreover, choosing a large L is most helpful when the geometric change time parameter η is small (Figure 4.13(left)). For $\eta = 0.1$ (Figure 4.13(left)), the expected reward rises anywhere from about 3-fold (for tolerance=0.2) to 7-fold (for tolerance=0.05), while for $\eta = 0.2$ (Figure 4.13(right)) it is anywhere from about 1.5-fold (for tolerance=0.2) to 3.5-fold (for tolerance=0.05). So

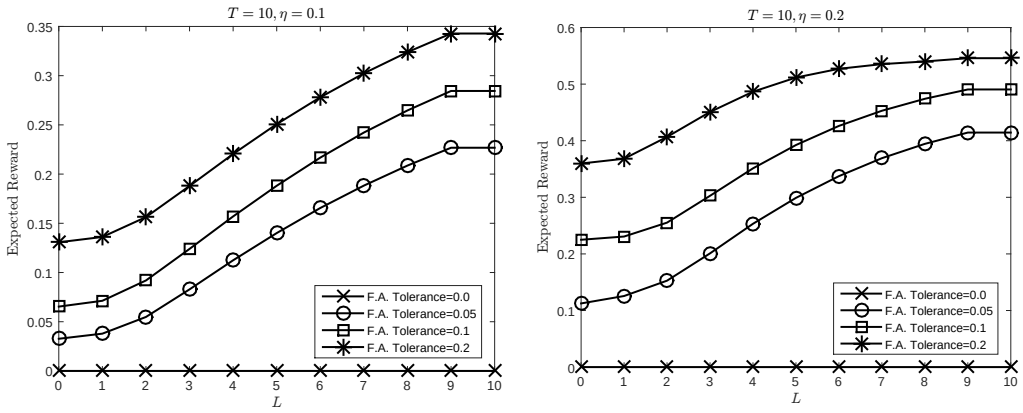


Figure 4.13: Results of CCL on Change Detection with $T = 10, \gamma = 0.8$.

for the same tolerance, lookahead makes twice the impact when $\eta = 0.1$ than $\eta = 0.2$. Small η suggests that the change is more likely to happen later, and therefore a large L is more likely to envision it. For both choices of η, as lookahead L increases, the relative increase in expected reward is smaller when F.A. tolerance is larger. This is because larger tolerance inherently gets more reward regardless of lookahead, and hence there is less reward for lookahead to recoup. These results suggest that, more generally, the value of L should be chosen based at least upon: (1) how far into the future the most meaningful changes to the belief state will occur (as captured by η in this case), (2) how sensitive the provider's reward is to making a more informed decision (as captured by F.A. tolerance in this case), and (3) how dramatically computation costs rise with farther lookahead (where in this case the branching factor of 2 (change or no change) is fairly low).

We have also experimented with a larger horizon, $H = T = 50$, where CCFL is not computationally affordable. The geometric change time parameter is $\eta = 0.04$. As we just saw, a low value like this makes the change more likely to happen later and thus emphasizes farther lookahead. The F.A. tolerance is set to 0.2. Table 4.4 contains the results of expected reward and run time for CCL and CCIL with various choices of L, and of I when applicable. The run time of CCL grows dramatically with L. The expected reward, though, grows relatively slowly, because these lookaheads are still very short for such a small η that requires large lookahead. This can be inferred from Figure 4.13 (left), where $\eta = 1.0$ is larger and we still see a steep increase in reward at $L = H/2$. Nevertheless, there is still a 3-fold increase in reward when we increase L for CCL until the computation budget is reached. For CCIL, we experiment with $L = 2, 4, 6$ and $I = 1, L/2, L$. Unsurprisingly, with more frequent iterative lookahead (smaller I), both the expected reward and the run time increase. CCIL($L = 4, I = 1$)

Table 4.4: Results on Change Detection with F.A. tolerance of 0.2, $T = 50, \eta = 0.04, \gamma = 0.8$. 1000s run time limit.

L	I	Expected Reward	Time(s)
1	n.a.	0.05	0.02
2	n.a.	0.06	0.05
3	n.a.	0.07	0.16
4	n.a.	0.09	0.46
6	n.a.	0.11	4.23
9	n.a.	0.15	125
10	n.a.	0.16	761
11	n.a.	n.a.	> 1000
2	1	0.06±0.02	1.62±0.08
2	2	0.04±0.02	0.99±0.04
4	1	0.28±0.04	16.33±0.98
4	2	0.17±0.04	9.85±0.78
4	4	0.09±0.03	4.04±0.30
6	1	0.32±0.03	117.11±7.84
6	3	0.31±0.04	33.01±2.56
6	6	0.13±0.04	28.41±1.98

achieves reward that is higher than any CCL within the computation budget. Both CCIL($L = 6, I = 1$) and CCIL($L = 6, I = 3$) double the reward of CCL($L = 10$), the largest L within the computation budget, yet use much less computation. These results verify again the effectiveness of the iterative lookahead strategy in CCIL. Recall that, in RockSample, setting $I = L$ achieves significantly larger reward than CCL with the same L. However, in Change Detection, $I = L$ achieves no higher reward than CCL for the values of L we consider. We conjecture that this is because the belief changes frequently in Change Detection (every time step) and perhaps in a way that is critical for the provider's future decisions, making it necessary to perform frequent iterative lookahead, while it might take several steps in RockSample to experience a change (after taking the $Check_i$ action). From the results of $L = 4, 6$ and $I = 1, L/2$, we observe that with larger L, the provider can use larger I without sacrificing too much reward. Overall, CCIL($L = 4, I = 1$) and CCIL($L = 6, I = 3$) achieve the best compromise for a wide range of tradeoffs between solution quality and computational cost.

4.4 Summary

This chapter defined a prescriptive semantics for a probabilistic commitment provider that is operating under model uncertainty. Our semantics is based on what a commitment provider can control—its own actions. Specifically, we considered a decision-theoretic setting where the provider is making sequential decisions in one out of several MDPs drawn from a known prior. Fulfilling a commitment corresponds to pursuing a course of action, beginning at the time the commitment was made, that has sufficient likelihood of realizing the intended state at a certain time prescribed by the commitment. In this semantics, the provider fulfills its commitment by following a commitment-constrained policy even if, due to bad luck, the desired outcome was not realized. Based on this semantics, we developed Commitment Constrained Lookahead (CCL), a novel algorithm parameterized by the belief-update lookahead boundary, that constructs commitment constrained policies offline for the provider. We empirically compared our new semantics, operationalized in CCL, with prior logical and predictive semantics concepts, to illustrate where and why our semantics is superior. We also analytically and empirically investigated the impact of the belief-update lookahead boundary that makes an explicit tradeoff between the computation cost and performance of the computed policy. Since the lookahead boundary, and therefore the performance, of CCL is directly limited by memory size, we have further extended CCL to Commitment Constrained Iterative Lookahead (CCIL) that iteratively adjusts the policy online according to the evolving posterior distribution about the true environment, while still satisfying the commitment constraint. Our empirical results show that CCIL can achieve the same performance as CCL with much less computational overhead. In a nutshell, the prescriptive semantics and the algorithms together offer tractable solutions for the provider to respond to its evolving model uncertainty without detriment to its trustworthy adherence to the commitment.

CHAPTER V

Robust Interpretation of Probabilistic Commitments

A probabilistic commitment constrains the provider's policy choice regarding the shared state feature at a single timestep, and while this gives the provider flexibility to adjust its policy on the fly, the recipient has to deal with the uncertainty about the shared feature at other timesteps. The question, then, is how should the recipient interpret the commitment, that is, how can the recipient approximate the true dynamics of the shared feature in a robust manner to yield a high quality policy? This is the problem we have formulated in Section 3.2. In this chapter, we focus on this question for both achievement and maintenance commitments, which are two types of commitment commonly modeled and studied in the literature. Our notion of robustness hinges on the suboptimality of the recipient's approximation of the influence, which is defined as the difference between the value of the optimal policy associated with the approximate influence and that associated with the true dynamics of the shared feature. This chapter presents theoretical analyses and empirical studies showing that, perhaps surprisingly, despite strong similarities in the provider's modeling of the two types of commitment, there is an inexpensive strategy for the recipient to create an approximate influence with low suboptimality for achievement commitments, while no such strategy exists for maintenance commitments.

5.1 Problem Statement Recapitulation

In this section, we revisit the problem of suboptimality of the recipient's approximate influence, as we have defined in Section 3.2. As the discussion will be focused on the recipient only in this chapter, we will drop superscripts r for the notations. Adopting the notations in Section 2.1 for MDPs, the recipient's MDP is denoted as

$M = (\mathcal{S}, \mathcal{A}, P, R, H)$ with initial state s_0. The optimal policy for M is denoted as π^*_M, and its value function $V_M^{\pi^*_M}$ is abbreviated as V^*_M. The value of the initial state for policy π is abbreviated as $v^\pi_M := V^\pi_M(s_0)$. As we have discussed in Section 2.3, we factor the recipient's state into features, $s = (l, u)$, where features l are locally controlled by the recipient and features u are shared and controlled by the provider. Accordingly, the recipient transition function is factored as $P = (P_l, P_u)$, where P_u is the dynamics of u that is determined purely by the provider and referred to as the provider's true influence.

For a given probabilistic commitment $c = (u_c, T_c, p_c)$, its specification and semantics constrain the provider's policy based on a single future timestep T_c: at time T_c, the value of u will be u_c with at least the promised probability p_c. By not committing to (bounds on) the probabilities at intervening (and subsequent) timesteps, the provider retains flexibility to revise its policy on the fly (for example, if its belief about the reward function changes, as we have discussed in Chapter IV).

The commitment specification is also the only information that the recipient has about P_u, and while information about only a single future timestep might give the provider flexibility, it imposes uncertainty on the recipient. That is, while the recipient knows something about P_u at the commitment's timestep T_c, it can only guess at the values of the influence at other timesteps.

Adopting the notations in Section 3.2 and dropping superscripts r, the recipient adopts a *strategy* $\widehat{P}_u(\cdot)$ that maps a given probabilistic commitment c to an *approximate influence* $\widehat{P}_u(c)$. When commitment c is fixed, there is no need to distinguish between a strategy $\widehat{P}_u(\cdot)$ and the approximate influence $\widehat{P}_u(c)$ it induces, and thus we will abbreviate them as $\widehat{P}_u$. The approximate influence $\widehat{P}_u$ is then used for planning in $\widehat{M} = (\mathcal{S}, \mathcal{A}, \widehat{P}, R, H)$ where $\widehat{P} = (P_l, \widehat{P}_u)$. For a fixed commitment, the suboptimality of $\widehat{P}_u$ is evaluated using the difference between the value of the optimal policy for $\widehat{M}$ and the value of the optimal policy for M when both policies are evaluated in M starting in s_0, i.e.

$$\text{Suboptimality}(\widehat{P}_u; P_u) = V^*_M(s_0) - V_M^{\pi^*_{\widehat{M}}}(s_0) = v^*_M - v_M^{\pi^*_{\widehat{M}}}.$$

Note that when the support of P_u is not fully contained in the support of $\widehat{P}_u$, the recipient's policy $\pi^*_{\widehat{M}}$ can associate zero occupancy (hence plan no action) for certain states when executed in M, which makes $V_M^{\pi^*_{\widehat{M}}}$ ill-defined. In this thesis, we resolve this by re-planning: during execution of $\pi^*_{\widehat{M}}$ in M, the recipient re-plans from any zero occupancy state that it happens to reach. Thus, the recipient's problem of

commitment interpretation is to identify a high-quality approximate influence that induces low suboptimality for the given commitment, while a high-quality strategy should robustly induce low suboptimality for a range of commitments.

5.2 Achievement and Maintenance

In this chapter, we focus on two types of commitment commonly studied in the literature, which are achievement commitments and maintenance commitments. In an achievement commitment, the provider commits to courses of action that probabilistically change the shared state features in a way desired by the recipient. For example, the recipient plans to take an action (e.g., move from one room to another) with a precondition (e.g., the door separating rooms is open) that the provider has promised to likely enable by some deadline. In a maintenance commitment, the provider instead commits to courses of action that, up until a promised time, are sufficiently unlikely to change features that are already the way the recipient wants them maintained. After that time, the provider can freely change the features. For example, a door the recipient wants open might initially be so, but the provider wants to close it to clean behind it during housekeeping tasks. The provider could postpone closing it (clean elsewhere first), but by changing other doors while cleaning elsewhere it might accidentally introduce a draft that could prematurely close the door the recipient wants left open.

To formally capture the differences between achievement and maintenance, we here describe the two types of commitment as subclasses of probabilistic commitments as defined in Section 2.3. Similar to prior work [HvR07, WD09], we assume that u contains a single state feature that takes binary value and can be toggled *at most once*. Let u^+, as opposed to u^-, be the value of u that is desirable for the recipient. Intuitively, $u^+(u^-)$ stands for an enabled (disabled) precondition needed by the recipient. In transactional settings, a feature (e.g., possession of goods) changing only once is common, as it is in multiagent planning domains where one agent enables a precondition needed by an action of another. Some cooperative agent work requires agents to return changed features to prior values (e.g., shutting the door after opening and passing through it), and in extreme cases where toggling reliably repeats (e.g., a traffic light) there may be no need for explicit commitments. In general, when the binary feature u can indeed toggle more than once, it can be modeled by a series of alternating togglings in opposite directions, and thus the discussion in this chapter can apply to such a general setting by dividing it into multiple stages, such that in

each stage the feature toggles at most once.

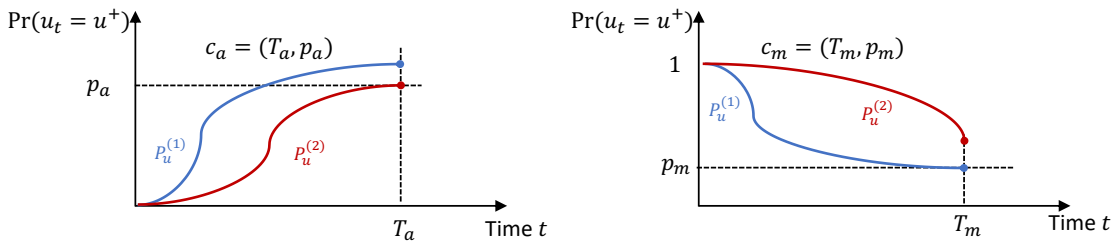
Achievement Commitments. Let the recipient's state at time t be factored as $s_t = (l_t, u_t)$. For achievement commitments, the initial value of the commitment feature is u^-, i.e. $u_0 = u^-$. An achievement commitment $c_a = (u^+, T_a, p_a)$ is a probabilistic commitment where the commitment feature value is u^+, the commitment time is T_a, and the commitment probability is p_a. Since the commitment feature value is fixed to u^+, we will abbreviate an achievement commitment $c_a = (u^+, T_a, p_a)$ as $c_a = (T_a, p_a)$ for the remainder of this chapter. The commitment semantics constrains the provider to follow a policy that changes the value of u to u^+ by time T_a with at least probability p_a, i.e.

$$\Pr(u_{T_a} = u^+ | u_0 = u^-) \geq p_a. \tag{5.1}$$

When planning with the achievement commitment, the provider can choose any policy that induces an influence that respects the commitment's semantics (5.1). Figure 5.1a illustrates two such influences as the provider's candidate influence for an achievement commitment. The recipient does not know the provider's true influence and adopts a strategy to create an approximate influence.

Maintenance Commitments. As a reminder, a maintenance commitment is appropriate in scenarios where the initial value of state feature u is desirable to the recipient, who wants it to maintain its initial value for some interval of time (e.g., [HvR07, DTH14]), but where the provider might want to take actions that could change it. Formally, for maintenance commitments, the initial value of the commitment feature is u^+, i.e. $u_0 = u^+$, and a maintenance commitment $c_m = (u^+, T_m, p_m)$ is a probabilistic commitment where the commitment feature value is u^+, the commitment time is T_m, and the commitment probability is p_m. As with an achievement commitment, we will abbreviate an achievement commitment $c_a = (u^+, T_a, p_a)$ as $c_a = (T_a, p_a)$ since u^+ is fixed. Given such a maintenance commitment, the provider is constrained to follow a policy that keeps u unchanged for the first T_m time steps with at least probability p_m. Since u can be toggled at most once, this is equivalent to probabilistically guaranteeing that u is still u^+ at the commitment time T_m, i.e.

$$\Pr(u_{T_m} = u_0 | u_0 = u^+) \geq p_m. \tag{5.2}$$



(a) Influences for Achievement (b) Influences for Maintenance

Figure 5.1: Candidate influences for an achievement commitment and a maintenance commitment.

As with an achievement commitment, the provider can choose any policy that induces an influence that respects the commitment's semantics (5.2), and the recipient adopts a strategy to create an approximate influence. Figure 5.1b illustrates two possibilities for the provider's candidate influence for a maintenance commitment.

Hence, from the provider's perspective, achievement and maintenance commitments are treated essentially identically, and from the recipient's perspective, the notions of approximate influence and suboptimality also identically apply to the two types of commitment. Even though decision-theoretic formulations of, and reasoning methods for, achievement and maintenance commitments are nearly identical, prior work has found it much harder to successfully coordinate for maintenance than achievement [CS08, GMDB08, Hia09]. In the past, it has been assumed that the difficulty lies on the provider's side—that it might be inherently harder for a provider to find good policies that maintain a feature than to change it. However, in this chapter we claim and justify that instead the challenge actually lies on the recipient's side: that a maintenance commitment is fundamentally harder for the recipient to model robustly than an achievement commitment is. We now substantiate this claim theoretically in Section 5.3 and empirically in Section 5.4.

5.3 Bounding the Suboptimality

In this section, we develop several strategies for the recipient to approximate the true influence, and present theoretical analyses that bound the worst-case suboptimality of these strategies. Our analyses make the following assumptions. Assumption V.1 states that the recipient's reward function only depends on its locally-controlled features, such that the cumulative reward of an episode is based only on the trajectory of l, $(l_0, l_1, ..., l_H)$. Note that, although the value of u_t does not directly affect the reward for time step t, it affects action choices that influence the value of l_{t+1} at the

next time step. Assumption V.2 intuitively says that u^+ establishes a precondition for an action that would be irrational to take when u^- holds. For example, if u^+ is a door being open, then the action of moving into the doorway could be part of an optimal plan, but taking that action if the door is closed (u^-) never is.

Assumption V.1. *For the recipient's reward function R, we assume*

$$R(s_t, a_t) = R(s_t) = R((l_t, u_t)) = R(l_t).$$

Assumption V.2. *Let $s^- = (l, u^-)$ and $s^+ = (l, u^+)$ be a pair of states that only differ in u. For any M with arbitrary influence P_u, there exists an optimal policy π^*_M such that*

$$P_l\left(\cdot|s^-, \pi^*_M(s^-)\right) = P_l\left(\cdot|s^+, \pi^*_M(s^-)\right).$$

To derive bounds on achievement and maintenance commitments, we will make use of the following lemma, where M^+ (M^-) is defined as the recipient's MDP identically to M except that u is always set to $u^+(u^-)$. Lemma V.1 directly follows from Assumption V.2, stating that the value of M^- is no more than that of M^+ and the value of any M is between the two.

Lemma V.1. *For any M with arbitrary influence P_u and initial value of u, we have $v^*_{M^-} \leq v^*_M \leq v^*_{M^+}$.*

Proof. Let's first consider the case in which P_u toggles u only at a single time step. We show $v^*_{M^-} \leq v^*_M$ by constructing a policy in M for which the value is $v^*_{M^-}$ by mimicking $\pi^*_{M^-}$. Whether u is initially u^- and later toggled to u^+ or *vice versa*, we can construct a policy π_M that chooses the same actions as $\pi^*_{M^-}$ assuming $u = u^-$ throughout the episode. Formally, for any $s^- = (l, u^-)$, letting $s^+ = (l, u^+)$,

$$\pi_M(s^+) = \pi_M(s^-) = \pi^*_{M^-}(s^-).$$

By Assumption V.2, π_M in M yields the same distribution over the trajectory of l as $\pi^*_{M^-}$ in M^-, and therefore $v^{\pi_M}_M = v^*_{M^-}$ since the cumulative reward only depends on the trajectory of l.

Similarly, we show $v^*_M \leq v^*_{M^+}$ by constructing a policy π_{M^+} in M^+ for which the value is v^*_M by mimicking π^*_M. Formally, for time steps when $u = u^-$ in M, let $\pi_{M^+}(s^+) = \pi^*_M(s^-)$. For time steps when $u = u^+$ in M, let $\pi_{M^+}(s^+) = \pi^*_M(s^+)$, where $s^- = (l, u^-)$, $s^+ = (l, u^+)$.

When P_u toggles u at $K > 1$ time steps, we can decompose the value function for P_u as the weighted average of K value functions corresponding to the K influences that toggle u at a single time step, and the weights of the average are the toggling probabilities of P_u at these K time steps. □

5.3.1 Minimal Enablement Duration

We begin by analyzing an intuitive and straightforward strategy to create approximate influences adopted in previous work for achievement commitments that models a single branch, *at the commitment time*, for when u^- probabilistically toggles to u^+ [WD10, ZDS+16]. Modelling the commitment with a single branch for toggling to u^+ at the latest possible time ignores possibilities of being enabled earlier than the deadline and of being enabled serendipitously after the deadline. Such an approximate influence models the achievement commitment *pessimistically*, in the sense that it minimizes the expected duration of u being enabled over all influences that respect the achievement commitment semantics (Equation (5.1)):

$$\min_{P_u \sim \ (5.1)} E_{P_u}\left[\sum_{t=0}^{H} 1_{\{u_t = u^+\}}\right]$$

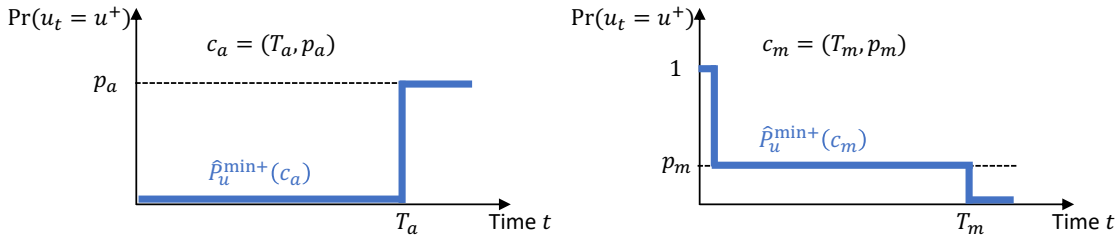
where $P_u \sim$ (5.1) means influence P_u satisfies Equation (5.1), and 1_E is the indicator function that takes value one if event E occurs and zero otherwise. We refer to this minimizer as the *minimal enablement duration* influence, as formalized in Definition V.1 and illustrated in Figure 5.2a.

Definition V.1. Given achievement commitment $c_a = (T_a, p_a)$, its minimal enablement duration influence $\widehat{P}_u^{\min+}(c_a)$ toggles u in the transition from time step $t = T_a - 1$ to $t = T_a$ with probability p_a, and does not toggle u at any other time step.

For maintenance commitments, the counterpart minimizes the expected enablement duration over all influences that respect the maintenance commitment semantics (Equation (5.2)):

$$\min_{P_u \sim \ (5.2)} E_{P_u}\left[\sum_{t=0}^{H} 1_{\{u_t = u^+\}}\right].$$

The minimizer models a probabilistic toggling to u^- at the earliest possible time, and a deterministic toggling to u^- (if it had not toggled earlier) after the commitment time, as formalized in Definition V.2 and illustrated in Figure 5.2b.



(a) $\widehat{P}_u^{\min+}(\cdot)$ for Achievement (b) $\widehat{P}_u^{\min+}(\cdot)$ for Maintenance

Figure 5.2: Minimal enablement duration for an achievement commitment and a maintenance commitment.

Definition V.2. Given maintenance commitment $c_m = (T_m, p_m)$, its minimal enablement duration influence $\widehat{P}_u^{\min+}(c_m)$ toggles u in the transition from time step $t = 0$ to $t = 1$ with probability $1 - p_m$, and (unless already toggled) from $t = T_m$ to $t = T_m + 1$ with probability one. It does not toggle u at any other time step.

As illustrated in Figure 5.2, the minimal enablement duration influence passes through the specific point of the commitment probability at the commitment time (i.e. (T_a, p_a), (T_m, p_m)), even though the provider's true influence does not have to (Figure 5.1). It is reasonable for the recipient to assume that the provider's true influence pass through the specific point, because otherwise the two agents could have agreed on a different commitment with the higher commitment probability for purpose of coordination. Therefore, in this thesis, we consider strategies, such as the minimal enablement duration, that pass the specific point, and focus on the core challenge that arises from the recipient's uncertainty about the true influence at time steps other than the commitment time.

Besides Assumptions V.1 and V.2, we also make Assumption V.3 for our analyses in this section, as a simplifying assumption stating that the true influence agrees with the minimal enablement duration influence after the commitment time, so that any suboptimality is caused by the imperfect modeling up until the commitment time.

Assumption V.3. *$P_u(u_{h+1}|u_h)$ agrees with the minimal enablement duration influence for $h \geq T$, where T is the commitment time.*

Bounding Suboptimality for Achievement. Here, we derive Theorem V.1 that bounds the suboptimality for achievement commitments as the difference between $v^*_{M^-}$ and $v^*_{M^+}$. We use Assumptions V.2 and V.3, and Lemma V.2 which states that, for achievement commitments, the possible ways the true influence differs from the minimal enablement duration influence can only improve the expected value.

Lemma V.2. *Given achievement commitment $c_a = (T_a, p_a)$, let $\widehat{P}_u = \widehat{P}_u^{\min+}(c_a)$, then we have $v_M^{\pi^*_{\widehat{M}}} \geq v_{\widehat{M}}^{\pi^*_{\widehat{M}}}$ where influence P_u in M respects the commitment semantics of c_a.*

Proof. For achievement commitments, the initial value of u is u^-. Let $P_u(t)$ be the probability that u is not enabled to u^+ until time step t in influence P_u, and $\overline{v}_t^{\pi}$ be the initial state's value under π when u is enabled from u^- to u^+ at t with probability one. By Assumption V.3, $v_M^{\pi^*_{\widehat{M}}}$ and $v_{\widehat{M}}^{\pi^*_{\widehat{M}}}$ can be decomposed as

$$\begin{aligned} v_M^{\pi^*_{\widehat{M}}} &= \textstyle\sum_{t=1}^{T_a} P_u(t)\overline{v}_t^{\pi^*_{\widehat{M}}} + (1-p_a)v_{M^-}^{\pi^*_{\widehat{M}}}, \\ v_{\widehat{M}}^{\pi^*_{\widehat{M}}} &= p_a\overline{v}_{T_a}^{\pi^*_{\widehat{M}}} + (1-p_a)v_{M^-}^{\pi^*_{\widehat{M}}}. \end{aligned}$$

When u is enabled at t in M, $\pi^*_{\widehat{M}}$ can be executed as if u is not enabled, by Assumption V.2, yielding identical trajectory distribution of l (therefore value) as in $\widehat{M}$. Therefore, the recipient's re-planning at t when $u = u^+$ will derive a better policy if possible. Therefore, the value of executing $\pi^*_{\widehat{M}}$ in M is no less than that in $\widehat{M}$, i.e. $\overline{v}_t^{\pi^*_{\widehat{M}}} \geq \overline{v}_{T_a}^{\pi^*_{\widehat{M}}}$. Therefore,

$$\begin{aligned} v_M^{\pi^*_{\widehat{M}}} =& \textstyle\sum_{t=1}^{T_a} P_u(t)\overline{v}_t^{\pi^*_{\widehat{M}}} + (1-p_a)v_{M^-}^{\pi^*_{\widehat{M}}} \\ \geq& \textstyle\sum_{t=1}^{T_a} P_u(t)\overline{v}_{T_a}^{\pi^*_{\widehat{M}}} + (1-p_a)v_{M^-}^{\pi^*_{\widehat{M}}} \\ \geq& p_a\overline{v}_{T_a}^{\pi^*_{\widehat{M}}} + (1-p_a)v_{M^-}^{\pi^*_{\widehat{M}}} \quad \text{(commitment semantics)} \\ =& v_{\widehat{M}}^{\pi^*_{\widehat{M}}}. \end{aligned}$$

□

Theorem V.1. *Given achievement commitment c_a, let $\widehat{P}_u = \widehat{P}_u^{\min+}(c_a)$. The suboptimality can be bounded as*

$$v_M^* - v_M^{\pi^*_{\widehat{M}}} \leq v_{M^+}^* - v_{M^-}^* \tag{5.3}$$

where influence P_u in M respects the commitment semantics of c_a. Further, there exists an achievement commitment for which the equality is attained.

Proof. The derivation of the bound in Equation (5.3) is straightforward from Lemma V.2:

$$v_M^* - v_M^{\pi^*_{\widehat{M}}} \leq v_{M^+}^* - v_{\widehat{M}}^{\pi^*_{\widehat{M}}} \leq v_{M^+}^* - v_{M^-}^*.$$

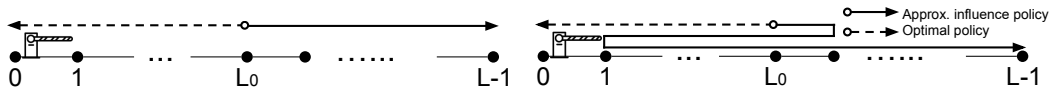


Figure 5.3: 1D Walk. *Left:* Example in the proof of Theorem V.1. *Right:* Example in the proof of Theorem V.2.

Next, we use a simple illustrative example to give an achievement commitment for which the equality is attained.

Example: An Achievement Commitment in 1D Walk. Consider the example of a 1D walk of L locations on $[0, L-1]$, as shown in Figure 5.3(left), where the recipient starts at L_0 and can move right, left, or stay still. There is a gate between 0 and 1 for which u^+ denotes the state of open and u^- closed. The provider toggles the gate stochastically according to P_u. For each time step the recipient is at neither end, it gets a reward of -1. Hence, the optimal policy is to reach either end as soon as possible in expectation. Note that the reward function makes Assumptions V.1 and V.2 hold.

Here, we derive an achievement commitment for which the bound in Theorem V.1 is attained. Consider $L = 10, L_0 = 3, H = 10$, achievement commitment ($T_a = L-1-L_0 = 6, p_a = 1$), and the true influence P_u in M that toggles the gate to open at $t = L_0 - 1 = 2$ with probability $p_a = 1$. The optimal policy in M is to move left to 0. Therefore, $v^*_M = v^*_{M^+} = -L_0 = -3$. Given the minimal enablement duration influence, moving right to L (arriving at time $L-1-L_0 = 6$) is faster than waiting for the gate to toggle at $T_a = 6$ and then reaching location 0 at time $T_a + 1 = 7$. Had the recipient known the gate would toggle at time $t = L_0 - 1 = 2$, it would have moved left, but by the time the gate toggles the recipient is at location $L_0 + L_0 - 1 = 5$, and continuing on to L is the faster choice. Therefore, $v^{\pi^*_{\widehat{M}}}_M = v^*_{M^-} = -(L-1-L_0) = -6$, and the bound in Theorem V.1 is attained. □

Bounding Suboptimality for Maintenance. We next ask if the bound in Equation (5.3) on suboptimality in achievement commitments also holds for maintenance commitments. Unfortunately, as stated in Theorem V.2, the optimal policy of the minimal enablement duration influence for maintenance commitments can be arbitrarily bad when evaluated in the true influence, incurring a suboptimality exceeding the bound in Equation (5.3). We give an example for an existence proof.

Theorem V.2. *Consider $\widehat{P}_u = \widehat{P}_u^{\min+}(c_m)$ to be the approximate influence when modelling the maintenance commitment in $\widehat{M}$. There exists an MDP M and a maintenance commitment c_m, such that the true influence P_u in M respects the commitment*

semantics of c_m, $v^*_M = v^*_{M^+}$, $v^{\pi^*_{\widehat{M}}}_M < v^*_{M^-}$, *and therefore the suboptimality*

$$v^*_M - v^{\pi^*_{\widehat{M}}}_M > v^*_{M^+} - v^*_{M^-} \tag{5.4}$$

exceeds the bound in Equation (5.3).

Proof. As an existence proof, we give an example of a maintenance commitment in 1D Walk for which $v^*_M = v^*_{M^+}$ and $v^{\pi^*_{\widehat{M}}}_M < v^*_{M^-}$. Consider 1D Walk with the same $L = 10, L_0 = 3, H = 10$ as in the example for Theorem V.1. Consider maintenance commitment ($T_m = L_0 + 1 = 4, p_m = 0$), and P_u toggles the gate to closed at $T_m = 4$ with probability $1 - p_m = 1$. As shown in Figure 5.3(right), the optimal policy should take L_0 steps to move directly to 0, for which the value is $v^*_M = v^*_{M^+}$. We have computed for Theorem V.1 that $v^*_{M^-} = -6$. With probability $1 - p_m = 1$, the gate is closed at $T_m = 4$, and $\pi^*_{\widehat{M}}$ takes $L + L_0 - 1 > H$ steps to reach $L - 1$. Thus, $v^{\pi^*_{\widehat{M}}}_M = -H = -10 < v^*_{M^-}$. □

In the example used in the existence proof above, the maximum suboptimality is incurred with maintenance commitment probability $p_m = 0$ (a no-guarantee commitment), because this is when the recipient is most uncertain about the influence and will be most negatively affected by the uncertainty. Note that for achievement, a no-guarantee commitment still falls within the Theorem V.1 bound.

Comparing the bound Equation (5.3) in Theorem V.1 with the bound Equation (5.4) in Theorem V.2 reveals a fundamental difference between achievement and maintenance commitments: maintenance commitments are inherently less tolerant to an unexpected change in the commitment feature. For achievement commitments, the easily-constructed minimal enablement duration influence has the property of being pessimistic, in that any unexpected changes to the feature, if they impact the recipient at all, *can only improve the expected value.* Thus, if despite its minimal enablement duration influence approximation, a recipient has chosen to follow a policy that exploits the commitment, it can never experience a true influence that would lead it to regret having done so. *The same cannot be said for maintenance commitments.* There, the easily-constructed minimal enablement duration influence is *not* pessimistic—it does not guarantee that any deviations from the influence can only improve the expected value. As our theoretical results show, the minimal enablement duration influence assuming toggling from u^+ to u^- right away can still lead to negative surprises, since if the toggling does not immediately occur the influence suggests that it is safe to assume no toggling until T_m, but that is not true since toggling could

happen sooner, after the recipient has incurred cost for a policy that would need to be abandoned. In the example for Theorem V.2, the worst time for toggling to u^- is not right away, but right before the precondition would be used, where the gate shuts just as the recipient is about to pass through it.

5.3.2 Alternative Influence Approximations

Besides using the minimal enablement duration strategy to create the approximate influence, we next consider and analyze several alternative strategies.

Maximal Enablement Duration. As opposed to the minimal enablement duration strategy, the maximal enablement duration strategy optimistically toggles u right after the initial time step for achievement commitments, and at the commitment time for maintenance commitments. Formally, given achievement commitment $c_a = (T_a, p_a)$, the maximal enable duration strategy, denoted as $\widehat{P}_u^{\max+}(\cdot)$, chooses the influence $\widehat{P}_u^{\max+}(c_a)$ that toggles u in the transition from time step $t = 0$ to $t = 1$ with probability p_a, and does not toggle u at any other time step; given maintenance commitment $c_m = (T_m, p_m)$, the maximal enablement duration strategy chooses the influence $\widehat{P}_u^{\max+}(c_m)$ that toggles u in the transition from time step $t = T_m - 1$ to $t = T_m$ with probability $1-p_m$, and (unless already toggled) from $t = T_m$ to $t = T_m+1$ with probability one. It does not toggle u at any other time step.

Constant Toggling. The constant toggling strategy, denoted as $\widehat{P}_u^{\text{const}}(\cdot)$, chooses the influence $\widehat{P}_u^{\text{const}}(c)$, for either an achievement or a maintenance commitment c, that toggles u at every time step up to the commitment time with a constant probability, and the probability is chosen such that the overall probability of toggling by the commitment time matches the commitment probability. The influence $\widehat{P}_u^{\text{const}}(c)$ agrees with the minimal enablement duration influence after the commitment time.

Minimal Value Timing. Both the minimal and maximal enablement duration strategies choose influences that model a single timestep no later than the commitment time and agree with Assumption V.3 thereafter. We denote the set of such influences as $\mathcal{P}_u^1(c)$ for either an achievement or a maintenance commitment c. The minimal value timing strategy, denoted as $\widehat{P}_u^{\text{minV}}(\cdot)$, chooses the influence from $\mathcal{P}_u^1(c)$ that has the minimal optimal value. Formally, for either an achievement or a maintenance commitment c, its minimal enablement duration influence $\widehat{P}_u^{\text{minV}}(c)$ is $\arg\min_{\widehat{P}_u \in \mathcal{P}_u^1(c)} v^*_{\widehat{M}}$ where $\widehat{P}_u$ is the influence in $\widehat{M}$.

Minimax Regret Timing. The minimax regret timing strategy $\widehat{P}_u^{\text{minimax}}(\cdot)$ chooses an influence from $\mathcal{P}_u^1(c)$ based on the minimax regret principle. Formally, for either an achievement or a maintenance commitment c, its minimax regret timing influence $\widehat{P}_u^{\text{minimax}}(c)$ is

$$\arg\min_{\widehat{P}_u \in \mathcal{P}_u^1(c)} \max_{P_u \in \mathcal{P}_u^1(c)} v_M^* - v_M^{\pi^*_{\widehat{M}}}$$

where $P_u, \widehat{P}_u$ are the influences in $M, \widehat{M}$, respectively.

The four strategies to create approximate influences, together with the minimal enablement duration strategy, include three heuristics that are computationally inexpensive to compute (minimal and maximal enablement duration, and constant toggling), and two more heuristics that are complex and expensive to compute (minimal value and minimax regret timing). Except for the constant toggling, all strategies create approximate influences that model a single branch for when u probabilistically toggles, and this single branching induces minimal computation cost for the recipient's planning. Recall that our theoretical analysis suggests, for maintenance commitments, the pessimistic time for toggling to u^- is not right away, but right before the recipient uses the precondition, and this causes the poor performance of the minimal enablement duration influence. One might hypothesize that the constant toggling can be more pessimistic for maintenance (and thus better) than the minimal enablement duration, because it projects the possibility of toggling to u^- at every single time step before the commitment time. One might also hypothesize that the latter two heuristics can be more pessimistic (and thus better) than the minimal enablement duration influence by identifying the worst possible toggling time. However, Theorem V.3 states that, while the minimal value timing influence coincides with the minimal enablement duration for achievement and therefore enjoys the same bound in Equation (5.3) for the worst-case suboptimality, *the bound does not hold for any of the alternative strategies in either achievement or maintenance.*

Theorem V.3. *For an achievement commitment c_a, the minimal value timing influence coincides with the minimal enablement duration, i.e. $\widehat{P}_u^{\text{minV}}(c_a) = \widehat{P}_u^{\text{min+}}(c_a)$, and thus the bound in Equation* (5.3) *holds for $\widehat{P}_u^{\text{minV}}(c_a)$. Except for this, the bound does not hold, i.e. for $\widehat{P}_u \in \{\ \widehat{P}_u^{\text{max+}}(c_a),\ \widehat{P}_u^{\text{const}}(c_a),\ \widehat{P}_u^{\text{minimax}}(c_a),\ \widehat{P}_u^{\text{max+}}(c_m),\ \widehat{P}_u^{\text{const}}(c_m),\ \widehat{P}_u^{\text{minV}}(c_m),\ \widehat{P}_u^{\text{minimax}}(c_m)\}$, there exists an MDP M, and an achievement or maintenance commitment, such that the true influence P_u in M respects the commitment semantics of $c \in \{c_m, c_a\}$, and the suboptimality*

$$v_M^* - v_M^{\pi^*_{\widehat{M}}} > v_{M^+}^* - v_{M^-}^*$$

exceeds the bound in Equation (5.3).

Proof. We first show that the minimal value timing influence coincides with the minimal enablement duration for achievement commitments, i.e. $\widehat{P}_u^{\text{minV}}(c_a) = \widehat{P}_u^{\text{min+}}(c_a)$. Consider achievement commitment $c_a = (T_a, p_a)$, and $\widehat{P}_u, \widehat{P}'_u \in \mathcal{P}_u^1(c_a)$ that toggles u at T and T' respectively with $T' < T \leq T_a$. We can construct a recipient's policy for the earlier toggling $\widehat{P}'_u$ that mimics the optimal policy for $\widehat{P}_u$, and hence the optimal value for T' is at least that for T, i.e. $v^*_{\widehat{M}} \leq v^*_{\widehat{M}'}$ where $\widehat{P}_u$ and $\widehat{P}'_u$ are the influences in $\widehat{M}$ and $\widehat{M}'$, respectively. Specifically, let $\pi^*_{\widehat{M}}$ be the optimal policy for $\widehat{P}_u$ and $\pi^*_{\widehat{M}}(\cdot|s)$ be the action probability distribution of $\pi^*_{\widehat{M}}$ in state s. For the earlier toggling time $T' < T$, we construct a policy $\pi_{T'}$ that mimics $\pi^*_{\widehat{M}}$: it chooses actions as if $u = u^-$ until T. Formally, for timesteps $t < T$, $\pi_{T'}(\cdot|s^-) = \pi^*_{\widehat{M}}(\cdot|s^-)$ for any state $s^- = (l, u^-)$ in which $u = u^-$, and $\pi_{T'}(\cdot|s^+) = \pi^*_{\widehat{M}}(\cdot|s^-)$ where $s^+ = (l, u^+)$ and $s^- = (l, u^-)$ only differ in u; for timesteps $t \geq T$, $\pi_{T'}(\cdot|s^r) = \pi^*_{\widehat{M}}(\cdot|s^r)$. Because $\pi_{T'}$ and $\pi^*_{\widehat{M}}$ yield the same trajectories of l and the reward only depends on l, they achieve the same value, and therefore $v^*_{\widehat{M}} \leq v^*_{\widehat{M}'}$.

As an existence proof, Table 5.1 summarizes examples for which the bound in Equation (5.3) does not hold for $\widehat{P}_u \in \{\, \widehat{P}_u^{\max+}(c_a),\, \widehat{P}_u^{\text{const}}(c_a),\, \widehat{P}_u^{\text{minimax}}(c_a),\, \widehat{P}_u^{\max+}(c_m),\, \widehat{P}_u^{\text{const}}(c_m),\, \widehat{P}_u^{\text{minV}}(c_m),\, \widehat{P}_u^{\text{minimax}}(c_m) \,\}$. All the examples are in the 1D Walk domain with fixed $L = 10, H = 20$. Besides the -1 reward for every time step until reaching either end, the recipient also gets a one-time reward when reaching the left end of r_{left}, which is an integer chosen from interval $[0, 10]$. We compute the suboptimality for a commitment c, achievement or maintenance, with initial location L_0 chosen from $\{1, 2, 3, ..., 8\}$, commitment time chosen from $\{1, 2, ..., H\}$ and commitment probability chosen from $\{0, 0.1, 0.2, ..., 1\}$, with the provider's true influence P_u chosen from $\mathcal{P}_u^1(c)$. For all possible combinations of r_{left}, c, L_0, and P_u, the corresponding suboptimality is evaluated, and Table 5.1 reports combinations for which the bound in Equation (5.3) does not hold. □

While the analysis in this section chooses the provider's true influence from $\mathcal{P}_u^1(c)$ in an adversarial manner from the recipient's perspective, a rational provider that maximizes its value can indeed induce such an influence in $\mathcal{P}_u^1(c)$ for any given commitment c, as formally stated in Theorem V.4.

Theorem V.4. *For any commitment c, achievement or maintenance, and any influence $P_u \in \mathcal{P}_u^1(c)$, there exists an MDP for the provider such that the optimal policy induces influence P_u.*

Table 5.1: 1D Walk Examples for Theorem V.3

	Achievement	Maintenance
Min Enablement	The bound in Eq. (5.3) holds	$L = 10, L_0 = 3, r_{\text{left}} = 0$ $T_m = 4, p_m = 0.0$ $v^*_{M^+} - v^*_{M^-} = -3 - (-6) = 3$ $P_u \in \mathcal{P}^1_{u,c}$ toggles at t=3 Suboptimality = 8.8
Max Enablement	$L = 10, L_0 = 6, r_{\text{left}} = 7$ $T_a = 4, p_a = 0.9$ $v^*_{M^+} - v^*_{M^-} = 1 - (-3) = 4$ $P_u \in \mathcal{P}^1_{u,c}$ toggles at t=3 Suboptimality = 4.7	$L = 10, L_0 = 3, r_{\text{left}} = 0$ $T_m = 3, p_m = 0.0$ $v^*_{M^+} - v^*_{M^-} = -3 - (-6) = 3$ $P_u \in \mathcal{P}^1_{u,c}$ toggles at t=1 Suboptimality = 4
Constant Toggling	$L = 10, L_0 = 3, r_{\text{left}} = 0$ $T_a = 7, p_a = 0.9$ $v^*_{M^+} - v^*_{M^-} = -3 - (-6) = 3$ $P_u \in \mathcal{P}^1_{u,c}$ toggles at t=6 Suboptimality = 4.0	$L = 10, L_0 = 3, r_{\text{left}} = 0$ $T_m = 7, p_m = 0.1$ $v^*_{M^+} - v^*_{M^-} = -3 - (-6) = 3$ $P_u \in \mathcal{P}^1_{u,c}$ toggles at t=1 Suboptimality = 3.3
Min Value	The bound in Eq. (5.3) holds	$L = 10, L_0 = 3, r_{\text{left}} = 9$ $T_m = 7, p_m = 0.3$ $v^*_{M^+} - v^*_{M^-} = 6 - (-6) = 12$ $P_u \in \mathcal{P}^1_{u,c}$ toggles at t=5 Suboptimality = 14.9
Minimax Regret	$L = 10, L_0 = 6, r_{\text{left}} = 7$ $T_a = 5, p_a = 1.0$ $v^*_{M^+} - v^*_{M^-} = 1 - (-3) = 4$ $P_u \in \mathcal{P}^1_{u,c}$ toggles at t=4 Suboptimality = 5.8	$L = 10, L_0 = 3, r_{\text{left}} = 0$ $T_m = 4, p_m = 0.0$ $v^*_{M^+} - v^*_{M^-} = -3 - (-6) = 3$ $P_u \in \mathcal{P}^1_{u,c}$ toggles at t=3 Suboptimality = 8.8

Proof. For achievement commitment $c = c_a = (T_a, p_a)$ and influence $P_u \in \mathcal{P}^1_u(c_a)$ that toggles at time step $T \leq T_a$, consider 1D Walk of $T_a + 1$ locations on $[0, T_a]$ as the provider's MDP, where the provider starts at location 0. The provider gets a reward of $+1$ for each time step at location T_a, and a reward of 0 everywhere else. For each time step at location T, the provider toggles the value of u from u^- to u^+ with probability p_a. Obviously, the provider's optimal policy is to move to and then stay at location T_a, which induces influence P_u.

Similarly, for maintenance commitment $c = c_m = (T_m, p_m)$ and influence $P_u \in \mathcal{P}^1_u(c_m)$ that toggles at time step $T \leq T_m$, consider the same 1D Walk of $T_m + 1$ locations as the provider's MDP, except that the provider toggles the value of u from u^+ to u^- with probability $1 - p_m$ at location T. The provider's optimal policy remains the same, which induces influence P_u. □

As a brief summary, in this section we have developed several strategies for the recipient to create the approximate influence, and theoretically analyzed their worst-case suboptimalities for both achievement and maintenance commitments. Our theoretical results show that there exists a strategy, minimal enablement duration, such that its worst-case suboptimality is reasonably bounded for achievement commitments. However, such a guarantee does not hold for maintenance commitments for *any* of the strategies we have considered. This not only includes the counterpart minimal enablement duration strategy but also the strategies that are purposely developed using insights about worst-case timing of the toggling, as well as the constant toggling strategy that models the toggling at every time step. While we cannot assert that a bounded strategy does not exist for maintenance commitments, we have shown that strategies specifically developed to account for the shortcomings of others nonetheless can still induce the worst-case unbounded suboptimality.

5.4 Empirical Study

In Section 5.3, we have developed several strategies for the recipient to create the approximate influence for a given (achievement or maintenance) commitment, and analyzed their worst-case suboptimalities. Specifically, we have contrived MDPs for the recipient in the 1D Walk domain, commitments, and the provider's true influences respecting the commitment semantics, to maximize the suboptimality induced by the approximate influences. We have shown that, for achievement, the worst-case suboptimality of the minimal enablement duration influence (or equivalently the minimal value timing influence) can be be bounded fairly tightly, while for maintenance the worst-case suboptimality of any approximate influence we have developed is effectively unbounded.

In this section, we conduct empirical evaluations of the suboptimality induced by those approximate influences besides the worst case. In Section 5.4.1, we measure suboptimality for general (achievement or maintenance) commitments in 1D Walk with various choices of commitment time and probability. In Section 5.4.2, we focus on value maximizer commitments, which either maximize the provider's or the recipient's local commitment value, or maximize the joint commitment value.

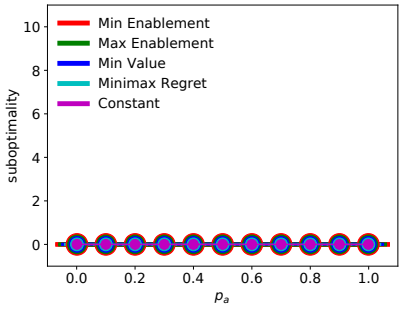
5.4.1 Suboptimality for General Commitments

Here, we measure the suboptimality of the strategies to create approximate influences developed in Section 5.3 for a general achievement commitment $c_a = (T_a, p_a)$

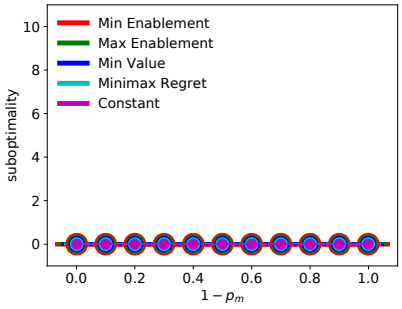
or maintenance commitment $c_m = (T_m, p_m)$ in 1D Walk, where the commitment time $T_a, T_m \in \{1, 2, ..., H\}$ can be any time step by the horizon and the commitment probability $p_a, p_m \in \{\frac{i}{n}\}_{i=0}^{n}$ is chosen from the interval $[0, 1]$ evenly discretized with $n = 10$. For a given (achievement or maintenance) commitment c, we measure the suboptimality with respect to all the influences in $\mathcal{P}_u^1(c)$ as the provider's true influence. The parameters for 1D Walk are the same as the example for Theorem V.1 except that the horizon is longer, $L = 10, L_0 = 3, H = 20$.

Figure 5.4 shows the mean, minimum, and maximum suboptimality over all realizations of the provider's true influence $P_u \in \mathcal{P}_u^1$ for commitment time $T_a, T_m \in \{1, 5, 10, 15\}$. We see that for achievement commitments, the suboptimality of the minimal enablement duration (or equivalently the minimal value timing) influence incurs the lowest suboptimality. The more expensive minimax regret timing influence has comparable suboptimality. The other two, maximal enablement duration and the constant toggling influences, incur the most suboptimality overall. For maintenance commitments, the minimal enablement duration and the minimax regret influences incur the *most* suboptimality overall, and, among the other three approximate influences, it is difficult to identify a single best influence that reliably reduces the suboptimality for all the maintenance commitments. The maximal enablement duration strategy has the lowest mean suboptimality overall, yet the maximum suboptimality it induces over candidate true influences can be quite high especially when p_m is close to one. On the contrary, the constant toggling strategy incurs higher mean suboptimality than the maximal enablement duration, yet its maximum suboptimality is consistently lower. The suboptimality of the minimum value timing strategy is the median among the five.

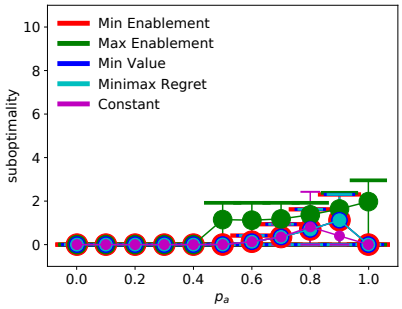
For both achievement and maintenance commitments, a larger commitment time and a larger commitment probability tend to induce higher suboptimality. This is because the recipient has more uncertainty about the provider's true influence when both the commitment time and probability are larger. In the extreme, as shown in Figures 5.4a and 5.4b, for commitment time $T_a = T_m = 1$ the recipient has no uncertainty about the toggling time, and hence the suboptimality is zero given that the provider's true influence $P_u \in \mathcal{P}_u^1$ matches the commitment probability. When the commitment probability is $p_a = p_m = 0$, the suboptimality is also zero since the recipient's approximate influence matches the provider's true influence in the sense that there is no toggling in either of the two influences. The same reasoning explains why the largest suboptimality occurs at $p_a = p_m = 1.0$.



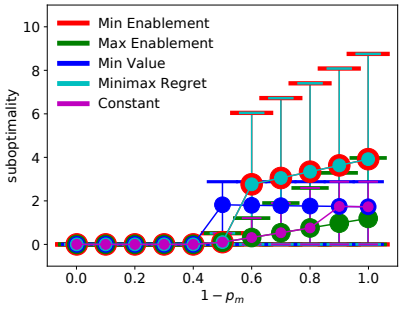
(a) Achievement, $T_a = 1$



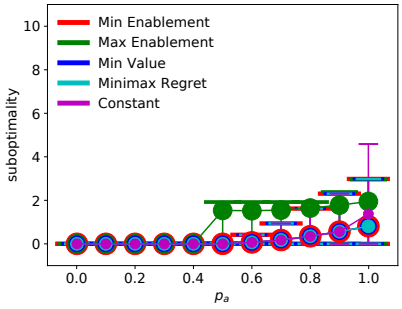
(b) Maintenance, $T_m = 1$



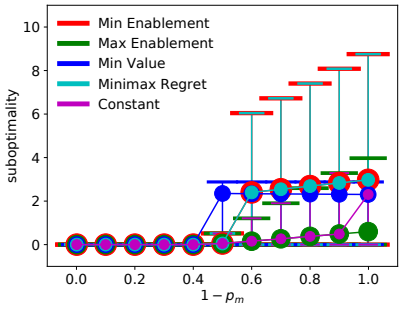
(c) Achievement, $T_a = 5$



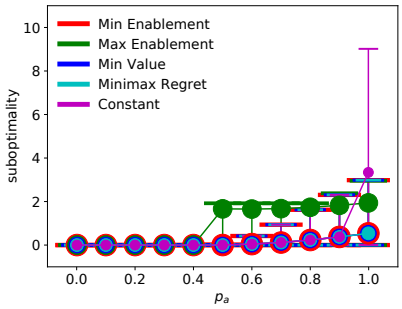
(d) Maintenance, $T_m = 5$



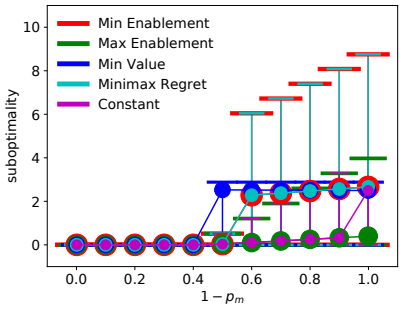
(e) Achievement, $T_a = 10$



(f) Maintenance, $T_m = 10$



(g) Achievement, $T_a = 15$



(h) Maintenance, $T_m = 15$

Figure 5.4: Suboptimality in 1D Walk. Please view in color. The results are for the recipient with $L = 10, L_0 = 3, H = 20$. Markers on the curves show the mean suboptimality over possible true influences that toggles at a single time step before the commitment time, $P_u \in \mathcal{P}^1_{u,c}$. Bars show the minimum and maximum.

5.4.2 Suboptimality for Value Maximizer Commitments

So far, in both Sections 5.3 and 5.4.1, we are concerned with the suboptimality that is concerned with general commitments with various commitment times and probabilities. Here, we introduce an environment that explicitly incorporates the provider's commitment value, and we focus on commitments that are rationally chosen to be value maximizers, which either maximize the provider's commitment value $v^{\mathrm{p}}(c)$, the recipient commitment value $v^{\mathrm{r}}(c)$, or the joint commitment value $v^{\mathrm{p}}(c) + v^{\mathrm{r}}(c)$. We make a note here that these rationally-chosen commitments are likely to be the ones adopted by the agents, and they are not chosen in favor of a particular type of commitment, nor in favor of a particular approximate influence. Moreover, in both Sections 5.3 and 5.4.1 we have been concerned with the virtual provider with its true influence $P_u \in \mathcal{P}_u^1(c)$ toggling u at a single time step no later than the commitment time. In this section, we are concerned with the more general situation in which the true influence P_u is not restricted to be an element in $\mathcal{P}_u^1(c)$; instead, P_u is naturally determined by the provider's policy that maximizes its own value while respecting the commitment semantics. We first describe the recipient's and the provider's environments below.

The recipient's environment. The recipient's environment is the same 1D Walk domain used for the proof of Theorem V.3. Specifically, the recipient is in a one-dimensional space with $L = 10$ locations represented as integers $\{0, 1, ..., 9\}$. The starting location L_0 is randomly chosen from locations $1 - 8$. The horizon for both agents is set to be $H = 20$. The one-time reward of r_{left} is randomly sampled from $[0, 10]$. In a specific instantiation of the recipient's MDP, L_0 and r_{left} are fixed, and they are randomly chosen to create various MDPs for the recipient. Since the left end has higher rewards than the right end, if the recipient's start position is close enough to the left end and the provider commits to opening the gate early enough with high enough probability, the recipient should utilize the commitment by checking if the gate is open by the commitment time, and pass through it if so; otherwise, the recipient should simply ignore the commitment and move to the right end. Thus, the various instances of the recipient's MDP include diverse preferences regarding the commitments.

The provider's environment. The provider's MDP is randomly generated from a distribution designed such that, in expectation, the provider's value when enabling the precondition is smaller than when not enabling it. This introduces tension in the

provider between enabling the precondition to help the recipient, versus increasing its own reward. We now describe the provider's MDP-generating distribution. The MDP has 10 states the provider can be in at any time step, one out of which is an absorbing state denoted as s^+, and where the initial state is chosen from the non-absorbing states. There are 3 actions. For each state-action pair $(s^{\mathrm{p}}, a^{\mathrm{p}})$ where $s^{\mathrm{p}} \neq s^+$, the transition function $P^{\mathrm{p}}(\cdot|s^{\mathrm{p}}, a^{\mathrm{p}})$ is determined independently by filling the 10 entries with values uniformly drawn from $[0, 1]$, and normalizing $P^{\mathrm{p}}(\cdot|s^{\mathrm{p}}, a^{\mathrm{p}})$. For achievement commitments, feature u takes the value of u^+ only in the absorbing state, i.e. $u^+ \in s^{\mathrm{p}}$ if and only if $s^{\mathrm{p}} = s^+$, and the reward $R^{\mathrm{p}}(s^{\mathrm{p}}, a^{\mathrm{p}})$ for a non-absorbing state $s^{\mathrm{p}} \neq s^+$ is sampled uniformly and independently from $[0, 1]$, and for the absorbing state $s^{\mathrm{p}} = s^+$ is zero, meaning the provider prefers to avoid the absorbing state, but that state is the only one that enables the precondition and realizes the achievement commitment. For maintenance commitments, feature u takes the value of u^+ only in the non-absorbing states, i.e. $u^+ \in s^{\mathrm{p}}$ if and only if $s^{\mathrm{p}} \neq s^+$, and the reward $R^{\mathrm{p}}(s^{\mathrm{p}}, a^{\mathrm{p}})$ for a non-absorbing state $s^{\mathrm{p}} \neq s^+$ is sampled uniformly and independently from $[-1, 0]$, and for the absorbing state $s^{\mathrm{p}} = s^+$ is zero, meaning the provider prefers to reach the absorbing state, but that state disables the precondition and fails the maintenance commitment.

We observe that, for small values of commitment time, the provider's maximum feasible probability of toggling u, or equivalently reaching s^+, by the commitment time is fairly low. Hence, in some experiments we also introduce a fourth action for the provider, a^+, such that, after taking a^+ in any non-absorbing state $s^{\mathrm{p}} \neq s^+$, the provider will transit to the absorbing state s^+ with probability $p^+_{s^{\mathrm{p}}}$, and will stay in the current state s^{p} with probability $1 - p^+_{s^{\mathrm{p}}}$. For each non-absorbing state $s^{\mathrm{p}} \neq s^+$, $p^+_{s^{\mathrm{p}}}$ is sampled from a Gaussian distribution and then clipped into $[0, 1]$. In a specific instantiation of the provider's MDP, the mean of the Gaussian distribution, denoted as p^+, is chosen from $\{0, 0.5, 0.9\}$, and standard deviation is fixed to 0.1.

Results. Tables 5.2, 5.3, 5.4, and 5.5 show the suboptimality for the value maximizer commitments without action a^+, with action a^+ and $p^+ = 0, 0.5$, and 0.9, respectively, each reporting the means and standard errors over 2500 randomly-generated pairs of the provider's MDP and the recipient's MDP. Since the problem instances have different reward scales, the suboptimality is normalized by the bound in Equation (5.3), i.e. $v^*_{M^+} - v^*_{M^-}$. The tables highlight strategies that induce low suboptimality for certain types of value maximizer commitments, with mean+error $\leq 5\%$ underlined and mean+error $\leq 1\%$ in bold.

Table 5.2: Suboptimality for maximizer commitments (without action a^+ for the provider). The suboptimality is normalized by $v^*_{M^+} - v^*_{M^-}$. The results are means and standard errors (in parentheses). Mean + standard error below 5% are underlined, and below 1% are in bold.

		Suboptimality (%)		
		Provider Value Maximizer	Joint Value Maximizer	Recipient Value Maximizer
Achv.	Min Enablement Min Value	**0.21 (0.03)**	**0.27 (0.03)**	**0.40 (0.03)**
	Max Enablement	26.51 (0.61)	29.25 (0.65)	28.75 (0.65)
	Minimax Regret	6.44 (0.23)	7.78 (0.26)	7.20 (0.26)
	Constant Toggling	**0.03 (0.01)**	**0.06 (0.01)**	**0.97 (0.04)**
Maint.	Min Enablement	9.93 (0.83)	4.00 (0.56)	1.55 (0.18)
	Max Enablement	11.04 (0.74)	11.04 (0.74)	11.04 (0.74)
	Min Value	15.02 (1.11)	10.82 (1.06)	8.74 (0.96)
	Minimax Regret	10.17 (0.83)	7.24 (0.62)	7.63 (0.58)
	Constant Toggling	9.47 (1.01)	7.56 (0.91)	**0.02 (0.01)**

Table 5.3: Suboptimality for maximizer commitments ($p^+ = 0$). The suboptimality is normalized by $v^*_{M^+} - v^*_{M^-}$. The results are means and standard errors (in parentheses). Mean + standard error below 5% are underlined, and below 1% are in bold.

		Suboptimality (%)		
		Provider Value Maximizer	Joint Value Maximizer	Recipient Value Maximizer
Achv.	Min Enablement Min Value	**0.01 (0.01)**	**0.01 (0.01)**	**0.31 (0.02)**
	Max Enablement	2.69 (0.21)	14.81 (0.33)	34.28 (0.69)
	Minimax Regret	**0.66 (0.09)**	6.01 (0.25)	10.15 (0.37)
	Constant Toggling	**0.01 (0.01)**	**0.01 (0.01)**	**0.46 (0.02)**
Maint.	Min Enablement	6.31 (0.67)	**0.68 (0.21)**	**0.01 (0.01)**
	Max Enablement	8.12 (0.63)	8.12 (0.63)	4.80 (0.49)
	Min Value	14.42 (1.15)	6.62 (0.87)	0.97 (0.33)
	Minimax Regret	7.30 (0.65)	7.56 (0.60)	4.45 (0.46)
	Constant Toggling	6.33 (0.83)	2.56 (0.91)	**0.01 (0.01)**

Table 5.4: Suboptimality for maximizer commitments ($p^+ = 0.5$). The suboptimality is normalized by $v^*_{M^+} - v^*_{M^-}$. The results are means and standard errors (in parentheses). Mean + standard error below 5% are underlined, and below 1% are in bold.

		Suboptimality (%)		
		Provider Value Maximizer	Joint Value Maximizer	Recipient Value Maximizer
Achv.	Min Enablement Min Value	**0.16(0.02)**	**0.12 (0.01)**	**0.14 (0.01)**
	Max Enablement	28.00 (0.63)	31.69 (0.69)	38.08 (0.63)
	Minimax Regret	6.82 (0.23)	9.67 (0.34)	4.53 (0.30)
	Constant Toggling	**0.01 (0.01)**	10.66 (0.43)	**0.02 (0.01)**
Maint.	Min Enablement	22.66 (0.70)	3.08 (0.36)	1.74 (0.20)
	Max Enablement	45.09 (1.54)	45.09 (1.54)	45.09 (1.54)
	Min Value	6.33 (0.39)	2.81 (0.35)	2.17 (0.31)
	Minimax Regret	22.99 (0.70)	4.72 (0.35)	10.62 (0.65)
	Constant Toggling	4.36 (0.37)	2.48 (0.34)	**0.01 (0.01)**

Table 5.5: Suboptimality for maximizer commitments ($p^+ = 0.9$). The suboptimality is normalized by $v^*_{M^+} - v^*_{M^-}$. The results are means and standard errors (in parentheses). Mean + standard error below 5% are underlined, and below 1% are in bold.

		Suboptimality (%)		
		Provider Value Maximizer	Joint Value Maximizer	Recipient Value Maximizer
Achv.	Min Enablement Min Value	**0.16 (0.02)**	**0.10 (0.01)**	**0.01 (0.01)**
	Max Enablement	27.89 (0.63)	32.40 (0.67)	32.36 (0.67)
	Minimax Regret	6.71 (0.23)	9.58 (0.36)	5.15 (0.31)
	Constant Toggling	**0.01 (0.01)**	52.79 (1.48)	**0.01 (0.01)**
Maint.	Min Enablement	10.40 (0.48)	**0.71 (0.16)**	1.72 (0.20)
	Max Enablement	46.83 (1.32)	50.00 (1.34)	50.00 (1.34)
	Min Value	1.66 (0.13)	**0.52 (0.11)**	**0.45 (0.10)**
	Minimax Regret	9.17 (0.42)	5.62 (0.24)	13.13 (0.64)
	Constant Toggling	1.80 (0.13)	**0.45 (0.10)**	**0.01 (0.01)**

For achievement commitments, the minimal enablement duration (or equivalently the minimal value timing) strategy consistently induces suboptimality below 1% with or without action a^+, for all the three types of maximizer commitment, while the maximal enablement duration and the minimax regret often induce suboptimality higher than 5%. Table 5.2 shows that, without action a^+, the constant toggling influence also induces suboptimality below 1% for all three types of maximizer commitment, and this also holds with action a^+ and a small $p^+ = 0$ as shown in Table 5.3. However, as p^+ increases, the constant toggling influence can induce suboptimality higher than 5%, especially for the joint value maximizer commitments, as shown in Tables 5.4 and 5.5. Generally, the provider value maximizers are those weak achievement commitments with late commitment time T_a and low commitment probability p_a, while the recipient value maximizers are those strong commitments with early T_a and high p_a. Since later commitment time T_a and higher commitment probability p_a often cause the recipient more uncertainty about the true influence and therefore higher suboptimality (as evidenced by the results in Figures 5.4a, 5.4c, 5.4e, and 5.4g), it is difficult to predict which type of value maximizer induces higher suboptimality. Thus, it should be unsurprising that some strategies work well for one type of value maximizer achievement commitment but not for another. Nonetheless, the minimal enablement duration (or equivalently the minimal value timing) strategy consistently induces low suboptimality for all types of value maximizer achievement commitment.

For maintenance commitments, the results show that none of the five strategies has suboptimality below 1% consistently for all three types of maximizer commitment, with or without action a^+. Overall, the suboptimality of all five strategies for maintenance is significantly higher than the suboptimality of the minimal enablement duration strategy for achievement. It is worth noting that, while the maximal enablement duration used to be an above-average strategy for maintenance commitments if the true influence is chosen from $\mathcal{P}_u^1(c_m)$ that toggles only at a single time step (shown in Figures 5.4b, 5.4d, 5.4f, and 5.4h), here we see that the maximal enablement duration is overall the worst among the five strategies, confirming that being optimistic does not result in robust interpretation of maintenance commitments. Similar to achievement commitments, it is difficult to predict which type of value maximizer maintenance commitment is harder for the recipient to model, and a strategy can work well for one value maximizer but not for another. For example, the constant toggling induces lowest suboptimality for recipient value maximizer maintenance commitments, suggesting that, when the commitment time T_m is late and the toggling probability $\leq 1 - p_m$ is low, it is empirically better to model the toggling more often

than a single time step. However, such a claim about the constant toggling strategy does not hold for joint value maximizers, as shown in Table 5.3.

5.5 Summary

In this chapter, we have focused on how the recipient should interpret the partial information specified in a probabilistic commitment. Specifically, a commitment specifies a lower bound on the probability of the commitment being realized at a single time step, and this partial specification imposes uncertainty for the recipient's planning. As described in Section 5.1, the recipient creates an approximate influence that approximates the provider's influence at other time steps. We are particularly interested in the quality of this approximate influence, quantified by its suboptimality, for two types of commitment, that of achievement and maintenance formally defined in Section 5.2 in the probabilistic commitment framework. In Section 5.3, we developed several strategies for the recipient to create the approximate influence, and studied their worst-case suboptimalities that is induced in simple examples of commitments and the provider's true influence. Using theorems in Section 5.3, we were able to identify a straightforward, computationally inexpensive strategy, referred to as the minimal enablement duration, whose worst-case suboptimality for achievement commitments can be bounded, while for maintenance commitments the worst-case suboptimality of any of the strategies is effectively unbounded. Our empirical study in Section 5.4 evaluated the strategies beyond worst-case examples. The results showed that the minimal enablement duration is effective for achievement commitments, while for maintenance none of the strategies can reliably yield low suboptimality.

With the recipient robustly interpreting an achievement commitment, successful coordination with the provider can be secured. On the other hand, the fact that interpreting a maintenance commitment is harder encourages future research in coordination with maintenance. As an immediate next step, one can try to develop and investigate better strategies than the ones we studied in this thesis. In a different direction, one can explore specifications that are more detailed than the single time step specification, which definitely reduce the recipient's uncertainty when creating the approximate influence, but, as a potential cost, could reduce the flexibility the provider needs. More broadly, our results provide insights to the community designing specifications and protocols for applying commitment-based coordination to domains involving both achievement and maintenance.

CHAPTER VI

Efficient Formulation of Cooperative Probabilistic Commitments

We have seen that the semantics of a commitment constrains the provider's policy choice, and thus the provider would prefer a weaker commitment (e.g., lower commitment probability, earlier commitment time for achievement, and later commitment time for maintenance) if it aims to maximize its own value. On the other hand, the recipient would prefer a stronger commitment (e.g., higher commitment probability, later commitment time for achievement, and earlier commitment time for maintenance) since the outcome specified by the commitment is desired. What commitment should they agree on? In this chapter, we focus on formulating a commitment that induces the optimal cooperative behavior between the agents in the sense that the sum of their two values is maximized. This optimal cooperative commitment problem is computationally challenging, because evaluating each commitment involves solving a linear program that is expensive, and thus we aim to avoid exhaustively searching the entire commitment space. We prove several structural properties of the provider's and the recipient's values as functions of the parameters in the commitment specification. This enable us to develop algorithms that exploit the properties to efficiently formulate (near-)optimal cooperative commitments for both the centralized setting (in which each agent's information is known to a centralized coordinator) and the decentralized setting (in which the information relevant to optimization is distributed between the agents).

6.1 Cooperative Probabilistic Commitments

As we have discussed in Section 2.3, for a given feasible probabilistic commitment $c = (u_c, T_c, p_c)$, the provider's commitment value function $v^{\mathrm{p}}(c)$ corresponds to the

provider's policy that maximizes the initial state value while satisfying the commitment constraint. The recipient's value of commitment c, v^{r}, is defined to be the optimal value of the recipient's initial state when planning with whatever it chooses for its approximate influence of the shared state feature. Let $v^{\mathrm{p+r}}(c) = v^{\mathrm{p}}(c) + v^{\mathrm{r}}(c)$ be the joint commitment value function. The optimal commitment is a feasible commitment that maximizes the joint value, i.e. $c^* = \arg\max_c v^{\mathrm{p+r}}(c)$.

In this chapter, we focus on achievement commitments, where the shared state feature u takes binary values of u^+ and u^- and is initially u^-. The provider is constrained to follow a policy that sets u to $u_c = u^+$ desired by the recipient by commitment time T_c with at least probability p_c. As we have shown in Chapter V, the minimal enablement duration, introduced in Section 5.3, is an effective strategy for the recipient to create the approximate influence for achievement, and thus we use it to compute the recipient's commitment value $v^{\mathrm{r}}(c)$. As the recipient's robust interpretation for maintenance is largely an open question, we leave the formulation of cooperative maintenance commitments as future work beyond this thesis. Since u_c is fixed to u^+, we use abbreviation $c = (T_c, p_c)$ for the remainder of this chapter. To formulate the optimal commitment for achievement, we need to specify $(T_c, p_c) \in [H] \times [0, 1]$ where $[H] = \{1, 2, ..., H\}$, i.e.

$$c^* = \arg\max_{(T_c, p_c) \in [H] \times [0,1]} v^{\mathrm{p+r}}(c). \tag{6.1}$$

A naïve strategy for solving the problem in Equation (6.1) is to discretize the commitment probability space, and evaluate every commitment in the discretized space. The finer the discretization is, the more commitments are considered and the better the solution will be. At the same time, the finer the discretization, the larger the computational cost of evaluating every commitment in the discretized commitment space. In Section 6.2, we prove structural properties of the provider's and the recipient's commitment value functions that enable us to develop algorithms that efficiently search for the exact optimal commitment.

6.2 Structure of the Probabilistic Commitment Space

We show that, as functions of the commitment probability, both commitment value functions are monotonic and piecewise linear; the provider's commitment value function is concave, and the recipient's is convex. The proofs for the properties of the provider's commitment value function is agnostic about the commitment type,

and thus can still apply to maintenance commitments. For the recipient, its commitment value function for achievement hinges on the minimal enablement duration influence, and the proofs of the structural properties cannot straightforwardly apply to maintenance.

6.2.1 Properties of the Provider's Commitment Value

Theorem VI.1. *Let $v^{\mathrm{p}}(c) = v^{\mathrm{p}}(T_c, p_c)$ be the provider's commitment value. For any fixed commitment time T_c, $v^{\mathrm{p}}(T_c, p_c)$ has the following properties as a function of commitment probability p_c:*

1. *$v^{\mathrm{p}}(T_c, p_c)$ is monotonically non-increasing in p_c.*
2. *$v^{\mathrm{p}}(T_c, p_c)$ is concave in p_c.*
3. *$v^{\mathrm{p}}(T_c, p_c)$ is piecewise linear in p_c.*

The proof of monotonicity is straightforward: by the inequality constraint in the commitment semantics of Equation (3.1), the set of commitment-constrained policies Π^{p}_c is non-increasing in p_c. To show the concavity and piecewise linearity, we consider the linear program that solves the provider's constrained planning problem in Equation (3.2). We now provide a full proof below.

Proof of Monotonicity. By the commitment semantics of Equation (3.1), $\Pi^{\mathrm{p}}_c = \Pi^{\mathrm{p}}_{T_c, p_c}$ is monotonically non-increasing in p_c for any fixed T_c, i.e. $\Pi^{\mathrm{p}}_{T_c, p'_c} \subseteq \Pi^{\mathrm{p}}_{T_c, p_c}$ for any $p'_c > p_c$. Therefore, $v^{\mathrm{p}}(T_c, p_c)$ is monotonically non-increasing in p_c. □

Proof of Concavity. Consider the linear program (LP) in Figure 3.2 for which the optimal value is $v^{\mathrm{p}}(c)$, as also presented below for convenience:

$$
\begin{aligned}
\max_{x^{\mathrm{p}}} \quad & \sum_{s^{\mathrm{p}}, a^{\mathrm{p}}} x^{\mathrm{p}}(s^{\mathrm{p}}, a^{\mathrm{p}}) R^{\mathrm{p}}(s^{\mathrm{p}}, a^{\mathrm{p}}) \\
\text{subject to } \forall s^{\mathrm{p}}, a^{\mathrm{p}} \quad & x^{\mathrm{p}}(s^{\mathrm{p}}, a^{\mathrm{p}}) \geq 0 \\
\forall s^{\mathrm{p}\prime} \quad & \sum_{a^{\mathrm{p}\prime}} x^{\mathrm{p}}(s^{\mathrm{p}\prime}, a^{\mathrm{p}\prime}) = \sum_{s^{\mathrm{p}}, a^{\mathrm{p}}} x^{\mathrm{p}}(s^{\mathrm{p}}, a^{\mathrm{p}}) P(s^{\mathrm{p}\prime} | s^{\mathrm{p}}, a^{\mathrm{p}}) + \delta(s^{\mathrm{p}\prime}, s^{\mathrm{p}}_0) \\
& \sum_{s^p_{T_c} \ni u_c} \sum_{a^p} x^{\mathrm{p}}(s^p_{T_c}, a^p) \geq p_c.
\end{aligned}
$$

For a fixed commitment time T_c and any two commitment probabilities p_c and p'_c, let $x^{\mathrm{p}}_c, x^{\mathrm{p}}_{c'}$ be the optimal solutions to the LP for commitments $c = (T_c, p_c)$, $c' =$

(T_c, p'_c), respectively. For any $\eta \in [0,1]$, let $p_{c,\eta} = \eta p'_c + (1-\eta)p_c$. Consider x^{p}_η that is the η-interpolation of $x^{\mathrm{p}}_c, x^{\mathrm{p}}_{c'}$,

$$x^{\mathrm{p}}_\eta(s^{\mathrm{p}}, a^{\mathrm{p}}) = \eta x^{\mathrm{p}}_{c'}(s^{\mathrm{p}}, a^{\mathrm{p}}) + (1-\eta) x^{\mathrm{p}}_c(s^{\mathrm{p}}, a^{\mathrm{p}}).$$

Note that x^{p}_η satisfies the first two constraints, and so it is the occupancy measure of policy π^{p}_η defined as

$$\pi^{\mathrm{p}}_\eta(a^{\mathrm{p}}|s^{\mathrm{p}}) = \frac{x^{\mathrm{p}}_\eta(s^{\mathrm{p}}, a^{\mathrm{p}})}{\sum_{a^{\mathrm{p}}} x^{\mathrm{p}}_\eta(s^{\mathrm{p}}, a^{\mathrm{p}})}.$$

Since the occupancy measure of π^{p}_η is the η-interpolation of x^{p}_c and $x^{\mathrm{p}}_{c'}$, it is easy to verify that π^{p}_η is feasible for commitment probability $p_{c,\eta}$. Therefore, the concavity holds because

$$\begin{aligned} v^{\mathrm{p}}(T_c, p_{c,\eta}) \geq V^{\pi^{\mathrm{p}}_\eta}_{M^{\mathrm{p}}}(s^{\mathrm{p}}_0) &= \sum_{s^{\mathrm{p}}, a^{\mathrm{p}}} x^{\mathrm{p}}_\eta(s^{\mathrm{p}}, a^{\mathrm{p}}) R^{\mathrm{p}}(s^{\mathrm{p}}, a^{\mathrm{p}}) \\ &= \sum_{s^{\mathrm{p}}, a^{\mathrm{p}}} \left(\eta x^{\mathrm{p}}_{c'}(s^{\mathrm{p}}, a^{\mathrm{p}}) + (1-\eta) x^{\mathrm{p}}_c(s^{\mathrm{p}}, a^{\mathrm{p}})\right) R^{\mathrm{p}}(s^{\mathrm{p}}, a^{\mathrm{p}}) \\ &= \eta v^{\mathrm{p}}(T_c, p'_c) + (1-\eta) v^{\mathrm{p}}(T_c, p_c). \end{aligned}$$

□

Piecewise Linearity. We first convert the LP into its standard form [BT97]:

$$\begin{aligned} \max_{\tilde{x}^{\mathrm{p}}} \; & r^T \tilde{x}^{\mathrm{p}} \\ \text{subject to } & A\tilde{x}^{\mathrm{p}} = b \\ & \tilde{x}^{\mathrm{p}} \geq 0. \end{aligned}$$

To convert the commitment constraint into an equality constraint, we introduce a slack variable $\xi \geq 0$:

$$\sum_{s^{\mathrm{p}}_{T_c} \ni u_c} \sum_{a^{\mathrm{p}}} x(s^{\mathrm{p}}_{T_c}, a^{\mathrm{p}}) - \xi = p_c.$$

The slack variable is a decision variable in the standard form, $\tilde{x}^{\mathrm{p}} = [x^{\mathrm{p}} \mid \xi] \in \mathbb{R}^{|\mathcal{S}^{\mathrm{p}}||\mathcal{A}^{\mathrm{p}}|+1}$. The standard form eliminates redundant constraints so that $A \in \mathbb{R}^{m \times (|\mathcal{S}^{\mathrm{p}}||\mathcal{A}^{\mathrm{p}}|+1)}$ is full row rank ($rank(A) = m$). Note that the elimination produces $b \in \mathbb{R}^m$ whose elements are linear in p_c.

Pick a set of indices B corresponding to m columns of the matrix A. We can think of A as the concatenation of two matrices A_B and A_N where A_B is the $m \times m$ matrix of these m linearly independent columns, and A_N contains the other columns. Correspondingly, $\tilde{x}^{\mathrm{p}}$ is decomposed into $\tilde{x}^{\mathrm{p}}_B$ and $\tilde{x}^{\mathrm{p}}_N$. Then, $\tilde{x}^{\mathrm{p}} = [\tilde{x}^{\mathrm{p}}_B \mid \tilde{x}^{\mathrm{p}}_N]$ is basic feasible if $\tilde{x}^{\mathrm{p}}_N = 0$, A_B is invertible, and $\tilde{x}^{\mathrm{p}}_B = A_B^{-1} b \geq 0$.

It is known that the optimal solution can be found in the basic feasible solutions,

$$\begin{aligned} v^{\mathrm{p}}(T_c, p_c) &= \max_{B:\tilde{x}^{\mathrm{p}} \text{ is basic feasible}} r^T \tilde{x}^{\mathrm{p}} \\ &= \max_{B:\tilde{x}^{\mathrm{p}} \text{ is basic feasible}} r_B^T \tilde{x}^{\mathrm{p}}_B \\ &= \max_{B:\tilde{x}^{\mathrm{p}} \text{ is basic feasible}} r_B^T A_B^{-1} b. \end{aligned}$$

Since b is linear in p_c, $v^{\mathrm{p}}(T_c, p_c)$ is the maximum of a set of linear functions in p_c, and therefore it is piecewise linear. □

6.2.2 Properties of the Recipient's Commitment Value

We here also make Assumptions V.1 and V.2 that we have made in Chapter V when analyzing the worst-case suboptimality of the recipient's approximate influence. Recall that these assumptions imply Lemma V.1 that formalizes the notion that u^+, as opposed to u^-, is the value of u that is desirable for the recipient.

Theorem VI.2. *Let $v^{\mathrm{r}}(c) = v^{\mathrm{r}}(T_c, p_c)$ be the recipient's commitment value. For any fixed commitment time T_c, under Assumptions V.1 and V.2, $v^{\mathrm{r}}(T_c, p_c)$ has the following properties as a function of commitment probability p_c:*

1. *$v^{\mathrm{r}}(T_c, p_c)$ is monotonically non-decreasing in p_c.*
2. *$v^{\mathrm{r}}(T_c, p_c)$ is convex in p_c.*
3. *$v^{\mathrm{r}}(T_c, p_c)$ is piecewise linear in p_c.*

The monotonicity is due to Lemma V.1: since u^+ is more desirable to the recipient, we can show that the recipient's value of any policy, including the recipient's policy that is optimal for a specific commitment, is non-decreasing in the toggling probability p_c. To prove convexity and piecewise linearity, the key idea is to express the recipient's commitment value as the maximum over its deterministic policies. We now provide a full proof below.

Proof of Monotonicity. We fix the commitment time T_c. For any recipient policy π^{r}, let $v_t^{\pi^{\mathrm{r}}}$ be the initial state value of π^{r} when u is enabled from u^- to u^+ with probability

1 at t, and let $v^{\pi^{\mathrm{r}}}_{M^{\mathrm{r}}}$ be the initial state value of π^{r} for M^{r}. It is useful to notice that, for commitment $c = (T_c, p_c)$,

$$v^{\pi^{\mathrm{r}}}_{\widehat{M}^{\mathrm{r}}} = p_c v^{\pi^{\mathrm{r}}}_{T_c} + (1 - p_c) v^{\pi^{\mathrm{r}}}_{M^{\mathrm{r}-}} \tag{6.4}$$

where $\widehat{M}^{\mathrm{r}}$ has the minimal enablement duration influence for commitment c. In words, the initial state value can be expressed as the weighted sum of the two scenarios, with the weight determined by the commitment probability. Consider the optimal policy $\pi^*_{\widehat{M}^{\mathrm{r}}}$ for $\widehat{M}^{\mathrm{r}}$. It is guaranteed that $v^{\pi^*_{\widehat{M}^{\mathrm{r}}}}_{T_c} \geq v^{\pi^*_{\widehat{M}^{\mathrm{r}}}}_{M^{\mathrm{r}-}}$ because, intuitively, u^+ is more desirable than u^- to the recipient. We will formally prove this later. Now consider $p'_c > p_c$ and let $c' = (T_c, p'_c)$:

$$\begin{aligned} v^{\mathrm{r}}(T_c, p_c) &= p_c v^{\pi^*_{\widehat{M}^{\mathrm{r}}}}_{T_c} + (1 - p_c) v^{\pi^*_{\widehat{M}^{\mathrm{r}}}}_{M^{\mathrm{r}-}} \\ &\leq p'_c v^{\pi^*_{\widehat{M}^{\mathrm{r}}}}_{T_c} + (1 - p'_c) v^{\pi^*_{\widehat{M}^{\mathrm{r}}}}_{M^{\mathrm{r}-}} \leq v^{\mathrm{r}}(T_c, p'_c). \end{aligned}$$

Now, we finish the proof by formally showing $v^{\pi^*_{\widehat{M}^{\mathrm{r}}}}_{T_c} \geq v^{\pi^*_{\widehat{M}^{\mathrm{r}}}}_{M^{\mathrm{r}-}}$. To this end, it is useful to recall Lemma V.1 that directly follows from Assumptions V.1 and V.2, stating that the value when u is always set to u^- is no more than the value of any arbitrary M^{r}, i.e. $v^*_{M^{r-}} \leq v^*_{M^{\mathrm{r}}}$. Now we can show $v^{\pi^*_{\widehat{M}^{\mathrm{r}}}}_{T_c} \geq v^{\pi^*_{\widehat{M}^{\mathrm{r}}}}_{M^{\mathrm{r}-}}$ because, otherwise, we have

$$\begin{aligned} v^{r}(T_c, p_c) &= p_c v^{\pi^*_{\widehat{M}^{\mathrm{r}}}}_{T_c} + (1 - p_c) v^{\pi^*_{\widehat{M}^{\mathrm{r}}}}_{M^{\mathrm{r}-}} \\ &< p_c v^{\pi^*_{\widehat{M}^{\mathrm{r}}}}_{M^{\mathrm{r}-}} + (1 - p_c) v^{\pi^*_{\widehat{M}^{\mathrm{r}}}}_{M^{\mathrm{r}-}} = v^{\pi^*_{\widehat{M}^{\mathrm{r}}}}_{M^{\mathrm{r}-}} \leq v^*_{M^-} \end{aligned}$$

where $v^{r}(T_c, p_c) < v^*_{M^-}$ contradicts Lemma V.1. □

Proof of Convexity and Piecewise Linearity. Let Π^{r}_D be the set of all the recipient's deterministic policies. It is well known [Put14] that the optimal value can be attained by a deterministic policy,

$$v^{\mathrm{r}}(T_c, p_c) = \max_{\pi^{\mathrm{r}} \in \Pi^{\mathrm{r}}_D} v^{\pi^{\mathrm{r}}}_{\widehat{M}^{\mathrm{r}}} = \max_{\pi^{\mathrm{r}} \in \Pi^{\mathrm{r}}_D} p_c v^{\pi^{\mathrm{r}}}_{T_c} + (1 - p_c) v^{\pi^{\mathrm{r}}}_{M^{\mathrm{r}-}}$$

which indicates that $v^{\mathrm{r}}(T_c, p_c)$ is the maximum of a finite number of value functions that are linear in p_c. Therefore, $v^{\mathrm{r}}(T_c, p_c)$ is convex and piecewise linear in p_c. □

6.3 Centralized Formulation of Cooperative Commitments

Here we show how the structure in the recipient's and provider's value functions presented above leads to a reduced search space for optimal commitments. This, in turn, will allow for an efficient *centralized* search algorithm for optimal commitments that we will use to benchmark the decentralized algorithms we develop.

As functions of the commitment probability, the provider's commitment value is non-increasing, the recipient's commitment value is non-decreasing, and both are piecewise linear. As an immediate consequence, the joint commitment value is piecewise linear in the probability, and any local maximum for a fixed commitment time T_c can be attained by a probability at the extremes of zero and the maximum feasible probability $\overline{p}(T_c)$, or where the slope of the provider's commitment value function changes. We refer to these probabilities as the provider's *linearity breakpoints*. Therefore, without loss of optimality, one can solve the problem in Equation (6.1) to find an optimal commitment by searching only over these linearity breakpoints, as formally stated in Theorem VI.3.

Theorem VI.3. *Let $\mathcal{P}(T_c)$ be the set of probabilities that are the linearity breakpoints of the provider's commitment value function for a fixed commitment time T_c. Let $\mathcal{C} = \{(T_c, p_c) : T_c \in [H], p_c \in \mathcal{P}(T_c)\}$ be the set of commitments in which the probability is a provider's linearity breakpoint. We have*

$$\max_{c \in [H] \times [0,1]} v^{\mathrm{p+r}}(c) = \max_{c \in \mathcal{C}} v^{\mathrm{p+r}}(c).$$

Proof. This directly results from the properties in Theorems VI.1 and VI.2. □

Further, the property of convexity/concavity assures that, for any fixed commitment time, the commitment value function is linear in a probability interval $[p_l, p_u]$ if and only if the value of an intermediate commitment probability $p_m \in (p_l, p_u)$ is the linear interpolation of the two extremes. This enables us to adopt a binary search procedure to efficiently identify the provider's linearity breakpoints. For any fixed commitment time T_c, the strategy first identifies the maximum feasible probability $\overline{p}(T_c)$. Beginning with the entire interval of $[p_l, p_u] = [0, \overline{p}(T_c)]$, it recursively checks the linearity of an interval by checking the middle point, $p_m = (p_l + p_u)/2$. The recursion continues with the two halves, $[p_l, p_m]$ and $[p_m, p_u]$, only if the commitment value function is verified to be nonlinear in interval $[p_l, p_u]$. This binary search procedure is outlined in Algorithm 2, implemented using a FIFO queue. Stepping

through $T_c \in [H]$ and doing the above binary search for each will find all probability breakpoint commitments $\mathcal{C}$.

Algorithm 2: Binary search for the provider's linearity breakpoints

Input: The provider's MDP M^p, commitment time T_c.
Output: $\mathcal{P}(T_c)$: the provider's linearity breakpoints for T_c.

Compute $\overline{p}(T_c)$, the maximum feasible probability for T_c
`queue` $\leftarrow$ A FIFO queue of probability intervals
`queue.push`$([0, \overline{p}(T_c)])$
Compute and save the provider's commitment value for $p_c = 0, \overline{p}(T_c)$, i.e. $v^{\mathrm{p}}(T_c, 0)$ and $v^{\mathrm{p}}(T_c, \overline{p}(T_c))$
Initialize $\mathcal{P}(T_c) \leftarrow \{\}$
while `queue` not empty **do**
    $[p_l, p_u] \leftarrow$ `queue.pop`()
    $\widehat{\mathcal{P}} \leftarrow \widehat{\mathcal{P}} \cup \{p_i, p_j\}$
    $p_m \leftarrow (p_l + p_u)/2$; compute and save $v^{\mathrm{p}}(T_c, p_m)$
    if $v^{\mathrm{p}}(T_c, p_m)$ is the linear interpolation of $v^{\mathrm{p}}(T_c, p_l)$ and $v^{\mathrm{p}}(T_c, p_u)$ **then**
        continue
    end
    else
        `queue.push`$([p_l, p_m])$
        `queue.push`$([p_m, p_u])$
    end
end

In summary, a centralized procedure to search for the optimal commitment first constructs $\mathcal{C}$ as just described, and then computes the value of each $c \in \mathcal{C}$ for both the provider and recipient. It returns the c with the highest summed value.

6.3.1 Empirical Evaluation

The principal theoretical result in this section is that, for centralized formulation, the commitment probabilities to consider can be restricted to the linearity breakpoints of the provider's commitment value function without loss of optimality. Our empirical evaluations here aim to confirm this optimality result, and test the hypothesis that the space of breakpoints would be relatively small, allowing the search to be faster. Our evaluations are in the domain used for studying value maximizer commitments in Section 5.4.2. Recall that, for the provider, the domain is designed to introduce tension between enabling the precondition to help the recipient versus increasing its own value. For the recipient, the domain includes diverse preferences regarding

the commitments. Thus, the commitments that maximize the joint value should be carefully formulated.

6.3.1.1 Alternative Discretizations

To test the hypothesis of greater efficiency, we compare the breakpoints commitments discretization to the following alternative discretizations:

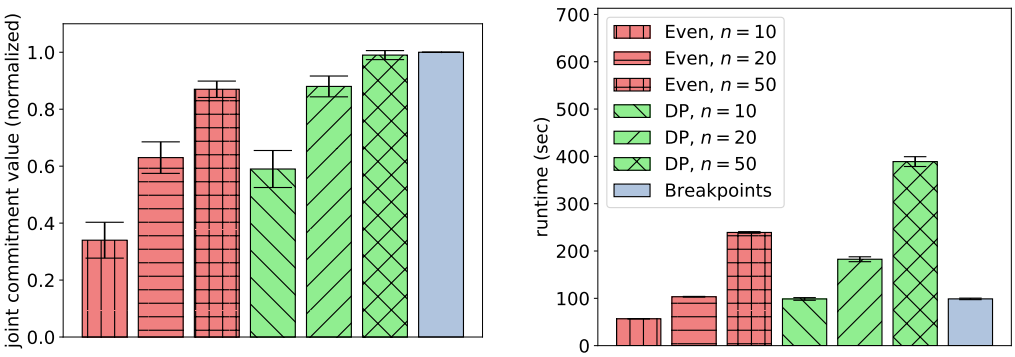
Even discretization. A simple method that discretizes the continuous probability space $[0, 1]$ evenly as $\{p_0, p_1, ..., p_n\}$ with $p_i = \frac{i}{n}$, where n is referred to as the resolution of the discretization. The larger resolution n is, the more commitments are included in the discretization and the better commitment found will be. At the same time, the larger resolution n is, the larger the computational cost of evaluating every commitment in the discretization.

Deterministic Policy (DP) discretization. We also consider another discretization, adopted in prior work [WD07, WD10], that finds all of the probabilities of toggling feature u at the commitment time that can be attained by the provider following a deterministic policy.

For the even discretization, we consider the resolutions $n \in \{10, 20, 50\}$. For the DP discretization, we found that the number of toggling probabilities of all the provider's deterministic policies is large, and the corresponding computational cost of identifying and evaluating them is high. To reduce the computational cost and for fair comparison, we group the probabilities in the DP discretization that are within $\frac{i}{n}$ of each other for $n \in \{10, 20, 50\}$.

6.3.1.2 Results

Figure 6.1 shows the joint values of the best commitments for the seven discretizations, along with runtimes for forming and evaluating the discretizations, where the provider does not have action a^+ to directly transit to the absorbing state that realizes the commitment. We report the mean and standard error over 50 randomly-generated pairs of the provider's MDP and the recipient's MDP. Since the problem instances have different reward scales, for each instance we normalize the joint value with the value of the best commitment for the breakpoints discretization, and the lowest value among the seven discretizations, such that the joint value for the breakpoints discretization normalizes to 1, and the lowest joint value among the seven discretizations normalizes to 0.



(a) Joint Commitment Value

(b) Runtime (sec.)

Figure 6.1: Centralized commitment formulation comparing the even, the DP, and the breakpoints discretizations. We report means and standard errors over 50 problem instances, each being a provider-recipient pair randomly generated as described in Section 5.4.2. Figure 6.1a shows the optimal joint commitment value for each discretization. Figure 6.1b shows the runtime for forming the discretization and evaluating the commitments in the discretization.

Figure 6.1 confirms that our breakpoints discretization yields the highest joint commitment value in a computationally efficient manner. In Figures 6.1a, we see that, for the even and the DP discretizations, the joint commitment value increases with the probability resolution n, and only once we reach $n = 50$ is the joint commitment value comparable to our breakpoints discretization. Figure 6.1b compares the runtimes of forming the discretization and evaluating the commitments in the discretization, confirming our hypothesis that using the breakpoints discretization is more computationally efficient than the even and the DP discretizations with a probability resolution that yields comparable (and no higher) joint commitment values. Table 6.1 compares the sizes of these discretizations, and confirms our intuition that the breakpoints discretization is the most computationally efficient because it identifies a smaller number of commitments that are sufficient for the joint value maximization.

Figure 6.2 visualizes the commitment value functions and their linearity breakpoints for a randomly-chosen problem instance for commitment time $T_c = 5$, 10, and 15. The visualizations confirm the structural properties of monotonicity, concavity/convexity, and piecewise linearity. We observe that although the maximum feasible probability $\overline{p}(T_c)$ unsurprisingly increases with the commitment time T_c, the number of the provider's linearity breakpoints does not increase proportionally to $\overline{p}(T_c)$. This confirms that the breakpoints discretization can be relatively small even

Table 6.1: Averaged discretization size per commitment time. The results are means and standard errors over the same 50 problem instances as in Figure 6.1.

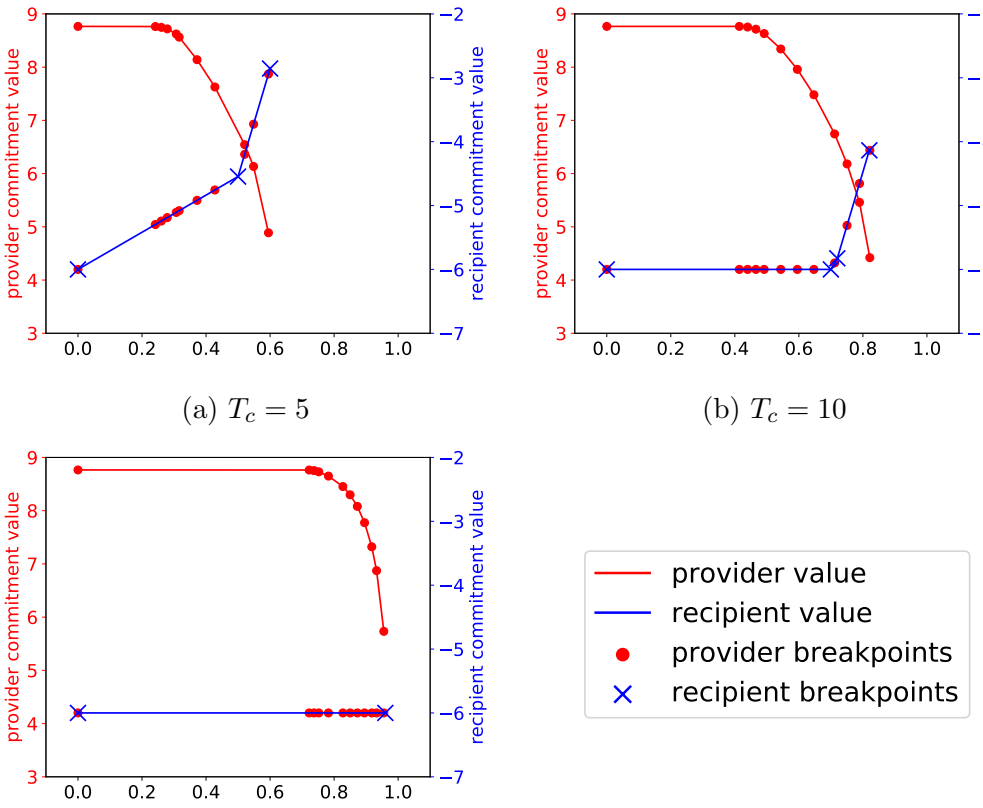
	$n = 10$	$n = 20$	$n = 50$
Even	7.8 ± 0.1	15.1 ± 0.1	37.1 ± 0.1
DP	6.1 ± 0.2	12.0 ± 0.3	26.5 ± 0.7
Breakpoints	10.0 ± 0.1		

though the feasible commitment probability space can be large. For this problem instance, the number of the recipient's linearity breakpoints is significantly smaller than that of the provider, indicating that the binary search procedure is more efficient that enumeration when evaluating the provider's breakpoints on the recipient's commitment value function.

Effect of the Commitment Space Size. For small values of commitment time T_c, the maximum feasible probability $\overline{p}(T_c)$ is fairly low without the provider's action a^+, and so is the size of the feasible commitment space. Here, we further compare the discretizations when the feasible commitment space is increased by the introduction of the provider's action a^+. Recall that we use p^+ to denote the mean of the Gaussian distribution associated with action a^+. For $p^+ \in \{0, 0.5\}$, Figure 6.3 shows the the maximum feasible probability $\overline{p}(T_c)$. For each p^+, we report the mean and standard error over 50 randomly generated pairs of the provider's MDP and the recipient's MDP. As shown in Figure 6.3, the maximum feasible probability dramatically increases with p^+ increased from 0 to 0.5. We hypothesize that, as feasibility increases, the runtime of the even discretization also increases because its size is proportional to the maximum feasible probability. We are interested in how the runtime of our breakpoints discretization changes with feasibility.

Figure 6.4 shows the joint values of the best commitments for the seven discretizations and the runtimes. The results confirm that, for both values of p^+, our breakpoints discretization yields the highest joint commitment value with the smallest runtime, which is consistent with Figure 6.1 that evaluates the discretizations without the provider's action a^+. The results also confirm our hypothesis on the runtime of the even discretization: comparing Figures 6.4b and 6.4d, it is noticeable that, for $n = 50$, the even discretization's runtime increases with p^+. The breakpoints discretization's runtime does not change with p^+. Perhaps surprisingly, DP's runtime for $p^+ = 0$ is much larger than that for $p^+ = 0.5$.

To have a detailed analysis on the runtime, we plot the runtimes for the even



(a) $T_c = 5$

(b) $T_c = 10$

(c) $T_c = 20$

Figure 6.2: Visualizations of commitment value functions and their linearity breakpoints for a randomly chosen problem instance. X-axis shows the commitment probability, and Y-axis shows the commitment value.

($n = 50$), DP ($n = 50$), and the breakpoints in Figure 6.5a, where DP's runtime is decomposed into the runtime for forming the discretization and the runtime for evaluating the commitments in the discretization. We see that the major reason why DP's runtime for $p^+ = 0$ is larger is because forming the discretization takes more time. For these six discretizations, Figure 6.5b shows their density defined as the number of commitments per commitment time normalized by the maximum feasibility commitment and averaged over the commitment time, i.e.

$$\text{Discretization Density} = \frac{1}{H}\sum_{T_c=1}^{H}\frac{\#\text{commitments for } T_c}{\overline{p}(T_c)} \tag{6.5}$$

As confirmed in Figure 6.5b, the density of the even discretization is determined by n, and therefore its size and runtime increases with the maximum feasible commitment determined by p^+. The density of the DP discretization is not only dependent on

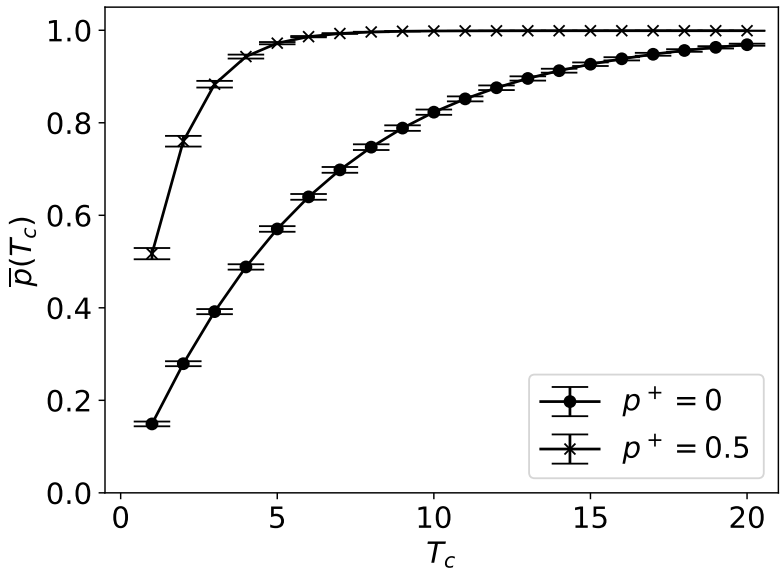
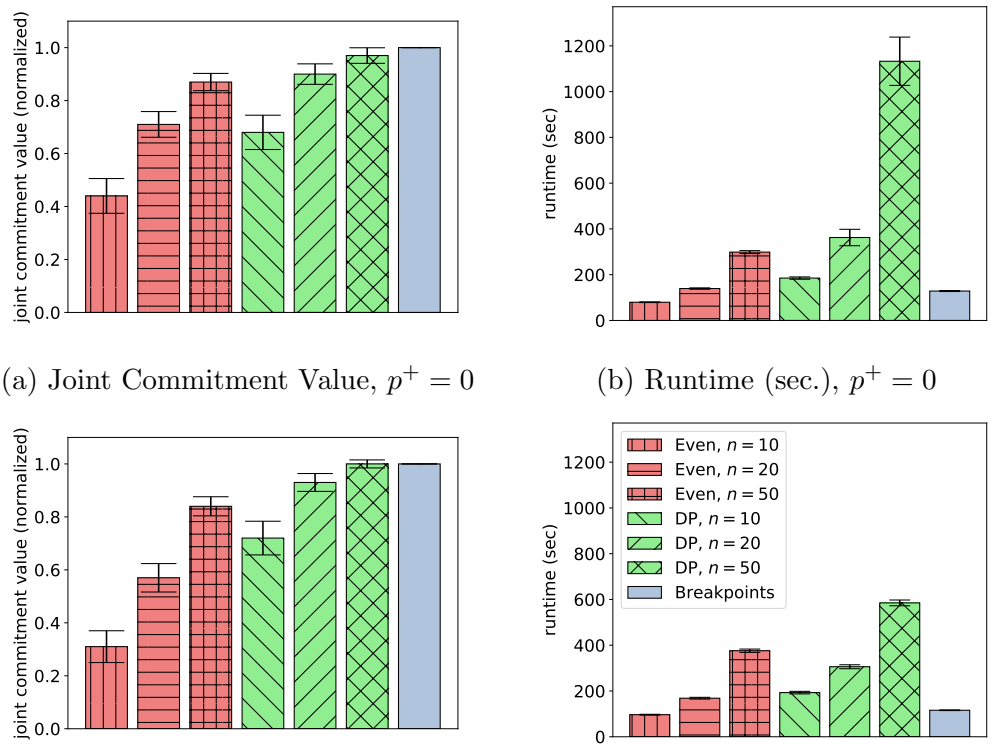


Figure 6.3: Maximum feasible commitment probability with the provider's action a^+. X-axis shows the commitment time, and Y-axis shows the maximum feasible commitment probability for a given commitment time. The results are means and standard errors over 50 randomly generated MDPs for the provider.

n but also p^+ that affects the structure of the state space and transition function. Actually, as shown in Figure 6.5b, the DP's density for $p^+ = 0$ is larger than that for $p^+ = 0.5$. Similarly, the density of our breakpoints discretization for $p^+ = 0$ is also larger, which explains why the runtime does not increase with feasibility.

6.4 Querying Approach for Decentralized Formulation of Cooperative Commitments

We now progress to the decentralized optimization setting where the agents try to find a commitment that maximizes their joint value, even though neither agent has full knowledge about the other's environment. Recall from Section 3.3 that we aim to develop a querying approach for eliciting the jointly-preferred commitment based on exchanged knowledge about feasible commitment options and their local values to an agent. In such a querying approach, the provider poses a *commitment query* consisting of information about a set of feasible commitments, and the recipient responds by selecting the commitment from the set that will best satisfy their joint preferences. The number of commitments in the query is often small to limit costs for communication and computation.



(a) Joint Commitment Value, $p^+ = 0$

(b) Runtime (sec.), $p^+ = 0$

(c) Joint Commitment Value, $p^+ = 0.5$

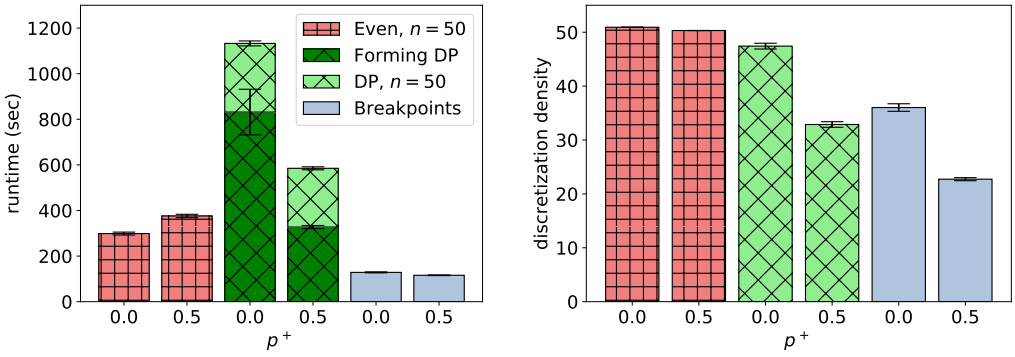
(d) Runtime (sec.), $p^+ = 0.5$

Figure 6.4: Centralized commitment formulation with the provider's action a^+. After taking action a^+, p^+ is the probability of transiting to the absorbing state (and hence realizing the commitment), which is sampled for each state from a Gaussian with standard deviation of 0.1 and then clipped into $[0, 1]$. The mean of the Gaussian distribution is chosen to be 0 (top) and 0.5 (bottom). The results are means and standard errors of the optimal joint commitment value (left) and the runtime (right) over 50 randomly generated problem instances, each being a provider-recipient pair.

Specifically, we consider a setting where the provider fully knows its MDP M^{p}, and where its uncertainty about the recipient's MDP is modeled as a probability distribution μ over a finite set of N candidate MDPs. Given uncertainty μ, the Expected Utility (EU) of a feasible commitment c is defined as the expected joint value of the commitment under μ:

$$EU(c; \mu) = E_{\mu}\left[v^{\mathrm{p+r}}(c)\right], \tag{6.6}$$

where the expectation is with respect to the uncertainty about the recipient's MDP. If the provider had to single-handedly select a commitment based on its uncertainty μ about the recipient, the best commitment is the one that maximizes the expected



(a) Runtime(sec)

(b) Discretization Density

Figure 6.5: Profiled runtime and discretization density. The left plot decomposes the runtime for DP ($n = 50$) in terms of forming the discretization (dark green, bottom bar) and evaluating the commitments in the discretization (light green, top bar). The right plot shows the discretization density (defined in Equation (6.5)) for even ($n = 50$), DP ($n = 50$), and breakpoints. The results are means and standard errors over the same 50 problem instances as in Figure 6.4.

utility:

$$c^*(\mu) = \arg\max_c EU(c;\mu), \text{ with } EU^*(\mu) = \max_c EU(c;\mu). \tag{6.7}$$

But through querying, the provider is given a chance to refine its knowledge about the recipient's actual MDP. Formally, the provider's commitment query $\mathcal{Q}$ consists of a (small) finite number of feasible commitments. The provider offers these choices to the recipient, where the provider also annotates each choice with its expected local value of its optimal policy respecting the commitment (Equation (3.2)). The recipient computes (using Equation (3.5)) its own expected value for each commitment offered in the query, and adds that to the annotated value from the provider. It then responds to the provider with the commitment that maximizes the summed value (with ties broken by selecting the smallest indexed) to be the commitment the two agents agree on. This querying approach is illustrated in Figure 6.6.

More formally, let $\mathcal{Q} \rightsquigarrow c$ denote the recipient's response that selects $c \in \mathcal{Q}$. With the provider's prior uncertainty μ, the posterior distribution given the response is denoted as $\mu \mid \mathcal{Q} \rightsquigarrow c$, which can be computed by Bayes' rule. To avoid large communication cost, the number of commitments in the query, $k = |\mathcal{Q}|$, is small, such that the response usually cannot fully resolve the provider's uncertainty. In that case, the value of a query $\mathcal{Q}$ is the EU with respect to the posterior distribution averaged

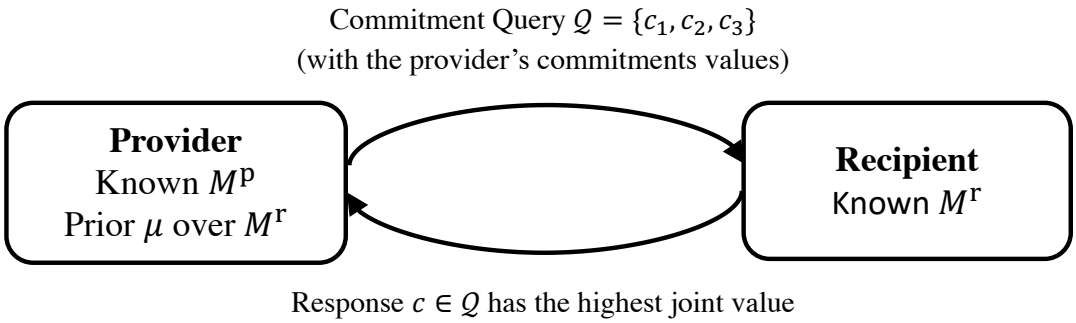


Figure 6.6: Illustration of the querying approach. Both the provider and the recipient fully know their own MDPs, M^{p} and M^{r}, respectively. The provider's uncertainty about the recipient's MDP is modeled as a prior distribution μ. The provider poses a commitment query $\mathcal{Q}$ consists of three commitments, along with their values for the provider. The recipient's response $c \in \mathcal{Q}$ has the highest joint value among the three commitments in the query.

over all the commitments in the query being a possible response, and, consistent with prior work [VB10, ZDS17], we refer to it as the query's Expected Utility of Selection (EUS):

$$EUS(\mathcal{Q}; \mu) = E_{\mathcal{Q} \rightsquigarrow c; \mu} \left[EU(c; \mu \mid \mathcal{Q} \rightsquigarrow c) \right].$$

Here, the expectation is with respect to the recipient's response under μ. The provider's *querying problem* thus is to formulate a query $\mathcal{Q} \subseteq [H] \times [0, 1]$ consisting of $|\mathcal{Q}| = k$ feasible commitments that maximizes EUS:

$$\max_{\mathcal{Q} \subseteq [H] \times [0,1], |\mathcal{Q}| = k} EUS(\mathcal{Q}; \mu). \tag{6.8}$$

Importantly, we can show that $EUS(\mathcal{Q}; \mu)$ is a submodular function of $\mathcal{Q}$, as formally stated in Theorem VI.4. Submodularity serves as the basis for a greedy optimization algorithm [NWF78], which we will discuss in detail in Section 6.5.

Theorem VI.4. *For any uncertainty μ, $EUS(\mathcal{Q}; \mu)$ is a submodular function of $\mathcal{Q}$. That is, given two queries $\mathcal{Q} \subseteq \mathcal{Q}'$, commitment $c \notin \mathcal{Q}$, we have:*

$$EUS(\mathcal{Q} \cup \{c\}; \mu) - EUS(\mathcal{Q}; \mu) \geq EUS(\mathcal{Q}' \cup \{c\}; \mu) - EUS(\mathcal{Q}'; \mu)$$

Proof. Since the recipient always chooses the commitment that maximizes the joint value over all commitments in the query, this reduces to the scenario referred to as the noiseless response model in prior work on EUS maximization [VB10]. The prior

work [VB10] proves the submodularity under the noiseless response model, which also proves Theorem VI.4. □

Submodularity means that adding a commitment to the query can increase the EUS, but the increase is diminishing with the size of the query. An upper bound on the EUS of any query of any size k can be obtained when $k \geq N$ such that the query can include the optimal commitment of each candidate recipient's MDP, i.e.

$$\overline{EUS} = E_\mu \left[\max_{c \in [H] \times [0,1]} v^{\mathrm{p+r}}(c) \right]. \tag{6.9}$$

Upper bound $\overline{EUS}$ can be computed with the centralized algorithm we described in Section 6.3.

6.4.1 Structure of the Commitment Query Space

Due to the properties of individual commitment value functions proved in Section 6.2, the expected utility $EU(c; \mu)$ defined in Equation (6.6), as calculated by the provider alone, becomes a summation of the non-increasing provider's commitment value function and the (provider-computed) weighted average of the non-decreasing recipient's commitment value functions. With the same reasoning as for Theorem VI.3, the optimality of the linearity breakpoint commitments can be generalized to any uncertainty. That is, for any uncertainty μ, the commitment probability of an expected utility maximizing commitment $c^*(\mu)$ is a linearity breakpoint of the provider's commitment value function, as formalized in Lemma VI.1.

Lemma VI.1. *Let $\mathcal{C}$ be defined in the same manner as in Theorem VI.3. We have*

$$\max_{c \in [H] \times [0,1]} EU(c; \mu) = \max_{c \in \mathcal{C}} EU(c; \mu).$$

Proof. This directly results from the properties in Theorems VI.1 and VI.2. □

As a consequence of Lemma VI.1, for EUS maximization, there is no loss in only considering the provider's linearity breakpoints, as formally stated in Theorem VI.5.

Theorem VI.5. *Let $\mathcal{C}$ be defined in the same manner as in Theorem VI.3. For any query size k and uncertainty μ, we have*

$$\max_{\mathcal{Q} \subseteq [H] \times [0,1], |\mathcal{Q}| = k} EUS(\mathcal{Q}; \mu) = \max_{\mathcal{Q} \subseteq \mathcal{C}, |\mathcal{Q}| = k} EUS(\mathcal{Q}; \mu).$$

Proof. We first give Lemma VI.2 that says any discretization that contains the linearity breakpoints is no worse than any other discretization.

Lemma VI.2. *Let $\mathcal{C}$ be defined in the same manner as in Theorem VI.5. Consider any finite set of commitments $\overline{\mathcal{C}}$ that contains $\mathcal{C}$, i.e. $\overline{\mathcal{C}} \supseteq \mathcal{C}$. For any query size k and any uncertainty μ,*

$$\max_{Q \subseteq \overline{\mathcal{C}}, |Q|=k} EUS(\mathcal{Q}; \mu) = \max_{Q \subseteq \mathcal{C}, |Q|=k} EUS(\mathcal{Q}; \mu). \tag{6.10}$$

Proof of Lemma 6.10. Because $\overline{\mathcal{C}} \supseteq \mathcal{C}$, it is obvious that "$\geq$" holds for Equation (6.10). We next show "$\leq$".

Given a commitment query $\mathcal{Q} = \{c_1, ..., c_k\}$, define $T(\mathcal{Q})$ as a commitment query where each commitment is the optimal commitment with respect to the posterior given a response for Q, i.e.

$$T(\mathcal{Q}) = \{c^*(\mu \mid \mathcal{Q} \rightsquigarrow c_1), ..., c^*(\mu \mid \mathcal{Q} \rightsquigarrow c_k)\}.$$

Previous work [VB10] shows that $EUS(T(\mathcal{Q}); \mu) \geq EUS(\mathcal{Q}; \mu)$. Due to Lemma VI.1, we now have $c^*(\mu) \in \mathcal{C}$ for any uncertainty μ. Thus, given an EUS maximizer $\mathcal{Q}^*$ for $\overline{\mathcal{C}}$, $T(\mathcal{Q}^*)$ is a subset of $\mathcal{C}$ with an EUS that is no smaller, which shows "$\leq$" holds for Equation (6.10). □

We are ready to prove Theorem VI.5. Consider the even discretization of $[0, 1]$, $\mathcal{P}_n = \{p_0, p_1, ..., p_n\}$ where $p_i = \frac{i}{n}$. Because v^{p+r} is bounded and piecewise linear in the commitment probability, for any $\epsilon > 0$, there exists a large enough discretization resolution n, such that for any size k query $\mathcal{Q} \subseteq [H] \times [0, 1]$, there is a size k query $\widehat{\mathcal{Q}} \in [H] \times \mathcal{P}_n$ that $|EUS(\mathcal{Q}; \mu) - EUS(\widehat{\mathcal{Q}}; \mu)| \leq \epsilon$. Therefore, we have

$$\begin{aligned} EUS(\mathcal{Q}; \mu) - \epsilon \leq \max_{\widehat{\mathcal{Q}} \subseteq [H] \times \mathcal{P}_n, |\widehat{\mathcal{Q}}|=k} EUS(\widehat{\mathcal{Q}}; \mu) &\leq \max_{\mathcal{Q} \subseteq (\mathcal{C} \cup [H] \times \mathcal{P}_n), |\mathcal{Q}|=k} EUS(\mathcal{Q}; \mu) \\ &= \max_{\mathcal{Q} \subseteq \mathcal{C}, |\mathcal{Q}|=k} EUS(\mathcal{Q}; \mu) \end{aligned}$$

for any query $\mathcal{Q} \subseteq [H] \times [0, 1]$ with $|\mathcal{Q}| = k$, where the equality is a direct result from Lemma VI.2. This concludes the proof. □

6.5 Efficient Commitment Query Formulation

Theorem VI.5 allows us to develop an efficient procedure for solving the query formulation problem (Equation (6.8)). The provider first identifies its linearity breakpoints commitments $\mathcal{C}$ and evaluates them for its MDP and each of the recipient's possible MDPs. Due to the concavity and convexity properties, commitments $\mathcal{C}$ can be identified and evaluated efficiently with the binary search procedure outlined in Algorithm 2. Finally, a size k query is formulated from commitments $\mathcal{C}$ that solves the EUS maximization problem either exactly with exhaustive search or approximately with greedy search:

Exhaustive query search. The finite EUS maximization problem can be exactly solved by exhaustively forming and evaluating each k-subset of breakpoint commitments, and selecting the best one.

Greedy query search. The finite EUS maximization problem can be approximately solved by a greedy procedure [VB10, CSD14] that iteratively grows the query by adding the breakpoint commitment that contributes maximum EUS. Formally, beginning with $\mathcal{Q}_0$ as an empty set, the algorithm iteratively performs $\mathcal{Q}_i \leftarrow \mathcal{Q}_{i-1} \cup \{c_i\}$ for $i = 1, ..., k$, where

$$c_i = \arg\max_{c \in \mathcal{C} \setminus \mathcal{Q}_{i-1}} EUS(\mathcal{Q}_{i-1} \cup \{c\}; \mu).$$

Since EUS is a submodular function of the query (Theorem VI.4), the greedily formed size k query $\mathcal{Q}_k$ is within a factor of $1 - (\frac{k-1}{k})^k$ of the EUS of the optimal query of size k [NWF78].

The overall procedure of formulating the greedy query is given in Algorithm 3.

6.5.1 Empirical Evaluation

Our empirical evaluations aim to answer the following questions regarding the decentralized commitment query formulation procedure, with the evaluations conducted on the same domain as in Section 6.3:

- For commitment query formulation, how effective and computationally more efficient is the breakpoints discretization compared with alternative discretization methods?

Algorithm 3: Greedy query formulation from the provider's linearity breakpoints commitments

Input: The provider's MDP M^p with horizon H
The provider's uncertainty μ over the recipient's MDP
The query size k.
Output: The greedy query of size k.

The provider's linearity breakpoints commitments $\mathcal{C} \leftarrow \{\}$
for commitment time $T_c = 1, 2, ..., H$ **do**
  Use Algorithm 2 to compute $\mathcal{P}(T_c)$, i.e. the provider's linearity breakpoints for T_c
  $\mathcal{C} \leftarrow \mathcal{C} \cup \{(T_c, p_c) : p_c \in \mathcal{P}(T_c)\}$
end
`// Formulate the greedy query from` $\mathcal{C}$
$\mathcal{Q}_0 \leftarrow \{\}$
for $i = 1, 2, ..., k$ **do**
  $EUS_{\max} \leftarrow -\infty$
  for $c \in \mathcal{C} \setminus \mathcal{Q}_{i-1}$ **do**
    $EUS_{\text{temp}} \leftarrow EUS(\mathcal{Q}_{i-1} \cup \{c\}; \mu)$
    if $EUS_{\text{temp}} > EUS_{\max}$ **then**
      $EUS_{\max} \leftarrow EUS_{\text{temp}}$
      $c_{\max} \leftarrow c$
    end
  end
  $\mathcal{Q}_i \leftarrow \mathcal{Q}_{i-1} \cup \{c_{\max}\}$
end
Return $\mathcal{Q}_k$

- How effective and computationally more efficient is greedy query search compared with exhaustive query search?

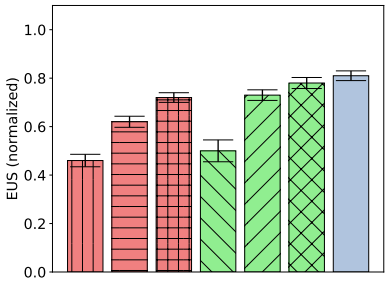
6.5.1.1 Evaluating the Breakpoints Discretization

The results in Figure 6.7 give the EUS for the seven discretizations, for $N = 10, 50$ as the number of the recipient's candidate MDPs and $k = 2, 5$ as the query size, along with the runtimes for forming the discretizations and evaluating the commitments in the discretizations for the provider's MDP and all the recipient's candidate MDPs. We report the mean and standard error over 50 randomly generated problem instances, each of which is generated by randomly sampling an MDP for the provider, and 10 candidate MDPs for the recipient, setting the provider's prior uncertainty μ over the recipient's MDP to be the uniform distribution over the 10 candidates. Since the problem instances have different reward scales (r_{left}), for each instance we normalize

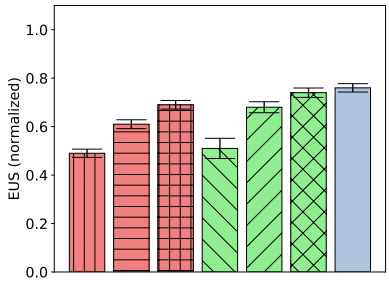
the EUS with the upper bound $\overline{EUS}$ defined in Equation (6.9) and the EUS of the optimal and greedy query of the even discretization for $k = 1, n = 10$, which we denote as $\underline{EUS}$. That is, a value for EUS is normalized as $(EUS - \underline{EUS})/(\overline{EUS} - \underline{EUS})$, such that the EUS of the query consisting of the optimal commitments for the recipient's candidate MDPs normalizes to 1, and the EUS consisting of the optimal commitment with respect to μ in the $n = 10$ even discretization normalizes to 0.

The results in Figure 6.7 show that, coupled with the greedy query algorithm (evaluated next in Section 6.5.1.2), our breakpoints commitments discretization yields the highest EUS in a computationally efficient manner. Figures 6.7a–6.7d compare the EUS for the seven discretizations. The EUS for the even and the DP discretizations increases with the probability resolution n as expected, and only once we reach $n = 50$ is the EUS comparable to our breakpoints discretization. In the comparison between the even and the DP discretizations, we see that, for the same n, the EUS of the DP discretization is consistently higher than that of the even discretization. This indicates that the inductive bias of using the provider's deterministic policies improves the greedy query's EUS. Recall that we use the same normalization constant for the EUS results for $k = 2$ and 5, and thus the results in Figures 6.7a–6.7d show that including more commitments in the query significantly improves the query's EUS. Specifically, for both $N = 10$ and $N = 50$, the normalized EUS for the breakpoints discretization is nearly one when including $k = 5$, a relatively small number compared with N, commitments in the greedy query. This demonstrates the effectiveness of the greedy query for EUS maximization, and we will evaluate the greedy query more thoroughly in Section 6.5.1.2.

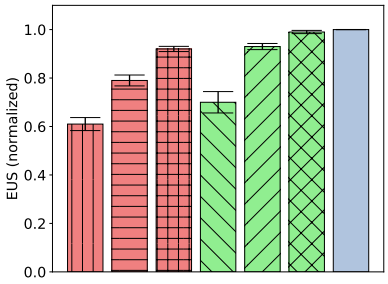
Figures 6.7e and 6.7f compare the runtimes of forming the discretization and evaluating the commitments in the discretization for the downstream query formulation procedure, showing that using breakpoints is significantly faster. The runtime for the downstream greedy query formulation is not included in Figure 6.7, but in Figure 6.8b for the breakpoints discretization. We observe from Figure 6.8b that the runtime for greedy query formulation is only a tiny fraction of that for forming and evaluating the discretization (shown in Figures 6.7e and 6.7f). This implies that, to efficiently perform the EUS maximization with the greedy query, it is crucial to first efficiently form and evaluate the discretization, which is exactly what is achieved by our breakpoints discretization.



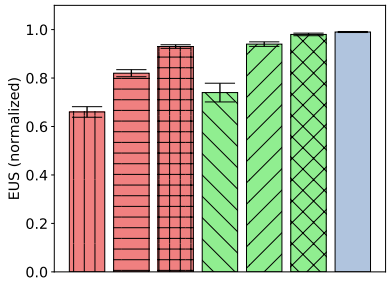
(a) EUS (normalized), $N = 10$, $k = 2$



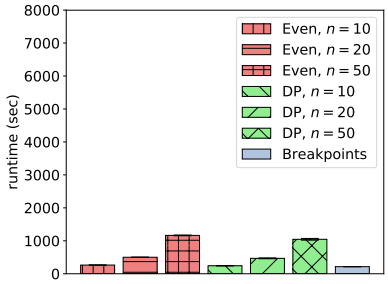
(b) EUS (normalized), $N = 50$, $k = 2$



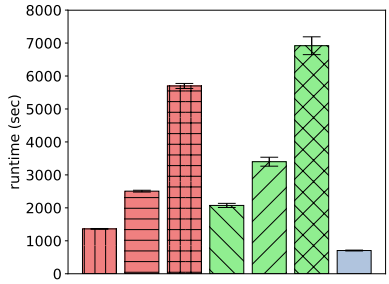
(c) EUS (normalized), $N = 10$, $k = 5$



(d) EUS (normalized), $N = 50$, $k = 5$



(e) Runtime (sec.), $N = 10$



(f) Runtime (sec.), $N = 50$

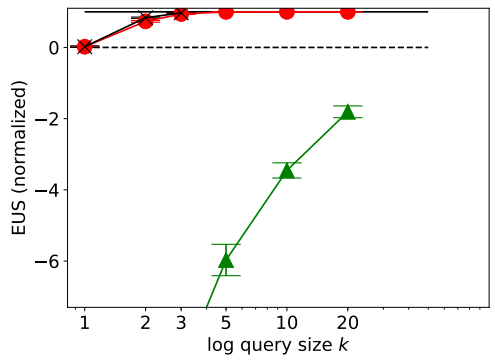
Figure 6.7: Decentralized commitment formulation comparing the even, the DP, and the breakpoints discretizations. We report means and standard errors of the EUS of the greedy query (Figures 6.7a–6.7d) and the runtime (Figures 6.7e and 6.7f) over 50 problem instances, each consisting of one provider MDP with a uniform prior over N recipient MDPs randomly generated as described in Section 5.4.2. The runtime is for forming the discretization and evaluating the commitments in the discretization. Figure 6.8b shows the runtime for forming the greedy query, which is only a tiny fraction of the runtimes shown in Figures 6.7e and 6.7f. The results are for $N = 10$ (left) and $N = 50$ (right).

6.5.1.2 Evaluating the Greedy Query

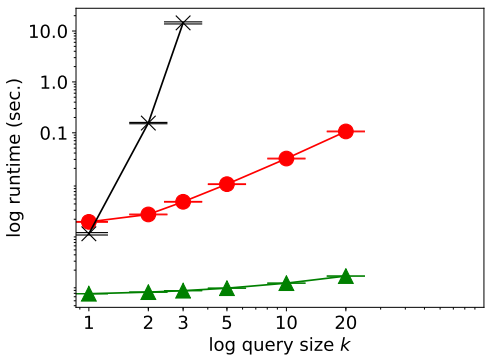
Next, we empirically confirm that greedy query search is effective for the commitment query EUS maximization. Given the results from Section 6.5.1.1, the query searches here are over the breakpoint commitments. Specifically, we show that, given the breakpoint commitments, formulating the commitment query greedily yields EUS that is comparable to the optimal, and is computationally much more efficient than exhaustively searching for the optimal query.

Figure 6.8c compares the EUS of the greedy query with the optimal (exhaustive search) query, and with a query comprised of randomly-chosen breakpoints. For the optimal query, we only show query size $k = 1$, 2, and 3, because we find that the exhaustive search is extremely time-consuming for $k > 3$. The EUS is normalized with $\overline{EUS}$ and the optimal EU prior to querying given uncertainty μ as defined in Equation (6.7). That is, a value for EUS is normalized as $(EUS - EU^*(\mu))/(\overline{EUS} - EU^*(\mu))$. (Note that $EU^*(\mu)$ is also the EUS of the optimal and greedy query when $k = 1$, since the recipient is only given one choice, which is the one optimizing the provider's model.) We vary the query size k, and report means and standard errors over the same 50 coordination problems as in Section 6.5.1.1. *We see that, notably, the EUS of the greedy query is very close to that of the optimal query.* Besides, unsurprisingly, for all three query formulation methods the EUS increases with the query size k; the random query's EUS after normalization is largely negative up to query size $k = 20$ as the EUS is normalized to 0 for optimal and greedy with $k = 1$, and therefore both optimal and greedy have significantly higher EUS than random. Figure 6.8b compares the runtimes of the three query formulation methods (excluding the runtime they all share for identifying the breakpoint commitments). Optimal relies on enumeration of the exponential space, so its runtime scales poorly with the query size k. In comparison, greedy scales much better and incurs moderate computational cost. These results confirm our hypothesis that greedy query search is a computationally effective method for formulating commitment queries that very nearly maximize EUS.

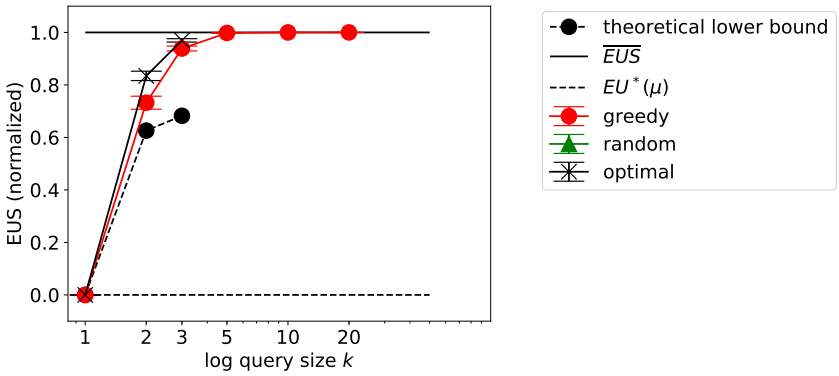
Robustness to diverse priors. Figure 6.8 has demonstrated the effectiveness of the greedy query for a particular type of the provider's prior μ, which is the uniform distribution over the recipient's $N = 10$ candidate MDP. Here, we further show that the greedy query's effectiveness is robust to diverse prior types. Besides the uniform prior, we consider two other prior types. For the random prior, the probability for each candidate recipient's MDP is proportional to a number that is randomly sampled from interval $[0, 1]$. For the Gaussian prior, the probability for each candidate



(a) EUS (normalized)



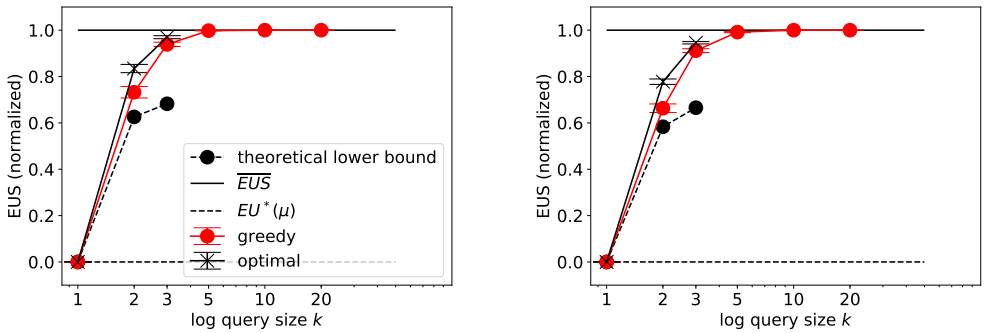
(b) Runtime (sec.)



(c) EUS (normalized) with the lower bound for greedy

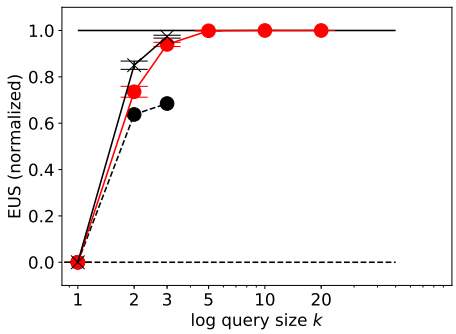
Figure 6.8: Comparison of the optimal, the greedy, and the random commitment queries. The commitment queries formed from the breakpoints discretization. The results are means and standard errors of the normalized EUS and the runtime over the same 50 problem instances with $N = 10$ candidate recipient's MDPs as in Figure 6.7. Figure 6.8b is a zoom-in version of Figure 6.8a comparing the the optimal and the greedy, also showing the theoretical lower bound of the greedy query's EUS.

recipient's MDP is proportional to the standard Gaussian distribution's probability density function evaluated at a number randomly sampled from the three-sigma interval $[-3, 3]$. Figure 6.9 shows the EUS, normalized in the same manner as Figure 6.8c, of the greedy query for the three prior types, with the number of candidate recipient's MDPs $N = 10$, and 50. For comparison, Figure 6.9 shows, for query size $k = 1, 2, 3$, the EUS of the optimal query and the greedy query's theoretical lower bound $(1 - (\frac{k-1}{k})^k$ of the EUS of the optimal query of size $k)$.

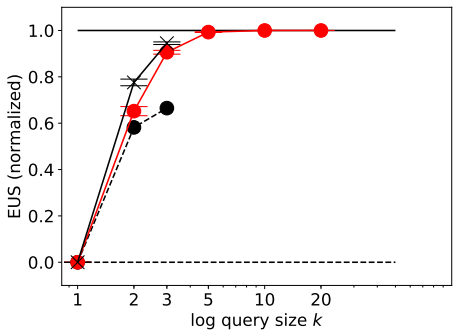


(a) Uniform Prior, $N = 10$

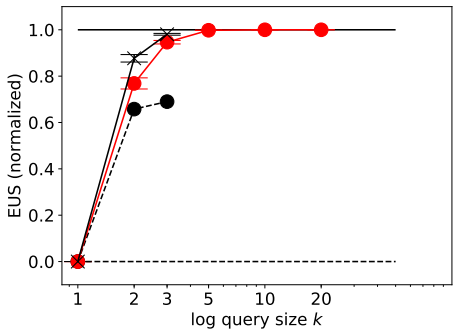
(b) Uniform Prior, $N = 50$



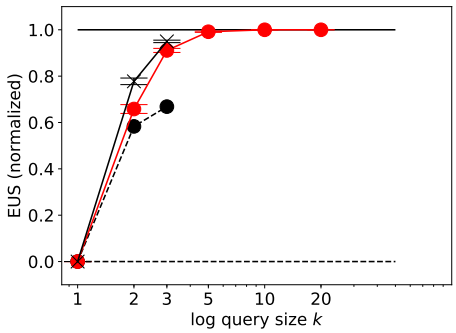
(c) Random Prior, $N = 10$



(d) Random Prior, $N = 50$



(e) Gaussian Prior, $N = 10$



(f) Gaussian Prior, $N = 50$

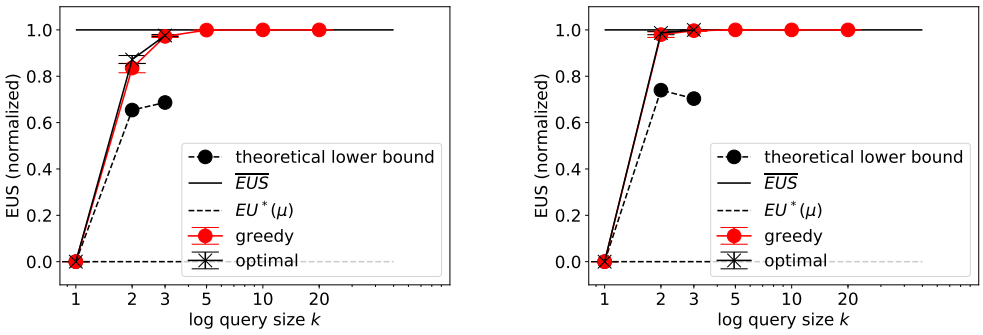
Figure 6.9: EUS of the greedy query for the uniform (top), random (middle), and Gaussian (bottom) priors. The queries are formed from the breakpoints discretization. The results are means and standard errors of the EUS over 50 problem instances, each consisting of one provider MDP and N recipient MDPs randomly generated as described in Section 5.4.2. The results are for $N = 10$ (left) and $N = 50$ (right).

We see that, as query size k increases, the greedy query's EUS also increases and quickly surpasses its theoretical lower bound. Moreover, besides the trivial case of $k = 1$, the discrepancy between the greedy query's EUS and the optimal query's EUS decreases as k increases. The greedy query's EUS reaches the upper bound $\overline{EUS}$ with query size k smaller than N, at $k = 5$ for $N = 10$ and $k = 10$ for $N = 50$, respectively. For comparison, the optimal query's EUS does not reach $\overline{EUS}$ at $k = 3$, indicating that the greedy method is an effective procedure for EUS maximization. Notably, these qualitative claims hold for both $N = 10$ and $N = 50$, and for all three types of priors. The results verify that the greedy query is effective for diverse prior types and is able to scale to a large number of candidates in the prior.

Robustness to multi-round querying. Besides priors that are synthetically generated, we here also explore priors that naturally emerge in a two-round querying process. Specifically, the provider's initial prior μ_0 is a random prior over N candidate recipient's MDPs generated as described above. The provider forms the first greedy query of size k_0, updates its prior to μ_1 based on the recipient's response, and then forms the second greedy query of size k for prior μ_1. We are interested in the quality of the second greedy query for the updated prior μ_1, which emerges from the first round of querying. Figure 6.10 shows the results for $N = 50$, $k_0 = 2$ and 5, comparing the greedy query with its theoretical lower bound and the optimal query. Consistent with the results in Figure 6.9, the results in Figure 6.10 show that the greedy query is effective for the priors that emerge from the first round of querying.

6.6 Summary

In this chapter, we focused on the problem of formulating cooperative probabilistic commitments, formally defined in Section 6.1. In Section 6.2, we proved several structural properties of the commitment value functions, which can be exploited to efficiently compute a discretization of the continuous commitment space that is guaranteed to contain the joint value maximizer. In Section 6.3, we studied the setting where there exists a centralized coordinator that has precise knowledge about both agents' MDPs and thus can directly exploit the discretization to efficiently search for the optimal commitment. Our empirical evaluations in Section 6.3.1 demonstrated such efficiency. In Section 6.4, we studied the decentralized setting where the coordinator does not exist and neither agent has precise knowledge about the other. We formulated the agents' partial knowledge using a Bayesian prior, for which we designed



(a) $k_0 = 2$ (b) $k_0 = 5$

Figure 6.10: EUS of the greedy query in the second round of querying. For the first round, the prior is the random prior over $N = 50$ candidate recipient's MDPs, and the provider forms the first greedy query of size k_0 and updates its prior based on the recipient's response. For the second round, the provider constructs the second query of size k (X-axis) for the updated prior, and the corresponding normalized EUS is shown along the Y-axis. The results are means and standard errors of the EUS over 50 problem instances, each consisting of this two-round querying process, for $N = 50$ and $k_0 = 2, 5$. The provider's MDP and $N = 50$ recipient MDPs are randomly generated as described in Section 5.4.2.

a querying approach for the agents to improve their knowledge about each other to agree on a better commitment. Our empirical evaluations in Section 6.5.1 demonstrated that, paired with the discretization identified using the properties proved in Section 6.2, high-quality queries can be formed efficiently to induce commitments that nearly optimize the joint value.

The efficiency of the algorithms, in both centralized and decentralized settings, is obtained from the properties of the commitment value functions. These properties capture regularity in how the agents' values change with the commitment specification, implying that there are usually a small number of commitments (i.e. the breakpoints discretization) that preserve the information of the entire commitment space. This suggests that these commitments are worth more attention than others, not only when formulating cooperative commitments considered in this thesis, but also when two non-cooperative agents are negotiating a commitment, or when agents are coordinating with a more detailed commitment specification involving more than one time step.

CHAPTER VII

Conclusion

Probabilistic commitments provide a general framework for coordinating agents that are coupled by shared state features. This thesis formulates and solves problems that arise from the two-phase procedure of commitment-based coordination between the provider and the recipient. In the commitment formulation phase, the agents agree on a probabilistic commitment regarding the provider's influence on the recipient via the shared state features. For this phase, this thesis focuses on the cooperative scenario in which the agents aim to agree on the commitment that maximizes their joint value. In the commitment execution phase when the commitment is already determined, the planning of the two agents is decoupled because the provider's influence to the recipient on the shared state features is abstracted in the probabilistic commitment. For this phase, this thesis formulates and solves the provider's and the recipient's decoupled planning problems. The contributions to these problems, as presented in Chapters IV, V, and VI, lay the foundation of the probabilistic commitment framework for multiagent coordination. We review these contributions below.

1. Chapter IV presents the commitment semantics for the provider that prescribes its policy selection under Bayesian uncertainty about the environment model, including the transition and reward functions. Results on an illustrative domain show how such prescriptive semantics outperforms several alternatives because it achieves better cooperative behavior between the provider and the recipient. The chapter presents the method, Commitment Constrained Full Lookahead (CCFL), for exactly optimizing the provider's policy, often at a huge computational cost, while respecting the commitment semantics. Further, the chapter develops Commitment Constrained Lookahead (CCL) and its online iterative version (CCIL), that construct policies for the provider that provably respect the commitment semantics. The empirical evaluations show that, compared

with CCFL, CCL and CCIL can make better tradeoffs between computation cost and policy quality. The novel prescriptive semantics and the new methods together offer practical solutions for the provider to maximize its autonomy by responding to its evolving model uncertainty without detriment to its trustworthiness to the recipient, and thus solve the provider's planning problem in the commitment execution phase.

Besides the setting formalized in Section 4.1, one can apply the techniques developed in the methods, such as CCL's parametrized posterior lookahead and CCIL's commitment-constrained online iteration, to other settings where an agent is learning about the environment while its behavior is required to meet spatial-temporal constraints. For example, such techniques can be straightforwardly applied to 1) more detailed specifications of the provider's influence that involve multiple time steps, as we have discussed in Section 2.3, 2) and settings where the agent's model uncertainty is formalized in a non-Bayesian manner. For example, we have considered the scenario where the provider's model uncertainty is non-Bayesian and applied the techniques developed in this thesis to such a setting that involves minimax regret policy optimization objectives [ZSD17].

2. Chapter V formulates the recipient's planning problem in the commitment execution phase as its robust interpretation of the commitment and focuses on two commonly-studied types of commitment, achievement and maintenance. The notion of robustness hinges on the suboptimality of the influence that the recipient creates from the commitment specification, which the recipient uses to approximate the provider's true influence for its subsequent planning. This chapter develops several strategies for creating the approximate influence, and presents theoretical and empirical results showing that, despite strong similarities in the provider's modeling of the two types of commitment, there is a strategy that induces low suboptimality for achievement, while no identifiable strategy can robustly reduce the suboptimality for maintenance.

   Although the idea of approximate influence has been explored in prior work, this thesis is the first to develop principled strategies and evaluation metrics for the recipient to interpret a probabilistic commitment. The results assure us that the recipient can robustly interpret achievement commitments, and thus successful coordination with the provider can be secured. On the other hand, the results suggest that successful coordination with maintenance commitments

is harder, encouraging us to explore specifications more detailed than the single time step abstraction, so as to reduce the recipient's uncertainty when creating the approximate influence, but at the same time also reduce the flexibility the provider has. This points out an important future direction to better understand the pros and cons of more detailed specifications for maintenance. Section 7.1 presents concrete ideas in this direction.

3. Chapter VI solves the cooperative commitment formulation problem in an computationally efficient manner. Specifically, for the centralized setting, the chapter formulates and solves the problem of searching for the commitment that exactly maximizes the joint commitment value. For the decentralized setting, the chapter develops a querying approach for the agents to agree on an approximately-optimal commitment. As the core contribution that leads to the computational efficiency, the chapter proves several structural properties of the commitment value functions, which can be exploited in both settings for efficiently searching for the optimal cooperative commitment or constructing valuable queries. The empirical evaluations show that exploiting the properties significantly improves the computational efficiency.

   The properties of the commitment value functions reveal the structure of commitments: although the commitment space is infinitely large, there are usually a small number of commitments that preserve all the information in the commitment space. Thus, we expect that our identification of these properties will be valuable to the broader community of multiagent research, especially on commitment-based multiagent coordination and optimization.

7.1 Discussion of Future Work

We briefly discuss a few possible directions for future work.

Measuring trustworthiness of the provider. In this thesis, we have presented semantics and algorithms for the provider to adhere to its probabilistic commitment. A follow on and related problem is how the recipient can measure the provider's trustworthiness, in order to decide whether it should trust the provider and agree on the commitment. In the decentralized setting, as we have discussed in the problem of commitment formulation, the recipient does not have full knowledge about the provider's environment and/or policy, thus making it a challenging problem to precisely assess

the probability of the commitment being realized. As a feasible approach, either communicating directly about the provider's environment and/or policy, or about the provider's historical interactions with its environment, or both, will facilitate the recipient's assessment. If the recipient can effectively measure the provider's trustworthiness, we can ask how the provider can earn trust with minimum communication with the recipient and/or interactions with the environment.

Improving the recipient's interpretation of maintenance commitments. In Chapter V, we have supported the claim that the recipient's interpretation of maintenance commitments is harder by studying several strategies for creating the approximate influence. A natural question to ask is whether there exists such an approximate influence for maintenance, other than the ones we have studied, that we can prove has a lower bound on its suboptimality (similar to the one in Theorem V.1 for achievement), and/or we can empirically show induces low suboptimality. If the answer is negative or it is expensive for the recipient to create such an approximate influence, then we might need to rethink how we represent maintenance commitments for multiagent coordination. For achievement, the customarily terse commitment abstraction gives the provider a lot of flexibility by only constraining it to meet the probability at the commitment time and so it can unilaterally change its influences before then. In many cases, the gain in flexibility for the provider can be worth the relatively small value loss to the recipient. However, for maintenance, as it is difficult to find an effective approximate influence, the potential for the recipient to lose more value could mean that the provider should commit to a more detailed specification—the loss of flexibility for the provider in this case is warranted because the recipient makes much better decisions. Potential future work can better understand such tradeoffs in using maintenance commitments, allowing the community to apply commitment-based coordination to domains involving both achievement and maintenance.

Efficient formulation of cooperative maintenance commitments. In Chapter VI, we have developed algorithms that efficiently formulate cooperative commitments for achievement by exploiting the structural properties of the commitment value functions. A natural question is whether these properties still apply to maintenance commitments, so that we can develop similar algorithms for efficient formulation of cooperative maintenance commitments. The proofs for the properties of the provider's commitment value function are agnostic about the commitment type, and thus can still apply to maintenance. For the recipient, its commitment value

function for achievement hinges on the minimal enablement duration influence where u^- probabilistically toggles to u^+ at the latest time step by the commitment time. Thus, the proofs of the structural properties of the recipient's commitment value for achievement cannot straightforwardly apply to maintenance. Moreover, as we have discussed, it remains an open question what approximate influence is best to use to compute the recipient's commitment value for maintenance in the first place.

Beyond binary commitment features. For the recipient's interpretation (Chapter V) and cooperative commitment formulation (Chapter VI), this thesis has solved these problems for the scenario where the commitment feature is binary, involving two types of commitment that toggle the feature in opposite directions. The provider's modelling of a probabilistic commitment is nearly identical for the the two types of commitment and can be easily extended beyond the binary commitment feature. However, future work is needed to better understand how the recipient should interpret and utilize a probabilistic commitment for which the commitment feature is more complicated than binary and how the two agents can efficiently formulate such a commitment for coordination.

Communication during commitment execution. We have focused on the scenario in which the communication between the agents is only allowed during commitment formulation, but not allowed during the commitment execution phase (including the provider's adherence and the recipient's interpretation). We could relax this restriction by allowing (limited) communication during execution, and a number of interesting questions could arise subsequently. Such a relaxation could lead to the problem of how the agents can best exploit the limited communication. For example, if the provider is allowed to inform the recipient, for a limited number of time steps during execution, of the probability of realizing the commitment from the current time step (e.g., the probability for the iterative lookahead in CCIL), how should the provider wisely decide when to inform the recipient? Such a relaxation could also lead to the problem of multi-commitment formulation. For example, if an achievement commitment ends up being unrealized by the commitment time, what if we allow the agents to formulate a second commitment for the execution after the commitment time? Moreover, once an achievement commitment is realized, the agents can start formulating a maintenance commitment for the subsequent execution about the precondition that was just enabled, and thus such a relaxation encourages us to develop a unified framework for both achievement and maintenance.

Scaling to more agents and commitments. Throughout this thesis, we are concerned with a single commitment between two agents, with one agent fixed as the provider and the other as the recipient. Much future work is needed for handling scenarios where there can be more than two agents for coordination using multiple commitments. The provider might make a commitment to multiple recipients. Instead of a single commitment, agents might need to coordinate with a chain of commitments that are temporally correlated. Mutual and cyclic commitments can exist, where an agent can shift from being a provider to being a recipient over time, or even can be both a provider and a recipient at the same time. These interesting scenarios naturally exist in multiagent coordination, and extending the work accomplished in this thesis to these scenarios requires scaling the problem formulations and solution methods to multiple agents and commitments.

7.2 Closing Remarks

With autonomous agents increasingly embedded in our daily lives, flexible and trustworthy coordination is crucial for the agents to collectively make beneficial societal impact. Inspired by how people work together, the commitment-based framework has emerged as one of the most promising ideas for achieving flexible and trustworthy multiagent coordination. This thesis develops formal notions and new techniques for the agents to represent, formulate, and plan with probabilistic commitments for coordination under inherent uncertainty about their environments. With strong provable guarantees and impressive empirical results in a range of classic multiagent planning domains, the developed methods lay the foundation of multiagent coordination with probabilistic commitments. We believe the contributions of this thesis are valuable to other researchers and engineers who are dedicated to building effective multiagent systems for complex, real-world settings.

BIBLIOGRAPHY

BIBLIOGRAPHY

[AGJ07] Thomas Agotnes, Valentin Goranko, and Wojciech Jamroga. Strategic commitment and release in logics for multi-agent systems (extended abstract). Technical Report IfI-08-01, Clausthal University, 2007.

[Alt99] Eitan Altman. *Constrained Markov Decision Processes*, volume 7. CRC Press, 1999.

[ASBSEM14] Faisal Al-Saqqar, Jamal Bentahar, Khalid Sultan, and Mohamed El-Menshawy. On the interaction between knowledge and social commitments in multi-agent systems. *Applied Intelligence*, 41(1):235–259, 2014.

[BAHZ09] Daniel S. Bernstein, Christopher Amato, Eric A. Hansen, and Shlomo Zilberstein. Policy iteration for decentralized control of Markov decision processes. *Journal of Artificial Intelligence Research*, 34:89–132, 2009.

[BDAMY13] José Bento, Nate Derbinsky, Javier Alonso-Mora, and Jonathan S. Yedidia. A message-passing algorithm for multi-agent trajectory planning. In *Advances in Neural Information Processing Systems*, pages 521–529, 2013.

[BGIZ02] Daniel S. Bernstein, Robert Givan, Neil Immerman, and Shlomo Zilberstein. The complexity of decentralized control of Markov decision processes. *Mathematics of Operations Research*, 27(4):819–840, 2002.

[BLG10] Hadi Bannazadeh and Alberto Leon-Garcia. A distributed probabilistic commitment control algorithm for service-oriented systems. *IEEE Transactions on Network and Service Management*, 7(4):204–217, 2010.

[Bou96] Craig Boutilier. Planning, learning and coordination in multiagent decision processes. In *Proceedings of the 6th Conference on Theoretical Aspects of Rationality and Knowledge*, pages 195–210, 1996.

[Bou02] Craig Boutilier. A POMDP formulation of preference elicitation problems. In *Proceedings of the Eighteenth National Conference on Artificial Intelligence*, pages 239–246, 2002.

[BPPS06] Craig Boutilier, Relu Patrascu, Pascal Poupart, and Dale Schuurmans. Constraint-based optimization and utility elicitation using the minimax decision criterion. *Artificial Intelligence*, 170(8-9):686–713, 2006.

[BT97] Dimitris Bertsimas and John N. Tsitsiklis. *Introduction to Linear Optimization*, volume 6. Athena Scientific Belmont, MA, 1997.

[Cas95] Cristiano Castelfranchi. Commitments: From individual intentions to groups and organizations. In *Proceedings of the International Conference on Multiagent Systems*, pages 41–48, 1995.

[CBG02] Jonathan Carter, Elijah Bitting, and Ali A. Ghorbani. Reputation formalization for an information–sharing multi–agent system. *Computational Intelligence*, 18(4):515–534, 2002.

[CKP00] Urszula Chajewska, Daphne Koller, and Ronald Parr. Making rational decisions using adaptive utility elicitation. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence*, pages 363–369, 2000.

[CL90] Philip R. Cohen and Hector J. Levesque. Intention is choice with commitment. *Artificial Intelligence*, 42(2-3):213–261, 1990.

[CMMT13] Federico Chesani, Paola Mello, Marco Montali, and Paolo Torroni. Representing and monitoring social commitments using the event calculus. *Autonomous Agents and Multi-Agent Systems*, 27(1):85–130, 2013.

[CPL] CPLEX. *IBM ILOG CPLEX 12.1*. `https://www.ibm.com/analytics/cplex-optimizer`.

[CS08] Bradley J. Clement and Steven R. Schaffer. Exploiting C-TÆMS models for policy search. In *Multiagent Planning Workshop at The Eighteenth International Conference on Automated Planning and Scheduling*, 2008.

[CSD14] Robert Cohn, Satinder Singh, and Edmund Durfee. Characterizing EVOI-sufficient k-response query sets in decision problems. In *The Seventeenth International Conference on Artificial Intelligence and Statistics*, pages 131–139, 2014.

[DD04] Dmitri A. Dolgov and Edmund H. Durfee. Optimal resource allocation and policy formulation in loosely-coupled Markov decision processes. In *Proceedings of the Fourteenth International Conference on Automated Planning and Scheduling*, pages 315–324, 2004.

[DD05] Dmitri Dolgov and Edmund Durfee. Stationary deterministic policies for constrained MDPs with multiple rewards, costs, and discount factors. In *International Joint Conference on Artificial Intelligence*, volume 19, pages 1326–1331, 2005.

[Dec87] Keith S. Decker. Distributed problem-solving techniques: A survey. *IEEE Transactions on Systems, Man, and Cybernetics*, 17(5):729–740, 1987.

[DNS08] Nirmit Desai, Nanjangud C. Narendra, and Munindar P. Singh. Checking correctness of business contracts via commitments. In *Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems*, volume 2, pages 787–794, 2008.

[DTH14] S. Duff, J. Thangarajah, and J. Harland. Maintenance goals in intelligent agents. *Computational Intelligence*, 30(1):71–114, 2014.

[Dur99] Edmund H. Durfee. Distributed problem solving and planning. In *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence* (Chapter 3). MIT Press, 1999.

[GBL+15] Jones Granatyr, Vanderson Botelho, Otto Robert Lessing, Edson Emílio Scalabrin, Jean-Paul Barthès, and Fabrício Enembreck. Trust and reputation models for multiagent systems. *ACM Computing Surveys (CSUR)*, 48(2):27, 2015.

[GKP02] Carlos Guestrin, Daphne Koller, and Ronald Parr. Multiagent planning with factored MDPs. In *Advances in Neural Information Processing Systems*, pages 1523–1530, 2002.

[GLZ16] Akın Günay, Yang Liu, and Jie Zhang. ProMoca: Probabilistic modeling and analysis of agents in commitment protocols. *Journal of Artificial Intelligence Research*, 57:465–508, 2016.

[GMDB08] Robert P. Goldman, David J. Musliner, Edmund H. Durfee, and Mark S. Boddy. Coordinating highly contingent plans: Biasing distributed MDPs towards cooperative behavior. In *Multiagent Planning Workshop at The Eighteenth International Conference on Automated Planning and Scheduling*, 2008.

[Gur] Gurobi. *Gurobi 8.1.* `http://www.gurobi.com/products/gurobi-optimizer`.

[Han99] Eric A. Hansen. *Finite-memory control of partially observable systems.* PhD thesis, University of Massachusetts, Amherst, 1999.

[HBZ04] Eric A. Hansen, Daniel S. Bernstein, and Shlomo Zilberstein. Dynamic programming for partially observable stochastic games. In *Proceedings of the Nineteenth National Conference on Artificial Intelligence*, volume 4, pages 709–715, 2004.

[Hia09] Laura M. Hiatt. *Probabilistic plan management.* PhD thesis, Carnegie Mellon University, 2009.

[HJS06] Trung Dong Huynh, Nicholas R. Jennings, and Nigel R. Shadbolt. An integrated trust and reputation model for open multi-agent systems. *Autonomous Agents and Multi-Agent Systems*, 13(2):119–154, 2006.

[HvR07] K. V. Hindriks and M. B. van Riemsdijk. Satisfying maintenance goals. In *5th International Workshop Declarative Agent Languages and Technologies*, pages 86–103, 2007.

[Jen93] N. R. Jennings. Commitments and conventions: The foundation of coordination in multi-agent systems. *The Knowledge Engineering Review*, 8(3):223–250, 1993.

[KK02] Spiros Kapetanakis and Daniel Kudenko. Reinforcement learning of coordination in cooperative multi-agent systems. *The Eighteenth National Conference on Artificial Intelligence*, 2002:326–331, 2002.

[KLC98] Leslie Pack Kaelbling, Michael L. Littman, and Anthony R. Cassandra. Planning and acting in partially observable stochastic domains. *Artificial intelligence*, 101(1-2):99–134, 1998.

[LR00] Martin Lauer and Martin Riedmiller. An algorithm for distributed reinforcement learning in cooperative multi-agent systems. In *Proceedings of the Seventeenth International Conference on Machine Learning*, pages 535–542, 2000.

[MAT] MATLAB. *MATLAB Optimization Toolbox*. `https://www.mathworks.com/products/optimization.html`.

[MH03] Ashok U. Mallya and Michael N. Huhns. Commitments among agents. *IEEE Internet Computing*, 7(4):90–93, 2003.

[MMS+18] Felipe Meneguzzi, Mauricio C. Magnaguagno, Munindar P. Singh, Pankaj R. Telang, and Neil Yorke-Smith. GoCo: Planning expressive commitment protocols. *Autonomous Agents and Multi-Agent Systems*, 32(4):459–502, 2018.

[MSB+08] Rajiv T Maheswaran, Pedro Szekely, Marcel Becker, Stephen Fitzpatrick, Gergely Gati, Jing Jin, Robert Neches, Narges Noori, Craig Rogers, Romeo Sanchez, Kevin Smyth, and Chris Van-Buskirk. Predictability & criticality metrics for coordination in complex environments. In *Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 647–654, 2008.

[MTYS15] Felipe Meneguzzi, Pankaj R. Telang, and Neil Yorke-Smith. Towards planning uncertain commitment protocols. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, pages 1681–1682, 2015.

[NTY+03] Ranjit Nair, Milind Tambe, Makoto Yokoo, David Pynadath, and Stacy Marsella. Taming decentralized POMDPs: Towards efficient policy computation for multiagent settings. In *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence*, pages 705–711, 2003.

[NWF78] George L. Nemhauser, Laurence A. Wolsey, and Marshall L. Fisher. An analysis of approximations for maximizing submodular set functions. *Mathematical Programming*, 14(1):265–294, 1978.

[OPT] OPTI. *OPTI Toolbox v2.2.* `https://www.inverseproblem.co.nz/OPTI`.

[OWK12] Frans Adriaan Oliehoek, Stefan J. Witwicki, and Leslie Pack Kaelbling. Influence-based abstraction for multiagent systems. In *Proceedings of the Twenty-sixth AAAI Conference on Artificial Intelligence*, pages 1422–1428, 2012.

[PL05] Liviu Panait and Sean Luke. Cooperative multi-agent learning: The state of the art. *Autonomous Agents and Multi-agent Systems*, 11(3):387–434, 2005.

[PMP+15] Pascal Poupart, Aarti Malhotra, Pei Pei, Kee-Eung Kim, Bongseok Goh, and Michael Bowling. Approximate linear programming for constrained partially observable Markov decision processes. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, pages 3342–3348, 2015.

[POM17] Ramon Fraga Pereira, Nir Oren, and Felipe Meneguzzi. Detecting commitment abandonment by monitoring sub-optimal steps during plan execution. In *Proceedings of the 16th Conference on Autonomous Agents and Multi-Agent Systems*, pages 1685–1687, 2017.

[PSM13] Isaac Pinyol and Jordi Sabater-Mir. Computational trust and reputation models for open multi-agent systems: A review. *Artificial Intelligence Review*, 40(1):1–25, 2013.

[Put14] Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, 2014.

[Raf82] H. Raffia. *The Art and Science of Negotiation*. Harvard University Press, 1982.

[RD11] Constantin A. Rothkopf and Christos Dimitrakakis. Preference elicitation and inverse reinforcement learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 34–48. Springer, 2011.

[RHJ04] Sarvapali D. Ramchurn, Dong Huynh, and Nicholas R. Jennings. Trust in multi-agent systems. *The Knowledge Engineering Review*, 19(1):1–25, 2004.

[Rob04] David Robertson. Multi-agent coordination as distributed logic programming. In *International Conference on Logic Programming*, pages 416–430. Springer, 2004.

[SCZ05] Daniel Szer, François Charpillet, and Shlomo Zilberstein. MAA*: A heuristic search algorithm for solving decentralized POMDPs. In *Proceedings of the Twenty-First Conference on Uncertainty in Artificial Intelligence*, pages 576–583, 2005.

[Shi63] Albert N. Shiryaev. On optimum methods in quickest detection problems. *Theory of Probability and Its Applications*, 8(1):22–46, 1963.

[Sin99] Munindar P. Singh. An ontology for commitments in multiagent systems. *Artificial Intelligence in the Law*, 7(1):97–113, 1999.

[Sin12] Munindar P. Singh. Commitments in multiagent systems: Some history, some confusions, some controversies, some prospects. In *The Goals of Cognition. Essays in Honor of Cristiano Castelfranchi*, pages 601–626, London, 2012.

[SL01] Tuomas Sandholm and Victor R. Lesser. Leveled commitment contracts and strategic breach. *Games and Economic Behavior*, 35:212–270, 2001.

[SS73] Richard D. Smallwood and Edward J. Sondik. The optimal control of partially observable Markov processes over a finite horizon. *Operations Research*, 21(5):1071–1088, 1973.

[SS02] Jordi Sabater and Carles Sierra. Reputation and social network analysis in multi-agent systems. In *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 475–482. ACM, 2002.

[SS04] Trey Smith and Reid Simmons. Heuristic search value iteration for POMDPs. In *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence*, pages 520–527, 2004.

[STW16] Pedro Santana, Sylvie Thiébaux, and Brian Williams. RAO*: An algorithm for chance-constrained POMDPs. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, pages 3308–3314, 2016.

[TMS13] Pankaj R. Telang, Felipe Meneguzzi, and Munindar P. Singh. Hierarchical planning about goals and commitments. In *Proceedings of the 2013 International Conference on Autonomous Agents and Multiagent Systems*, pages 877–884, 2013.

[VB10] Paolo Viappiani and Craig Boutilier. Optimal Bayesian recommendation sets and myopically optimal choice query sets. In *Advances in Neural Information Processing Systems*, pages 2352–2360, 2010.

[VKP09] Jirí Vokrínek, Antonín Komenda, and Michal Pechoucek. Decommitting in multi-agent execution in non-deterministic environment: Experimental approach. In *Proceedings of the 8th International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 977–984, 2009.

[WD07] Stefan J. Witwicki and Edmund H. Durfee. Commitment-driven distributed joint policy search. In *Proceedings of the 6th International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 480–487, 2007.

[WD09] Stefan J. Witwicki and Edmund H. Durfee. Commitment-based service coordination. *International Journal of Agent-Oriented Software Engineering*, 3:59–87, 01 2009.

[WD10] Stefan J. Witwicki and Edmund H. Durfee. Influence-based policy abstraction for weakly-coupled Dec-POMDPs. In *Proceedings of the 20th International Conference on Automated Planning and Scheduling*, pages 185–192, 2010.

[Win06] Michael Winikoff. Implementing flexible and robust agent interactions using distributed commitment machines. *Multiagent and Grid Systems*, 2(4):365–381, 2006.

[XL00] Ping Xuan and Victor R. Lesser. Incorporating uncertainty in agent commitments. In *Intelligent Agents VI. Agent Theories, Architectures, and Languages*, pages 57–70. Springer, 2000.

[XS01] Jie Xing and Munindar P. Singh. Formalization of commitment-based agent interaction. In *Proceedings of the 2001 ACM Symposium on Applied Computing*, pages 115–120, 2001.

[ZDS+16] Qi Zhang, Edmund H. Durfee, Satinder Singh, Anna Chen, and Stefan J. Witwicki. Commitment semantics for sequential decision making under reward uncertainty. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, pages 3315–3323, 2016.

[ZDS17] Shun Zhang, Edmund Durfee, and Satinder Singh. Approximately-optimal queries for planning in reward-uncertain Markov decision processes. In *Proceedings of the Twenty-Seventh International Conference on Automated Planning and Scheduling*, pages 339–347, 2017.

[ZDS18] Qi Zhang, Edmund H. Durfee, and Satinder Singh. Challenges in the trustworthy pursuit of maintenance commitments under uncertainty. In *Proceedings of the 20th International Trust Workshop co-located with AAMAS/IJCAI/ECAI/ICML 2018*, pages 75–86, 2018.

[ZDS20a] Qi Zhang, Edmund Durfee, and Satinder Singh. Efficient querying for cooperative commitments. In *11th International Workshop on Optimization in Multiagent Systems at AAMAS*, 2020.

[ZDS20b] Qi Zhang, Edmund H. Durfee, and Satinder Singh. Semantics and algorithms for trustworthy commitment achievement under model uncertainty. *Autonomous Agents and Multi-Agent Systems*, 34(1):19, 2020.

[ZSD17] Qi Zhang, Satinder Singh, and Edmund Durfee. Minimizing maximum regret in commitment constrained sequential decision making. In *Twenty-Seventh International Conference on Automated Planning and Scheduling*, pages 348–356, 2017.

[ZSD20] Qi Zhang, Satinder Singh, and Edmund Durfee. Modeling probabilistic commitments for maintenance is inherently harder than for achievement. In *Proceedings of the Thirty-fourth AAAI Conference on Artificial Intelligence*, 2020.