Formalizing and Enforcing Purpose Restrictions

Michael Carl Tschantz

May 9, 2012
CMU-CS-12-117

School of Computer Science
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15213

Thesis Committee:
Anupam Datta, co-chair
Jeannette M. Wing, co-chair
Lorrie Faith Cranor
Manuela M. Veloso
Joseph Y. Halpern, Cornell University

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy

This research was supported by the U.S. Army Research Office grants DAAD19-02-1-0389 and W911NF-09-1-0273 to CyLab, by the National Science Foundation (NSF) grants CCF0424422 and CNS1064688, and by the U.S. Department of Health and Human Services grant HHS 90TR0003/01. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of any sponsoring institution, the U.S. government or any other entity.

Some of this dissertation appears in a conference paper in the proceedings of the 33rd IEEE Symposium on Security and Privacy (San Francisco, May 2012) as "Formalizing and Enforcing Purpose Restrictions in Privacy Policies" [TDW12a] and in the full version of that paper [TDW12b]. Some of the material presented in those papers appeared in an earlier technical report [TDW11]. This version of this dissertation corrects information on the title page about funding from an earlier version published on May 8, 2012.

Keywords: Privacy, Formal Methods, Auditing, Compliance Checking, Planning, MDPs, POMDPs

Abstract

Privacy policies often place restrictions on the purposes for which a governed entity may use personal information. For example, regulations, such as the Health Insurance Portability and Accountability Act (HIPAA), require that hospital employees use medical information for only certain purposes, such as treatment, but not for others, such as gossip. Thus, using formal or automated methods for enforcing privacy policies requires a semantics of *purpose restrictions* to determine whether an action is *for* a purpose. We provide such a semantics using a formalism based on *planning*. We model planning using a modified version of Markov Decision Processes (MDPs), which exclude redundant actions for a formal definition of *redundant*. We argue that an action is for a purpose if and only if the action is part of a plan for optimizing the satisfaction of that purpose under the MDP model. We use this formalization to define when a sequence of actions is *only for* or *not for* a purpose. This semantics enables us to create and implement an algorithm for automating auditing, and to describe formally and compare rigorously previous enforcement methods. We extend this formalization to Partially Observable Markov Decision Processes (POMDPs) to answer when *information* is used for a purpose. To validate our semantics, we provide an example application and conduct a survey to compare our semantics to how people commonly understand the word “purpose”.

Dedicated to Euclid, Ohio:
the people, the schools, the swimming pools,
the lake, the creek, the August twilight,
my teachers, my coaches, my childhood friends,
and especially my loving parents.
You made my travels possible.

Acknowledgments

My advisors have provided not just general guidance and kind support but also specific contributions to my research. For example, the idea of examining purpose restrictions came from Jeannette M. Wing, and Anupam Datta proposed applying the formalism presented in this dissertation more generally to operating procedures.

I could not have covered the breadth of this work without the aid of the members of my thesis committee. I would not have attempted the survey validating the semantics had it not been for the advice of Lorrie Faith Cranor. Joseph Y. Halpern steered me toward related works in philosophy. Manuela M. Veloso pushed me to better understand the previous work in artificial intelligence, greatly improving the algorithms.

My colleagues Dilsun Kaynar and Divya Sharma provided many helpful comments improving my arguments. Anonymous reviewers of related conference papers also improved the expression of my thesis.

My arguments build upon and use a great many theories and frameworks developed by others not directly involved in this research. In addition to the authors listed in the bibliography, numerous texts and websites, such as Wikipedia, informed my work.

Others contributed by allowing me to focus on my thesis. The people involved in many development efforts, such as GNU, Latex, Linux, PLT Racket, R, and Xfig, gave me the tools I needed to test and document my ideas. The supportive staff of Carnegie Mellon University kept me fed and away from paperwork. The United States government, among others, provided funds enabling me to be a full-time student.

My thesis uses models of artificial intelligence to solve a problem of formal foundations. I could not have combined these areas without the strong foundations I had in each from undergraduate research opportunities at Brown University (class of 2005). In particular, I learned of Markov Decision Processes while conducting research on autonomous trading agents with Amy Greenwald. Kathi Fisler and Shriram Krishnamurthi introduced me to formal policy analysis when I researched XACML policy analysis. Shriram also sent me down the path of formalizing concepts with my undergraduate thesis on reasonability properties for access-control policy languages. I have also benefited from conducting research with Sagar Chaki at the Software Engineering Institute of CMU, with Aditya V. Nori at Microsoft Research India, and with John E. Savage at Brown.

More generally, I have benefited from my many teachers and colleagues at CMU, at Brown, and in the public school system of Euclid, Ohio, including Euclid High School (class of 2001). Lastly and most fundamentally, my education would have been possible without the unwavering support of my parents.

I thank everyone who has helped me with my education.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation of Problem

Purpose is a key concept for privacy policies. For example, the European Union requires that [The95]:

> Member States shall provide that personal data must be [...] collected for specified, explicit and legitimate purposes and not further processed in a way incompatible with those purposes.

The United States also has laws placing purpose restrictions on information in some domains such as the Health Insurance Portability and Accountability Act (HIPAA) [Off03] for medical information and the Gramm-Leach-Bliley Act [Uni10] for financial records. These laws and best practices motivate organizations to discuss in their privacy policies the purposes for which they will use information.

Some privacy policies warn users that the policy provider may use certain information for certain purposes. For example, the privacy policy of a medical provider states, "We may disclose your [protected health information] for public health activities and purposes [...]" [Was03]. Such warnings do not constrain the behavior of the policy provider.

Other policies that prohibit using certain information for a purpose do constrain the behavior of the policy provider. Examples include the privacy policy of Yahoo! Email, which states that "Yahoo!'s practice is *not* to use the content of messages stored in your Yahoo! Mail account *for* marketing purposes" [Yah10b, emphasis added].

Some policies even limit the use of certain information to an explicit list of purposes. The privacy policy of The Bank of America states, "Employees are authorized to access Customer Information *for* business purposes *only*." [Ban05, emphasis added]. The HIPAA Privacy Rule [Off03] requires that covered entities (e.g., health care providers and business partners) only use or disclose protected health information about a patient with that patient's written authorization or:

> [...] for the following purposes or situations: (1) To the Individual [...]; (2) Treatment, Payment, and Health Care Operations; (3) Opportunity to Agree or Object; (4) Incident to an otherwise permitted use and disclosure; (5) Public Interest and Benefit Activities; and (6) Limited Data Set for the purposes of research, public health or health care operations.

These examples show that verifying that an organization obeys a privacy policy requires a semantics of *purpose restrictions*. In particular, enforcement requires the ability to determine that the organization under scrutiny obeys at least two classes of purpose restrictions. As shown in the example rule from Yahoo!, the first requirement is that the organization does *not* use certain sensitive information *for* a given purpose. The second, as the example rule from HIPAA shows, is that the organization uses certain sensitive information *only for* a given list of purposes. We call the first class of restrictions *prohibitive rules* (not-for) and the second class *exclusivity rules* (only-for). A prohibitive rule disallows an action for a particular purpose. An exclusivity rule disallows an action for every purpose other than the exceptions the rule lists. Each class of rule requires determining whether the organization's behavior is *for* a purpose, but they differ in whether this indicates a violation or compliance, respectively.

For example, consider a physician accessing a medical record. Under the HIPAA Privacy Rule, the physician may access the record only for certain purposes such as treatment, research, and billing. Thus, for an trusted auditor (either internal or external) to determine whether the physician has obeyed the Privacy Rule requires the auditor to determine the purposes for which the physician accessed the record. The auditor's ability to determine the purposes behind actions is limited since the auditor can only observe the behavior of the physician. As a physician may perform the exact same actions for different purposes, the auditor can never be sure of the purposes behind an action. However, if the auditor determines that the record access could not have possibly been for any of the purposes allowed under the Privacy Rule, then the auditor knows that the physician violated the policy.

Manual enforcement of these privacy policies is labor intensive and error prone. Thus, to reduce costs and make their operations more trustworthy, organizations would like to automate the enforcement of the privacy policies governing their operations; tool support for this activity is beginning to emerge in the market. For example, FairWarning and Cerner's P2Sentinel offer automated services for the detection of privacy breaches in a hospital setting [Fai, Cer]. Meanwhile, previous research has proposed formal methods to enforce purpose restrictions [AKSX02, BBL05, HA05, AF07, BL08, PGY08, JSNS09, NBL$^+$10, EKWB11].

However, each of these endeavors starts by assuming that actions or sequences of actions are labeled with the purposes they are *for*. They avoid analyzing the meaning of *purpose* and provide no method of performing this labeling other than through intuition alone. The absence of a formal semantics to guide this determination has hampered the development of methods for ensuring policy compliance. Such a definition would provide insights into how to develop tools that identify suspicious accesses in need of detailed auditing and algorithms for determining which purposes an action could possibly be for. Such a definition would also show which enforcement approaches are most accurate. More fundamentally, such a definition could frame the scientific basis of a societal and legal understanding of purpose and of privacy policies that use the notion of purpose. Such a foundation can, for example, guide implementers as they codify in software an organization's interpretation of internal and government-imposed privacy policies.

1.2 Motivation of Our Approach

We start with an informal example that suggests that *an action is for a purpose if the action is part of a plan for achieving that purpose*. Consider a physician working at a hospital who, as a specialist, also owns a private practice that tests for bone damage using a novel technique for extracting information from X-ray images. After seeing a patient and taking an X-ray, the physician forwards the patient's medical record including the X-ray to his private practice to apply this new technology. As this action entails the transmission of protected health information, the physician will have violated HIPAA if this transmission is not for one of the purposes HIPAA allows. The physician would also run afoul of the hospital's own policies governing when outside consultations are permissible unless this action was for a legitimate purpose. Finally, the patient's insurance will only reimburse the costs associated with this consultation if a medical reason (purpose) exists for them. The physician claims that this consultation was for reaching a diagnosis. As such, it is for the purpose of treatment and, therefore, allowed under each of these policies. The hospital auditor, however, has selected this action for investigation since the physician's making a referral to his own private practice makes possible the alternate motivation of profit.

Whether the physician violated these policies depends upon details not presented in the above description. For example, we would expect the auditor to ask questions such as:

1. Was the test relevant to the patient's condition?
2. Did the patient benefit medically from having the test?
3. Was this test the best option for the patient?

We will introduce these details as we introduce each of the factors relevant to the purposes behind the physician's actions.

States and Actions. Sometimes the purposes for which an agent takes an action depend upon the previous actions and the state of the system. In the above example, whether the test is relevant depends upon the condition of the patient, that is, the state that the patient is in.

While an auditor could model the act of transmitting the record as two (or more) different actions based upon the state of the patient, modeling two concepts with one formalism could introduce errors. A better approach is to model the state of the system. The state captures the context in which the physician takes an action and enables the purposes of an action to depend upon the actions that precede it.

The physician's own actions also affect the state of the system and, thus, the purposes for which his actions are. For example, had the physician transmitted the patient's medical record before taking the X-ray, then the transmission could not have been for treatment since the physician's private practice only operates on X-rays and would have no use for the record without the X-ray.

The above example illustrates that when an action is for a purpose, the action is part of a sequence of actions that can lead to a state in which some goal associated with the purpose is achieved. In the example, the goal is reaching a diagnosis. Only when the X-ray is first added to the record is this goal reached.

Non-redundancy. Some actions, however, may be part of such a sequence without actually being for the purpose. For example, suppose that the patient's X-ray clearly shows the patient's problem. Then, the physician can reach a diagnosis without sending the record to the private practice. Thus, while both taking the X-ray and sending the medical record might be part of a sequence of actions that leads to achieving a diagnosis, the transmission does not actually contribute to achieving the diagnosis: the physician could omit it and the diagnosis could still be reached.

From this example, it may be tempting to conclude that an action is *for* a purpose only if that action is *necessary* to achieve that purpose. However, consider a physician who, to reach a diagnosis, must either send the medical record to a specialist or take an MRI. In this scenario, the physician's sending the record to the specialist is not necessary since he could take an MRI. Likewise, taking the MRI is not necessary. Yet, the physician must do one or the other and that action will be for the purpose of diagnosis. Thus, an action may be for a purpose without being necessary for achieving the purpose.

Rather than *necessity*, we use the weaker notion of *non-redundancy* found in work on the semantics of *causation* (e.g., [Mac74]). Given a sequence of actions that achieves a goal, an action in it is *redundant* if that sequence with that action removed (and otherwise unchanged) also achieves the goal. An action is *non-redundant* if removing that action from the sequence would result in the goal no longer being achieved. Thus, non-redundancy may be viewed as necessity under an otherwise fixed sequence of actions.

For example, suppose the physician decides to send the medical record to the specialist. Then, the sequence of actions modified by removing this action would not lead to a state in which a diagnosis is reached. Thus, the transmission of the medical record to the specialist is non-redundant. However, had the X-ray revealed to the physician the diagnosis without needing to send it to a specialist, the sequence of actions that results from removing the transmission from the original sequence would still result in a diagnosis. Thus, the transmission would be redundant.

Quantitative Purposes. Above we implicitly presumed that the diagnosis from either the specialist or an MRI had equal quality. This need not be the case. Indeed, many purposes are actually fulfilled to varying degrees. For example, the purpose of marketing is never completely achieved since there is always more marketing to do. Thus, we model a purpose by assigning to each state-action pair a number that describes how well that action fulfills that purpose when performed in that state. We require that the physician selects the test that maximizes the quality of the diagnosis as determined by total purpose score accumulated over all his actions.

We must adjust our notion of non-redundancy accordingly. An action is non-redundant if removing that action from the sequence would result in the purpose being satisfied less. Now, even if the physician can make a diagnosis himself, sending the record to a specialist would be non-redundant if getting a second opinion improves the quality of the diagnosis.

Probabilistic Systems. The success of many medical tests and procedures is probabilistic. For example, with some probability the physician's test may fail to reach a diagnosis. The physician would still have transmitted the medical record for the purpose of diagnosis even if the test failed to reach one. This possibility affects our semantics of purpose: now an action may be for a purpose even if that purpose is never achieved.

To account for such probabilistic events, we model the outcome of the physician's actions as probabilistic. For an action to be for a purpose, we require that there be a non-zero probability of the purpose being achieved and that the physician attempts to maximize the expected reward. In essence, we require that the physician attempts to achieve a diagnosis. Thus, the auditee's *plan* determines the purposes behind his actions rather than just the actions themselves.

Overview of Approach. This example illustrates key factors in determining whether an action is for a purpose. In particular, the auditor should model the auditee as an agent that interacts with an *environment model*. The environment model shows how the actions the auditee can perform affect the state of the environment. It also models how well each state and action satisfies each purpose that the modeled auditee might possibly find motivating. Limiting consideration to one purpose, the environment model becomes a Markov Decision Process (MDP) where the degree of satisfaction of that purpose is the reward function of the MDP. If the auditee is motivated to act by only that purpose, then the auditee's actions must correspond to an optimal *plan* for this MDP and these actions are *for* that purpose. Additionally, we use a stricter definition of optimal than used for standard MDPs to reject redundant actions that neither decrease nor increase the total reward.

In this example, the auditor would examine an MDP modeling the physician's environment with the quality of treatment as the reward function to be optimized. If no optimal plans for this MDP involve ordering the test, then the auditor can conclude definitively that the physician did not order the test for treatment.

1.3 Statement of Thesis

The goal of this work is to study the meaning of *purpose* in the context of enforcing privacy policies. We aim to provide formal definitions suitable for automating the enforcement of purpose restrictions. Since post-hoc auditing provides the perspective often required to determine the purpose of an action, we focus on automated auditing. However, we believe our semantics is applicable to other enforcement mechanisms and may also clarify informal reasoning. For example, in Chapter 4, we use it to create an operating procedure that encourages compliance with a purpose restriction.

We find that *planning* is central to the meaning of purpose. We see the role of planning in the definition of the sense of the word "purpose" most relevant to our work [SW89]:

> The object for which anything is done or made, or for which it exists; the result or effect intended or sought; end, aim.

Similarly, work on cognitive psychology calls purpose "the central determinant of behavior" [DKP96, p19]. If our auditors are concerned with rational auditees (the person or organization being audited), then we may assume the auditee uses a plan to determine what actions it will perform in its attempt to achieve its purposes. We (as have philosophers [Tay66]) conclude that if an auditee chooses to perform an action a while planning to achieve the purpose p, then the auditee's action a is *for the purpose* p.

Our goal is to make these notions formal in a manner useful for automation and computation. In particular, this dissertation argues the following thesis:

> A model of planning underlies a formalization of purpose restrictions that enables their automated enforcement.

As suggested by the example in the previous section, we start by using the MDP formalism as the model of planning. However, when we consider purpose restrictions on information use, we use Partially Observable Markov Decision Processes (POMDPs) instead. In either case, we compare the behaviors of the auditee as recorded in the log to how the auditee would behave when selecting a plan of action using a model for the purpose in question.

To argue our thesis, this dissertation presents various contributions as summarized in Section 1.5 and is structured as explained in Section 1.6. In particular, we provide a formal semantics using a planning model to determine whether a sequence of actions is for a purpose, and we build upon this formalization algorithms for applying auditing to purpose restrictions. Before discussing these contributions, we put our thesis in context by summarizing prior work.

1.4 Prior Work

In this section, we provide a brief overview of prior work to show that prior work has not demonstrated our thesis. In Chapter 7, we cover prior work in more detail.

The planning-based semantics of purpose we use follows from informal philosophical inquiry [Tay66]. Taylor notes that whether an action is for a purpose depends upon the plan that leads the actor to perform the action and not on whether the actor succeeds in furthering that purpose. While presenting numerous examples illustrating the distinction, this prior work did not provide formal definitions, discuss information use, formalize purpose restrictions, discuss their enforcement, provide algorithms, or present empirical validation.

Works that do provide formal models fall into one of either two strands: enforcing purpose restrictions and goal inference. Our work builds on both of these strands.

Enforcing Purpose Restrictions. Most prior work on using formal methods for enforcing purpose restrictions has focused on when observable actions further a purpose [AKSX02, BBL05, AF07, BL08, PGY08, JSNS09, NBL$^+$10, EKWB11]. These works do not empirically show that their formalism corresponds to the actual meaning of purpose restrictions. Our thesis argues that an action is for a purpose when that action is part of a plan for that purpose, as opposed to furthering that purpose. Furthermore, none of these works formalize information use.

The prior work of Hayati and Abadi on enforcing purpose restricts does not fit into the above mold [HA05]. It provides a type system for tracking information flow in programs with purpose restrictions in mind. However, their work does not formalize when information use is for a purpose since it presupposes that the programmer can determine whether a function uses information for a certain purpose and provides no formal guidance for making this determination.

A closely related enforcement problem is that of *minimal disclosure*, which requires that the amount of information used in granting a request for access should be as little as possible while still achieving the purpose behind the request. However, purpose restrictions do not require the amount of information used to be minimal and often involve purposes that are never fully achieved (e.g., more marketing is always possible). Furthermore, works on minimal disclosure [MMZ06, BMDS07] do not use a planning-based formalism. They also lack the probabilistic transitions necessary to see the distinction between information use furthering a purpose and being part of a plan for furthering a purpose.

Goal Inference. The essence of our formalization of purpose restrictions is to reduce the problem to one of *goal inference*. Goal inference is the problem of determining, from the actions and states of a planning agent, the goal that agent is pursuing. Under our formalization, the auditee is the planning agent and the possible purposes are the possible goals.

The previous work on goal inference most closely related to ours use models similar to the MDP and POMDP models we use [RSM04, SGBR04, VR05, BTS06, VR06, BTS07, BST09, BKvdWvR10, BST11, RG11]. However, none of these works provides a goal inference algorithm suitable for our auditing task. In particular, they are each concerned with determining the probability that a sequence of actions are for a purpose, whereas we are concerned with whether an action *or use of information* could be for a purpose. Thus, we must develop a formalism for information use and a method of determining when a plan depends upon information use. We must also concern ourselves with the soundness of our audit algorithm rather than its accuracy in terms of a predicted probability.

1.5 Summary of Contributions

To argue our thesis, we offer the following novel contributions:

1. A formal treatment of purpose restrictions on actions using a planning-based formalism of purpose;

2. A semantic formalism of when information use is for a purpose;

3. An empirical validation that our planning-based formalism closely corresponds to how people understand the word "purpose" as used in purpose restrictions;

4. An algorithm and its implementation for auditing employing our formalism;

5. The application of our formalism to aid the understanding of privacy concerns found in the healthcare domain; and

6. The characterization of previous policy enforcement methods in our formalism and a comparative study of their expressiveness.

We believe that these contributions together are sufficient to demonstrate our thesis. In particular, the first three contributions illustrate that planning can formalize purpose restrictions. The last three illustrate that our formalism may aid automated auditing and analysis. Our success, however, must be qualified by the limited ability of our formalism to handle multiple purposes and the intricacies of human planning.

Although motivated by our goal to formalize the notions of *use* and *purpose* prevalently found in privacy policies, our work is more generally applicable to a broad range of policies, such as fiscal policies governing travel reimbursement or statements of ethics proscribing conflicts of interest.

1.6 Structure of Dissertation

Chapter 2 discusses when a sequence of actions is for a purpose. In Section 2.1, we present a formalism providing a semantics to purpose restrictions based upon planning with MDPs. Section 2.2 provides an auditing method and discusses the ramifications of the auditor observing only the behaviors of the auditee and not the underlying planning process of the auditee that resulted in these behaviors. We show that in some circumstances, the auditor can still acquire enough information to determine that the auditee violated the privacy policy. To do so, the auditor must first use our MDP model to construct all the possible behaviors that the privacy policy allows and then compare it with all the behaviors of the auditee that could have resulted in the observed auditing log. Section 2.3 presents an algorithm for auditing based on our formal definitions, illustrating the relevance of our work.

In Chapter 3, we extend our formalism to answer the question of when *information use* is for a purpose. Many uses of information may be modeled as an action, which makes the formalism of Chapter 2 applicable. However, this formalism cannot detect when information is used by the planning process itself. Thus, we extend the formalism to use Partially Observable Markov Decision Processes (POMDPs) that can capture such information usage. The explicitness of partial observations in the POMDP model allows us to consider how the agent would plan if some observations were conflated to ignore information of interest. We provide an algorithm for auditing that tests whether an agent uses information for a purpose by comparing the behaviors of the agent to the behaviors it would manifest had it planned its actions in this simulated state of ignorance.

To validate our work, we consider its application and perform an empirical study. In Chapter 4, we address a concern in the healthcare domain involving the rise of Regional Health Information Organizations (RHIOs). We use the formalism of Chapter 2 to create an operating procedure that encourages compliance with a purpose restriction.

In Chapter 5, we present the results of a survey testing how people understand the word "purpose". The survey compares our planning-based approach to the prior approach based on whether an action improves the satisfaction of a purpose. We find that our approach matches the survey participants' responses much more closely than the prior approach.

Most auditees are actually interested in multiple purposes and select plans that simultaneously satisfy as many of the desired purposes as possible. Handling the interactions among purposes complicates our semantics. In particular, actions selected by a single plan may be for different purposes. In Chapter 6, we present examples showing when our semantics can extend to handle multiple purposes and when difficulties arise in determining which purposes an action is for. Currently, the state-of-the-art in the understanding of human planning limits our abilities to improve upon our semantics. However, as this understanding improves, one may replace our formalism based on MDPs and POMDPs with more detailed ones while retaining our general framework of defining purpose restrictions in terms of planning.

Chapter 7 discusses related work. Even without a formalism for multiple purposes, our work is sufficient to put the previous work on enforcing privacy policies on firm semantic ground. In Section 7.1, we use our formalism to discuss the strengths and weaknesses of each such approach. In particular, we find that each approach enforces the policy given the set of all possible allowed behaviors, which is a set that our approach can construct. We also compare the previous auditing approaches, which differ in their trade-offs between auditing complexity and accuracy of representing this set of behaviors. The remaining sections discuss works related to ours by methodology rather than goals.

We end in Chapter 8 by presenting interesting directions for future work and conclusions. Appendix A provides additional background on POMDPs. Appendix B presents details about the empirical study. Appendix C summarizes notation.

Chapter 2

Action for a Purpose

2.1 Planning for a Purpose

In this section, we present a formalism for planning that accounts for quantitative purposes, probabilistic systems and non-redundancy. We first review Markov Decision Processes (MDPs)—a natural model for planning with probabilistic systems. In general, an agent planning for some purpose constructs an MDP to help select its actions. The MDP models the agent's environment and how the agent's actions affect the environment's state. We use the reward function of the MDP to quantify the degree of satisfaction of a purpose upon taking an action from a state. The agent selects a plan that determines for each state, the action that the agent will perform if the agent reaches that state. The plan the agent selects optimizes the expected total discounted reward (degree of purpose satisfaction) under the MDP.

We then develop a stricter definition of optimal than used with standard MDPs. We use this definition to create models we call "NMDPs" for *Non-redundant MDPs*. In addition to requiring that strategies optimize the expected total discounted reward, NMDPs exclude strategies that employ redundant actions that neither decrease nor increase the total reward. We end with an example illustrating the use of an NMDP to model an audited environment.

2.1.1 Markov Decision Processes

An MDP may be thought of as a probabilistic automaton where each transition is labeled with a reward in addition to an action. Rather than having accepting or goal states, the "goal" of an MDP is to maximize the total reward over time. Furthermore, we distinguish between the MDP, which is a model of an environment, and the agent, which is an entity using the model to select its actions. Thus, while it is convenient to speak informally of actions arising from an MDP, strictly speaking actions are performed by an agent because of the agent's use of the MDP model to select these actions.

To define partially observable MDPs, let $\mathsf{Dist}(X)$ denote the space of all distributions over the set X. That is, $f \in \mathsf{Dist}(X)$ is a function from X to the reals between 0 and 1 that obeys the standard of axioms of probability theory making it a distribution over X. An MDP is a tuple $m = \langle \mathcal{S}, \mathcal{A}, t, r, \gamma \rangle$ where

- $\mathcal{S}$ is a set of states;
- $\mathcal{A}$ is a set of actions;

- $t : \mathcal{S} \times \mathcal{A} \to \mathsf{Dist}(\mathcal{S})$, a transition function from a state and an action to a distribution over states;
- $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, a reward function; and
- γ, a discount factor such that $0 < \gamma < 1$.

where $\mathbb{R}$ is the set of real numbers. For each state s in $\mathcal{S}$, the agent using the MDP to plan selects an action a from $\mathcal{A}$ to perform. Upon performing the action a in the state s, the agent receives the reward $r(s, a)$. The environment then transitions to a new state s' with probability $\mu(s')$ where μ is the distribution provided by $t(s, a)$. The goal of the agent is to select actions to maximize its expected total discounted reward $\mathbb{E}\left[\sum_{i=0}^{\infty} \gamma^i \rho_i\right]$ where $i \in \mathbb{N}$ (the set of natural numbers) ranges over time modeled as discrete steps, ρ_i is the reward at time i, and the expectation is taken over the probabilistic transitions. The discount factor γ accounts for the preference of people to receive rewards sooner than later. It may be thought of as similar to inflation. We require that $\gamma < 1$ to ensure that the expected total discounted reward is bounded.

We formalize the agent's plan as a *stationary strategy* (commonly called a "policy", but we reserve that word for privacy policies). A stationary strategy is a function σ from the state space $\mathcal{S}$ to the set $\mathcal{A}$ of actions (i.e., $\sigma : \mathcal{S} \to \mathcal{A}$) such that at a state s in $\mathcal{S}$, the agent always selects to perform the action $\sigma(s)$. The value of a state s under a strategy σ is

$$v_m(\sigma, s) = \mathbb{E}\left[\sum_{i=0}^{\infty} \gamma^i r(s_i, \sigma(s_i))\right]$$

(v_m is typically written as V_m, but we reserve V_m for POMDPs in Chapter 3.) The Bellman equation [Bel52] shows that

$$v_m(\sigma, s) = r(s, \sigma(s)) + \gamma \sum_{s' \in \mathcal{S}} t(s, \sigma(s))(s') * v_m(\sigma, s')$$

A strategy σ^* is optimal if and only if for all states s, $v_m(\sigma^*, s) = \max_\sigma v_m(\sigma, s)$. At least one optimal policy always exists (see, e.g., [RN03]). Furthermore, if σ^* is optimal, then

$$\sigma^*(s) \in \underset{a \in \mathcal{A}}{\mathsf{argmax}} \left[r(s, a) + \gamma \sum_{s' \in \mathcal{S}} t(s, \sigma(s))(s') * v_m(\sigma, s') \right]$$

We denote this set of optimal strategies as $\mathsf{opt}(\langle \mathcal{S}, \mathcal{A}, t, r, \gamma \rangle)$, or when the transition system is clear from context, as $\mathsf{opt}(r)$. Such strategies are sufficient to maximize the agent's expected total discounted reward despite only depending upon the current state of the MDP.

Under this formalism, the auditee plays the role of the agent optimizing the MDP to plan. We presume that each purpose may be modeled as a reward function. That is, we assume the degree to which a purpose is satisfied may be captured by a function from states and actions to a real number. The higher the number, the higher the degree to which that purpose is satisfied. When the auditee wants to plan for a purpose p, it uses a reward function, r^p, such that $r^p(s, a)$ is the degree to which taking the action a from state s aids the purpose p. We also assume that the expected total discounted reward can capture the degree to which a purpose is satisfied over time. We say that the auditee plans *for* the purpose p when the auditee adopts a strategy σ that is optimal for the MDP $\langle \mathcal{S}, \mathcal{A}, t, r^p, \gamma \rangle$.

Executions and Behaviors. Given the strategy σ and the actual results of the probabilistic transitions yielded by t, the agent exhibits an *execution*. We represent this execution as an infinite sequence $e = [s_1, a_1, s_2, a_2, \ldots]$ of alternating states and actions starting with a state, where s_i is the ith state that the agent was in and a_i is the ith action the agent took, for all i in $\mathbb{N}$. We call a finite prefix b of an execution e a *behavior*.

Not every sequence of states and actions is a possible execution of the agent under an MDP. For an execution to be possible under an MDP, it must be consistent with some strategy and the transitions relation t. We say an execution e is *consistent* with a strategy σ if and only if $a_i = \sigma(s_i)$ for all i in $\mathbb{N}$ where a_i is the ith action in e and s_i is the ith state in e. A behavior is consistent with a strategy if it can be extended to an execution consistent with that strategy.

To determine whether an execution is possible under t, let a *contingency* κ be a function from $\mathcal{S} \times \mathcal{A} \times \mathbb{N}$ to $\mathcal{S}$ such that $\kappa(s, a, i)$ is the state that results from taking the action a in the state s as the ith action. We say that a contingency κ is *consistent* with an MDP if and only if κ only picks states to which the transition function t of the MDP assigns a non-zero probability to (i.e., for all s in $\mathcal{S}$, a in $\mathcal{A}$, and i in $\mathbb{N}$, $t(s, a)(\kappa(s, a, i)) > 0$).

Given an MDP m, let $m(s, \kappa)$ be the possibly infinite state model that results of having κ resolve all the probabilistic choices in m and having the model start in state s. Let $m(s, \kappa, \sigma)$ denote the execution that results from using the strategy σ and state s in the non-probabilistic model $m(s, \kappa)$. Formally, $m(s, \kappa, \sigma) = [s_1, a_1, s_2, a_2, \ldots]$ where $s_1 = s$ and for all $i \in \mathbb{N}$, $a_i = \sigma(s_i)$ and $s_{i+1} = \kappa(s_i, a_i, i)$.

Consistent contingencies capture the idea of possible executions. Formally, we say that an execution $e = [s_1, a_1, s_2, a_2, \ldots]$ is *possible* for m if and only if there exists a state s of m, a contingency κ consistent with m, and a strategy σ for m such that $e = m(s, \kappa, \sigma)$ Similarly, we say that a behavior $b = [s_1, a_1, \ldots, s_n, a_n]$ is *possible* for m if and only if there exists a state s of m, a contingency κ consistent with m, and a strategy σ for m such that $b \sqsubset m(s, \kappa, \sigma)$ where $\sqsubset$ denotes the proper-prefix relation. The following lemma reduces the global property of a behavior being possible for an MDP to local properties of the MDP.

Lemma 1. *For all MDPs m and behaviors $b = [s_1, a_1, \ldots, s_n, a_n] \in (\mathcal{S} \times \mathcal{A})^*$, b is possible for m if and only if for all $i < n$, $t(s_i, a_i)(s_{i+1}) > 0$ and for all $i \leq n$ and $j \leq n$, $s_i = s_j$ implies that $a_i = a_j$.*

Proof. Suppose that b is possible for m. Then, there exists a state s of m, a contingency κ consistent with m, and a strategy σ for m such that $b \sqsubset m(s, \kappa, \sigma)$. Since $b \sqsubset m(s, \kappa, \sigma)$, for all $i < n$, $\kappa(s_i, a_i, i) = s_{i+1}$. Since κ is consistent with m, for all $i < n$, $t(s_i, a_i)(s_{i+1}) > 0$. Since σ is stationary, $a_i = \sigma(s_i) = \sigma(s_j) = a_j$ for all $i, j \leq n$ such that $s_i = s_j$.

Suppose that for all $i < n$, $t(s_i, a_i)(s_{i+1}) > 0$ and for all $i \leq n$ and $j \leq n$, $s_i = s_j$ implies that $a_i = a_j$. Let $s = s_1$. Let σ be some strategy such $\sigma(s_i) = a_i$ for all $i \leq n$. Such a σ exists since $s_i = s_j$ implies that $a_i = a_j$ for all $i \leq n$ and $j \leq n$. Let κ be some contingency consistent with m such that for all $i < n$, $\kappa(s_i, a_i, i) = s_{i+1}$. Such a κ exists since for all $i < n$, $t(s_i, a_i)(s_{i+1}) > 0$. $b \sqsubset m(s, \kappa, \sigma)$. □

2.1.2 Non-redundancy

MDPs do not require that strategies be non-redundant. Even given that the auditee had an execution e from using a strategy σ in $\mathsf{opt}(r^p)$, some actions in e might not be *for* the purpose p. The reason is that some actions may be redundant despite being costless. The MDP optimization criterion behind opt prevents redundant actions from delaying the achievement of a goal as the reward associated with that goal would be further discounted making such redundant actions sub-optimal. However, the optimization criterion is not affected by redundant actions when they appear after all actions that provide non-zero rewards. Intuitively, the hypothetical agent planning only for the purpose in question would not perform such unneeded actions even if they have zero reward. Thus, to create our formalism of non-redundant MDPs (NMDPs), we replace opt with a new optimization criterion nopt that prevents these redundant actions while maintaining the same transition structure as a standard MDP.

To account for redundant actions, we must first contrast such actions with doing nothing. Thus, we introduce a distinguished action stop that stands for stopping and doing nothing. For all states s, stop labels a transition with zero reward (i.e., $r(s, \mathsf{stop}) = 0$) that is a self-loop (i.e., $t(s, \mathsf{stop})(s) = 1$). (We could put stop on only the subset of states that represent possible stopping points by slightly complicating our formalism.) Since we only allow deterministic stationary strategies and stop only labels self-loops, this decision is irrevocable: once the agent stops and does nothing, the agent does nothing forever. As selecting to do nothing results in only zero rewards henceforth, it may be viewed as stopping with the previously acquired total discounted reward.

Proposition 1. *For all NMDPs m, strategies σ for m, and states s, if $\sigma(s) = \mathsf{stop}$, then $v_m(\sigma, s) = 0$.*

Proof.

$$v_m(\sigma, s) = \mathbb{E}\left[\sum_{i=0}^{\infty} \gamma^i r(s_i, \sigma(s_i))\right]$$

where s_i is the ith state that the environment modeled by the NMDP enters starting with $s = s_0$.

Proof by induction shows that for all i, $s_i = s$. The base case follows from the definition of s_0. For the inductive case, the inductive hypothesis shows that $s_i = s$. $s_{i+1} = s'$ with probability $t(s_i, \sigma(s_i))(s') = t(s, \sigma(s))(s') = t(s, \mathsf{stop})(s') = \mathsf{degen}(s)$ by the definition of NMDPs where $\mathsf{degen}(s)(s'') = 1$ if and only if $s'' = s$ and is equal to 0 for all other s''. Thus, with certainty, $s_{i+1} = s$.

Thus,

$$v_m(\sigma, s) = \mathbb{E}\left[\sum_{i=0}^{\infty} \gamma^i r(s_i, \sigma(s_i))\right] = \mathbb{E}\left[\sum_{i=0}^{\infty} \gamma^i r(s, \mathsf{stop})\right] = \mathbb{E}\left[\sum_{i=0}^{\infty} \gamma^i 0\right] = 0$$

□

We use the idea of stopping and doing nothing to make formal when one execution contains more actions than another despite both being of infinite length. Given an execution e, let $\mathsf{active}(e)$ denote the prefix of e before the first instance of stop. $\mathsf{active}(e)$ will be equal to e in the case where e does not contain stop. An execution e_1 is a *proper sub-execution* of an execution e_2 if and only if $\mathsf{active}(e_1) \sqsubset \mathsf{active}(e_2)$ where $\sqsubset$ is the proper prefix relation. (We also use $\sqsubseteq$ for the prefix-or-equal relation.) Note if e_1 does not contain the stop, it cannot be a proper sub-execution of any execution.

We use contingencies to compare strategies. Given two strategies σ and σ', we write $\sigma' \prec \sigma$ if and only if for all contingencies κ and states s, $m(s, \kappa, \sigma')$ is a proper sub-execution of or equal to $m(s, \kappa, \sigma)$, and for at least one contingency κ' and state s', $m(s', \kappa', \sigma')$ is a proper sub-execution of $m(s', \kappa', \sigma)$. Intuitively, σ' proves that σ produces a redundant execution under κ' and s'. As we would expect, $\prec$ is a strict partial ordering on strategies.

Proposition 2. *$\prec$ is a strict partial order.*

Proof. The proper sub-execution relation is a strict partial order. This follows directly from the proper-prefix relation $\sqsubset$ being a strict partial order. We write $\lhd$ for *proper sub-execution* and $\unlhd$ for *proper sub-execution or equal*.

Now, we show that $\prec$ is also a strict partial ordering.

- Irreflexivity: for no σ is $\sigma \prec \sigma$. For $\sigma \prec \sigma$ to be true, there would have to exist a $\sigma \in \mathsf{opt}$ such that for at least one contingency κ' and s', $m(s', \kappa', \sigma')$ is a proper sub-execution of itself. However, this is impossible since the sub-execution relation is strict partial order.

- Asymmetry: for all σ_1 and σ_2, if $\sigma_1 \prec \sigma_2$, then it is not the case that $\sigma_2 \prec \sigma_1$. To show a contradiction, suppose $\sigma_1 \prec \sigma_2$ and $\sigma_2 \prec \sigma_1$ are both true. It would have to be the case that for all contingencies κ and states s, $m(s, \kappa, \sigma_1) \unlhd m(s, \kappa, \sigma_2)$ and $m(s, \kappa, \sigma_2) \unlhd m(s, \kappa, \sigma_1)$. Since $\lhd$ is a strict partial order, this implies that for all s and κ, $m(s, \kappa, \sigma_1) = m(s, \kappa, \sigma_2)$. Thus, there cannot exist a contingency κ' and state s' such that $m(s', \kappa', \sigma_2) \lhd m(s', \kappa', \sigma_1)$. Then $\sigma_2 \prec \sigma_1$ cannot be true, a contradiction.

- Transitivity: for all σ_1, σ_2, and σ_3, if $\sigma_1 \prec \sigma_2$ and $\sigma_2 \prec \sigma_3$, then $\sigma_1 \prec \sigma_3$. Suppose $\sigma_1 \prec \sigma_2$ and $\sigma_2 \prec \sigma_3$. Then for all for all contingencies κ and states s, $m(s, \kappa, \sigma_1) \unlhd m(s, \kappa, \sigma_2)$ and $m(s, \kappa, \sigma_2) \unlhd m(s, \kappa, \sigma_3)$. Since $\unlhd$ has transitivity, this implies that $m(s, \kappa, \sigma_1) \unlhd m(s, \kappa, \sigma_3)$ for all κ and s.

  Furthermore, it must be the case that there exists a contingency κ' and state s' such that $m(s', \kappa', \sigma_1) \lhd m(s', \kappa', \sigma_2)$. From above, $m(s', \kappa', \sigma_2) \unlhd m(s', \kappa', \sigma_3)$. Thus, by the transitivity of $\lhd$, $m(s', \kappa', \sigma_1) \lhd m(s', \kappa', \sigma_3)$ as needed. This implies that $\sigma_1 \prec \sigma_3$.

□

We define $\mathsf{nopt}(m)$ to be the subset of $\mathsf{opt}(m)$ holding only strategies σ such that for no $\sigma' \in \mathsf{opt}(m)$ does $\sigma' \prec \sigma$. $\mathsf{nopt}(m)$ is the set of non-redundant optimal policies.

The next lemma converts the requirements for being non-redundant from being about the executions of an MDP to being a local property. It uses the definition that $q^*_m(s, a) = r(s, a) + \gamma \sum_{s'} t(s, a)(s') * v^*_m(s')$ and the proof uses that $q_m(\sigma, s, a) = r(s, a) + \gamma \sum_{s'} t(s, a)(s') * v_m(\sigma, s')$. ($q^*$ is typically written as Q^*, but we reserve Q^* for POMDPs in Chapter 3.)

Lemma 2. *For all NMDPs m and σ in $\mathsf{opt}(m)$, σ is in $\mathsf{nopt}(m)$ if and only if for all states s such that $\sigma(s) \neq \mathsf{stop}$, $q^*_m(s, \sigma(s)) > 0$.*

Proof. If Direction. Suppose that for all s such that $\sigma(s) \neq \mathsf{stop}$, $q^*_m(s, \sigma(s)) > 0$. For the purposes of showing a contradiction, assume that $\sigma \notin \mathsf{nopt}(m)$. Then there exists σ' such that $\sigma' \in \mathsf{opt}(m)$ and $\sigma' \prec \sigma$. This implies that there exists κ' and s' such that $\mathsf{active}(m(s', \kappa', \sigma'))$ is a strict prefix of $\mathsf{active}(m(s', \kappa', \sigma))$. $m(s', \kappa', \sigma')$ must have the form $[s_1, a_1, s_2, \ldots, s_n, \mathsf{stop}, \ldots]$ and $m(s', \kappa', \sigma)$ must have the form $[s_1, a_1, s_2, \ldots, s_n, a_n, \ldots]$ for some n where $a_n \neq \mathsf{stop}$. Since $\sigma(s_n) = a_n \neq \mathsf{stop}$, $q^*_m(s, \sigma(s)) > 0$. Since both σ and σ' are in $\mathsf{opt}(m)$, $0 < q^*_m(s_n, \sigma(s)) = q^*_m(s_n, \sigma'(s)) = q^*_m(s_n, \mathsf{stop}) = q_m(\sigma, s_n, \mathsf{stop})$. However, by Proposition 1, $q_m(\sigma, s_n, \mathsf{stop}) = v_m(\sigma, s_n) = 0$, a contradiction. Thus, our assumption that $\sigma \notin \mathsf{nopt}(m)$ is false and σ is $\mathsf{nopt}(m)$.

Only-If Direction. Suppose σ is in $\mathsf{nopt}(m)$. Consider a state s such that $\sigma(s) \neq \mathsf{stop}$. Since σ is in $\mathsf{nopt}(m)$, there exists no σ' in $\mathsf{opt}(m)$ such that $\sigma' \prec \sigma$. That is, there exists no σ' such that σ' is in $\mathsf{opt}(m)$; for all contingencies κ' consistent with m, states s', $\mathsf{active}(m(s', \kappa', \sigma')) \sqsubseteq \mathsf{active}(m(s', \kappa', \sigma))$; and there exists a contingency κ'' and s'' such that $\mathsf{active}(m(s'', \kappa'', \sigma')) \sqsubset \mathsf{active}(m(s'', \kappa'', \sigma))$. That is, for all σ', either (1) σ' is not in $\mathsf{opt}(m)$; (2) it is not the case that for all contingencies κ' consistent with m, states s', $\mathsf{active}(m(s', \kappa', \sigma')) \sqsubseteq \mathsf{active}(m(s', \kappa', \sigma))$; or (3) it is not the case that there exists a contingency κ'' and a state s'' such that $\mathsf{active}(m(s'', \kappa'', \sigma')) \sqsubset \mathsf{active}(m(s'', \kappa'', \sigma))$.

We consider each of those three possibilities for σ' such that σ' is equal to σ except $\sigma'(s) = \mathsf{stop}$.

1. Case: σ' is not in $\mathsf{opt}(m)$. Since σ' is not in $\mathsf{opt}(m)$, there must exist $s^\dagger$ such that $\sigma'(s^\dagger) \notin \mathsf{argmax}_a\, q^*_m(s^\dagger, a)$. Since σ is in $\mathsf{opt}(m)$, for all $s' \neq s$, $\sigma'(s') = \sigma(s') \in \mathsf{argmax}_a\, q^*_m(s', a)$. Thus, $s^\dagger$ must be s. Since $\sigma'(s) \notin \mathsf{argmax}_a\, q^*_m(s, a)$ and $q^*_m(s, \sigma'(s)) = q^*_m(s, \mathsf{stop}) = 0$, $\max_a q^*_m(s, a) = v^*_m(s) > 0$. Since σ is in $\mathsf{opt}(m)$, $q^*_m(s, \sigma(s)) = v^*_m(s) > 0$.

2. Case: It is not the case that for all contingencies κ' consistent with m, and for all states s',

$$\mathsf{active}(m(s', \kappa', \sigma')) \sqsubseteq \mathsf{active}(m(s', \kappa', \sigma))$$

   For all κ' and s', $m(s', \kappa', \sigma)$ and $m(s', \kappa', \sigma')$ only differ if they reach the state s since σ and σ' only differ at the state s. If s is never reached, then $\mathsf{active}(m(s', \kappa', \sigma')) = \mathsf{active}(m(s', \kappa', \sigma))$. If s is reached, then $m(s', \kappa', \sigma')$ has the form $[s', a_1, s_2, a_2, \ldots, s, \mathsf{stop}, \ldots]$ and $m(s', \kappa', \sigma)$ has the form $[s', a_1, s_2, a_2, \ldots, s, \sigma(s), \ldots]$. Thus, either way, $\mathsf{active}(m(s', \kappa', \sigma')) \sqsubseteq \mathsf{active}(m(s', \kappa', \sigma))$. Thus, it is the case that for all contingencies κ' consistent with m, states s', $\mathsf{active}(m(s', \kappa', \sigma')) \sqsubseteq \mathsf{active}(m(s', \kappa', \sigma))$. Since this is a contradiction, the result trivially holds.

3. Case: There does not exist a contingency κ'' and a state s'' such that

$$\mathsf{active}(m(s'', \kappa'', \sigma')) \sqsubset \mathsf{active}(m(s'', \kappa'', \sigma))$$

   Let s'' be s. Then for all κ'', $m(s'', \kappa'', \sigma') = m(s, \kappa'', \sigma')$ has the form $[s, \mathsf{stop}, \ldots]$. $m(s, \kappa'', \sigma)$ has the form $[s, \sigma(s), \ldots]$ for some $\sigma(s) \neq \mathsf{stop}$. Thus, there exists a contingency κ'' and s'' such that $\mathsf{active}(m(s'', \kappa'', \sigma')) \sqsubset \mathsf{active}(m(s'', \kappa'', \sigma))$. Since this is a contradiction, the result trivially holds.

Thus, the result holds under all three possible cases. □

One of the reasons that the MDP model is useful is that an optimal strategy is guaranteed to exist. Fortunately, we can prove that $\mathsf{nopt}(m)$ is also guaranteed to be non-empty. One way to prove this result would use reasoning about well-ordered sets, Proposition 2, and the fact that space of all possible strategies is finite for NMDPs with finite state and action spaces. However, we provide a proof that depends more upon the structure of NMDPs since it can extend to NMDPs with infinite state spaces, which becomes important in the next chapter.

Theorem 1. *For all MDPs m, $\mathsf{nopt}(m)$ is not empty.*

Proof. $\mathsf{opt}(m)$ is non-empty (see, e.g., [RN03]). Let σ be some element of $\mathsf{opt}(m)$. Let σ' be σ except whenever $q^*_m(s, \sigma(s)) \leq 0$, $\sigma'(s) = \mathsf{stop}$. For any such state s, $q^*_m(s, \sigma(s)) \leq 0 = v_m(\sigma', s) = q_m(\sigma', s, \sigma'(s))$ by Proposition 1. For all other states $q^*_m(s, \sigma'(s)) = q^*_m(s, \sigma(s))$ since $\sigma'(s) = \sigma(s)$. In either case, $v^*_m(s) = q^*_m(s, \sigma(s))$ since s is optimal. Thus, for all states s, $v^*_m(s) \leq q^*_m(s, \sigma'(s))$. Thus, σ' is in $\mathsf{opt}(m)$. Furthermore, by construction, for all s, $\sigma'(s) \neq \mathsf{stop}$ implies that $q^*_m(s, \sigma'(s)) = q^*_m(s, \sigma(s)) > 0$. Thus, σ' is in $\mathsf{nopt}(m)$ by Lemma 2. □

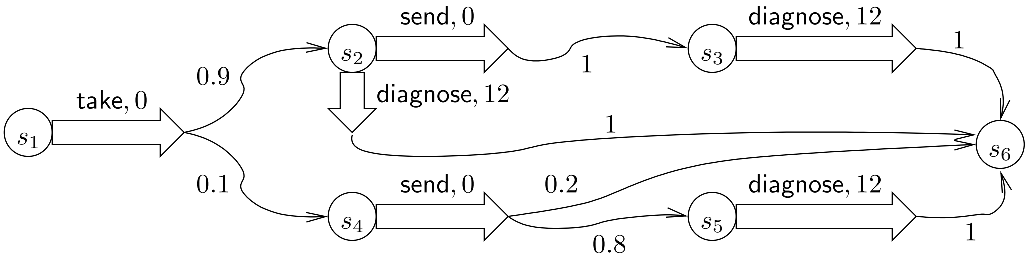

Figure 2.1: The MDP m_{ex1} that the physician used. Circles represent states, block arrows denote possible actions, and squiggly arrows denote probabilistic outcomes. Self-loops of zero reward under all actions, including the special action stop, are not shown.

2.1.3 Example: Modeling the Physician's Environment

Suppose an auditor is inspecting a hospital and comes across a physician referring a medical record to his own private practice for analysis of an X-ray as described in Section 1.2. As physicians may only make such referrals for the purpose of treatment (treat), the auditor may find the physician's behavior suspicious. To investigate, the auditor may formally model the hospital using our formalism.

After studying the hospital and how the physician's actions affect it, the auditor would construct the NMDP $m_{\mathsf{ex1}} = \langle \mathcal{S}_{\mathsf{ex1}}, \mathcal{A}_{\mathsf{ex1}}, t_{\mathsf{ex1}}, r^{\mathsf{treat}}_{\mathsf{ex1}}, \gamma_{\mathsf{ex1}} \rangle$ shown in Figure 2.1. The figure conveys all components of the NMDP except γ_{ex1}. For instance, the block arrow from the state s_1 labeled take and the squiggly arrows leaving it denote that after the agent performs the action take from state s_1, the environment will transition to the state s_2 with probability 0.9 and to state s_4 with probability of 0.1 (i.e., $t_{\mathsf{ex1}}(s_1, \mathsf{take})(s_2) = 0.9$ and $t_{\mathsf{ex1}}(s_1, \mathsf{take})(s_4) = 0.1$). The number over the block arrow further indicates the degree to which the action satisfies the purpose of treat. In this instance, it shows that $r^{\mathsf{treat}}_{\mathsf{ex1}}(s_1, \mathsf{take}) = 0$. This transition models the physician taking an X-ray. With probability 0.9, he is able to make a diagnosis right away (from state s_2); with probability 0.1, he must send the X-ray to his practice to make a diagnosis. Similarly, the transition from state s_4 models that his practice's test has a 0.8 success rate of making a diagnosis; with probability 0.2, no diagnosis is ever reached. For simplicity, we assume that all diagnoses have the same quality of 12 and that second opinions do not improve the quality; the auditor could use a different model if these assumptions are false. (For simplicity, in this example, we construe the meaning of the purpose *treatment* very narrowly. An auditor could construe it more broadly to include goals, such as research, that improve treatment in the long run.)

Using the model, the auditor computes $\mathsf{opt}(r^{\mathsf{treat}}_{\mathsf{ex1}})$, which consists of those strategies that maximizes the expected total discounted degree of satisfaction of the purpose of treatment where the expectation is over the probabilistic transitions of the model. $\mathsf{opt}(r^{\mathsf{treat}}_{\mathsf{ex1}})$ includes the appropriate strategy σ_1 where $\sigma_1(s_1) = \mathsf{take}$, $\sigma_1(s_4) = \mathsf{send}$, $\sigma_1(s_2) = \sigma_1(s_3) = \sigma_1(s_5) = \mathsf{diagnose}$, and $\sigma_1(s_6) = \mathsf{stop}$. Furthermore, $\mathsf{opt}(r^{\mathsf{treat}}_{\mathsf{ex1}})$ excludes the redundant strategy σ_2 that performs a redundant send where σ_2 is the same as σ_1 except for $\sigma_2(s_2) = \mathsf{send}$. Performing the extra action send delays the reward of 12 for achieving a diagnosis resulting in its discounted reward being $\gamma^2_{\mathsf{ex1}} * 12$ instead of $\gamma_{\mathsf{ex1}} * 12$ and, thus, the strategy is not optimal.

However, $\mathsf{opt}(r^{\mathsf{treat}}_{\mathsf{ex1}})$ does include the redundant strategy σ_3 that is the same as σ_1 except for $\sigma_3(s_6) = \mathsf{send}$. $\mathsf{opt}(r^{\mathsf{treat}}_{\mathsf{ex1}})$ includes this strategy despite the send actions from state s_6 being redundant since no positive rewards follow the send actions. Fortunately, $\mathsf{nopt}(r^{\mathsf{treat}}_{\mathsf{ex1}})$ does not include σ_3 since σ_1 is both in $\mathsf{opt}(r^{\mathsf{treat}}_{\mathsf{ex1}})$ and $\sigma_1 \prec \sigma_3$. To see that $\sigma_1 \prec \sigma_3$ note that for every contingency κ and state s, the $m_{\mathsf{ex1}}(s, \kappa, \sigma_1)$ has the form b followed by an finite sequence of stop (interleaved with the state s_6) for some finite prefix b. For the same κ, $m_{\mathsf{ex1}}(s, \kappa, \sigma_3)$ has the form b followed by an infinite sequence of send actions (interleaved with the state s_6) for the same b. Thus, $m_{\mathsf{ex1}}(s, \kappa, \sigma_1)$ is a proper sub-execution of $m_{\mathsf{ex1}}(s, \kappa, \sigma_3)$.

The above modeling implies that the strategy σ_1 can be for the purpose of treatment but σ_2 and σ_3 cannot be for treatment.

2.2 Auditing

In the above example, the auditor constructed a model of the environment in which the auditee operates. The auditor must use the model to determine if the auditee obeyed the policy. We first discuss this process for auditing exclusivity policy rules and revisit the above example. Then, we discuss the process for prohibitive policy rules. In the next section, we provide an auditing algorithm that automates comparing the auditee's behavior, as recorded in a log, to the set of allowed behaviors.

2.2.1 Auditing Exclusivity Rules

Suppose that an auditor would like to determine whether an auditee performed some logged actions *only for* the purpose p. The auditor can compare the logged behavior to the behavior that a hypothetical agent would perform when planning for the purpose p. In particular, the hypothetical agent selects a strategy from $\mathsf{nopt}(\langle \mathcal{S}, \mathcal{A}, t, r^p, \gamma \rangle)$ where $\mathcal{S}$, $\mathcal{A}$, and t models the environment of the auditee; r^p is a reward function modeling the degree to which the purpose p is satisfied; and γ is an appropriately selected discounting factor. If the logged behavior of the auditee would never have been performed by the hypothetical agent, then the auditor knows that the auditee violated the policy.

In particular, the auditor must consider all the possible behaviors the hypothetical agent could have performed. For a model m, let $\mathsf{nbehv}(r^p)$ represent this set where a finite prefix b of an execution is in $\mathsf{nbehv}(r^p)$ if and only if there exists a strategy σ in $\mathsf{nopt}(r^p)$, a contingency κ, and a state s such that b is a prefix of $m(s, \kappa, \sigma)$.

The auditor must compare $\mathsf{nbehv}(r^p)$ to the set of all behaviors that could have caused the auditor to observe the log that he did. We presume that the log ℓ was created by a process log that records features of the current behavior. That is, $\mathsf{log}\colon B \to L$ where B is the set of behaviors and L the set of logs, and $\ell = \mathsf{log}(b)$ where b is the prefix of the actual execution of the environment available at the time of auditing. The auditor must consider all the behaviors in $\mathsf{log}^{-1}(\ell) = \{\, b \in B \mid \mathsf{log}(b) = \ell \,\}$ as possible where log^{-1} is the inverse of the logging function. In the best case for the auditor, the log records the whole prefix b of the execution that transpired until the time of auditing, in which case $\mathsf{log}^{-1}(\ell) = \{\ell\}$. However, the log may be incomplete by missing actions, or may include only partial information about an action such as that it was one of a set of actions.

If $\mathsf{log}^{-1}(\ell) \cap \mathsf{nbehv}(r^p)$ is empty, then the auditor may conclude that the auditee did not plan for the purpose p, and, thus, violated the rule that auditee must only perform the actions recorded in ℓ for the purpose p; otherwise, the auditor must consider it possible that the auditee planned for the purpose p.

If $\mathsf{log}^{-1}(\ell) \subseteq \mathsf{nbehv}(r^p)$, the auditor might be tempted to conclude that the auditee surely obeyed the policy rule. However, as illustrated in the second example below, this is not necessarily true. The problem is that $\mathsf{log}^{-1}(\ell)$ might have a non-empty intersection with $\mathsf{nbehv}(r^{p'})$ for some other purpose p'. In this case, the auditee might have been actually planning for the purpose p' instead of p. Indeed, given the likelihood of such other purposes for non-trivial scenarios, we consider proving compliance practically impossible. However, this incapability is of little consequence: $\mathsf{log}^{-1}(\ell) \subseteq \mathsf{nbehv}(r^p)$ does imply that the auditee is behaving as though he is obeying the policy. That is, in the worse case, the auditee is still doing the right things even if for the wrong reasons.

2.2.2 Example: Auditing the Physician

Below we revisit the example of Section 2.1.3. We consider two cases. In the first, the auditor shows that the physician violated the policy. In the second, auditing is inconclusive.

Violation Found. Suppose after constructing the model as above in Section 2.1.3, the auditor maps the actions recorded in the access log ℓ_1 to the actions of the model m_{ex1}, and finds $\mathsf{log}^{-1}(\ell_1)$ holds only a single behavior: $b_1 = [s_1, \mathsf{take}, s_2, \mathsf{send}, s_3, \mathsf{diagnose}, s_6, \mathsf{stop}, s_6, \mathsf{stop}]$. Next, using $\mathsf{nopt}(r^{\mathsf{treat}}_{\mathsf{ex1}})$, as computed above, the auditor constructs the set $\mathsf{nbehv}(r^{\mathsf{treat}}_{\mathsf{ex1}})$ of all behaviors an agent planning for treatment might exhibit. The auditor would find that b_1 is not in $\mathsf{nbehv}(r^{\mathsf{treat}}_{\mathsf{ex1}})$.

To see this, note that every execution e_1 that has b_1 as a prefix is generated from a strategy σ such that $\sigma(s_2) = \mathsf{send}$. The strategy σ_2 from Section 2.1.3 is one such strategy. None of these strategies are members of $\mathsf{opt}(r^{\mathsf{treat}}_{\mathsf{ex1}})$ for the same reason as σ_2 is not a member. Thus, b_1 cannot be in $\mathsf{nbehv}(r^{\mathsf{treat}}_{\mathsf{ex1}})$. As $\mathsf{log}^{-1}(\ell) \cap \mathsf{nbehv}(r^{\mathsf{treat}}_{\mathsf{ex1}})$ is empty, the audit reveals that the physician violated the policy.

Inconclusive. Now suppose that the auditor sees a different log ℓ_2 such that $\mathsf{log}^{-1}(\ell_2) = \{b_2\}$ where $b_2 = [s_1, \mathsf{take}, s_4, \mathsf{send}, s_5, \mathsf{diagnose}, s_6, \mathsf{stop}, s_6, \mathsf{stop}]$. In this case, our formalism would not find a violation since b_2 is in $\mathsf{nbehv}(r^{\mathsf{treat}}_{\mathsf{ex1}})$. In particular, the strategy σ_1 from above produces the behavior b_2 under the contingency that selects the bottom probabilistic transition from state s_1 to state s_4 under the action take.

Nevertheless, the auditor cannot be sure that the physician obeyed the policy. For example, consider the NMDP m'_{ex1} that is m_{ex1} altered to use the reward function $r^{\mathsf{profit}}_{\mathsf{ex1}}$ instead of $r^{\mathsf{treat}}_{\mathsf{ex1}}$. $r^{\mathsf{profit}}_{\mathsf{ex1}}$ assigns a reward of zero to all transitions except for the send actions from states s_2 and s_4, to which it assigns a reward of 9. σ_1 is in $\mathsf{nopt}(r^{\mathsf{profit}}_{\mathsf{ex1}})$ meaning that not only the same actions (those in b_2), but even the exact same strategy can be either for the allowed purpose treat or the disallowed purpose profit. Thus, if the physician did refer the record to his practice for profit, he cannot be caught as he has tenable deniability of his ulterior motive of profit.

2.2.3 Auditing Prohibitive Rules

In the above example, the auditor was enforcing the rule that the physician's actions be *only for* treatment. Now, consider auditing to enforce the rule that the physician's actions are *not for* personal profit. To obey this purpose restriction, the auditee need not have attempted to minimize the degree of satisfaction of the purpose. Rather the auditee, need merely to have ignored the prohibited purpose.

To audit for compliance with a rule prohibiting the purpose p, after seeing the log ℓ, the auditor could check whether $\mathsf{log}^{-1}(\ell) \cap \mathsf{nbehv}(r^p)$ is empty. If so, then the auditor knows that the policy was obeyed because the auditee could not have been planning for the purpose p. If not, then the auditor cannot prove

```
AuditNMDP($m = \langle \mathcal{S}, \mathcal{A}, t, r, \gamma \rangle,\ b = [s_1, a_1, s_2, a_2, \ldots, s_n, a_n]$):
if (ImpossibleMDP($m, b$))
  return true    //  behavior impossible for NMDP
$v^*_m$ := SolveMDP($m$)
for ($i := 1; i \leq n; i{+}{+}$):
  if ($\mathsf{q}^*(m, v^*_m, s_i, a_i) < v^*_m(s_i)$):
    return true    //  action suboptimal
  if ($\mathsf{q}^*(m, v^*_m, s_i, a_i) \leq 0$ and $a_i \neq \mathsf{stop}$):
    return true    //  action redundant
return false
```

Figure 2.2: The algorithm AuditNMDP. SolveMDP may be any MDP solving algorithm. Figure 2.3 shows ImpossibleMDP.

nor disprove a violation. In the above example, just as the auditor is unsure whether the actions were *for* the required purpose of treatment, the auditor is unsure whether the actions are *not for* the prohibited purpose of profit.

An auditor might decide to investigate some of the cases where $\mathsf{log}^{-1}(\ell) \cap \mathsf{nbehv}(r^p)$ is not empty. In this case, the auditor could limit his attention to only those possible violations of a prohibitive rule that cannot be explained away by some allowed purpose. For example, in the inconclusive example above, the physician's actions can be explained with the allowed purpose of treatment. As the physician has tenable deniability, it is unlikely that investigating his actions would be a productive use of the auditor's time. Thus, the auditor should limit his attention to those logs ℓ such that both $\mathsf{log}^{-1}(\ell) \cap \mathsf{nbehv}(r^{\mathsf{profit}}_{\mathsf{ex1}})$ is non-empty and $\mathsf{log}^{-1}(\ell) \cap \mathsf{nbehv}(r^{\mathsf{treat}}_{\mathsf{ex1}})$ is empty.

A similar additional check using disallowed purposes could be applied to enforcing exclusivity rules. However, for exclusivity rules, this check would identify cases where the auditee's behavior could have been either for the allowed purpose or a disallowed purpose. Thus, it would serve to find additional cases to investigate and increase the auditor's workload rather than reduce it. Furthermore, the auditee would have tenable deniability for these possible ulterior motives, making these investigations a poor use of the auditor's time.

2.3 Auditing Algorithm

We would like to automate the auditing process described above. To this end, we present in Figure 2.2 an algorithm AuditNMDP that aids the auditor in comparing the log to the set of allowed behaviors. The algorithm is closely related to a goal inference algorithm that use MDPs [BTS06, BST09], but our algorithm focuses on soundness rather than predictive ability. (See Section 7.4 for a more detailed discussion.)

Since we are not interested in the details of the logging process and would like to focus on the planning aspects of our semantics, we limit our attention to the case where $\mathsf{log}(b) = b$ (i.e., the log is simply the behavior of the auditee). However, future work could extend our algorithm to handle incomplete logs by constructing the set of all possible behaviors that could give rise to that log.

The algorithm presumes that the MDP m is finite. That is, both $\mathcal{S}$ and $\mathcal{A}$ are finite. As proved below (Theorem 2), AuditNMDP(m, b) returns *true* if and only if $\mathsf{log}^{-1}(b) \cap \mathsf{nbehv}(m)$ is empty. In the case of

```
ImpossibleMDP($m = \langle \mathcal{S}, \mathcal{A}, t, r, \gamma \rangle,\ b = [s_1, a_1, s_2, a_2, \ldots, s_n, a_n]$):
11 for ($i := 1; i \leq n; i$++):
12   if ($s_i \notin \mathcal{S}$):
13     return true       //  $s_i$ is not a state
14   if ($a_i \notin \mathcal{A}$):
15     return true       //  $a_i$ is not an action
16 for ($i := 1; i < n; i$++):
17   if ($t(s_i, a_i)(s_{i+1}) \leq 0$):
18     return true       //  $s_{i+1}$ unreachable from $s_i$
19   for ($j := i + 1; j \leq n; j$++):
20     if ($s_i = s_j$ and $a_i \neq a_j$):
21       return true       //  no stationary strategy could have produced the behavior
22 return false
```

Figure 2.3: The algorithm ImpossibleMDP. Returns whether the given behavior is possible for the given MDP.

an exclusivity rule, the auditor may conclude that the policy was violated when AuditNMDP returns *true*. In case of a prohibitive rule, the auditor may conclude the policy was obeyed when AuditNMDP returns *true*.

The algorithm operates by checking a series of local conditions of the NMDP m and behavior b that are equivalent to the global property of whether $\mathsf{log}^{-1}(b) \cap \mathsf{nbehv}(m)$ is empty (as proved by Lemma 3). First, AuditNMDP checks whether the behavior b is possible for m using the sub-routine ImpossibleMDP shown in Figure 2.3. ImpossibleMDP checks whether every state and action is valid (Lines 12 and 14), every state is reachable by the state proceeding it (Line 17), and that the same action is performed from equal states in b (Line 20).

Next, the AuditNMDP checks whether the behavior b is optimal (Line 05) and non-redundant (Line 07). To do so, AuditNMDP uses a sub-routine SolveMDP to compute v^*_m, which for each state s records $v^*_m(s)$, the optimal value for s. The fact that NMDPs are a type of MDP allows AuditNMDP to use any MDP optimization algorithm for SolveMDP, such as reducing the optimization to a system of linear equations [d'E63].

AuditNMDP uses a function q^* that computes q^* from v^*:

$$\mathsf{q}^*(m, v^*_m, s, a) = r(s_i, a_i) + \gamma \sum_{s' \in \mathcal{S}} t(s_i, a_i)(s') * v^*_m(s')$$

Thus, $\mathsf{q}^*(m, v^*_m, s, a)$ is equal to $q^*_m(s, a)$.

The essence of the algorithm is checking whether $\mathsf{log}^{-1}(\ell) \cap \mathsf{nbehv}(m)$ is empty. For simplicity, our algorithm presumes that $\mathsf{log}^{-1}(\ell)$ holds only one behavior. This restriction manifests itself in that each of the local checks (Lines 01, 05, and 07) only considers a single sequence of states and actions.

If $\mathsf{log}^{-1}(\ell)$ holds more than a single behavior but is a small set, then the auditor may run the algorithm for each behavior in $\mathsf{log}^{-1}(\ell)$. Alternatively, in some cases the set $\mathsf{log}^{-1}(\ell)$ may have structure that a modified algorithm could leverage. For example, if $\mathsf{log}^{-1}(\ell)$ is missing what action is taken at some states of the execution or only narrows down the taken action to a set of possible alternatives, a conjunction of constraints on the action taken at each state may identify the set. Furthermore, if the log only records some of the states reached by the auditee, the algorithm IMPOSSIBLEMDP could be changed to allow from such discontinuities.

2.3.1 Correctness

To prove correctness, we use the following lemma that allows us to reduce checking for violations to local properties of the NMDP and the auditee's behavior.

Lemma 3. *For an NMDP m, the behavior $b = [s_1, a_1, \ldots, s_n, a_n]$ is in $\mathsf{nbehv}(m)$ if and only if b is a possible behavior of m, and for all $i \leq n$, $q^*_m(s_i, a_i) = v^*_m(s_i)$ and $a_i \neq \mathsf{stop}$ implies that $q^*_m(s_i, a_i) > 0$.*

Proof. First, for the only-if direction, suppose $b \in \mathsf{nbehv}(m)$. Since b is in $\mathsf{nbehv}(m)$, there exists a state s, a contingency κ consistent with m, and strategy σ in $\mathsf{nopt}(m)$ such that $b \sqsubset m(s, \kappa, \sigma)$. Thus, b is possible since κ is consistent with with m. Since $b \sqsubset m(s, \kappa, \sigma)$, for all $i \leq n$, $\sigma(s_i) = a_i$. Since σ is in $\mathsf{nopt}(m)$, for all $i \leq n$, $q^*_m(s_i, a_i) = q^*_m(s_i, \sigma(s_i)) = v^*_m(s_i)$. Since σ is in $\mathsf{nopt}(m)$, by Lemma 2, for all s such that $\sigma(s) \neq \mathsf{stop}$, $q^*_m(s, \sigma(s)) > 0$. Thus, for all $i \leq n$, $\sigma(s_i) \neq \mathsf{stop}$, $q^*_m(s, \sigma(s)) > 0$.

Second, for the if direction, suppose b is a possible behavior of m, and for all $i \leq n$, $q^*_m(s_i, a_i) = v^*_m(s_i)$ and $a_i \neq \mathsf{stop}$ implies that $q^*_m(s_i, a_i) > 0$. By Theorem 1, $\mathsf{nopt}(m)$ is not empty. Let σ be some element of $\mathsf{nopt}(m)$. Let σ' be identical to σ except for all i, $\sigma'(s_i) = a_i$, which is well defined since b is possible. For all i, $q^*_m(s_i, \sigma'(s_i)) = q^*_m(s_i, a_i) = v^*_m(s_i)$ and $\sigma'(s_i) = a_i \neq \mathsf{stop}$ implies that $q^*_m(s_i, \sigma'(s_i)) = q^*_m(s_i, a_i) > 0$. For all other states s, $q^*_m(s_i, \sigma'(s)) = q^*_m(s, \sigma(s)) = v^*_m(s)$ and $\sigma'(s) \neq \mathsf{stop}$ implies that $q^*_m(s, \sigma'(s)) > 0$ by Lemma 2 since $\sigma'(s) = \sigma(s)$. Thus, for all s, $q^*_m(s, \sigma'(s)) = v^*_m(s)$, which implies that σ' is in $\mathsf{opt}(m)$. Furthermore, for all s, $\sigma'(s) \neq \mathsf{stop}$ implies that $q^*_m(s, \sigma'(s)) > 0$, which implies that σ' is in $\mathsf{nopt}(m)$ by Lemma 2.

By Lemma 1, b being possible implies that for all $i < n$, $t(s_i, a_i)(s_{i+1}) > 0$. Thus, there exists a contingency κ that is consistent with m such that $\kappa(s_i, a_i, i) = s_{i+1}$. Furthermore, $b \sqsubset m(s, \kappa, \sigma')$ for $s = s_1$. Thus, since σ' is in $\mathsf{nopt}(m)$, b is in $\mathsf{nbehv}(m)$. □

The above lemma combines with reasoning about the actual code of the program to yield its correctness. First, we prove the correctness of IMPOSSIBLEMDP as a lemma.

Lemma 4. *For all MDPs m and behaviors b,* IMPOSSIBLEMDP(m, b) *is a decision procedure for whether b is not a possible behavior of m.*

Proof. To show that IMPOSSIBLEMDP is a decision procedure, we must show that it always terminates, that b is not possible for m if and only if IMPOSSIBLEMDP(m, b) returns true, and that b is possible for m if and only if IMPOSSIBLEMDP(m, b) returns false.

To show that IMPOSSIBLEMDP terminates note that all the for loops involve a monotonically increasing counter (i or j) and that they all terminate after the counter reaches finite number (n or $n + 1$).

IMPOSSIBLEMDP returns true if and only if one of the following is true: (1) there exists $i \leq n$ such that s_i is not a state of m, (2) there exists $i \leq n$ such that a_i is not an action of m, (3) there exists $i < n$ such that $t(s_i, a_i)(s_{i+1}) \leq 0$, (4) there exists $i < n$ and j where $i < j \leq n$ such that $s_i = s_j$ and $a_i \neq a_j$. IMPOSSIBLEMDP returns false if and only if all of the conditions (1), (2), (3), and (4) are false. Conditions (1) and (2) are both false if and only if b is in $(\mathcal{S} \times \mathcal{A})^*$. Condition (3) is false if and only if for all $i < n$, $t(s_i, a_i)(s_{i+1}) > 0$. Condition (4) is false if and only if all $i \leq n$ and $j \leq n$, $s_i = s_j$ implies that $a_i = a_j$.

Thus, by Lemma 1, b is possible for m if and only if the conditions (1), (2), (3), and (4) are all false, which is exactly when IMPOSSIBLEMDP returns false. Furthermore, b is not possible for m if and only if one of the conditions (1), (2), (3), and (4) is true, which is exactly when IMPOSSIBLEMDP returns true. □

Theorem 2. *For all finite NMDPs m and behaviors b,* AUDITNMDP *is a decision procedure for whether $\mathsf{log}^{-1}(b) \cap \mathsf{nbehv}(m)$ is empty.*

Proof. To show that AUDITNMDP is a decision procedure, we must show that it always terminates, that $\mathsf{log}^{-1}(b) \cap \mathsf{nbehv}(m)$ is empty if and only if AUDITNMDP(m, b) returns true, and that $\mathsf{log}^{-1}(b) \cap \mathsf{nbehv}(m)$ is non-empty if and only if AUDITNMDP(m, b) returns false.

To show that AUDITNMDP terminates, note that SOLVEMDP is also guaranteed to terminate because m is finite. Thus, each iteration of the for loop terminates. Furthermore, n is a finite number and i monotonically increases toward it. Thus, the loop will execute only a finite number of times. Furthermore q^* will terminate since $\mathcal{S}$ is finite.

Now, we show that $\mathsf{log}^{-1}(b) \cap \mathsf{nbehv}(m)$ is empty if and only if AUDITNMDP(m, b) returns true. AUDITNMDP(m, b) returns true if and only if at least one of the following is true: (1) b is not possible (see Lemma 4), (2) there exists $i \leq n$ such that $\mathsf{q}^*(m, v^*_m, s_i, a_i) < v^*_m(s_i)$, (3) there exists $i \leq n$ such that $\mathsf{q}^*(m, v^*_m, s_i, a_i) \leq 0$ and $a_i \neq \mathsf{stop}$. At least one of the Conditions (1), (2), or (3) is true if and only if the following is false: b is a possible behavior of m, for all $i \leq n$, $q^*_m(s_i, a_i) = v^*_m(s_i)$ and $a_i \neq \mathsf{stop}$ implies that $q^*_m(s_i, a_i) > 0$. Thus, by Lemma 3, AUDITNMDP(m, b) returns true if and only if b is not in $\mathsf{nbehv}(m)$. Since $\mathsf{log}^{-1}(b) = \{b\}$, AUDITNMDP$(m, b)$ returns true if and only if $\mathsf{log}^{-1}(b) \cap \mathsf{nbehv}(m)$ is empty.

Since AUDITNMDP(m, b) always terminates and can only return true or false, and returns true if and only if $\mathsf{log}^{-1}(b) \cap \mathsf{nbehv}(m)$ is empty, AUDITNMDP(m, b) returns false if and only if $\mathsf{log}^{-1}(b) \cap \mathsf{nbehv}(m)$ is non-empty. □

2.3.2 Running Time

The running time of the algorithm is dominated by the MDP optimization conducted by SOLVEMDP. SOLVEMDP may be done exactly by reducing the optimization to a system of linear equations [d'E63]. Such systems may be solved in polynomial time [Kha79, Kar84]. However, in practice, large systems are often difficult to solve. Fortunately, a large number of algorithms for making iterative approximations exist whose running time depends on the quality of the approximation. (See [LDK95] for a discussion.) In the next section, we discuss an implementation using such a technique.

2.4 Approximation Algorithm and Implementation

Rather than implement the exact algorithm AUDITNMDP found in Section 2.3, we implemented an approximation algorithm using the standard value iteration algorithm to solve MDPs (see, e.g., [RN03]). The value iteration algorithm starts with an arbitrary guess of an optimal strategy for an MDP and the value of each state under that policy. With each iteration, the algorithm improves its estimation of the optimal strategy and its value. It continues to make successively more accurate estimations until the improvement between one iteration and next is below some threshold ϵ. At this point, the algorithm returns its estimations. The difference between its estimation of the value of each state under the optimal policy and the

```
AuditNMDPApprox($m = \langle \mathcal{S}, \mathcal{A}, t, r, \gamma \rangle,\ b = [s_1, a_1, s_2, a_2, \ldots, s_n, a_n]$):
if (ImpossibleMDP($m, b$))
  return true        // behavior impossible for NMDP
$\langle \mathsf{v}^*_{\mathsf{low}}, \mathsf{v}^*_{\mathsf{up}} \rangle :=$ SolveMDPApprox($m$)
for ($i := 1; i \leq n; i{+}{+}$):
  if ($\mathsf{q}^*(m, \mathsf{v}^*_{\mathsf{up}}, s_i, a_i) < \mathsf{v}^*_{\mathsf{low}}(s_i)$):
    return true        // action suboptimal
  if ($\mathsf{q}^*(m, \mathsf{v}^*_{\mathsf{up}}, s_i, a_i) \leq 0$ and $a_i \neq \mathsf{stop}$):
    return true        // action redundant
return false
```

Figure 2.4: The algorithm AuditNMDPApprox. SolveMDPApprox is an MDP approximation algorithm. Figure 2.3 shows ImpossibleMDP.

true value is bounded by $2\epsilon\gamma/(1-\gamma)$ where γ is the discount factor of the MDP [WB93, WB94]. Each iteration takes $O(|\mathcal{S}|^2 * |\mathcal{A}|)$ time. The number of iterations needed to reach convergence grows quickly in γ making the algorithm pseudo-polynomial time in γ and polynomial time in $|\mathcal{A}|$ and $|\mathcal{S}|$ [Tse90]. Despite the linear programming approach having better worst-case complexity, value iteration tends to perform well in practice. Using value iteration in our algorithm results in it having the same asymptotic running time of pseudo-polynomial in γ.

To maintain soundness, the approximate auditing algorithm differs from the exact algorithm to account for the approximations made by the value-iteration algorithm. Figure 2.4 shows a general framework for auditing with approximation algorithms. SolveMDPApprox is an approximation algorithm for solving MDPs. It returns lower and upper bounds on the value of $v^*_m(s, a)$ for each s and a. AuditNMDPApprox uses these bounds to soundly audit.

For example, the auditor may select to use value iteration for SolveMDPApprox. In this case, $\mathsf{v}^*_{\mathsf{low}}(s, a) = \mathsf{v}^*_{\mathsf{app}}(s, a) - 2\epsilon\gamma/(1-\gamma)$ and $\mathsf{v}^*_{\mathsf{up}}(s, a) = \mathsf{v}^*_{\mathsf{app}}(s, a) + 2\epsilon\gamma/(1-\gamma)$ where $\mathsf{v}^*_{\mathsf{app}}(s, a)$ is the value of the approximation returned by value iteration using ϵ for the accuracy parameter.

With these changes, the approximation algorithm is sound in that it will return *true* only when the original algorithm AuditNMDP solving the MDPs exactly would return *true*.

Theorem 3. *For all finite NMDPs m and behaviors b, if* AuditNMDPApprox(m, b) *returns true, then* $\mathsf{log}^{-1}(b) \cap \mathsf{nbehv}(m)$ *is empty.*

Proof. If AuditNMDPApprox(m, b) returns true, then one of the following is true: (1) ImpossibleMDP returns true, (2) there exists $i \leq n$ such that $\mathsf{q}^*(m, \mathsf{v}^*_{\mathsf{up}}, s_i, a_i) < \mathsf{v}^*_{\mathsf{low}}(s_i)$, or (3) there exists $i \leq n$ such that $\mathsf{q}^*(m, \mathsf{v}^*_{\mathsf{up}}, s_i, a_i) \leq 0$ and $a_i \neq \mathsf{stop}$. If (1) is true, then b is not a possible behavior of m by Lemma 4. If (2) is true, then for that i, $q^*_m(s_i, a_i) \neq v^*_m(s_i)$ since $q^*_m(s_i, a_i) \leq \mathsf{q}^*(m, \mathsf{v}^*_{\mathsf{up}}, s_i, a_i) < \mathsf{v}^*_{\mathsf{low}}(s_i) \leq v^*_m(s_i)$. If (3) is true, then for that i, $a_i \neq \mathsf{stop}$ does not imply that $q^*_m(s_i, a_i) > 0$ since $a_i \neq \mathsf{stop}$ and $q^*_m(s_i, a_i) \leq \mathsf{q}^*(m, \mathsf{v}^*_{\mathsf{up}}, s_i, a_i) \leq 0$. Thus, under each of these cases, Lemma 3 shows that $b = [s_1, a_1, s_2, a_2, \ldots, s_n, a_n]$ is not in $\mathsf{nbehv}(m)$. This fact implies that $\mathsf{log}^{-1}(b) \cap \mathsf{nbehv}(m)$ is empty since $\mathsf{log}^{-1}(b) = \{b\}$. □

AuditNMDPApprox is not complete: it may return *false* in cases where AuditNMDP would return *true*. These additional results of *false* mean that additional violations of exclusivity rules might go uncaught and additional compliance with prohibitive rules might go unproven. However, since *false* indicates an inconclusive audit, they do not alter soundness of the implementation.

When AUDITNMDPAPPROX returns *false*, the auditor may use a more accurate approximation algorithm for SOLVEMDPAPPROX in hopes that improving accuracy of the approximations will produce the conclusive response of *true*. For the value iteration algorithm, the auditor just needs to rerun the algorithm with a lower value for ϵ. There always exists a value of ϵ small enough to show that $\mathsf{q}^*(m, \mathsf{v}^*_{\mathsf{up}}, s_i, a_i) < \mathsf{v}^*_{\mathsf{low}}(s_i)$ when it is actually the case that $q^*_m(s_i, a_i) < v^*_m(s_i)$. However, when $q^*_m(s_i, a_i) = 0$, there will be no value of ϵ small enough to make $\mathsf{q}^*(m, \mathsf{v}^*_{\mathsf{up}}, s_i, a_i) \leq 0$ true. Thus, AUDITNMDPAPPROX using value iteration will never catch when $\mathsf{log}^{-1}(b) \cap \mathsf{nbehv}(m)$ is empty because an action of b is redundant but otherwise optimal ($v^*_m(s_i) = q^*_m(s_i, a_i) = 0$ but $a_i \neq \mathsf{stop}$ for some a_i).

We programmed our implementation in the Racket dialect of Scheme [FP10]. The implementation is available at:

`http://www.cs.cmu.edu/~mtschant/thesis/`

The implementation uses an explicit representation of the state and actions spaces. The transition and reward functions are represented using hash maps. Since we did not optimize the implementation, we did not benchmark its performance. However, in Section 4.5, we use it to aid understanding a complex example and report its performance in that section.

Chapter 3

Information Use for a Purpose

3.1 Actions Using Information

In Chapter 2, we saw how to formalize performing an action for a purpose. In this chapter, we consider how to formalize using information for a purpose. Before providing an overview of this chapter in Section 3.1.3, we present two examples that motivate our approach. The first example (Section 3.1.1) illustrates the difficulties of modeling information use as an action in the sense of Chapter 2 even when the information used is easily mapped to specific actions. After finding this approach unsatisfactory, we introduce Partially Observable Markov Decision Processes (POMDPs) to provide a formalization of information use. In the second example (Section 3.1.2), we show that POMDPs are sufficient even when the information used cannot be directly mapped to actions. Since this section is motivational, we defer all formalism to later sections.

3.1.1 Physician Example: Parametric Information Use

An auditee may perform actions that manipulate, save, transmit, collect, or otherwise involve information governed by a privacy policy. Consider the example in Section 2.1.3. Here the action send involves transmitting the information conveyed or revealed by the X-ray. Thus, the formalism presented in Chapter 2 may already deal with actions involving information. The auditor may use this formalism to determine that an auditee used information for a purpose by determining that the auditee performed an action involving that information for that purpose.

However, the involvement of information is not explicit. Rather, the auditor must track information with means outside the formal model. For example, the auditor might informally determine which actions involve information and supply suggestive names to identify them such as send. While the auditor may understand from context that the action send involves the information in the X-ray, the auditor may desire a more detailed model that makes this information usage explicit. Furthermore, when the usage of information by a system is unclear, the auditor needs a model of the system to determine its information usage.

Such a model could be constructed by modeling the different values that the X-ray could take on and modeling each as an input such as in work on noninterference [GM82]. However, this model would be disconnected from the formal models presented in the previous chapter: the MDP formalism is incompatible with the nondeterminism used in these models for inputs since such nondeterminism makes evaluating a policy impossible. Nevertheless, we may adapt such a model to the MDP setting by instead encoding

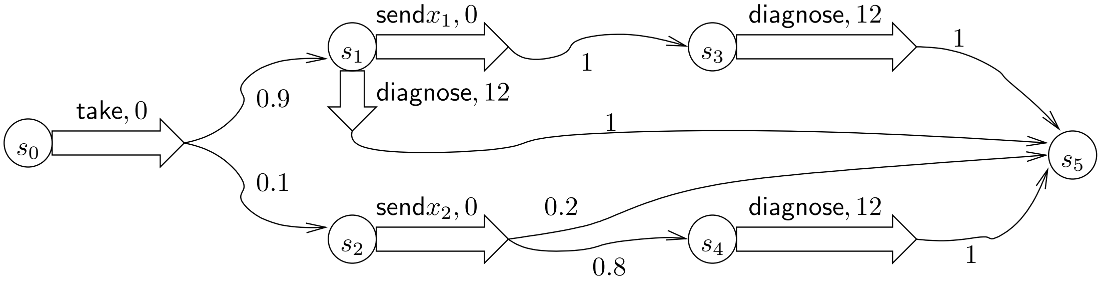

Figure 3.1: MDP making the involvement of information explicit for the physician example. This figure uses the conventions as earlier figures of MDPs, such as Figure 2.1.

these inputs in the probabilistic transitions of the model. Such an encoding requires assigning a probability distribution over the possible inputs.

Figure 3.1 shows such a model. In this example, from the point of view of the physician deciding whether to send the X-ray to the specialist, only two classes of X-ray exist: X-rays from which the physician himself can form a diagnosis and those from which he cannot. Thus, as a simplification, in our model, we employ only two different X-rays. (Alternatively, one can view these two X-rays as two classes of X-rays where an X-ray is put into one or the other class depending upon whether the physician can form a diagnosis from the X-ray.) The physician can form a diagnosis from the X-ray x_1, but not x_2. To see this difference in the model, note that, from state s_1, the action diagnose leads to a reward of 12 and to a new state s_5, whereas, from the state s_2, the action diagnosis is a self-loop of zero reward (which is inferred from the absence of the action labeling any arrow leaving the state s_2 in the figure). The single send action is replaced with one for each value of the X-ray.

Despite the X-rays x_1 and x_2 intuitively corresponding to inputs to the physician or observations made by the physician, the physician becoming aware of which of the two possible X-rays he has taken is not explicitly represented in the model as an input or an action. Rather, these inputs determine which of the *states* s_1 or s_2 the physician reaches. For example, input x_1 results in the transition to the state s_1. Since the physician may observe what state he is in, he may learn the value of the input (i.e., which X-ray he took). In this simple model, we assume that the physician can only send that X-ray. In some systems, the physician might have a choice of which X-ray to send.

Consider a physician who plans to send the X-ray to a specialist if and only if it is x_2. Using a formalization similar to noninterference [GM82], we can determine that his plan uses the X-ray. However, such a formalism would be unnatural as it relies on modeling *inputs*, which do not show up in the MDP model. Thus, we instead switch to the formalism of Partially Observable Markov Decision Processes (POMDPs). Under the POMDP model, the agent using the model to plan does not know *a priori* what state it is in. Rather the agent makes *observations* that adjusts its beliefs about its current state. The input of the X-ray would be modeled as such an observation. Furthermore, rather than modeling the uncertainty over which value of the X-ray will be produced as a probabilistic transition, a POMDP model represents such uncertainty as uncertainty over the initial state of the system.

Figure 3.2 shows a POMDP model of the above example. Unlike the MDP model (shown in Figure 3.1), the POMDP model explicitly represents the physician receiving the X-ray as an observation. The obser-

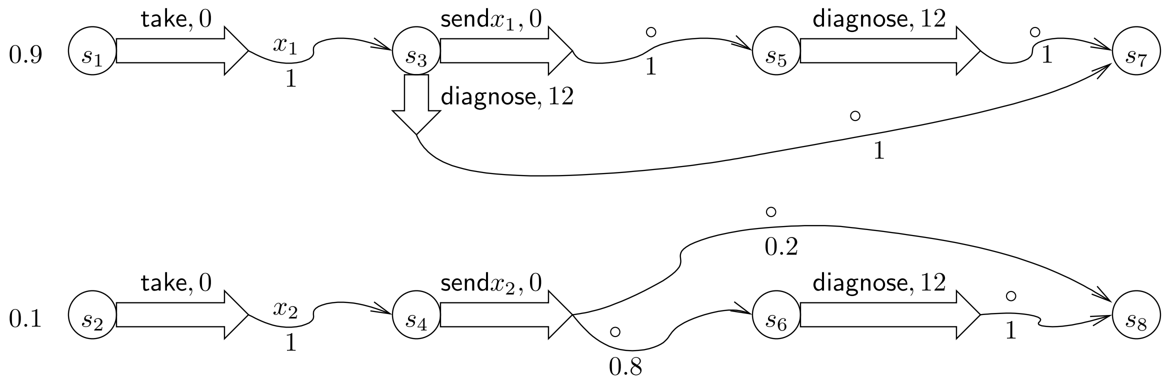

Figure 3.2: POMDP m_{phy} making the involvement of information explicit for the physician example. This figure is suggestive of the POMDP model of the physician example. While focusing on the key features of the POMDP, it not a complete representation. See Section 3.2.2 for the complete description of the POMDP m_{phy} and further discussion of the figure.
The figure follows the conventions of our figures showing MDPs: Circles represent states, block arrows denote possible actions, and squiggly arrows denote probabilistic outcomes including their probability (under the arrow); self-loops of zero reward are not shown. Additionally, each probabilistic outcome (squiggly arrow) is labeled with the observation that accompanies it (this convention does not generalize well for more complex POMDPs). To the left of the possible initial states s_1 and s_2 are the probabilities of the physician starting in each of these states. Strictly speaking, these probabilities are not part of the POMDP model, but rather part of physician's initial beliefs about the state of the POMDP. However, we include these beliefs in this figure since, in this example, the physician's initial beliefs are known to the auditor.

vations x_1 and x_2 that label the probabilistic transitions following the action take, which models taking the X-ray, represent which value of the X-ray the physician observes. The value seen by the physician is determined by which of the possible initial states is the actual initial state of the environment. Intuitively, each initial state represents a different condition the patient may have that will affect the value of the X-ray. As the physician does not know *a priori* what state it is in, he must take the X-ray to determine which of these conditions afflicts the patient. Note that the dummy observation $\circ$ labels transitions that produce no additional information other than that a transition occurred.

Intuitively, if the physician were to ignore whether he makes the observation x_1 or x_2 after taking the X-ray, then he will not learn which of the two possible conditions afflicts the patient. Furthermore, he will not learn whether he can make a diagnosis from the X-ray. To compensate, the physician will have to always send the X-ray to the specialist to ensure a diagnosis. This difference in behavior may enable an auditor to learn whether the physician used X-ray.

Up until now, we have only seen information affecting the action chosen by the physician parametrically. That is, after the physician observes x_2, it chooses the action sendx_2, an action labeled with the used information. This might lead an auditor to conclude that only actions labeled by a piece of information (such as x_2 labeling sendx_2) involves information. However, the parametric view of information use may be generalized. In some cases, the agent may use information to choose among completely different courses of action. For example, upon seeing a certain value for the X-ray, the physician may decide to perform emergency surgery rather than sending the X-ray for further analysis.

This distinction is similar to one made in works on checking programs for noninterference: the first example shown in Figure 3.2 (parametric) suggests *direct information flow* or a *data dependence* in which a variable takes on the value of some input. The second example involving emergency surgery (non-parametric) suggests *indirect information flow* or a *control dependence* in which a control statement affects the value of a variable by affecting whether an assignment to that variable executes (e.g., [FOW87]). While, formalisms such as noninterference do not apply to our optimization models (MDPs and POMDPs), we would like to ensure that our formalization of information use may handle all forms of information use. Fortunately, the POMDP model naturally captures both parametric and non-parametric use of information. The next example illustrates modeling non-parametric information flow with a POMDP.

3.1.2 Advertising Example: Non-Parametric Information Use

The previous example dealt with parametric information use in which the actions that involve information directly show that information (i.e., the actions are parametrically labeled with observations relevant to a class of information). In this example, the agent uses information but none of its actions directly show that information.

Consider an website attempting to determine which advertisement to show a website visitor. The website has access to a database of information about potential visitors that it can use to select advertisements. Since some advertisements are more effective for some demographics than others, it is in the website's interest to use this information.

However, a privacy policy governs what information the website may use for the purpose of marketing, including determining the advertisement to show the visitor. The policy states

> We will not use information you provide about your sex for the purpose of marketing.

Since the entry of the website's database for a visitor's sex is created from the information that the visitor provides to the website, the policy prohibits the use of the database's entry for the visitor's to select advertisements.

For simplicity, we assume that the only information relevant to advertising is the sex of the visitor. We further assume that the website is choosing among three advertisements: ad_1, ad_2, and ad_3. We presume ad_1 is the best for females and the worst for males (on average), ad_3 is the best for males and the worst for females, and ad_2 strikes a middle ground. We further presume that these advertisements are generic and do not directly contain any information about the visitor to whom the website shows it.

Figure 3.3 shows part a POMDP model m_{adv} of a such an advertising website. Unlike the POMDP m_{phy} of the physician example in Figure 3.2, the state space of this model is factored: it's a tuple with each component representing some factor about the state. This state space involves three factors (components of the tuple). The first factor is f if the visitor is a female and m if the visitor is male. The second factor shows what the database records about the visitor (with $\bot$ representing that the database does not have a record for the visitor). The third factor records what advertisement the website has shown to the visitor (with $\varnothing$ indicating that the website has not shown an advertisement). For example, the state $\langle \mathsf{f}, \bot, \mathsf{ad}_2 \rangle$ indicates that the visitor is a female, the database does not record her sex, and the website has shown her ad_2.

Intuitively, we would expect that the website will first perform the action lookup and show ad_1 to a female and ad_3 to a male. Our reasoning is that this plan maximizes the effectiveness of the advertisements.

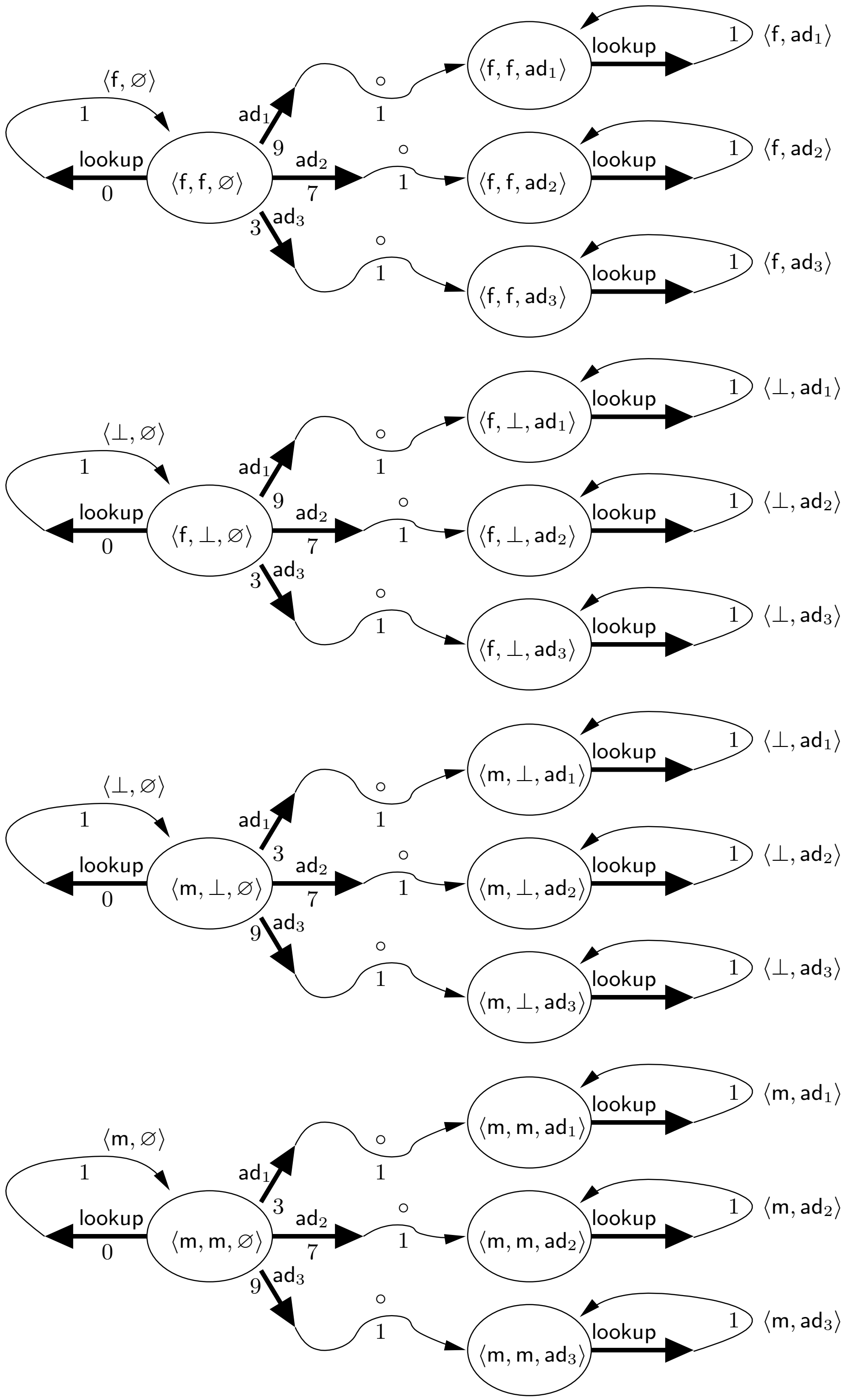
$\langle \mathsf{f}, \varnothing \rangle$
1
lookup
0
$\langle \mathsf{f}, \mathsf{f}, \varnothing \rangle$
ad_1
9
ad_2
7
ad_3
3
○
1
○
1
○
1
$\langle \mathsf{f}, \mathsf{f}, \mathsf{ad}_1 \rangle$
lookup
1
$\langle \mathsf{f}, \mathsf{ad}_1 \rangle$
$\langle \mathsf{f}, \mathsf{f}, \mathsf{ad}_2 \rangle$
lookup
1
$\langle \mathsf{f}, \mathsf{ad}_2 \rangle$
$\langle \mathsf{f}, \mathsf{f}, \mathsf{ad}_3 \rangle$
lookup
1
$\langle \mathsf{f}, \mathsf{ad}_3 \rangle$
$\langle \perp, \varnothing \rangle$
1
lookup
0
$\langle \mathsf{f}, \perp, \varnothing \rangle$
ad_1
9
ad_2
7
ad_3
3
○
1
○
1
○
1
$\langle \mathsf{f}, \perp, \mathsf{ad}_1 \rangle$
lookup
1
$\langle \perp, \mathsf{ad}_1 \rangle$
$\langle \mathsf{f}, \perp, \mathsf{ad}_2 \rangle$
lookup
1
$\langle \perp, \mathsf{ad}_2 \rangle$
$\langle \mathsf{f}, \perp, \mathsf{ad}_3 \rangle$
lookup
1
$\langle \perp, \mathsf{ad}_3 \rangle$
$\langle \perp, \varnothing \rangle$
1
lookup
0
$\langle \mathsf{m}, \perp, \varnothing \rangle$
ad_1
3
ad_2
7
ad_3
9
○
1
○
1
○
1
$\langle \mathsf{m}, \perp, \mathsf{ad}_1 \rangle$
lookup
1
$\langle \perp, \mathsf{ad}_1 \rangle$
$\langle \mathsf{m}, \perp, \mathsf{ad}_2 \rangle$
lookup
1
$\langle \perp, \mathsf{ad}_2 \rangle$
$\langle \mathsf{m}, \perp, \mathsf{ad}_3 \rangle$
lookup
1
$\langle \perp, \mathsf{ad}_3 \rangle$
$\langle \mathsf{m}, \varnothing \rangle$
1
lookup
0
$\langle \mathsf{m}, \mathsf{m}, \varnothing \rangle$
ad_1
3
ad_2
7
ad_3
9
○
1
○
1
○
1
$\langle \mathsf{m}, \mathsf{m}, \mathsf{ad}_1 \rangle$
lookup
1
$\langle \mathsf{m}, \mathsf{ad}_1 \rangle$
$\langle \mathsf{m}, \mathsf{m}, \mathsf{ad}_2 \rangle$
lookup
1
$\langle \mathsf{m}, \mathsf{ad}_2 \rangle$
$\langle \mathsf{m}, \mathsf{m}, \mathsf{ad}_3 \rangle$
lookup
1
$\langle \mathsf{m}, \mathsf{ad}_3 \rangle$

Figure 3.3: POMDP model m_{adv} of the advertising example. This figure uses the same conventions as our other POMDP figure shown in Figure 3.2. Due to space constraints, we do not show states that correspond to the database being incorrect, such as $\langle \mathsf{f}, \mathsf{m}, \varnothing \rangle$.

However, how the website will actually behave depends upon the website's initial beliefs. For example, if the website believes that all visitors are female or that the database is inaccurate, it will not bother to check the database.

The behavior we intuitively expect corresponds to the initial beliefs we implicitly presumed from the example description: that database is accurate (e.g., assigns a low probability to states such as $\langle \mathsf{f}, \mathsf{m}, \varnothing \rangle$), that the database is useful (e.g., assigns a higher probability to the state $\langle \mathsf{f}, \mathsf{f}, \varnothing \rangle$ than the state $\langle \mathsf{f}, \perp, \varnothing \rangle$), that website has not shown an advertisement to the visitor yet (i.e., assigns zero probability to states of the form $\langle g, d, \mathsf{ad}_i \rangle$), and that visitors are equally likely to be female or male. Under these presumptions, optimal strategies behave as we intuitively expect: The website is to first check whether the database contains information about the visitor. If the database records that the visitor is a female, then the website shows her ad_1. If it records a male, the website shows ad_3. If the database does not contain the visitor's sex (holds $\perp$), then the website shows ad_2.

3.1.3 Summary of Chapter

The reminder of this chapter makes the intuitions already introduced formal. In particular, we discuss using the POMDP model to audit purpose restrictions involving the use of information. To do so, we must provide a semantics of *information use*. To that end, we note that purpose restrictions typically do not govern the use of knowledge gained from information sources not mentioned by the policy even if this knowledge can also be inferred from information sources prohibited by the policy. For example, Yahoo! promises not to use the contents of a user's emails for the purpose of marketing. We expect this to prohibit Yahoo! from reading a user's emails to determine what advertisements to show him. However, we do not expect Yahoo! to avoid using for marketing knowledge of the user that it has collected by other means even if some of this knowledge is separately implied by the user's emails.

Thus, a purpose restriction limits the use of information from a source rather than some class of knowledge. Limiting information from a source includes limiting any direct observations of that source or inferences that would be impossible without such observations. Understanding these restrictions does not require epistemic models of knowledge (e.g., [FHMV95]) nor fine-grain inference control (e.g., [FJ02]). Rather, similar to how noninterference characterizes information use for computer programs [GM82], these restrictions require understanding how observations of information change the agent's behavior. However, whereas noninterference starts with the automaton model of programs, enforcing purpose restrictions requires understanding a purpose-driven planning agent with a model such as the POMDP model. The POMDP model allows us to model the agent's environment with the purpose in question defining the reward function of the POMDP (Section 3.2).

The explicitness of partial observations in the POMDP model allows us to formalize information use by considering how the agent would plan if some observations were conflated to ignore information of interest (Section 3.3). We do so by quotienting the space of observations by an equivalence relation that treats two observations as indistinguishable if they only differ by information whose use is prohibited by the purpose restriction. By ignoring this distinguishing information, we simulate ignorance of the information. Such quotienting is well-defined for POMDPs since observations only probabilistically constrain the space of possible current states of the agent's environment, and quotienting just decreases the accuracy of this constraining.

We test whether an agent uses information for a purpose by comparing the behaviors of the agent to the behaviors it would manifest had it planned its actions in this simulated state of ignorance (Section 3.4). We provide an auditing algorithm using our formalism to compare the behavior of an agent to how it would behave under such ignorance (Section 3.5). Our algorithm use an off-the-shelf approximation algorithm for POMDPs. Our algorithm automates much of the enforcement of purpose restrictions governing information use.

Throughout this chapter we employ the two examples already introduced. These examples together show that our formalism can handle both parametric and non-parametric information use.

3.2 Planning under Partial Observations

To model information use, we first present the formalism of Partially Observable Markov Decision Processes (POMDPs) [Son71]. In particular, we present previous work on the formal model and on how to reduce the optimization of POMDPs to the optimization of a related *belief MDP*. We then present two examples and discuss applying the idea of non-redundancy to this model.

In general, an agent planning for some purpose constructs a POMDP to help select its actions. The POMDP models the agent's environment and how its actions affects the environment's state and the satisfaction of the purpose it is pursuing. The agent selects a plan that optimizes the expected total discounted reward (degree of purpose satisfaction) under the POMDP. (For a survey, see [Mon82].)

3.2.1 Partially Observable Markov Decision Processes

A POMDP is a tuple $\langle \mathcal{S}, \mathcal{A}, t, r, \mathcal{O}, \nu, \gamma \rangle$ where

- $\mathcal{S}$ is a finite state space;
- $\mathcal{A}$, a finite set of actions;
- $t : \mathcal{S} \times \mathcal{A} \to \mathsf{Dist}(\mathcal{S})$, a transition function from a state and an action to a distribution over states;
- $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, a reward function;
- $\mathcal{O}$, a finite observation space containing any observations the agent may perceive while performing actions;
- $\nu : \mathcal{A} \times \mathcal{S} \to \mathsf{Dist}(\mathcal{O})$, a distribution over observations given an action and the state resulting from performing that action; and
- γ, a discount factor such that $0 \leq \gamma < 1$.

An execution of a POMDP m is an infinite interleaving of states, actions, and observations. For example, $[s_1, a_1, o_1, s_2, a_2, o_2, \ldots]$ is an execution in which the modeled agent started in the state s_1 from which the agent performs action a_1 causing the observation o_1 and the agent's environment transitions to state s_2 from which the agent performs action a_2 causing the observation o_2, and so forth.

An agent does not know *a priori* which of the possible states of the POMDP is the current state of its environment. Rather it holds beliefs about which state is the current state. In particular, the agent assigns a probability to each state s according to how likely the agent believes that the current state is the state s. A *belief state* β captures these beliefs as a distribution over states of $\mathcal{S}$ (i.e., $\beta \in \mathsf{Dist}(\mathcal{S})$). An agent's belief state is updated as it performs actions and makes observations. When an agent takes the action a and makes the observation o starting with the beliefs β, the agent develops the new beliefs β' (also a distribution over $\mathcal{S}$). $\beta'(s')$ is the probability that s' is the next state.

We define $\mathsf{update}_m(\beta, a, o)$ to equal the updated beliefs β'. β' assigns to the state s' the probability $\beta'(s') = \Pr[S'{=}s' \mid O{=}o, A{=}a, B{=}\beta]$ where S' is a random variable over next states, $B{=}\beta$ identifies the agent's current belief state as β, $A{=}a$ identifies the agent's current action as a, and $O{=}o$ identifies the observation the agent makes while performing action a as o. To compute the value of $\mathsf{update}_m(\beta, a, o)$, we may reduce it to a formula in terms of the POMDP model m as follows:

$$\mathsf{update}_m(\beta, a, o)(s') \tag{3.1}$$

$$= \Pr[S'{=}s' \mid O{=}o, A{=}a, B{=}\beta] \tag{3.2}$$

$$= \frac{\Pr[O{=}o \mid S'{=}s', A{=}a, B{=}\beta] \Pr[S'{=}s' \mid A{=}a, B{=}\beta]}{\Pr[O{=}o \mid A{=}a, B{=}\beta]} \tag{3.3}$$

$$= \frac{\Pr[O{=}o \mid S'{=}s', A{=}a, B{=}\beta] \sum_{s \in \mathcal{S}} \Pr[S{=}s \mid A{=}a, B{=}\beta] \Pr[S'{=}s' \mid A{=}a, B{=}\beta, S{=}s]}{\Pr[O{=}o \mid A{=}a, B{=}\beta]} \tag{3.4}$$

$$= \frac{\nu(a, s')(o) \sum_{s \in \mathcal{S}} \beta(s) * t(s, a)(s')}{\Pr[O{=}o \mid A{=}a, B{=}\beta]} \tag{3.5}$$

Line 3.4 follows since the POMDP can have only one current states making different possible current states mutually exclusive events. That is, for any two differing states s_1 and s_2, the event of the current state being s_1 (i.e., $S = s_1$) and the event of the current state being s_2 (i.e., $S = s_2$) are mutually exclusive. Since $\Pr[O{=}o \mid A{=}a, B{=}\beta]$ is independent of s', it may be treated as a normalization factor equal to

$$\sum_{s' \in \mathcal{S}} \nu(a, s')(o) \sum_{s \in \mathcal{S}} \beta(s) * t(s, a)(s') \tag{3.6}$$

Similar to MDPs, the agent does not need to track its history of actions and observations independently of its beliefs as such beliefs are a sufficient statistic. Thus, the agent's strategies need only deal with beliefs and are formalized as a function from beliefs to actions. That is, the space of possible strategies that agent may employ given a POMDP is $\mathsf{Dist}(\mathcal{S}) \to \mathcal{A}$.

The goal of the agent is find the optimal strategy. By the Bellman equation [Bel52], the expected value of a belief state β under a strategy σ for $m = \langle \mathcal{S}, \mathcal{A}, t, r, \mathcal{O}, \nu, \gamma \rangle$ is

$$V_m(\sigma, \beta) = R_m(\beta, \sigma(\beta)) + \gamma \sum_{o \in \mathcal{O}} N_m(\beta, \sigma(\beta))(o) * V_m(\sigma, \mathsf{update}_m(\beta, \sigma(\beta), o))$$

where R and N are r and ν raised to work over beliefs:

$$R_m(\beta, a) = \sum_{s \in \mathcal{S}} \beta(s) * r(s, a) \tag{3.7}$$

and

$$N_m(\beta, a)(o) = \Pr[O{=}o \mid B{=}\beta, A{=}a] = \sum_{s \in \mathcal{S}} \beta(s) * \sum_{s' \in \mathcal{S}} t(s, a)(s') * \nu(a, s')(o) \tag{3.8}$$

A strategy σ is optimal if for every state β, it is optimal (i.e., if for all β, $V_m(\sigma, \beta)$ equals $V^*_m(\beta) = \max_{\sigma'} V_m(\sigma', \beta)$).

We are also interested in the optimal value of a performing an action from a belief state. For a POMDP m, we define the quality of an action given a belief state as follows:

$$Q^*_m(\beta, a) = R_m(\beta, a) + \gamma \sum_{o \in \mathcal{O}} N_m(\beta, a)(o) * V^*_m(\mathsf{update}_m(\beta, \sigma(\beta), o))$$

An action a is optimal for a belief state if and only if $Q^*_m(\beta, a) = V^*_m(\beta)$.

The theory of POMDPs reduces the process of finding the optimal strategy for a POMDP to that of finding the optimal strategy for a related MDP that uses belief states as its state space (e.g., [Son78]). This reduction allows us to reuse the theory of MDPs to find the optimal strategies of POMDPs. Let bmdp be a function that takes a POMDP and produces its *belief MDP*. We define bmdp as follows:

$$\mathsf{bmdp}(\langle \mathcal{S}, \mathcal{A}, t, r, \mathcal{O}, \nu, \gamma \rangle) = \langle \mathcal{B}, \mathcal{A}, \tau, R_m, \gamma \rangle$$

where R_m is as defined in Equation 3.7, the (infinite) state space $\mathcal{B}$ is the set of all possible beliefs $\mathsf{Dist}(\mathcal{S})$, and $\tau : \mathcal{B} \times \mathcal{A} \to \mathsf{Dist}(\mathcal{B})$ is a transition function that depends upon the agent's beliefs and observations:

$$\tau(\beta, a)(\beta') = \Pr[B'{=}\beta' \mid B{=}\beta, A{=}a]$$

for all β and β' in $\mathcal{B}$ and a in $\mathcal{A}$.

To compute the value of $\tau(\beta, a)(\beta')$, we reduce it to features of the POMDP model m. In particular, we express $\tau(\beta, a)(\beta')$ as a formula of N_m as defined in Equation 3.8 and $\Theta_m(\beta, a, \beta')$, the set of observations that can accompany a belief state β transitioning under the action a to the belief state β' under the model m: $\Theta_m(\beta, a, \beta') = \{\, o \in \mathcal{O} \mid \mathsf{update}_m(\beta, a, o) = \beta' \,\}$.

$$\begin{aligned} \tau(\beta, a)(\beta') &= \Pr[B'{=}\beta' \mid B{=}\beta, A{=}a] \\ &= \sum_{o \in \Theta_m(\beta, a, \beta')} N_m(\beta, a)(o) \end{aligned}$$

For a POMDP m, the optimal strategy of $\mathsf{bmdp}(m)$ is a function from the state space of $\mathsf{bmdp}(m)$ to an action. As the state space is the set of all beliefs about states in m, such a strategy is also a strategy for the POMDP m. Furthermore, the optimal strategy of $\mathsf{bmdp}(m)$ is also the optimal strategy of m. The optimal value that a belief MDP $\mathsf{bmdp}(m)$ assigns to a belief state is equal to the optimal value that the POMDP m assigns to it. This result, proved as Proposition 3 below, implies that the belief MDP is equivalent to the POMDP in that each shares the same optimal strategies.

Proposition 3. *For all POMDPs m, for all belief states β and actions a of m, $V^*_m(\beta) = v^*_{\mathsf{bmdp}(m)}(\beta)$ and $Q^*_m(\beta, a) = q^*_{\mathsf{bmdp}(m)}(\beta, a)$.*

Since belief MDPs are the subject of previous work, we defer details and proofs to Appendix A.

To define the behaviors of a POMDP, we do not focus on prefixes of executions that refer to the state space of POMDP since the agent does not have knowledge of the current state of its environment. Rather, we use the belief states that the agent could hold. We say that a sequence $[\beta_1, a_1, o_1, \beta_2, a_2, o_2, \ldots, \beta_n, a_n, o_n]$ in $(\mathcal{B} \times \mathcal{A} \times \mathcal{O})^*$ is a *possible* behavior of a POMDP m if and only if $[\beta_1, a_1, \beta_2, a_2, \ldots \beta_n, a_n]$ is a possible behavior of $\mathsf{bmdp}(m)$ and for all $i < n$, $\beta_{i+1} = \mathsf{update}_m(\beta_i, a_i, o_i)$.

3.2.2 Advertising Example: Model

The POMDP Model. The model of the example involving an advertising website shown in Figure 3.3 can be formalized as a POMDP $m_{\mathsf{adv}} = \langle \mathcal{S}, \mathcal{A}, t, r, \mathcal{O}, \nu, \gamma \rangle$. Formally, the state space is $\mathcal{S} = \{\mathsf{f}, \mathsf{m}\} \times \{\mathsf{f}, \mathsf{m}, \bot\} \times \{\mathsf{ad}_1, \mathsf{ad}_2, \mathsf{ad}_3, \varnothing\}$ where f indicates a female; m, a male; $\bot$, that the database does not contain the visitor's sex; ad_i, that the website has shown the visitor the ith advertisement; and $\varnothing$, that the website has not shown an advertisement.

The action space is $\mathcal{A} = \{\mathsf{lookup}, \mathsf{ad}_1, \mathsf{ad}_2, \mathsf{ad}_3\}$. The actions ad_1, ad_2, and ad_3 correspond to the website showing the visitor one of the three possible advertisements while lookup corresponds to the website looking up information on the visitor.

The actions and states are related by the transition relation t. While the website has uncertainty about the sex of the visitor, the transition relation t is deterministic given the current state of the environment and the website's action. Thus, in this model, for all states s and actions a, the distribution $t(s, a)$ is always a *degenerate distribution*. The degenerate distribution $\mathsf{degen}(x)$ is equal to 1 at the value of x and 0 everywhere else (i.e., $\mathsf{degen}(x)(x) = 1$ and for all $y \neq x$, $\mathsf{degen}(x)(y) = 0$). In our model, $t(\langle g, d, \varnothing \rangle, \mathsf{ad}_i) = \mathsf{degen}(\langle g, d, \mathsf{ad}_i \rangle)$ for all g in $\{\mathsf{f}, \mathsf{m}\}$, d in $\{\mathsf{f}, \mathsf{m}, \bot\}$, and i in $\{1, 2, 3\}$ reflecting that showing an advertisement does not change the visitor's sex or the website's database. Furthermore, $t(\langle g, d, \mathsf{ad}_i \rangle, \mathsf{ad}_j) = \mathsf{degen}(\langle g, d, \mathsf{ad}_i \rangle)$ since the website can show the visitor only one advertisement. Lastly, $t(s, \mathsf{lookup}) = \mathsf{degen}(s)$ since looking up information in the database does not change the state of the environment. (It can, however, change the belief state of the agent.)

These transitions are accompanied by a reward. The reward for showing an advertisement depends upon the visitor's sex as follows:

$$\begin{aligned} r(\langle \mathsf{f}, d, \varnothing \rangle, \mathsf{ad}_1) &= 9 & \qquad r(\langle \mathsf{m}, d, \varnothing \rangle, \mathsf{ad}_1) &= 3 \\ r(\langle \mathsf{f}, d, \varnothing \rangle, \mathsf{ad}_2) &= 7 & \qquad r(\langle \mathsf{m}, d, \varnothing \rangle, \mathsf{ad}_2) &= 7 \\ r(\langle \mathsf{f}, d, \varnothing \rangle, \mathsf{ad}_3) &= 3 & \qquad r(\langle \mathsf{m}, d, \varnothing \rangle, \mathsf{ad}_3) &= 9 \end{aligned}$$

with $r(s, a) = 0$ for all other states s and actions a.

The observation space is $\mathcal{O} = \{\mathsf{f}, \mathsf{m}, \bot\} \times \{\mathsf{ad}_1, \mathsf{ad}_2, \mathsf{ad}_3, \varnothing\}$. The observation $\langle d, \alpha \rangle$ reveals that the database holds information d about the visitor's sex and that the visitor has seen advertisement α (with $\alpha = \varnothing$ if the visitor has not seen one).

The function ν relates these observations to actions and states. Again we restrict our attention to degenerate distributions since this example contains uncertainty but not truly random processes. For each state $\langle g, d, \alpha \rangle$, the lookup action results in the observation $\langle d, \alpha \rangle$. Thus, $\nu(\mathsf{lookup}, \langle g, d, \alpha \rangle) = \mathsf{degen}(\langle d, \alpha \rangle)$. (Note that the second argument to ν is the next state that results from performing the action, not the current state. However, the transition relation t is such that lookup does not change the state of the system and the next state is the current state.) We model showing an advertisement as providing the dummy observation $\circ$: $\nu(\mathsf{ad}_i, s) = \mathsf{degen}(\circ)$ for all i and states s.

We use a discounting factor of $\gamma = 0.9$ (to pick an arbitrary but reasonable value). These components define the example POMDP to be $m_{\mathsf{adv}} = \langle \mathcal{S}, \mathcal{A}, t, r, \mathcal{O}, \nu, \gamma \rangle$. The first part of a possible execution of m_{adv} with a total reward of 7 is

$$[\langle \mathsf{f}, \bot, \varnothing \rangle, \quad \mathsf{ad}_2, \quad \circ, \quad \langle \mathsf{f}, \bot, \mathsf{ad}_2 \rangle, \quad \ldots]$$

As mentioned above, all of the transitions of the POMDP model result in degenerate distributions. The paucity of non-degenerate probabilistic transitions is a feature of this example in which none of the actions involve random processes. Rather, the uncertainty in this example results from the website not knowing *a priori* the sex of the visitor. This uncertainty is captured by the model by having the website not knowing *a priori* what its initial state is. Intuitively, the action lookup attempts to remove this uncertainty by providing the website with an observation that reduces the set of states that it has to consider possible. In the case of making the observation $\langle \mathsf{f}, \alpha \rangle$ or $\langle \mathsf{m}, \alpha \rangle$, this reduction identifies the visitor's sex.

The way we represent the POMDP m_{adv} in Figure 3.3 fails to represent many parts of the formal model. For example, the figure does not show that performing the action ad in state $\langle \mathsf{f}, \mathsf{f}, \mathsf{ad}_1 \rangle$ results in the physician observing $\circ$. While this figure serves to focus the reader's attention on the interesting aspects of the POMDPs illustrated, the figure is not intended to replace its formal representation.

Beliefs and Optimal Actions. We can formalize the intuitive initial beliefs we assigned to the website as a belief state β_0. The belief that database is accurate requires that $\beta_0(\langle g_1, g_2, \alpha \rangle) = 0$ for α in $\{\mathsf{ad}_1, \mathsf{ad}_2, \mathsf{ad}_3, \varnothing\}$ and all g_1 and g_2 in $\{\mathsf{f}, \mathsf{m}\}$ such that $g_1 \neq g_2$. That the website has not shown an advertisement to the visitor yet requires that $\beta_0(\langle g, d, \mathsf{ad}_i \rangle) = 0$ for all g in $\{\mathsf{f}, \mathsf{m}\}$, d in $\{\mathsf{f}, \mathsf{m}, \bot\}$, and i in $\{1, 2, 3\}$. That visitors are equally likely to be female or male requires that $\beta_0(\langle \mathsf{f}, d, \alpha \rangle) = \beta_0(\langle \mathsf{m}, d, \alpha \rangle)$ for all d and α. The final constraint that the database be useful does not yield an exact constraint on β_0. Rather, we interpret it to imply the inequality $\beta_0(\langle g, g, \alpha \rangle) > \beta_0(\langle g, \bot, \alpha \rangle)$. To be concrete, we take β_0 to be as follows:

$$\begin{aligned}
\beta_0(\langle \mathsf{f}, \mathsf{f}, \varnothing \rangle) &= 0.4 \\
\beta_0(\langle \mathsf{m}, \mathsf{m}, \varnothing \rangle) &= 0.4 \\
\beta_0(\langle \mathsf{f}, \bot, \varnothing \rangle) &= 0.1 \\
\beta_0(\langle \mathsf{m}, \bot, \varnothing \rangle) &= 0.1 \\
\beta_0(\langle g, d, \mathsf{ad}_i \rangle) &= 0 \qquad\qquad \text{For all } g \text{ in } \{\mathsf{f}, \mathsf{m}\}, d \text{ in } \{\mathsf{f}, \mathsf{m}, \bot\}, \text{ and } i \text{ in } \{1, 2, 3\}
\end{aligned}$$

The optimal action to perform from the belief state β_0 is lookup. To see this result, note that after showing the visitor an advertisement, no further rewards are possible. Thus,

$$\begin{aligned}
&(3.9) & Q^*_{m_{\mathsf{adv}}}(\beta_0, \mathsf{ad}_1) &= R_{m_{\mathsf{adv}}}(\beta_0, \mathsf{ad}_1) + \gamma \sum_{o \in \mathcal{O}} N_{m_{\mathsf{adv}}}(\beta_0, \mathsf{ad}_1)(o) * V^*_{m_{\mathsf{adv}}}(\mathsf{update}_{m_{\mathsf{adv}}}(\beta_0, \mathsf{ad}_1, o)) \\
&(3.10) & &= R_{m_{\mathsf{adv}}}(\beta_0, \mathsf{ad}_1) + \gamma \sum_{o \in \mathcal{O}} N_{m_{\mathsf{adv}}}(\beta_0, \mathsf{ad}_1)(o) * 0 \\
&(3.11) & &= \sum_{s \in \mathcal{S}} \beta_0(s) * r(s, \mathsf{ad}_1) \\
&(3.12) & &= \sum_{d,\alpha} \beta_0(\langle \mathsf{f}, d, \alpha \rangle) * r(\langle \mathsf{f}, d, \alpha \rangle, \mathsf{ad}_1) + \sum_{d,\alpha} \beta_0(\langle \mathsf{m}, d, \alpha \rangle) * r(\langle \mathsf{m}, d, \alpha \rangle, \mathsf{ad}_1) \\
&(3.13) & &= (0.4 * 9 + 0.1 * 9) + (0.4 * 3 + 0.1 * 3) \\
&(3.14) & &= 6
\end{aligned}$$

where Line 3.10 comes from the fact that no further rewards are possible after showing a single advertisement. Similarly,

$$Q^*_{m_{\mathsf{adv}}}(\beta_0, \mathsf{ad}_2) = (0.4 * 7 + 0.1 * 7) + (0.4 * 7 + 0.1 * 7) = 7$$
$$Q^*_{m_{\mathsf{adv}}}(\beta_0, \mathsf{ad}_3) = (0.4 * 3 + 0.1 * 3) + (0.4 * 9 + 0.1 * 9) = 6$$

To compute $Q^*_{m_{\mathsf{adv}}}(\beta_0, \mathsf{lookup})$, we must examine the rewards that can occur after performing lookup. These rewards depend upon the observations made from lookup. Three observations are possible after performing lookup from β_0: $\langle \mathsf{f}, \varnothing\rangle$, $\langle \mathsf{m}, \varnothing\rangle$, and $\langle \bot, \varnothing\rangle$. In the case of observing $\langle \mathsf{f}, \varnothing\rangle$, the next belief state will be $\mathsf{update}_{m_{\mathsf{adv}}}(\beta_0, \mathsf{lookup}, \langle \mathsf{f}, \varnothing\rangle) = \beta_{\mathsf{f}}$ where

$$\beta_{\mathsf{f}}(\langle \mathsf{f}, \mathsf{f}, \varnothing\rangle) = \frac{\nu(\mathsf{lookup}, \langle \mathsf{f}, \mathsf{f}, \varnothing\rangle)(\langle \mathsf{f}, \varnothing\rangle) \sum_{s \in \mathcal{S}} \beta(s) * t(s, \mathsf{lookup})(\langle \mathsf{f}, \mathsf{f}, \varnothing\rangle)}{\sum_{s' \in \mathcal{S}} \nu(\mathsf{lookup}, s')(\langle \mathsf{f}, \varnothing\rangle) \sum_{s \in \mathcal{S}} \beta(s) * t(s, \mathsf{lookup})(s')} \tag{3.15}$$
$$= \frac{1 * (0.4 * 1)}{1 * (0.4 * 1)} \tag{3.16}$$
$$= 1 \tag{3.17}$$

where Line 3.15 comes from Lines 3.5 and 3.6. Line 3.16 comes from the fact that $t(s, \mathsf{lookup})(s')$ is equal to 1 for $s = s'$ and that $\nu(\mathsf{lookup}, \langle g, d, \alpha\rangle) = \mathsf{degen}(\langle d, \alpha\rangle)$. Similarly, $\mathsf{update}_{m_{\mathsf{adv}}}(\beta_0, \mathsf{lookup}, \langle \mathsf{m}, \varnothing\rangle) = \beta_{\mathsf{m}} = \mathsf{degen}(\langle \mathsf{m}, \mathsf{m}, \varnothing\rangle)$ and $\mathsf{update}_{m_{\mathsf{adv}}}(\beta_0, \mathsf{lookup}, \langle \bot, \varnothing\rangle) = \beta_\bot$ where $\beta_\bot(\langle \mathsf{f}, \bot, \varnothing\rangle) = 0.5$ and $\beta_\bot(\langle \mathsf{m}, \bot, \varnothing\rangle) = 0.5$. Using calculations similar to those done for $Q^*_{m_{\mathsf{adv}}}(\beta_0, \mathsf{ad}_1)$, we can compute $Q^*_{m_{\mathsf{adv}}}(\beta_{\mathsf{f}}, \mathsf{ad}_1)$. Doing these calculations for every action and each of the belief states β_{f}, β_{m}, and $\beta_\bot$, we can find the optimal actions and their values for each of these three belief states:

$$V^*_{m_{\mathsf{adv}}}(\beta_{\mathsf{f}}) = Q^*_{m_{\mathsf{adv}}}(\beta_{\mathsf{f}}, \mathsf{ad}_1) = 9$$
$$V^*_{m_{\mathsf{adv}}}(\beta_{\mathsf{m}}) = Q^*_{m_{\mathsf{adv}}}(\beta_{\mathsf{m}}, \mathsf{ad}_3) = 9$$
$$V^*_{m_{\mathsf{adv}}}(\beta_\bot) = Q^*_{m_{\mathsf{adv}}}(\beta_\bot, \mathsf{ad}_2) = 7$$

Furthermore,

$$N_{m_{\mathsf{adv}}}(\beta_0, \mathsf{lookup})(\langle \mathsf{f}, \varnothing\rangle) = \sum_{s \in \mathcal{S}} \beta_0(s) * \sum_{s' \in \mathcal{S}} t(s, \mathsf{lookup})(s') * \nu(\mathsf{lookup}, s')(\langle \mathsf{f}, \varnothing\rangle) \tag{3.18}$$
$$= \sum_{s \in \mathcal{S}} \beta_0(s) * \nu(\mathsf{lookup}, s)(\langle \mathsf{f}, \varnothing\rangle) \tag{3.19}$$
$$= \beta_0(\langle \mathsf{f}, \mathsf{f}, \varnothing\rangle) * \nu(\mathsf{lookup}, \langle \mathsf{f}, \mathsf{f}, \varnothing\rangle)(\langle \mathsf{f}, \varnothing\rangle) \tag{3.20}$$
$$= 0.4 * 1 \tag{3.21}$$
$$= 0.4 \tag{3.22}$$

where Line 3.19 results from the fact that $t(s, \mathsf{lookup}) = \mathsf{degen}(s)$. Line 3.20 follows from the fact that the only state s such that both $\beta_0(s)$ and $\nu(\mathsf{lookup}, s)(\langle \mathsf{f}, \varnothing\rangle)$ are non-zero is $\langle \mathsf{f}, \mathsf{f}, \varnothing\rangle$. Similarly,

$$N_{m_{\mathsf{adv}}}(\beta_0, \mathsf{lookup})(\langle \mathsf{m}, \varnothing\rangle) = \beta_0(\langle \mathsf{m}, \mathsf{m}, \varnothing\rangle) * \nu(\mathsf{lookup}, \langle \mathsf{m}, \mathsf{m}, \varnothing\rangle)(\langle \mathsf{m}, \varnothing\rangle)$$
$$= 0.4$$

and

$$\begin{aligned}
&N_{m_{\mathsf{adv}}}(\beta_0, \mathsf{lookup})(\langle\bot, \varnothing\rangle)\\
&\qquad = \beta_0(\langle\mathsf{f}, \bot, \varnothing\rangle) * \nu(\mathsf{lookup}, \langle\mathsf{f}, \bot, \varnothing\rangle)(\langle\bot, \varnothing\rangle) + \beta_0(\langle\mathsf{m}, \bot, \varnothing\rangle) * \nu(\mathsf{lookup}, \langle\mathsf{m}, \bot, \varnothing\rangle)(\langle\bot, \varnothing\rangle)\\
&\qquad = 0.1 * 1 + 0.1 * 1\\
&\qquad = 0.2
\end{aligned}$$

and

$$\begin{aligned}
N_{m_{\mathsf{adv}}}(\beta_0, \mathsf{lookup})(\circ) &= \sum_{s\in\mathcal{S}} \beta_0(s) * \sum_{s'\in\mathcal{S}} t(s, \mathsf{lookup})(s') * \nu(\mathsf{lookup}, s')(\circ)\\
&= \sum_{s\in\mathcal{S}} \beta_0(s) * \sum_{s'\in\mathcal{S}} t(s, \mathsf{lookup})(s') * 0\\
&= 0
\end{aligned}$$

Putting these parts together, we find that

$$\begin{aligned}
&Q^*_{m_{\mathsf{adv}}}(\beta_0, \mathsf{lookup})\\
&\qquad = R_{m_{\mathsf{adv}}}(\beta_0, \mathsf{lookup}) + \gamma \sum_{o\in\mathcal{O}} N_{m_{\mathsf{adv}}}(\beta_0, \mathsf{lookup})(o) * V^*_{m_{\mathsf{adv}}}(\mathsf{update}_{m_{\mathsf{adv}}}(\beta_0, \mathsf{lookup}, o))\\
&\qquad = 0 + 0.9 \sum_{o\in\mathcal{O}} N_{m_{\mathsf{adv}}}(\beta_0, \mathsf{lookup})(o) * V^*_{m_{\mathsf{adv}}}(\mathsf{update}_{m_{\mathsf{adv}}}(\beta_0, \mathsf{lookup}, o))\\
&\qquad = 0.9 \left(\begin{array}{cl}
 & N_{m_{\mathsf{adv}}}(\beta_0, \mathsf{lookup})(\langle\mathsf{f}, \varnothing\rangle) * V^*_{m_{\mathsf{adv}}}(\mathsf{update}_{m_{\mathsf{adv}}}(\beta_0, \mathsf{lookup}, \langle\mathsf{f}, \varnothing\rangle))\\
+ & N_{m_{\mathsf{adv}}}(\beta_0, \mathsf{lookup})(\langle\mathsf{m}, \varnothing\rangle) * V^*_{m_{\mathsf{adv}}}(\mathsf{update}_{m_{\mathsf{adv}}}(\beta_0, \mathsf{lookup}, \langle\mathsf{m}, \varnothing\rangle))\\
+ & N_{m_{\mathsf{adv}}}(\beta_0, \mathsf{lookup})(\langle\bot, \varnothing\rangle) * V^*_{m_{\mathsf{adv}}}(\mathsf{update}_{m_{\mathsf{adv}}}(\beta_0, \mathsf{lookup}, \langle\bot, \varnothing\rangle))\\
+ & N_{m_{\mathsf{adv}}}(\beta_0, \mathsf{lookup})(\circ) * V^*_{m_{\mathsf{adv}}}(\mathsf{update}_{m_{\mathsf{adv}}}(\beta_0, \mathsf{lookup}, \circ))
\end{array}\right)\\
&\qquad = 0.9 \left(0.4 * V^*_{m_{\mathsf{adv}}}(\beta_{\mathsf{f}}) + 0.4 * V^*_{m_{\mathsf{adv}}}(\beta_{\mathsf{m}}) + 0.2 * V^*_{m_{\mathsf{adv}}}(\beta_{\bot}) + 0\right)\\
&\qquad = 0.9\,(0.4 * 9 + 0.4 * 9 + 0.2 * 7)\\
&\qquad = 7.74
\end{aligned}$$

Comparing this value to $Q^*_{m_{\mathsf{adv}}}(\beta_0, \mathsf{ad}_1)$, $Q^*_{m_{\mathsf{adv}}}(\beta_0, \mathsf{ad}_2)$, and $Q^*_{m_{\mathsf{adv}}}(\beta_0, \mathsf{ad}_3)$, we find that lookup is the optimal action for the website to take from belief state β_0.

Thus, an optimal strategy σ^* to m_{adv} must be such that $\sigma^*(\beta_0) = \mathsf{lookup}$, $\sigma^*(\beta_{\mathsf{f}}) = \mathsf{ad}_1$, $\sigma^*(\beta_{\mathsf{m}}) = \mathsf{ad}_3$, and $\sigma^*(\beta_\bot) = \mathsf{ad}_2$. These results match our intuitions. Various optimal strategies differ as to what the website does after showing the advertisement as such actions do not affect the reward. (We return to this point later when we consider non-redundancy in Section 3.2.4.)

3.2.3 Physician Example: Model

The POMDP Model. The example shown in Figure 3.2 corresponds to a POMDP m_{phy}. m_{phy} is equal to $\langle\mathcal{S}, \mathcal{A}, t, r, \mathcal{O}, \nu, \gamma\rangle$ where

- $\mathcal{S} = \{s_1, s_2, \ldots, s_7, s_8\}$;

state s	action a	distribution $t(s, a)$ over next states
s_1	take	$\mathsf{degen}(s_3)$
s_2	take	$\mathsf{degen}(s_4)$
s_3	$\mathsf{send}x_1$	$\mathsf{degen}(s_5)$
s_3	diagnose	$\mathsf{degen}(s_7)$
s_4	$\mathsf{send}x_2$	$t(s_4, a)(s_8) = 0.2$, $t(s_4, a)(s_6) = 0.8$, $t(s, a)(s) = 0$ for $s \notin \{s_6, s_8\}$
s_5	diagnose	$\mathsf{degen}(s_7)$
s_6	diagnose	$\mathsf{degen}(s_8)$
s	a	$\mathsf{degen}(s)$ for all remaining s and a

Table 3.1: Transition relation for the POMDP m_{phy}. Recall that the degenerate distribution $\mathsf{degen}(x)$ is equal to 1 at the value of x and 0 everywhere else.

- $\mathcal{A} = \{\mathsf{take}, \mathsf{send}x_1, \mathsf{send}x_2, \mathsf{diagnose}\}$;
- t is defined in Table 3.1;
- r is such that $r(s_3, \mathsf{diagnose}) = r(s_5, \mathsf{diagnose}) = r(s_6, \mathsf{diagnose}) = 12$ and $r(s, a) = 0$ for all other values of $s \in \mathcal{S}$ and $a \in \mathcal{A}$;
- $\mathcal{O} = \{x_1, x_2, \circ\}$;
- ν is defined such that $\nu(\mathsf{take}, s_3) = \mathsf{degen}(x_1)$, $\nu(\mathsf{take}, s_4) = \mathsf{degen}(x_2)$, and $\nu(a, s') = \mathsf{degen}(\circ)$ for all other actions a and states s'; and
- γ, which is not represented in the figure, is 0.9 (to pick an arbitrary but reasonable value).

As with the POMDP m_{adv} of Section 3.2.2, this POMDP does not feature many non-degenerate probabilistic transitions. Again, this paucity is a feature of the example in which few of the actions involve random processes. Much of the uncertainty in this example results from the physician not knowing the status of his patient, which is captured by the model by having the physician not knowing *a priori* what his initial state is. Intuitively, the action take removes this uncertainty by providing the physician with an observation that identifies his current state.

As with the model m_{adv}, the way we represent the POMDP m_{phy} in Figure 3.2 also does not represent many parts of the formal model. For example, the figure does not show that performing the action take in state s_3 results in the physician observing x_1. This fact is implied since ν only depends upon the resulting state (s_3) and the action, not the original state (s_1 or s_3).

Beliefs and Optimal Actions. Consider the initial beliefs β_0 discussed in Section 3.1.1 that assigns non-zero probabilities to only the possible initial states ($\beta_0(s_1) = 0.9$ and $\beta_0(s_2) = 0.1$). Under the model m_{phy}, starting from the initial belief state β_0, the physician will learn with certainty which state he is in after observing the value of the X-ray. Thus, his belief state will be a degenerate distribution after performing the action take. Performing calculations similar to those performed in Section 3.2.2, we may find that to be an optimal strategy for m_{phy}, a strategy σ^* must be such that $\sigma^*(\beta_0) = \mathsf{take}$, $\sigma^*(\mathsf{degen}(s_3)) = \sigma^*(\mathsf{degen}(s_5)) = \sigma^*(\mathsf{degen}(s_6)) = \mathsf{diagnose}$, and $\sigma^*(\mathsf{degen}(s_4)) = \mathsf{send}x_2$.

3.2.4 Non-redundancy

In the advertising example, the actions of the website after showing the advertisement are unconstrained. The reason is that showing the advertisement will result in the current state of the POMDP becoming of the form $\langle g, d, \mathsf{ad}_i \rangle$. States of this form are all *absorbing*: all possible actions result in returning to the same state. Furthermore, all the actions possible from these states result in zero reward. Since the only criterion of an optimal strategy is its expected total discounted reward, a strategy may assign any action to these states without changing whether it is optimal.

This effect leads to the counterintuitive result that performing lookup in the belief state $\mathsf{degen}(\langle \mathsf{f}, \mathsf{f}, \mathsf{ad}_1 \rangle)$ is for the purpose of marketing. This result is counterintuitive since the same total reward is possible regardless of whether the agent performs the lookup action from the state $\langle \mathsf{f}, \mathsf{f}, \mathsf{ad}_1 \rangle$. Thus, the agent believes that it is performing an action that cannot further its total reward and yet it is still for the purpose represented by the reward.

Indeed, this counterintuitive result is true of all the actions in $\mathcal{A}$. Yet, by the definition of the POMDP, the agent must continue to perform one of four actions in $\mathcal{A}$ despite none of them adding to the total reward. Intuitively, the agent should just stop.

We have already formalized a solution to this counterintuitive result for MDPs using the idea of non-redundancy in Section 2.1.2. We may apply the same idea to POMDPs. We add to each POMDP a distinguished action stop that indicates that the agent stops and does nothing more (for the purpose in question). The stop action always produces zero reward and results in no state change (i.e., $r(s, \mathsf{stop}) = 0$ and $t(s, \mathsf{stop}) = \mathsf{degen}(s)$ for all s in $\mathcal{S}$). The action stop is always followed by the dummy observation $\circ$ (i.e., $\nu(\mathsf{stop}, s) = \mathsf{degen}(\circ)$ for all s in $\mathcal{S}$).

An action a from a belief state β is *redundant* if it is no better than stopping (i.e., if $Q^*_m(\beta, a) \leq Q^*_m(\beta, \mathsf{stop}) = 0$). A strategy is *non-redundant* if it never requires a redundant action from any belief state.

In Section 2.1.2, we converted the MDP model into the NMDP model by considering strategies that contain redundant actions to be suboptimal. Similarly, we can create a non-redundant POMDP model (NPOMDP). To do so, we require that the agent selects not just any strategy from the set of those that maximizes the expected total discounted reward, but rather that it selects only a strategy that both maximizes the reward and is non-redundant. More formally, we define the set of optimal strategies for an NPOMDP m to be $\mathsf{nopt}(m) = \mathsf{nopt}(\mathsf{bmdp}(m))$ where nopt is defined for MDPs in Section 2.1.2. By Theorem 1, $\mathsf{nopt}(m)$ is not empty since $\mathsf{bmdp}(m)$ is an MDP.

We define $\mathsf{nbehv}(m)$ such that a sequence $[\beta_1, a_1, o_1, \beta_2, a_2, o_2, \ldots, \beta_n, a_n, o_n]$ in $(\mathcal{B} \times \mathcal{A} \times \mathcal{O})^*$ is in $\mathsf{nbehv}(m)$ if and only if it is a possible behavior of m and $[\beta_1, a_1, \beta_2, a_2, \ldots \beta_n, a_n]$ is in $\mathsf{nbehv}(\mathsf{bmdp}(m))$.

Henceforth, we will use m_{phy} and m_{adv} to refer to the NPOMDP versions of the model presented in Sections 3.2.3 and 3.2.2, respectively. For example, m_{phy} will now refer to the NPOMDP such that

- $\mathcal{S}$ is as before: $\mathcal{S} = \{s_1, s_2, \ldots, s_7, s_8\}$;

- $\mathcal{A} = \{\mathsf{take}, \mathsf{send}x_1, \mathsf{send}x_2, \mathsf{diagnose}, \mathsf{stop}\}$;

- t as before except with domain of actions now including stop: that is, as defined in Table 3.1 with the last line of the table showing that $t(s, \mathsf{stop}) = \mathsf{degen}(s)$ for all states s;

- r is as before except with the domain of actions now including stop:

$$r(s_3, \mathsf{diagnose}) = r(s_5, \mathsf{diagnose}) = r(s_6, \mathsf{diagnose}) = 12$$

and $r(s, a) = 0$ for all other values of $s \in \mathcal{S}$ and $a \in \mathcal{A}$ (including stop);

- $\mathcal{O}$ is as before: $\mathcal{O} = \{x_1, x_2, \circ\}$ (if $\mathcal{O}$ did not already have the dummy observation $\circ$ as a member, we would had added it);
- ν is as before except with the set of actions now including stop: $\nu(\mathsf{take}, s_3) = \mathsf{degen}(x_1)$, $\nu(\mathsf{take}, s_4) = \mathsf{degen}(x_2)$, and $\nu(a, s') = \mathsf{degen}(\circ)$ for all other actions a (including stop) and states s'; and
- γ is as before: $\gamma = 0.9$.

The requirement of non-redundancy forces non-redundant optimal strategies σ^* to be such that

$$\sigma^*(\mathsf{degen}(s_7)) = \sigma^*(\mathsf{degen}(s_8)) = \mathsf{stop}$$

in addition to the requirements resulting from optimality:

$$\begin{aligned} \sigma^*(\beta_0) &= \mathsf{take} \\ \sigma^*(\mathsf{degen}(s_3)) = \sigma^*(\mathsf{degen}(s_5)) = \sigma^*(\mathsf{degen}(s_6)) &= \mathsf{diagnose} \\ \sigma^*(\mathsf{degen}(s_4)) &= \mathsf{send}x_2 \end{aligned}$$

As with our figures depicting NMDPs, our figures of NPOMDPs do not show the do-nothing action stop that is always a self-loop of zero reward. Also implicit is that the observation from stop is always the dummy observation $\circ$. Since the features that distinguish a NPOMDP from a POMDP are not represented in our figures of either, a single figure may be reused to represent both a POMDP and the corresponding NPOMDP. For example, while we introduced Figure 3.2 to represent the POMDP formalized in Section 3.2.3, it also serves to represent the NPOMDP version described in this section.

3.3 Modeling Information Use

To gain information is to see a distinction. Thus, to ignore information corresponds to ignoring this distinction. Below we formalize this idea using an equivalence relation that conflates information. We then apply the formalization to our two examples.

3.3.1 Formal Model

To formalize the idea of using or ignoring information, we use an equivalence relation $\equiv$ over an observation space $\mathcal{O}$. For each equivalence class of $\equiv$, the agent will conflate its members by treating every observation in it as indistinguishable from one another. Let $\equiv[o]$ denote the equivalence class that holds the observation o (i.e., $\equiv[o] = \{\, o' \in \mathcal{O} \mid o' \equiv o \,\}$).

To ignore these distinctions, when the agent observes o, it updates its beliefs as though it has seen some element of $\equiv[o]$ but is unsure of which one. That is, if the agent starts with beliefs β and observes o after performing the action a, it will develop the new beliefs β' where $\beta'(s') = \Pr[S'{=}s' \mid O \in {\equiv}[o], A{=}a, B{=}\beta]$. Let $\mathsf{update}'_m(\beta, a, o, \equiv)$ denote these updated beliefs β'. To show how to compute $\mathsf{update}'_m(\beta, a, o, \equiv)$, we rewrite update' in terms of m and $\equiv$ as follows:

$$
\begin{aligned}
&\mathsf{update}'(\beta, a, o, \equiv)(s') \\
&= \Pr[S'{=}s' \mid O \in {\equiv}[o], A{=}a, B{=}\beta] \\
&= \frac{\Pr[O \in {\equiv}[o] \mid S'{=}s', A{=}a, B{=}\beta] \Pr[S'{=}s' \mid A{=}a, B{=}\beta]}{\Pr[O \in {\equiv}[o] \mid A{=}a, B{=}\beta]} \\
&= \frac{\Pr[\vee_{o_1 \in \equiv[o]} O{=}o_1 \mid S'{=}s', A{=}a, B{=}\beta] \Pr[S'{=}s' \mid A{=}a, B{=}\beta]}{\Pr[\vee_{o_1 \in \equiv[o]} O{=}o_1 \mid A{=}a, B{=}\beta]} \\
&= \frac{\sum_{o_1 \in \equiv[o]} \Pr[O{=}o_1 \mid S'{=}s', A{=}a, B{=}\beta] \Pr[S'{=}s' \mid A{=}a, B{=}\beta]}{\sum_{o_1 \in \equiv[o]} \Pr[O{=}o_1 \mid A{=}a, B{=}\beta]} \\
&= \frac{\sum_{o_1 \in \equiv[o]} \Pr[O{=}o_1 \mid S'{=}s', A{=}a, B{=}\beta] \sum_{s \in \mathcal{S}} \Pr[S{=}s \mid A{=}a, B{=}\beta] \Pr[S'{=}s' \mid A{=}a, B{=}\beta, S{=}s]}{\sum_{o_1 \in \equiv[o]} \Pr[O{=}o_1 \mid A{=}a, B{=}\beta]} \\
&= \frac{\sum_{o_1 \in \equiv[o]} \nu(a, s')(o_1) \sum_{s \in \mathcal{S}} \beta(s) * t(s, a)(s')}{\sum_{o_1 \in \equiv[o]} \Pr[O{=}o_1 \mid A{=}a, B{=}\beta]}
\end{aligned}
$$

The third line is well defined since the agent observing o after performing the action a from belief state β implies that $\Pr[O \in {\equiv}[o] \mid A{=}a, B{=}\beta] \geq \Pr[O{=}o \mid A{=}a, B{=}\beta] > 0$. We may move from the disjunction in the fourth line to the summation in the fifth line since the agent can make only a single observation after each action making the different possible observations mutually exclusive events.

Looking at the denominator of the last line, since

$$\sum_{o_1 \in \equiv[o]} \Pr[O{=}o_1 \mid A{=}a, B{=}\beta]$$

is independent of s', it may be treated as a normalization factor equal to

$$\sum_{o_1 \in \equiv[o]} \sum_{s' \in \mathcal{S}} \nu(a, s')(o) \sum_{s \in \mathcal{S}} \beta(s) * t(s, a)(s')$$

We show how to construct from a POMDP m and an equivalence relation $\equiv$ a new POMDP that updates its beliefs in manner consistent with the modified updating function update'_m. To do so, we alter the observation space $\mathcal{O}$ and observation distribution ν of m. This construction enables defining information use with POMDPs and obviates the need to introduce a new class of models specialized for the modified belief updating function update'. Since the construction only affects the observation space and observation distribution, it works identically on NPOMDPs to produce an NPOMDP that models information use and purpose restrictions.

Given a POMDP m and an equivalence relation $\equiv$ on $\mathcal{O}$, let $m/{\equiv}$ denote the restricted POMDP that results from the agent ignoring the distinctions among observations related by $\equiv$. We define the quotient POMDP $m/{\equiv}$ by quotienting the observation space $\mathcal{O}$ of m with $\equiv$ so that $m/{\equiv}$ ignores information while using the standard update function. Given $m = \langle \mathcal{S}, \mathcal{A}, t, r, \mathcal{O}, \nu, \gamma \rangle$, let $m/{\equiv}$ equal the POMDP $\langle \mathcal{S}, \mathcal{A}, t, r, \mathcal{O}/{\equiv}, \nu/{\equiv}, \gamma \rangle$ where $\mathcal{O}/{\equiv}$ is the partitioning of $\mathcal{O}$ under $\equiv$ (i.e., $\mathcal{O}/{\equiv}$ is the set of equivalence classes of $\equiv$) and $\nu/{\equiv}(a, s')(O) = \sum_{o \in O} \nu(a, s')(o)$ where O is an element of $\mathcal{O}/{\equiv}$ (i.e., O is an equivalence class of $\equiv$).

The following proposition (Proposition 4) verifies that $m/{\equiv}$ is a POMDP by showing that $\nu/{\equiv}$ satisfies the requirements of being a probability distribution.

Proposition 4. *For all POMDPs m and equivalence relations $\equiv$ over the observation space of m, $m/{\equiv}$ is a POMDP.*

Proof. We must prove that $\nu/{\equiv}$ is a well-defined probability distribution over the space of observations $\mathcal{O}/{\equiv}$.

$$\begin{aligned}\sum_{O \in \mathcal{O}/\equiv} \nu/{\equiv}(O) &= \sum_{O \in \mathcal{O}/\equiv} \sum_{o \in O} \nu(o) \\ &= \sum_{o \in \mathcal{O}} \nu(o) \\ &= 1\end{aligned}$$

where the second line follows from $\mathcal{O}/{\equiv}$ being a partition of $\mathcal{O}$ and the last line follows from ν being a distribution over $\mathcal{O}$.

For all $O \in \mathcal{O}/{\equiv}$,

$$\nu/{\equiv}(O) = \sum_{o \in O} \nu(o)$$

As ν is a distribution, $\nu(o) \geq 0$ for all o in O. Thus, $\sum_{o \in O} \nu(o) \geq 0$. As $O \subseteq \mathcal{O}$ and ν is a distribution over $\mathcal{O}$, $\sum_{o \in O} \nu(o) \leq \sum_{o \in \mathcal{O}} \nu(o) = 1$. □

The next proposition (Proposition 5) shows that $m/{\equiv}$ captures ignoring information according to the definition of update'.

Proposition 5. *For all POMDPs m and equivalence relations $\equiv$ over the observation space of m, update'_m and $\mathsf{update}_{m/\equiv}$ are equivalent.*

Proof. For all β, a, o, $\equiv$, and s',

$$\begin{aligned}\mathsf{update}'_m(\beta, a, o, \equiv)(s') &= \frac{\sum_{o_1 \in \equiv[o]} \nu(a, s')(o_1) \sum_{s \in \mathcal{S}} \beta(s) * t(s, a)(s')}{\sum_{o_1 \in \equiv[o]} \sum_{s' \in \mathcal{S}} \nu(a, s')(o) \sum_{s \in \mathcal{S}} \beta(s) * t(s, a)(s')} \\ &= \frac{\sum_{o_1 \in \equiv[o]} \nu(a, s')(o_1) \sum_{s \in \mathcal{S}} \beta(s) * t(s, a)(s')}{\sum_{s' \in \mathcal{S}} \sum_{o_1 \in \equiv[o]} \nu(a, s')(o_1) \sum_{s \in \mathcal{S}} \beta(s) * t(s, a)(s')} \\ &= \frac{\nu/{\equiv}(a, s')(\equiv[o]) \sum_{s \in \mathcal{S}} \beta(s) * t(s, a)(s')}{\sum_{s' \in \mathcal{S}} \nu/{\equiv}(a, s')(\equiv[o]) \sum_{s \in \mathcal{S}} \beta(s) * t(s, a)(s')} \\ &= \mathsf{update}_{m/\equiv}(\beta, a, \equiv[o])(s')\end{aligned}$$

where the last line comes from the same reasoning found in Lines 3.1 to 3.5 applied in reverse. □

These two propositions (Propositions 4 and 5) together show that $m/{\equiv}$ is a POMDP with a new observation space $\mathcal{O}/{\equiv}$ that ignores information conflated by $\equiv$. Thus, we may model ignoring information using the POMDP model and do not need to construct new model based around the modified updating function update'. While using a model $m/{\equiv}$, the actual observations made by the agent continue to lie in $\mathcal{O}$, not $\mathcal{O}/{\equiv}$. Thus, to use the model $m/{\equiv}$ as a POMDP, one must map the observation o to $\equiv[o]$ before updating the agent's beliefs.

An agent does not use the information conveyed by the distinctions among the observations $O \subseteq \mathcal{O}$ if the agent plans using a POMDP $m/{\equiv}$ where all the observations of O are related by $\equiv$. Strictly speaking, the agent uses the information if he plans with the model m even if the strategy that it would choose under m and $m/{\equiv}$ are identical.

Note that if $\equiv$ relates every element of $\mathcal{O}$ to one another, this does not imply that the agent never learns anything. For example, suppose that the agent can perform an action a that always leads to a single state s_a. After performing this action, the agent will learn with certainty that it is in the state s_a even in the absence of any meaningful observations. Thus, our formalism cannot capture policies that restrict an agent from using information about what actions it performed. However, we have not seen any such policies in practice.

More generally, the POMDP model itself contains information about the agent's environment that the agent will continue to use. The auditor must ensure that the agent does not construct a model using prohibited information. Typically, privacy policies govern information about specific individuals (e.g., a visitor's sex). The agents subjected to auditing typically handle many such individuals (e.g., our website will show advertisements to many visitors). Thus, an agent typically constructs its model from general information (e.g., the ratio of the sexes) and uses observations to make it parametric in the individual. Thus, we expect using prohibited information to create a model would be conspicuous.

At the opposite extreme, $m/{=}$ is a restricted POMDP that behaves identically to m in that $=$ (i.e., equality) ignores no distinctions between any two observations. That is, for $m/{=}$, every observation o is mapped to the singleton $\{o\}$ in $\mathcal{O}/{=}$ meaning that it is conflated with no other observations. This results in $\mathsf{update}'(\beta, a, o, =) = \mathsf{update}_m(\beta, a, =[o])$ being equal to $\mathsf{update}_m(\beta, a, o)$.

3.3.2 Advertising Example: Information Use

Returning to our running example formally modeled in Section 3.2.2, the policy governing the website states that the website will not use the database's entry about the visitor's sex for determining the advertisement to show the visitor. The auditor must decide how to formally model this restriction. One way would be to use the smallest equivalence relation $\equiv_{\mathsf{adv}}$ such that for all d_1 and d_2 in $\{\mathsf{f}, \mathsf{m}, \bot\}$ and all α in $\{\mathsf{ad}_1, \mathsf{ad}_2, \mathsf{ad}_3, \varnothing\}$, $\langle d, \alpha\rangle \equiv_{\mathsf{adv}} \langle d_2, \alpha\rangle$, conflating the database's entry for all observations. Under this equivalence relation, $m_{\mathsf{adv}}/{\equiv_{\mathsf{adv}}}$ is $\langle \mathcal{S}, \mathcal{A}, t, r, \mathcal{O}/{\equiv_{\mathsf{adv}}}, \nu/{\equiv_{\mathsf{adv}}}, \gamma\rangle$. $\mathcal{O}/{\equiv_{\mathsf{adv}}}$ holds five observations created from the observations of $\mathcal{O}$ as the equivalence classes of $\equiv_{\mathsf{adv}}$. For each value of α in $\{\mathsf{ad}_1, \mathsf{ad}_2, \mathsf{ad}_3, \varnothing\}$, $\mathcal{O}/{\equiv_{\mathsf{adv}}}$ holds the observation $\{\langle \mathsf{f}, \alpha\rangle, \langle \mathsf{m}, \alpha\rangle, \langle \bot, \alpha\rangle\}$. Furthermore, $\mathcal{O}/{\equiv_{\mathsf{adv}}}$ holds $\{\circ\}$ since $\circ$ is only related to itself by $\equiv_{\mathsf{adv}}$. For all states $\langle g, d, \alpha\rangle$, $\nu/{\equiv}(\mathsf{lookup}, \langle g, d, \alpha\rangle) = \mathsf{degen}(\{\langle \mathsf{f}, \alpha\rangle, \langle \mathsf{m}, \alpha\rangle, \langle \bot, \alpha\rangle\})$. For all states s' and all i, $\nu/{\equiv}(\mathsf{ad}_i, s') = \mathsf{degen}(\{\circ\})$.

The website planning with the model $m_{\mathsf{adv}}/{\equiv_{\mathsf{adv}}}$ does not use any information from database. In this case, the website's initial beliefs will solely determine its optimal strategies. Furthermore, under $m_{\mathsf{adv}}/{\equiv_{\mathsf{adv}}}$, performing the action lookup will provide no benefit to the website since the website will conflate the observations to ignore the information it provides. Any optimal strategy for $m_{\mathsf{adv}}/{\equiv_{\mathsf{adv}}}$ will call for performing ad_2 from the initial beliefs β_0 discussed in Section 3.2.2.

Alternatively, the auditor might conclude that the policy only forces the website to ignore whether the database records the visitor as a female or male and not whether the database contains this information. In this case, the auditor would use a different equivalence relation $\equiv'_{\mathsf{adv}}$ such that $\langle \mathsf{f}, \alpha \rangle \equiv'_{\mathsf{adv}} \langle \mathsf{m}, \alpha \rangle$ but $\langle \mathsf{f}, \alpha \rangle \not\equiv'_{\mathsf{adv}} \langle \bot, \alpha \rangle \not\equiv'_{\mathsf{adv}} \langle \mathsf{m}, \alpha \rangle$ for all α in $\{\mathsf{ad}_1, \mathsf{ad}_2, \mathsf{ad}_3, \varnothing\}$. Under the initial beliefs β_0, the website would behave identically under $\equiv_{\mathsf{adv}}$ and $\equiv'_{\mathsf{adv}}$. However, if the website's initial beliefs were such that it is much more likely to not know a female's sex than a male's, then it might choose to show ad_1 instead of ad_2 in the case of observing $\langle \bot, \varnothing \rangle$.

In our opinion, $\equiv'_{\mathsf{adv}}$ more accurately reflects the policy that the visitor's sex will not be used for marketing. We feel that the distinction between $\langle \bot, \alpha \rangle$ and $\langle \mathsf{f}, \alpha \rangle$ is provides information about the visitor's presence in the database. However, an auditor may interpret the privacy policy differently or be enforcing a privacy policy that clearly prohibits the use of any information about the patient's sex or willingness to provide that information. In either of these cases, the auditor may require the auditee to treat all three observations as indistinguishable.

3.3.3 Physician Example: Information Use

We return to the previously discussed example formally modeled in Section 3.2.3. The naming of actions in that example conveys that the actions $\mathsf{send}x_1$ and $\mathsf{send}x_2$ each involves the information of the X-ray, which is represented as the observations x_1 and x_2. An informal inspection of the POMDP m_{phy} reveals that we intuitively agree that these actions involve that information.

It may be tempting to apply our formalism with the goal of showing that $\mathsf{send}x_1$ and $\mathsf{send}x_2$ actions use the information of X-rays. However, our formalization does not determine whether an *action* uses information. The reason is that a single action could either use information or not depending upon why the agent selected to perform that action. For example, despite our intuition that the action $\mathsf{send}x_2$ uses the information of the X-ray, if the physician decided to perform the action $\mathsf{send}x_2$ regardless of any observations he made, then that action does not use the information of the X-ray. Thus, rather than determine whether an *action* uses information, our formalism determines whether an *agent* uses information based upon how the agent plans.

Our formalization does show that an agent performing the action $\mathsf{send}x_2$ because the X-ray had the value x_2 used that information. To see this result, consider the equivalence relation $\equiv_{\mathsf{phy}}$ that is the smallest such that $x_1 \equiv_{\mathsf{phy}} x_2$ ($\circ$ is not related to anything but itself). Under this equivalence relation, $m_{\mathsf{phy}}/{\equiv_{\mathsf{phy}}}$ is $\langle \mathcal{S}, \mathcal{A}, t, r, \mathcal{O}/{\equiv_{\mathsf{phy}}}, \nu/{\equiv_{\mathsf{phy}}}, \gamma \rangle$. The observation space $\mathcal{O}/{\equiv_{\mathsf{phy}}}$ is $\{\{\circ\}, \{x_1, x_2\}\}$. $\nu/{\equiv_{\mathsf{phy}}}$ is such that $\nu/{\equiv_{\mathsf{phy}}}(\mathsf{take}, s_3)$ and $\nu/{\equiv_{\mathsf{phy}}}(\mathsf{take}, s_4)$ are each equal to $\mathsf{degen}(\{x_1, x_2\})$; and $\nu/{\equiv_{\mathsf{phy}}}(a, s') = \mathsf{degen}(\{\circ\})$ for all other actions a and states s'.

Recall the initial beliefs β_0 discussed in Section 3.2.3 such that $\beta_0(s_1) = 0.9$ and $\beta_0(s_2) = 0.1$. Under the model $m_{\mathsf{phy}}/{\equiv_{\mathsf{phy}}}$ that prevents using the X-ray, the physician will not learn what state he is in after taking the X-ray. Thus, his beliefs about his current state will be determined by his initial beliefs. Assuming the initial beliefs β_0, after taking the X-ray, the physician will first attempt to make a diagnosis himself since s_3 is significantly more likely than s_4. However, in the case where s_4 is actually the current state, diagnosis will not work and the physician will receive no reward. To account for this case, the physician will then send the X-ray x_2 to the specialist and then try to make a diagnosis again.

Note that in this case, the physician sends the X-ray x_2 to the specialist regardless of the taken X-ray's value. Thus, the physician does not use the taken X-ray despite performing an action involving an X-ray. This result is a kin to the fact that a tabloid asserting that a celebrity has a disease based upon no information about the celebrity cannot be accused of using medical information about the celebrity even when the tabloid's assertion is correct by luck. (While the tabloid did not violate the celebrity's privacy rights by using protected medical information, the tabloid may still have violated the celebrity's rights for other reasons.) This result may appear odd in the context of our example involving X-rays for two reasons. First, the model is an abstraction of the example compressing many possible X-rays into just two values (x_1 and x_2) based solely upon whether the physician may reach a diagnosis from it. In a less abstract model of the example, many different values would exist for the X-ray and physician would only be able to reach a diagnosis by sending the correct one to the specialist. Second, the example (originally from Chapter 2) presupposes that the physician may use the X-ray to reach a diagnosis. Thus, we should expect the quotient POMDP $m_{\mathsf{phy}}/{\equiv_{\mathsf{phy}}}$ to be counterintuitive. For this example, the real value of creating quotient POMDPs lays in modeling purposes other than diagnosis for which physician may not use the X-ray. Thus, we discontinue this example until discussing auditing for other disallowed purposes in Section 3.4.5.

3.4 Auditing

In the previous section, we presented the quotienting operator $\cdot/\cdot$ for restricting the information used in a POMDP. This operation works identically for non-redundant POMDPs (NPOMDPs). Thus, we may use information quotienting for auditing purpose restrictions.

Most policies do not categorically rule out using a type of information, but rather restricts the purposes for which the auditee may use certain types of information. In this case, the auditor can construct an NPOMDP m that models optimizing the satisfaction of the purpose in question. The auditor must then construct $m/{\equiv}$ where $\equiv$ is an equivalence relation constructed from the restrictions the policy puts on information usage. The auditor may then check whether optimizing $m/{\equiv}$ can justify the auditee's actions.

However, auditing for disallowed uses of information is more complex than auditing for disallowed actions. Whereas actions are observable and in many contexts may be recorded in a log file, when information is only used in the planning process itself, it is not directly observable. Thus, the auditor must infer from the actions of the auditee what information it used. We consider this process for both prohibitive and exclusivity rules before discussing automating the process in Section 3.5.

3.4.1 Prohibitive Rules

Consider a rule of a privacy policy that demands that some information is not used for a certain purpose p (a prohibitive rule). The auditee must treat observations that only differ in that information as the same while planning for that purpose. Thus, if the auditee is planning for the purpose p and m is a POMDP modeling the auditee's environment whose reward function measures the satisfaction of p, then the auditor may expect that the auditee's actions as recorded in the log are consistent with a strategy in $\mathsf{nopt}(m/{\equiv})$ where $\equiv$ relates each observation to all the other observations that differ only by the restricted information. If the auditee's actions are inconsistent with every strategy in $\mathsf{nopt}(m/{\equiv})$, then the auditor knows that auditee performed one or more of the following acts:

1. performed actions for some purpose other than p,

2. used the prohibited information,

3. failed to properly optimize $m/{\equiv}$ despite trying (is incompetent), or

4. used some model other m.

Each of these acts warrant the auditor's attention.

The auditor may further check whether the auditee's actions are consistent with $\mathsf{nopt}(m)$. If the actions are inconsistent with both $\mathsf{nopt}(m)$ and $\mathsf{nopt}(m/{\equiv})$, then the auditee either planned for a different purpose, was incompetent, or used a different model. If the auditee's actions are consistent with $\mathsf{nopt}(m)$, but not with $\mathsf{nopt}(m/{\equiv})$, then the auditor receives a strong suggestion that the auditee made use of the prohibited information while planning for the purpose p. However, it does not prove that the auditee used the information for the purpose p since it could also be the case that the auditee's actions result from the auditee following a strategy for some other purpose.

3.4.2 Advertising Example: Auditing for a Prohibitive Rule

Consider the example modeled in Section 3.2.2 and discussed in Section 3.3.2 in which the website is prohibited from using the visitor's sex for the purpose of marketing. Let the initial beliefs β_0 be as before: $\beta_0(\langle \mathsf{f}, \mathsf{f}, \varnothing\rangle) = 0.4$, $\beta_0(\langle \mathsf{f}, \bot, \varnothing\rangle) = 0.1$, $\beta_0(\langle \mathsf{m}, \mathsf{m}, \varnothing\rangle) = 0.4$, and $\beta_0(\langle \mathsf{m}, \bot, \varnothing\rangle) = 0.1$.

Suppose that the log shows $[\beta_0, \mathsf{lookup}, \langle \mathsf{f}, \varnothing\rangle, \beta_{\mathsf{f}}, \mathsf{ad}_1]$ where β_{f} is as defined in Section 3.2.2: $\beta_{\mathsf{f}} = \mathsf{degen}(\langle \mathsf{f}, \mathsf{f}, \varnothing\rangle)$. In this case, the auditor can easily tell that the website used the prohibited information since the intermediate belief state β_{f} is equal to $\mathsf{update}_{m_{\mathsf{adv}}}(\beta_0, \mathsf{lookup}, \langle \mathsf{f}, \varnothing\rangle)$ but is not equal to $\mathsf{update}_{m_{\mathsf{adv}}/\equiv'_{\mathsf{adv}}}(\beta_0, \mathsf{lookup}, \{\langle \mathsf{f}, \varnothing\rangle, \langle \mathsf{m}, \varnothing\rangle\})$ where the observation $\{\langle \mathsf{f}, \varnothing\rangle, \langle \mathsf{m}, \varnothing\rangle\}$ is the equivalence class of $\langle \mathsf{f}, \varnothing\rangle$ under $\equiv'_{\mathsf{adv}}$.

While not represented in the formal model m_{adv} we constructed for this example, in many contexts the auditor may be comfortable with the assumption that the only purpose for which the auditee could have performed the action ad_1 is advertising. Under this assumption, the auditor may determine that either the website used the information for a disallowed purpose or is incompetent. Thus, while the auditor cannot rule out with certainty the possibility that some other purpose led to the auditee's actions, the auditor may do so convincingly.

The above example used the intermediate beliefs of the website. For some systems modeled as an MDP, obtaining the intermediate states in a log seems plausible. However, the intermediate states under a belief MDP correspond to the auditee's subjective beliefs. Obtaining this information without asking the auditee, who could lie, seems difficult if not impossible. However, even without access to β_{f}, the auditor may reach the same conclusion provided he knows β_0. That is, suppose that the log just records that website started with initial beliefs β_0, performed lookup, and then performed ad_1. The auditor may determine the website's actions are consistent with $\mathsf{nopt}(m_{\mathsf{adv}})$ but not $\mathsf{nopt}(m_{\mathsf{adv}}/{\equiv'_{\mathsf{adv}}})$ since performing the action ad_2 but not ad_1 is optimal for $m_{\mathsf{adv}}/{\equiv'_{\mathsf{adv}}}$ under the initial beliefs β_0. In fact, the auditor does not even need to know that the website performed lookup since performing ad_1 implies that the website did so: without the information made available by lookup, the action ad_2 optimizes m_{adv} given the initial beliefs β_0. In this example, having access to the initial beliefs β_0 seems conceivable if common knowledge includes that females and males are equally likely, the database is correct and contains one 80% of the visitors, and the visitor has not already seen an advertisement.

Without access to the initial beliefs of the website, the auditor cannot conclude from just the fact that the website performed the action ad_1 that the website used the database's entry on the visitor's sex for marketing. The website might have started with the initial belief that all visitors are female. In this case, the website would optimally show ad_1 without checking the database. While this uses the website's beliefs about the visitor's sex, it does not use the restricted information, the database.

3.4.3 Exclusivity Rules

Consider a rule of a privacy policy that demands that some information is used only for a certain purpose p (an exclusivity rule). Given the logged behaviors of the auditee, the auditor must first determine whether they could be for the purpose p. That is, the auditor must determine whether they are consistent with $\mathsf{nopt}(m)$ where m is a model for optimizing the purpose p. If they could be for the purpose p, the audit finishes without finding any violation.

If the sequence of actions performed by the auditee cannot be for the purpose p, then the auditor must determine whether the auditee used the restricted information in selecting the sequence of actions. This process is similar to the process used for auditing prohibitive rules. However, now the auditor must find a second purpose p' that can explain the actions. If the actions are consistent with $\mathsf{nopt}(m')$ but not $\mathsf{nopt}(m'/{\equiv})$ where m' is a model for the purpose p' and $\equiv$ relates observations that only differ in the restricted information, then the auditor may conclude that if the auditee performed the actions for the purpose p', then it violated the policy.

The auditor may repeat this process for every purpose p' of which the auditor can conceive. For each such alternative purpose p' the auditor may test whether the actions of the auditee are consistent with optimizing $m'/{\equiv}$ where m' is a POMDP for the purpose p'. If the auditor finds an alternative p' such that the auditee's actions are consistent with optimizing $m'/{\equiv}$, then the auditor has found an innocent explanation for the auditee's behavior. If the auditor can find no such purpose, then the auditor may be fairly confident that the auditee violated the policy. However, the auditor cannot be absolutely certain since the auditor might have failed to conceive of an exculpatory purpose or the auditee could be incompetent and failed to correctly optimize for a purpose while using only allowed information.

Alternatively, the auditor may ask the auditee to supply the alternative purpose p' and the auditor may check only that purpose. If the auditee cannot provide an alternative purpose p' such that its actions are consistent with optimizing p' without using the restricted information, the auditor may conclude that the auditee either violated the policy or is incompetent.

3.4.4 Advertising Example: Auditing for an Exclusivity Rule

Again returning to the example formalized as m_{adv}, suppose that the policy stated that a visitor's sex may only be used for the purpose of identification. Suppose that the website performed the action lookup followed by ad_1. Under the assumption that the website would only perform the action ad_1 for the purpose of advertising, the auditor may focus on advertising as the single alternative purpose. As explained in the example for prohibitive rules (Section 3.4.2), the auditor may conclude that the auditee used the patient's sex for the purpose of marketing indicating a violation of this exclusivity rule.

3.4.5 Physician Example: Auditing for an Exclusivity Rule

Let us return to the physician example formalized in Section 3.2.3. In this example, the physician is allowed to use the X-ray only for diagnosis. The physician starts with initial beliefs β_0 such that $\beta_0(s_1) = 0.9$,

$\beta_0(s_2) = 0.1$, and $\beta_0(s) = 0$ for all other states s.

Suppose the auditor has a log that shows that the physician starts with the initial beliefs β_0, performs that action take, observes x_1, and performs $\mathsf{send}x_1$. Despite not recording the intermediate beliefs of the physician, this log shows that the physician's actions were not for the purpose of diagnosis since the physician could make the diagnosis without sending the X-ray to the specialist.

Since these actions are not for the purpose of the diagnosis, the physician is prohibited from using the X-ray while selecting these actions. If the physician obeyed the policy, there must exist another purpose p' such that these actions are optimal for $m'/{\equiv_{\mathsf{phy}}}$ where m' is a model for satisfying p' and $\equiv_{\mathsf{phy}}$ relates the two X-rays. The existence of such a purpose may strike the auditor as highly suspect since the primary effect of the action take is to produce information that the physician must ignore. Furthermore, that the physician observes x_1 and then performs the corresponding action $\mathsf{send}x_1$ is a striking coincidence.

The auditor may decided to interview the physician to determine what purpose he might have had in mind. The physician could offer purposes that would explain the actions without using the X-ray. For example, the alternative purpose of increasing costs is served by taking an unused X-ray and sending some X-ray (not necessarily the correct one) to a specialist. That the X-ray taken and sent could be the same by coincidence is possible in this example. (In a more realistic example with many more possible X-rays, it becomes significantly less plausible.) However, while not previously discussed, the physician is likely to be governed by additional policies that would make such a purpose illegitimate.

If the physician cannot produce a legitimate purpose for which his actions are optimal and non-redundant, then the auditor has found that the physician committed a violation. The nature of the physician's violation could take one of two forms. First, it might have been violation of the prohibition against using the X-ray for a purpose other than diagnosis. Second, the physician might have performed an action for an otherwise illegitimate purpose.

3.5 Auditing Algorithm

In this section, we provide an algorithm that determines whether a behavior could have resulted from optimizing a POMDP modeling a purpose and quotiented by an equivalence relation modeling a restricted class of information. The above examples (Sections 3.4.2, 3.4.4, and 3.4.5) illustrate not only that such an algorithm can aid an auditor, but also that the auditor must make numerous other determinations. For example, the auditor must also determine whether a purpose is illegitimate for an auditee given all the purpose restrictions of a policy. We leave automating these other determinations to future work.

Furthermore, while the above examples illustrate cases where the audit may make determinations without access to the auditee's beliefs, we focus on the case where the auditor interviews the auditee to determine its (purported) belief states. The auditor then checks whether the auditee's story is consistent with itself and with any logs that the auditor has. Our algorithm aids the auditor in determining whether the auditee's story is consistent.

Performing auditing in this fashion must be more focused than auditing using the algorithm of Section 2.3. The degree of automation of that algorithm allows the auditor to run it looking suspicious actions for every auditee. The costs of interviewing auditees may prohibit such routine auditing in the case of POMDPs. Rather, our approach for auditing POMDPs is better restricted to investigating auditees found to be suspicious through other means.

```
AuditNPOMDPApprox($m = \langle \mathcal{S}, \mathcal{A}, t, r, \mathcal{O}, \nu, \gamma \rangle$, $\equiv$, $b = [\beta_1, a_1, o_1, \beta_2, a_2, o_2, \ldots, \beta_n, a_n, o_n]$):
  $m' = \langle \mathcal{S}, \mathcal{A}, t, r, \mathcal{O}/{\equiv}, \nu/{\equiv}, \gamma \rangle$
  if (ImpossiblePOMDP($m'$, $b$))
    return true        //  behavior impossible for NPOMDP $m'$
  $\langle \mathsf{V}^*_{\mathsf{low}}, \mathsf{V}^*_{\mathsf{up}} \rangle :=$ SolvePOMDPApprox($m'$)
  for ($i := 1$; $i \leq n$; $i$++):
    if ($\mathtt{Q}^*(m', \mathsf{V}^*_{\mathsf{up}}, \beta_i, a_i) < \mathsf{V}^*_{\mathsf{low}}(\beta_i)$):
      return true        //  action suboptimal
    if ($\mathtt{Q}^*(m', \mathsf{V}^*_{\mathsf{up}}, \beta_i, a_i) \leq 0$ and $a_i \neq \mathsf{stop}$):
      return true        //  action redundant
  return false
```

Figure 3.4: The algorithm AuditNPOMDPApprox. $\mathtt{Q}^*$ is defined in text.

While we provide both an exact and an approximation algorithm for MDPs, we only provide an approximation algorithm for POMDPs since exactly solving POMDPs is undecidable [Mad00] (conjectured earlier in [PT87]). Figure 3.4 shows our algorithm, called AuditNPOMDPApprox. The core of the algorithm is similar to AuditNMDPApprox in that it also checks whether each action that auditee performed is optimal for the state from which the auditee performed the action. This core of the algorithm is also closely related to goal inference algorithms that use POMDPs [BST11, RG11]. (See Section 7.4 for a detailed discussion.) However, AuditNPOMDPApprox differs from these algorithms by considering information use.

The algorithm takes as inputs a POMDP m, an equivalence relation $\equiv$, and a log that records a behavior $b = [\beta_1, a_1, o_1, \beta_2, a_2, o_2, \ldots, \beta_n, a_n, o_n]$ such that the audited agent is operating in the environment m under a policy prohibiting information as described by $\equiv$ and took action a_i from belief state b_i for all $i \leq n$. AuditNPOMDPApprox returns whether the agent's behavior, as recorded in b, is inconsistent with optimizing the POMDP $m/{\equiv}$.

The inputs to AuditNPOMDPApprox may either be created by the auditor based upon his examination of the auditee's behavior or constructed by the auditee and provided to the auditor. In the second case, the auditor must be mindful that the auditee might provide false information. However, even in this case, the algorithm can still help auditor determine whether the auditee's explanation is consistent.

AuditNPOMDPApprox operates by first constructing the quotient POMDP $m' = m/{\equiv}$ from m and $\equiv$ using its definition. Second, it uses the sub-routine ImpossiblePOMDP(m', b), shown in Figure 3.5, to check whether the given behavior b is possible for m'. For each i, AuditNPOMDPApprox then checks whether performing the recorded action a_i in belief state β_i is optimal under $m/{\equiv}$. We use an approximation algorithm to solve for the value of performing a_i in β_i (i.e., $Q^*_{m/\equiv}(\beta_i, a_i)$) and the optimal value $V^*_{m/\equiv}(\beta_i)$. For soundness, we require an approximation algorithm SolvePOMDPApprox that produces both lower bounds $\mathsf{V}^*_{\mathsf{low}}$ and upper bounds $\mathsf{V}^*_{\mathsf{up}}$ on $V^*_{m/\equiv}(\beta_i)$. Many such algorithms exist (e.g., [Son78, KLC98, ZH01, SS05, KHL08, PKK11]). For each β_i and a_i in ℓ, AuditNPOMDPApprox checks whether these bounds show that $Q^*_{m/\equiv}(\beta_i, a_i)$ is strictly less than $V^*_{m/\equiv}(\beta_i)$. If so, then the action a_i is sub-optimal for β_i and AuditNPOMDPApprox returns true.

$\mathtt{Q}^*(m', \mathsf{V}^*_{\mathsf{up}}, \beta, a)$ is a function that uses $\mathsf{V}^*_{\mathsf{up}}$ to return an upper bound on $Q^*_{m/\equiv}(\beta, a)$. $\mathtt{Q}^*(m', \mathsf{V}^*_{\mathsf{up}}, \beta, a)$

```
ImpossiblePOMDP$(m = \langle \mathcal{S}, \mathcal{A}, t, r, \mathcal{O}, \nu, \gamma \rangle,\ b = [\beta_1, a_1, o_1, \beta_2, a_2, o_2, \ldots, \beta_n, a_n, o_n])$:
for ($i := 1; i \leq n; i{+}{+}$):
  if ($\beta_i \notin \mathsf{Dist}(\mathcal{S})$):
    return true    // $\beta_i$ is not a belief state
  if ($a_i \notin \mathcal{A}$):
    return true    // $a_i$ is not an action
  if ($o_i \notin \mathcal{O}$):
    return true    // $o_i$ is not an observation
for ($i := 1; i < n; i{+}{+}$):
  if ($\mathsf{update}_m(\beta_i, a_i, o_i) = \beta_{i+1}$):
    return true    // $\beta_{i+1}$ unreachable from $\beta_i$ under $a_i$ and $o_i$
  for ($j := i + 1; j \leq n; j{+}{+}$):
    if ($\beta_i = \beta_j$ and $a_i \neq a_j$):
      return true    // no stationary strategy could have produced the behavior
return false
```

Figure 3.5: The algorithm ImpossiblePOMDP. Returns whether the given behavior is possible for the given POMDP.

equals:

$$R_{m'}(\beta, a) + \gamma \sum_{O \in \mathcal{O}/\equiv} N_{m'}(\beta, a))(O) * \mathsf{V}^*_{\mathsf{up}}(\mathsf{update}_{m'}(\beta, \sigma(\beta), O))$$

with $R_{m'}$ and $N_{m'}$ as defined in Equations 3.7 and 3.8 of Section 3.2.

3.5.1 Correctness

As proved in Theorem 4, AuditNPOMDPApprox is sound: if AuditNPOMDPApprox$(m, \equiv, b)$ returns true, then the behavior b is a not a non-redundant and optimal behavior for the NPOMDP $m/\equiv$. If the auditor provides to AuditNPOMDPApprox a model m that accurately describes the environment of an auditee, an equivalence relation $\equiv$ that accurately characterizes the information restrictions of a policy, and a behavior b that accurately describes the behavior of the auditee, then to AuditNPOMDPApprox$(m, \equiv, b)$ returning true implies that the auditee deviated from behavior acceptable under the policy. This deviation could either be the auditee optimizing some other purpose, using information it should not have, using a different POMDP model of its environment, or failing to correctly optimize the POMDP. Each of these possibilities should concern the auditor and is worthy of further investigation.

The proof of correctness follows the same outline as the proof for the MDP approximation algorithm. First, we show local conditions for testing the global property of a behavior being non-redundant and optimal (Lemma 5). Second, we reason about the code to show that it correctly checks these local conditions (Lemma 6 and Theorem 4).

The following lemma (Lemma 5) shows that a behavior b being optimal and non-redundant for a NPOMDP m (i.e., that b is in $\mathsf{nbehv}(m)$) implies the following three properties about b. First, b must actually be a possible behavior of m. Second, every action a_i in b must be optimal for the belief state β_i in which the auditee performed the action a_i (i.e., $Q^*_m(\beta_i, a_i) = V^*_m(\beta_i)$). Third, every action a_i in b must be non-redundant in that it must produce a higher expected total discounted reward when taken in the belief state β_i in which the auditee performed it than the action stop when performed from β_i (i.e., $a_i \neq \mathsf{stop}$ implies that $Q^*_m(\beta_i, a_i) > Q^*_m(\beta_i, \mathsf{stop}) = 0$).

Lemma 5. *For all NPOMDPs m, if the behavior $b = [\beta_1, a_1, o_1, \ldots, \beta_n, a_n, o_n]$ is in $\mathsf{nbehv}(m)$, then b is a possible behavior of m, and for all $i \leq n$, $Q^*_m(\beta_i, a_i) = V^*_m(\beta_i)$ and $a_i \neq \mathsf{stop}$ implies that $Q^*_m(\beta_i, a_i) > 0$.*

Proof. Suppose that b is in $\mathsf{nbehv}(m)$. By the definition of $\mathsf{nbehv}(m)$, b is a possible behavior of m and $b' = [\beta_1, a_1, \beta_2, a_2, \ldots \beta_n, a_n]$ is in $\mathsf{nbehv}(\mathsf{bmdp}(m))$. By the definition of *possible* for POMDPs, b' is possible behavior of $\mathsf{bmdp}(m)$ and for all $i < n$, $\beta_{i+1} = \mathsf{update}_m(\beta_i, a_i, o_i)$.

By Lemma 3, b' being in $\mathsf{nbehv}(\mathsf{bmdp}(m))$ implies that b' is a possible behavior of $\mathsf{bmdp}(m)$, and for all $i \leq n$, $q^*_{\mathsf{bmdp}(m)}(\beta_i, a_i) = v^*_{\mathsf{bmdp}(m)}(\beta_i)$ and $a_i \neq \mathsf{stop}$ implies that $q^*_{\mathsf{bmdp}(m)}(\beta_i, a_i) > 0$.

By Proposition 3, for all $i \leq n$, $v^*_{\mathsf{bmdp}(m)}(\beta_i) = V^*_m(\beta_i)$ and $q^*_{\mathsf{bmdp}(m)}(\beta_i, a_i) = Q^*_m(\beta_i, a_i)$. Thus, for all $i \leq n$, $Q^*_m(\beta_i, a_i) = V^*_m(\beta_i)$ since $q^*_{\mathsf{bmdp}(m)}(\beta_i, a_i) = v^*_{\mathsf{bmdp}(m)}(\beta_i)$. Furthermore, for all $i \leq n$, $a_i \neq \mathsf{stop}$ implies that $Q^*_m(\beta_i, a_i) > 0$ since for all $i \leq n$, $a_i \neq \mathsf{stop}$ implies that $q^*_{\mathsf{bmdp}(m)}(\beta_i, a_i) > 0$. □

AUDITNPOMDPAPPROX checks each of the three conditions that Lemma 5 shows are implied by a behavior being optimal and non-redundant. If any of them are false, then the behavior is not optimal and non-redundant by the contrapositive of Lemma 5. The following lemma (Lemma 6) shows that the sub-routine IMPOSSIBLEPOMDP(m, b) correctly checks whether the behavior b is possible for the NPOMDP m.

Lemma 6. *For all POMDPs m and finite sequences b, if* IMPOSSIBLEPOMDP(m, b) *returns true, then b is not a possible behavior of m.*

Proof. b must have the form $[\beta_1, a_1, o_1, \beta_2, a_2, o_2, \ldots, \beta_n, a_n, o_n]$ for some n. If IMPOSSIBLEMDP returns true, then at least one of the following is true: (1) there exists $i \leq n$ such that β_i is not a belief state of m (Line 53), (2) there exists $i \leq n$ such that a_i is not an action of m (Line 55), (3) there exists $i \leq n$ such that o_i is not an observation of m (Line 57), (4) there exists $i < n$ such that $\mathsf{update}_m(\beta_i, a_i, o_i) \neq \beta_{i+1}$ (Line 60), or (5) there exists $i < n$ and j where $i < j \leq n$ such that $s_i = s_j$ and $a_i \neq a_j$ (Line 63). Conditions (1), (2), (3) each imply that b is not in $(\mathcal{S} \times \mathcal{A} \times \mathcal{O})^*$, which implies that b is not a possible behavior for m. Condition (4) implies that b is not a possible behavior of m as well.

Let $b' = [\beta_1, a_1, \beta_2, a_2, \ldots \beta_n, a_n]$. If Condition (5) holds, there exists $i < n$ and j where $i < j \leq n$ such that $s_i = s_j$ and $a_i \neq a_j$. This implies that b' is not a possible behavior of $\mathsf{bmdp}(m)$ by Lemma 1. Thus, b is not a possible behavior of m. □

The soundness theorem below (Theorem 4) reasons about the code of AUDITNPOMDPAPPROX to show that it correctly checks the location conditions mentioned by Lemma 5. It uses Lemma 6 to justify the sub-routine IMPOSSIBLEPOMDP.

Theorem 4 (Soundness)**.** *For all POMDPs m, equivalence relations $\equiv$ over the observation space of m, and finite sequences b, if* AUDITNPOMDPAPPROX$(m, \equiv, b)$ *returns true, then b is not in* $\mathsf{nbehv}(m/\equiv)$.

Proof. b must have the form $[\beta_1, a_1, o_1, \beta_2, a_2, o_2, \ldots, \beta_n, a_n, o_n]$ for some n. If AUDITNPOMDPAPPROX returns true, then at least one of the following conditions is true: (1) IMPOSSIBLEPOMDP(m', b) returns true (Line 33), (2) there exists $i \leq n$ such that $\mathtt{Q}^*(m', \mathtt{V}^*_{\mathsf{up}}, s_i, a_i) < \mathtt{V}^*_{\mathsf{low}}(s_i)$ (Line 37), or (3) there exists

$i \leq n$ such that $\mathtt{Q}^*(m', \mathtt{V}^*_{\mathsf{up}}, s_i, a_i) \leq 0$ and $a_i \neq \mathsf{stop}$ (Line 39). If (1) is true, then b is not a possible behavior of m' by Lemma 6. If (2) is true, then for that i, $Q^*_{m'}(s_i, a_i) \neq V^*_{m'}(s_i)$ since $Q^*_{m'}(s_i, a_i) \leq \mathtt{Q}^*(m', \mathtt{V}^*_{\mathsf{up}}, s_i, a_i) < \mathtt{V}^*_{\mathsf{low}}(s_i) \leq V^*_{m'}(s_i)$. If (3) is true, then for that i, $a_i \neq \mathsf{stop}$ does not imply that $Q^*_{m'}(s_i, a_i) > 0$ since $a_i \neq \mathsf{stop}$ and $Q^*_{m'}(s_i, a_i) \leq \mathtt{Q}^*(m', \mathtt{V}^*_{\mathsf{up}}, s_i, a_i) \leq 0$. Thus, under each of these cases, Lemma 5 shows that b is not in $\mathsf{nbehv}(m')$. This fact implies that $\mathsf{log}^{-1}(b) \cap \mathsf{nbehv}(m/\!\equiv)$ is empty since $\mathsf{log}^{-1}(b) = \{b\}$ and $m' = m/\!\equiv$. □

Thus, if AUDITNPOMDPAPPROX returns true, either the agent optimized some other purpose, used information it should not have, used a different POMDP model of its environment, or failed to correctly optimize the POMDP.

If the algorithm returns false, then auditor cannot find the agent's behavior inconsistent with an optimal strategy and the auditor should spend his time auditing other agents. However, AUDITNPOMDPAPPROX is incomplete and such a finding does not mean that the agent surely obeyed the policy. For one, a better approximation of $V^*_{m/\equiv}$ might actually show that $Q^*_{m/\equiv}(\beta_i, a_i) < V^*_{m/\equiv}(\beta_i)$ for some i. More fundamentally, incompleteness remains even with an exact POMDP solver: it is possible that the agent was planning with a different purpose in mind or that the agent used disallowed information, but that, by coincidence, the agent performed actions consistent with the allowed purpose and information use. While the auditor might want to find such illicit motivations, the agent has tenable deniability and the auditor cannot determine whether the agent obeyed the policy. Given the impossibility of such a determination and that the behavior is consistent with allowed behavior (the agent did the right thing for the wrong reasons), the auditor's time is better spent elsewhere.

Chapter 4

Application to Medical Records

4.1 The Healthcare Domain

In this chapter, we apply our work to the healthcare domain. The move to electronic health records both introduces new threats to the privacy of patients and allows old ones to be addressed more completely than before. Some of these threats are:

1. curious employees looking up celebrity records,
2. curious employees looking up the records of people they know,
3. massive data loss from an insider copying records,
4. massive data loss from accidents such as losing a laptop,
5. massive data loss from outsiders attacking the system,
6. healthcare providers systematically taking liberties with privacy for profit by taking advantage of vague laws,
7. honest employees inadvertently violating privacy by accident, and
8. honest employees not sharing needed information despite being allowed to.

The threats involving massive data loss are enabled by the move to electronic health records as paper records are impractical to steal or copy in large qualities. Threat 6 is acerbated by electronic health records (EHRs) as they make new uses of information profitable such as data mining for corporate research. We hope the improved auditing abilities of EHR will reduce the risks of the remaining ones.

The first three involve purpose based violations. The next two appear to be standard security problems and, thus, we do not consider them further. The antepenultimate one could be a purpose violation depending upon one's interpretation of the law. The last two often involve purposes.

Our formalism may mitigate those threats involving purposes. For example, it provides the basis for automated auditing, which could discourage curious employees. In addition to clarifying the meaning of purpose restrictions found in laws, our formalism may aid understanding other vague requirements. For example, HIPAA requires healthcare providers to limit disclosure of medical records to the “minimum necessary to accomplish the intended purpose” [U.S10a]. Our formalism for *purpose* may be combined with a formalism for *minimum necessary* to understand this requirement.

In the remainder of this chapter, we explore one possible use of our formalism in the healthcare domain. We look at how it can help policy subjects understand the implications of a policy in the emerging domain of *Regional Health Information Organizations* (RHIOs). The American Reinvestment and Recovery Act provides funding to promote RHIOs. RHIOs are to collect and store health records for individuals living in a defined region. Health care providers working in those areas may gain access to patient records from their local RHIO. By making records more available, RHIOs hope to improve patient care while lowering costs.

As RHIOs are a new technology and do not directly provide treatment, arguments may arise over whether their use is actually for treatment. A physician considering reading such a record may find the circumstances too complex to understand without help. However, we cannot expect the physician to perform the modeling required to use our auditing algorithm either.

To shed light on this issue, we show how to apply our formalism to two uses of RHIOs and ask whether they are for the purpose of treatment. Compliance officers at an RHIO may use our algorithm to audit simulated logs of possible future uses and determine which actions the restriction allows. The compliance officers may generalize these quantitative results to a qualitative operating procedure, such as *the physician may read records of patients with whom he does not have a current relationship only when seeing that patient in the future is highly likely*.

Below, we show an example of reasoning that could lead to this procedure. First, we look at uploading information to an RHIO. Second, we look at reading information from an RHIO. Our examination of the reading information considers multiple different models of a RHIO of varying complexity. After starting with a simple model, we consider extensions modeling other actions a physician may take, how reading a patient’s record could help other patients, multiple time steps, and learning information. We find that in some cases, a physician is justified in reading records of patients with whom the physician does not have a current relationship.

Compliance officers at an RHIO may find these results helpful while creating operating procedures. We find, for example, under a model of a large hospital, that physician should not read a patient’s record unless the physician has a reason to believe that the patient is much more likely than average to seek care (under the policy that physicians may read patient records only for treatment). However, the policy is more lenient for a model of a small hospital.

4.2 Uploading Information

Our formalism shows that uploading information to RHIOs is for the purpose of treatment even when they go unused. For example, consider a physician Dr. X who uploads a patient Y’s record to an RHIO, which goes unused. Nevertheless, Dr. X’s action is for treatment since a reasonable model would show with some probability patient Y could end up in an accident resulting in an emergency room visit at a hospital he has never attended before. In such a case, immediate access to Y’s record from the RHIO would improve the abilities of the emergency physicians to treat Y. Thus, under our formalism of *purpose*, Dr. X’s posting the medical record is for treatment.

Figure 4.1 shows such a model. In it, the physician has a choice between posting the record or not. Either way, with some probability p, the patient will need care from a facility that has access to the RHIO.

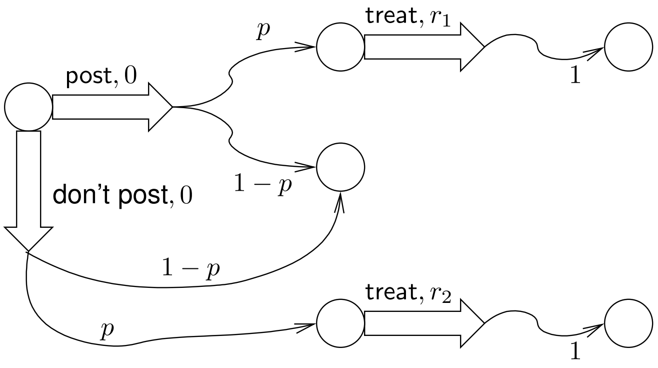

Figure 4.1: MDP representing posting records to an RHIO. $r_1 > r_2$.

In the case where Dr. X posted the record, the facility provides treatment at a high quality level of r_1. In the case the were Dr. X did not post the record, the facility delivers treatment at a lower level r_2 where $r_1 > r_2$. For such a model, the optimal action is to post the record. Implicit in the model is that posting a patient's record does not alters the probability that the patient will seek care.

4.3 Reading Information

Now we consider whether a physician may read a record of an RHIO for the purpose of treatment even when that physician currently has no plans to treat that patient. At first, the answer appears to be clearly *no* as a physician who is not engaged in treating a patient appears to have no reason to look at that patient's record. A simple model could formalize this intuition by showing the action of reading the record having value for treatment and not leading to a state that is any better for treatment.

However, such a physician could use similar reasoning to that which justified posting medical records in Section 4.2 to argue that such record reading is for the purpose of treatment. For example, suppose that Dr. Z reads patient Y's record despite not being involved in Y's treatment. Dr. Z may argue that he is "proactively" reading Y's record because with some non-zero probability Y will come to Dr. Z in the future as a patient and will receive better treatment as a result of Dr. Z already being familiar with Y. Dr. Z might formalize his argument with a model like the one shown in Figure 4.1, but with reading the record in place of posting it. Such a model shows more details of the environment than the our initial model. Unlike our initial model, this more detailed model vindicates the physician.

However, this model does not help guide the physician in picking which record to read. To do so, we need to replace the generic read action with a separate one for each record that the physician could read. Figure 4.2 shows such a model. For simplicity, we presume that reading a record will only affect Dr. Z's treatment of the next patient. (Later in this section, we will remove this restriction by modeling multiple sequential patient treatments.) This model has n patients in the RHIO. p_i represents the probability that Dr. Z will see patient i next. $p_\varnothing$ represents the probability that Dr. Z does not have a next patient. The action read_i represents reading the ith patient's record. The reward ρ^{t}_i is a reward measuring how well Dr. Z treats patient i without seeing that patient's record beforehand; δ^{r}_i represents the improvement in how well

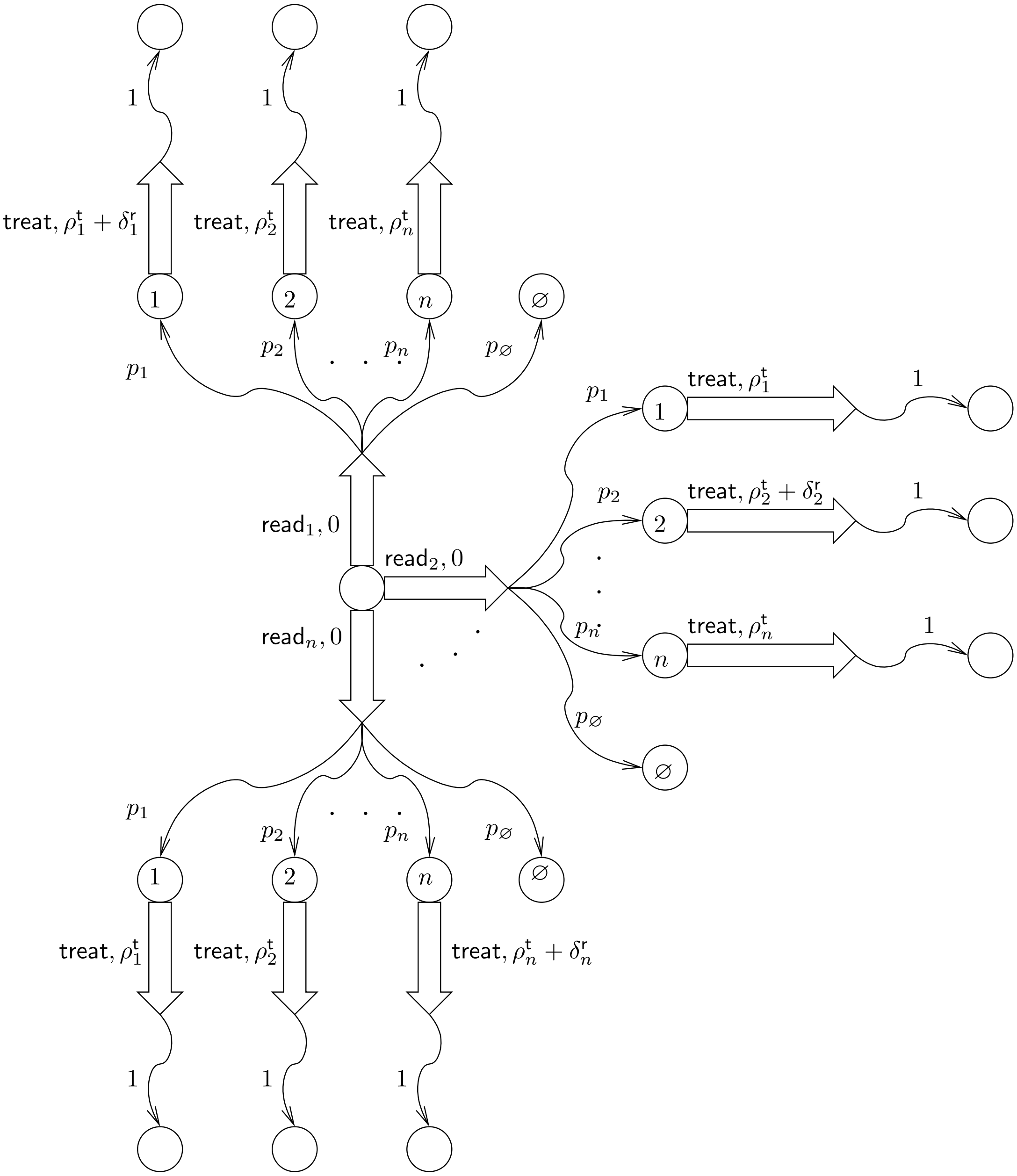
1
1
1
treat, $\rho_1^{\mathsf{t}} + \delta_1^{\mathsf{r}}$
treat, ρ_2^{t}
treat, ρ_n^{t}
1
2
n
$\varnothing$
p_1
p_2
p_n
$p_\varnothing$
p_1
treat, ρ_1^{t}
1
p_2
treat, $\rho_2^{\mathsf{t}} + \delta_2^{\mathsf{r}}$
1
read$_1$, 0
read$_2$, 0
read$_n$, 0
p_n
treat, ρ_n^{t}
1
$p_\varnothing$
p_1
p_2
p_n
$p_\varnothing$
treat, ρ_1^{t}
treat, ρ_2^{t}
treat, $\rho_n^{\mathsf{t}} + \delta_n^{\mathsf{r}}$
1
1
1

Figure 4.2: A simple MDP representing reading records from an RHIO. Each state is labeled with the identifying number of the patient that needs treatment.

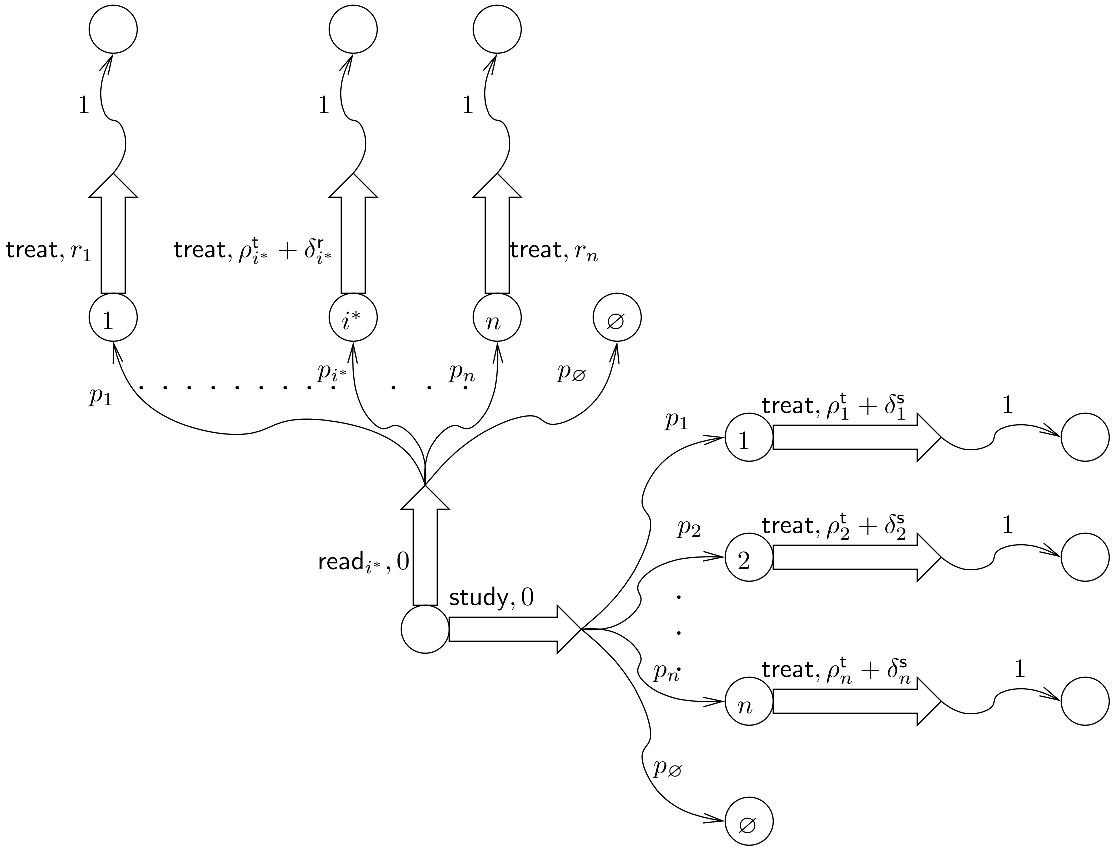
1
1
1
treat, r_1
treat, $\rho^{\mathsf{t}}_{i^*} + \delta^{\mathsf{r}}_{i^*}$
treat, r_n
1
i^*
n
$\varnothing$
p_1
p_{i^*}
p_n
$p_\varnothing$
read$_{i^*}$, 0
study, 0
p_1
treat, $\rho^{\mathsf{t}}_1 + \delta^{\mathsf{s}}_1$
1
1
p_2
treat, $\rho^{\mathsf{t}}_2 + \delta^{\mathsf{s}}_2$
1
2
p_n
treat, $\rho^{\mathsf{t}}_n + \delta^{\mathsf{s}}_n$
1
n
$p_\varnothing$
$\varnothing$

Figure 4.3: A more detailed MDP representing reading records from an RHIO. Each state is labeled with the identifying number of the patient that needs treatment.

Dr. Z may treat patient i with seeing the record beforehand. This model is optimized by reading the record for patient i that maximizes the expected improvement $p_i * \delta^{\mathsf{r}}_i$. Let i^* represent the value of i that optimizes $p_i * \delta^{\mathsf{r}}_i$. (For simplicity, we presume uniqueness.)

At this point, the reader might be worried that our formalism would allow anyone to read Y's record in the case where $i^* = Y$ because with some non-zero odds this could improve treatment. However, with an even more detailed model, the auditor can show that this is not the case. In particular, the auditor could also model the ability of the physician to further his studying by studying medical advances instead of patient records. Such a model is shown in Figure 4.3. The model does not show all the possible read$_i$ actions. Rather it shows only read$_{i^*}$ where i^* is the patient that maximizes the expected improvement in treatment as above. Since this is the only read$_i$ action that an optimizing agent would choose (as shown in the model of Figure 4.2), this simplification does not affect planning. The model now includes the action study. Studying yields an improvement of δ^{s}_i in the level of treatment the physician may provide to patient i. The model shows that the physician should only read the record of patient i^* when $p_{i^*} * \delta^{\mathsf{r}}_{i^*} \geq \sum_{i=1}^{n} p_i \delta^{\mathsf{s}}_i$.

It may be difficult to assign exact values to the rewards ρ^{t}_i, δ^{r}_i, and δ^{s}_i. However, the auditor can reason more abstractly. For example, if the auditor can shown that p_{i^*} is about 0.01, then the improvement $\delta^{\mathsf{r}}_{i^*}$ from reading the record of i^* must be at least 100 times that of the expected improvement from studying (i.e., $\sum_{i=1}^{n} p_i \delta_i^{\mathsf{s}}$). For many patients and physicians, such a large improvement from reading the record ahead of seeing the patient seems unlikely. As p_{i^*} goes to 1, the physician will be justified in reading the record even if doing so is no better than studying. This effect matches our intuition that physicians should be allowed to read the records of patients that they are scheduled to see (i.e., with p_{i^*} near 1). Likewise, as the number of possible patients decreases, we would expect p_{i^*} to increase and justify the physician reading the record. This effect matches our intuition that a physician inspecting the records of his small family practice is less suspicious than a physician looking over records of a large health system, which is still less suspicious than a physician looking over records of an entire RHIO.

In summary, while the above modeling does not provide a definitive answer without computing all the reward values, it does provide insight into the problem. In particular, it shows that unless p_{i^*} is high or $\delta_{i^*}^{\mathsf{r}}$ is large compared to the value of studying, then the physician will be better off studying.

4.4 Interactions among Patient Information

We went from a simple model that showed that Dr. Z could not read any records, to a more detailed one showing that Dr. Z could always read the records of i^*, to an even more detailed model showing that it depends upon whether studying would be better use of Dr. Z's time. This progression of models shows that unlike some properties (such as safety properties), an over- or under-approximation might not be sound. Thus, an auditor might worry that the above model is still not detailed enough to produce the correct result. While we cannot prove that it is detailed enough, we can explore extensions of the model and show that result remains fairly stable under them.

One might object to the dichotomy of studying and reading records since, in some cases, reading a record for one patient might aid in the treatment of another patient. We can model this secondary effect of reading a record by having a different reward ρ_{ij}^{t} for all i and j from 1 to n representing the level of treatment that the physician provides to patient i after reading the record j. (δ_i^{r} would equal $\rho_{ii}^{\mathsf{t}} - \rho_i^{\mathsf{t}}$ under this notation.) In this case, instead of calculating i^* as above, we would calculate it as that value of j that maximizes $\sum_i p_i \rho_{ij}^{\mathsf{t}}$. Despite complicating the calculation, the final result still depends upon whether reading the records is more helpful for treatment than studying.

With the increasing use of approaches like data mining for diagnosis, we could consider the reading of every medical record for the purpose of comparative study of patients. One could model this either as a series of reads and rewards that depends upon every previously read record, or as a single *read every record* action. However, a physician does not need to actually read every record to use machine learning. Rather the system could process the records and show the physician only the results. While the results might contain sensitive information, it is likely to be less sensitive than the actual records themselves. Thus, the physician could benefit from the information in the records without actually seeing them in their entirety To capture this in our model, we would have to decompose the action into the smaller actions that could make it up: reading records, processing the records, reading results. As before, we could find that under normal circumstances, the physician's time is better spent studying aggregate results than reading records individually. This case shows the effects of another way to make a model more detailed: decomposing an action allows an auditor to pass more fine-grained judgment.

One might want to make the purpose of treatment parametric in each patient and interpret *treatment* as meaning treatment for some patient. (We do not believe this interpretation to have been the one intended by the authors of HIPAA. Indeed, this is not the interpretation taken by DeYoung et al. in their formalization of HIPAA.) Making this change results in looking at patient i's record as being for the *treatment of patient* i whenever $\delta_i^{\mathsf{r}} > \delta_i^{\mathsf{s}}$, which seems rather likely. Intuitively, if one is concerned with the treatment of a single patient instead of treatment in general, it makes sense to read that patient's medical record. (Previously, we implicitly modeled the purpose of treatment as the sum of the values for treating each of the patients.)

4.5 Multiple Time Steps

In the above examples, for simplicity, we modeled the physician as having only a single time step during which to either read a record or study. In actuality, physicians have multiple chunks of time that may be spent on some combination of these two actions. We are interested in finding out how the presence of multiple time steps affects which behaviors are allowable to the physician. In particular, we would like to know if it is still the case that The model shows that the physician should only read the record of patient i^* when $p_{i^*} * \delta_{i^*}^{\mathsf{r}} \geq \sum_{i=1}^{n} p_i * \delta_i^{\mathsf{s}}$.

After constructing a model of multiple time steps, we will approach this problem by using the implementation of AUDITNMDPAPPROX algorithm discussed in Section 2.4 to see how accurately the above rule predicts when a physician may read a record. We find that at least for small numbers of steps, the above rule approximately holds.

4.5.1 Modeling

We assume that the effects of reading a medical record or studying wears off over time. Not only does this assumption model the limited nature of human memory, but also allows us to model an infinite number of time steps using a finite model. In particular, we encode the last h actions (reading, studying, and treating) of the physician in the states of the model. The reward for treating a patient depends upon this history.

We model this example as a family of NMDPs that depend upon the parameter h, the number of steps before a physician forgets something. For simplicity, we assume that the number of patients is equal to h as well. (Having more patients than the physician can remember cannot change his behavior.) In the case, where more than h patients are stored in an RHIO, we consider the subset of the RHIO that holds the patients that would benefit the most from having their records read (those that maximizes $p_i * \delta_i^{\mathsf{r}}$).

Formally, let m_{ex4}^{h} be the model for the parameter h (and others introduced below). $m_{\mathsf{ex4}}^{h} = \langle \mathcal{S}, \mathcal{A}, t, r, \gamma \rangle$ where the action space $\mathcal{A}$ is equal to $\{\mathsf{stop}, \mathsf{treat}, \mathsf{study}, \mathsf{read}_1, \ldots, \mathsf{read}_h\}$. The state space $\mathcal{S}$ is equal to $\{\mathsf{treat}, \mathsf{study}, \mathsf{read}_1, \ldots, \mathsf{read}_h\}^h \times \mathcal{C}$ where $\{\mathsf{treat}, \mathsf{study}, \mathsf{read}_1, \ldots, \mathsf{read}_h\}^h$ encodes a h-step history of the physician's actions and $\mathcal{C}$ is the set of possible conditions in which the physician may currently find himself. The history records, in order, which action the physician made in each of the h most recent steps before the current step unless the physician has taken the do-nothing action stop. Once the physician performs stop, the history is frozen at its current value and does not record the current or future stop actions. The history does not record the stop actions since they always result in returning to the current state making updating the history impossible. However, this failure to record the stop action does not alter the optimal strategy of the NMDP m_{ex4}^{h} since stop is of zero reward and results in a self-loop at every state. The set of conditions $\mathcal{C}$ is equal to $\{\emptyset, \mathsf{o}, 1, \ldots, h\}$ where $\emptyset$ represents no patients currently wanting to see the physician, o (short for *other*) represents a patient not in the RHIO attempting to see the physician, and i in $\{1, \ldots, h\}$ represents the ith patient of the RHIO attempting to see the physician.

form of state	action	reward
$\langle\langle a_1, \ldots a_h\rangle, c\rangle$ (any state)	study	0
$\langle\langle a_1, \ldots a_h\rangle, c\rangle$ (any state)	read_i	0
$\langle\langle a_1, \ldots a_h\rangle, \emptyset\rangle$	treat	0
$\langle\langle a_1, \ldots a_h\rangle, \mathsf{o}\rangle$	treat	$\rho^{\mathsf{t}}_{\mathsf{o}} + n * \delta^{\mathsf{s}}_{\mathsf{o}}$
$\langle\langle a_1, \ldots, \mathsf{read}_i, \ldots a_h\rangle, i\rangle$	treat	$\rho^{\mathsf{t}}_{\mathsf{o}} + \delta^{\mathsf{r}}_i + n * \delta^{\mathsf{s}}_i$
$\langle\langle a_1, \ldots a_h\rangle, i\rangle$ where read_i is not in $\langle a_1, \ldots a_h\rangle$	treat	$\rho^{\mathsf{t}}_i + n * \delta^{\mathsf{s}}_i$

Table 4.1: The rewards for m^h_{ex4}. In the last three rows, n stands for the number of instances of study in $\langle a_1, \ldots a_h\rangle$.

At each time step, the physician chooses whether to treat the current patient (if any), read a patient's record, study, or do nothing. This updates his history by replacing the oldest of the h entries with this choice. The condition also probabilistically updates to a value of $\mathcal{C}$. The transition function t is such that $t(\langle\langle a_1, a_2, \ldots, a_h\rangle, c\rangle, a)$ is equal to a distribution d over these possible next states. In particular, d depends upon additional parameters p_c for each c in $\mathcal{C}$. p_c provides the probability of c being the next condition in which the physician finds the hospital. The distribution d assigns the probability of p_c to the next state $\langle\langle a_2, \ldots, a_h, a\rangle, c\rangle$ for each c and the probability of 0 for all other states where a is the current action of the physician.

Table 4.1 lists the rewards for each state and action. The reward for the actions read_i or study is always 0. The reward for treat depends upon the state. For the state $\langle\langle a_1, \ldots a_h\rangle, \emptyset\rangle$, the reward is also 0. For the state $\langle\langle a_1, \ldots a_h\rangle, \mathsf{o}\rangle$, the reward is $\rho^{\mathsf{t}}_{\mathsf{o}} + n * \delta^{\mathsf{s}}_{\mathsf{o}}$ where $\rho^{\mathsf{t}}_{\mathsf{o}}$ is the base reward for treating a patient not in the database, n is the number of instances of study in $\langle a_1, \ldots, a_h\rangle$, and $\delta^{\mathsf{s}}_{\mathsf{o}}$ is the additional reward achieved per studying action. For the state $\langle\langle a_1, \ldots a_h\rangle, i\rangle$, if there exists j in $\{1, \ldots, h\}$ such that $a_j = \mathsf{read}_i$, the reward will be $\rho^{\mathsf{t}}_i + \delta^{\mathsf{r}}_i + n * \delta^{\mathsf{s}}_i$ where ρ^{t}_i is base reward for treating patient i, δ^{r}_i is the additional reward for having read the patient's record, and n and δ^{s}_i are as before. For the state $\langle\langle a_1, \ldots a_h\rangle, i\rangle$, if there does not exist j in $\{1, \ldots, h\}$ such that $a_j = \mathsf{read}_i$, the reward will be $\rho^{\mathsf{t}}_i + n * \delta^{\mathsf{s}}_i$. ρ^{t}_i, δ^{r}_i, δ^{s}_i, $\rho^{\mathsf{t}}_{\mathsf{o}}$, and $\delta^{\mathsf{s}}_{\mathsf{o}}$ are all additional parameters to the model. We also treat the discounting factor γ as a parameter.

The number of actions is $|\{\mathsf{stop}, \mathsf{treat}, \mathsf{study}, \mathsf{read}_1, \ldots, \mathsf{read}_h\}| = 3 + h$. The number of states is

$$|\{\mathsf{treat}, \mathsf{study}, \mathsf{read}_1, \ldots, \mathsf{read}_h\}^h \times \mathcal{C}| = (2 + h)^h * (2 + h) = (h + 2)^{h+1}$$

For every state s and action a except stop, each of the possible $h + 2$ conditions in $\mathcal{C}$ could arise in the next state from performing action a in state s. Presuming all the probability parameters p_c are non-zero, the resulting number of non-zero transitions is

$$|\mathcal{S}| * |\mathcal{A} - \{\mathsf{stop}\}| * |\mathcal{C}| + |\mathcal{S}| = (h+2)^{h+1} * (h+3-1) * (h+2) + (h+2)^{h+1} = (h+2)^{h+3} + (h+2)^{h+1}$$

where the second summand accounts for the self-loop under stop at each state.

Since m^2_{ex4} has 64 states and 1088 non-zero transitions, we cannot easily represent the whole model in a diagram. Thus, in Figure 4.4, we show just part of m^2_{ex4}. It shows only the part of the NMDP relevant to transitions from the states $\langle\langle \mathsf{s}, 2\rangle, \emptyset\rangle$ or $\langle\langle 2, \mathsf{s}\rangle, 2\rangle$. The part of m^2_{ex4} is sufficient to illustrate the possibility of multi-state cycles. In particular, it shows the possibility of executions of the following form

$$[\langle\langle \mathsf{study}, \mathsf{read}_2\rangle, \emptyset\rangle, \quad \mathsf{study}, \quad \langle\langle \mathsf{read}_2, \mathsf{study}\rangle, 2\rangle, \quad \mathsf{read}_2, \quad \langle\langle \mathsf{study}, \mathsf{read}_2\rangle, \emptyset\rangle, \quad \ldots]$$

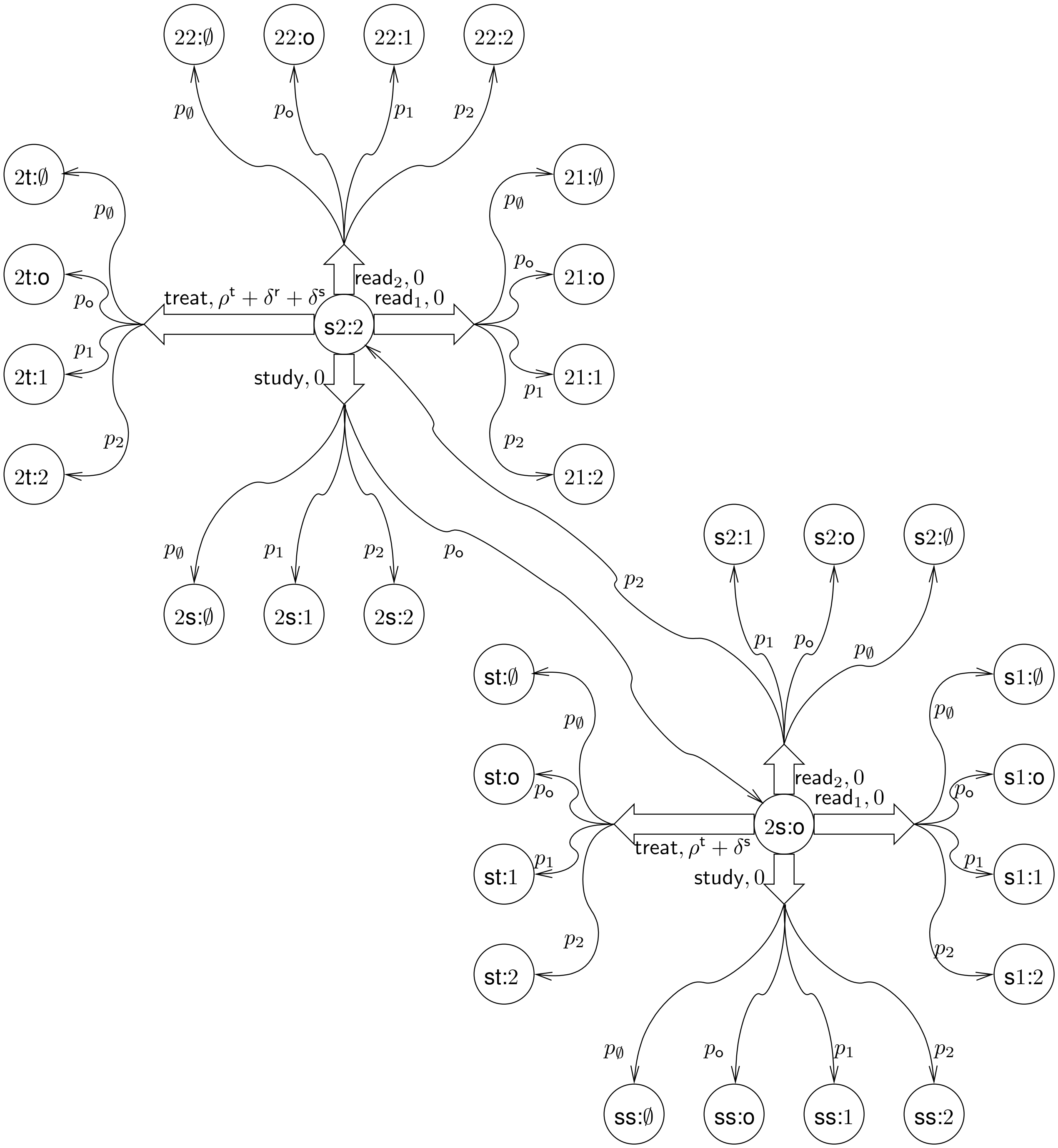

Figure 4.4: Part of the NMDP m^2_{ex4}. The figure only shows transitions and rewards originating at either state $\langle\langle \mathsf{s}, 2\rangle, \mathsf{none}\rangle$ or $\langle\langle 2, \mathsf{s}\rangle, 2\rangle$. It only shows states involved in one of these transitions. It abbreviates the state $\langle\langle a_1, a_2\rangle, c\rangle$ as $\bar{a}_1\bar{a}_2{:}c$ where $\bar{a}_1$ and $\bar{a}_2$ abbreviations for actions: read_1 becomes 1; read_2 becomes 2; and treat, t.

Reasoning about this family of models abstractly as we did for the one-step model is much more difficult since we now have the possibility of cycles. Thus, we will instead reason about concrete models created by fixing the values of the parameters. To do so, we will employ the AuditNMDPApprox algorithm of Section 2.3 to aid us and illustrate the usefulness of our formalism.

4.5.2 Methodology

We conducted experiments with our implementation to gain a feel for how the values of the parameters affects the allowed behavior. For simplicity, in our experiments, we treat all patients in the RHIO as identical and use the same rewards in the case of a patient not in the RHIO (i.e., $p_i = p_j$, $\rho^{\mathsf{t}}_i = \rho^{\mathsf{t}}_j = \rho^{\mathsf{t}}_{\mathsf{o}}$, $\delta^{\mathsf{r}}_i = \delta^{\mathsf{r}}_j$, and $\delta^{\mathsf{s}}_i = \delta^{\mathsf{s}}_j = \delta^{\mathsf{s}}_{\mathsf{o}}$ for all i and j in $\{1, \ldots, h\}$). Thus, we simply write p_{i} in the place of p_i for all i in $\{1, \ldots, h\}$. We also write δ^{r} in the place of δ^{r}_i, ρ^{t} for ρ^{t}_i or $\rho^{\mathsf{t}}_{\mathsf{o}}$, and δ^{s} for δ^{s}_i or $\delta^{\mathsf{s}}_{\mathsf{o}}$ for all i in $\{1, \ldots, h\}$.

Our experiments study how large the improvement δ^{r} must be compared to the improvement δ^{s} for the obeying the policy means that the physician must read a patient's record instead of studying. Given fixed values for all other parameters, we call this lowest value of δ^{r} for which the physician may read the record of a patient without violating the policy the *reading threshold*.

For the single step model, we can deduce the reading threshold from the rule that reading a record is acceptable if and only if $\max_i p_i * \delta^{\mathsf{r}}_i \geq \sum_{i=1}^{h} p_i * \delta^{\mathsf{s}}$. Under the assumption that all patients in the RHIO are identical and allowing possibility of a patient not in the RHIO, this inequality becomes $p_{\mathsf{i}} * \delta^{\mathsf{r}} \geq p_{\mathsf{o}} * \delta^{\mathsf{s}} + \sum_{i=1}^{h} p_{\mathsf{i}} * \delta^{\mathsf{s}}$, which provides the reading threshold of $\frac{h * p_{\mathsf{i}} + p_{\mathsf{o}}}{p_{\mathsf{i}}} \delta^{\mathsf{s}}$.

Analytically determining the reading threshold for m^{h}_{ex4} is possible using a manner similar to how we did so for the single-step model. However, this analysis is complicated by the presence of non-trivial cycles in the model and would involve solving a system of equations. Thus, we instead estimate the reading threshold in three manners. For the first estimation, we use the reading threshold of the single step model. In the context of our multi-step model, we call this value the *analytically estimated reading threshold* (AERT) since we find it by analysis of a simplified model.

For our second estimation, we use our implementation to estimate the reading threshold using simulations. We call this estimation the *simulatively estimated reading threshold* (SERT). Each simulation corresponds to setting the value of δ^{r} to some value v and testing with the AuditNMDPApprox implementation whether reading is allowed at the value v. In particular, we test whether studying (as opposed to reading a record) at the state $\langle\langle \mathtt{treat}, \ldots, \mathtt{treat}\rangle, \emptyset\rangle$ is a violation of the policy. If so, then v is an upper bound on the reading threshold; if not, then v is a lower bound. The algorithm establishes the initial lower bound as 1 and finds an initial upper bound by exponentially increasing the value of v until AuditNMDPApprox returns *true*. After establishing initial lower and upper bounds, the estimation algorithms iteratively uses their average for the next value of v tested by AuditNMDPApprox to find either a tighter lower or upper bound. The estimation algorithm continues until the bounds are within 1% of one another and uses their average as the SERT. If we were using AuditNMDP, then this procedure would guarantee that the resulting SERT is within 0.5% of the true reading threshold. However, since we use the approximate AuditNMDPApprox, the SERT may be further from the true reading threshold.

For our third estimation, we also use primarily simulation. Thus, we denote it as SERT′. However, it is a hybrid approach using the AERT as the initial value v tested with our AuditNMDPApprox implementation. Depending upon whether AuditNMDPApprox establishes the AERT to be an lower or upper bound, the algorithm will search for the missing bound using an exponential search. Then, the algorithm zeros in on an estimation using the same method as for the SERT. This hybrid approach is not necessarily quicker than the method for the SERT since the AERT could be very inaccurate. Nor is the SERT′ necessarily more accurate than the AERT since its attempts to improve upon the AERT could actually decrease its accuracy.

We implemented these estimation techniques in the Racket dialect of Scheme to use our implementation of the AUDITNMDPAPPROX algorithm. They may be downloaded from:

`http://www.cs.cmu.edu/~mtschant/thesis/`

We ran our implementations on a Lenovo U110 laptop computer with 3GB of memory and a 1.60 GHz Intel Core 2 Duo CPU running the DrRacket interpreter in Windows Vista.

4.5.3 Results

We compare the estimations AERT, SERT, and SERT′ across several models in the family m^h_{ex4}. Table 4.2 summarizes the results for each model we studied. The table also reports the running time required to compute the SERT and SERT′ for each model.

For all experiments, we use δ^{s} equal to 1. Most of the experiments used $h = 2$. For each of these experiments, we used three different values for the discounting factor γ: 0.9, 0.1, and 0.01. We ran four experiment with $h = 3$ and $\gamma = 0.01$. In two cases, we computed the SERT′ but not the SERT due to the long running time of these cases.

The table does not report upon the running time for a single call to AUDITNMDPAPPROX. For the $h = 2$ cases, the running time for a single call of AUDITNMDPAPPROX varied between 1.3 and 29 seconds. For the $h = 3$ cases, the running time varied between 202 seconds and 70 minutes. Each computation of SERT took between 10 and 12 calls to AUDITNMDPAPPROX. Each computation of SERT′ took 9 calls.

Examining Table 4.2, we see the following patterns. As expected from the fact they are computed in very similar manners the SERT and the SERT′ are always very close to one another. With the exception of four outliers in two different rows (in boldface), the AERT, SERT, and SERT′ are all very close. The AERT is consistently less than the other two estimations. This divergence increases as γ increases, which is intuitively because larger values of γ increase the importance of steps after the first step (the only step accounted for in the AERT).

The outliers use the very low value of 1 for ρ^{t}. We compared the optimal strategies for these outliers with similar models. Only in the case of the outliers does the physician read patient records rather than provide treatment even when a patient is present. Intuitively, the physician shows this behavior since the reward ρ^{t} for treating a patient without having either read the patient's record or studying is less than the expect increase in rewards for studying or reading a record. This effect disappears for lower values of γ since increases in future rewards become more heavily discounted.

As expected from known complexity results [Tse90], increasing γ increases the running time. The computation of SERT′ is consistently faster than the computation of the SERT.

4.5.4 Discussion

A base reward of $\rho^{\mathsf{t}} = 1$ with a studying bonus of $\delta^{\mathsf{s}} = 1$ seems unreasonable for most hospital settings since it implies that a physician may double his ability to treat the average patient in the amount of time it takes him to treat a patient. (These values might be reasonable for interns at a teaching a hospital.) Putting

s	p_{i}	p_{o}	$p_{\emptyset}$	ρ^{t}	γ	AERT	SERT	time	SERT$'$	time
2	0.01	0.95	0.03	1000	0.01	97	97.539	15 sec	97.379	13 sec
2	0.01	0.95	0.03	1000	0.1	97	97.539	20 sec	97.379	18 sec
2	0.01	0.95	0.03	1000	0.9	97	98.945	234 sec	98.895	211 sec
2	0.01	0.95	0.03	10000	0.01	97	97.539	16 sec	97.379	15 sec
2	0.01	0.95	0.03	10000	0.1	97	97.539	22 sec	97.379	19 sec
2	0.01	0.95	0.03	10000	0.9	97	98.945	272 sec	98.895	246 sec
2	0.01	0.95	0.03	100	0.01	97	97.539	16 sec	97.379	13 sec
2	0.01	0.95	0.03	100	0.1	97	97.539	19 sec	97.379	16 sec
2	0.01	0.95	0.03	100	0.9	97	98.945	196 sec	98.895	177 sec
2	0.01	0.95	0.03	10	0.01	97	97.539	15 sec	97.379	13 sec
2	0.01	0.95	0.03	10	0.1	97	97.539	16 sec	97.379	15 sec
2	0.01	0.95	0.03	10	0.9	97	98.945	162 sec	98.895	146 sec
2	0.01	0.95	0.03	1	0.01	97	97.539	13 sec	97.379	12 sec
2	0.01	0.95	0.03	1	0.1	97	97.539	16 sec	97.379	15 sec
2	0.01	0.95	0.03	1	0.9	97	**74.336**	128 sec	**74.076**	115 sec
2	0.0001	0.9698	0.03	1000	0.01	9700	9753.906	18 sec	9737.891	13 sec
2	0.0001	0.9698	0.03	1000	0.1	9700	9753.906	24 sec	9737.891	18 sec
2	0.0001	0.9698	0.03	1000	0.9	9700	9894.531	283 sec	9889.453	213 sec
2	0.0001	0.9698	0.03	10	0.01	9700	9753.906	18 sec	9737.891	13 sec
2	0.0001	0.9698	0.03	10	0.1	9700	9753.906	20 sec	9737.891	15 sec
2	0.0001	0.9698	0.03	10	0.9	9700	9894.531	194 sec	9889.453	145 sec
2	0.0001	0.9698	0.03	1	0.01	9700	9753.906	16 sec	9737.891	12 sec
2	0.0001	0.9698	0.03	1	0.1	9700	9753.906	20 sec	9737.891	15 sec
2	0.0001	0.9698	0.03	1	0.9	9700	**7363.281**	157 sec	**7369.727**	117 sec
2	0.0001	0.95	0.05	1000	0.01	9502	9542.969	18 sec	9539.117	13 sec
2	0.0001	0.95	0.05	1000	0.1	9502	9542.969	24 sec	9539.117	18 sec
2	0.0001	0.95	0.05	1000	0.9	9502	9683.594	283 sec	9687.586	211 sec
2	0.01	0.8	0.18	1000	0.01	82	82.07	15 sec	82.32	13 sec
2	0.01	0.8	0.18	1000	0.1	82	82.07	20 sec	82.32	18 sec
2	0.01	0.8	0.18	1000	0.9	82	84.18	234 sec	84.242	211 sec
3	0.01	0.94	0.03	1000	0.01	97	97.539	45 min	97.379	31 min
3	0.01	0.94	0.03	1000	0.1	97	97.539	63 min	97.379	54 min
3	0.01	0.94	0.03	1000	0.9	97	98.242	12 hours	98.137	461 min

Table 4.2: Results of experiments on m^{h}_{ex4}. In all cases $\delta^{\mathsf{s}} = 1$. The values for the estimations are rounded to three decimal places. Four outliers, two each in two rows, are in boldface.

aside the outliers for this reason, we find that the AERT, SERT, and SERT$'$ are close. We conclude that the AERT is a good approximation of the reading threshold at least for small values of h.

The complexity of the above calculations highlights how our model of planning does not correspond to how humans plan (further discussed in Chapter 6). We cannot expect physicians to perform complex modeling and analysis let alone to use computer simulations before deciding whether to read a record or study. However, compliance officers at hospitals may find these results helpful while drafting policy manuals.

For example, consider a large hospital where the probability of a physician seeing a typical patient in the RHIO is less than 1 in $10,000$. At such a hospital, the reading threshold of about 9700 holds across various of values for ρ^{t} and γ. Extrapolating from the results for the tests using $h = 3$ with $p_{\mathsf{i}} = 0.01$ instead of 0.0001, one may conclude that the value is likely to remain around 9700 for larger values of h. In many settings, managers may find an improvement from reading a patient's record of 9700 times the improvement from studying inconceivable. In this case, a policy manual may quantitatively summarize the quantitative results shown in Table 4.2 as prohibiting a physician from reading patient records unless the physician has a reason to believe that the patient is much more likely than average to be seeking care. Such a prohibition may not make sense at a small practice where the probability of seeing an average patient is 1 in 100 since reading a record could conceivably produce an improvement of 97 times the improvement from studying.

4.6 Learning Additional Information

Now we consider the effects of the physician learning additional information, which requires POMDPs to model. The above example assumes that Dr. Z has a fixed probability distribution over next patients. However, Dr. Z might learn information that leads him to consider some patients more likely than others. In the extreme case, a patient might present himself at Dr. Z's practice leading him to assign the probability of 1 to that patient. In this case, the model would change and reading that patient's record would become the optimal plan.

In a more complex example, Dr. Z might learn that someone with the named "John Smith" has been in an accident and will require treatment soon. As "John Smith" is a common name, Dr. Z might have more than one record bearing the name. If the number is small enough (less than a 100 in the above model), then the best use of Dr. Z's time will be reading as many of these records as possible. Even in the case where Dr. Z has time to read only a single record making it unlikely he will read the record of the John Smith coming for treat, Dr. Z's reading will be for the purpose of treatment.

In this case, Dr. Z should select one of the records bearing the name "John Smith" that has the highest expected improvement in treatment. In the case where more than one record has the same expected improvement, Dr. Z might be tempted choose amongst these records with another illicit purpose in mind. To avoid Dr. Z satisfying this other illicit purpose, his selection should be uniformly random amongst all these records. A medical record system could perform the selection for him to insure randomness.

In more complex situations, in coming information might also adjust the probabilities without ruling any one patient out. For example, learning that a patient has a chronic condition and is likely to seek treatment could make that patient more likely relative to the others without ruling out any of the other patients.

Each of these the above scenarios can be modeled using POMDPs. Using our formalism for information use, we can examine whether the incoming information is used for the purpose of treatment. In particular, if the incoming information causes the physician to read a patient medical record rather than study, our formalize will show that the information was used for the purpose of treatment. In the case where the information resulted in no change in the physician's behavior, our formalism will show neither that it was used for treatment nor that it was not used for treatment as the physician may have decided to study either with or without the additional information.

4.7 Revisiting Uploading

The above arguments about a physician having to make good use of his time does not invalidate the reasoning used to justify Dr. X posting Y's record to the RHIO. In particular, the probability that some doctor in the region would need Y's record at some point in the future is fairly high. However, the probability p_{i^*} from above is comparatively small as it is limited to just Dr. Z and to only the period of time over which Dr. Z can remember the information. Furthermore, presuming suitable automation Dr. X posting the record will take so little time as that little else productive could be done in that time. Dr. Z reading a record, on the other hand, takes much longer.

Now, the reader may be worried that our formalism allows Dr. X to post the record anywhere because with some odds it might be retrieved from that location to improve treatment. While our formalism allows such posting, the above example involving Dr. Z shows that our formalism does not allow people with access to these postings to read them unless there is a good reason. If we trust those with access not to abuse their access (which is implicit in HIPAA), then distributing the record does not have negative privacy implications. However, as we are not that trusting, one must weigh the risks of abuse with the possible benefits of access. While a formalism based on MDPs and planning may be helpful for such balancing, it is outside of the scope of this work formalizing purpose.

Chapter 5

Empirical Study of Semantics

5.1 Goals

Both previous work and this work offer methods for enforcing privacy policies that feature purpose restrictions. These methods test whether a sequence of actions violates a clause of a privacy policy that restricts certain actions to be only for certain purposes. By providing a test for whether the purpose restriction is violated, these methods implicitly provide a semantics for these restrictions.

To ensure that these methods correctly enforce the privacy policy, one must show that the semantics employed by a method matches the intended meaning of the policy. Unfortunately, determining the intended meaning of a policy from its text is often impossible. Furthermore, these policies often act as agreements among multiple parties who may differ in their interpretation of the policy. For these reasons, we compare the semantics proposed by these methods of policy enforcement to the most common interpretations of policies.

While previous works have not provided a formal semantics, it appears that many works (e.g., [AF07, JSNS09]) flag actions as a violation if they do not further the purpose in question. (See Section 7.1 for a description of past works.) In particular, these works make assumptions about how people think about *purpose* in the context of enforcing a privacy policy that restricts an agent to only performing a certain class of actions for a certain purpose. The following hypothesis characterizes these assumptions:

H1. The agent obeys the restriction if and only if the action furthered the purpose.

This hypothesis entails the following hypothesis about how people interpret the meaning of *purpose*:

H1'. An action is for a purpose if and only if that action furthers that purpose.

Our work instead asserts that an action may be for a purpose even if that purpose is never furthered. In particular, we assert that the action merely has to be part of a plan for furthering that purpose. Thus, our formalism assumes the following hypothesis (in the same context as above):

H2. The auditee obeys the restriction if and only if the auditee performed that action as part of a plan for furthering that purpose.

(We do not construct our algorithms directly from Hypothesis H2. Rather they are approximations using only observable information.) Similarly, this hypothesis entails the following:

H2’. An action is for a purpose if and only if the auditee performed that action as part of a plan for furthering that purpose.

To show that our work provides a method of enforcing purpose restrictions more faithful to their common meaning, we would like to disprove Hypotheses H1 and H1’ while proving Hypotheses H2 and H2’.

As Hypothesis H1 is a bi-implication, we can disprove it by disproving either of the following hypotheses (here and henceforth, in the same context as above):

H1a. If the action furthers a purpose, then the auditee obeys the restriction.

H1b. If the auditee obeys the restriction, then the action furthers a purpose.

We will attempt to disprove both Hypotheses H1a and H1b.

Similarly, Hypothesis H2 breaks into two sub-hypotheses:

H2a. If the auditee performed an action as part of a plan for furthering a purpose, then the auditee obeyed the restriction.

H2b. If the auditee obeyed the restriction, then the auditee performed the action as part of a plan for furthering that purpose.

We will test both of these hypotheses by providing example scenarios of an auditee performing actions with descriptions of his plans. However, these tests will not prove either of these hypotheses as doing so would require testing them under all scenarios. Indeed, given that some tests could be carefully crafted to bring about success for reasons unrelated to planning, such testing does not necessarily provide good evidence in favor of these hypotheses. To provide better evidence for the truth of Hypothesis H2, we will also test the following related hypothesis:

H2c. Describing an action as being part of a plan for furthering purpose as opposed to not being part of such a plan in a scenario causes people to think that the auditee obeyed the restriction.

H2c may be viewed a causal or directional version of H2. Unlike H2a and H2b, which may be tested with unrelated scenarios, H2c must be tested with scenarios that only differ from one another in whether the action is part of a plan for the purpose in question.

For completeness we also test the causal version of H1:

H1c. Describing an action as furthering a purpose as opposed to not furthering a purpose in a scenario causes people to think that the auditee obeyed the restriction.

As Hypothesis H1 leads to Hypotheses H1a, H1b, and H1c, Hypothesis H1’ leads to corresponding hypotheses H1a’, H1b’, and H1c’. Similarly, H2’ leads to H2a’, H2b’, H2c’. We also test these hypotheses to provide additional evidence for our formalism.

5.2 Methodology

Approach. We may disprove Hypothesis H1a by exhibiting a scenario in which an action of an auditee furthers a purpose, but people feel that the auditee did not obey a purpose restriction stating that the action

	Furthered purpose	Did not further purpose
Planned for purpose	C_{pf}	$C_{\mathsf{p}\bar{\mathsf{f}}}$
Not planned for purpose	$C_{\bar{\mathsf{p}}\mathsf{f}}$	$C_{\bar{\mathsf{p}}\bar{\mathsf{f}}}$

Table 5.1: Classes of Scenarios for Survey Questionnaire. Each position in the grid identifies the scenario class associated with the values of the two factors given on each axis.

may only be performed for that purpose. We may disprove Hypothesis H1b by exhibiting an scenario in which an action does not further a purpose, but people feel that the auditee obeyed the restriction. To test Hypothesis H1c, we construct a pair of scenarios that differs only in whether the action furthered the purpose in question, and show that people's feelings about whether the auditee obeyed the restriction is unchanged across the two scenarios.

Testing Hypotheses H2a, H2b, and H2c is similar to testing the corresponding hypothesis for H1. However, we expect the opposite results. For example, to test Hypothesis H2c, we construct a pair of scenarios that differs only in whether the auditee performed that action as part of a plan for furthering that purpose. We expect to show that people feel that the auditee obeyed the restriction only in the scenario in which the action is part of a plan for furthering that purpose.

To these ends, we use four classes of scenarios: Classes C_{pf}, $C_{\mathsf{p}\bar{\mathsf{f}}}$, $C_{\bar{\mathsf{p}}\mathsf{f}}$, and $C_{\bar{\mathsf{p}}\bar{\mathsf{f}}}$. Each class is determined by two factors: (1) whether the action furthers the purpose in question in the scenario and (2) whether the auditee performs the action as part of a plan for furthering the purpose. Table 5.1 identifies these classes along these two axes. (E.g., $C_{\bar{\mathsf{p}}\mathsf{f}}$ stands for the scenario that was *not* **p**lanned ($\bar{\mathsf{p}}$) for the purpose but **f**urthered (f) it.)

Showing that people think the auditee does not obey the restriction in Scenario Class $C_{\bar{\mathsf{p}}\mathsf{f}}$ is sufficient for disproving Hypothesis H1 by disproving Hypothesis H1a. Showing that people think the auditee obeys the restriction in Class $C_{\mathsf{p}\bar{\mathsf{f}}}$ provides additional evidence that previous approaches are insufficient by disproving the other direction, H1b, of the bi-implicational Hypothesis H1. Comparing Class C_{pf} against $C_{\mathsf{p}\bar{\mathsf{f}}}$ tests Hypothesis H1c. Comparing Class $C_{\bar{\mathsf{p}}\mathsf{f}}$ against $C_{\bar{\mathsf{p}}\bar{\mathsf{f}}}$ also tests Hypothesis H1c.

For Hypothesis H2, showing that people think the auditee obeyed the restriction in Classes C_{pf} and $C_{\mathsf{p}\bar{\mathsf{f}}}$ each provides evidence for Hypothesis H2a. Showing that people think the auditee does not obey the restriction in Classes $C_{\bar{\mathsf{p}}\mathsf{f}}$ and $C_{\bar{\mathsf{p}}\bar{\mathsf{f}}}$ each provides evidence for Hypothesis H2b by way of the contrapositive. Comparing Class C_{pf} against $C_{\bar{\mathsf{p}}\mathsf{f}}$ and comparing Class $C_{\mathsf{p}\bar{\mathsf{f}}}$ against $C_{\bar{\mathsf{p}}\bar{\mathsf{f}}}$ test Hypothesis H2c.

Questionnaire Construction. We constructed a questionnaire with four scenarios, one from each of the four scenario classes above. The auditee in these four scenarios is subject to a privacy policy that states that the auditee may only use a type of information for a single purpose. The policy we used for the questionnaire is as follows:

> Metropolis General Hospital and its employees will share a patient's medical record with an outside specialist only for the purpose of providing that patient with treatment.

Table 5.2 presents the scenarios where Scenario S_{xy} is the scenario in Scenario Class C_{xy}.

For each scenario, we ask the participant five questions. The first two are simple questions, Questions Q1 and Q2, about each scenario. These questions have objectively correct answers that the participant can easily

$\mathbf{S}_{\mathsf{pf}}$. A case worker employed by Metropolis General Hospital meets with a patient. The case worker develops a plan with the sole goal of treating the patient. The plan includes sharing the patient's medical record with an outside specialist. Upon receiving the record, the specialist succeeds in treating the patient.

$\mathbf{S}_{\mathsf{p\bar{f}}}$. A case worker employed by Metropolis General Hospital meets with a patient. The case worker develops a plan with the sole goal of treating the patient. The plan includes sharing the patient's medical record with an outside specialist. Upon receiving the record, the specialist did *not* succeed in treating the patient.

$\mathbf{S}_{\mathsf{\bar{p}f}}$. A case worker employed by Metropolis General Hospital meets with a patient. The case worker develops a plan with the sole goal of reducing costs for the hospital. The plan includes sharing the patient's medical record with an outside specialist. Upon receiving the record, the specialist succeeds in treating the patient.

$\mathbf{S}_{\mathsf{\bar{p}\bar{f}}}$. A case worker employed by Metropolis General Hospital meets with a patient. The case worker develops a plan with the sole goal of reducing costs for the hospital. The plan includes sharing the patient's medical record with an outside specialist. Upon receiving the record, the specialist did *not* succeed in treating the patient.

Table 5.2: Questionnaire Scenarios. For each scenario class, the scenario used on the questionnaire.

find by reading the scenarios. Checking that the participant chooses the correct answer allowed us to ensure that the participants actually read the scenario and answered accordingly rather than arbitrarily.

After priming the participant to have read the scenario with Questions Q1 and Q2, we ask the key questions for our study. While we are most interested in whether the participant believes that the auditee obeyed the policy, we start with the more basic Question Q3: whether the action was for the allowed purpose of treatment. We then ask Question Q4: whether the auditee obeyed the policy. For each of these four questions, the participant may select among *yes*, *no*, and *I don't know*. We expected Questions Q3 and Q4 to be answered identically for each scenario. We included both to help determine whether the questionnaire was well worded and to test the Hypotheses H1' and H2'. We conjecture that the majority of participants will answer this Questions Q3 and Q4 with *yes* for the scenarios in classes C_{pf} and $\mathrm{C}_{\mathsf{p\bar{f}}}$, and with *no* for $\mathrm{C}_{\mathsf{\bar{p}f}}$ and $\mathrm{C}_{\mathsf{\bar{p}\bar{f}}}$.

To help determine the reasoning used by the participants, which would be especially useful if our survey results deviated from the expected, we included the free form Question Q5 asking why the participant chose the answer he did to Question Q4. Table 5.3 shows the wording of these questions.

We presented these scenarios and questions as a questionnaire. For each survey participant, we randomly ordered the scenarios to reduce the effects that scenario ordering may have. For each scenario, we ordered the questions with the objective questions, Q1 and Q2, first to encourage the participant to read the scenario before answering the subjective questions in which we are interested. The subjective questions are ordered as follows: Q3, Q4, and, lastly, Q5. Appendix B.1 shows a sample questionnaire.

Q1. Was the goal of the case worker's plan to treat the patient?

Q2. Did the specialist succeed in treating the patient?

Q3. Did the case worker share the record with the specialist for the purpose of treatment?

Q4. Did the case worker obey the above privacy policy?

Q5. Why did you answer Question 4 as you did?

Table 5.3: Questionnaire Questions

Pilot Study. Before running the main survey we conducted a small scale pilot study of ten participants. The participants were recruited on Amazon Mechanical Turk (`www.mturk.com`) using a small payment of $1.50 (USD). (Appendix B.2 shows the advertisement.) Participants took the survey online using Mechanical Turk's survey functionality without randomly ordering the scenarios.

The goal of this pilot study was to ensure that our recruitment and survey mechanisms worked. We also closely examined the responses to determine whether the participants were seriously answering the questions, and whether Questions Q1 and Q2 identified arbitrary responses. As the goal of this study was not to collect data on our hypotheses, we did not statistically analyze the data. However, we will qualitatively describe the results below.

In the pilot study, seven of the ten respondents matched our predictions perfectly. One respondent deviated for a single answer in a manner inconsistent with the other answers provided by the respondent. Thus, we suspect that his response is most likely an error in selecting the answer.

A second respondent said that the action was not for the purpose of treatment in Scenarios $S_{\bar{p}f}$ and $S_{\bar{p}\bar{f}}$, but that, nevertheless, the case worker obeyed the policy since the specialist would try to provide treatment. This response suggests that Hypotheses H2 and H2' are more than trivially different.

The third respondent to deviate from our hypothesis claimed that the action was for the purpose of treatment and the case worker obeyed the policy in all of the scenarios including Scenarios $S_{\bar{p}f}$ and $S_{\bar{p}\bar{f}}$ where goal of the case worker was cost reduction. This respondent's answer to Question Q5 suggests that the case worker did not violate the policy as the scenarios provide evidence that the specialist provided treatment whereas they provide no evidence that any of the actions reduced costs. For example, this respondent provided the following for Question Q5 given Scenario $S_{\bar{p}f}$:

> Though the case worker's goal was cost-reduction, the medical records were still provided for the purpose of treating the patient; simply giving medical records to outside specialists, with no further actions, would not be a way to reduce costs for a hospital.

This response highlights that our scenarios discuss treatment in more detail than cost reduction, which could have unintended effects on people's analysis of them.

Interestingly, while these two deviations do not match our Hypothesis H2, they are consistent with the approximations our algorithm makes. While these deviations suggest interesting directions for future studies, we decided that these issues did not warrant rewriting the scenarios to include more information on cost reduction or to examine more carefully the differences between Hypotheses H2 and H2'.

None of the respondents said that the policy was violated in Scenario $\mathsf{S}_{\mathsf{p}\bar{\mathsf{f}}}$, providing evidence against Hypothesis H1. None of the respondents answered Questions Q1 or Q2 incorrectly and none of their responses appeared arbitrary.

Survey Protocol. The main survey consisted of two hundred participants. We conducted the survey in the same manner as the pilot study but with three changes. First, given the ease with which we recruited participants for the pilot study, we reduced the payment to \$0.50.

Second, while still using Mechanical Turk to recruit and pay participants, we used Survey Gizmo (`www.surveygizmo.com`) to conduct the survey. This change allowed us to randomly order the scenarios for each participant.

Third, given the success of Questions Q1 and Q2, we decided before the survey to exclude from the results any participants who got more than one of them wrong in total across all four scenarios. The odds of correctly guessing either all the answers or all but one is less than 4% presuming the participant knows that *I don't know* is never a correct answer.[1]

We analyzed the survey responses according to the statistical model presented in the next section.

5.3 Statistical Modeling

In this section, we provide a detailed description of the statistical tests we employ in the next section. Those with a background in hypothesis testing and statistics may find the following summary sufficient.

Summary. Each of the hypotheses H1a, H1b, H2a, and H2b makes predictions about whether Question Q4 will be answered with *yes* or *no*. We model these answers as a draw from a binomial distribution and we interpret these predictions as predictions about probability of success for the binomial distribution. For Hypotheses H1a and H1b, we treat their predictions as the null hypotheses about the probability of success and attempt to reject them to disprove H1. We treat the predictions of H2a and H2b as the alternative hypotheses and attempt to reject their negations as null hypotheses to provide evidence in favor of H2. Table 5.5 presents how to convert these predictions in testable hypotheses. In short, we interpret a prediction that a question will be answered with a certain response as an assertion that the probability of success (seeing that response) is at least 0.5.

To test Hypothesis H1c, we use McNemar's Test to test whether an action furthering a purpose has a statistically significant effect on how people answer Question Q4. We test Hypothesis H2c using McNemar's Test across scenarios that only differ in the goal of the auditee's plan.

We test Hypotheses H1' and H2' analogously using Question Q3 in the place of Q4. For all statistical tests, we use $\alpha = 0.05$ for the threshold of statistical significance.[2]

[1] The odds of guessing correctly one of the questions is $\frac{1}{2}$ since there are two possible answers (ruling out *I don't know*). Each of the four scenarios have two questions meaning that seven or eight would have to be correctly guessed for a guessing participant to avoid rejection. We model these guesses using the binomial distribution, which has the cumulative distribution function $F(x; n, p) = \Pr[X \leq x] = \sum_{i=0}^{x} \binom{n}{i} p^i (1-p)^{n-i}$ where x is the number of successes, n the number of trails, and p is the probablity of success. In particular, we find that odds of getting 7 or more success is $1 - F(6; 8, \frac{1}{2})$ where $F(6; 8, \frac{1}{2})$ is the odds of getting 6 or fewer successes. This is $1 - F(6; 8, \frac{1}{2}) = 1 - \sum_{i=0}^{6} \binom{8}{i} \frac{1}{2}^i (1 - \frac{1}{2})^{8-i} < 1 - 0.96 = 0.04$. Without ruling out the option of *I don't know*, the odds of successfully avoiding detection would be $1 - F(6; 8, \frac{1}{3}) < 0.01$.

[2] The statistical tests used in this chapter are all from the "orthodox" frequentist interpretation of statistics. Bayesian statistics

5.3.1 Hypothesis Testing

An underlying presumption of this work is that *purpose* has an objective definition on which people generally agree. However, even under this presumption, we cannot expect that, for each question, every participant will respond with the same answer. Some participants might misread the question or hold non-standard views. Thus, we model each response to Question Q4 as a trial of a distribution over the three possible responses: *yes*, *no*, and *I don't know*.

The hypotheses H1a, H1b, H2a, and H2b each make predictions about how people will answer Question Q4 in various scenarios. For example, Hypothesis H1a predicts that people will answer Question Q4 with *yes* rather than *no* when given a scenario of Class $\mathrm{C}_{\bar{\mathsf{p}}\mathsf{f}}$. Literally interpreted, Hypothesis H1a predicts that the probability of answering *yes* under Scenario $\mathrm{S}_{\bar{\mathsf{p}}\mathsf{f}}$, which we denote as $p_{\bar{\mathsf{p}}\mathsf{fy}}$, will be 1. However, as discussed above, we would expect to see the probability $p_{\bar{\mathsf{p}}\mathsf{fy}}$ being somewhat less than 1 even if Hypothesis H1a is true. The lower the probability, the more questionable the truth of the hypothesis becomes. The lower limit at which we reject the hypothesis as false depends upon how one formalizes the hypothesis. We choose to set this limit at the probability 0.5 since a hypothesis that does not correctly predict the majority of outcomes appears clearly false to us. Thus, we formalize this prediction as:

H1a$_{\mathsf{0y}}$**.** $p_{\bar{\mathsf{p}}\mathsf{fy}} \geq 0.5$

As we hope to disprove Hypothesis H1a, we would like to cast doubt on the Hypotheses H1a$_{\mathsf{0y}}$, which makes it a *null hypothesis* we hope to reject. Rejecting the null hypothesis provides evidence in favor of the *alternative hypothesis* we hope to show:

H1a$_{\mathsf{ay}}$**.** $p_{\bar{\mathsf{p}}\mathsf{fy}} < 0.5$

Since H1a$_{\mathsf{0y}}$ predicts a large number of *yes* responses and H1a$_{\mathsf{ay}}$ predicts a small number, the smaller the number of *yes* responses observed among the survey responses, the more likely H1a$_{\mathsf{ay}}$ seems relative to H1a$_{\mathsf{0y}}$. That is, seeing a small number y or fewer *yes* responses is more unlikely under the assumption of H1a$_{\mathsf{0y}}$ than under the assumption H1a$_{\mathsf{ay}}$. As this small number y decreases the probability of seeing y or fewer *yes* responses under the assumption of H1a$_{\mathsf{0y}}$ decreases. This probability is called the *p-value*. It is convenient to represent the p-value as $\Pr[Y \leq y \mid \mathrm{H1a}_{\mathsf{0y}}]$ where Y is a random variable over the number of observed *yes* responses and y is the actual number of observed *yes* responses. However, the hypothesis H1a$_{\mathsf{0y}}$ is a composite hypothesis asserting that $p_{\bar{\mathsf{p}}\mathsf{fy}} = p$ for some $p \geq 0.5$. Since we would like to disprove the null hypothesis for all these possible values of p, we use the upper bound as the p-value: $\max_{p:0.5 \leq p \leq 1} \Pr[Y \leq y \mid p_{\bar{\mathsf{p}}\mathsf{fy}} = p]$.

If the number y of observed *yes* responses is small enough, then the p-value may become so small that we may confidently reject the null hypothesis H1a$_{\mathsf{0y}}$ in favor of the alternative H1a$_{\mathsf{ay}}$. Since we are looking for a low value for the number of *yes* responses to reject the null hypothesis, we are using a *lower-tail rejection region*.

We must decide how unlikely the observation must be before we are willing to reject the null hypothesis. This choice must balance the risk of incorrectly rejecting a null hypothesis that is actually true (called *Type I*

offers many advantages over frequentist statistics (see, e.g., [Jay03]). However, the analysis in this chapter is the current standard for the area. Furthermore, given the overwhelming evidence in favor Hypothesis H2 and against H1, the flaws of frequentist methods should not affect any of the outcomes. Lastly, the author desires to perform the statistical analysis he selected before seeing the data to avoid the impression of changing analyses to reach a desired outcome. For these reasons, the author will leave a Bayesian analysis of the data as future work despite believing that one would be more mathematically justified and accurate.

Error) with the risk of incorrectly accepting a null hypothesis that is false (called *Type II Error*). Following convention, we choose the level of Type I Error to be $\alpha = 0.05$. That is, we reject H1a$_{\mathsf{0y}}$ in favor of H1a$_{\mathsf{ay}}$ if the p-value (the probability of seeing observed number of *yes* responses or fewer under the assumption that H1a$_{\mathsf{0y}}$ is true) is less than 0.05.

Hypothesis H1a also produces another prediction: that the number of *no* responses to Question Q4 will be low for Scenario S$_{\bar{\mathsf{p}}\mathsf{f}}$. We can also formalize this prediction as a null hypothesis that we hope to reject:

H1a$_{\mathsf{0n}}$. $p_{\bar{\mathsf{p}}\mathsf{fn}} \leq 0.5$

The alternative hypothesis we hope to accept in favor of the null hypothesis is

H1a$_{\mathsf{an}}$. $p_{\bar{\mathsf{p}}\mathsf{fn}} > 0.5$

In this case, we become more willing to reject the null hypothesis H1a$_{\mathsf{0n}}$ as the number of *no* responses *increases*. This creates an *upper-tail rejection region* in which we are interested in the probability of seeing the observed number of *no* responses or more under the assumption that the null hypothesis is true. As before this quantity is called the p-value. We will again reject if the p-value is less than $\alpha = 0.05$.

We can also formalize the predictions made by Hypothesis H2a. However, as we hope to provide evidence in favor of Hypothesis H2 instead of disproving it, we treat its predictions as alternative hypotheses rather than null hypotheses. For the null hypotheses we use the negations of its predictions and attempt to disprove them. For example, Hypothesis H2a predicts that the number of *yes* responses to Question Q4 for Scenario S$_{\mathsf{p}\bar{\mathsf{f}}}$ will be high. Thus, we attempt to provide evidence for the following alternative hypothesis:

H2a$_{\mathsf{ay}}$. $p_{\mathsf{p}\bar{\mathsf{f}}\mathsf{y}} > 0.5$

We do so by showing the probability of seeing the observed number of *yes* responses or more (the p-value using an upper tail rejection region) is unlikely (less than $\alpha = 0.05$) under the assumption that the following null hypothesis is true:

H2a$_{\mathsf{0y}}$. $p_{\mathsf{p}\bar{\mathsf{f}}\mathsf{y}} \leq 0.5$

We may similarly, formalize other predictions of Hypotheses H1a and H2a as well as the predictions of Hypotheses H1b and H2b. We show each of these formalizations in Table 5.5 in the next section while presenting the survey results.

These formalizations are, however, only useful if we can compute the value of the p-value under each of them. That is, we must have a formal model of the survey responses that allows us to compute the probability of seeing the responses we observe under the null hypothesis. We now turn to describing such a model.

5.3.2 Binomial Model of the Survey

Each null hypotheses that we test is an assertion about the probability of observing either a *yes* or a *no* response. In the case that the null hypothesis is an assertion about the probability of observing *yes*, we consider the response of *yes* to be a *success* outcome representing successfully observing the response about which the assertion is. We may collapse the responses of *no* and *I don't know* into a single *failure* outcome that represents failure to see *yes*. Likewise, in the case where the null hypothesis is an assertion about the probability of observing *no*, we may treat *no* as a success outcome while treating *yes* and *I don't know*, jointly, as a failure outcome.

By using only two outcomes (success and failure), we may model each survey response as a Bernoulli trial, which models the flipping of a possibly biased coin. The degree of bias determines the probability of success, which models the probability of a respondent answering the question in the manner we are testing.

We model all the responses to a single question of our survey collectively as a series of identical independent Bernoulli trials with each respondent corresponding to one trial. For a given number of trials and probability of success for each trial, the binomial distribution provides the probability of seeing each possible number of successes. (As we do not allow the same individual to take the survey more than once, the assumption of identical independent trials is not completely satisfied since later responses are from a smaller pool of possible respondents that does not include the previous respondents. This factor results in the hypergeometric distribution being a more accurate model. However, since we are drawing our participants from a pool much larger than the sample size, the binomial distribution provides a good approximation.) In particular, the binomial distribution has the cumulative distribution function $F(x; n, p) = \Pr[X \leq x] = \sum_{i=0}^{x} \binom{n}{i} p^i (1-p)^{n-i}$ where x is the number of successes, n the number of trails, and p is the probability of success.

Our null hypotheses are assumptions about the value of the success probability p (not to be confused with the idea of a p-value). Using the binomial distribution, we may determine the probability of seeing the responses observed under the null hypothesis. However, we are actually interested in the p-value: the probability of seeing a set of responses at least as extreme as the observed one where the meaning of *extreme* depends upon whether we are using a lower-tail or an upper-tail rejection region.

For example, consider the null hypothesis H1a$_{\mathsf{0y}}$ that $p_{\bar{\mathsf{p}}\mathsf{fy}} \geq 0.5$. We will reject H1a$_{\mathsf{0y}}$ using a lower-tail rejection region if its p-value is less than $\alpha = 0.05$ where the p-value is the probability of seeing the observed number of *yes* responses (success outcomes) or fewer. Under our binomial model, the p-value for H1a$_{\mathsf{0y}}$ is

$$\max_{p:0.5 \leq p \leq 1} \Pr[Y \leq y \mid p_{\bar{\mathsf{p}}\mathsf{fy}} = p] = \max_{p:0.5 \leq p \leq 1} \Pr[Y \leq y \mid Y \sim \mathsf{B}(n, p)] = \max_{p:0.5 \leq p \leq 1} F(x; n, p)$$

where $Y \sim \mathsf{B}(n, p)$ asserts that Y is a random variable obeying the binomial distribution with a sample size of n and success probability of p.

We may use $F(x; n, 0.5)$ in the place of $\max_{p:0.5 \leq p \leq 1} F(x; n, p)$ since we will reject the null hypothesis under the first value if and only if we reject it under the second value. The reason for this equivalence is that $F(x; n, p)$ is an decreasing function in p and is always maximized at $p = 0.5$ when $0.5 \leq p$.

For Hypothesis H2a, we are interested in the null hypothesis that $p_{\mathsf{p}\bar{\mathsf{f}}\mathsf{y}} \leq 0.5$ using an upper-tail rejection region. For this null hypothesis, the p-value equals

$$\max_{p:0 \leq p \leq 0.5} 1 - F(x; n, p) = 1 - \min_{p:0 \leq p \leq 0.5} F(x; n, p)$$

Similar to the case with the lower-tail rejection region, $\min_{p:0 \leq p \leq 0.5} F(x; n, p)$ is equal to $F(x; n, 0.5)$ since $F(x; n, p)$ is minimized at the largest available value of p, that is, 0.5.

The ability to use $F(x; n, 0.5)$ in computing the p-value for both lower-tail and upper-tail rejection regions justifies the convention of writing the null hypotheses using an equality rather than an inequality relation. Whether the equality is short hand for a greater-than-equal or a less-than-equal relation may be inferred from the alternative hypothesis paired with the null hypothesis. We will adopt this convention for the remainder of this work.

5.3.3 McNemar's Test

To test hypotheses H1c and H2c, we must compare the responses across scenarios. These responses are not independent since the same respondent produces responses for both scenarios. That is, the responses are produced as *matched-pairs*. McNemar's test provides a method of determining from these matched-pairs the effects of switching between the two scenarios [McN47]. In particular, McNemar's test examines the number of pairs where the response switches either from *yes* to *no* or from *no* to *yes*. The test approximates the probability of the number of switches being produced by two dependent draws from one distribution. If this probability is small, then one may reject the null hypothesis that switching between the two scenarios had no effect. By rejecting this null hypothesis, one provides evidence for the alternative hypothesis that the difference between the two scenarios affected the responses.

For example, for hypothesis H2c, we compare the responses to Question Q4 across the Scenarios S_{pf} and $\mathrm{S}_{\bar{\mathsf{p}}\mathsf{f}}$. We use the null hypothesis that whether the case worker employed a plan for treating the patient has no effect on whether survey participants think the case worker violated the policy. If we find that a large number of respondents have different responses across the two scenarios, then we would reject the null hypothesis and conclude that case worker's planning does have an effect.

We test Hypothesis H1a' in a manner similar to how we test Hypothesis H1a. However, we use Question Q3 instead of Question Q4. Analogously, we test Hypotheses H1b', H1c', H2a', H2b', and H2c' in a manner similar to Hypotheses H1b, H1c, H2a, H2b, and H2c, respectively, using Question Q3 in place of Question Q4.

5.4 Results

While we only offered to pay the first 200 respondents, we received 207 completed surveys. The extra surveys may have resulted from people misunderstanding the instructions and not collecting payment.

Of these completed surveys, we excluded 20 respondents for missing two or more of the objective questions. All of the statistics shown in this section are calculated from the remaining 187 respondents. Appendix B.4 shows the same statistics for all 207 respondents. Including the 20 excluded respondents does not change the significance of any of our hypothesis tests.

Table 5.4 shows the distributions of responses for each question. Informally examining the tables shows that the vast majority of the respondents conform to Hypothesis H2. For example, 177 (95%) of the respondents answered Question Q4 for Scenario $\mathrm{S}_{\mathsf{p}\bar{\mathsf{f}}}$ with the answer of *yes* as predicted by Hypothesis H2, whereas only eight (4%) answered with *no* as predicted by Hypothesis H1. However, the difference is less pronounced for Scenario $\mathrm{S}_{\bar{\mathsf{p}}\mathsf{f}}$ where 133 (71%) match Hypothesis H2's prediction of *no* and 45 (24%) matches H1's prediction of *yes*. Interestingly, 31 (17%) answered *yes* for Scenario $\mathrm{S}_{\bar{\mathsf{p}}\bar{\mathsf{f}}}$ despite both hypotheses predicting *no*.

Table 5.5 shows the hypothesis tests we conducted using the binomial model. The top half of the table shows tests intended to disprove Hypothesis H1 while the bottom half shows tests attempting to confirm Hypothesis H2. Every test in favor of Hypothesis H2 obtains statistical significance. Eight of the 16 tests against Hypothesis H1 obtain statistical significance. The eight that do not obtain significance are the cases where the two hypotheses agree. In every case where the two disagree, both the test confirming Hypothesis H2 and the one against Hypothesis H1 obtains significance.

Scenario	Yes	I don't know	No
S_{pf}	186 (99%)	0 (00%)	1 (01%)
$S_{p\bar{f}}$	184 (98%)	1 (01%)	2 (01%)
$S_{\bar{p}f}$	12 (06%)	1 (01%)	174 (93%)
$S_{\bar{p}\bar{f}}$	6 (03%)	0 (00%)	181 (97%)

Q1: Was the goal treatment? (question with an objectively correct answer)

Scenario	Yes	I don't know	No
S_{pf}	187 (100%)	0 (00%)	0 (00%)
$S_{p\bar{f}}$	2 (01%)	0 (00%)	185 (99%)
$S_{\bar{p}f}$	179 (96%)	0 (00%)	8 (04%)
$S_{\bar{p}\bar{f}}$	3 (02%)	0 (00%)	184 (98%)

Q2: Was the treatment successful? (question with an objectively correct answer)

Scenario	Yes	I don't know	No
S_{pf}	185 (99%)	2 (01%)	0 (00%)
$S_{p\bar{f}}$	183 (98%)	1 (01%)	3 (02%)
$S_{\bar{p}f}$	43 (23%)	6 (03%)	138 (74%)
$S_{\bar{p}\bar{f}}$	38 (20%)	10 (05%)	139 (74%)

Q3: Was the action for the purpose?

Scenario	Yes	I don't know	No
S_{pf}	182 (97%)	2 (01%)	3 (02%)
$S_{p\bar{f}}$	177 (95%)	2 (01%)	8 (04%)
$S_{\bar{p}f}$	45 (24%)	9 (05%)	133 (71%)
$S_{\bar{p}\bar{f}}$	31 (17%)	9 (05%)	147 (79%)

Q4: Was the policy obeyed?

Table 5.4: Survey Responses. In Scenario S_{pf}, the case worker's goal was treatment and the treatment was successful; in $S_{p\bar{f}}$, the goal was treatment and it failed; in $S_{\bar{p}f}$, the goal was cost reduction and the treatment succeeded; and in $S_{\bar{p}\bar{f}}$, the goal was cost reduction and the treatment failed.

Testing	Alternative Hypothesis	Null Hypothesis	p-Value	Significant?
Against H1a	$p_{\mathsf{pfy}} < 0.5$	$p_{\mathsf{pfy}} = 0.5$	1	No
Against H1a	$p_{\mathsf{pfn}} > 0.5$	$p_{\mathsf{pfn}} = 0.5$	1	No
Against H1a	$p_{\mathsf{\bar{p}fy}} < 0.5$	$p_{\mathsf{\bar{p}fy}} = 0.5$	3.28889e-013	Yes
Against H1a	$p_{\mathsf{\bar{p}fn}} > 0.5$	$p_{\mathsf{\bar{p}fn}} = 0.5$	3.527326e-009	Yes
Against H1a'	$p'_{\mathsf{pfy}} < 0.5$	$p'_{\mathsf{pfy}} = 0.5$	1	No
Against H1a'	$p'_{\mathsf{pfn}} > 0.5$	$p'_{\mathsf{pfn}} = 0.5$	1	No
Against H1a'	$p'_{\mathsf{\bar{p}fy}} < 0.5$	$p'_{\mathsf{\bar{p}fy}} = 0.5$	3.08316e-014	Yes
Against H1a'	$p'_{\mathsf{\bar{p}fn}} > 0.5$	$p'_{\mathsf{\bar{p}fn}} = 0.5$	2.662347e-011	Yes
Against H1b	$p_{\mathsf{p\bar{f}n}} < 0.5$	$p_{\mathsf{p\bar{f}n}} = 0.5$	1.699463e-043	Yes
Against H1b	$p_{\mathsf{p\bar{f}y}} > 0.5$	$p_{\mathsf{p\bar{f}y}} = 0.5$	6.090736e-041	Yes
Against H1b	$p_{\mathsf{\bar{p}\bar{f}n}} < 0.5$	$p_{\mathsf{\bar{p}\bar{f}n}} = 0.5$	1	No
Against H1b	$p_{\mathsf{\bar{p}\bar{f}y}} > 0.5$	$p_{\mathsf{\bar{p}\bar{f}y}} = 0.5$	1	No
Against H1b'	$p'_{\mathsf{p\bar{f}n}} < 0.5$	$p'_{\mathsf{p\bar{f}n}} = 0.5$	5.556827e-051	Yes
Against H1b'	$p'_{\mathsf{p\bar{f}y}} > 0.5$	$p'_{\mathsf{p\bar{f}y}} = 0.5$	2.570485e-049	Yes
Against H1b'	$p'_{\mathsf{\bar{p}\bar{f}n}} < 0.5$	$p'_{\mathsf{\bar{p}\bar{f}n}} = 0.5$	1	No
Against H1b'	$p'_{\mathsf{\bar{p}\bar{f}y}} > 0.5$	$p'_{\mathsf{\bar{p}\bar{f}y}} = 0.5$	1	No
For H2a	$p_{\mathsf{pfy}} > 0.5$	$p_{\mathsf{pfy}} = 0.5$	9.461645e-048	Yes
For H2a	$p_{\mathsf{pfn}} < 0.5$	$p_{\mathsf{pfn}} = 0.5$	5.556827e-051	Yes
For H2a	$p_{\mathsf{p\bar{f}y}} > 0.5$	$p_{\mathsf{p\bar{f}y}} = 0.5$	6.090736e-041	Yes
For H2a	$p_{\mathsf{p\bar{f}n}} < 0.5$	$p_{\mathsf{p\bar{f}n}} = 0.5$	1.699463e-043	Yes
For H2a'	$p'_{\mathsf{pfy}} > 0.5$	$p'_{\mathsf{pfy}} = 0.5$	8.961588e-053	Yes
For H2a'	$p'_{\mathsf{pfn}} < 0.5$	$p'_{\mathsf{pfn}} = 0.5$	5.097894e-057	Yes
For H2a'	$p'_{\mathsf{p\bar{f}y}} > 0.5$	$p'_{\mathsf{p\bar{f}y}} = 0.5$	2.570485e-049	Yes
For H2a'	$p'_{\mathsf{p\bar{f}n}} < 0.5$	$p'_{\mathsf{p\bar{f}n}} = 0.5$	5.556827e-051	Yes
For H2b	$p_{\mathsf{\bar{p}fn}} > 0.5$	$p_{\mathsf{\bar{p}fn}} = 0.5$	3.527326e-009	Yes
For H2b	$p_{\mathsf{\bar{p}fy}} < 0.5$	$p_{\mathsf{\bar{p}fy}} = 0.5$	3.28889e-013	Yes
For H2b	$p_{\mathsf{\bar{p}\bar{f}n}} > 0.5$	$p_{\mathsf{\bar{p}\bar{f}n}} = 0.5$	7.078408e-016	Yes
For H2b	$p_{\mathsf{\bar{p}\bar{f}y}} < 0.5$	$p_{\mathsf{\bar{p}\bar{f}y}} = 0.5$	1.479279e-021	Yes
For H2b'	$p'_{\mathsf{\bar{p}fn}} > 0.5$	$p'_{\mathsf{\bar{p}fn}} = 0.5$	2.662347e-011	Yes
For H2b'	$p'_{\mathsf{\bar{p}fy}} < 0.5$	$p'_{\mathsf{\bar{p}fy}} = 0.5$	3.08316e-014	Yes
For H2b'	$p'_{\mathsf{\bar{p}\bar{f}n}} > 0.5$	$p'_{\mathsf{\bar{p}\bar{f}n}} = 0.5$	9.252051e-012	Yes
For H2b'	$p'_{\mathsf{\bar{p}\bar{f}y}} < 0.5$	$p'_{\mathsf{\bar{p}\bar{f}y}} = 0.5$	4.896385e-017	Yes

Table 5.5: Binomial Hypothesis Tests

Testing	Alternative Hypothesis	Null Hypothesis
Proving H2a	$p_{\mathsf{pfy}} > 0.94$	$p_{\mathsf{pfy}} = 0.94$
Proving H2a	$p_{\mathsf{pfn}} < 0.05$	$p_{\mathsf{pfn}} = 0.05$
Proving H2a	$p_{\mathsf{p\bar{f}y}} > 0.91$	$p_{\mathsf{p\bar{f}y}} = 0.91$
Proving H2a	$p_{\mathsf{p\bar{f}n}} < 0.08$	$p_{\mathsf{p\bar{f}n}} = 0.08$
Proving H2a’	$p'_{\mathsf{pfy}} > 0.96$	$p'_{\mathsf{pfy}} = 0.96$
Proving H2a’	$p'_{\mathsf{pfn}} < 0.02$	$p'_{\mathsf{pfn}} = 0.02$
Proving H2a’	$p'_{\mathsf{p\bar{f}y}} > 0.95$	$p'_{\mathsf{p\bar{f}y}} = 0.95$
Proving H2a’	$p'_{\mathsf{p\bar{f}n}} < 0.05$	$p'_{\mathsf{p\bar{f}n}} = 0.05$
Proving H2b	$p_{\mathsf{\bar{p}fn}} > 0.65$	$p_{\mathsf{\bar{p}fn}} = 0.65$
Proving H2b	$p_{\mathsf{\bar{p}fy}} < 0.3$	$p_{\mathsf{\bar{p}fy}} = 0.3$
Proving H2b	$p_{\mathsf{\bar{p}\bar{f}n}} > 0.73$	$p_{\mathsf{\bar{p}\bar{f}n}} = 0.73$
Proving H2b	$p_{\mathsf{\bar{p}\bar{f}y}} < 0.22$	$p_{\mathsf{\bar{p}\bar{f}y}} = 0.22$
Proving H2b’	$p'_{\mathsf{\bar{p}fn}} > 0.67$	$p'_{\mathsf{\bar{p}fn}} = 0.67$
Proving H2b’	$p'_{\mathsf{\bar{p}fy}} < 0.29$	$p'_{\mathsf{\bar{p}fy}} = 0.29$
Proving H2b’	$p'_{\mathsf{\bar{p}\bar{f}n}} > 0.68$	$p'_{\mathsf{\bar{p}\bar{f}n}} = 0.68$
Proving H2b’	$p'_{\mathsf{\bar{p}\bar{f}y}} < 0.26$	$p'_{\mathsf{\bar{p}\bar{f}y}} = 0.26$

Table 5.6: Extreme Binomial Hypothesis Tests. This table shows the hypothesis test using the most extreme probability for which statistical significance is still achieved and is accurate up to two places after the decimal point.

Since the results of the hypothesis testing where so strongly in favor of Hypothesis H2 using the probability of 0.5 as the null hypothesis, we decided to calculate the most extreme probabilities that still obtains significance. For testing that a probability is less than a value (lower tail rejection region), the most extreme value is the minimal value, whereas it is the maximum value for testing that a probability is greater than a value (upper tail rejection region). Table 5.6 shows these probabilities conservatively calculated up to 0.01 away from the true extreme probability. For example, the bottom row shows $p'_{4\mathsf{y}}$ is less than 0.26 with statistical significance but not less than 0.25 with statistical significance. (This does not imply that $p'_{4\mathsf{y}} > 0.25$ with statistical significance.) As these probabilities are more extreme for Hypotheses H2a and H2a’ than Hypotheses H2b and H2b’, H2a and H2a’ appear to be more accurate. However, as we added these statistics to the analysis after having conducted the survey, they may suffer from confirmation bias.

Table 5.7 shows the results of using McNemar’s Test to compare the distribution of responses to one question across two scenarios. For example, the last row compares the distribution producing responses to Question Q3 for Scenario $\mathsf{S}_{\mathsf{p\bar{f}}}$ to that producing responses for Scenario $\mathsf{S}_{\mathsf{\bar{p}\bar{f}}}$. McNemar’s Test shows that the differences in the observed responses are statistically significant. This result indicates that the two distributions differ as predicted by Hypothesis H2c’. On the other hand, the fourth line of Table 5.7 shows that the responses for Question Q3 do not differ significantly across Scenarios $\mathsf{S}_{\mathsf{\bar{p}f}}$ and $\mathsf{S}_{\mathsf{\bar{p}\bar{f}}}$. This result differs from Hypothesis H1, which predicts that people would answer the question differently across the two scenarios. McNemar’s Test validates all four predictions of Hypothesis H2. It validates one of the predictions of Hypothesis H1. The statistic could not be computed in one case as the data was too sparse for the calculation.

Testing	Question	Scenarios	p-Value	Significant?
For H1c	Q4	S_{pf} vs. $S_{\mathsf{p\bar{f}}}$	NaN	No
For H1c	Q4	$S_{\mathsf{\bar{p}f}}$ vs. $S_{\mathsf{\bar{p}\bar{f}}}$	0.02674664	Yes
For H1c’	Q3	S_{pf} vs. $S_{\mathsf{p\bar{f}}}$	0.3916252	No
For H1c’	Q3	$S_{\mathsf{\bar{p}f}}$ vs. $S_{\mathsf{\bar{p}\bar{f}}}$	0.3951831	No
For H2c	Q4	S_{pf} vs. $S_{\mathsf{\bar{p}f}}$	1.020173e-029	Yes
For H2c	Q4	$S_{\mathsf{p\bar{f}}}$ vs. $S_{\mathsf{\bar{p}\bar{f}}}$	3.112267e-031	Yes
For H2c’	Q3	S_{pf} vs. $S_{\mathsf{\bar{p}f}}$	5.186851e-031	Yes
For H2c’	Q3	$S_{\mathsf{p\bar{f}}}$ vs. $S_{\mathsf{\bar{p}\bar{f}}}$	8.40055e-031	Yes

Table 5.7: McNemar’s Tests Across Scenarios

Scenario	Questions	p-Value	Significant?
S_{pf}	Q4 vs. Q3	NaN	No
$S_{\mathsf{p\bar{f}}}$	Q4 vs. Q3	NaN	No
$S_{\mathsf{\bar{p}f}}$	Q4 vs. Q3	0.3843414	No
$S_{\mathsf{\bar{p}\bar{f}}}$	Q4 vs. Q3	0.2239329	No

Table 5.8: McNemar’s Tests Across Questions

We were surprised to see the degree of difference between how people answered Questions Q4 and Q3. For example, for Scenario $S_{\mathsf{\bar{p}\bar{f}}}$, 79% of respondents answered Question Q5 with *no* whereas only 74% answered Q3 with *no* despite our belief that both questions should be answered identically (see Table 5.4). To test whether these differences are statistically significant, we used McNemar’s test to compare the responses to these two questions within a single scenario. Table 5.8 shows the results. None of the tests showed a statistically significant difference in how the questions were answered, but two of the tests failed to produce a numeric p-value.

5.5 Limitations of Study

Various factors affect the validity of our conclusions. We discuss each of them below.

By mentioning whether the auditee is performing the action as part of a plan, it forces the participant to consider the relationship between purposes and plans. It is possible that participants not primed to think about planning would substantiate H1.

The use of Mechanical Turk raises questions about how representative our population sample is. Ross et al. look at the demographics of Mechanical Turk workers and find that among U.S. workers, a disproportionate number are female [RIS$^+$10]. However, Berinsky, Huber, and Lenz find that Mechanical Turk studies are as representative, if not more representative, than convenience samples commonly used in research [BHL11]. While we attempted to limit our sample to adults in the United States, Mechanical Turk’s ability to verify the qualification criteria is limited. Even given a representative pool of Mechanical Turk workers for our sampling frame, our sample may be biased as the participants selected to take our survey rather than us having randomly selected them from the pool.

In many cases, lawyers write privacy policies using the jargon of their profession. We have not studied whether lawyers or others involved in the writing or enforcing of privacy policies (e.g., auditors) understand purpose restrictions in a manner different from the population at large, which we attempted to sample for this survey. While surveying such professions is interesting future work, we believe that the lay understanding is important since public-facing privacy policies often contain purpose restrictions.

The use of paid but unmonitored participants, also raises concerns that participants might provide arbitrary answers to speed through the questionnaire. Kittur, Chi, and Suh present experimental results of using Mechanical Turk for user studies [KCS08]. They conclude that Mechanical Turk can be useful if one eliminates such spurious submissions by including questions with known answers and rejecting participants who fail to correctly answer these questions. We follow this protocol by using Questions Q1 and Q2 to force the participant to read the scenarios and by notifying survey participants that we may withhold payment if they answer arbitrarily. Answering the remaining questions (Q3, Q4, and Q5) becomes fairly easy after having correctly answered Questions Q1 and Q2. By making the additional work required for meaningful participation small, we hope to have reduced arbitrary responses. However, by threatening to withhold payment, we may have increased the *demand effect*, the tendency of participants to provide the answers they believe the surveyor would like to observe as opposed to their honest opinions (see, e.g., [Orn09]).

Some respondents might answer later questions in a manner consistent with their answers to earlier questions despite having differing opinions. This bias could arise since some of the differences between questions may appear trivial, especially since we made each question similar to the others to reduce confounding factors. As no scenario has the same answers to both Questions Q1 and Q2 together as any other scenario, we hope to have reduced this bias.

Nonattitudes occur when a participant arbitrarily selects a response since they do not have an opinion on a question. To reduce the effect of nonattitudes, we included the option of a *I don't know* response.

We do not claim that the questionnaire tests all relevant factors (i.e., we do not claim high content validity). Indeed, we did not test some factors that we suspect may affect respondents such as whether the policy is perceived as good or bad.

Another concern is that respondents may change their opinions over time. We did not perform a follow-up study to determine how reliable our survey is over time.

It is also possible that our survey questions are not understood by the respondents in a manner consistent with testing the meaning of *purpose*. The various forms of validity discussed below attempt to determine whether our survey actually measured the concepts in which we are interested.

We believe that our survey has face validity. That is, we believe that our questions are, on their face, well worded for testing our hypotheses.

Including both Questions Q3 and Q4 not only allowed us to compare the truth of Hypothesis H1a to H1a' (and likewise with the other unprimed-primed pairs of hypotheses), but also to see the effects of the changing the wording of the questions. As the respondents typically answered these two questions in the same manner, we believe that our results are not overly influenced by the wording of the questions and pertain to the underlying concepts. That is, we believe our survey has convergent validity. However, that some respondents varied their responses across Questions Q4 and Q3 within a single scenario deserves further investigation.

As we know of no previous empirical research addressing the issues tested by our study, we cannot compare our results to those already proved about the meaning of *purpose*. Thus, we cannot that argue that our survey has construct validity by showing that it agrees with previous results. However, prior work has studied how people assign *goals* to actions [BTS06, BTS07, BST09, BST11]. These studies have found that planning-based models similar to ours predict the goals people assign to animated characters better than heuristics.

A survey respondent may confuse the concepts we are testing with related ones reducing the divergent validity of our survey. For example, rather than actually answer Question Q4, they may instead provide the answer to the following question: "Was the case worker's action consistent with someone seeking treatment for the patient?" Such confusion may explain some of the unexpected variation in responses between Questions Q3 and Q4.

The ultimate goal of our work is to determine how people think policies involving the concept of *purpose* should be enforced. Our survey is detached from any actual enforcement. Respondents might behave differently than their responses suggest given the task of actually enforcing a policy. They may also differ from their responses in their feelings if they were actually subject to such a policy. Our survey is most similar to the respondent acting as a neutral third-party or judge in a dispute over the meaning of a policy. However, even in such a role, the respondent's behavior may differ from that suggested from his responses. Ideally, our survey will predict with a high degree of accuracy how the respondents would behave in each of these three roles (policy enforcer, policy subject, or neutral third-party) establishing that our survey actually corresponds to the behavior we wish to study (i.e., has criterion validity). However, we have not established this form of validity.

5.6 Discussion

The results shown above provide evidence in favor of defining an action to be for a purpose if and only if an agent performed the action as part of a plan for furthering that purpose (Hypothesis H2). The binomial tests provide strong evidence against defining an action to be for a purpose if and only if that action furthered the purpose (Hypothesis H1). McNemar's test provides some support for Hypothesis H1. Indeed, informally examining the response distributions (Table 5.4), it appears Hypothesis H1 does accurately model a small minority of respondents. However, Hypothesis H2 appears to accurately model a much larger number of respondents. For these reasons, we conclude that Hypothesis H2 provides a superior model to that of Hypothesis H1.

Nevertheless, the relative strength of Hypothesis H2a compared to Hypothesis H2b suggests that some people feel that an action being for a purpose is sufficient but not necessary for an action to be for a purpose. Examining free-form responses to Question Q5 suggests that some people feel that the action of sharing a record is for the purpose of treatment since it is the same action that would be taken had the case worker been planning for treatment. This suggests a third class of hypotheses:

H3 The auditee obeys the (purpose) restriction if and only if the auditee performed an action that a hypothetical agent would take had it planned for the purpose.

H3' An action is for a purpose if and only if that action is the action a hypothetical agent would take had it planned for the purpose.

These hypotheses place strictly weaker restrictions on the auditee's behavior consistent with the idea that H2 is sufficient but not necessary. Interestingly, they match the approximations our algorithm makes in attempting to enforce Hypothesis H2. Unfortunately, by not mentioning whether the case worker's choice to forward the record in Scenarios $\mathrm{S}_{\bar{\mathsf{p}}\mathsf{f}}$ and $\mathrm{S}_{\bar{\mathsf{p}}\bar{\mathsf{f}}}$ is consistent with the actions of a hypothetical agent planning for treatment, we cannot test these hypotheses using the conducted survey.

Chapter 6

Multiple Purposes and Limitations

6.1 Introduction

So far, our formalism allows our hypothetical agent to consider only a single purpose. However, auditees may perform an action for more than one purpose. In many cases, the auditor may simply ignore any action that is not governed by the privacy policy and not relevant to the plans the auditee is employing that uses governed actions.

In the physician example of Chapter 2, the physician already implicitly considered many other purposes before even seeing this current patient. For example, the physician presumably performed many actions not mentioned in the model in between taking the X-ray, sending it, and making a diagnosis, such as going on a coffee break. As these actions are not governed by the privacy policy and neither improves nor degrades the diagnosis even indirectly, the auditor may safely ignore them. Thus, our semantics can handle multiple purposes in this limited fashion.

However, in other cases, the interactions between purposes become important. Below we discuss two complementary ways that an auditee can consider multiple purposes that produce interactions. In the first, the auditee considers one purpose after another. In the second, the auditee attempts to optimize for multiple purposes simultaneously. We find that our semantics may easily be extended to handle the first, but difficulties arise for the second. We end the section by considering what features a formalism would need to handle simultaneous consideration of purposes and the challenges they raise for auditing.

6.2 Sequential Consideration

Yahoo!'s privacy policy states that they will not contact children for the purpose of marketing [Yah10a]. Suppose Yahoo! decides to change the name of `games.yahoo.com` to `fun.yahoo.com` because they believe the new name will be easier to market. They notify users of `games.yahoo.com`, including children, of the upcoming change so that they may update their bookmarks.

In this example, the decision to change names, made for marketing, causes Yahoo! to contact children. However, we do not feel that this is a violation of Yahoo!'s privacy policy. A decision made for marketing altered the expected future of Yahoo! in such a way that customer service would suffer if Yahoo! did nothing. Thus, to maintain good customer service, Yahoo! made the decision to notify users without further consideration of marketing. Since Yahoo! did not consider the purpose of marketing while making this decision, contacting the children was not *for* marketing despite Yahoo! considering the implications of changing the name for marketing while making its decision to contact children.

Bratman describes such planning in his work formalizing *intentions* in the Belief-Desire-Intention (BDI) model [Bra87]. He views it as a sequence of planning steps in which the intention to act (e.g., to change the name) at one step may affect the plans formed at later steps. In particular, each step of planning starts with a model of the environment that is refined by the intentions formed by each of the previous planning steps. The step then creates a plan for a purpose that further refines the model with new intentions resulting from this plan. A purpose associated with a previous step constrains the choices available at a later step of planning. Thus, a purpose associated with a previous step may affect the plan formed in a later step for a different purpose. We adopt the stance that an action selected at a step is *for* the purpose optimized at that step. However, the action is not also for other previous purposes even if the constraints created previously by planning for those purposes affect which action the agent chooses at the current step.

Baker et al. formalizes a simple form of sequential planning for an MDP model with multiple goals, but they do not support intentions from previous goals affecting future goals [BTS07, BST09].

6.3 Simultaneous Consideration

At other times, an auditee might consider more than one purpose in the same step. For example, the physician may have to both provide quality treatment and to respect the patient's financial concerns. In this case, the physician may not be able to simultaneously provide the highest quality care at the lowest price. The two competing concerns must be balanced and the result may not maximize the satisfaction of either of them.

The traditional way of modeling the simultaneous optimization of multiple rewards is to combine them into a single reward using a weighted average over the rewards. Each reward would be weighted by how important it is to the auditee performing the optimization. This amalgamation of the various purpose rewards makes it difficult to determine for which purpose various actions are selected.

One possibility is to analyze the situation using counterfactual reasoning (see, e.g., [Mac74]). For example, given that the auditee performed an action a while optimizing a combination of purposes p_1 and p_2, the auditor could ask if the auditee would have still performed the action a even if the auditee had not considered the purpose p_1 and had only optimized the purpose p_2. If not, than the auditor could determine that the action was for p_1. However, as the next example shows, such reasoning is not sufficient to determine the purposes of the actions.

To show the generality of purposes, we consider an example involving travel reimbursement instead of privacy. Consider a Philadelphian who needs to go to New York City for a business meeting with his employer and is invited to give a lecture at a conference in Washington, D.C., with his travel expenses reimbursed by the conference. He could drive to either New York or Washington (modeled as the actions $\mathsf{driveNY}$ and $\mathsf{driveDC}$, respectively). However, due to time constraints he cannot drive to both of them. To attend both events, he needs to fly to both (modeled as actions flyNY and flyDC). As flying is more expensive, both driving actions receive a higher reward than flying (2 instead of 1), but flying is better than not going (0). Figure 6.1 models the traveler's environment.

Given these constraints, he decides to fly to both only to find auditors at both events scrutinizing his decision. For example, an auditor working for the conference could find that his flight to Washington was not for the lecture since the traveler would have driven had it not been for work. If the conference's policy requires that reimbursed flights are *only for* the lecture, the auditor might deny reimbursement. However,

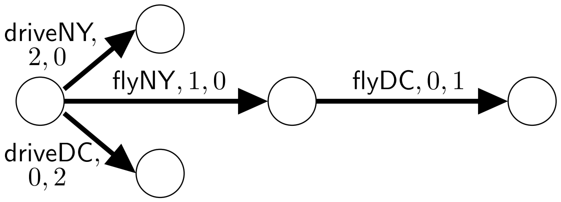

Figure 6.1: Model of a traveler deciding whether to fly or drive. Since every transition is deterministic, we represent each as a single arrow. Each is labeled with the action name, the rewards for business and the rewards for lecturing in that order. Self-loops of zero reward are not shown including all those labeled with the do-nothing action stop.

the employer seems even less likely to reimburse the traveler for his flight to Washington since the flight is redundant for getting to New York.

However, under the semantics discussed above, each flight would be for both purposes since only when the traveler considers both does he decide to take either flight. While having the conference reimburse the traveler for his flight to Washington seems reasonable, the idea that they should also reimburse him for his flight to New York appears counterintuitive.

Our approach of sequential planning also cannot explain this example. To plan sequentially, the traveler must consider one of the two events first. If, for example, he considers New York first, he will decide to drive to New York and then decline the invitation to Washington. Only by considering both events at once, does he decide to fly and is able to satisfy both purposes.

We believe resolving this conflict requires extending our semantics to consider requirements that an action be *for* a purpose (as opposed to *not for* or *only for*). Furthermore, we believe that the optimization of combinations of purposes does not accurately model human planning with multiple purposes. Intuitively, the traveler selects flyDC not *for* work but also not *only for* the conference. Rather flyDC seems be *for* the conference under the constraint that it must not prevent the traveler from attending the meeting. In the next section, we consider the possibility of modeling human planning more accurately.

Handling multiple purposes is likely to lead to computational intractability. Blokpoel, Kwisthout, van der Weide, and Rooij examine goal inference for an MDP-like model with multiple goals [BKvdWvR10]. They find that supporting numerous goals (or purposes) results in intractability. Given that their model does not handle intentions, we suspect that the approach we envision will be even more complex.

6.4 Modeling Human Planning

MDPs and POMDPs are useful for automated planning and producing optimal strategies for informing operating procedures as discussed in Chapter 4. However, they are not specialized for modeling planning by humans who plan in a significantly different manner. First, while planning, humans tend not to combine their purposes into a single reward or utility function as suggested above, but rather maintain a vector of reward functions each corresponding to a separate purpose [Sim55]. Second, except while playing well defined games, constructing an MDP or POMDP model of the environment is a costly process involving the collection of information and its analysis. Indeed, the auditee will need a plan to construct his model [Sti61], but such a planning problem is even more complex leading to an infinite digression [Sel02]. Third, while the algorithms used for MDP optimization are conceptually simple, they involve numeric calculations beyond the comfort of humans [Sim55]; POMDP optimization is difficult even for computers [Mad00]. Fourth, experiments suggest that humans tend not to use a discounting factor such as γ, but rather discount in a more ad hoc manner resembling hyperbolic discounting [Ain92].

For these reasons, others have searched for more tailored models of human planning [Sim55, GS02]. Simon proposed to model humans as having *bounded rationality* to account for their limitations and their lack of information [Sim55]. Work on formalizing bounded rationality has resulted in a variety of planning principles ranging from the systematic (e.g., Simon's *satisficing*) to the heuristic (see, e.g., [Gig02]). However, "[a] comprehensive, coherent theory of bounded rationality is not available" [Sel02, p14] and there still is "a significant amount of unpredictability in how an animal or a human being will undertake to solve a problem" such as planning [DKP96, p40].

We view creating semantics more closely tied to human planning interesting future work. However, modeling human planning may prove complex enough to justify accepting the imperfections of semantics such as ours or even heuristic based approaches for finding violations such as the query intrusion model discussed above [AKSX02].

Despite these difficulties, one could look for discrepancies between a semantics of purpose restrictions and experimental results on planning. In this manner one could judge how closely a semantics approximates human planning in the ways relative to purpose restrictions.

In particular, our semantics appears to hold human auditees to too high of a standard: they are unlikely to always be able to pick the optimal strategy for a purpose. When enforcing an exclusivity rule, this strictness could result in the auditor investigating some auditees who honestly planned for the only allowed purpose, but failed to find the optimal policy. While such investigations would be false positives, they do have the pleasing side-effect of highlighting areas in which an auditee could improve his planning.

In the case of enforcing prohibitive rules, this strictness could cause the auditor to miss some violations that do not optimize the prohibited purpose, but, nevertheless, are for the purpose. The additional checks proposed at the end of Section 2.2.3 could be useful for detecting these violations: if the auditee's actions are not consistent with a strategy that optimizes any of the allowed purposes but does improve to some degree the prohibited purpose, the actions may warrant extra scrutiny.

Prior work on goal inference has used more relaxed criterion than optimality to account for a human's inability to optimize (e.g., [BST11, RG11]). These relaxations amount to allowing the auditee to deviate from the optimal strategy with a probability proportional to the suboptimality of the deviation (or an approximation of the suboptimality in the case of [BST11]). We believe that an accurate model would also have to allow for deviations due to the difficulty of determining the consequences of an action.

While our semantics is limited by our understanding of human planning, it still reveals concepts crucial to the meaning of *purpose*. Ideas such as planning and non-redundancy will guide future investigations on the topic.

Chapter 7

Related Work

7.1 Applying our Formalism to Past Methods

Past methods of enforcing purpose restrictions have not provided a means of assigning purposes to sequences of actions. Rather, they presume that the auditor (or someone else) already has a method of determining which behaviors are for a purpose. In essence, these methods presuppose that the auditor already has the set of allowed behaviors $\mathsf{nbehv}(r^p)$ for the purpose p that he is enforcing. These methods differ in their intensional representations of the set $\mathsf{nbehv}(r^p)$. Thus, some may represent a given set exactly while others may only be able to approximate it. These differences mainly arise from the different mechanisms they use to ensure that the auditee only exhibits behaviors from $\mathsf{nbehv}(r^p)$. We use our semantics to study how reasonable these approximations are.

Research Efforts. Byun et al. use role-based access control [San96] to present a methodology for organizing privacy policies and their enforcement [BBL05, BL08, NBL$^+$10]. They associate purposes with sensitive resources and with roles, and their methodology only grants the user access to the resource when the purpose of the user's role matches the resource's purpose. The methodology does not, however, explain how to determine which purposes to associate with which roles. Furthermore, a user in a role can perform actions that do not fit the purposes associated with his role allowing him to use the resource for a purpose other than the intended one. Thus, their method is only capable of enforcing policies when there exists some subset A of the set of actions $\mathcal{A}$ such that $\mathsf{nbehv}(r^p)$ is equal to the set of all interleavings of A with $\mathcal{S}$ of finite but unbounded length (i.e., $\mathsf{nbehv}(r^p) = (\mathcal{S} \times A)^*$). The subset A corresponds to those actions that use a resource with the same purpose as the auditee's role. Despite these limitations, their method can implement the run-time enforcement used at some organizations, such as a hospital that allows physicians access to any record to avoid denying access in time-critical emergencies. However, it does not allow the fine-grain distinctions used during post-hoc auditing done at some hospitals to ensure that physicians do not abuse their privileges. *Group-centric access control* has similar advantages and limitations offers similar advantages but suffers from the same shortcomings [KSNW09].

Al-Fedaghi uses the work of Byun et al. as a starting point but concludes that rather than associating purposes with roles, one should associate purposes with sequences of actions [AF07]. Influenced by Al-Fedaghi, Jafari et al. adopt a similar position calling these sequences *workflows* [JSNS09]. The set of workflows allowed for a purpose p corresponds to $\mathsf{nbehv}(r^p)$. They do not provide a formal method of determining which workflows belong in the allowed set leaving this determination to the intuition of the auditor. They do not consider probabilistic transitions and the intuition they supply suggests that they would only include workflows that successfully achieve or improve the purpose. Thus, our approach appears more lenient by including some behaviors that fail to improve the purpose. As shown in Chapter 5, this leniency is key to capturing the semantics of purpose restrictions. An auditor could encode a workflow in the state of the environment to get results similar to Al-Fedaghi's or Jafari et al.'s results while using *Contextual Role-Based Access Control* [MF03] or *Situation-Based Access Control* [PBDD08].

Others have adopted a hybrid approach allowing the roles of an auditee to change based on the state of the system [PGY08, EKWB11]. These dynamic roles act as a level of indirection assigning an auditee to a state. This indirection effectively allow role-based access control to simulate the workflow methods to be just as expressive.

Agrawal et al. propose a methodology called *Hippocratic databases* for protecting the privacy of subjects of a database [AKSX02]. They propose to use a *query intrusion model* to enforce privacy polices governing purposes. Given a request for access and the purpose for which the requester claims the request is made, the query intrusion model compares the request to previous requests with the same purpose using an approach similar to intrusion detection. If the request is sufficiently different from previous ones, it is flagged as a possible violation. While the method may be practical, it lacks soundness and completeness. Furthermore, by not being semantically motivated, it provides no insight into the semantics of purpose. To avoid false positives, the set of allowed behaviors $\mathsf{nbehv}(r^p)$ would have to be small or have a pattern that the query intrusion model could recognize.

Jif is a language extension to Java designed to enforce requirements on the flows of information in a program [CMVZ09]. Hayati and Abadi explain how to reduce purpose restrictions to information flow properties that Jif can enforce [HA05]. Their method requires that inputs are labeled with the purposes for which the policy allows the program to use them and that each unit of code be labeled with the purposes for which that code operates. If information can flow from an input statement labeled with one purpose to code labeled for a different purpose, their method produces a compile-time type error. (For simplicity, we ignore their use of sub-typing to model sub-purposes.) In essence, their method enforces the rule *if information i flows to code c, then i and c must be labeled with the same purpose*. The interesting case is when the code c uses the information i to perform some observable action $a_{c,i}$, such as producing output. Under our semantics, we treat the program as the auditee and view the policy as limiting these actions. By directly labeling code, their method does not consider the contexts in which these actions occur. Rather the action $a_{c,i}$ is either aways allowed or always disallowed based on the purpose labels of c and i. By not considering context, their method is subject to the same limitations as the method of Byun et al. with the subset A being equal to the set of all actions $a_{c,i}$ such that c and i have the same label. However, using more advanced type systems (e.g., typestate [SY86]), they might be able extend their method to consider the context in which code is executed and increase the method's expressiveness.

Commercial Products. FairWarning and Cerner's P2Sentinel are commercially available auditing systems designed for the hospital setting [Fai, Cer]. They log employee accesses to medical records by collecting the accesses across the various computer systems a typical hospital uses. The systems come with a selection of queries over the audit log that recognize common suspicious actions such as an employee looking up the record of a celebrity or someone with the same last name (a possible relative). They also allow the auditor create his own queries. In principle, the auditor my craft any query including ones that embed our auditing algorithms. Thus, these systems are, in theory, capable of enforcing purpose restrictions with our semantics.

In practice, auditors using these systems have employed much simpler queries. Thus, we compare our auditing algorithms to the ability of these systems to enforcement purpose restrictions using only the typical queries employed by users of them. For example, the marketing material for FairWarning highlights nine example queries (see `http://www.fairwarning.com/subpages/monitoring.asp`):

1. VIP record snooping
2. Executive record snooping
3. Patient / employee record snooping
4. Family member and self-examination of records
5. Neighbor record snooping
6. Identity Theft
7. Medical Identity Theft
8. Simultaneous logins from addresses or terminals
9. Repeated login failures

Of these, the last four are authentication issues orthogonal purpose restrictions. The first five are examples of purpose violations. Employees should only access records for valid purposes such as treatment. These five examples are cases in which an employee accesses a record for the illegitimate purpose of curiosity (i.e., to snoop). In each of these examples, the illegitimate purpose of curiosity is highly suggested by the identities of the record's user (the employee) and the record's subject (the patient). For example, in the case of VIP record snooping, the patient being a VIP makes every access to his record suspicious since many employees would be curious about a VIP's condition.

Under the approach of these systems, to enforce a prohibitive rule restricting behavior to being not for a purpose p, the auditor would select a set of queries that check whether an auditee's behavior lays in the set $\mathsf{nbehv}(r^p)$ and report a violation if so. In the above five examples, these queries take the form of checks on the identities of the record's user and the record's subject. Such identity checks are only capable of recognizing sets $\mathsf{nbehv}(r^p)$ where a behavior b is either in or not in $\mathsf{nbehv}(r^p)$ based on these identities and not how the record is used or any further context.

More generally, the auditor may craft more complex queries to recognize $\mathsf{nbehv}(r^p)$. However, since the auditor might not think of all behaviors in $\mathsf{nbehv}(r^p)$ or all the queries needed to detect such behaviors, the auditor might only recognize a subset of $\mathsf{nbehv}(r^p)$, making the method incomplete. Like our auditing methodology, this approach is also unsound in that a behavior identified as being in $\mathsf{nbehv}(r^p)$ might actually be permissible by also being in $\mathsf{nbehv}(r^{p'})$ for some allowed purpose p'.

To enforce an exclusivity rule restricting behavior to being for only the purpose p, the auditor would select a set of illegitimate purposes that may tempt the auditee into violating the restriction. The auditor would then enforce a set of prohibitive rules against these illegitimate purposes. Whereas this approach looks for suspicious behaviors, our algorithms verify that the observed behavior could be for the allowed purpose.

Thus, unlike an auditor using our methodology, an auditor using queries with a system like FairWarning or P2Sentinel must determine all the tempting illegitimate purposes or risk undetected violations. Furthermore, as discussed above, enforcing each possible prohibitive rule is neither sound nor complete.

The approach of FairWarning and P2Sentinel offers both advantages and disadvantages compared to our methodology. The commercial approaches allow the auditor to select those behaviors most concerning to him for detection. While the auditor must have a good understanding of the threats facing the hospital, he does not need to formalize this knowledge into an environment model. Since each query is independent of the others, the auditor can expect reasonable performance even when understanding only a subset of the threats. Furthermore, the auditor may add to or refine the queries he uses as his knowledge of the hospital and the threats improves. Our method, on the other hand, requires the auditor to formalize the functioning of the hospital as an environment model. While the auditor can refine this model as the auditor's knowledge of the hospital improves, our algorithms require a fairly accurate model from the beginning to avoid false positives and false negatives. (See Chapter 4 for an example of model refinement and its consequences.) The amount of work involved in formally modeling the environment implies that our methodology is less appropriate when auditing a small number of accesses. However, our approach computes from this model all the behaviors that are not for the purpose in question. Thus, while enforcing an exclusivity rule, our approach does not require the auditor to understand the threats facing the hospital. Given these trade-offs, the commercial approach does comparably well given an environment that the auditor does not understand well in comparison to the threats facing it; our approach does comparably well when the auditor understands the environment but the environment faces emerging threats that the auditor knows less about.

7.2 Related Problems in Policy Enforcement

We have already covered the most closely related work in Section 7.1. Below we discuss work on related problems in computer science.

Minimal Disclosure. The works most similar to ours in approach have been on *minimal disclosure*, which requires that the amount of information used in granting a request for access should be as little as possible while still achieving the purpose behind the request. Massacci, Mylopoulos, and Zannone define minimal disclosure for Hippocratic databases [MMZ06]. Barth, Mitchell, Datta, and Sundaram study minimal disclosure in the context of workflows [BMDS07]. They model a workflow as meeting a utility goal if it satisfies a temporal logic formula. Minimizing the amount of information disclosed is similar to an agent maximizing his reward and thereby not performing actions that have costs but no benefits. However, we consider several factors that these works do not, including quantitative purposes that are satisfied to varying degrees and probabilistic behavior resulting in actions being for a purpose despite the purpose not being achieved, which is necessary to capture the semantics of purpose restrictions (Chapter 5).

Expressing Privacy Policies with Purpose. Work on understanding the components of privacy policies has shown that *purpose* is a common component of privacy rules (see, e.g., [BA05, BA08]).

Some languages for specifying access-control policies allow the purpose of an action to partially determine if access is granted. For example, EPAL is a language in which privacy policies are expressed by listing all the conditions under which a system should grant a request for access to sensitive resources [PS03]. These conditions may depend upon four factors: the identity of the requester for access, the resource requested, the action the requester would like to perform on the resource, and the purpose for which the requester would like to perform the action. However, EPAL lacks a formal semantics that describes when an action is for a purpose and treats purposes as syntactic labels. Rather, it depends on the system making use of the language to determine what actions are for what purposes and provides no formal guidance as to how the system should make this determination.

The Platform for Privacy Preferences (P3P) offers a language for specifying the privacy policies of websites [Cra02]. These policies must state the purposes for which the website collects information. The policy may either reference one of the predefined purposes that the language offers or provide a custom purpose. The specification of the language provides a description of each of the predefined purposes in natural language [CLM$^+$02]. The policy author must provide such a description for any custom purposes he uses. We hope our work will provide a method of formalizing when information use meets the requirements of these descriptions.

SPARCLE is a system for authoring and examining privacy policies [BKKF05, BKK06]. The system consumes policies written in a restricted form of natural language and parses them into standard components. The system then allows the user to examine the policy by focusing on different components, edit the policy, and translate the policy into machine readable formats (e.g., EPAL). One of the standard components SPARCLE considers is purpose. While SPARCLE is capable of identifying restrictions on purpose in a policy, it does not assign a semantics to these restrictions.

Hanson et al. provide an algebra for tracking the permissible uses of data as it is transferred from system to system and is combined with other information [HBLK$^+$07]. However, this work is not concerned with the meaning of *purpose* or *for*.

7.3 Works from Philosophy and Psychology

Philosophy concerns defining the meaning of words. Philosophers typically proceed by iteratively refining a definition to match their intuitions about each new example of the word's use. The experimental methods of psychology (defined broadly to include linguistics and cognitive science) have given rise to experimental philosophy. This hybrid methodology studies the meaning of the words by looking at the most common view of a population rather than the intuitions of experts. Our work uses intuition until Chapter 5, which presents a survey. Both philosophy and psychology apply their methodologies to understanding the nature of human planning. We discuss these efforts below.

Philosophical Foundations. Taylor provides a detailed explanation of the importance of planning to the meaning of *purpose*, but does not provide any formalism [Tay66]. Taylor concludes that one must distinguish the purpose of actions from their effects: the effects are the actual results of the actions whereas the purpose is merely the desired effects (page 216). Our model formalizes this distinction by allowing an action to be for a purpose despite that purpose not being achieved.

Unlike Taylor, most philosophical works use *purpose* in the sense of *the purpose of life*, which differs from the sense in which privacy policies use the word. Many works use *desire* and *motivation* to refer to the feelings and attitudes that cause an agent to act. That is, desires correspond to purposes in our formalism. (However, the usage and connotations do differ: while a jaded physician may order tests for the purpose of treatment, we would not say "he desires treatment" and probably not even "he desires to provide treatment", but “he desires to keep his job and get paid” sounds reasonable. Whether these differences have ramifications for our formalism is future work.) These works discuss desires to define *intentions*. Intentions typically refers to the modifications the agent hopes to make to the state of the world. That is, in our formalism, intentions are actions the agent plans to take under the strategy it has selected. For example, the desire for satiety motivates the intention to go grocery shopping.

The modern philosophical work in the area of intentions starts with Anscombe who argues that the intention of an action is the answer offered to the question Why did you perform that action? [Ans57].

Bratman builds on Anscombe’s work by emphasizing the importance of agent planning in determining intentions to create the Belief-Desire-Intention (BDI) model [Bra87]. In Bratman’s work, an intention is an action an agent plans to take where the plan is formed while attempting to maximize the satisfaction of the agent’s desires. To some extent, Section 2.1 may be viewed as a formalization of a simplification of Bratman’s view. (The plans of Bratman are more complex than our strategies to account for the limited reasoning abilities of humans.)

Using Bratman’s work as a starting point, Cohen and Levesque present a logical formalization of when an agent intends to perform an action or intends to bring about a state of affairs [CL90]. Roughly speaking, under their formalism, an agent intends to satisfy a predict p over states if and only if the agent has knowingly performing a sequence of actions that makes p true as a goal that it believes it can achieve and will continue to attempt to make p true until it believes it is impossible to do so. These predicates are related to binary purpose scores, and our formalism produces strategies that roughly correspond to the intentional actions of Cohen and Levesque. However, our formalism also handles quantitative purposes and information use. Cohen and Levesque comment on the existence of quantitative purposes and propose to model them as a series of intentions, but do not provide a formalism to do so.

Intentions also affect planning and will become important as we search for more accurate models of human planning. Roy use logics and game theory to formalize how intentions can affect an agent’s planning [Roy08]. He uses his formalism to study when an agent’s plan is rational given the agent’s intentions. Given the auditee’s intentions, we could replace our MDP formalism of planning with Roy’s intention-driven formalism.

By modeling auditees as planning agents with desires (purposes) and beliefs, the auditor has adopted what Dennett calls *the intentional stance* toward them [Den87]. While our algorithms assume just the basic intentional stance that the auditor may accurately *model* auditees as planning agents, our discussions of issues such as tenable deniability suggests the stronger view that auditees actually plan and have desires and beliefs. (See page 34 of [Den87] for a discussion of the difference.) Applying the intentional stance to auditees such as computer programs or whole organizations is less familiar than applying it to humans, but fits Dennett’s scheme. However, the case of computer programs is problematic when the point of auditing is to assign moral blame or punishment: computer programs are not moral entities and will not respond to punishment. In such cases, the auditor should consider the programmer, not the program, as the auditee. For example, in Section 3.2.2, we create a POMDP for a website to determine how it uses information for advertising. In the example, we audited “the website”, which could either refer to the program running the website or the programmers of the website. While we can model this program as having a purpose, it is unreasonable to punish it. Thus, the auditee is better understood as the programmers (an organization) of the website, whose behavior is codified by the program. Under this view, the program may be identified with the strategy of the website programmers.

Causality. Our treatment of *for* in Section 2.1 is motivated by the counterfactual definition of *causality*. This definition requires that for an action to cause an effect that both the effect actually occurs and that the effect might not have occurred if the action did not occur. For example, Mackie defines a *cause* to be insufficient and non-redundant parts of unnecessary but sufficient causes (INUS conditions) for an effect [Mac74]. Mackie models causes and effects as facts. Working with sets of causes, this means that a fact c is a cause of an effect e if there exists a set C such that C is sufficient to entail e (sufficiency) and no subset of C is sufficient to entail e (non-redundancy).

We borrow the notion of *non-redundancy* from Mackie's definition of causality. Roughly speaking, we replace the causes with actions and the effect with a purpose. The extension to our semantics proposed in Section 6.3, may be seen as another instance of non-redundancy. This time, we replace the causes with purposes and the effect with an action. This suggests that for an action to be for a purpose, we expect both that the action was non-redundant for improving that purpose and that the purpose was non-redundant in motivating the action. That is, we expect planning to be parsimonious. We then use counterfactual reasoning to determine which purposes motivate the action.

Experimental Philosophy. Experimental philosophy has found some inconsistencies in how people tend to use the word "intent" called the *Knobe effect* [Kno03]. When it comes to benefits for purposes that are good, people tend to only say that the actor intended for the benefits if the actor selected his action taking the purpose into consideration, which agrees with our model. However, when it comes to bad purposes, people tend to say that the actor intended for the (bad) benefits even if the actor did not select his action with the goal of achieving the bad purpose in mind, which disagrees with our model. (See [Fel08] for a survey.)

Human Planning. Psychological studies have produced models of human thought (see, e.g., [ABB$^+$04]). However, these are too low-level and incomplete for our needs [DKP96]. The GOMS (Goals, Operators, Methods, and Selection rules) formalism provides a higher level model, but is limited to selecting behavior using simple planning approaches [CMN83, JK96]. Simon's approach of *bounded rationality* [Sim55] and related heuristic-based approaches [GS02] model more complex planning, but with less precise predictions.

7.4 Related Algorithms

Plan Recognition. Attempting to infer the plan that an agent has while performing an action is *plan recognition* [SSG78]. Plan recognition may predict the future actions of agents allowing systems to anticipate them. Often, plan recognition algorithms model how "low-level" actions contribute to achieving a "top-level" action that is done for its own sake (see, e.g., [KA86]). These top-level actions are similar to purposes. However, our auditing algorithm checks whether a sequence of actions is consistent with a given purpose rather than attempting to predict the most likely purpose or plan motivating the actions.

The work billed as plan recognition most closely related to our work is by Ramírez and Geffner [RG09, RG10, RG11]. They approach the problem by first attempting to determine what goal state the agent is attempting to reach rather than attempting to recognize the agent's plan directly. Thus, this has more in common with work on goal inference than standard approaches to plan recognition and we discuss it with the other works on goal inference.

Most work on plan recognition assumes that the agent is not attempting to mislead the plan recognizer since they are designed to aid cooperation with the agent. Our work is related to work on *adversarial* plan detection [AFFH86].

Particularly related is the work of Geib and Goldman, who use adversarial plan recognition to aid intrusion detection [GG01]. Similar to standard works, they model plans as a graph that represents a space of possible plans. Nodes of the graph represent actions and directed edges represent the order in which the adversary must perform the actions. Intrusions are paths in the graph from an initial node to a goal node. However, unlike most work on plan recognition, owing to the hostile nature of the actor, they do not assume that all relevant actions are observable. Thus, rather than simply comparing the observed actions to paths in the graph to determine possible plans, their recognition algorithm also considers unobserved actions consistent with the state of the system that the adversary might have performed.

Relatedly, Cuppens, Autrel, Miège, and Benferhat attempt to recognize malicious intentions for intrusion detection [CAMB02]. They model attacks as consisting of multiple actions each with pre-conditions and post-conditions. An adversary attempts to perform a *malicious* action by first performing all the *suspicious* actions needed to enable the pre-condition of the malicious action. Their approach is to observe these suspicious actions and predict from their model what other actions the adversary might have performed or will be performing. In particular, they try to predict which (if any) malicious action the adversary is attempting to perform using a shortest path heuristic. The distinction between suspicious and malicious actions does not apply to our work since we consider purposes, not actions, to be malicious. Indeed, in our setting many actions, such as looking up a medical record, could be either acceptable or malicious depending upon the context.

The models of planning used in both of these works differ from ours in two ways. First, we model purposes quantitatively instead of qualitatively. Second, our work considers probabilistic effects of the environment that might cause the agent to fail to achieve its plan.

Goal Inference. Works on goal inference attempt to determine from observations of an agent's actions what goal (typically, a goal state) the agent is attempting to reach. Identifying purposes with goals, our work uses goal inference to enforce purpose restrictions.

The goal inference works most closely related to ours are those using planning models similar to ours to compare the agent's actions to the actions that the agent would plan to perform given various possible goals. Early work by Rao, Shon, and Meltzoff motivate such reasoning by comparing such a process using a Bayesian graphical model to models of infant imitation [RSM04]. Shon, Grimes, Baker, and Rao consider an algorithm in such a setting [SGBR04]. While the Bayesian graphical models used in these works can represent partial observations, like POMDPs, they presuppose a fixed upper-bound on the number of actions the agent can perform.

Verma and Rao consider alternative algorithms and extend the model to include a distinguished action $\mathsf{StayPut}$ that is similar to our distinguished action stop [VR05, VR06]. Unlike our irrevocable action stop, an agent may perform $\mathsf{StayPut}$ before other actions. However, they presume a prior probability distribution (i.e., prior to observing the actions the agent takes) over actions that makes $\mathsf{StayPut}$ very unlikely before the agent reaches the goal state and equally likely to other actions afterwards. While this presumption is sufficient for their goal of ensuring that the agent takes the shortest path to the goal state (with high probability), it does not satisfy our stronger goal that the agent also only performs some no-op action once the goal is reached (or, under our model, once no further reward is possible).

The work of Baker, Saxe, and Tenenbaum focuses on the planning aspect of goal inference and attempt to ensure that they use a model of planning simple enough to correspond to human planning [BTS06]. They extend their model to an MDP model similar to ours [BTS07, BST09]. Under these MDP models, rather than having a reward function, the agent attempts to reduce the *costs* of reaching a *goal* state. For each possible goal state, their algorithms use the degree to which the agent's actions minimizes the costs of reaching the goal state to assign a probability to that goal state being the one pursued by the agent. Our reward functions are similar to the negation of their cost functions, but these works predict which goal state the agent is pursuing rather than which cost function it is using. Furthermore, rather than determine whether an action is optimal for a purpose, they assign a probability to the action that is proportional to how close to optimal it is. To validate their models, they performed experiments showing participants animated characters moving around a scene with various possible goal destinations. They compared the goals the participants assigned the animated characters to the most probable goal produced by their models.

Baker, Saxe, and Tenenbaum generalize their methods for MDPs to POMDPs and perform similar experiments to validate the generalized model [BST11]. After performing goal inference using the Strips model of planning [RG09, RG10], Ramírez and Geffner also adopts a POMDP model [RG11]. For the model of Ramírez and Geffner, the POMDP model uses costs, goal states, and a relaxed sense of optimality similar to MDP models of Baker et al. The POMDP model of Baker et al. is more closely related to our models. This model uses a reward function on states and actions like ours. Like their previous works, they do not require that an action be optimal to be for a purpose and assign a probability to an action based on how close to optimal it is. However, rather setting the probability of an action to be proportional to the quality of the action (as computed by Q^*), they set it to an approximation of the quality (Hauskrecht's look-ahead approximation Q^{LH} [Hau00]). Our POMDP algorithm for auditing is similar to their algorithm. However, to maintain soundness, our algorithm accounts for the error of approximate POMDP solving. Furthermore, their algorithms may assign a non-zero probability to a goal (or purpose) even if the agent's actions are inconsistent with pursuing that goal under our strict definition. Lastly, they do not consider non-redundancy nor information use.

Baker et al. formalizes a simple form of sequential planning for an MDP model with multiple goals that is similar to the sequential planning discussed in Section 6.2 [BTS07, BST09]. However, they do not support intentions from previous goals affecting future goals. Blokpoel, Kwisthout, van der Weide, and Rooij examine extending this model to handle simultaneous purposes, which we discuss in Section 6.3, but do not consider intentions [BKvdWvR10]. Ullman, Baker, Macindoe, Evans, Goodman, and Tenenbaum extend this line of work for handling social interactions (see, e.g., [UBM$^+$09]).

Also related is the work of Mao and Gratch [MG04]. While it differs from our work in the same ways as the work of Baker et al., it also differs in that rewards track how much the agent wants to achieve the goal rather than the degree of satisfaction of the goal.

Automated Planning. *Decision-theoretic planning* is planning to optimize some criteria, such as a purpose. (Blythe provides a survey [Bly99].) Optimizing MDPs or POMDPs to create plans are just two instances of decision-theoretic planning. Other instances may be more accurate, convenient, or general models of human planning.

For example, due to uncertainty the auditor may have about the model used by the auditee, we are interested in environment models that are like MDPs but without fixed probabilities assigned to transitions. Discrete-time Markov chains without fixed probabilities are known as *interval-valued discrete-time Markov* *chains* (IDTMCs). The form of IDTMC most similar to our model is the Uncertain Markov Chain (UMC) model [JL91]. We hope the algorithm of Sen et al. [SVA06] for a model checking problem related to UMCs may shed light on how to generalize our algorithm found in Section 2.3.

The POMDP model presented in Chapter 3 provides a model for planning as the auditee over time gains a better understanding of the environment. While the POMDP model allows the auditee's beliefs about its current state to change over time, the model itself is static. Bethke, Bertuccelli, and How present an adaptive MDP model that changes over time based on the auditee's current understanding of the model [BBH08]. We do not consider such adaptive models.

Chapter 8

Conclusions and Future Work

8.1 Conclusion

We use planning to create the first formal semantics for determining when a sequence of actions or information use is allowed under a purpose restriction. In particular, our formalism uses models similar to MDPs and POMDPs for planning, which allows us to automate auditing for both exclusivity and prohibitive purpose restrictions (Chapters 2 and 3). We have provided an auditing algorithms and an implementation base on our formalism (Sections 2.3, 2.4, and 3.5).

We validate that our approach based on planning accurately captures the meaning of purpose restrictions with numerous intuitive examples (Sections 2.1.3, 2.2.2, 2.2.3, 3.3.3, and 3.3.2 and Chapter 4) and an empirical study of how people understand the word "purpose" in the context of privacy policy enforcement (Chapter 5).

We apply our formalism to understand the ramifications of Regional Health Information Organizations (Chapter 4). Furthermore, we use our formalism to explain and compare previous methods of policy enforcement in terms of a formal semantics (Section 7.1). Our formalism highlights that an action can be for a purpose even if that purpose is never achieved, a point present in philosophical works on the subject (e.g., [Tay66]), but whose ramifications on policy enforcement had been unexplored.

These contributions lead us to conclude our thesis:

> A model of planning underlies a formalization of purpose restrictions that enables their automated enforcement.

However, we recognize the limitations of our formalism: it imperfectly models human planning and only captures some forms of planning for multiple purposes (Chapter 6). Nevertheless, we believe the essence of our work is correct: an action is for a purpose if the actor selects to perform that action while planning for the purpose. Future work will instantiate our semantic framework with more complete models of human planning.

8.2 Future Work

Future work may improvement the accuracy, practicality, or generality of our formalism. Unfortunately, these three directions may not lead to the same place: improving the accuracy or generality of the formalism may make it harder to use in practice. Future work may also apply our formalism. We consider these four areas in turn.

8.2.1 Improving Accuracy

While Chapter 5 presented a survey showing that a relation exists between planning and purpose restrictions, more studies will clarify this relation. We would like to study whether exclusivity rules require the agent to actually follow a plan for the allowed purpose (Hypothesis H2) or whether the agent just needs to perform actions consistent with such a plan (Hypothesis H3). We would also like know whether people consider an incompetent agent in violation of a purpose restriction when the agent thinks it is performing actions consistent with a plan for a purpose but is mistaken. As discussed in Section 6.4, a better model of human planning may improve the accuracy of our predictions and allow us to generalize our formalism to multiple purposes.

8.2.2 Furthering Practicality

Under our formulation of purpose restrictions, to check whether a system obeys a policy, an auditor must not only model the system and policy, but also how the system *could* have behaved with an environment model. Auditors can often automatically extract system models from source code (see, e.g., [CGP00]) or create system models from descriptions of operating procedures. However, in many cases, the auditor will have to create the environment model by hand after researching all the possible behaviors. Given the difficulty of this task, we desire methods of finding policy violations that do not require a full environment model even if they are only approximately correct.

Creating environment models can be difficult even if the auditor has an intuitive idea of its form. Thus, a tool that describes the possible environment models consistent with a policy and a specification of the system would be useful. For an auditor to find such a tool useful, he must be able to understand the tool's output. An interactive query engine might aid the auditor in understanding the results.

Often systems, such as those storing electronic medical records, have no model but produce logs detailing their use. For such systems, the auditor may be able to learn the environment model from the logs. Previous work on reinforcement learning, such as Q-learning [Wat89], optimizes MDPs using observations of their behavior, often available from logs, instead of the MDP model itself. Similarly, SARSA can allow POMDP optimization without the model itself [RN94]. Bauer constructs libraries of possible plans from observed sequences of actions [Bau98]. *Process mining* attempts to create a model of the processes used at an organization from logs [WvdA01].

We hope these techniques may aid future research on auditing without a pre-created environment model. We envision a system using Experience-Based Access Management (EBAM), which attempts to iteratively construct a model for access control as a system is operating [GLM11]. Over time, EBAM refines a model based upon an auditor identifying results that are false positives or false negatives. We see our work fitting into this framework by providing a formalism for the model with a semantics that allows the system to generalize from its experiences. This approach would be similar to how Zhang et al. use Role-Based Access Control as a formalism for EBAM models [ZGL$^+$11].

8.2.3 Generalizations

Interactions of Multiple Purposes and Agents. Most pressingly, we would like to address the interactions among multiple purposes discussed in Chapter 6.

We would also like to study how the definition extends to interactions among multiple agents. By employing MDPs, we implicitly limited the agent's ability to reason about how other agents may behave. To allow strategic reasoning, we would have to adopt game theoretic models in the place of MDPs (see, e.g., [OR94]). Work on goal inference during social interactions could also prove relevant (see, e.g., [UBM$^+$09]).

Interaction with Obligations. Many privacy policies contain *obligations*: requirements imposed upon the system as the result of performing some action (see, e.g., [DFK07]). For example, a company may be required to delete a credit card number six months after completing a transaction. Designing a system to obey obligations is complicated since the system must discharge some obligations, such as the one above, after the use that triggered them. Thus, systems may benefit from automated methods for enforcing obligations.

Obligations may have a non-trivial interaction with purposes. For example, consider a policy that requires that funds only be used to purchase books. Suppose that an employee purchases a book with a credit card and later uses funds to pay the credit card bill. Many would argue that the employee obeyed the policy despite not strictly using the funds to purchase the book since paying the bill was an obligation incurred for the purpose of purchasing the book. We would like our semantics to explain such circumstances.

Context-Dependent Purposes. In some cases a purpose restriction might only indirectly refer to a purpose. For example, the policy of the website of the U.S. Social Security Administration states [U.S10b]:

> By providing your personal information, you give us consent to use the information only for the purpose for which it was collected. We describe those purposes when we collect information.

Enforcing such policies may pose difficulties for the auditor since the policy refers to a purpose not by name but rather by the context in which the website collected the information. While the auditor may use our semantics and algorithms after determining the appropriate purposes from the context in which the website collected the information, we desire methods to make this task easier.

More burdensome still, in some cases, the auditor might not have the domain expertise to specify the meaning of a purpose even after determining it. For example, the Facebook policy governing Facebook applications states, "You will only request the data you need to operate your application." and "A user's friends' data can only be used in the context of the user's experience on your application." [Fac12]. While these are not explicitly purpose restrictions, the auditor may view them as implying a purpose restriction: *An application will collect and use Facebook user data only for the purpose that the application serves.* If Facebook were to attempt to enforce this purpose restriction, it would need to determine the purpose of the application, which might be unclear to the auditor working for Facebook. Facebook could require the application developer to provide the purpose and even the environment model to the auditor. In this case, the auditor may use our methodology to audit the application under the provided model. However, the auditor might still not have enough information to judge the accuracy or appropriateness of the developer's representations. We consider developing methods of enforcing purpose restrictions that reference context-dependent purposes, such as this one, an interesting direction for future work.

8.2.4 Applications

We would like to conduct a case study to determine the applicability of our semantics to a real system. Possible systems to study is the admissions process for graduate school or the privacy practices of a hospital. This exercise will aid us in understanding the process of creating complex environment models.

Our approach to auditing considered the policy and environment to be a fixed entities passively modeled. However, our formalism can shed light on how organizations may design their systems to comply with policies or create enforceable policies. Since the agent will attempt to maximize his personal benefits, a system designer must assure that these align with the purposes that the system would like to further. Mechanisms such as punishments, which decrease personal benefits for disallowed actions, can help align these benefits. Mechanisms forcing an auditee to declare its strategy can make violations easier to detect.

The idea of *minimum necessary* shows up in HIPAA [U.S10a]:

> The Privacy Rule generally requires covered entities to take reasonable steps to limit the use or disclosure of, and requests for, protected health information to the minimum necessary to accomplish the intended purpose.

We would like to apply our formalism to minimum necessary use, disclosure, and requests. We could than compare an approach based on our formalism of purpose to previous work on minimum disclosure [MMZ06, BMDS07]. In particular, we like to study how to represent one system using more information than another with the equivalence relations $\equiv$ on information used in Chapter 3. We are also interested in formalizing the idea of “accomplishing” a qualitative purpose that can always be further satisfied, such as marketing or profit.

8.3 Perspective

Fundamentally, our work shows the difficulties of enforcing purpose restrictions due to issues such as the tenable deniability of ulterior motives (Sections 2.2.2 and 2.2.3). These difficulties justify policies proscribing conflicts of interest and requiring the separation of duties despite possibly causing inefficiencies. For example, many hospitals would err on the side of caution and disallow a referral from a physician to his own private practice or require a second opinion to do so, thereby restraining the ulterior motive of profit. Indeed, despite the intuition that *privacy is security with a purpose*, due to these difficulties, purpose possibly plays the role of guidance in crafting more operational internal policies that organizations enforce rather than the role of a direct input to the formal auditing process itself. In light of this possibility, one may view our work as a way to judge the quality of these operational policies relevant to the intent of the purpose restrictions found in the actual privacy policy.

However, we do view purpose restrictions as important to privacy policies. We believe that privacy policies are best thought of as a balancing act between the security of an *information subject* and the utility of an *information holder*. While abuse of the information in question harms the information subject, the information holder controls use of the information. Traditional data security typically envisions an entity that holds information about itself. As the information holder is one and the same as the information subject, the entity can internally balance its competing concerns of wanting to make use of the information while also protecting it.

In the case of privacy, on the other hand, the information holder is not the information subject. For example, while a hospital stores a medical record and is in charge of protecting it, it is the patient that bears the direct harm if that record becomes public. Indeed, the hospital might even stand to profit from sharing the record with entities (such as insurance companies) that the patient my deem adversarial.

Thus, we view privacy policies as a balancing act between the information subject and the information holder. The role of privacy policies and laws requiring privacy precautions (such as HIPAA) is to strike a balance between these competing concerns. Purpose plays a role by restricting the actions for which the information holder may use the information to only those where the benefit has been deemed by the policy maker as commensurate with the risks to the information subject.

That purpose restrictions do not actually reference this balancing act limits their ability to provide privacy. Indeed, under our formalism, an action could be for a purpose even if that action is part of a plan that barely improves the purpose while having grave privacy implications. In light of this, we believe that policies should consider the actual balancing of competing concerns. Unfortunately, whereas we can easily express purpose restrictions, the trade-offs between two different goals, such as treatment and patient privacy, are often difficult to compare. For this reason, we expect purpose restrictions to continue to a play a key role in privacy policies.

Bibliography

[ABB$^+$04] John R. Anderson, Daniel Bothell, Michael D. Byrne, Scott Douglass, Christian Lebiere, and Yulin Qin. An integrated theory of the mind. *Psychological Review*, 111:1036–1060, 2004.

[AF07] Sabah S. Al-Fedaghi. Beyond purpose-based privacy access control. In *ADC '07: Proceedings of the Eighteenth Australasian Database Conference*, pages 23–32. Australian Computer Society, Darlinghurst, Australia, 2007. Presented in Ballarat, Victoria, Australia.

[AFFH86] Jerome Azarewicz, Glenn Fala, Ralph Fink, and Christof Heithecker. Plan recognition for airborne tactical decision making. In *National Conference on Artificial Intelligence*, pages 805–811. 1986.

[Ain92] George Ainslie. *Picoeconomics: The Interaction of Successive Motivational States within the Person*. Studies in Rationality and Social Change. Cambridge University Press, 1992.

[AKSX02] Rakesh Agrawal, Jerry Kiernan, Ramakrishnan Srikant, and Yirong Xu. Hippocratic databases. In *VLDB '02: Proceedings of the 28th International Conference on Very Large Data Bases*, pages 143–154. VLDB Endowment, 2002. Presented in Hong Kong, China.

[Ans57] G.E.M. Anscombe. *Intention*. Harvard University Press, Cambridge, MA, USA, 1957.

[BA05] Travis D. Breaux and Annie I. Antón. Analyzing goal semantics for rights, permissions, and obligations. In *RE '05: Proceedings of the 13th IEEE International Conference on Requirements Engineering*, pages 177–188. IEEE Computer Society, Washington, DC, USA, 2005.

[BA08] Travis D. Breaux and Annie I. Antón. Analyzing regulatory rules for privacy and security requirements. *IEEE Trans. Softw. Eng.*, 34(1):5–20, 2008.

[Ban05] Bank of America Corporation. Bank of America privacy policy for consumers, September 2005. Accessed on February 4, 2011, at `http://www.bankofamerica.com/privacy/pdf/eng-boa.pdf`.

[Bau98] Mathias Bauer. Acquisition of abstract plan descriptions for plan recognition. In *Proceedings of the fifteenth national conference on Artificial intelligence*, pages 936–941. American Association for Artificial Intelligence, Menlo Park, CA, USA, 1998. Presented at AAAI '98, Madison, Wisconsin, United States.

[BBH08] Brett Bethke, Luca F. Bertuccelli, and Jonathan P. How. Experimental demonstration of adaptive mdp-based planning with model uncertainty. In *Guidance, Navigation, and Control Conference*. American Institute of Aeronautics and Astronautics, 2008.

[BBL05] Ji-Won Byun, Elisa Bertino, and Ninghui Li. Purpose based access control of complex data for privacy protection. In *SACMAT '05: Proceedings of the Tenth ACM Symposium on Access Control Models and Technologies*, pages 102–110. ACM, New York, NY, USA, 2005. Presented in Stockholm, Sweden.

[Bel52] Richard Bellman. On the theory of dynamic programming. *Proceedings of the National Academy of Sciences*, 38:716–719, 1952.

[BHL11] Adam J. Berinsky, Gregory A. Huber, and Gabriel S. Lenz. Using Mechanical Turk as a subject recruitment tool for experimental research. Submitted for review, June 2011.

[BKK06] Carolyn A. Brodie, Clare-Marie Karat, and John Karat. An empirical study of natural language parsing of privacy policy rules using the SPARCLE policy workbench. In *SOUPS '06: Proceedings of the Second Symposium on Usable Privacy and Security*, pages 8–19. ACM, New York, NY, USA, 2006. Presented in Pittsburgh, Pennsylvania.

[BKKF05] Carolyn Brodie, Clare-Marie Karat, John Karat, and Jinjuan Feng. Usable security and privacy: a case study of developing privacy management tools. In *SOUPS '05: Proceedings of the 2005 Symposium on Usable Privacy and Security*, pages 35–43. ACM, New York, NY, USA, 2005. Presented in Pittsburgh, Pennsylvania.

[BKvdWvR10] Mark Blokpoel, Johan Kwisthout, Theo P. van der Weide, and Iris van Rooij. How action understanding can be rational, Bayesian and tractable. In S. Ohlsson and R. Catrambone, editors, *Proceedings of the 32nd Annual Conference of the Cognitive Science Society*, pages 1643–1648. Cognitive Science Society, Austin, TX, USA, 2010.

[BL08] Ji-Won Byun and Ninghui Li. Purpose based access control for privacy protection in relational database systems. *The VLDB Journal*, 17(4):603–619, 2008.

[Bly99] Jim Blythe. Decision-theoretic planning. *AI Magazine*, 20(2):37–54, 1999.

[BMDS07] Adam Barth, John Mitchell, Anupam Datta, and Sharada Sundaram. Privacy and utility in business processes. In *CSF '07: Proceedings of the 20th IEEE Computer Security Foundations Symposium*, pages 279–294. IEEE Computer Society, Washington, DC, USA, 2007.

[Bra87] Michael E. Bratman. *Intention, Plans, and Practical Reason*. Harvard University Press, Cambridge, MA, USA, 1987.

[BST09] Chris L. Baker, Rebecca Saxe, and Joshua B. Tenenbaum. Action understanding as inverse planning. *Cognition*, 113(3):329–349, 2009.

[BST11] Chris L. Baker, Rebecca R. Saxe, and Josh B. Tenenbaum. Bayesian theory of mind: Modeling joint belief-desire attribution. In *Proceedings of the Thirty-Third Annual Conference of the Cognitive Science Society*, pages 2469–2474. Cognitive Science Society, Austin, TX, USA, 2011.

[BTS06] Chris L. Baker, Josh B. Tenenbaum, and Rebecca R. Saxe. Bayesian models of human action understanding. In Y. Weiss, B. Schölkopf, and J. Platt, editors, *Advances in Neural Information Processing Systems 18*, pages 99–106. MIT Press, Cambridge, MA, 2006. Presented at NIPS 2005.

[BTS07] Chris L. Baker, Josh B. Tenenbaum, and Rebecca R. Saxe. Goal inference as inverse planning. In *Proceedings of the Twenty-Ninth Annual Conference of the Cognitive Science Society*, pages 779–784. Cognitive Science Society, Austin, TX, USA, 2007.

[CAMB02] Frédéric Cuppens, Fabien Autrel, Alexandre Miège, and Salem Benferhat. Recognizing malicious intention in an intrusion detection process. In Ajith Abraham, Javier Ruiz del Solar, and Mario Köppen, editors, *Second International Conference on Hybrid Intelligent Systems*, volume 87 of *Frontiers in Artificial Intelligence and Applications*, pages 806–817. IOS Press, 2002.

[Cer] Cerner. Security solution for a healthcare network: P2Sentinel powered by SenSage. Flyer. Accessed on May 5, 2012, at `http://www.cerner.com/uploadedFiles/P2Sentinel_flyer.pdf`.

[CGP00] Edmund M. Clarke, Orna Grumberg, and Doron A. Peled. *Model Checking*. MIT Press, 2000.

[CL90] Philip R. Cohen and Hector J. Levesque. Intention is choice with commitment. *Artif. Intell.*, 42:213–261, March 1990.

[CLM$^+$02] Lorrie Cranor, Marc Langheinrich, Massimo Marchiori, Martin Presler-Marshall, and Joseph Reagle. The Platform for Privacy Preferences 1.0 (P3P1.0) specification. W3C Recommendation, April 2002.

[CMN83] Stuart Card, Thomas P. Moran, and Allen Newell. *The Psychology of Human Computer Interaction*. Lawrence Erlbaum Associates, 1983.

[CMVZ09] Stephen Chong, Andrew C. Myers, K. Vikram, and Lantian Zheng. *Jif Reference Manual*, February 2009. Accessed on May 05, 2012, at `http://www.cs.cornell.edu/jif/doc/jif-3.3.0/manual.html`.

[Cra02] Lorrie Faith Cranor. *Web Privacy with P3P*. O'Reilly, 2002.

[d'E63] F. d'Epenoux. A probabilistic production and inventory problem. *Management Science*, 10(1):98–108, October 1963.

[Den87] Daniel C. Dennett. *The Intentional Stance*. MIT Press, 1987.

[DFK07] Daniel J. Dougherty, Kathi Fisler, and Shriram Krishnamurthi. Obligations and their interaction with programs. In J. Biskup and J. Lopez, editors, *Computer Security – ESORICS 2007*, volume 4734 of *Lecture Notes in Computer Science*, pages 375–389. Springer-Verlag Berlin Heidelberg, 2007.

[DKP96] Jagannath Prasad Das, Binod C. Kar, and Rauno K. Parrila. *Cognitive Planning: The Psychological Basis of Intelligent Behavior*. Sage, 1996.

[EKWB11] Md. Enamul Kabir, Hua Wang, and Elisa Bertino. A conditional purpose-based access control model with dynamic roles. *Expert Syst. Appl.*, 38:1482–1489, March 2011.

[Fac12] Facebook. Facebook platform policies, March 2012. Accessed on April 13, 2011, at `http://developers.facebook.com/policy/`.

[Fai] FairWarning. FairWarning: Privacy breach detection for healthcare. Accessed on February 7, 2011, at `http://fairwarningaudit.com/`.

[Fel08] Adam Feltz. The knobe effect: A brief overview. *Journal of Mind and Behavior*, 28:265–278, 2008.

[FHMV95] Ronald Fagin, Joseph Y. Halpern, Yoram Moses, and Moshe Y. Vardi. *Reasoning About Knowledge*. MIT Press, 1995.

[FJ02] Csilla Farkas and Sushil Jajodia. The inference problem: a survey. *SIGKDD Explor. Newsl.*, 4:6–11, December 2002.

[FOW87] Jeanne Ferrante, Karl J. Ottenstein, and Joe D. Warren. The program dependence graph and its use in optimization. *ACM Trans. Program. Lang. Syst.*, 9(3):319–349, 1987.

[FP10] Matthew Flatt and PLT. Reference: Racket. Technical Report PLT-TR-2010-1, PLT Inc., 2010. Accessed on May 5, 2012, at `http://racket-lang.org/tr1/`. Version 5.2.1, revised Feb. 2, 2012.

[GG01] Christopher W. Geib and Robert P. Goldman. Plan recognition in intrusion detection systems. In *DARPA Information Survivability Conference and Exposition*. 2001. Presented at DISCEX.

[Gig02] Gerd Gigerenzer. The adaptive toolbox. In Gerd Gigerenzer and Reinhard Selten, editors, *Bounded Rationality: The Adaptive Toolbox*, Dahlem Workshop Reports, pages 37–50. MIT Press, 2002.

[GLM11] Carl A. Gunter, David M. Liebovitz, and Bradley Malin. Experience-based access management: A life-cycle framework for identity and access management systems. *IEEE Security and Privacy*, 9:48–55, 2011.

[GM82] Joseph A. Goguen and Jose Meseguer. Security policies and security models. In *Proceedings of the IEEE Symposium on Security and Privacy*, pages 11–20. 1982.

[GS02] Gerd Gigerenzer and Reinhard Selten, editors. *Bounded Rationality: The Adaptive Toolbox*. Dahlem Workshop Reports. MIT Press, 2002.

[HA05] Katia Hayati and Martín Abadi. Language-based enforcement of privacy policies. In *PET 2004: Workshop on Privacy Enhancing Technologies*, pages 302–313. Springer-Verlag, 2005.

[Hau00] Milos Hauskrecht. Value-function approximations for partially observable markov decision processes. *J. Artif. Int. Res.*, 13(1):33–94, August 2000.

[HBLK$^+$07] Chris Hanson, Tim Berners-Lee, Lalana Kagal, Gerald Jay Sussman, and Daniel Weitzner. Data-purpose algebra: Modeling data usage policies. In *POLICY '07: Proceedings of the Eighth IEEE International Workshop on Policies for Distributed Systems and Networks*, pages 173–177. IEEE Computer Society, Washington, DC, USA, 2007. Presented in Bologna, Italy.

[Jay03] E.T. Jaynes. *Probability Theory: The Logic of Science*. Cambridge University Press, Cambridge, UK, 2003.

[JK96] Bonnie E. John and David E. Kieras. The GOMS family of user interface analysis techniques: comparison and contrast. *ACM Trans. Comput.-Hum. Interact.*, 3:320–351, December 1996.

[JL91] B. Jonsson and K. G. Larsen. Specification and refinement of probabilistic processes. In *Proceedings of Sixth Annual IEEE Symposium on Logic in Computer Science*, LICS, pages 266–277. IEEE Press, July 1991.

[JSNS09] Mohammad Jafari, Reihaneh Safavi-Naini, and Nicholas Paul Sheppard. Enforcing purpose of use via workflows. In *WPES '09: Proceedings of the 8th ACM Workshop on Privacy in the Electronic Society*, pages 113–116. ACM, New York, NY, USA, 2009. Presented in Chicago, Illinois, USA.

[KA86] Henry A. Kautz and James F. Allen. Generalized plan recognition. In *Proceedings of the Fifth National Conference on Artificial Intelligence*, pages 32–37. AAAI, 1986.

[Kar84] N. Karmarkar. A new polynomial-time algorithm for linear programming. In *STOC '84: Proceedings of the Sixteenth Annual ACM Symposium on Theory of Computing*, pages 302–311. ACM, New York, NY, USA, 1984.

[KCS08] Aniket Kittur, Ed H. Chi, and Bongwon Suh. Crowdsourcing user studies with Mechanical Turk. In *Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, pages 453–456. ACM, New York, NY, USA, 2008. Presented at CHI '08, Florence, Italy.

[Kha79] L. G. Khachian. A polynomial algorithm in linear programming. *Dokl. Akad. Nauk SSSR*, 244:1093–1096, 1979. English translation in Soviet Math. Dokl. 20, 191-194, 1979.

[KHL08] Hanna Kurniawati, David Hsu, and Wee Sun Lee. SARSOP: Efficient point-based POMDP planning by approximating optimally reachable belief spaces. In *Proc. Robotics: Science and Systems*. 2008.

[KLC98] Leslie Pack Kaelbling, Michael L. Littman, and Anthony R. Cassandra. Planning and acting in partially observable stochastic domains. *Artif. Intell.*, 101:99–134, May 1998.

[Kno03] J. Knobe. Intentional action and side effects in ordinary language. *Analysis*, 63:190–193, 2003.

[KSNW09] Ram Krishnan, Ravi Sandhu, Jianwei Niu, and William H. Winsborough. A conceptual framework for group-centric secure information sharing. In *ASIACCS '09: Proceedings of the 4th International Symposium on Information, Computer, and Communications Security*, pages 384–387. ACM, New York, NY, USA, 2009. Presented in Sydney, Australia.

[LDK95] Michael L. Littman, Thomas L. Dean, and Leslie P. Kaelbling. On the complexity of solving Markov decision problems. In *Proceedings of the Eleventh Annual Conference on Uncertainty in Artificial Intelligence*, pages 394–402. Morgan Kaufmann, 1995. Presented at UAI 95, Montreal, Québec, Canada.

[Mac74] John L. Mackie. *The Cement of the Universe: A Study of Causation*. Oxford University Press, 1974.

[Mad00] Omid Madani. *Complexity Results for Infinite-Horizon Markov Decision Processes*. PhD thesis, University of Washington, 2000.

[McN47] Quinn McNemar. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12:153–157, 1947.

[MF03] Gustavo H.M.B. Motta and Sergio S. Furuie. A contextual role-based access control authorization model for electronic patient record. *Information Technology in Biomedicine, IEEE Transactions on*, 7(3):202–207, September 2003.

[MG04] Wenji Mao and Jonathan Gratch. A utility-based approach to intention recognition. In *AAMAS 2004 Workshop on Agent Tracking: Modeling Other Agents from Observations*. July 2004.

[MMZ06] Fabio Massacci, John Mylopoulos, and Nicola Zannone. Hierarchical hippocratic databases with minimal disclosure for virtual organizations. *The VLDB Journal*, 15(4):370–387, 2006.

[Mon82] George E. Monahan. A survey of partially observable Markov decision processes: Theory, models, and algorithms. *Management Science*, 28(1):pp. 1–16, 1982.

[NBL$^+$10] Qun Ni, Elisa Bertino, Jorge Lobo, Carolyn Brodie, Clare-Marie Karat, John Karat, and Alberto Trombetta. Privacy-aware role-based access control. *ACM Trans. Inf. Syst. Secur.*, 13:24:1–24:31, July 2010.

[Off03] Office for Civil Rights. Summary of the HIPAA privacy rule. OCR Privacy Brief, U.S. Department of Health and Human Services, 2003.

[OR94] Martin J. Osborne and Ariel Rubinstein. *A Course in Game Theory*. MIT Press, Cambridge, MA, USA, 1994.

[Orn09] Martin T. Orne. Demand characteristics and the concept of quasi-controls. In Robert Rosenthal and Ralph L. Rosnow, editors, *Artifacts in Behavioral Research: Robert Rosenthal and Ralph L. Rosnow's Classic Books*, page 110. Oxford University Press, 2009.

[PBDD08] Mor Peleg, Dizza Beimel, Dov Dori, and Yaron Denekamp. Situation-based access control: Privacy management via modeling of patient data access scenarios. *J. of Biomedical Informatics*, 41(6):1028–1040, December 2008.

[PGY08] Huanchun Peng, Jun Gu, and Xiaojun Ye. Dynamic purpose-based access control. In *International Symposium on Parallel and Distributed Processing with Applications*, pages 695–700. IEEE Computer Society, Los Alamitos, CA, USA, 2008.

[PKK11] Pascal Poupart, Kee-Eung Kim, and Dongho Kim. Closing the gap: Improved bounds on optimal POMDP solutions. In Fahiem Bacchus, Carmel Domshlak, Stefan Edelkamp, and Malte Helmert, editors, *Proceedings of the International Conference on Automated Planning and Scheduling*. AAAI, 2011. Presented at ICAPS 2011.

[PS03] Calvin Powers and Matthias Schunter. Enterprise privacy authorization language (EPAL 1.2). W3C Member Submission, November 2003.

[PT87] Christos Papadimitriou and John N. Tsitsiklis. The complexity of Markov decision processes. *Math. Oper. Res.*, 12:441–450, August 1987.

[RG09] Miquel Ramírez and Hector Geffner. Plan recognition as planning. In *Proceedings of the 21st international joint conference on Artifical intelligence*, pages 1778–1783. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2009. Presented at IJCAI'09, Pasadena, California, USA.

[RG10] Miquel Ramírez and Hector Geffner. Probabilistic plan recognition using off-the-shelf classical planners. In Maria Fox and David Poole, editors, *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence*. AAAI Press, July 2010. Presented at AAAI 2010, Atlanta, Georgia, USA.

[RG11] Miquel Ramírez and Hector Geffner. Goal recognition over POMDPs: Inferring the intention of a POMDP agent. In Toby Walsh, editor, *Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, pages 2009–2014. IJCAI/AAAI, 2011. Presented at IJCAI 2011, Barcelona, Catalonia, Spain.

[RIS$^+$10] Joel Ross, Lilly Irani, M. Six Silberman, Andrew Zaldivar, and Bill Tomlinson. Who are the crowdworkers? Shifting demographics in Mechanical Turk. In *Proceedings of the 28th of the international conference extended abstracts on Human factors in computing systems*, pages 2863–2872. ACM, New York, NY, USA, 2010. Presented at CHI EA '10, Atlanta, Georgia, USA.

[RN94] G. A. Rummery and M. Niranjan. On-line Q-learning using connectionist systems. Technical Report CUEF/F-INFENG/TR 166, Cambridge University Engineering Department, 1994.

[RN03] Stuart J. Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Pearson Education, 2nd edition, 2003.

[Roy08] Olivier Roy. *Thinking before Acting: Intentions, Logic, Rational Choice*. PhD thesis, Institute for Logic, Language and Computation; Universiteit van Amsterdam, 2008.

[RSM04] Rajesh P. N. Rao, Aaron P. Shon, and Andrew N. Meltzoff. A bayesian model of imitation in infants and robots. In K. Dautenhahn and C. Nehaniv, editors, *Imitation and Social Learning in Robots, Humans, and Animals*, pages 217–247. Cambridge University Press, 2004.

[San96] Ravi S. Sandhu. Role hierarchies and constraints for lattice-based access controls. In *ESORICS '96: Proceedings of the 4th European Symposium on Research in Computer Security*, pages 65–79. Springer-Verlag, London, UK, 1996.

[Sel02] Reinhard Selten. What is bounded rationality? In Gerd Gigerenzer and Reinhard Selten, editors, *Bounded Rationality: The Adaptive Toolbox*, Dahlem Workshop Reports, pages 13–36. MIT Press, 2002.

[SGBR04] Aaron P. Shon, David B. Grimes, Chris L. Baker, and Rajesh P.N. Rao. A probabilistic framework for model-based imitation learning. In *Proceedings of the 26th Annual Meeting of the Cognitive Science Society*. Cognitive Science Society, Austin, TX, USA, 2004.

[Sim55] Herbert A. Simon. A behavioral model of rational choice. *Quarterly Journal of Economics*, 69:99–118, 1955.

[Son71] Edward Jay Sondik. *The optimal control of partially observable Markov processes*. PhD thesis, Stanford University, 1971.

[Son78] Edward J. Sondik. The optimal control of partially observable Markov processes over the infinite horizon: Discounted costs. *Operations Research*, 26(2):pp. 282–304, 1978.

[SS05] Trey Smith and Reid Simmons. Point-based POMDP algorithms: Improved analysis and implementation. In *Proc. of the Conference on Uncertainty in Artificial Intelligence*. July 2005.

[SSG78] C.F. Schmidt, N.S. Sridharan, and J.L. Goodson. The plan recognition problem: An intersection of psychology and artificial intelligence. *Artificial Intelligence*, 11(1-2):45 – 83, 1978.

[Sti61] George J. Stigler. The economics of information. *The Journal of Political Economy*, 69(3):213–225, 1961.

[SVA06] Koushik Sen, Mahesh Viswanathan, and Gul Agha. Model-checking markov chains in the presence of uncertainties. In *International Conference on Tools and Algorithms for the Construction and Analysis of Systems*, volume 3920 of *Lecture Notes in Computer Science*, pages 394–410. Springer-Verlag Berlin Heidelberg, 2006. Presented at TACAS.

[SW89] John A. Simpson and Edmund S. C. Weiner. purpose, n. In *The Oxford English Dictionary*. Oxford University Press, 2nd edition, 1989.

[SY86] R. E. Strom and S. Yemini. Typestate: A programming language concept for enhancing software reliability. *IEEE Trans. Softw. Eng.*, 12:157–171, January 1986.

[Tay66] Richard Taylor. *Action and Purpose*. Prentice-Hall, 1966.

[TDW11] Michael Carl Tschantz, Anupam Datta, and Jeannette M. Wing. On the semantics of purpose requirements in privacy policies. Technical Report CMU-CS-11-102, School of Computer Science, Carnegie Mellon University, February 2011. Also available at `http://arxiv.org/abs/1102.4326`.

[TDW12a] Michael Carl Tschantz, Anupam Datta, and Jeannette M. Wing. Formalizing and enforcing purpose restrictions in privacy policies. In *Proceedings of the IEEE Symposium on Security and Privacy*. IEEE Computer Society, Los Alamitos, CA, USA, 2012. To appear.

[TDW12b] Michael Carl Tschantz, Anupam Datta, and Jeannette M. Wing. Formalizing and enforcing purpose restrictions in privacy policies (full version). Technical Report CMU-CS-12-106, School of Computer Science, Carnegie Mellon University, March 2012.

[The95] The European Parliament and the Council of the European Union. Directive 95/46/EC. *Official Journal of the European Union*, L 281:31–50, November 1995.

[Tse90] Paul Tseng. Solving h-horizon stationary Markov decision process in time proportional to log(h). *Operations Research Letters*, 9(5):287–297, 1990.

[UBM$^+$09] Tomer D. Ullman, Chris L. Baker, Owen Macindoe, Owain Evans, Noah D. Goodman, and Joshua B. Tenenbaum. Help or hinder: Bayesian models of social goal inference. In Y. Bengio, D. Schuurmans, J. Lafferty, C. K. I. Williams, and A. Culotta, editors, *Advances in Neural Information Processing Systems 22*, pages 1874–1882. NIPS Foundation, 2009. Presented at NIPS 2009.

[Uni10] United States Congress. Financial services modernization act of 1999. Title 15, United States Code, Section 6802, February 2010.

[U.S10a] U.S. National Archives and Records Administration. Title 45 – public welfare. Code of Federal Regulations, October 2010. Sections 164.502(b) and 164.514(d). Commonly cited as 45 CFR 164.502(b) and 45 CFR 164.514(d).

[U.S10b] U.S. Social Security Administration. Internet privacy policy. Webpage., 2010. This privacy policy was available at `http://www.ssa.gov/privacy.html` in 2010, when we retrieved it, but changed by April 13, 2012. It is still archived at `http://web.archive.org/web/20110605201734/http://www.ssa.gov/privacy.html`.

[VR05] Deepak Verma and Rajesh Rao. Graphical models for planning and imitation. Technical Report 2005-02-01, Department of CSE, University of Washington, Seattle, WA, USA, February 2005.

[VR06] Deepak Verma and Rajesh P. N. Rao. Goal-based imitation as probabilistic inference over graphical models. In Y. Weiss, B. Schölkopf, and J. Platt, editors, *Advances in Neural Information Processing Systems 18*, pages 1393–1400. MIT Press, Cambridge, MA, 2006. Presented at NIPS 2005.

[Was03] Washington Radiology Associates, P.C. Notice of privacy practices, April 2003. Accessed on February 4, 2011, at `http://www.washingtonradiology.com/office-guide/privacy.asp`.

[Wat89] Christopher John Cornish Hellaby Watkins. *Learning from Delayed Rewards*. PhD thesis, Cambridge University, 1989.

[WB93] Ronald Williams and Leemon C. Baird. Tight performance bounds on greedy policies based on imperfect value functions. Technical Report NU-CCS-93-14, Northeastern University, November 1993.

[WB94] Ronald Williams and Leemon C. Baird. Tight performance bounds on greedy policies based on imperfect value functions. In *Proceedings of the Tenth Yale Workshop on Adaptive and Learning Systems*. Yale University, June 1994.

[WvdA01] A.J.M.M. Weijters and W.M.P van der Aalst. Process mining: Discovering workflow models from event-based data. In *Proceedings of the 13th Belgium-Netherlands Conference on Artificial Intelligence*, pages 283–290. 2001. Presented at BNAIC 2001.

[Yah10a] Yahoo! Privacy policy: Information collection and use, 2010. Accessed on May 5, 2012, at `http://info.yahoo.com/privacy/us/yahoo/details.html#2`.

[Yah10b] Yahoo! Privacy policy: Yahoo Mail, 2010. Accessed on May 5, 2012, at `http://info.yahoo.com/privacy/us/yahoo/mail/details.html`.

[ZGL$^+$11] Wen Zhang, Carl A. Gunter, David Liebovitz, Jian Tian, and Bradley Malin. Role prediction using electronic medical record system audits. In *AMIA 2011 Annual Symposium*, pages 858–867. American Medical Informatics Association, October 2011.

[ZH01] Rong Zhou and Eric A. Hansen. An improved grid-based approximation algorithm for POMDPs. In *Proceedings of the 17th International Joint Conference on Artificial Intelligence*, volume 1, pages 707–714. Morgan Kaufmann Publishers, San Francisco, CA, USA, 2001. Presented in Seattle, WA, USA.

Appendix A

Further Background on POMDPs

A.1 Details of the Belief MDPs

First, we reduce $\tau(\beta, a)(\beta')$ for $\mathsf{bmdp}(m)$ to an expression in terms of the features of the POMDP m. Doing so involves using Kronecker's delta: $\delta(x, y)$ is equal to 1 if $x = y$ and 0 otherwise. The reduction also uses the set $\Theta'_m(\beta, a)$ of observations that are possible under the belief state β after performing the action a: $\Theta'_m(\beta, a) = \{\, s \in \mathcal{O} \mid \Pr[O{=}o \mid B{=}\beta, A{=}a] \neq 0 \,\}$. The reduction is as follows:

$$
\begin{aligned}
\tau(\beta, a)(\beta') &= \Pr[B'{=}\beta' \mid B{=}\beta, A{=}a] && \text{(A.1)}\\
&= \sum_{o \in \mathcal{O}} \Pr[B'{=}\beta' \wedge O{=}o \mid B{=}\beta, A{=}a] && \text{(A.2)}\\
&= \sum_{o \in \Theta'_m(\beta, a)} \Pr[B'{=}\beta' \wedge O{=}o \mid B{=}\beta, A{=}a] && \text{(A.3)}\\
&= \sum_{o \in \Theta'_m(\beta, a)} \Pr[B'{=}\beta' \mid B{=}\beta, A{=}a, O{=}o] \Pr[O{=}o \mid B{=}\beta, A{=}a] && \text{(A.4)}\\
&= \sum_{o \in \Theta'_m(\beta, a)} \delta(\beta', \mathsf{update}_m(\beta, a, o)) \Pr[O{=}o \mid B{=}\beta, A{=}a] && \text{(A.5)}\\
&= \sum_{o \in \Theta_m(\beta, a, \beta')} \delta(\beta', \mathsf{update}_m(\beta, a, o)) \Pr[O{=}o \mid B{=}\beta, A{=}a] && \text{(A.6)}\\
&= \sum_{o \in \Theta_m(\beta, a, \beta')} \Pr[O{=}o \mid B{=}\beta, A{=}a] && \text{(A.7)}\\
&= \sum_{o \in \Theta_m(\beta, a, \beta')} N_m(\beta, a)(o) && \text{(A.8)}
\end{aligned}
$$

Line A.2 follows from the Law of Total Probability since the agent can make only a single observation after each action making the different possible observations mutually exclusive events. Line A.3 follows since $\Pr[B'{=}\beta' \wedge O{=}o \mid B{=}\beta, A{=}a]$ is zero for any o not in $\Theta'_m(\beta, a)$. Line A.4 uses the Multiplication Rule, which is well defined in this case since for all o in $\Theta'_m(\beta, a)$, $\Pr[B{=}\beta, A{=}a, O{=}o]$ is non-zero. Using that update is a deterministic function, we know that for each value of β, a, and o, the probability

$\Pr[B'{=}\beta' \mid B{=}\beta, A{=}a, O{=}o]$ is non-zero for only one value of β' and that value is $\mathsf{update}_m(\beta, a, o)$. Thus, $\Pr[B'{=}\beta' \mid B{=}\beta, A{=}a, O{=}o] = \delta(\beta', \mathsf{update}_m(\beta, a, o))$ and Line A.5 is justified. Line A.6 follows since $\delta(\beta', \mathsf{update}_m(\beta, a, o))$ is zero for all o that is in $\Theta'_m(\beta, a)$ but not in $\Theta_m(\beta, a, \beta')$ and $\Theta_m(\beta, a, \beta') \subseteq \Theta'_m(\beta, a)$. Line A.7 follows since $\delta(\beta', \mathsf{update}_m(\beta, a, o))$ is 1 for all o in $\Theta_m(\beta, a, \beta')$.

The state space of $\mathsf{bmdp}(m)$ is uncountably infinite. Thus, some equations used for MDPs are not well defined for the belief MDP $\mathsf{bmdp}(m)$. For example, the equation

$$v_{\mathsf{bmdp}(m)}(\sigma, \beta) = R_m(\beta, \sigma(\beta)) + \gamma \sum_{\beta' \in \mathcal{B}} \tau(\beta, \sigma(\beta))(\beta') * v_{\mathsf{bmdp}(m)}(\sigma, \beta')$$

involves a summation over an uncountably infinite number of belief states, which is not well defined.

Fortunately, we can adjust such equations to restrict summations to index over a countable subset of the belief state space $\mathcal{B}$. This restriction is possible because for each current belief state, action, and observation, only a single next belief state is possible. Thus, for each current belief state and action, only a finite number of next belief states are possible, one for each observation. Let $\Gamma_m(\beta, a)$ denote these possible next states: $\Gamma_m(\beta, a) = \{\, \beta' \in \mathcal{B} \mid \exists o \in \mathcal{O} \text{ s.t. } \mathsf{update}(\beta, a, o) = \beta' \,\}$. The subset $\Gamma_m(\beta, a)$ is finite since $\mathcal{O}$ is finite. β' is in $\Gamma_m(\beta, a)$ if and only if $\Theta_m(\beta, a, \beta')$ is not empty. Thus, if $\tau(\beta, a)(\beta') > 0$, then $\beta' \in \Gamma_m(\beta, a)$ since $\tau(\beta, a)(\beta') = \sum_{o \in \Theta_m(\beta, a, \beta')} N_m(\beta, a)(o)$. Thus, we may replace $\mathcal{B}$ in the above summation with $\Gamma_m(\beta, a)$ to get

$$v_{\mathsf{bmdp}(m)}(\sigma, \beta) = R_m(\beta, \sigma(\beta)) + \gamma \sum_{\beta' \in \Gamma_m(\beta, \sigma(\beta))} \tau(\beta, \sigma(\beta))(\beta') * v_{\mathsf{bmdp}(m)}(\sigma, \beta')$$

Similar adjustments rescue other definitions that involve summations over $\mathcal{B}$. For example, $q^*_{\mathsf{bmdp}(m)}(\beta, a)$ becomes $R_m(\beta, a) + \gamma \sum_{\beta' \in \Gamma_m(\beta, a)} \tau(\beta, a)(\beta') * v^*_{\mathsf{bmdp}(m)}(\beta')$.

After making these adjustments, we can prove that the optimal value that a belief MDP $\mathsf{bmdp}(m)$ assigns to a belief state is equal to the optimal value that the POMDP m assigns to it to prove Proposition 3 in the next section.

A.2 Proof of Proposition 3

For all β and σ,

$$
\begin{aligned}
&\text{(A.9)} && \sum_{o\in\mathcal{O}} N_m(\beta,\sigma(\beta))(o) * V_m(\sigma, \mathsf{update}_m(\beta,\sigma(\beta),o)) \\
&\text{(A.10)} && = \sum_{\beta'\in\Gamma_m(\beta,\sigma(\beta))} \sum_{o\in\mathcal{O}} \delta(\beta', \mathsf{update}_m(\beta,\sigma(\beta),o)) * N_m(\beta,\sigma(\beta))(o) * V_m(\sigma,\beta') \\
&\text{(A.11)} && = \sum_{\beta'\in\Gamma_m(\beta,\sigma(\beta))} \sum_{o\in\Theta_m(\beta,\sigma(\beta),\beta')} \delta(\beta', \mathsf{update}_m(\beta,\sigma(\beta),o)) * N_m(\beta,\sigma(\beta))(o) * V_m(\sigma,\beta') \\
&\text{(A.12)} && = \sum_{\beta'\in\Gamma_m(\beta,\sigma(\beta))} \sum_{o\in\Theta_m(\beta,\sigma(\beta),\beta')} N_m(\beta,\sigma(\beta))(o) * V_m(\sigma,\beta') \\
&\text{(A.13)} && = \sum_{\beta'\in\Gamma_m(\beta,\sigma(\beta))} \left(\sum_{o\in\Theta_m(\beta,\sigma(\beta),\beta')} N_m(\beta,\sigma(\beta))(o) \right) * V_m(\sigma,\beta') \\
&\text{(A.14)} && = \sum_{\beta'\in\Gamma_m(\beta,\sigma(\beta))} \tau(\beta,\sigma(\beta))(\beta') * V_m(\sigma,\beta')
\end{aligned}
$$

Line A.10 is justified since $\mathsf{update}_m(\beta,\sigma(\beta),o)$ is in $\Gamma_m(\beta,\sigma(\beta))$ by definition and $\delta(\beta', \mathsf{update}_m(\beta,\sigma(\beta),o))$ is equal to 1 at exactly one value of β', when $\beta' = \mathsf{update}_m(\beta,\sigma(\beta),o)$, and is 0 for all other values of β'. Line A.11 follows from the fact that for all β, β', σ, and o, $\delta(\beta', \mathsf{update}_m(\beta,\sigma(\beta),o))$ is equal to 0 if o is not in $\Theta_m(\beta,\sigma(\beta),\beta') = \{\, o \in \mathcal{O} \mid \mathsf{update}_m(\beta,\sigma(\beta),o) = \beta' \,\}$. Line A.12 is justified since for all β, β', σ, and o, $\delta(\beta', \mathsf{update}_m(\beta,\sigma(\beta),o))$ is equal to 1 if o is in $\Theta_m(\beta,\sigma(\beta),\beta')$.

The above equation allows us to conclude the following:

$$
\begin{aligned}
V_m^*(\beta) &= \max_\sigma V_m(\sigma,\beta) \\
&= \max_\sigma R_m(\beta,\sigma(\beta)) + \gamma \sum_{o\in\mathcal{O}} N_m(\beta,\sigma(\beta))(o) * V_m(\sigma, \mathsf{update}_m(\beta,\sigma(\beta),o)) \\
&= \max_\sigma R_m(\beta,\sigma(\beta)) + \gamma \sum_{\beta'\in\Gamma_m(\beta,\sigma(\beta))} \tau(\beta,\sigma(\beta))(\beta') * V_m(\sigma,\beta') \\
&= \max_\sigma v_{\mathsf{bmdp}(m)}(\sigma,\beta) \\
&= v^*_{\mathsf{bmdp}(m)}(\beta)
\end{aligned}
$$

Furthermore,

$$
\begin{aligned}
Q_m^*(\beta,a) &= R_m(\beta,a) + \gamma \sum_{o\in\mathcal{O}} N_m(\beta,a)(o) * V_m^*(\mathsf{update}_m(\beta,\sigma(\beta),o)) \\
&= R_m(\beta,a) + \gamma \sum_{\beta'\in\Gamma_m(\beta,a)} \tau(\beta,a)(\beta') * v^*_{\mathsf{bmdp}(m)}(\beta') \\
&= q^*_{\mathsf{bmdp}(m)}(\beta,a)
\end{aligned}
$$

where the middle lines follow from the same reasoning as above.

Appendix B

Details of Empirical Study

B.1 Questionnaire

Below is the content of the questionnaire. The formatting differed in that it was broken up into multiple webpages. Initial instructions were shown on Mechanical Turk's website (Appendix B.1.1). The additional instructions, questions, and payment information were shown on Survey Gizmo's website (Appendix B.1.2). Survey Gizmo always showed the additional instructions first and the payment information last. For each participant, Survey Gizmo presented the scenarios in a random order and on its own webpage. Survey Gizmo numbers the questions dynamically based upon the order in which Survey Gizmo presents the scenarios.

Recall that, for each scenario, the first two questions (Q1 and Q2) have objectively correct answers that the survey participant may easily find by reading the scenario and we use them to determine whether the participant put thought into answering the questions.

B.1.1 Mechanical Turk

If you choose to participate, you will be asked a series questions [sic] about when an action is for a purpose. If you fill out the survey reasonably (do not just randomly select answers), you will be paid for your participation. The risks of taking this survey are equivalent to every day computer use. Your participation is voluntary.
If you choose to participate, then fill out the survey at SurveyGizmo using the following link:
`http://edu.surveygizmo.com/s3/621146/Hospital-Survey`
Upon completion enter the last four digits of your phone number here:

□

We ask for this number so we can track who successfully completed the survey. We will ask you to enter the same number at SurveyGizmo.

B.1.2 Survey Gizmo

Instructions. Metropolis General Hospital has the following privacy policy:

> Metropolis General Hospital and its employees will share a patient's medical record with an outside specialist only for the purpose of providing that patient with treatment.

For each scenario below, please answer the following questions based on your understanding of the above policy.

Scenario 1. Metropolis General Hospital has the following privacy policy:

> Metropolis General Hospital and its employees will share a patient's medical record with an outside specialist only for the purpose of providing that patient with treatment.

Please answer the following questions based on your understanding of the above policy and the following scenario:
A case worker employed by Metropolis General Hospital meets with a patient. The case worker develops a plan with the sole goal of treating the patient. The plan includes sharing the patient's medical record with an outside specialist. Upon receiving the record, the specialist succeeds in treating the patient.

1. Was the goal of the case worker's plan to treat the patient?

   (a) Yes

   (b) No

   (c) I don't know

2. Did the specialist succeed in treating the patient?

   (a) Yes

   (b) No

   (c) I don't know

3. Did the case worker share the record with the specialist for the purpose of treatment?

   (a) Yes

   (b) No

   (c) I don't know

4. Did the case worker obey the above privacy policy?

   (a) Yes

   (b) No

   (c) I don't know

5. Why did you answer Question 4 as you did?

   ____

Scenario 2. Metropolis General Hospital has the following privacy policy:

> Metropolis General Hospital and its employees will share a patient's medical record with an outside specialist only for the purpose of providing that patient with treatment.

Please answer the following questions based on your understanding of the above policy and the following scenario:
A case worker employed by Metropolis General Hospital meets with a patient. The case worker develops a plan with the sole goal of treating the patient. The plan includes sharing the patient's medical record with an outside specialist. Upon receiving the record, the specialist did *not* succeed in treating the patient.

1. Was the goal of the case worker's plan to treat the patient?

    (a) Yes

    (b) No

    (c) I don't know

2. Did the specialist succeed in treating the patient?

    (a) Yes

    (b) No

    (c) I don't know

3. Did the case worker share the record with the specialist for the purpose of treatment?

    (a) Yes

    (b) No

    (c) I don't know

4. Did the case worker obey the above privacy policy?

    (a) Yes

    (b) No

    (c) I don't know

5. Why did you answer Question 4 as you did?

Scenario 3. Metropolis General Hospital has the following privacy policy:

> Metropolis General Hospital and its employees will share a patient's medical record with an outside specialist only for the purpose of providing that patient with treatment.

Please answer the following questions based on your understanding of the above policy and the following scenario:
A case worker employed by Metropolis General Hospital meets with a patient. The case worker develops a plan with the sole goal of reducing costs for the hospital. The plan includes sharing the patient's medical record with an outside specialist. Upon receiving the record, the specialist succeeds in treating the patient.

1. Was the goal of the case worker's plan to treat the patient?

    (a) Yes

    (b) No

    (c) I don't know

2. Did the specialist succeed in treating the patient?

    (a) Yes

    (b) No

    (c) I don't know

3. Did the case worker share the record with the specialist for the purpose of treatment?

    (a) Yes

    (b) No

    (c) I don't know

4. Did the case worker obey the above privacy policy?

    (a) Yes

    (b) No

    (c) I don't know

5. Why did you answer Question 4 as you did?

Scenario 4. Metropolis General Hospital has the following privacy policy:

> Metropolis General Hospital and its employees will share a patient's medical record with an outside specialist only for the purpose of providing that patient with treatment.

Please answer the following questions based on your understanding of the above policy and the following scenario:
A case worker employed by Metropolis General Hospital meets with a patient. The case worker develops a plan with the sole goal of reducing costs for the hospital. The plan includes sharing the patient's medical record with an outside specialist. Upon receiving the record, the specialist did *not* succeed in treating the patient.

1. Was the goal of the case worker’s plan to treat the patient?
   (a) Yes
   (b) No
   (c) I don’t know

2. Did the specialist succeed in treating the patient?
   (a) Yes
   (b) No
   (c) I don’t know

3. Did the case worker share the record with the specialist for the purpose of treatment?
   (a) Yes
   (b) No
   (c) I don’t know

4. Did the case worker obey the above privacy policy?
   (a) Yes
   (b) No
   (c) I don’t know

5. Why did you answer Question 4 as you did?

Payment Information. To receive payment on Mechanical Turk, please enter the last four digits of your phone number here:

B.2 Mechanical Turk Advertisement

```
Research survey on the meaning of privacy

Requester: Michael Carl Tschantz    HIT Expiration Date: Jul 20, 2011 (2 weeks 5 days)   Reward: $0.50
                                    Time Allotted: 10 minutes                            HITs Available: 200
Description: Take a short survey about how you interpret a privacy policy to help
             research on the topic taking place at Carnegie Mellon University.

Keywords: Survey, Research

Qualifications Required:
HIT approval rate (%) is greater than 95
Location is US
```

B.3 Tables of Matched Pairs

Question Q3		$S_{p\bar{f}}$		
		Yes	I don’t know	No
S_{pf}	Yes	182	1	2
	I don’t know	1	0	1
	No	0	0	0

Question Q3		$S_{\bar{p}\bar{f}}$		
		Yes	I don’t know	No
$S_{\bar{p}f}$	Yes	25	2	16
	I don’t know	0	4	2
	No	13	4	121

Question Q4		$S_{p\bar{f}}$		
		Yes	I don’t know	No
S_{pf}	Yes	176	0	6
	I don’t know	0	2	0
	No	1	0	2

Question Q4		$S_{\bar{p}\bar{f}}$		
		Yes	I don’t know	No
$S_{\bar{p}f}$	Yes	26	3	16
	I don’t know	1	5	3
	No	4	1	128

Question Q3		$S_{\bar{p}f}$		
		Yes	I don’t know	No
S_{pf}	Yes	43	6	136
	I don’t know	0	0	2
	No	0	0	0

Question Q3		$S_{\bar{p}\bar{f}}$		
		Yes	I don’t know	No
$S_{p\bar{f}}$	Yes	37	9	137
	I don’t know	0	0	1
	No	1	1	1

Question Q4		$S_{\bar{p}f}$		
		Yes	I don’t know	No
S_{pf}	Yes	45	8	129
	I don’t know	0	1	1
	No	0	0	3

Question Q4		$S_{\bar{p}\bar{f}}$		
		Yes	I don’t know	No
$S_{p\bar{f}}$	Yes	30	8	139
	I don’t know	0	1	1
	No	1	0	7

Scenario S_{pf}		Q3		
		Yes	I don’t know	No
q4	Yes	181	1	0
	I don’t know	1	1	0
	No	3	0	0

Scenario $S_{p\bar{f}}$		Q3		
		Yes	I don’t know	No
q4	Yes	176	1	0
	I don’t know	2	0	0
	No	5	0	3

Scenario $S_{\bar{p}f}$		Q3		
		Yes	I don’t know	No
q4	Yes	32	2	11
	I don’t know	1	3	5
	No	10	1	122

Scenario $S_{\bar{p}\bar{f}}$		Q3		
		Yes	I don’t know	No
q4	Yes	22	0	9
	I don’t know	2	5	2
	No	14	5	128

B.4 Results Using All Respondents

Scenario	Yes	I don't know	No
S_{pf}	205 (99%)	1 (00%)	1 (00%)
$S_{p\bar{f}}$	202 (98%)	2 (01%)	3 (01%)
$S_{\bar{p}f}$	25 (12%)	5 (02%)	177 (86%)
$S_{\bar{p}\bar{f}}$	18 (09%)	2 (01%)	187 (90%)

Q1: Was the goal treatment? (question with an objectively correct answer)

Scenario	Yes	I don't know	No
S_{pf}	206 (100%)	1 (00%)	0 (00%)
$S_{p\bar{f}}$	3 (01%)	1 (00%)	203 (98%)
$S_{\bar{p}f}$	196 (95%)	3 (01%)	8 (04%)
$S_{\bar{p}\bar{f}}$	5 (02%)	0 (00%)	202 (98%)

Q2: Was the treatment successful? (question with an objectively correct answer)

Scenario	Yes	I don't know	No
S_{pf}	205 (99%)	2 (01%)	0 (00%)
$S_{p\bar{f}}$	202 (98%)	2 (01%)	3 (01%)
$S_{\bar{p}f}$	59 (29%)	9 (04%)	139 (67%)
$S_{\bar{p}\bar{f}}$	51 (25%)	14 (07%)	142 (69%)

Q3: Was the action for the purpose?

Scenario	Yes	I don't know	No
S_{pf}	201 (97%)	2 (01%)	4 (02%)
$S_{p\bar{f}}$	195 (94%)	3 (01%)	9 (04%)
$S_{\bar{p}f}$	61 (29%)	11 (05%)	135 (65%)
$S_{\bar{p}\bar{f}}$	44 (21%)	12 (06%)	151 (73%)

Q4: Was the policy obeyed?

Table B.1: Survey Results for All Respondents

Testing	Alternative Hypothesis	Null Hypothesis	p-Value	Significant?
Against H1a	$p_{\mathsf{pfy}} < 0.5$	$p_{\mathsf{pfy}} = 0.5$	1	No
Against H1a	$p_{\mathsf{pfn}} > 0.5$	$p_{\mathsf{pfn}} = 0.5$	1	No
Against H1a	$p_{\mathsf{\bar{p}fy}} < 0.5$	$p_{\mathsf{\bar{p}fy}} = 0.5$	1.59774e-009	Yes
Against H1a	$p_{\mathsf{\bar{p}fn}} > 0.5$	$p_{\mathsf{\bar{p}fn}} = 0.5$	7.097797e-006	Yes
Against H1a’	$p'_{\mathsf{pfy}} < 0.5$	$p'_{\mathsf{pfy}} = 0.5$	1	No
Against H1a’	$p'_{\mathsf{pfn}} > 0.5$	$p'_{\mathsf{pfn}} = 0.5$	1	No
Against H1a’	$p'_{\mathsf{\bar{p}fy}} < 0.5$	$p'_{\mathsf{\bar{p}fy}} = 0.5$	2.606856e-010	Yes
Against H1a’	$p'_{\mathsf{\bar{p}fn}} > 0.5$	$p'_{\mathsf{\bar{p}fn}} = 0.5$	4.514694e-007	Yes
Against H1b	$p_{\mathsf{p\bar{f}n}} < 0.5$	$p_{\mathsf{p\bar{f}n}} = 0.5$	8.206618e-048	Yes
Against H1b	$p_{\mathsf{p\bar{f}y}} > 0.5$	$p_{\mathsf{p\bar{f}y}} = 0.5$	4.833563e-044	Yes
Against H1b	$p_{\mathsf{\bar{p}\bar{f}n}} < 0.5$	$p_{\mathsf{\bar{p}\bar{f}n}} = 0.5$	1	No
Against H1b	$p_{\mathsf{\bar{p}\bar{f}y}} > 0.5$	$p_{\mathsf{\bar{p}\bar{f}y}} = 0.5$	1	No
Against H1b’	$p'_{\mathsf{p\bar{f}n}} < 0.5$	$p'_{\mathsf{p\bar{f}n}} = 0.5$	7.187894e-057	Yes
Against H1b’	$p'_{\mathsf{p\bar{f}y}} > 0.5$	$p'_{\mathsf{p\bar{f}y}} = 0.5$	1.503496e-053	Yes
Against H1b’	$p'_{\mathsf{\bar{p}\bar{f}n}} < 0.5$	$p'_{\mathsf{\bar{p}\bar{f}n}} = 0.5$	1	No
Against H1b’	$p'_{\mathsf{\bar{p}\bar{f}y}} > 0.5$	$p'_{\mathsf{\bar{p}\bar{f}y}} = 0.5$	1	No
For H2a	$p_{\mathsf{pfy}} > 0.5$	$p_{\mathsf{pfy}} = 0.5$	5.08808e-052	Yes
For H2a	$p_{\mathsf{pfn}} < 0.5$	$p_{\mathsf{pfn}} = 0.5$	3.684324e-055	Yes
For H2a	$p_{\mathsf{p\bar{f}y}} > 0.5$	$p_{\mathsf{p\bar{f}y}} = 0.5$	4.833563e-044	Yes
For H2a	$p_{\mathsf{p\bar{f}n}} < 0.5$	$p_{\mathsf{p\bar{f}n}} = 0.5$	8.206618e-048	Yes
For H2a’	$p'_{\mathsf{pfy}} > 0.5$	$p'_{\mathsf{pfy}} = 0.5$	1.046682e-058	Yes
For H2a’	$p'_{\mathsf{pfn}} < 0.5$	$p'_{\mathsf{pfn}} = 0.5$	4.861731e-063	Yes
For H2a’	$p'_{\mathsf{p\bar{f}y}} > 0.5$	$p'_{\mathsf{p\bar{f}y}} = 0.5$	1.503496e-053	Yes
For H2a’	$p'_{\mathsf{p\bar{f}n}} < 0.5$	$p'_{\mathsf{p\bar{f}n}} = 0.5$	7.187894e-057	Yes
For H2b	$p_{\mathsf{\bar{p}fn}} > 0.5$	$p_{\mathsf{\bar{p}fn}} = 0.5$	7.097797e-006	Yes
For H2b	$p_{\mathsf{\bar{p}fy}} < 0.5$	$p_{\mathsf{\bar{p}fy}} = 0.5$	1.59774e-009	Yes
For H2b	$p_{\mathsf{\bar{p}\bar{f}n}} > 0.5$	$p_{\mathsf{\bar{p}\bar{f}n}} = 0.5$	1.443359e-011	Yes
For H2b	$p_{\mathsf{\bar{p}\bar{f}y}} < 0.5$	$p_{\mathsf{\bar{p}\bar{f}y}} = 0.5$	1.440142e-017	Yes
For H2b’	$p'_{\mathsf{\bar{p}fn}} > 0.5$	$p'_{\mathsf{\bar{p}fn}} = 0.5$	4.514694e-007	Yes
For H2b’	$p'_{\mathsf{\bar{p}fy}} < 0.5$	$p'_{\mathsf{\bar{p}fy}} = 0.5$	2.606856e-010	Yes
For H2b’	$p'_{\mathsf{\bar{p}\bar{f}n}} > 0.5$	$p'_{\mathsf{\bar{p}\bar{f}n}} = 0.5$	4.581869e-008	Yes
For H2b’	$p'_{\mathsf{\bar{p}\bar{f}y}} < 0.5$	$p'_{\mathsf{\bar{p}\bar{f}y}} = 0.5$	7.161858e-014	Yes

Table B.2: Binomial Hypothesis Tests for All Respondents

Testing	Alternative Hypothesis	Null Hypothesis
Proving H2a	$p_{\mathsf{pfy}} > 0.94$	$p_{\mathsf{pfy}} = 0.94$
Proving H2a	$p_{\mathsf{pfn}} < 0.05$	$p_{\mathsf{pfn}} = 0.05$
Proving H2a	$p_{\mathsf{p\bar{f}y}} > 0.9$	$p_{\mathsf{p\bar{f}y}} = 0.9$
Proving H2a	$p_{\mathsf{p\bar{f}n}} < 0.08$	$p_{\mathsf{p\bar{f}n}} = 0.08$
Proving H2a’	$p'_{\mathsf{pfy}} > 0.96$	$p'_{\mathsf{pfy}} = 0.96$
Proving H2a’	$p'_{\mathsf{pfn}} < 0.02$	$p'_{\mathsf{pfn}} = 0.02$
Proving H2a’	$p'_{\mathsf{p\bar{f}y}} > 0.94$	$p'_{\mathsf{p\bar{f}y}} = 0.94$
Proving H2a’	$p'_{\mathsf{p\bar{f}n}} < 0.04$	$p'_{\mathsf{p\bar{f}n}} = 0.04$
Proving H2b	$p_{\mathsf{\bar{p}fn}} > 0.59$	$p_{\mathsf{\bar{p}fn}} = 0.59$
Proving H2b	$p_{\mathsf{\bar{p}fy}} < 0.36$	$p_{\mathsf{\bar{p}fy}} = 0.36$
Proving H2b	$p_{\mathsf{\bar{p}\bar{f}n}} > 0.67$	$p_{\mathsf{\bar{p}\bar{f}n}} = 0.67$
Proving H2b	$p_{\mathsf{\bar{p}\bar{f}y}} < 0.27$	$p_{\mathsf{\bar{p}\bar{f}y}} = 0.27$
Proving H2b’	$p'_{\mathsf{\bar{p}fn}} > 0.61$	$p'_{\mathsf{\bar{p}fn}} = 0.61$
Proving H2b’	$p'_{\mathsf{\bar{p}fy}} < 0.35$	$p'_{\mathsf{\bar{p}fy}} = 0.35$
Proving H2b’	$p'_{\mathsf{\bar{p}\bar{f}n}} > 0.62$	$p'_{\mathsf{\bar{p}\bar{f}n}} = 0.62$
Proving H2b’	$p'_{\mathsf{\bar{p}\bar{f}y}} < 0.31$	$p'_{\mathsf{\bar{p}\bar{f}y}} = 0.31$

Table B.3: Extreme Binomial Hypothesis Tests for All Respondents. This table shows the hypothesis test using the most extreme probability for which statistical significance is still achieved and is accurate up to two places after the decimal point.

Question Q3		$S_{p\bar{f}}$		
		Yes	I don’t know	No
	Yes	201	2	2
S_{pf}	I don’t know	1	0	1
	No	0	0	0

Question Q3		$S_{\bar{p}\bar{f}}$		
		Yes	I don’t know	No
	Yes	38	3	18
$S_{\bar{p}f}$	I don’t know	0	7	2
	No	13	4	122

Question Q4		$S_{p\bar{f}}$		
		Yes	I don’t know	No
	Yes	194	1	6
S_{pf}	I don’t know	0	2	0
	No	1	0	3

Question Q4		$S_{\bar{p}\bar{f}}$		
		Yes	I don’t know	No
	Yes	39	4	18
$S_{\bar{p}f}$	I don’t know	1	7	3
	No	4	1	130

Question Q3		$S_{\bar{p}f}$		
		Yes	I don’t know	No
	Yes	59	9	137
S_{pf}	I don’t know	0	0	2
	No	0	0	0

Question Q3		$S_{\bar{p}\bar{f}}$		
		Yes	I don’t know	No
	Yes	50	12	140
$S_{p\bar{f}}$	I don’t know	0	1	1
	No	1	1	1

Question Q4		$S_{\bar{p}f}$		
		Yes	I don’t know	No
	Yes	61	10	130
S_{pf}	I don’t know	0	1	1
	No	0	0	4

Question Q4		$S_{\bar{p}\bar{f}}$		
		Yes	I don’t know	No
	Yes	43	10	142
$S_{p\bar{f}}$	I don’t know	0	2	1
	No	1	0	8

Scenario S_{pf}		Q3		
		Yes	I don’t know	No
	Yes	200	1	0
q4	I don’t know	1	1	0
	No	4	0	0

Scenario $S_{p\bar{f}}$		Q3		
		Yes	I don’t know	No
	Yes	194	1	0
q4	I don’t know	2	1	0
	No	6	0	3

Scenario $S_{\bar{p}f}$		Q3		
		Yes	I don’t know	No
	Yes	47	3	11
q4	I don’t know	1	5	5
	No	11	1	123

Scenario $S_{\bar{p}\bar{f}}$		Q3		
		Yes	I don’t know	No
	Yes	34	1	9
q4	I don’t know	2	8	2
	No	15	5	131

Table B.4: Matched Pairs for All Respondents

Testing	Question	Scenarios	p-Value	Significant?
For H1c	Q4	S_{pf} vs. $S_{\mathsf{p\bar{f}}}$	NaN	No
For H1c	Q4	$S_{\mathsf{\bar{p}f}}$ vs. $S_{\mathsf{\bar{p}\bar{f}}}$	0.008449127	Yes
For H1c’	Q3	S_{pf} vs. $S_{\mathsf{p\bar{f}}}$	0.3430301	No
For H1c’	Q3	$S_{\mathsf{\bar{p}f}}$ vs. $S_{\mathsf{\bar{p}\bar{f}}}$	0.2147006	No
For H2c	Q4	S_{pf} vs. $S_{\mathsf{\bar{p}f}}$	2.300576e-030	Yes
For H2c	Q4	$S_{\mathsf{p\bar{f}}}$ vs. $S_{\mathsf{\bar{p}\bar{f}}}$	2.598558e-032	Yes
For H2c’	Q3	S_{pf} vs. $S_{\mathsf{\bar{p}f}}$	7.115157e-032	Yes
For H2c’	Q3	$S_{\mathsf{p\bar{f}}}$ vs. $S_{\mathsf{\bar{p}\bar{f}}}$	4.269341e-032	Yes

Table B.5: McNemar’s Test Across Scenarios for All Respondents

Scenario	Questions	p-Value	Significant?
S_{pf}	Q4 vs. Q3	NaN	No
$S_{\mathsf{p\bar{f}}}$	Q4 vs. Q3	NaN	No
$S_{\mathsf{\bar{p}f}}$	Q4 vs. Q3	0.2997806	No
$S_{\mathsf{\bar{p}\bar{f}}}$	Q4 vs. Q3	0.3736321	No

Table B.6: McNemar’s Test Across Questions for All Respondents

Appendix C

Notation

a	an action
A	a random variable over actions
$\mathcal{A}$	the action space of an MDP or POMDP
active	the prefix of a given execution before the first instance of the action stop
ad	an advertising action in the advertising example
b	a behavior
B	a random variable over belief states
$\mathcal{B}$	space of all possible belief states
behv	the set of optimal behaviors for a given MDP or POMDP
B	the binomial distribution for the provided parameters
bmdp	the belief MDP of a given POMDP
d	ranges over $\{\mathsf{f}, \mathsf{m}, \bot\}$, the possible values of the database in the advertising example
degen	a degenerate distribution: for any x, $\mathsf{degen}(x)$ assigns probability 1 to x
Dist	the space of distributions over a given set
e	an execution of an MDP or POMDP
$\mathbb{E}$	expectation
f	stands for *female* in the advertising example
F	cumulative distribution function
g	ranges over $\{\mathsf{f}, \mathsf{m}\}$, the possible sexes of a website user in the advertising example

h	the number of steps modeled for multi-step RHIO application
H	a hypothesis about the survey
i	an index
j	an index
ℓ	a log
L	set of all possible logs
log	function from a behavior to a log
log^{-1}	the inverse of log
m	an MDP or POMDP
m_{adv}	a POMDP modeling the advertising example
m_{phy}	a POMDP modeling the physician example
m	stands for *male* in the advertising example
n	the length of a sequence or the sample size of the survey
N	ν lifted for POMDPs
$\mathbb{N}$	the set of natural numbers: $\{1, 2, 3, \ldots\}$
nbehv	the set of non-redundant optimal behaviors of a given MDP or POMDP
nopt	set of non-redundant optimal strategies of a given MDP or POMDP
o	an observation of POMDP
O	a random variable over observations of POMDP (often denoted by Ω) of an equivalence class of $\equiv$
$\mathcal{O}$	space of observations of a POMDP
opt	set of optimal strategies of a given MDP or POMDP
p	a purpose or a probability in Ch 5
$\Pr$	the probability of an expression
q	value function of a state and an action of an MDP given a strategy (typically denoted by Q)
q^*	optimal value function of a state and an action of an MDP (typically denoted by Q^*)
q^*	a sub-routine for computing q^* from v^*

Q	value function of a state and an action of a POMDP given a strategy
Q^*	optimal value function of a state and an action of a POMDP
$\mathtt{Q}^*$	a sub-routine for computing Q^* from V^*
Q	a question of the survey
r	reward function for an MDP or POMDP
R	reward function r raised for POMDPs
$\mathbb{R}$	set of real numbers
s	a state
S	a random variable over states
$\mathcal{S}$	state space
S	a scenario in the survey
stop	the distinguished action of NMDP or NPOMDP that indicates stopping and doing nothing more
t	transition relation of an MDP or POMDP
update	the function that updates an agents beliefs given an action and an observation of a POMDP
v	value function of a state of an MDP given a strategy (typically denoted by V)
v^*	optimal value function of a state of an MDP (typically denoted by V^*)
$\mathsf{v}^*_{\mathsf{low}}$	lower bounds on v^*
$\mathsf{v}^*_{\mathsf{up}}$	upper bounds on v^*
V	value function of a state of a POMDP given a strategy
V^*	optimal value function of a state of a POMDP
$\mathsf{V}^*_{\mathsf{low}}$	lower bounds on V^*
$\mathsf{V}^*_{\mathsf{up}}$	upper bounds on V^*
x	ranges over X-rays in the physician example or over the elements of any set
y	the number of *yes* responses to the survey or ranges over the elements of any set
Y	random variable over number of *yes* responses to the survey
α	ranges over what advertisement (or none) the website could have shown in the advertising example; the level used for significance for hypothesis testing the survey

β	a belief state of a belief MDP
γ	discounting factor of an MDP or POMDP
Γ	set of belief states with non-zero probability
δ	Kronecker's delta
δ^{r}	the increase in quality of treatment from reading a patient's record in the RHIO application
δ^{s}	the increase in quality of treatment from studying medical literature in the RHIO application
κ	a contingency
ν	probability of an observation given a state and an action for a POMDP (often denoted by O)
ρ	a reward
ρ^{t}	the reward for treating a patient in the RHIO application
σ	a strategy (typically called "policy" and denoted by π)
τ	transition relation of a belief MDP
Θ	set of observations that could have resulted in a given updated belief for a POMDP
Θ'	set of observations possible for a given belief state and action for a POMDP
$\bot$	stands for no information known about the user's sex in the advertising example
$\circ$	a dummy observation providing no information
$\varnothing$	stands for the website having not shown any advertisement to the user in the advertising example
$\times$	cross product
$\equiv$	an equivalence relation over observations
$\equiv_{\mathsf{adv}}$	an equivalence relation used in the advertising example
$\equiv_{\mathsf{phy}}$	an equivalence relation used in the physician example
$\sqsubset$	prefix relation over sequences
$\sqsubseteq$	prefix-or-equal relation over sequences
$\prec$	sub-strategy relation
$\lhd$	sub-execution relation
$\sim$	denotes that a random variable obeys a distribution