



Prognostic normative reasoning

Jean Oh^{a,*}, Felipe Meneguzzi^b, Katia Sycara^a, Timothy J. Norman^c

^a Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, USA

^b School of Computer Science, Pontificia Universidade Católica do Rio Grande do Sul, Porto Alegre, RS, Brazil

^c Department of Computing Science, University of Aberdeen, Aberdeen, UK

ARTICLE INFO

Article history:

Received 8 December 2011

Received in revised form

13 November 2012

Accepted 10 December 2012

Available online 27 December 2012

Keywords:

Proactive agents

Normative reasoning

Agent architecture

ABSTRACT

Human users planning for multiple objectives in complex environments are subjected to high levels of cognitive workload, which can severely impair the quality of the plans created. This paper describes a software agent that can proactively assist cognitively overloaded users by providing normative reasoning about prohibitions and obligations so that the user can focus on her primary objectives. In order to provide proactive assistance, we develop the notion of prognostic normative reasoning (PNR) that consists of the following steps: (1) recognizing the user's planned activities, (2) reasoning about norms to evaluate those predicted activities, and (3) providing necessary assistance so that the user's activities are consistent with norms. The idea of PNR integrates various AI techniques, namely, user intention recognition, normative reasoning over a user's intention, and planning, execution and replanning for assistive actions. In this paper, we describe an agent architecture for PNR and discuss practical applications.

© 2012 Elsevier Ltd. All rights reserved.

1. Introduction

Human users planning for multiple objectives in highly complex environments are subjected to high levels of cognitive workload, which can severely impair the quality of the plans created. The cognitive workload is significantly increased when a user must not only cope with a complex environment, but also with a set of complex rules (or norms) that prescribe how the planning process must be carried out. For example, military planners during peacekeeping operations have to plan to achieve their own unit's objectives while following standing operating procedures that regulate how interaction and collaboration with Non-Governmental Organizations (NGOs) must take place. These procedures generally prescribe conditions upon which the military should perform escort missions, for example, to ensure that humanitarian NGO personnel are kept safe in conflict areas. To assist cognitively overloaded human users, we develop a software assistant agent that can proactively provide timely reasoning support.

In this paper, we specifically aim to assist the user's *normative reasoning*. Norms generally define a set of rules that is enforced among the members of a society. In this paper, normative reasoning refers to the reasoning about prohibitions and obligations specified

in those rules of a society. A recent study shows that dynamically changing normative stipulations hinder human ability to plan to both accomplish goals and abide by the norms (Sycara et al., 2010). This result is not surprising, and the difficulty is intensified in a multi-national or multi-cultural team context that is common in various operations today. To help the users cope with cognitive overload, there have been growing interests in automated normative reasoning. Existing work on automated norm management relies on a deterministic view of the planning model (Modgil et al., 2009), where norms are specified in terms of classical logic; in this approach, violations are detected only after they have occurred, consequently assistance can only be provided after the user has already committed actions that caused the violation (Sycara et al., 2010). By contrast, our approach aims to predict potential future violations and proactively take actions to help prevent the user from violating the norms.

In order to provide a user with a timely support, it is crucial that the agent recognizes the user's needs in advance so that the agent can work in parallel with the user to ensure that the assistance is ready by the time the user actually needs it. This desideratum imposes several technical challenges for the assistant agent to: (1) *recognize* the user's planned activities, (2) *reason* about potential needs of assistance for those predicted activities to comply with norms as much as possible, and (3) *plan* to provide appropriate assistance suitable for newly identified user needs.

Our approach to tackle these challenges is realized in a proactive planning agent framework. As opposed to planning for a *given* task, the key challenge we address here is to *identify* a new

* Corresponding author.

E-mail addresses: jeanoh@cs.cmu.edu (J. Oh), felipe.meneguzzi@pucrs.br (F. Meneguzzi), katia@cs.cmu.edu (K. Sycara), t.j.norman@abdn.ac.uk (T.J. Norman).

set of tasks for the agent, *i.e.*, the agent needs to figure out when and what it can do for the user. Specifically, we employ a probabilistic plan recognition technique to predict a user's plan for her future activities. The agent then evaluates the predicted user plan to detect any potential norm violations, generating a set of new tasks for the agent to prevent the occurrence of such norm violations. After identifying new tasks, the agent plans, executes, and replans a series of actions to perform the tasks. As the user's environment changes the agent continuously updates its predictions. Subsequently, the agent must frequently revise its plans during execution. To enable a full cycle of autonomy, we present an agent architecture that seamlessly integrates techniques for plan recognition; normative reasoning over a user's plan; and planning, execution and replanning for assistive actions.

We have published abstract descriptions of our approach in Oh et al. (2011a, 2011b). Several readers of our previous publications expressed interests in the parts that were omitted due to space limitation. This paper provides the detail of our approach that has not been fully published earlier. The main contributions of this paper are the following. We present a principled agent architecture for prognostic reasoning assistance by integrating probabilistic plan recognition with reasoning about norm compliance. We develop the notion of prognostic normative reasoning to predict the user's likely normative violations, allowing the agent to plan and take remedial actions before the violations actually occur. To the best of our knowledge, our approach is the first that manages norms in a proactive and autonomous manner. Our framework supports interleaved planning and execution for the assistant agent to adaptively revise its plans during execution, taking time constraints into consideration to ensure timely support to prevent violations. In order to avoid the agent's interference with the user's actions, although the agent can observe the variables representing the user's state, we assume that the user and the agent can act upon a disjoint set of variables. Thus, although the assistant does not directly enforce the norms, it tries to steer the user towards compliance through communication. For a proof of concept experiment, our approach has been fully implemented in the context of a military peacekeeping scenario.

The rest of this paper is organized as follows. After reviewing related work in Section 2, we describe a high-level architecture of the agent system in Section 3. The three main components are described in detail in the following sections: Section 4 describes the agent's plan recognition algorithm for predicting the user's future plan; Section 5 describes how the agent evaluates the norms to maintain a normative state and to detect potential violations; and Section 6 presents how the agent plans and executes actions to accomplish identified goals. We present a fully implemented system in a peacekeeping problem domain, followed by other potential applications in Section 8, and conclude the paper in Section 9.

2. Related work

We develop the notion of prognostic normative reasoning by bridging two significant branches of AI research: plan recognition and normative reasoning. Here, we discuss only the work closely related to ours and point the readers to survey articles for broader background.

2.1. Plan recognition

Plan recognition refers to the task of identifying the user's high-level goals (or intentions) by observing the user's current activities. In order to recognize a user plan, the agent should have

some model of how the user typically performs certain tasks; for instance, given two locations the user may have a set of preferred routes to drive between the two locations. Such a model is referred to as a *plan library*, and it represents a set of alternative ways to solve a domain-specific problem. The majority of existing work in plan recognition relies on plan libraries; that is, plan recognition algorithms aim to find a plan in the library that best explains the observed behavior. For specific techniques, we refer the readers to survey articles such as Armentano and Amandi (2007).

Constructing a plan library is, however, an elaborate process. In order to facilitate the cumbersome step of building a plan library, recent work proposed the idea of formulating plan recognition as a planning problem. Notably, one approach uses classical planners (Ramírez and Geffner, 2009), whereas the other approach uses decision-theoretic planners (Baker et al., 2009). Following the plan recognition as planning principle, our approach utilizes a decision-theoretic planner, specifically a Markov Decision Process (MDP) (Bellman, 1957). The decision-theoretic planners aim to find an optimal plan with respect to the objective of maximizing a discounted long-term reward (or minimizing a cost). The work presented in Baker et al. (2009) uses an MDP to represent how people recognize the plans of others. During their experiments, human subjects watch an agent moving in a two-dimensional space and are asked to predict the agent's goal positions. Their results show that decision-theoretic predictions match well with those of the human subjects. In this respect, our approach is to design a software assistant to make predictions in a similar manner to a human assistant.

2.2. Normative reasoning

In order to ensure that certain global properties of a society or organization are maintained, rules (or norms) that express permissions, prohibitions and obligations have been developed (Jones, 1990). These concepts represent, respectively, situations that must, must not and can be the case in the world for it to comply with the societal rules. Mathematical study of norms has been carried out in the context of deontic logic (von Wright, 1968), while computational treatment of these stipulations has been studied recently by the agents community as normative systems. These efforts led to the development of various formal models of norms (Vázquez-Salceda et al., 2005), as well as practical approaches to reasoning about norms within individual agents (Lopez y Lopez and Luck, 2003) and in a society (García-Camino et al., 2009). The formalisms that allow modeling of norms for agent systems can also be used for the specification of the rules that humans must follow. Since this work is concerned with assisting a user to mitigate the cognitive load of planning under normative constraints, we leverage the formalisms to create an internal representation of the norms that the assistant must consider when providing assistance.

In order for norms to be enforced in a norm-regulated system, various mechanisms were devised to monitor norm compliance within a system. The state of compliance of a set of norms within a system is known as the *normative state* (Farrell et al., 2005) and describes which agents are complying (or violating) which norms. Although various approaches to norm monitoring have been proposed (Farrell et al., 2005; Modgil et al., 2009; Hübner et al., 2010), they all rely on a deterministic logic view of the normative state. Without a probabilistic model of agent behavior, a norm monitoring mechanism can only assert whether a norm is definitely violated or not, lacking a gradual notion of how *likely* an agent is to violate a norm or *when* an agent is *about* to violate a norm. Thus, an assistant aiming to warn a user of potential violations can either *constantly* remind the user of *all* the norms

Download English Version:

<https://daneshyari.com/en/article/380839>

Download Persian Version:

<https://daneshyari.com/article/380839>

[Daneshyari.com](https://daneshyari.com)