Journal of Computer and System Sciences 78 (2012) 370–391

Contents lists available at ScienceDirect

Journal of Computer and System Sciences

www.elsevier.com/locate/jcss

Maintaining awareness using policies; Enabling agents to identify relevance of information

Amir Talaei-Khoei [a,*], Terje Solvoll [b], Pradeep Ray [a], Nandan Parameshwaran [a]

[a] School of Information Systems, Technology and Management, The University of New South Wales, No. 1039, Quadrangle Building, UNSW, Sydney, NSW 2052, Australia
[b] Norwegian Centre for Integrated Care and Telemedicine, University Hospital of North-Norway, Tromsø, Norway

ARTICLE INFO

Article history:
Received 29 November 2010
Received in revised form 25 March 2011
Accepted 31 May 2011
Available online 6 June 2011

Keywords:
Computer supported cooperative work
Awareness
Intelligent agents
Policy

ABSTRACT

The field of computer supported cooperative work aims at providing information technology models, methods, and tools that assist individuals to cooperate. The presented paper is based on three main observations from literature. First, one of the problems in utilizing information technology for cooperation is to identify the relevance of information, called awareness. Second, research in computer supported cooperative work proposes the use of agent technologies to aid individuals to maintain their awareness. Third, literature lacks the formalized methods on how software agents can identify awareness. This paper addresses the problem of awareness identification. The main contribution of this paper is to propose and evaluate a formalized structure, called Policy-based Awareness Management (PAM). PAM extends the logic of general awareness in order to identify relevance of information. PAM formalizes existing policies into Directory Enabled Networks-next generation structure and uses them as a source for awareness identification. The formalism is demonstrated by applying PAM to the space shuttle Columbia disaster occurred in 2003. The paper also argues that efficacy and cost-efficiency of the logic of general awareness will be increased by PAM. This is evaluated by simulation of hypothetical scenarios as well as a case study.

1. Introduction

In a time when information technologies (IT) are increasingly involving in the people's lives, cooperative environments need to adapt themselves to the new approaches in cooperation. The evolution of cooperative environments has been marked by the emphasis given to methods on how to utilize intelligent IT tools to enhance cooperation among participants. In order to enhance cooperation, computer supported cooperative work (CSCW) proposes the use of awareness, while it is defined as the understanding of information relevant to one's goals [1–4]. Daneshgar and Wang [1] state that there is currently no definitive method to identify such awareness.

CSCW has, recently, evolved to embrace complexity-based paradigm [5]. This paradigm replaces deterministic perspectives of the internal and external views of systems by agency principles [6]. The agency principles emphasize on the role of individuals in a system. Zacarias et al. [5] define agency relationship as interactions between individuals and software agents to perform tasks on individuals' behalf. Much research proposes use of agents to aid individuals maintaining their awareness [7,8]. In this paper, we employ intelligent agents to assist individuals in awareness identification.

The motivation of this research is explained by the space shuttle Columbia disaster, US, 2003, which also goes through the paper and demonstrates the proof of concepts. The paper aims at two main contributions. First, it proposes a formalized

* Corresponding author.
E-mail address: amirtk@kth.se (A. Talaei-Khoei).

0022-0000/$ – see front matter
doi:10.1016/j.jcss.2011.05.013

Table 1
Comparison of awareness definitions.

Awareness type	Definition	About artifact	About people	About activity	About context
Workspace awareness [21]	Up-to-the-minutes information about the existence of entities in a shared workspace	×	×		
Common-sense awareness [3]	General sense of who is around and what does belong to them	×	×		
Group awareness [3]	Understanding of people in the group, their responsibilities, and their status		×	×	
Social awareness [22]	Information about presence of people and their activities	×	×	×	
Context awareness [23]	Non-ignorance of an internal or external entity that causes change in a situation	×	×	×	×

structure, called Policy-based Awareness Management (PAM), which enables software agents to identify relevance of information. In order to do so, this paper extends the logic of general awareness [9–11] and uses a given set of policies as a source of awareness. PAM formalizes policies in Directory Enabled Networks-next generation (DEN-ng) structure [12]. Second, the paper evaluates cost-efficiency and efficacy of PAM in a triangulation [13] of two simulation studies: (I) simulations with hypothetical example scenarios, and (II) simulations with a scenarios found in the case study of wireless communication systems at St. Olavs Hospital, Trondheim University Hospital, Norway.

The paper is organized in the following way: Section 2 describes analyzing the space shuttle Columbia disaster in the US in 2003. Section 3 presents a background and related work. Section 4 presents the PAM framework for modeling awareness in policy-based agent systems. Section 5 presents a three-step process to create awareness of agents based on policies. Section 6 presents the evaluation of PAM in both lab-base simulations and case study. Section 8 discusses the implications and limitations of PAM. Section 8 concludes the paper.

2. Demonstrative case

The following scenario took place during the re-entry of space shuttle Columbia to the earth atmosphere over Texas, USA on February 1, 2003. The disaster was disintegration of the shuttle that claimed the lives of all seven of its crew.

In the space shuttle Columbia flight in 2003, very soon after launch, a part of temporal protection system broke and shuttle began to shake [14]. In the time, the NASA engineering team had only very low resolution pictures of the shuttle's situation. Therefore, they recognized two possible causes for the shakes in the shuttle. The shakes could be caused by the shuttle turning around for the re-entry to the atmosphere or it could be caused by damage to the temporal protection system (TPS). Therefore, the NASA engineering team requested high-resolution imaging [15]. Unfortunately, all of the sudden, the NASA management declared the shakes as a turnaround issue and ignored the request of the engineering team. Therefore, the relation between the possible damage to the TPS and the shakes was not considered [15].

Had NASA recognized the relevance of the damage in the TPS, they would have requested high-resolution imaging from DoD to find out whether there is damage or not. There would also have been a procedure by spacewalk for repairing the damage [15]. At the time, there were policy guidelines [16] in NASA stating that when an aircraft experiences unusual shakes, if there is any TPS damage, the spacewalk procedure must be granted. Although the capability and the guidelines were available, the NASA management team did not recognize the relevance of information, which led to deny the image request and accordingly the disaster [14].

The concept of awareness in CSCW is defined for such situations when we need to recognize the relevance of information. One of the challenging questions is how to recognize awareness or in the other words, how the NASA management could have recognized the relevance of the TPS damage to the shakes in the shuttle.

Prior research has proposed that agents can be used to assist individuals in obtaining awareness. In this paper, we employ intelligent software agents and propose a step-wised process to identify awareness based on a given set of policies, called Policy-based Awareness Management (PAM).

3. Background and related work

The comprehensive literature reviews on awareness in CSCW appear in [17–20]. In brief, literature discerns five types of awareness as it explained in Table 1. As it is shown in the table the context awareness covers the other types of awareness in terms of the type of information that it addresses. Such awareness is highly contextual and cannot be addressed by the other types of awareness [18]. Ray et al. [4] define context awareness as an understanding of relevant information that is required for an individual. In this definition, although, the notation of context awareness shows the relevance of information, it does not refer to the knowledge about the information [18]. Our concept of awareness is most closely related to context awareness.

There are bodies of work in application of information technology in cooperative environments that propose the agent technology to aid roles maintaining their awareness [7,8]. Research in intelligent agents has been interested in the natural

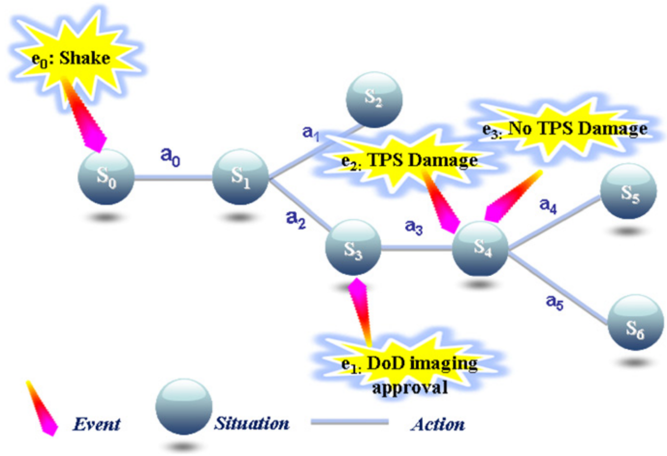

Fig. 1. World for the space shuttle Columbia disaster – NASA management. a_0: realizing the shake, a_1: announcing shake as a turn-around effect, a_2: requesting imaging from DoD, a_3: understanding that DoD sent the images to the engineering team. a_4: announcing no TPS damage. a_5: granting spacewalk rescue procedure.

semantics for awareness as a mental attitude. The classical approach is the possible-worlds model in which a situation can be considered possible in addition to true or false [24]. In the following, we explain the problem of logical omniscience and perfect reasoning in the possible-world model.

3.1. The problem of logical omniscience and perfect reasoning

The possible-worlds model provides an intuitive semantics for mental attitudes of agents [25], but it also commits us to logical omniscience and perfect reasoning: (1) the agent is omniscient, i.e. it knows all the valid formulas; (2) the agent is a perfect reasoner, i.e. if the agent knows p and knows that p implies q, it knows q. This is clearly an idealization. In real life, we only know a certain sub-set of valid formulas, we only know the relevant formulas. In the following we first show these problems via the space shuttle Columbia disaster and then we address the approaches available in the literature for these problems. Due to the space limitation, we only discuss the awareness approach as it is foundation of this research.

3.1.1. Logical omniscience and perfect reasoning in space shuttle Columbia disaster

As an example, Fig. 1 shows the world for the mental states of NASA management. The nodes represent the situations. The arrows going into the nodes represent what event occurred in each of these situations. The lines between different worlds represent actions that the NASA management could take to change the situation. The detailed specification of this diagram will be discussed in Section 4.

While being in the situation s_1, although the TPS on the left wing of the space shuttle had damage, NASA management did not know it, i.e. logical omniscience. In the same time, although the policy guideline was available to initiate a spacewalk procedure, NASA management did not consider it relevant to the situation, i.e. perfect reasoning.

3.1.2. Dealing with logical omniscience and perfect reasoning

There are several approaches to represent logical omniscience and perfect reasoning, including the algorithmic approach, syntactic approach, impossible worlds approach, and the awareness approach. Readers are advised to refer [26]. However, Halpern and Pucella [26] claim that the above approaches are equi-expressive and in practice, there may be a natural interpretation for each of these approaches, which makes the pragmatic approach as the selective criteria. Looking at the definition of awareness and the nature of space shuttle Columbia disaster, our problem matches with awareness approach.

On the awareness approach, e.g. the logic of general awareness [9–11], the essential idea is relevance of knowledge. Under the possible-worlds interpretation, a valid sentence and its consequences are true at every world that the agent considers possible. However, a known sentence and its known consequences may or may not be relevant. Therefore, in the logic of general awareness, an agent implicitly knows all the valid sentences, but it changes its implicit knowledge to the explicit knowledge if and only if, the agent is aware of the sentence. Sillari [11] defines awareness of a propositional sentence as the relevance of that sentence to a situation. Therefore, the notation of awareness does not refer to validity of a sentence. Regardless that a sentence is valid or not, an agent becomes of aware of sentence if and only if it identifies the relevance of the sentence to the situation. Halpern and Pucella [26] argue that with the awareness approach we must explain how awareness can be obtained. In the other words, how agents identify the relevance of information to a given situation. Policy-based Awareness Management (PAM) is an extension to the logic of general awareness and uses policies as a source for identifying and obtaining awareness.

4. Policy-based Awareness Management (PAM) framework

The objective of the PAM framework is to set a foundation consisting of different ideas borrowed from literature. The framework is supported by different definitions given in [9,11,26–28].

4.1. Informal semantics: Intuitions

We consider a system involved with different agents. These agents are being run in the same system, although they have their own model of the world. We model the world using a temporal tree [29] with a single pat and multiple futures. This structure is called *branching-time model* of the world. An example of branching-time model is given in Fig. 1. A specific time-point in a specific world is called *situation*. In Fig. 1, situations are presented by blue circles. Situations present the different circumstances in the world. Once an agent perceives a change, it goes from a situation to the other and holds the new situations.

Events are also ways by which we classify any change in the system. However, these changes do not necessarily change the agent's situation in its world. In fact, an agent might change its situation while another one might be apathetic about the event. Therefore, the event itself is very cognitive in nature. That is, the agents involved in the system treat the events in different ways depending on their mental attitude in the situation that they receive the event. The difference between events and situations to express a change is clear when we say an event occurs or is received but we say a situation is hold. That is, in events the agent seeks a way of reacting to the circumstances while situations are more informatics about the different true or false propositions. In a world, there are a set of events that they have been received and an agent has also a queue of received events.

Actions transform one situation into another. Actions here are primitive, meaning that they are performable directly by the agent who does the action. A primitive event uniquely determines the next situation in the branching-time model. Therefore, an action is defined as a connection between two different situations. Having been transformed from a situation to another one, the agent takes the action as a done action in the destination situation.

The branches in the world or in the other words actions emanating from a situation – can be viewed as representing the choices available to the agent at each situation. For example, if there are two possible actions for an agent, then there are two different situations for this agent to go. In fact, an action equals to another action if and only if it connects the same situations.

The system consists of a set of *variables* and a defined set of *domain values* that can be assigned to the variables. Situations are expressed by different *propositional sentences*, which can be true or false in each situation. Each proposition consists of a set of variables while in each situation there is a defined assignment of values to the variables. The *interpreter* calculates truth or falsity of a proposition in a situation considering the value of variables that the proposition has.

We use formalism similar to Computation Tree Logic (CTL) [30]. We evaluate a formula representing a proposition in two different ways: (1) in a specific situation in the branching-time model, (2) in a path, in the branching-time model. We use the model operators of CTL, *inevitable* and *optional*. The modal operator *inevitable* is to be true of a formula at a particular situation if and only if the formula is true of all paths starting from that situation. The operator *optional* is true for a formula at a particular situation in a branching-time model if it is true of at least in one path starting from that situation. We also apply the standard temporal operators $\circ$ (next), $\Diamond$ (eventually) and *Until*. They are defined respectively as the formula is true in the next situation, or finally in a situation, or it is true until another formula becomes true. The modal and the standard operators can be combined in various ways to describe the available choices to the agent.

An agent implicitly knows a proposition if and only if the truth or falsity of the proposition is available for the agent. In the other word, being in the situation, the interpreter can calculate the value of the proposition based on the valued variables that the propositions need to be identified as a valid formula or invalid one. The agent implicitly knows all the valid propositions and accordingly, implicitly does not know all the invalid propositions. An agent is aware of a proposition if and only if it takes it relevant. An agent explicitly knows a proposition, if and only if it is aware of the proposition and it has implicit knowledge to that [9–11]. The question is still remained that how we calculate the awareness [26]. This is addressed, in this paper, by using policy rules.

Policy has long been the topic of interest in network management. However the most of the work have been done in security issues [31]. PAM has borrowed its policy structure from Directory Enabled Networks-next generation (DEN-ng) [12], where a meta-model to implement awareness with policies is given. However DEN-ng does not address the use of policies to identify and obtain awareness, this is proposed by PAM. DEN-ng advises that a policy is a composition of different policy rules, where each rule is defined as an inherence of a propositional sentence with "event-condition-action" semantics. The semantics is such that a rule is triggered when an event occurs. When the condition clause is satisfied, then the modality of action will be applied which may or may not result in executing the action. Twidle et al. [32] define modalities of actions applied in policy rules such as permitting, forbidding, requiring and deterring. Therefore, policy rules can be defined as any of these four types. Defining policy rules as an inherence of proposition gives the agent the opportunity to take a policy rule applicable in a world, if and only if it the agent implicitly knows the policy rule is true. This means the policy rule in that particular world is a true proposition. This idea has been borrowed from [33].

4.2. Formal semantics: Definitions

In this section, we form a set of formal definitions building a logical framework for policy-based awareness in PAM. The PAM framework is supported by different definitions in the literature. In the following, the basic definitions are given.

Definition 1. A model $\mathbf{M}$ of a system with agents $0, 1, \ldots, n$ is a tuple $\mathbf{M} = (\mathbf{M}_0, \mathbf{M}_1, \ldots, \mathbf{M}_n)$.

Definition 2. A model $\mathbf{M}_i$ of agent i from the system is defined to be a tuple, $\mathbf{M}_i = (W, T, S, E, V, D, U, \pi \prec, I, A_i, \prec_i^{Done}, E_i^{Received}, P)$. W is a set of the worlds. T is a set of time-points. Therefore, possible situations in a world $w \in W$ at the time $t \in T$ are represented by $w_t \subseteq W \times T$ which is $w_t = \{s_{t_0}, s_{t_1}, \ldots\}$. In fact, $S = W \times T$. E is a set of events. V is the set of variables, and D is the set of domain values that can be assigned to the variables. U is a set of formulas while each of which represents a propositional sentence. U is closed under $\wedge$ and $\neg$, $\prec$ is a binary relationship on situations that represents a set of actions. As such, $s_{t'}$ describes the situation next after s_t, if and only if there exists an action $a = (s_t, s_{t'}) \in \prec$. A sequence of actions, i.e. $((s_t, s_{t+1}), \ldots, (s_{t+\varepsilon-1}, s_{t+\varepsilon}))$ is called path with the length of ε, which we show it as $(s_t, s_{t+\varepsilon})$. The assignment function π is a relation that maps a variable in a situation to a domain value. More formally, $\pi \in V \times S \times D$. In a situation s_t, for $\nu \in V$ and $d \in D$, we write $\pi_{s_t}^{\nu} = d$. The interpreter I maps a formula from U to a set of true-sets. A true-set for a formula is a set of valued variables that makes the formula true, regardless of the situation. Therefore a formula can be written as $\varphi(\nu_0, \ldots, \nu_r) \in U$ while I_φ shows the true-set for φ. The relation awareness of agent i maps the situation to a sub-set of U. More formally, $A_i \in S \times \{\varphi\colon\ \varphi \in U\}$. The relation done-action of agent i maps the situation to a sub-set of $\prec$. Therefore, $\prec_i^{Done} \in W \times T \times \{a \in \prec\}$. In a situation, the relation received-event of agent i maps the situation to a sub-set of E. As such, $E_i^{Received} \in S \times \{e \in E\}$. P is a set of policy rules.

Definition 3. A world $w \in W$ is a tuple, $w = (T_w, S_w, \prec_w, E_w, P_w)$. In the world w, $T_w \in T$ is a set of time-points, $S_w \in S$ is a set of situations, $\prec_w \in \prec$ is a set of actions, $E_w \in E$ is a set of events. $P_w \in P$ is a set of policy rules.

Definition 4. Consider the model $\mathbf{M}$; the satisfiability relations are defined as below:

$$(\mathbf{M}, s_t) \vDash \varphi(x_1, \ldots, x_r) \quad \text{iff } \left(\pi_{s_t}^{\nu_i}, \ldots, \pi_{s_t}^{\nu_r}\right) \in I_\varphi, \tag{1}$$

$$(\mathbf{M}, s_t) \vDash \neg\varphi \quad \text{iff } (\mathbf{M}, s_t) \nvDash \varphi, \tag{2}$$

$$(\mathbf{M}, s_t) \vDash \varphi \wedge \psi \quad \text{iff } (\mathbf{M}, s_t) \vDash \varphi \text{ and } (\mathbf{M}, s_t) \vDash \psi, \tag{3}$$

$$(\mathbf{M}, s_t) \vDash K_i\varphi \quad \text{iff } (\mathbf{M}, s_t) \vDash \varphi. \tag{4}$$

Therefore, from (4), we can say $K_i\varphi \leftrightarrow \varphi$. Provided (3) and (4), we also can conclude $K_i(\varphi \wedge \psi) \leftrightarrow K_i\varphi \wedge K_i\psi$,

$$(\mathbf{M}, s_t) \vDash A_i\varphi \quad \text{iff } \varphi \in A_i^{s_t}, \tag{5}$$

$$(\mathbf{M}, s_t) \vDash X_i\varphi \quad \text{iff } (\mathbf{M}, s_t) \vDash K_i\varphi \wedge A_i\varphi, \tag{6}$$

$$(\mathbf{M}, s_t) \vDash \prec_i a \quad \text{iff } a \in \prec_i^{Done\, s_t}, \tag{7}$$

$$(\mathbf{M}, s_t) \vDash E_i e \quad \text{iff } e \in E_i^{Received\, s_t}, \tag{8}$$

$$(\mathbf{M}, s_t) \vDash \circ\varphi \quad \text{iff there exists an } s_{t'} \text{ such that } (s_t, s_{t'}) \in \prec \text{ and } (\mathbf{M}, s_{t'}) \vDash \varphi, \tag{9}$$

$$\begin{aligned}(\mathbf{M}, s_t) \vDash \Diamond\varphi \quad &\text{iff there exists a path } (s_t, \ldots, s_{t+m}) \text{ while } (\mathbf{M}, s_{t+m}) \vDash \varphi \\ &\text{and for each } (s_{t+\varepsilon}, s_{t+\varepsilon+1}) \text{ of this path, } (s_{t+\varepsilon}, s_{t+\varepsilon+1}) \in \prec,\end{aligned} \tag{10}$$

$$\begin{aligned}(\mathbf{M}, s_t) \vDash \Diamond_{\leqslant m}\varphi(\nu_0, \ldots, \nu_r) \quad &\text{iff there exists a path } (s_t, \ldots, s_{t+m}) \text{ associated with a set of actions } \prec^m \\ &\text{such that } \left|\prec^m\right| = m \text{ while } (\mathbf{M}, s_{t+m}) \vDash \varphi(\nu_0, \ldots, \nu_r) \text{ and } \prec^d \text{ is the set of each } (s_{t+\varepsilon}, s_{t+\varepsilon+1}) \\ &\text{of this path that } (s_{t+\varepsilon}, s_{t+\varepsilon+1}) \in \prec,\end{aligned} \tag{11}$$

$$\begin{aligned}(\mathbf{M}, s_t) \vDash \varphi\, \mathit{Until}\, \psi \quad &\text{iff there exists a path } (s_t, \ldots, s_{t+m}) \text{ with an action } (s_{t+\varepsilon}, s_{t+\varepsilon+1}) \text{ such that} \\ &(\mathbf{M}, s_{t+\varepsilon}) \vDash \varphi \text{ and } (\mathbf{M}, s_{t+\varepsilon+1}) \vDash \neg\varphi \wedge \psi,\end{aligned} \tag{12}$$

$$\begin{aligned}(\mathbf{M}, s_t) \vDash \varphi\, \mathit{Until}_{\leqslant m}\, \psi, \quad &\text{iff there exists a path } (s_t, \ldots, s_{t+m}) \text{ associated with a set of actions } \prec^m \text{ such that} \\ &\left|\prec^m\right| = m \text{ and also there exists an action } (s_{t+m-1}, s_{t+m}) \text{ such that } (\mathbf{M}, s_{t+m-1}) \vDash \varphi \text{ and } (\mathbf{M}, s_{t+m}) \vDash \neg\varphi \wedge \psi,\end{aligned} \tag{13}$$

$$\begin{aligned}(\mathbf{M}, s_t) \vDash \mathit{inevitable}\,\varphi \quad &\text{iff for each path starting from } w_t \text{ like } (s_t, \ldots, s_{t+m}) \text{ there is an action such as} \\ &(s_{t+m-1}, s_{t+m}) \in \prec \text{ where } (\mathbf{M}, s_{t+m-1}) \vDash \varphi,\end{aligned} \tag{14}$$

$$(\mathbf{M}, s_t) \vDash optional\, \varphi \quad \text{iff there exists at least one path starting from } s_t \text{ like } (s_t, \ldots, s_{t+m})$$
$$\text{such that there is an action } (s_{t+m-1}, s_{t+m}) \in \prec \text{ where } (\mathbf{M}, s_{t+m-1}) \vDash \varphi. \tag{15}$$

Considering the awareness satisfiability relation, some properties of the logic are given below:

$$A_i(\neg\varphi) \leftrightarrow A_i\varphi, \tag{16}$$

$$A_i(\varphi \wedge \psi) \leftrightarrow A_i\varphi \wedge A_i\psi, \tag{17}$$

$$A_i(\varphi \to \psi) \leftrightarrow A_i\varphi, \tag{18}$$

$$A_i K_i\varphi \leftrightarrow A_i\varphi, \tag{19}$$

$$A_i A_i\varphi \leftrightarrow A_i\varphi. \tag{20}$$

Therefore, we can generate the (A_5):

$$A_i X_i\varphi \leftrightarrow A_i\varphi. \tag{21}$$

Definition 5. A policy rule $\rho \in P$ is defined as a formula that belongs to U while forbidding, permitting, deterring and requiring policy rules are defined in the following way:

Forbidding policy rule: A policy rule $\rho \in P$ is forbidding, i.e., $\rho \in P_{Forbidding}$, if and only if there exist a formula $\varphi \in U$, an action $a \in \prec$, and an event $e \in E$ such that

$$\rho = \big(\varphi \wedge E_i e \to inevitable \circ (\neg \prec_i a)\big).$$

Permitting policy rule: A policy rule $\rho \in P$ is permitting, i.e., $\rho \in P_{Permitting}$, if and only if there exist a formula $\varphi \in U$, an action $a \in \prec$, and an event $e \in E$ such that

$$\rho = (\varphi \wedge E_i e \to optional \circ \prec_i a).$$

Deterring policy rule: A policy rule $\rho \in P$ is deterring, i.e., $\rho \in P_{Deterring}$, if and only if there exist a formula $\varphi \in U$, an action $a \in \prec$, and an event $e \in E$ such that

$$\rho = \big(\varphi \wedge E_i e \to optional \circ (\neg \prec_i a)\big).$$

Requiring policy rule: A policy rule $\rho \in P$ is requiring, i.e., $\rho \in P_{Requiring}$, if and only if there exist a formula $\varphi \in U$, an action $a \in \prec$, and an event $e \in E$ such that

$$\rho = (\varphi \wedge E_i e \to inevitable \circ \prec_i a).$$

Note that defining a policy rule as a propositional sentence allows the agent to define a variable ensuring the validity constraint for the policy, such as the period of policy applicability [34]. In fact, the agent implicitly knows a policy rule if and only if it is valid. This is because of the definition of policy rule as a propositional sentence. Therefore, the agent's implicit knowledge can be inferred once the policy rule is valid.

4.2.1. Commitments: Implicit knowledge leading to actions

In this section, we show how implicit knowledge of an agent guides the agent to select an action. The following definition of agent's commitment to an action shows how an agent selects an action considering its current implicit knowledge.

Note that $K_i(\circ\, inevitable \prec a)$ provides us $\circ inevitable \prec a$, which means that the agent is going to do the action a. Therefore, $K_i(\circ\, inevitable \prec a)$ is defined as commitment. In the following, Definition 6 shows how an agent updates its implicit knowledge to commit an action.

Definition 6. An agent i commits to an action $a = (w_t, w_{t'}) \in \prec$ by updating its implicit knowledge following the axiom below:

$$K_i\big(inevitable \Diamond K_i\varphi \vee inevitable \Diamond (\neg K_i\varphi)\big) \wedge K_i\big[optional \circ \big(\prec_i a \wedge K_i(inevitable \Diamond \varphi)\big)\big]$$
$$\wedge (\neg K_i\varphi) \to K_i(\circ\, inevitable \prec_i a).$$

Note that the problem with this definition is that there might be no action that takes the agent to $K_i(inevitable \Diamond \varphi)$ or there might be several actions available. We address this problem later in Section 5.3.

Table 2
Value assignment in different situations – space shuttle Columbia disaster.

Situation	Value assignment to variables	Situation	Value assignment to variables
S_0	$\pi_{s_0}^{\nu_0} = d_3$	S_4	$\pi_{s_4}^{\nu_0} = d_2$, $\pi_{s_4}^{\nu_3} = d_4$ and $\pi_{s_4}^{\nu_4} = d_4$
S_1	$\pi_{s_1}^{\nu_0} = d_2$	S_5	$\pi_{s_5}^{\nu_0} = d_2$, $\pi_{s_5}^{\nu_3} = d_4$, $\pi_{s_5}^{\nu_4} = d_4$, $\pi_{s_5}^{\nu_1} = d_0$, $\pi_{s_5}^{\nu_2} = d_0$, and $\pi_{s_2}^{\nu_5} = d_2$
S_2	$\pi_{s_2}^{\nu_0} = d_2$ and $\pi_{s_2}^{\nu_5} = d_2$	S_6	$\pi_{s_5}^{\nu_0} = d_2$, $\pi_{s_5}^{\nu_3} = d_4$, $\pi_{s_5}^{\nu_4} = d_4$, $\pi_{s_5}^{\nu_1} = d_0$, $\pi_{s_5}^{\nu_2} = d_1$, and $\pi_{s_2}^{\nu_5} = d_3$
S_3	$\pi_{s_3}^{\nu_0} = d_2$, $\pi_{s_3}^{\nu_3} = d_4$ and $\pi_{s_3}^{\nu_4} = d_5$		

4.3. PAM framework and space shuttle Columbia disaster

In this section, we show how PAM framework can model awareness in the space shuttle Columbia disaster (see Table 2).

Let us assume three agents in the system: (1) NASA management, (2) engineering team, and (3) DoD. In this system we define a set of variables as $\{\nu_0, \nu_1, \nu_2, \nu_3, \nu_4, \nu_5\}$ such that ν_0: status of the shuttle, ν_1: physical status for the TPS on the right wing, ν_2: physical status for the TPS on the left wing, ν_3: image request, and ν_4: image approval to engineering team. ν_5: turn-around effect. $\{d_0, d_1, d_2, d_3, d_4, d_5\}$ presents the set of domain variables such that d_0: no damage, d_1: damage, d_2: shake, d_3: fine, d_4: sent, and d_5: pending. The formulas that represent the propositional sentences are given in $\{\varphi_0, \varphi_1, \varphi_2, \varphi_3, \varphi_4\}$ while φ_0: the shuttle is shaking, φ_1: an imaging request has been sent to DoD, φ_2: DoD has approved the images to the engineering team, φ_3: the shuttle is suffering of turn-around effect and φ_4: the shuttle is suffering of damage on its TPS. The interpreter is also defined as $I = \{I_{\varphi_0}, I_{\varphi_1}, I_{\varphi_2}, I_{\varphi_3}, I_{\varphi_4}\}$ where $I_{\varphi_0} = \{((\nu_0, d_2))\}$, $I_{\varphi_1} = \{((\nu_3, d_4))\}$, $I_{\varphi_2} = \{((\nu_4, d_4))\}$, $I_{\varphi_3} = \{((\nu_5, d_2))\}$, and $I_{\varphi_4} = \{((\nu_1, d_1)), ((\nu_2, d_1))\}$ show how values are assigned to the variables in this scenario.

In situation S_5, given $\pi_{s_2}^{\nu_5} = d_2$ and provided $I_{\varphi_3} = \{((\nu_5, d_2))\}$, similar to S_2, we can say $K_{Mng}\varphi_3$. Also, in situation S_5, given $\pi_{s_5}^{\nu_1} = d_0$, $\pi_{s_5}^{\nu_2} = d_0$, we have $K_{Mng}(\neg\varphi_4)$. Therefore, NASA management, in this situation, implicitly knows that the shake is because of turn-around effect and not because of TPS damage. The difference of this situation with S_2 is that the management implicitly knows the shake is not because of TPS damage. However, the NASA management does not know that in S_2.

In situation S_6, given $\pi_{s_5}^{\nu_2} = d_1$, and provided $I_{\varphi_4} = \{((\nu_1, d_1)), ((\nu_2, d_1))\}$ we have $K_{Mng}\varphi_4$. Also, from $\pi_{s_2}^{\nu_5} = d_3$, we conclude $K_{Mng}(\neg\varphi_3)$. Therefore, NASA management, in this situation, implicitly knows that the shake is because of TPS damage and not because of turn-around effect.

If we come back to situation S_1, looking at Fig. 1, in situation S_1, the NASA management has two choices: (I) a_1: announcing the shake as a turn-around effect. (II) a_2: requesting imaging from DoD.

In situation S_1, we have $K_i(\textit{inevitable} \Diamond K_i\varphi_3 \vee \textit{inevitable} \Diamond (\neg K_i\varphi_3))$. While in this situation, $K_{Mng}(\neg\varphi_3)$, we also have $K_{Mng}(\textit{optional} \circ (\prec_i a_1 \wedge K_i\, \textit{inevitable} \Diamond \varphi_3))$. Taking the action commitment definition into account, we can conclude that $K_{Mng}(\circ\, \textit{inevitable} \prec_i a_1)$, which means that the NASA management team commits to announce the shake as a turn-around effect.

We also analyze the commitment of NASA management to request imaging from DoD in situation S_1. Although $\neg K_{Mng}\varphi_4$, the first and the second promise of the action commitment are not satisfied for φ_4. Therefore, the management agent does not commit to a_2 and does not request imaging from DoD.

According to the discussion above when the management agent is in situation S_1 it announces the shake as a turn-around effect. In the following, we introduce the process of awareness identification and show how management agent requests imaging, if becomes aware of φ_4. The process uses policy rules to identify the relevance of information to a situation.

5. Steps towards identifying and obtaining awareness in the PAM

PAM defines the notion of a *policy-aware agent* and proposes a three-step process to (1) recognize relevant policy rules, (2) recognize the relevant information required for invoking the relevant policy rules, and (3) change behavior based on the recognized relevant information. The contribution of PAM is using policy rules as a source to obtain awareness.

5.1. Step one: Recognize relevant policy rules

The objective of this step is to show how agents, given a situation, identify a policy rule as relevant. We define *policy-aware agent* as an agent that identifies awareness of a policy rule when it recognizes that there is a possibility in now or in future that it may violate the rule.

Taken this point of view to policy-aware agent, when the agent is not going to violate a policy rule, the agent simply follows its current implicit knowledge and updates it to the action commitment following Definition 6. However, we are going to show violating a policy rule, changing the agent's awareness updates its implicit knowledge. As such, this might result in committing to a different action.

Note since permitting and deterring are not in force [32], violating the policy rules can only happen in forbidding and requiring policy rules, which is also shown more formally in Theorem 1.

Theorem 1 *(Satisfiability of action commitment).*

(a) $\forall \rho \in P_{Forbidding}$: $K_i(\rho) \wedge K_i(\varphi) \wedge K_i(E_i e) \wedge K_i(optional \circ \prec_i a)$ *is not satisfiable.*
(b) $\forall \rho \in P_{Permitting}$: $K_i(\rho) \wedge K_i(\varphi) \wedge K_i(E_i e) \wedge K_i(optional \circ \neg(\prec_i a))$ *is satisfiable.*
(c) $\forall \rho \in P_{Deterring}$: $K_i(\rho) \wedge K_i(\varphi) \wedge K_i(E_i e) \wedge K_i(optional \circ \prec_i a)$ *is satisfiable.*
(d) $\forall \rho \in P_{Requiring}$: $K_i(\rho) \wedge K_i(\varphi) \wedge K_i(E_i e) \wedge K_i(optional \circ \neg(\prec_i a))$ *is not satisfiable.*

Proof. (a) Provided $K_i(\rho \in P_{Forbidding})$, we can say, $K_i(\varphi \wedge E_i e \to inevitable \circ (\neg \prec_i a))$. Given $K_i(\rho) \wedge K_i(\varphi) \wedge K_i(E_i e)$, we have $K_i(inevitable \circ (\neg \prec_i a))$, which has a clear contradiction with the $K_i(optional \circ \prec_i a)$. Therefore, the given phrase is not satisfiable.

(b), (c) and (d) can be proved in similar ways. □

Taking the fact that only forbidding and requiring policy rules are in force, we now provide the formal definition of policy-aware agent based on these two types of policy rules.

Definition 7. A *policy-aware agent* is an agent who creates its awareness of a policy rule if and only if the agent recognizes a possibility to violate the policy rule sometime now or in the future. There are two situations in which an agent violates a policy rule and accordingly needs to become aware of the rule:

1. The agent becomes aware of a forbidding policy rule, upon which there is a possibility sometime now or in the future that the agent will implicitly know that (a) the policy rule as a property is valid, (b) the event has been received, (c) the policy's condition is satisfied and (d) there will be an option to do the forbidden action given in the policy rule. In such a situation, there is a possibility to violate the policy rule.
   Therefore, $\forall \rho \in P_{Forbidding}$:
   $$A_i \rho \, Until\big[\neg optional \lozenge \left(K_i \rho \wedge K_i \varphi \wedge K_i E_i e \wedge K_i(optional \circ \prec_i a)\right)\big].$$
2. The agent becomes aware of a requiring policy rule, upon which there is a possibility sometime now or in the future that the agent will implicitly know that (a) the policy rule is valid, (b) the event has been received, (c) the policy's condition is satisfied, and (d) there will be an option to *not* do the required action given in the policy rule. In such a situation, there is a possibility to violate the policy rule.
   Therefore, $\forall \rho \in P_{Requiring}$:
   $$A_i \rho \, Until\big[\neg optional \lozenge \left(K_i \rho \wedge K_i \varphi \wedge K_i E_i e \wedge K_i\big(optional \circ \neg(\prec_i a)\big)\right)\big].$$

In the definition above for policy-aware agent, we consider the commitment phrases, which Theorem 2 proved that they are not satisfiable. In fact, $K_i(optional \circ \prec_i a)$ is not satisfiable for forbidding policy rules and so $K_i(optional \circ \neg(\prec_i a))$ for requiring policy rules.

5.1.1. Recognize relevant policy rules in the space shuttle Columbia disaster

In Section 2, we mentioned a policy available at the time of the disaster. The policy rule says when an aircraft experiences unusual shakes, if there is any TPS damage, the spacewalk procedure must be granted. The formalization of this policy rule in the PAM framework is $\rho = (\varphi_0 \wedge E_{Mng} e_0 \to inevitable \circ \prec_{Mng} a_5)$. Assume that the policy rule ρ is valid in all the situations, so $K_{Mng}\rho$ is valid as well. By Definition 5, the policy rule is requiring, as such and by Theorem 1, it is eligible to maintain the NASA management's awareness. By Definition 7(2), the agent becomes aware of ρ, in both paths: (S_1, S_3, S_4, S_5) and (S_1, S_3, S_4, S_6).

5.2. Step two: Recognize the relevant information in the relevant policy rules

The objective of this step is to show how agents become aware of relevant proposition that is required to invoke the policy rule. A policy-aware agent identifies information awareness about the conditions referenced in the rule until it is aware of that policy rule. Theorem 2 shows how policy awareness in agents renders them aware of relevant information that is required to invoke the policy rule.

Theorem 2 *(Awareness to the policy rule's condition). A policy-aware agent creates information awareness from the conditions referenced in a forbidding or requiring rule, until it is aware of that policy rule*:

$$\forall \rho \in P_{Forbidding} \cup P_{Requiring}: \quad A_i \varphi \, Until \neg A_i \rho.$$

Proof. First, we prove the theorem for forbidding policy rules. Taking the definition of *Until* into account, we can say $A_i \rho \, Until \neg A_i \rho$. By replacing the policy rule ρ with its definition in the promise, we have $A_i(\varphi \wedge E_i e \to inevitable \circ (\neg \prec_i a)) \, Until \neg A_i \rho$. For requiring policy rules, we can prove the theorem in the similar way. □

According to the properties of the logic given in (A_4), (A_5) and (A_6), if an agent is aware of other agents' awareness, or aware of implicit knowledge or explicit knowledge about a propositional sentence, then the agent will be aware of that sentence. Therefore, if an agent is aware the fact that another agent, in the system, identifies a policy rule as relevant rule, i.e., awareness of awareness, then it will be aware of the condition of that particular policy rule. This awareness can be useful when the system involves different agents. Theorem 3 formalizes this claim.

Theorem 3 *(Awareness under the other's awareness and knowledge). A policy-aware agent becomes aware of the condition for a policy rule which it is aware of the other's awareness, implicit or explicit knowledge to that rule.*

(a) $A_i\varphi\, \mathit{Until} \neg A_i A_j \rho$.
(b) $A_i\varphi\, \mathit{Until} \neg A_i K_j \rho$.
(c) $A_i\varphi\, \mathit{Until} \neg A_i X_j \rho$.

Proof. (a) Replacing $A_i A_j \rho$ with $A_i \rho$ in Theorem 3 results in (a).
Cases (b) and (c) can be proved in the similar way. □

This is actually an approach to find out the awareness about the other individuals' awareness or knowledge. In fact, based on our previous definitions borrowed from the logic of general awareness, awareness about awareness or knowledge of information can infer the awareness of that information.

5.2.1. Recognizing the relevance information in the relevant policy rules for the space shuttle Columbia disaster

As we discussed in Section 5.1.1, NASA management, applying Step 1, becomes aware of the policy rule ρ all the way in both paths (S_1, S_3, S_4, S_5) and (S_1, S_3, S_4, S_6). By Theorem 2, we can conclude that management agent becomes aware of φ_0 in these situations. In the next step, we show how this awareness results in committing the management agent to a_2.

5.3. Step three: Change behavior based on recognized relevant information

The logic, so far, treated policy-based awareness as a mental attitude of agents, which indicates the relevance of a proposition to the situation. However, we have not formalized how awareness guides or determines the agent's future knowledge. In the other words, we have not discussed how the agent's current awareness leads to its future implicit knowledge and how it results in selecting an action.

An alternative is to look at the relationship between current awareness and future implicit knowledge as what we can think of strategies for knowledge update. Different types of agents will have different types of strategies. In answer to the aforementioned problem of agents' commitments in Section 4.2.1, we propose three different strategies: Volitional Agent, Cautious Volitional Agent and Hasty Volitional Agent. We will call the last two ones shortly as Cautious and Hasty.

Being aware of a propositional sentence, the volitional agent implicitly knows that eventually and inevitably, it will have implicit knowledge that the proposition is true or false. There are two problems with this strategy:

1. There might be no path in the branching-time model that takes the agent to such implicit knowledge. Therefore, we define Cautious Volitional Agent or in short Cautious Agent. In this strategy, being aware of a proposition, the cautious agent checks the possibility of achieving implicit knowledge to the proposition or its negation and then updates its implicit knowledge to eventually and inevitably have implicit knowledge to the proposition or its negation.
2. There might be several paths in the branching-time model that take the agent to such implicit knowledge. Therefore, we define Hasty Volitional Agent or in short Hasty Agent. In this strategy, being aware of a proposition, the hasty agent updates its implicit knowledge that it will eventually comes to the implicit knowledge of the proposition in the shortest path possible.

Hence, in the following we give the formal definition of each of these proposed strategies:

Definition 8. Provided $A_i\varphi$, we define three types of agents:

(1) Volitional Agent:

$$A_i\varphi \to \big[\mathit{inevitable}\big(K_i(\mathit{inevitable} \lozenge K_i\varphi)\, \mathit{Until}\, K_i\varphi\big)\big] \vee \big[\mathit{inevitable}\big(K_i\big(\mathit{inevitable} \lozenge (\neg K_i\varphi)\big)\mathit{Until}(\neg K_i\varphi)\big)\big].$$

(2) Cautious Agent:

$$\begin{aligned} A_i\varphi \to {} & \big[\mathit{inevitable}\big(K_i(\mathit{inevitable} \lozenge K_i\varphi)\, \mathit{Until}\big(\neg K_i(\mathit{optional} \lozenge K_i\varphi)\big) \vee K_i\varphi\big)\big] \\ & \vee \big[\mathit{inevitable}\big(K_i\big(\mathit{inevitable} \lozenge (\neg K_i\varphi)\big)\, \mathit{Until}\big(\neg K_i\big(\mathit{optional} \lozenge K_i(\neg\varphi)\big)\big) \vee (\neg K_i\varphi)\big)\big]. \end{aligned}$$

(3) Hasty Agent: $\forall d_{\min} \in N$ such that

$$A_i\varphi \to \big[\mathit{inevitable}\big(K_i(\mathit{inevitable} \Diamond K_i\varphi)\, \mathit{Until}_{\leqslant d_{\min}} \big(\neg K_i(\mathit{optional} \Diamond K_i\varphi)\big) \vee K_i\varphi\big)\big]$$
$$\vee \big[\mathit{inevitable}\big(K_i\big(\mathit{inevitable} \Diamond (\neg K_i\varphi)\big)\, \mathit{Until}_{\leqslant d_{\min}} \big(\neg K_i\big(\mathit{optional} \Diamond K_i(\neg\varphi)\big)\big) \vee (\neg K_i\varphi)\big)\big]$$

there exists no $d < d_{\min}$ such that

$$A_i\varphi \to \big[\mathit{inevitable}\big(K_i(\mathit{inevitable} \Diamond K_i\varphi)\, \mathit{Until}_{\leqslant d} \big(\neg K_i(\mathit{optional} \Diamond K_i\varphi)\big) \vee K_i\varphi\big)\big]$$
$$\vee \big[\mathit{inevitable}\big(K_i\big(\mathit{inevitable} \Diamond (\neg K_i\varphi)\big)\, \mathit{Until}_{\leqslant d} \big(\neg K_i\big(\mathit{optional} \Diamond K_i(\neg\varphi)\big)\big) \vee (\neg K_i\varphi)\big)\big].$$

A volitional agent reaches an identical conclusion only if she continues to implicitly, until the time it has implicit knowledge that it has realized what was relevant to its situation, i.e. awareness. As such, being aware of a proposition updates the agent's implicit knowledge that it will eventually and inevitably has implicit knowledge to the proposition or its negation. Similarly, a cautious agent has the same knowledge if it is possible. A hasty agent also has this knowledge that it implicitly, inevitably and in the shortest path knows the aware proposition sometime now or in future. This will knowledge update solves the problem mentioned in action commitment. More formally, we have the following theorem for different types of agents. The theorem shows how awareness leads to update the agent's implicit knowledge.

Theorem 4 *(Awareness leading to knowledge update).*

(1) *Volitional Agent*:

$$A_i\varphi \to K_i\big(\mathit{inevitable} \Diamond K_i\varphi \vee \mathit{inevitable} \Diamond (\neg K_i\varphi)\big).$$

(2) *Cautious Agent*:

$$A_i\varphi \wedge \big[\big(\mathit{inevitable}(K_i\, \mathit{optional} \Diamond K_i\varphi\, \mathit{Until}\, K_i\varphi)\big) \vee \big(\mathit{inevitable}\big(K_i\, \mathit{optional} \Diamond K_i(\neg\varphi)\, \mathit{Until}\, K_i(\neg\varphi)\big)\big)\big]$$
$$\to K_i\big(\mathit{inevitable} \Diamond K_i\varphi \vee \mathit{inevitable} \Diamond (\neg K_i\varphi)\big).$$

(3) *Hasty Agent*: $\forall d_{\min} \in N$ *such that*

$$A_i\varphi \wedge \big[\big(\mathit{inevitable}(K_i\, \mathit{optional} \Diamond K_i\varphi\, \mathit{Until}_{\leqslant d_{\min}}\, K_i\varphi)\big) \vee \big(\mathit{inevitable}\big(K_i\, \mathit{optional} \Diamond K_i(\neg\varphi)\, \mathit{Until}_{\leqslant d_{\min}}\, K_i(\neg\varphi)\big)\big)\big]$$
$$\to K_i\big(\mathit{inevitable} \Diamond_{\leqslant d_{\min}} K_i\varphi \vee \mathit{inevitable} \Diamond (\neg K_i\varphi)\big).$$

there exists no $d < d_{\min}$ *such that*

$$A_i\varphi \wedge \big[\big(\mathit{inevitable}(K_i\, \mathit{optional} \Diamond K_i\varphi\, \mathit{Until}_{\leqslant d}\, K_i\varphi)\big) \vee \big(\mathit{inevitable}\big(K_i\, \mathit{optional} \Diamond K_i(\neg\varphi)\, \mathit{Until}_{\leqslant d}\, K_i(\neg\varphi)\big)\big)\big]$$
$$\to K_i\big(\mathit{inevitable} \Diamond_{\leqslant d} K_i\varphi \vee \mathit{inevitable} \Diamond (\neg K_i\varphi)\big).$$

Proof. (1) Assume the promise $A_i\varphi$. By Definition 7(1), we can conclude to the following axiom:

$$\big[\mathit{inevitable}\big(K_i(\mathit{inevitable} \Diamond K_i\varphi)\, \mathit{Until}\, K_i\varphi\big)\big] \vee \big[\mathit{inevitable}\big(K_i\big(\mathit{inevitable} \Diamond (\neg K_i\varphi)\big)\, \mathit{Until}(\neg K_i\varphi)\big)\big].$$

By the definition of *Until*, the axiom $\varphi\, \mathit{Until}\, \psi$ interferes $\mathit{inevitable} \Diamond \psi$ [34]. Therefore, we can say $\mathit{inevitable} \Diamond K_i\varphi \vee \mathit{inevitable} \Diamond (\neg K_i\varphi)$. By (4), we conclude to $K_i(\mathit{inevitable} \Diamond K_i\varphi \vee \mathit{inevitable} \Diamond (\neg K_i\varphi))$.

Cases (2) and (3) follow the similar proof. □

So far, we have shown how an agent who is aware of a proposition updates its implicit knowledge. Here, in Theorem 5, we take Definition 6 into consideration to see how the agent commits to an action by being aware of a proposition, which is the purpose of this step.

Theorem 5 *(Awareness leading to action commitment).*

(1) *Volitional Agent*:

$$A_i\varphi \wedge K_i\big[\mathit{optional} \circ \big(\prec_i a \wedge K_i(\mathit{inevitable} \Diamond \varphi)\big)\big] \wedge (\neg K_i\varphi) \to K_i(\circ\, \mathit{inevitable} \prec_i a).$$

(2) *Cautious Agent*:

$$A_i\varphi \wedge \big[\big(\mathit{inevitable}(K_i\, \mathit{optional} \Diamond K_i\varphi\, \mathit{Until}\, K_i\varphi)\big) \vee \big(\mathit{inevitable}\big(K_i\, \mathit{optional} \Diamond K_i(\neg\varphi)\, \mathit{Until}\, K_i(\neg\varphi)\big)\big)\big]$$
$$\wedge K_i\big[\mathit{optional} \circ \big(\prec_i a \wedge K_i(\mathit{inevitable} \Diamond \varphi)\big)\big] \wedge (\neg K_i\varphi) \to K_i(\circ\, \mathit{inevitable} \prec_i a).$$

(3) *Hasty Agent*: $\forall d_{\min} \in N$ *such that*

$$A_i\varphi \wedge \big[\big(inevitable(K_i\, optional \Diamond K_i\varphi\, Until_{\leqslant d_{\min}} K_i\varphi)\big) \vee \big(inevitable\big(K_i\, optional \Diamond K_i(\neg\varphi)\, Until_{\leqslant d_{\min}} K_i(\neg\varphi)\big)\big)\big] \wedge K_i\big[optional \circ \big(\prec_i a \wedge K_i(inevitable \Diamond_{\leqslant d_{\min}} \varphi)\big)\big] \wedge (\neg K_i\varphi) \rightarrow K_i(\circ\, inevitable \prec_i a)$$

there exists no $d < d_{\min}$ *such that*

$$A_i\varphi \wedge \big[\big(inevitable(K_i\, optional \Diamond K_i\varphi\, Until_{\leqslant d} K_i\varphi)\big) \vee \big(inevitable\big(K_i\, optional \Diamond K_i(\neg\varphi)\, Until_{\leqslant d} K_i(\neg\varphi)\big)\big)\big] \wedge K_i\big[optional \circ \big(\prec_i a \wedge K_i(inevitable \Diamond_{\leqslant d} \varphi)\big)\big] \wedge (\neg K_i\varphi) \rightarrow K_i(\circ\, inevitable \prec_i a).$$

Proof. (1) Assume the promise $A_i\varphi$. By Theorem 4, we can conclude to $K_i(inevitable \Diamond K_i\varphi \vee inevitable \Diamond (\neg K_i\varphi))$. Provided the following promise $K_i[optional \circ (\prec_i a \wedge K_i(inevitable \Diamond \varphi))] \wedge (\neg K_i\varphi)$ and according to Definition 6, we can conclude $K_i(\circ\, inevitable \prec_i a)$.

Cases (2) and (3) follow the similar proof. □

5.3.1. Change of behavior based on awareness in the space shuttle Columbia disaster

In order to simplify the problem, we choose NASA management to be a volitional agent. Having had $A_i\varphi_0 \wedge K_i[optional \circ (\prec_i a_2 \wedge K_i(inevitable \Diamond \varphi_0))] \wedge (\neg K_i\varphi_0)$, by Theorem 5, we can conclude to $K_i(\circ\, inevitable \prec_i a_2)$. Therefore, the management agent commits to a_2 and requests imaging from DoD.

All in all, when the unusual shakes happened in the shuttle the TPS damage as the condition of the policy rule was relevant information that NASA overlooked. This oversight resulted in the failure to ask the DoD for the high-resolution images.

6. Evaluation

Our approach in evaluation of PAM is triangulation [35] of (I) lab simulations and (II) case study of wireless communication system at St. Olavs Hospital, Trondheim University Hospital, Norway.

6.1. Simulations: Hypothetical experiments

We opted to conduct a simulation study on PAM, since we intend to see the efficiency and efficacy of this method by large-scale change in different configuration parameters, which will be introduced later on in this section. In this section, we apply PAM in hypothetical worlds and see the behavior of a system that uses PAM.

6.1.1. Experimental settings

In our experiments, we used a java program to randomly generate 1000 different branching-time models including randomized events and propositions. While generating the worlds, the random function used different random seeds. Having generated the worlds, the program selected a situation for each model as success with a given initiative event. Therefore, in a random basis some of these models needed recognizing the relevance of propositions, i.e. awareness and some did not. Then, the program generated 200 policy rules based on the set of all propositions, event and actions. We ran the simulation on a computer with 2.33 GHz Intel dual-core CPU, 2 GB of RAM, and professional Windows XP. Fig. 2 shows the overall settings in the experiments.

6.1.2. Methodology

We defined different configuration parameters, which affect on the success rate of the system. By varying each of these parameters, a number of the system conditions such as cost-efficiency and efficacy can be simulated.

Success rate, i.e. S is defined as the percentage that agents achieve a specific situation given as a pre-defined input to the experiments. We compare developing an agent system using PAM against ignoring awareness. In each experiment, we measure attained success rate and spent cost in PAM approach compared to ignoring awareness. In fact, the more actions the agents perform, the cost of the system will increase. This is how we measured the success. Table 3 shows the configuration factors. Based on these factors, we designed three different simulations. In each simulation, we categorized the worlds in different groups. We ran the simulations by taking different worlds or policy rules randomly and we calculated the average rate for the success factor and cost. According to the discussion made in the following, we also measured the cost-efficiency and efficacy of updating awareness ignorance to PAM.

Efficacy is a comparative measure. The efficacy of PAM characterizes the correspondence between the success rates obtained by using PAM and by awareness ignorance. The efficacy, i.e. ς is defined as the ratio between attained success rate in PAM and the success rate by ignoring awareness, i.e. see (22):

$$\varsigma_{PAM} = \frac{S_{PAM}}{S_{Awareness\,ignorance}}. \tag{22}$$

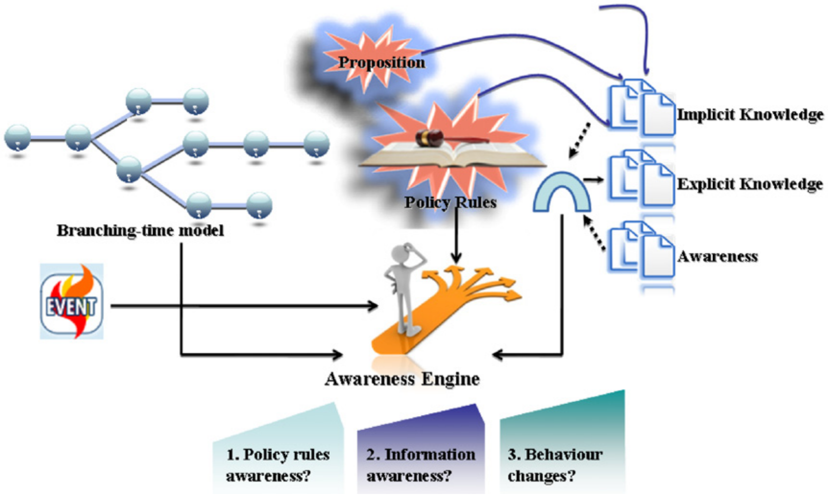

Fig. 2. Experiment settings.

Table 3
Configuration parameters.

Name	Description
Size	The number of situations in the branching-time model of the world.
Complexity	The arrangement of the situations meaning how flat the branching-time model of the world looks like. We measure the complexity by the maximum number of the horizontal situations, i.e. situations that have the same distance (number of actions) from initial situation. Therefore, the more horizontal situations that the branching-time model has, the more complex the world is.
Availability of policy rules	The number of the policy rules that have been triggered by events and actions involved in the world.

Comparing PAM and awareness ignorance approach, the cost-efficiency, i.e. ε is defined as the extra success rate that we obtain by spending extra cost, i.e. see (23). Increasing the success rate and decreasing the cost increase the cost-efficiency of the system. In each experiment, study the cost-efficiency of updating the standard approach to PAM by varying the configuration parameters:

$$\varepsilon = \frac{S_{PAM} - S_{Awareness\,ignorance}}{C_{PAM} - C_{Awareness\,ignorance}}. \tag{23}$$

Readers wishing more details about efficacy and cost-efficiency are recommended to [36] where we borrowed our definitions.

6.1.3. Results

In this section, we present the results of our experiments. In each experiment, we chose a configuration parameter and changed it to see its effect on success rate, cost, efficiency, and efficacy of PAM. This gives us a picture on how and when to use PAM. In the following, we present the results in terms of the aforementioned configuration parameters.

Size. The objective of this experiment was to see how the size of worlds affects on PAM. We measured the size of the worlds by the number of the situations involved in them. In our simulation, the size was in the range of 3 to 184 situations. We categorized the branching-time model based on their size and put the worlds with the same size in the same group. We repeated the experiment for each group 100 times. In each time, we randomly selected one of the worlds involved in the related group. We ran the system once with PAM and once without awareness. The results are presented in Fig. 3.

Complexity. We define the complexity of a world as the number of different situations involved in its branching-time model, which the agent can reach in an equal distance (i.e. number of actions) from the initial situation. Therefore, we measure the complexity of the world by the maximum number of the horizontal situation. Horizontal situations are situations with the same distance from the initial situation. In this experiment, we categorized the branching-time models in regard to their complexity. We could find branching-time models that were flat, while there were also models with 34 horizontal situations. We repeated the experiment, for each specific complexity, 100 times. We ran the system with awareness approach and by ignoring awareness. The results are presented in Fig. 4.

Availability of the policy rules. Having had the situations, propositions, events, and worlds, the java program generated 200 policy rules. We ran the system developed by PAM with zero policy rules – which is equal to awareness ignorance, with one policy rule, with two policy rules and so on to 200 policy rules. In each time, we ran 100 selected branching-time models from the 1000 generated models and recorded the success of system and the number of navigated situations as cost. Then,

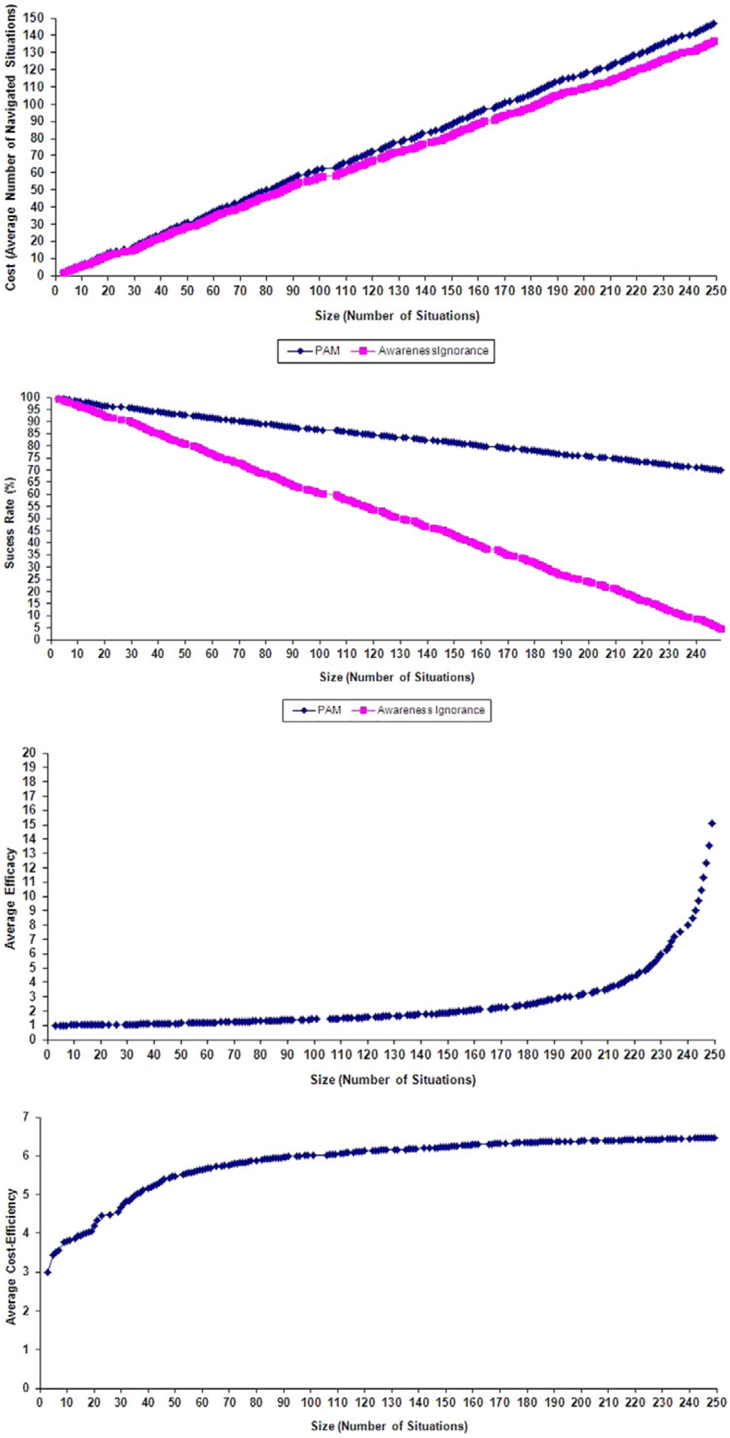

Fig. 3. Size: Comparing PAM with awareness ignorance approach.

we calculated the success rate and average cost. These experiments evaluate the effect of policy rules. We operated these experiments only for PAM and we can consider PAM with zero policy rules to be the same as the awareness ignorance approach. In this section, we discuss the change of success rate by increasing the number of policy rules. Since, PAM with no policy rule is similar to awareness ignorance; the efficacy of updating awareness ignorance method to PAM can be calculated by running PAM with no policy rule as awareness ignorance approach. Therefore, the efficacy curve, in terms of growth by increasing the number of policy rules, has the same shape as success rate curve. In fact, at each point of interest

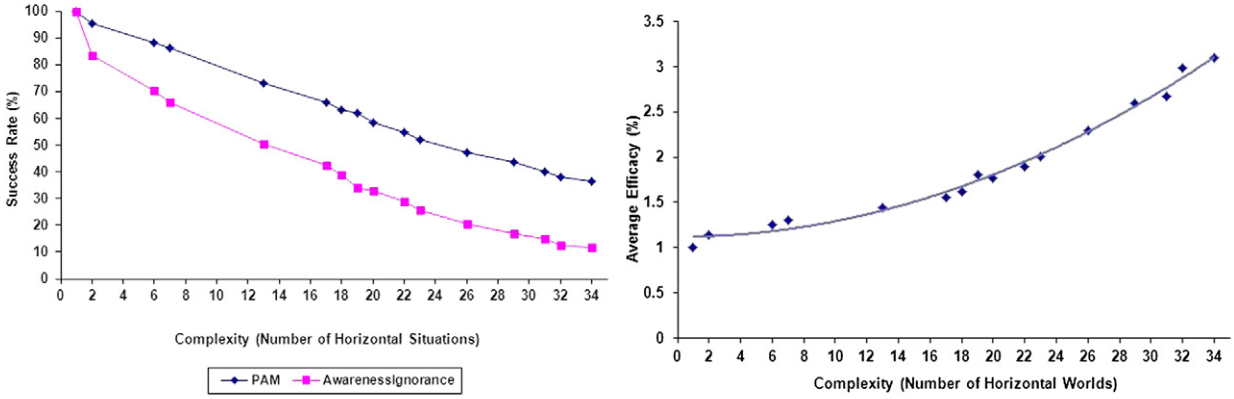

Fig. 4. Complexity: Comparing PAM with awareness ignorance approach.

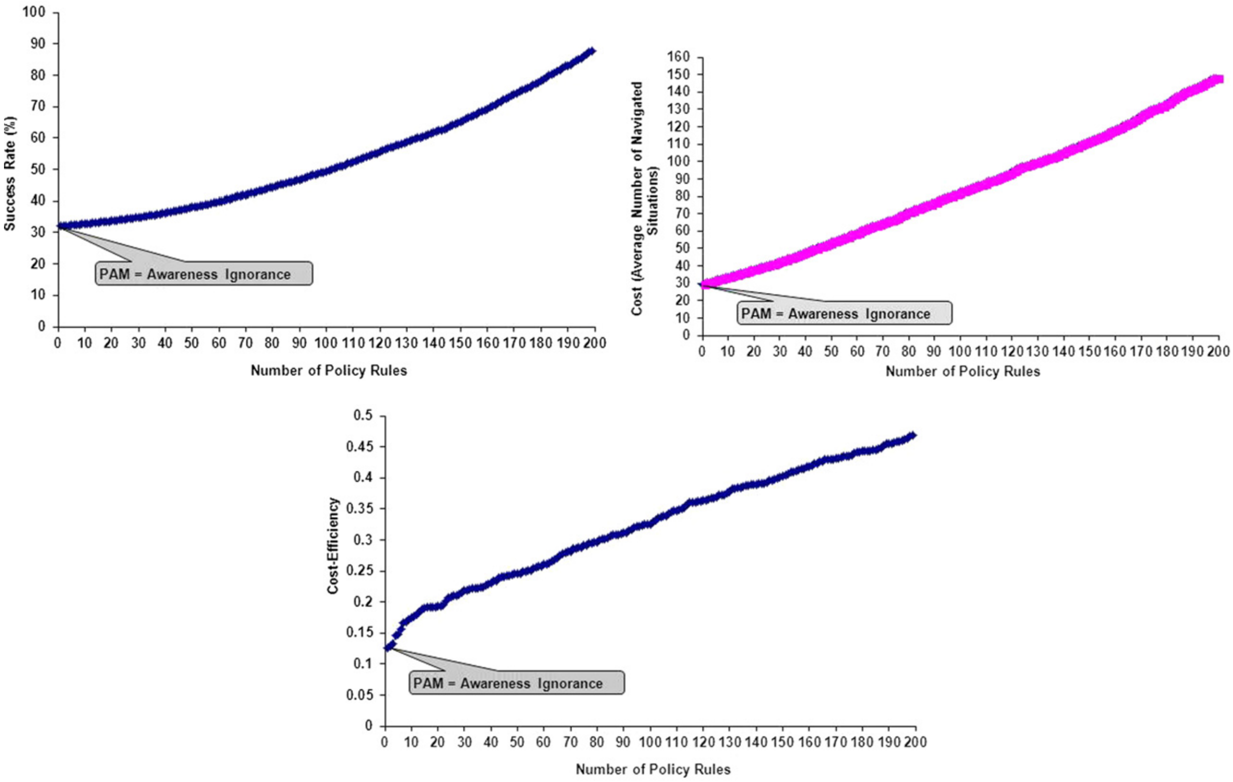

Fig. 5. Number of policy rules: Comparing PAM with ignoring awareness.

in the curve, average success rate of PAM will be divided by a constant number as average success rate of PAM with no policy rule (i.e. awareness ignorance). Therefore, updating awareness ignorance to PAM becomes more effective with more policy rules. The results are presented in Fig. 5.

6.2. Case study: Experiments on the wireless communication system at St. Olavs Hospital

In this section, we present an interpretive case study [37] conducted at St. Olavs Hospital, Trondheim University Hospital, Norway. We found physicians were being called by wireless devices when they should have not been interrupted. In this section, we are going to apply PAM to develop a cooperative management application for wireless communication at the hospital for experimental purposes.

The experiments were based on the different scenarios obtained from data collection in the hospital. The methodology for the experiments and the configuration parameters followed the methodology for the hypothetical experiments, explained in Section 6.1.

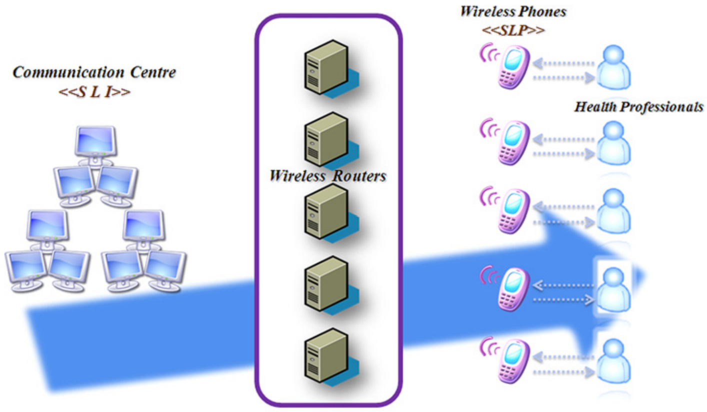

Fig. 6. Intelligent wireless communication system at St. Olavs Hospital.

6.2.1. *Data collection*

In order to collect data for our experiments, one of the authors moderated an interpretive data collection [37] at St. Olavs Hospital. This consisted of participatory observations, non-structured and mostly ad hoc interviews, and discussions. The study was conducted among a selected group of physicians at various levels of hierarchy and roles, within two clinics at the hospital. Regarding sensitive information gathered during the study, a non-discloser agreement had been signed before data collection.

Observations. The moderator followed the independent work of 11 physicians at the clinics, for a total of 135 hours between May and June 2009. In order to have a more realistic picture of the communication system at the hospital, the moderator took the role of a first year medical student, dressed and acted like a physician to blend in as much as possible for a more realistic picture of the communication situation at the clinic. He followed each physician in their everyday work for at least two workdays/nights/duties. The moderator contacted each physician to make an appointment in the morning meeting at each clinic. The moderator recorded every call/page/message, type of device, reaction and physicians' situation.

Interviews and discussions. During the observation at each clinic, the moderator had an open office with other assistant physicians. This gave rise to opportunities for several discussions on communication scenarios. The moderator did not use any pre-defined interview guidelines. However, he asked questions related to what he had observed. The moderator asked two types of questions from interviewees; (1) specific questions to the role of interviewees, (2) similar questions to everybody and then compared the answers.

Here, our focus is on wireless communication among health professionals. The moderator, during the data collection, mainly on his observations found 43 different time-branching models and 24 policy rules related to interruption via the wireless communication in the clinics.

6.2.2. *Deployment of PAM*

PAM can be developed using intelligent communication systems such as the one installed in the hospital – Cisco Call Manager. The objective of intelligent communication systems is to provide communications processing that includes (a) a Service Logical Program (SLP) to receive users' requests and (b) a Service Logical Interpreter (SLI) to execute the users' requests and return the results to the users. The SLP and SLI can be implemented by software agents. This implementation can benefit from using PAM to recognize the relevance of information and request it from the users (see Fig. 6). At St. Olavs Hospital, the SLI can be implemented on top of the Cisco Call Manager server, which stores information such as availability of physicians in the hospital (see Fig. 6). The SLP can be implemented on the wireless phones. In the following section, we explain a simple sample scenario to clarify the experimental settings.

Sample scenario. The following scenario occurred, during one of the observations of a physician while examining a patient. The physician was in a sterile dress and gloves. When the physician received a call from a nurse, the moderator observed that the physician stopped the examination, took off his gloves, and answered the phone. Then, after finishing the call, he washed his hands and he treated them in antibacterial liquid again, took the new gloves on, and started the examination, all over again. This scenario is presented in Fig. 7.

The moderator asked the physician for his opinion about the call. The call was about a patient's medication and could have answered by another physician. The physician believed that most of the phone calls that he receives during his examinations can be answered by his available colleagues. As a matter of fact, the physician was not available and the call should have been diverted to another physician. To do so, the nurse should have asked for the availability of the physician (w_2) from the communication center, before calling.

Looking at the system in S_1, PAM proposes that the nurse agent should take the following three steps to obtain awareness:

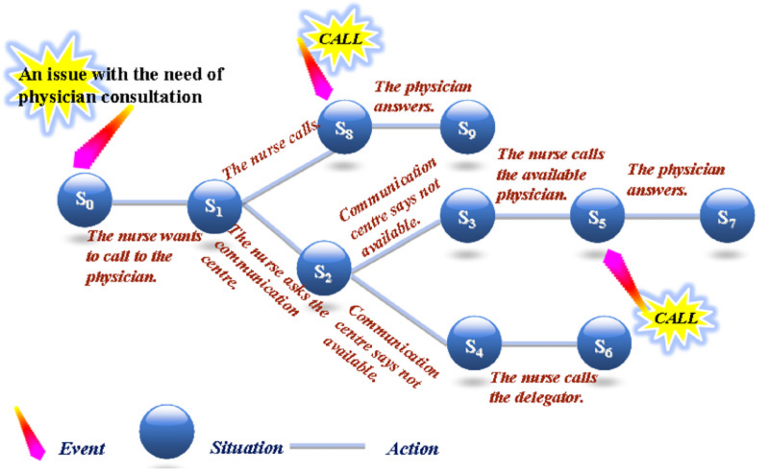

Fig. 7. World in the sample scenario.

1. *Recognize relevant policy rules*: The nurse agent implicitly knows that the physician is required. However, the agent does not know whether the physician is available or not. Therefore, if it simply calls the physician, it goes to S_8. Looking at the branching-time model and the nurse, the agent finds out that there is a possibility at S_4 that it implicitly knows about the physician's unavailability. Therefore, there is a possibility for the agent to do the forbidden action involved in a mentioned policy rule. The policy rule says once an issue comes which requires a physician consultation, if the physician is not available; the nurse must not call the physician. Therefore, calling to the physician when he/she is not available is forbidden and breaks the policy rule. As such, the agent takes this policy rule as relevant and becomes aware of that (see Fig. 7).
2. *Recognize the relevant of information*: Being aware of the policy rule, the nurse agent needs to implicitly know the conditions for the rule, which makes the availability of the physician relevant. Therefore, the nurse agent needs to become aware of the physician's availability.
3. *Change behavior based on recognized relevant information*: While the nurse is aware that the physician is not available, the nurse agent needs to implicitly know whether "the physician is not available" is a true or false sentence. Therefore, it takes an action that eventually brings it to this knowledge. The nurse agent asks the communication center and goes to S_2. This is because that the truth or falsity of the sentence is given in S_4 and S_3 which are accessible from the path begins from S_2.

Fig. 8 shows the mentioned branching-time model as an example on how a PAM-supported system can be deployed on the Cisco Call Manager at the hospital. Everyday, the physicians give their availability schedule to the communication center via their wireless phones. Then, once a nurse dials the physician, the wireless phone checks the availability of the system with the SLI agent sitting on the server at communication center. If the physician is available, the nurse SLP agent will continue dialing. If the physician is not available, then the nurse agent based on the uploaded delegator list calls the delegator.

6.2.3. Results

We ran 43 branching-time models with 24 policy rules obtained from data collection while SLP agents were implemented using PAM and awareness ignorance approach. We studied the behavior of the system comparing these two approaches. Tables 4, 5 and 6 present the average cost and success rate for each of the configuration parameters. Figs. 9, 10 and 11 also show the efficacy and cost-efficiency of updating awareness ignorance to PAM in the wireless communication system at the hospital.

Table 4 shows that following what we have found in our simulations, the success rate in both approaches decreased in wireless communication system of the hospital, while size was increasing. However, the average cost increased by increasing the size. Fig. 9 shows while increasing the size of branching-time models, the efficacy and cost-efficacy of updating SLI agents to use PAM increased in wireless communication system at the hospital. Having plotted the trend chart using Analysis of Variance (ANOVA) [38], Fig. 9 exhibits the similar trend in our hypothetical simulations and the case of wireless communication system of the hospital.

Table 5 shows that increasing the complexity of the system in both approaches, PAM and awareness ignorance, has a negative impact on success rate of the wireless communication of the hospital. However, Fig. 10 exhibits the higher rate of dropping in PAM compared to awareness ignorance. This also confirms by what we found in our hypothetical simulations. Fig. 10 shows while updating SLI agents to use PAM, the efficacy trend, plotted using ANOVA – improves. Comparing the

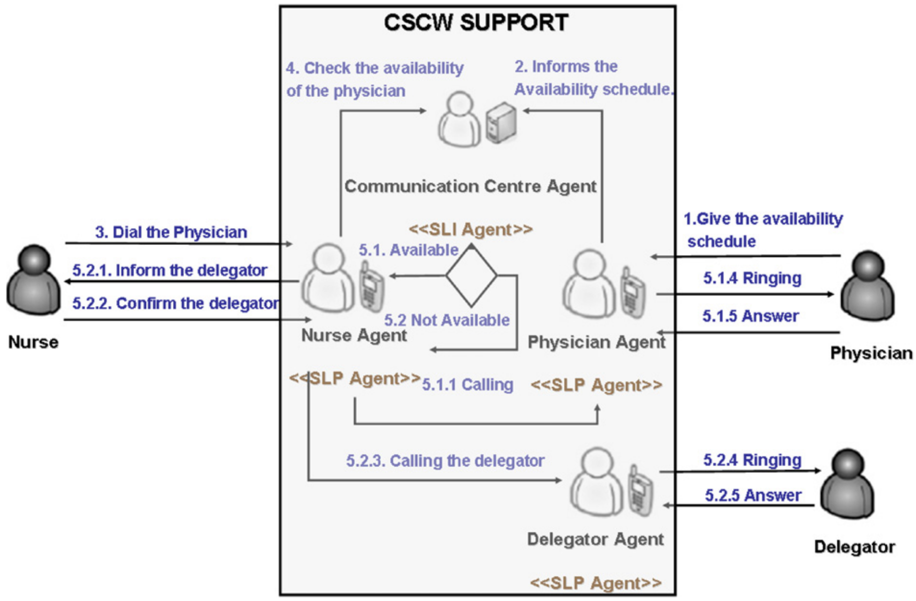

Fig. 8. Deployment of the PAM.

Table 4
The results of the experiments in the hospital related to size.

Size	Ignoring awareness		PAM	
	Average cost	Success rate (%)	Average cost	Success rate (%)
4	3.22	86.61	3.07	87.33
8	5.17	79.92	4.61	82.66
12	6.2	72.33	5.22	77.31
14	7.33	59.28	5.9	67.36
15	7.51	48.74	6	57.74
16	7.83	38.93	6.23	49.93
17	8.11	35.86	6.39	48.99
19	8.84	24.41	6.97	39.86
20	9.4	16.9	7.54	32.51
22	10.17	9.36	8.32	25.30
23	10.43	5.96	8.79	21.52
24	11.27	4.23	9.48	21.29
25	11.82	4.01	10.13	21.09
26	12.26	3.5	10.72	20.90

Table 5
The results of the experiments in the hospital related to complexity.

Complexity	Ignoring awareness	PAM
	Success rate (%)	Success rate (%)
1	100	100
2	69.92	82.66
3	60.56	77.31
4	41.12	64.32
5	22.92	50.34
6	10.96	38.2
7	4.45	23.45

impact of complexity on efficacy in hypothetical simulations with experimenting wireless communication system at St. Olavs Hospital, we can see the similar growth.

Table 6 shows the results of our experiment at St. Olavs Hospital. This table highlights the finding of our hypothetical simulations stating that increasing the number of policy rules lifts up the success rate as well as the average cost. Plotting the ANOVA trend for efficacy and cost-efficiency in Fig. 11 shows the support of the finding in the hospital by the efficacy and cost-efficiency improvement in the hypothetical simulations.

Table 6
The results of the experiments in the hospital related to availability of policy rules.

Number of policy rules	PAM		Number of policy rules	PAM		Number of policy rules	PAM		Number of policy rules	PAM	
	Success rate (%)	Average cost		Success rate (%)	Average cost		Success rate (%)	Average cost		Success rate (%)	Average cost
0	21.32	11.41	7	32.93	14.21	14	65.4	17.73	21	85.72	19.11
1	22.11	12	8	35.1	14.56	15	68.43	18.07	22	89.34	19.37
2	23.43	12.2	9	41.12	15.37	16	70.2	18.14	23	90.45	19.44
3	24.52	12.46	10	48.91	16.2	17	72.6	18.26	24	93.6	19.5
4	25.92	12.83	11	54.72	16.93	18	75.9	18.53			
5	27.64	13.21	12	59.92	17.28	19	79.6	18.69			
6	29.14	13.45	13	62.11	17.44	20	83.1	18.91			

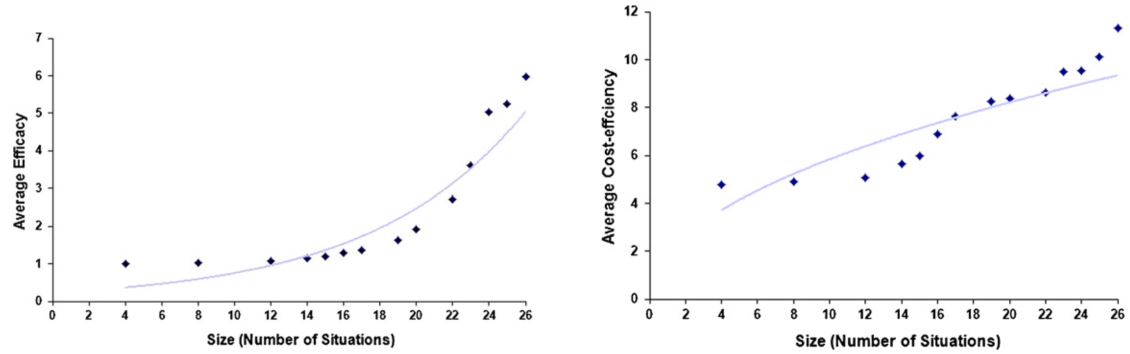

Fig. 9. Impact of size on efficacy and cost-efficiency in the hospital: Comparing PAM with ignoring awareness.

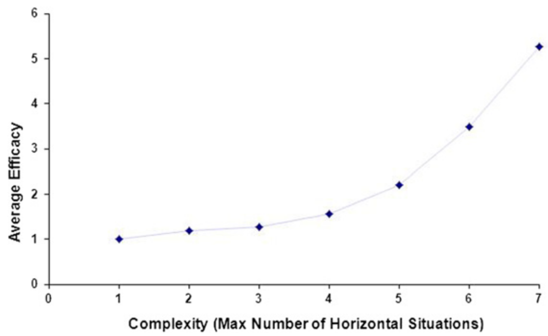

Fig. 10. Impact of complexity on efficacy and cost-efficiency in the hospital: Comparing PAM with ignoring awareness.

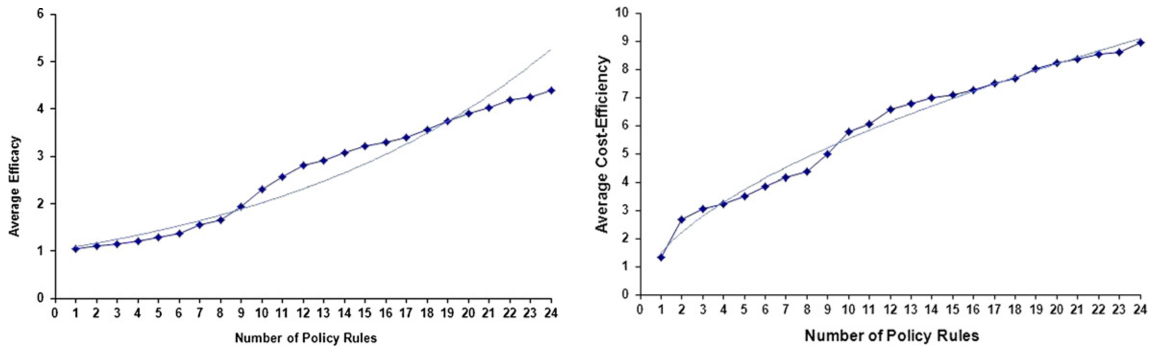

Fig. 11. Impact of number of policy rules on efficacy and cost-efficiency in the hospital: Comparing PAM with ignoring awareness.

6.3. Evaluation outcomes and summary

In the experiments on the size, we found that bigger branching-time models with more worlds have less success rate and more average cost, which are both undesirable but also expected. In fact, increasing the size brings more opportunities for the agents to select among different possible actions, which decreases the success rate, if the relevant policy rule is not provided. The bigger branching-time models clearly require more actions to be taken, which increases the average cost. The data analysis of the experiments on the size shows that updating the system to a PAM-supported system is more effective and cost-efficient in more number of words. In fact, if the cost in the system is a critical issue, then the PAM is not a proper choice. This is because by increasing the size of the branching-time model the average cost of using PAM increases more than the cost while using awareness ignorance. However, our study shows in bigger number of worlds, PAM becomes both more effective and cost-efficient. This means if in a system such as the interruptions in the wireless communication system at St. Olavs Hospital, success is the main issue, then PAM is a good choice especially in bigger size. PAM is also a good method, if in a system we are concerned about cost and success together. Since our study shows that by increasing the size, the cost-efficiency increases, PAM is also a proper choice especially in bigger branching-time model when we consider both the success rate and cost.

In the experiments on the complexity, we found that as more complicated scenarios are involved more action selections are needed, which decreases the success of the system. Comparing the success rate in PAM with the awareness ignorance, we argue that when the complexity of the scenarios increases, PAM becomes more effective.

In the experiments on number of policy rules, we examined the idea of using policy rules as a source of awareness to have a higher success rate. The results show that by increasing the policy rules, PAM becomes a more expensive but also more effective and cost-efficient method. In the other words, if the cost is an issue in a system, then using PAM with a large number of policy rules is not a wise choice. However, when the success comes to the picture and becomes an important factor for the system such as interruptions in wireless communication system at the hospital, then using PAM with a large number of policy rules is effective and cost-efficient.

6.3.1. Significance and generalizability of the experiments

The experiments presented in this paper have so far shown that PAM increases the success factor. However, we have not explained whether the finding is significant enough to call it contribution. We should also provide how limited these experiments are in terms of generalizablity. In order to do so, we use a statistical method called t-test. This method assesses whether the data collected from two groups are statistically different from each other. This analysis is appropriated when we want to compare the two groups and claim a group of data is significantly different from the other. Readers wishing more details about t-test are recommended to refer [39].

In t-test, we first set our statistical hypothesis such as a set of data μ_{PAM} is greater than the other set of data $\mu_{Awareness\,ignorance}$. We look to show that this statement is true and, if so in which degree of probability we can generalize our finding. We calculate the population means by (24) for each of these sets of data where $\bar{S}_{PAM}$ is the population means for μ_{PAM}, $S_{PAM,j}$ is the jth success rate for μ_{PAM} and n_{PAM} is the size of data set μ_{PAM}. Similar definitions are given for $\mu_{Awareness\,ignorance}$:

$$\bar{S}_{PAM} = \frac{\sum_{j=1}^{n_{PAM}} S_{PAM,j}}{n_{PAM}}. \tag{24}$$

Then, (25) estimates the variance for μ_{PAM}:

$$\theta_{PAM}^2 = \frac{\sum_{j=1}^{n_{PAM}} S_{PAM,j} - \bar{S}_{PAM}}{n_{PAM} - 1}. \tag{25}$$

The formula (26) pools the individual sample variances while the degree of freedom for the sampling is calculated by $n_{PAM} + n_{Awareness\,ignorance} - 2$:

$$\theta_p^2 = \frac{(n_{PAM} - 1)\theta_{PAM}^2 + (n_{Awareness\,ignorance} - 1)\theta_{Awareness\,ignorance}^2}{n_{PAM} + n_{Awareness\,ignorance} - 2}. \tag{26}$$

Once we calculate the t-value from (27) we look at the table of t-values. Considering the degree of freedom, if we can find the minimum probability in which the calculated t-value is still greater than provided number by the table, then the hypothesis is significant:

$$t_0 = \frac{\bar{S}_{PAM} - \bar{S}_{Awareness\,ignorance}}{\theta_p \sqrt{\frac{1}{n_{PAM}} + \frac{1}{n_{Awareness\,ignorance}}}}. \tag{27}$$

We name this probability as α and define confidence interval as $100 \times (1 - \alpha)$, which shows the generalizability of the findings.

Table 7
Significance and generalizability of the hypothetical experiments.

Experiment	Significant	Confidence interval
Size	Yes	73.8%
Complexity	Yes	82.4%
Availability of policy rules	Yes	70%

Table 7 shows the results for the hypothetical experiments with regard to their significance and generalizability. Despite the fact that the experiments are limited to confidence interval, the rates for generalizability of these experiments are acceptable.

6.3.2. Randomized inputs to avoid biasing in the experiments

It is essential to avoid biasing input data, which can harm the results. Two major reasons for biasing are (1) selection bias and (2) observation bias. In this study, we use random inputs to avoid selection biases as recommended by [40]. In order to avoid observation biases, we study PAM in wireless communication system at St. Olavs Hospital. Due to possibility of observation biasing in the case study, during the conducted discussion, the moderator double-checked what had been found in the observations and discussed them with interviewees.

7. Discussion

In this section, we explain the contributions of this research as well as the limitations. The contributions are discussed in terms of contributions for academia and contributions in practice. The limitations are given based on the few assumptions in PAM, which can open directions for future research.

7.1. Contributions

Our contributions to the academia are in the three fields; CSCW, awareness of intelligent agents and policy in agent systems:

- We borrowed the concept of awareness from CSCW. However, there is currently no definitive method for recognizing the required awareness. Given policy-aware agents, participants in cooperative endeavors can recognize the relevance of policy rules and the information that they require to enhance cooperation.
- PAM is an extension of the logic of general awareness. While the literature on this logic emphasizes the importance of a method to identify awareness of agents, PAM proposes policies as a source to identify and obtain awareness, resulting in favorable behaviors.
- Although Directory Enabled Networks-next generation (DEN-ng) are being used to implement awareness in agents, the use of policies as a guideline to find the required information for awareness has been ignored. This gap is addressed by PAM.

Our contributions to the practice can be summarized as providing an experimental study on a method with the following practical applications:

- In intelligent communication systems, PAM, as shown in the paper, can implement SLP agents to recognize the relevance of information which are required to be requested from the SLI agent.
- Practice in developing software agents typically addresses awareness in terms of programming intelligent agents. Such software agents should reason and make suggestions to assist cooperative roles. The communication system at the hospital shows that programming software agents have two weaknesses which can be solved by considering policies of agent–base systems: First, the suggestions made by software agents are dependent on their understanding of the situation and limited by their implementation. As such, agents are not able to recognize the relevance of information to a situation. Second, standard approaches to software agents are technology dependent, but involved cooperative roles often use different technologies simultaneously. Therefore, integrated cooperation can be difficult to achieve while ignoring policies as a source of awareness.

7.2. Limitations

The research presented in this paper has got some limitations that we would like to point out here.

7.2.1. Relying on the quality of policies

PAM is not a process to design policies; rather, it assumes a given set of policies. Therefore, PAM relies on the quality of the policies, which indicates the following limitations:

- Interactions and conflicts between policies: policy rules may interact with each other, and a new added policy rule may conflict with existing previous rules.
- Refining high-level policies to computational policy rules: high-level policies in ordinary English need to be translated into machine-readable form before applying PAM.

7.2.2. Primitive actions

PAM only faces with primitive actions. Primitive actions as we defined in Section 4.1 are those actions that they are directly performable by the agents, and they can be assigned to nodes, i.e. situations in the branching-time model of the world. This can be one of the future research directions to explore a method which can take non-primitive actions. Such method must map the non-primitive actions to composition of situations in branching-time model. Another approach could be decomposition of the non-primitive actions to primitive actions. The initiative for this decomposition would be the structure of the branching-time model.

7.2.3. Lack of practical production evaluation

Although PAM is untried in mass production environments for application in the different domains, an initial experimental proof of concept as an exemplar is made in the wireless communication system at St. Olavs Hospital. We also showed the applicability of PAM in space shuttle Columbia disaster in 2003. While exemplars are a common way to provide initial validation to a new method in software engineering [41], further analysis, productions, and more real-world experimentation are needed.

We admit to the following limitations of the illustrative examples, which can be methodologically handled by the triangulation:

- *Simplicity*: The objective of taking an example from the space shuttle Columbia disaster was to illustrate the motivation of the research as a proof of concept. This example also illustrates the PAM framework and process.
- *Hindsight biasing*[1]: Hindsight bias tends to occur when we believe (after an incident) that the onset of the incident was predictable. However, this is an overestimation since we benefit from the feedbacks given about the outcome of the incident [42]. Le Coze [43] acknowledges this issue in incident investigation research and advises to triangulate evaluation methods. However, we admit that the space shuttle Columbia disaster suffers from hindsight biasing; we use triangulation in our evaluation to handle this issue.

8. Conclusion

Awareness in the field of computer supported cooperative work and in the area of intelligent agents lacks a definitive method to identify relevant information. We also indicate that although currently several policy frameworks such as Directory Enabled Networks-next generation (DEN-ng) are being used to implement awareness in agent systems, the use of policies as a guideline to find the awareness has not been forthcoming. In this paper, we have shown that policies can be a source of awareness, which changes the behavior of roles without the need for direct orders. We proposed a method called Policy-based Aware Management (PAM) that refines policies to awareness. We demonstrated PAM by applying the method to the space shuttle Columbia disaster in US in 2003.

The PAM framework is based on a tree-like structure with a single past and multiple futures, called branching-time model of worlds. At any change of environment, i.e. event, PAM defines a possible world represented by a branching-time model. In each model, an arrangement of situations identifies the actions that change one situation to another. PAM framework follows the logic of general awareness, where implicit knowledge about a proposition can be inferred in a situation where the proposition is valid. Awareness is defined as a relevant proposition to a situation. PAM framework also defines explicit knowledge where both of implicit knowledge and awareness are gained.

The PAM process uses policy rules as a source to identify awareness in three steps. The first step is to identify the relevant policy rules. In this step, the agent becomes aware of a policy rule if and only if in the branching-time model of the world, there is a possibility, sometimes now or in future, that the agent violates the rule. The second step is to identify the relevance of information. When an agent becomes aware of a policy rule, it takes the condition of the rule relevant and becomes aware of that. In third step, the agent changes its behaviors to gain implicit knowledge to what it is aware of.

In this paper, we triangulated series of hypothetical simulations with our experiments on the application of PAM to the wireless communication system at St. Olavs Hospital.

Acknowledgments

A part of this research is supported by the Research Council of Norway, grant No. 176852/S10. We would also like to thank all the physicians at St. Olavs Hospital for all their helps and collaborations.

[1] In some psychology literature, it is also called knew-it-all-along effect.

References

[1] F. Daneshgar, J. Wang, Validation of the awareness net model for the Australian security investment processes, Knowl.-Based Syst. 20 (8) (2007) 736–744.
[2] P. Dourish, V. Bellotti, Awareness and coordination in shared workspaces, in: Proceedings of the 1992 ACM Conference on Computer-Supported Cooperative Work (CSCW'92), ACM, New York, NY, 1992, pp. 107–114, http://doi.acm.org/10.1145/143457.143468.
[3] C. Gutwin, S. Greenberg, Support for group awareness in real-time desktop conferences, University of Waikato, 2007.
[4] P. Ray, S.A. Shahrestani, F. Daneshgar, The role of fuzzy awareness modelling in cooperative management, Inform. Syst. Front. 7 (3) (2005) 299–316.
[5] M. Zacarias, H. Pinto, R. Magalhães, J. Tribolet, A context-aware and agent-centric perspective for the alignment between individuals and organizations, Inf. Syst. 35 (4) (2010) 441–466.
[6] R. Magalhães, Organizational Knowledge and Technology: An Action-Oriented Perspective on Organization and Information Systems, Edward Elgar Publishing, 2004.
[7] H. Dommel, J.J. Garcia-Luna-Aceves, A coordination framework and architecture for internet groupware, J. Netw. Comput. Appl. 23 (4) (2000) 401–427.
[8] Y. Li, W. Shen, H. Ghenniwa, Agent-based web services framework and development environment, Comput. Intell. 20 (2004) 678–692.
[9] R. Fagin, J. Halpern, Belief, awareness, and limited reasoning, Artificial Intelligence 4 (1988).
[10] G. Sillari, Models of awareness, in: Logic and the Foundations of Game and Decision Theory, vol. 3, 2008.
[11] G. Sillari, Quantified logic of awareness and impossible possible worlds, Rev. Symb. Log. 1 (4) (2008) 514–529.
[12] J. Strassner, et al., The design of a new policy model to support ontology-driven reasoning for autonomic networking, J. Netw. Syst. Manag. 17 (1) (2009) 5–32.
[13] R.B. Johnson, A.J. Onwuegbuzie, L.A. Turner, Toward a definition of mixed methods research, J. Mix. Methods Res. 1 (2) (2007) 112–133.
[14] S. McDanels, An overview of the space shuttle Columbia accident from recovery through reconstruction, J. Fail. Anal. Prev. 6 (1) (2006) 2–91.
[15] J. Wilson, Report of Columbia accident investigation board, vol. I, NASA, USA, 2003.
[16] NASA, Columbia crew survival investigation report, USA, 2003.
[17] M.D. Storey, D. Čubranić, D.M. German, On the use of visualization to support awareness of human activities in software development, in: Proceedings of ACM Symposium, St. Louis, MO, 2005.
[18] I. Omoronyia, J. Ferguson, M. Roper, M. Wood, A review of awareness in distributed collaborative software engineering, Softw. Pract. Exper. 40 (2010).
[19] A. Sarma, A survey of collaborative tools in software development, UCI, ISR Technical Report, UCI-ISR-05-3, 2005.
[20] N. Bricon-Souf, C.R. Newman, Context awareness in health care: A review, Int. J. Med. Inform. 76 (1) (2007) 2–12.
[21] C. Gutwin, G. Stark, S. Greenberg, in: Proceedings of the First International Conference on Computer Support for Collaborative Learning, 1995, pp. 147–156.
[22] A. Acquisti, R. Gross, Imagined communities: Awareness, information sharing, and privacy on the Facebook, in: Proceedings of Sixth Workshop on Privacy Enhancing Technologies, 2006, pp. 36–58.
[23] T. Gross, C. Stary, A. Totter, User-centered awareness in computer-supported cooperative work-systems: Structured embedding of findings from social sciences, Int. J. Hum. Comput. Interact. 18 (3) (2005).
[24] J. Hintikka, Impossible possible worlds vindicated, J. Philos. Logic 4 (3) (1975) 475–484.
[25] J. Bates, The role of emotion in believable agents, Commun. ACM 37 (7) (1994) 122–125.
[26] J.Y. Halpern, R. Pucella, Dealing with logical omniscience: Expressiveness and pragmatics, Artificial Intelligence (2010), doi:10.1145/1324249.1324273.
[27] A.S. Rao, M.P. Georgeff, Modeling rational agents within a BDI-architecture, in: Proceedings of the Second International Conference on Principles of Knowledge Representation and Reasoning, 1991.
[28] J. Allen, G. Fergusson, Actions and events in interval temporal logic, J. Logic Comput. 4 (5) (1994) 531–579.
[29] O. Kupferman, M. Vardi, Logic in computer science, in: Proceedings of 56 Twenty First Annual IEEE Symp. on Logic in Computer Science, 2006, pp. 265–274.
[30] L.M. Surhone, M.T. Timpledon, S.F. Marseken, Temporal Logic, Betascript Publishing, 2010.
[31] D.W. Marsh, R.O. Baldwin, B.E. Mullins, R.F. Mills, M.R. Grimaila, A security policy language for wireless sensor networks, J. Syst. Softw. 82 (1) (2009) 101–111.
[32] K. Twidle, N. Dulay, E. Lupu, M. Sloman, Autonomic and autonomous systems, in: Proceedings of Fifth International Conference on Autonomic and Autonomous Systems, 2009, pp. 330–335.
[33] D. Fallis, Epistemic value theory and information ethics, Minds Mach. 14 (1) (2004) 101–117.
[34] F. Kröger, S. Merz, Temporal Logic and State Systems, Springer, 2008.
[35] M. Johnson, Policy-based coordination in joint human-agent activity, in: Proceedings of the IEEE Int. Conf. on Sys., Man., and Cybr., 2004, pp. 10–13.
[36] J. Akhtman, L. Hanzo, Power versus bandwidth-efficiency in wireless communications: The economic perspective, in: Proceedings of 70th IEEE Conference of Vehicular Technology, 2009, doi:10.1109/VETECF.2009.5379027.
[37] R. Stockdale, C. Standing, An interpretive approach to evaluating information systems: A content, context, process framework, European J. Oper. Res. 173 (3) (2006) 1090–1102.
[38] D.E.R. Girden, ANOVA: Repeated Measures, first ed., Sage Publications, Inc., 1991.
[39] D.C. Montgomery, Design and Analysis of Experiments, John Wiley and Sons, 2008.
[40] R.A. Bailey, Design of Comparative Experiments, first ed., Cambridge University Press, 2008.
[41] L.M. Cysneiros, V. Werneck, E. Yu, Evaluating methodologies: A requirements engineering approach through the use of an exemplar, in: Proceedings of Seventh Workshop in Requirement Engineering, 2004, pp. 40–55.
[42] U. Hoffrage, R. Hertwig, G. Gigerenzer, Hindsight bias: A by-product of knowledge updating?, J. Exper. Psychol. Learn. Mem. Cogn. 26 (3) (2000) 566–581.
[43] J. Le Coze, Disasters and organisations: From lessons learnt to theorising, Safety Sci. 46 (1) (2008) 132–149.