2020

# G2i: A principled approach to leveraging the goal-to-information nexus in BDI agents

Kinzang Chhogyal

Angela Consoli

Aditya K. Ghose
*University of Wollongong*, aditya@uow.edu.au

Hoa K. Dam
*University of Wollongong*, hoa@uow.edu.au

# G2i: A principled approach to leveraging the goal-to-information nexus in BDI agents


## Abstract

In a number of settings, information requirements must be guided by the current goals of an agent. This paper shows how information requirements can be systematically derived from goals by examining the executability conditions of plans and actions that help achieve that goal. We extend the standard BDI framework by annotating beliefs, plans and actions with probabilities and propose the use of information-seeking actions to meet the information requirements. We show how agents can use such information-seeking actions, which can be expensive to execute (often at the cost of other mission-critical actions), to increase the expected utility of their goals, and thereby present a novel variant of the standard BDI framework.


## Disciplines

Engineering | Science and Technology Studies


## Publication Details

Chhogyal, K., Consoli, A., Ghose, A. & Dam, H. (2020). G2i: A principled approach to leveraging the goal-to-information nexus in BDI agents. Procedia Computer Science, 176 2675-2684.

Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 176 (2020) 2675–2684

Procedia Computer Science

www.elsevier.com/locate/procedia

24th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems

# G2I: A principled approach to leveraging the goal-to-information nexus in BDI agents

Kinzang Chhogyal[a], Angela Consoli[b,*], Aditya Ghose[a], Hoa Dam[a]

[a]*School of Computing and IT, University of Wollongong, NSW 2522, Australia*
[b]*Defence Science and Technology Group, PO Box 1500, Edinburgh SA 5111, Australia*

**Abstract**

In a number of settings, information requirements must be guided by the current goals of an agent. This paper shows how information requirements can be systematically derived from goals by examining the executability conditions of plans and actions that help achieve that goal. We extend the standard BDI framework by annotating beliefs, plans and actions with probabilities and propose the use of *information-seeking* actions to meet the information requirements. We show how agents can use such information-seeking actions, which can be expensive to execute (often at the cost of other mission-critical actions), to increase the expected utility of their goals, and thereby present a novel variant of the standard BDI framework.



*Keywords:* BDI agents; Goal-to-information nexus; Information-seeking actions

## 1. Introduction

In the BDI (belief-desire-intention) agent programming framework [**?** ], agents have the capability to achieve goals by selecting and executing plans from a plan library. Plans comprise of goals and subgoals which ultimately decompose into a sequence of primitive actions. There is a vast body of work in this area that addresses different issues related to BDI systems. However, the relationship between goals and the information required to achieve these goals has received limited attention. We call this the *goal-to-information* nexus, which we abbreviate as *G2I*.

Where does one begin to examine the relationship between goals and information? A starting point that holds much promise lies with the plans (and actions) used for achieving goals. Plans and actions in BDI systems consist of pre-conditions and post-conditions; the former are the executability conditions for plans and actions, whereas the latter prescribe the results of the execution. The executability conditions are verified by the agent against its set of beliefs, including the assumption made in typical BDI systems, that the verification proceeds smoothly. This begs

* Corresponding author. Tel.: +0-000-000-0000 ; fax: +0-000-000-0000.
*E-mail address:* Angela.Consoli@dst.defence.gov.au


10.1016/j.procs.2020.09.295

the question of what happens if an agent cannot verify the executability conditions because there is no belief in its belief set that is relevant for making such verifications? It is clear that this problem arises because the agent lacks the necessary information it requires, and this is where we see the first connection between goals and information. One way to handle this is by making the *closed-world assumption*, where anything that is not known is assumed to be false. The main advantage of making the closed-world assumption is that less effort is required on the agent's part but making decisions based on this assumption could be dangerous, especially in critical situations. Alternatively, if an agent is aware it lacks information, it may be endowed with the capability to launch specific actions to obtain the necessary information in order to make informed decisions. This requires the agent to do more work but it may be well worth it. We call these actions *information-seeking actions*.

Standard BDI systems, which are popular in the literature, are deterministic; an agent either believes in something or it does not; the pre-condition of a plan, or action, is either true or false. However, if we allow for non-deterministic (probabilistic) beliefs and pre-conditions, a second connection between goals and information arises. The concept of the *expected utility* follows naturally from the probabilistic nature of beliefs, plans and actions. The expected utility of executing a plan is wholly determined by the actions that are present in the body of the plan. Agents desire to get the maximum expected utility possible, but if they are considerably uncertain about the pre- and post-conditions of plans and actions, such desires are futile. Thus, agents must increase the certainty surrounding plans and actions. This too, can be achieved with the help of information-seeking actions, which highlights another connection between goals and information.

This paper is largely motivated by the second connection that we identified. We begin by formalising the notions of probabilistic belief states, actions and plans. We then proceed to show, via a motivating example, how the expected utility may be computed, and how agents may utilise information-seeking actions in order to determine the value of information that such actions offer. We also briefly discuss the role that the goal-information nexus can play in providing heightened situational awareness for operators in hostile environments with multiple agents.

## 2. Preliminaries

We use a propositional language $\mathscr{L}$ consisting of a finite set of atoms for our presentation but the ideas discussed are also applicable to a first-order language. Literals are atoms, also called positive literals, or their negation (negative literals). For instance, if $a$ is an atom, $a$ and $\neg a$ are literals. We will reserve the letter $l$ to denote literals. If $l$ is a literal, we use the upper-case letter $L$ to range over the literals $l$ and $\neg l$. The current state of the world, denoted by $S$, is a complete assignment of truth values to all the atoms, i.e. for each atom, it consists of either the positive or negative literal but not both. We use $\mathscr{A}$ to denote a BDI agent.

**Belief State:** An agent $\mathscr{A}$'s belief state, denoted as $\mathscr{B}$, is a probability distribution over the set of all literals $\mathscr{L}$.[1] $\mathscr{B}(l)$ is the probability that the agent believes literal $l$ is true. If $\mathscr{B}(l) = k$, then $\mathscr{B}(\neg l) = 1 - k$, where $0 \leq k \leq 1$. Given a set of literals, $\{l_1, l_2, \ldots, l_n\}$, we take $\mathscr{B}(\{l_1, l_2, \ldots, l_n\})$ to be the agent's belief that all the literals in the set are true, i.e. the conjunction of literals, and write it as $\mathscr{B}(l_1, l_2, \ldots, l_n)$. We will say that a belief state $B$ is consistent iff for all literals $l \in \mathscr{L}$, $\mathscr{B}(l) + \mathscr{B}(\neg l) = 1$.

**Goals, Actions, Plans and Belief Update:** In implementations of BDI systems such as in *AgentSpeak - Jason*[**? ?** ], agents have goals that are achieved by selecting applicable plans from a plan library. Various types of goals, such as *achievement* and *maintenance* goals, have been identified in the literature. In our case, we focus on *achievement* goals - a description of the desired state of the world that an agent would like the world to evolve to. Such goals can be represented *declaratively* and this facilitates the means to reason about goals such as whether a goal has been achieved, or whether two goals are in conflict [**? ? ?** ]. Formally, a goal $\mathscr{G}$ is a consistent set of literals.

The definitions for actions and plans that we use are similar to standard definitions that can be found in the literature except that we annotate the literals in the precondition of an action and the context condition of a plan with probabilities. Formally, an action $a$ is a tuple consiting of a *name*, precondition $pre(a)$ and postcondition $post(a)$ where *name*

[1] The assumption is that the agent's belief state satisfies the axioms of probability. See the Dutch Book Arguments [**?** ].

is the name of an action and, $pre(a)$ is a set of 2-tuples of the form $\langle l,\ \mathbb{P}(l)\rangle$ with $0 < \mathbb{P}(l) \leq 1$, and $post(a)$ is a consistent sets of literals. $\mathbb{P}(l)$ is the minimum probability with which $\mathscr{A}$ believes $l$ should be true for action $a$ to be considered executable. More on this will be provided in the section on executability that follows shortly. The reader may object that $\mathbb{P}$ also constitutes an agent's beliefs and should be included in $\mathscr{B}$. This is a valid point, but for ease of presentation we will represent them separately.

In BDI agent programming, an agent $\mathscr{A}$ may have several plans that can help achieve its goal $\mathscr{G}$. To choose an applicable plan, the agent uses the context condition $C$ of a plan to test its applicability. If it passes the test, then the body of the plan, consisting of a sequence of sub-goals ($\mathscr{G}_i$) and actions ($a_j$) may be executed. Note that there may be more than one applicable plan and other ways of selecting a plan for execution may be needed. More formally, a plan has the following structure: $\mathscr{G} : \mathscr{C} : \mathscr{G}_1, \mathscr{G}_2, \ldots, a_1, a_2, \ldots$, where $\mathscr{G}$ is the goal, the context condition $\mathscr{C}$ is also a set of 2-tuples of the form $\langle l,\ \mathbb{P}(l)\rangle$ with $0 < \mathbb{P}(l) \leq 1$, $\mathscr{G}_i$ are sub-goals and $a_j$ are actions. The sequence of sub-goals and actions is referred to as the *body* of the plan.[2]

**Executability of actions and plans:** Typically, in AI planning, the agent has to be certain that each literal in the goal's context condition or action precondition is true for a plan or action to be deemed executable. In our case, we must take into account the probabilistic nature of action preconditions and goal context conditions to determine when an action or plan is executable. Given an agent $\mathscr{A}$ with current belief state $\mathscr{B}$, $\mathscr{A}$ deems

- an action $a$ executable iff $\forall \langle l_i, \mathbb{P}(l_i)\rangle \in \text{pre}(a),\ \mathscr{B}(l_i) \geq \mathbb{P}(l_i)$, and
- a plan $\mathscr{P}$ executable iff $\forall \langle l_i, \mathbb{P}(l_i)\rangle \in \mathscr{C},\ \mathscr{B}(l_i) \geq \mathbb{P}(l_i)$, where $\mathscr{C}$ is the context condition of $\mathscr{P}$.

One may ask how the agent obtains the probabilities $\mathbb{P}(l_i)$. Such probabilities could be derived either from the agent's causal knowledge about actions or from the agent's prior experience in executing these plans and actions.

**Belief Update:** Agents may update their beliefs either after receiving/sensing new information or after they perform an action. Thus, they must be equipped with a mechanism that enables them to do so. In this paper, we assume that an agent's knowledge is encoded in a Bayesian Network [**?** ] and, depending on the level of certainty of the evidence, we use two belief update mechanisms. Let $\mathscr{B}$ be $\mathscr{A}$'s prior belief state and $l_1$ and $l_2$ be two literals. $\mathscr{B}'$ represents the updated or new belief state.

> *Certain Evidence*: If the agent is certain about the evidence (probability 1) that $l_2$ is true, then $\mathscr{B}'(l_2) = 1$. The updated belief in $l_1$ is given by Simple Conditionalization [**?** ] as $\mathscr{B}'(l_1) = \mathscr{B}(l_1 \mid l_2)$.
>
> *Uncertain Evidence*: If the evidence is uncertain, i.e. $0 < \mathscr{B}'(l_2) < 1$, Simple Conditionalization can no longer be used. Instead, we use Jeffrey Conditionalization [**?** ], and the updated belief in $l_1$ is computed as $\mathscr{B}'(l_1) = \mathscr{B}(l_1 \mid l_2)\mathscr{B}'(l_2) + \mathscr{B}(l_1 \mid \neg l_2)\mathscr{B}'(\neg l_2)$.

Note that in both cases the assumption is that the probabilities associated with the agent's conditional beliefs remain unchanged in updated belief states, i.e. $\mathscr{B}(l_1 \mid l_2) = \mathscr{B}'(l_1 \mid l_2)$.

## 3. Expected Utility of Plans

**Single Action Plans:** We assume that an agent assigns utility values to goals. Intuitively, the utility value can be taken to be the reward that the agent receives for achieving a goal - the greater the utility value, the greater the reward. The utility of a goal $\mathscr{G}$ is denoted as $\mathscr{U}(\mathscr{G})$. We will first focus on the case where a goal can be achieved by a plan $P$ with a single action $a$ in the body of the plan. This will suffice to illustrate the core ideas in this paper and we will see later that it easily extends to the multiple actions case.

[2] Note that sub-goals and actions may appear in any order although what we have shown suggests there is a sequence of sub-goals followed by a sequence of actions.

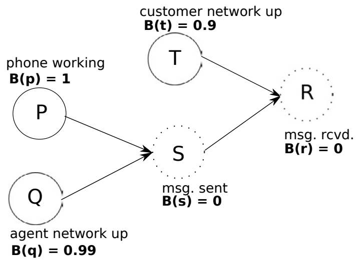


Figure 1. The Bayesian Network for Ex. 1. The prior beliefs in $\mathscr{B}(s)$ and $\mathscr{B}(r)$ are 0 as the action to send the message has not been executed yet. They are only updated after the action *send_sms* is executed.

*E*xample 1: Consider that a bank employs a BDI agent that can send SMS security codes to customers for verifying transactions. Also, assume that the agent wants to message a particular customer in a rural area. The mobile network is very reliable in the agent's area and very rarely do its messages not get sent for delivery. On the other hand, the mobile network in the customer's area is slightly less reliable, and sometimes the customer does not receive messages sent to her. The Bayesian network for this example is shown in Fig. 1. The conditional probability tables (CPTs) relating parent and child nodes is simple for this example and shown in Table 1. For the agent, knowing that either the phone is not working or that its network is down means also knowing that its message will not get sent. Also, knowing either that its message has not been sent or the rural network is down means it knows the customer will not receive the message.

Table 1. Conditional probability table (CPT) for Ex. 1

| $P$ | $Q$ | $\mathscr{B}(s)$ |
|---|---|---|
| $p$ | $q$ | 1 |
| $p$ | $\neg q$ | 0 |
| $\neg p$ | $q$ | 0 |
| $\neg p$ | $\neg q$ | 0 |

| $S$ | $T$ | $\mathscr{B}(r)$ |
|---|---|---|
| $s$ | $t$ | 1 |
| $s$ | $\neg t$ | 0 |
| $\neg s$ | $t$ | 0 |
| $\neg s$ | $\neg t$ | 0 |

Let us formalise the example above. Let $\mathscr{L}$ consist of the literals: $p$: the BDI agent's phone is working,[3] $q$: the mobile network is up in the agent's area, $r$: the customer received the message, $s$: the message was sent to the customer, and $t$: the mobile network is up in the customer's area

Let the agent's current belief state be $\mathscr{B}$ with: $\mathscr{B}(p) = 1$, $\mathscr{B}(q) = 0.99$, $\mathscr{B}(s) = 0$ $\mathscr{B}(r) = 0$ and $\mathscr{B}(t) = 0.9$. $\mathscr{B}(s)$ and $\mathscr{B}(r)$ are zero because the agent has not sent the message yet. They are in this sense special nodes and that is why they are shown with dotted lines in Fig.1. We assume that the probabilities for $p$, $q$ and $t$ are based on the agent's knowledge and/or its prior experience. Let goal $\mathscr{G}$ be: $\mathscr{G} = \{r, s\}$. Let the plan that has been selected to achieve $\mathscr{G}$ be: $\{r, s\} : \{\langle q, 0.90\rangle\} : \mathit{send\_sms}$ where *send_sms* is an action. The context condition for the plan, $\{\langle q, 0.90\rangle\}$, states that the agent must believe with at least probability 0.9 that its network is up to be able to execute this plan.[4] If this is not the case, alternative plans may be pursued. The action *send_sms* has the following properties: $\mathit{pre}(\mathit{send\_sms}) = \{\langle p, 1\rangle\}$ and $\mathit{post}(\mathit{send\_sms}) = \{r, s\}$. The precondition of *send_sms* states that the agent must be certain that its phone is working before sending a SMS message. The postcondition states that the result of executing the action is that the SMS is sent and the customer receives the SMS. However, life is not ideal - the message may not get sent and/or the customer may not receive it.

[3] We say phone here but in reality it may be just be a software that communicates with a mobile network.

[4] The reader may question why $p$ is not included in the context condition. It could be but we decided to include $p$ in the pre-condition of the action *send_sms*. In either case, the analysis would change slightly but the principle remains the same.

**Belief Update Single Action:** Given an agent's current set of beliefs, we are interested in computing the expected utility that an agent may derive from executing a plan. However, we first look at how beliefs may be updated after executing an action.

Example 1(cont.): In our example, as $\mathscr{C} = \{\langle q, 0.90\rangle\}$, we have $\mathbb{P}(q) = 0.90$. Since $\mathscr{B}(q) = 0.99$ and $\mathscr{B}(q) > \mathbb{P}(q)$, the plan is applicable. Likewise, for $pre(send\_sms) = \{\langle p, 1\rangle\}$, $\mathscr{B}(p) \geq \mathbb{P}(p)$, therefore *send_sms* is executable. Assume the agent presses the *send* button and receives no error message. Note that pressing the *send* button doesn't necessarily mean the message has been sent. The agent now has to update its beliefs. Technically, there is no new evidence about $p$, $q$ and $t$, i.e. their probabilities remain unchanged, but $s$ and $r$ must be conditioned by them, so we take it there is newly received evidence about them. The probabilities associated with $p$, $q$ and $t$ in the updated belief state $B'$ are:

$\mathscr{B}'(p) = 1$, $\mathscr{B}'(q) = 0.99$ and $\mathscr{B}'(t) = 0.9$.

The probabilities for $s$ and $r$ must be updated given new evidence about $p$, $q$ and $t$. We must deal with $s$ first. It will become clear why so. Due to conditional independence assumptions in Bayesian Networks, we know that $s$ is independent of any other node given its parents $p$ and $q$. However, since $\mathscr{B}'(q) < 1$, we use Jeffrey Conditionalization to get the agent's updated belief in $s$ as follows:

$\mathscr{B}'(s) = \sum_{P,Q} \mathscr{B}(s \mid P, Q)\ \mathscr{B}'(P, Q) = \mathscr{B}(s \mid pq)\ \mathscr{B}'(p, q) = 0.99$ [5]

Similarly, we get $\mathscr{B}'(r)$ :

$\mathscr{B}'(r) = \sum_{S,T} \mathscr{B}(r \mid S, T)\ \mathscr{B}'(S, T) = \mathscr{B}(r|st) \times \mathscr{B}'(st) \approx 0.89.$ [6]

To see how we obtained $\mathscr{B}'(st)$, see the footnote.[7] As seen in above, the computation of $\mathscr{B}'(r)$ relies on $\mathscr{B}'(s)$ and this is the reason why the value of $\mathscr{B}'(s)$ had to be established first. It shows that: a) probabilities associated with the nodes are updated in a specific order, b) belief update is not always be a one-shot process but rather a sequence of updates, and c) some nodes only have their probabilities updated after the execution of a particular action.

**Expected Utility:** We next calculate the *expected utility* (EU) obtained by executing the plan in the single action case.

**Definition 3.1** (Expected Utility of executing a plan)**.** *Given a belief state B, the expected utility of executing plan $\mathscr{P}$ w.r.t goal a $\mathscr{G} = \{l_1, l_2, \ldots l_n\}$ is denoted as $EU_{\mathscr{P}}(\mathscr{G})$ and is defined as: $EU_{\mathscr{P}}(\mathscr{G}) = \mathscr{U}(\mathscr{G}) \times \mathscr{B}'(\mathscr{G})$, where $\mathscr{B}'$ represents the updated belief state after executing $\mathscr{P}$, i.e. the single action in the body. Note that $EU_{\mathscr{P}}(\mathscr{G})$ cannot be greater than $\mathscr{U}(\mathscr{G})$ as $\mathscr{B}'(\mathscr{G})$ is at most 1.*

Example 1(cont.): Let $\mathscr{U}(\mathscr{G}) = \mathscr{U}(\{r, s\}) = 5$. Then, $EU_{\mathscr{P}}(\{s, r\}) = \mathscr{U}(\{s, r\}) \times \mathscr{B}'(s \wedge r) = \mathscr{U}(\{s, r\}) \sum_{P,Q,T} \mathscr{B}'(P, Q, r, s, T) = \mathscr{U}(\{s, r\})\ \mathscr{B}'(s \mid pq)\ \mathscr{B}'(r|st)\ \mathscr{B}'(t)\ \mathscr{B}'(p)\ \mathscr{B}'(q) = 4.45.$ [8]

The expected utility is just under 90% of the maximum of 5 possible. Depending on the situation, this may or may not be acceptable. In our example, making sure messages are delivered may be critical and 90% may not be good enough. If this is so, the question then becomes whether the agent can do anything to increase the expected utility of executing the plan. We address this next.

---

[5] Observe from the CPT that $\mathscr{B}(s \mid P, Q) \neq 0$ iff $P = p$ and $Q = q$. All other terms in the summation can be eliminated. So $\mathscr{B}'(s) = \mathscr{B}(s \mid p, q)\ \mathscr{B}'(pq) = 1 \times \mathscr{B}'(p) \times \mathscr{B}'(q)$ [Independence Assumption] $= 1 \times 1 \times 0.99 = 0.99$

[6] $\sum_{S,T} \mathscr{B}(r \mid S, T)\ \mathscr{B}'(S, T) = \mathscr{B}(r|st) \times \mathscr{B}'(st) = \mathscr{B}(r|st) \times \mathscr{B}'(st) = 1 \times 0.89 \approx 0.89$

[7] We know $\mathscr{B}'(st) = \mathscr{B}'(PQRst)$. By conditional independence assumption, $\mathscr{B}'(PQRst) = \mathscr{B}'(s \mid PQ)\mathscr{B}'(R \mid st)\mathscr{B}'(t)\mathscr{B}'(P)\mathscr{B}'(Q) = \mathscr{B}'(s \mid pq)\mathscr{B}'(r \mid st)\mathscr{B}'(t)\mathscr{B}'(p)\mathscr{B}'(q) = 1 \times 1 \times 0.9 \times 1 \times 0.99 \approx 0.89$.

[8] From conditional independence assumptions, we get: $EU_{\mathscr{P}}(\{s, r\}) = \mathscr{U}(\{s, r\}) \sum_{P,Q,T} \mathscr{B}'(s|PQ)\ \mathscr{B}'(r|sT)\ \mathscr{B}'(T)\ \mathscr{B}'(P)\ \mathscr{B}'(Q)$. As done in the previous example, the only conditional probabilities that are non-zero due to the CPTs are when $P = p$, $Q = q$ and $T = t$ $EU_{\mathscr{P}}(\{s, r\}) = \mathscr{U}(\{s, r\})\ \mathscr{B}'(s \mid pq)\ \mathscr{B}'(r|st)\ \mathscr{B}'(t)\ \mathscr{B}'(p)\ \mathscr{B}'(q) = 5 \times 1 \times 1 \times 0.9 \times 1 \times 0.99 = 5 \times 0.89 = 4.45$

**Expected Utility with Perfect Information:** Prior to sending the message to the client and assuming that the status of mobile networks can be checked, the agent could: i) check that its mobile network is up and/or ii) check that the customer's network is up. We will call such actions *information-seeking actions* (ISA). These actions provide the agent with perfect information so that after launching them, it is certain about the truth of the literals of interest.[9] ISAs could be used for two primary purposes. First, confirming the context conditions and action preconditions where absolutely certainty is required. Second, for increasing the expected utility of executing a plan. More formally,

**Definition 3.2** (ISA)**.** *Given a belief state $\mathscr{B}$ and a set of literals $\mathscr{V} = \{v_1, \ldots, v_m\}$, an information-seeking action $isa_{\mathscr{V}}$ is an action such that if $\mathscr{B}'$ represent the updated belief state after launching $isa_{\mathscr{V}}$, then for all $v_i \in \mathscr{V}$ either $\mathscr{B}'(v_i) = 0$ or $\mathscr{B}'(v_i) = 1$.*

Before launching $isa_{\mathscr{V}}$, the agent is uncertain about truth of these literals in $\mathscr{V}$, i.e. $0 < \mathscr{B}(v_i) < 1$.[10] The assumption is that post-launch, the agent receives information that makes it certain about the truth of the literals in $\mathscr{V}$, i.e. either $\mathscr{B}'(v_i) = 0$ or $\mathscr{B}'(v_i) = 1$. ISAs do not come for free. Additional energy must be expended in order to use them which can often be costly. We use $k_{\mathscr{V}}$ to denote the cost of launching the action $isa_{\mathscr{V}}$. Are ISAs are worth being launched? One way to answer this question is by computing the *value of information* gained by launching ISAs.

**Definition 3.3** (EU with Perfect Information)**.** *Let an agent's current belief state be $\mathscr{B}$, let information-seeking action $isa_{\mathscr{V}}$, where $\mathscr{V} = \{v_1, \ldots, v_m\}$, be launched and let $B'$ be the updated belief state after launching $isa_{\mathscr{V}}$. The expected utility of executing plan $\mathscr{P}$ w.r.t goal a $\mathscr{G}$ with perfection information about the literals in $\mathscr{V}$ is denoted as $EU_{\mathscr{P}}(\mathscr{G} \mid \mathscr{V})$ and is defined as $EU_{\mathscr{P}}(\mathscr{G} \mid \mathscr{V}) = \mathscr{U}(\mathscr{G}) \times \mathscr{B}'(\mathscr{G})$, where for each literal l: if $l \in \mathscr{V}$ or $\neg l \in \mathscr{V}$, then $\mathscr{B}'(l) = 0$ or $1$, else $\mathscr{B}'(l)$ obtained using either Simple or Jeffrey condtionalization.*

Note that there are $m$ literals in $\mathscr{V}$ and for each one of them, the agent could either observe $v_i$ as being true or false, i.e. $\mathscr{B}'(v_i) = 0$ or $\mathscr{B}'(v_i) = 1$. Thus, there are $2^m$ *potential* expected utility functions that result from Def. 3.3 above. We will denote them as $EU_1(\cdot), \ldots, EU_{2^m}(\cdot)$. They will be required for computing the *value of information* that we take up shortly, but first let us look at an example of the expected utility of executing a plan after obtaining perfect information about a set of literals. If $\mathscr{V} = \{v_1, v_2\}$, when we write $EU_{\mathscr{P}}(\mathscr{G} \mid \{v_1, v_2\})$, we mean the expected utility after observing $v_1$ and $v_2$ as being true, i.e. probability of 1. Likewise, by $EU_{\mathscr{P}}(\mathscr{G} \mid \{v_1, \neg v_2\})$, we mean the expected utility when observing $v_1$ as being true and $v_2$ as being false, i.e. probability of 1 and 0 respectively, and so on.

*E*xample 1(cont.) Say prior to messaging the customer, the agent decides to launch an ISA, $isa_{\{t\}}$, to verify the status of the customer's network. Here $\mathscr{V} = \{t\}$, therefore we can have two functions, $EU_1 = EU_{\mathscr{P}}(\mathscr{G} \mid \{t\})$ and $EU_2 = EU_{\mathscr{P}}(\mathscr{G} \mid \{\neg t\})$, one for possibly observing the network is up and one for possibly observing it is down. Let us compute $EU_1(\cdot)$ first. Observing $t$, the agent updates its belief state so that $\mathscr{B}'(t) = 1$. Belief updates of $s$ and $r$ are done as before, and we get $B'(s) = B'(r) = 0.99$. Again, the only conditional probabilities that are non-zero due to the CPTs are when $P = p$, $Q = q$ and $T = t$. Therefore:

$EU_{\mathscr{P}}(\{s, r\} \mid \{t\}) = 4.95$ [11] and $EU_{\mathscr{P}}(\{s, r\} \mid \{\neg t\}) = 0.$ [12]

As we can see, for the agent to know with certainty that the rural network is up increases the expected utility of executing the plan. If it is down, there is no utility obtained by executing the plan.

**Value of Information:** The expected utility values we just calculated are premised on the condition that an ISA to confirm $t$'s value is actually launched. Since launching an ISA may be costly, we need to determine if it is worth our while. We cannot be certain about the truth conditions of the literals we want to verify, thus in the defining the value of information for an ISA every alternative must be considered.

[9] While we do not consider it here, there may be ISAs that do not resolve the truth of a literal but merely increases/decreases an agent's degree of belief in a literal. See the discussion section.

[10] An agent could be certain about the truth of the literals and still launch an $isa_{\mathscr{V}}$ but it wouldn't change its degree of belief in the literal.

[11] $EU_{\mathscr{P}}(\{s, r\} \mid \{t\}) = \mathscr{U}(\{s, r\})\ \mathscr{B}'(s \mid pq)\ \mathscr{B}'(r|st)\ \mathscr{B}'(t)\ \mathscr{B}'(p)\ \mathscr{B}'(q) = 5 \times 1 \times 1 \times 1 \times 1 \times 0.99 = 4.95$

[12] $EU_{\mathscr{P}}(\{s, r\} \mid \{\neg t\}) = \mathscr{U}(\{s, r\})\mathscr{B}'(s \mid pq)\mathscr{B}'(r|s\neg t)\mathscr{B}'(\neg t)\mathscr{B}'(p)\mathscr{B}'(q) = 5 \times 1 \times 0 \times 1 \times 1 \times 0.99 = 0$

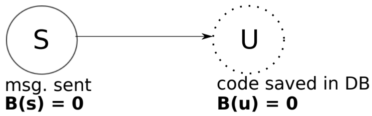


Figure 2. The Extended Bayesian Network for Ex. 2 showing only the new node $U$ and it's parent $S$ with the beliefs prior to any actions being executed.

**Definition 3.4** (Value of Information VOI)**.** *Given a belief state $\mathscr{B}$, goal $\mathscr{G}$, plan $\mathscr{P}$, isa$_{\mathscr{V}}$ where $\mathscr{V} = \{V_1, \ldots, V_m\}$, the value of information w.r.t $\mathscr{G}$, $\mathscr{P}$ and isa$_{\mathscr{V}}$ is denoted as $VOI_{\mathscr{G},\mathscr{P},isa_{\mathscr{V}}}$ or simply VOI when the context is clear, and is defined as $VOI = (\sum \mathscr{B}(\mathscr{V}) \times EU_{\mathscr{P}}(G \mid \mathscr{V})) - EU_{\mathscr{P}}(\mathscr{G}) - k_{\mathscr{V}}$*

The intuition behind this definition is that each possible outcome of the literals in $\mathscr{V}$ are considered as the agent cannot know what it will observe and the utility obtained by using that information is weighted by the prior belief in the conjunction of the literals in $\mathscr{V}$.

*E*xample 1 (cont.): As in the previous example, let $\mathscr{V} = \{t\}$. Also, let the cost $k_{\mathscr{V}} = 0.1$. Then $VOI = (\sum \mathscr{B}(\{t\}) \times EU_{\mathscr{P}}(\{r, s\} \mid \{t\})) - EU_{\mathscr{P}}(\{r, s\}) - 0.1$. We know from the previous examples that $EU_{\mathscr{P}}(\{r, s\}) = 4.45$, $EU_{\mathscr{P}}(\{s, r\} \mid \{t\}) = 4.95$ and $EU_{\mathscr{P}}(\{s, r\} \mid \{\neg t\}) = 0$. We also know $\mathscr{B}(t) = 0.9$ and $\mathscr{B}(\neg t) = 0.1$. We get, $VOI = \mathscr{B}(t)\, EU_{\mathscr{P}}(\{r, s\} \mid \{t\}) + \mathscr{B}(\neg t)\, EU_{\mathscr{P}}(\{r, s\} \mid \{\neg t\}) - 4.45 - 0.1 = -0.095$. [13]

So if the agent were to launch an ISA to detect the status of the rural network, there is negative information gain or loss of information. This can be attributed to the cost of launching the ISA being too high. Observe that for the information gain to be positive the cost must be less than 0.005.

**Multiple Action Plans:** We now consider plans with multiple actions in the body as shown below. Since sub-goals can be decomposed into a sequence of actions, this case will also cover plans that have a mix of sub-goals and actions in their body. $\mathscr{G} : \mathscr{C} : a_1, a_2, \ldots, a_n$. Compared to the single action case, since there are multiple actions here, belief updates must also occur after the execution of each action. Although the remaining analysis is the same as in the single action case, for the sake of completeness, we extend the example we have been working with by including an additional node in the Bayesian network to illustrate belief update and expected utility calculations in the multiple actions case.

Table 2. Conditional probability table for Ex. 2

| $S$ | $\mathscr{B}(u)$ |
|---|---|
| $s$ | 0.99 |
| $\neg s$ | 0 |

*E*xample 2: Consider that after sending a SMS code to the customer, the agent must subsequently execute an action *save_code* that saves the code in a database so that it can be used to verify the code the customer is expected to send. Let the variable $U$ ranging over the literals $u$ and $\neg u$ stand for the belief that *code is saved* and *unsaved* respectively. Let $U$'s parent node be $S$ as shown in Fig. 2. The conditional probability $\mathscr{B}(U \mid S)$ in shown Table 2 which says even if the agent is certain the message has been sent, it believes the code will only be saved in the database with a probability of 0.99.[14] If the agent is certain that the message has not been sent, then it will also know the code has not been saved.

The goal of the agent is as before to send a message and make sure the customer receives the message. Additionally, the agent must now save the code in the database. For this example, the goal is $\mathscr{G} = \{r, s, u\}$ and

[13] $\mathscr{B}(t)\, EU_{\mathscr{P}}(\{r, s\} \mid \{t\}) + \mathscr{B}(\neg t)\, EU_{\mathscr{P}}(\{r, s\} \mid \{\neg t\}) - 4.45 - 0.1 = 0.9 \times 4.95 + 0.1 \times 0 - 4.55 = 4.455 - 4.55 = -0.095$

[14] This could be due to transient faults can be cause by electro-magnetic noise, power supply fluctuations, etc. that may affect the database.

the plan is $\{r, s, u\}$ : $\{\langle q, 0.90\rangle\}$ : *send_sms*, *save_code*. The new action *save_code* has the following properties: $pre(save\_code) = \{\langle s, 1\rangle\}$ and $post(save\_code) = \{u\}$.

**Belief Update Multiple Actions:** Recall that in the single acton case, belief update happened in stages. After executing the *send_sms* action, the belief in $s$ was updated first, and then followed by $r$. In the multiple actions case, we also have to contend with additional belief update stages that are introduced by other actions in the plan body. As we show below, in our example the agent updates its beliefs first after executing *send_sms* and then again after executing *save_code*. We will use $\mathscr{B}', \mathscr{B}'', \mathscr{B}''', \ldots$ to denote the updated belief state due to a sequence of actions. We often use $\mathscr{B}^*$ to indicate the final belief state after executing all actions and completing all belief updates.

*E*xample 2 (cont.): Let $\mathscr{B}$' be the agent's updated belief state after executing *send_sms*. The literals $p, q, r$, $s$ and $t$ have the same probabilities in $\mathscr{B}$' as in Ex. 1. Recall that $\mathscr{B}'(s) = 0.99$. Note that belief update of $u$ only happens if *save_code* is executed, so $\mathscr{B}(u) = \mathscr{B}'(u) = 0$. Let $\mathscr{B}'' = \mathscr{B}^*$ be the agent's updated belief state after executing *save_code*. The probabilities of $p$, $q$, $r$, $s$ and $t$ remain unchanged in $\mathscr{B}''$. The only thing left to calculate is $\mathscr{B}''(u)$. This again is given by Jeffrey conditionalization as $\mathscr{B}''(s) = 0.99$. We get:

$\mathscr{B}''(u) = \sum_S \mathscr{B}(u \mid S)\, \mathscr{B}''(S) = 0.98$ [15]

Let us compute the expected utility of the goal $\mathscr{G} = \{r, s, u\}$. Let $\mathscr{U}(G) = 5$. We get:

$EU_{\mathscr{P}}(\{r, s, u\}) = \mathscr{U}(\{r, s, u\}) \times \mathscr{B}^*(r, s, u) \approx 4.41.$ [16]

**EU With Perfect Information:** Let us calculate the expected utility with perfect information for this case.

*E*xample 2 (cont): As done earlier, let $isa_{\{t\}}$ that verifies the status of the customer's network be launched. Again, there are two potential functions, $EU_{\mathscr{P}}(\{r, s, u\} \mid \{t\})$ and $EU_{\mathscr{P}}(\{r, s, u\} \mid \{\neg t\})$. We get:

$EU_{\mathscr{P}}(\{r, s, u\} \mid \{t\}) = \mathscr{U}(\{r, s, u\}) \sum_{P,Q} \mathscr{B}^*(P, Q, r, s, t, u) = 4.90$ [17], and

$EU_{\mathscr{P}}(\{r, s, u\} \mid \{\neg t\}) = \mathscr{U}(\{r, s, u\}) \sum_{P,Q} \mathscr{B}'(P, Q, r, s, \neg t, u) = = 0$ [18]

We can see from the examples above that much of the analysis in the multiple actions case is very similar to the single action case. The only difference is that belief updates can be more complex because of multiple actions.

**VOI - Multiple Actions:** We now look at the VOI calculations in the multiple actions case.

*E*xample 2 (cont.) : VOI calculations are also like in the single actions case. Let $\mathscr{V} = \{t\}$ and cost $k_{\mathscr{V}} = 0$. We get:

$VOI = (\sum \mathscr{B}(\{t\}) \times EU_{\mathscr{P}}(\{r, s, u\} \mid \{t\})) - EU_{\mathscr{P}}(\{r, s, u\}) - 0$

We know from the previous examples that $EU_{\mathscr{P}}(\{r, s, u\}) = 4.41$, $EU_{\mathscr{P}}(\{r, s, u\} \mid \{t\}) = 4.90$ and $EU_{\mathscr{P}}(\{r, s, u\} \mid \{\neg t\}) = 0$. We also know $\mathscr{B}(t) = 0.9$ and $\mathscr{B}(\neg t) = 0.1$. Thus:

$VOI = \mathscr{B}(t)\, EU_{\mathscr{P}}(\{r, s, u\} \mid \{t\}) + \mathscr{B}(\neg t)\, EU_{\mathscr{P}}(\{r, s, u\} \mid \{\neg t\}) - 4.41 - 0 = 0.$ [19]

---

[15] $\mathscr{B}''(u) = \sum_S \mathscr{B}(u \mid S)\, \mathscr{B}''(S) = \mathscr{B}(u \mid s)\mathscr{B}''(s) + \mathscr{B}(u \mid \neg s)\mathscr{B}''(\neg s) = 0.99 \times 0.99 + 0 \times 0.01 = 0.98$

[16] $EU_{\mathscr{P}}(\{r, s, u\}) = \mathscr{U}(\{r, s, u\}) \times \mathscr{B}^*(r, s, u) = 5 \times \sum_{P,Q,T} \mathscr{B}^*(P, Q, r, s, T, u) = 5 \times \sum_{P,Q,T} \mathscr{B}^*(u \mid s)\, \mathscr{B}^*(s \mid PQ)\, \mathscr{B}^*(r \mid sT)\, \mathscr{B}^*(P)\, \mathscr{B}^*(Q)\, \mathscr{B}^*(T) = 5 \times \mathscr{B}^*(u \mid s)\, \mathscr{B}^*(s \mid pq)\, \mathscr{B}^*(r \mid st)\, \mathscr{B}^*(p)\, \mathscr{B}^*(q)\, \mathscr{B}^*(t) = 5 \times 0.99 \times 1 \times 1 \times 1 \times 0.99 \times 0.9 \approx 4.41$

[17] $EU_{\mathscr{P}}(\{r, s, u\} \mid \{t\}) = \mathscr{U}(\{r, s, u\}) \sum_{P,Q} \mathscr{B}^*(P, Q, r, s, t, u) = 5 \times \mathscr{B}^*(u \mid s)\, \mathscr{B}^*(s \mid PQ)\, \mathscr{B}^*(r \mid st)\, \mathscr{B}^*(P)\, \mathscr{B}^*(Q)\, \mathscr{B}^*(t) = 5 \times \mathscr{B}^*(u \mid s)\, \mathscr{B}^*(s \mid pq)\, \mathscr{B}^*(r \mid st)\, \mathscr{B}^*(p)\, \mathscr{B}^*(q)\, \mathscr{B}^*(t) = 5 \times 0.99 \times 1 \times 1 \times 1 \times 0.99 \times 1 \approx 4.90$

[18] $EU_{\mathscr{P}}(\{r, s, u\} \mid \{\neg t\}) = \mathscr{U}(\{r, s, u\}) \sum_{P,Q} \mathscr{B}'(P, Q, r, s, \neg t, u) = = 5 \times \mathscr{B}'(u \mid s)\, \mathscr{B}'(s \mid PQ)\, \mathscr{B}'(r \mid s\neg t)\, \mathscr{B}'(P)\, \mathscr{B}'(Q)\, \mathscr{B}'(\neg t) = 0.$

[19] $VOI = \mathscr{B}(t)\, EU_{\mathscr{P}}(\{r, s, u\} \mid \{t\}) + \mathscr{B}(\neg t)\, EU_{\mathscr{P}}(\{r, s, u\} \mid \{\neg t\}) - 4.41 - 0 = 0.9 \times 4.90 + 0.1 \times 0 - 4.41 = 4.41 - 4.41 = 0$

In this case, there is no information gained by launching this ISA. Perhaps, the conditional probability ($\mathscr{B}(u \mid s)$) is already is quite high so that not much gain is seen even by launching an ISA.

## 4. Discussion

**Information-Seeking Actions with Imperfect Information:** The information-seeking actions that we considered returned perfect information, i.e. the agent could know with certainty whether a literal is true or false. In reality, this may not be possible with respect to all literals and information-seeking actions may instead increase, decrease or keep the degree of belief in a literal unchanged. For instance, consider that an agent believes with 90% probability that it is raining in a particular location; this belief could be premised on the weather forecast that it saw this morning which said that there was a 90% chance of a shower. Now, later in the afternoon if it has a goal to achieve but one that requires knowing the weather at the current moment at that location, the best it can do (assuming it has no one to ask who lives there) is to look up the most recent forecast. The forecast could now say 95% chance of a shower which means the agent doesn't have perfect information but still increases its belief about rain. The expected utility of its goals will however increase. Belief updates in such cases will have to rely on Jeffrey conditionalization which we have already used in our example so this framework can handle such cases without major modifications.

**Which Information-Seeking Action?** As we have seen, there can be multiple ISAs that can be launched. Should all possible ISAs be launched? This would certainly increase the expected utility of executing a plan but this may not always be worthwhile. For instance, consider that there are two ISAs: $isa_1$ and $isa_2$ with costs $k_1$ and $k_2$ and with value of information $VOI_1$ and $VOI_2$ respectively. If $\mid k_1 - k_2 \mid$ is large but $\mid VOI_1 - VOI_2 \mid$ is negligible, it will be better to launch the ISA that incurs the lower cost. Of course, there may be other factors to consider too. If it is critical for the expected utility of the plan to be as close as possible to the utility of the goal, then all ISAs should be launched.

**BDI Plan Selection:** In agent-programming languages like Jason [**?** ], if there are multiple applicable plans, a *applicable plan selection function* can be used to select one plan. The default selection mechanism in Jason is to choose the first plan that appears in the library. The framework that we proposed can be used to provide another plans selection function for the applicable plans based on the value of information. If there are two plans $\mathscr{P}$ and $\mathscr{P}$ for achieving goal $\mathscr{G}$, a simple selection function could be one that chooses the plan that returns a higher VOI.

**Belief Update:** From the analysis of the motivating example, it can be seen that updating beliefs form the most complex piece in our framework. This is not surprising as exact inference in Bayesian networks is NP-hard and techniques for approximate inference such as Monte Carlo sampling might be required [**?** ] . In our example, additionally, we had to be careful about the order in which the nodes were updated after executing an action. Also, there were nodes that are only affected when certain actions takes place. The Bayesian network in our example is simple enough that we knew: i) the order in which the updates should happen and ii) which actions affect which nodes. One can easily imagine a more complicated Bayesian network where knowing the two might not be as easy. Perhaps, these things can be established in a systematic way and we hope to take this up as future work. Another interesting line of inquiry is the connection between ISAs and the *do calculus* [**?** ]. The $do(\cdot)$ operator is an interventional action that sets the value of a variable. One can see that it is analogous to ISAs which also sets the value of a variable although the means by which it sets the value is not by direct intervention. This connection would be interesting to explore.

**G2I and Enhancing Situational Understanding in the Tactical Environment:** The high level objective for any mission within a environment with competing agents is to preserve or enhance situation understanding and deliver decision superiority to the agent of interest. SU is achieved by defining a set of mission goals, where information requirements, as well as information sources, are derived to ensure each mission goal can be achieved. These goals must be managed, usually through the situational understanding of not only the agent, but the tactical platform they are utilising. Situational Understanding (SU) is defined as the "product of applying analysis and judgment to the unit's situation awareness to determine the relationships of the factors present and form logical conclusions concerning threats to the force or mission accomplishment, opportunities for mission accomplishment, and gaps in information" [**?** ]. Of note, the most accepted models of SU is the Observe-Orient-Decide-Act and Endsley's Perception-Comprehension-Projection model [**?** ]. The management of goals within the competitive environment is based on the current situation being perceived. This is where agents will *reason* whether goals that were deemed achievable are now considered unattainable, or just a lower priority. With the increase of objects being situated within the environment, *tactical situations* are becoming more complex for goal reasoning to be human centric. This view is supported by [**?** ], who believes that intelligent agents can assist in streamlining goal reasoning, due to their ability and flexibility to deliberate quickly on an evolving situation. [? ] proposes the Tactical Battle Manager (TBM) that utilises a goal reasoning agent as the central component of the system. [? ] TBM utilises belief update and belief revision based on the current information presented to the goal reasoning agents. Furthermore, TBM relies on apriori set of actions. However, this is where the similarities between [? ] and our proposed G2I end. [? ] states that in order to utilise their goal reasoning agent within dynamic and unpredictable environments, some sort of scene- and situational- understanding component needs to exist within TBM. Furthermore, TBM agents need to possess the ability to reason with mental models to infer information related to the given situation, and exhibit an understanding of intent for both the human operator and the object, as well as the information gathered within the environment . Finally, [? ] states that methods need to be developed to proceed with goals deemed to be unachievable in its current form.

The G2I can assist tactical platforms in assessing whether an operator is diverging from a mission goal, since the information being generated by the user, as well as the information gathered and processed from the tactical environment can determine if an operators belief set is causing goal refinement. The nexus between goals and information can assist in the introduction of situation-aware systems at the tactical edge. [? ] agrees, however the G2I will ensure that the tracking of semantic and affective information through computational analysis is far more bounded, timely and adaptive.

## 5. Conclusion

The connection between between an agent's goal and the information requirements for achieving that goal has received little attention in the literature. This paper shows that the executability condition of plans and actions that help achieve the goal is one way to establish such a connection and that the standard BDI framework may be extended by including probabilistic beliefs, goals and actions, which allows us to compute the expected utility of goals. We also introduce the notion of information-seeking actions that can help meet these information requirement and show how they may be exploited by the agent to increase the expected utility of its goals. Finally, we discuss how this work may be further extended specifically with regards to its role in enhancing situational understanding.