

# An adaptive, probabilistic, cognitive agent architecture for modelling sugarcane growers' operational decision-making

C.S. Price <sup>a, b</sup> , D. Moodley <sup>c, b</sup> , A.W. Pillay <sup>a, b</sup> , G.B. Rens <sup>d</sup> 

<sup>a</sup> University of KwaZulu-Natal, Private Bag X 54001, 4000 Durban, South Africa

<sup>b</sup> Centre for Artificial Intelligence Research, South Africa

<sup>c</sup> University of Cape Town, Private Bag X3, Rondebosch, 7701 Cape Town, South Africa

<sup>d</sup> KU Leuven, Dept. of Computer Science, Celestijnenlaan 200A, 3001 Heverlee, Belgium

---

## ABSTRACT

Building computational models of agents in dynamic, partially observable and stochastic environments is challenging. We propose a cognitive computational model of sugarcane growers' daily decision-making to examine sugarcane supply chain complexities. Growers make decisions based on uncertain weather forecasts; cane dryness; unforeseen emergencies; and the mill's unexpected call for delivery of a different amount of cane. The Belief-Desire-Intention (BDI) architecture has been used to model cognitive agents in many domains, including agriculture. However, typical implementations of this architecture have represented beliefs symbolically, so uncertain beliefs are usually not catered for. Here we show that a BDI architecture, enhanced with a dynamic decision network (DDN), suitably models sugarcane grower agents' repeated daily decisions. Using two complex scenarios, we demonstrate that the agent selects the appropriate intention, and suggests how the grower should act adaptively and proactively to achieve his goals. In addition, we provide a mapping for using a DDN in a BDI architecture. This architecture can be used for modelling sugarcane grower agents in an agent-based simulation. The mapping of the DDN's use in the BDI architecture enables this work to be applied to other domains for modelling agents' repeated decisions in partially observable, stochastic and dynamic environments.

**Keywords:** Probabilistic BDI agent architecture, dynamic decision network, intention selection, sugarcane grower, sequential operational farmer decisions

**Categories:** • Computing methodologies ~ Artificial intelligence, Knowledge representation and reasoning, Probabilistic Probabilistic reasoning • Mathematics of computing ~ Probability and statistics, Probabilistic representations, Bayesian networks, Decision diagrams

## Email:

C.S. Price [pricec@ukzn.ac.za](mailto:pricec@ukzn.ac.za) (CORRESPONDING),

D. Moodley [deshen.moodley@uct.ac.za](mailto:deshen.moodley@uct.ac.za),

A.W. Pillay [pillayw4@ukzn.ac.za](mailto:pillayw4@ukzn.ac.za),

G.B. Rens [gavinrens@gmail.com](mailto:gavinrens@gmail.com)

## Article history:

Received: 14 June 2020

Accepted: 19 April 2022

Available online: 22 July 2022

---

Price, C.S., Moodley, D., Pillay, A.W. and Rens, G.B. (2022). An adaptive, probabilistic, cognitive agent architecture for modelling sugarcane growers' operational decision-making. *South African Computer Journal* 34(1), 152–191. <https://doi.org/10.18489/sacj.v34i1.857>

Copyright © the author(s); published under a [Creative Commons NonCommercial 4.0 License \(CC BY-NC 4.0\)](https://creativecommons.org/licenses/by-nc/4.0/).

SACJ is a publication of the South African Institute of Computer Scientists and Information Technologists.

ISSN 1015-7999 (print) ISSN 2313-7835 (online).

## 1 INTRODUCTION

Farming is a decision-intense enterprise: farmers make operational decisions daily, e.g. whether to plant, weed, irrigate or harvest a crop (Martin-Clouaire, 2017). These choices are often mutually exclusive and conflicting: choosing leisure time means not doing a farm-related task at the same time (Daydé et al., 2014). Farmers take decisions to move them closer to their goals (Martin-Clouaire, 2017). However, as the farmer receives new information from the environment, his goals can change (Daydé et al., 2014).

The environment in which farmers make these decisions is complex: it is dynamic, stochastic and partially observable (Martin-Clouaire, 2017; Martin-Clouaire & Rellier, 2009) In this environment, farmers make flexible plans which they can update as information about uncertain events (e.g. weather, availability of resources) becomes available (Daydé et al., 2015). Their decision-making process is therefore sequential, not “once-off” (Martin-Clouaire, 2017). Based on their experience and the information from the environment, they assess the situation they are currently in (Martin-Clouaire, 2017; McCown, 2012), then proactively select which goal they should pursue, and take steps to implement it, thus adapting to the changes in the environment. Farmers do this in a “satisficing” way (Simon, 1996), i.e. selecting a “good enough” option, given the available information (Robert et al., 2017).

An example of such farmers is the sugarcane grower. For these growers, the 38 week long milling season dominates the year, and they are contractually obligated to deliver cane to the mill six days a week. To stay current with cane deliveries, the grower needs to decide how much cane to burn daily so that it can be harvested and delivered on the following day. This complex cane burning decision is affected by what the grower believes about the weather forecast, how dry the cane is, and the status of his deliveries. In addition to his adapting to these dynamic and uncertain conditions, the mill may issue a call for more, or for less, than the grower’s daily cane quota. His burning decisions are sequential, in that his current decisions are dependent on previous decisions taken, and the same decision is made daily during the milling season. During this time, the grower may also have other goals, for example resting once a week, ideally on a Sunday, or resolving an emergency.

A number of computational models that aim to capture the complexity of farmers’ decision-making have been created. For example, Dury et al. (2010) and Robert, Dury et al. (2016) modelled the tactical-level crop selection decision of farmers amidst the uncertainty of expected rainfall, underground water availability and changing market prices. In these papers, the authors used the Belief-Desire-Intention (BDI) architecture (M. E. Bratman et al., 1988) to represent their farmer agents. This commonly-used architecture is based on practical human reasoning (M. Bratman, 1987). Using the BDI architecture, a course of action can be selected based on limited information, in a boundedly rational way (M. E. Bratman et al., 1988). This mimics how humans take decisions (Simon, 1996).

However, the computational models mentioned previously did not explicitly represent uncertainty or causal reasoning. Uncertain knowledge was only used prior to the agent’s decision of which crop to plant. These decisions took place a few times a year. Harvesting decisions

were taken by invoking rules in their simulations (Dury, 2011; Robert et al., 2017), By contrast, the sugarcane grower needs to reason with uncertain, causal information at an operational level, on a sequential, daily basis during the milling season, as he repeatedly determines how much cane to burn.

The BDI architecture is suitable for representing sugarcane growers, since it mimics human decision processes and has been used to model other types of farmers (e.g Dury et al., 2010; Liang et al., 2016; Robert, Dury et al., 2016; Truong et al., 2015). However, typical implementations of the BDI architecture have represented beliefs symbolically e.g. using binary logic propositions (Luz et al., 2013; Silva & Gluz, 2011; Wooldridge, 2000). This means that uncertain beliefs (Luz et al., 2013) are typically not catered for. An exception is Rens and Moodley (2017).

Bayesian networks (BNs) offer a compact way of representing causal relationships between variables under uncertainty (Darwiche, 2010). BNs with decision and utility nodes (i.e. Bayesian decision networks, or BDNs) can be used to model decision-making under uncertainty (Korb & Nicholson, 2011). BNs, and to a limited extent, BDNs have been used in BDI architectures to represent the dependencies and causal relationships in an agent's uncertain beliefs, and reason with them (e.g. Y. Chen et al., 2013; Fagundes et al., 2009). BDNs have also been used to select the plan a BDI agent should follow (Y. Chen et al., 2013). However, BDNs represent an episodic (once-off), rather than sequential, decision-making process. Dynamic decision networks (DDNs) extend BDNs to support sequential decision-making (Korb & Nicholson, 2011). A DDN which is incorporated into the BDI architecture would enable the agent to represent and reason with uncertain, causal knowledge, and to choose the most appropriate course of action at each time step.

The objective of this study is to design an adaptive, probabilistic, cognitive architecture of a sugarcane grower agent, based on the BDI architecture, using a DDN as the primary deliberation and knowledge representation and reasoning mechanism in the agent. The agent architecture needs to mimic how sugarcane growers deliberate and decide how much cane to burn prior to harvesting and delivering cane to the mill, or whether to do other tasks. It needs to demonstrate how the grower agent represents and reasons with uncertain knowledge in a stochastic, dynamic and partially observable environment. Such an agent design can be used in computational models of sugarcane growers, e.g. agent-based models of sugarcane supply chains, where growers deliver their cane to the mill. It may also be useful to others whose agents need to operate in stochastic, partially observable and dynamic environments.

The architecture was evaluated using two complex scenarios which represent realistic features of the agent's decision-making, e.g. resting one day a week on Sundays; anticipating upcoming weather and burning in advance; not being able to burn due to rain; catching up with deliveries if behind; responding to a different call for cane from the mill; and responding to an emergency. The DDN in the architecture was evaluated by sugarcane supply chain experts using criteria for validating expert-elicited BNs (Pitchforth & Mengersen, 2013).

The contribution of this paper is fourfold: firstly, we present an adaptive, probabilistic, cognitive BDI agent architecture for the sugarcane grower; secondly, we demonstrate how

DDNs may be used in the BDI architecture to select intentions; thirdly, we present an analysis of the sugarcane grower as a BDI agent; finally, we present an agent architecture representing growers making repeated operational decisions in stochastic, partially observable and dynamic environments. To our knowledge, no existing work has explored the use of a DDN to drive the inference and decision-making process of a BDI agent.

The outline of the paper is as follows: a literature review is followed by details about the case study upon which the architecture is based. The agent architecture is then presented. This includes an analysis of the sugarcane grower as a BDI agent, the architecture overview and the decision-making mechanism, which is the DDN. The agent's analysis and evaluation follows, and finally, the conclusion, limitations and future work are presented.

## 2 LITERATURE REVIEW

### 2.1 BDI agent architectures

The Belief-Desire-Intention (BDI) architecture (M. E. Bratman et al., 1988), is based on practical human reasoning (M. Bratman, 1987). This architecture represents the mental states of the agent's beliefs (what it believes about its environment and itself), its desires (the goals it wants to achieve), and its intentions (goals it is currently committed to achieving) (Rao & Georgeff, 1995). Algorithm 1 shows the BDI agent's control loop (Wooldridge, 2000). When the agent receives percepts (line 4), it revises what it believes about the world and itself (i.e. belief revision function, *brf()* in line 5). The agent deliberates about its goals and intentions (lines 6-7), and based on those deliberations, selects a goal to follow, which becomes its intention (line 7). It finds the most suitable plan (line 8), and executes it (line 9).

Using the BDI architecture, a course of action can be selected based on limited information, in a boundedly-rational way (M. E. Bratman et al., 1988). This mimics how humans take decisions, using "satisficing" (Simon, 1996). By contrast, a rational agent would have recalculated what it should do based on all the options and all information at each time step (Martin-Clouaire, 2017).

Typical implementations of the BDI architecture have represented beliefs symbolically, e.g. using binary logic propositions (Luz et al., 2013; Silva & Gluz, 2011; Wooldridge, 2000). This means that uncertain beliefs are not catered for (Luz et al., 2013). An alternative mechanism is needed to represent and reason with uncertainty.

### 2.2 Bayesian networks

#### 2.2.1 Preliminaries

Bayesian networks (BNs), also called "belief networks", are acyclic graphs which represent cause-effect relationships under uncertainty in a compact way (Darwiche, 2010; Korb & Nicholson, 2011). Variables are represented by nodes, which are linked to other variables by directed arcs. The arcs denote conditional dependencies. The BN represents a joint probability

**Algorithm 1** BDI agent's control loop (Wooldridge, 2000)

---

```

1: B := B0;                                ▷ B0 are the initial beliefs
2: I := I0;                                ▷ I0 are the initial intentions
3: while true do
4:   get next percept ρ;
5:   B := brf(B, ρ);
6:   D := options(B, I);
7:   I := filter(B, D, I);
8:   π := plan(B, I);
9:   execute(π);
10: end while

```

---

distribution over all the variables in the graph. Nodes typically have discretised mutually exclusive states. Each node also contains a conditional probability table (CPT), which defines the probability of being in each of the nodes' states, given the parent node's states. The BN's CPTs can be populated from data or from experts (Aguilera et al., 2011; Uusitalo, 2007), or a combination of both (Korb & Nicholson, 2011) making them a flexible modelling tool. Laypeople find them intuitive to understand (S. H. Chen & Pollino, 2012), and the causal links between variables reflect how humans think (Kocabas & Dragicevic, 2013).

When a BN is compiled, Bayes' rule (see Equation 1) is used by the inference engine to infer the probabilities of a variable being in a particular state (called belief updating). If there is certainty about a particular node's state, this is entered as evidence in the BN, and the state's probability changes to 100%. As the values of one node change, the values of other nodes also change using this equation (called belief propagation).

$$\Pr(A|B) = \frac{\Pr(B|A) \Pr(A)}{\Pr(B)} \quad (1)$$

Including a decision node and utility node turns the BN into a Bayesian decision network (BDN), also called an "influence diagram". The BDN determines the action with the highest expected utility of the decision by using Equation 2 (Russell & Norvig, 2009).

$$\text{Action to take} = \underset{A}{\operatorname{argmax}} EU(A|E) = \sum_i \Pr(S_i|E, A)(U, S_i|A) \quad (2)$$

where:

$E$  is the available evidence

$A$  is an action with possible outcome states  $S_i$

$U(S_i|A)$  is the utility of each outcome state, given that action  $A$  is taken

$\Pr(S_i|E, A)$  is the conditional probability distribution over the possible outcome states, given that  $E$  has been observed, and action  $A$  has been taken (Korb & Nicholson, 2011)

However, BNs and BDNs are limited in that they cannot represent how a system changes over time. A BDN which is repeated in many time slices, with arrows linking nodes in the different slices, is called a dynamic decision network (DDN) (Korb & Nicholson, 2011) or a dynamic Bayesian decision network (DBDN) (Woodberry et al., 2014). DDNs are able to represent how variables change over time (Preuveneers et al., 2016), and represent causal knowledge about the domain (Woodberry et al., 2014). They make “acceptably good” decisions (i.e. they satisfice), given the information at hand, rather than an optimal decision (Bencomo et al., 2013). They are able to operate within uncertain, partially observable environments. When presented with unexpected evidence, they can change their plans quickly (Russell & Norvig, 2009). In this way, they have adaptive behaviour (Woodberry et al., 2014). DDNs have been used to model dynamic domains such as fighter pilots’ auto-defense management (Da Costa, 1999); satisficing non-functional requirements in a self-adapting software system (Bencomo et al., 2013); ecological management of grasslands (Woodberry et al., 2014); and accessing relevant parts of patients’ electronic health records by travelling general practitioners (Preuveneers et al., 2016).

### 2.2.2 BNs in the sugarcane supply chain domain

In the sugarcane domain, Drury et al. (2016) generated a BN of factors affecting sugarcane yield using text mining. Price et al. (2018) developed a DDN to model how much cane growers decided to burn before harvesting. However, this work had the following shortcomings: the DDN could not anticipate imminent unsuitable burning weather soon enough so the grower could burn cane in advance. The “Status of deliveries” node had too few states to record how far behind/ahead the grower was with his deliveries. There was no differential behaviour between how small and large growers burn cane. The mill’s call for an increase/reduction in cane deliveries was not present. Finally, the DDN also assumed a “burn today-deliver today” rule, rather than a “burn today-deliver tomorrow” rule. A DDN presented in Price et al. (2019) addressed these issues. To our knowledge, apart from Price et al. (2018) and Price et al. (2019), no other models have been developed to represent these types of adaptive, uncertain decisions in the sugarcane supply chain.

## 2.3 Incorporation of BNs in the BDI architecture

There are three main places in the BDI architecture where different types of BNs have been used, which are outlined below. To our knowledge, DDNs have not been used in BDI architectures.



### 2.3.1 Using BNs to represent and reason with uncertain, causal knowledge

BNs have been used to represent and reason with uncertain, causal knowledge by a number of authors. A BN has been used to represent all the agent's beliefs, since the beliefs may be uncertain (Fagundes, 2007; Fagundes et al., 2009; Luz et al., 2013). Inputs from the environment are represented as entering evidence into the BN. The BN's inference engine performs the BDI belief revision function (Fagundes, 2007) to infer the most likely state of the world. A BN has also been used in a BDI agent to represent uncertain beliefs and reason diagnostically; in so doing, it determines possible explanations of the situation perceived by the agent (Wanyana & Moodley, 2021).

Jason is an implementation of AgentSpeak, an extended version of the AgentSpeak(L) language, which is based on the BDI architecture (Bordini et al., 2007). Five papers have reported on incorporating BNs into Jason. Firstly, in BayesJason (Calcin, 2006, cited in Fagundes (2007), Fagundes et al. (2009) and Kieling and Vicari (2011)), the AgentSpeak(L) grammar was extended so that each belief is represented by a node in a BN (Fagundes, 2007). The Jason API was extended so that agent perceptions, i.e. evidence, could be entered and the beliefs could be updated. The probabilistic beliefs programmed by the developer could be converted into a BN, rather than including a predefined BN into Jason (Kieling & Vicari, 2011). Secondly, Kieling and Vicari (2011) extended Jason by representing uncertain knowledge from the environment in a BN. By incorporating a plugin for probabilistic knowledge for BDI agents (named "COPA"), the evidence or updated beliefs were communicated to the Jason agent. Thirdly, in similar work, AgentSpeak(PL) offers a way of defining a BN in code (Silva & Gluz, 2011). Fourthly, Y. Chen et al. (2013) also implemented a BN to interpret the noisy input from the environment in their AgentSpeak agent. In their work, the BN interprets the environment's input signal, and the resultant inferred belief is then stored in the AgentSpeak belief base. Fifthly, a dynamic BN (DBN) was used to correct noisy percepts for an AgentSpeak agent, implemented in Jason (Coelho & Nogueira, 2015). The output was then fed into the symbolic set of beliefs via a belief revision function. The agent using the corrected perception function performed better than those which did not use this feature.

In their review of BDI agent architectures, De Silva et al. (2020) only mention using a BN to represent uncertain information in the environment, and not in other areas, which are discussed below.

### 2.3.2 Using BNs to reason about desires and intentions

BNs were used in two papers to reason about desires and intentions (Fagundes et al., 2009; Luz et al., 2013). In both of these works, the BN represented all the agent's beliefs, and desires and intentions were represented by a state in a BN node. Since the desires and intentions are modelled by mutually exclusive states in the nodes, the agent could determine which desires were feasible, based on the current set of beliefs. The agent's belief in a particular intention needed to have belief support – the agent believed that this desire could be achieved; additionally, its belief value had to be above a certain threshold. The intention with the highest

probability above this threshold was then selected as the desire to follow. Fagundes et al. (2009) used a BN with the threshold value because they wanted to model agents that were more risk averse or risk seeking: for risk averse agents, the threshold value could be higher than for the more risk seeking agents. They noted that were it not for the implementation of the risk characterisation of the agents, a BDN would be the more obvious choice for determining to which desire the agent should commit. Luz et al. (2013) extended the work of Fagundes et al. (2009). In their work, they considered alternatives to using a threshold value to select a desire from all the options. They demonstrated that a desire could be achieved if selected, even though the probability of that desire being valid could fall under the selected threshold value, based on the agent's current beliefs. They proposed three alternative methods of selecting desires which gave the agent better results than using Fagundes et al. (2009)'s. threshold method.

### 2.3.3 Using BNs to select plans

Four papers looked at how incorporating BNs into an AgentSpeak(L) agent could be used to select plans. These works all extended the Jason interpreter of AgentSpeak(L) by allowing them to include BNs to represent uncertain knowledge. In BayesJason (Calcin, 2006, cited in Fagundes (2007), Fagundes et al. (2009) and Kieling and Vicari (2011)) each belief was represented by a BN node, and the probability of the belief was used to trigger plans (Kieling & Vicari, 2011). Secondly, Kieling and Vicari (2011) added an alternative plan selection algorithm to handle the probabilistic information represented in the BN. They demonstrated that including a BN makes coding the agent much simpler, as the developer does not have to enumerate all the combinations of possibilities that the agent will face. Thirdly, Silva and Gluz (2011)'s AgentSpeak(PL) (also called JasonBayes) used the probabilistic beliefs to check a plan's context conditions or as a trigger event of the plan. Gluz and Jaques (2014) used this AgentSpeak(PL) to model emotional BDI agents. They selected the intention, based on inputs to the BDN, and used the BDN's utilities to infer the intensity of the agent's emotion as a result of following a plan. Finally, Y. Chen et al. (2013) incorporated a BDN into their AgentSpeak agent to select the appropriate plan to adopt. The plan selection part of the AgentSpeak architecture is replaced by the BDN. To our knowledge, they are they only authors who implemented plan selection with a BDN.

## 2.4 BDI representations of farmer agents

The BDI architecture has been used by a number of authors to represent farmers. This architecture has been used to model farmers' land-use change decisions (e.g. Liang et al., 2016; Muto et al., 2020; Truong et al., 2015) and crop selection decisions (e.g. Dury et al., 2010; Robert, Dury et al., 2016; Taillandier et al., 2012). These land-use and crop selection decisions are taken at a strategic or tactical level. In the crop selection decision models of Dury et al. (2010) and Robert, Dury et al. (2016), the uncertainty of expected rainfall, groundwater availability and changing market prices was considered before the crop decision was taken. This crop



selection decision was taken twice a year. However, the harvesting decision for those crops was based on rules (Dury, 2011; Robert et al., 2017), including one to wait for two days after it had rained before continuing with harvesting (Robert et al., 2017). Apart from two papers, these works did not explicitly represent uncertainty in their models. Taillandier et al. (2012)'s work is an exception here: their view was that farmers' desires were not mutually exclusive. They used a multi-criteria approach to selecting the farmers' desires. Uncertain, incomplete or missing information was represented using Shafer (1976)'s belief theory. Muto et al. (2020) used fuzzy logic to assess the expected benefit of achieving a particular goal.

## 2.5 Summary

The BDI framework for agent architectures replicates the way humans think and reason about their goals (Wooldridge, 2000). In typical implementations of the BDI framework, uncertainty is not included. BNs offer an accessible way of causal reasoning under uncertainty. Some authors have incorporated BNs into their BDI agent architectures to represent all the agent's uncertain beliefs and to perform the agent's belief revision function. Others have used BNs to reason about which desires and intentions to follow. BNs have also been used to select which plan the agent must follow. To our knowledge, DDNs have not been incorporated into the BDI architecture.

BDI architectures have been used to represent farmers as they make strategic or tactical-level decisions. In crop selection decision systems, operational decisions still seem to be represented as rules. In this study, sugarcane growers need to incorporate uncertainty into their operational decision-making process to a much greater extent. Incorporating a DDN into a BDI agent would help the grower agent to select which intention to follow, including making the pre-harvest burning decisions. To our knowledge, no one has proposed a BDI agent architecture representing the operational decisions of a sugarcane grower. Also, apart from Taillandier et al. (2012) and Muto et al. (2020), to our knowledge, no one has proposed a farmer BDI agent which reasons with uncertainty.

## 3 CASE STUDY

This study is based on a sugarcane supply chain in KwaZulu-Natal, South Africa. In this chain, the all growers prefer to rest on Sundays. During the 38-week long milling season, they are contractually obligated to deliver their daily quota of cane each day, six days a week. There are penalties for not doing so, as the mill faces mechanical problems if it is starved of cane.

Before harvesting, cane is burned to remove the leaves, as it makes harvesting, transportation and processing easier (Smithers, 2014). Cane burned on one day can be delivered on the following day. However, since the cane quality starts to decline as the cane is burned and harvested (SASRI, 2017), the grower does not want to burn too much cane, as this would negatively affect his income from the crop. The grower aims to deliver his cane "just-in-time". The mill will also not accept cane that was harvested more than four days prior to delivery.

The frequency of burning depends on the grower's size: large-scale growers burn more frequently than small-scale growers (e.g. daily versus on alternate days, respectively). The amount of cane the grower needs to burn depends on whether he is behind/ahead of deliveries and on his maximum daily delivery capacity. If he is behind, he can burn more cane, up to his maximum daily delivery capacity, to catch up. In addition, the grower's deliveries can be affected by the mill's call for cane: this is typically 100% of the grower's daily cane quota. However, the mill could call for more or for much less (e.g. 50% or 0%), depending on the situation at the mill. If the mill calls for less than 100%, the grower may not deliver more than that amount; otherwise, the grower can deliver more than 100% (e.g. if he is behind with deliveries).

Weather can also affect the cane burning decision: the grower cannot burn in rain, or in high winds. Long periods of soft soaking drizzle leave the cane wet, unable to be burned (Kadwa & Bezuidenhout, 2015). After rain, cane dries based on the daily temperature and wind. High winds pose a risk of runaway fires during burning (Kadwa & Bezuidenhout, 2015). If the grower thinks it will rain or be windy on the following day, he can burn cane in advance to mitigate these conditions (Sanjika & Bezuidenhout, 2015). If there was an emergency on the grower's farm, or that of his neighbour, like a runaway fire, the grower would help to resolve the emergency before continuing with his burning and delivery tasks.

## 4 AGENT ARCHITECTURE

In this section, the approach used to design the agent architecture is discussed first (Section 4.1), followed by an analysis of the sugarcane grower as a BDI agent (Section 4.2). The architecture is then presented (Section 4.3) followed by a description of the DDN, which selects intentions in stochastic, dynamic and partially observable environments (Section 4.4).

### 4.1 Design approach

The objective of this study is to design an adaptive, probabilistic, cognitive model of a sugarcane grower agent, based on the BDI architecture and a DDN decision model. The most complex goal of the agent is to decide whether to burn cane on a given day and the quantity of cane to burn. This goal dominates around 73% of the grower's year.

The BDI agent architecture replicates the way humans think and reason about their goals (Wooldridge, 2000). The architecture takes account of different goals that humans have, and how they deliberate to decide on which goal to try to achieve. However, deliberation could cause the current goal to be dropped in favour of another one. This practical human reasoning is how farmers also approach their decision-making problems (Robert, Dury et al., 2016).

BDI implementations do not typically support reasoning with uncertainty (Luz et al., 2013). In addition, the BDI architecture is an abstract architecture and does not prescribe how the belief maintenance, revision and intention selection mechanism should be implemented. A DDN is a suitable mechanism to use in the BDI architecture, as it represents and reasons with

uncertain beliefs, and it revises beliefs based on new information. Since DDNs are BDNs which are repeated over many time slices, connected with inter-slice arcs, they can represent more complex, time-dependent causal relationships between variables, for example where time lags are present. Compared to a BDN, which makes a single “one shot” decision, DDNs make decisions in each time slice. This enables DDNs to represent more adaptive, human-like behaviour: based on the states of variables in the previous time slice and new input from the environment, the DDN makes decisions which can correct previous errors and helps the agent to reach its goals. This represents how farmers make decisions in practice (Robert, Thomas & Bergez, 2016). In addition, DDNs represent boundedly rational reasoning, in that they consider only the most important features which could influence the decision. The uncertainty represented by DDNs shows that the agent does not have perfect knowledge.

To ensure that the agent architecture adequately represents growers in the KZN sugarcane supply chain chosen as the study site, two sugarcane growers and the mill’s cane procurement manager were interviewed<sup>1</sup>. Based on these interviews, the grower agent was analysed in three ways. Firstly, the grower’s BDI characteristics were categorised (see Section 4.2.1). Secondly, since the pre-harvest cane burning decision plays such a pivotal role for the grower’s delivery process, a flowchart of the grower’s decision-making processes was developed. This flowchart highlights the BDI characteristics of the grower, shows how different intentions are chosen, and shows the key factors growers consider when making their pre-harvest burning decisions (Section 4.2.2). Thirdly, the grower’s task environment was classified using Russell and Norvig (2009)’s characteristics (Section 4.2.3).

Based on the flowchart and interviews, a DDN of the sugarcane grower’s cane-burning decision-making was developed. This is described in Section 4.4. Finally, the cognitive loop of the probabilistic BDI agent representing a grower was developed, based on the workings of the DDN (see Section 4.3).

## 4.2 Analysis of the sugarcane grower as a BDI agent

### 4.2.1 The sugarcane grower’s BDI characteristics

The sugarcane grower was analysed as a BDI agent. The full analysis is presented in Table 1 (Appendix A). The grower has beliefs about his environment, including the weather, cane dryness, status of deliveries. He also has beliefs about things like which day of the week it is and the presence of emergencies, such as a fire danger on his neighbour’s farm. His desires or goals are issues such as remaining up-to-date with cane deliveries during the milling season; harvesting and extracting all burned cane; burning the correct amount of cane daily before harvesting; working a six-day week; and never to have runaway fires. During the milling season, the grower needs to deliver sufficient fresh cane to the mill daily. To do this, he has to burn a particular amount of cane each day. Other intentions include harvesting and extracting all available burned cane; attending to an emergency, like helping a neighbouring grower to

<sup>1</sup>Ethical clearance HSS/0204/101

contain a runaway fire; or taking a day off work. The grower’s plans are the sequences of actions the grower takes to achieve the intention. For the cane burning intention, actions would include determining which field(s) to burn on the current day, burning it, and harvesting it. He would also deliver as much of the cane harvested on the previous day(s) to the mill.

### 4.2.2 Flowchart of the sugarcane grower’s decision-making process

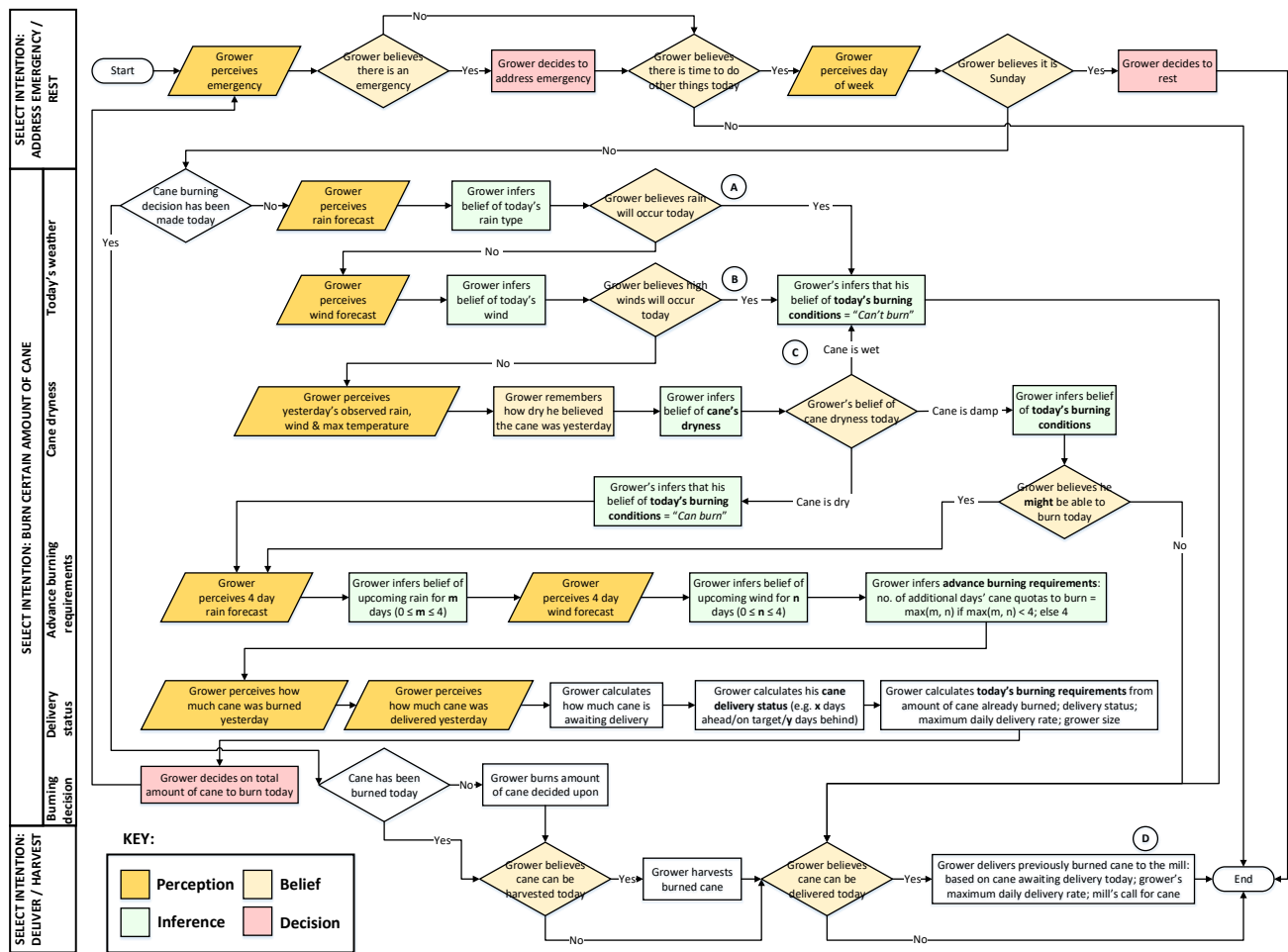


Figure 1: Flowchart of sugarcane grower’s daily decision-making process for different desires (goals). The goal “burn cane” shows more detail, as it is a more complex decision which dominates the grower’s year.

In order to design the BDI agent, the grower’s daily decision-making process was analysed in a flowchart (see Figure 1). The flowchart highlights the grower’s BDI characteristics, as outlined in Section 4.2.1. The analysis identified the perceptions, inferences and beliefs required to collect and evaluate observations from the environment, so that the grower could deliberate

about which goal to pursue, given the current situation, and select the appropriate course of action. In the diagram, inputs from the environment are shown in orange; inferences are pale green; beliefs are yellow; and the decision to choose an intention is in pink.

The grower has many different goals (desires). Three such goals are exemplified in Figure 1: resolving emergencies, like helping a neighbour to extinguish a runaway fire; not working on Sundays; burning cane; harvesting cane; and delivering cane. Other goals not displayed in the flowchart can be found in Table 1 (Appendix A). The cane burning desire is analysed in more detail, as it is the most prominent and most complex goal amongst those the grower aims to achieve.

At the top of the flowchart, the grower checks if there is an emergency, and will adopt the goal of addressing it if it is present. If the grower perceives it is a Sunday, he will adopt his goal to rest. Otherwise, the grower adopts the goal of burning cane, and needs to determine how much to burn, based on the key factors influencing that decision.

The cane burning decision has five stages, shown on the left-hand side. For the “Today’s weather” stage, the grower gets the day’s weather forecast. If the grower believes it will rain and/or there will be high winds, he infers that he cannot burn (flows labelled **A** and **B**). In this case, he will stop deliberating, and deliver as much previously burned and harvested cane as possible, based on his maximum daily delivery rate and the mill’s call for cane (label **D**). If he can burn, he turns his attention to the “Cane dryness”. Based on the previous day’s observed weather conditions and his belief of the previous day’s cane dryness, he infers how dry the cane is. Wet cane cannot be burned (label **C**), and the grower delivers as much cane as possible (label **D**).

However, if the cane is dry enough, the grower assesses the weather for the next four days. If he believes that rain or high wind will commence on the following day, he will need to infer how much additional cane to burn (“Advance burning requirements” step). The grower then considers his “Delivery status”. To deliver the freshest possible cane, the grower will not burn if enough cane is awaiting delivery. He will not burn more than his capacity to deliver. He determines his delivery status. If he is behind, he will burn slightly more cane than if he is on target, so that he can catch up. From this, he calculates his burning requirements for that day. From the cane requirements and the advance burning requirements, the grower can calculate the most appropriate amount of cane to burn on that day in the “Burning decision” stage. Cane burned on that day will be available for delivery from the following day. The grower then delivers as much cane as he can (label **D**), subject to his delivery capacity and the mill’s call for cane. Since the cane burning and delivery are independent operations, they can be thought of as a tuple (cane to burn, cane to deliver) each day.

#### 4.2.3 The sugarcane grower’s task environment

Using the flowchart (Figure 1) and analysis of the decision-making process, the task environment of the grower was analysed using the characteristics specified by Russell and Norvig (2009). The environment is partially observable, single agent, stochastic, dynamic and

discrete. In this environment, the grower takes sequential decisions. In the flowchart, the partially observable environment can be seen where the grower perceives weather forecast information, and has to infer his beliefs about that (see “Today’s weather” and “Advance burning requirements” sections). The single-agent environment is evident since the grower’s decisions are taken by him alone, and he does not have to consider others’ actions to decide what to do. The cane’s dryness follows a stochastic process (see “Cane dryness” section). Because the weather is dynamic, the grower has to infer his beliefs about it and act accordingly. Should the mill call for a different amount of cane than the default “100% of daily quota” (see flowchart, label **D**), this will dynamically change the amount of cane available for the following day’s deliveries. The environment is discrete as there are a finite number of percepts and actions that can be taken. The grower makes the same sequential decisions daily, based on his beliefs and new information from the environment.

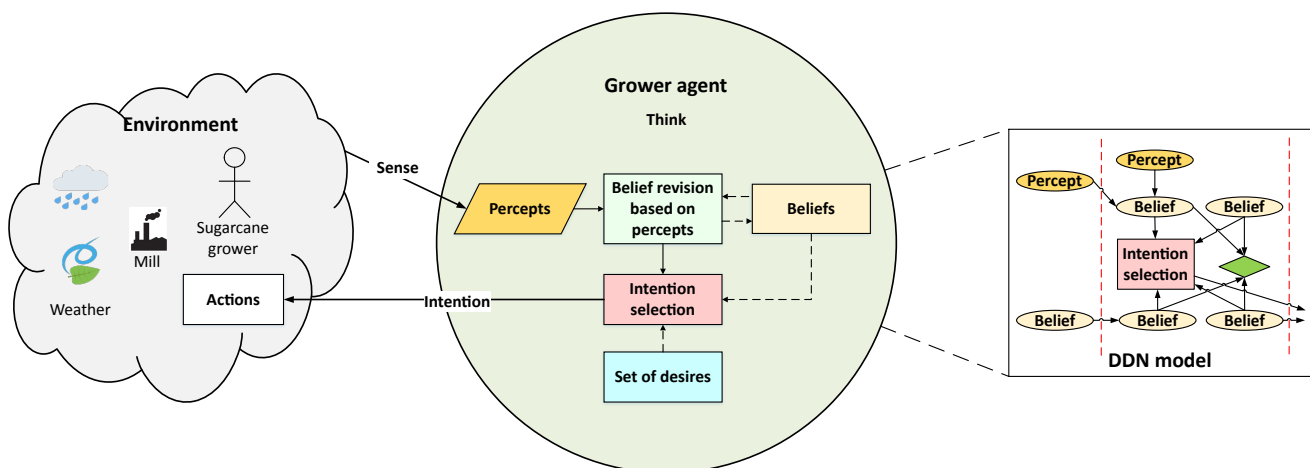


Figure 2: Probabilistic BDI agent architecture

### 4.3 Architecture overview

Based on the analysis presented in Section 4.2 an agent architecture was designed to represent sugarcane growers’ adaptive behaviour during the milling season. The overview of the agent architecture is shown in Figure 2. In the grower agent, the diagram shows how it gets percepts from the environment and updates its beliefs. Based on its new beliefs, it determines the most appropriate goals (desires) to focus upon. When the agent selects a goal, this becomes its intention. This intention is output, so that the sugarcane grower in the environment can select the most appropriate plan and act upon it.

The DDN is used to achieve a number of aspects of the BDI architecture: representing causal knowledge about the domain, with which the agent can reason; recording percepts as evidence in the DDN; revising the agent’s beliefs, based on the evidence, using the DDN’s Bayesian inference algorithm. For more complex deliberations, e.g. deciding how much cane to burn,



the DDN is also used to determine which desires (goals) are possibilities, since they have belief support; and to select the most appropriate intention, based on the desires, using the DDN's utility maximisation algorithm. This last feature selects from among mutually exclusive desires.

The control loop for the probabilistic BDI agent of a sugarcane grower agent is shown in Algorithm 2. This algorithm is based on the BDI agent control loop of Wooldridge (2000) (Algorithm 1). The inputs to the algorithm are the DDN, which contains the beliefs of the agent, and the desires relating to burning between 0 and 4 days' cane quota. Other desires not contained in the DDN, for example, responding to an emergency, and resting on a Sunday, are also inputs. Other inputs are: the grower's size (large/small) and the grower's maximum daily delivery rate: these two values are assumed to remain constant over the duration of the milling season. Three types of observations are also input from the environment: those about weather, delivery and the presence of emergencies/day of rest. The weather observations include the current day's forecast of rain, wind and maximum temperature; the observations of these values for the previous day; and the forecast of upcoming rain/wind starting on the following day. The output is the selected intention.

At time  $t = 0$ , the beliefs are stored in the DDN and no intention is selected. There is a pre-defined set of Desires from which the agent can select. In line 7, whether the grower agent represents a large or small-scale grower is initialised, as well as its maximum daily delivery rate. The grower agent gets a number of observations from the environment (line 10). These include the forecast for the current day, the previous day's observed weather, the weather forecast for the next four days, the grower's deliveries for the previous day, the previous day's burned cane supply, the previous day's burning decision, and the previous day's delivery status. The day of the week and presence of emergencies are also observed. Observations are discretised to fit into one of the DDN node's state categories, if applicable (line 11). These processed observations form the agent's percepts. The cane burning requirements for the current day are calculated *CalcBR()* in line 12, is described further in Equation 3 Section 4.4). The percepts (weather and cane burning requirements) are entered as evidence into the DDN (line 13). The Bayesian inference engine, which is the agent's belief revision function, revises the agent's beliefs in line with the new evidence (line 14). The grower agent must first check if an emergency is present, or if the day of the week is a Sunday. If either of these are true, the agent's intention will be set accordingly (lines 15-16). If neither is true, the cane burning intention is selected by the DDN: it is used to infer the optimal amount of cane to be burn under the current conditions, i.e. the decision state with the maximum utility value (line 17). The algorithm outputs the optimal intention. The Grower can then choose the appropriate plan for that intention and implement actions in the environment to execute the plan.

#### 4.4 The decision-making model

Using the analysis of the grower's decisions presented in the flowchart (Figure 1) and interviews, a DDN was developed (see Figure 3) to represent the grower's most complex decision

**Algorithm 2** Probabilistic BDI agent's control loop, adapted from Wooldridge (2000)

---

```

1: Input: DDN, containing initial beliefs and desires relating to cane burning;
    Other desires the agent can consider;
    GrowerSize, MaxDailyDeliveryRate;
    Observations about weather:
        • current day's weather forecast; rain/wind forecast for next 4 days; previous day's observed weather
          data;
    Observations about delivery information:
        • CaneAwaitingDeliveryt-1; CaneDeliveredt-1; CaneBurnedt-1; DeliveryStatust-1
    Other observations:
        • EmergencyIsPresentt; DayOfWeekt
2: Output: Selected intention
3:  $t = 0$  ▷ time
4:  $B_0 =$  Beliefs stored in the DDN
5:  $D =$  set of desires
6:  $I_0 = \emptyset$ 
7: set GrowerSize (large/small) and MaxDailyDeliveryRate
8: while true do
9:    $t = t + 1$ 
10:  get observations from environment
11:   $P_t =$  discretised observations ▷ percepts
12:  BurningRequirementst = calcBR(CaneAwaitingDeliveryt, DeliveryStatust, MaxDailyDeliveryRate,
    GrowerSize)
13:  enter relevant percepts  $P_t$  into DDN as evidence
14:   $B_t =$  revised beliefs, using Bayesian inference
15:  if EmergencyIsPresentt then
     $I_t =$  address emergency
    enter "Burn no cane" into DDN decision node as evidence
16:  else if DayOfWeekt = Sunday then
     $I_t =$  rest
    enter "Burn no cane" into DDN decision node as evidence
17:  else if it is MillingSeason then
     $I_t =$  DDN decision state with highest utility
18:  end if
19:  output  $I_t$ 
20: end while

```

---

on a given day – i.e. how much cane to burn on a given day. A description of the DDN's nodes and their states can be found in Table 2 (Appendix B). Figure 3 shows causal nodes and the decision node "Intention selection" for two days: the current day on the right and the previous day on the left. The intention selection chooses the most appropriate amount of cane to burn on the current day. This intention selection on the current day considers the conditions of the current and previous days.

The orange nodes in the DDN are where the percepts are entered as evidence (see line 13 of

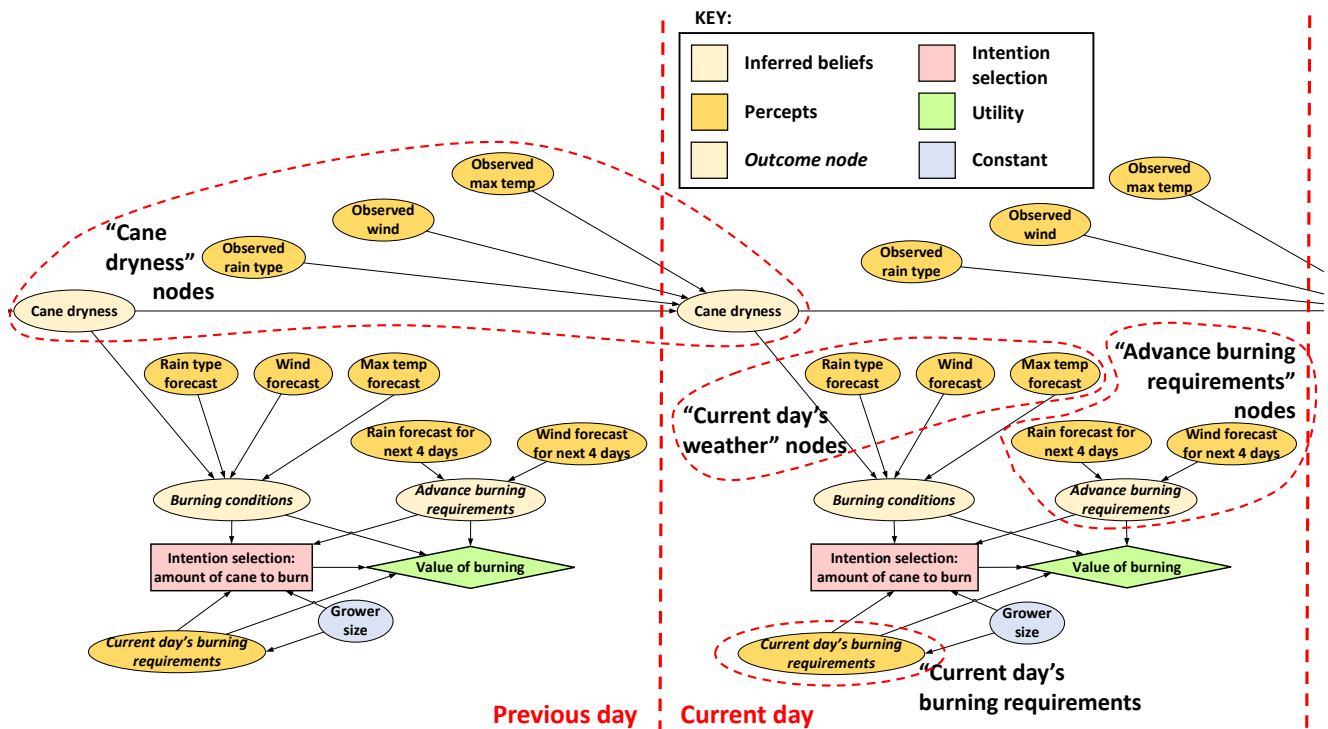


Figure 3: The DDN which represents the grower agent’s mechanism for selecting an intention for the goal “burn certain amount of cane”

Algorithm 2). These percepts fall into four categories: those necessary for understanding the agent’s belief of the “Current day’s weather”, given the forecast; those necessary for determining the “Advance burning requirements”, given a four-day forecast; those for determining the “Current day’s burning requirements”; and those for determining the “Cane’s dryness”. In the case of the latter, “Cane dryness” is a hidden node which is affected by the previous day’s dryness, and the observed rain, wind and temperatures of the previous day. The nodes for inferring the current day’s “Cane dryness” thus span two time slices.

On entering the percepts, the Bayesian inference engine revises its beliefs (line 14 of Algorithm 2). The revised beliefs propagate through the network using Bayes’ rule (see Equation 1). The belief propagation culminates in three outcome nodes shown in italics: *Burning conditions* summarises the weather forecast and cane dryness to assess whether burning can take place or not on that day. *Advance burning requirements* summarises the need to burn a number of day’s cane quota in advance, due to forecast high winds or rain which start on the following day. *Current day’s burning requirements* summarises how many days’ cane quota to burn for that day, assuming that burning is possible. *Current day’s burning requirements* is based on the number of days’ cane quota already burned and awaiting delivery, the number of days’ cane quota that the grower is behind or ahead of deliveries, the maximum number of days’ quota that the grower can deliver on any one day, and the grower’s size (see Equation 3

to Equation 5). The input to Current day's burning requirements is calculated by the function  $CalcBR()$  in Equation 3 which is expanded upon in Algorithm 3 in Appendix C. If the grower is behind with deliveries, the amount required for that day is the maximum that the grower can deliver in one day. The function which calculates the current delivery status,  $CalcDS()$  in Equation 5, can also be found in Appendix C (Algorithm 4).  $MaximumDailyDeliveryRate$  and  $GrowerSize$  (large or small) are constant for each grower.

$$\begin{aligned}
 BurningRequirements_t = & CalcBR( CaneAwaitingDelivery_t, \\
 & DeliveryStatus_t, \\
 & MaximumDailyDeliveryRate, \\
 & GrowerSize)
 \end{aligned} \tag{3}$$

where

$$\begin{aligned}
 CaneAwaitingDelivery_t = & CaneAwaitingDelivery_{t-1} \\
 & - CaneDelivered_{t-1} \\
 & + CaneBurned_{t-1}
 \end{aligned} \tag{4}$$

and

$$\begin{aligned}
 DeliveryStatus_t = & CalcDS( DeliveryStatus_{t-1}, \\
 & CaneDelivered_{t-1}, \\
 & DayOfWeek)
 \end{aligned} \tag{5}$$

The first two of the DDN's three outcome variables, *Burning conditions*, and *Advance burning requirements* play the role of "situation detection" (Adeleke & Moodley, 2015) for the agent. These variables reflect the agent's belief of whether burning can or cannot take place that day; and the agent's belief of future burning conditions due to forecasted inclement weather, starting on the following day. Based on the outcome, the *Current day's burning requirements* is evaluated. This follows the flow of decision-making in the flowchart (Figure 1).

The three outcome nodes feed into the decision node "Intention selection", which selects the mutually exclusive intention of how much cane to burn (Algorithm 2, line 17). Possible intention options are to burn 0; 1; 1.25; 1.5; 1.75; 2; 2.25; ... 4 days' cane quota. These intentions are chosen by the DDN's maximum expected utility calculation (Russell & Norvig, 2009) (see Equation 2).

The DDN was developed iteratively using the knowledge engineering approach to developing BNs (Korb & Nicholson, 2011). The DDN's structure was developed by hand. The wind and maximum temperature nodes' states were discretised automatically based on observed weather data for the mill area (SASA, 2018). Node and state names were checked for vagueness (Marcot, 2017) and edited if necessary. States were checked to ensure that they covered

the full range of possible values of a node (Korb & Nicholson, 2011; Pitchforth & Mengersen, 2013). The CPTs values were developed by hand, based on business rules, using an anchoring technique (Korb & Nicholson, 2011) to compare different options. Each part of the DDN was tested before adding more functionality to it (Hänninen, 2014). The DDN was designed and tested in ‘HUGIN Expert A/S’ (2019). The assumptions underlying the DDN can be found in Appendix D, Table 3.

## 5 ANALYSIS AND EVALUATION OF THE AGENT ARCHITECTURE

The evaluation of the probabilistic agent using two scenarios is presented in Section 5.1 and the analysis of the agent’s decision-making process is discussed in Section 5.2. This includes a brief description of how the DDN was validated using Pitchforth and Mengersen (2013)’s framework for expert-elicited BNs.

### 5.1 Evaluation of the probabilistic BDI agent

The probabilistic BDI agent was evaluated by running two complex scenarios. These scenarios were chosen to demonstrate the features of the agent’s decision-making, e.g. resting one day a week on Sundays; anticipating upcoming weather and burning in advance; not being able to burn due to rain; catching up with deliveries if behind; responding to a different call for cane from the mill; and responding to an emergency. The scenarios illustrate how the agent responds and adapts to percepts from its partially observable, stochastic and dynamic environment. Based on the percepts, the agent needs to select an intention from among mutually-exclusive desires, i.e. to rest (on Sundays), burn a particular amount of cane, or attend to an emergency. Both scenarios show the behaviour of the agent of a large grower, with a maximum daily delivery rate of 150% of a day’s cane quota – i.e. 1.5 days’ cane quota. It is assumed that the forecasted weather materialises, and the grower in the environment accepts and acts on the intention identified by the agent. The grower delivers as much cane as possible each day, except for rest days (Sundays).

The run for Scenario 1 starts on Saturday. The grower is one day ahead with deliveries, and the cane is damp. One day’s cane quota is awaiting delivery. The agent needs to infer the cane dryness to decide if burning is possible – if it is not a rest day. Drizzle materialises on Tuesday and a thundershower on Friday (blue rectangles at top). On Thursday, the temperature rises to high (also highlighted in a blue rectangle). In addition to these weather changes, on Friday, the mill calls for only 50% of the normal cane quota (blue rectangle at bottom). The run for Scenario 2 has the exact same conditions as Scenario 1, except that an emergency is perceived on the Monday in addition to the weather conditions and mill’s call for cane outlined above.

Scenario 1 (see Figure 4) shows how the grower agent reacts and adapts to the percepts from the environment over 10 days starting on a Saturday. The days are columns of the table. In addition, the first column shows the BDI aspects of the run: percepts (P), beliefs (B) and intentions (I) at 6am. The second column shows how these are implemented by the DDN:

evidence (E), hidden nodes (HN), outcome nodes (ON) and decision nodes (DN). The agent's intention selection of the amount of cane which needs to be burned is based on the DDN decision (bottom bar chart). In this chart, intentions of cane over 2.5 days' quota to be burned were not selected. They are thus not displayed in the chart to simplify the diagram.



Figure 4: Scenario 1 results. Key: For BDI (first column): P = Percept; B = Belief; I = Intention; for BN (second column): E = Evidence; HN = Hidden node; ON = Outcome node; DN = Decision node

On Day 1, the grower is one day ahead with deliveries and has one day's cane awaiting delivery. Consequently, the *Current day's burning requirement* is that the agent needs to burn no cane. In addition, based on the agent's belief of cane dryness (damp cane), the agent infers that the *Burning conditions* for the day are that cane cannot be burned. The best intention for the agent to follow is to burn no cane, so the agent selects that intention (third row from bottom). The one day's quota that was awaiting delivery is delivered (bottom row). This means that the grower is still one day ahead. By Sunday, the cane has dried sufficiently, and even though conditions are suitable for burning, the agent knows that the grower prefers to rest and not burn or deliver on that day. On Monday, the grower is still 1 day ahead, as deliveries are not expected on a Sunday. On Monday, the cane is dry enough to burn, and the agent perceives that it will rain for one day starting on the following day (*Advance burning requirements*). The agent's decision-making mechanism selects the intention to burn 2 days' cane (1 day's quota for Monday; 1 day in advance for Tuesday). On Monday, since there was no cane awaiting delivery, none was delivered. As a result, by Tuesday, the grower is



on target with deliveries. On Tuesday, the anticipated rain (drizzle) materialises, meaning that no burning can take place. 1.5 days' quota is delivered (the grower's maximum delivery capacity), after which he has caught up half a day by the beginning of Wednesday, and is 0.5 days ahead. The agent has thus adapted to the upcoming weather forecast. Tuesday's drizzle causes the cane to be too wet to burn on Wednesday. The remaining 0.5 day's cane, burned on Monday, is delivered on Wednesday. By Thursday, the agent is on target, but has no cane awaiting delivery: the grower will soon fall behind. On Thursday, the cane is marginally dry enough to burn. Rain is forecast for one day starting on the following day. Therefore 2.5 days' quota is burned (1.5 days' quota to catch up with deliveries, and 1 day's quota in advance for Friday's expected rain). On Friday, burning cannot proceed because of the rain. In addition, the mill calls for 50% of a day's quota (blue rectangle at bottom). The grower can only deliver 0.5 days' cane, even though 2.5 is available. This leaves the grower 1.5 days behind by Saturday. By Saturday, the cane has dried quickly from the thundershower, and cane can be burned again: 1.5 days' quota. The grower once again adaptively catches up with deliveries on Saturday, and by Sunday, it is 1 day behind. Once again, the agent chooses to rest on the Sunday. On Monday, cane can be burned again. There are 2 days' quota awaiting delivery. The agent therefore suggests burning 1 days' quota.

In Scenario 2 (see Figure 5), the agent perceives an emergency on Monday (Day 3). The effects of this emergency are highlighted in red rectangles in Figure 5. Instead of burning cane for Monday, and in advance of the predicted rain on Tuesday, the agent selects the intention to attend to the emergency (third last row). This means that no cane is burned on Monday, meaning that no cane is delivered on Tuesday, and the grower falls further and further behind: by day 10, the grower is 3 days behind with deliveries.

## 5.2 Analysis of the agent's decision-making process

### 5.2.1 Analysis of scenarios

The two scenarios outlined in Section 5.1 illustrate the working of the agent's control loop (Algorithm 2). They also provide a mapping of how a DDN can be used in the BDI architecture to select intentions. To our knowledge, this is the first time that such a mapping has been described.

"Uncertainty calls for flexibility" (Simon, 1996, p 43). The two scenarios demonstrate the agent's flexibility in being able to react to its partially observable, stochastic and dynamic environment. It flexibly selects intentions like burning a particular amount of cane, resting on a Sunday, and attending to an emergency where necessary. The intentions demonstrated in the two scenarios show how the agent architecture works. The architecture can be expanded to include other intentions. The DDN enhances the agent's deliberations in that it is only used to identify which is the most appropriate intention to adopt in a complex, uncertain environment. Easier intention selections can be made with if-then-else statements.

The agent is able to recover from sudden changes in the environment, e.g. an emergency (Day 3, Scenario 2); weather (Days 4 and 7, both scenarios); and the mill's reduced call for



Figure 5: Scenario 2 results with differences from Scenario 1 highlighted in red. Key: For BDI (first column): P=Percept; B=Belief; I=Intention; for BN (second column): E=Evidence; HN=Hidden node; ON=Outcome node; DN=Decision node

cane (Day 7, both scenarios). This confirms Russell and Norvig (2009)’s statement that DDNs allow agents to change quickly when faced with sudden changes. Given new percepts, the agent is able to adapt and devise a course of action which would get it to its goals. For example, the agent anticipates rain, and burns cane in advance (Days 3 and 6, Scenario 1; Day 6, Scenario 2). This reactive and proactive behaviour mimics farmers’ behaviour when making operational decisions (Martin-Clouaire, 2017), allowing them to deliver their cane “just-in-time”. Even if the grower did not accept the agent’s suggested intention but rather did something else, the agent would still be able to recover and decide what to do based on new inputs from the environment. This aligns with Simon’s opinion of effective system behaviour, when he said that “a system can generally be steered more accurately if it uses feed forward, based on prediction of the future, in combination with feedback, to correct the errors of the past” (Simon, 1996, p 36).

Understanding how important it is to deliver fresh cane to the mill in a “just-in-time” manner, the agent selects the intentions to burn cane which allow the grower to catch up with deliveries adaptively if he is behind. However, the grower’s ability to catch up are limited by his maximum delivery capacity (see cane quota delivered in Scenario 1, Day 8). If the grower has more than his daily delivery capacity awaiting delivery, the agent will not advise that

additional cane be burned, even though it recognises that the grower needs to catch up with deliveries (e.g. Scenario 2, Day 10). In this way, the agent only suggests amounts of cane to be burned that will ensure the freshest cane is delivered.

### 5.2.2 Incorporation of DDN in BDI architecture

The adaptive, probabilistic, cognitive agent architecture presented in Section 4 uses the DDN to represent and reason with uncertain, causal knowledge; interpret noisy inputs from the environment; and select the most suitable intention.

The DDN reasons with uncertain beliefs and stores the agent's beliefs in a causal manner. This is also how the BNs in Calcin (2006), Fagundes (2007), Fagundes et al. (2009), Kieling and Vicari (2011), Luz et al. (2013), Silva and Gluz (2011) and Wanyana and Moodley (2021) were used in BDI architectures. The DDN implicitly interprets noisy information from the environment, as do Y. Chen et al. (2013) with a BN and Coelho and Nogueira (2015) with a DBN.

In addition to these features, the DDN provides a mechanism for the agent to infer probabilistic beliefs, including those of hidden variables over time (e.g. cane dryness), and of outcome variables (e.g. the *current day's burning conditions*, *burning requirements* and *advance burning requirements*). The three outcome variables, shown in italics in Figure 3 to Figure 5, are used to select the most appropriate intention by the DDN, and underlie the agent's bounded rationality: the selection of the intention to burn cane depends only on these three variables. The agent is also boundedly rational since it infers a result which is good enough – i.e. it satisfies – even some of the beliefs with which it reasons are uncertain. This agrees with Bencomo et al. (2013). The *Burning conditions* and *Advanced burning requirements* outcome nodes play the role of situation detection (Adeleke & Moodley, 2015), i.e. whether the agent can burn cane or not on the current day, and on the following day(s). This situation detection mirrors how farmers approach decision-making (Martin-Clouaire, 2017; McCown, 2012).

The DDN calculates the best intention to pursue amongst the cane burning decisions using each time slice's decision and utility nodes and utility expectation formula (Equation 2). Fagundes et al. (2009) suggested that this was the obvious way to select intentions, but instead of using a BDN, they used a BN. Obtaining the maximum utility in each time slice does not reduce the way that the DDN models the grower's boundedly rational behaviour. The DDN only considers selected variables (e.g. weather) as input, rather than all the variables available. This mimics how the growers make decisions; they are always monitoring weather forecasts. The DDN also reasons with uncertain knowledge: i.e. the agent does not have perfect knowledge.

The DDN has other benefits for the architecture: comparing the DDN to Fagundes et al. (2009)'s suggestion of using a BDN, the dynamic nature of the DDN – i.e. its inter-slice arcs – enable it to keep track of variables that affect others in the next time slice (Preuveneers et al., 2016). As such, the DDN is a richer way of modelling the agent's uncertain beliefs and knowledge, as well as complex causal relationships between variables. DDNs can also represent time lags which are present in the farmer's decision-making environment (Martin-

Clouaire, 2017).

The architecture is flexible, in that simple intentions, like resting, or attending to an emergency, can be selected without using the DDN. However, for more complex deliberations, like selecting the intention of how much cane to burn, the adaptive nature of the agent is made possible due to the sequential decision-making mechanism of the DDN. This concurs with Martin-Clouaire (2017) in that operational decisions are not “one-shot” decisions, as one would find for a strategic time horizon. Rather, they are “process oriented”. The intentions which the agent selects are those which can correct previous errors and help the agent to reach its goals (Robert, Thomas & Bergez, 2016). The DDN also offers mechanisms for representing the time lags (Martin-Clouaire, 2017) in the farmer’s decision process, and can represent the farmer’s cognitive processes.

One of the disadvantages of using a DDN is the fact that the BN designer has to enumerate all the CPT options (Kieling & Vicari, 2011). However, this could be seen in a positive light as it ensures that the designer considers all the combinations of options in a systematic manner.

### 5.2.3 DDN validation

To ensure that the DDN reflects the sugarcane growers’ cognitive processes adequately, the DDN was evaluated according to Pitchforth and Mengersen’s validation framework for expert-elicited Bayesian networks (Pitchforth & Mengersen, 2013). The behaviour of the DDN aligns with the literature on how growers burn their cane, confirming nomological validity. All nodes and states necessary to produce the output are present and no unnecessary nodes or states were present, confirming content validity.

The structure, discretisation and parameterisation of the DDN was compared to other BNs of similar domains to determine convergent validity. The summary or outcome nodes follow the “definitional/synthesis idiom” of BNs (Kyrimi et al., 2020). The previous day’s observed weather and “Cane dryness” nodes together causing the current day’s “Cane dryness” node follow the “cause-consequence idiom” of BNs (Kyrimi et al., 2020). The sequential structure of the DDN is similar to the farmer example given in Clemen (1991). The states of the nodes in the *Current day’s burning requirements* and decision nodes are both discretised with the same accuracy of 0.25 day’s cane quota. These factors confirm the DDN’s convergent validity.

Predictive validity evaluates the DDN’s behaviour and output, and consists of three subcategories. Firstly, behaviour sensitivity evaluates the BN’s sensitivity to evidence; to changes in parameters; and the value of information – only applicable to BDNs and DDNs (Korb & Nicholson, 2011). Secondly, extreme conditions tests evaluate how the network/part thereof behave when extreme values are input. Thirdly, qualitative features tests evaluate of the DDN’s output by modellers and experts.

Behaviour sensitivity tests were conducted on the outcome nodes leading to the decision node. For example, the “sensitivity set” (Coupé & van der Gaag, 2002) for the node “Burning conditions, Day 1” (circled in red, Figure 6) was evaluated in Hugin. The more intense the colour of the node, the more the target node is sensitive to the values of the intensely coloured



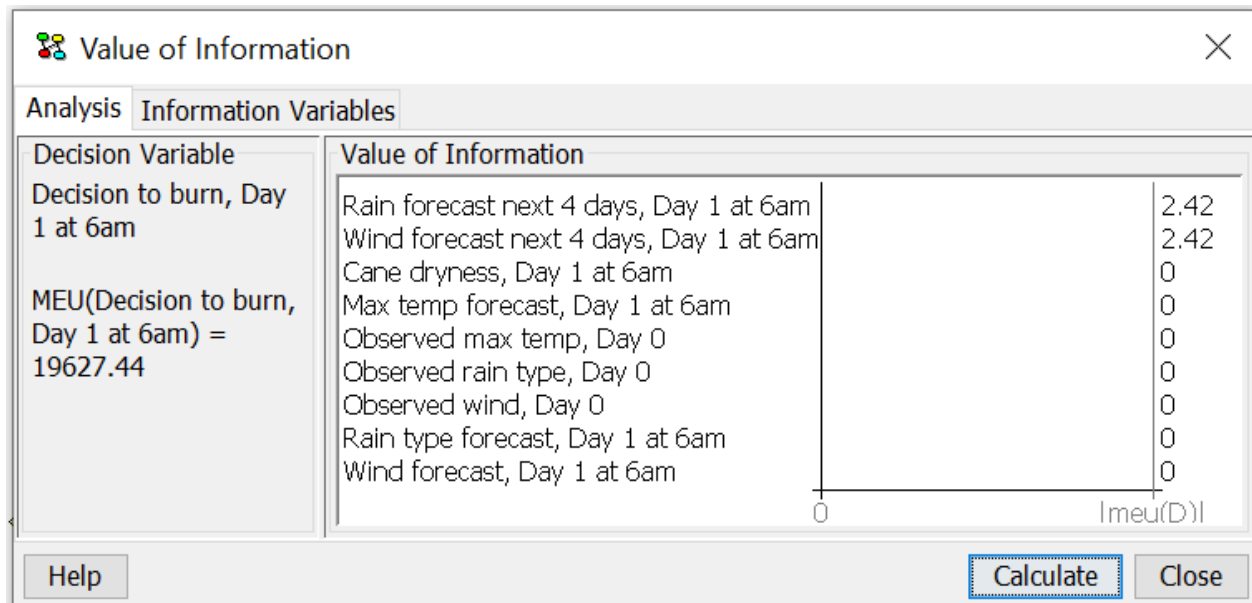


Figure 7: Value of information for current day decision variable

the nodes, the participants agreed that the model was behaving in the way that was expected. However, the group identified that the CPT values in the “cane dryness” nodes were not allowing the cane to dry as fast as they would expect. These CPT values have since been updated in the DDN. Two focus group participants thought that the model represents what happens in reality in the mill’s supply chain “Well”, whereas one scored it as “Neutral”. The fourth participant said that the model “represents 80% of what happens in reality.” Based on the validation of the DDN and the feedback from the focus group participants, the DDN can be seen as an effective way of representing the cognitive processes that sugarcane growers use to make pre-harvest burning decisions. However, the DDN should be tested in other sugarcane supply chains before deployment into industry.

The agent architecture presented here, and the scenarios run, represent how sugarcane growers behave when taking repeated operational decisions in a stochastic, partially observable and dynamic environment. This contributes to the literature on modelling farmers’ repeated, sequential decisions at an operational level, where uncertainty and causal reasoning need to be represented explicitly. To our knowledge, such architectures for farmers’ repeated operational decisions have not been presented before.

## 6 CONCLUSION, LIMITATIONS AND FUTURE WORK

The BDI architecture is abstract and does not stipulate how the belief maintenance, revision and intention selection mechanism should be implemented. This paper has presented an architecture for a BDI agent which makes sequential, repeated decisions about which mutually



exclusive intention to select in a stochastic, partially observable and dynamic environment. A DDN is used to represent and reason with uncertain knowledge in a probabilistic manner, and determine the best intention to follow where complex, causal and probabilistic knowledge is involved. The DDN is used to track variables which change over time, but where simpler intention decisions need to be made, these can be made without the DDN, making it a flexible architecture. The scenario runs show a mapping of how a DDN can be used within the BDI architecture.

The architecture has been evaluated by modelling how sugarcane growers from a KwaZulu-Natal, South Africa, sugarcane supply chain decide what intention to select during the milling season. These growers burn cane prior to harvesting and delivering it in a “just-in-time” manner. In addition, they rest on Sundays and attend to emergencies should they arise. The scenario runs show how the grower agent reacts and adapts to a number of percepts from the environment; the agent also proactively pursues a goal and burns cane in advance, should inclement weather be imminent. Sugarcane supply chain role players from a KwaZulu-Natal mill area evaluated the DDN and found it to represent their pre-harvest burning decision-making process, confirming the agent’s cognitive representation of their cane-burning behaviour during the milling season.

The DDN has been modelled to identify situations in which the agent can and cannot burn. This mimics farmers’ assessing of situations before deciding what to do (Martin-Clouaire, 2017; McCown, 2012). DDNs can be used to model the time lags which are present in farmers’ decisions (Martin-Clouaire, 2017).

The agent described here is boundedly rational, in that it does not use all possible information to select the intention at each time step. Rather it uses the most important information, which is also uncertain. This mimics how farmers make decisions (Martin-Clouaire, 2017). As far as we are aware, computational models of farmers’ operational decision-making have not represented and reasoned under uncertainty. As a computational model of repeated, sequential operational decisions, in a stochastic, partially observable and dynamic environment, this architecture thus contributes to the literature on modelling farmers’ repeated operational decisions.

The architecture described here can be expanded to include other intentions. The agent described here can be used to represent growers in other sugarcane supply chains by adjusting the CPT values of the DDN, if necessary, and testing it in those settings. The architecture could be applied to other cases where repeated sequential decisions are being made in a stochastic, partially observable and dynamic environment. The adaptive, probabilistic, cognitive BDI architecture presented here would be useful to represent a sugarcane grower agent in an agent-based simulation of a sugarcane supply chain.

## ACKNOWLEDGEMENTS

The authors gratefully thank: the cane procurement managers and focus group participants of the KwaZulu-Natal sugarcane supply chain for their inputs into and feedback about the model;

University Capacity Development Programme (UCDP) for funding the upgrade to Hugin 8.6 and 8.8 and writing retreats; and College of Law and Management Studies, UKZN, for funding transport to the mill. The authors also thank the reviewers for their comments which have improved the paper.

## References

- Adeleke, J. A. & Moodley, D. (2015). An ontology for proactive indoor environmental quality monitoring and control. *2015 Annual Research Conference on South African Institute of Computer Scientists and Information Technologists (SAICSIT '15)*, 1–10. <https://doi.org/10.1145/2815782.2815816>
- Aguilera, P. A., Fernández, A., Fernández, R., Rumí, R. & Salmerón, A. (2011). Bayesian networks in environmental modelling. *Environmental Modelling & Software*, 26(12), 1376–1388. <https://doi.org/10.1016/j.envsoft.2011.06.004>
- Bencomo, N., Belaggoun, A. & Issarny, V. (2013). Dynamic decision networks for decision-making in self-adaptive systems: A case study. *2013 8th International Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS)*, 113–122. <https://doi.org/10.1109/SEAMS.2013.6595498>
- Bordini, R. H., Hübner, J. F. & Wooldridge, M. (2007). *Programming multi-agent systems in AgentSpeak using Jason*. Chichester, UK: John Wiley & Sons. <https://doi.org/10.1002/9780470061848>
- Bratman, M. (1987). *Intention, plans, and practical reason*. Cambridge MA: Harvard University Press.
- Bratman, M. E., Israel, D. J. & Pollack, M. E. (1988). Plans and resource-bounded practical reasoning. *Computational Intelligence*, 4(3), 349–355. <https://doi.org/10.1111/j.1467-8640.1988.tb00284.x>
- Calcin, O. G. P. (2006). BayesJason: Bayesian networks and Jason interpreter. *6th Iberoamerican Workshop on Multi-Agent Systems (IBERAGENTS 2006)*.
- Chen, S. H. & Pollino, C. A. (2012). Good practice in Bayesian network modelling. *Environmental Modelling & Software*, 37, 134–145. <https://doi.org/10.1016/j.envsoft.2012.03.012>
- Chen, Y., Hong, J., Liu, W., Godo, L., Sierra, C. & Loughlin, M. (2013). Incorporating PGMs into a BDI architecture. In G. Boella, E. Elkind, B. T. R. Savarimuthu, F. Dignum & M. K. Purvis (Eds.), *PRIMA 2013: Principles and practice of multi-agent systems* (pp. 54–69). Berlin Heidelberg: Springer-Verlag. [https://doi.org/10.1007/978-3-642-44927-7\\_5](https://doi.org/10.1007/978-3-642-44927-7_5)
- Clemen, R. T. (1991). *Making hard decisions: An introduction to decision analysis*. Belmont CA: Duxbury.
- Coelho, F. & Nogueira, V. (2015). Probabilistic perception revision in AgentSpeak (L). In Q. Chen, P. Torroni, S. Villata, J. Hsu & A. Omicini (Eds.), *Prima 2015: Principles and practice of multi-agent systems* (pp. 613–621). Cham, Switzerland: Springer. [https://doi.org/10.1007/978-3-319-25524-8\\_44](https://doi.org/10.1007/978-3-319-25524-8_44)

- Coupé, V. M. H. & van der Gaag, L. C. (2002). Properties of sensitivity analysis of Bayesian belief networks. *Annals of Mathematics and Artificial Intelligence*, 36(4), 323–356. <https://doi.org/10.1023/A:1016398407857>
- Da Costa, P. C. G. (1999). *The fighter aircraft's autodefense management problem: A dynamic decision network approach* (Master's thesis). Department of Systems Engineering and Operations Research, George Mason University. Fairfax VA.
- Darwiche, A. (2010). Bayesian networks. *Communications of the ACM*, 53(12), 80–90. <https://doi.org/10.1145/1859204.1859227>
- Daydé, C., Couture, S., Garcia, F. & Martin-Clouaire, R. (2014). Investigating operational decision-making in agriculture. In D. P. Ames, N. W. T. Quinn & A. E. Rizzoli (Eds.), *7th International Congress on Environmental Modelling and Software (iEMSs)* (pp. 2188–2195).
- Daydé, C., Couture, S. & Martin-Clouaire, R. (2015). Interview-based structuring of operational decision-making by farmers. *5th International Symposium for Farming Systems Design (AGRO2015)*.
- De Silva, L., Meneguzzi, F. R. & Logan, B. (2020). BDI agent architectures: A survey. In B. Bessiere (Ed.), *International Joint Conferences on Artificial Intelligence (IJCAI 2020)*.
- Drury, B., Rocha, C., Moura, M.-F. & de Andrade Lopes, A. (2016). The extraction from news stories a causal topic centred Bayesian graph for sugarcane. *20th International Database Engineering & Applications Symposium (IDEAS '16)*, 364–369. <https://doi.org/10.1145/2938503.2938521>
- Dury, J. (2011). *The cropping-plan decision-making: A farm level modelling and simulation approach* (Doctoral dissertation). Agrosystèmes, Ecosystèmes et Environnement, Institut National Polytechnique de Toulouse. Toulouse, France.
- Dury, J., Garcia, F., Reynaud, A., Therond, O. & Bergez, J.-E. (2010). Modelling the complexity of the cropping plan decision-making. In D. A. Swayne, W. Yang, A. A. Voinov, A. Rizzoli & T. Filatova (Eds.), *5th International Congress on Environmental Modelling and Software (iEMSs2010)*.
- Fagundes, M. S. (2007). *Integrating BDI model and Bayesian networks* (Master's thesis). School of Computer Science, Universidade Federal do Rio Grande do Sul. Porto Alegre, Brazil.
- Fagundes, M. S., Vicari, R. M. & Coelho, H. (2009). Deliberation process in a BDI model with Bayesian networks. In A. Ghose, G. Governatori & R. Sadananda (Eds.), *Agent Computing and Multi-Agent Systems (PRIMA 2007)* (pp. 207–218). Berlin Heidelberg: Springer-Verlag. [https://doi.org/10.1007/978-3-642-01639-4\\_18](https://doi.org/10.1007/978-3-642-01639-4_18)
- Gluz, J. C. & Jaques, P. A. (2014). A probabilistic implementation of emotional BDI agents. *6th International Conference on Agents and Artificial Intelligence (ICAART-2014)*, 121–129. <https://doi.org/10.5220/0004815501210129>
- Hänninen, M. (2014). Bayesian networks for maritime traffic accident prevention: Benefits and challenges. *Accident Analysis & Prevention*, 73, 305–312. <https://doi.org/10.1016/j.aap.2014.09.017>
- HUGIN Expert A/S [Accessed: 11-10-2019]. (2019). <https://www.hugin.com>

- Kadwa, M. & Bezuidenhout, C. N. (2015). Modelling sugarcane supply consistency at a sugar mill. *Computers and Electronics in Agriculture*, 111, 107–111. <https://doi.org/10.1016/j.compag.2014.12.019>
- Kieling, G. L. & Vicari, R. M. (2011). Insertion of probabilistic knowledge into BDI agents construction modeled in Bayesian networks. *International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS 2011)*, 115–122. <https://doi.org/10.1109/CISIS.2011.26>
- Kocabas, V. & Dragicevic, S. (2013). Bayesian networks and agent-based modeling approach for urban land-use and population density change: A BNAS model. *Journal of Geographical Systems*, 15(4), 403–426. <https://doi.org/10.1007/s10109-012-0171-2>
- Korb, K. B. & Nicholson, A. E. (2011). *Bayesian artificial intelligence* (2nd). Boca Raton FL: CRC Press. <https://doi.org/10.1201/b10391>
- Kyrimi, E., Neves, M. R., McLachlan, S., Neil, M., Marsh, W. & Fenton, N. (2020). Medical idioms for clinical Bayesian network development. *Journal of Biomedical Informatics*, 108, 103495. <https://doi.org/10.1016/j.jbi.2020.103495>
- Liang, X., Chen, H., Wang, Y. & Song, S. (2016). Design and application of a CA-BDI model to determine farmers' land-use behavior. *SpringerPlus*, 5(1), 1–18. <https://doi.org/10.1186/s40064-016-3245-7>
- Luz, B., Meneguzzi, F. & Vicari, R. (2013). Alternatives to threshold-based desire selection in Bayesian BDI agents. In M. Cossentino, A. E. F. Seghrouchni & M. Winikoff (Eds.), *Engineering Multi-Agent Systems (EMAS 2013)* (pp. 176–195). Berlin Heidelberg: Springer. [https://doi.org/10.1007/978-3-642-45343-4\\_10](https://doi.org/10.1007/978-3-642-45343-4_10)
- Marcot, B. G. (2017). Common quandaries and their practical solutions in Bayesian network modeling. *Ecological Modelling*, 358, 1–9. <https://doi.org/10.1016/j.ecolmodel.2017.05.011>
- Martin-Clouaire, R. (2017). Modelling operational decision-making in agriculture. *Agricultural Sciences*, 8(7), 527–544. <https://doi.org/10.4236/as.2017.87040>
- Martin-Clouaire, R. & Rellier, J.-P. (2009). Modelling and simulating work practices in agriculture. *International Journal of Metadata, Semantics and Ontologies*, 4(1–2), 42–53. <https://doi.org/10.1504/IJMSO.2009.026253>
- McCown, R. L. (2012). A cognitive systems framework to inform delivery of analytic support for farmers' intuitive management under seasonal climatic variability. *Agricultural Systems*, 105(1), 7–20. <https://doi.org/10.1016/j.agsy.2011.08.005>
- Muto, T. J., Bolivar, E. B. & González, E. (2020). BDI multi-agent based simulation model for social ecological systems. In F. De La Prieta, P. Mathieu, J. A. R. Arango, A. El Bolock, E. del Val, J. J. Prunera, J. Carneiro, R. Fuentes, F. Lopes & V. Julian (Eds.), *Highlights in Practical Applications of Agents, Multi-Agent Systems, and Trust-worthiness. The PAAMS Collection (PAAMS 2020)* (pp. 279–288). Cham, Switzerland: Springer. [https://doi.org/10.1007/978-3-030-51999-5\\_23](https://doi.org/10.1007/978-3-030-51999-5_23)

- Pitchforth, J. & Mengersen, K. (2013). A proposed validation framework for expert elicited Bayesian networks. *Expert Systems with Applications*, 40(1), 162–167. <https://doi.org/10.1016/j.eswa.2012.07.026>
- Preuveneers, D., Naqvi, N. Z., Ramakrishnan, A., Berbers, Y. & Joosen, W. (2016). Adaptive dissemination for mobile electronic health record applications with proactive situational awareness. *2016 49th Hawaii International Conference on System Sciences (HICSS)*, 3229–3238. <https://doi.org/10.1109/HICSS.2016.405>
- Price, C. S., Moodley, D. & Pillay, A. W. (2018). Dynamic Bayesian decision network to represent growers' adaptive pre-harvest burning decisions in a sugarcane supply chain. In J. van Niekerk & B. Haskins (Eds.), *2018 Annual Conference of the South African Institute of Computer Scientists and Information Technologists (SAICSIT '18)* (pp. 89–98). <https://doi.org/10.1145/3278681.3278693>
- Price, C. S., Moodley, D. & Pillay, A. W. (2019). Modelling uncertain adaptive decisions: Application to KwaZulu-Natal sugarcane growers. In M. Davel & E. Barnard (Eds.), *South African Forum for Artificial Intelligence Research (FAIR 2019)* (pp. 145–160).
- Rao, A. S. & Georgeff, M. P. (1995). BDI agents: From theory to practice. *First International Conference on Multi-Agent Systems (ICMAS 95)*, 312–319.
- Rens, G. & Moodley, D. (2017). A hybrid POMDP-BDI agent architecture with online stochastic planning and plan caching. *Cognitive Systems Research*, 43, 1–20. <https://doi.org/10.1016/j.cogsys.2016.12.002>
- Robert, M., Dury, J., Thomas, A., Therond, O., Sekhar, M., Badiger, S., Ruiz, L. & Bergez, J.-E. (2016). CMFDM: A methodology to guide the design of a conceptual model of farmers' decision-making processes. *Agricultural Systems*, 148, 86–94. <https://doi.org/10.1016/j.agsy.2016.07.010>
- Robert, M., Thomas, A. & Bergez, J.-E. (2016). Processes of adaptation in farm decision-making models. A review. *Agronomy for Sustainable Development*, 36(4), 1–15. <https://doi.org/10.1007/s13593-016-0402-x>
- Robert, M., Thomas, A., Sekhar, M., Badiger, S., Ruiz, L., Raynal, H. & Bergez, J.-E. (2017). Adaptive and dynamic decision-making processes: A conceptual model of production systems on Indian farms. *Agricultural Systems*, 157, 279–291. <https://doi.org/10.1016/j.agsy.2016.08.001>
- Russell, S. & Norvig, P. (2009). *Artificial intelligence: A modern approach* (3rd). Englewood Cliffs NJ: Prentice-Hall.
- Sanjika, T. M. & Bezuidenhout, C. N. (2015). Driving factors-based approach for identifying performance indicators in sugarcane supply and processing systems. *British Food Journal*, 117(6), 1652–1669. <https://doi.org/10.1108/BFJ-09-2014-0323>
- SASA. (2018). SASA weather portal [Accessed: 29 May 2018]. [http://portal.sasa.org.za/weatherweb/weatherweb.wv\\_menus.menu\\_frame?menuid=1](http://portal.sasa.org.za/weatherweb/weatherweb.wv_menus.menu_frame?menuid=1)
- SASRI. (2017). Information Sheet: 15. CANE QUALITY - 15.3 Losses in cane quality due to harvest to crush delays [Accessed: 26 October 2021]. [https://sasri.org.za/wp-content/uploads/Information\\_Sheets/IS\\_15.3-Harvest-to-crush-delays.pdf](https://sasri.org.za/wp-content/uploads/Information_Sheets/IS_15.3-Harvest-to-crush-delays.pdf)



- Shafer, G. (1976). *A mathematical theory of evidence*. Princeton NJ: Princeton University Press. <https://doi.org/10.1515/9780691214696>
- Silva, D. G. & Gluz, J. C. (2011). AgentSpeak (PL): A new programming language for BDI agents with integrated Bayesian network model. *2011 International Conference on Information Science and Applications*, 1–7. <https://doi.org/10.1109/ICISA.2011.5930301>
- Simon, H. A. (1996). *The sciences of the artificial* (3rd Edition). Cambridge MA: MIT Press.
- Smithers, J. (2014). Review of sugarcane trash recovery systems for energy cogeneration in South Africa. *Renewable and Sustainable Energy Reviews*, 32, 915–925. <https://doi.org/10.1016/j.rser.2014.01.042>
- Taillandier, P., Therond, O. & Gaudou, B. (2012). A new BDI agent architecture based on the belief theory. Application to the modelling of cropping plan decision-making. In R. Seppelt, A. A. Voinov, S. Lange & D. Bankamp (Eds.), *International Environmental Modelling and Software Society (iEMSs)*.
- Truong, Q. C., Taillandier, P., Gaudou, B., Vo, M. Q., Nguyen, T. H. & Drogoul, A. (2015). Exploring agent architectures for farmer behavior in land-use change. A case study in coastal area of the Vietnamese Mekong Delta. In B. Gaudou & J. S. Sichman (Eds.), *Multi-Agent Based Simulation XVI (MABS 2015)* (pp. 146–158). Cham, Switzerland: Springer. [https://doi.org/10.1007/978-3-319-31447-1\\_10](https://doi.org/10.1007/978-3-319-31447-1_10)
- Uusitalo, L. (2007). Advantages and challenges of Bayesian networks in environmental modelling. *Ecological Modelling*, 203(3-4), 312–318. <https://doi.org/10.1016/j.ecolmodel.2006.11.033>
- Wanyana, T. & Moodley, D. (2021). An agent architecture for knowledge discovery and evolution. In S. Edelkamp, R. Möller & E. Rueckert (Eds.), *KI 2021: Advances in Artificial Intelligence* (pp. 241–256). Cham, Switzerland, Springer. [https://doi.org/10.1007/978-3-030-87626-5\\_18](https://doi.org/10.1007/978-3-030-87626-5_18)
- Woodberry, O., Millett-Riley, J., Sinclair, S. & Nicholson, A. (2014). An object-oriented dynamic Bayesian decision network model for grasslands adaptive management. *Eleventh UAI Conference on Bayesian Modeling Applications Workshop (BMAW'14)*, 1218, 101–101.
- Wooldridge, M. (2000). *Reasoning about rational agents*. Massachusetts: MIT Press.



## A SUGARCANE GROWER'S BDI CHARACTERISTICS

The BDI characteristics of the sugarcane grower are given in Table 1.

Table 1: BDI characteristics of the sugarcane grower

BDI characteristic	Sugarcane grower
Belief	<p>Beliefs are what the sugarcane believes about itself and its environment, and how the environment changes, given its current state. For example:</p> <ul style="list-style-type: none"> <li>• the grower's beliefs about that day's weather (max temperature, amount of wind, type of rain that may fall)</li> <li>• his beliefs that adverse rain and wind might occur in the next four days</li> <li>• his beliefs about how dry the cane is</li> <li>• his beliefs about what causes the cane to become more wet or dry</li> <li>• his beliefs about how much cane is already burned and is awaiting delivery, how much cane was delivered on the previous day, and how far ahead/on target/behind he is with deliveries</li> <li>• his beliefs about how well the mill is running</li> <li>• his beliefs that a fire danger is present on a neighbour's farm</li> <li>• his beliefs about which day of the week it is</li> <li>• his beliefs about the productivity of a particular field</li> </ul>
Goal = Desire	<p>The grower's goals are</p> <ul style="list-style-type: none"> <li>• to keep up-to-date with deliveries during the milling season</li> <li>• to harvest and extract all burned cane</li> <li>• to burn the correct amount of cane daily before harvesting</li> <li>• to avoid penalties (e.g. late deliveries, delivered cane that is too old)</li> <li>• to ensure that sugarcane fields remain sufficiently productive</li> <li>• never to have runaway fires</li> <li>• to work a maximum of 6 days per week</li> <li>• to ensure that the sugarcane farm remains a profitable business</li> </ul>

*continued on next page –*

*Table 1 – continued from previous page*

<b>BDI characteristic</b>	<b>Sugarcane grower</b>
Intention	<p>The sugarcane grower's intentions are things like</p> <ul style="list-style-type: none"> <li>• delivering sufficient fresh cane for each day during the milling season</li> <li>• harvesting and extracting all available burned cane</li> <li>• burning a particular amount of cane each day before harvesting</li> <li>• taking a day off work once a week</li> <li>• replanting a field</li> <li>• maintaining trucks in delivery fleet, and fixing a broken-down sugarcane delivery truck(s) where necessary</li> <li>• helping neighbouring growers to contain runaway fires, should they occur</li> </ul>
Plan	<p>Sequence of actions the grower takes to achieve each intention. For example, for the intention “delivering sufficient fresh cane for each day during the milling season”, the plan includes actions such as</p> <ul style="list-style-type: none"> <li>• determine which field(s) to burn</li> <li>• burn the cane</li> <li>• harvest (and extract) the burned cane</li> <li>• deliver as much cane that was awaiting delivery (up to and including the previous day) to the mill (this is subject to transport limitations and mill limitations)</li> <li>• update daily tally of deliveries and delivery status (at end of the day)</li> <li>• update daily tally of cane awaiting delivery (at end of the day)</li> </ul>
Action	One of the bulleted steps in the plan above

## B DESCRIPTION OF THE DDN'S NODES

The description of the DDN's nodes and their states (see Figure 3) is given in Table 2.

Table 2: Description of the DDN's nodes and their states

Nodes	Description	States
<i>"Cane dryness" nodes</i>		
Cane dryness	Dryness of the cane	Dry, damp, wet
Observed rain type	Type of rain observed on previous day	No rain, thundershower, drizzle
Observed wind	Average wind run (km per day) on previous day	0–110 (Low), 110.1–180 (Medium), 180.1–560 (High)
Observed max temp	Maximum temperature on previous day (in °C)	5–22 (Low), 22.1–27 (Medium), 27.1–45 (High)
<i>"Current day's weather" nodes</i>		
Rain type forecast	Rain type forecast on current day	No rain, Thundershower, Drizzle
Wind forecast	Wind forecast on current day (in km per day)	0–110 (Low), 110.1–180 (Medium), 180.1–560 (High)
Max temp forecast	Maximum temperature forecast (in °C)	5–22 (Low), 22.1–27 (Medium), 27.1–45 (High)
<i>"Advance burning requirements" nodes</i>		
Rain forecast for next 4 days	Rain forecast for the next four days, starting on the following day	No rain forecast, Rain lasting 1 day starts tomorrow, ... Rain lasting 4 days starts tomorrow
Wind forecast for next 4 days	High wind forecast for the next four days, starting on the following day	No high wind forecast, High wind lasting 1 day starts tomorrow, ... High wind lasting 4 days starts tomorrow
Advance burning requirements	No. of days' quota to be burned in advance of inclement weather	Burn no additional cane, Burn 1 day's additional cane, ... Burn 4 days' additional cane

*continued on next page –*

Table 2 – continued from previous page

Nodes	Description	States
<i>Other nodes</i>		
Burning conditions	Whether it is possible to burn or not on the current day	Can burn, Can't burn
Grower size	Whether the grower is a large- or small-scale grower	Large, Small
Current day's burning requirements	No. of days' quota to be burned to satisfy current day's requirements	Need to burn no cane, Need to burn 0.25 days' cane, ... Need to burn 4 days' cane
Intention selection: Amount of cane to burn	Decision node: no. of days' quota to be burned	Don't burn cane, Burn 1 day's cane, Burn 1.25 days' cane, ... Burn 4 days' cane

## C CALCULATION OF CURRENT DAY'S BURNING REQUIREMENTS, DELIVERY STATUS, AND MAXIMUM CANE WHICH CAN BE DELIVERED

This section contains the calculation of three variables: the current day's burning requirements; the delivery status; and the maximum amount of cane which can be delivered.

The calculation for the current day's burning requirements (in units of one day's cane quota) is presented in Algorithm 3.

Algorithm 3: Calculation of Current day's burning requirements on day  $t$

---

CalcBR(CaneAwaitingDelivery $_t$ , StatusOfDeliveries $_t$ , MaximumDailyDeliveryRate, GrowerSize)

**Output:** BurningRequirements $_t$

```

/* CaneAwaitingDelivery $_t$ , StatusOfDeliveries $_t$  given in units of daily cane quota */
/* If deliveries are ahead, StatusOfDeliveries $_t$  > 0; if deliveries are on target,
   StatusOfDeliveries $_t$  = 0; if deliveries are behind, StatusOfDeliveries $_t$  < 0 */
/* MaximumDailyDeliveryRate given as % of 1 day's cane quota, and is constant for
   each grower; GrowerSize (large/small) is constant for each grower */
/* Large growers burn daily; small growers burn every other day */
if (MaximumDailyDeliveryRate = 100%) then
  if (GrowerSize = Large) then
    if CaneAwaitingDelivery $_t$  < 1
      /* less than 1 days' cane is awaiting delivery */
      BurningRequirements $_t$  = (1 - CaneAwaitingDelivery $_t$ )
    else BurningRequirements $_t$  = 0
  else /* GrowerSize = Small */
    if CaneAwaitingDelivery $_t$  < 2 then
      /* less than 2 days' cane is awaiting delivery */
      BurningRequirements $_t$  = (2 - CaneAwaitingDelivery $_t$ )
    else BurningRequirements $_t$  = 0
else /* MaximumDailyDeliveryRate > 100% */
  if (GrowerSize = Large) then
    if (StatusOfDeliveries $_t$  >= 0) and (CaneAwaitingDelivery $_t$  >= 2)
      then
        /* On target or ahead of deliveries & have 2+ days'
           cane quota awaiting delivery */
        BurningRequirements $_t$  = 0
        /* 2 days' cane quota: enough for today & tomorrow */
      else
        if (StatusOfDeliveries $_t$  < 0) and
           (CaneAwaitingDelivery $_t$  >= 2*MaximumDailyDeliveryRate)
          then /* Behind with deliveries & have 2+ days' cane deliveries
              awaiting delivery */
            BurningRequirements $_t$  = 0
            /* 2 days' cane deliveries: enough for today & tomorrow */
          else
            BurningRequirements $_t$  = 2 - (CaneAwaitingDelivery $_t$  +
              StatusOfDeliveries $_t$ )
            if BurningRequirements $_t$  > MaximumDailyDeliveryRate then

```

---

```

        BurningRequirementst = MaximumDailyDeliveryRate
    if BurningRequirementst < 0 then
        BurningRequirementst = 0
    end else
else /* GrowerSize = Small */
    if ((StatusOfDeliveriest >= 2) or ((StatusOfDeliveriest >= 0) and
        (CaneAwaitingDeliveryt >= 2))
    then /* On target or ahead of deliveries & have 2+ days'
        cane quota awaiting delivery */
        BurningRequirementst = 0
        /* ahead or have 2 days' cane quota: enough for today &
        tomorrow */
    else
    if (StatusOfDeliveriest < 0) and
        (CaneAwaitingDeliveryt >= 2*MaximumDailyDeliveryRate)
    then
        /* Behind with deliveries & have 2+ days' cane deliveries
        awaiting delivery */
        BurningRequirementst = 0
        /* 2 days' cane deliveries: enough for today & tomorrow;
        burn tomorrow */
    else
        BurningRequirementst = 3 - (CaneAwaitingDeliveryt +
            StatusOfDeliveriest)
        if BurningRequirementst > 2*MaximumDailyDeliveryRate then
            BurningRequirementst = 2*MaximumDailyDeliveryRate
        if BurningRequirementst < 0 then
            BurningRequirementst = 0
return BurningRequirementst

```

---

The calculation for the cane delivery status on day  $t$  is presented in Algorithm 4.

Algorithm 4: Calculation of delivery status on day  $t$

---

```

CalcDS(DeliveryStatust-1, CaneDeliveredt-1, DayofWeek) returns DeliveryStatust
/* CaneDeliveredt is given in units of daily cane quota */
/* DeliveryStatust-1 is given in no. of days' quota deliveries are ahead
(+ve number), behind (-ve number) or on target (0). */
/* Growers rest on Sundays */
if DayOfWeek = Monday then
    DeliveryStatust = DeliveryStatust-1 /* Deliveries not expected on Sundays */
else
    DeliveryStatust = DeliveryStatust-1 + (CaneDeliveredt-1 - 1)
return DeliveryStatust

```

---



The calculation for the maximum amount of cane the grower can deliver on day  $t$  (in units of one day's cane quota) is presented in Algorithm 5. If the mill calls for less than one day's cane quota, the grower may not deliver more than that call. However, if the mill calls for 100% or more of a day's quota, the grower may deliver up to his maximum daily delivery capacity, if he has cane that is awaiting delivery.

Algorithm 5: Maximum amount of cane the grower can deliver on day  $t$

---

```

CalcCGCD(CaneAwaitingDeliveryt, MaximumDailyDeliveryRate, MillsCallForCanet)
    returns MaxCaneGrowerCanDelivert
/* MaximumDailyDeliveryRate and MillsCallForCanet given as %
of 1 day's cane quota. */
/* CaneAwaitingDeliveryt and MaxCaneGrowerCanDelivert given in
units of 1 day's cane quota */
if (MillsCallForCanet < 1) then
    if (CaneAwaitingDeliveryt < MillsCallForCanet) then
        MaxCaneGrowerCanDelivert = CaneAwaitingDeliveryt
    else MaxCaneGrowerCanDelivert = MillsCallForCanet
else /* MillsCallForCanet >= 1 */
    if (CaneAwaitingDeliveryt > MaximumDailyDeliveryRate) then
        MaxCaneGrowerCanDelivert = MaximumDailyDeliveryRate
    else MaxCaneGrowerCanDelivert = CaneAwaitingDeliveryt
return MaxCaneGrowerCanDelivert

```

---

## D DDN ASSUMPTIONS

The DDN's assumptions are given in Table 3.

Table 3: DDN assumptions

---

DDN assumptions
<ul style="list-style-type: none"> <li>• The grower's size (large/small), and maximum daily delivery rate are constant over the duration of the DDN runs.</li> <li>• Cane which is burned today can be delivered tomorrow.</li> <li>• Large growers burn daily and deliver daily; small growers burn on alternate days and deliver daily.</li> <li>• No more than 4 day's cane quota is ever suggested to be burned at one time.</li> <li>• Cane can be burned in multiples of 0.25 of a day's quota: 0; 1; 1.25; 1.5; 1.75; 2; 2.25; ... , 4. If <math>0 &lt; \text{Current day's burning requirements} &lt; 1</math>, then 0 days' cane quota will be suggested to be burned.</li> <li>• Growers can deliver up to two days' cane quota on any one day.</li> <li>• Growers deliver cane depending on how much cane has been burned (and harvested); the grower's maximum daily delivery rate; and the mill's call for cane.</li> <li>• If the mill calls for 100% (normal) daily deliveries, or more, the grower may deliver more than his one day's quota, if cane is available (e.g. if the grower is behind with deliveries); however, if the mill calls for less than one day's cane, the grower will not deliver more than the amount called for by the mill.</li> </ul> <p>Rain and wind affect the decision to burn in the following ways:</p> <ul style="list-style-type: none"> <li>• Wet cane cannot be burned.</li> <li>• Cane cannot be burned in high winds.</li> <li>• Cane cannot be burned if the grower believes that drizzle or thundershowers are forecast for that day.</li> <li>• However, if cane is dry, and the grower believes that no rain will be forecast, and that high winds are not forecast, cane can be burned.</li> </ul> <p>Cane dryness is affected by the weather:</p> <ul style="list-style-type: none"> <li>• Cane dries quickly after a thundershower, but soft soaking drizzle leaves the cane damp or wet.</li> <li>• Wet or damp cane dries depending on the maximum temperature and the strength and duration of wind.</li> </ul> <p>If wet or windy weather is forecast for the following four days, starting on the following day, the grower may burn more cane in advance so that there is cane to deliver when burning cannot take place. However, the grower will never burn more than four days' cane.</p>

---