Context-based and Explainable Decision Making with Argumentation

Zhiwei Zeng[*]
LILY Research Centre, IGS
Nanyang Technological University
Alibaba-NTU Singapore JRI
i160001@e.ntu.edu.sg

Xiuyi Fan[†]
LILY Research Centre, NTU
Department of Computer Science
Swansea University
xiuyi.fan@swansea.ac.uk

Chunyan Miao[‡]
LILY Research Centre, SCSE
Nanyang Technological University

Cyril Leung
LILY Research Centre
Nanyang Technological University

Chin Jing Jih[§]
LKC School of Medicine, NTU
Centre for Geriatric Medicine, TTSH

Ong Yew Soon
SCSE
Nanyang Technological University

ABSTRACT

Argumentation-based approaches to decision making have gained considerable research interest, due to their ability to select and justify decisions. In order to make better decisions, context is a key piece of information that needs to be considered. However, most existing argumentation-based models and frameworks have not modelled or reasoned with context explicitly. In this paper, we present a new argumentation-based approach for making context-based and explainable decisions. We propose a graphical representation for modelling decision problems involving varying contexts, Decision Graph with Context (DGC), and a reasoning mechanism for making context-based decisions which relies on the Assumption-based Argumentation formalism. Based on these constructs, we introduce two types of explanations, *argument explanation* and *context explanation*, identifying the reasons for the decisions made from an argument-view and a context-view respectively.

KEYWORDS

Decision making; context-awareness; argumentation; explanation

ACM Reference Format:
Zhiwei Zeng, Xiuyi Fan, Chunyan Miao, Cyril Leung, Chin Jing Jih, and Ong Yew Soon. 2018. Context-based and Explainable Decision Making with Argumentation. In *Proc. of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018), Stockholm, Sweden, July 10–15, 2018,* IFAAMAS, 9 pages.

1 INTRODUCTION

Amongst various approaches to decision making, argumentation-based approaches have gained increasing amount of research interest recently [1, 14, 21, 32]. Argumentation can play two different roles in decision making, namely help to select, or to explain and justify decisions. Argumentation-based approaches to decision making are expected to be more akin with the way humans deliberate, evaluate alternatives and make decisions [2]. This endows argumentation-based approaches with unique benefits, including transparent decision making process and the ability to offer understandable reasons underlying the decisions made.

Context, the particular situation, environment or domain in which a decision is to be made, is a key piece of information that needs to be taken into account in order to make an optimal decision. Contexts add additional dynamics and complexity to decision making in a sense that a decision may be "good" in a particular context but less "good" in other contexts. Studies in [4] have shown that humans evaluate arguments differently depending on implicit domain knowledge. Hence, incorporating context in problem modelling and reasoning can help to establish better understanding of the decision parameters and make holistic evaluation of the decision alternatives. However, existing argumentation-based approaches to decision making have not modelled or reasoned with context explicitly during the decision making process, also making it hard to study and understand the effects of context on decisions.

In this paper, we present an argumentation-based approach for making context-based and explainable decisions. To introduce context into decision making, the first step is to model context in the formal representation of a decision problem. We propose *Decision Graphs with Context (DGC)* for this purpose. DGCs can capture the varying relationships between decisions and goals in different contexts, offering greater expressiveness and flexibility in modelling decision problems. To select "good" decisions, we map DGCs to Assumption-based Argumentation (ABA) frameworks and transform the process of making context-based decisions in DGCs to determining argument admissibility in ABA frameworks.

To make the decision making process more transparent to humans, we propose two types of explanations for the decisions made. It is useful to study the reasons for not selecting a decision for the purpose of improving the decision alternatives or adapting decisions in different contexts. We introduce *argument explanations* to explain the sources of failure of a decision alternative by identifying the attackers from which it cannot be defended. Sometimes an alternative is not "good" in a given context but "good" in another. Benefiting from the new graphical representation and the reasoning

[*]IGS stands for the Interdisciplinary Graduate School. Alibaba-NTU Singapore JRI stands for the Alibaba-NTU Singapore Joint Research Institute.
[†]NTU stands for Nanyang Technological University.
[‡]SCSE stands for the School of Computer Science and Engineering.
[§]TTSH stands for Tan Tock Seng Hospital, Singapore

Proc. of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018), M. Dastani, G. Sukthankar, E. André, S. Koenig (eds.), July 10–15, 2018, Stockholm, Sweden.

mechanism proposed, we also introduce *context explanations* which give context-specific reasons to explain whether the failure of a decision can be attributed to contexts.

The remaining part of this paper is organised as follows. We introduce relevant background in Section 2 and present Decision Graph with Context (DGC) in Section 3. We then illustrate how to compute "good" decisions with Assumption-Based Argumentation (ABA) in Section 4 and how to derive the two forms of explanations in Section 5. Finally, we discuss related works in Section 6 and conclude in Section 7.

2 BACKGROUND

Abstract Argumentation (AA) *frameworks* [8] are pairs $AF = \langle \mathcal{B}, \mathcal{K} \rangle$, consisting of a set of arguments, $\mathcal{B}$, and a binary *attack* relation, $\mathcal{K}$. Given an AA framework $AF = \langle \mathcal{B}, \mathcal{K} \rangle$, a set of arguments $B \subseteq \mathcal{B}$ is *admissible* in *AF* iff $\forall a, b \in B$, there exists no $(a, b) \in \mathcal{K}$ (B is conflict free) and $\forall a \in B$, if $(c, a) \in \mathcal{K}$, then there exists some $b \in B$ such that $(b, c) \in \mathcal{K}$.

We say that an argument a is *in AF* iff $a \in \mathcal{B}$, and an attack (a, b) is *in AF* or *a attacks b* in *AF* iff $(a, b) \in \mathcal{K}$.

Assumption-based Argumentation (ABA) *frameworks* [29] are tuples $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, C \rangle$ where

- $\langle \mathcal{L}, \mathcal{R} \rangle$ is a deductive system, with a *language* $\mathcal{L}$ and a rule set $\mathcal{R}$ of the form $\beta_0 \leftarrow \beta_1, \ldots, \beta_m (m \geq 0, \beta_i \in \mathcal{L})$;
- $\mathcal{A} \subseteq \mathcal{L}$ is a non-empty set, referred to as *assumptions*;
- C is a total mapping from $\mathcal{A}$ into $2^{\mathcal{L}}$, where each $c \in C(\alpha)$ is a *contrary* of α.

Given a rule ρ of the form $\beta_0 \leftarrow \beta_1, \ldots, \beta_m$, β_0 is referred to as the *head* and $\beta_1, \ldots, \beta_m$ as the *body* of ρ. All ABA frameworks are *flat*, i.e. assumptions do not occur in the head of rules.

In ABA frameworks, *arguments* are deductions of claims with sets of rules and supported by sets of assumptions. Attacks against arguments are directed at the assumptions in the support of arguments. Informally, adapted from [9, 29]:

- an *argument for* $\beta \in \mathcal{L}$ supported by $\Delta \subseteq \mathcal{A}$ with $\mathsf{R} \subseteq \mathcal{R}$ (denoted $\Delta \vdash_{\mathsf{R}} \beta$) is a finite tree with nodes labelled by sentences in $\mathcal{L}$ or by τ[1], the root labelled by β, leaves either τ or assumptions in Δ, and non-leaves β' with the elements of the body of some rule in R with head β' as children, and R contains no other rules except the ones in the tree.
- an argument $\Delta_1 \vdash_{\mathsf{R}_1} \beta_1$ *attacks* an argument $\Delta_2 \vdash_{\mathsf{R}_2} \beta_2$ iff β_1 is a contrary of one of the assumptions in Δ_2.

When there is no ambiguity, $\Delta \vdash \beta$ is used as the shorthand form for $\Delta \vdash_{\mathsf{R}} \beta$.

Admissibility and other semantics introduced in AA can also be applied to ABA [12]. Formally, given an ABA framework $ABF = \langle \mathcal{L}, \mathcal{R}, \mathcal{A}, C \rangle$, a set of assumptions is *admissible* in *ABF* iff it does not attack itself and it attacks all $\Delta \subseteq \mathcal{A}$ that attack it. We say that an argument $\Delta \vdash \beta$ is *admissible* in *ABF* iff there is an admissible set $\Delta' \subseteq \mathcal{A}$ for which $\Delta \subseteq \Delta'$. We also say that an argument $\Delta \vdash_{\mathsf{R}} \beta$ is in *ABF* iff $\mathsf{R} \subseteq \mathcal{R}$ and $\Delta \subseteq \mathcal{A}$.

Since ABA is an instance of AA, given an ABA framework, a *corresponding* AA framework can be constructed by following the procedures described in [10].

[1] $\tau \notin \mathcal{L}$ represents"true" and stands for the empty body of rules

Explanations for non-admissible arguments in AA are defined using the *pruning operator*, $\setminus$. Given an AA framework $AF = \langle \mathcal{B}, \mathcal{K} \rangle$ and a set of arguments $B \subseteq \mathcal{B}$, the *repaired* framework is $AF \setminus B = \langle \mathcal{B}', \mathcal{K}' \rangle$, where $\mathcal{B}' = \mathcal{B} \setminus B$ and $\mathcal{K}' = \{(a, b) | (a, b) \in \mathcal{K}$ and $a \in \mathcal{B}', b \in \mathcal{B}'\}$.

Given an AA framework $AF = \langle \mathcal{B}, \mathcal{K} \rangle$, let $a \in \mathcal{B}$ be some non-admissible argument in *AF*. Then, $B \subseteq \mathcal{B}$ is an *explanation* of a iff: (1) a is admissible in $AF \setminus B$, and (2) there exists no $B' \subset B$ such that a is admissible in $AF \setminus B'$. If no such B exists in *AF*, then $\{a\}$ is the *explanation* of a [15].

A decision problem can be represented by a *decision framework* which describes the relationships between decisions and attributes and between goals and attributes with two tables, as follows:

A **Decision Framework (DF)** [11] is a tuple $\langle \mathsf{D}, \mathsf{A}, \mathsf{G}, \mathsf{T}_{\mathsf{DA}}, \mathsf{T}_{\mathsf{GA}} \rangle$, consisting of:

- a finite set of decisions $\mathsf{D} = \{d_1, \cdots, d_n\}, (n > 0)$,
- a finite set of attributes $\mathsf{A} = \{a_1, \cdots, a_m\}, (m > 0)$,
- a finite set of goals $\mathsf{G} = \{g_1, \cdots, g_l\}, (l > 0)$, and
- two tables T_{DA}, of size $n \times m$, and T_{GA}, of size $l \times m$, such that
  - for every $\mathsf{T}_{\mathsf{DA}}[i, j]$ $(1 \leq i \leq n, 1 \leq j \leq m)$, $\mathsf{T}_{\mathsf{DA}}[i, j]$ is either 1, representing d_i has a_j, or 0, otherwise.
  - for every $\mathsf{T}_{\mathsf{GA}}[k, j]$ $(1 \leq k \leq l, 1 \leq j \leq m)$, $\mathsf{T}_{\mathsf{GA}}[k, j]$ is either 1, representing g_k is satisfied by a_j, or 0, otherwise.

Given a decision framework $DF = \langle \mathsf{D}, \mathsf{A}, \mathsf{G}, \mathsf{T}_{\mathsf{DA}}, \mathsf{T}_{\mathsf{GA}} \rangle$, a decision $d_i \in \mathsf{D}$ *meets* a goal $g_k \in \mathsf{G}$, with respect to *DF*, if and only if there exists an attribute $a_j \in \mathsf{A}$, such that $\mathsf{T}_{\mathsf{DA}}[i, j] = 1$ and $\mathsf{T}_{\mathsf{GA}}[k, j] = 1$.

We use $\Gamma(d) = S$, where $d \in \mathsf{D}$, $S \subseteq \mathsf{G}$ to denote the set of goals met by d.

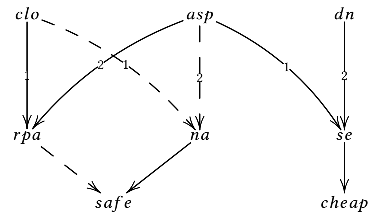
3 MODELLING CONTEXT IN DECISION MAKING

We introduce Decision Graphs with Context (DGC) as a new representation for modelling decision problems.

A DGC contains two parts of information: (1) a directed acyclic graph with nodes and edges that represents the relationship between decisions and goals; (2) contexts in which the decision is to be made. In a DGC, there are three types of nodes, namely *decisions*, *goals* and *intermediates*, corresponding to the candidate decisions, goals and decision attributes. Edges represent relations amongst the nodes, e.g. an edge from a decision to an intermediate attribute represents that the decision possesses the attribute; an edge from an intermediate attribute to a goal represents that the attribute satisfies the goal; an edge from one intermediate attribute to another intermediate attribute represents the former leads to the latter. In a DGC, the relationship between two nodes can be either *definite* or *defeasible*. A definite relationship holds in all contexts while a defeasible relationship generally holds but becomes inapplicable in certain contexts. Formally:

Definition 3.1. A *Decision Graph with Context (DGC)* is a tuple $\langle \mathsf{N}, \mathsf{E}, \mathsf{C} \rangle$, in which $\langle \mathsf{N}, \mathsf{E} \rangle$ is an acyclic graph, such that:

- $\mathsf{N} = \mathsf{N}_{\mathsf{d}} \cup \mathsf{N}_{\mathsf{int}} \cup \mathsf{N}_{\mathsf{g}}$ is a set of *nodes*, such that N_{d}, $\mathsf{N}_{\mathsf{int}}$ and N_{g} are pairwise disjoint, in which
  - $\mathsf{N}_{\mathsf{d}} \neq \emptyset$ is a set of *decision* nodes (decisions);
  - $\mathsf{N}_{\mathsf{int}}$ is a set of *intermediate* nodes (intermediates);
  - $\mathsf{N}_{\mathsf{g}} \neq \emptyset$ is a set of *goal* nodes (goals).



$$\mathsf{C_r} = \{ \quad clo\_allergy \rightarrow \neg dEdge(clo, na, 1)$$
$$asp\_allergy \rightarrow \neg dEdge(asp, na, 2)$$
$$gastritis \wedge \neg inhibitor \rightarrow \neg dEdge(rpa, safe, 1) \quad \}$$

Figure 1: A DGC Example. Solid arrows (⟶) represent strict edges $\mathsf{E_s}$ whereas dashed arrows (⇢) represent defeasible edges $\mathsf{E_d}$. Intuitive readings of the nodes in this graph are: *clo* stands for *administer_clopidogrel*, *asp* stands for *administer_aspirin*, *dn* means *do_nothing*, *rpa* means *reduced_platelet_adhesion*, *na* means *no_allergy*, and *se* represents *smaller_expense*.

- $\mathsf{E} = \mathsf{E_s} \cup \mathsf{E_d}$ is a set of *edges*, such that $\mathsf{E_s} \cap \mathsf{E_d} = \emptyset$, and
  - $\mathsf{E_s}$ is a set of *strict* edges;
  - $\mathsf{E_d}$ is a set of *defeasible* edges;
  - $[n_i|n_j]$ denotes an edge from n_i to n_j, for $n_i, n_j \in \mathsf{N}$;
  - $[n_i|n_j]$ is in E iff either $n_i \in \mathsf{N_d}$ and $n_j \in \mathsf{N_{int}} \cup \mathsf{N_g}$, or $n_i \in \mathsf{N_{int}}$ and $n_j \in \mathsf{N_{int}} \cup \mathsf{N_g}$;
  - each edge $e \in \mathsf{E}$ is associated with a *tag* i such that $i \in \mathbb{N}$, denoted by $t(e) = i$. If $t(e) = 1$ for an edge e, the tag is often omitted.
- $\mathsf{C} = \mathsf{C_p} \cup \mathsf{C_r}$ is a set of *defeasible context* such that $\mathsf{C_p} \cap \mathsf{C_r} = \emptyset$, in which
  - each sentence $c \in \mathsf{C}$ is an implication of the form $t_n \wedge \ldots \wedge t_1 \rightarrow t_0$ $(n \geq 0)$ over a language $\mathcal{L}_\mathsf{C}$ such that for each defeasible edge $e = [n|n'] \in \mathsf{E_d}$ with $t(e) = i$ there is a defeasible context sentence $\neg dEdge(n, n', i) \in \mathcal{L}_\mathsf{C}$;
  - $\mathsf{C_p}$ is a set of context primitives;
  - $\mathsf{C_r}$ is a set of rules;
  - for each $c \in \mathsf{C}$ in the form $t_n \wedge \ldots \wedge t_1 \rightarrow t_0$, $n = 0$ if $c \in \mathsf{C_p}$; and $n > 0$ if $c \in \mathsf{C_r}$.

Given a DGC, strict edges in $\mathsf{E_s}$ represent definite relationships between nodes while defeasible edges in $\mathsf{E_d}$ capture defeasible relationships. $\mathsf{C_p}$ consists of context primitives whereas $\mathsf{C_r}$ contains rules that specify how the context primitives, either by themselves or in conjunction, would influence the defeasibility of an edge. The notion $\neg dEdge(n, n', i)$ represents that the defeasible edge $[n|n']$ is *inapplicable* (refer to Definition 3.3), and hence cannot be traversed. Tags associated with the edges allow more complex relationships to be captured and are used to determine whether a node is *reachable* (refer to Definition 3.5) from a set of nodes in a DGC.

Example 3.2. As a running example to illustrate our approach, we discuss the decision making problem of choosing the appropriate treatment for a patient threatened by blood clotting. This example is adapted from an example by Modgil on the treatment of heart disease [23].

There are three decision alternatives available for treating blood clotting: *administer_clopidogrel* (*clo*), *administer_aspirin* (*asp*) and simply *do_nothing* (*dn*). The use of clopidogrel or aspirin can lead to *reduced_platelet_adhesion* (*rpa*), which reduces the risk of heart disease. The patient should have *no_allergy* (*na*) to the medicine administered in order to use it safely. In addition, if the patient has a history of gastritis, clopidogrel and aspirin should be used in conjunction with a proton pump inhibitor [2] to prevent gastrointestinal bleeding (which is considered unsafe) due to reduced platelet adhesion. In terms of cost considerations, aspirin is more affordable and incurs *smaller_expenses* (*se*) than clopidogrel. Doing nothing incurs no monetary costs.

Specifically for Example 3.2, we consider a patient who has a history of gastritis and an allergy to clopidogrel. Meanwhile, inhibitors are in stock.

We construct the DGC for deciding the appropriate treatment as follows:

- the decisions are: $\mathsf{N_d} = \{clo, asp, dn\}$;
- the goals are: $\mathsf{N_g} = \{safe, cheap\}$;
- the intermediate attributes are: $\mathsf{N_{int}} = \{rpa, na, se\}$;
- the strict edges are: $\mathsf{E_s} = \{\ [clo|rpa], [asp|rpa], [na|safe], [asp|se], [dn|se], [se|cheap]\ \}$;
- the defeasible edges are: $\mathsf{E_d} = \{\ [clo|na], [asp|na], [rpa|safe]\ \}$;
- the context primitives are: $\mathsf{C_p} = \{\rightarrow gastritis, \rightarrow clo\_allergy, \rightarrow inhibitor\}$;
- the context rules $\mathsf{C_r}$ are listed in Fig. 1.

The defeasible context $\mathsf{C} = \mathsf{C_p} \cup \mathsf{C_r}$ specifies the contexts in which the defeasible edges become inapplicable and hence untraversable. In this example, the context primitives $\mathsf{C_p}$ include medical history of the patient and the availability of the inhibitor. The context rules $\mathsf{C_r}$ can be used to determine the defeasibility of $\mathsf{E_d}$ given $\mathsf{C_p}$. For instance, an intuitive reading of the context rule

$$gastritis \wedge \neg inhibitor \rightarrow \neg dEdge(rpa, safe, 1)$$

associated with edge $[rpa|safe]$ is: if the patient has a history of gastritis and no inhibitor can be used together with the medicine, reducing platelet adhesion does not achieve the goal *safe*.

Formally, with the defeasible context specified by C, we define *inapplicable edges* as follows.

Definition 3.3. Given a DGC $CG = \langle \mathsf{N}, \mathsf{E}, \mathsf{C} \rangle$, with $\mathsf{E} = \mathsf{E_s} \cup \mathsf{E_d}$ and $\mathsf{C} = \mathsf{C_p} \cup \mathsf{C_r}$, the *inapplicable edges* of CG is a subset of $\mathsf{E_d}$, such that:

$$\mathsf{E_{ia}} = \{e \in \mathsf{E_d} | e = [n_i|n_j], t(e) = k, \mathsf{C} \vdash_{\mathsf{MP}} \neg dEdge(n_i, n_j, k)\}$$

where $\vdash_{\mathsf{MP}}$ stands for repeated applications of the modus ponens inference rule to the set of defeasible context C until the elements of $\mathsf{E_{ia}}$ do not change any more. [3]

We use $\Psi(CG) = \mathsf{E_{ia}}$ to denote the set of inapplicable edges in the context C.

Example 3.4. (Example 3.2 continued.) According to Definition 3.3, only defeasible edges can become inapplicable. With currently defined C, among the three defeasible edges $\mathsf{E_d} = \{\ [clo|na], [asp|na],$

[2] There are still controversies regarding the concomitant use of clopidogrel and proton pump inhibitors.

[3] The modus ponens inference rule amounts to deriving c from either $\rightarrow c$ or $a \rightarrow c$ and a, for any set (conjunction) of sentences a and sentences c.

$[rpa|safe]\}$ in the DGC in Fig. 1, only edge $[clo|na]$ is inapplicable, i.e. $\mathsf{E_{ia}} = \{[clo|na]\}$.

In order to provide means for determining whether a decision meets a goal, we first introduce the notion of *reachability* from a set of nodes to a node in DGC as follows.

Definition 3.5. Given a DGC $CG = \langle \mathsf{N}, \mathsf{E}, \mathsf{C} \rangle$, let $n \in \mathsf{N}$, $N \subseteq \mathsf{N}$. We say that n is *reachable* from N if and only if one of the following two conditions hold:

(1) there exists a tag k such that $N = \{n_i | \ e_i = [n_i|n] \in \mathsf{E} \setminus \Psi(CG)$ and $t(e_i) = k\}$; or
(2) there exists some $N' \subseteq \mathsf{N}$ such that n is reachable from N' and for each $n' \in N'$, n' is reachable from N.

The reachability to a node is defined recursively. Item 1 specifies the base condition that a node n is reachable from a set of nodes N. Item 2 specifies the "transitive" characteristic of reachability that if a node n is reachable from some intermediate set N' such that each node n' in N' is reachable from N, then n is reachable from N.

Tags associated with the edges allow more complex relationships to be captured and are used when determining whether a node is reachable. A node n is reachable from a set of nodes N if all nodes in N lead to n via applicable edges labelled by the same tag. When the tag is 1, it is often omitted.

Example 3.6. In Fig. 1, there are two paths to reach node $safe$ from node *clo*, one via node *rpa* and the other via node *na*. The two edges leading to node $safe$, $[rpa|safe]$ and $[na|safe]$, have the same tag 1 (both are 1, hence both are omitted). Since the two edges have the same tag, they have an "AND" relationship. The node $safe$ is only reachable via the two paths simultaneously. Hence, node $safe$ is reachable from $\{clo\}$ only when both *rpa* and *na* are reachable from $\{clo\}$ and $safe$ is reachable from $\{rpa, na\}$. However, if the two edges, $[rpa|safe]$ and $[na|safe]$, have different tags (e.g. 1 and 2), then they have an "OR" relationship. In this case, node $safe$ is reachable from $\{clo\}$ if $safe$ can be reached from either $\{rpa\}$ or $\{na\}$ which can be reach from $\{clo\}$, i.e. node $safe$ can be reached via either of the paths.

Referring to the notion of reachability, we can now define the set of goals satisfied by a decision as follows.

Definition 3.7. Given a DGC $CG = \langle \mathsf{N}, \mathsf{E}, \mathsf{C} \rangle$, $\mathsf{N} = \mathsf{N_d} \cup \mathsf{N_{int}} \cup \mathsf{N_g}$, in which $\mathsf{N_d}$ and $\mathsf{N_g}$ are the decisions and goals respectively, a decision $d \in \mathsf{N_d}$ *meets* (in the context of C) a goal $g \in \mathsf{N_g}$, iff g is reachable from $\{d\}$ in the context C.

We use $\Gamma(d) = S$, $S \subseteq \mathsf{N_g}$ to denote the set of goals met by decision d.

Example 3.8. (Example 3.4 continued.) As shown in Fig. 1, in the given context $\mathsf{C} = \{\rightarrow gastritis, \rightarrow clo\_allergy, \rightarrow inhibitor\} \cup \mathsf{C_r}$:

– *rpa* is reachable from $\{clo\}$ and $\{asp\}$;
– *na* is reachable from $\{asp\}$;
– *se* is reachable from $\{asp\}$ and $\{dn\}$;
– $safe$ is reachable from $\{rpa, na\}$ and hence $\{asp\}$;
– *cheap* is reachable from $\{se\}$ and hence $\{asp\}$, $\{dn\}$;
– *na*, and hence $safe$, is **not** reachable from $\{clo\}$ since edge $[clo|na]$ is inapplicable.

We can derive that $\Gamma(asp) = \{safe, cheap\}$ and $\Gamma(dn) = \{cheap\}$, i.e. decision *asp* meets the goal $safe$ and *cheap* while decision *dn* only meets the goal *cheap*; and the decision *clo* meets neither of the two goals.

According to the definitions given above, we can see that, given a DGC, whether a decision can meet a goal depends on the reachability of the goal from the decision. This reachability in turn depends on whether there exists a path leading from the decision to the goal and the applicability of the defeasible edges in the path. Even for two DGCs with exactly the same acyclic graph, i.e. the same N and E, a change in context C may render some formerly applicable defeasible edges inapplicable or some inapplicable ones applicable, and hence may change the goals that can be met by the decisions. We modify Example 3.2 to illustrate how different contexts would affect the goals met by the decisions.

Example 3.9. Suppose the inhibitor is currently out-of-stock, i.e. $\{\rightarrow \neg inhibitor\}$, while other contexts remain the same. In the new contexts, we now have $\mathsf{E_{ia}} = \{\ [clo|na], [rpa|safe]\ \}$. Thus, the goal $safe$ is no longer reachable from $\{rpa, na\}$ and hence from $\{asp\}$. We can derive that $\Gamma(asp) = \{cheap\}$ and $\Gamma(dn) = \{cheap\}$. Both decision *asp* and *dn* meet the the goal *cheap* while no goal meets the goal $safe$.

Comparing Example 3.8 and 3.9, we can see that just by changing the contexts, the goals met by the decisions also change.

With the ability to capture context information, DGC offers greater expressiveness and flexibility in modelling decision problems. As an example, we show that DGCs generalise Decision Frameworks as follows.

Proposition 3.10. *For any decision framework* $DF = \langle \mathsf{D}, \mathsf{A}, \mathsf{G}, \mathsf{T_{DA}}, \mathsf{T_{GA}} \rangle$, *there is a DGC* $CG = \langle \mathsf{N}, \mathsf{E}, \mathsf{C} \rangle$, $\mathsf{N} = \mathsf{N_d} \cup \mathsf{N_{int}} \cup \mathsf{N_g}$, *with* $\mathsf{N_d} = \mathsf{D}$ *the decisions,* $\mathsf{N_g} = \mathsf{G}$ *the goals, such that for each* $d \in \mathsf{D}$, $g \in \mathsf{G}$, *it holds that:* $g \in \Gamma(d)$ *in DF if and only if* $g \in \Gamma(d)$ *in CG.*

Proof. $g_k \in \Gamma(d_i)$ in *DF* if and only if there exists an attribute $a_j \in \mathsf{A}$, such that $\mathsf{T_{DA}}[i, j] = 1$ and $\mathsf{T_{GA}}[k, j] = 1$. $\mathsf{T_{DA}}[i, j] = 1$ if and only if there exists an applicable edge $[d_i|a_j]$ in *CG*, and $\mathsf{T_{GA}}[k, j] = 1$ if and only if there exists an applicable edge $[a_j|g_k]$ in *CG*. By Definition 3.5 and 3.7, g_k is reachable from d_i. Hence, d_i meets g_k in *CG*. □

This proposition holds as DGCs are capable of capturing any decision-attribute-goal relations that can be captured by DFs.

4 MAKING CONTEXT-BASED DECISIONS

After representing the decision problem as a DGC defined in the previous section, we illustrate how to map a DGC to an ABA framework and determine whether a decision is "good" with the ABA framework. An ABA framework can incorporate both the decision problem and the decision criteria simultaneously. The problem of identifying "good" decisions in a DGC can then be transformed into the problem of determining the admissibility of arguments in the corresponding ABA framework.

With the *decisions_meet_goals* information, different decision criteria can be adopted to evaluate the decisions. Dominance is used in [11]. In this paper, we focus on dominant decisions.

Following [11], a decision $d \in \mathsf{N_d}$ is *dominant* iff it meets all goals that are ever met by other decisions, i.e. let $S = \Gamma(d)$, there exists no d' such that $d' \neq d$, $g' \in \Gamma(d')$ and $g' \in \mathsf{N_g} \setminus S$.

Definition 4.1. Given a DGC $CG = \langle \mathsf{N}, \mathsf{E}, \mathsf{C} \rangle$, $\mathsf{N} = \mathsf{N_d} \cup \mathsf{N_{int}} \cup \mathsf{N_g}$ with $\mathsf{N_d}$ the decisions, $\mathsf{N_g}$ the goals, $\mathsf{E} = \mathsf{E_s} \cup \mathsf{E_d}$ with $\mathsf{E_s}$ the strict edges, $\mathsf{E_d}$ the defeasible edges, $\mathsf{C} = \mathsf{C_p} \cup \mathsf{C_r}$ with $\mathsf{C_p}$ the context primitives, and $\mathsf{C_r}$ the context rules, the *Dominant ABA Framework drawn from CG* is $ABF = \langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \mathcal{C} \rangle$, in which:

- $\mathcal{R}$ is such that:
  - for all $e = [n|n'] \in \mathsf{E_s}$: $edge(n, n', t(e)) \leftarrow \in \mathcal{R}$;
  - $edge(x, y, t) \leftarrow dEdge(x, y, t) \in \mathcal{R}$;
  - $reach(x, y) \leftarrow edge(x, y, t) \in \mathcal{R}$;
  - $reach(x, y) \leftarrow reach(x, w_1), edge(w_1, y, t), \neg unreachableSib(w_1, y, t, x) \in \mathcal{R}$;
  - $unreachableSib(w_1, y, t, x) \leftarrow edge(w_2, y, t), \neg reach(x, w_2), w_1 \neq w_2, \in \mathcal{R}$;
  - $met(d, g) \leftarrow reach(d, g) \in \mathcal{R}$;
  - $notDom(d) \leftarrow notMet(d, g), othersMet(d, g) \in \mathcal{R}$;
  - $noOthers(d_i, g) \leftarrow notMet(d_1, g), \cdots, notMet(d_{i-1}, g), notMet(d_{i+1}, g), \cdots, notMet(d_n, g) \in \mathcal{R}$;
  - for all $t_n \wedge \ldots \wedge t_1 \rightarrow t_0 \in \mathsf{C}$: $t_0 \leftarrow t_1, \ldots, t_n \in \mathcal{R}$;
  - nothing else is in $\mathcal{R}$.
- $\mathcal{A}$ is such that:
  - for all $n \in \mathsf{N}$, $e = [n'|n''] \in \mathsf{E}$, $n \neq n', n''$, $\neg unreachableSib(n', n'', t(e), n) \in \mathcal{A}$;
  - for all $n, n' \in \mathsf{N}$, $n \neq n'$, $\neg reach(n, n') \in \mathcal{A}$;
  - for all $d \in \mathsf{N_d}$, $dom(d) \in \mathcal{A}$;
  - for all $d \in \mathsf{N_d}$ and $g \in \mathsf{N_g}$, $notMet(d, g)$, $othersMet(d, g) \in \mathcal{A}$;
  - for all $e = [n|n'] \in \mathsf{E_d}$: $dEdge(n, n', t(e)) \in \mathcal{A}$;
  - nothing else is in $\mathcal{A}$.
- $\mathcal{C}$ is such that:
  - $\mathcal{C}(dEdge(x, y, t)) = \{\neg dEdge(x, y, t)\}$;
  - $\mathcal{C}(\neg unreachableSib(w_1, y, t, x)) = \{unreachableSib(w_1, y, t, x)\}$;
  - $\mathcal{C}(\neg reach(x, y)) = \{reach(x, y)\}$;
  - $\mathcal{C}(dom(d)) = \{notDom(d)\}$;
  - $\mathcal{C}(notMet(d, g)) = \{met(d, g)\}$;
  - $\mathcal{C}(othersMet(d, g)) = \{noOthers(d, g)\}$;
  - nothing else is in $\mathcal{C}$.

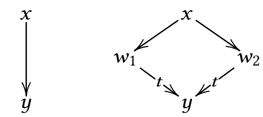
The intuition of Definition 4.1 is the following. We know a decision d meets a goal g if node g is reachable from the set $\{d\}$. We also know that a node y is reachable from another set of nodes $\{x\}$ under either of the two conditions, as illustrated in Fig. 2:

(1) if there exists an edge leading from node x to y: $reach(x, y) \leftarrow edge(x, y, k)$;
(2) there is an intermediate node w_1 such that:
  - w_1 is reachable from $\{x\}$, and there exists an edge from w_1 to y tagged with t, and
  - if w_1 has a "sibling" node $w_2 \neq w_1$ such that there is an edge from w_2 to y that is also tagged with t, then w_2 is also reachable from x, given by rules:

$$reach(x, y) \leftarrow reach(x, w_1), edge(w_1, y, t) ,\neg unreachableSib(w_1, y, t, x) \quad \text{and}$$
$$unreachableSib(w_1, y, t, x) \leftarrow edge(w_2, y, t), \neg reach(x, w_2).$$

A decision d is dominant if it meets all goals that are ever met by other decisions. Hence, the two premises of $notDom(d)$ are $notMet(d, g)$ and $othersMet(d, g)$, representing "the decision does not meet this goal" and "some other decisions can meet this goal" respectively. The contrary of "some other decisions can meet this goal" is "no other decisions meet this goal", represented by the rule:

$$noOthers(d_i, g) \leftarrow notMet(d_1, g), \cdots, notMet(d_{i-1}, g), notMet(d_{i+1}, g), \cdots, notMet(d_n, g).$$



(1) (2)

Figure 2: Two conditions under which node x is reachable from node y: (1) direct edge from x to y; (2) through intermediate node w_1 and its sibling w_2.

Each defeasible edge $e = [n|n'] \in \mathsf{E_{ia}}$ has an associated assumption $dEdge(n, n', t(e))$ with a contrary $\neg dEdge(x, y, t)$ for some $x = n, y = n'$ and $t = t(e)$. Decision contexts in C are represented as elements of $\mathcal{R}$ in the ABA framework. With such formalization, context information can be captured by the framework and influence the decision making by moderating the applicability of the defeasible edges. For example, if a defeasible edge $e = [n|n']$ tagged with $t(e) = i$ becomes inapplicable in a new context C', then we have $\{\} \vdash \neg dEdge(n, n', i)$ which can form an attack on the assumption $dEdge(n, n', i)$.

We already know that the problem of identifying "good" decisions in a decision graph is equivalent to the problem of determining the admissibility of the decisions in the corresponding ABA framework [21]. This also applies to a DGC, which is formally proved as follows:

Proposition 4.2. *Given a DGC $CG = \langle \mathsf{N}, \mathsf{E}, \mathsf{C} \rangle$, $\mathsf{N} = \mathsf{N_d} \cup \mathsf{N_{int}} \cup \mathsf{N_g}$ with $\mathsf{N_d}$ the decisions, $\mathsf{N_g}$ the goals, let ABF be the dominant ABA framework drawn from CG. Then for all decisions $d \in \mathsf{N_d}$, d is dominant in CG iff $\{dom(d)\} \vdash dom(d)$ is admissible in ABF.*

Proof. (*Sketch.*) First, we prove dominance implies admissibility for $d_i \in \mathsf{N_d}$. Since d_i is dominant, for each goal g_j, either (1) d_i meets goal g_j, therefore argument $\{\} \vdash met(d_i, g_j)$ exists and is not attacked; or (2) there is no argument $\{\} \vdash met(d_k, g_j)$ for all $d_k \in \mathsf{N_d}$, therefore argument $\{\} \vdash noOthers(d_i, g_j)$ exists and is not attacked. In both cases, the attackers of the argument $\{dom(d_i)\} \vdash dom(d_i)$, i.e. $\{notMet(d_i, g_j), othersMet(d_i, g_j)\} \vdash notDom(d_i)$, are always counter attacked. Thus, $\{dom(d_i)\} \vdash dom(d_i)$ withstands all attacks. Moreover, since $\{dom(d_i)\} \cup \{notMet(d_1, g_j), \cdots, notMet(d_{i-1}, g_j), notMet(d_{i+1}, g_j), \cdots notMet(d_n, g_j)\}$ is conflict-free, $\{dom(d_i)\} \vdash dom(d_i)$ is admissible.

We then prove admissibility implies dominance. Since the argument $\{dom(d_i)\} \vdash dom(d_i)$ is admissible, all its attackers, i.e. $\{notMet(d_i, g_j), othersMet(d_i, g_j)\} \vdash notDom(d_i)$ for all $g_j \in \mathsf{N_g}$, must be counter attacked. We know that each such attacker is counter attacked either because there exists an argument $\{\} \vdash met(d_i, g_j)$ or argument $\{\} \vdash noOthers(d_i, g_j)$, i.e. either d_i meets g_j or no other $d_k \in \mathsf{N_d}$ meets g_j. Thus, d_i is dominant. □

Proposition 4.2 shows that ABA frameworks can be used to identify dominant decisions.

Example 4.3. (Example 3.8 continued.) Given the DGC in Fig. 1, a dominant ABA framework $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, C \rangle$ can be constructed according to Definition 4.1, with the variables x, y, w_1, w_2 instantiated to the elements in N, variable t instantiated to the tags, variable d instantiated to the elements of $\mathsf{N_d}$ and variable g instantiated to the elements of $\mathsf{N_g}$ in the DGC. Due to space limitation, we list a few elements of each component instead of the fully instantiated framework here:

$\mathcal{R}$ consists of:

$edge(clo, rpa, 1) \leftarrow \quad edge(asp, na, 2) \leftarrow$
$edge(dn, se, 2) \leftarrow \quad edge(rpa, safe, 1) \leftarrow \ldots$
$edge(clo, na, 1) \leftarrow dEdge(clo, na, 1)$
$edge(asp, na, 2) \leftarrow dEdge(asp, na, 2)$
$edge(rpa, safe, 1) \leftarrow dEdge(rpa, safe, 1) \ldots$
$reach(clo, rpa) \leftarrow edge(clo, rpa, 1) \ldots$
$reach(clo, safe) \leftarrow reach(clo, rpa), edge(rpa, safe, 1)$
$, \neg unreachableSib(rpa, safe, 1, clo) \ldots$
$unreachableSib(rpa, safe, 1, clo)$
$\leftarrow edge(na, safe, 1), \neg reach(clo, na) \ldots$
$met(clo, safe) \leftarrow reach(clo, safe) \ldots$
$notDom(asp) \leftarrow notMet(asp, safe), othersMet(asp, safe) \ldots$
$noOthers(asp, safe) \leftarrow notMet(clo, safe), notMet(dn, safe) \ldots$
$\neg dEdge(rpa, safe, 1) \leftarrow gastritis, \neg inhibitor$
$clo\_allergy \leftarrow \quad \neg dEdge(clo, na, 1) \leftarrow clo\_allergy$
$gastritis \leftarrow \quad \neg dEdge(asp, na, 2) \leftarrow asp\_allergy$
$inhibitor \leftarrow$

$\mathcal{A}$ consists of:

$\neg unreachable(asp, rpa, 2, clo) \quad \neg unreachable(asp, na, 2, clo)$
$\neg unreachable(rpa, safe, 1, clo) \quad \ldots$
$\neg reach(clo, asp) \quad \neg reach(clo, rpa) \quad \neg reach(clo, safe)$
$\neg reach(asp, na) \quad \neg reach(asp, safe) \quad \neg reach(dn, se)$
$\neg reach(dn, safe) \quad \ldots$
$dom(clo) \quad dom(asp) \quad dom(dn)$
$notMet(clo, safe) \quad notMet(asp, cheap) \ldots$
$othersMet(clo, safe) \quad othersMet(dn, cheap) \ldots$
$dEdge(clo, na, 1) \quad dEdge(rpa, safe, 1) \ldots$

C is as given in Definition 4.1.

In this ABA framework, argument $\{dom(asp)\} \vdash dom(asp)$ is admissible. However, argument $\{dom(clo)\} \vdash dom(clo)$ and argument $\{dom(dn)\} \vdash dom(dn)$ are not admissible. By Proposition 4.2, decision asp is the only dominant decision.

Given a DGC $CG = \langle \mathsf{N}, \mathsf{E}, \mathsf{C} \rangle$, a change in context has no impact on the graph part $\langle \mathsf{N}, \mathsf{E} \rangle$, but affects the elements in the set C, which are represented as rules in $\mathcal{R}$ in the corresponding ABA framework. Hence, to obtain a new ABA framework which corresponds to the new context, we only need to update the elements in $\mathcal{R}$, by replacing the old rules affected with new ones derived from the new context. As we will see in the following example, as context changes, the applicability of defeasible edges may change which may affect the reachability of goal nodes from decision nodes. Thus, the dominance of decisions may be different when context varies.

Example 4.4. (Example 3.9 continued.) When the inhibitor is out-of-stock, the dominant ABA framework that can be drawn from the DGC in this example is almost the same $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, C \rangle$ as given in Example 4.3, but with the following modification:

- Remove the rule $inhibitor \leftarrow$ in $\mathcal{R}$;
- Add the rule $\neg inhibitor \leftarrow$ into $\mathcal{R}$.

In this context, both $\{dom(asp)\} \vdash dom(asp)$ and $\{dom(dn)\} \vdash dom(dn)$ are admissible. Hence, by Proposition 4.2, decisions asp and dn are dominant.

From Example 4.3 and 4.4, we can see that when the context changes, the dominance of the decisions may also change as a result. Thus, context can affect the decision. In other words, different decisions may be chosen in different context.

5 EXPLAINING NON-DOMINANT DECISIONS

The process of making decisions with argumentation paves the way for generating meaningful explanations for the decisions. For each non-dominant decision, it is useful to identify the reasons why it is not a "good" decision in the given context. In this section, we describe two types of explanations, *argument explanation (arg-explanation)* and *context explanation (cont-explanation)* for non-dominant decisions. Arg-explanations focus on identifying the cause of non-admissibility. By Proposition 4.2, we know that choosing dominant decisions in a DGC is equivalent to identifying admissible arguments in the corresponding ABA framework. In other words, if a decision is non-admissible, it will not be selected. Thus, identifying the cause of non-admissibility can help to explain why a decision is not selected. Cont-explanations provide more informative explanations when the non-admissibility can be traced to the contexts. It tells whether the failure of a decision can be attributed to the contexts and why it fails in this context.

Based on the ABA framework drawn from a DGC, as defined in Definition 4.1, we formalize the arg-explanation for non-admissible ABA arguments in Definition 5.1, which is adapted from [15]. Note that, the following definition is building on the explanations for AA arguments given in the background, and the definition of the pruning operator, \, also follows that given before in the Background.

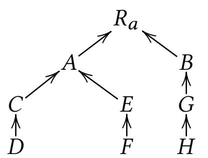
Definition 5.1. Given an ABA framework $ABF = \langle \mathcal{L}, \mathcal{R}, \mathcal{A}, C \rangle$ with the corresponding AA framework $AF = \langle \mathcal{B}, \mathcal{K} \rangle$, let $a \in \mathcal{B}$ be a non-admissible argument in AF. Then, $B \subseteq \mathcal{B}$ is an explanation of a if and only if the following two conditions hold:

(1) a is admissible in $AF \setminus B$, and
(2) there exists no $B' \subset B$ such that a is admissible in $AF \setminus B'$.

If no such B exists, $\{a\}$ is the arg-explanation of a.

The non-admissibility of an argument a can be attributed to a set of attackers B from which a cannot be defended. The intuitive idea of Definition 5.1 is: if we remove all arguments in set B from the framework and argument a becomes admissible in the pruned framework, then B contains all attackers from which a cannot be defended and they constitute an arg-explanation for why a is not admissible in the unpruned framework. Note that set B needs to be minimal. In the cases when no such set B exist, the non-admissibility of a can only be attribute to itself, i.e. when a attacks itself.

Although arg-explanations are computed using admissibility semantics, they can provide interpretable reasons to explain the decisions. We continue to use the dominant decision criterion as



$R_a = \{dom(clo)\} \vdash dom(clo)$ $\quad$ $A = \{notMet(clo, safe), othersMet(clo, safe)\} \vdash notDom(clo)$
$B = \{notMet(clo, cheap), othersMet(clo, cheap)\} \vdash notDom(clo)$
$D = \{\} \vdash_{\mathsf{R}} \neg dEdge(clo, na, 1)$ with $\mathsf{R} = \{\neg dEdge(clo, na, 1) \leftarrow clo\_allergy, clo\_allergy \leftarrow\}$
$E = \{notMet(asp, safe), notMet(dn, safe)\} \vdash noOthers(clo, safe)$
$G = \{notMet(asp, cheap), notMet(dn, cheap)\} \vdash noOthers(clo, cheap)$
$H = \{\neg unreachableSib(se, cheap, 1, dn)\} \vdash met(dn, cheap)$
$C = \{dEdge(clo, na, 1), dEdge(rpa, safe, 1), \neg unreachableSib(rpa, safe, 1, clo), \neg unreachableSib(na, safe, 1, clo)\} \vdash met(clo, safe)$
$F = \{dEdge(asp, na, 2), dEdge(rpa, safe, 1), \neg unreachableSib(rpa, safe, 1, asp), \neg unreachableSib(na, safe, 1, asp)\} \vdash met(asp, safe)$

Figure 3: AA framework for Example 5.3. Six arg-explanations of argument R_a: $\{A, B\}$, $\{A, H\}$, $\{B, D\}$, $\{B, F\}$, $\{D, H\}$ and $\{F, H\}$.

an illustration. For a non-dominant decision d, which fails to meet all goals that are ever met by other decisions, arguments in an arg-explanation of $\{dom(d)\} \vdash dom(d)$ identify goals that are not met by d but met by some other decisions. Formally:

Proposition 5.2. *Given a DGC $CG = \langle \mathsf{N}, \mathsf{E}, \mathsf{C} \rangle$, $\mathsf{N} = \mathsf{N_d} \cup \mathsf{N_g} \cup \mathsf{N_{int}}$, with $\mathsf{N_d}$ the decisions, $\mathsf{N_g}$ the goals, let $ABF = \langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \mathcal{C} \rangle$ be the dominant ABA framework drawn from CG. Then, for all $R_a = \{dom(d)\} \vdash dom(d)$ in ABF where $d \in \mathsf{N_d}$ is not dominant,*

1. *if for some $g \in \mathsf{N_g}$, $\{notMet(d, g), othersMet(d, g)\} \vdash notDom(d)$ is in an arg-explanation of R_a, then g is not met by d;*
2. *if for some $g \in \mathsf{N_g}$, argument $\{\} \vdash \neg dEdge(n_i, n_j, t)$ is in an arg-explanation of R_a and is attacking an argument $\Delta_1 \vdash met(d, g)$, then g is not met by d;*
3. *if for some $d' \in \mathsf{D}, d' \neq d$ and $g \in \mathsf{N_g}$, $\Delta_2 \vdash met(d', g)$ is in an arg-explanation R_a, then d does not meet g and d' meets g.*

Proof. By the construction of *ABF* (Definition 4.1), for a decision d and a goal g, argument $A = \{notMet(d, g), othersMet(d, g)\} \vdash notDom(d)$ exists in *ABF*. In order to defend the root argument $R_a = \{dom(d)\} \vdash dom(d)$, argument A needs to be counter-attacked by either argument $C = \Delta_1 \vdash met(d, g)$ or $E = \Delta_3 \vdash noOthers(d, g)$. For (1), it is easy to see that if A is removed, all its sub-level attackers and defenders are also removed, and thus R_a would become admissible. Thus, A is an arg-explanation of R_a and d does not meet g. To show (2), we observe that since $\neg dEdge(n_i, n_j, t)$ attacks argument C, d does not meet g. Similarly for (3), since the argument attacked by $F = \Delta_2 \vdash met(d', g)$ is $\Delta_3 \vdash noOthers(d, g)$ which in turn attacks argument A, F defends A and d does not meet g. Also, since argument F exists in *ABF*, d' meets g. □

In order for a decision d to be dominant, for each goal g, it must be either d meets g or all goals do not meet g. Thus, it is easy to see that d does not meet g and some other decision meets g would be in an arg-explanation. The inapplicability of a defeasible edge can also be in an arg-explanation as it may affect the reachability of g from d.

Example 5.3. (Example 4.3 continued.) Given the ABA framework *ABF* in Example 4.3, the corresponding AA framework constructed from *ABF* for the decision *administer_clopidogrel* (*clo*) is shown in Fig.3. There are six arg-explanations $\{A, B\}$, $\{A, H\}$, $\{B, D\}$, $\{B, F\}$, $\{D, H\}$ and $\{F, H\}$ for the root argument R_a. Removing any one of them from this AA framework makes the argument R_a admissible. By Proposition 5.2, we interpret this arg-explanation as follows:

> *The decision of administering clopidogrel is not dominant as it does not meet the goal safe or the goal cheap.*

An arg-explanation for a decision contains the reasons for the non-admissibility of the argument that embeds the decision. In other words, it contains the reasons why the decision is not dominant, which may or may not be due to contexts in which the decision is made. Hence, it is useful to know why the decision is non-admissible in a particular context. We use cont-explanation, which only contains the reasons that can be traced down to the decision contexts, for this purpose. Formally:

Definition 5.4. Given a DGC $CG = \langle \mathsf{N}, \mathsf{E}, \mathsf{C} \rangle$, $\mathsf{N} = \mathsf{N_d} \cup \mathsf{N_g} \cup \mathsf{N_{int}}$, with $\mathsf{N_d}$ the decisions, $\mathsf{N_g}$ the goals, let $ABF = \langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \mathcal{C} \rangle$ be the dominant ABA framework drawn from *CG*. Let $d \in \mathsf{N_d}$ be a non-dominant decision in *CG* and the set of arguments E_{arg} be an arg-explanation for the non-admissible argument $R_a = \{dom(d)\} \vdash dom(d)$ in ABF. Then for each argument $\{\} \vdash_{\mathsf{R}} \neg dEdge(n_i, n_j, t) \in E_{arg}$, $\mathsf{R} \subset \mathcal{R}$ is in a cont-explanation E_{cont} of R_a.

According to Definition 5.4, if an argument has a cont-explanation, it must also have an arg-explanation, but not vice versa. A cont-explanation is derived from the ABA rules in arguments of inapplicable edges. It is worth noting that inapplicable edges are not always relevant to the computation of cont-explanations. Only the ones that affects the reachability of goals from the concerned decision are relevant to the derivation of cont-explanations. Formally:

Proposition 5.5. *Given a DGC $CG = \langle \mathsf{N}, \mathsf{E}, \mathsf{C} \rangle$, $\mathsf{N} = \mathsf{N_d} \cup \mathsf{N_g} \cup \mathsf{N_{int}}$, with $\mathsf{N_d}$ the decisions, $\mathsf{N_g}$ the goals, let $ABF = \langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \mathcal{C} \rangle$ be the dominant ABA framework drawn from CG, in which*

- *$d \in \mathsf{N_d}$ be a non-dominant decision in ABF that fails to meet some goal $g \in \mathsf{N_g}$, and there exists an argument $\Delta \vdash met(d, g)$;*
- *$d' \in \mathsf{N_d}$ meets g, $d' \neq d$;*
- *edge $[n_i | n_j] \in \mathsf{E_{ia}}$ is inapplicable in the context C and the argument for its inapplicability is $\{\} \vdash_{\mathsf{R}} \neg dEdge(n_i, n_j, t)$, $\mathsf{R} \subset \mathcal{R}$.*

Then R is in a cont-explanation of $\{dom(d)\} \vdash dom(d)$ if and only if $dEdge(n_i, n_j, t) \in \Delta$.

Proof. An intuitive proof of Proposition 5.5 is as follows. According to Definition 5.4, if R is in a cont-explanation of $R_a = \{dom(d)\} \vdash dom(d)$, argument $E = \{\} \vdash_{\mathsf{R}} \neg dEdge(n_i, n_j, t)$ must be in an arg-explanation of R_a. Thus, E must be an attack on the argument $D = \Delta \vdash met(d, g)$, and D cannot be defended from E. Since in ABA attacks can only be directed at the assumptions in the support of an argument, $dEdge(n_i, n_j, t)$ must be in the support Δ of D. Conversely, if $dEdge(n_i, n_j, t)$ is in the support of argument D, then argument E forms an attack from which D cannot be defended and renders R_a non-admissible. Thus, E is in an arg-explanation of R_a and R is in a cont-explanation of R_a. □

Example 5.6. (Example 5.3 continued.) The argument $\{\} \vdash_{\mathsf{R}} \neg dEdge(clo, na, 1)$ where $\mathsf{R} = \{\neg dEdge(clo, na, 1) \leftarrow clo\_allergy, clo\_allergy \leftarrow\}$ is in the arg-explanation of $R_a = \{dom(clo)\} \vdash dom(clo)$. By Definition 5.4, the cont-explanation for R_a is $E_{cont} = \{\neg dEdge(clo, na, 1) \leftarrow clo\_allergy, clo\_allergy \leftarrow\}$. According to Proposition 5.2, we can interpret this cont-explanation as follows:

The decision of administering clopidogrel does not meet the goal safe as the patient has an allergy to clopidogrel.

We can also observe that in this example, $dEdge(clo, na, 1)$ is in the support of argument $\Delta \vdash met(clo, safe)$.

6 RELATED WORK

Researches on context-based decision making have focused on building knowledge representation models and developing decision making methods. Logical models, such as ontologies [6, 30] and logic rules [19, 22, 27], have been widely used due to their high interpretability and expressibility. Subsumption checking in an ontology is used to perform activity recognition in [6]. A first order logic model is introduced in [27] to express complex rules with context. Several Bayesian approaches [18, 25] have also been proposed to model the decision making process. A combined approach is adopted in [5], which employs ontologies and logic rules for representation and makes decisions with Markov Logic Networks. In [20] and [31], case-based reasoning is used to generate context awareness by referring back to similar previous scenarios. Different from these works, we take an argumentation-based approach, which enjoys the benefits of transparent decision making and ease of explanation generation.

In an argumentation-based approach, argumentation has been used as a formalism for representing decision problems as well as a reasoning mechanism for computing decisions guided by some decision criteria. In [1], arguments in favour or against each decision alternative are constructed, and then evaluated against a pessimistic or an optimistic criterion. In [2], arguments for decisions are built using AA and their acceptability are evaluated with the classical semantics. With only the accepted arguments, binary comparisons among the decisions are performed using a unipolar, bipolar or non-polar decision criterion. Similar to our approach, a unified process is used in [11] and [21] to map the decision model to an ABA framework and rely on the admissibility semantics to compute dominant decisions. Our main contribution, as compared to other argumentation-based works including [11] and [21], is proposing a new approach that is able to incorporate and reason with context. We provided constructs for modelling defeasible relationships between decisions and intermediates and between intermediates and goals, which are moderated by contexts. Contexts are modelled as rules in an ABA framework, which influence the decision making by moderating the reachability of goals from decisions.

Recently, more research efforts have been spent on realizing the explanatory power of argumentation. Argumentative explanations for the case-based reasoning process are studied in [7], while explanations for solutions to Answer Set Programming problems are studied in [28]. Formal definitions of argumentative explanations in AA and ABA have been introduced in [13] and [14] respectively, which are based on the related admissibility semantics proposed in the articles. While [13] and [14] focus on generating explanations for adopted decisions, [15] focuses on generating explanations for unadopted decisions. In [32], natural language explanations were generated for the preferred decision after pairwise comparisons using information extracted from dispute trees. Similar to [15], we also studied explanations for unadopted decisions from an argument-view. However, our explicit modelling of context has enabled the study of explanation from a new context point-of-view.

7 CONCLUSION AND FUTURE WORK

In this paper, we presented an argumentation-based approach for making context-based and explainable decisions. To incorporate context into problem representation and the decision making process, Decision Graphs with Context (DGC) was proposed to model decision problems with varying contexts. The problem of computing decisions in DGCs was then converted into determining argument admissibility in ABA frameworks by mapping DGCs to ABA frameworks. To improve the transparency of the proposed decision making approach to humans, we formalised two types of explanations with their computations, *argument explanation* and *context explanation*, which can help to explain the underlying reasons for not choosing a decision from two different point-of-views.

We have focused on incorporating context into problem representation and the decision making process. Our contributions include: (1) introduction of a new graphical representation for context-based decision problems, (2) computation of context-based decisions and (3) formalisation of two notions of argumentative explanations, including one based on contexts.

As our proposed approach is more for the case of a single-agent system, we are interested to explore how it can be applied to a multi-agent system. ABA dialogues and frameworks can be used to communicate arguments and compute decisions in a multi-agent system. In [16], the agents exchange arguments in the form of an ABA dialogue. A joint ABA framework can be constructed from the dialogue to model the decision-making process of two agents. Our proposed approach is also based on ABA framework. Hence, theoretically, the methods we used to make context-based decisions and to generate the two types of explanations can also be applied to the joint ABA framework for two or more agents. However, protocols are required to govern the dialogues among agents. Also, since agents may have different goals and candidate decisions, multiple Decision Graph with Context (DGC) (one for each agent) may be needed to model the problem.

We have only focused on explanations for non-admissible decisions in this paper. Currently, we are working on generating the two types of explanations for admissible decisions. Some of our discussions regarding the applicability of edges share similar ideas as the reasoning in higher order argumentation semantics, such as [3, 17, 24, 26], where arguments are allowed to attack relations and other arguments. In the future, we will study the relevancy between our approach and these higher order argumentation semantics. We are also interested in incorporating preferences in addition to contexts.

ACKNOWLEDGMENTS

This research is supported by the National Research Foundation, Prime Minister's Office, Singapore under its IDM Futures Funding Initiative, the Singapore Ministry of Health under its National Innovation Challenge on Active and Confident Ageing (NIC Project No. MOH/NIC/COG04/2017), the Interdisciplinary Graduate School, Nanyang Technological University under its Graduate Research Scholarship and the Alibaba-NTU Singapore Joint Research Institute.

REFERENCES

[1] Leila Amgoud and Henri Prade. 2006. Explaining Qualitative Decision under Uncertainty by Argumentation.. In *Proceedings of the National Conference on Artificial Intelligence*, Vol. 21. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 219.

[2] Leila Amgoud and Henri Prade. 2009. Using arguments for making and explaining decisions. *Artificial Intelligence* 173, 3-4 (2009), 413–436.

[3] Pietro Baroni, Federico Cerutti, Massimiliano Giacomin, and Giovanni Guida. 2011. AFRA: Argumentation framework with recursive attacks. *International Journal of Approximate Reasoning* 52, 1 (2011), 19–37.

[4] Federico Cerutti, Nava Tintarev, and Nir Oren. 2014. Formal Arguments, Preferences, and Natural Language Interfaces to Humans: an Empirical Evaluation.. In *ECAI*. 207–212.

[5] Pedro Chahuara, François Portet, and Michel Vacher. 2013. Making context aware decision from uncertain information in a smart home: A markov logic network approach. In *Proceedings of 4th International Joint Conference on Ambient Intelligence (AmI'13)*. Springer, 78–93.

[6] Liming Chen and Chris Nugent. 2009. Ontology-based activity recognition in intelligent pervasive environments. *International Journal of Web Information Systems* 5, 4 (2009), 410–430.

[7] Kristijonas Cyras, Ken Satoh, and Francesca Toni. 2016. Explanation for case-based reasoning via abstract argumentation. In *6th International Conference on Computational Models of Argument (COMMA 2016)*. IOS PRESS, 243–254.

[8] Phan Minh Dung. 1995. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence* 77, 2 (1995), 321–357.

[9] Phan Minh Dung, Robert A Kowalski, and Francesca Toni. 2009. Assumption-based argumentation. In *Argumentation in Artificial Intelligence*. Springer, 199–218.

[10] Phan Minh Dung, Paolo Mancarella, and Francesca Toni. 2007. Computing ideal sceptical argumentation. *Artificial Intelligence* 171, 10 (2007), 642–674.

[11] Xiuyi Fan and Francesca Toni. 2013. Decision making with assumption-based argumentation. In *International Workshop on Theorie and Applications of Formal Argumentation*. Springer, 127–142.

[12] Xiuyi Fan and Francesca Toni. 2014. A general framework for sound assumption-based argumentation dialogues. *Artificial Intelligence* 216 (2014), 20–54.

[13] Xiuyi Fan and Francesca Toni. 2014. On computing explanations in abstract argumentation. In *Proceedings of the 21st European Conference on Artificial Intelligence*. IOS Press, 1005–1006.

[14] Xiuyi Fan and Francesca Toni. 2015. On Computing Explanations in Argumentation.. In *Proceedings of 29th AAAI Conference on Artificial Intelligence (AAAI-15)*. 1496–1502.

[15] Xiuyi Fan and Francesca Toni. 2015. On Explanations for Non-Acceptable Arguments. In *International Workshop on Theorie and Applications of Formal Argumentation*. Springer, 112–127.

[16] Xiuyi Fan, Francesca Toni, Andrei Mocanu, and Matthew Williams. 2014. Dialogical two-agent decision making with assumption-based argumentation. In *Proceedings of the 2014 International Conference on Autonomous Agents and Multi-agent Systems (AAMAS 2014)*. International Foundation for Autonomous Agents and Multiagent Systems, 533–540.

[17] Dov M Gabbay. 2009. Semantics for higher level attacks in extended argumentation frames part 1: Overview. *Studia Logica* 93, 2-3 (2009), 357.

[18] Tao Gu, Hung Keng Pung, and Da Qing Zhang. 2005. A service-oriented middleware for building context-aware services. *Journal of Network and computer applications* 28, 1 (2005), 1–18.

[19] Hsien-Chou Liao and Chien-Chih Tu. 2007. A RDF and OWL-Based Temporal Context Reasoning Model for Smart Home. *Information Technology Journal* 6, 8 (2007), 1130–1138.

[20] Tinghuai Ma, Yong-Deak Kim, Qiang Ma, Meili Tang, and Weican Zhou. 2005. Context-aware implementation based on CBR for smart home. In *Proceedings of 12th IEEE International Conference on Wireless And Mobile Computing, Networking And Communications (WiMob'05)*, Vol. 4. IEEE, 112–115.

[21] Paul-Amaury Matt, Francesca Toni, and Juan R Vaccari. 2009. Dominant Decisions by Argumentation Agents. *ArgMAS* 6057 (2009), 42–59.

[22] Alessandra Mileo, Davide Merico, and Roberto Bisiani. 2011. Reasoning support for risk prediction and prevention in independent living. *Theory and Practice of Logic Programming* 11, 2-3 (2011), 361–395.

[23] S Modgil and J Fox. 2009. A guardian agent approach to safety in medical multi-agent systems. In *Safety and security in multiagent systems*. Springer, 67–79.

[24] Sanjay Modgil and Henry Prakken. 2013. A general account of argumentation with preferences. *Artificial Intelligence* 195 (2013), 361–397.

[25] Takashi Nishiyama, Shinpei Hibiya, and Tetsuo Sawaragi. 2011. Development of agent system based on decision model for creating an ambient space. *AI & society* 26, 3 (2011), 247–259.

[26] Sylwia Polberg. 2017. *Intertranslatability of abstract argumentation frameworks*. Technical Report. Technical Report DBAI-TR-2017-104, Institute for Information Systems, Technical University of Vienna.

[27] Anand Ranganathan and Roy H Campbell. 2003. An infrastructure for context-awareness based on first order logic. *Personal and Ubiquitous Computing* 7, 6 (2003), 353–364.

[28] Claudia Schulz. 2015. Explaining Answer Set Programming in Argumentative Terms.. In *Proceedings of 29th AAAI Conference on Artificial Intelligence (AAAI-15)*. 4253–4254.

[29] Francesca Toni. 2014. A tutorial on assumption-based argumentation. *Argument & Computation* 5, 1 (2014), 89–117.

[30] Peter Wolf, Andreas Schmidt, and Michael Klein. 2008. SOPRANO-An extensible, open AAL platform for elderly people based on semantical contracts. In *3rd Workshop on Artificial Intelligence Techniques for Ambient Intelligence (AITAmI'08)*.

[31] Bingchuan Yuan and John Herbert. 2014. Context-aware hybrid reasoning framework for pervasive healthcare. *Personal and ubiquitous computing* 18, 4 (2014), 865–881.

[32] Qiaoting Zhong, Xiuyi Fan, Francesca Toni, and Xudong Luo. 2014. Explaining Best Decisions via Argumentation.. In *ECSI*. 224–237.