Abstract

In a normative environment an agent’s actions are not only directed by its goals but also by norms. Here, potential conflicts among the agent’s goals and norms makes decision-making challenging. We therefore seek to answer the following questions: (i) how should an agent act in a normative environment? and (ii) how can the agent explain why it acted in a certain way? We propose a solution in which a normative planning problem serves as the basis for a practical reasoning approach based on argumentation. The properties of the best plan(s) w.r.t. goal achievement and norm compliance are mapped to arguments that are used to explain why a plan is justified, using a dialogue game.

1 Introduction

Agents in normative systems must be able to reason about actions in pursuit of their goals, but must also consider the regulative norms imposed on them. Such norms define obligations and prohibitions on their behavior, and to avoid punishment, agents must comply with norms while pursuing their goals. However, if norm compliance hinders a more important goal or norm, an agent should consider violating it. To decide how to act, an agent thus needs to generate all plans and weigh up the importance of goal achievement and norm compliance against the cost of goal failure and norm violation in different plans. Although some reasoning frameworks do this [Broersen et al., 2001; Kollingbaum and Norman, 2003], little attention has been paid to explaining the agents’ decision making in such frameworks. Such explanation is important in contexts including human-agent teams and agent debugging, and to provide explanation, we propose utilising formal argumentation.

Argumentation has been applied to inconsistency handling and decision-making [Dung, 1995; Amgoud and Prade, 2009], and its dialogical interpretation makes it an appropriate tool to generate explanations for decisions [Fan and Toni, 2015; Caminada et al., 2014b]. Although argumentation has been extensively used in practical reasoning (e.g., [Atkinson and Bench-Capon, 2007]), integrating the reasoning and dialogical aspect of argumentation for decision-making and its explanation has not been addressed by existing approaches.

In this paper we propose an argumentation-based approach to normative practical reasoning using a dialogue game to provide an intuitive overview of agent’s reasoning. In achieving this aim, the following contributions are made: (i) we formalise a set of argument schemes and critical questions [Walton, 1996] aimed at checking plan justifiability with respect to goal satisfaction and norm compliance/violation; (ii) we offer a novel decision criterion that identifies the best plan(s) both in the presence and absence of preferences over goals and norms; and (iii) we investigate the properties of the best plan(s). These properties, together with Caminada’s Socratic dialogu game [Caminada et al., 2014a], are used to generate an explanation for the justifiability of the best plan(s).

2 Model

This section introduces a model for normative practical reasoning based on STRIPS planning [Fikes and Nilsson, 1971].

Definition 1 (Normative Planning Problem). A normative planning problem is a tuple \( P = \langle FL, \Delta, A, G, N \rangle \) where \( FL \) is a set of fluents; \( \Delta \subseteq FL \) is the initial state; \( A \) is a finite, non-empty set of durative actions; \( G \) is the set of agent goals; and \( N \) is a set of action-based norms imposed on the agent.

Fluents \( FL \) is a set of domain fluents. A literal \( l \) is a fluent or its negation. For a set of literals \( L \), we define \( L^+ = \{ l \text{ s.t. } l \in L \} \) and \( L^- = \{ l \text{ s.t. } \neg l \in L \} \). \( L \) is well-defined if \( L^+ \cap L^- = \emptyset \). For a state \( s \subseteq FL \), \( s^+ \) are fluents considered true, and \( s^- = FL \setminus s^+ \). A state \( s \) satisfies literal \( l \), denoted as \( s \models l \), if \( l \in s \), and satisfies literal \( \neg l \), denoted \( s \models \neg l \), if \( l \notin s \).

Actions An action \( a = \langle pr, ps, d \rangle \) is composed of well-defined sets of literals \( pr, ps \) that represent \( a \)’s pre- and postconditions respectively, and a number \( d \in \mathbb{N} \) representing the action’s duration. Given an action \( a = \langle pr, ps, d \rangle \), we write \( pr(a), ps(a) \) and \( d(a) \) for \( pr, ps, \) and \( d \). Postconditions are divided into add \( (ps(a)^+) \) and delete \( (ps(a)^-) \) postcondition sets. An action \( a \) can be executed in state \( s \) iff the state satisfies its preconditions. The postconditions of a durative action are applied in the state \( s \) at which the action ends, by adding the positive postconditions belonging to \( ps(a)^+ \) and deleting the negative postconditions belonging to \( ps(a)^- \).

Goals Achievement goals instantaneously achieve a certain state of affairs. Each \( g \in G \) is a well-defined set of literals
g = \{r_1, \ldots, r_n\} \subseteq D, known as goal requirements (denoted as r_i, that should be satisfied in the state to satisfy the goal.

**Norms** An action-based norm is defined as a tuple n = \langle d, o, a_{con}, a_{sub}, dl \rangle, where d, o \in \{o, f\} is the deontic operator denoting obligation or prohibition; a_{con} \in A is the action that activates the norm; a_{sub} \in A is the action that is the subject of the obligation or prohibition; and dl \in N is the norm deadline relative to the completion of the execution of the action a_{con}, the activation condition of the norm.

### 2.1 Semantics

Let P = \langle FL, \Delta, A, G, N \rangle be a normative planning problem. Also let \pi = \langle (a_0, 0), \ldots, (a_n, t_a) \rangle be a sequence of actions such that \pi(a_i, t_a) = (a_j, t_j) \in \pi \iff t_a < t_j < t_a + d(a_i), (a_i, a_j) \in cf_{action}, where cf_{action} is defined below.

**Definition 2 (Conflicting Actions).** Actions a_i and a_j have a concurrency conflict iff the preconditions or postconditions of a_i contradict the preconditions or postconditions of a_j.

\[
\text{cf}_{\text{action}} = \{(a_i, a_j) : \exists r \in \text{ps}(a_i) \cup \text{ps}(a_j), \neg r \in \text{ps}(a_j) \cup \text{ps}(a_i)\}
\]

The duration of a sequence of actions \pi is calculated as Makespan(\pi) = max(t_a + d(a_i)). The execution of \pi from a starting state s_0 brings about a sequence of states S(\pi) = \langle s_0, \ldots, s_m \rangle for every discrete time interval from 0 to m = Makespan(\pi). The transition relation between two states is as follows. Let A_k be the set of action, time pairs such that the actions end at state s_k. State s_k results from removing all delete postconditions and adding all add postconditions of actions in A_k to state s_{k-1}. I.e., \forall 0 < k \leq m:

\[
s_k = \left\{ \begin{array}{l}
(s_{k-1} \setminus \bigcup_{a \in A_k} \text{ps}(a)^-) \cup \bigcup_{a \in A_k} \text{ps}(a)^+ & \text{if } A_k \neq \emptyset \\
A_k = \emptyset & \text{otherwise}
\end{array} \right.
\]

\[\pi \text{ satisfies a goal if there is a state that satisfies the goal: } \pi \models g \iff \exists s_k \in S(\pi) \text{ s.t. } s_k \models g.\] The set of satisfied goals by \pi is denoted \Gamma_\pi.

\[\pi \text{ complies with an obligation} \text{ if the action that is the subject of the obligation, } a_{sub}, \text{occurs during the compliance period (i.e., between when the condition holds and when the deadline expires):} \]

\[
\pi \models n \iff (a_{con}, t_{a_{con}}), (a_{sub}, t_{a_{sub}}) \in \pi \text{ s.t. } \]  
\[
t_{a_{sub}} \in [t_{a_{con}} + d(a_{con}), t_{a_{con}} + d(a_{con})]
\]

If a_{sub} does not occur during the compliance period, the obligation is violated: \pi \not\models n.

\[\pi \text{ complies with a prohibition} \text{ if the prohibition’s subject action } a_{sub} \text{ does not occur during the compliance period:} \]

\[
\pi \models n \iff (a_{con}, t_{a_{con}}) \in \pi, \bar{\pi}(a_{sub}, t_{a_{sub}}) \in \pi \text{ s.t. } \]  
\[
t_{a_{sub}} \in [t_{a_{con}} + d(a_{con}), t_{a_{con}} + d(a_{con})]
\]

If a_{sub} occurs during the compliance period, the prohibition is violated: \pi \not\models n.

We assume that all norm deadlines end before Makespan(\pi). Therefore, all activated norms in \pi (denoted as N_\pi) are either complied with (denoted N_{com}(\pi)) or violated (denoted N_{viol}(\pi)) by time m. Another assumption made is deontic detachment [Andrighetto et al., 2013], meaning that norm instances are unique, even if a norm is invoked several times.

### 2.2 Conflict

In this section we consider several types of conflict, defining which sequences of action within a plan are determined to be in conflict. We consider a running example where an agent has the goals of going on strike; submitting a report; and getting a certificate of some sort. However, if the agent goes on maternity leave, it cannot go to office or attend meetings.

**Definition 3 (Conflicting Goals).** Goal g_i and g_j are in conflict if satisfying them requires bringing about conflicting state of affairs.

\[cf_{goal} = \{ (g_i, g_j) : \exists r \in g_i, \neg r \in g_j \} \]

**Example 1.** The goal strike, made up of fluenets \{union_member, ~at_office, ~meeting_attended\} and goal submission, with fluenets \{at_office, report_finalised\} are in conflict.

**Definition 4 (Conflicting Obligations and Goals).** Norm n = \langle o, a_{con}, a_{sub}, dl \rangle and goal g are in conflict if executing action a_{sub} — the subject of the obligation — brings about postconditions that are in conflict with the requirements of g.

\[cf_{goalobl} = \{ (g, n) : \exists r \in g, \neg r \in \text{ps}(a_{sub}) \} \]

**Example 2.** Goal strike and norm n_1 = \langle o, get_company_funding, attend_meeting 2 \rangle, obligating meeting attendance if company funding is used are in conflict, since postcondition meeting_attended of attend_meeting is incompatible with the fluenets of strike.

**Definition 5 (Conflicting Prohibitions and Goals).** A prohibition norm n = \langle f, a_{con}, a_{sub}, dl \rangle and a goal g are in conflict, if the postconditions of a_{sub} contribute to satisfying g via r (and r cannot be brought about by any other action), but executing action a_{sub} is prohibited by norm n.

\[cf_{goalpro} = \{ (g, n) : \exists r \in g, r \in \text{ps}(a_{sub}) \} \]

**Example 3.** Submission = \{at_office, report_finalised\} and n_2 = \langle f, take_maternity_leave, go_to_office, 6 \rangle are in conflict since taking maternity leave prevents the agent from going to the office and hence prevents fulfilling the goal of submission: (submission, n_2) \in cf_{goalpro}.

The entire set of conflicting goals and norms is defined as:

\[cf_{goalnorm} = cf_{goalobl} \cup cf_{goalpro} \]

**Definition 6 (Conflicting Obligations).** n_1 = \langle o, a_{con}, a_{sub}, dl \rangle and n_2 = \langle o, b_{con}, b_{sub}, dl' \rangle are in conflict in the context of \pi if the obliged actions in n_1, i.e., a_{sub}, and n_2, i.e., b_{sub} have a concurrency conflict; and action a_{sub} is in progress during the entire period over which the agent is obliged to execute action b_{sub}.

\[cf_{obold} = \{ (n_1, n_2) : \exists a_{con}, a_{sub}, (b_{con}, b_{sub}), t_{a_{con}} \in \pi \} \]

\[
\text{such that } t_{a_{con}} + d(a_{con}) + d(a_{sub}) + dl' \subseteq [t_{a_{con}} + d(a_{con}), t_{a_{con}} + d(a_{con}) + dl] \cup [t_{b_{con}} + d(b_{con}) + d(b_{sub})] \]

\]
Example 4. Due to the concurrency conflict between actions `attend_meeting` and `attend_interview`, in \( n_1 = (o, get\_company\_funding, attend\_meeting, 2) \) and \( n_4 = (o, take\_theory\_test, attend\_interview, 2) \) and depending on the way actions are sequenced in a plan, it is possible that in some \( \pi: (n_1, n_4) \in cf_{\text{oblpro}} \).

**Definition 7** (Conflicting Obligations and Prohibitions). An obligation \( n_1 = (o, a_{con}, a_{sub}, dl) \) and a prohibition \( n_2 = (f, b_{con}, a_{sub}, dl') \) are in conflict in the context of \( \pi \) if \( n_2 \) forbids the agent to execute \( a_{sub} \) during the entire period over which obligation \( n_1 \) obliges the agent to take \( a_{sub} \).

\[
e_{\text{oblpro}}^\pi = \{(n_1, n_2) \mid (a_{con}, t_{a_{con}}), (b_{con}, t_{b_{con}}) \in \pi; [t_{a_{con}} + d(a_{con}), t_{a_{con}} + d(a_{con}) + dl) \subseteq [t_{b_{con}} + d(b_{con}), t_{b_{con}} + d(b_{con}) + dl')\}
\]

Example 5. The obligation and prohibition \( n_1 = (o, get\_company\_funding, attend\_meeting, 2) \) and \( n_3 = (f, take\_maternity\_leave, attend\_meeting, 6) \) can be in conflict in some \( \pi \) as they require and forbid attending a meeting. Thus, for some \( \pi, (n_1, n_3) \in cf_{\text{oblpro}} \).

The two sets \( cf_{\text{oblpro}}^\pi \) and \( cf_{\text{oblpro}}^\pi \) constitute the set of conflicting norms: \( cf_{\text{norm}}^\pi = cf_{\text{oblpro}}^\pi \cup cf_{\text{oblpro}}^\pi \).

**Definition 8** (Plan). A sequence of actions \( \pi = ((a_0, 0), \ldots, (a_n, t_a)) \) s.t. \( \pi((a_i, t_{a_i}), (a_j, t_{a_j}) \in \pi \mid t_{a_i} \leq t_{a_j} < t_{a_i} + d(a_i), (a_i, a_j) \in cf_{\text{action}} \) is a plan for the normative planning problem \( \mathcal{P} = (FL, \Delta, A, G, N) \) iff:

- Only the flents in \( \Delta \) hold in the initial state: \( s_0 = \Delta \)
- The preconditions of action \( a_i \) holds at time \( t_{a_i} \) and throughout the execution of \( a_i \):
  \[
  \forall k \in [t_{a_i}, t_{a_i} + d(a_i)], s_k \models pr(a_i)
  \]
- The set of goals satisfied by plan \( \pi \) is a non-empty (\( G_\pi \neq \emptyset \)) consistent subset of goals:
  \[
  G_\pi \subseteq G \text{ and } \exists g_1, g_j \in G_\pi \text{ s.t. } (g_1, g_j) \in cf_{\text{goal}}
  \]
- There is no conflict between the goals satisfied and norms complied with:
  \[
  \exists g \in G_\pi \text{ and } n \in N_{\text{cmp}(\pi)} \text{ s.t. } (g, n) \in cf_{\text{goalnorm}}
  \]

Note that since norms are action-based and there is no possibility of executing conflicting actions, there will be no conflict between the norms complied with in a plan.

We consider a set of plans \( \Pi \), and in the next section deal with the problem of choosing the best plan from this set.

3 Identifying the Best Plan

The conflict between an agent’s goals and norms often makes it impossible for the agent to satisfy all its goals while complying with all norms triggered in a plan. We begin by considering how to treat each plan as a proposal of actions and how to use argumentation schemes to check the justifiability of a plan proposal with respect to conflicts and preferences. In Section 3.2, we then identify the best plan from the justified subset.

3.1 Generating Arguments

An argumentation framework (AF) consists of a set of arguments and attacks between them [Dung, 1995]: \( AF = (Arg, Att) \), \( Att \subseteq Arg \times Arg \). In scheme-based approaches [Walton, 1996] arguments are expressed in natural language and a set of critical questions is associated with each scheme, identifying how the scheme can be attacked. Below, we introduce a set of argument schemes and critical questions to reason about a plan proposal with respect to the goals it satisfies and norms it complies with or violates.

**Definition 9** (Plan Argument Scheme \( Arg_\pi \)). A plan argument claims that a proposed sequence of actions should be executed because it satisfies a set of goals, and complies with a set of norms while violating some other norms:

- In the initial state \( \Delta \)
- The agent should execute sequence of actions \( \pi \)
- Which will satisfy set of goals \( G_\pi \) and complies with set of norms \( N_{\text{comp}(\pi)} \) and violates set of norms \( N_{\text{vol}(\pi)} \)

**Definition 10** (Goal Argument Scheme \( Arg_g \)). A goal argument claims that a feasible goal should be satisfied:

- Goal \( g \) is feasible\(^1\) for the agent
- Therefore, satisfying \( g \) is required.

The set of goal argument is denoted as \( Arg_G \).

**Definition 11** (Norm Argument Scheme \( Arg_n \)). A norm argument claims that an activated norm should be complied with:

- \( n \) is an activated norm imposed on the agent in plan \( \pi \)
- Therefore, complying with \( n \) is required in \( \pi \).

The set of norm argument for a plan is denoted as \( Arg_N_\pi \).

Critical Questions for the Plan Argument Scheme

**CQ1**: Is there a goal argument which attacks \( Arg_\pi \)? This CQ results in an undercut attack (asymmetric by nature) from a goal argument to a plan argument, when the goal is not satisfied in the plan:

\[
\forall \text{Arg}_g \in Arg_G \text{ if } \pi \nmid g \text{ then } (\text{Arg}_g, \text{Arg}_\pi) \in Att
\]

**CQ2**: Is there a norm argument which attacks \( Arg_\pi \)? This CQ results in an undercut from a norm argument to a plan argument, when the norm is violated in the plan:

\[
\forall \text{Arg}_n \in Arg_N_\pi \text{ if } \pi \nmid n \text{ then } (\text{Arg}_n, \text{Arg}_\pi) \in Att
\]

Critical Questions for the Goal Argument Scheme

**CQ3**: What goal arguments might attack \( Arg_g \)? This CQ results in a rebuttal attack (symmetric by definition) between arguments for conflicting goals:

\[
\forall \text{Arg}_g, \text{Arg}_g' \in Arg_G \text{ if } (g, g') \in cf_{\text{goal}} \text{ then } (\text{Arg}_g, \text{Arg}_g') \in Att
\]

**CQ4**: What norm arguments might attack \( Arg_g \)? This CQ results in a rebuttal attack between arguments for a goal and a norm that are in conflict:

\[
\forall \text{Arg}_g \in Arg_G, \text{Arg}_n \in Arg_N_\pi \text{ if } (g, n) \in cf_{\text{goalnorm}} \text{ then } (\text{Arg}_g, \text{Arg}_n) \in Att
\]

\(^1\)A goal is feasible if there is at least one plan that satisfies it.
Critical Questions for the Norm Argument Scheme

CQ4: What goal arguments might attack the norm presented by Arg\(_g\)? The previous critical question is associated with argument schemes for norms as well as goals, hence the repetition of the critical question.

\[\forall Arg_g \in Arg_G, Arg_n \in Arg_{N_e}\]
\[\text{if } (n, g) \in c_f_{goalnorm} \text{ then } (Arg_n, Arg_g) \in Att\]

CQ5: What norm arguments might attack the norm presented by Arg\(_n\)? Conflict between two norms is defined as a contextual conflict that depends upon the context of the plan in which the norms are activated.

\[\forall Arg_n, Arg_{n'} \in Arg_{N_e}\]
\[\text{if } (n, n') \in c_f_{norm} \text{ then } (Arg_n, Arg_{n'}) \in Att\]

Preferences between arguments distinguish an attack from a defeat (i.e., a successful attack [Amgoud and Cayrol, 2002]). The attack from one argument to another is a defeat if the latter argument is not preferred over the former. However, as discussed in [Prakken, 2012], rebuttal attacks are preference-dependent, whereas undercuts are preference-independent. Thus, attacks due to CQ3, CQ4 and CQ5 need preferences to be resolved, while attacks caused by CQ1 and CQ2 are preference-independent, always resulting in defeat.

We define \(\succeq^{gn}\) as a partial preorder on \(G \cup N\). \(\succeq^{gn}\) denotes the strict relation corresponding to \(\succeq^{gn}\). Also, \((\alpha, \beta) \in \sim^{gn}\) iff \((\alpha, \beta) \in \succeq^{gn}\) and \((\beta, \alpha) \in \succeq^{gn}\). The preferences between the goal and norm arguments result from the preference between these entities: \((Arg_{g_0}, Arg_{g_1}) \in \succeq^{gn}\) \((\alpha, \beta) \in \sim^{gn}\).

An AF for a plan proposal consists of the argument for the plan itself, a set of arguments for goals and arguments for norms that are activated in that plan. Although the set of goal arguments in AFs for plan proposals remain the same across the AFs, the set of norm arguments differs between AFs depending on the norms that are activated by the plan proposal in each AF.

**Definition 12** (Plan Proposal AF). The AF for plan proposal \(\pi\) is \(AF_\pi = \langle Arg, Def_f\rangle\), where \(Arg = Arg_g \cup Arg_G \cup Arg_{N_e}\) and \(Def_f\) is defined as: \(\forall Arg_{g_0}, Arg_{g_1} \in Arg, (Arg_{g_0}, Arg_{g_1}) \in Def_f\) iff \((Arg_{g_0}, Arg_{g_1}) \in Att_{CQ1-5}\) and \((Arg_g, Arg_{g_1}) \not\in Att_5\).

The next section explains how an AF for a plan proposal is evaluated and used to identify the best plan(s).

### 3.2 Evaluating the Argumentation Framework

Argumentation semantics [Dung, 1995] are a means for evaluating arguments in an AF. Among the proposed semantics are the credulous preferred semantics which several authors have suggested [Caminada, 2006; Prakken, 2006; Oren, 2013] are appropriate for reasoning about actions. Caminada [2006] provides an intuitive way to identify the status of arguments w.r.t. various semantics through labellings. An argument is, respectively, labelled \(in\), \(out\) and \(undec\), if it is acceptable, rejected and undecided under a certain semantics. In a complete labellings \(L_{cmp}\), an argument is labelled \(in\) iff all its attackers are labelled \(out\), and is labelled \(out\) iff there exists an attacker for it that is labelled \(in\). A complete labelling in which the set of arguments labelled \(in\) are maximal (w.r.t. set inclusion) is a preferred labelling \(L_{pr}\). An argument is credulously accepted under preferred semantics if it is labelled \(in\) by at least one preferred labelling.

**Definition 13** (Justified Plans). Plan \(\pi\) is justified if \(Arg_{g}\) is labelled \(in\) by at least one preferred labelling for \(AF_\pi\): \(\exists L_{pr} \text{ s.t. } Arg_{g} \in in(L)\).

Although all justified plans are internally consistent, they can still be disagreed with externally. That is, there might be further criteria to take into account when identifying the best plan among justified plans. We define the criteria for the best plan(s) using an established set ordering principle in argumentation, the Democratic principle: \((S_\alpha, S_\beta) \in \succeq \iff \forall \beta \in S_j \setminus S_i, \exists \alpha \in S_i \setminus S_j \text{ s.t. } (\alpha, \beta) \in \succeq\). Since preferences over goals and norms are partial, comparing two plans based on the set of goals and norms is not always possible. Therefore, absent such preference information, the best plan(s) satisfies the most goals while violating the fewest norms. We start by defining the goal-dominant and norm-dominant plans, based on which a better than relation between plans is defined.

Let \(G_{\Pi} = \{G_{\pi_1}, G_{\pi_2}, \ldots, G_{\pi_n}\}\), where \(G_{\pi}\) is the set of goals satisfied in plan \(\pi\).

**Definition 14** (Goal-dominance). Plan \(\pi_1, \pi_2\) goal-dominates \(\pi_j\) denoted as \((\pi_1, \pi_j) \in \succeq^{gn}\) if
1. \(\succeq^{G}\) is a total preorder on \(G_{\Pi}\) and \((G_{\pi_1}, G_{\pi_j}) \in \succeq^{G}\); or
2. \(|G_{\pi_1}| \geq |G_{\pi_j}|\) (i.e., if \(\succeq^{G}\) is not a total preorder on \(G_{\Pi}\)).

**Definition 15** (Norm-dominance). Plan \(\pi_1\) norm-dominates \(\pi_2\) denoted as \((\pi_1, \pi_2) \in \succeq^{gn}\) if
1. \(\succeq^{N}\) is a total preorder on \(N_{vol(\Pi)}\) and \((N_{vol(\pi_1)}, N_{vol(\pi_2)}) \in \succeq^{N}\); or
2. \(|N_{vol(\pi_1)}| \geq |N_{vol(\pi_2)}|\) (i.e., if \(\succeq^{N}\) is not a total preorder on \(N_{vol(\Pi)}\)).

**Definition 16** (Plan Comparison). Plan \(\pi_1\) is better than \(\pi_j\) denoted \((\pi_1, \pi_j) \in \succeq^{\pi}\) iff:
1. \(\pi_1\) is justified and \(\pi_1\) is not; or
2. \(\pi_1, \pi_2\) and \(\pi_j\) are both justified and \((\pi_1, \pi_j) \in \succeq^{G}\); or
3. \(\pi_1, \pi_2\) and \(\pi_j\) are both justified and \((\pi_1, \pi_j) \in \succeq^{G}\) but \((\pi_j, \pi_1) \in \succeq^{N}\).

Plan \(\pi_1\) is as good as \(\pi_j\) denoted \((\pi_1, \pi_j) \in \sim^{\pi}\) iff \((\pi_1, \pi_j) \not\in \succeq^{\pi}\) and \((\pi_j, \pi_1) \not\in \succeq^{\pi}\).

The relation \(>_{\pi}\) is irreflexive, asymmetric and transitive, while \(\sim_{\pi}\) is an equivalence relation on \(\Pi\).

**Definition 17** (Equivalence Classes). Given \(\pi \in \Pi\), let \([\pi]\) denote the equivalence class to which \(\pi\) belongs. \(\{[\pi],[\pi]\} \in \succeq\) iff \((\pi_1, \pi_j) \in \succeq^{G}\) or \((\pi_1, \pi_j) \in \sim^{\pi}\).

**Definition 18** (Best Plan(s)). Plan \(\pi_i\) is (one of) the best plan(s) for the agent to execute iff
- \(\pi_i\) is justified, and
- \(\exists \pi_j\) such that \(([\pi_j],[\pi_j]) \in \succeq^{\pi}\).

**Example 6**. Assume an agent with three goals \(strike, submission\) (c.f., Example 1), and certificate =
Theorem 1 (Extension Consistency). Suppose the conclusions of extension $E$ are inconsistent, i.e., there are arguments $Arg_\alpha, Arg_\beta \in E$ such that:

1. $Arg_\alpha$’s conclusion requires executing plan $\pi$ and $Arg_\beta$’s conclusion requires satisfying goal $g$ with norm $n$, while $g$ is not satisfied/ violated in $\pi$. Thus, $Arg_\beta$ defeats $Arg_\alpha$; $E$ is not conflict-free and cannot be an extension.

2. $Arg_\alpha$’s conclusion requires satisfying goal $g$ with norm $n$ and $Arg_\beta$’s conclusion requires satisfying goal $g$ with norm $n'$, while $g/n$ and $g/n'$ are inconsistent. Thus, $Arg_\alpha$ attacks $Arg_\beta$ and vice versa. Due to the preferences, at least one of these attacks is identified as defeat and therefore $E$ is not conflict-free and not an extension.

Theorem 2 (Indirect Consistency). The closure under strict rules of the conclusions of any extension is consistent.

Proof. Follows immediately from lack of strict rules.

Theorem 3 (Extension Composition). If a plan argument is labelled in by preferred labelling $L$, the arguments representing all the goals that it does not satisfy and norms it violates are labelled out by $L$ and vice versa:

$$Arg_\pi \in in(L) \Rightarrow Arg_\pi \in in(G) \cup Arg_n \in N_{out}(n) \subseteq out(L).$$

Proof. Every preferred labelling is a complete labelling. An argument is labelled in by a complete labelling iff all its attackers are labelled out. Therefore, a plan argument is labelled in by a preferred labelling iff all its attackers, namely the arguments for goals that it does not satisfy and norms that it violates, are labelled out by that labelling.

Theorem 4. If a plan argument is labelled in by preferred labelling $L$, the arguments representing all the goals that it satisfies and norms it complies with are also labelled in:

$$Arg_\pi \in in(L) \Rightarrow Arg_\pi \in in(G) \cup Arg_n \in N_{com}(n) \subseteq in(L).$$

Proof. Since $Arg_\pi \in in(L)$, from Property 4 we know that $Arg_\pi \in in(G) \cup Arg_n \in N_{com}(n) \subseteq out(L)$. We also know from the definition of a plan that $Arg_\pi \in in(G) \cup Arg_n \in N_{com}(n)$ is conflict free. Since all possible attackers of $Arg_\pi \in in(G) \cup Arg_n \in N_{com}(n)$ belong to $Arg_\pi \in in(G) \cup Arg_n \in N_{com}(n)$ and $Arg_\pi \in in(G) \cup Arg_n \in N_{com}(n)$ are all labelled out, we conclude that $Arg_\pi \in in(L)$.

Note that from $Arg_\pi \in in(L)$ one cannot conclude that $Arg_\pi \in in(L)$, as there might be justified goals or norms not satisfied or complied with in the plan.

Theorem 5. There is no more than one preferred labelling in which $Arg_\pi \in in(L)$.

Proof. From Properties 4 and 5 we know that if $Arg_\pi \in in(L)$ then $Arg_\pi \in in(G) \cup Arg_n \in N_{com}(n) \subseteq out(L)$ and $Arg_\pi \in in(G) \cup Arg_n \in N_{com}(n) \subseteq in(L)$. Since every preferred labelling is a complete labelling and the following property holds for complete labellings: if $out(L_{cmp1}) = out(L_{cmp2})$ then $L_{cmp1} = L_{cmp2}$, we conclude that there is no more than one preferred labelling in which $Arg_\pi \in in(L)$.

Theorem 6. $L$ is a stable labelling.

Proof. Every preferred labelling is a complete labelling. An argument is labelled in by a complete labelling iff all its attackers are labelled out. Therefore, a plan argument is labelled in by a preferred labelling iff all its attackers, namely the arguments for goals that it does not satisfy and norms that it violates, are labelled out by that labelling.

Theorem 7. $L$ is a stable labelling.

Proof. In Property 4 we showed that if $Arg_\pi \in in(L)$ then $Arg_\pi \in in(L) \Rightarrow Arg_\pi \in in(G) \cup Arg_n \in N_{com}(n) \subseteq out(L)$ and $Arg_\pi \in in(G) \cup Arg_n \in N_{com}(n) \subseteq in(L)$, which makes the undec($L$) = $\emptyset$. A preferred labelling with undec($L$) = $\emptyset$ is a stable labelling. Therefore, $L$ is a stable labelling.

Theorem 8. Let $\succsim^gn$ be a total preorder on G∪N and therefore $\succsim^gn$ be a total preorder on goal and norm arguments. If $Arg_\pi \in in(L)$, and the set of arguments for the most preferred goals and norms, Pref($Arg$), is conflict free, all arguments belong to Pref($Arg$) are labelled in by $L$.

Proof. Elements of set Pref($Arg$) cannot be defeated, since the set is conflict-free and the remaining arguments belong to $Arg \setminus Pref(A)$. The latter cannot defeat elements of Pref($Arg$), because this would imply an attack from a less preferred argument to a more preferred one has resulted in a defeat, which is contrary to assumption. Assume that $\exists Arg_\alpha \in Pref(A)$ such that $Arg_\alpha \not\in in(L)$. If $\exists Arg_\beta \in in(L)$ s.t. ($Arg_\alpha, Arg_\beta$) $\in$ Def then $Arg_\alpha$ should have been labelled in by $L$ otherwise it is contrary to the assumption of maximality of preferred labellings. If $\exists Arg_\beta \in in(L)$ s.t. ($Arg_\alpha, Arg_\beta$) $\in$ Def then $\exists Arg_\alpha \in in(L)$ s.t. ($Arg_\alpha, Arg_\alpha$) $\in$ Def, which is contradictory to

\{course_fee_paid, theory_test_done, interviewed\} and four norms $n_1, n_2, n_3$, and $n_4$ (c.f., Examples 2, 3, 4, and 5).
the fact that \( Arg_s \) cannot be defeated. Therefore, all elements of \( \text{Pref}(Arg) \) are labelled in by \( \text{in}(\mathcal{L}) \).

4 Explaining the Justifiability of the Best Plan

In this section we exploit an existing dialogue for preferred semantics known as Socratic Discussion [Caminada et al., 2014a] to provide an explanation for the justifiability of the best plan(s). Deciding if an argument is in at least one preferred extension amounts to deciding if it is in at least one admissible extension (i.e. it is labelled in by at least one admissible labelling). In an admissible labelling if an argument is labelled in, all its attackers are labelled out, and if an argument is labelled out, it has an attacker that is labelled in.

Definition 19 (Socratic Discussion [Caminada et al., 2014a]). Let \( AF = \langle Arg, Def \rangle \). The sequence of moves \( \{\Delta_1, \Delta_2, \ldots, \Delta_n\} \) \((n \geq 1)\) is a Socratic discussion iff: (i) each odd move (M-move) is an argument labelled in; (ii) each even move (S-move) is an argument labelled out; (iii) each argument moved by \( S \) attacks an argument moved by \( M \) earlier in the dialogue; (iv) each argument moved by \( M \) attacks an argument moved by \( S \) in the previous step; (v) S-moves cannot be repeated. Player \( S \) wins the discussion if there is an M-move and an S-move containing the same argument. Otherwise, the winner is the player that makes the last move.

Given that the agent’s best plans(s) \( \pi \) is labelled in by at least one preferred labelling, player \( M \) is guaranteed a winning strategy in a Socratic discussion with \( \Delta_1 = \text{in}(Arg_P) \). The even moves in the rest of dialogue are arguments labelled out, which according to Property 4 are goals not satisfied or norms violated in \( \pi \). On the other hand, the rest of odd moves in the dialogue are arguments labelled in, which according to Property 5 are goals satisfied or norms complied with in \( \pi \). Since each odd move attacks the even move in the previous step, during a dialogue the agent is able to dialectically explain why it did not satisfy a goal or violate a norm, which are the two causes of attacks on plan proposals.

Example 7. This example shows a Socratic discussion \( \Delta = \langle \text{in}(Arg_{x4}), \text{out}(Arg_{x2}), \text{in}(Arg_{s1}), \text{out}(Arg_{n3}), \text{in}(Arg_{n1}) \rangle \) for plan \( \pi_4 \).
- M: Plan \( \pi_4 \) is (one of) the best plan(s) and is justifiable.
- S: Why does the plan not satisfy goal submission?
- M: As it satisfies goal strike, attacking goal submission.
- S: Why does the plan violate norm \( n_1 \)?
- M: Because the plan satisfies norm \( n_4 \) that attacks norm \( n_1 \).

5 Related Work

One of the most well-known scheme-based approach in practical reasoning is [Atkinson and Bench-Capon, 2007]. Recently, [Oren, 2013] has proposed a similar scheme-based approach for normative practical reasoning where arguments are constructed for a sequence of actions. Similar to the latter approach, we construct arguments for plans rather than actions. [Oren, 2013] assumes that conflicts within and between goals and norms are inferred from paths, rather than being obtained from the model. Thus, although it is possible to explain why one path is preferred over another, it is not possible to understand why a path does not satisfy a goal or violate a norm. In contrast, we explicitly consider why an agent does not satisfy a goal, or violate a norm. In addition, in this work the explanation of justifiability of why a plan is (one of) the best plan(s) for the agent to execute is formulated using a dialogue game for preferred semantics.

There are several applications where dialogue games are used for explanation purposes [Zhong et al., 2014; Fan and Toni, 2015; Caminada et al., 2014b]. In [Zhong et al., 2014] and [Fan and Toni, 2015] admissible dispute trees developed for Assumption-based Argumentation [Dung et al., 2009] are used to provide explanation for why a certain decision is better than another. In [Caminada et al., 2014b] a dialogical proof based on the grounded semantics [Caminada and Podałaszewski, 2012] is created to justify the actions executed in a plan. Despite the popularity of the preferred semantics, they have not been used for explanation in such contexts, and our work is the first to do so in the practical reasoning domain.

6 Conclusion and Future Work

In contrast to existing argument based practical reasoning approaches, we propose a framework that integrates both the reasoning and dialogical aspects of argumentation to perform normative practical reasoning. In doing so, we answer the question of how an agent should act in a normative environment under conflicting goals and norms. Moreover, our approach can generate explanation for agent behaviour using an argumentation-based dialogue.

In future work we will investigate temporal solutions to addressing goal-goal and goal-norm conflict, similar to how conflicts between norms are handled. We also intend to empirically evaluate the effectiveness of our explanations, determining how likely a human is to accept the recommendation of a system regarding the best plan(s).
References


