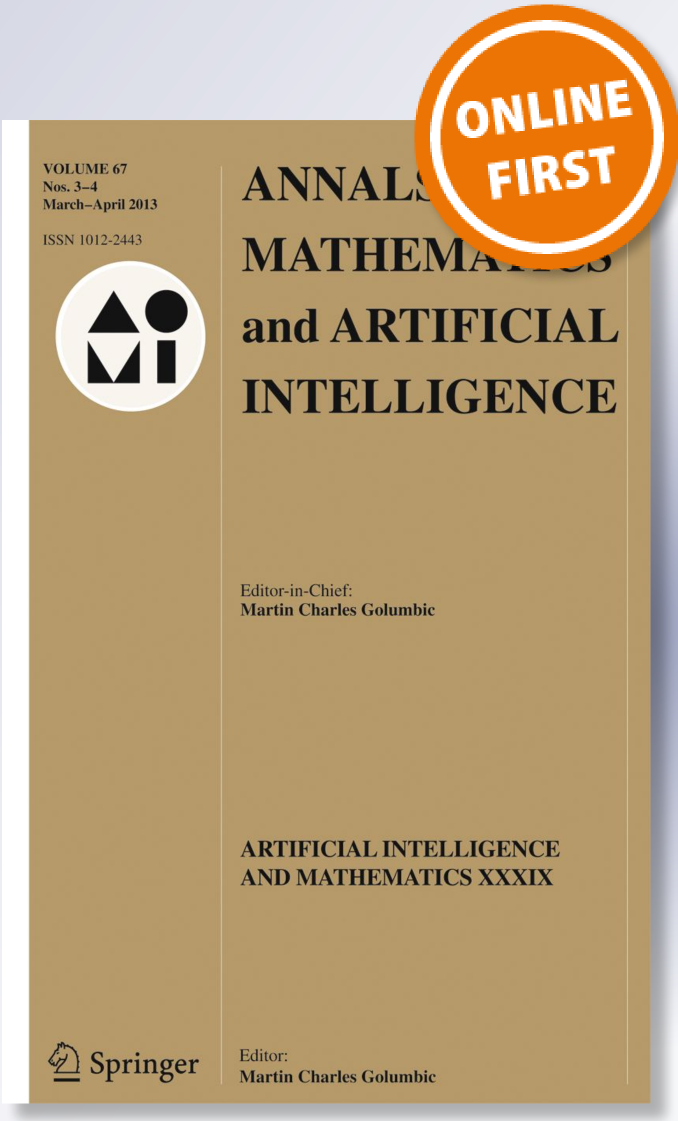
Practical reasoning using values: an argumentative approach based on a hierarchy of values

Juan C. L. Teze, Antoni Perelló-Moragues, Lluis Godo & Pablo Noriega

Annals of Mathematics and Artificial Intelligence

ISSN 1012-2443

Ann Math Artif Intell
DOI 10.1007/s10472-019-09660-8



Springer

Annals of Mathematics and Artificial Intelligence
https://doi.org/10.1007/s10472-019-09660-8

Practical reasoning using values: an argumentative approach based on a hierarchy of values

Juan C. L. Teze[1,2] · **Antoni Perelló-Moragues**[3] · **Lluis Godo**[3] · **Pablo Noriega**[3]

Published online: 06 August 2019

Abstract

Values are at the heart of human decision-making. They are used to decide whether something or some state of affairs is good or not, and they are also used to address the moral dilemma of the right thing to do under given circumstances. Both uses are present in several everyday situations, from the design of a public policy to the negotiation of employee benefit packages. Both uses of values are specially relevant when one intends to design or validate that artificial intelligent systems behave in a morally correct way. In real life, the choice of policy components or the agreed upon benefit package are processes that involve argumentation. Likewise, the design and deployment of value-driven artificial entities may be well served by embedding practical reasoning capabilities in these entities or using argumentation for their design and certification processes. In this paper, we propose a formal framework to support the choice of actions of a value-driven agent and arrange them into plans that reflect the agent's preferences. The framework is based on defeasible argumentation. It presumes that agent values are partially ordered in a hierarchy that is used to resolve conflicts between incommensurable values.

Keywords Practical reasoning · Defeasible argumentation · Hierarchy of values

✉ Juan C. L. Teze
jct@cs.uns.edu.ar

Antoni Perelló-Moragues
tperello@iiia.csic.es

Lluis Godo
godo@iiia.csic.es

Pablo Noriega
pablo@iiia.csic.es

1 Department of Computer Science and Engineering, Institute for Computer Science and Engineering (UNS-CONICdET), Universidad Nacional del Sur - Alem 1253, (8000) Bahía Blanca Bs. As., Argentina

2 Agents and Intelligent Systems Area, Faculty of Management Sciences, Universidad Nacional de Entre Ríos - Tavella 1424, (3200) Concordia E. R., Argentina

3 Artificial Intelligence Research Institute (IIIA-CSIC), Campus UAB - 08193 Bellaterra, Barcelona, Spain

Mathematics Subject Classification (2010) 91B69

1 Introduction

The point of policy-making is to "achieve a better state of affairs". Thus, the key responsibility of policy-makers is to choose the means that are likely to achieve such improvement. Policy means are usually a combination of actions that stakeholders are encouraged either to take or to avoid; together with the tools that help instrument them (norms, incentives, persuasive messages, and so on). The choice of means is part of the policy-design process which is a phase of a complex policy-making cycle that also involves negotiation, enactment, evaluation and back to (re)-design.

Added complexity resides in the fact that policy-making is relevant in socio-political contexts involving multiple stakeholders with conflicting interests. For instance, in the domain of agricultural water use, farmers may want to increase yield while environmentalists are concerned with the health of the ecosystem [24, 41]. Argumentation is a device that is often used to address these conflicts. Policy-makers may argue about the advantages of choosing specific means or to compensate, refine or discard them. These arguments usually involve values. Because values are directly involved in determining whether a state of affairs is "better" or not than other, and also in determining what might be the "right" action to take, under given circumstances.

Evidently, policy-making is but an instance of this dual use of values, and several other contexts that involve deciding about or from values involve some form of argumentation. In this paper, we use policy-making as a typical problem domain to illustrate how argumentation about values may be used. We limit our exploration to dialectical argumentation about a proposal of a set of value-aligned actions situated within an explicit context, and use the dialectical processes that may support selection of policy means to illustrate our ideas.

More precisely, in this paper we introduce an argumentation-based formalisation for agents that reason with values. We propose a formalism based in Defeasible Logic Programming [20] to analyse, on one hand, how the actions of an agent that has a goal in mind lead to transitions from one state of the world to another promoting certain values; and, on the other hand, how the set of actions that an agent may execute in order to achieve that goal can be identified. A distinctive feature of our proposal is to characterise value-driven agents that take into account hierarchical relations of values in order to choose action plans when several values are involved in the decision.

Our task responds to a relevant challenge: to foster moral behaviour in artificial systems. The rationale is that a distinctive aspect of our hyperconnected society is the existence of autonomous artificial entities that make decisions that not infrequently may have substantial consequences [19]. For example, autonomous vehicles, personal assistants, social coordination platforms, bot-based political campaigns, consumer profiling and micro-marketing. The fact that these decisions may be driven by questionable or no values at all is a concern that is being addressed from different approaches. One approach is to make explicit the standards that such systems should adhere to in order to guarantee that certain values are supported. This is the approach championed by the IEEE committee on ethical design of autonomous systems [37]. A second approach is to make moral awareness an essential part of system design, so that the resulting systems avoid pitfalls and support the intended values. This is the approach postulated by the Value-by-Design supporters [8, 33, 34]. A third approach, also described as the *value alignment problem* (VAP), is to condition autonomous entities to "do the right thing" and make sure that the autonomy they may exhibit complies with a set of values. A line of research within this approach is the development of artefacts that enable formal support for the design and enforcement of accountable value aligned behaviour [40]. Our paper belongs to this effort.

The focus on values, in our case, is cognitive. We are interested in the roles values may play in the decision-making processes of individuals (including artificial autonomous entities) and, more specifically in the use of practical reasoning to support value-driven action.

There is wide consensus about the importance of values to guide behaviour. A dramatic instance in favour of their significance is the editorial opinion about the Trump Administration published in the New York Times (09/05/2018) [4] where the anonymous author condemns the president's lack of values.[1]

Indeed, the values that people hold, together with other cognitive constructs like personality, needs and motivation, have a significant effect on their goals and preferences [27, 28, 30, 36]. Schwartz et al. [44] define values as "concepts or beliefs, about desirable *end states* or *behaviours*, that transcend specific situations, guide selection or evaluation of behaviour and events, and are ordered by relative importance". Moreover, Schwartz theory of values [42] —which has substantial empirical support— recognises six properties of values:

(i) Values are beliefs.
(i) Values refer to desirable goals.
(iii) Values transcend specific actions.
(iv) Values serve as standards or criteria.
(v) Values are ordered by importance.
(vi) The relative importance of multiple values guides action.

This understanding of values makes explicit the two main roles of values we require: to assess objective states of the world ("moral judgements"), and the relevance of values in individual's choices ("ethical dilemmas").

Using values in practice —to validate value-imbued autonomous systems or in actual policy-making negotiation— involves weighing their relevance and significance to choose (often through negotiation) among several courses of action. In our proposal, we address this issue, first, by holding a consequentialist view of values. In this view, one assumes that the meaning of values is found in their consequences [34, 45] and, therefore, one may say that an action is aligned with a value if the outcome of its performance improves the state of the world. Implicit in this assumption is that one may observe the relevant part of the world —through "indicators" (which are usually taken to be certain variables, parameters or facts in the representation of the world) that reflect that value— and compare any two states with respect to that value. Consequentialism only solves the problem of comparing two states of the world with respect to a specific value but in most cases choosing a course

[1] "...We [many senior officials of the Trump's administration] believe our first duty is to this country, and the president continues to act in a manner that is detrimental to the health of our republic. That is why many Trump appointees have vowed to do what we can to preserve our democratic institutions while thwarting Mr. Trump's more misguided impulses until he is out of office. *The root of the problem is the president's amorality*. Anyone who works with him knows he is not moored to any discernible first principles that guide his decision making".

of action involves taking several values into account, simultaneously. To solve this problem we propose to order values in a hierarchy. Although, in [43], Schwartz proposes a structure between value types as a way to deal with universal requirements for human action. As we shall discuss in Section 6, the use this type of structures associated with values has not been extensively studied in the practical reasoning context.[2]

The rest of the paper is structured as follows. The next two sections provide background, namely in Section 2 we introduce a running example and in Section 3 we give an overview of Defeasible Logic Programming (DeLP). In Section 4 we present our characterisation of agents that reason with values. Then, in Section 5 we discuss how our proposal handles the comparison of values, and study some criteria based on values for comparing plans. We discuss some related work in Section 6 and, finally, in Section 7, we present our conclusions and possible directions for future work.

2 A running example

Visualise a geopolitical region where several stakeholders, including politicians and farmers, are interested in making a "better use of water". Policy-makers have the task of translating their (political) principles into some goals or *ends* that reflect this improvement, and accompany these with some *means* that are conducive to those ends. Thus, in one case, policy-makers that hold *utilitarian values* would strive to obtain more revenue from every cubic meter of water used. In order to achieve this improvement they may propose the use of fertilisers or subsidising the adoption of modern irrigation systems. Ideally, farmers that adopt these means would have better crops and larger income and therefore a "better level of life". Alternatively, *ecologically motivated* stakeholders who are interested in the well-being of the community may claim that the utilitarian policy would lead to ecological disaster unless other means, like restrictions and incentives, are implemented. Thus, even when the goals of both parties may be compatible, if parties want to agree on a policy, they still need to agree on means that may lead to states of the world that are acceptable to both (maybe for different reasons). Consequently, in order to agree on a policy, parties may need to argue in favour or against specific actions that may lead to particular outcomes.

We will build on this simplistic situation to motivate and illustrate our proposal. More specifically, we look into a community of farmers who use traditional irrigation methods to grow their crops, drawing water from a river and their own wells. Thus, farmers may improve their use of water by making a more "productive" use of the amount of water; for instance, by changing to better paid crops, using fertilisers to improve yield or introducing drop irrigation or sprinklers to avoid evaporation and runaway losses. These changes may have direct effect on income and the general well-being of the community but come along with some expenses, financial risks and ecological effects. In this context policy-makers would consider taking or fostering some actions that balance the potential improvements with their undesirable effects.

[2] It should be noted that consequentialism is consistent with the notion that when several values are involved in a decision, these values may be made "commensurable". This means that one may combine their evaluations into a single "index" that is used to compare states of the world and prioritise actions and plans of actions, and the actual combination may be achieved through different aggregation models. In this paper, we do not assume commensurable values, in fact our proposal to use a value hierarchy is precisely a way of dealing with incommensurability. However, in the running example we suggest that although we deal only with three values, these are in fact the aggregation of several variables that may stand for other (closely associated) values.

Table 1 Facts involved in policy actions

Label	Meaning
hCP	High Crop Prices
lRZ	Low rainfall zone
lIT	Low investment in farmers training
mIF	Monthly (frequent) inspection of farms
dWZ	Dry weather zone
sFA	Supervised fertiliser application

To make matters simple, let us assume that policy-makers take three values into consideration: *economic development, environmental protection* and *workers well-being*. Let us also assume that these values may be objectively assessed through some indicators that reflect the state of the world. Such indicators may then be combined into *indexes* to decide whether the values are satisfactorily met at any point in time or not. For illustration purposes we combine the indicators of the state of the world into *three indexes* —one for each value– and use them to "score" the state of the world as satisfactory or not at any time. Namely, if we decide that a combination of several well-being indicators like salaries, housing facilities, transportation, education, consumer prices, etc. is high enough, then we claim that a satisfactory level of social well-being is met, and label that state as sH "employee salary is high"; otherwise, we label $\sim sH$ those states of the world where this index is not met. The index for satisfactory economic development –again combining several indicators into it— is labelled iH "farmers income is high enough". Finally, the index for environmental protection is reflected into the index gQ, "water quality is good". The salient consequence of this representation is that these three indexes allow us to reduce whatever state the world may be in as one of the eight possible combinations of the indexes.

As we mentioned above, policy makers try to reach those satisfactory value levels by introducing means that foster desired behaviour and thus the evolution of the state of the world towards those ends. Although there may be several types of means, in this illustration we reduce available means to actions. The point is that actions change the state of the world and may thus affect the level of satisfaction of each value. In particular we only consider three actions that policy-makers may introduce into the system: (i) to subsidise fertilisers (sF) that has a positive effect on economic development and workers well-being (because ideally it improves yield and therefore increases income and labour productivity); (ii) to modernise irrigation systems (mIS) that again contributes to economic development; and (iii) to control the use of fertilisers (cFU) whose aim is to improve environmental protection. These actions are not performed in a void, an action can only be taken if, on one hand, some pre-conditions are met and, on the other hand, if the action succeeds then it has effects on the state of the world. In other words, an action may happen only if certain facts hold in the world, and if it happens, different facts may hold.

Note that the "world" of the farmers community contains millions of facts, however for policy-making, there is only a part of the world that is "relevant", the one that is related to actions and events that may change the scoring of the values endorsed by the policy. In the rest of this paper we limit the notion of the *state of the world* to include only the set of facts that are involved in the pre-conditions and post-conditions of available actions. In our simplistic example, we make explicit the six facts in Table 1.

Figure 1 illustrates the scenario, in the previously mentioned policy domain, as a state transition system. In this example a policy-maker agent that starts in state $\{\sim iH, \sim gQ, \sim sH\}$

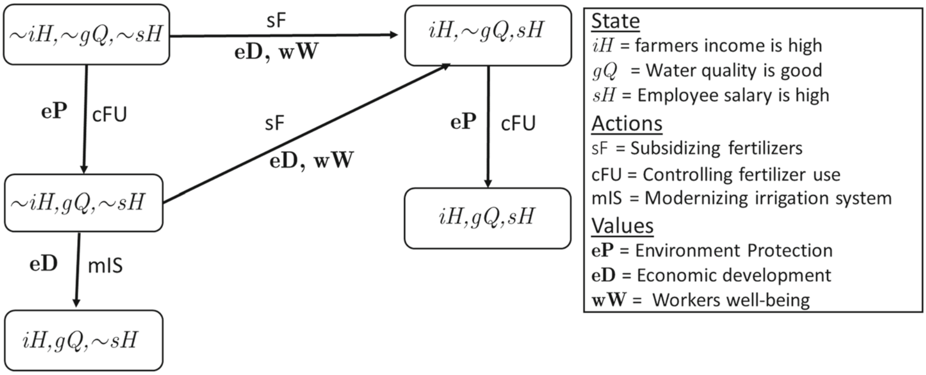


Fig. 1 Transitions from an initial state $\{\sim iH, \sim gQ, \sim sH\}$

chooses actions that lead to different states. Note that states are represented as boxes with (binary) state indexes, and arcs are labelled by actions and by the values that those actions promote.

In this paper, we will introduce an agent specification that describes what actions can be taken by the agent to reach a target state, using defeasible logic programming as a formalism to represent domain knowledge.

3 A refresher on defeasible logic programming

We will use Defeasible Logic Programming (DeLP) as the central component for integrating argumentation into our formalism introduced later; below we will provide an introduction to the elements of the framework. DeLP is a formalization that combines results of Logic Programming and Defeasible Argumentation that allows representing information declaratively using rules, and employing a defeasible inference mechanism based on argumentation for warranting the entailed conclusions.

A DeLP-program $\mathcal{P}$ is a pair (Π, Δ) where Π is a set of facts and strict rules and Δ is a set of defeasible rules.

Facts are ground literals representing atomic information or the negation of atomic information using the strong negation "$\sim$". *Strict Rules* represent non-defeasible information and are denoted $L_0 \leftarrow L_1, \ldots, L_n$, where L_0 is a ground literal and $\{L_i\}_{0<i\leq n}$ is a set of ground literals.

Defeasible Rules represent tentative information that may be used when nothing could be posed against it and are denoted $L_0 \prec L_1, \ldots, L_n$, where L_0 is a ground literal and $\{L_i\}_{0<i\leq n}$ is a set of ground literals. A defeasible rule *Head* $\prec$ *Body* expresses that *reasons to believe in the antecedent Body give reasons to believe in the consequent Head*. Following [20], DeLP ground rules can also be represented as schematic rules with variables; as usual in Logic Programming (see [26]), schematic variables are denoted with an initial uppercase letter.

Example 1 Continuing with the example illustrated in Fig. 1, let $\mathcal{K}_1 = (\Pi_1, \Delta_1)$ be a DeLP-program that represents information of application domain presented in the running example of Section 2:

$$\Pi_1 = \left\{ \begin{array}{l} hCP \\ \sim sH \\ \sim iH \\ lRZ \\ lIT \\ mIF \end{array} \right\} \Delta_1 = \left\{ \begin{array}{l} dWZ \prec lRZ \\ sFA \prec mIF, lRZ \\ \sim sFA \prec lIT \end{array} \right\} \tag{1}$$

Observe that the set Π_1 has six facts. These facts represent information about farmers and crops where they irrigate which can be used by a policy-maker agent to decide what policy action to perform.

The set Δ_1 contains three defeasible rules.

The first rule expresses that if farms are in low rainfall zones (lRZ), then there is a reason for establishing that they are in a dry weather zone (dWZ). Whereas the second rule expresses that there is a tentative reason for establishing that there is a supervised application of fertilizers (sFA) if there is a monthly technical inspection of farms (mIF) and farms are in low rainfall zones (lRZ). Finally, the last rule represents a reason against establishing that there is a supervised application of fertilizers ($\sim sFA$) if the investment in farmers' training is low (lIT).

From a DeLP-program $\mathcal{K}$, tentative information can be inferred; when it is possible to infer a literal L from $\mathcal{K}$ this is denoted $\mathcal{K} \mid\!\sim L$, and these inferences are called *defeasible derivations*. A defeasible derivation of a literal L from $\mathcal{K}$, is a finite sequence of ground literals $L_1, L_2, \ldots, L_n = L$, where each literal L_i is in the sequence because: (a) L_i is a fact in $\mathcal{K}$, or (b) there exists a rule R_i in $\mathcal{K}$ (strict or defeasible) with head L_i and body $B_1, B_2, \ldots, B_k$ and every literal of the body is an element L_j of the sequence appearing before L_i ($j < i$). We will say that L has a *strict derivation* from $\mathcal{K}$, denoted $\mathcal{K} \vdash L$, if either L is a fact or all the rules used for obtaining the sequence $L_1, L_2, \ldots, L_n$ are strict rules. For instance, from the DeLP-program $\mathcal{K}_1$ of Example 1 there are defeasible derivations for sFA and $\sim sFA$.

Strong negation may be used to represent contradictory knowledge; two literals are contradictory if they are complementary; that is, given a literal L, the complement with respect to the strong negation will be noted $\sim L$, i.e.,if L is a propositional variable p, then $\sim L = \sim p$ and if L is the negation of a propositional variable $\sim p$, then $\sim L = p$. Thus, a set of literals is contradictory iff it contains a pair of complementary literals. A set of facts and rules is contradictory if it derives a pair of contradictory literals. Actually, given a program (Π, Δ), $\Pi \cup \Delta$ can be contradictory (e.g.,both sFA and $\sim sFA$ can be derived from $\mathcal{K}_1$), but Π can not.

An *argument* for a literal L from (Π, Δ) is denoted $\langle \mathcal{A}, L \rangle$, where $\mathcal{A} \subseteq \Delta$ is a minimal and non-contradictory set, such that together with Π allows a defeasible derivation of L. Given a DeLP-argument $\langle \mathcal{A}, L \rangle$, L is called the *conclusion* of the argument, and sometimes for simplicity we will say that $\mathcal{A}$ is the argument that *supports* L. For instance, the following two arguments can be constructed from the program $\mathcal{K}_1$ of Example 1: $\langle \mathcal{B}, sFA \rangle$ where

$\mathcal{B} = \{sFA \prec mIF, lRZ\}$, and $\langle \mathcal{C}, \sim sFA \rangle$ where $\mathcal{C} = \{\sim sFA \prec lIT\}$. An argument $\langle \mathcal{A}, L \rangle$ is said to be a subargument of $\langle \mathcal{A}_1, L_1 \rangle$, if $\mathcal{A} \subseteq \mathcal{A}_1$.

Two literals L and L_1 *disagree* in the context of a program (Π, Δ), when the set $\Pi \cup \{L, L_1\}$ is contradictory. We say that the argument $\langle \mathcal{A}_1, L_1 \rangle$ *counterargues* or *attacks* $\langle \mathcal{A}_2, L_2 \rangle$ at literal L, if and only if there exists a subargument $\langle \mathcal{A}, L \rangle$ of $\langle \mathcal{A}_2, L_2 \rangle$ such that L and L_1 disagree. For instance, the two arguments introduced above: $\langle \mathcal{A}_3, sFA \rangle$ and $\langle \mathcal{A}_4, \sim sFA \rangle$, counterargue or attack each other because literals they support (sFA and $\sim sFA$) disagree. Note that in DeLP an argument can attack the conclusion or an inner point of other arguments.

To decide which argument prevails in an attack situation, the arguments are compared using an *argument comparison criterion* that establishes a preference relation between the arguments involved; this criterion is a modular part of system and can be replaced. We will denote that an argument $\langle \mathcal{A}_1, L_1 \rangle$ is preferred to $\langle \mathcal{A}_2, L_2 \rangle$, as $\langle \mathcal{A}_1, L_1 \rangle \rhd \langle \mathcal{A}_2, L_2 \rangle$. In DeLP, an argument $\langle \mathcal{A}_1, L_1 \rangle$ is a defeater for an argument $\langle \mathcal{A}_2, L_2 \rangle$, iff there exists a subargument $\langle \mathcal{A}, L \rangle$ of $\langle \mathcal{A}_2, L_2 \rangle$ such that $\langle \mathcal{A}_1, L_1 \rangle$ counterargues $\langle \mathcal{A}_2, L_2 \rangle$ at L and it holds that:

1. $\langle \mathcal{A}_1, L_1 \rangle \rhd \langle \mathcal{A}, L \rangle$ (proper defeater), or
2. $\langle \mathcal{A}_1, L_1 \rangle \not\rhd \langle \mathcal{A}, L \rangle$ and $\langle \mathcal{A}, L \rangle \not\rhd \langle \mathcal{A}_1, L_1 \rangle$ (blocking defeater).

For instance, if $\rhd$ stands for the specificity criterion [20], then it is clear that, regarding the two arguments introduced above, we have $\langle \mathcal{B}, sFA \rangle \rhd \langle \mathcal{C}, \sim sFA \rangle$. Informally, an argument $\langle \mathcal{A}_1, L_1 \rangle$ is preferred to another $\langle \mathcal{A}_2, L_2 \rangle$ under the specificity criterion if $\langle \mathcal{A}_1, L_1 \rangle$ uses more information than $\langle \mathcal{A}_2, L_1 \rangle$ (i.e. $\mathcal{A}_1$ contains more literals in the premises of its rules than $\mathcal{A}_2$), or supports its conclusion more directly than $\langle \mathcal{A}_2, L_1 \rangle$ (i.e., $\mathcal{A}_1$ has less rules than $\mathcal{A}_2$).

Given a DeLP-program $\mathcal{K} = (\Pi, \Delta)$ from which all arguments are built, to establish whether an argument $\langle \mathcal{A}, L \rangle$ is an undefeated argument, all the defeaters for $\langle \mathcal{A}, L \rangle$ are considered, and the defeaters for these defeaters, and so on. As each defeater could in turn be defeated, a sequence of arguments called *argumentation line* $\Lambda = [\langle \mathcal{A}_0, L_0 \rangle, \langle \mathcal{A}_1, L_1 \rangle, \langle \mathcal{A}_2, L_2 \rangle, \ldots, \langle \mathcal{A}_n, L_n \rangle]$ arises where each argument (except the first one) is a defeater of its predecessor. In Λ, in regard to the initial argument (position 0), the arguments in an even position plays a role as a supporting argument and the ones in an odd position act as an interfering argument. In DeLP, an argumentation line Λ is considered acceptable if the following conditions hold: (1) Λ is finite, (2) the set of supporting arguments in Λ is non contradictory and the set of interfering arguments in Λ is non contradictory, (3) no argument $\langle \mathcal{A}_j, L_j \rangle$ in Λ is a subargument of an argument $\langle \mathcal{A}_i, L_i \rangle$ in Λ, $i < j$, and (4) every blocking defeater $\langle \mathcal{A}_i, L_i \rangle$ in Λ is defeated by a proper defeater $\langle \mathcal{A}_{i+1}, L_{i+1} \rangle$ in Λ. These four constraints are necessary to avoid fallacious situations, but they are not independent; in particular, finiteness could be obtained from the others (see [20] for a complete discussion).

Since there might be more than one defeater for a given argument, a set of argumentation lines could originate from one argument; this leads to a tree structure called *dialectical tree*. In a dialectical tree, each path from the root to a leaf corresponds to a different acceptable argumentation line. Also, note that every node (except the root) is a defeater of its parent, and the leaves of the tree are undefeated arguments.

The *marking of a dialectical tree* ([20]) is a process that assigns every node the mark of defeated ("D") or the mark of undefeated ("U") as follows. Leaf nodes are marked as "U", an inner node is marked as "D" if it has at least a child marked as "U", while an inner node is marked as "U" if all its children are marked as "D". Thus, a ground literal Q is said that it is *warranted* from the program $\mathcal{K}$, if there exists an argument $\langle \mathcal{A}, L \rangle$ for Q from $\mathcal{K}$ and the root of its associated dialectical tree is marked as "U", i.e.,the argument $\langle \mathcal{A}, L \rangle$ cannot be defeated.

Given a DeLP-program $\mathcal{K}$, and a ground literal Q, a DeLP-interpreter will return YES, if the literal Q is warranted from $\mathcal{K}$; NO, if the complement of Q is warranted from $\mathcal{K}$; UNDECIDED, if neither Q nor its complement are warranted from $\mathcal{K}$; or UNKNOWN, if Q is not in the signature of $\mathcal{K}$.

4 Value-based agents

In this section, we introduce a formalism that combines actions, values, and argumentation to model policy-making agents. We focus on both the formal description of a value-based agent specification and the necessary mechanisms to integrate defeasible argumentation into agent reasoning process. In our proposed approach, agents look for achieving an objective state through a set of actions applicable in a given specific situation. The agent's knowledge will be represented by a DeLP-program, therefore an agent will be able to perform defeasible reasoning over this knowledge in order to determine what actions can be applied.

In the rest of the paper we will assume the following components in our formalism:

(i) A finite set of propositional *state variables* $Var = Var_s \cup Var_c$.[3] Literals Lit built from Var will be used to describe what is known about the state of the world.
(ii) A finite set $\mathcal{V}$ denoting the agent's *values*.
(iii) A finite set $\mathcal{A}$ denoting *actions*.

We will further assume that $Var_s \cup Var_c$, $\mathcal{V}$ and $\mathcal{A}$ are pairwise disjoint sets.

We assume that, at any moment, the (incomplete) agent's representation of the state of the world is by means of a set of non-contradictory literals. We will refer to it as *world state*.

Definition 1 (State) A state for an agent is a consistent set of literals $\Psi \subseteq Lit$.

In our formalism, literals in a given state will be represented as facts in the DeLP-program representing the agent's knowledge.

Example 2 Continuing with the scenario depicted in Fig. 1, we fix for this and subsequent examples the following sets of state variables and actions:[4]

$$Var = \{iH, gQ, sH, hCP, lRZ, lIT, mIF, dWZ, sFA\}$$

$$\mathcal{A} = \{\mathsf{sF}, \mathsf{sCFU}, \mathsf{mIS}\}.$$

As a matter of example, a possible description of the current state of the world by the policy-maker agent is given by the following consistent set of literals from Var:

$$\Psi_2 = \{hCP, \sim sH, \sim iH, lRZ, lIT, mIF\}$$

Note that this is an incomplete description, as Ψ_2 nothing tells about the variables gQ, lIT and sFA.

[3] Actually, one could distinguish two subsets of variables of Var: the indexes or variables that relevant for the agent as for policy making purposes (Var_s), and other variables that provide additional descriptions of the world and that can be used to determine if an action is applicable or not (Var_c).

[4] Following the above distinction among variables, in this running example $Var_s = \{iH, gQ, sH\}$ would be the relevant variables for the agent as shown in Fig. 1.

Following this idea, we introduce below the concept of transition action A promoting values V.

Definition 2 (Transition action) A *transition action* is a tuple $\mathsf{A} = \langle \mathsf{V}, \mathsf{Pos}, \mathsf{Pre} \rangle$, also denoted $(\mathsf{V}, \mathsf{Pos} \leftarrow\!\!\prec \mathsf{Pre} : \mathsf{A})$, where $\mathsf{Pre} \subseteq Lit$ is a set of literals representing preconditions for executing A, $\mathsf{Pos} \subseteq Lit$ is a consistent set of literals representing the consequences or effects of executing the action $\mathsf{A} \in \mathcal{A}$, and $\mathsf{V} \subseteq \mathcal{V}$ is the set of values promoted by A.

The intended meaning of an action A is thus "*if all literals of* Pre *representing the preconditions of* A *are warranted at a given state* Ψ*, then after executing* A *the literals of* Pos *will hold true in the new resulting state* Ψ'". Later, we will define when an action is applicable and the result of its execution.

Example 3 Consider the application domain presented in the example of Fig. 1. The agent has three actions $\Gamma_3 = \{\mathsf{sF}, \mathsf{cFU}, \mathsf{mIS}\}$. Roughly speaking, action cFU transforms $\sim gQ$-states into gQ-states, while actions sF and mIS transform $\sim iH$-states into iH-states, but action mIS can only applied on gQ-states. Their full specification, with preconditions, postconditions and values promoted, is as follows:

$$\Gamma_3 = \left\{ \begin{array}{l} \{\mathbf{eD,wW}\}, \{iH, sH, \sim gQ\} \leftarrow\!\!\prec \{\sim iH, \sim sH, hCP\} : \mathsf{sF} \\ \{\mathbf{eP}\}, \{gQ\} \leftarrow\!\!\prec \{\sim gQ, sFA\} : \mathsf{cFU} \\ \{\mathbf{eD}\}, \{iH\} \leftarrow\!\!\prec \{\sim iH, dWZ, gQ\} : \mathsf{mIS} \end{array} \right\}$$

An action promoting a particular value is performed to achieve some new state. That state will contain certain desirable aspects

being the promoted value the reason why these aspects are currently desirable. Note that the same state could be achieved while promoting several values in different circumstances. In the example above: if the policy maker is productivist, then it may modernize the irrigation system (mIS) for increasing farmers' income while promoting both well-being and economic development when farmers' income and the employees salary are not high ($\sim iH$ and $\sim sH$) if the water quality is good (gQ). Nevertheless, promoting only economic development may be an alternative by subsidizing fertlizers (sF) when the water quality is not good ($\sim gQ$). Similarly a same action may be used to achieve different states when applied in different circumstances.

Obviously, an action $\langle \mathsf{V}, \mathsf{Pos}, \mathsf{Pre} \rangle$ will be ready to be executed as soon as its preconditions are satisfied. In our case, using the DeLP argumentative framework for knowledge representation, the agent's knowledge base will be represented by a set Δ of DeLP defeasible rules. Then, checking the satisfaction of the preconditions amounts to checking whether the literals in Pre can be warranted by the DeLP program (Δ, Ψ), consisting of the agents' defeasible rules together with the facts holding in the current world state Ψ. We will denote by $\mathsf{warrL}(\Delta, \Psi)$ the set of warranted literals by the DeLP program (Δ, Ψ).

Definition 3 (Applicable transition action) Let Δ a set of defeasible rules representing an agent's knowledge base, Ψ a state and Γ a set of transition actions available to this agent. A transition action $\langle \mathsf{V}, \mathsf{Pos}, \mathsf{Pre} \rangle \in \Gamma$ is applicable at Ψ if for each precondition $P_i \in \mathsf{Pre}$, it holds that $P_i \in \mathsf{warrL}(\Psi, \Delta)$.

After an applicable action A is executed, the state Ψ is updated by taking each effect of Pos as a new fact (and overriding the old contradictory ones). We will denote by Ψ^{A} the new state resulting from executing the action A at Ψ.

Definition 4 (Resulting state) Let

Ψ be a state, Γ a set of available transition actions, and $\mathsf{A} = \langle \mathsf{V}, \mathsf{Pos}, \mathsf{Pre} \rangle \in \Gamma$ a transition action applicable in Ψ. The resulting state after executing A at Ψ is defined as follows:

$$\Psi^{\mathsf{A}} = (\Psi \backslash {\sim}\mathsf{Pos}) \cup \mathsf{Pos},$$

where ${\sim}\mathsf{Pos}$ is the set of complemented literals of Pos.[5]

It is very easy to check that, by definition, the resulting state Ψ^{A} keeps indeed being a consistent set of literals

Example 4 Consider the set of actions Γ_3 specified in Example 3, the knowledge base represented by the defeasible rules in Δ_1 introduced in Example 1,

and state Ψ_2 presented in Example 2 where

$$\Psi_2 = \{hCP, {\sim}sH, {\sim}iH, lRZ, lIT, mIF\}.$$

Consider now the action sF in Γ_3. The action sF is applicable in Ψ_2 because

${\sim}iH$, ${\sim}sH$ and hCP are facts in Ψ_2, and thus they clearly belong to $\mathsf{warrL}(\Psi_2, \Delta_1)$. The resulting state of executing sF at Ψ_2 is the set

$$(\Psi_2)^{\mathsf{sF}} = \{hCP, sH, iH, {\sim}gQ, lRZ, lIT, mIF\}.$$

Given that the execution of a transition action results in a new state, another action could apply over this state and so on. This leads to the notion of an applicable sequence of actions or action plan, which is defined as follows.

Definition 5 (Applicable sequence / Action plan) Let Δ be a set of defeasible rules representing the agent's knowledge base,

Ψ a state and Γ the set of available transition actions for the agent. An action plan $\mathsf{P} = [\mathsf{A}_1, \mathsf{A}_2, \ldots, \mathsf{A}_n]$ is a non-empty sequence of transition actions, where each $\mathsf{A}_i \in \Gamma$. Then:

– The action plan P is applicable at Ψ if, for every $1 \leq i \leq n$, action A_i is applicable at $\Psi^{[\mathsf{A}_1,\ldots,\mathsf{A}_{i-1}]}$, where the latter is a shorthand for $(\cdots(\Psi^{\mathsf{A}_1})\cdots)^{\mathsf{A}_{i-1}}$.
– The resulting state of executing an applicable action plan $\mathsf{P} = [\mathsf{A}_1, \mathsf{A}_2, \ldots, \mathsf{A}_n]$ at Ψ is $\Psi^{[\mathsf{A}_1,\mathsf{A}_2,\ldots,\mathsf{A}_n]}$.

Again, it is immediate to check that indeed the resulting state $\Psi^{[\mathsf{A}_1,\mathsf{A}_2,\ldots,\mathsf{A}_n]}$ of executing an applicable sequence of actions $\mathsf{P} = [\mathsf{A}_1, \mathsf{A}_2, \ldots, \mathsf{A}_n]$ consists of a new consistent set of literals.

[5] That is, for every $p \in Var$, ${\sim}p \in {\sim}\mathsf{Pos}$ iff $p \in \mathsf{Pos}$, and $p \in {\sim}\mathsf{Pos}$ iff ${\sim}p \in \mathsf{Pos}$.

Example 5 Consider state Ψ_2 introduced in Example 2, and the set of transition actions Γ_3 introduced in Example 3. Different sequence of actions from Γ_3 can be analysed in order to establish whether they are applicable. Namely:

- Consider the sequence of actions $\mathsf{P}_1 = [\mathsf{cFU}]$. The action cFU it is not applicable in Ψ_2 since $\sim gQ \notin \Psi_2$.
- Consider the sequence $\mathsf{P}_2 = [\mathsf{sF}, \mathsf{cFU}]$. The action sF is applicable at Ψ_2 (since all its preconditions $\sim iH$, $\sim sH$, hCP belong to Ψ_2). The resulting state of executing sF is

$$(\Psi_2)^{\mathsf{sF}} = \{hCP, sH, iH, \sim gQ, lRZ, lIT, mIF\}.$$

  The next action cFU is also applicable since now $\sim gQ \in (\Psi_2)^{\mathsf{sF}}$ and sFA, the other precondition, is a warranted literal in the program $((\Psi_2)^{\mathsf{sF}}, \Delta_1)$. Indeed, recall from Section 3, that under the specificity criterion the argument $\langle\{sFA \prec mIF, lRZ\}, sFA\rangle$ is preferred to the argument $\langle\{\sim sFA \prec lIT\}, \sim sFA\rangle$. Therefore, P_2 is an applicable action plan at Ψ_2.

Values are at the core of decision making criteria, motivations, preferences, and attitudes [31]. Values play a substantial role in an agent's decision making process. For instance, in a policy making domain, it is clear that the adoption of a policy depends on the importance order that the agent has over her own values. In the context of our running example, one may think that a productivist policy maker will prioritize actions that promote an economic development value, whereas an environmentalist policy maker will probably give priority to actions that promote an environment protection value.

When an agent reasons about what action to choose next, it may well be the case that different available actions promote different values. In such a case, a conflict may arise and we need a way to deal with it. In this context, value systems can be used to resolve this situation. In our approach, values will be associated to agent's actions and serve as criteria to guide the selection and evaluation of the actions.

In [38], Rokeach describes the notion of value system as follows: "*A value system is a learned organization of principles and rules to help one choose between alternatives, resolve conflicts, and make decisions*". Loosely inspired by [38], we propose to simply define a value system as a set of values (or labels) together with an importance preference relation over them. Indeed, this will allow for an interesting mechanism to compare action plans based on the values the plans promote and the agent's preferences over those values.

In other words, we will adopt the following definition of value system.

Definition 6 (Value system) A value system is a partially ordered set $(\mathcal{V}, \succeq)$, where $\mathcal{V}$ is a set of labels representing an agent's set of values and $\succeq \subseteq \mathcal{V} \times \mathcal{V}$ is a partial order, that is, $\succeq$ fulfills the following properties:[6]

- reflexive: $V \succeq V$, for all $V \in \mathcal{V}$
- anti-symmetric: if $V \succeq U$ and $U \succeq V$ then $V = U$
- transitive: if $V \succeq U$ and $U \succeq W$ then $U \succeq W$

When $V_i \succeq V_j$, we will say that the value V_i is at least as preferred as the value V_j. Moreover:

[6]As usual, we will write $V \succeq U$ instead of $(V, U) \in \succeq$.

- when $V_i \succeq V_j$ and $V_j \neq V_i$, we will say that V_i is strictly preferred to V_j, and we will write $V_i \succ V_j$.
- when $V_i \not\succeq V_j$ and $V_j \not\succeq V_i$, we will say that V_i and V_j are incomparable.

As it is well-known, the strict counterpart $\succ$ of a partial order $\succeq$ as defined above is irreflexive ($V \not\succ V$), asymmetric ($V_i \succ V_j$ implies $V_j \not\succ V_i$) and transitive. Conversely, from the strict partial order $\succ$, one can recover the original (non-strict) partial order by taking the reflexive closure of $\succ$, given by: $V \succeq U$ if either $V \succ U$ or $V = U$.

Example 6 Considering the application domain presented in the example of Fig. 1, the set $\mathcal{V}_6 = \{\mathbf{eP}, \mathbf{eD}, \mathbf{wW}, \mathbf{sW}, \mathbf{w}\}$ can be seen as a set of possible values for a policy-maker, where '**eP**' stands for *environment protection*, '**eD**' for *economic development*, '**wW**' for *workers well-being*, '**sW**' for *social well-being*, and '**w**' for *general well-being*. Assume the policy-maker agent only expresses her (strict) preference for promoting the environment protection over the economic development, i.e. $\mathbf{eP} \succ_6 \mathbf{eD}$ and the rest of values are assumed to be incomparable. In such a case, $\{\mathcal{V}_6, \succeq_6\}$ would be the value system for the policy-maker, where $\succeq_6$ is the reflexive closure of $\succ_6$.

The main aim of an agent is to find a list of actions that leads to a goal state from current situation state. To accomplish this, a value-driven agent will be characterized by: a knowledge base consisting on by a set of defeasible domain rules, an initial state, a set of actions, and a set of goals. Besides these components, an agent will be associated to a particular value system that guides its behaviour on selection of actions. We will capture this notion in the following definition.

Definition 7 (Value-driven agent specification) An agent specification under a value system $\boldsymbol{v} = (\mathcal{V}, \succeq)$ is a tuple $T_{\boldsymbol{v}} = (\Psi, \Delta, \Gamma, G)$, where:

- Ψ is a consistent set of literals representing the agent's current state.
- Δ is a set of defeasible rules representing the agent's knowledge base.
- $\Gamma = \{\mathsf{A}_1, \mathsf{A}_2, \ldots, \mathsf{A}_n\}$ is a set of transition actions available to the agent such that for every value V involved in $\mathsf{A}_i \in \Gamma$, $V \in \mathcal{V}$.
- G a consistent set of literals representing agent's goals.

From the agent's knowledge base, and using the set of actions available to this agent it is possible to determine desirable future states intended to be achieved by the agent. We will call *achievable goals* to those goals that can be warranted after the execution of an applicable sequence of actions in the current situation.

Definition 8 (Achievable goal) Let $T_\upsilon = (\Psi, \Delta, \Gamma, G)$ be an agent's specification and let $L \in G$. We will say that L is an achievable goal under the specification T_υ iff there exists a plan $\mathsf{P} = [\mathsf{A}_1, \mathsf{A}_2, \ldots, \mathsf{A}_n]$, where $\mathsf{A}_i \in \Gamma$ for each $1 \leq i \leq n$, such that:

- P is applicable at Ψ, and
- L is warranted by the program $(\Psi^{[\mathsf{A}_1, \mathsf{A}_2, \ldots, \mathsf{A}_n]}, \Delta)$, that is, $L \in \mathsf{warrL}(\Psi^{[\mathsf{A}_1, \mathsf{A}_2, \ldots, \mathsf{A}_n]}, \Delta)$.

In such a case, we will say that P is a plan for L.

Example 7 Consider the value system $\boldsymbol{v} = \{\mathcal{V}_6, \succ_6\}$ introduced in Example 6, $G_7 = \{iH\}$ and $T_{\boldsymbol{v}} = (\Psi_2, \Delta_1, \Gamma_3, G_7)$ the specification for a policy-maker. Consider the sequence of actions $\mathsf{P}_2 = [\mathsf{sF}, \mathsf{cFU}]$ from Example 5. We have shown there that P_2 is a plan applicable at Ψ_2, and since iH is a postcondition of the action cFU, P_2 is a plan to achieve iH.

As discussed in Section 1, in policy-making process, conflicts between actions are often resolved using ordering on values considered. Accordingly, given two different action plans P_1 and P_2 for a same current goal L, in order to decide which one prevails, the plans will be compared using a strategy based on the agent's preferences on the values.

From now on, we will abstract away from concrete comparison strategies, assuming there exists a comparison strategy among plans based on the agent's value system, that we will denote $>$, i.e., we will denote that P_1 is preferred to P_2 as $\mathsf{P}_1 > \mathsf{P}_2$.

The study of different strategies for comparing plans is one of our aims for the next section.

Definition 9 (Preferred plans) Let $T_{\boldsymbol{v}} = (\Psi_2, \Delta_1, \Gamma_3, G_7)$ be a specification of an agent under a value system $\boldsymbol{v} = (\mathcal{V}, \succeq)$, and L a current goal under T_{υ}.

The plan P is a *preferred plan* for L iff P is a plan for L and there does not exist another plan P' for L such that $\mathsf{P}' > \mathsf{P}$.

So far, we have presented a formalism that integrates values into actions that an agent can perform and it uses defeasible logic programming as reasoning mechanism and knowledge representation. In our proposed approach, agents perform actions in order to achieve a desirable future state of the world. We called achievable goal to a desirable state achieved from an applicable sequence of actions. Clearly, an agent may have more than one plan for an achievable goal. In this case, it has to select one among those using some strategy. Next, we introduce some particular strategies to obtain preferred plans based on the agent's value system.

5 Value-based preferences

In this section, we will first consider the problem of deciding between plans presenting different alternative strategies. Then, we will introduce a way to complete an agent's value preferences on the basis of a value hierarchy-based formalization.

5.1 Comparison strategies

In this section, we perform a more detailed analysis of how the value system serves as formal tool when comparing plans, by showing some concrete ways to compare plans based on agent's value. Many strategies can be defined. The plan comparison criteria defined below are but some examples.

An action plan is formed by actions, these actions can promote one o more values. Depending on an attitude more skeptical or credulous of agents different comparison criteria can be formulated. Considering this type of attitudes some value comparison criteria are formally defined below.

Let P be an action plan for a given goal L, and let us define $val(\mathsf{P})$ as the set of values promoted by the actions of the plan. In other words,

$$val(\mathsf{P}) = \bigcup_{\langle \mathsf{V}_i, \mathsf{Pos}_i, \mathsf{C}_i, \mathsf{Pre}_i \rangle \in \mathsf{P}} \mathsf{V}_i$$

Given a value system $(\mathcal{V}, \succeq)$, to compare two plans P and P' based on the values promoted by each of them, $val(\mathsf{P})$ and $val(\mathsf{P}')$ respectively, we have to somehow extend the order between values $\succeq$ to an ordering between subsets of values. To this end, let us consider the following extensions of $\succeq$ to subsets of $\mathcal{V}$: for any $\mathsf{A}, \mathsf{B} \subseteq \mathcal{V}$,

- $\mathsf{A} \succeq_{\forall\forall} \mathsf{B}$ if for all $V_1 \in \mathsf{A}$ and $V_2 \in \mathsf{B}$ we have $V_1 \succeq V_2$.
- $\mathsf{A} \succeq_{\exists\exists} \mathsf{B}$ if there is $V_1 \in \mathsf{A}$ and $V_2 \in \mathsf{B}$ such that $V_1 \succeq V_2$.
- $\mathsf{A} \succeq_{\forall\exists 1} \mathsf{B}$ if for all $V_1 \in \mathsf{A}$ there exists $V_2 \in \mathsf{B}$ such that $V_1 \succeq V_2$.
- $\mathsf{A} \succeq_{\forall\exists 2} \mathsf{B}$ if for all $V_2 \in \mathsf{B}$ there exists $V_1 \in \mathsf{A}$ such that $V_1 \succeq V_2$.
- $\mathsf{A} \succeq^*_{\forall\exists} \mathsf{B}$ if $\mathsf{A} \succeq_{\forall\exists 1} \mathsf{B}$ and $\mathsf{A} \succeq_{\forall\exists 2} \mathsf{B}$

In general these relations are not partial orders on the power set $2^{\mathcal{V}}$ since the anti-symmetry property does not hold any longer for them. However, it is easy to check that $\succeq_{\forall\exists 1}$, $\succeq_{\forall\exists 2}$ and $\succeq^*_{\forall\exists} = \succeq_{\forall\exists 1} \cap \succeq_{\forall\exists 2}$ are always reflexive and transitive relations, i.e. they are weak orders. In particular, when $\succeq$ is a total order, then $\succeq^*_{\forall\exists}$ is usually known as the *interval order*.[7] On the other hand, in general, the relation $\succeq_{\forall\forall}$ is transitive but not reflexive, while the relation $\succeq_{\exists\exists}$ is reflexive but not transitive. The former can be easily made a weak order by taking its reflexive closure, and will be of some use later, but the latter it will not be considered any longer.

Although they are not partial orders, for our purposes, the extensions considered above can be used to define different suitable (strict) comparison criteria among plans. Indeed, given two plans P_1 and P_2, the following strict criteria can be defined:

- *Fully skeptical criterion*: $\mathsf{P}_1 >_{s0} \mathsf{P}_2$ if $val(\mathsf{P}_1) \succ_{\forall\forall} val(\mathsf{P}_2)$.

  By definition, $\mathsf{A} \succ_{\forall\forall} \mathsf{B}$ iff

  (i) for all $V_1 \in \mathsf{A}$ and for all $V_2 \in \mathsf{B}$ $V_1 \succeq V_2$, and
  (ii) there is $U_1 \in \mathsf{A}$ and $U_2 \in \mathsf{B}$ such that $U_1 \succ U_2$.

- *Skeptical criterion 1*: $\mathsf{P}_1 >_{s1} \mathsf{P}_2$ if $val(\mathsf{P}_1) \succ_{\forall\exists 1} val(\mathsf{P}_2)$.

  By definition, $\mathsf{A} \succ_{\forall\exists 1} \mathsf{B}$ iff

  (i) for all $V_1 \in \mathsf{A}$ there is $V_2 \in \mathsf{B}$ such that $V_1 \succeq V_2$, and
  (ii) there is $U_2 \in \mathsf{B}$ such that for all $U_1 \in \mathsf{A}$, $U_2 \not\succeq U_1$.

- *Skeptical criterion 2*: $\mathsf{P}_1 >_{s2} \mathsf{P}_2$ if $val(\mathsf{P}_1) \succ_{\forall\exists 2} val(\mathsf{P}_2)$.

  By definition, $\mathsf{A} \succ_{\forall\exists 2} \mathsf{B}$ iff

  (i) for all $V_2 \in \mathsf{B}$ there is $V_1 \in \mathsf{A}$ such that $V_1 \succeq V_2$, and
  (ii) there is $U_1 \in \mathsf{A}$ such that for all $U_2 \in \mathsf{B}$, $U_2 \not\succeq U_1$.

- *Interval-order criterion*: $\mathsf{P}_1 >_{s12} \mathsf{P}_2$ if $val(\mathsf{P}_1) \succ^*_{\forall\exists} val(\mathsf{P}_2)$.

Note that the strict criteria $>_{s0}$, $>_{s1}$, $>_{s2}$ and $>_{s12}$ are irreflexive, asymmetric and transitive comparison criteria. Of course, $>_{s0}$ is the strongest, most restrictive criterion, while $>_{s12}$ is in turn stronger than both $>_{s1}$ and $>_{s2}$. Intuitively, P_1 is preferred to P_2 according to $>_{s0}$ when all the values promoted by P_1 are least as preferred to all values promoted

[7] Indeed, if $(V, \succeq)$ is a total order, it turns out that, if $a \succeq b$ and $c \succeq d$, then $[a, b] \succeq^*_{\forall\exists} [c, d]$ iff $a \succeq c$ and $b \succeq d$.

by P_2, so it is a sort of full dominance criterion. On the other hand, P_1 is preferred to P_2 according to $>_{s1}$ when any value of P_1 is at least as preferred to some value of P_2, but there can be values of P_2 that are strictly more preferred than values of P_1. P_1 is preferred to P_2 according to $>_{s2}$ whenever any value of P_2 is dominated by some value of P_1. This sounds as a more reasonable criterion than $>_{s1}$ if we want to choose a plan that promotes as many good values as possible. The combination of $>_{s1}$ and $>_{s2}$, i.e. $>_{s12}$, seems like even a better compromise.

Example 8 To see the intuitive behaviour of the four criteria $\succeq_{\forall\forall}$, $\succeq_{\forall\exists 1}$, $\succeq_{\forall\exists 2}$ and $\succeq^*_{\forall\exists}$, consider an abstract set of values $\mathcal{V} = \{a, b, c, d, e, f\}$, with the following total order: $a \succ b \succ c \succ d \succ e \succ f$. Next we consider comparing four different pairs of sets of values from $\mathcal{V}$:

Case 1: For $\mathsf{A} = \{b, c\}$, $\mathsf{B} = \{a, b, d\}$, we have the following comparisons:

- $\mathsf{A} \not\succeq_{\forall\forall} \mathsf{B}$, $\mathsf{B} \not\succeq_{\forall\forall} \mathsf{A}$
- $\mathsf{A} \succeq_{\forall\exists 1} \mathsf{B}$, $\mathsf{B} \not\succeq_{\forall\exists 1} \mathsf{A}$ and $\mathsf{A} \succ_{\forall\exists 1} \mathsf{B}$
- $\mathsf{B} \succeq_{\forall\exists 2} \mathsf{A}$, $\mathsf{A} \not\succeq_{\forall\exists 2} \mathsf{B}$ and $\mathsf{B} \succ_{\forall\exists 2} \mathsf{A}$.
- $\mathsf{A} \not\succeq^*_{\forall\exists} \mathsf{B}$ and $\mathsf{B} \not\succeq^*_{\forall\exists} \mathsf{A}$.

Case 2: For $\mathsf{A} = \{b, c, e\}$, $\mathsf{B} = \{c, d\}$, we have:

- $\mathsf{A} \not\succeq_{\forall\forall} \mathsf{B}$, $\mathsf{B} \not\succeq_{\forall\forall} \mathsf{A}$
- $\mathsf{A} \not\succeq_{\forall\exists 1} \mathsf{B}$, $\mathsf{B} \succeq_{\forall\exists 1} \mathsf{A}$ and $\mathsf{B} \succ_{\forall\exists 1} \mathsf{A}$
- $\mathsf{B} \not\succeq_{\forall\exists 2} \mathsf{A}$, $\mathsf{A} \succeq_{\forall\exists 2} \mathsf{B}$ and $\mathsf{A} \succ_{\forall\exists 2} \mathsf{B}$
- $\mathsf{A} \not\succeq^*_{\forall\exists} \mathsf{B}$ and $\mathsf{B} \not\succeq^*_{\forall\exists} \mathsf{A}$.

Case 3: For $\mathsf{A} = \{a, b, c\}$ and $\mathsf{B} = \{b, c, d\}$, we have:

- $\mathsf{A} \not\succeq_{\forall\forall} \mathsf{B}$, $\mathsf{B} \not\succeq_{\forall\forall} \mathsf{A}$
- $\mathsf{A} \succeq_{\forall\exists 1} \mathsf{B}$, $\mathsf{B} \not\succeq_{\forall\exists 1} \mathsf{A}$ and $\mathsf{A} \succ_{\forall\exists 1} \mathsf{B}$
- $\mathsf{A} \succeq_{\forall\exists 2} \mathsf{B}$, $\mathsf{B} \not\succeq_{\forall\exists 2} \mathsf{A}$ and $\mathsf{A} \succ_{\forall\exists 2} \mathsf{B}$
- $\mathsf{A} \succeq^*_{\forall\exists} \mathsf{B}$, $\mathsf{B} \not\succeq^*_{\forall\exists} \mathsf{A}$, $\mathsf{A} \succ^*_{\forall\exists} \mathsf{B}$.

Case 4: For $\mathsf{A} = \{a, b\}$ and $\mathsf{B} = \{b, c, d\}$, we have:

- $\mathsf{A} \succeq_{\forall\forall} \mathsf{B}$, $\mathsf{B} \not\succeq_{\forall\forall} \mathsf{A}$ and $\mathsf{A} \succ_{\forall\forall} \mathsf{B}$
- $\mathsf{A} \succeq_{\forall\exists 1} \mathsf{B}$, $\mathsf{B} \not\succeq_{\forall\exists 1} \mathsf{A}$ and $\mathsf{A} \succ_{\forall\exists 1} \mathsf{B}$.
- $\mathsf{A} \succeq_{\forall\exists 2} \mathsf{B}$, $\mathsf{B} \not\succeq_{\forall\exists 2} \mathsf{A}$ and $\mathsf{A} \succ_{\forall\exists 2} \mathsf{B}$.
- $\mathsf{A} \succeq^*_{\forall\exists} \mathsf{B}$, $\mathsf{B} \not\succeq^*_{\forall\exists} \mathsf{A}$, $\mathsf{A} \succ^*_{\forall\exists} \mathsf{B}$.

Example 9 Considering the specification of a policy-maker $T_{\boldsymbol{\nu}} = (\Psi_2, \Delta_1, \Gamma_3, G_7)$ introduced in Example 7, two plans $\mathsf{P}_2 = [\mathsf{sF}, \mathsf{cFU}]$ and $\mathsf{P}_3 = [\mathsf{mIS}]$ can be obtained for the current policy goal $G_7 = iH$. Given the set of values $val(\mathsf{P}_1) = \{\mathbf{eP,eD,wW}\}$ and $val(\mathsf{P}_2) = \{\mathbf{eD}\}$,[8] and the partial order $\succeq_6$ given in Example 6 ($\mathbf{eP} \succ_6 \mathbf{eD}$ and its reflexive closure), using skeptical criterion $>_{s1}$, we have that P_1 is preferred to P_2, and hence

[8]Recall from Example 6, '**eP**' stands for *environment protection*, '**eD**' for *economic development*, '**wW**' for *workers well-being*,, and '**w**' for *general well-being*.

{**eP**,**eD**,**wW**} $\succeq_{\forall\exists 1}$ {**eD**}. However, no plan can be selected according to the other criteria because plans are incomparable under those comparison criteria.

More sophisticated criteria could be used combining the different comparison methods mentioned above in a lexicographic way. That is, one can, for instance, consider first the interval-order criterion $>_{s12}$; if no plan is preferred, then one can apply another comparison method, for instance the skeptical criterion $>_{s2}$, and finally, if needed, the remaining criterion $>_{s1}$. In this way, one can consider a number of refinements of each comparison criteria.

On the other hand, in the existing literature, several approaches to compare plans can be found, e.g.,majority rules widely used for social choice problems [12]. Following this particular approach, a plan could be considered better to another one by majority of preferred values.

In the context of social choice functions, given a preference ordering on a set of alternatives, Gärdenfors proposed in [22] a comparison criterion among subsets of alternatives based on the following reading of the sure-thing principle: “if some alternative has been added, it should be at least as good as all the other alternatives, and if some alternative has been deleted, it should be worse than the remaining alternatives”. In the next definition, we adapt this idea to also compare plans.[9]

Definition 10 [cf. [22]] Let $(\mathcal{V}, \succeq)$ be a value system, and P_1 and P_2 two plans.

The plan P_1 is G-preferred to P_2, denoted $\mathsf{P}_1 >_G \mathsf{P}_2$, iff one of the following conditions is satisfied:

(C1) $val(\mathsf{P}_1) \subset val(\mathsf{P}_2)$ and $val(\mathsf{P}_1) \succ_{\forall\forall} val(\mathsf{P}_2) \setminus val(\mathsf{P}_1)$

(C2) $val(\mathsf{P}_2) \subset val(\mathsf{P}_1)$ and $val(\mathsf{P}_1) \setminus val(\mathsf{P}_2) \succ_{\forall\forall} val(\mathsf{P}_2)$

(C3) $val(\mathsf{P}_1) \not\subset val(\mathsf{P}_2)$, $val(\mathsf{P}_2) \not\subset val(\mathsf{P}_1)$, $val(\mathsf{P}_2) \neq val(\mathsf{P}_1)$ and $val(\mathsf{P}_1) \setminus val(\mathsf{P}_2) \succ_{\forall\forall} val(\mathsf{P}_2) \setminus val(\mathsf{P}_1)$

Note that in the conditions (C1), (C2) and (C3) above, since $\succ_{\forall\forall}$ compares disjoint sets, we could safely replace $\succ_{\forall\forall}$ by $\succeq_{\forall\forall}$. It is shown in [22] that $>_G$ is a irreflexive and transitive relation.

In the following, given a value system $(\mathcal{V}, \succeq)$ and $n \in \{1, 2, 3\}$, we will write $(Cn)(\mathsf{P}_1, \mathsf{P}_2)$ to denote that condition (Cn) introduced in Def. 10 above holds for plans P_1 and P_2.

Proposition 1 *It holds that:*

1. *If* $\mathsf{P}_1 >_{s0} \mathsf{P}_2$ *then* $\mathsf{P}_1 >_G \mathsf{P}_2$.
2. *If* $\mathsf{P}_1 >_G \mathsf{P}_2$ *then* $\mathsf{P}_1 >_{s1} \mathsf{P}_2$.
3. *If* $\mathsf{P}_1 >_G \mathsf{P}_2$ *then* $\mathsf{P}_1 >_{s2} \mathsf{P}_2$.
4. *If* $\mathsf{P}_1 >_G \mathsf{P}_2$ *then* $\mathsf{P}_1 >_{s12} \mathsf{P}_2$.

Proof We prove items 1 and 2, since item 3 is very similar to item 2, and item 4 is just a combination of items 2 and 3.

[9] In [22], the preference relation on alternatives is supposed to be a total weak order (i.e. a reflexive transitive and connected relation), while in a value system we assume the preference relation on values $\succeq$ is assumed to be a partial order (i.e. a reflexive, anti-symmetric and transitive relation).

1. Clearly $\mathsf{P}_1 >_{s0} \mathsf{P}_2$ implies $val(\mathsf{P}_1) \succeq_{\forall\forall} val(\mathsf{P}_2)$, and this in turn implies $A \succeq_{\forall\forall} B$ for any subsets $A \subseteq val(\mathsf{P}_1)$ and $B \subseteq val(\mathsf{P}_2)$. Finally, the claim follows by observing (as mentioned above) that $A \succeq_{\forall\forall} B$ iff $A \succ_{\forall\forall} B$ whenever $A \cap B = \emptyset$.
2. It is enough to check that each of the three conditions (C1), (C2) and (C3) implies $val(\mathsf{P}_1) \succeq_{\forall\exists 1} val(\mathsf{P}_2)$. Consider for instance (C1) and assume $val(\mathsf{P}_1) \subset val(\mathsf{P}_2)$. Then, if $val(\mathsf{P}_1) \succ_{\forall\forall} val(\mathsf{P}_2) \setminus val(\mathsf{P}_1)$, then we also have $val(\mathsf{P}_1) \succ_{\forall\exists 1} val(\mathsf{P}_2) \setminus val(\mathsf{P}_1)$ and hence $val(\mathsf{P}_1) \succ_{\forall\exists 1} val(\mathsf{P}_2)$ as well due to the reflexivity of $\succeq$. The other cases are proved in a similar way □

Therefore, Gärdenfors criterion $>_G$ can be seen as an intermediate comparison criterion between the strongest one $>_{s0}$ and the remaining criteria $>_{s1}$, $>_{s2}$ and $>_{s12}$.

Finally, let us mention that all these comparison criteria satisfy a sort of strong dominance principle: if P_1 and P_2 are two plans such that $val(\mathsf{P}_1) \cap val(\mathsf{P}_2) = \emptyset$ and such that all values in $val(\mathsf{P}_1)$ are strictly preferred to all values in $val(\mathsf{P}_2)$, then $\mathsf{P}_1 > \mathsf{P}_2$.

5.2 Value hierarchy-based comparison

As we have already mentioned, one of our goals is to consider values promoted by actions to be a very important aspect when deciding among action plans. So the definition of a criterion for resolving conflicts between values becomes a central issue for our formalization. The approach we consider in this section is that, when the set of values come organised as a hierarchy according to a specificity or subtype relation, a suitable criterion should not only depend on the available agent's preferences over (some) values, but also on such hierarchical order. The underlying idea is that if a value is a subtype of, or more specific than, another value, then if an action promotes the first value, it should also promote the second one.

Therefore, in this section we consider a scenario where, besides assuming to be partially ordered by an importance/preference criterion, we also assume the set of values to be organised in a hierarchy tree, from more general to more specific values. We show how, under some constraints, both structures can be combined and yield a suitable new partial order on values. Note that this order on values should then be extended to compare sets of values, and eventually to some strategy to compare plans, as done in the previous section.

More formally, we assume the (finite) set of values $\mathcal{V}$ is equipped with a partial order of specificity $\gg$ having a least element τ (the root) and the property that for each $V \in \mathcal{V}$, the set $\mathsf{Ancestors}(V) = \{U \in \mathcal{V} \mid V \gg U\}$[10] is a chain (i.e. totally ordered). In this way, the ordered set $(\mathcal{V}, \gg)$ corresponds to a rooted tree, where a value in a path from the root to a leaf value is all the more specific as it is nearer to the leaf. Namely, if $V_1 \gg V_2$, then it will be said that V_1 is a subtype of V_2 or that V_1 is a value more specific than V_2. In other words, V_1 is a *descendant* of V_2, and V_2 is an *ancestor* of V_1. Moreover, in the following, for $V \neq \tau$ we will denote by $\mathsf{parent}(V)$ the closest ancestor of V according to $\gg$, i.e. $\mathsf{parent}(V) = U$ iff $U \in \mathsf{Ancestors}(V)$, $U \neq V$, and there is no other value U' different from U and V such that $V \gg U' \gg U$. Also, the siblings of a value V will be those values sharing the same parent, namely $\mathsf{siblings}(V) = \{U \in \mathcal{V} \mid \mathsf{parent}(U) = \mathsf{parent}(V)\}$. Note that $U \in \mathsf{siblings}(V)$ iff $V \in \mathsf{siblings}(U)$.

In such a scenario, we further assume an agent expresses her (partial) preferences over values by means of a value system $(\mathcal{V}, \succeq)$, where $\succeq$ is a preference partial order. The

[10] Note that since $\gg$ is assumed to be a partial order, it is reflexive, and hence $V \in \mathsf{Ancestors}(V)$.

question is then how to make use of the hierarchy order in order to complement the preference order. Consider for instance actions A_1 and A_2 promoting values V_1 and V_2 respectively, and further assume that $V_1 \gg V_2$ and $V_1 \succ V_2$, A_1 can be deemed to be preferred to A_2 because it promotes a value more important. However, A_2 also indirectly promotes V_1, because V_1 is a subtype of V_2. Therefore both actions are promoting the same value and it would not be a coherent decision to prefer A_1 to A_2. Consequently, in such a case, making the preference $V_1 \succ V_2$ explicit can lead to some indeterminism. For this reason, and to avoid this kind of situations, we will only allow the agent to express preferences over siblings of the hierarchical structure $(V, \gg)$. In this way, comparing two values that are not siblings is carried out by comparing their nearest ancestors that are siblings.

This combined notion of preference order is formalized in the following definitions.

Definition 11 (PH-structure) Let $\mathcal{V}$ be the set of agent's values. A preference-hierarchy structure on $\mathcal{V}$ is a tuple $(\mathcal{V}, \gg, \succeq)$ such that:

1. $(\mathcal{V}, \gg)$ is a rooted tree, that is, $\gg$ is partial order with a least element τ (root) satisfying the following property: for each $V \in \mathcal{V}$, the set $\mathsf{Ancestors}(V) = \{U \in \mathcal{V} \mid V \gg U\}$ is a chain.
2. $(\mathcal{V}, \succeq)$ is a partial order.
3. $\gg$ and $\succeq$ are orthogonal, i.e.,distinct values cannot be comparable both with $\gg$ and $\succeq$.
4. If $V \succeq U$ then $V \in \mathsf{siblings}(U)$.

If the preference order $\succeq$ is total on the sibling sets of $(\mathcal{V}, \gg)$, then we will call $(\mathcal{V}, \gg, \succeq)$ to be a complete PH-structure

In short, a PH-structure is a hierarchy together with a preference ordering over sibling values. In fact, when the order $\succeq$ on siblings is total, then the structure $(\mathcal{V}, \gg, \succeq)$ is usually known as an *ordered tree* [25].

Example 10 Consider the set of values $\mathcal{V}_{10} = \{\mathbf{eP}, \mathbf{eD}, \mathbf{wW}, \mathbf{sW}, \mathbf{w}, \tau\}$. We can define a PH-structure $(\mathcal{V}_{10}, \gg_{10}, \succeq_{10})$, where $\gg_{10} = \{(\mathbf{eP}, w), (\mathbf{sW}, \mathbf{w}), (\mathbf{eD}, \mathbf{sW}), (\mathbf{wW}, \mathbf{sW}), (wW, w), (eD, w), (w, \tau), (eP, \tau), (eD, \tau), (sW, \tau), (wW, \tau)\}$, and $\succeq_{10} = \{(\mathbf{eP}, \mathbf{sW})\}$.

As already announced, given a PH-structure $(\mathcal{V}, \gg, \succeq)$, one can extend the preference order on siblings sets of values to other values by making use of the hierarchy relation $\gg$ in a natural way by looking at their sibling ancestors.

Definition 12 (Combined value system) Let be $(\mathcal{V}, \gg, \succeq)$ a PH-structure. Then the combined value system is the pair $(\mathcal{V}, \succeq_{\mathsf{H}})$, where the (strict) preference order $\succ_{\mathsf{H}}$ is defined as follows:

$$V \succ_{\mathsf{H}} U \quad \text{iff} \quad V' \succ U' \text{ for } V' \in \mathsf{Ancestors}(V), U' \in \mathsf{Ancestors}(U) \text{ such that } V' \in \mathsf{siblings}(U').$$

In other words, $\succ_{\mathsf{H}}$ is the following composition of partial orders: $\succ_{\mathsf{H}} = \gg \circ \succ \circ \ll$, where $\ll$ is the converse relation of $\gg$, and $\circ$ denotes composition of relations. Then $\succeq_{\mathsf{H}}$ is just the reflexive closure of $\succ_{\mathsf{H}}$.

First of all, we check that $\succeq_{\mathsf{H}}$ is well defined in the sense that it properly extends $\succeq$.

Proposition 2 $\succeq_{\mathsf{H}}$ *is a partial order that extends* $\succeq$.

Proof We prove first that $\succ_{\mathsf{H}}$ is a strict partial order and then that it extends $\succ$:

(i) - The relation $\succ_{\mathsf{H}}$ is irreflexive:
Assume $V \succ_{\mathsf{H}} V$, then $V \gg A$, $A \succ B$ and $B \ll V$, for some siblings $A, B \in \mathcal{V}$. That is, $A, B \in \mathsf{Ancestors}(V)$, hence either $A \gg B$ or $B \gg A$. But this is impossible if A and B are siblings.
- The relation $\succ_{\mathsf{H}}$ is asymmetric:
Assume $V \succ_{\mathsf{H}} U$ and $U \succ_{\mathsf{H}} V$. There exist siblings $A, B \in \mathcal{V}$ such that $V \gg A$, $A \succ B$ and $B \ll U$, and conversely $V \gg A$, $B \succ A$ and $B \ll U$, that is impossible since $\succ$ is asymmetric.
- The relation $\succ_{\mathsf{H}}$ is transitive:
Assume $V \succ_{\mathsf{H}} U$ and $U \succ_{\mathsf{H}} W$. Then, there are A, B, C, D such that (1) $V \gg A$, $A \succ B$ and $B \ll U$, and (2) $U \gg C$, $C \succ D$ and $D \ll W$. Since both B and C are ancestors of U, then either $B = C$ or $B \neq C$. If $C = B$ then $A \succ B = C \succ D$, and then we have $V \gg A$, $A \succ D$ and $D \ll W$, and hence $V \succ_{\mathsf{H}} W$. Otherwise, if $B \neq C$ then either $B \gg C$ or $C \gg B$. W.l.o.g., assume $B \gg C$. Then, C is also an ancestor of A, and thus of V as well. Hence, we have $V \gg C$, $C \succ D$ and $D \ll W$, that is, $V \succ_{\mathsf{H}} W$.

(ii) - The order $\succ_{\mathsf{H}}$ extends $\succ$:
Indeed, if V and U are siblings such that $V \succ U$, then we have $V \gg V$, $V \succ U$ and $U \gg U$. Hence $V \succ_{\mathsf{H}} U$ holds as well. In fact we have something stronger, i.e. if V and U are siblings, then $V \succ_{\mathsf{H}} U$ iff $V \succ U$. □

Next, we gather (without proof) three properties of the combined order $\succ_{\mathsf{H}}$ that easily follow from the definition.

Proposition 3 *Let* $(\mathcal{V}, \gg, \succeq)$ *be a preference-hierarchy structure, and let two different values* $V, U \in \mathcal{V}$.

- *If* $V \succ U$, *then* $V' \succ_{\mathsf{H}} U'$ *for any* $V' \in \mathsf{Descendants}(V)$ *and* $U' \in \mathsf{Descendants}(U)$.
- *If* V *and* U *are incomparable w.r.t.* $\succeq$, *then for any* $V' \in \mathsf{Descendant}(V)$ *and* $U' \in \mathsf{Descendant}(U)$, V' *is incomparable to* U' *w.r.t.* $\succ_{\mathsf{H}}$.
- *If* $V \gg U$, *then* V *and* U *are incomparable w.r.t.* $\succ_{\mathsf{H}}$.

Example 11 Given the PH-structure $(\mathcal{V}_{10}, \gg_{10}, \succeq_{10})$ introduced in Example 10, the following two sets of ancestor values can be obtained from $\gg_{10}$: $\mathsf{Ancestors}(\mathbf{eD}) = \{\mathbf{sW}, \mathbf{w}, \tau\}$ and $\mathsf{Ancestors}(\mathbf{eP}) = \{\mathbf{w}, \tau\}$. Note that the values **eP** and **eD** are not comparable according to the relation $\succeq$. Nevertheless, since $\mathbf{eP} \succ_{10} \mathbf{sW}$ and by Definition 12, it is possible to stablish the following combined preference $\mathbf{eP} \succ_{\mathsf{H}} \mathbf{eD}$.

Several conclusions can be drawn from what we obtained in this section. In general, we would like to emphasize the fact that combined order criterion not only helps to understand how agent's values can be used for comparing plans, but also allows us to discuss several aspects related to the hierarchy structure associated to such values. For instance, requiring $(\mathcal{V}, \gg)$ be a tree allows us to always find ancestors of any pair of values that are siblings, and hence the possibility of comparing the values whenever the sibling ancestors are comparable as well. This would not be possible in case either dealing with value hierarchies involving only some of agent's values or with a weaker structure. Finally, let us remark that the combined value system $(\mathcal{V}, \succeq_{\mathsf{H}})$ that can be obtained from a given PH-structure $(\mathcal{V}, \gg, \succeq)$ is not a final tool by itself (it only serves to compare values), rather $\succeq_{\mathsf{H}}$ is meant to be used to define corresponding comparison strategies for plans according to the different extensions to compare subsets of values discussed in the previous version.

6 Related work

There are several approaches that combine argumentation with practical reasoning theory. Atkinson *et al.* [7] propose the use of argument schemes for justifying actions and critical questions as objections to the justifications. The argument scheme is stated as follows: in current circumstances R, I should perform action A, to bring about new circumstances S, which will achieve goal G and promote value V. An Action-Based Alternating Transition System (AATS), originally presented by [49], is a formal structure for an explicit representational model of states, actions, and transitions. As defined in [49], the AATS does not include a reference to the values of agents. However, in order to adapt this system to use with argument schemes, Atkinson and Bench-Capon in [5] extended AATS labelling transitions to indicate which values are promoted and which are demoted. To represent a particular problem with an AATS with values, one first identifies a set of propositional variables that are considered relevant to the problem. Each combination of the corresponding positive or negative literals will be a potential state of the system. Then, one identifies the incumbent agents, the different possible actions they can perform and how these will lead to those states; each transition represents a joint action of the agents involved. Finally, one should determine values and relate them to the transitions between states. Inspired by [7], we formalise a particular way to specify an agent's actions and propose different criteria to decide how to choose between them depending on a hierarchical order over values. The concept of state, however, is different between both approaches. When an agent performs an action in a given state of the world, it produces a new state. A state transition can thus be seen as a truth assignment to a set of propositions (since due to the agent actions some facts in the world become true and others may cease to be). In our proposal facts can adopt three different values: true, false, or unknown.

In [35] the authors have proposed an argumentation-based approach for practical reasoning that extends the works of [2] and [1] presenting three different instantiations of Dung's framework to reason about beliefs, desires and plans, respectively.

Similarly to us, [35] use a defeasible argumentation formalism to represent and reason over contradictory information. However, and although [35] is based on structured argumentation, there are some differences with our proposal. Firstly, our focus does not follow a BDI architecture model. Moreover, although their work uses the concept of plan, their approach determines which plan prevails after an argumentative analysis. This approach is related more to our notion of argument than to our formalisation of action plan.

Van der Weide *et al.*, in [47], present an argumentation approach for reasoning over state preferences where argument schemes [7] are used to determine preferences between states. The values that agents hold are used to compare states. In this formalism the use of perspectives as pre-orders over states is central: when it is not possible to determine which of two states is preferred from the perspective of a value, then this preference can be inferred by means of the notion of influence between perspectives. One of the main contributions of this work is the proposed value system that is used for resolving conflicts that can appear when a transition state promotes some values and demotes other values. In a similar development, we start from the notion of argument scheme proposed in [7] but, unlike them, our main aim is not to provide a way of using argument schemes to give meaning to values and to determine whether values are promoted or demoted. To find preferred states they propose to use argumentation, but no particular argumentation framework is proposed. In our approach the situation is different, an agent has a set of actions promoting different values and a concrete argumentation formalism is used to represent and reason over contradictory knowledge. Despite sharing the motivation of using values, we do not focus on introducing a number of argument schemes for reasoning about whether values are promoted. In [47], the influence relation denotes that one perspective is an aspect of another which clearly can see as a way to arrange values hierarchically. However, this approach differs from ours where we provide an agent with the capability to complete their preferences over values using the hierarchy under which such values are arranged.

Another approach for practical reasoning was proposed in [48]. This work extends the proposal of [47] considering several argument schemes, but formalised as an argumentative system to argue about what decision an agent should take based on its preferences over outcomes. In this approach, the agent's preferences are represented in terms of values and goals. The originality of this approach lies in representing values with a perspective, which is associated with an ordering over outcomes. Like us, the authors combine argumentation with values; however, they are used for a different purpose. In our approach, values constitute a central component in actions because the agent can use them to decide between different action plans, whereas argumentation is the reasoning mechanism that the agent will use to reason over conflicting domain information. In [48], an argumentation system is proposed to reasons about what decision is best. To accomplish that, agents have an ordering over perspectives that represents what values they find most important. An interesting characteristic of our model is that the approach proposed in [48] could be extended incorporating the mechanism to complete agent's value preferences introduced here.

There are several works that relate a general model for agent reasoning to argumentative systems [3, 13, 23, 39, 46]; however, none of these provides grounds to formalise the relationship between agent's values and the actions that they can perform.

Gottifredi *et al*. propose, in [23], a declarative agent programming language where mental components are represented and uses defeasible logic programming as the reasoning formalism. An agent will be able to represent conflicting goals and beliefs, and the DeLP is used for deciding which beliefs and goals are warranted. In particular, the focus is in showing how each agent deliberative cycle is performed using argumentation and how actions are affected by this deliberation. In order to decide how to act, agents use plan rules. These rules establish the action plan to execute with the objective of achieving its associated goals. Similarly, our approach proposes a declarative agent specification based on defeasible argumentation but there are differences between the two approaches: First, our aim is not to formalise an agent programming language describing each agent's mental constructs and emphasising the model dynamics. We assume agents can decide how to act taking into account those values that they consider important. In other words, we adopt an argumentation-based approach that enables an explicit representation of states, values, and actions of an agent. Second, we focus on the use of a set of values arranged hierarchically to guide the selection of plan actions, whereas in [23] there are no specific mechanisms to compare plans. In our approach, the hierarchy of values plays an important role in the choice of a particular plan.

Our proposal is also related to work in belief revision. A key work in revision for Dung's argumentation frameworks was presented in [14], where the problem of revision of argumentation frameworks is translated into propositional logic favouring the use of AGM revision operators for dealing with this matter. The output of revision is a set of argumentation frameworks instead of a single argumentation framework. More recently, Diller *et al*. [15] propose two revision approaches to produce a single argumentation framework as an outcome of the extensions of a set of involved argumentation frameworks. First, they introduce a revision by propositional formulas. Similar to the approach in [14], they revise an argumentation framework so that its extensions are modified according to the models of the formula at issue. Then, they introduce a revision operator that maps two argumentation framework into a new argumentation framework. In [17], Doutre and Maiilly discuss other contributions on dynamics of argumentation systems.

A salient point of those framework revision approaches is the fact that the revision process takes into account an epistemic evolution of the world. However, other approaches take into account a different perspective: a physical evolution of the world, what in belief change theory is called an *update*. For instance, [17] focuses in update operations for handling dynamics in abstract argumentation graphs. The originality of this work lies in that they present a first-order logical language YALLA capable of expressing attack relation between sets of arguments in abstract argumentation graphs. Although it is not their study focus, the authors state that their approach can be used to deal with practical aspects of how an agent can achieve goals, as we do.

Another interesting work in belief update is the one by Doutre *et al*. [16]. In that paper, the authors encode argumentation frameworks and their dynamics in Dynamic Logic of Propositional Assignments (DL-PA); and show how extensions of an argumentation framework can be constructed by means of DL-PA programs. In particular, enforcing operations consist in minimally updating the attack relation in an argumentation framework. This minimal update is done with respect to a specific formula representing a goal, so that this goal holds either in all extensions or in some extensions of the argumentation framework. Besides having a different research focus (practical reasoning with values vs dynamic in argumentation), in our case the way for identifying the goals that can be considered as acceptable conclusions deviates from Dung's semantics. In particular, we use DeLPs warrant procedure which uses a mechanism based on the construction of dialectical trees to determine if an argument is warranted or not. A noteworthy advantage of incorporating DeLP in our approach is the possibility to exploit planning algorithms proposed in [21] to achieve agent's goals, and resolve different types of threats. Although dynamics in argumentation is not our focus in this paper, a detailed analysis of properties leading to characterise update operators for our application domain would be an appealing line for future work.

We should mention that there are several approaches to structured argumentation, like ABA [11] and ASPIC$^+$ [29] or Besnard and Hunter's deductive argumentation [10], that, analogously to DeLP, could be adapted as well to our value-driven argumentation proposal. Although these approaches are similar in many regards, also differ at several points, as mentioned in [9]. Namely, first of all, ABA and ASPIC$^+$ are explicitly within Dung's semantics [18], while DeLP deviates from Dung's approach to select accepted arguments and justified conclusions through the construction and marking of dialectical trees. As shown in this paper, besides an action's effects, there could be more effects that can be deduced using the inference process of DeLP. Second, DeLP and ASPIC$^+$ use two different types of rules, called strict and defeasible rules, while ABA reduces defeasible rules to strict rules plus assumptions. In contrast, in Besnard and Hunter's deductive argumentation the inference rules are the inference rules of the base logic. Finally, ASPIC$^+$, DeLP, and deductive argumentation are approaches that allow for the use of explicit preferences to resolve conflicts between arguments, while ABA encodes preferences in assumption-premises of rules.

7 Closing remarks

In this paper we have looked into policy-making as value-driven processes [32] to illustrate how an agent may use its values to choose among alternative actions and then assemble these into plans that reflect the agent's preferences. We claimed that argumentation, as a cognitive construct, constitutes a significant resource to deal with conflicts and showed how an agent can represent its values and exploit them to decide among action plans. As noted before [6, 47, 48], resorting to the values held by an agent is an interesting way of reflecting the agent's preferences into the decision-making process. However, in most approaches the preferences of an agent are expressed in terms of a linear ordering over values.

In Section 4 we introduced an agent model whose knowledge base is represented as sets of defeasible rules that involve values, and uses defeasible argumentation (based on Defeasible Logic Programming (DeLP)) as its reasoning formalism. Thus, an agent is able to represent conflicting information and use defeasible argumentation to determine whether some particular information is accepted as warranted. Moreover, agents use argumentation to move from one desirable state to another and thus generate action plans in order to achieve their goals. In particular, we showed how substitutions for free variables obtained from DeLP argumentation mechanism can be used during the generation of action plans, a valuable feature in dynamic environments.

In Section 5 we organise the values of an agent as a partial order and use this system to establish criteria to compare plans to decide which plan prevails. In order to deal with values that are not commensurable, we interpret the partial ordering of values as a hierarchical structure, and then base the decision on the analysis of preferences between ancestors of such values. Finally, we also analysed several properties that explain why some types of value hierarchies would not be suitable to decide over actions.

Some lines of future work are formal: In this paper we used an austere notion of the relevant state of the world that involves only value-specific indexes and facts that are directly involved in actions. Future work would be the extension of our framework to a richer notion of relevant state that could include variables that would allow different interpretations of the same value. Similarly, one could explore a richer notion of transition between states and include, for example, events that are independent of agents' actions, actions that are not directly oriented towards values, norms that restrict or condition transitions (beyond the current association with facts as pre- and post-conditions for actions) and cost-like features like incentives or sanctions that would affect preferences among states and thus plan-formation and choices.

Another research line is the introduction of other forms of (dynamic?) ordering of values like "saliency" (to respond to urgent or unforeseen situations) or "local relevance" (for different classes of relevant states of the world). Likewise, one may explore alternative ways of ordering values. For instance explore other forms of dealing with incommensurable values —like aggregation "satisfising" aggregation models or welfare functions— and provide criteria for deciding among these and value hierarchies.

An additional line for future work is to investigate how to integrate an automated planning system to our proposal, and provide tools for modelling agent's behaviour using the value system proposed in our formalisation. Although we have chosen DeLPas our structured argumentation framework for structured argumentation, several extensions of DeLP have been proposed and many of them are mainly related to applications. We intend to explore in the future the work of García et al. in [21] where a DeLP-based planning formalism, and an extension of the traditional POP algorithm to consider arguments as planning steps is proposed. A concrete topic for future work is to follow the approach proposed in

[21] and use our approach of value hierarchies to resolve interference (threats) that can appear in a plan.

We mentioned above that the characterisation of update operators holds research promise. For instance, to study means to determine the literals that should prevail in action pre-conditions. In this respect, it would be interesting to study the construction of operators in collaborative multi-agent applications where agents act as information sources proposing arguments for and against supporting pre-conditions; and in this context, explore the relevance of the credibility degree of agents. Similarly, one may work towards improving the capabilities and scope of current practical reasoning mechanisms to a wider number of domains. One possibility is to study domain-dependent update operators, for instance with different semantics for dealing the degree of domain-dependent "reliability" attached to agents, and study how these semantics affect the behaviour of the system.

Form an empirical standpoint there are also lines for future work. In this paper we discussed a formalism for value-driven preferences for the comparison of plans that is suitable to model policy-making processes. It would be useful to study other application domains to get a better understanding of the applicability of our proposal and, complementary, the characterisation of domains where other forms of value-driven preferences are involved. For instance, terminal and palliative health care, international trade conflict resolution and assistive robotics.

In terms of other types of applications, one may explore the use our formalism to model agent based simulations of other value-dependent social phenomena. A side objective of these simulations may be to explore the use of our framework with other cognitive constructs that complement values and argumentation — for instance motivation, personality or needs— in order to explain goal-setting and planning.

Acknowledgements The authors are indebted to the anonymous reviewers for their in-depth constructive comments and suggestions that have help to significantly improve the paper. Teze acknowledges partial support by CONICET, Universidad Nacional del Sur (UNS), and Universidad Nacional de Entre Ríos (UNER), Argentina. Perelló-Moragues is supported with an AGAUR industrial doctoral grant sponsored by FCC AQUALIA, IIIA-CSIC, and UAB. Godo, Perelló-Moragues and Noriega acknowledge the AppPhil project (funded by Caixa Bank, RecerCaixa 2017) and the Spanish FEDER/ MINECO project CIMBVAL (TIN2017-89758-R). Godo's work is also supported by the Spanish FEDER/ MINECO project RASO (TIN2015-71799-C2-1-P).

References

1. Amgoud, L.: A formal framework for handling conflicting desires. In: 7th European conference symbolic and quantitative approaches to reasoning with uncertainty, ECSQARU 2003, Aalborg, Denmark, July 2-5, 2003. Proceedings. pp. 552–563 (2003)
2. Amgoud, L., Cayrol, C.: A reasoning model based on the production of acceptable arguments. Ann. Math. Artif. Intell. **34**(1-3), 197–215 (2002)
3. Amgoud, L., Devred, C., Lagasquie-Schiex, M.C.: A constrained argumentation system for practical reasoning. In: AAMAS. pp. 429–436 (2008)
4. Anonymous: I Am Part of the Resistance Inside the Trump Administration. Op-Ed, The New York Times, Sep 5. https://en.wikipedia.org/wiki/I_Am_Part_of_the_Resistance_Inside_the_Trump_Administration (2018)
5. Atkinson, K., Bench-Capon, T.: Practical reasoning as presumptive argumentation using action based alternating transition systems. Artif. Intell. **171**(10-15), 855–874 (2007)
6. Atkinson, K., Bench-Capon, T.J.M.: States, goals and values: Revisiting practical reasoning. Argument & Computation **7**(2-3), 135–154 (2016)
7. Atkinson, K., Bench-Capon, T.J.M., McBurney, P.: Computational representation of practical argument. Synthese **152**(2), 157–206 (2006)

8. Awad, E., Dsouza, S., Kim, R., Schulz, J., Henrich, J., Shariff, A., Bonnefon, J.F., Rahwan, I.: The moral machine experiment. Nature p. 1 (2018)
9. Besnard, P., Garcia, A., Hunter, A., Modgil, S., Prakken, H., Simari, G., Toni, F.: Introduction to structured argumentation. Argument & Computation **5**(1), 1–4 (2014)
10. Besnard, P., Hunter, A.: Elements of argumentation, vol. 47. MIT Press, Cambridge (2008)
11. Bondarenko, A., Toni, F., Kowalski, R.A.: An assumption-based framework for non-monotonic reasoning. In: Proceedings of the 2nd International Workshop, Logic Programming and Non-monotonic Reasoning, Lisbon, Portugal, June 1993. pp. 171–189 (1993)
12. Brandt, F., Conitzer, V., Endriss, U., Lang, J., Procaccia, A.D. (eds.): Handbook of Computational Social Choice. Cambridge University Press, Cambridge (2016)
13. Bratman, M.E., Israel, D.J., Pollack, M.E.: Plans and resource-bounded practical reasoning. Comput. Intell. **4**, 349–355 (1988)
14. Coste-Marquis, S., Konieczny, S., Mailly, J., Marquis, P.: On the revision of argumentation systems: Minimal change of arguments statuses. In: Principles of Knowledge Representation and Reasoning: Proceedings of the 14th International Conference, KR 2014, Vienna, Austria, July 20-24, 2014 (2014)
15. Diller, M., Haret, A., Linsbichler, T., Rümmele, S., Woltran, S.: An extension-based approach to belief revision in abstract argumentation. Int. J. Approx. Reasoning **93**, 395–423 (2018)
16. Doutre, S., Herzig, A., Perrussel, L.: A dynamic logic framework for abstract argumentation. In: Principles of knowledge representation and reasoning: Proceedings of the 14th International Conference, KR 2014, Vienna, Austria, July 20-24, 2014 (2014)
17. Doutre, S., Mailly, J.: Constraints and changes: a survey of abstract argumentation dynamics. Argument & Computation **9**(3), 223–248 (2018)
18. Dung, P.M.: On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. Artif. Intell. **77**(2), 321–358 (1995)
19. Floridi, L. (ed.): The Onlife Manifesto. Springer International Publishing, Berlin (2015)
20. García, A.J., Simari, G.R.: Defeasible logic programming: An argumentative approach. Theory and Practice of Logic Programming (TPLP) **4**(1-2), 95–138 (2004)
21. García, D.R., García, A.J., Simari, G.R.: Defeasible reasoning and partial order planning. In: 5th International Symposium Foundations of Information and Knowledge Systems, FoIKS 2008, Pisa, Italy, February 11-15, 2008, Proceedings. pp. 311–328 (2008)
22. Gärdenfors, P.: Manipulation of social choice functions. J. Econ. Theory **13**(2), 217–228 (1976)
23. Gottifredi, S., García, A.J., Simari, G.R.: Argumentation systems and agent programming languages. In: AAAI Fall Symposium: the uses of Computational Argument. pp. 27–32 (2009)
24. Jager, W., Edmonds, B.: Policy making and modelling in a complex world. In: Janssen, M., Wimmer, M.A., Deljoo, A. (eds.) Policy Practice and Digital Science: Integrating Complex Systems, Social Simulation and Public Administration in Policy Research, pp. 57-73. Springer International Publishing, Cham (2015)
25. Kuboyama, T.: Matching and Learning in Trees. University of Tokyo, Doctoral Thesis (2007)
26. Lifschitz, V.: Foundations of logic programs. In: Brewka, G. (ed.) Principles of Knowledge Representation, pp. 69-128, CSLI Pub (1996)
27. Mercuur, R., Dignum, V., Jonker, C.: The use of values for modeling social agents. In: Bai, Q., Ren, F., Zhang, M., to, T. (eds.) Proceedings of the 3rd International Workshop on Smart Simulation and Modelling for Complex Systems (2017)
28. Miceli, M., Castelfranchi, C.: A cognitive approach to values. J. Theory Soc. Behav. **19**(2), 169–193 (1989)
29. Modgil, S., Prakken, H.: The ASPIC$^+$ framework for structured argumentation: a tutorial. Argument & Computation **5**(1), 31–62 (2014)
30. Parks, L., Guay, R.P.: Personality, values, and motivation. Personal. Individ. Differ. **47**(7), 675–684 (2009)
31. Parks, L., Guay, R.P.: Personality, values, and motivation. Pers. Individ. Differ. **47**(7), 675–684 (2009)
32. Perelló-Moragues, A., Noriega, P.: Using agent-based simulation to understand the role of values in policy-making. In: Proceedings of the Social Simulation Conference 2018 (SSC2018) (In Press)
33. Van de Poel, I.: Values in engineering design. In: Meijers, A.W.M. (ed.) Handbook of the Philosophy of Science, pp. 973–1006. Elsevier (2009)
34. van de Poel, I.: Translating values into design requirements. In: Michelfelder, D.P., McCarthy, N., Goldberg, D.E. (eds.) Philosophy and Engineering: Reflections on Practice, Principles and Process, pp. 253–266. Springer, Netherlands (2013)
35. Rahwan, I., Amgoud, L.: An argumentation based approach for practical reasoning. In: AAMAS. pp. 347–354 (2006)
36. Reiss, S.: Multifaceted nature of intrinsic motivation: the theory of 16 basic desires. Rev. Gen. Psychol. **8**(3), 179–193 (2004)

37. Rizzo, A.: Ethically aligned design, version 2. https://standards.ieee.org/industry-connections/ec/ead-v1.html (2017)
38. Rokeach, M.: The Nature of Human Values. Free Press (1973)
39. Rotstein, N.D., García, A.J., Simari, G.R.: Reasoning from desires to intentions: a dialectical framework. In: AAAI. pp. 136–141 (2007)
40. Russell, S.: Provably beneficial artificial intelligence. The Next Step: Exponential Life BBVA-Open Mind (2017)
41. Schulz, C., Martin-Ortega, J., Glenk, K., Ioris, A.A.: The value base of water governance. Ecol. Econ. **131**, 241–249 (2017)
42. Schwartz, S.H.: Universals in the content and structure of values: theoretical advances and empirical tests in 20 countries. In: Advances in Experimental Social Psychology, vol. 25, pp. 1–65. Elsevier (1992)
43. Schwartz, S.H., Bilsky, W.: Toward a universal psychological structure of human values. J. Pers. Soc. Psychol. **53**(3), 550 (1987)
44. Schwartz, S.H., Bilsky, W.: Toward a theory of the universal content and structure of values: extensions and cross-cultural replications. J. Pers. Soc. Psychol. **58**(5), 878 (1990)
45. Sinnott-Armstrong, W.: Consequentialism. In: Zalta, E.N. (ed.) The Stanford Encyclopedia of Philosophy. Metaphysics Research Lab, Stanford University, winter 2015 edn (2015)
46. Velmovitsky, P.E., Briot, J., Viana, M.L., Lucena, C.: Practical reasoning in an argumentation-based decision BDI agent: a case study for participatory management of protected areas. In: The 29th International Conference on Software Engineering and Knowledge Engineering, Wyndham Pittsburgh University Center, Pittsburgh, PA, USA, July 5-7, 2017. pp. 527–530 (2017)
47. van der Weide, T.L., Dignum, F., Meyer, J.C., Prakken, H., Vreeswijk, G.: Practical reasoning using values. In: Argumentation in Multi-agent Systems, 6th International Workshop, ArgMAS 2009, Budapest, Hungary, May 12, 2009. Revised Selected and Invited Papers. pp. 79–93 (2009)
48. van der Weide, T.L., Dignum, F., Meyer, J.C., Prakken, H., Vreeswijk, G.: Arguing about preferences and decisions. In: Argumentation in Multi-agent Systems - 7th International Workshop, ArgMAS 2010, Toronto, ON, Canada, May 10, 2010 Revised, Selected and Invited Papers. pp. 68–85 (2010)
49. Wooldridge, M., van der Hoek, W.: On obligations and normative ability: towards a logical analysis of the social contract. J. Appl. Log. **3**(3-4), 396–420 (2005)

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.