Reasoning and Decision Making in an Inconsistent World

Labeled Logical Varieties as a Tool for Inconsistency Robustness

Cornelis N.J. de Vey Mestdagh and Mark Burgin

Abstract The goal of this paper is the development of foundations for robust reasoning and decision-making in pervasively inconsistent theories and deductive databases. The pervasiveness of these inconsistencies is partly inherent to the human epistemic condition (i.e. the need to make decisions on the basis of perspective bound knowledge) and partly inherent to practical limitations (e.g. incomplete knowledge). In the first case, we do not want to eliminate the inconsistencies. In the second case, we cannot eliminate them. So, in both cases, we will have to incorporate them in our descriptions of reasoning and decision making. For this reason, inconsistency handling is one of the central problems in many areas of AI. There are different approaches to dealing with contradictions and other types of inconsistency. In this paper, we develop an approach based on logical varieties and prevarieties, which are complex structures constructed from logical calculi. Being locally isomorphic to a logical calculus, globally logical varieties form a logical structure, which allows representation of inconsistent knowledge in a consistent way and provides much more flexibility and efficacy for AI than standard logical methods. Logical varieties and prevarieties are efficiently used in database theory and practice, expert systems and knowledge representation and processing. To increase efficiency, flexibility and capabilities of logical varieties and prevarieties and to model perspective bound decision making, we introduce labeling of their elements and study labeled logical varieties and prevarieties. We illustrate the viability of this by an example of a labeled logical variety, the extended Logic of Reasonable Inferences, which is and has been applied in the legal domain.

C.N.J. de Vey Mestdagh (✉)
Centre for Law & ICT, University of Groningen, Groningen, The Netherlands
e-mail: c.n.j.de.vey.mestdagh@rug.nl

M. Burgin
Department of Computer Science, UCLA, Los Angeles, USA

R. Neves-Silva et al. (eds.), *Intelligent Decision Technologies*,
Smart Innovation, Systems and Technologies 39,
DOI 10.1007/978-3-319-19857-6_36

1 Introduction

As far as we (humans) are concerned, the world *is* inconsistent. Our reproductive survival probability seems to be enhanced by our ability to perceive and process inconsistent data establishing inconsistent (alternative, mutually exclusive) projections and using them to make the more beneficial choices. If this was not the case, we would not be here to reflect upon our inconsistency handling abilities as evolution theory and the anthropic principle suggest. The argument that this ability is an independent feature of our cognition can be countered by the observation that perceiving inconsistency in a consistent world would just result in confusion and therefore be counter-*re*productive. The time dimension imposes constant change upon us. If this change *is* predetermined, then there would be only one consistent universe. Actually, this change can be brought about by humans, who are often able to foresee alternative futures caused by their actual actions. They can even experience a feeling of choice preceding the actual action because they dispose of a palette of imaginable alternative actions and, at least in their minds, they have the power to decide which one of these alternatives to select and perform. From our perspective every transition from one state to another therefore presupposes a selection of the next state from a set of accessible possibilities. To describe our universe, we therefore need a descriptive model of knowledge of the universe that incorporates the perception of the alternatives and some tentative outcomes of the selection process. Even if the selection process is predetermined, we need a description of the antecedents (that are called considerations if constituted by conscious cognitive processes) that led to the derived conclusion. It is also important to be able to explain this process to ourselves and to others, because we will have to communicate about it and to cooperate with each other in its application to be able to make collective decisions. The more we know about and share these antecedents the more we perceive them as *facts*, while the less we know about them and the less we share them the more we are inclined to call them *opinions*. An advanced scientific description of relations between antecedents and corresponding conclusions is called a theory. The situation is even more complicated because a state of mind itself can also play the role of a possible future. Indeed, when we want to describe changes of our states of mind, we consider a number of possible states of mind (e.g. opinions) to which we come, in making our decisions and in adhering to certain opinions.

In the next section we will formulate the *theoretical* requirements for the description of our inconsistency handling capabilities.

2 Reasoning and Decision Making in an Inconsistent World

Humans are able to imagine and reason about different futures (i.e. to describe their distinct characteristics and the relations between these characteristics) by perceiving relations between their (mental) actions, making selections of possible futures, reasoning about relations between themselves as individuals and some of these futures and coming into a state in which this relation is narrowed down to one possible future, which then becomes the present. Classical logic does not cover and usually denies this human capacity. Some modern logics try to incorporate this process into their scope (see Sect. 3). However, modern logics like belief revision and default reasoning still deny the fact that the state of inconsistency is not transient but permanent - there is no decisive factor which renders a perception (a fact or opinion) into the universal knowledge - while these logics consider knowledge to be inherently universal. Other modern logics like probabilistic and fuzzy logics have the same limitations and even more, they deny the presence of the context dependent factors that constitute opinions and that are used in the selection of the individual consequences of these opinions. In particular, the mean of all perceived or even possible contexts is not a realistic prediction of the future context in an actual case, while these logics consider knowledge to be valid in one common (or even universal) context. To meet these requirements for the description of our cognition, we need two things:

1. A description of our cognition that allows reasoning about (mutually exclusive) alternative actions and situations.
2. A description of our cognition that allows tagging the alternatives and situations, e.g., describing their identity and context, in order to be able to explain the connection between alternative antecedents (considerations/arguments) and alternative imagined futures (decisions for a certain perspective or opinion).

As we will see in Sect. 3, the most common forms of modeling decision making in logic (like belief revision and default reasoning; probabilistic and fuzzy logics) are not suited to realistically model reasoning and decision making in an inconsistent domain of knowledge. Logical varieties do solve the first limitation of these logics: they allow for the preservation of all individual opinions [17, 18, 25, 26] (Burgin and de Vey Mestdagh 1191). Labeled logics solve the second limitation of these logics: they allow for the identification of individual opinions and enable us to reason about these opinions. Labeled logical varieties, introduced and studied in this paper, combine advantages of both approaches essentially extending the power of logic in solving problems in all practical and theoretical areas since they both resolve the epistemic need to conserve inconsistency as the practical need to handle inconsistency even when it cannot be eliminated.

The previous argument can be illustrated most easily by taking a knowledge domain that is situated on the opinion end of the perception (fact-opinion) continuum. The legal knowledge domain is such a domain. Not only is its foundation based on the existence of different normative opinions, but it also does not presuppose the aspiration for a permanent (universal) resolution of these differences. In practice, only local and temporal legal solutions enable humans to make a move allowing them immediately to reconsider their previous decision when necessary. The rejected opinion is a part of the (context of) the selected opinions (see Sect. 7). Judicial and other social laws are not laws of nature. They allow for discretion (e.g., for balancing specific interests), qualification (e.g., they have an open texture), interpretation (e.g., they are vague) and individualization (e.g., they can be instantiated by a personal local temporal context). Even if the contents of my opinion are exactly the same as your opinion, it is still a unique opinion, which has another potential effect upon the outcome of a reasoning process. So, reasoning in the legal domain demands non explosive means of describing and processing alternatives. Furthermore it demands the preservation of opinions, even if they are locally and or temporally or even on average defeated. Finally it demands the possibility to tag each opinion and to describe its context.

The most obvious way in which the context of an opinion plays a role in reasoning is when an individual or a group tries to decide which opinion should prevail in order to be able to make an opinionated move. In other words, to make a personal local temporal decision, we need to be able to reason about opinions and their contexts. For both purposes, the opinions have to be identified. Criteria for making decisions between opinions do use this identification to compare opinions. To make things more complex: even the decision criteria are opinions and therefore alternative decision criteria compete when we make a decision. Mathematicians and logicians immediately recognize the recursion, which takes away the burden of modeling each of these levels of decision making separately. An extended example of such a decision making process including the contexts, the tags for opinions (valid, excludes, preferred) and their semantics and the decision rules (legal principles/metarules) and derivation rules (ranking principles) is given in de Vey Mestdagh and Hoepman [81]. We will develop a decision logic on the basis of the class this example belongs to and show its efficacy using inter alia this example.

In the next section we will formulate the *formal* requirements for the description of our inconsistency handling capabilities.

3 Approaches to Dealing with Inconsistent Knowledge

There is an active current research in the area of inconsistent knowledge, comprising various directions and approaches [25, 26]. Although inconsistencies bothered logicians from the time of Aristotle, the first logical systems treating contradictions appeared only in the 20th century. At first, they had the form of multivalued logics developed by Vasil'év and Łukasiewicz. Then the first relevant logics were built by Orlov. However, their work did not make any impact at the time and the first logician to have developed a formal paraconsistent logic was a student of Łukasiewicz, Jaśkowski [48]. Starting from this time, a diversity of different paraconsistent logics, including fuzzy logics, multivalued logics and relevant logics, has been elaborated (cf. [73]).

The perspective-bound character of information and information processing often results in natural inconsistency coming from different perspectives or from a faulty perception or from faulty information processing, such as processing on the basis of incomplete knowledge from a single perspective. As a result, now many understand that contradiction handling is one of the central problems in AI. Inconsistent knowledge/belief systems exist in many areas of AI, such as distributed knowledge base and databases, defeasible reasoning, dynamic expert systems, merging ontologies, ontology evolution, knowledge transition from one formalism to another, and belief revision.

There are three basic approaches to dealing with inconsistency in knowledge systems. The first one, which we call the *restoration approach*, is aimed at restoring consistency of an inconsistent knowledge system, e.g., a database [69]. The most popular direction in the first approach is called *coherence-based methodology* [60, 61]. It is performed in two steps:

1. Selection of one or several consistent subsystems in an inconsistent knowledge system;
2. Application of the classical inference on these subsystems.

Examples of systems that select one consistent subsystem are possibilistic logic [8], linear ordering [61], Adjustment and Maxi-adjustment [86, 87]. Examples of techniques that select several consistent subsystems are acceptable subbases [70], preferred subbases [14], Papini's revision function [63] and lexicographical approach [9, 50, 51].

Another way to restore consistency is to exclude those formulas (statements) that cause contradictions. This is achieved by utilizing logics that allow dynamical changes in the process of their functioning. Examples of such strategies are non-monotonic logics [53, 55, 56] and belief revision [29, 34, 39, 66, 84].

A *non-monotonic logic* is a formal logic that learning a new piece of knowledge can transform the set of what is known aimed at contradiction elimination. A non-monotonic logic can handle various reasoning tasks, such as *default reasoning* when consequences may be derived only because of lack of evidence of the contrary, *abductive reasoning* when consequences are deduced as most plausible explanations and belief revision when new knowledge may contradict old beliefs and some beliefs that cause contradictions are excluded [55, 53].

The second approach, which we call the *tolerance approach*, is to tolerate inconsistency [12] by using non-classical logics and including an inconsistent knowledge system into such a logic. As a result, inconsistency is treated at a higher level [78]. Examples of such methods are paraconsistent logics [13, 28, 67, 72] and argumentation [2, 10, 31].

The third direction, which we call the *structuring approach*, is based on implicit or explicit utilization of logical varieties, quasi-varieties and prevarieties [16–18].

In comparison with non-monotonic logics, which form the base for the first approach, logical varieties, quasi-varieties and prevarieties provide tools for preserving all points of view, approaches and positions even when some of them taken together lead to contradiction. Due to their flexibility, logical varieties, quasi-varieties and prevarieties allow treating any form of logical contradictions in a rigorous and consistent way.

Paraconsistent logics, which form the base for the second approach, are inferentially weaker than classical logic; that is, they deem *fewer* inferences valid. Thus, in comparison with paraconsistent logics, logical varieties, quasi-varieties and prevarieties allow utilization of sufficiently powerful means of logical inference, for example, deductive rules of the classical predicate calculus. Besides, Weinzierl [85] explains why paraconsistent reasoning is not acceptable for many real-life scenarios and other approaches are necessary. In addition, paraconsistent logics attempts to deal with contradictions in a discriminating way, while logical varieties, quasi-varieties and prevarieties treat contradictions and other inconsistencies by a separation technique.

Although conventional logical systems based on logical calculi have been successfully used in mathematics and beyond, they have definite limitations that often restrict their applications. For instance, the principal condition for any logical calculus is its consistency. At the same time, knowledge about large object domains (in science or in practice) is essentially inconsistent [16, 62, 79, 80]. From this perspective, Partridge and Wilks [64] write, "because of privacy and discretionary concerns, different knowledge bases will contain different perspectives and conflicting beliefs. Thus, all the knowledge bases of a distributed AI system taken together will be perpetually inconsistent." Consequently, when conventional logic is used for formalization, it is possible to represent only small fragments of the object domain. Otherwise, contradictions appear.

To eliminate these limitations in a logically correct way, logical prevarieties and varieties were introduced [16]. Logical varieties represent the natural development of logical calculi, being more advanced systems of logic, and thus, they show the direction in which mathematical logic will inevitably go. Including logical calculi as the simplest case, logical varieties and related systems offer several advantages over the conventional logic:

1. Logical varieties, prevarieties and quasi-varieties give an exact and rigorous structure to deal with all kinds of inconsistencies
2. Logical varieties allow modeling/realization of all other approaches to inconsistent knowledge. For instance, it is possible to use any kind of paraconsistent logics as components of logical varieties. In [16], it is demonstrated how logical varieties realize non-monotonic inference.
3. Theoretical results on logical varieties provide means for more efficient application of logical methods to problems in different areas (e.g. the application of the Logic of Reasonable Inferences (a logical variety) to represent and process contradicting opinions in the legal domain as described below).
4. Logical varieties allow partitioning of an inconsistent knowledge system into consistent parts and to use powerful tools of classical logic for reasoning.

5. Logical varieties allow utilization of different kinds of logic (multi-functionality) in the same knowledge system. For instance, it is possible to use a combination of the classical predicate calculus and non-monotonic calculus to represent two perspectives one of which is based on complete knowledge and the other on incomplete knowledge.
6. Logical varieties allow separation of different parts in a knowledge system and working with them separately.
7. Logical varieties provide means to reflect change of beliefs, knowledge and opinions without loss of previously existed beliefs, knowledge and opinions even in the case when new beliefs, knowledge and opinions contradict to what was before. In [18], they are applied to temporal databases.

These qualities of logic varieties are especially important for normative, in particular, legal, knowledge because this knowledge consists of a collection of formalized systems, a collection of adopted laws, a collection of existing traditions and precedents, and a collection of people's opinions. In addition, in the process of functioning, normative (legal) knowledge involves a variety of situational knowledge, beliefs and opinions. To analyze and use this diversity, it is necessary to have a flexible system that allows one to make sense of all different approaches without discarding them in an attempt to build a unique consistent system. To formalize these characteristics of normative knowledge, a form of a logical variety, the **Logic of Reasonable Inferences** (LRI) was developed [79]. The LRI was subsequently used as specification for the implementation of a *knowledge based system shell*, **Argumentator** [80]. This shell has consequently been used to acquire and represent legal knowledge. The resulting legal knowledge based system has been successfully used to test the empirical validity of the theory about legal reasoning and decision making modeled by the LRI [80].

It is interesting that several other systems used for inconsistency resolution, e.g., Multi-Context Systems [85], are also logical varieties and prevarieties. For instance, bridge rules used in Multi-Context Systems for non-monotonic information exchange are functions that glue together components of a logical variety or prevariety.

The goal of this work is the development of logical tools for working with inconsistency in knowledge systems. That is why we do not consider here specific approaches to inconsistency in data sets and databases. This is an important area and various means are used to perform data mining on inconsistent data. It was demonstrated that approaches based on *rough set theory* are especially useful for this task (cf. [42–45, 59, 65, 83]), as well as approaches based on *fuzzy set theory* (cf. [49, 71]). However, it is necessary to mention that data sets are model logical varieties and prevarieties (see Sect. 5) and it is possible to apply techniques based on model logical varieties and prevarieties to building tools for working with inconsistency in data sets and databases.

In the next sections we will describe labeling in logic (Sect. 4), logical varieties (Sect. 5) and the function of labels in logical varieties as a formal approach to handle inconsistency (Sect. 6), which overcomes the limitations of the other approaches described in this section.

4 Labeling in Logic and Named Sets

Labels are a kind of names and labeling is a specific naming. Namely, a *label* is a name of an object attached to this object. In a similar way, labeling is a process of naming studied in named set theory [22], and namely, it is a process of attaching the label to the object. This informal conception is formalized in the following way.

Let us consider a logical language L and a labeling structure A.

Definition 4.1 A *labeling* of the logical language L in the structure A is a partial mapping $l: L \to A$.
According to this definition, the mapping l assigns labels to logical formulas from L.

Treating labels in the context of names, we see that names, naming and named sets came to logic from the very beginning. To understand this, we start with the definition of a named set.

Definition 4.2 A *named set* (or a *fundamental triad*) **X** is the structure that consists of three components $\mathrm{X} = (X, f, I)$.

Each component of a named set (fundamental triad) plays its unique role and has a specific name. Namely, X is called the *support*, I is called the *component of names* or *set of names* and f is called the *naming correspondence* of the named set **X**.

This construction is prevalent in mathematics in general and in mathematical logic in particular. At the same time, *named sets* are abounding in many areas, in particular, in the legal domain. An example is elaborated on in Sect. 7.

Traditionally, mathematical logic is divided into three parts: *formal* (usually, axiomatic) *theories* (such as axiomatic set theories or formal arithmetic), *proof theory*, and *model theory*.

A formal theory includes a *logical calculus* and *logical semantics*. A *logical calculus* consists of a set A of initial statements called *axioms*, the *deductive system* as a set r of rules for correct inference, and a set T of deduced statements called *theorems*. *Logical semantics* studies inner interpretation (or truth evaluation) of logical systems for codification of the meaning of logical formulas in the form of truth-conditions, or possible truth conditions.

Proof theory studies the syntax of logic, that is, logical languages and means of logical inference, mostly, deduction. In this area, the main logical structures are propositions, predicates, logical operations (such as conjunction or disjunction), logical formulas, logical transformations (such as deduction or induction), logical calculi and syntactic or deductive logical varieties and prevarieties.

Model theory studies outer interpretations of logical systems in general structures called models.

At first, names and named sets were used only in model theory. For instance, statements of Aristotle's syllogistics contained variables and as it was demonstrated, variables are named sets.

A *variable* $\boldsymbol{x}$ is a named set (fundamental triad) $x = (x, r, D_x)$ where the support is the name of the variable, the set of names form the *domain* D_x of the variable $\boldsymbol{x}$, and correspondence r called the *interpretation mapping* connects the name x with the domain of this variable.

Names and consequently named sets play an imperative role in logic. For instance, Alonzo Church explains the importance of names for mathematical logic, contemplating the concept of name as one of the basic terms used in logic [27]. At the beginning of his book, Church clarifies the main concepts of logic. At first, he elucidates what *logic* is as a discipline. Then he makes clear what a *name* is in logic, taking into account only proper names and following the theory of (proper) names developed by Frege [33]. Only after this exposition, Church goes to such concepts as variables, constants, functions, propositions and so on. Thus, according to Church, the concept of a name is the first after the concept of logic. Shoenfield [75] also describes how names for individuals and expressions are constructed and used. To indicate importance of names in logic, Smullian [77] includes the word *name* in the title of his book, which provides a popular and informal introduction to logic.

Much later labels and corresponding named sets appeared in logical semantics with the advent of fuzzy logic. Namely, the truth value of a formula (statement) instead of taking two values {true, false} as in classical logics acquired multiple values from the interval [0, 1]. Each of these values is a number from the interval [0, 1], which means that the formula (statement) is true only to some degree. Thus, this degree c is a label, which displays the degree of truthfulness.

Fuzzy logics have been extensively studied, having many applications [5, 46, 58].

In proof theory, labels and corresponding named sets appeared in the labeled deductive systems introduced and studied by Gabbay [36]. *Labeled logics* and *labeled deductive systems* form a new and actively expanding direction in logic (cf. [37, 82]). Labeled logics use *labeled signed formulas* where labels (names of the formulas) are taken from information frames. As the result, the set of formulas in a labeled logic becomes an explicit named set, the support of which consists of logical formulas, while the set of names is an information frame, i.e., the system of labels. The derivation rules act on the labels as well as on the formulae, according to certain fixed rules of propagation. It means that derivation rules are morphisms (mappings) of the corresponding named sets.

Application of logic to programming gave birth to the direction called logic programming. Naturally, labeled logics brought forth labeled logic programs [6].

Special tools have been developed for working with names in logic, and naming always involves building new or transforming existing named sets (cf. Sect. 5.2). For instance, Gabbay and Malod [38] extend predicate modal and temporal logics introducing a special predicate $W(x)$, which names the world under consideration. Such a naming allows one to compare the different states the world (universe or individual) can be in after a given period of time, depending on the alternatives taken on the way. As Gabbay and Malod [38] remark, the idea of naming the worlds and/or time points goes back to Prior [68].

Labels play *two roles* in logic:

1. *Substitution labels* are used to represent logical formulas.
2. *Attached labels* provide additional information about logical formulas.

Substitutions are used in all kinds and types of logic. For instance, when we have a metaformula in the sense of [20] or logical schema in the sense of [54] $A \wedge B \rightarrow A$ of the propositional calculus, in it, A and B are substitution labels because it is possible to substitutes by arbitrary propositions obtaining a true formula of the classical propositional calculus. In essence, any variable in logic is a substitution label.

It is necessary to remark that a substitution is a kind of renaming [54], while renaming is one of the basic operations studied in the theory of named sets [22]. As a matter of fact, renaming is abounding in many areas, in particular, in the legal domain (see the example in Sect. 7).

There are *three modalities* of attached labels:

1. *Control labels* impact, e.g., control or direct, logical inference.
2. *Coordinated labels* are transformed in the process of inference according to deduction rules.
3. *Independent labels* and logical formulas are independently transformed in the process of inference.

Control labels have become rather popular in logic. For instance, in labeled deductive systems, consequence is defined by proof rules that work with both logical formulas and their labels [36], while in traditional logical systems, consequence is defined by proof rules only over the logical formulas. Examples of the use of different types of attached labels are given in Sect. 7.

There are also many possible functions of attached labels: Often attached labels form some structure, e.g., an algebra or information frame.

A labeled logical language L consists of pairs (φ, a) where φ is a logical formula and a is an attached label. Note that, in general, one label can consist of several other labels. For instance, if $a_1, a_2, \ldots, a_n$ are labels, then it is possible to treat the n–tuple $(a_1, a_2, \ldots, a_n)$ as another label.

Applications of logic in software technology on the one hand, e.g., in logic programming, and applications of computers to logic, e.g., in automated theorem proving, bring us to labeled inference languages. As Manna and Waldinger [54] write, proofs in formal theories that use simplistic application of deduction rules are usually far too expensive to be of practical value. So, it is necessary to build efficient proof strategies. Here labels come into the play. They can be used for labeling deduction rules in strategies, for naming proof strategies and organizing interaction between proof strategies.

Besides, proof strategies are specific types of second-order algorithms [23, 24]. As a result, labels can be used in second-order algorithms for the same purposes as in proof strategies.

In the next section we will describe the application of labels to logical varieties.

5 Structuring Inconsistency by Means of Labeled Logical Quasi-Varieties, Prevarieties, and Varieties

There are different types and kinds of logical varieties and prevarieties: *deductive or syntactic* varieties and prevarieties, *functional or semantic* varieties and prevarieties and *model or pragmatic* varieties and prevarieties [17]. Semantic logical varieties and prevarieties are formed by separating those parts that represent definite semantic units and determining logical semantics in these parts. In contrast to syntactic and semantic varieties, model varieties are essentially formalized structures. The structure of data in a distributed relational database is an example of logical model varieties.

Syntactic varieties, quasi-varieties and prevarieties are built from logical calculi as buildings are built from blocks. That is why, we, at first remind the concept of a logical calculus.

Let us consider a labeled logical language L and a labeled inference language R.

Definition 5.1 A *syntactic* or *deductive labeled logical calculus* is a triad (a named set) of the form $\boldsymbol{C} = (A, H, T)$ where $H \subseteq R$ and $A, T \subseteq L, A$ is the set of axioms, H consists of inference rules (rules of deduction) by which from axioms the theorems of the calculus are deduced, and the set of theorems T is obtained by applying algorithms/procedures/rules from H to elements from A.

Note that there are not only logical calculi. For instance, there are differential calculus and integral calculus in mathematical analysis.

Let **K** be a class of syntactic labeled logical calculi, K be a set of labeled inference rules, and **F** be a class of partial mappings from L to L.

Definition 5.2 A triad **M** = (A, K, M), where A and M are sets of expressions that belong to the labeled logical language L (A consists of axioms and M consists of theorems) and H is a set of inference rules, which belong to K, is called:

(1) a *labeled projective syntactic* (**K,F**)-*quasi-prevariety* if there exists a set of labeled logical calculi $\boldsymbol{C_i} = (A_i, H_i, T_i)$ from **K** and a system of mappings f_i: $A_i \to L$ and g_i: $M_i \to L (i \in I)$ from **F** in which A_i consists of axioms and M_i consists of some (not necessarily all) theorems of the logical calculus $\boldsymbol{C}_i$, i.e., $M_i \subseteq T_i$ and for which the equalities $A = \cup_{i \in I} f_i(A_i)$, $K = \cup_{i \in I} H_i$ and $M = \cup_{i \in I} g_i(M_i)$ are valid (it is possible that $\boldsymbol{C_i} = \boldsymbol{C_j}$ for some $i \neq j$).

(2) a *labeled syntactic* **K**-*quasi-prevariety* if it is a projective syntactic (**K,F**)-quasi-prevariety where all mappings f_i and g_i that define **M** are inclusions, i.e., $A = \cup_{i \in I} A_i$ and $M = \cup_{i \in I} M_{i.}$

(3) a *labeled projective syntactic* (**K,F**)-*quasi-variety* with the depth k if it is a projective syntactic (**K,F**)-quasi-prevariety and for any $i_{1,} i_2, i_3, \ldots, i_k \in I$ either the intersections $\cap_{j=1^k} f_{ij}(A_{ij})$ and $\cap_{j=1^k} g_{ij}(T_{ij})$ are empty or there exists a calculus $C = (A, H, T)$ from **K** and projections $f \to \cap_{j=1^k} f_{ij}(A_{ij})$ and $g: N \to \cap_{j=1^k} g_{ij}(M_{ij})$ from **F** where $N \subseteq T$.

(4) a *labeled syntactic* **K**-*quasi-variety* with the depth k if it is a projective syntactic (**K**,**F**)- quasi-variety with depth k in which all mappings f_i and g_i that define **M** are bijections on the sets A_i and $M_{i,}$ correspondingly, and for all intersections (cf. Definition 4.2(3)), mappings f and g are bijections.
(5) a (*full*) *labeled projective syntactic* (**K**,**F**)-*quasi-variety* if for any $k > 0$, it is a projective syntactic (**K**,**F**)-*quasi-variety* with the depth k.
(6) a (*full*) *labeled syntactic* **K**-*quasi-variety* if for any $k > 0$, it is a **K**-quasi-variety with the depth k.
(7) a *labeled projective syntactic* (**K**,**F**)-prevariety if it is a projective syntactic (**K**,**F**)-quasi-prevariety in which $M_i = T_i$ for all $i \in I$.
(8) a *labeled syntactic* **K**-*prevariety* if it is a syntactic **K**-quasi-variety in which $M_i = T_i$ for all $i \in I$.
(9) a *labeled projective syntactic* (**K**,**F**)-*variety with the depth* k if it is a projective syntactic (**K**,**F**)-quasi-prevariety with the depth k in which $M_i = T_i$ for all $i \in I$.
(10) a *labeled syntactic* **K**-*variety with the depth* k if it is a projective syntactic (**K**, **F**)-quasi-variety with depth k in which $M_i = T_i$ for all $i \in I$ and for all intersections (cf. Definition 4.2(3)), $N = T$.
(11) a (*full*) *labeled projective syntactic* (**K**,**F**)-*variety* if for any $k > 0$, it is a projective syntactic (**K**,**F**)-variety with the depth k.
(12) a (*full*) *labeled syntactic* **K**-*variety* if for any $k > 0$, it is a **K**-variety with the depth k.

We see that the collection of mappings f_i and g_i makes a unified system called a prevariety or quasi-prevariety out of separate logical calculi $\boldsymbol{C}_{i,}$ while the collection of the intersections $\cap_{j=1^k} f_{ij}(A_{ij})$ and $\cap_{j=1^k} g_{ij}(T_{ij})$ makes a unified system called a variety out of separate logical calculi $\boldsymbol{C}_{i.}$ For instance, mappings f_i and g_i allow one to establish a correspondence between norms/laws that were used in one country during different periods of time or between norms/laws used in different countries.

The main goal of syntactic logical varieties is in presenting sets of formulas as a structured logical system using logical calculi, which have means for inference and other logical operations. Semantically, it allows one to describe a domain of interest, e.g., a database, knowledge of an individual or the text of a novel, by a syntactic logical variety dividing the domain in parts that allow representation by calculi.

In comparison with varieties and prevarieties, logical quasi-varieties and quasi-prevarieties are not necessarily closed under logical inference. This trait allows better flexibility in knowledge representation.

Definition 5.3 The calculi $\boldsymbol{C}_i$ used in the formation of the labeled syntactic quasivariety, prevariety or variety **M** are called *components* of **M**.

Note that it is possible that different components of a syntactic prevariety (variety) **M** have different deduction (inference) rules and even different types of deduction (inference) rules. For instance, some components have conventional, i.e., recursive, inference rules, while other components have super-recursive inference rules, which are more powerful [19]. If we consider the formal arithmetic *A*, then the famous result of Gödel asserts impossibility of proving consistency of the arithmetic *A* by recursive, inference rules [41]. However, later consistency of the arithmetic *A* was proved using more powerful superrecursive inference rules [1, 40, 74].

An example of a logical variety is a distributed database or knowledge base, each component of which consists of consistent knowledge/data. Then components of this knowledge/database are naturally represented by components of a logical variety. Moreover, very often data in a database have various labels. For example, in temporary database data elements have timestamps [21]. Besides, in one knowledge base different object domains may be represented. In these domains, some object may have properties that contradict properties of an object from another domain.

One more example of naturally formed logical varieties is the technique *Chunk and Permeate* built by [15]. This technique suggests to begin reasoning from inconsistent premises and proceeds by separating the assumptions into consistent theories (called *chunks* by the authors). These chunks are components of the logical variety shaped by them. After this, appropriate consequences are derived in one component (chunk). Then those consequences are transferred to a different component (chunk) for further consequences to be derived. This is exactly the way how logical varieties are used to realize and model non-monotonic reasoning [16]. Brown and Priest suggest that Newton's original reasoning in taking derivates in the calculus, was of this form.

An interesting type of logical varieties was developed in artificial intelligence and large knowledge bases. As Amir and McIlraith write [4, 52, 57], there is growing interest in building large knowledge bases of everyday knowledge about the world, comprising tens or hundreds of thousands of assertions. However working with large knowledge bases, general-purpose reasoning engines tend to suffer from combinatorial explosion when they answer user's queries. A promising approach to grappling with this complexity is to structure the content into multiple domain- or task-specific partitions. These partitions generate a logical variety comprising the knowledge base content. For instance, a first-order predicate theory or a propositional theory is partitioned into tightly coupled sub-theories according to the language of the axioms in the theory. This partitioning induces a graphical representation where a node represents a particular partition or sub-theory and an arc represents the shared language between sub-theories.

The technology of content partitioning allows reasoning engines to improve the efficiency of theorem proving in large knowledge bases by identifying and exploiting the implicit structure of the knowledge [4, 52, 57]. The basic approach is to convert a graphical representation of the problem into a tree-structured representation, where each node in the tree represents a tightly-connected sub-problem, and the arcs represent the loose coupling between sub-problems. To maximize the effectiveness of partition-based reasoning, the coupling between partitions is minimized, information being passed between nodes is reduced, and local inference within each partition is also minimized.

Additional advantage of partitioning is a possibility to reason effectively with multiple knowledge bases that have overlap in content [4].

The tools and methodology of content partitioning and thus, implicitly of logical varieties are applied for the design of logical theories describing the domain of robot motion and interaction [3]. Other examples are given in Burgin and de Vey Mestdagh [25, 26].

Concepts of logical varieties and prevarieties provide further formalization for local logics of Barwise and Seligman [7], many-worlds model of quantum reality of Everett [30, 32], and pluralistic quantum field theory of Smolin related to the many-worlds theory [76].

In the next section we will describe the functions of labels in logical varieties

6 Labels and Their Functions in Logical Varieties

Labels can perform multiple functions in logical varieties:

1. Labels that indicate the source of information (knowledge) represented by a component of a logical variety, assuming that each component of a logical variety is connected to a source of information (knowledge). For instance the source of a particular observation or derivation, a specific legal source (legislation, case law, expert opinion, etc.);
2. Labels that show the reliability of (the source of) information (knowledge). For instance an accepted fact finding method, a valid legal source (see Sect. 7), the reliability of a witness, etc.;
3. Labels that refer the logical formulas to the problem-knowledge which these formulas represent. For instance, items, e.g. formulas, related to the same source (a person, a body, etc.) can have the same label (see Sect. 7);
4. Labels that estimate the relevance of logical formulas to the corresponding problems;
5. Labels that estimate the importance of information (knowledge) to the related conjectures;
6. Labels that refer the logical formulas to the topics (attributes) knowledge about which these formulas represent;
7. Labels that estimate the relevance of logical formulas to the corresponding topics (attributes);
8. Labels that describe relations between objects and people involved in the process;
9. Labels that estimate the importance of information (knowledge);
10. Labels that refer the logical formulas to conjectures (see for example, Sect. 7);
11. Labels that estimate importance of information (knowledge) to the related conjectures;

12. Labels that estimate probability of conjectures represented by logical formulas in a logical variety;
13. Labels that show the degree of support of a conjecture or belief. Utilization of such labels is described in [35, 47];
14. Labels that describe connections and relations of a formula;
15. Labels that indicate where information represented by a formula is stored;
16. Labels that show time of information creation and/or acceptance (see for example, Sect. 7);
17. Labels that show time of information validity.

Labeling in information systems allows users to discern when the same information represented by a formula comes from different sources. Thus, it is possible that one and the same formula has different labels. Such a formula can belong to different components or even to the same component of the syntactic logical quasivariety, prevariety or variety. For instance, when a label indicates the reliability of information sources and the source 1 has the reliability 5, while the source 1 has the reliability 7, then a formula φ coming from the source 1 has the reliability label 5, while coming from the source 1 has the reliability label 7.

Let us consider some types of labeling.

Definition 6.1 A labeling $l: M \to A$ of a syntactic logical quasivariety (prevariety or variety) $\mathrm{M} = (A, K, M)$ is called:

(1) *structured* if formulas from different components of **M** have different labels;
(2) *discerning* if different elements from M have different labels.
(3) *univalent* if any element from M has, at most, one label.

An example of structured labeling of logical quasivarieties is a locally stratified knowledge base or database in the approach of Benferhat and Garcia [11] who suggest handling inconsistent knowledge bases by local stratification based on priorities, such as a preference relation or reliability relation, which are assigned to knowledge (data) items, allowing to provide more meaningful and reliable information to the user. Priority values are used as labels of formulas, while priorities stratify the database or its part forming a logical variety, in which a component is determined by a fixed level (value) of priority. Thus, priority labeling is structured.

Proposition 6.1. A syntactic quasivariety $\mathbf{M} = (A, K, M)$ has a structured labeling if it is discrete.

Corollary 6.1. A syntactic prevariety (variety) $\mathbf{M} = (A, K, M)$ has a structured labeling if it is discrete.

An important operation in labeled logical structures, such as logical deductive systems, logical calculi and logical varieties, is relabeling or changing labels. It is a particular case of renaming studied in the theory of named sets [22].

Labels allow logicians to connect semantic logical varieties studied in [17] and syntactic logical varieties studied here.

Let us consider a labeled logical language L and a structure Q. Note that it is possible to treat a logical language without labels as a specific labeled logical language, namely, as logical language formulas of which have empty labels.

Definition 6.2. A named set $\mathrm{Sem} = (L, c, Q)$ where $c: L \to Q$ is a partial mapping is called a *semantics* in the language L with the scale Q.

The most popular semantic scale is the truth-scale $Q = \{\mathrm{T}, \mathrm{F}\}$ used in classical logics. Another popular semantic scale $Q = \{\mathrm{T}, \mathrm{F}, \mathrm{U}\}$ where U means *undefined* or *unknown* is used in intuitionistic logics.

Semantics in a labeled logical language allows us to define semantic calculi. Let us consider a subset E of the structure Q.

Definition 6.3. A labeled semantic E-calculus $\mathbf{C}$ is the name set $\mathbf{C} = (C, c, E)$ in which the subset C of L defined in the following way

$$C = \{\varphi \in L; c(\varphi) \in E\}$$

Let $\mathbf{K}$ be a class of semantic labeled logical calculi, Q be a semantic scale, and $\mathbf{F}$ be a class of partial mappings.

Definition 6.4. (a) A *projective labeled semantic logical* ($\mathbf{K}$, $\mathbf{F}$)-*prevariety* $\mathbf{M}$ = (M, k, E) is the named set components of which are defined in the following way:

There are semantic labeled logical calculi $\mathbf{C}_i = (C_i, c_i, E_i)$ from $\mathbf{K}$ and three systems of mappings $c_i: C_i \to E_i$, $k_i: E_i \to E$ and $g_i: M_i \to L$ from $\mathbf{F}$ such that

$$M = \cup_{i \in I} g_i(M_i), E = \cup_{i \in I} k_i(E_i) \text{ and } k = \cup_{i \in I} k_i c_i$$

(b) A *labeled semantic logical* $\mathbf{K}$-*prevariety* is a projective labeled semantic logical ($\mathbf{K}$, $\mathbf{F}$)-prevariety $\mathbf{M}$ = (M, k, E) in which all g_i and k_i are inclusions, i.e., $M = \cup_{i \in I} M_i$ and $E = \cup_{i \in I} E_i$.

Definition 6.5. A *labeled semantic logical* $\mathbf{K}$-*variety* is a labeled semantic logical $\mathbf{K}$-prevariety $\mathbf{M}$ = (M, k, E) in which all intersections $\cap_{j=1^k} M_{ij}$ are empty or there exists a calculus $\boldsymbol{C} = (\boldsymbol{C}, \boldsymbol{c}, \boldsymbol{H})$ from $\mathbf{K}$ such that $\cap_{j=1^k} M_{ij} = C, \cap_{j=1^k} E_{ij} = H$ and each restriction of c on M_{ij} coincides with c_{ij}.

Definition 6.6. Labeling $l: M \to L$ of a syntactic logical quasivariety, prevariety or variety $\mathbf{M} = (A, K, M)$ is called *semantic* if there is a semantic logical prevariety or variety $\mathbf{M}_\mathrm{s} = (N, k, E)$ such that $N = M$ and $l(\varphi) = k(\varphi)$ for any formula φ from M.

Definitions imply the following result.

Proposition 6.2 For any syntactic logical variety $\mathbf{M} = (A, K, M)$ and any semantic logical variety $\mathbf{M}_\mathrm{s} = (N, k, E)$, there is a semantic labeling of $\mathbf{M}$ defined by $\mathbf{M}_\mathrm{s.}$

Control labels can essentially change properties of logical calculi and logical varieties.

By the famous Gödel theorem, the first-order formal arithmetic (Peano arithmetic) A is undecidable [41], and in particular, it has formulas that are true in the natural semantics of A but are not recursively provable (recursively deducible). An appropriate labeling can change this situation.

Proposition 6.3 There is a binary labeling l of all formulas in the language of A and a deduction operator such that all true in A formulas and only true in A formulas are recursively provable.

Proof. Let us take the standard semantic scale of labeling $Q = \{\mathrm{T}, \mathrm{F}\}$ and a formula φ from the language $L(A)$ of the arithmetic A. We attach the label T to φ when φ is true in A and the label F when φ is false in A. After doing this, we define the following deduction rule

$$(\varphi, d) \Vdash (\psi, c) \text{ if and only if } c = d = T$$

We can see that the labeled calculus (D, $\{\Vdash\}$, H) where the set of axioms D consists of single axiom $(n = n, \mathrm{T})$ and H is the set of all theorems deduced from D by means of the deduction operator $\Vdash$ is complete and consistent assuming that A is consistent.

Proposition is proved.

In the above example, labeling extends provability making deduction more powerful because in traditional logics without labeling, it is impossible to build sufficiently powerful complete and consistent calculus as the first Gödel theorem tells us. At the same time, it is possible not only to extend power of deduction but also to restrict provability by means of an appropriate labeling as the following example demonstrates.

Let us consider a formal theory P with the classical deduction operator $\vdash$, which includes application of standard deduction rules, such as $(\varphi, \varphi \to \psi) \Rightarrow \psi$, and the set of axioms A, and take the classical semantic scale $Q = \{\mathrm{T}, \mathrm{F}\}$, We attach the label T to a formula φ from the language $L(P)$ of the theory P when φ is true in A and the label F when φ is false in A. For a set of formulas D and a formula φ from $L(P)$, we define another deduction rule

$D \Vdash (\varphi, d)$ if and only if $D \vdash \varphi$ in P and there are no formulas in D with the label F.

This gives us a syntactic calculus $\mathbf{C} = (A, \{\Vdash\}, T)$ where T are all theorems deduced from A by means of the deduction operator $\Vdash$. In this calculus, false formulas do not participate in the inference and thus, even when A or T has contradictions (false formulas), it does not cause explosion, i.e., T is not necessarily equal to L. Note that in a general case, $\mathbf{C}$ is a kind of a paraconsistent logical calculus.

Various other applications of control labels to enhancement of inference processes are described in the area of labelled deductive systems [36, 37].

In the next section we will apply a species of a labeled logic variety (the extended Logic of Reasonable Inferences) to an example from the legal domain.

7 The Logic of Reasonable Inferences, an Implementation of a Labeled Logical Variety in the Legal Domain

Our knowledge of the world is always perspective bound and therefore fundamentally inconsistent, even if we agree to a common perspective, because this agreement is necessarily local and temporal due to the human epistemic condition. The natural inconsistency of our knowledge of the world is particularly manifest in the legal domain (de Vey Mestdagh et al. 2011).

In the legal domain, on the object level (that of case facts and opinions about legal subject behavior), alternative (often contradicting) legal positions compete. All of these positions are a result of reasoning about the facts of the case at hand and a selection of preferred behavioral norms presented as legal rules. At the meta-level meta-positions are used to make a choice for one of the competing positions (the solution of an internal conflict of norms, a successful subject negotiation or mediation, a legal judgement). Such a decision based on positions that are inherently local and temporal is by definition also local and temporal itself. The criteria for this choice are in most cases based on legal principles. We call these legal principles *metaprinciples* because they are used to evaluate the relations between different positions at the object level.

To formalize this natural characteristic of (legal) knowledge we developed the Logic of Reasonable Inferences (LRI, [79]). The LRI is a logical variety that handles inconsistency by preserving inconsistent positions and their antecedents using as many independent predicate calculi as there are inconsistent positions [25, 26]. The original LRI was implemented and proved to be effective as a model of and a tool for knowledge processing in the legal domain [80]. In order to be able to make inferences about the relations between different positions (e.g. make local and temporal decisions), labels were added to the LRI. In de Vey Mestdagh et al. (2011) formulas and sets of formulas are named and characterized by labelling them in the form (A_i, H_i, P_i, C_i). These labels are used to define and restrict different possible inference relations (***Axioms*** A_i and ***Hypotheses*** H_i, i.e. ***labeled signed formulas*** and ***control labels***, see Sect. 4) and to define and restrict the composition of consistent sets of formulas (***Positions*** P_i and ***Contexts*** C_i). Formulas labeled A_i must be part of any position and context and therefore are not (allowed to be) inconsistent. Formulas labeled H_i can only be part of the same position or context if they are mutually consistent. A set of formulas labeled P_i represents a ***position***, i.e. a consistent set of formulas including all Axioms (e.g., ***a perspective on a world, without inferences about that world***). A set of formulas labeled C_i represents a ***context*** (a maximal set of consistent formulas within the (sub)domain and their justifications, c.f. ***the world under consideration***, see Sect. 4). All these labels can be used as predicate variables and if individualized to instantiate predicate variables and consequently as constants (***variables as named sets*** see Sect. 4). Certain metacharacteristics of formulas and pairs of formulas were finally described by labels (e.g., metapredicates like Valid, Excludes, Prefer) describing some of their legal source characteristics and their legal relations which could be used to rank the different positions externally. The ***semantics*** of these three Predicates (Valid, Exclude and Prefer) are described in de Vey Mestdagh et al. [81]. These three predicates describe the elementary relations between legal positions that are prescribed by the most fundamental sets of legal principles (i.e. principles regarding the legal validity of positions, principles regarding the relative exclusivity of legal positions even if they do not contradict each other and principles regarding the preference of one legal position over another). It was also demonstrated that the first requirement of Sect. 2, a *description of our cognition should allow for reasoning about (mutually exclusive) alternatives*, is attained by the LRI.

In this section we show that labels can be used formally to describe the ranking process of positions and contexts. With that the second requirement of Sect. 2 will be also satisfied, namely, *a description of our cognition should allow for local and temporal decisions for a certain alternative, which means without discarding the non-preferred alternatives like belief revision does and without using the mean of all alternatives like probabilistic logics do.* This extends the LRI from a logical variety that could be used to formalize the non-explosive inference of inconsistent contexts (opinions) and naming (the elements of) these contexts to a labeled logical variety, in which tentative decisions can be formally represented by using a labelling that allows for expressing the semantics of the aforementioned meta-predicates and prioritizing (***priority labelling***). We illustrate the use of these labels by examples.

In de Vey Mestdagh et al. 2011 (Sect. 4), a complex legal case is presented about a threat to the **tower of Pisa**. It would take too much space to repeat the informal and the full formal case description here. Just the necessary parts of the formal case description are repeated here. The full case description can be read at and downloaded from:

http://www.rug.nl/staff/c.n.j.de.vey.mestdagh/inconsistent_knowledge_as_a_natural_phenomenon_cnjdvm_def-en-reftobook.pdf (p. 29, Sect. 4).

As a part of this case four existing legal positions (P_i) consisting of Axioms (A_i) and Hypotheses (H_i) are presented:

P1	Subjectivist's view	: A1..A11; H1, H4(H1), H7(H1,H2)
P2	Objectivist's view	: A1..A11; H2, H4(H2), H5(H2), H6(H2,H1)
P3	Impressionist's view	: A1..A11; H3, H7(H3,H2)
P4	Subjectivist's + Impressionist's	: A1..A11; H1, H3, H4(H1), H7(H1,H2), H7(H3, H2)

It is shown that these positions can be formally reduced to two mutually inconsistent contexts (general perspectives C1 justified by P1, P3, P4 and C2 justified by P2):

C1	Reasonable inference	: Criminal_attempt(One_of_us,Toppling_Top)
	Justification	: P1, P3, P4
C2	Reasonable inference	: ¬Criminal_attempt(One_of_us,Toppling_Top)
		Valid(H1), Valid(H2), Excludes(R2,R1)
	Justification	: P2

There are several possible perspectives on the criteria used to decide between these two contexts. In this Section four of these tentative '***decision models***' and a combination of them are described. We use the term 'tentative', because the decision model does not discard contexts but just adds inferences about their preference to the contexts. The decision models are extensions of and are applied to the theorems of the case presented in de Vey Mestdagh et al. 2011.

Decision model 'Prefer the most valid position'

Prefer a minimal position which contains more *uninstantiated* valid basic formulas over a position that contains less *uninstantiated* valid basic formulas (the minimal position predicate and the uninstantiated formula predicates are left out to preserve readability).

A6 Precedent(H1), Precedent(H2)
A7 Complies_with_section_1_Penal_Code(H2)
H4 $\text{Precedent}(H_i) \rightarrow \text{Valid}(H_i)$ [*meta – rule precedent validity*]
H5 $\text{Complies\_with\_section\_1\_Penal\_Code}(H_i) \rightarrow \text{Valid}(H_i)$ [*meta – rule principle of legality*]
H8 $\neg\ (\text{Precedent}(H_i) \vee \text{Complies\_with\_section\_1\_Penal\_Code}(H_i)) \rightarrow \neg\text{Valid}(H_i)$
H9 $\text{In}(P_i, H_i) \wedge \text{Valid}(H_i) \rightarrow \text{Valid}(P_i)$
H10 $\text{In}(P_i, H_i) \wedge \neg\text{Valid}(H_i) \rightarrow \neg\text{Valid}(P_i)$
H11 $\text{Valid}(P_i) \wedge \neg\text{Valid}(P_j) \rightarrow \text{Prefer}(P_{i,} P_j)$

On the basis of this extended set of rules hypotheses H1 and H2 are just valid and hypothesis H3 is just invalid (all hypotheses contained in de positions P1, P2, P3 and P4 presented above). As a consequence the positions P1 (just valid uninstantiated formulas), P2 (just valid uninstantiated formulas) and P4 (valid and invalid uninstantiated formulas) are preferred over P3 (just invalid uninstantiated formulas). The double evaluation of P4 (valid and invalid) causes a split of contexts.

Decision model 'Prefer the legal position'

Prefer a position which contains more uninstantiated legal basic formulas (i.e. a formula based on legislation) over a position that contains less uninstantiated legal basic formulas:

A7 Complies_with_section_1_Penal_Code(H2)
H5 $\text{Complies\_with\_section\_1\_Penal\_Code}(Hi) \rightarrow \text{Valid}(Hi)$ [*meta – rule principle of legality*]
H12 $\text{Complies\_with\_section\_1\_Penal\_Code}(Hi) \rightarrow \text{Legal}(Hi)$
H13 $\neg\ \text{Complies\_with\_section\_1\_Penal\_Code}(Hi) \rightarrow \neg\text{Legal}(Hi)$

H14 $\mathrm{In(Pi, Hi)} \wedge \mathrm{Legal(Hi)} \rightarrow \mathrm{Legal(Pi)}$
H15 $\mathrm{In(Pi, Hi)} \wedge \neg\mathrm{Legal(Hi)} \rightarrow \neg\mathrm{Legal(Pi)}$
H16 $\mathrm{Legal(Pi)} \wedge \neg\mathrm{Legal(Pj)} \rightarrow \mathrm{Prefer(Pi, Pj)}$

On the basis of this set of rules P2 (containing a legal uninstantiated formula), is preferred over P1, P3 and P4 (containing no legal uninstantiated formulas).

Decision model 'Prefer the specific position'

Prefer a position which contains the more specific uninstantiated legal basic formulas over a position that contains the more general uninstantiated legal basic formulas.

A8 $\mathrm{Specialis(H2, H1)}$
H6 $\mathrm{Specialis}(\mathrm{H_i}, \mathrm{H_j}) \rightarrow \mathrm{Excludes}(\mathrm{H_i}, \mathrm{H_j})$ $[\mathit{meta - rule\ lex\ specialis}]$
H18 $\mathrm{In}(\mathrm{P_i}, \mathrm{H_i}) \wedge \mathrm{In}(\mathrm{P_j}, \mathrm{H_j}) \wedge \mathrm{H6\ Excludes}(\mathrm{H_i}, \mathrm{H_j}) \rightarrow \mathrm{Prefer}(\mathrm{P_{i,}}\, \mathrm{P_j})$

On the basis of this set of rules P2 (containing the specialis H2) is preferred over P1 and over P4 (both containing the generalis H1)

Decision model 'Prefer the position which soothes public alarm' (terrorism induced policies)

A decision model which is popular in this time of perceived terror threats is 'Prefer a position which soothes public alarm'.

A5 $\mathrm{Recent\_public\_alarm(Toppling\_Top)}$
A9 $\mathrm{Soothes\_public\_alarm(H1, Toppling\_Top)}$
A10 $\neg\mathrm{Soothes\_public\_alarm(H2, Toppling\_Top)}$
A11 $\mathrm{Soothes\_public\_alarm(H3, Toppling\_Top)}$
H7 $\mathrm{Recent\_public\ \_alarm(y)} \wedge \mathrm{Soothes\_public\_alarm}(\mathrm{H_i},\ \mathrm{y}) \wedge \neg\mathrm{Soothes\_public\_}$
$\mathrm{alarm}(\mathrm{H_j},\ \mathrm{y}) \rightarrow \mathrm{Prefer}(\mathrm{H_i}, \mathrm{H_j})$ $[\mathit{meta - rule\ terrorism\ induced\ policies}]$
H19 $\mathrm{In}(\mathrm{P_i}, \mathrm{H_i}) \wedge \mathrm{In}(\mathrm{P_j}, \mathrm{H_j}) \wedge \mathrm{H7\ Prefer}(\mathrm{H_i}, \mathrm{H_j}) \rightarrow \mathrm{Prefer}(\mathrm{P_{i,}}\, \mathrm{P_j})$

On the basis of this set of rules P1, P3 and P4 (based on one or two of the soothing rules H1 and H3) are preferred over P2 (based on the non-soothing rule H2).

Decision model 'Prefer the context with the most preferred position as a justification'

The LRI adds the inferred preference relations by definition to the contexts (cf. de Vey Mestdagh et al. 2011). The two contexts C1 and C2 described in de Vey Mestdagh et al. 2011 split into four contexts: two (equal to the originals) extended with valid formulas P4 and two (C3 and C4, equal to the originals C1 and C2) with invalid formulas P4. A simple decision model just counts the number of times a position is preferred and prefers the context(s) with the most preferred position as a justification. Position P1 is preferred 3 times in the four decision models, position

P2 6 times, position P3 once and position P4 twice. In the contexts the occurrence of the preference for position P2 is reduced to 4 times and the other preferences stay the same (less than 4), because equal inferences are just registered once (so a more elegant decision model would use the number of preferred positions based on the different decision models, but the LRI does not allow for labelling the decision models because they do not belong to the formal part of the logic).

The following decision rule will conclude to prefer C2 (and C4) over C1 (and C2):
Most_preferred(Pi) ∧ In_justification(Ci,Pi) ∧ ¬ In_justification(Cj,Pi) → Prefer (Ci,Cj)

The reasonable inference in both C2 and C4 is that it is no criminal attempt if one of us tries to topple the Tower of Pisa just by pushing it, which is actually the dominant (but not the only!) opinion of the majority of Dutch courts. Note that this preference is just added to (in this case) all contexts and that none of the contexts is discarded because of the preference. Actually the logic does not decide but just suggests a decision (that a human or computer can process further). This is exactly what we want. Interestingly enough in a more complex case the LRI can derive multiple preference relations between contexts including contradictory ones. In that case metapredicates can be added to evaluate the preference relation between these preference relations (if we had the brains this would end up in an infinite recursion, but fortunately we lawyers as all humans are fairly limited in our capacity to think new metalevels of decision making through).

8 Conclusion

In this paper, we have described the theory and application of labelled logical varieties. It was demonstrated that labelled logical varieties provide means for overcoming some severe restrictions of common logical approaches to inconsistency. It allows to preserve all knowledge positions instead of discarding them (like belief revision) and instead of replacing them by an aggregate such as a mean (probabilistic approaches).

Labelled logical varieties allow us to formally describe and to model the underlying reality of:

1. *Reasoning about (mutually exclusive) alternatives without discarding or adapting them*
2. *Deciding locally and temporally for a certain alternative*

The preservation of all alternatives does satisfy the epistemic need to conserve inconsistency (natural inconsistent perspectives) and in doing that can en passant satisfy the practical need to handle unwanted inconsistency when it cannot be eliminated.

We think that this is in accord with the human epistemic condition, which does not know and does not accept universal and infinite closure under any circumstance.

Further research will focus on the extension and application of labelled logical varieties in knowledge systems, particularly in the legal domain. It would also be useful to develop techniques based on model logical varieties and prevarieties for building tools that would allow working with inconsistency in data sets and databases.

References

1. Ackermann, W.: Zur Widerspruchsfreiheit der Zahlentheorie. Math. Ann. **117**, pp. 162–194 (1940)
2. Amgoud, L., Cayrol, C.: On the acceptability of arguments in preference-based argumentation. In: Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence (UAI'98), pp. 1–7 (1998)
3. Amir, E.: Dividing and Conquering Logic. Ph.D. thesis, Stanford University, Computer Science Department, 2002
4. Amir, E., McIlraith, S.: Partition-based logical reasoning for first-order and propositional theories. Artif. Intell. **162**(1/2), 49–88 (2005)
5. Bandemer, H., Gottwald, S.: Fuzzy Sets, Fuzzy Logic, Fuzzy Methods With Applications. Wiley, New York (1996)
6. Barker, C.: Continuations in natural language. In: Thielecke, H. (ed.), Proceedings of the 4th Continuations Workshop, pp. 55–64. Technical report CSR-04-1, School of Computer Science, University of Birmingham (2004)
7. Barwise, J., Seligman, J.: Information Flow: The Logic of Distributed Systems. Cambridge Tracts in Theoretical Computer Science 44, Cambridge University Press (1997)
8. Benferhat, S., Dubois, D., Prade. H.: Representing default rules in possibilistic logic. In: Proceedings of the 3rd International Conference of Principles of Knowledge Representation and Reasoning (KR'92), pp. 673–684 (1992)
9. Benferhat, S., Cayrol, C., Dubois, D., Lang, J., Prade, H.: Inconsistency management and prioritized syntax-based entailment. In: Proceedings of the 13th International Joint Conference on Artificial Intelligence (IJCAI'93), pp. 640–645 (1993)
10. Benferhat, S., Dubois, D., Prade, H.: How to infer from inconsistent beliefs without revising? In: Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI'95), pp. 1449–1455 (1995)
11. Benferhat, S., Garcia, L.: Handling locally stratified inconsistent knowledge bases. Stud. Logica. **70**, 77–104 (2002)
12. Bertossi, L.E., Hunter, A., Schaub, T. (eds.): Inconsistency Tolerance. LNCS, vol. 3300, Springer, Heidelberg (2005)
13. Besnard, P., Hunter, A.: Quasi-classical logic: non-trivializable classical reasoning from inconsistent information. In: Proceedings of ECSQARU'95, LNAI, vol. 946, pp. 44–51 (1995)
14. Brewka, G.: Preferred subtheories: an extended logical framework for default reason. In: Proceedings of the 11th Int Joint Conference on Artificial Intelligence (IJ CAI'89), pp. 1043–1048 (1989)

15. Brown, B., Priest, G.: Chunk and permeate: a paraconsistent inference strategy, part I: the infinitesimal calculus. J. Philos. Logic **33**(4), 379–388 (2004)
16. Burgin, M.: Logical methods in artificial intelligent systems (in Russian). In: Vestnik of the Computer Society, No. 2, pp. 66–78 (1991)
17. Burgin, M.: Logical varieties and covarieties. (in Russian) In: Methodological and Theoretical Problems of Mathematics and Information and Computer Sciences, pp. 18–34. Kiev (1997)
18. Burgin, M.: Logical Tools for Program Integration and Interoperability. In: Procedings of the IASTED International Conference on Software Engineering and Applications, pp. 743–748, MIT, Cambridge (2004)
19. Burgin, M.: Super-recursive algorithms. Monographs in computer science. Springer, New York (2005). ISBN 0-387-95569-0
20. Burgin, M.: Languages, Algorithms, Procedures, Calculi, and Metalogic. Preprint in Mathematics LO/0701121, 31 pp. (electronic edition: http://arXiv.org) (2007)
21. Burgin, M.: structural organization of temporal databases. In: Proceedings of the 17th International Conference on Software Engineering and Data Engineering (SEDE-2008), ISCA, pp. 68–73, Los Angeles, California (2008)
22. Burgin, M.: Theory of Named Sets. Nova Science Publishers, New York (2011)
23. Burgin, M., Debnath, N.: Reusability as design of second-level algorithms. In: Proceedings of the ISCA 25th International Conference "Computers and their Applications" (CATA-2010), ISCA, pp. 147–152. Honolulu, Hawaii (2010)
24. Burgin, M., Gupta, B.: Second-level algorithms, superrecursivity, and recovery problem in distributed systems. Theor. Comput. Syst. **50**(4), 694–705 (2012)
25. Burgin, M., de Vey Mestdagh, C.N.J.: The Representation of Inconsistent Knowledge in advanced knowledge based systems. In: Koenig, A., Dengel, A., Hinkelmann, K., Kise, K., Howlett, R.J., Jain, L.C. (eds.) Knowlege-Based and Intelligent Information and Engineering Systems, vol. 2, pp. 524–537. Springer, ISBN 978-3-642-23862-8 (2011)
26. Burgin, M., de Vey Mestdagh, C.N.J.: Consistent structuring of inconsistent knowledge. In: Journal of Intelligent Information Systems, pp. 1–24. Springer US, Sept 2013
27. Church, A.: Introduction to Mathematical Logic. Princeton University Press, Princeton (1956)
28. Da Costa, N.C.A.: Calcul propositionnel pour les systemes formels inconsistants. Compte Rendu Academie des Sciences (Paris) **257**, 3790–3792 (1963)
29. Dalal, M.: Investigations into a theory of knowledge base revision: preliminary report. In: Proceedings of the Seventh National Conference on Artificial Intelligence (AAAI'88), pp. 475–479 (1988)
30. DeWitt, B.S.: The Many-Universes interpretation of quantum mechanics. In: Foundations of Quantum Mechanics, pp. 167–218. Academic Press, New York (1971)
31. Dung, P.M.: On the acceptability of arguments and its fundamental role in non-monotonic reasoning, logic programming and n-person games. Artif. Intell. **77**, 321–357 (1995)
32. Everett, H.: 'Relative State' formulation of quantum mechanics. Rev. Mod. Phys. **29**, 454–462 (1957)
33. Frege, F.L.G.: Über Sinn und Bedeutung. In: Zeitschrift für Philosophie und philosophische Kritik, pp. 25–50 (1892)
34. Friedman, N., Halpern, J.Y.: A knowledge-based framework for belief change, part II: revision and update. In: Proceedings of the Fourth International Conference on the Principles of Knowledge Representation and Reasoning (KR'94), pp. 190–200 (1994)
35. Gabbay, D.M.: Labelled deductive systems and the informal fallacies. In: Van Eemeren F.H. et al. (eds.). Proceedings of the 3rd International Conference on Argumentation, vol. 2: Analysis and Evaluation, Sponsored by ISSA, International Society for the Study of Argumentation, pp. 308–319 (1994)
36. Gabbay, D.M.: Labelled deductive systems. Oxford Logic Guides, vol. 33, Clarendon Press/Oxford Science Publications, Oxford (1996)

37. Gabbay, D.M., D'Agostino, M.: A generalization of analytic deduction via labelled deductive systems part 1: basic substructural logics. J. Autom. Reasoning **13**, 243–281 (1994)
38. Gabbay, D.M., Malod, G.: Naming worlds in modal and temporal logic. J. Log. Lang. Inf. **11**, 29–65 (2002)
39. Gärdenfors, P., Rott, H.: Belief revision. In: Handbook of Logic in Artificial Intelligence and Logic Programming, vol. 4, pp. 35–132. Oxford University Press (1995)
40. Gentzen, G.: Die Widerspruchfreiheit der reinen Zahlentheorie. Math. Ann. **112**, 493–565 (1936)
41. Gödel, K.: Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme I. Monatsh. Math. Phys. **38**(1), 173–198 (1931–1932)
42. Gogoi, P., Das, R., Borah, B., Bhattacharyya, D.K.: Efficient rule set generation using rough set theory for classification of high dimensional data. Int. J. Smart Sens. Ad Hoc Netw. (IJSSAN) **1**(2), 13–20 (2011)
43. Greco, S., Matarazzo, B., Slowinski, R., Stefanowski, J.: Variable consistency model of dominance-based rough sets approach. In: Rough Sets and Current Trends in Computing. Lecture Notes in Computer Science, vol. 2005, pp. 170–181 (2001)
44. Grzymala-Busse, J.W.: Knowledge acquisition under uncertainty-A rough set approach. J. Intell. Rob. Syst. **1**, 3–16 (1988)
45. Grzymala-Busse, J.W.: Rough set theory with applications to data mining. Real World Appl. Comput. Intell. Stud. Fuzziness Soft Comput. **179**, 221–244 (2005)
46. Hájek, P.: Metamathematics of Fuzzy Logic. kluwer Academic publishers, Dordrecht (1998)
47. Hunter, A., Liu, W.: Knowledge base stratification and merging based on degree of support. In: Quantitative and Qualitative Approaches to Reasoning and Uncertainty (ECSQARU'09), LNCS, vol. 5590, pp. 383–395. Springer (1994)
48. Jaśkowski, S.: Rachunek zdań dla systemów dedukcyjnych sprzecznych. Studia Societatis Scientiarun Torunesis (Sectio A) **1**(5), 55–77 (1948)
49. Kong, H., Xue, G., He X., Yao, S.: A proposal to handle inconsistent ontology with fuzzy OWL. In: 2009 WRI World Congress on Computer Science and Information Engineering, pp. 599–603 (2009)
50. Lehmann, D.: Another perspective on default reasoning. Ann. Math. Artif. Intell. **15**, 61–82 (1995)
51. Lehmann, D.: Belief revision, revised. In: Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI'95), pp. 1534–1540 (1995)
52. MacCartney, B., McIlraith, S.A., Amir, A., Uribe, T.: Practical partition-based theorem proving for large knowledge bases. In: Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence (IJCAI-03), pp. 89–96 (2003)
53. Makinson, D.: Bridges from Classical to Nonmonotonic Logic. College Publications (2005)
54. Manna, Z., Waldinger, R.: The Deductive Foundations of Computer Programming. Addison-Wesley, Boston/New York/Toronto (1993)
55. Marek, W., Truszczynski, M.: Nonmonotonic Logics: Context-Dependent Reasoning. Springer, New York (1993)
56. McDermott, D., Doyle, J.: Non-monotonic logic I. Artif. Intell. **25**, 41–72 (1980)
57. McIlraith, S., Amir, E.: Theorem proving with structured theories. In: Proceedings of the 17th International Joint Conference on Artificial Intelligence, (IJCAI'01), pp. 624–631 (2001)
58. McNeill, D., Freiberger, P.: Fuzzy Logic. Simon and Schuster (1993)
59. Mollestad, T., Skowron, A.: A rough set framework for data mining of propositional default rules. In: Proceedings of the 9th International Symposium on Foundations of Intelligent Systems, pp. 448–457 (1996)
60. Nebel, B.: Belief revision and default reasoning: syntax-based approaches. In: Proceedings of the Second International Conference on the Principles of Knowledge Representation and Reasoning (KR'91), pp. 417–428 (1991)

61. Nebel, B.: Base revision operations and schemes: semantics, representation and complexity. In: Proceedings of the Eleventh European Conference on Artificial Intelligence (ECAI'94), pp. 341–345 (1994)
62. Nguen, N.T.: Inconsistency of knowledge and collective intelligence. Cybern. Syst. **39**(6), 542–562 (2008)
63. Papini, O.: A complete revision function in propositional calculus. In: Proceedings of the 10th European Conference on Artificial Intelligence (ECAI'92) (1992)
64. Partridge, D., Wilks, Y.: The Foundations of Artificial Intelligence. Cambridge University Press, Cambridge (1990)
65. Pawlak, Z.: Rough set and intelligent data analysis. Int. J. Inform. Sci. **147**, 1–12 (2002)
66. Pollock, J.L., Gillies, A.: Belief revision and epistemology. Synthese **122**, 69–92 (2000)
67. Priest, G., Routley, R., Norman, J. (eds.): Paraconsistent Logic: Essays on the Inconsistent. Philosophia Verlag, München (1989)
68. Prior, A.N.: Past, Present and Future. Clarendon Press, Oxford (1967)
69. Rescher, N., Manor, R.: On inference from inconsistent premisses. Theor. Decis. **1**(2), 179–217 (1970)
70. Rescher, N.: Plausible reasoning: an introduction to the theory and practice of plausibilistic inference (1976)
71. Resconi, G., Hinde, C.J.: Active sets, fuzzy sets and inconsistency. In: IEEE International Conference on Fuzzy Systems (FUZZ), pp. 1–8, Barcelona, Spain (2010)
72. Ross, T.J.: Fuzzy Logic with Engineering Applications. McGraw-Hill P. C. (1994)
73. Routley, R., Plumwood, V., Meyer, R.K., Brady, R.T.: Relevant Logics and Their Rivals. Atascadero, Ridgeview, CA (1982)
74. Schütte, K.: Beweistheorie. Springer, Berlin (1960)
75. Shoenfield, J.R.: Mathematical Logic. Addison-Wesley, Reading (2001)
76. Smolin, L.: The Bekenstein bound, topological quantum field theory and pluralistic quantum field theory. Penn State preprint CGPG-95/8-7; Los Alamos Archives preprint in physics, gr-qc/9508064, electronic edition: http://arXiv.org (1995)
77. Smullian, R.: What is the Name of this Book?. Prentice Hall, Englewood Cliffs (1978)
78. Toulmin, S.: The Uses of Argument. Cambridge University Press (1956)
79. de Vey Mestdagh, C.N.J., Verwaard, W., Hoepman, J.H.: The logic of reasonable inferences. In: Breuker, J.A., de Mulder, R.V., Hage, J.C. (eds.) Legal Knowledge Based Systems, Model-based legal reasoning, Proceedings of the 4th annual JURIX Conference on Legal Knowledge Based Systems, pp. 60–76. Vermande, Lelystad (1991)
80. de Vey Mestdagh, C.N.J.: Legal expert systems. Experts or Expedients? The Representation of Legal Knowledge in an Expert System for Environmental Permit Law. In: Ciampi, C., Marinai, E. (eds.) The Law in the Information Society, Conference Proceedings on CD-Rom, Firenze, 8 pp (1998)
81. de Vey Mestdagh, C.N.J., Hoepman, J.H.: Inconsistent knowledge as a natural phenomenon: the ranking of reasonable inferences as a computational approach to naturally inconsistent (legal) theories. In: Dodig-Crnkovic, G. & Burgin, M. (eds.) Information and Computation, pp. 439–476. World Scientific, New Jersey (2011)
82. Viganò, L., Volpe, M.: Labeled natural deduction systems for a family of tense logics. In: Demri, S., Christian S., Jensen, C.S. (eds.) 15th International Symposium on Temporal Representation and Reasoning (TIME 2008), pp. 118–126, University of Quebec, Montreal, Canada, 16–18 June 2008. IEEE Computer Society (2008)
83. Wang, G., Liu, F.: The inconsistency in rough set based rule generation. In: Rough Sets and Current Trends in Computing, Lecture Notes in Computer Science, vol. 2005, pp. 370–377 (2001)
84. Wassermann, R.: An algorithm for belief revision. In: Proceedings of the 7th International Conference of Principles of Knowledge Representation and Reasoning (KR'2000) (2000)
85. Weinzierl, A.: Comparing inconsistency resolutions in multi-context systems. In: Slavkovik, M. (ed.) Student Session of the European Summer School for Logic, Language, and Information, pp. 17–24 (2010)

86. Williams, M.A.: Transmutations of knowledge systems. In: Proceedings of the 4th International Conference of Principles of Knowledge Representation and Reasoning (KR'94), pp. 619–629 (1994)
87. Williams, M.A.: A practical approach to belief revision: reason-based change. In: Proceedings of the 5th International Confeence of Principles of Knowledge Representation and Reasoning (KR'96), pp. 412–421 (1996)