Artificial Intelligence 259 (2018) 52–90

Contents lists available at ScienceDirect

Artificial Intelligence

www.elsevier.com/locate/artint

Measuring inconsistency with constraints for propositional knowledge bases

Kedian Mu

School of Mathematical Sciences, Peking University, Beijing 100871, PR China

A R T I C L E I N F O

Article history:
Received 16 May 2016
Received in revised form 12 February 2018
Accepted 20 February 2018
Available online 2 March 2018

Keywords:
Inconsistency
Constraints
Causality
Bipartite graph

A B S T R A C T

Measuring inconsistency has been considered as a necessary starting point to understand the nature of inconsistency in a knowledge base better. For practical applications, however, we often have to face some constraints on resolving inconsistency. In this paper, we propose a graph-based approach to measuring the inconsistency for a propositional knowledge base with one or both of two typical types of constraints on modifying formulas. Here the first type of constraint, called the hard constraint, describes a pair of sets of formulas such that all the formulas in the first set should be protected from being modified on the condition that all the formulas in the second set must be modified in order to restore the consistency of that base, while the second type, called the soft constraint, describes a set of pairs of formulas that are not allowed to be modified together. At first, we use a bipartite graph to represent the relation between formulas and minimal inconsistent subsets of a knowledge base. Then we show that such a graph-based representation allows us to characterize the inconsistency with constraints in a concise way. Based on this characterization, we thus propose measures for the degree of inconsistency and for the responsibility of each formula for the inconsistency of a knowledge base with constraints, respectively. Finally, we show that these measures can be well explained based on Halpern and Pearl's causal model and Chockler and Halpern's notion of responsibility.

1. Introduction

Inconsistency is one of the important issues in knowledge and information systems. Techniques for inconsistency handling have been given much attention in the community of artificial intelligence and its application domains. Recently, measuring inconsistency has been considered as a useful way of better understanding the nature of inconsistency, and then provides a promising starting point to promote the process of inconsistency handling in knowledge and information systems in many applications such as requirements engineering [22,23], network security and intrusion detection [18,19], and medical experts systems [29]. A growing number of inconsistency measures have been proposed so far. Hunter et al. classified these inconsistency measures into two categories, i.e., base-level measures and formula-level ones [8]. Roughly speaking, the base-level measures focus on describing how inconsistent a knowledge base is, while the formula-level ones aim to grasp the responsibility (or contribution) of each formula of a knowledge base for the inconsistency in that base.

In particular, minimal inconsistent subsets of a knowledge base are attractive to measuring inconsistency in applications of syntax-based inconsistency handling [7]. Here a minimal inconsistent subset (MIS for short) refers to an inconsistent

E-mail address: mukedian@math.pku.edu.cn.

https://doi.org/10.1016/j.artint.2018.02.003

subset without an inconsistent proper subset. Please note that minimal inconsistent subsets of a knowledge base may be considered as a natural characterization of inconsistency in that base, since we only need to remove one formula from each minimal inconsistent subset in order to restore the consistency of that base [30]. In this sense, inconsistency measures built upon minimal inconsistent subsets may help us link measuring inconsistency with resolving inconsistency in a natural way. Along this line, minimal inconsistent subsets have been used to develop base-level inconsistency measures [7,8,24,26,9] as well as formula-level measures [7,8,21,9].

Theoretically, removing any formula of a minimal inconsistent subset can break the minimal inconsistent subset. Then removing a minimal part that contains at least one formula for each minimal inconsistent subset may be considered as a potential proposal for resolving the inconsistency. However, not all of such proposals for resolving inconsistency are interesting to a given practical application. For example, in requirements engineering, changing different sets of software requirements may involve different stakeholders with their own demands and benefit expectations, then a final proposal is often a trade-off between different stakeholders [22]. Generally, domain experts and users often have a good sense of which proposals are more appropriate for resolving inconsistency in that application. They may also have a sense of "conditions" for acceptable proposals for the application domain, which would rule out the proposals that they know would not be of interest. Thus, a good heuristic is to specify such intuition or expectations on resolving inconsistency as *constraints* to facilitate inconsistency handling in practical applications. For example, the integrity constraints in merging an inconsistent multiset of information from different sources are used to characterize the behavior that any expected merging operator has to obey [15]. In requirements engineering, essential requirements are not allowed to be involved in any feasible proposal for resolving contradictions between requirements in general case [27].

Besides that such constraints can help us to select proposals we want, they may be pushed deep into the process of inconsistency handling to improve the effectiveness of related activities involved in resolving inconsistency. In particular, incorporating such constraints in measuring inconsistency can help us establish more practical relations between measuring inconsistency and resolving inconsistency.

In this paper, we focus on two typical kinds of constraints on modifying formulas of a knowledge base. A constraint of the first type is a pair of sets of formulas such that all the formulas in the first set should be protected from being modified, on the condition that all the formulas in the second must be changed in resolving inconsistency. We call such a constraint a *hard constraint*. Generally, a hard constraint represents some partial compromise on resolving inconsistency in practical applications. In contrast, a constraint of the second type is given as a set of pairs of formulas that are not allowed to be modified together in resolving inconsistency. We call such a constraint a *soft constraint*.

As mentioned above, minimal inconsistent subsets can be considered as a promising starting point to connect inconsistency measures and syntax-based inconsistency handling. However, selecting formulas that have to be modified to break minimal inconsistent subsets from their own respective perspectives does not necessarily lead to an effective proposal for resolving inconsistency. Intuitively, overlaps between minimal inconsistent subsets are often of interest to breaking all the minimal inconsistent subsets by removing as few formulas as possible. Then such two overlapping minimal inconsistent subsets should be associated with each other when we want to break them. Along this line, given a minimal inconsistent subset, any two minimal inconsistent subsets that have their own respective overlaps with the minimal inconsistent subset will be also associated with each other even if the two subsets have no overlap. More generally, two minimal inconsistent subsets M and M' are associated with each other if there is a chain of minimal inconsistent subsets $M_1, M_2, \cdots, M_n$ with $M_1 = M$ and $M_n = M'$ such that M_{i+1} and M_i overlap each other for all $i = 1, 2, \cdots, n-1$. Evidently, these associations bring a partition of the set of minimal inconsistent subsets such that only minimal inconsistent subsets in the same part (called a cluster in this paper) are associated with one another (but not necessarily overlap), moreover, they should be broken as a whole instead of from their own respective perspectives. Then we need to know how the minimal inconsistent subsets in a cluster are associated with each other in order to break them together. That is, we need to capture both the interconnection relation between minimal inconsistent subsets and formulas that play important roles in the interconnection so as to help us understand the role of each formula in causing the inconsistency from a perspective of causality. To address this, we construct a bipartite graph for a knowledge base, which represents both the inner structure of each minimal inconsistent subset and the interconnection between minimal inconsistent subsets due to their overlaps. Then we show that such a graph-based representation allows us to incorporate the two types of constraints in characterizing inconsistency in a concise way. Based on this incorporation, we propose approaches to measuring inconsistency with one or both of the two types of constraints, respectively. In particular, both our base-level and formula-level measures can be reduced to their respective corresponding measures presented in [21] when there is no constraint. Some intuitive logical properties and complexity issues for these inconsistency measures are also discussed, respectively.

On the other hand, it is often expected that an inconsistency measure under development will be interpretable. That is, inconsistency measures may need to be tied in with some specific interpretations that can help us gain an intuitive insight into the inconsistency. Generally, causality plays an important role in analyzing and resolving inconsistency in practical applications. Formulas identified as causes of the inconsistency of a knowledge base are of interest when we take some actions for restoring the consistency of that base. Thus causality-based explanations for inconsistency measures can help us establish a significant linkage between inconsistency measuring and inconsistency resolving. However, causality is a subtle topic in itself. The *counterfactual dependence* is considered as a common ground of many attempts presented to define causality from Hume to the present [2]. Informally speaking, A counterfactually depends on B if it is the case that if B had not happened, then A would not have happened. Recently, Halpern and Pearl's structural causal model [5], one of