Repairing Socially Aggregated Ontologies Using Axiom Weakening

Daniele Porello, Nicolas Troquard, Roberto Confalonieri, Pietro Galliani, Oliver Kutz, and Rafael Peñaloza

Free University of Bozen-Bolzano, Italy
{Firstname.Lastname}@unibz.it

Abstract. Ontologies represent principled, formalised descriptions of agents' conceptualisations of a domain. In the case of a community of agents, these descriptions may differ among agents. We propose an aggregative view of the integration of ontologies based on Social Choice Theory (SCT) and Judgement Aggregation (JA). Agents may vote on statements from the ontologies, and we aim at constructing a collective, integrated ontology, that reflects the individual conceptualisations as much as possible. As several results in SCT and JA show, many attractive and widely used aggregation procedures are prone to return inconsistent collective ontologies. In this paper, we propose to solve the possible inconsistencies in the collective ontology by applying suitable weakenings of axioms derived from generalisations and specialisations of concepts in axioms belonging to minimally inconsistent subsets.

1 Introduction

Social choice theory (SCT) is a branch of economic theory that deals with the design and analysis of mechanisms for aggregating opinions of individual agents to arrive at a basis for a collective decision [10]. A ubiquitous example of such a mechanism is voting, usually intended as voting on preferences in standard social choice. Recently, the model of aggregation has been applied to judgements, or more generally to propositional attitudes, expressed in some logical setting, in an area termed Judgement Aggregation (JA) [6,16,18].

Ontologies are widely used in Knowledge Representation to provide principled descriptions of agents' knowledge, by presenting a clear formalisation of their conceptualisations. The meaning of the concepts is then represented by means of a number of axioms, which may be written in a variety of logical systems of varying expressivity [13]. With the exception of [20], the usual approaches to SCT and JA are usually applied to propositional logics, modal logics, or even more general logics, but they do not touch the problem of the possibly heterogeneous definitions of concepts used by the agents to formalise their individual conceptualisation. Understanding what is the meaning of a concept for a community of agents and deciding how to elect a common conceptualisation out of possibly conflicting ones is an interesting open problem that has several applications, for instance, in the context of political applications of SCT. Understanding what is the meaning of a concept for a community of agents is crucial for modelling electoral campaigning, where parties try to maximise their electorate by appealing to widely shareable world views. Moreover, in the case of coalition formation, it is important to assess whether viable coalitions of parties, besides sharing a possible agenda of interests, may also share the concepts that they use in order to promote their alliance to their respective electorates. Finally, concept aggregation is important in opinion pooling, in trying to define the mean opinion, which is a statement involving concepts [8].

In the context of ontology aggregation, we may think of each ontology as a voter, and these voters try to 'elect' a collective ontology that adequately and fairly represents their conceptualisations. SCT then provides the formal means to assess the suitable aggregation procedures for a given aggregation scenario, by defining a number of properties that aggregators may or may not satisfy. However, many results in SCT and JA show that a significant number of important aggregation procedures, e.g., the majority rule, fail in preserving the consistency of the individual inputs [18,20]. This means that, although we assume that all ontologies that agents submit for aggregation are consistent, the outcome of the aggregation may lead to an inconsistent collective conceptualisation. A number of strategies to circumvent inconsistency have been pursued in SCT and JA, for instance, abandoning well-known aggregators in favour of aggregators that indeed preserve consistency, or restricting the set of propositions about which the agents cast their vote to those for which consistency can be ensured.

In this paper, we follow a novel approach. We discuss well-known justified aggregation procedures that are actually used in real collective decision problems, viz absolute majority rule and quota rule, and we propose a computational viable methodology based on *axiom weakening* to repair their possibly inconsistent outcomes.

The idea behind axiom weakening is to generalise or specialise possibly conflicting concepts with concepts that are, in some sense, as close as possible to the original ones, but do not yield an inconsistency. Preventing inconsistencies by appealing to 'general' concepts, which may then be prone to agreement although they have not been voted on by any individual, has been suggested and legitimated in the literature on social choice and deliberation [7, 17, 19].

This is an important issue, and it also relates to the distinction between fine vs. coarse integration of ontologies. In the case of a coarse integration, the ontology to be constructed will always contain some of the formulas included in the individual ontologies; in the fine integration, new formulas shall be constructed. The approach in [20] provides an example of coarse integration. In this paper, we are after a viable definition of fine integration.

To resume, the contributions of this paper are as follows. We consider possible conceptualisations of agents as represented by means of ontologies written in Description Logic (DL). In particular, we focus on the basic DL $\mathcal{ALC}$ [1], which is a popular language for ontology development. Secondly, we use the methodology of SCT and JA of [20] to define a framework for ontology aggregation. Thirdly, we use refinement operators for concept generalisations and specialisations, and we apply them to repair the collective ontology by selecting adequate refinement of the axioms that caused the inconsistency.

2 Ontologies and Description logics

We take an ontology to be a set of formulas in an appropriate logical language, describing our domain of interest. Which logic we use precisely is not crucial for illustrating the proposed approach, but as much formal work on ontologies makes use of description logics (DLs), we will use these logics for all of our examples. A significant widely used basic description logic is $\mathcal{ALC}$, which is the logic we shall be working with here.

We present the basics of $\mathcal{ALC}$. For full details we refer to the literature [1]. The language of $\mathcal{ALC}$ is based on an alphabet consisting of *atomic concepts names* N_C, and *roles names* N_R. The set of *concept descriptions* is generated by the following grammar (where A represents atomic concepts and R role names):

$$C ::= A \mid \neg C \mid C \sqcap C \mid C \sqcup C \mid \forall R.C \mid \exists R.C$$

We collect all $\mathcal{ALC}$ concepts over N_C and N_R in $\mathcal{L}(\mathcal{ALC}, N_C, N_R)$. We assume a linear order $\prec_{\mathcal{ALC}}$ over $\mathcal{ALC}$ formulas. We do not need to attach any particular meaning to it, but it will be helpful for coping with non-determinism and for tie-breaking.

A *TBox* is a finite set of concept inclusions of the form $C \sqsubseteq D$ (where C and D are concept descriptions). It is used to store terminological knowledge regarding the relationships between concepts. An *ABox* is a finite set of formulas of the form $A(a)$ ("object a is an instance of concept A") and $R(a, b)$ ("objects a and b stand to each other in the R-relation").[1] It is used to store assertional knowledge regarding specific objects. The semantics of $\mathcal{ALC}$ is defined in terms of *interpretations* $I = (\Delta^I, \cdot^I)$ that map each object name to an element of its domain Δ^I, each atomic concept to a subset of the domain, and each role name to a binary relation on the domain. The truth of a formula in such an interpretation is defined in the usual manner [1]. For instance, a point $a \in \Delta^I$ belongs to the interpretation of the concept $(\forall R.C)^I$ if all elements related to a via (the interpretation of) R belong to the (interpretation of) C.

In the remainder of this paper, we restrict our attention to TBox axioms. As usual, a TBox $\mathcal{T}$ is *consistent* if it has a model, and *inconsistent* otherwise. A concept C is *satisfiable* with respect to a TBox if there exists an interpretation I of the TBox that makes C^I non-empty. A consequence relation $\models$ is defined on top of this semantics in the standard way. The relation $\models_O$ denotes the consequence relation w.r.t. an ontology O.

[1] Note that limiting the ABox to 'atomic' formulas is not a restriction, as A may be given a complex definition in the TBox.

3 Aggregating Ontologies

Consider an arbitrary but fixed finite set Φ of $\mathcal{ALC}$ TBox statements over this alphabet.[2] We call Φ the *agenda* and any set $O \subseteq \Phi$ an *ontology*. We denote the set of all those ontologies that are *consistent* by $\mathrm{On}(\Phi)$.

Let $\mathcal{N} = \{1, \ldots, n\}$ be a finite set of *agents* (or *individuals*). Each agent $i \in \mathcal{N}$ provides a consistent ontology $O_i \in \mathrm{On}(\Phi)$. An *ontology profile* is a vector $\boldsymbol{O} = (O_1, \ldots, O_n) \in \mathrm{On}(\Phi)^{\mathcal{N}}$ of consistent ontologies, one for each agent. We write $N_{\varphi}^{\boldsymbol{O}} := \{i \in \mathcal{N} \mid \varphi \in O_i\}$ for the set of agents that include φ in their ontology under profile $\boldsymbol{O}$. Our object of study are *ontology aggregators*.

Definition 1 (Ontology aggregators). *An ontology aggregator is a function* $F : \mathrm{On}(\Phi)^{\mathcal{N}} \to 2^{\Phi}$ *mapping any profile of consistent ontologies to an ontology.*

Observe that, according to this definition, the ontology we obtain as the outcome of an aggregation process needs not be consistent. Ontology aggregators that are *consistent* would be very desirable in general. Unfortunately, they also suffer certain drawbacks. The *unanimous aggregator* is one of these.

Definition 2 (Unanimous aggregator). *The unanimous aggregator is the ontology aggregator* F_{un} *mapping any given profile* $\boldsymbol{O} \in \mathrm{On}(\Phi)^{\mathcal{N}}$ *to the ontology*

$$F_{un}(\boldsymbol{O}) := O_1 \cap \cdots \cap O_n \ .$$

The unanimous aggregator indeed preserves consistency. That is, if every ontology O_j is consistent, so is $F_{un}(\boldsymbol{O})$. However, if the individual ontologies are heterogeneous enough, the unanimous aggregator is likely to provide a very poor collective ontology. Often in practice, on large enough electorate and agendas, the aggregated ontology will be empty.

At the opposite side of the spectrum, we can define the union aggregator, that accepts any piece of information provided by at least one agent.

Definition 3 (Union aggregator). *The union aggregator is the ontology aggregator* F_u *mapping any given profile* $\boldsymbol{O} \in \mathrm{On}(\Phi)^{\mathcal{N}}$ *to the ontology*

$$F_u(\boldsymbol{O}) := O_1 \cup \cdots \cup O_n \ .$$

In this case, the collective ontology is very likely to be inconsistent.

A way to balance the contributions of agents better than with the unanimous and the union aggregators, we can adapt the majority rule, which is widely applied in any political scenarios. In our setting, the majority rule is defined as follows.

[2] The finite set of TBox formulas in Φ might be all TBox formulas of a certain maximum length or the union of all TBox formulas that a given population of agents choose to include in their TBoxes.

LeftPolicy $\sqsubseteq$ RaiseWages
LeftPolicy $\sqsubseteq$ RaiseWelfare
RaiseWages $\sqcap$ RaiseWelfare $\sqsubseteq \bot$

Fig. 1. The TBox agenda of the agents.

Definition 4 (Absolute majority rule). *The absolute majority rule is the ontology aggregator F_m mapping any given profile $\boldsymbol{O} \in \mathrm{On}(\Phi)^{\mathcal{N}}$ to the ontology*

$$F_m(\boldsymbol{O}) := \{\varphi \in \Phi \mid \#N_\varphi^{\boldsymbol{O}} > \frac{n}{2}\} \ .$$

That is, under the absolute majority rule, a formula gets accepted if and only if more than half of the individual agents accept it. A simple generalisation of the majority rule provides the class of *quota* rules, where the threshold of $\frac{n}{2}$ is replaced by any threshold q.

Definition 5 ((Uniform) quota rules). *The uniform quota rule with threshold q is the ontology aggregator F_q mapping any given profile $\boldsymbol{O} \in \mathrm{On}(\Phi)^{\mathcal{N}}$ to the ontology*

$$F_q(\boldsymbol{O}) := \{\varphi \in \Phi \mid \#N_\varphi^{\boldsymbol{O}} > q\} \ .$$

The majority rule, and more generally quota rules, return a consistent ontology only on very simple agendas, i.e., on very simple ontologies [20]. In the next section we introduce an example to discuss the possible inconsistency caused by majoritarian aggregation and we informally present our strategy to solve the problem.

4 Possibly Inconsistent Collective Ontologies

The following example shows that also the absolute majority rule, which is widely used in any political scenario, is not a consistent aggregator. Our example is a simple adaptation of the *doctrinal paradox* familiar from the literature on judgement aggregation to the case of concept definitions [12, 18].

Consider three left-wing political leaders, i.e., three agents 1, 2, and 3, who must agree on what is a left policy in order to coordinate their campaigns. They vote on possible definitions of left-wing policy by casting their votes on the TBox agenda shown in Figure 1. Each individual ontology, in particular, formalises possible meanings that agents ascribe to what is a left-wing policy. Suppose that the agents vote as in Table 1.

Every individual set of axioms is consistent and the concept LeftPolicy is satisfiable in each of the individual ontologies. Agent 1, for instance, believes that a left policy must raise both the wages and the levels of welfare, accordingly this agent believes that it is possible to promote the levels of both. Agent 2 believes that a left policy only has to raise wages, not the level of welfare, as they believe that it is not possible to do both. Agent 3 believes that what counts as a left

Table 1. A voting scenario

	LeftPolicy $\sqsubseteq$ RaiseWages	LeftPolicy $\sqsubseteq$ RaiseWelfare	RaiseWages $\sqcap$ RaiseWelfare $\sqsubseteq \bot$
1	yes	yes	no
2	yes	no	yes
3	no	yes	yes
Maj.	yes	yes	yes

LeftPolicy $\sqsubseteq$ RaiseWages
LeftPolicy $\sqsubseteq$ RaiseWelfare
RaiseWages $\sqsubseteq$ ReduceInequality
RaiseWelfare $\sqsubseteq$ ReduceInequality
LeftPolicy $\sqsubseteq$ ReduceInequality
ReduceInequality $\sqsubseteq$ Policy
LeftPolicy $\sqsubseteq$ Policy

Fig. 2. A reference ontology

policy is that it promotes the levels of welfare and that it is not possible to increase welfare and wages at the same time.

Although all individual ontologies are consistent and the concept LeftPolicy is indeed satisfiable in each O_i, the ontology obtained by applying the absolute majority rule is not. The ontology $F_m(O_1, O_2, O_3)$ in this case coincides with the full agenda of Figure 1. In particular, it contains the following definitions:

LeftPolicy $\sqsubseteq$ RaiseWages
LeftPolicy $\sqsubseteq$ RaiseWelfare
RaiseWages $\sqcap$ RaiseWelfare $\sqsubseteq \bot$

By accepting both LeftPolicy $\sqsubseteq$ RaiseWages and LeftPolicy $\sqsubseteq$ RaiseWelfare, we could infer LeftPolicy $\sqsubseteq$ RaiseWages$\sqcap$RaiseWelfare, which together with the axiom RaiseWages $\sqcap$ RaiseWelfare $\sqsubseteq \bot$ makes the concept of LeftPolicy unsatisfiable. Moreover, as soon as we assume that there are indeed candidates for a left-wing policy, e.g., we add an ABox formula LeftPolicy(a), for some constant a, to the ontology $F_m(O_1, O_2, O_3)$, then the collective ontology becomes inconsistent.

To repair the outcome of the majority rule, we assume that the agents agree to use a reference ontology (Figure 2). There is more than one way of repairing the collective ontology. With respect to the reference ontology, the concept ReduceInequality is a generalisation of RaiseWelfare, and of RaiseWages. Therefore, one way of correcting the collective ontology is to weaken the axiom LeftPolicy $\sqsubseteq$ RaiseWages, by substituting the concept RaiseWages with the concept ReduceInequality. Symmetrically, one can weaken LeftPolicy $\sqsubseteq$ RaiseWelfare, by generalising the concept RaiseWelfare also with ReduceInequality. In both cases, we obtain a consistent set of axioms.

Another way of fixing the collective ontology would also be to weaken the axiom RaiseWages $\sqcap$ RaiseWelfare $\sqsubseteq \bot$, for instance by specialising the concept RaiseWages $\sqcap$ RaiseWelfare into $\bot$. However, the repaired ontology would contain the uninformative axiom $\bot \sqsubseteq \bot$. Although we effectively obtain a consistent ontology, a repair strategy would ideally avoid such an outcome when possible.

In the remainder of this paper, we study a number of strategies of removing possibly inconsistent definitions by generalising and/or specialising the concepts involved in the inconsistent collective ontologies.

5 Generalisation and Specialisation of $\mathcal{ALC}$ Concepts

Part of our strategy for fixing the collective ontology (obtained by means of an aggregation function) relies on weakening the axioms present in a TBox w.r.t. an ontology. Weakening an axiom essentially amounts to refine its premise or its conclusion, as we shall see.

Refinement operators are a well-known notion in Inductive Logic Programming where they are used to structure a search process for learning concepts from examples. In this setting, two types of refinement operators exist: specialisation refinement operators and generalisation refinement operators. While the former constructs specialisations of hypotheses, the latter constructs generalisations [14].

Given the quasi-ordered set $\langle \mathcal{L}(\mathcal{ALC}, N_c, N_R), \sqsubseteq \rangle$, a generalisation refinement operator is defined as follows:

$$\gamma_{\mathcal{T}}(C) \subseteq \{C' \in \mathcal{L}(\mathcal{ALC}, N_c, N_R) \mid C \sqsubseteq_{\mathcal{T}} C'\} \ .$$

Whereas a specialisation refinement operator is defined as follows:

$$\rho_{\mathcal{T}}(C) \subseteq \{C' \in \mathcal{L}(\mathcal{ALC}, N_c, N_R) \mid C' \sqsubseteq_{\mathcal{T}} C\} \ .$$

Roughly speaking, a generalisation refinement operator takes a concept C as input and returns a set of descriptions that are more general than C by taking a TBox $\mathcal{T}$ into account. A specialisation operator, instead, returns a set of descriptions that are more specific. Whilst specialisation operators for $\mathcal{ALC}$ (and other description logics) have been studied in the literature [15], few proposals have been made for generalisation operators in less expressive logics due to the complexity of dealing with concept generalisations [3, 4, 23].

In what follows, we present the novel generalisation refinement operator for $\mathcal{ALC}$ which is a variant of the one proposed in [5]. Then, we will define a specialisation operator based on it. In order to define this generalisation refinement operator for $\mathcal{ALC}$, we need some auxiliary definitions.[3] In the following, we assume that complex concepts C are rewritten into negation normal form, and thus negation only appears in front of atomic concepts.

Definition 6. *Let $\mathcal{T}$ be an $\mathcal{ALC}$ TBox with concept names from N_C. The set of* non-trivial subconcepts *of $\mathcal{T}$ is given by*

$$\mathsf{sub}(\mathcal{T}) = \{\top, \bot\} \cup \bigcup_{C \sqsubseteq D \in \mathcal{T}} \mathsf{sub}(C) \cup \mathsf{sub}(D) \ .$$

[3] To avoid any confusion, we point out that we use $=$ and $\neq$ between $\mathcal{ALC}$ concepts to denote syntactic identity and difference, respectively.

where sub *is defined over the structure of concept descriptions as follows:*

$$\begin{aligned}
\mathit{sub}(A) &= \{A\} \\
\mathit{sub}(\bot) &= \{\bot\} \\
\mathit{sub}(\top) &= \{\top\} \\
\mathit{sub}(\neg A) &= \{\neg A, A\} \\
\mathit{sub}(C \sqcap D) &= \{C \sqcap D\} \cup \mathit{sub}(C) \cup \mathit{sub}(D) \\
\mathit{sub}(C \sqcup D) &= \{C \sqcup D\} \cup \mathit{sub}(C) \cup \mathit{sub}(D) \\
\mathit{sub}(\forall R.C) &= \{\forall R.C\} \cup \mathit{sub}(C) \\
\mathit{sub}(\exists R.C) &= \{\exists R.C\} \cup \mathit{sub}(C)
\end{aligned}$$

Based on $\mathsf{sub}(\mathcal{T})$, we define the upward and downward cover sets of atomic concepts. Intuitively, the upward set of A collects the most specific subconcepts found in the Tbox $\mathcal{T}$ that are more general (subsume) A; conversely, the downward set of A collects the most general subconcepts from $\mathcal{T}$ that are subsumed by A. The downcover is only needed for the base case of generalising a negated atom. The properties of $\mathsf{sub}(\mathcal{T})$ guarantee that the upward and downward cover sets are finite.

Definition 7. *Let $\mathcal{T}$ be an $\mathcal{ALC}$ TBox over N_C. The* upward cover set *of the concept C with respect to $\mathcal{T}$ is:*

$$\mathsf{UpCov}_{\mathcal{T}}(C) := \{D \in \mathsf{sub}(\mathcal{T}) \mid C \sqsubseteq_{\mathcal{T}} D \text{ and there is no } D' \in \mathsf{sub}(\mathcal{T}) \text{ with } C \sqsubset_{\mathcal{T}} D' \sqsubset_{\mathcal{T}} D\} \ . \quad (1)$$

The downward cover set *of the concept C with respect to $\mathcal{T}$ is:*

$$\mathsf{DownCov}_{\mathcal{T}}(C) := \{D \in \mathsf{sub}(\mathcal{T}) \mid D \sqsubseteq_{\mathcal{T}} C \text{ and there is no } D' \in \mathsf{sub}(\mathcal{T}) \text{ with } D \sqsubset_{\mathcal{T}} D' \sqsubset_{\mathcal{T}} C\} \ . \quad (2)$$

We can now define our generalisation refinement operator for $\mathcal{ALC}$ as follows.

Definition 8. *Let $\mathcal{T}$ be an $\mathcal{ALC}$ TBox. We define $\gamma_{\mathcal{T}}$, the* generalisation refinement operator *w.r.t. $\mathcal{T}$, inductively over the structure of concept descriptions as:*

$$\begin{aligned}
\gamma_{\mathcal{T}}(A) &= \mathsf{UpCov}_{\mathcal{T}}(A) \\
\gamma_{\mathcal{T}}(\neg A) &= \{\neg B \mid B \in \mathsf{DownCov}_{\mathcal{T}}(A)\} \cup \mathsf{UpCov}_{\mathcal{T}}(\neg A) \\
\gamma_{\mathcal{T}}(\top) &= \{\top\} \\
\gamma_{\mathcal{T}}(\bot) &= \mathsf{UpCov}_{\mathcal{T}}(\bot) \\
\gamma_{\mathcal{T}}(C \sqcap D) &= \{C' \sqcap D \mid C' \in \gamma_{\mathcal{T}}(C)\} \cup \{C \sqcap D' \mid D' \in \gamma_{\mathcal{T}}(D)\} \cup \mathsf{UpCov}_{\mathcal{T}}(C \sqcap D) \\
\gamma_{\mathcal{T}}(C \sqcup D) &= \{C' \sqcup D \mid C' \in \gamma_{\mathcal{T}}(C)\} \cup \{C \sqcup D' \mid D' \in \gamma_{\mathcal{T}}(D)\} \cup \mathsf{UpCov}_{\mathcal{T}}(C \sqcup D) \\
\gamma_{\mathcal{T}}(\forall R.C) &= \{\forall R.C' \mid C' \in \gamma_{\mathcal{T}}(C)\} \cup \mathsf{UpCov}_{\mathcal{T}}(\forall R.C) \\
\gamma_{\mathcal{T}}(\exists R.C) &= \{\exists R.C' \mid C' \in \gamma_{\mathcal{T}}(C)\} \cup \mathsf{UpCov}_{\mathcal{T}}(\exists R.C)
\end{aligned}$$

When there is no ambiguity or to refer to an arbitrary TBox, we often omit the subscript $\mathcal{T}$ from the operator $\gamma_{\mathcal{T}}$.

Given a generalisation refinement operator γ, $\mathcal{ALC}$ concepts are related by refinement paths as described next.

Definition 9. *For every concept C, we note $\gamma^i(C)$ the i-th iteration of its generalisation. It is inductively defined as follows:*

- $\gamma^0(C) = \{C\}$;
- $\gamma^{j+1}(C) = \gamma^j(C) \cup \bigcup_{C' \in \gamma^j(C)} \gamma(C') \quad , j \geqslant 0$.

Definition 10. *The minimal number of generalisations to be applied in order to generalise C to D is called the distance between C and D, noted $\lambda(C \xrightarrow{\gamma} D)$. Formally, $\lambda(C \xrightarrow{\gamma} D) = \min\{j \mid j \geqslant 0$ and $D \in \gamma^j(C)\}$.*

$\lambda(\_ \xrightarrow{\gamma} \_)$ is a partial function that is defined only when the concept passed as first argument can eventually be refined into the concept passed as second argument.

Definition 11. *The set of all concepts that can be reached from C by means of γ in a finite number of steps is*

$$\gamma^*(C) = \bigcup_{i \geqslant 0} \gamma^i(C) \ .$$

This following lemma says that it is always possible to generalise a concept into $\top$ in a finite number of steps.

Lemma 12. *For every concept C, it is the case that $\top \in \gamma^*(C)$.*

It is possible to define a specialisation operator with definitions analogous to the ones for γ. Here, we simply define the specialisation operator ρ from the generalisation operator γ as follows:

$$\rho_{\mathcal{T}}(C) = \{C' \mid C' \in \mathsf{sub}(\mathcal{T}) \text{ and } C \in \gamma_{\mathcal{T}}(C')\} \ .$$

The definitions in this section can be easily adapted for ρ and we omit them. As we did for γ, we denote ρ^i to be the i-th iteration of ρ, and we denote ρ^* to be the unbounded iteration of the specialisation operator ρ. The integer $\lambda(C \xrightarrow{\rho} D)$ denotes the minimal number of specialisations required to reach D from C. Like in Lemma 12, we have that for every concept C, it is the case that $\bot \in \rho^*(C)$.

6 Repairing Collective Ontologies

We introduce a strategy for fixing the collective ontology obtained by means of an aggregation function. As we have discussed, a number of important aggregators (e.g., the majority or quota rules) might fail to preserve consistency. We propose a methodology to cope with collective inconsistency by manipulating the axioms that cause the inconsistency via axiom weakening.

6.1 Axiom Weakening

Roughly speaking, weakening an axiom $C \sqsubseteq D$ amounts to *enlarging* the set of interpretations that satisfy the axiom. This could be done in different ways: Either by substituting $C \sqsubseteq D$ with $C \sqsubseteq D'$, where D' is a more general concept than D (i.e., its interpretation is larger); or, by modifying the axiom $C \sqsubseteq D$ to $C' \sqsubseteq D$, where C' is a more specific concept than C; or even by generalising and specialising simultaneously to obtain $C' \sqsubseteq D'$.

Given an ontology O, we denote the set of its concept names of O by N_C^O. We want to define a procedure to change axioms gradually by replacing them with less restrictive axioms. Recall that γ_O denotes the generalisation of a concept and ρ_O denotes its specialisation with respect to a given ontology O.

Definition 13 (Axiom weakening). *Given an axiom $C \sqsubseteq D$ of O, the set of* weakenings *of $C \sqsubseteq D$ in O, denoted by $g_O(C \sqsubseteq D)$ is the set of all axioms $C' \sqsubseteq D'$ such that*

$$C' = \rho_O^*(C) \text{ and } D' = \gamma_O^*(D) \ .$$

If the ontology O is consistent, the weakening of an axiom in O is always satisfied by a super set of the interpretations that satisfy the axiom. Let $I = (\Delta^I, \cdot^I)$ be an interpretation. Then by definition the class of all entities that fulfil the axiom $C \sqsubseteq D$ is $(\Delta^I \setminus C^I) \cup D^I$. A weakening of $C \sqsubseteq D$ either specialises C, therefore restricting C^I, and accordingly extending $\Delta^I \setminus C^I$, or generalises D, therefore, extending D^I. Hence, the set of entities for which $C \sqsubseteq D$ holds is a subset of the set of entities for which any axiom in $g_O(C \sqsubseteq D)$ holds. The following result holds.

Lemma 14. *For every axiom φ, if $\varphi \in g_O(\psi)$, then $\psi \models_O \varphi$.*

Proof. Suppose $\varphi = A' \sqsubseteq B' \in g_O(A \sqsubseteq B)$. Then, by definition of g_O, $A' \sqsubseteq A$ and $B \sqsubseteq B'$ follows from O. Thus, by transitivity of subsumption, we obtain that $A \sqsubseteq B \models_O A' \sqsubseteq B'$. □

Moreover, note that $\bot \sqsubseteq \top$ always belongs to $g_O(C \sqsubseteq D)$. We want to model how to repair any inconsistent set of axioms Y of $\mathcal{ALC}$, by appealing to a consistent reference ontology R. Notice that, even though it is not desirable, R can be dissociated from the axioms in the collective ontology. If the ontology R does not refer to some of the atomic concepts in C or D, then their generalisation is the most general concept $\top$ and their specialisation is the most specific concept $\bot$.[4]

Any inconsistent set of axioms Y can in principle be repaired by means of a sequence of weakenings of the axioms in Y with respect to R.

Lemma 15. *Let* R *be a consistent reference ontology and Y a minimally inconsistent set of axioms. There exists a subset $\{\psi_1, \ldots, \psi_n\} \subseteq Y$ and weakenings $\psi'_i \in g_{\mathrm{R}}(\psi_i)$ for $i, 1 \leq i \leq n$ such that $(Y \setminus \{\psi_1, \ldots, \psi_n\}) \cup \{\psi'_1, \ldots, \psi'_n\} \cup \mathrm{R}$ is consistent.*

[4] Notice that $\gamma_{\mathcal{T}}$ and $\rho_{\mathcal{T}}$ are defined on arbitrary $\mathcal{ALC}$ formulas.

Algorithm 1 Fixing ontologies through weakening.

Procedure FIX-ONTOLOGY(O,R) ▷ O inconsistent ontology, R reference ontology

while O is inconsistent **do**
  $\mathcal{Y} \leftarrow \textsc{mis}(O)$ ▷ find all minimally inconsistent subsets of O
  for $Y \in \mathcal{Y}$ **do**
    choose $\psi \in Y, \psi' \in g_{\mathrm{R}}(\psi)$ with $Y \setminus \{\psi\} \cup \{\psi'\}$ consistent, $\lambda_O(\psi, \psi')$ minimal
    $O \leftarrow (O \setminus \{\psi\}) \cup \{\psi'\}$
return O

Proof. It suffices to notice that for every $\varphi \in Y$, the set $g_O(\varphi)$ contains the tautological axiom $\bot \sqsubseteq \top$. Because of Lemma 12 for γ_O (and analogous lemma for ρ_O), in the worst case, one can weaken every axiom ψ_i in Y into $\psi'_i = \bot \sqsubseteq \top$. This is equivalent to removing ψ. Since R is consistent, it follows that the set $(Y \setminus \{\psi_1, \ldots, \psi_m\}) \cup \{\psi'_1, \ldots, \psi'_m\} \cup \mathrm{R}$ is also consistent. □

The lemma ensures that a minimally inconsistent set of axioms can be adequately weakened to be integrated consistently into another consistent ontology. As the proof shows, in the worst case these axioms are weakened to become a tautology. However, we are interested in weakening axioms as little as possible to remain close to the original axioms. Notice that every axiom in $g_O(C \sqsubseteq D)$ is obtained by applying γ and ρ a finite number of times. Hence, we can define λ_O to be a refinement distance in an ontology O, such that for every $C' \sqsubseteq D' \in g_O(C \sqsubseteq D)$,

$$\lambda_O(C \sqsubseteq D, C' \sqsubseteq D') = \lambda(C \xrightarrow{\rho_O} C') + \lambda(D \xrightarrow{\gamma_O} D') \ .$$

Repair strategies can exploit this distance to guide the weakening of axioms that are the least stringent. Indeed, by minimising λ_O, we are trying to find the weakening of the axiom that is as close as possible to the original axiom in the context of O. Moreover, by trying to minimise the distance, we are trying to prevent non-informative axioms to be selected as weakenings. In other words, axioms like $\bot \sqsubseteq \top$, $\bot \sqsubseteq D$, or $C \sqsubseteq \top$ should only be selected if no other options are available. In principle, we can also provide refined constraints on the generalisation and specialisation paths, e.g. by fixing an ordering of the concepts of the ontology O that determines which concepts are to be generalised or specialised first.

6.2 Fixing Collective Ontologies via Axiom Weakenings

When $F(\boldsymbol{O})$ is inconsistent, we can adopt the general strategy described in Algorithm 1 to repair it w.r.t. a given (fixed) reference ontology R.[5] The algorithm finds all the minimally inconsistent subsets $Y_1, \ldots, Y_n$ of $F(\boldsymbol{O})$ (e.g., using the methods from [2,21]) and repairs each of them by weakening one of its axioms to regain consistency. From all the possible choices made to achieve this goal, the

[5] We discuss a few strategies for deciding a reference ontology in the next section.

$\mathsf{LeftPolicy} \sqsubseteq \mathsf{RaiseWages}$
$\mathsf{LeftPolicy} \sqsubseteq \mathsf{RaiseWelfare}$
$\mathsf{RaiseWages} \sqcap \mathsf{RaiseWelfare} \sqsubseteq \bot$

Fig. 3. The ontology $F_m(\boldsymbol{O})$.

algorithm selects one that minimizes the distance λ_O (line 4). This process corrects all original causes for inconsistency, but may still produce an inconsistent ontology due to masking [11]. Hence, the process is repeated until a consistent ontology is found. Notice that the algorithm is non-deterministic, since it depends on the choice of the axiom to weaken, and the weakening selected. As such, it can also be seen as a strategy returning a non-singleton *set* of ontologies. That is, the procedure is non-resolute [20].

In order to define the strategy that we have presented as a function returning a single ontology, two policies for breaking ties are required. For both, we can capitalize on the linear order over formulas $\prec_{\mathcal{ALC}}$ introduced earlier. We can define a linear order $\prec^x_{\mathcal{ALC}}$ over axioms as follows: $C \sqsubseteq D \prec^x_{\mathcal{ALC}} E \sqsubseteq F$ iff $C \prec_{\mathcal{ALC}} E$, or $C = D$ and $D \prec_{\mathcal{ALC}} F$.

Firstly, on line 4 of the strategy, one needs to pick one axiom out of every minimally inconsistent sets. The first policy that we adopt is to choose the axiom ψ out of Y that is minimal with respect to $\prec^x_{\mathcal{ALC}}$.

Secondly, also on line 4, there may be more than one closest generalisation ψ' of a chosen ψ that minimise $\lambda_O(\psi, \psi')$. To break the tie, the second policy that we adopt, we again to pick the one that is minimal with respect to $\prec^x_{\mathcal{ALC}}$.

Now, with a reference ontology R and the linear order $\prec_{\mathcal{ALC}}$ fixed, the strategy returns an aggregation procedure $g_{R,\prec_{\mathcal{ALC}}}(F(\boldsymbol{O}))$: firstly aggregate the individual ontologies in $\boldsymbol{O}$, then generalise the axioms in any possible inconsistent set of $F(\boldsymbol{O})$ with respect to the reference ontology R, and obtain $g_{R,\prec_{\mathcal{ALC}}}(F(\boldsymbol{O}))$.

We leave a detailed treatment of possible rules to break ties and their respective merits for future work.

6.3 An Application

We illustrate our strategy by discussing the example in Section 4. We have seen that the absolute majority rule can return inconsistent collective ontologies. The collective ontology $F_m(\boldsymbol{O})$ is presented in Figure 3. The collective ontology $F_m(\boldsymbol{O})$ is inconsistent.

To apply our strategy, we have firstly to select a reference ontology R. Suppose we choose the ontology in Figure 1. We exemplify how $g_R(F_m(\boldsymbol{O}))$ works by assuming in this case that it is non-resolute.

We start by choosing an axiom in a minimally inconsistent subset of $F_m(\boldsymbol{O})$ that needs to be weakened. The whole collective ontology $F_m(\boldsymbol{O})$ is a minimally inconsistent set. So suppose we start by $\mathsf{LeftPolicy} \sqsubseteq \mathsf{RaiseWages}$. Then, we have to select a concept to generalise or specialise. Suppose we select $\mathsf{RaiseWages}$. Thus, to generalise the axiom $\mathsf{LeftPolicy} \sqsubseteq \mathsf{RaiseWages}$ we can replace it by

$\mathsf{LeftPolicy} \sqsubseteq \mathsf{ReduceInequality}$, since $\mathsf{ReduceInequality}$ is the closest generalisation to $\mathsf{RaiseWages}$ in the reference ontology R. We obtain then the new ontology, where the axiom $\mathsf{LeftPolicy} \sqsubseteq \mathsf{RaiseWages}$ has been replaced by the weaker $\mathsf{LeftPolicy} \sqsubseteq \mathsf{ReduceInequality}$. That is, $g_R(F_m(\boldsymbol{O}))$ contains the ontology:

$$\begin{array}{|l|}\hline \mathsf{LeftPolicy} \sqsubseteq \mathsf{ReduceInequality} \\ \mathsf{LeftPolicy} \sqsubseteq \mathsf{RaiseWelfare} \\ \mathsf{RaiseWages} \sqcap \mathsf{RaiseWelfare} \sqsubseteq \bot \\ \hline \end{array}$$

Alternatively, we could have started by generalising $\mathsf{RaiseWages} \sqcap \mathsf{RaiseWelfare} \sqsubseteq \bot$. In this case, we have two choices, either we generalise $\bot$, or we specialise $\mathsf{RaiseWages} \sqcap \mathsf{RaiseWelfare}$. $\bot$ can be generalised by any concept in the reference ontology. $\mathsf{RaiseWages} \sqcap \mathsf{RaiseWelfare}$ can here be specialised only by replacing it with $\bot$, obtaining therefore $\bot \sqsubset \bot$, which is a (non-informative) logical axiom. By replacing an axiom with a logical one, the effect on the final ontology is the same as removing the original axiom (a logical axiom does not restrict the models of the ontology). Thus, in this case, the repaired ontology is the following ontology which is an element of $g_R(F_m(\boldsymbol{O}))$:

$$\begin{array}{|l|}\hline \mathsf{LeftPolicy} \sqsubseteq \mathsf{ReduceInequality} \\ \mathsf{LeftPolicy} \sqsubseteq \mathsf{RaiseWelfare} \\ \hline \end{array}$$

6.4 Selecting the Reference Ontology

The procedure illustrated in the previous section relies on the availability and specific choice of a reference ontology to repair the outcome of the aggregation procedure. A reference ontology may provide more information that the individual ontologies, as in the example of Section 4, however we have to decide how to deal with the axioms of the reference ontology that overlap with the formulas of the agenda.

For the sake of discussion, we briefly illustrate a few viable solutions. As we shall see, the choice of R affects the quality of the fixing of the collective ontology, both in terms of fairness of the outcome (i.e. how many agents would accept the generalisations obtained via R) and in terms of the amount of information available.

The first solution is to assume that R contains a maximally consistent subset X of $F(\boldsymbol{O})$. That is, we select, among those axioms that have been collectively accepted according to F, a maximally consistent subset. Notice that, in the definition of generalisation of an axiom, we use γ and ρ, the definition of which requires the concepts of upcover and downcover (c.f. Definition 8). Both of these definitions imply that there may be formulas that are inferred by $F(\boldsymbol{O})$, even though they are not in $F(\boldsymbol{O})$. This is due to the fact that $F(\boldsymbol{O})$ needs not be deductively closed and that the formulas in $F(\boldsymbol{O})$ are in fact a subset of an arbitrarily chosen agenda of formulas, as usual in JA.

If F is the majority rule, the generalisations in R that may replace or repair the axioms in $F(\boldsymbol{O})$ are at least consistent with *some* of the formulas of the ontologies for a majority of agents. Since X must contain some of the axioms accepted by majority, we can at least say that $O_i \cap X$ is consistent for a majority of agents.

A second choice is to take a more informative ontology as a reference ontology. We can assume that R includes a maximally consistent subset Z of $O_1 \cup \cdots \cup O_n$. In this case, although the reference ontology is in general richer than in the previous case, we can only ensure that the generalisations provided by R are consistent with some of the axioms of at least one agent: an axiom is in Z only if at least one agent vote for that. Therefore $O_i \cap X$ is consistent for at least one agent.

Thirdly, we could assume that the reference ontology should include the intersection $O_1 \cap \cdots \cap O_n$. Recall that by taking the intersection of all individual ontologies, consistency is ensured. In this case, the reference ontology may be very poor, however if φ follows from $O_1 \cap \cdots \cap O_n$, then φ follows from every individual ontology. Therefore, any generalisation of axioms is in this case acceptable for every agent. In the case $O_1 \cap \cdots \cap O_n$ is empty, R might only contain logical axioms. In this case, repairing an ontology by appealing to R amounts to only replacing possibly conflicting axioms with logical ones. Thus, that has the effect of removing axioms that cause the collective inconsistency.

Another strategy for selecting a reference ontology is to say that it has to include one agent's ontology as the representative one. Strategies for selecting the most representative voter out of a profile of voters have been discussed in [9]. For the case of ontology aggregation, we can find the most representative ontology by exploiting the definition of Hamming distance between an ontology and a profile [20], that is the most representative ontology is the one that minimises the sum of the distances with respect to all the other agents ontologies.

Finally, a strategy that tries to balance between the number of voters that may consistently accept the generalisations of the axioms provided by R and the amount of information conveyed by R is the following. Given a profile of individual ontologies $\boldsymbol{O}$, we take a $W = O_{i_1} \cup \cdots \cup O_{i_l}$ where $\{i_1 \dots i_l\} \subseteq \mathcal{N}$ in such a way that $\mid \{i_1 \dots i_l\} \mid$ is maximal, W is consistent, and for every voter $i_j \notin \{i_1, \dots, i_l\}$, it is the case that $W \cup O_{i_j}$ is inconsistent. The reference ontology R is then assumed to include W. That is, in this case the reference ontology tries to join together the highest number of individual ontologies, as far as they are consistent. The sets $O_{i_1} \cup \cdots \cup O_{i_n}$ for which the cardinality of $\{i_1 \dots i_l\}$ is maximal are in general not unique, therefore again here we have to take into account a tie-breaking rule; again, tie-breaking can exploit the order $\prec_{\mathcal{ALC}}$. Observe that, in this case, W is not necessarily a maximally consistent subset of $O_1 \cup \cdots \cup O_n$, as W has the constraint that either it contains all the formulas of an individual ontology or none. With respect to this reference ontology, every agent whose ontology is in W can in principle accept all the generalisations provided by R, as they are consistent with her or his ontology.

7 Discussion and Future Work

In this paper, we proposed a novel approach to repair an inconsistent ontology, which is obtained by aggregating the individual ontologies of a community of agents. Our approach is based on the notion of axiom weakening, which amounts to generalise or to specialise the concepts in axioms that belong to minimally inconsistent subsets. We also discussed the problem of engineering a reference ontology for repairing the inconsistent collective ontology. Whilst we proposed different possible viable solutions, a more extensive evaluation is needed.

Nevertheless, we believe that this paper is already an important contribution both to Social Choice Theory (SCT), Judgement Aggregation (JA) and to the Description Logic fields. As far as SCT and JA are concerned, we showed how it is possible to repair an inconsistent collective ontology while maintaining well-known real-world aggregators. As far as DL is concerned, we showed how the notion of refinement operators, typically encountered in machine learning, can be transposed to symbolic reasoning to define axiom weakening.

As future work, apart from evaluating the reference ontology engineering already mentioned, we aim at taking into account the notion of coherence by Thagard [22]. Coherence theory, and in particular conceptual coherence, could be used, for instance, to decide which minimally inconsistent subset to choose; or which axiom to weaken, depending on how much it contributes to maximising the overall coherence of the collective ontology.

References

1. Baader, F., Calvanese, D., McGuinness, D.L., Nardi, D., Patel-Schneider, P.F. (eds.): The Description Logic Handbook: Theory, Implementation, and Applications. Cambridge University Press, New York, NY, USA (2003)
2. Baader, F., Peñaloza, R.: Axiom pinpointing in general tableaux. Journal of Logic and Computation 20(1), 5–34 (February 2010), special Issue: Tableaux and Analytic Proof Methods
3. Baader, F., Sertkaya, B., Turhan, A.Y.: Computing the least common subsumer w.r.t. a background terminology. Journal of Applied Logic 5(3), 392 – 420 (2007)
4. Confalonieri, R., Eppe, M., Schorlemmer, M., Kutz, O., Peñaloza, R., Plaza, E.: Upward Refinement Operators for Conceptual Blending in $\mathcal{EL}^{++}$. Annals of Mathematics and Artificial Intelligence (2016), doi:10.1007/s10472-016-9524-8
5. Confalonieri, R., Kutz, O., Troquard, N., Galliani, P., Porello, D., Peñaloza, R., Schorlemmer, M.: Coherence, Similarity, and Concept Generalisation. In: Proc. of DL 2017. CEUR (2017)
6. Dietrich, F.: A generalised model of judgment aggregation. Social Choice and Welfare 28(4), 529–565 (2007)
7. Dietrich, F., List, C.: Judgment aggregation by quota rules: Majority voting generalized. Journal of Theoretical Politics 19(4), 391–424 (2007)
8. Dietrich, F., List, C.: Probabilistic opinion pooling generalized. part one: general agendas. Social Choice and Welfare 48(4), 747–786 (2017)
9. Endriss, U., Grandi, U.: Binary aggregation by selection of the most representative voters. In: AAAI. pp. 668–674 (2014)
10. Gaertner, W.: A Primer in Social Choice Theory. Oxford University Press (2006)

11. Horridge, M., Parsia, B., Sattler, U.: Justification masking in ontologies. In: KR 2012. AAAI Press (2012)
12. Kornhauser, L.A., Sager, L.G.: The one and the many: Adjudication in collegial courts. California Law Review 81(1), 1–59 (1993)
13. Kutz, O., Mossakowski, T., Lücke, D.: Carnap, Goguen, and the Hyperontologies: Logical Pluralism and Heterogeneous Structuring in Ontology Design. Logica Universalis 4(2), 255–333 (2010), special Issue on 'Is Logic Universal?'
14. van der Laag, P.R., Nienhuys-Cheng, S.H.: Completeness and properness of refinement operators in inductive logic programming. The Journal of Logic Programming 34(3), 201 – 225 (1998)
15. Lehmann, J., Hitzler, P.: Concept learning in description logics using refinement operators. Machine Learning 78(1-2), 203–250 (2010)
16. List, C., Pettit, P.: Aggregating sets of judgments: An impossibility result. Economics and Philosophy 18(1), 89–110 (2002)
17. List, C.: Deliberation and agreement. In: Deliberation, Participation and Democracy, pp. 64–81. Springer (2007)
18. List, C., Puppe, C.: Judgment aggregation: A survey. In: Handbook of Rational and Social Choice. Oxford University Press (2009)
19. Ottonelli, V., Porello, D.: On the elusive notion of meta-agreement. Politics, Philosophy and Economics 12(1), 68–92 (2013)
20. Porello, D., Endriss, U.: Ontology merging as social choice: Judgment aggregation under the open world assumption. Journal of Logic and Computation 24(6), 1229–1249 (2014)
21. Schlobach, S., Cornet, R.: Non-standard reasoning services for the debugging of description logic terminologies. In: Proc. IJCAI-03. pp. 355–362. Morgan Kaufmann (2003)
22. Thagard, P.: Coherence in thought and action. The MIT Press (2000)
23. Zarrieß, B., Turhan, A.Y.: Most Specific Generalizations w.r.t. General $\mathcal{EL}$-TBoxes. In: Proceedings of the 23th International Joint Conference on Artificial Intelligence. pp. 1191–1197. IJCAI '13, AAAI Press (2013)