Formal Ontology in Information Systems
N. Aussenac-Gilles et al. (Eds.)

doi:10.3233/FAIA231126

A Quest for Identity Criteria in Computational Ontologies

Pawel GARBACZ [1]
The John Paul II Catholic University of Lublin

Abstract. The notion of identity criteria marked the dawn of contemporary formal ontology. Despite a number of issues this notion has raised, the quest for them still seems to be worthwhile, in particular in the case of a formal ontology built in the context of information systems. In the current paper I investigate the benefits and costs of using automatic theorem provers in the task of identifying such criteria for formal ontologies that are expressed in a prover-processable language. To this end two detailed case studies were performed – each concerned an upper-level ontology presented in a recent volume of the Applied Ontology journal. The identity criteria found by the process described in this paper turned out to be not particularly illuminating. The respective theorems that define them are rather direct consequences of the axioms, so proofs and models provided by the prover do not provide any new insights into the actual conceptual contents of the formal ontologies.

Keywords. criteria of identity, prover, automation, Vampire

1. Introduction

The notion of identity criteria had marked the dawn of contemporary formal ontology as exemplified by such papers, as [1] or [2]. Despite all conceptual issues this notion opens up, cf., for example, [3], many ontologists still believe that further research into its applicability is worth pursuing. One of the reasons for such insistence may follow from the belief that only entities with clear identity criteria are ontologically respectable or at least more respectable than those without – as famously stated by Quine [4]. This theoretical respectability may be cashed out in a computer system in a number of ways:

1. as a theoretical foundation for managing unique identifiers, including complex primary keys in relational databases;
2. as an operational principle to distinguish between the case where a new object needs to be recognised and the case when an existing object is specified in more detail;
3. as a working method of counting objects.

However, the clear-cut examples of identity criteria are scarce, i.e., most of the principles recommended as such criteria are at best objectionable, i.e., they are vulnerable to

[1]This research was funded by the Minister of Education and Science Republic of Poland within the program under the name "Regional Initiative of Excellence" in 2019-2023 (project number: 028/RID/2018/19, the amount of funding: 11 742 500 PLN)

various argumentative patterns that aim to show that a given purported criterion cannot achieve its purpose. Some of these patterns are based on philosophers', sometimes conflicting, intuitions regarding imagined or real situations where the purported principles of identity appear to fail.

The context of formal ontologies may give us a higher ground in this debate. After all, one of the main reasons why we develop such theories is to clarify certain philosophical insights so that they become less subjective or even verifiable to a limited degree. This lends some support to the idea that one can more rigorously investigate the notion of identity criteria within a certain formal ontology and hope for more reliable findings than when our conceptual grounds are less firm.

In this paper I will show what kind of results one may expect using the automatic theorem provers in the quest for identity criteria in such a context. The results reported here build upon the research presented in [5]. For the reasons explained in section 4, I had to scope this study just to two cases of formal ontologies: DOLCE and UFO.

2. Definitions and paraphernalia

The context of this research is fixed by the current developments in formal applied ontologies, due to which it is now possible to use theorem provers and model finders to investigate the formal properties of such theories.

If one takes this context into account, certain conceptualisations of formal applied ontologies seem to be simpler than others. For instance, it is convenient to interpret a formal applied ontology as a text file that can be an input to a computer application – obviously this assumption is to be taken here as ontologically innocent as possible, e.g., I do not assume that a formal ontology *is* a file. I will say that a formal ontology is *computationally friendly*, or just *computational*, if it can be an input to a theorem prover/model finder – again, this definition makes the concept of computationally friendly ontologies rather fluid without any logical or philosophical implications regarding the choice of the appropriate language.[2] All theorem provers/model finders I am aware of assume that an input text file consists of a number of strings, i.e., sequences of ASCII characters, of a certain form. Although sometimes one can make meta-logical distinctions between such formulae, e.g., one can mark some of them as axioms and others as conjectures, for the sake of simplicity I will assume that a formal applied ontology is just a set of formulae to be interpreted by theorem prover/model finder as axioms.

A *criterion of identity* is usually understood as a principle that accounts for the identity of a thing. T. Williamson [7, p. 144-153] introduced the well-known distinction between two kinds of such principles.

A *one-level criterion of identity* is expressed as a sentence of the form:

$$\forall x,y[\kappa(x)\wedge\kappa(y)\rightarrow(\rho(x,y)\equiv x=y)] \tag{1}$$

where: 'κ' denotes an (ontological) kind or sort of objects and 'ρ' stands for a certain relation between x and y. For example, the axiom of extensionality may be seen as a one-level criterion of identity for sets:

[2] I derived the term 'computational ontology' from [6].

$$\forall x, y[\texttt{Set}(x) \wedge \texttt{Set}(y) \rightarrow (\texttt{SameMembers}(x,y) \equiv x = y)] \quad (2)$$

where '$\texttt{SameMembers}$' is defined as below:

$$\forall x, y[\texttt{SameMembers}(x,y) \equiv \forall z(z \in x \equiv z \in y)] \quad (3)$$

A *two-level criterion of identity* is expressed as a sentence of the form:

$$\forall x, y[\kappa(x) \wedge \kappa(y) \rightarrow (\rho(x,y) \equiv \delta(x) = \delta(y))] \quad (4)$$

where: 'κ' and ρ are understood as in the previous case and 'δ' denotes a function from objects within one ontological level to objects from another level. A typical example of such criteria concerns the directions of lines:

$$\forall x, y[\mathrm{Line}(x) \wedge \mathrm{Line}(y) \rightarrow (x||y \equiv \mathrm{dir}(x) = \mathrm{dir}(y))] \quad (5)$$

where '$x||y$' means that x is parallel to y and '$\mathrm{dir}(x)$' denotes the direction of (line) x.

Despite its ontological significance I will ignore in this paper this kind of principles, focusing on one-level criteria of identity – an excuse for this selection may be the fact that available computational ontologies rarely use functional symbols required by the two-level identity criteria.

Usually a criterion of identity, say relation ρ, is supposed to satisfy some additional constraints – below I quote the list from ([8]):

- IC1 non-vacuousness: ρ cannot have parts that are vacuously satisfiable, i.e., ρ holds between elements of a certain kind such that all of them are alike with respect to the properties associated to that kind;
- IC2 informativeness: criterion ρ (for objects of kind κ) should contribute to the specification of the nature of these objects;
- IC3 partial-exclusivity: criterion ρ (for objects of kind κ) cannot be so general that it can be applied to other kinds of objects;
- IC4 minimality: criterion ρ (for objects of kind κ) specifies the smallest number of determinables, i.e., aspects, such that determinates that fall under them turn out to be necessary and sufficient to ensure the identity of two objects of kind κ;
- IC5 non-circularity: criterion ρ (for objects of kind κ) cannot make use of kind κ;
- IC6 non-tautologicity: ρ is a proper subset of $K \times K$ (where K is the set of all κ-objects);
- IC7 κ-maximality: ρ is the $\subseteq$-greatest relation that makes condition 1 above true;
- IC8 uniqueness: ρ is the unique identity criterion for κ.
- IC9 formal properties of identity: ρ is an equivalence relation;

Some of these constraints can be expressed in a formal way and therefore can be automatically tested in the case of a computationally friendly ontology: IC3, IC6, IC7, IC8, and IC9 – I will let the reader to figure out (or consult [8]) the obvious details of such formalisation except for IC3 which probably require some interpretation or restriction. Due to the formal structure of identity criteria defined by 1 a criterion of identity for a given type κ is also an identity criterion for all types κ subsumes. So if one wants to be able to define identity criteria for such categories as physical objects or temporal qualities, which are subsumed by other types, one needs to redefine condition IC3 in the following way: (IC3') partial-exclusivity: if ρ is an identity criterion for kind κ, then it is not a criterion for other types except for all types κ subsumes. In the terminology of [2] one can say that IC3 is restricted to types supplying, and not just carrying, identity criteria.

Other constraints from the list above, i.e., IC1, IC2, IC4, and IC5, seem to resist such procedure and require some more sophisticated logical and/or conceptual analysis, which cannot be automated as of now.

In what follows I will consider a subclass of (one-level) identity criteria, which are structurally similar to the aforementioned criterion of identity for sets. Consider the following generalisation of 2:

$$\forall x, y[\kappa(x) \wedge \kappa(y) \rightarrow (\forall z(\pi(z,x) \equiv \pi(z,y)) \rightarrow x = y)] \tag{6}$$

Similarly, one can have another form of one-level identity criteria:

$$\forall x, y[\kappa(x) \wedge \kappa(y) \rightarrow (\forall z(\pi(x,z) \equiv \pi(y,z)) \rightarrow x = y)] \tag{7}$$

Both 6 and 7 can be generalised in an obvious way up to:

$$\begin{gathered}\forall x, y[\kappa(x) \wedge \kappa(y) \rightarrow \\ (\forall z_1 \ldots z_{n-1} z_{n+1}, \ldots z_m(\pi(z_1, \ldots, z_{n-1}, x, z_{n+1}, \ldots, x_m) \equiv \pi(z_1, \ldots, z_{n-1}, y, z_{n+1}, \ldots, x_m)) \\ \rightarrow x = y]\end{gathered} \tag{8}$$

where: $1 \leq n \leq m$ (and m is the arity of π).

For the sake of brevity, "$\forall z_1 \ldots z_{n-1} z_{n+1}, \ldots z_m(\pi(z_1, \ldots, z_{n-1}, x, z_{n+1}, \ldots, x_m) \equiv \pi(z_1, \ldots, z_{n-1}, y, z_{n+1}, \ldots, x_m)$" will be abbreviated to "$\texttt{same}_{\pi,n,m}(x,y)$" so that 8 can be written as:

$$\forall x, y[\kappa(x) \wedge \kappa(y) \rightarrow (\texttt{same}_{\pi,n,m}(x,y) \rightarrow x = y)]] \tag{9}$$

For example, `SameMembers` is now $\texttt{same}_{\in,1,2}$.

Any sentence that is of the form of 9 will be called here *a statement of an indiscerniblity-driven criterion of identity*. Relation $\texttt{same}_{\pi,n,m}(x,y)$ in such a statement will be called *an indiscerniblity-driven criterion of identity* and π be referred to as the *base* for such criterion. Note that all indiscerniblity-driven identity criteria satisfy condition IC9.

Note that the notion of indiscerniblity-driven criterion of identity explicitly specifies only a sufficient and not necessary condition for identity – the reason for this change is that all indiscerniblity-driven identity criteria are reflexive, so by definition they also are necessary conditions.

Although not all one-level criteria of identity fall under formula 9, the usual examples of one-level criteria do – see, for example, [7], [9], or [10]. The main exception to this observation is the spatio-temporal continuity as an identity criterion for material objects (as discussed in [11]).

To simplify the narrative, from now on, the term 'identity criterion' will be restricted in the rest of the paper to indiscerniblity-driven criteria of identity – this does not imply any view on the existence or value of other types of identity criteria.

3. Task and method

Suppose that there is a computationally friendly applied ontology or any computationally friendly theory for that matter. Each such theory has its signature, which specifies, among other things:

1. set of monadic predicates, which will be called the ontology's *types*;
2. set of binary and higher arity predicates, which will be called the ontology's *relations*.

The task is to find within this theory all criteria of identity, i.e., all formulae of the form given by 9 that are the theory's theorems. In other words, if κ is a type in the theory and π is an m-ary relation there, we investigate whether there is an n such that 9 is a theorem of this theory. If there is, then I will say that the respective relation of indiscernibility, i.e., $\texttt{same}_{\pi,n,m}(x,y)$, is a *rough criterion of identity* for κ (within the theory). If a rough identity criterion satisfies a constraint ICn from the list in the previous section (IC1 - IC8), then it will also be called *ICn-refined*. If a criterion satisfies all these constraints, it can be said to be *fully-refined*.

Now the fact that we deal with computationally friendly applied ontologies, makes it possible to automate the following processes:

1. find all rough identity criteria
2. for each ICn constraint: IC3 , IC6, IC7, IC8, find all ICn-refined identity criteria.

In order to show how these tasks may be accomplished, I will follow the path specified in [5], i.e., I will use the computationally friendly upper-level ontologies from a recent volume of Applied Ontology, Applied Ontology 17 (2022) and feed them to an automatic prover in search for these criteria.

To this end I will extend each ontology in scope in the following way – see the caveat in section 4: for each type 'κ' and each relation 'π' (whose arity is m) in its signature and for each $1 < n \leq m$, I will add four additional axioms:

$$\kappa(\alpha) \tag{10}$$

$$\kappa(\beta) \tag{11}$$

$$\texttt{same}_{\pi,n,m}(\alpha,\beta) \tag{12}$$

$$\neg\alpha = \beta. \tag{13}$$

where 'α' and 'β' are individual constants that do not occur anywhere in the original ontology. Then I used an automatic prover to check the consistency of this extended ontology. Then:

1. if it turned out to be inconsistent, $\texttt{same}_{\pi,n,m}$ is an (at least) rough identity criterion for 'κ';
2. if it turned out to be consistent, $\texttt{same}_{\pi,n,m}$ is not (even) a rough identity criterion for 'κ'.

Note that it may be the case that, within the resources allocated to the prover, that the prover be not able to find either a proof of inconsistency or a model of the extended ontology, in which case it is undecided whether $\mathtt{same}_{\pi,n,m}$ is a rough identity criterion for 'κ'.

I will also automate the respective checks for the following ICn constraints: IC3 , IC6, IC7, IC8.

4. Implementation

The aforementioned method requires a prover capable of both building proofs and models. As in [5] I selected the Vampire prover, whose native language is the TPTP syntax ([12]). The Python code that adapts Vampire for the purpose of searching for rough and refined identity criteria can be accessed from the GitHub repository https://github.com/mereolog/isniffer. The adaptation in question concerns:

- generating formulae like 12 for all types and relations in the ontology's signature;
- generating formulae needed for automated checks for: IC3 , IC6, IC7, IC8;
- selecting the proper modes for multiple Vampire runs - this includes:
  - ∗ defining the order of Vampire modes to be run: casc_sat, casc, portfolio.
  - ∗ setting the maximal time to be spent in one run: 60 seconds
- performance improvements due to which criteria inherited from supertypes of a given type are not checked again.

For the reasons detailed in [5] I will be able to use only two of these six computationally friendly ontologies, namely, DOLCE and UFO.

- the question of BFO's consistency still cannot be answered using the standard provers and model finders;[3]
- TUpper still remains inconsistent, although for a different reason that the one reported in [5]: commit 9fc7551 produces an inconsistency proof;
- all other ontologies investigated in [5] are rendered in OWL whose expressivity does not allow for such formulae as 1.

For each ontology I will present the identity criteria found in the tabular form, where rows identify the ontology's types and columns will point to the identity criteria. For the latter I will slightly simplify the notation – instead of $\mathtt{same}_{\pi,n,m}$ I will write $\pi@n$. For example, `SameMembers` is now represented by '$\in @1$'. The value '+' in a cell will indicate that the type identified by the cell's row has the identity criterion identified by the cell's column. All empty cells indicate that the respective relation was found not to be an identity criterion for a given type. '?' will mark out those cases where the prover was not able to decide within the allocated resources. In fact, for each ontology I will provide four tables: one for rough identity criteria and one for each of the three constraints: IC3, IC7, IC8 – note that I show the '?' character only in the tables for rough criteria of identity.

[3]One may expect that a search for identity criteria may be decided in at least some case because we check the BFO ontology extended with four additional axioms. However, the way that types are represented in BFO is different from other computational ontologies because BFO represent them as individuals. Therefore, the search for BFO identity criteria requires a different set of these additional axioms.

Obviously, if type κ_1 is (proven to be) a subtype of κ_2, then I will only show the latter – otherwise these tables will not fit the page.

All processes were run on a MacBook Pro (Intel Core i7 with six 2.6 GHz cores and 16 GB of DDR4 RAM).

4.1. DOLCE

The formalisation of DOLCE I used is provided by the link http://www.loa.istc.cnr.it/wp-content/uploads/2021/07/dolce-mace4-prover9.zip, where one can get Prover9/Mace4 input file that indeed contains a consistent first-order theory. Using the LADR-2009-11A application I converted this file into the TPTP syntax, which Vampire "understands". Then I ran the process described above.

Table 1 shows all rough identity criteria found by Vampire. No relation in DOLCE is tautological in the sense of IC6. Tables 2 – 4 show the refined criteria. Comparing these results one can easily note that DOLCE has no fully refined identity criteria for any of its types, despite having quite a few of rough ones. Note that table 1 does not list all undecided cases, but only those that pertain to the types and relations listed there. In fact there were 20 undecided questions about identity criteria in DOLCE (out of 2306 sent to the Vampire prover).

The full search took 8948 seconds, with the average search for a single identity criterion candidate lasting 3.88 seconds. All artefacts produced in this process can be found at https://github.com/mereolog/isniffer/tree/main/outputs/dolce.

Although some rough identity criteria in DOLCE are rather direct consequences of its axioms, others are less obvious. For instance, compare axiom direct_quality_Ad43, which gives rise to dqt@2 being an identity criterion to axiom definition_unicity_a8, which does not yield a similar result for df@2.

(all X all Y all V ((dqt(X,Y) & dqt(X,V)) -> ((Y)=(V)))) # label(direct_quality_Ad43).

(all X all Y all Z ((df(X,Y) & df(X,Z)) -> (Y=Z))) # label(definition_unicity_a8).

Also note that such criteria as ov@1, ov@2, p@1 and p@1 has reflexive relations as their bases – this can be seen as a breach of the non-circularity constraint (IC5). Consider the theorem that establishes p@1 (being an improper part) as a criterion of identity for pd (perdurants):

$$\forall x, y[\mathtt{pd}(x) \wedge \mathtt{pd}(y) \rightarrow [(\forall z(\mathtt{p}(x,z) \equiv \mathtt{p}(y,z)) \rightarrow x = y)]. \tag{14}$$

Now the conceptual problem with this principle is that one of the improper parts of every perdurant is this very perdurant, so 14 can be unbundled into:

$$\forall x, y[\mathtt{pd}(x) \wedge \mathtt{pd}(y) \rightarrow [(\forall z(z = x \vee \mathtt{pp}(x,z) \equiv z = y \vee \mathtt{pp}(y,z)) \rightarrow x = y)]. \tag{15}$$

where 'pp' denotes the relation of proper parthood. Since '$\forall z(z = x \vee \mathtt{pp}(x,z) \equiv z = y \vee \mathtt{pp}(y,z))$' is equivalent (within DOLCE) to '$x = y \vee \forall z(\mathtt{pp}(x,z) \equiv \mathtt{pp}(y,z))$', 15 amounts to, informally speaking, a tautological claim that two perdurants are identical if they are identical or share the same proper parts. In sum, it seems that the list of identity criteria requirements from [8] needs to be reinterpreted or extended for the type of criteria discussed in this paper.

	atp@1	atp@2	dif@1	dif@2	dif@3	dqt@1	dqt@2	ov@1	ov@2	p@1	p@2	sum@1	sum@2	sum@3
ab		?	?	+	+			+	+	+	+	+	+	+
acc		?	+											
ar		?	+											
at	+	+						+	+	+	+	+	+	+
nped							+							
pd		?	?	+	+		+	+	+	+	+	+	+	+
ped							+							

Table 1. Rough identity criteria in DOLCE

	atp@1	atp@2	dif@1	dif@2	dif@3	dqt@1	dqt@2	ov@1	ov@2	p@1	p@2	sum@1	sum@2	sum@3
ab														
acc														
ar														
at	+	+												
nped														
pd														
ped														

Table 2. IC3-refined identity criteria in DOLCE

	atp@1	atp@2	dif@1	dif@2	dif@3	dqt@1	dqt@2	ov@1	ov@2	p@1	p@2	sum@1	sum@2	sum@3
ab				+	+									
acc			+											
ar			+											
at														
nped														
pd				+	+		+							
ped							+							

Table 3. IC7-refined identity criteria in DOLCE

	atp@1	atp@2	dif@1	dif@2	dif@3	dqt@1	dqt@2	ov@1	ov@2	p@1	p@2	sum@1	sum@2	sum@3
ab														
acc			+											
ar			+											
at														
nped							+							
pd														
ped							+							

Table 4. IC8-refined identity criteria in DOLCE

Comparing table 3 to table 1 it occurred to me that constraint IC7 may be seen as too restrictive – perhaps being ⊆-maximal instead of being ⊆-greatest would be sufficient. Then the mereological relations as having the same parts or overlapping the same particulars could then become IC7-refined identity criteria in DOLCE.

4.2. UFO

UFO The TPTP formalisation of UFO is provided via the https://github.com/unibz-core/ufo-formalization.git repository. The commit 2da0df3 there, which I made use of in this paper, still contains a typo in one of the axioms – I fixed it as described in [5].

Before I present the detailed outcomes of the search for identity criteria in UFO I should mention three formal issues that this search "discovered". First, the following two relations were found to be tautological in the sense of IC6:

$$\texttt{mediates@1}(x,y) \equiv \forall z[\texttt{mediates}(x,z) \equiv \texttt{mediates}(y,z)]. \tag{16}$$

$$\texttt{mediates@2}(x,y) \equiv \forall z[\texttt{mediates}(z,x) \equiv \texttt{mediates}(z,y)]. \tag{17}$$

In other words, the following two formulae are theorems of UFO:

$$\forall x,y,z[\texttt{mediates}(x,z) \equiv \texttt{mediates}(y,z)]. \tag{18}$$

$$\forall x,y,z[\texttt{mediates}(z,x) \equiv \texttt{mediates}(z,y)]. \tag{19}$$

The proofs of these facts can be found here and here. I suspect that these are unintended consequences of the UFO axioms.

Secondly, initially the type relator was found to have more than 100 identity criteria. Since this looked wrong to me, I checked that this type is not satisfiable, it is interpreted as the empty set in all UFO models. The proof can be found here.

Thirdly, the following types can be proved to be (pairwise) equivalent (see: proofs in https://github.com/mereolog/isniffer/tree/main/outputs/ufo/subsumptions):

1. moment, intrinsicMoment, momentType and intrinsicMomentType ;
2. relator, relatorKind, and relatorType;
3. semiRigidNonSortal, mixin, semiRigid and rigidNonSortal.

Again, although such equivalences are acceptable from the formal point of view, they look to me as unintended consequences of the UFO axioms.

Table 5 shows all rough identity criteria found by Vampire – except for mediates@1 and mediates@2 as explained above. Tables 6 – 8 show the refined criteria. Comparing these results one can easily verify that UFO has qualndividuaOf@1 as the fully refined identity criterion for quaindividual:

$$\forall x,y[\texttt{quaIndividual}(x) \land \texttt{quaIndividual}(y) \rightarrow (\forall z(\texttt{quaIndividualOf}(x,z) \equiv \texttt{quaIndividualOf}(y,z)) \rightarrow x = y)] \tag{20}$$

Note that table 5 does not list all undecided cases, but only those that pertain to the types and relations listed there. In fact there were 304 undecided questions about identity criteria in UFO (out of 3392 sent to the Vampire prover).

	ifd@4	iof@2	overlap@1	overlap@2	partOf@1	partOf@2	quaIndividualOf@1	specializes@1	specializes@2	sum@1	sum@2	sum@3
kind	+	+						+	+			
quaIndividual							+					
thing			+	+	+	+				+	+	+

Table 5. Rough identity criteria in UFO

	ifd@4	iof@2	overlap@1	overlap@2	partOf@1	partOf@2	quaIndividualOf@1	specializes@1	specializes@2	sum@1	sum@2	sum@3
kind	+	+						+	+			
quaIndividual							+					
thing			+	+	+	+				+	+	+

Table 6. IC3-refined identity criteria in UFO

	ifd@4	iof@2	overlap@1	overlap@2	partOf@1	partOf@2	quaIndividualOf@1	specializes@1	specializes@2	sum@1	sum@2	sum@3
kind	+	+						+	+			
quaIndividual							+					
thing												

Table 7. IC7-refined identity criteria in UFO

	ifd@4	iof@2	overlap@1	overlap@2	partOf@1	partOf@2	quaIndividualOf@1	specializes@1	specializes@2	sum@1	sum@2	sum@3
kind												
quaIndividual							+					
thing												

Table 8. IC8-refined identity criteria in UFO

The full search took 38196 seconds with the average search for a single identity criterion candidate lasting 11.26 seconds. All artefacts produced in the quest for UFO identity criteria can be found at https://github.com/mereolog/isniffer/tree/main/outputs/ufo.

5. Discussion

The main goal of this paper was to investigate another application of theorem provers in the context of formal applied ontology, where the first, obvious application is a standard test of consistency. It was thought of as an experiment in a green field area of research, so I did not expect any groundbreaking discoveries. And indeed the results may be considered disappointing.

The fact that the count of refined identity criteria in upper-level ontologies is so low may be surprising even if we keep in mind that the results reported here concern only indiscernibility-driven criteria of identity. However, this fact is mainly due to rather restrictive constraints stipulated in [8]: IC1 – IC9. Then one may interpret these results as a motive to reconsider these constraints as they may exclude too many identity criteria candidates. On the other hand, as I indicated in section 4.1, the list may also require some more restrictive clauses for indiscernibility-driven identity criteria.

A reviewer of this paper emphasized that the intitial scope of identity criteria candidates restricted just to the relations in the signature of a given ontology is a serious restriction of this research. The results I reported illustrate this comment, which points to an obvious extension of this experiment, whereby we would consider some relations defined by means of the relations in the current selection.

The actual cases of rough identity criteria found by the process described in this paper are not particularly illuminating if one considers the axiomatisations of DOLCE and UFO. The respective theorems that define them are rather direct consequences of the axioms, so proofs and models provided by Vampire do not provide any new insights into the actual conceptual contents of the formal ontologies. Nevertheless, I am not aware of any such attempt where one tried to find all identity criteria in an ontology in a systematic way.

This attempt reported in this paper revealed surplus benefits one might gain from such an effort. Namely, the search for identity criteria in UFO draws to our attention suspicious theorems of this theory that are related to the search in question. Thus, even if its results are of limited value, one may find these by-products useful in improving the conceptual and formal qualities of computational ontologies.

Also such an exercise as outlined in this paper provides a list of relations that are not identity criteria, so what matters is not just the list of rough identity criteria for a given ontology, but also a list of those relations that are not. However, the amount of understanding yielded by the latter is defined by the last consideration.

Finally, there is a performance consideration. A consistency check for a formal ontology, as exemplified in [5], is a one-off process, which usually takes a few seconds. Thus the Vampire prover did not spend a lot of time to find a model for DOLCE and UFO, but searching for identity criteria turned out to be more challenging, so many questions about identity criteria were left undecided. The performance of this process has two aspects:

1. a check for a single identity criterion candidate in a formal ontology may take longer than the consistency check for this ontology due to, for example, the fact that the ontology needs to be extended with a set of additional axioms;
2. the number *n* of such checks depends on the number of types, relations, and their arities, e.g., in the case of UFO, one needs to run more than 3000 of those.

So a computer-supported quest for identity criteria within a computational ontology may be rather computationally expensive, at least much more expensive than a simple consistency check. It has already been observed, for instance: [13], that the ”computational price” of each such process, in particular model finding, heavily depends on the number of predicates with higher arities. This dependence could interpret the difference in the length of an average search between DOLCE and UFO. Also it explains why I got many more undecided questions about identity criteria for UFO than for DOLCE: the former contains quite a few predicates with arities higher than the highest arity in the latter.

References

[1] Carrara M, Giaretta P. Identity Criteria and Sortal Concepts. In: Proceedings of the International Conference on Formal Ontology in Information Systems - Volume 2001. FOIS ’01. New York, NY, USA: Association for Computing Machinery; 2001. p. 234–243.

[2] Guarino N, Welty C. 7. In: Green R, Bean CA, Myaeng SH, editors. Identity and Subsumption. Dordrecht: Springer Netherlands; 2002. p. 111-26.

[3] Carrara M, Giaretta P. The many facets of identity criteria. Dialectica. 2004;58(2):221-32.

[4] Quine WV. Ontological Relativity and Other Essays. Columbia University Press; 1969.

[5] Garbacz P. ". In: Sales TP, Hedblom MM, Tan H, Righetti G, Kutz O, Glauer M, et al., editors. Proceedings of the Joint Ontology Workshops 2022 Episode VIII: The Svear Sommar of Ontology, Jönköping, Sweden, August 15-19, 2022. vol. 3249 of CEUR Workshop Proceedings. CEUR-WS.org; 2022. Available from: http://ceur-ws.org/Vol-3249/paper1-FOUST.pdf.

[6] Fitelson B, Zalta EN. Steps toward a computational metaphysics. Journal of Philosophical Logic. 2007;36(2):227-47.

[7] Williamson T. Identity and Discrimination. Basil Blackwell; 1990.

[8] Carrara M, Gaio S. Towards a formal account of identity criteria. In: Majda Trobok BZ Nenad Miscevic, editor. Between logic and reality. Springer; 2012. p. 227-42.

[9] Lowe EJ. One-Level versus Two-Level Identity Criteria. Analysis. 1991;51(4):192-4.

[10] Clercq RD. Locke’s Principle is an Applicable Criterion of Identity. Noûs. 2013;47(4):697-705.

[11] Oderberg DS. 2. In: Continuity as the Criterion of Identity over Time: Continuity without Stages? London: Palgrave Macmillan UK; 1993. p. 37-62.

[12] Sutcliffe G. The Logic Languages of the TPTP World. Logic Journal of the IGPL. 2022.

[13] Stephen S, Hahmann T. Identifying Bottlenecks in Practical SAT-Based Model Finding for First-Order Logic Ontologies with Datasets. Proceedings of the AAAI Conference on Artificial Intelligence. 2019 07;33:10039-40.