Applied Ontology 0 (2011) 1–0
IOS Press

The modular structure of an ontology: an empirical study *

Chiara Del Vescovo ??, Bijan Parsia ??, Uli Sattler ?? and Thomas Schneider ??

[a] *School of Computer Science, University of Manchester, Oxford Road, Manchester, M13 9PL, UK*
Email: {delvescc,bparsia,sattler}@cs.man.ac.uk

[b] *Fachrichtung Informatik, Universität des Saarlandes, Campus E1.3, Saarbrücken, Germany*
Email: schneider@ps.uni-saarland.de

Abstract Efficiently extracting a module from a given ontology that captures all the ontology's knowledge about a set of specified terms is a well-understood task. This task can be based, for instance, on locality-based modules.

In contrast, extracting *all* modules of an ontology is computationally difficult because there can be exponentially many. However, it is reasonable to assume that, by revealing the modular structure of an ontology, we can obtain information about its topicality, connectedness, structure, superfluous parts, or agreement between actual and intended modeling. Furthermore, incremental reasoning makes use of a number of, although not all possible, modules of an ontology.

We report on experiments to estimate the number of modules of real-life ontologies. We also evaluate the modular structure of ontologies that we succeeded to fully modularize. In that evaluation, we look at the number and sizes of the modules, as well as the relation between module size and number and size of signatures that lead to the module. Chances are that the understanding we report about small ontologies can be applied to all ontologies.

Keywords: Ontologies, description logics, module extraction, syntactic locality, ontology comprehension

1. Introduction

Why modularize an ontology? In software engineering, modularly structured systems are desirable, all other things being equal. Given a well-designed modular program, it is generally easier to process, modify, and analyze it and to reuse parts by exploiting the modular structure. As a result, support for modules (or components, classes, objects, packages, aspects) is a commonplace feature in programming languages.

Ontologies are computational artefacts akin to programs and, in notable examples, can get quite large as well as complex, which suggests that exploiting modularity might be fruitful, and research into modularity for ontologies has been an active area for ontology engineering. Recently, a lot of effort has gone into developing *logically sensible* modules, that is, modules which offer strong logical guarantees for intuitive modular properties. One such guarantee is called *coverage* and means that the module captures all the ontology's knowledge about a given set of terms (signature)—a kind of dependancy isolation. A module in this sense is therefore a subset of the axioms in an ontology that provides coverage for a signature, and each possible signature determines such a module. Coverage is provided by modules based on conservative extensions, but also by efficiently computable approximations, such as modules based on syntactic locality **?**.

The task of extracting one such module given a signature, which we call GetOne in this section, is well understood and starting to be deployed in standard ontology development environments, such as Protégé 4,[1] and online.[2] The extraction of locality-based modules has already been effectively used in the field for ontology reuse **?** as well as a subservice for incremental reasoning **?**.

While GetOne is an important and useful service, it, by itself, tells us nothing about the modular structure of the ontology as a whole. The modular structure is determined by the set of *all* modules and their inter-

*This work has been supported by the UK EPSRC grant no. EP/E065155/1.

[1] `http://www.co-ode.org/downloads/protege-x`

[2] `http://owl.cs.manchester.ac.uk/modularity`

relations, or at least a suitable subset thereof. We call the task of a-posteriori determining the modular structure of an ontology GetStruct and, in order to determine that structure, we investigate here the task GetAll of extracting all modules. While GetOne is well-understood and often computationally cheap, GetAll has hardly been examined for module notions with strong logical guarantees, with the work described in **?** being a promising exception. GetOne also requires the user to know in advance the right set of terms to input to the extractor: we call this a *seed* signature for the module and note that one module can have several such seed signatures. Since there are non-obvious relations between the final signature of a module and its seed signature, users are often unsure how to generate a proper request and confused by the results. If they had access to the overall modular structure of the ontology determined by GetAll, they could use it to guide their extraction choices. In general, supported by the experience described in **?**, we believe that, by revealing the modular structure of an ontology, we can obtain information about its topicality, connectedness, structure, superfluous parts, or agreement between actual and intended modeling. Our use-cases include: for ontology engineers, the possibility of checking the ontology design—for example, if the module relative to some terms corresponds to the intuitive "knowledge encapsulation" about that term; for end users, the possibility to support the understanding of what the ontology deals with, and where the topic they want to focus on is placed within the ontology.

In the worst case, the number of all modules of an ontology is exponential in the number of terms or axioms in the ontology, in fact in the minimum of these numbers. Hence, it is possibly the case that ontologies have too many modules to extract all of them, even with an optimized extraction methodology. Even with only polynomially many modules, there may be too many for direct user inspection. Then, some other form of analysis would have to be designed.

We report on experiments to obtain or estimate this number and to evaluate the modular structure of an ontology where we succeeded to compute it.

Related work. One solution to GetStruct is described in **??** via partitions related to $\mathcal{E}$-connections. The resulting modules are disjoint, and this technique is of limited applicability—when it succeeds, it divides an ontology into three kinds of modules: (A) those which import vocabulary from others, (B) those whose vocabulary is imported, and (C) isolated parts. In experiments and user experience, the numbers of parts extracted were quite low and often corresponded usefully to user understanding. For instance, the tutorial ontology Koala, consisting of 42 logical axioms, is partitioned into one A-module about animals and three B-modules about genders, degrees and habitats.

It has also been shown in **?** that certain combinations of these parts provide coverage. For Koala, such a combination would still be the whole ontology. In general, partitions were observed to be too coarse grained; sometimes extraction resulted in a single partition even though the ontology seemed well structured. Furthermore, the robustness properties of the parts (e.g., under vocabulary extension) are not as well-understood as those of locality-based modules. Finally, there is only a preliminary implementation of the partition algorithm[3]. However, partitions share efficient computability with locality-based modules.

Another approach to GetStruct is described in **?**. It underlies the tool ModOnto, which aims at providing support for working with ontology modules that is similar to, and borrows intuitions from, software modules. This approach is logic-based and a-posteriori but, to the best of our knowledge, it has not been examined whether such modules provide coverage in the above sense. Furthermore, ModOnto does not aim at obtaining *all* modules from an ontology.

Another procedure for partitioning an ontology is described in **?**. However, this method only takes the concept hierarchy of the ontology into account and can therefore not provide the strong logical guarantee of coverage.

Among the a-posteriori approaches to GetOne, some provide logical guarantees such as coverage, and others do not. The latter are not of interest for this paper. The former are usually restricted to DLs of low expressivity, where deciding conservative extensions—which underly coverage—is tractable. Prominent

[3] Partitioning is implemented in Swoop (`http://code.google.com/p/swoop/`), but it turned out that this implementation is incomplete: for the ontologies we tried, not all axioms were included in the partition, and some of the links between the parts were erroneous. Therefore, comparisons between partitions and our modularization technique are future work.

examples are the module extraction feature of CEL **?** and the system MEX **?**. However, we aim at an approach that covers DLs up to OWL 2.

There are a number of logic-based approaches to modularity that function a-priori, i.e., the modules of an ontology have to be specified in advance by features that are added to the underlying (description) logic and whose semantics is well-defined. These approaches often support distributed reasoning; they include C-OWL **?**, $\mathcal{E}$-connections **?**, Distributed Description Logics **?**, and Package-Based Description Logics **?**. Even in these cases, however, we may want to understand the modular structure of the syntactically delineated parts. Furthermore, with imposed structure, it is not always clear that that structure is correct. Decisions about modular structure have to be taken early in the modeling which may enshrine misunderstandings. Examples were reported in **?**, where user attempts to capture the modular structure of their ontology by separating the axioms into separate files were totally at odds with the analyzed structure.

Outline. In the following, we will report on analytic and experimental results towards estimating the number of modules of an ontology. This is a first step towards investigating whether GetAll is feasible for practical purposes. We will see that this is not the case for the unrestricted form of GetAll because module numbers are too large to handle in most cases.

In the theoretical part, we look at ontologies that are only subsumption hierarchies and examine the characteristics of such hierarchies that lead to the combinatorial explosion in the module number. We will see that already very simple shapesm of hierarchies can lead to an exponential number of modules.

The experimental part consists of extracting all modules from several ontologies. We have considered three notions of modules based on syntactic locality—they all provide coverage, but differ in the size of the modules and in other useful properties of modules, see **?**—and extracted such modules for all subsets of the terms in the respective ontology. We are mainly interested in module *numbers* rather than sizes or interrelations: the main concern is whether the suspected combinatorial explosion occurs. In order to test the latter, we have sampled subsets of each ontology and performed a full modularization on each subontology, measuring the relation between module number and subontology size for each ontology. We also report on several approaches to reduce the number of modules to the most "interesting" ones.

Finally, we investigate whether the overall number of modules can be estimated by randomly sampling seed signatures and recording how often the corresponding modules are identical.

Additional material for the evaluation of the experiments, such as spreadsheets and charts, are available online **?**.

2. Preliminaries

Underlying description logics. We assume the reader to be familiar with OWL and the underlying description logics (DLs) **??**. We consider an ontology to be a finite set of axioms, which are of the form $C \sqsubseteq D$ or $C \equiv D$, where C, D are (possibly complex) concepts, or $R \sqsubseteq S$, where R, S are (possibly inverse) roles. Since we are interested in the logical part of an ontology, we disregard non-logical axioms. However, it is easy to add the corresponding annotation and declaration axioms in retrospect once the logical part of a module has been extracted. This is included in the publicly available implementation of locality-based module extraction in the OWL API.[4]

Let $\mathsf{N_C}$ be a set of concept names, and $\mathsf{N_R}$ a set of role names. A *signature* Σ is a set of terms, i.e., $\Sigma \subseteq \mathsf{N_C} \cup \mathsf{N_R}$. We can think of a signature as specifying a topic of interest. Axioms that only use terms from Σ can be thought of as "on-topic", and all other axioms as "off-topic". For instance, if $\Sigma = \{\mathsf{Animal}, \mathsf{Duck}, \mathsf{Grass}, \mathsf{eats}\}$, then $\mathsf{Duck} \sqsubseteq \exists \mathsf{eats}.\mathsf{Grass}$ is on-topic, while $\mathsf{Duck} \sqsubseteq \mathsf{Bird}$ is off-topic.

Any concept or role name, ontology, or axiom that uses only terms from Σ is called a Σ-*concept*, Σ-*role*, Σ-*ontology*, or Σ-*axiom*. Given any such object X, we call the set of terms in X the *signature of* X and denote it with $\widetilde{X}$.

[4] `http://owlapi.sourceforge.net`

Conservative extensions and locality. Conservative extensions (CEs) capture the above described encapsulation of knowledge. They are defined as follows. Let $\mathcal{L}$ be a DL, $\mathcal{M} \subseteq \mathcal{O}$ be $\mathcal{L}$-ontologies, and Σ be a signature.

1. $\mathcal{O}$ is a *deductive* Σ*-conservative extension* (Σ*-dCE*) of $\mathcal{M}$ w.r.t. $\mathcal{L}$ if for all GCI axioms α over $\mathcal{L}$ with $\widetilde{\alpha} \subseteq \Sigma$, it holds that $\mathcal{M} \models \alpha$ if and only if $\mathcal{O} \models \alpha$.
2. $\mathcal{M}$ is a *dCE-based module for* Σ of $\mathcal{O}$ if $\mathcal{O}$ is a Σ-dCE of $\mathcal{M}$ w.r.t. $\mathcal{L}$.

Unfortunately, CEs are hard or even impossible to decide for many DLs, see **???**. Therefore, approximations have been devised. We focus on *syntactic locality* (here for short: locality). Locality-based modules can be efficiently computed and provide coverage, that is, they capture *all* the relevant entailments, but not necessarily *only* those **??**. Although locality is defined for the DL $\mathcal{SHIQ}$, it is straightforward to extend it to $\mathcal{SHOIQ}(D)$ (see **??**), and the implementation of locality-based module extraction in the OWL API. We are using the notion of locality from **?**.

An axiom α is called *syntactically* $\bot$*-local* ($\top$*-local*) *w.r.t. signature* Σ if it is of the form $C^{\bot} \sqsubseteq C$, $C \sqsubseteq C^{\top}$, $R^{\bot} \sqsubseteq R$ ($R \sqsubseteq R^{\top}$), or $\mathsf{Trans}(R^{\bot})$ ($\mathsf{Trans}(R^{\top})$), where C is an arbitrary concept, R is an arbitrary role name, $R^{\bot} \notin \Sigma$ ($R^{\top} \notin \Sigma$), and $C^{\bot}$ and $C^{\top}$ are from $\mathrm{Bot}(\Sigma)$ and $\mathrm{Top}(\Sigma)$ as defined in Figure **??** (a) (Figure **??** (b)).

(a) $\bot$*-Locality*

Let $A^{\bot}, R^{\bot} \notin \Sigma,\ C^{\bot} \in \mathrm{Bot}(\Sigma),\ C^{\top}_{(i)} \in \mathrm{Top}(\Sigma),\ \bar{n} \in \mathbb{N} \setminus \{0\}$

$$\mathrm{Bot}(\Sigma) ::= A^{\bot} \mid \bot \mid \neg C^{\top} \mid C \sqcap C^{\bot} \mid C^{\bot} \sqcap C \mid \exists R.C^{\bot} \mid \geqslant \bar{n} R.C^{\bot} \mid \geqslant \bar{n} R^{\bot}.C$$

$$\mathrm{Top}(\Sigma) ::= \top \mid \neg C^{\bot} \mid C^{\top}_1 \sqcap C^{\top}_2 \mid \geqslant 0 R.C$$

(b) $\top$*-Locality*

Let $A^{\top}, R^{\top} \notin \Sigma,\ C^{\bot} \in \mathrm{Bot}(\Sigma),\ C^{\top}_{(i)} \in \mathrm{Top}(\Sigma),\ \bar{n} \in \mathbb{N} \setminus \{0\}$

$$\mathrm{Bot}(\Sigma) ::= \bot \mid \neg C^{\top} \mid C \sqcap C^{\bot} \mid C^{\bot} \sqcap C \mid \geqslant \bar{n} R.C^{\bot}$$

$$\mathrm{Top}(\Sigma) ::= A^{\top} \mid \top \mid \neg C^{\bot} \mid C^{\top}_1 \sqcap C^{\top}_2 \mid \geqslant \bar{n} R^{\top}.C^{\top} \mid \geqslant 0 R.C$$

Figure 1. Syntactic locality conditions

It has been shown in **?** that $\mathcal{M} \subseteq \mathcal{O}$ and all axioms in $\mathcal{O} \setminus \mathcal{M}$ being $\bot$-local (or all axioms being $\top$-local) w.r.t. $\Sigma \cup \widetilde{M}$ is sufficient for $\mathcal{O}$ to be a Σ-dCE of $\mathcal{M}$. The converse does not hold: e.g., the axiom $A \equiv B$ is neither $\bot$- nor $\top$-local w.r.t. $\{A\}$, but the ontology $\{A \equiv B\}$ is an $\{A\}$-dCE of the empty ontology.

It is described in **?** how to obtain modules of $\mathcal{O}$ for $\top$- and $\bot$-locality. We are using the notions of $\top$-, $\bot$-, $\top\bot^*$- and $\bot\top^*$-modules from (**?**, Def. 4). That is, given an ontology $\mathcal{O}$, a *seed signature* Σ and a module notion $x \in \{\top, \bot, \top\bot^*, \bot\top^*\}$, we denote the x-module of $\mathcal{O}$ w.r.t. Σ by x-mod$(\Sigma, \mathcal{O})$. If we do not specify x, we generally speak of a *locality-based* module. It is straightforward to show that $\top\bot^*$-mod$(\Sigma, \mathcal{O}) =$ $\bot\top^*$-mod$(\Sigma, \mathcal{O})$ for each $\mathcal{O}$ and Σ. In contrast, $\top$- and $\bot$-modules do not have to be equal—in fact, the former are usually larger than the latter. Through the nesting, $\top\bot^*$-mod$(\Sigma, \mathcal{O})$ is always contained in $\top$-mod$(\Sigma, \mathcal{O})$ and $\bot$-mod$(\Sigma, \mathcal{O})$. Finally, we want to point out that, for $\mathcal{M} = x$-mod$(\Sigma, \mathcal{O})$, neither $\Sigma \subseteq \widetilde{\mathcal{M}}$ nor $\widetilde{\mathcal{M}} \subseteq \Sigma$ needs to hold.

The following property of locality-based modules will be of interest for our modularization. For $x \in \{\bot, \top\}$, Proposition **??** has been shown in **?**. The transfer to nested modules is straightforward.

Proposition 1. *Let* $\mathcal{O}$ *be an ontology,* Σ *be a signature,* $x \in \{\bot, \top, \top\bot^*\}$*; let* $\mathcal{M} = x$*-mod*$(\Sigma, \mathcal{O})$ *and* Σ' *be a signature with* $\Sigma \subseteq \Sigma' \subseteq \Sigma \cup \widetilde{\mathcal{M}}$*. Then x-mod*$(\Sigma', \mathcal{O}) = \mathcal{M}$*.*

Genuine modules. In order to limit the overall number of modules, we introduce the notion of a *genuine module*. Intuitively, a given module $\mathcal{M}$ of an ontology is *fake* if it can be partitioned into a set $\{\mathcal{M}_1, \ldots, \mathcal{M}_n\}$ of smaller modules such that each "relevant" entailment of $\mathcal{M}$ follows from some $\mathcal{M}_i$.

Since the definition of relevance of an entailment within a module is still in progress, we use a computable approximation, described in Definition **??**. We first introduce some useful notions. Let $\mathcal{O}$ be an ontology and $\mathfrak{M}$ be the set of all modules of $\mathcal{O}$. An atomic concept C is called *top-level for $\mathcal{M}$ (bottom-level for $\mathcal{M}$)* if $\mathcal{O} \models A \sqsubseteq C$ ($\mathcal{O} \models C \sqsubseteq A$) for all atomic concepts $A \in \widetilde{M}$. A set $\{\Sigma_1, \dots, \Sigma_n\}$ of signatures is called *$\mathcal{M}$-almost pairwise disjoint* if every two signatures Σ_i, Σ_j with $i \neq j$ are disjoint or share at most one symbol, which is an atomic concept, and if the set of all these shared atomic concepts contains at most one top-level and at most one bottom-level concept for $\mathcal{M}$. A module $\mathcal{M} \in \mathfrak{M}$ is *fake* if there exist modules $\mathcal{M}_1, \dots, \mathcal{M}_n \in \mathfrak{M}$ such that $\mathcal{M} = \mathcal{M}_1 \uplus \dots \uplus \mathcal{M}_n$ and the set $\{\widetilde{\mathcal{M}_1}, \dots, \widetilde{\mathcal{M}_n}\}$ is *$\mathcal{M}$-almost pairwise disjoint*. Otherwise $\mathcal{M}$ is called *genuine*.

In particular, if a module is fake, then it consists of disjoint modules whose signatures *almost* disjoint. For example, in Koala, we have a fake module about habitat that consists of a rainforest and a dryforest submodule, which only overlap in the term habitat and do not share any other terms and no axioms. Fake modules are uninteresting because $\mathcal{M}$ being fake means that different seed signatures of the $\mathcal{M}_i$ do not interact with each other. Given that often the overall number of modules appears to grow exponentially with the size of the subontology, a natural question arising is whether this is only caused by the fact that there are exponentially many fake modules.

3. Analytic module number determination

In general, an ontology $\mathcal{O}$ can have exponentially many modules because every subset of $\mathcal{O}$'s signature can lead to a distinct module.

Fortunately, we have good reasons to believe that there are significantly fewer modules than seed signatures in realistic ontologies: first, Proposition **??** says that, given the locality-based module $\mathcal{M} = x\text{-mod}(\Sigma, \mathcal{O})$, every seed signature Σ' that extends Σ and is a subset of $\Sigma \cup \widetilde{\mathcal{M}}$ yields the same module $\mathcal{M}$. Second, even if two seed signatures Σ and Σ' are not in such a relationship, the modules for Σ and Σ' may still coincide.

Theoretically, the number of modules of an ontology can therefore range between 1 (all modules coincide) and $2^{|\Sigma|}$, where $|\Sigma|$ is the cardinality of Σ. Ontologies of arbitrary size with exactly one module are, for instance, those that consist of only non-local axioms or only tautologies.

In oder to find out whether GetAll has a chance to be useful in practice at all, we will need to identify families of ontologies with an asymptotic behaviour of the overall module number that is below exponential, i.e., polynomial. Our current understanding is restricted to ontologies that are pure subsumption hierarchies, i.e., which consist of axioms $A \sqsubseteq B$, where A, B are concept names. We call them *simple taxonomies* here. A simple taxonomy $\mathcal{S}$ can be considered as a graph whose nodes are the concept names in $\mathcal{S}$'s signature, and whose edges are given by the subsumption axioms in $\mathcal{S}$. For the remainder of this section, we use "(subsumption) graph", "node", "edge" interchangeably with "simple taxonomy", "concept name", "subsumption axiom". We look at different shapes of such *subsumption graphs* and how the number of modules is related to the shape.

The extraction algorithm for locality-based modules has the following consequences, given a subsumption graph $\mathcal{S}$ and a seed signature Σ that contains the nodes A, B.

1. $\bot$-mod$(\Sigma, \mathcal{S})$ contains all paths in $\mathcal{S}$ that start in A or B.
2. $\top$-mod$(\Sigma, \mathcal{S})$ contains all paths in $\mathcal{S}$ that end in A or B.
3. $\top\bot^*$-mod$(\Sigma, \mathcal{S})$ contains all paths in $\mathcal{S}$ that start in A and end in B.

To understand (1), consider an outgoing edge of A, which corresponds to an axiom $\alpha = A \sqsubseteq A'$. This axiom cannot be $\bot$-local w.r.t. a signature that contains A. Therefore α is included in the module, which extends the module signature with A'. All axioms corresponding to outgoing edges of A' are now not $\bot$-local w.r.t. the extended signature and included in the module etc. The general consequence of (3) is even stronger:

4. $\top\bot^*$-mod$(\Sigma, \mathcal{S})$ consists of exactly the paths between pairs of nodes in Σ.

It is now easy to see that every graph $\mathcal{S}$ that consists of exactly one path has a quadratic number of modules: Let $\mathcal{S}$ be $A_0 \to A_1 \to \cdots \to A_n$. Then, due to (3) and (4), every module is a subpath, and every subpath is a module. Therefore there are $\frac{n(n+1)}{2}$ $\top\bot^*$-modules of $\mathcal{S}$. The same conclusion can be drawn for the number of dCE-based modules of $\mathcal{S}$.

We now consider graphs that consist of multiple paths which have a common start node, i.e., $\mathcal{S}$ is the union of the paths $A \to B_1^j \to \cdots \to B_{d_j}^j$ with $j = 1,\ldots,w$ and $w \geqslant 1$. We call this shape a *hand* and the j-th path the j-th *finger*; then w stands for the width of the graph, and the maximum of the d_j for its depth. Now every module is a set of paths that contains at most one subpath of every finger. However, not every such set is a module: A is either on every subpath in the module, or on none. If we assume the fingers to have equal depth $d = d_1 = \cdots = d_w$, then the number of modules of the hand is $\left(\frac{d(d-1)}{2}\right)^w + 2^w$, which is still quadratic in the depth of the graph, but already exponential in its width. This suggests that the width of the subsumption graph may be the source of an exponential behaviour of the module number. Before we consider other graph shapes to test this hypothesis, we observe that the number of *genuine* modules of a hand, according to Definition **??**, is quadratic in the depth of the hand as well, but linear in its width. The reason is that the combination of subpaths of different fingers is fake; therefore there are $\left(\frac{d(d+1)}{2}\right) \cdot w$ genuine modules.

From here, rewrite.

It is not hard to observe that there are simple families of ontologies that already have exponentially many genuine modules, i.e., in the worst case, an exponential number of modules cannot be avoided. For instance, each taxonomy of the form $\mathcal{T}_n = \{C_i \sqsubseteq B \mid 1 \leqslant i \leqslant n\}$ has exponentially many (locality and dCE based) modules: each subset of $\{C_1,\ldots,C_n\}$ as a seed signature leads to a different $\bot$-module, which contains the axiom $C_i \sqsubseteq B$ if and only if C_i is in this set. For $\top$-, $\top\bot^*$- and dCE-based modules, we can add B to each of these subsets and argue in the same way. This example taxonomy still has only linearly many genuine modules—namely all $\{C_i \sqsubseteq B\}$. However, if we add the axiom $B \sqsubseteq A$ to $\mathcal{T}_n$, we obtain an ontology having an exponential number of genuine modules. For every set $J \subseteq \{1,\ldots,n\}$, the module $\mathcal{M}_J := x\text{-mod}\big(\{A\} \cup \{C_j \mid j \in J\},\ \mathcal{T}_n\big)$ is genuine for $x = \top, \bot, \top\bot^*$. A relaxation of the genuinity definition does not help because we can replace the axiom $B \sqsubseteq A$ with a longer inclusion chain or an even more complex inclusion structure.

...

Conclusion: current understanding restricted to pure class graphs. Do other features increase or decrease module numbers? Directions?

Other patterns that lead to exponentially many genuine modules include atomic disjointness axioms and axioms involving simple existential restrictions and conjunctions. Consider, for example, the taxonomy family $\mathcal{T}_n' = \mathcal{T}_n \cup \{D_i \sqsubseteq C_i \mid 1 \leqslant i \leqslant n\}$, where each $\mathcal{T}_n'$ has only $3n+1$ genuine modules, namely each nonempty subpath of any of the n paths in the concept inclusion hierarchy plus the empty module. As soon as we add axioms $\{C_i \sqsubseteq \neg C_j \mid 1 \leqslant i < j \leqslant n\}$ or $\{C_i \sqsubseteq \exists R_{ij}.C_j, D_i \sqsubseteq \exists S_{ij}.D_j \mid 1 \leqslant i < j \leqslant n\}$ or $\{C_i \sqcap X_{ij} \sqsubseteq C_j, D_i \sqcap Y_{ij} \sqsubseteq D_j \mid 1 \leqslant i < j \leqslant n\}$, all combinations of such paths become genuine modules.

Thus, while the worst case number of modules is high, it is not analytically impossible that real ontologies would have a reasonable number of modules. Unfortunately, empirically, as discussed in Section **??**, this does not seem to be the case.

...

4. Experimental module number determination

4.1. Description of the experiments

Ontologies. We performed the experiments on several existing ontologies that we consider to be well designed and sufficiently diverse. “Well designed” means that these ontologies cover a specific domain to a certain level of detail; they are axiomatically rich, for example, they do not only connect terms via atomic subsumptions, which would make module extraction rather uninteresting because the terms in the signature of a module would hardly cause other terms to be included in the module. We concentrate on well-designed ontologies because we want to *understand* their structure. "Diverse" means that these ontologies have different sizes, expressivities, ratios of axiom and term numbers, and cover different domains.

We also selected some ontologies which have had successful and insightful full modularization by other techniques (in particular, Koala and OWL-S). Unfortunately, we have had to restrict our attention to rather small ontologies for practical reasons. However, the selection constitutes a set of ontologies which are commonly discussed in ontology engineering circles and for which people have strong instincts about their modular structure.

Figure **??** gives an overview; most of these ontologies can be found in the TONES ontology repository[5].

Name	DL expressivity	#Axioms[a]	#Terms[b]
Koala	$\mathcal{ALCON}(D)$	42	25
Mereology	$\mathcal{SHIN}$	44	25
University	$\mathcal{SOIN}(D)$	52	39
People	$\mathcal{ALCHOIN}$	108	73
miniTambis	$\mathcal{ALCN}(D)$	173	226
OWL-S	$\mathcal{ALCHOIN}(D)$	277	137
Tambis	$\mathcal{ALCN}(D)$	595	494
Galen	$\mathcal{ALEHF}+$	4,528	3,161

[a] We only count logical axioms here.

[b] We only count atomic concepts as well as abstract and concrete roles here.

Figure 2. Ontologies used in the experiments

Full modularization. Let $\mathcal{O}$ be the ontology to be modularized. Our goal is to find all modules of $\mathcal{O}$, i.e., to compute $\{x\text{-mod}(\Sigma,\mathcal{O}) \mid \Sigma \in \widetilde{\mathcal{O}}\}$. In order to keep track of the seed signatures, we seek an algorithm which, given $\mathcal{O}$ as input, returns a representation of all pairs $(\Sigma,\mathcal{M})$ with $\Sigma \subseteq \widetilde{\mathcal{O}}$ and $\mathcal{M} = x\text{-mod}(\Sigma,\mathcal{O})$.

The most naïve procedure is to simply traverse through all seed signatures Σ, extract the corresponding module and add it to the output. Since there are exponentially many seed signatures, this is not feasible—even for Koala, 2^{25} runs of even the easiest test is unrealistic. However, we can try to rely on Proposition **??**, save the extraction of certain modules, and hope that this leads to significantly less module extractions than the number of seed signatures.

Since a module can have several seed signatures, we represent a module as a pair consisting of $\mathcal{M}$ and the set $\mathcal{S}$ of all *minimal* seed signatures Σ for which $\mathcal{M}$ is a module. Whenever a module for a new seed signature Σ' is to be computed, we first check whether Σ' satisfies $\Sigma \subseteq \Sigma' \subseteq \Sigma \cup \widetilde{\mathcal{M}}$ for some already extracted module $\mathcal{M}$ and some associated minimal seed signature Σ. Only if this is not the case, the module $\mathcal{M}' = x\text{-mod}(\Sigma',\mathcal{O})$ is computed. If $\mathcal{M}'$ coincides with some already extracted module $\mathcal{M}$, then Σ' is added to the set of minimal seed signatures associated with $\mathcal{M}$; otherwise the pair $(\{\Sigma'\},\mathcal{M}')$ is added to the set of extracted modules. This is performed by Algorithm **??**, which calls Alg. **??**.

Algorithm **??** is sound and complete, i.e., the following properties are satisfied, for its input $\mathcal{O}$ and output $\mathfrak{M}$.

1. For each $(\mathcal{S},\mathcal{M}) \in \mathfrak{M}$ and $\Sigma \in \mathcal{S}$, the ontology $\mathcal{M}$ is an x-module of $\mathcal{O}$ w.r.t. Σ.
2. For each $\Sigma \subseteq \widetilde{\mathcal{O}}$ with $\Sigma \neq \emptyset$, there is some $(\mathcal{S},\mathcal{M}) \in \mathfrak{M}$ and some $\Sigma' \in \mathcal{S}$ such that $\Sigma' \subseteq \Sigma \subseteq \Sigma' \cup \widetilde{\mathcal{M}}$.

Soundness is obvious, and completeness can be shown easily using Prop. **??**.

It is now possible to minimize the runtime of this algorithm via several optimizations. One is rather technical and consists in representing axiom sets and signatures via bit vectors, which makes their comparisons fast. Another optimization consists in imposing an order on the terms in the signature of the on-

[5] `http://owl.cs.manchester.ac.uk/repository`

Algorithm 1 Extract all x-modules

```
1:  Input:   an ontology $\mathcal{O}$ with signature $\widetilde{\mathcal{O}}$
2:  Output:  a set $\mathfrak{M} = \{(\mathcal{S}_1, \mathcal{M}_1), \ldots, (\mathcal{S}_n, \mathcal{M}_n)\}$
             of all $x$-modules of $\mathcal{O}$,
             associated with their sets of
             minimal seed signatures (SSigs)

3:  {Start: extract $x$-modules for all singleton SSigs}
4:  $\mathfrak{M} \leftarrow \emptyset$
5:  for all $t \in \widetilde{\mathcal{O}}$ do
6:    $\mathcal{M} \leftarrow$ extract $x$-module of $\mathcal{O}$ w.r.t. $\{t\}$
7:    call integrate($\mathfrak{M}, \{t\}, \mathcal{M}$)
8:  end for

9:  {Extension: iteratively add single terms to SSigs}
10: while $\mathfrak{M}$ contains $(\mathcal{S}, \mathcal{M})$ with marked $\Sigma \in \mathcal{S}$ do
11:   $(\mathcal{S}, \mathcal{M}) \leftarrow$ some elem. of $\mathfrak{M}$ with marked $\Sigma \in \mathcal{S}$
12:   $\Sigma \leftarrow$ some marked element of $\mathcal{S}$
13:   for all $t \in \widetilde{\mathcal{O}} \setminus (\Sigma \cup \widetilde{\mathcal{M}})$ do
14:     $\mathcal{M}' \leftarrow$ extract $x$-module of $\mathcal{O}$ w.r.t. $\Sigma \cup \{t\}$
15:     call integrate($\mathfrak{M}, \Sigma \cup \{t\}, \mathcal{M}'$)
16:   end for
17:   unmark $\Sigma$ in $(\mathcal{S}, \mathcal{M})$
18: end while
19: return $\mathfrak{M}$
```

Algorithm 2
integrate($\mathfrak{M}, \Sigma, \mathcal{M}$)

```
for all $(\mathcal{S}', \mathcal{M}') \in \mathfrak{M}'$ do
  if $\mathcal{M} = \mathcal{M}'$ then
    $\mathcal{S}' \leftarrow \mathcal{S}' \cup \{\Sigma\}$
    mark $\Sigma$ in $(\mathcal{S}', \mathcal{M}')$
    return
  end if
end for
$\mathfrak{M} \leftarrow \mathfrak{M} \cup (\{\Sigma\}, \mathcal{M})$
mark $\Sigma$ in $(\{\Sigma\}, \mathcal{M})$
return
```

tology and, in Line **??**, choosing only those terms t to extend Σ which are lexicographically larger than all terms in Σ. This does not affect completeness and drastically reduces runtime.

Sampling via subsets. In preliminary testing it soon became apparent that even our optimized algorithm would not reasonably terminate on even fairly small ontologies. Since we have a search space exponential in the size of the ontology and potentially exponentially many modules, it was not clear whether the problem was that our algorithm was not sufficiently optimized (so that the search space dominated) or that the output was impossible to generate. Since it is pointless to try to optimize an algorithm for a function whose output is exponentially large in the size of the typical input, it is imperative to determine whether real-world ontologies do have an exponential number of modules. This last question is one goal of the experiments described in this paper.

In order to test the hypothesis that real-life ontologies have an exponential number of modules, we have sampled subsets of different sizes from the ontologies listed in Table **??**. By fully modularizing each of these subsets, we can draw conclusions about the asymptotic relation between its size and the number of modules obtained. Randomly generated subsets would tend to contain unrelated axioms, taken out of the context in which they have been included by the ontology developers. Since unrelated axioms, or ontologies with many unrelated terms, generally yield many modules, it would be harder to justify the hypothesis that real-world ontologies tend to have significantly less than exponentially many modules if we used arbitrary, less coherent subsets.

We have therefore chosen to let each subset be a module for a randomly generated signature—although we are aware that such subsets are more modular than necessary because ontologies are not normally developed modularly. But this is not a problem: it can only cause us to *understate* the number of modules.

We have sampled 10 signatures of each size between 0 and a threshold of 50 (or ontology's signature size if that was smaller). In some cases where the subset sizes were not optimally distributed (e.g., when small subsets were missing), we sampled 30 signatures of each size. For these signatures, we have extracted the $\top\bot^*$-modules, excluding duplicates, and ordered them by size. Then we have fully modularized all subsets in descending order, aborting when a single modularization took longer than a preset timeout of 20, 60 or 600 minutes, see Section **??** for an explanation of that choice. For each subset, we counted the number of all modules and of its genuine modules.

Computer specifications. For the experiments, we used the implementation of locality-based module extraction algorithms in the OWL API, with minor modifications allowing for a more efficient representation of ontology and signature subsets and which neglect non-logical axioms. We ran most of the experiments on a notebook with a 2.4 GHz Intel Core 2 Duo processor, 4 GB RAM, Mac OS 10.5 and Java 1.5. Some computationally intensive processes were run on a PC with two 2.66 GHz Dual-Core Intel Xeon processors, 16 GB RAM with the same software.

4.2. *Results*

Module numbers for full modularization. Figure **??** shows the full modularization of Koala and Mereology for the module types $\top$, $\bot$ and $\top\bot^*$. In the case of $\top\bot^*$, we also determined genuine modules, denoted by $\top\bot^*_g$. In addition to the number of modules, we have listed the runtime and four aggregations of module sizes (i.e., minimum, maximum, average, standard deviation), where "size" refers to the number of logical axioms. Since the number of axioms is a syntax-dependent measure, we plan to include other measures, such as the number of terms and the sum of the sizes of all axioms, in future work.

	Koala				Mereology			
	$\top$	$\bot$	$\top\bot^*$	$\top\bot^*_g$	$\top$	$\bot$	$\top\bot^*$	$\top\bot^*_g$
#Modules	**12**	**520**	**3,660**	**2,143**	**40**	**552**	**1952**	**272**
Time [s]	*0*	*1*	*9*	*34*	*0*	*6*	*158*	*158*
Min size	29	6	0	0	18	0	0	0
Avg size	35	27	23	23	26	25	20	22
Max size	42	42	42	42	40	40	40	38
Std. dev.	4	6	6	6	6	7	8	8

$\top\bot^*_g$ = *genuine* $\top\bot^*$ modules. "Size" = number of logical axioms.

Figure 3. Full modularization of Koala and Mereology

For both ontologies, the number of modules increases from $\top$- via $\bot$- to $\top\bot^*$ modules as expected: as mentioned before, $\top$-modules tend to be bigger, and therefore more modules coincide in this case. However, $\top$-modules are too coarse-grained: most of them comprise almost the whole ontology, and all have a size of at least 29 (69% of Koala) or 18 (41% of Mereology).

The extracted $\bot$-modules yield a more fine-grained picture, although all their sizes for Koala are still above 6 (14%). We already pay for this with an increase in the number of modules by a factor of more than 43 (Koala) and 14 (Mereology). With $\top\bot^*$, smaller modules are included, but for the price of another increase in module numbers by a factor of 7 (Koala) and 3.5 (Mereology). Mereology does not only have fewer $\top\bot^*$ modules than Koala, but also a much smaller proportion of genuine modules (14% versus 59% for Koala). This can be explained with a peculiarity in Mereology's structure: it imports six axioms from the ontology Lkif-top, but only reuses one of the atomic concepts therein. In terms of the intuition behind the definition of of genuine modules, the lower ratio for Mereology reflects the loose connectedness between imported and remaining terms.

Apart from module numbers, another price we pay for a more fine-grained modularization of the same ontology is increased extraction time. On the other hand, the extraction time for all 1,952 $\top\bot^*$ modules of Mereology is significantly larger than that for all 3,660 $\top\bot^*$ modules of Koala—although the same number of terms from each ontology went into seed signatures. This discrepancy has the following explanation. On average, the module signatures are smaller than for Koala, and therefore the difference between a minimal seed signature Σ of a module $\mathcal{M}$ and the extended signature $\Sigma \cup \widetilde{\mathcal{M}}$ is smaller. Therefore, more extensions of minimal seed signatures need to take place.

Attempts to fully modularize ontologies larger than Koala and Mereology with the described algorithm did not succeed. We cancelled each such computation after several hours, when thousands of modules have been extracted.

Reducing the overall number of modules. Although the total number of modules is far from the theoretical upper bound of 2^{25} for Koala and Mereology, it is still too large to inspect each module separately or expect ontology users to do so on a regular basis. For this reason, we have tried two more ways to reduce the overall numbers to fewer "interesting" ones.

Apart from distinguishing genuine from fake modules following the extraction, we have also experimented with a technique of unifying similar modules. It consisted in replacing a large enough number of modules that differ by a small enough number of axioms with the union and intersection of all these modules, where "large enough" and "small enough" are adjustable parameters. In order to obtain a noticeable decrease in module numbers for Koala, we had to choose parameter values so extreme that the unified modules could not reasonably be called similar anymore.

Another attempt at reducing module sizes was to vary the ways to obtain the first modules in Line **??** of Algorithm **??** (*start strategy*) and to extend the module list in Line **??** (*extension strategy*). One such strategy was to use the signatures of all axioms in $\mathcal{O}$ for start and extension instead of single terms. The underlying intuition is that the presence of some axiom in $\mathcal{O}$ indicates that its signature constitutes a topic that is relevant to the ontology. By thus restricting the number of seed signatures, we hoped to restrict the total number of modules to the more relevant ones. This turned out to have almost no effect on the number of modules extracted, but increase runtime significantly, partly because the lexicographic optimization to Line **??** of Algorithm **??** could not be used.

Module numbers for subset sampling. After carrying out the subset sampling technique described in Section **??**, we are strongly convinced that most of the ontologies examined exhibit the feared exponential behavior. Figure **??** shows scatterplots of the number of $\top\bot^*$ modules (genuine $\top\bot^*$ modules) versus the size of the subset for People and Koala. Each chart shows an exponential trendline, which is the least-squares fit through the data points by using the equation $m = ce^{bn}$, where n is the size of the subset, m is the number of modules, e is the base of the natural logarithm, and c, b are constants. This equation and the corresponding determination coefficient (R^2 value) are given beneath each chart. Spreadsheets with the underlying data, as well as spreadsheets and charts for the other ontologies, can be found at **?**. The R^2 values and trendline equations for the examined ontologies are summarized in Figure **??**, where we also included the estimated number of modules for the full ontology as per the equation, the timeout used and the overall runtime.

The scatterplots and determination coefficients for the first six ontologies in Figure **??** provide strong evidence that the number of modules depends exponentially on the size of the subset. In most cases, the

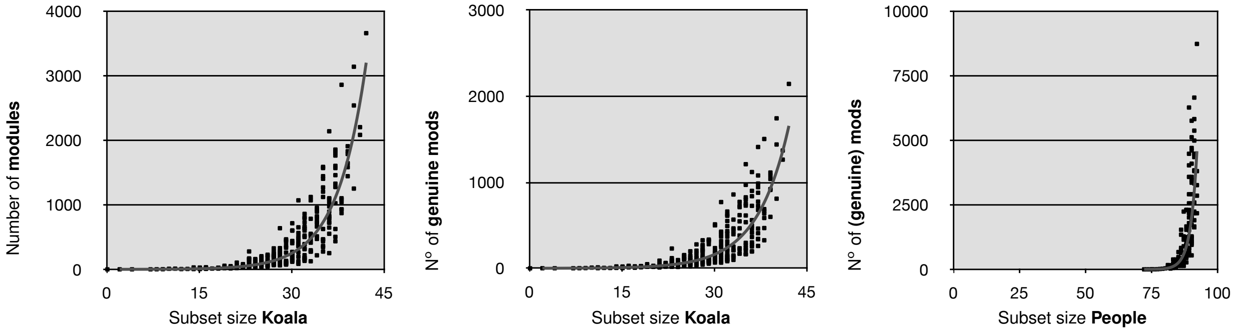

Figure 4. Numbers of modules versus subset sizes for Koala and People

	Confidence		Trendline equation		Estimate		Timeout	Runtime
Ontology	R^2_{m}	R^2_{g}	m	g	m	g	[min]	[min]
People	.95	.95	both $2 \cdot 10^{-13} e^{.41n}$		10^6	10^6	20	148
Mereology	.87	.94	$1.2e^{.16n}$	$1.1e^{.13n}$	10^3	10^2	—	4
Koala	.90	.88	$.45e^{.21n}$	$.50e^{.19n}$	10^3	10^3	—	4
Galen	.94	.86	$1.2e^{.24n}$	$1.6e^{.16n}$	NaN	NaN	60	288
University	.84	.83	$1.7e^{.19n}$	$1.6e^{.14n}$	10^4	10^3	20	354
OWL-S	.82	.84	$.0027e^{.17n}$	$.0032e^{.16n}$	10^{17}	10^{17}	60	73
Tambis	.75	.70	$1.1e^{.22n}$	$1.4e^{.13n}$	10^{58}	10^{33}	600	681
miniTambis	.47	.52	$2.6e^{.18n}$	$2.5e^{.14n}$	10^{14}	10^{10}	600	963

m, g	$\top\bot^*$ modules, genuine $\top\bot^*$ modules
R^2_{m}, R^2_{g}	Determination coefficient of fitted trendlines
Estimate	Module numbers for full ontology as per trendline
NaN	Estimate is larger than 10^{142}

Figure 5. Witnesses for exponential behavior

exponential behavior was observable with no timeout or a 20-minute timeout. For Galen, and OWL-S, we increased the timeout to 60 minutes.

For the remaining two ontologies, Tambis and miniTambis, even with a timeout of 600 minutes, we do not have a strong evidence of an exponential behavior. However, the most likely explanation is that we are timing out before the exponential break in the trendline. We observed this with several other ontologies which led us to double our original timeout from 10 to 20 minutes and then to treble that for Galen and OWL-S. We have longer timing out experiments planned for Tambis and miniTambis, but they will require considerably more time to run. In this context, it is interesting to note that the size of the subset whose modularization exceeded the timeout varied between 15 (miniTambis, 600 minutes) and 92 (People, 20 minutes).

The "Estimate" columns of Figure **??** show that we cannot always expect many fake modules, the most prominent example being People. Its two trendlines are almost identical, with the highest confidence value among the examined ontologies. In addition, the exponent in the equation is the largest. However, for miniTambis, there could be almost quadratically more fake than genuine modules.

Weight analysis for Koala. Even if we consider only genuine modules, there are ontologies that have exponentially many of them. In order to focus on even fewer, "interesting" modules, we have devised the measures *cohesion* and *pulling power*. Thy are based on the number of seed signatures (SSigs) of a module $\mathcal{M}$ and the number of terms in $\widetilde{\mathcal{M}}$. An SSig Σ of $\mathcal{M}$ is called *minimal* (MSSig) if there is no signature $\Sigma' \subset \Sigma$ that is an SSig of $\mathcal{M}$. If we ignore terms not present in the module, we speak of a *real MSSig* for $\mathcal{M}$: this is a signature $\Sigma' = \Sigma \cap \widetilde{\mathcal{M}}$ where Σ is an MSSig for $\mathcal{M}$. Let r, s, m be the number of real MSSigs for $\mathcal{M}$, the size of the smallest MSSig for $\mathcal{M}$, and the size of $\widetilde{\mathcal{M}}$.

The *cohesion* of $\mathcal{M}$ measures how strongly the terms in $\mathcal{M}$ are held together, as indicated by the number of seed signatures for $\mathcal{M}$. More precisely, the cohesion of $\mathcal{M}$ is defined to be the ratio r/s. The *pulling power* of $\mathcal{M}$ measures how many terms are needed in an MSSig to "pull" all terms into $\mathcal{M}$ that we find there. We define the *pulling power* of $\mathcal{M}$ to be the ratio m/s.

As a first draft, we define the *weight* of a module $\mathcal{M}$ to be the product of its cohesion and pulling power: $w = \frac{r \cdot m}{s^2}$. We computed the weight of all 3660 modules of Koala. The 11 heaviest modules and their set differences yield a partition of almost the whole ontology into 10 parts, each of which consists of terms that intuitively form a topic (subconcepts included): Animal; Person and isHardWorking; Student; Parent; Koala and Marsupial; TasmanianDevil; Quokka; Habitat; Degree; Gender. These topics reflect the core parts of the ontology. Those axioms that do not occur among the heaviest modules tend to be those that we intuitively would call less important for the ontology, for instance RainForest $\sqsubseteq$ Forest.The first 11 modules cover almost all of Koala's logical axioms (39 out of 42), and all axioms are covered from the 34th heaviest module on. The first 19 heaviest modules are also genuine.

The next step will be to refine this measure and apply it to more ontologies. Since we cannot expect to fully modularise ontologies bigger than Koala, we will need to find ways to extract heavy-weight modules separately.

5. Module number estimation via seed signature sampling

The results of the experiments strongly suggest that there is no hope for a robustly scalable algorithm that computes *all* modules of an ontology. However, if we are only interested in the *number* of all modules, it is possible that we can estimate this number. A straightforward approach would be as follows. For an ontology $\mathcal{O}$ with N terms, there can be at most 2^N many modules if we assume for the moment that the $\mathcal{O}$ has more axioms than terms. We can now randomly draw $n \ll N$ seed signatures and compute their modules. Since some of the modules are likely to coincide, we assume that the n drawn seed signatures yield $k < n$ modules. The task is now to find an estimate for the number K of all modules of $\mathcal{O}$ using N, n and k, and we also need to find out what values of n guarantee for a statistically reliable estimate.

The problem can be reformulated as follows: given a bag of 2^N marbles (seed signatures) from which we have randomly drawn n marbles which turned out to have k colors (modules), what is a reliable estimate for the number K of all colors in the bag? It is clear that we should be looking for a maximum likelihood estimator, i.e., a value K_0 for which the probability that n drawn marbles have k colors under the assumption $K = K_0$ is maximal. The problem with this criterion is that it very much depends on the distribution of marbles over the colors, i.e., whether the number of marbles of the same color differs among the colors or is roughly equal.

Therefore, in a first step, we took the number of marbles per color into account. Let the bag contain $N_1, \ldots, N_K$ marbles of color $1, \ldots, K$, where $N_1 + \cdots + N_K = N$. Suppose we draw $n_1, \ldots, n_k$ marbles of color $1, \ldots, k$, where $n_1 + \cdots + n_k = n$. The probability that the random drawing of n marbles has this outcome, is

$$P = \mathsf{V}_{n_1}^{N_1} \ldots \mathsf{V}_{n_k}^{N_k} \cdot \frac{1}{\mathsf{V}_n^N} \cdot \mathsf{C}_{n_1}^{n} \cdot \mathsf{C}_{n_2}^{n-n_1} \ldots \mathsf{C}_{n_k}^{n-n_1-\cdots-n_{k-1}},$$

where $\mathsf{C}_b^a = \binom{a}{b} = \frac{a!}{b!(a-b)!}$ is the number of k-combinations of n elements and $\mathsf{V}_b^a = \frac{a!}{(a-b)!}$ is the number of k-variations of n elements.

It can easily be seen that this value takes on a maximum when $n_i = N_i$ for $i = 1, \ldots, k$, regardless of the distribution of colors among the marbles that remain in the bag. It is therefore convenient to unify the drawn colors as well as the colors not drawn. The problem can then be simplified as follows: given a bag of 2^N marbles, each black or white, from which we have randomly drawn n marbles which turned out to be black, what is a reliable estimate for the number of black marbles in the bag?

According to the draw, there bag could have contained i black and $N-i$ white marbles, for $i = n, \dots, N$. For each i, let H_i the hypothesis "there were exactly i black marbles in the bag". The probability of drawing exactly n black marbles under H_i is the quotient of the number of draws of n black marbles out of the i black ones, divided by the number of draws of n marbles out of all N marbles, i.e., $P_i = \frac{\binom{i}{n}}{\binom{N}{n}}$. It is now easy to see that $P_N = 1$ and $P_N > P_{N-1} > \cdots > P_n$. Therefore, the maximum likelihood estimator for the number of black marbles is N, which corresponds to estimating K to be equal to k. In order to minimize the error when accepting H_N and rejecting $H_{N-1}, \dots, H_n$, we have to make sure that all corresponding P_i are below a certain threshold value t, which is usually taken to be 0.05. Due to the observed monotony, it suffices to ensure $P_{N-1} < 0.05$, i.e., $\frac{\binom{N-1}{n}}{\binom{N}{n}} = \frac{N-n}{N} < t$, therefore, $n > 0.95N$. This means that, in order to achieve that the estimate for the number of colors has the usual confidence, we would have to draw 95% of all seed signatures, which is not a significant saving compared to drawing them all.

In order to avoid the problem that the intended high confidence requires us to draw too many samples, we can extend the null hypothesis to "the marbles in the bag had between k and $k+d$ colors", for an adjustable parameter d denoting the interval size or tolerance. In the black-white view, this would mean "the bag contained between n and $N-d$ black marbles". The error we would make in accepting this new hypothesis and rejecting the remaining $H_{N-d+1}, \dots, H_N$, would be at most P_{N-d+1}. This means that we have to ensure $P_{N-d+1} < t$, i.e., $\frac{\binom{N-d+1}{n}}{\binom{N}{n}} < t$. Now this equation is difficult to solve for N without making d explicit. However, if we insert realistic values for d and N and try different values for n via binary search, we can find the smallest value of n (sample size) for which the inequation is satisfied. For $N = 2^{25} = 33,554,432$, which is the number of all seed signatures of Koala (marbles in the bag) and a tolerance of $d = 100$, which is almost 3% of what we happen to know to be the number of all modules, the minimal sample size is $n = 1,006,633$.

This last figure means that we would have to draw about 3% of all seed signatures in order to get a confident estimate of the number of all modules in the form of an interval of size 100. Now it might seem that having to extract only 3% of all modules is a significant improvement. But there are at least two counter-arguments. First, for a smaller number of randomly drawn seed signatures, the optimizations performed to Algorithm **??** based on Proposition **??** will be far less effective than for the complete power set of $\widetilde{\mathcal{O}}$ where the signatures can be traversed ordered by size. In the latter case, many more module extractions can be saved by checking containment of one signature in another. Second, if it should turn out that acceptable tolerances d are achieved for the ratio $\frac{n}{N} = 3\%$ *independently* of the original ontology's size, then we would still have to extract 3% of an exponential number of modules. This would mean that this new approach might be able to handle ontologies slightly bigger than Koala, but it would still not be scalable. Although we plan to verify this last conjecture experimentally, we are convinced that we cannot expect to be able to estimate the number of modules using any of the discussed approaches to seed signature sampling.

6. Discussion and outlook

The fundamental conclusion is clear: the number of modules, even when we restrict our attention to genuine modules, is exponential in the size of the ontology for real ontologies. Our most reasonable estimates of the total number of modules in small to midsize ontologies (i.e., anything over 100 axioms) show that full modularization is practically impossible. As we are computing locality based modules, which tend to be larger than conservative extension based modules, our results give us a lower bound on the number of modules.

It is, of course, possible that there are principled ways to reduce the target number of modules. We could use a coarser approximation, though that would be hard to justify on logical grounds. Attempts to use "less minimal" modules or to heuristically merge modules have exhibited bad behavior, with a strong tendency to collapse to very few modules that comprise most of the ontology.

We believe that this conclusion is robust, even with the failure of our experiments on Tambis and miniTambis to uncover exponential behavior. As we said in Section 4, our expectation is that a longer timeout will reveal the problematic behavior.

Furthermore, we observe that these ontologies have a large number of unsatisfiable concepts, with large justifications for those, and comparatively long axioms with large signatures. Since each module for such a concept contains at least one justification[6], modules for these ontologies tend to be large, which decreases the overall number of modules. Similarly, large axioms with large signatures tend to raise the chances of interaction between terms as well as increasing the signature size of modules which, in turn, make for large numbers of non-minimal seed signatures. However, these facts do not indicate a difference in kind between Tambis and miniTambis and other ontologies we examined, such as University, or even Koala. Both Koala and University have unsatisfiable concepts. The justifications for the unsatisfiable concepts in Koala have a max size of 5 axioms, whereas University tops out at 9, with most being below 6. miniTambis and Tambis's justifications have a max size of 13 axioms, with a large percentage over 6. If our hypothesis about the role of the justifications is correct, then it seems likely that the exponential break is merely delayed. Thus it is still possible, and we believe probable, that an exponential behavior is present but is only visible with a sufficiently higher timeout. Furthermore, the large size of the justifications in Tambis and miniTambis is a bit artificial as it is dependent on the unsatisfiability. These ontologies have large chains of "dependent" unsatisfiabilities **?** which increase the size of justifications along the chain. When the unsatisfiabilities are resolved, those concepts will no longer have those particularly lengthy justifications as part of all of their modules.

These considerations suggest that, in general, the ratio between genuine and fake modules can be seen as a measure of axiomatic richness, at least indicating how strongly the axioms in the ontology connect its terms: the fewer of its modules are fake, the more "mutually touching" its terms are.

While the outcome of the experiments is discouraging from the point of view of using the complete modularization in order to analyze the ontology, it does suggest several interesting lines of future work. First, we have already seen several features of ontologies that correlate well with a large or small number of modules. However, except for the phenomenon seen in Mereology, we do not have a verified explanation. Thus, for example, we need to get a precise picture of the relationship between justificatory and modular structure. Second, even if we cannot compute all modules, we may be able to compute a better approximation of their number. Given that signature sampling did not seem to help, we intend to explore sources of module number increase or reduction, such as the shape of the inferred concept hierarchy and patterns of axioms. Methodologically, it seems that artificial ontologies should be used, e.g., for confirmation of the relationship between justificatory structure and module number. Third, our preliminary experiments aimed at computing *heavy weight* ontologies are promising: our weights seem to capture nicely the cohesion and pulling power of a module, and the resulting heavy modules seem to correlate nicely with topics. We are currently investigating whether it is possible to compute all heavy modules without computing all modules, and also looking into a suitable notion of *building blocks* of modules. The latter concept is closely related to fake and genuine modules, which we are also investigating in more detail.

[6] We have strong reason to believe that a locality-based module, due to being depleting **?**, always contains *all* justifications for each entailment within its extended signature.