

# Axiom Dependency Hypergraphs for Fast Modularisation and Atomic Decomposition

Francisco Martín-Recuerda<sup>1</sup> and Dirk Walther<sup>2</sup>

<sup>1</sup> Universidad Politécnica de Madrid, Spain  
fmartinrecuerda@fi.upm.es

<sup>2</sup> TU Dresden, Theoretical Computer Science  
Center for Advancing Electronics Dresden, Germany  
Dirk.Walther@tu-dresden.de

**Abstract.** In this paper we use directed hypergraphs to represent the locality-based dependencies between the axioms of an OWL ontology. We define a notion of an axiom dependency hypergraph, where axioms are represented as nodes and dependencies between axioms as hyperedges connecting possibly several nodes with one node. We show that a locality-based module of an ontology corresponds to a connected component in the hypergraph, and an atom of an ontology to a strongly connected component. Collapsing the strongly connected components into single nodes yields a condensed axiom dependency hypergraph, which contains the atomic decomposition of the ontology. To condense the axiom dependency hypergraph we exploit linear time graph algorithms on its graph fragment. This optimization can significantly reduce the time needed to compute the atomic decomposition of an ontology. We provide an experimental evaluation for computing the atomic decomposition of large biomedical ontologies, and for computing syntactic locality-based modules using the condensed axiom dependency hypergraph.

## 1 Introduction

An *axiom dependency hypergraph (ADH)* for OWL ontologies is a directed hypergraph which explicitly represents the locality-based dependencies between axioms. This graph can be used to compute the atomic decomposition (AD) of an ontology [5]. Different to other hypergraph representations of ontologies [12, 10], the nodes represent axioms that are connected by hyperedges which explicitly describe the locality-based dependencies between axioms. These hyperedges mimic the order in which axiom are included in a locality-based module by the module extraction algorithm [7]. Due to this particular hypergraph representation the correspondence between atoms of an ontology and the strongly connected components (SCCs) of its hypergraph becomes apparent. This allows us to employ standard algorithms from graph theory to compute atoms. Similarly, modules can be characterized as connected components in an ADH. The

---

The second author is funded by the German Research Foundation (DFG) via the Cluster of Excellence ‘Center for Advancing Electronics Dresden’.

notion of an ADH provides a new perspective on previous work on locality and atomic decomposition from the point of view of graph theory.

Computing all SCCs in a directed hypergraph is an inherently quadratic process [1], whereas in a directed graph, it can be done in linear time wrt. the size of the graph [11, 13]. For several well-known biomedical ontologies from the NCBO Bioportal,<sup>1</sup> many (if not all) of the locality-based dependencies between axioms can be represented by simple edges [9]. In the case that all dependencies between axioms of an ontology can be represented using simple edges only, the ADH is a directed graph, and its SCCs can be computed very efficiently. For ontologies containing axioms that exhibit dependencies which can only be represented using complex hyperedges, we can still benefit from the use of linear-time algorithm for computing SCCs in the graph fragment of the ADH.

We have implemented a Java prototype for computing the atomic decomposition and syntactic locality-based modules. We confirm a significant improvement in running time for prominent biomedical ontologies from the NCBO Bioportal compared against state-of-the-art implementations [14, 15].

The paper is organised as follows. In Section 2, we introduce the notion of minimal non-locality signature of an axiom which can be used to check whether an axiom is non-local wrt. a signature. In Section 3, we introduce the notion of an axiom dependency hypergraph, and we show how to characterize locality-based modules as well as atoms together with their dependencies. We demonstrate the applicability of the hypergraph-based approach with an evaluation of our Java prototype in Section 4. The paper is concluded in a final section.

## 2 Non-locality of Axioms

In this paper, we consider ontologies formulated in the expressive description logic  $SR\mathcal{OIQ}$  [8] which underlies the Web Ontology Language OWL 2.<sup>2</sup> For the evaluation, we consider prominent biomedical ontologies that are formulated in the light-weight description logic  $\mathcal{EL}^{++}$  [2] which is at the core of the OWL 2 EL profile.<sup>3</sup> For a detailed introduction to description logics, we refer to [3].

A module  $\mathcal{M}$  of an ontology  $\mathcal{O}$  wrt. a signature  $\Sigma$  is a subset of  $\mathcal{O}$  that preserves all entailments formulated using symbols from  $\Sigma$  only. More formally,  $\mathcal{M} \subseteq \mathcal{O}$  is a *module* of  $\mathcal{O}$  wrt.  $\Sigma$  if for all axioms  $\alpha$  with  $\text{sig}(\alpha) \subseteq \Sigma$ :  $\mathcal{M} \models \alpha$  iff  $\mathcal{O} \models \alpha$ , where  $\text{sig}(\alpha)$  denotes the signature of  $\alpha$ . The notion of *syntactic locality* has been suggested as a practical approach to computing approximations of minimal modules in polynomial time [7]. Syntactic locality for a signature  $\Sigma$  defines a set of axioms that are syntactically local wrt.  $\Sigma$ . We consider the syntactic locality-based notions  $\perp$ -locality and  $\top$ -locality, and we do not consider any semantic locality notion. An axiom  $\alpha$  is local wrt.  $\Sigma$  if  $\alpha$  is equivalent to a tautology after all symbols in  $\alpha$  that are not in  $\Sigma$  have been replaced by either  $\top$  or  $\perp$ , respectively. Intuitively,  $\alpha$  is local wrt.  $\Sigma$  if it does not state anything

<sup>1</sup> <http://bioportal.bioontology.org/>

<sup>2</sup> <http://www.w3.org/TR/owl2-overview/>

<sup>3</sup> [http://www.w3.org/TR/owl2-profiles/#OWL\\_2\\_EL](http://www.w3.org/TR/owl2-profiles/#OWL_2_EL)

about the symbols in  $\Sigma$ . An ontology can *safely* be extended with  $\alpha$  or it can *safely* import  $\alpha$ , where ‘safe’ means not changing the meaning of terms in  $\Sigma$ . We denote with  $\text{Mod}_{\mathcal{O}}^x(\Sigma)$ , for  $x \in \{\perp, \top\}$ , the  $x$ -local module of ontology  $\mathcal{O}$  wrt. the signature  $\Sigma$  consisting of axioms that are not  $x$ -local wrt.  $\Sigma$ .

We can check for the state of syntactic non-locality of an axiom in terms of signature containment. To this end, we introduce the notion of minimal non-locality signature for *SROLQ* axioms.

**Definition 1.** *Let  $\alpha$  be an axiom, and let  $x \in \{\perp, \top\}$  denote a locality notion. A Minimal non- $x$ -Locality Signature for an axiom  $\alpha$  is a signature  $\Sigma \subseteq \text{sig}(\alpha)$  such that  $\alpha$  is not  $x$ -local wrt.  $\Sigma$ , and  $\Sigma$  is minimal (wrt. set inclusion) with this property. The set of minimal non- $x$ -locality signatures of  $\alpha$  is denoted by  $\text{MLS}^x(\alpha)$ .  $\dashv$*

The notion of minimal non-locality signature turns out to be equivalent to the notion of minimal globalising signatures, which were introduced specifically for computing modules from an atomic decomposition [4].

The following example shows that there can be exponentially many minimal non-locality signatures for axioms formulated in a simple concept description language using merely conjunction and disjunction as operators.

*Example 1.* Let  $\alpha = (X_{11} \sqcup X_{12} \sqcup \dots \sqcup X_{1m}) \sqcap \dots \sqcap (X_{n1} \sqcup X_{n2} \sqcup \dots \sqcup X_{nm}) \sqsubseteq Y$  be an axiom. The minimal non- $\perp$ -locality signature  $\text{MLS}^{\perp}(\alpha)$  of  $\alpha$  is as follows:

$$\text{MLS}^{\perp}(\alpha) = \{\{X_{1i_1}, X_{2i_2}, \dots, X_{ni_n}\} \mid i_1, i_2, \dots, i_n \in \{1, \dots, m\}\}$$

Then:  $|\text{MLS}^{\perp}(\alpha)| = m^n$ .  $\triangleleft$

However, exponentially many minimal non-locality signatures can be avoided if the axiom is normalised. An ontology  $\mathcal{O}$  is normalised by applying the normalisation rules presented in [10], which are an extension of the normalisation for  $\mathcal{EL}$  in [12]. Axioms of a normalised ontology have one of the following forms, where  $A_i \in \text{N}_{\mathcal{C}} \cup \{\top\}$ ,  $B_i \in \text{N}_{\mathcal{C}} \cup \{\perp\}$ ,  $R_i \in \text{N}_{\mathcal{R}} \cup \text{inv}(\text{N}_{\mathcal{R}})$ ,  $X, Y \in \{\exists R.B, (\geq n R.B), \exists R.Self \mid B \in \text{N}_{\mathcal{C}}, R \in \text{N}_{\mathcal{R}} \cup \text{inv}(\text{N}_{\mathcal{R}}), n \geq 0\}$  and  $\ell, m \geq 0$ :

$$\begin{array}{ll} \alpha_1 : A_1 \sqcap \dots \sqcap A_{\ell} \sqsubseteq B_1 \sqcup \dots \sqcup B_m & \alpha_5 : X \sqsubseteq Y \\ \alpha_2 : X \sqsubseteq B_1 \sqcup \dots \sqcup B_m & \alpha_6 : R_1 \sqsubseteq R_2 \\ \alpha_3 : A_1 \sqcap \dots \sqcap A_{\ell} \sqsubseteq Y & \alpha_7 : Dis(R_1, R_2) \\ \alpha_4 : R_1 \circ \dots \circ R_{\ell} \sqsubseteq R_{\ell+1} & \end{array}$$

where  $\text{N}_{\mathcal{C}}$ ,  $\text{N}_{\mathcal{R}}$  are mutually disjoint sets of concept names and role names,  $\text{inv}(\text{N}_{\mathcal{R}})$  is the set of inverse roles  $r^-$ , for  $r \in \text{N}_{\mathcal{R}}$ , and  $\exists R.Self$  expresses the local reflexivity of  $R$ . The normalisation of  $\mathcal{O}$  runs in linear time in the size of  $\mathcal{O}$ . The normalised ontology is a conservative extension of  $\mathcal{O}$  wrt.  $\text{sig}(\mathcal{O})$  [10].<sup>4</sup>

The following proposition can readily be seen.

<sup>4</sup> The normalisation in [10] can straightforwardly be extended to *SROLQ*-ontologies. Then a normalised axiom can be of the forms as described, where  $A_i$  and  $B_i$  additionally range over nominals. However, nominals are not contained in any minimal non-locality signature of a normalised axiom.

**Proposition 1.** *Let  $\alpha$  be a normalised axiom. Then:  $|MLS^\perp(\alpha)| = 1$  and  $|MLS^\top(\alpha)| \leq 2$ . ⊣*

We can apply additional normalisation rules to reduce the number of symbols on the left- and right-hand side of normalised axioms. Bounding the number of symbols in an axiom results in bounding the size of the minimal non-locality signatures of the axiom.

We now give simple conditions under which normalised axioms are not syntactic local. Similar non-locality conditions are presented in the notions of  $\perp$ - and  $\top$ -reachability in [10].

**Proposition 2 (Non-locality via Signature Containment).** *Let  $\alpha$  be a normalised axiom. Let  $\Sigma$  be a signature. Then:  $\alpha$  is not  $\perp$ -local wrt.  $\Sigma$  iff one of the following holds:*

- $\text{sig}(LHS(\alpha)) \subseteq \Sigma$  if  $\alpha$  is of the form  $\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5, \alpha_6$ ;
- $\text{sig}(\alpha) \subseteq \Sigma$  if  $\alpha$  is of the form  $\alpha_7$ ;

*Then:  $\alpha$  is not  $\top$ -local wrt.  $\Sigma$  iff  $\alpha$  is of the form  $\alpha_7$  or one of the following holds:*

- $\text{sig}(RHS(\alpha)) \cap \Sigma \neq \emptyset$  if  $\alpha$  is of the form  $\alpha_3, \alpha_4, \alpha_5, \alpha_6$ ;
- $\text{sig}(RHS(\alpha)) \subseteq \Sigma$  if  $\alpha$  is of the form  $\alpha_1, \alpha_2$ . ⊣

### 3 Axiom Dependency Hypergraph

A *directed hypergraph* [6] is a tuple  $\mathcal{H} = (\mathcal{V}, \mathcal{E})$ , where  $\mathcal{V}$  is a non-empty set of *nodes* (vertices), and  $\mathcal{E}$  is a set of *hyperedges* (*hyperarcs*). A *hyperedge*  $e$  is a pair  $(T(e), H(e))$ , where  $T(e)$  and  $H(e)$  are non-empty disjoint subsets of  $\mathcal{V}$ .  $H(e)$  ( $T(e)$ ) is known as the *head* (*tail*) and represents a set of nodes where the hyperedge ends (starts). A *B-hyperedge* is a directed hyperedge with only one node in the head. Directed Hypergraphs with only *B-hyperedges* are called *Directed B-hypergraphs*. This is the only type of hypergraphs that we consider in this paper.

A node  $v$  is *B-connected* (or forward reachable) from a set of nodes  $V'$  (written  $\mathcal{V}' \geq_B v$ ) if (i)  $v \in V'$ , or (ii) there is a *B-hyperedge*  $e$  such that  $v \in H(e)$  and all tail nodes in  $T(e)$  are *B-connected* from  $V'$ . For a set of nodes  $\mathcal{V}' \subseteq \mathcal{V}$ , we denote with  $\geq_B(\mathcal{V}')$  the set  $\geq_B(\mathcal{V}') = \{v \in \mathcal{V} \mid \mathcal{V}' \geq_B v\}$  of *B-connected* nodes from  $\mathcal{V}'$ .

In a directed hypergraph  $\mathcal{H}$ , two nodes  $v_1$  and  $v_2$  are *strongly B-connected* if  $v_2$  is *B-connected* to  $v_1$  and *vice versa*. In other words, both nodes,  $v_1$  and  $v_2$ , are *mutually reachable*. A *strongly B-connected component (SCC)* is the set of all nodes from  $\mathcal{H}$  which are all mutually reachable [1]. We allow an SCC to be a singleton set as the reachability relation is reflexive, i.e., any axiom is mutually reachable from itself.

Directed *B-hypergraphs* can be used to explicitly represent the locality-based dependencies between axioms. *Axiom dependency hypergraphs* for ontologies wrt. the locality-based modularity notions are defined as follows. We use  $\text{sig}(S)$  to denote the signature of a set  $S$  of axioms.

**Definition 2 (Axiom Dependency Hypergraph).** Let  $\mathcal{O}$  be an ontology. Let  $x \in \{\perp, \top\}$  denote a locality notion. The Axiom Dependency Hypergraph  $\mathcal{H}_{\mathcal{O}}^x$  for  $\mathcal{O}$  wrt.  $x$ -locality ( $x$ -ADH) is defined as the directed  $B$ -hypergraph  $\mathcal{H}_{\mathcal{O}}^x = (\mathcal{V}^x, \mathcal{E}^x)$ , where

- $\mathcal{V}^x = \mathcal{O}$ ; and
- $e = (T(e), H(e)) \in \mathcal{E}^x$  iff  $T(e) \subseteq \mathcal{V}^x$  and  $H(e) = \{\beta\}$ , for some  $\beta \in \mathcal{V}^x$ , such that:
  - (i)  $\beta \notin T(e)$ , and
  - (ii)  $\beta$  is not  $x$ -local wrt.  $\text{sig}(T(e))$ . –

The nodes of the axiom dependency hypergraph are the axioms in the ontology. Hyperedges are directed and they connect possibly many tail nodes with one head node. Note that a head node of a hyperedge is not allowed to occur in its tail. Intuitively, the tail nodes of an hyperedge  $e$  correspond to axioms that provide the signature symbols wrt. which the axiom represented by the head node of  $e$  is non-local. We can think of walking along hyperedges to access  $B$ -connected nodes as part of the process of how the module extraction algorithm computes a module by successively including axioms into the module.

The notion of ADH for ontologies depends on the notion of syntactic locality. Using Prop. 2, we can equivalently define this notion using minimal non-locality signatures by replacing Item (ii) of Def. 2 with:

- (iib)  $\Sigma \subseteq \text{sig}(T(e))$ , for some  $\Sigma \in \text{MLS}^x(\beta)$ .

An ADH  $\mathcal{H}_{\mathcal{O}}$  contains all locality-based dependencies between different axioms of the ontology  $\mathcal{O}$ . These dependencies are represented by the hyperedges in  $\mathcal{H}_{\mathcal{O}}$ . Note that  $\mathcal{H}_{\mathcal{O}}$  contains exponentially many hyperedges, many of which can be considered redundant in the following sense.

**Definition 3.** A hyperedge  $e$  in a directed  $B$ -hypergraph  $\mathcal{H}$  is called redundant if there is a hyperedge  $e'$  in  $\mathcal{H}$  such that  $H(e) = H(e')$  and  $T(e') \subsetneq T(e)$ . –

A compact version of a directed  $B$ -hypergraph  $\mathcal{H}$  is obtained from  $\mathcal{H}$  by removing all redundant hyperedges while the  $B$ -connectivity relation between axioms is preserved. In the remainder of the paper, we consider ADHs that are compact. Notice that compact ADHs are unique and they may still contain exponentially many hyperedges. The number of hyperedges can be reduced to polynomially many by first normalising the ontology as discussed in Section 2. In the following, we characterise modules and atoms together with their dependencies in terms of ADHs for which  $B$ -reachability is crucial.

### 3.1 Locality-based modules in an ADH

$B$ -connectivity in an ADH can be used to specify locality-based modules in the corresponding ontology. A locality-based module of an ontology  $\mathcal{O}$  for the signature of an axiom  $\alpha$  (or a subset of axioms  $\mathcal{O}' \subseteq \mathcal{O}$ ) corresponds to the  $B$ -connected component in the ADH for  $\mathcal{O}$  from  $\alpha$  (or  $\mathcal{O}'$ ) [9].

**Proposition 3.** Let  $\mathcal{O}$  be an ontology,  $\mathcal{O}' \subseteq \mathcal{O}$  and  $\Sigma = \text{sig}(\mathcal{O}')$ . Let  $\geq_B$  be the  $B$ -connectivity relation of the  $x$ -ADH for  $\mathcal{O}$ , where  $x \in \{\perp, \top\}$ . Then:  $\text{Mod}_{\mathcal{O}}^x(\Sigma) = \geq_B(\mathcal{O}')$ .  $\dashv$

However, ADHs do not contain sufficient information for computing a module for an arbitrary signature as the following simple example shows.

*Example 2.* Let  $\mathcal{O} = \{\alpha_1 = A \sqsubseteq C, \alpha_2 = C \sqcap B \sqsubseteq D, \alpha_3 = D \sqsubseteq A\}$  and  $\Sigma = \{A, B\}$ . We have that  $\text{Mod}_{\mathcal{O}}^{\perp}(\Sigma) = \{\alpha_1, \alpha_2, \alpha_3\}$ . The  $\perp$ -ADH for  $\mathcal{O}$  contains no hyperedge  $e$  with  $H(e) = \{\alpha_2\}$  and, consequently,  $\alpha_2$  cannot be reached via a hyperedge.  $\triangleleft$

The problem can be solved by incorporating the signature  $\Sigma$  into the ADH. The  $\Sigma$ -extension  $\mathcal{H}_{\mathcal{O}, \Sigma}^x$  of an  $x$ -ADH  $\mathcal{H}_{\mathcal{O}}^x$  for an ontology  $\mathcal{O}$  wrt.  $x$ -locality,  $x \in \{\perp, \top\}$ , is defined as the ADH according Def. 2 but with Item (ii) replaced with:

(iii)  $\beta$  is not  $x$ -local wrt.  $\Sigma \cup \text{sig}(T(e))$ .

Intuitively, no symbol in  $\Sigma$  contributes to the dependencies between axioms. Consequently, less axioms in the tail are needed to provide the signature for non-locality of  $\beta$ . Note that hyperedges that are non-redundant in the original ADH may become redundant in the  $\Sigma$ -extended ADH. The remaining hyperedges represent the dependencies between axioms *modulo*  $\Sigma$ .

*Example 3.* Let  $\mathcal{O}$  and  $\Sigma$  as in Ex. 2. The  $\Sigma$ -extension of  $\perp$ -ADH for  $\mathcal{O}$  contains the edge  $e = \{\{\alpha_1\}, \{\alpha_2\}\}$ . Hence,  $\alpha_2$  can be reached via the hyperedge  $e$ . Axiom  $\alpha_1$  is the only axiom that is not- $\perp$  local wrt.  $\Sigma$ . The  $B$ -connected nodes from  $\alpha_1$  are the axioms in  $\text{Mod}_{\mathcal{O}}^{\perp}(\Sigma)$ .  $\triangleleft$

Given the  $\Sigma$ -extension of an ADH for an ontology,  $B$ -connectivity can be used to determine the axioms that are not local wrt. to  $\Sigma$  and to compute the corresponding locality-based module.

**Proposition 4.** Let  $\mathcal{O}$  be an ontology,  $\Sigma$  a signature and  $x \in \{\perp, \top\}$ . Let  $\mathcal{O}_{\Sigma}^x$  be the set of axioms from  $\mathcal{O}$  that are not  $x$ -local wrt.  $\Sigma$ . Let  $\geq_B$  be the  $B$ -connectivity relation of the  $\Sigma$ -extension of the  $x$ -ADH for  $\mathcal{O}$ . Then:  $\text{Mod}_{\mathcal{O}}^x(\Sigma) = \geq_B(\mathcal{O}_{\Sigma}^x)$ .  $\dashv$

### 3.2 ADH Atomic Decomposition

Atoms represent sets of highly related axioms in the sense that they always co-occur in modules [5]. We denote with  $\text{Atoms}_{\mathcal{O}}^x$  the set of all atoms of  $\mathcal{O}$  wrt.  $x$ -locality modules, for  $x \in \{\perp, \top\}$ . The atoms of an ontology partition the set of its axioms (i.e., every axiom occurs in exactly one atom). A dependency relation between atoms can be established: an atom  $\mathbf{a}_2$  *depends on* an atom  $\mathbf{a}_1$  in an ontology  $\mathcal{O}$  (written  $\mathbf{a}_1 \succ_{\mathcal{O}} \mathbf{a}_2$ ) if  $\mathbf{a}_2$  occurs in every module of  $\mathcal{O}$  containing  $\mathbf{a}_1$ . The binary relation  $\succ_{\mathcal{O}}$  is a partial order. For a given ontology, the poset  $\langle \text{Atoms}_{\mathcal{O}}^x, \succ_{\mathcal{O}} \rangle$  was introduced as *Atomic Decomposition (AD)* and it represents the modular structure of the ontology [5].

In the previous section, we have established that locality-based modules of an ontology  $\mathcal{O}$  correspond to sets of  $B$ -connected nodes in the axiom dependency hypergraph for  $\mathcal{O}$ . An atom of  $\mathcal{O}$  consists of axioms  $\alpha$  that share the same modules wrt. the signature of  $\alpha$ . It holds that for every  $x$ -local atom  $\mathbf{a} \subseteq \mathcal{O}$  with  $x \in \{\perp, \top\}$ :  $\alpha, \beta \in \mathbf{a}$  if, and only if,  $\text{Mod}_{\mathcal{O}}^x(\text{sig}(\alpha)) = \text{Mod}_{\mathcal{O}}^x(\text{sig}(\beta))$  [5]. Together with Proposition 3, we can now characterise the notion of an atom with a corresponding notion in axiom dependency hypergraphs. We have that two nodes in an ADH represent axioms that are contained in the same atom if, and only if, the nodes agree on the set of nodes that are  $B$ -connected from them. Formally:  $\alpha, \beta \in \mathbf{a}$  if, and only if,  $\geq_B(\alpha) = \geq_B(\beta)$ , where  $\geq_B$  be the  $B$ -connectivity relation of the ADH  $\mathcal{H}_{\mathcal{O}}$  for  $\mathcal{O}$ . It follows that all axioms of an atom are mutually  $B$ -connected in  $\mathcal{H}_{\mathcal{O}}$ . Axioms that are mutually  $B$ -connected constitute strongly  $B$ -connected components of  $\mathcal{H}_{\mathcal{O}}$ . Consequently, the set of atoms for an ontology  $\mathcal{O}$  corresponds to the set of strongly  $B$ -connected components in the axiom dependency hypergraph for  $\mathcal{O}$ . Let  $\text{SCCs}(\mathcal{H}_{\mathcal{O}}^x)$  be the set of strongly connected components of the hypergraph  $\mathcal{H}_{\mathcal{O}}^x$ , where  $x \in \{\perp, \top\}$ .

**Proposition 5.** *Let  $\mathcal{O}$  be an ontology and let  $x \in \{\perp, \top\}$  denote a locality notion. Then:  $\text{Atoms}_{\mathcal{O}}^x = \text{SCCs}(\mathcal{H}_{\mathcal{O}}^x)$ .  $\dashv$*

The condensed ADH is formed by collapsing the strongly  $B$ -connected components into single nodes and turning hyperedges between axioms into hyperedges between sets of axioms. The condensed ADH corresponds to the quotient hypergraph  $\mathcal{H}_{\mathcal{O}}/\simeq_B$  of  $\mathcal{H}_{\mathcal{O}}$  under the mutual  $B$ -connectivity relation  $\simeq_B$  in  $\mathcal{H}_{\mathcal{O}}$ . The  $\simeq_B$ -equivalence classes are the strongly  $B$ -connected components of  $\mathcal{H}_{\mathcal{O}}$ . The partition of a hypergraph under an equivalence relation is defined as follows.

**Definition 4 (Quotient Hypergraph).** *Let  $\mathcal{H} = (\mathcal{V}, \mathcal{E})$  be a hypergraph. Let  $\simeq$  be an equivalence relation over  $\mathcal{V}$ . The quotient of  $\mathcal{H}$  under  $\simeq$ , written  $\mathcal{H}/\simeq$ , is the graph  $\mathcal{H}/\simeq = (\mathcal{V}/\simeq, \mathcal{E}/\simeq)$ , where*

- $\mathcal{V}/\simeq = \{[x]_{\simeq} \mid x \in \mathcal{V}\}$ ; and
- $e = (T(e), H(e)) \in \mathcal{E}/\simeq$  iff there is an  $e' \in \mathcal{E}$  such that  $T(e) = \{[x]_{\simeq} \mid x \in T(e')\}$ ,  $H(e) = \{[x]_{\simeq} \mid x \in H(e')\}$  and  $T(e) \cap H(e) = \emptyset$ .  $\dashv$

We can now define the notion of a condensed ADH as the partition of the ADH under the mutual  $B$ -reachability relation.

**Definition 5 (Condensed Axiom Dependency Hypergraph).** *Let  $\mathcal{H}_{\mathcal{O}}^x = (\mathcal{V}^x, \mathcal{E}^x)$  be the  $x$ -ADH for an ontology  $\mathcal{O}$ , where  $x \in \{\perp, \top\}$ . Let  $\simeq_B$  be the mutual  $B$ -connectivity relation in  $\mathcal{H}_{\mathcal{O}}^x$ . The condensed axiom dependency hypergraph for  $\mathcal{O}$  wrt.  $x$ -locality ( $x$ -cADH) is defined as the quotient  $\mathcal{H}_{\mathcal{O}}^x/\simeq_B$  of  $\mathcal{H}_{\mathcal{O}}^x$  under  $\simeq_B$ .  $\dashv$*

The dependency relation  $\succ_{\mathcal{O}}^x$  between  $x$ -local atoms of  $\mathcal{O}$ , for  $x \in \{\perp, \top\}$ , is defined as follows [5]. For atoms  $\mathbf{a}, \mathbf{b} \in \text{Atoms}_{\mathcal{O}}^x$  and axioms  $\alpha \in \mathbf{a}$  and  $\beta \in \mathbf{b}$ :  $\mathbf{a} \succ_{\mathcal{O}}^x \mathbf{b}$  if, and only if,  $\mathbf{b} \subseteq \text{Mod}_{\mathcal{O}}^x(\alpha)$  if, and only if,  $\text{Mod}_{\mathcal{O}}^x(\beta) \subseteq \text{Mod}_{\mathcal{O}}^x(\alpha)$ .

**Proposition 6.** Let  $\mathcal{O}$  be an ontology with  $\alpha, \beta \in \mathcal{O}$ . Let  $\mathbf{a}, \mathbf{b} \in \text{Atoms}_{\mathcal{O}}^x$  such that  $\alpha \in \mathbf{a}$  and  $\beta \in \mathbf{b}$ , where  $x \in \{\perp, \top\}$ . Let  $\simeq$  be the mutual  $B$ -connectivity relation in the  $x$ -ADH for  $\mathcal{O}$  and  $\geq$  the  $B$ -connectivity relation in the  $x$ -cADH for  $\mathcal{O}$ . Then:  $\mathbf{a} \succ_{\mathcal{O}}^x \mathbf{b}$  iff  $[\alpha]_{\simeq} \geq [\beta]_{\simeq}$ .  $\dashv$

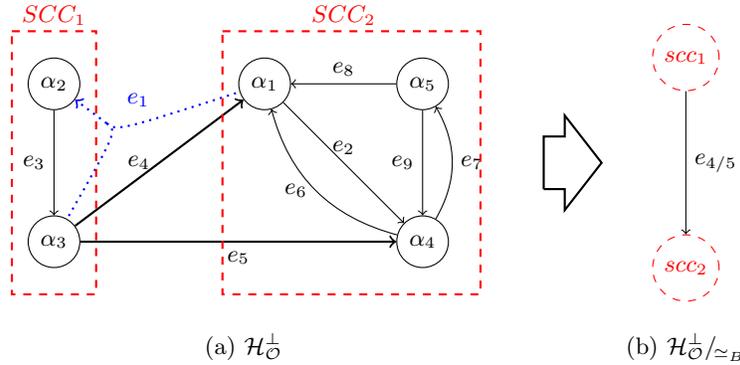
*Example 4.* Let  $\mathcal{O} = \{\alpha_1, \dots, \alpha_5\}$ , where  $\alpha_1 = A \sqsubseteq B$ ,  $\alpha_2 = B \cap C \cap D \sqsubseteq E$ ,  $\alpha_3 = E \sqsubseteq A \cap C \cap D$ ,  $\alpha_4 = A \sqsubseteq X$ ,  $\alpha_5 = X \sqsubseteq A$ . The  $\perp$ -ADH  $\mathcal{H}_{\mathcal{O}}^{\perp}$  contains the following hyperedges:

$$\begin{aligned} e_1 &= (\{\alpha_1, \alpha_3\}, \{\alpha_2\}) & e_2 &= (\{\alpha_1\}, \{\alpha_4\}) & e_3 &= (\{\alpha_2\}, \{\alpha_3\}) & e_4 &= (\{\alpha_3\}, \{\alpha_1\}) \\ e_5 &= (\{\alpha_3\}, \{\alpha_4\}) & e_6 &= (\{\alpha_4\}, \{\alpha_1\}) & e_7 &= (\{\alpha_4\}, \{\alpha_5\}) & e_8 &= (\{\alpha_5\}, \{\alpha_1\}) \\ e_9 &= (\{\alpha_5\}, \{\alpha_4\}) \end{aligned}$$

We obtain the following  $\perp$ -local modules for the axioms:

$$\begin{aligned} \text{Mod}_{\mathcal{O}}^{\perp}(\alpha_1) &= \text{Mod}_{\mathcal{O}}^{\perp}(\alpha_4) = \text{Mod}_{\mathcal{O}}^{\perp}(\alpha_5) = \{\alpha_1, \alpha_4, \alpha_5\} \\ \text{Mod}_{\mathcal{O}}^{\perp}(\alpha_2) &= \text{Mod}_{\mathcal{O}}^{\perp}(\alpha_3) = \{\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5\} \end{aligned}$$

The resulting atoms in  $\text{Atoms}_{\mathcal{O}}^{\perp}$  are  $\mathbf{a}_1 = \{\alpha_2, \alpha_3\}$  and  $\mathbf{a}_2 = \{\alpha_1, \alpha_4, \alpha_5\}$ , where  $\mathbf{a}_1 \succ \mathbf{a}_2$ , i.e.  $\mathbf{a}_2$  depends on  $\mathbf{a}_1$ . The ADH  $\mathcal{H}_{\mathcal{O}}^{\perp}$  with the SCCs and the condensed ADH  $\mathcal{H}_{\mathcal{O}}^{\perp}/\simeq_B$  is depicted in Figure 1.



**Fig. 1.** Example 4: From the  $\perp$ -ADH to the condensed  $\perp$ -ADH

The strongly connected components of  $\mathcal{H}_{\mathcal{O}}^{\perp}$  are  $SCC_1 = \{\alpha_2, \alpha_3\}$  and  $SCC_2 = \{\alpha_1, \alpha_4, \alpha_5\}$ . Moreover,  $SCC_2$  depends on  $SCC_1$  as any axiom in  $SCC_2$  is  $B$ -connected from any axiom in  $SCC_1$ . Note that the atoms  $\mathbf{a}_1$  and  $\mathbf{a}_2$  of  $\mathcal{O}$  and their dependency coincide with the strongly connected components of  $\mathcal{H}_{\mathcal{O}}^{\perp}$ .  $\triangleleft$

Analogously to the previous section, we can characterise modules in terms of  $B$ -reachability in condensed axiom dependency hypergraphs. Proposition 3 can be lifted to cADHs as follows.

**Proposition 7.** *Let  $\mathcal{O}$  be an ontology,  $\mathcal{O}' \subseteq \mathcal{O}$  and  $\Sigma = \text{sig}(\mathcal{O}')$ . Let  $\simeq$  be the mutual  $B$ -connectivity relation of the  $x$ -ADH for  $\mathcal{O}$  and  $\geq_B$  the  $B$ -connectivity relation of the  $\Sigma$ -extension of the  $x$ -cADH for  $\mathcal{O}$ , where  $x \in \{\perp, \top\}$ . Then:  $\text{Mod}_{\mathcal{O}}^x(\Sigma) = \bigcup_{\geq_B}(\{[\alpha]_{\simeq} \mid \alpha \in \mathcal{O}'\})$ .  $\dashv$*

## 4 Implementation and Evaluation

For a collection of well-known biomedical ontologies from the NCBO Bioportal, we observe that for many (if not all) axioms, the locality-based dependencies to other axioms can be represented using only *simple* directed hyperedges (i.e., hyperedges with only one tail node) [9]. For instance, the ADH for ontologies like CHEBI can be seen as a directed graph without *complex* hyperedges (i.e., hyperedges with more than one tail node). Computing strongly connected components in a graph can be done in linear-time using standard algorithms from graph theory [11, 13]. That is, for ontologies like CHEBI we compute the strongly connected components of the respective ADH in linear time.

For ADHs of ontologies  $\mathcal{O}$  like SNOMED CT that contain both, simple and complex hyperedges, we compute the strongly connected components in four steps. First, we build the axiom dependency graph  $\mathcal{G}_{\mathcal{O}}$ , which is the fragment of the ADH  $\mathcal{H}_{\mathcal{O}}$  for  $\mathcal{O}$  without complex hyperedges. Second, we compute the strongly connected components of  $\mathcal{G}_{\mathcal{O}}$  using a linear-time algorithm [11, 13]. Note that the strongly connected components give rise to an equivalence relation  $\simeq_{\mathcal{G}_{\mathcal{O}}}$  on the nodes in  $\mathcal{G}_{\mathcal{O}}$ . In the third step, we reduce  $\mathcal{H}_{\mathcal{O}}$  by computing the quotient graph  $\mathcal{H}_{\mathcal{O}}/\simeq_{\mathcal{G}_{\mathcal{O}}}$  of  $\mathcal{H}_{\mathcal{O}}$  using  $\simeq_{\mathcal{G}_{\mathcal{O}}}$  (cf. Def. 4). Finally, in step four, we obtain the strongly connected components of  $\mathcal{H}_{\mathcal{O}}$  by determining for any two nodes in  $\mathcal{H}_{\mathcal{O}}/\simeq_{\mathcal{G}_{\mathcal{O}}}$  whether they are mutually reachable. Note that computing mutual reachability this way is a quadratic process. However, using  $\mathcal{H}_{\mathcal{O}}/\simeq_{\mathcal{G}_{\mathcal{O}}}$  instead of  $\mathcal{H}_{\mathcal{O}}$  it is usually more efficient as the number of nodes is typically reduced.

The number of hyperedges may be exponential in the size of the ontology, which makes it impractical to represent the entire ADH explicitly. We implement an ADH  $\mathcal{H} = (\mathcal{V}, \mathcal{E})$  as a directed labelled graph  $\mathcal{G}_{\mathcal{H}} = (\mathcal{V}, \mathcal{E}', \mathcal{L})$  containing the simple hyperedges of  $\mathcal{H}$  and encoding the complex hyperedges in the node labels as follows. A node  $v_{\alpha}$  in  $\mathcal{G}_{\mathcal{H}}$  for an axiom  $\alpha$  is labelled with the pair  $\mathcal{L}(v_{\alpha}) = (\text{MLS}^{\perp}(\alpha), \text{sig}(\alpha))$  consisting of the minimal non- $\perp$ -locality signatures of  $\alpha$  and the signature of  $\alpha$ . Reachable nodes in  $\mathcal{G}_{\mathcal{H}}$  can be computed by walking along the edges in  $\mathcal{E}'$  and via signature containment checks. Condensed ADHs are implemented in a similar way with the difference that nodes represent sets of axioms. For a set  $S$  of axioms, we set  $\text{MLS}^{\perp}(S) = \bigcup_{\alpha \in S} \text{MLS}^{\perp}(\alpha)$ . We note that when computing modules for an arbitrary signature  $\Sigma$  using  $\mathcal{G}_{\mathcal{H}}$ , it is not necessary to compute the  $\Sigma$ -extension of  $\mathcal{G}_{\mathcal{H}}$  (cf. Prop. 4 and 7). Reachable nodes in the  $\Sigma$ -extension of  $\mathcal{G}_{\mathcal{H}}$  can be computed via a modified signature containment check that accounts for  $\Sigma$  using the node labels in  $\mathcal{G}_{\mathcal{H}}$ .

We have implemented a Java prototype that computes the atomic decomposition and locality-based modules of several prominent biomedical ontologies. The implementation takes an  $\mathcal{EL}^{++}$ -ontology as an input and computes the atomic

decomposition wrt.  $\perp$ -locality, and the  $\perp$ -local module of an arbitrary input signature. The current version of our prototype does not normalize the input ontology. For the evaluation of our prototype, we have selected nine well-known biomedical ontologies that are available (with the exception of Snomed CT) in the NCBO Bioportal and in the ORE 2013 repository.<sup>5</sup> We divide the ontologies into two groups. The first group of six ontologies only contains axioms whose dependencies wrt.  $\perp$ -locality can be represented using simple hyperedges, whereas the second group of three ontologies requires both, simple and complex hyperedges.

We compare the performance of our prototype against the performance of two systems for computing the atomic decomposition of OWL2 ontologies which implement the same algorithm [14]: FaCT++ v1.6.2 which is implemented in C++ [14]<sup>6</sup> and OWLAPITools v1.0.0 which is implemented in Java [15]<sup>7</sup> as an extension of the OWLAPI.<sup>8</sup> All experiments were conducted on an Intel Xeon E5-2640 2.50GHz with 100GB RAM running Debian GNU/Linux 7.3. We use Java 1.7.0.51 and the OWLAPI version 3.4.8. The following table lists the results, where each time is the average of 10 executions.

Ontology $\mathcal{O}$	Properties of $\mathcal{O}$				AD time for $\mathcal{O}$		
	$ \text{sig}(\mathcal{O}) $	#axioms $A \sqsubseteq C$	#axioms $C \equiv D$	#role axioms	FaCT++	OWLAPI TOOLS	ADH
CHEBI	37 891	85 342	0	5	137 s	1 619 s	4 s
FMA-lite	75 168	119 558	0	3	18 481 s	13 258 s	17 s
Gazetteer	517 039	652 355	0	6	31 595 s	–	24 s
GO	36 945	72 667	0	2	47 s	1 489 s	4 s
NCBI	847 796	847 755	0	0	49 228 s	–	66 s
RH-Mesh	286 382	403 210	0	0	6 921 s	9 159 s	17 s
CPO	136 090	306 111	73 461	96	9 731 s	26 480 s	2 283 s
Full-Galen	24 088	25 563	9 968	2 165	640 s	781 s	115 s
Snomed CT	291 207	227 698	63 446	12	16 081 s	57 282 s	2 540 s

We applied a timeout of 24h, which aborted the executions of the OWLAPI-TOOLS on the ontologies Gazetteer and NCBI.

Our prototype consistently outperforms FaCT++ which in turn (considerably) outperforms the OWLAPITools, with the exception of FMA-lite. In the case of the first group of six ontologies, an over 1000-fold speedup could be achieved compared to the performance of FaCT++ on FMA-lite and Gazetteer. For the smallest ontology in this group, which is GO, the prototype is 13 times faster than FaCT++. The prototype also scales better than the other systems. For the second group of three ontologies, the speedup is reduced but our prototype is still 4–7 times faster than FaCT++, and 11–23 faster than the OWLAPI-TOOLS. Collapsing the strongly connected components of the graph fragment of

<sup>5</sup> <http://ore2013.cs.manchester.ac.uk/>

<sup>6</sup> <http://code.google.com/p/factplusplus/>

<sup>7</sup> <http://owlapitools.sourceforge.net/>

<sup>8</sup> <http://owlapi.sourceforge.net/>

the ADH helps reducing the number of nodes by nearly 50% in some cases. The use of a tree datastructure to represent the set of reachable nodes computed for each node of the ADH reduces the time needed to identify mutually reachable nodes.

We also compare the performance of our prototype for extracting  $\perp$ -locality modules with the performance of FaCT++ and the OWLAPI. The following table presents for every implementation the time needed to extract a module from an ontology for a signature consisting of 500 symbols selected at random.

Ontology $\mathcal{O}$	Time to compute $\text{Mod}_{\mathcal{O}}^{\perp}(\Sigma)$			Speedup wrt.	
	FaCT++	OWLAPI	cADH	FaCT++	OWLAPI
CHEBI	38.6 ms	175.8 ms	2.1 ms	18.4	83.7
FMA-lite	326.9 ms	1 042.3 ms	3.4 ms	96.1	306.6
Gazetteer	177.9 ms	1 503.0 ms	15.9 ms	11.2	94.5
GO	512.2 ms	1 398.7 ms	6.1 ms	84.0	229.3
NCBI	236.2 ms	9 193.6 ms	16.3 ms	14.5	564.0
RH-Mesh	91.2 ms	1 811.3 ms	8.9 ms	10.2	203.5
CPO	564.7 ms	3 026.8 ms	51.6 ms	10.9	58.7
Full-Galen	75.2 ms	215.4 ms	2.9 ms	25.9	74.3
SNOMED CT	525.0 ms	2 841.3 ms	84.4 ms	6.2	33.7

Our prototype outperforms FaCT++ and the OWLAPI in all cases. For the first group of six ontologies, the best speedup of over 95 times wrt. FaCT++ was achieved in the case of FMA-lite. For the second group of three ontologies, the best performance improvement was realised in the case of Full-Galen with a speedup of over 25-times. The speedup wrt. the OWLAPI is even higher.

## 5 Conclusion

We have introduced the notion of an axiom dependency hypergraph that represents explicitly the locality-based dependencies between axioms. We have shown that locality-based modules of an ontology correspond to a set of connected nodes in the hypergraph, and atoms of an ontology to strongly connected components. We have implemented a prototype in Java that computes, based on axiom dependency hypergraphs, the atomic decomposition of  $\mathcal{EL}^{++}$ -ontologies wrt.  $\perp$ -locality. Our prototype outperforms FaCT++ and the OWLAPITOOLS in computing the atomic decomposition of all biomedical ontologies tested. In some cases a staggering speedup of over 1 000 times could be realised. Moreover, our prototype outperforms FaCT++ and the OWLAPI in extracting syntactic  $\perp$ -locality modules. Here a speedup of over 95 times could be realised.

We plan to extend the prototype implementation to support both  $\top$ -locality and full *SRQI*-ontologies. Moreover, it would be interesting to investigate the possibility of computing strongly connected components in hypergraphs in less than quadratic time. Such a result would improve the performance of computing mutual reachability in the axiom dependency hypergraph for ontologies whose locality-based dependencies can only be represented by complex hyperedges.

## References

1. Allamigeon, X.: Strongly connected components of directed hypergraphs. CoRR abs/1112.1444 (2011)
2. Baader, F., Brandt, S., Lutz, C.: Pushing the EL envelope further. In: Proc. of OWLED'08 (2008)
3. Baader, F., Calvanese, D., McGuinness, D.L., Nardi, D., Patel-Schneider, P.F. (eds.): The description logic handbook: theory, implementation, and applications. Cambridge University Press (2007)
4. Del Vescovo, C., Gessler, D.D.G., Klinov, P., Parsia, B., Sattler, U., Schneider, T., Winget, A.: Decomposition and modular structure of biportal ontologies. In: Proc. of ISWC'11, pp. 130–145. Springer-Verlag (2011)
5. Del Vescovo, C., Parsia, B., Sattler, U., Schneider, T.: The modular structure of an ontology: Atomic decomposition. In: Proc. of IJCAI'11, pp. 2232–2237 (2011)
6. Gallo, G., Longo, G., Pallottino, S., Nguyen, S.: Directed hypergraphs and applications. *Discrete Applied Mathematics* 42(23), 177–201 (1993)
7. Grau, B.C., Horrocks, I., Kazakov, Y., Sattler, U.: Modular reuse of ontologies: theory and practice. *JAIR* 31, 273–318 (2008)
8. Horrocks, I., Kutz, O., Sattler, U.: The even more irresistible SROIQ. In: Proc. of KR'06, pp. 57–67. AAAI Press (2006)
9. Martín-Recuerda, F., Walther, D.: Towards fast atomic decomposition using axiom dependency hypergraphs. In: Proc. of WoMO'13. CEUR Workshop Proceedings, vol. 1081. CEUR-WS.org (2013)
10. Nortje, R., Britz, A., Meyer, T.: Reachability modules for the description logic SRIQ. In: Proc. of LPAR-13 (2013)
11. Sharir, M.: A strong connectivity algorithm and its applications to data flow analysis. *Computers & Mathematics with Applications* 7(1), 67–72 (1981)
12. Suntisrivaraporn, B.: Polynomial time reasoning support for design and maintenance of large-scale biomedical ontologies. Ph.D. thesis, TU Dresden, Germany (2009)
13. Tarjan, R.E.: Depth-first search and linear graph algorithms. *SIAM J. Comput.* 1(2), 146–160 (1972)
14. Tsarkov, D.: Improved algorithms for module extraction and atomic decomposition. In: Proc. of DL'12. CEUR Workshop Proceedings, vol. 846. CEUR-WS.org (2012)
15. Tsarkov, D., Vescovo, C.D., Palmisano, I.: Instrumenting atomic decomposition: Software apis for owl. In: Proc. of OWLED'13. CEUR Workshop Proceedings, vol. 1080. CEUR-WS.org (2013)