Hydra Battles and AC Termination

Nao Hirokawa
School of Information Science, JAIST, Ishikawa, Japan

Aart Middeldorp
Department of Computer Science, Universität Innsbruck, Austria

Abstract

We present a new encoding of the Battle of Hercules and Hydra as a rewrite system with AC symbols. Unlike earlier term rewriting encodings, it faithfully models any strategy of Hercules to beat Hydra. To prove the termination of our encoding, we employ type introduction in connection with many-sorted semantic labeling for AC rewriting and AC-RPO.

2012 ACM Subject Classification Theory of computation → Equational logic and rewriting; Theory of computation → Rewrite systems; Theory of computation → Computability

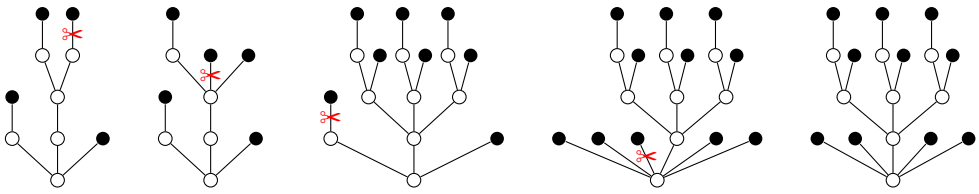
Keywords and phrases battle of Hercules and Hydra, term rewriting, termination

Digital Object Identifier 10.4230/LIPIcs.FSCD.2023.12

Funding *Nao Hirokawa*: JSPS KAKENHI Grant Number JP22K11900.
Aart Middeldorp: Part of this work was performed when the second author was employed at the Future Value Creation Research Center of Nagoya University, Japan.

1 Introduction

The mythological monster Hydra is a dragon-like creature with multiple heads. Whenever Hercules in his fight chops off a head, more and more new heads can grow instead, since the beast gets increasingly angry. Here we model a Hydra as an unordered tree. If Hercules cuts off a leaf corresponding to a head, the tree is modified in the following way: If the cut-off node h has a grandparent n, then the branch from n to the parent of h gets multiplied, where the number of copies depends on the number of decapitations so far. Hydra dies if there are no heads left, in that case Hercules wins. The following sequence shows an example fight:



Though the number of heads can grow considerably in one step, it turns out that the fight always terminates, and Hercules will win independent of his strategy. Proving termination of the Battle is challenging since Kirby and Paris proved in their landmark paper [11] that termination for an arbitrary (computable) strategy is independent of Peano arithmetic. In [11] a termination argument based on ordinals is used.

Starting with [4, p. 271], several TRS encodings of the Battle of Hercules and Hydra have been proposed and studied [3, 5, 7, 17, 21]. Touzet [21] was the first to give a rigorous termination proof and in [24] the automation of ordinal interpretations is discussed. In this paper we present yet another encoding. In contrast to earlier TRS encodings that model a specific strategy, it uses AC matching to represent *arbitrary* battles. To prove its termination, we apply and extend existing termination methods for AC rewriting.

The remainder of the paper is organized as follows. After recalling some basic definitions in Section 2, we present our new encoding of the Battle in Section 3. We give a rigorous proof that our encoding faithfully represents the Battle. In Section 4 we present many-sorted semantic labeling for AC rewriting and apply it to our encoding. This results in an infinite AC rewrite system, which is subjected to AC-RPO [20] in Section 5. Related work is discussed in Section 6. In particular, we comment on earlier encodings of the Battle. We conclude in Section 7 with suggestions for future research.

2 Preliminaries

Let $\mathcal{S}$ be a set of *sorts*. An *$\mathcal{S}$-sorted signature* $\mathcal{F}$ consists of function symbols f having a sort declaration $S_1 \times \cdots \times S_n \to S$. Here $S_1, \ldots, S_n$ and S are sorts in $\mathcal{S}$ and n is the *arity* of f. By $f^{(n)}$ we indicate that f has arity n. Let $\mathcal{V}$ be a countably infinite set of variables, where every variable has its own sort. We assume the existence of infinitely many variables of each sort. *Terms* of sort S are inductively defined as usual: Every variable of sort S is a term of sort S and if f has sort declaration $S_1 \times \cdots \times S_n \to S$ and t_i is a term of sort S_i for all $1 \leqslant i \leqslant n$ then $f(t_1, \ldots, t_n)$ is a term of sort S. *Ground terms* are terms without variables. By $\mathcal{T}(\{f_1, \ldots, f_m\})$ we denote the set of all ground terms over $\{f_1, \ldots, f_m\}$. The *root symbol* $\mathsf{root}(t)$ of a term t is t if it is a variable, and f if $t = f(t_1, \ldots, t_n)$. For every sort S we introduce a fresh constant $\Box_S$, called the *hole*. A term over $\mathcal{F} \uplus \{\Box_S \mid S \in \mathcal{S}\}$ is a *context* over $\mathcal{F}$ if it contains exactly one hole. Given a context C and a term t, we write $C[t]$ for the term resulting from replacing the hole in C by t. A mapping σ that associates each variable to a term of the same sort is a *substitution* if its domain $\{x \in \mathcal{V} \mid \sigma(x) \neq x\}$ is finite. The application $t\sigma$ of σ to a term t is defined as $\sigma(t)$ if t is a variable and $f(t_1\sigma, \ldots, t_n\sigma)$ if $t = f(t_1, \ldots, t_n)$. A binary relation $\to$ on terms is *closed under substitutions* if $s\sigma \to t\sigma$ whenever $s \to t$, for all substitutions σ. It is *closed under contexts* if $C[s] \to C[t]$ whenever $s \to t$, for all contexts C. Moreover, the relation $\to$ is said to be a *rewrite relation* if it is closed under contexts and substitutions.

A *rewrite rule* $\ell \to r$ consists of two terms ℓ and r of the same sort such that all variables in r occur in ℓ. A (many-sorted) *term rewrite system* (TRS) is a set of rewrite rules. We denote by $\to_{\mathcal{R}}$ the smallest rewrite relation that contains the pairs of the TRS $\mathcal{R}$. A rule $\ell \to r$ is *non-collapsing* if r is not a variable. A TRS is called *non-collapsing* if all rules are non-collapsing. Let $\mathcal{F}_{\mathsf{AC}}$ be a subset of the binary function symbols in $\mathcal{F}$ that have sort declarations of the form $S \times S \to S$. We denote by AC the set of equations

$$f(f(x,y),z) \approx f(x,f(y,z)) \qquad\qquad f(x,y) \approx f(y,x)$$

expressing the associativity and commutativity of each $f \in \mathcal{F}_{\mathsf{AC}}$. The relation $\to_{\mathsf{AC}}$ is defined as expected and its reflexive, transitive, and symmetric closure is denoted by $=_{\mathsf{AC}}$. Let $\mathcal{R}$ be a TRS. The relation $=_{\mathsf{AC}} \cdot \to_{\mathcal{R}} \cdot =_{\mathsf{AC}}$ is called AC rewriting and abbreviated by $\to_{\mathcal{R}/\mathsf{AC}}$. We say that $\mathcal{R}$ is *AC terminating* if $\to_{\mathcal{R}/\mathsf{AC}}$ is well-founded. A rewrite relation is a *reduction order* if it is a well-founded order. A reduction order $>$ is *AC-compatible* if the inclusion $=_{\mathsf{AC}} \cdot > \cdot =_{\mathsf{AC}} \subseteq >$ holds. AC termination of a TRS $\mathcal{R}$ can be shown by finding an AC-compatible reduction order such that $\mathcal{R} \subseteq >$ holds.

The above definitions specialize to the usual unsorted setting when the set of sorts is a singleton set.

Finally, we recall two order extensions. Let $>$ be a strict order on a set A. The *lexicographic extension* $>^{\mathsf{lex}}$ of $>$ is defined on tuples over A as follows: $(a_1, \ldots, a_m) >^{\mathsf{lex}} (b_1, \ldots, b_n)$ if $n = m$ and there exists an index $1 \leqslant k \leqslant n$ such that $a_k > b_k$ and $a_i = b_i$ for all $i < k$. The

multiset extension $>^{\mathsf{mul}}$ of $>$ is defined on multisets over A as follows: $M >^{\mathsf{mul}} N$ if there exist multisets X and Y such that $M = (N - X) \uplus Y$, $\varnothing \neq X \subseteq M$, and every $b \in Y$ admits an element $a \in X$ with $a > b$.

3 Encoding

▶ **Definition 1.** *To represent Hydras, we use a signature containing a constant symbol* h *representing a head, a binary symbol* $\mid$ *for siblings, and a unary function symbol* i *representing the internal nodes. We use infix notation for* $\mid$ *and declare it to be an* AC *symbol.*

▶ **Example 2.** The Hydras in the above example fight are represented by the terms

$$
\begin{aligned}
H_0 &= \mathsf{i}(\mathsf{i}(\mathsf{h}) \mid \mathsf{i}(\mathsf{i}(\mathsf{i}(\mathsf{h}) \mid \mathsf{i}(\mathsf{h}))) \mid \mathsf{h}) \\
H_1 &= \mathsf{i}(\mathsf{i}(\mathsf{h}) \mid \mathsf{i}(\mathsf{i}(\mathsf{i}(\mathsf{h}) \mid \mathsf{h} \mid \mathsf{h})) \mid \mathsf{h}) \\
H_2 &= \mathsf{i}(\mathsf{i}(\mathsf{h}) \mid \mathsf{i}(\mathsf{i}(\mathsf{i}(\mathsf{h}) \mid \mathsf{h}) \mid \mathsf{i}(\mathsf{i}(\mathsf{h}) \mid \mathsf{h}) \mid \mathsf{i}(\mathsf{i}(\mathsf{h}) \mid \mathsf{h})) \mid \mathsf{h}) \\
H_3 &= \mathsf{i}(\mathsf{h} \mid \mathsf{h} \mid \mathsf{h} \mid \mathsf{i}(\mathsf{i}(\mathsf{i}(\mathsf{h}) \mid \mathsf{h}) \mid \mathsf{i}(\mathsf{i}(\mathsf{h}) \mid \mathsf{h}) \mid \mathsf{i}(\mathsf{i}(\mathsf{h}) \mid \mathsf{h})) \mid \mathsf{h} \mid \mathsf{h}) \\
H_4 &= \mathsf{i}(\mathsf{h} \mid \mathsf{h} \mid \mathsf{i}(\mathsf{i}(\mathsf{i}(\mathsf{h}) \mid \mathsf{h}) \mid \mathsf{i}(\mathsf{i}(\mathsf{h}) \mid \mathsf{h}) \mid \mathsf{i}(\mathsf{i}(\mathsf{h}) \mid \mathsf{h})) \mid \mathsf{h} \mid \mathsf{h})
\end{aligned}
$$

▶ **Definition 3.** *The* TRS $\mathcal{H}$ *consists of the following 14 rewrite rules:*

$$
\begin{aligned}
\mathsf{A}(n, \mathsf{i}(\mathsf{h})) &\xrightarrow{1} \mathsf{A}(\mathsf{s}(n), \mathsf{h}) & \mathsf{D}(n, \mathsf{i}(\mathsf{i}(x))) &\xrightarrow{8} \mathsf{i}(\mathsf{D}(n, \mathsf{i}(x))) \\
\mathsf{A}(n, \mathsf{i}(\mathsf{h} \mid x)) &\xrightarrow{2} \mathsf{A}(\mathsf{s}(n), \mathsf{i}(x)) & \mathsf{D}(n, \mathsf{i}(\mathsf{i}(x) \mid y)) &\xrightarrow{9} \mathsf{i}(\mathsf{D}(n, \mathsf{i}(x)) \mid y) \\
\mathsf{A}(n, \mathsf{i}(x)) &\xrightarrow{3} \mathsf{B}(n, \mathsf{D}(\mathsf{s}(n), \mathsf{i}(x))) & \mathsf{D}(n, \mathsf{i}(\mathsf{i}(\mathsf{h} \mid x) \mid y)) &\xrightarrow{10} \mathsf{i}(\mathsf{C}(n, \mathsf{i}(x)) \mid y) \\
\mathsf{C}(0, x) &\xrightarrow{4} \mathsf{E}(x) & \mathsf{D}(n, \mathsf{i}(\mathsf{i}(\mathsf{h} \mid x))) &\xrightarrow{11} \mathsf{i}(\mathsf{C}(n, \mathsf{i}(x))) \\
\mathsf{C}(\mathsf{s}(n), x) &\xrightarrow{5} x \mid \mathsf{C}(n, x) & \mathsf{D}(n, \mathsf{i}(\mathsf{i}(\mathsf{h}) \mid y)) &\xrightarrow{12} \mathsf{i}(\mathsf{C}(n, \mathsf{h}) \mid y) \\
\mathsf{i}(\mathsf{E}(x) \mid y) &\xrightarrow{6} \mathsf{E}(\mathsf{i}(x \mid y)) & \mathsf{D}(n, \mathsf{i}(\mathsf{i}(\mathsf{h}))) &\xrightarrow{13} \mathsf{i}(\mathsf{C}(n, \mathsf{h})) \\
\mathsf{i}(\mathsf{E}(x)) &\xrightarrow{7} \mathsf{E}(\mathsf{i}(x)) & \mathsf{B}(n, \mathsf{E}(x)) &\xrightarrow{14} \mathsf{A}(\mathsf{s}(n), x)
\end{aligned}
$$

The Battle is started with the term $\mathsf{A}(0, t)$ where t is the term representation of the initial Hydra. Rule 1 takes care of the dying Hydra ○——●. Rule 2 cuts a head without grandparent node, and so no copying takes place. Due to the power of AC matching, the removed head need not be the leftmost one. With rule 3, the search for locating a head with grandparent node starts. The search is performed with the auxiliary symbol D and involves rules 8–13. When the head to be cut is located (in rules 10–13), copying begins with the auxiliary symbol C and rules 4 and 5. The end of the copying phase is signaled with E, which travels upwards with rules 6 and 7. Finally, rule 14 creates the next stage of the Battle. Note that we make extensive use of AC matching to simplify the search process.

▶ **Theorem 4.** *If* H *and* H' *are the encodings in* $\mathcal{T}(\{\mathsf{h}, \mathsf{i}, \mid\})$ *of successive Hydras in an arbitrary battle then* $\mathsf{A}(n, H) \to^{+}_{\mathcal{H}/\mathsf{AC}} \mathsf{A}(\mathsf{s}(n), H')$ *for some* $n \in \mathcal{T}(\{0, \mathsf{s}\})$.

Before presenting the proof, we illustrate how the rewrite rules transform H_0 to H_1 in Example 2.

▶ **Example 5.** The following sequence simulates the first step in the example fight:

$$
\begin{aligned}
\mathsf{A}(0, H_0) &\xrightarrow{3} \mathsf{B}(0, \mathsf{D}(\mathsf{s}(0), H_0)) \\
&=_{\mathsf{AC}} \cdot \xrightarrow{9} \mathsf{B}(0, \mathsf{i}(\mathsf{D}(\mathsf{s}(0), \mathsf{i}(\mathsf{i}(\mathsf{i}(\mathsf{h}) \mid \mathsf{i}(\mathsf{h})))) \mid \mathsf{i}(\mathsf{h}) \mid \mathsf{h})) \\
&\xrightarrow{8} \mathsf{B}(0, \mathsf{i}(\mathsf{i}(\mathsf{D}(\mathsf{s}(0), \mathsf{i}(\mathsf{i}(\mathsf{h}) \mid \mathsf{i}(\mathsf{h})))) \mid \mathsf{i}(\mathsf{h}) \mid \mathsf{h}))
\end{aligned}
$$

$$
\begin{aligned}
&\xrightarrow{12} \mathsf{B}(0, \mathsf{i}(\mathsf{i}(\mathsf{i}(\mathsf{C}(\mathsf{s}(0), \mathsf{h}) \mid \mathsf{i}(\mathsf{h}))) \mid \mathsf{i}(\mathsf{h}) \mid \mathsf{h})) \\
&\xrightarrow{5} \mathsf{B}(0, \mathsf{i}(\mathsf{i}(\mathsf{i}(\mathsf{h} \mid \mathsf{C}(0, \mathsf{h}) \mid \mathsf{i}(\mathsf{h}))) \mid \mathsf{i}(\mathsf{h}) \mid \mathsf{h})) \\
&\xrightarrow{4} \mathsf{B}(0, \mathsf{i}(\mathsf{i}(\mathsf{i}(\mathsf{h} \mid \mathsf{E}(\mathsf{h}) \mid \mathsf{i}(\mathsf{h}))) \mid \mathsf{i}(\mathsf{h}) \mid \mathsf{h})) \\
=_{\mathsf{AC}} \cdot &\xrightarrow{6} \mathsf{B}(0, \mathsf{i}(\mathsf{i}(\mathsf{E}(\mathsf{i}(\mathsf{h} \mid \mathsf{h} \mid \mathsf{i}(\mathsf{h})))) \mid \mathsf{i}(\mathsf{h}) \mid \mathsf{h})) \\
&\xrightarrow{7} \mathsf{B}(0, \mathsf{i}(\mathsf{E}(\mathsf{i}(\mathsf{i}(\mathsf{h} \mid \mathsf{h} \mid \mathsf{i}(\mathsf{h})))) \mid \mathsf{i}(\mathsf{h}) \mid \mathsf{h})) \\
&\xrightarrow{6} \mathsf{B}(0, \mathsf{E}(\mathsf{i}(\mathsf{i}(\mathsf{i}(\mathsf{h} \mid \mathsf{h} \mid \mathsf{i}(\mathsf{h}))) \mid \mathsf{i}(\mathsf{h}) \mid \mathsf{h}))) \\
&\xrightarrow{14} \mathsf{A}(\mathsf{s}(0), \mathsf{i}(\mathsf{i}(\mathsf{i}(\mathsf{h} \mid \mathsf{h} \mid \mathsf{i}(\mathsf{h}))) \mid \mathsf{i}(\mathsf{h}) \mid \mathsf{h})) =_{\mathsf{AC}} \mathsf{A}(\mathsf{s}(0), H_1) \qquad \lrcorner
\end{aligned}
$$

It is important to note that the TRS $\mathcal{H}$ defined above is *unsorted* and we establish in this paper the result that it is AC terminating on all terms. When simulating a battle, like in the statement of the Theorem 4, we deal with well-behaved terms adhering to the sort discipline introduced shortly. The restriction to sorted terms is crucial for our termination proof, but entails no loss of generality. This is due to the following result, which is a special case of [15, Corollary 3.9].

▶ **Theorem 6.** *A non-collapsing* TRS *over a many-sorted signature is* AC *terminating if and only if the corresponding* TRS *over the unsorted version of the signature is* AC *terminating.*

The idea of using sorts to simplify termination proof goes back to Zantema [25]. The TRS $\mathcal{H}$ can be seen as a TRS over the many-sorted signature $\mathcal{F}'$:

$$
\begin{array}{llll}
\mathsf{h} : \mathsf{O} & \mathsf{i}, \mathsf{E} : \mathsf{O} \to \mathsf{O} & \mid : \mathsf{O} \times \mathsf{O} \to \mathsf{O} & \mathsf{A}, \mathsf{B} : \mathsf{N} \times \mathsf{O} \to \mathsf{S} \\
0 : \mathsf{N} & \mathsf{s} : \mathsf{N} \to \mathsf{N} & & \mathsf{C}, \mathsf{D} : \mathsf{N} \times \mathsf{O} \to \mathsf{O}
\end{array}
$$

where N, O and S are sort symbols. Since $\mathcal{H}$ is non-collapsing, Theorem 6 guarantees that AC termination of $\mathcal{H}$ follows from AC termination of well-sorted terms over $\mathcal{F}'$.

In the remainder of this section we present a proof of Theorem 4 and its converse.

▶ **Definition 7.** *Let n be a natural number. The* TRS $\mathcal{R}_n$ *operates on encodings of Hydras and consists of the following four rules:*

$$
\begin{array}{rlrl}
\mathsf{i}(\mathsf{i}(\mathsf{h})) & \xrightarrow{1} \mathsf{i}(\mathsf{h}^{n+2}) & \mathsf{i}(\mathsf{i}(\mathsf{h}) \mid y) & \xrightarrow{3} \mathsf{i}(\mathsf{h}^{n+2} \mid y) \\
\mathsf{i}(\mathsf{i}(\mathsf{h} \mid x)) & \xrightarrow{2} \mathsf{i}(\mathsf{i}(x)^{n+2}) & \mathsf{i}(\mathsf{i}(\mathsf{h} \mid x) \mid y) & \xrightarrow{4} \mathsf{i}(\mathsf{i}(x)^{n+2} \mid y)
\end{array}
$$

Here t^k for $k \geqslant 1$ is defined inductively as follows:

$$
t^k = \begin{cases} t & \text{if } k = 1 \\ t^{k-1} \mid t & \text{if } k > 1 \end{cases}
$$

The following lemma relates successive Hydras in a battle to the rules in Definition 7. The easy proof is omitted.

▶ **Lemma 8.** *If H and H' be the encodings of Hydras at stages n and $n+1$ in a battle then*

1. *$H = \mathsf{i}(\mathsf{h})$ and $H' = \mathsf{h}$, or*
2. *$H =_{\mathsf{AC}} \mathsf{i}(\mathsf{h} \mid t)$ and $H' = \mathsf{i}(t)$ for some term t, or*
3. *$H \to_{\mathcal{R}_n/\mathsf{AC}} H'$.*

In the following we write n for $\mathsf{s}^n(0)$. Moreover, $\mathcal{T}_{\mathcal{H}}$ denotes the set of ground terms over $\{\mathsf{h}, \mathsf{i}, \mid\}$, and $\mathcal{C}_{\mathcal{H}}$ denotes the set of ground contexts over $\{\mathsf{h}, \mathsf{i}, \mid\}$.

▶ **Lemma 9.** *If $n > 0$ then $\mathsf{C}(\mathsf{n}, t) \to^*_{\mathcal{H}/\mathsf{AC}} t^n \mid \mathsf{E}(t)$ for all terms t.*

Proof. We use induction on n. If $n = 1$ then

$$\mathsf{C}(\mathsf{n}, t) \xrightarrow{5} t \mid \mathsf{C}(0, t) \xrightarrow{6} t \mid \mathsf{E}(t) = t^n \mid \mathsf{E}(t)$$

Suppose the result holds for $n \geqslant 1$ and consider $n + 1$. The induction hypothesis yields $\mathsf{C}(\mathsf{n}, t) \to^*_{\mathcal{H}/\mathsf{AC}} t^n \mid \mathsf{E}(t)$. Hence

$$\mathsf{C}(\mathsf{n{+}1}, t) \xrightarrow{5} t \mid \mathsf{C}(\mathsf{n}, t) \to^*_{\mathcal{H}/\mathsf{AC}} t \mid (t^n \mid \mathsf{E}(t)) =_{\mathsf{AC}} t^{n+1} \mid \mathsf{E}(t)$$ ◀

▶ **Lemma 10.** *If $n \geqslant 0$ then $\mathsf{D}(\mathsf{n{+}1}, H) \to^*_{\mathcal{H}/\mathsf{AC}} \mathsf{E}(H')$.*

Proof. We use structural induction on H and consider the following two cases.

- First suppose $H \to_{\mathcal{R}_n/\mathsf{AC}} H'$ is a root step. If the first rule of $\mathcal{R}_n$ is used then $H = \mathsf{i}(\mathsf{i}(\mathsf{h}))$ and $H' =_{\mathsf{AC}} \mathsf{i}(h^{n+2})$. We have $\mathsf{D}(\mathsf{n{+}1}, H) \xrightarrow{13} \mathsf{i}(\mathsf{C}(\mathsf{n{+}1}, \mathsf{h}))$. Using Lemma 9 we obtain

  $$\mathsf{i}(\mathsf{C}(\mathsf{n{+}1}, \mathsf{h})) \to^*_{\mathcal{H}/\mathsf{AC}} \mathsf{i}(\mathsf{h}^{n+1} \mid \mathsf{E}(\mathsf{h})) =_{\mathsf{AC}} \cdot \xrightarrow{6} \mathsf{E}(\mathsf{i}(\mathsf{h} \mid \mathsf{h}^{n+1})) =_{\mathsf{AC}} \mathsf{E}(H')$$

  If the second rule of $\mathcal{R}_n$ is used then $H =_{\mathsf{AC}} \mathsf{i}(\mathsf{i}(\mathsf{h} \mid t))$ and $H' =_{\mathsf{AC}} \mathsf{i}(\mathsf{i}(t)^{n+2})$ for some term t. We have $\mathsf{D}(\mathsf{n{+}1}, H) =_{\mathsf{AC}} \cdot \xrightarrow{11} \mathsf{i}(\mathsf{C}(\mathsf{n{+}1}, \mathsf{i}(t)))$. Using Lemma 9 we obtain

  $$\mathsf{i}(\mathsf{C}(\mathsf{n{+}1}, \mathsf{i}(t))) \to^*_{\mathcal{H}/\mathsf{AC}} \mathsf{i}(\mathsf{i}(t)^{n+1} \mid \mathsf{E}(\mathsf{i}(t))) =_{\mathsf{AC}} \cdot \xrightarrow{6} \mathsf{E}(\mathsf{i}(\mathsf{i}(t) \mid \mathsf{i}(t)^{n+1})) =_{\mathsf{AC}} \mathsf{E}(H')$$

  If the third rule of $\mathcal{R}_n$ is used then $H =_{\mathsf{AC}} \mathsf{i}(\mathsf{i}(\mathsf{h}) \mid t)$ and $H' =_{\mathsf{AC}} \mathsf{i}(\mathsf{h}^{n+2} \mid t)$ for some term t. We have $\mathsf{D}(\mathsf{n{+}1}, H) =_{\mathsf{AC}} \cdot \xrightarrow{12} \mathsf{i}(\mathsf{C}(\mathsf{n{+}1}, \mathsf{h}) \mid t)$. The remaining argument is the same as in the preceding cases. If the fourth rule of $\mathcal{R}_n$ is used then $H =_{\mathsf{AC}} \mathsf{i}(\mathsf{i}(\mathsf{h} \mid s) \mid t)$ and $H' =_{\mathsf{AC}} \mathsf{i}(\mathsf{i}(s)^{n+2} \mid t)$ for some terms s and t. Using Lemma 9 we obtain

  $$\begin{aligned}\mathsf{D}(\mathsf{n{+}1}, H) &=_{\mathsf{AC}} \cdot \xrightarrow{10} \mathsf{i}(\mathsf{C}(\mathsf{n{+}1}, \mathsf{i}(s)) \mid t) \to^*_{\mathcal{H}/\mathsf{AC}} \mathsf{i}((\mathsf{i}(s)^{n+1} \mid \mathsf{E}(\mathsf{i}(s))) \mid t)\\ &=_{\mathsf{AC}} \cdot \xrightarrow{6} \mathsf{E}(\mathsf{i}(\mathsf{i}(s) \mid (\mathsf{i}(s)^{n+1} \mid t))) =_{\mathsf{AC}} \mathsf{E}(H')\end{aligned}$$

- Otherwise, $H =_{\mathsf{AC}} \mathsf{i}(H_1 \mid H_2 \mid \cdots \mid H_m)$ and $H' =_{\mathsf{AC}} \mathsf{i}(H_1' \mid H_2 \mid \cdots \mid H_m)$ for some $m \geqslant 1$ and Hydras $H_1, \ldots, H_m, H_1'$ with $H_1 \to_{\mathcal{R}_n/\mathsf{AC}} H_1'$. We obtain $\mathsf{D}(\mathsf{n{+}1}, H_1) \to^*_{\mathcal{H}/\mathsf{AC}} \mathsf{E}(H_1')$ from the induction hypothesis. Note that $\mathsf{root}(H_1) = \mathsf{i}$. If $m = 1$ then

  $$\begin{aligned}\mathsf{D}(\mathsf{n{+}1}, H) &=_{\mathsf{AC}} \mathsf{D}(\mathsf{n{+}1}, \mathsf{i}(H_1)) \xrightarrow{8} \mathsf{i}(\mathsf{D}(\mathsf{n{+}1}, H_1)) \to^*_{\mathcal{H}/\mathsf{AC}} \mathsf{i}(\mathsf{E}(H_1')) \xrightarrow{7} \mathsf{E}(\mathsf{i}(H_1'))\\ &=_{\mathsf{AC}} \mathsf{E}(H_1)\end{aligned}$$

  and if $m > 1$ we reach the same conclusion using rules 9 and 6 instead of 8 and 7. ◀

Proof of Theorem 4. Our task is to show

$$\mathsf{A}(\mathsf{n}, H) \to^*_{\mathcal{H}/\mathsf{AC}} \mathsf{A}(\mathsf{n{+}1}, H')$$

For the first two cases in Lemma 8 the claim is immediate by rules 1 and 2 of $\mathcal{H}$. To verify the third case, assume $H \to_{\mathcal{R}_n/\mathsf{AC}} H'$. This implies $\mathsf{root}(H) = \mathsf{i}$. Using rules 3 and 14 together with Lemma 10 yields

$$\mathsf{A}(\mathsf{n}, H) \xrightarrow{3} \mathsf{B}(\mathsf{n}, \mathsf{D}(\mathsf{n{+}1}, H)) \to^*_{\mathcal{H}/\mathsf{AC}} \mathsf{B}(\mathsf{n}, \mathsf{E}(H')) \xrightarrow{14} \mathsf{A}(\mathsf{n{+}1}, H')$$ ◀

In the remaining part of this section we prove the converse of Theorem 4.

▶ **Theorem 11.** *Let $H, H' \in \mathcal{T}_\mathcal{H}$ be encodings of Hydras and let n be a natural number. If $\mathsf{A}(\mathsf{n}, H) \to^*_{\mathcal{H}/\mathsf{AC}} \mathsf{A}(\mathsf{n{+}1}, H')$ then H and H' are successive Hydras in a battle.*

In order to show the claim we need a few auxiliary lemmata.

▶ **Definition 12.** *We define U as the set consisting of all terms of the forms $\mathsf{A}(\mathsf{n}, t)$, $\mathsf{B}(\mathsf{n}, C[\mathsf{C}(\mathsf{m}, t)])$, $\mathsf{B}(\mathsf{n}, C[\mathsf{D}(\mathsf{n{+}1}, t)])$, and $\mathsf{B}(\mathsf{n}, C[\mathsf{E}(t)])$, where $n, m \in \mathbb{N}$, $t \in \mathcal{T}_\mathcal{H}$, and $C \in \mathcal{C}_\mathcal{H}$.*

The set U contains all terms reachable from $\mathsf{A}(\mathsf{n}, H)$.

▶ **Lemma 13.** *If $t \in U$ and $t \to^*_{\mathcal{H}\cup\mathsf{AC}} u$ then $u \in U$.*

In order to analyze the rewrite sequence $\mathsf{A}(\mathsf{n}, H) \to^*_{\mathcal{H}/\mathsf{AC}} \mathsf{A}(\mathsf{n{+}1}, H')$ we define three subsets of $\mathcal{H}$: $\mathcal{H}_1 = \{1, 2\}$, $\mathcal{H}_2 = \{3, 4, 5, 6, 7, 8, 9, 14\}$, and $\mathcal{H}_3 = \{10, 11, 12, 13\}$. The rewrite sequence in Example 5 can then be described as follows:

$$\begin{aligned}\mathsf{A}(0, H_0) &\to^*_{\mathcal{H}_2/\mathsf{AC}} \mathsf{B}(0, \mathsf{i}(\mathsf{i}(\mathsf{D}(\mathsf{s}(0), \mathsf{i}(\mathsf{i}(\mathsf{h}) \mid \mathsf{i}(\mathsf{h})))) \mid \mathsf{i}(\mathsf{h}) \mid \mathsf{h}))\\ &\to_{\mathcal{H}_3/\mathsf{AC}} \mathsf{B}(0, \mathsf{i}(\mathsf{i}(\mathsf{i}(\mathsf{C}(\mathsf{s}(0), \mathsf{h}) \mid \mathsf{i}(\mathsf{h}))) \mid \mathsf{i}(\mathsf{h}) \mid \mathsf{h}))\\ &\to^*_{\mathcal{H}_2/\mathsf{AC}} \mathsf{A}(1, H_1)\end{aligned}$$

▶ **Definition 14.** *We define V as the extension of U with $\mathcal{T}_\mathcal{H}$ and all terms of the forms $C[\mathsf{C}(\mathsf{n}, t)]$ $C[\mathsf{D}(\mathsf{n}, t)]$, and $C[\mathsf{E}(t)]$ where $n \in \mathbb{N}$, $t \in \mathcal{T}_\mathcal{H}$, and $C \in \mathcal{C}_\mathcal{H}$. The mapping $\pi\colon V \to \mathcal{T}_\mathcal{H}$ is defined as follows:*

$$\pi(t) = \begin{cases} \mathsf{h} & \text{if } t = \mathsf{h} \\ \mathsf{i}(\pi(u)) & \text{if } t = \mathsf{i}(u) \\ \pi(u) \mid \pi(v) & \text{if } t = u \mid v \\ u & \text{if } t = \mathsf{A}(\mathsf{n}, u) \text{ or } t = \mathsf{D}(\mathsf{n}, u) \text{ or } t = \mathsf{E}(u) \\ \pi(u) & \text{if } t = \mathsf{B}(\mathsf{n}, u) \\ u^{n+1} & \text{if } t = \mathsf{C}(\mathsf{n}, u) \end{cases}$$

Taking the role of C into account, the mapping π computes the Hydra in a given term. Applying π to the terms in the above rewrite sequence of $\mathcal{H}_2/\mathsf{AC}$ and $\mathcal{H}_3/\mathsf{AC}$, we obtain

$$\begin{aligned}H_0 = \pi(\mathsf{A}(0, H_0)) &=_\mathsf{AC} \pi(\mathsf{B}(0, \mathsf{i}(\mathsf{i}(\mathsf{D}(\mathsf{s}(0), \mathsf{i}(\mathsf{i}(\mathsf{h}) \mid \mathsf{i}(\mathsf{h})))) \mid \mathsf{i}(\mathsf{h}) \mid \mathsf{h})))\\ &\to_{\mathcal{R}_0/\mathsf{AC}} \pi(\mathsf{B}(0, \mathsf{i}(\mathsf{i}(\mathsf{i}(\mathsf{C}(\mathsf{s}(0), \mathsf{h}) \mid \mathsf{i}(\mathsf{h}))) \mid \mathsf{i}(\mathsf{h}) \mid \mathsf{h})))\\ &=_\mathsf{AC} \pi(\mathsf{A}(1, H_1)) = H_1\end{aligned}$$

This verifies that H_1 is a successor of H_0.

▶ **Lemma 15.** *The following properties hold.*

1. *$\pi(t) = t$ for all terms $t \in \mathcal{T}_\mathcal{H}$,*
2. *$\pi(C[t]) = C[\pi(t)]$ for all terms $t \in V$ and contexts $C \in \mathcal{C}_\mathcal{H}$,*
3. *$\pi(C[t]) =_\mathsf{AC} \pi(D[u])$ for all terms $t, u \in \mathcal{T}_\mathcal{H}$ and contexts $C, D \in \mathcal{C}_\mathcal{H}$ with $t =_\mathsf{AC} u$ and $C =_\mathsf{AC} D$.*

Proof. The first statement is proved by induction on $t \in \mathcal{T}_\mathcal{H}$. If $t = \mathsf{h}$ then $\pi(t) = \mathsf{h} = t$. If $t = \mathsf{i}(u)$ with $u \in \mathcal{T}_\mathcal{H}$ then $\pi(t) = \mathsf{i}(\pi(u)) = \mathsf{i}(u) = t$. If $t = u \mid v$ with $u, v \in \mathcal{T}_\mathcal{H}$ then $\pi(t) = \pi(u) \mid \pi(v) = u \mid v = t$. For the second statement we use induction on the context $C \in \mathcal{C}_\mathcal{H}$. If $C = \Box$ then $\pi(C[t]) = \pi(t) = C[\pi(t)]$. If $C = \mathsf{i}(D)$ then $\pi(C[t]) = \mathsf{i}(\pi(D[t])) = \mathsf{i}(D[\pi(t)]) = C[\pi(t)]$. If $C = D \mid u$ then $D \in \mathcal{C}_\mathcal{H}$ and $u \in \mathcal{T}_\mathcal{H}$ and thus $\pi(C[t]) = \pi(D[t]) \mid \pi(u) = D[\pi(t)] \mid u = C[\pi(t)]$. If $C = u \mid D$ then $D \in \mathcal{C}_\mathcal{H}$ and $u \in \mathcal{T}_\mathcal{H}$ and thus $\pi(C[t]) = \pi(u) \mid \pi(D[t]) = u \mid D[\pi(t)] = C[\pi(t)]$. The third statement follows from statements (1) and (2): $\pi(C[t]) = C[\pi(t)] = C[t] =_\mathsf{AC} D[t] =_\mathsf{AC} D[u] = D[\pi(u)] = \pi(D[u])$. ◀

The following lemma relates AC rewriting of $\mathcal{H}$ to rewriting of Hydras according to Definition 7.

▶ **Lemma 16.** *The following statements hold for all terms $s \in U$.*

1. *If $s =_{\mathsf{AC}} t$ then $\pi(s) =_{\mathsf{AC}} \pi(t)$.*
2. *If $s \to_{\mathcal{H}_2} t$ then $\pi(s) =_{\mathsf{AC}} \pi(t)$.*
3. *If $s \to_{\mathcal{H}_3} t$ then $\pi(s) \to_{\mathcal{R}_n/\mathsf{AC}} \pi(t)$ with $s = \mathsf{B}(\mathsf{n}, s')$ for some $n \geqslant 0$.*

Proof. Let $s \in U$.

1. If $s = \mathsf{A}(\mathsf{n}, u)$ with $u \in \mathcal{T}_{\mathcal{H}}$ then $t = \mathsf{A}(\mathsf{n}, v)$ for some term $v \in \mathcal{T}_{\mathcal{H}}$ with $u =_{\mathsf{AC}} v$. Since $\pi(s) = u$ and $\pi(t) = v$, $\pi(s) =_{\mathsf{AC}} \pi(t)$ follows. If $s = \mathsf{B}(\mathsf{n}, C[\mathsf{C}(\mathsf{m}, u)])$ with $n, m \in \mathbb{N}$, $C \in \mathcal{C}_{\mathcal{H}}$ and $u \in \mathcal{T}_{\mathcal{H}}$ then $t = \mathsf{B}(\mathsf{n}, D[\mathsf{C}(\mathsf{m}, v)])$ with $C =_{\mathsf{AC}} D$ and $u =_{\mathsf{AC}} v$. Using Lemma 15(1,2) we obtain $\pi(s) = \pi(C[\mathsf{C}(\mathsf{m}, u)]) = C[\pi(\mathsf{C}(\mathsf{m}, u))] = C[u^{m+1}]$ and $\pi(t) = D[v^{m+1}]$. From $u =_{\mathsf{AC}} v$ we infer $u^{m+1} =_{\mathsf{AC}} v^{m+1}$ and thus $\pi(s) =_{\mathsf{AC}} \pi(t)$ by Lemma 15(3). The cases $s = \mathsf{B}(\mathsf{n}, C[\mathsf{D}(\mathsf{n}{+}1, u)])$ and $s = \mathsf{B}(\mathsf{n}, C[\mathsf{E}(u)])$ are treated in the same way.
2. For the second statement we make a case analysis based on the employed rule in $\mathcal{H}_2$.
   - If $s \xrightarrow{3} t$ then $s = \mathsf{A}(\mathsf{n}, \mathsf{i}(u))$ and $t = \mathsf{B}(\mathsf{n}, \mathsf{D}(\mathsf{n}{+}1, \mathsf{i}(u)))$ for some $n \geqslant 0$ and $u \in \mathcal{T}_{\mathcal{H}}$. We have $\pi(s) = \mathsf{i}(u) = \pi(\mathsf{D}(\mathsf{n}{+}1, \mathsf{i}(u))) = \pi(t)$ by the definition of t.
   - If $s \xrightarrow{4} t$ then $s = \mathsf{B}(\mathsf{n}, C[\mathsf{C}(\mathsf{0}, u)])$ and $t = \mathsf{B}(\mathsf{n}, C[\mathsf{E}(u)])$ for some $n \geqslant 0$, $C \in \mathcal{C}_{\mathcal{H}}$ and $u \in \mathcal{T}_{\mathcal{H}}$. We have $\pi(s) = \pi(C[\mathsf{C}(\mathsf{0}, u)]) = C[u^1] = C[u] = \pi(C[u]) = \pi(t)$.
   - If $s \xrightarrow{5} t$ then $s = \mathsf{B}(\mathsf{n}, C[\mathsf{C}(\mathsf{m}, u)])$ and $t = \mathsf{B}(\mathsf{n}, C[u \mid \mathsf{C}(\mathsf{m}{-}1, u)])$ for some $n \geqslant 0$, $m > 0$, $C \in \mathcal{C}_{\mathcal{H}}$ and $u \in \mathcal{T}_{\mathcal{H}}$. We have $\pi(s) = C[u^{m+1}] =_{\mathsf{AC}} C[u \mid u^m] = C[\pi(u \mid u^m)] = \pi(C[u \mid \mathsf{C}(\mathsf{m}{-}1, u)]) = \pi(t)$.
   - If $s \xrightarrow{6} t$ then $s = \mathsf{B}(\mathsf{n}, C[\mathsf{i}(\mathsf{E}(u) \mid v)])$ and $t = \mathsf{B}(\mathsf{n}, C[\mathsf{E}(\mathsf{i}(u \mid v))])$ for some $n \geqslant 0$, $C \in \mathcal{C}_{\mathcal{H}}$ and $u, v \in \mathcal{T}_{\mathcal{H}}$. We have $\pi(s) = \pi(C[\mathsf{i}(\mathsf{E}(u) \mid v)]) = C[\mathsf{i}(u \mid v)] = \pi(C[\mathsf{E}(\mathsf{i}(u \mid v))]) = \pi(t)$.
   - If $s \xrightarrow{7} t$ then $s = \mathsf{B}(\mathsf{n}, C[\mathsf{i}(\mathsf{E}(u))])$ and $t = \mathsf{B}(\mathsf{n}, C[\mathsf{E}(\mathsf{i}(u))])$ for some $n \geqslant 0$, $C \in \mathcal{C}_{\mathcal{H}}$ and $u \in \mathcal{T}_{\mathcal{H}}$. We have $\pi(s) = \pi(C[\mathsf{i}(\mathsf{E}(u))]) = C[\mathsf{i}(u)] = \pi(C[\mathsf{E}(\mathsf{i}(u))]) = \pi(t)$.
   - If $s \xrightarrow{8} t$ then $s = \mathsf{B}(\mathsf{n}, C[\mathsf{D}(\mathsf{n}{+}1, \mathsf{i}(\mathsf{i}(u)))])$ and $t = \mathsf{B}(\mathsf{n}, C[\mathsf{i}(\mathsf{D}(\mathsf{n}{+}1, \mathsf{i}(u)))])$ for some $n \geqslant 0$, $C \in \mathcal{C}_{\mathcal{H}}$ and $u \in \mathcal{T}_{\mathcal{H}}$. We have $\pi(s) = C[\mathsf{i}(\mathsf{i}(u))] = \pi(t)$.
   - If $s \xrightarrow{9} t$ then $s = \mathsf{B}(\mathsf{n}, C[\mathsf{D}(\mathsf{n}{+}1, \mathsf{i}(\mathsf{i}(u) \mid v))])$ and $t = \mathsf{B}(\mathsf{n}, C[\mathsf{i}(\mathsf{D}(\mathsf{n}{+}1, \mathsf{i}(u)) \mid v)])$ for some $n \geqslant 0$, $C \in \mathcal{C}_{\mathcal{H}}$ and $u, v \in \mathcal{T}_{\mathcal{H}}$. In this case we obtain $\pi(s) = C[\mathsf{i}(\mathsf{i}(u) \mid v)] = \pi(t)$.
   - If $s \xrightarrow{14} t$ then $s = \mathsf{B}(\mathsf{n}, \mathsf{E}(u))$ and $t = \mathsf{A}(\mathsf{n}{+}1, u)$ for some $n \geqslant 0$ and $u \in \mathcal{T}_{\mathcal{H}}$. In this case we have $\pi(s) = \pi(\mathsf{E}(u)) = u = \pi(t)$.
3. Again we make a case analysis on the applied rewrite rule.
   - If $s \xrightarrow{10} t$ then $s = \mathsf{B}(\mathsf{n}, C[\mathsf{D}(\mathsf{n}{+}1, \mathsf{i}(\mathsf{i}(\mathsf{h} \mid u) \mid v))])$ and $t = \mathsf{B}(\mathsf{n}, C[\mathsf{i}(\mathsf{C}(\mathsf{n}{+}1, \mathsf{i}(u)) \mid v)])$ for some $n \geqslant 0$, $C \in \mathcal{C}_{\mathcal{H}}$ and $u, v \in \mathcal{T}_{\mathcal{H}}$. We obtain $\pi(s) = C[\mathsf{i}(\mathsf{i}(\mathsf{h} \mid u) \mid v)]$ and $\pi(t) = C[\mathsf{i}(\mathsf{i}(u)^{n+2} \mid v)]$. Hence $\pi(s) \to_{\mathcal{R}_n} \pi(t)$ by applying rule 4 of $\mathcal{R}_n$.
   - If $s \xrightarrow{11} t$ then $s = \mathsf{B}(\mathsf{n}, C[\mathsf{D}(\mathsf{n}{+}1, \mathsf{i}(\mathsf{i}(\mathsf{h} \mid u)))])$ and $t = \mathsf{B}(\mathsf{n}, C[\mathsf{i}(\mathsf{C}(\mathsf{n}{+}1, \mathsf{i}(u)))])$ for some $n \geqslant 0$, $C \in \mathcal{C}_{\mathcal{H}}$ and $u, v \in \mathcal{T}_{\mathcal{H}}$. We obtain $\pi(s) = C[\mathsf{i}(\mathsf{i}(\mathsf{h} \mid u))]$ and $\pi(t) = C[\mathsf{i}(\mathsf{i}(u)^{n+2})]$. Hence $\pi(s) \to_{\mathcal{R}_n} \pi(t)$ by applying rule 2 of $\mathcal{R}_n$.
   - If $s \xrightarrow{12} t$ then $s = \mathsf{B}(\mathsf{n}, C[\mathsf{D}(\mathsf{n}{+}1, \mathsf{i}(\mathsf{i}(\mathsf{h}) \mid v))])$ and $t = \mathsf{B}(\mathsf{n}, C[\mathsf{i}(\mathsf{C}(\mathsf{n}{+}1, \mathsf{h}) \mid v)])$ for some $n \geqslant 0$, $C \in \mathcal{C}_{\mathcal{H}}$ and $v \in \mathcal{T}_{\mathcal{H}}$. We obtain $\pi(s) = C[\mathsf{i}(\mathsf{i}(\mathsf{h}) \mid v)]$ and $\pi(t) = C[\mathsf{i}(\mathsf{h}^{n+2} \mid v)]$. Hence $\pi(s) \to_{\mathcal{R}_n} \pi(t)$ by applying rule 3 of $\mathcal{R}_n$.
   - If $s \xrightarrow{13} t$ then $s = \mathsf{B}(\mathsf{n}, C[\mathsf{D}(\mathsf{n}{+}1, \mathsf{i}(\mathsf{i}(\mathsf{h})))])$ and $t = \mathsf{B}(\mathsf{n}, C[\mathsf{i}(\mathsf{C}(\mathsf{n}{+}1, \mathsf{h}))])$ for some $n \geqslant 0$ and $C \in \mathcal{C}_{\mathcal{H}}$. We obtain $\pi(s) = C[\mathsf{i}(\mathsf{i}(\mathsf{h}))]$ and $\pi(t) = C[\mathsf{i}(\mathsf{h}^{n+2})]$. Hence $\pi(s) \to_{\mathcal{R}_n} \pi(t)$ by applying rule 1 of $\mathcal{R}_n$. ◀

So we are ready to prove the main claim.

Proof of Theorem 11. Suppose $s = \mathsf{A}(\mathsf{n}, H) \to^+_{\mathcal{H}/\mathsf{AC}} \mathsf{A}(\mathsf{n}{+}1, H') = t$. Inspection of $\mathcal{H}$ reveals that one of the following two cases holds:

(a) $s \to_{\mathcal{H}_1/\mathsf{AC}} t$, or

(b) $s \to^*_{\mathcal{H}_2/\mathsf{AC}} \cdot \to_{\mathcal{H}_3/\mathsf{AC}} \cdot \to^*_{\mathcal{H}_2/\mathsf{AC}} t$.

We first consider (a). If $s \to_{\mathcal{H}_1/\mathsf{AC}} t$ is a root step using rule 1 then $H = \mathsf{i}(\mathsf{h})$ and $H' = \mathsf{h}$. If $s \to_{\mathcal{H}_1/\mathsf{AC}} t$ is a root step using rule 2 then $H =_{\mathsf{AC}} \mathsf{i}(\mathsf{h} \mid u)$ and $H' =_{\mathsf{AC}} \mathsf{i}(u)$ for some term u. Next we consider (b). We have $s \to^*_{\mathcal{H}_2/\mathsf{AC}} s' \to_{\mathcal{H}_3/\mathsf{AC}} t' \to^*_{\mathcal{H}_2/\mathsf{AC}} t$ for some s' and t'. From Lemma 13 we obtain $s, s', t', t \in U$. Hence

$$H = \pi(s) =_{\mathsf{AC}} \pi(s') \to_{\mathcal{R}_n/\mathsf{AC}} \pi(t') =_{\mathsf{AC}} \pi(t) = H'$$

is obtained by Lemma 16 and thus also $H \to_{\mathcal{R}_n/\mathsf{AC}} H'$. ◀

4 Many-Sorted Semantic Labeling modulo AC

The mutual dependence between the function symbols A and B in rules 3 and 14 of $\mathcal{H}$ makes proving termination of $\mathcal{H}/\mathsf{AC}$ a non-trivial task. We use the technique of semantic labeling (Zantema [26]) to resolve the dependence by labeling both A and B by the ordinal value of the Hydra encoded in their second arguments. Semantic labeling for rewriting modulo has been investigated in [19]. We need, however, a version for many-sorted rewriting since the distinction between ordinals and natural numbers is essential for the effectiveness of semantic labeling for $\mathcal{H}/\mathsf{AC}$.

Before introducing semantic labeling, we recall some basic semantic definitions. An *algebra* $\mathcal{A}$ for an $\mathcal{S}$-sorted signature $\mathcal{F}$ is a pair $(\{S_\mathcal{A}\}_{S\in\mathcal{S}}, \{f_\mathcal{A}\}_{f\in\mathcal{F}})$, where each $S_\mathcal{A}$ is a non-empty set, called the *carrier of sort* S, and each $f_\mathcal{A}$ is a function of type $f : (S_1)_\mathcal{A} \times \cdots \times (S_n)_\mathcal{A} \to S_\mathcal{A}$, called the *interpretation function* of $f : S_1 \times \cdots \times S_n \to S$. A mapping that associates each variable of sort S to an element in $S_\mathcal{A}$ is called an *assignment*. We write $\mathcal{A}^\mathcal{V}$ for the set of all assignments. Given an assignment $\alpha \in \mathcal{A}^\mathcal{V}$, the *interpretation* of a term t is inductively defined as follows:

$$[\alpha]_\mathcal{A}(t) = \begin{cases} \alpha(t) & \text{if } t \text{ is a variable} \\ f_\mathcal{A}([\alpha]_\mathcal{A}(t_1), \ldots, [\alpha]_\mathcal{A}(t_n)) & \text{if } t = f(t_1, \ldots, t_n) \end{cases}$$

Let $\mathcal{A} = (\{S_\mathcal{A}\}_{S\in\mathcal{S}}, \{f_\mathcal{A}\}_{f\in\mathcal{F}})$ be an $\mathcal{S}$-sorted $\mathcal{F}$-algebra. We assume that each carrier set $S_\mathcal{A}$ is equipped with a well-founded order $>_S$ such that the interpretation functions are weakly monotone in all argument positions, and call $(\mathcal{A}, \{>_S\}_{S\in\mathcal{S}})$ a weakly monotone many-sorted algebra. Given terms s and t of sort S, we write $s \geqslant_\mathcal{A} t$ ($s =_\mathcal{A} t$) if $[\alpha]_\mathcal{A}(s) \geqslant_S [\alpha]_\mathcal{A}(t)$ ($[\alpha]_\mathcal{A}(s) =_S [\alpha]_\mathcal{A}(t)$) holds for all $\alpha \in \mathcal{A}^\mathcal{V}$.

A labeling L for $\mathcal{F}$ consists of sets of labels $L_f \subseteq S_\mathcal{A}$ for every $f : S_1 \times \cdots \times S_n \to S$. The labeled signature $\mathcal{F}_\mathsf{lab}$ consists of function symbols $f_a : S_1 \times \cdots \times S_n \to S$ for every function symbol $f : S_1 \times \cdots \times S_n \to S$ in $\mathcal{F}$ and label $a \in L_f$ together with all function symbols $f \in \mathcal{F}$ such that $L_f = \varnothing$. A *labeling* (L, lab) for $(\mathcal{A}, \{>_S\}_{S\in\mathcal{S}})$ consists of a labeling L for the signature $\mathcal{F}$ together with a mapping $\mathsf{lab}_f \colon (S_1)_A \times \cdots \times (S_n)_A \to L_f$ for every function symbol $f : S_1 \times \cdots \times S_n \to S$ in $\mathcal{F}$ with $L_f \neq \varnothing$. We call (L, lab) *weakly monotone* if all its labeling functions lab_f are weakly monotone in all coordinates. The mapping lab_f determines the label of the root symbol f of a term $f(t_1, \ldots, t_n)$, based on the values of its arguments $t_1, \ldots, t_n$. Formally, for every assignment $\alpha \in \mathcal{A}^\mathcal{V}$ we define a mapping lab_α inductively as follows:

$$\mathsf{lab}_\alpha(t) = \begin{cases} t & \text{if } t \in \mathcal{V} \\ f(\mathsf{lab}_\alpha(t_1), \dots, \mathsf{lab}_\alpha(t_n)) & \text{if } t = f(t_1, \dots, t_n) \text{ and } L_f = \varnothing \\ f_a(\mathsf{lab}_\alpha(t_1), \dots, \mathsf{lab}_\alpha(t_n)) & \text{if } t = f(t_1, \dots, t_n) \text{ and } L_f \neq \varnothing \end{cases}$$

where a denotes the label $\mathsf{lab}_f([\alpha]_{\mathcal{A}}(t_1), \dots, [\alpha]_{\mathcal{A}}(t_n))$. Note that $\mathsf{lab}_\alpha(t)$ and t have the same sort. Given a TRS $\mathcal{R}$ over a (many-sorted) signature $\mathcal{F}$, we define the *labeled* TRS $\mathcal{R}_{\mathsf{lab}}$ over the signature $\mathcal{F}_{\mathsf{lab}}$ as follows:

$$\mathcal{R}_{\mathsf{lab}} = \{\, \mathsf{lab}_\alpha(\ell) \to \mathsf{lab}_\alpha(r) \mid \ell \to r \in \mathcal{R} \text{ and } \alpha \in A^{\mathcal{V}} \,\}$$

Since the AC symbol | in the encoding of the Hydra battle is a constructor, there is no need to label it. Hence we assume for simplicity that $L_f = \varnothing$ for every AC symbol $f \in \mathcal{F}$. The TRS $\mathcal{D}\mathsf{ec}$ consists of all rewrite rules

$$f_a(x_1, \dots, x_n) \to f_b(x_1, \dots, x_n)$$

with $f : S_1 \times \cdots \times S_n \to S$ a function symbol in $\mathcal{F}$, $a, b \in L_f$ such that $a >_S b$, and pairwise different variables $x_1, \dots, x_n$. A weakly monotone algebra $(\mathcal{A}, >)$ is a *quasi-model* of $\mathcal{R}/\mathsf{AC}$ if $\ell \geqslant_{\mathcal{A}} r$ for all rewrite rules $\ell \to r$ in $\mathcal{R}$ and $\ell =_{\mathcal{A}} r$ for all equations $\ell \approx r$ in AC.

▶ **Theorem 17.** *Let $\mathcal{R}/\mathsf{AC}$ be a* TRS *over a many-sorted signature $\mathcal{F}$, $(\mathcal{A}, \{>_S\}_{S \in \mathcal{S}})$ a quasi-model of $\mathcal{R}/\mathsf{AC}$ with a weakly monotone labeling (L, lab). If $(\mathcal{R}_{\mathsf{lab}} \cup \mathcal{D}\mathsf{ec})/\mathsf{AC}$ is terminating then $\mathcal{R}/\mathsf{AC}$ is terminating.*

Proof. We show

1. if $t \to_{\mathcal{R}} u$ then $\mathsf{lab}_\alpha(t) \to^*_{\mathcal{D}\mathsf{ec}} \cdot \to_{\mathcal{R}_{\mathsf{lab}}} \mathsf{lab}_\alpha(u)$
2. if $t =_{\mathsf{AC}} u$ then $\mathsf{lab}_\alpha(t) =_{\mathsf{AC}} \mathsf{lab}_\alpha(u)$

for all sorts S, terms $t, u \in \mathcal{T}_S(\mathcal{F}, \mathcal{V})$, and assignments $\alpha \in \mathcal{A}^{\mathcal{V}}$. First suppose $t \to_{\mathcal{R}} u$ is a root step using the rewrite rule $\ell \to r$. So $t = \ell\sigma$ and $u = r\sigma$ for some substitution σ. Define the assignment $\beta = [\alpha]_{\mathcal{A}} \circ \sigma$ and the (labeled) substitution $\tau = \mathsf{lab}_\alpha \circ \sigma$. An easy induction proof yields $\mathsf{lab}_\alpha(s\sigma) = \mathsf{lab}_\beta(s)\tau$ for all terms s. By definition $\mathsf{lab}_\beta(\ell) \to \mathsf{lab}_\beta(r) \in \mathcal{R}_{\mathsf{lab}}$. Hence $\mathsf{lab}_\alpha(t) = \mathsf{lab}_\beta(\ell)\tau \to_{\mathcal{R}_{\mathsf{lab}}} \mathsf{lab}_\beta(r)\tau = \mathsf{lab}_\alpha(u)$. Next suppose $t \to_{\mathcal{R}} u$ takes place below the root. So $t = f(t_1, \dots, t_i, \dots t_n)$ and $u = f(t_1, \dots, u_i, \dots t_n)$ with $t_i \to_{\mathcal{R}} u_i$. Let $S_1 \times \cdots \times S_n \to S$ be the sort declaration of f. The induction hypothesis yields $\mathsf{lab}_\alpha(t_i) \to^*_{\mathcal{D}\mathsf{ec}} \cdot \to_{\mathcal{R}_{\mathsf{lab}}} \mathsf{lab}_\alpha(u_i)$. We obtain $[\alpha]_{\mathcal{A}}(t_i) \geqslant_{S_i} [\alpha]_{\mathcal{A}}(u_i)$ from the quasi-model assumption. If $L_f = \varnothing$ then

$$\begin{aligned} \mathsf{lab}_\alpha(t) = f(\mathsf{lab}_\alpha(t_1), \dots, \mathsf{lab}_\alpha(t_i), \dots, \mathsf{lab}_\alpha(t_n)) &\to^*_{\mathcal{D}\mathsf{ec}} \cdot \to_{\mathcal{R}_{\mathsf{lab}}} \\ f(\mathsf{lab}_\alpha(t_1), \dots, \mathsf{lab}_\alpha(u_i), \dots, \mathsf{lab}_\alpha(t_n)) &= \mathsf{lab}_\alpha(u) \end{aligned}$$

Suppose $L_f \neq \varnothing$ and let

$$\begin{aligned} a &= \mathsf{lab}_f([\alpha]_{\mathcal{A}}(t_1), \dots, [\alpha]_{\mathcal{A}}(t_i), \dots, [\alpha]_{\mathcal{A}}(t_n)) \\ b &= \mathsf{lab}_f([\alpha]_{\mathcal{A}}(t_1), \dots, [\alpha]_{\mathcal{A}}(u_i), \dots, [\alpha]_{\mathcal{A}}(t_n)) \end{aligned}$$

We obtain $a \geqslant_S b$ from the weak monotonicity of the labeling function lab_f. Therefore, the following rewrite sequence is constructed:

$$\begin{aligned} \mathsf{lab}_\alpha(t) = f_a(\mathsf{lab}_\alpha(t_1), \dots, \mathsf{lab}_\alpha(t_i), \dots, \mathsf{lab}_\alpha(t_n)) &\to^*_{\mathcal{D}\mathsf{ec}} \\ f_b(\mathsf{lab}_\alpha(t_1), \dots, \mathsf{lab}_\alpha(t_i), \dots, \mathsf{lab}_\alpha(t_n)) &\to^*_{\mathcal{D}\mathsf{ec}} \cdot \to_{\mathcal{R}_{\mathsf{lab}}} \\ f_b(\mathsf{lab}_\alpha(t_1), \dots, \mathsf{lab}_\alpha(u_i), \dots, \mathsf{lab}_\alpha(t_n)) &= \mathsf{lab}_\alpha(u) \end{aligned}$$

This concludes the proof of the first statement. For the second statement we use induction on the number of applications of AC axioms in $t =_{\mathsf{AC}} u$. If this number is one, the conclusion is reached by reasoning as above (with $L_f = \varnothing$ because AC symbols are not labeled and hence the rules of $\mathcal{D}\mathsf{ec}$ do not come into play). ◀

After these preliminaries, we are ready to put many-sorted semantic labeling to the test. Consider the many-sorted algebra $\mathcal{A}$ with carriers $\mathbb{N}$ for sort N and $\mathbb{O}$, the set of ordinal numbers smaller than ϵ_0, for sorts O and S and the following interpretation functions:

$$\begin{aligned} 0_{\mathcal{A}} = \mathsf{h}_{\mathcal{A}} = 1 \qquad & \mathsf{s}_{\mathcal{A}}(n) = n + 1 \qquad & \mathsf{i}_{\mathcal{A}}(x) = \omega^x \\ x \mid_{\mathcal{A}} y = x \oplus y \qquad & \mathsf{E}_{\mathcal{A}}(x) = x + 1 \qquad & \mathsf{C}_{\mathcal{A}}(n, x) = x \cdot n + 1 \\ \mathsf{A}_{\mathcal{A}}(n, x) = \mathsf{B}_{\mathcal{A}}(n, x) = \mathsf{D}_{\mathcal{A}}(n, x) = x & & \end{aligned}$$

Here $\oplus$ denotes natural addition on ordinals, which is strictly monotone in both arguments.

▶ **Lemma 18.** *The algebra $(\mathcal{A}, \{>_{\mathsf{O}}, >_{\mathsf{N}}\})$ is a quasi-model of $\mathcal{H}/\mathsf{AC}$.*

Proof. First note that the interpretation functions are weakly monotone. The rewrite rules in $\mathcal{H}$ are oriented by $\geqslant_{\mathsf{O}}$:

$$\begin{aligned}
\mathsf{A}_{\mathcal{A}}(n, \mathsf{i}_{\mathcal{A}}(\mathsf{h}_{\mathcal{A}})) = \omega &>_{\mathsf{O}} 1 = \mathsf{A}_{\mathcal{A}}(\mathsf{s}_{\mathcal{A}}(n), \mathsf{h}_{\mathcal{A}}) && (1)\\
\mathsf{A}_{\mathcal{A}}(n, \mathsf{i}_{\mathcal{A}}(\mathsf{h}_{\mathcal{A}} \mid_{\mathcal{A}} x)) = \omega^{x+1} &>_{\mathsf{O}} \omega^x = \mathsf{A}_{\mathcal{A}}(\mathsf{s}_{\mathcal{A}}(n), \mathsf{i}_{\mathcal{A}}(x)) && (2)\\
\mathsf{A}_{\mathcal{A}}(n, \mathsf{i}_{\mathcal{A}}(x)) = \omega^x &=_{\mathsf{O}} \omega^x = \mathsf{B}_{\mathcal{A}}(n, \mathsf{D}_{\mathcal{A}}(\mathsf{s}_{\mathcal{A}}(n), \mathsf{i}_{\mathcal{A}}(x))) && (3)\\
\mathsf{C}_{\mathcal{A}}(0_{\mathcal{A}}, x) = x + 1 &=_{\mathsf{O}} x + 1 = \mathsf{E}_{\mathcal{A}}(x) && (4)\\
\mathsf{C}_{\mathcal{A}}(\mathsf{s}_{\mathcal{A}}(n), x) = x \cdot n + x + 1 &=_{\mathsf{O}} x \cdot n + x + 1 = x \mid_{\mathcal{A}} \mathsf{C}_{\mathcal{A}}(n, x) && (5)\\
\mathsf{i}_{\mathcal{A}}(\mathsf{E}_{\mathcal{A}}(x) \mid_{\mathcal{A}} y) = \omega^{x \oplus y + 1} &>_{\mathsf{O}} \omega^{x \oplus y} + 1 = \mathsf{E}_{\mathcal{A}}(\mathsf{i}_{\mathcal{A}}(x \mid_{\mathcal{A}} y)) && (6)\\
\mathsf{i}_{\mathcal{A}}(\mathsf{E}_{\mathcal{A}}(x)) = \omega^{x+1} &>_{\mathsf{O}} \omega^x + 1 = \mathsf{E}_{\mathcal{A}}(\mathsf{i}_{\mathcal{A}}(x)) && (7)\\
\mathsf{D}_{\mathcal{A}}(n, \mathsf{i}_{\mathcal{A}}(\mathsf{i}_{\mathcal{A}}(x))) = \omega^{\omega^x} &=_{\mathsf{O}} \omega^{\omega^x} = \mathsf{i}_{\mathcal{A}}(\mathsf{D}_{\mathcal{A}}(n, \mathsf{i}_{\mathcal{A}}(x))) && (8)\\
\mathsf{D}_{\mathcal{A}}(n, \mathsf{i}_{\mathcal{A}}(\mathsf{i}_{\mathcal{A}}(x) \mid_{\mathcal{A}} y)) = \omega^{\omega^x \oplus y} &=_{\mathsf{O}} \omega^{\omega^x \oplus y} = \mathsf{i}_{\mathcal{A}}(\mathsf{D}_{\mathcal{A}}(n, \mathsf{i}_{\mathcal{A}}(x)) \mid_{\mathcal{A}} y) && (9)\\
\mathsf{D}_{\mathcal{A}}(n, \mathsf{i}_{\mathcal{A}}(\mathsf{i}_{\mathcal{A}}(\mathsf{h}_{\mathcal{A}} \mid_{\mathcal{A}} x) \mid_{\mathcal{A}} y)) = \omega^{\omega^{x+1} \oplus y} &>_{\mathsf{O}} \omega^{\omega^x \cdot n \oplus y + 1} = \mathsf{i}_{\mathcal{A}}(\mathsf{C}_{\mathcal{A}}(n, \mathsf{i}_{\mathcal{A}}(x)) \mid_{\mathcal{A}} y) && (10)\\
\mathsf{D}_{\mathcal{A}}(n, \mathsf{i}_{\mathcal{A}}(\mathsf{i}_{\mathcal{A}}(\mathsf{h}_{\mathcal{A}} \mid_{\mathcal{A}} x))) = \omega^{\omega^{x+1}} &>_{\mathsf{O}} \omega^{\omega^x \cdot n + 1} = \mathsf{i}_{\mathcal{A}}(\mathsf{C}_{\mathcal{A}}(n, \mathsf{i}_{\mathcal{A}}(x))) && (11)\\
\mathsf{D}_{\mathcal{A}}(n, \mathsf{i}_{\mathcal{A}}(\mathsf{i}_{\mathcal{A}}(\mathsf{h}_{\mathcal{A}}) \mid_{\mathcal{A}} y)) = \omega^{\omega \oplus y} &>_{\mathsf{O}} \omega^{(n+1) \oplus y} = \mathsf{i}_{\mathcal{A}}(\mathsf{C}_{\mathcal{A}}(n, \mathsf{h}_{\mathcal{A}}) \mid_{\mathcal{A}} y) && (12)\\
\mathsf{D}_{\mathcal{A}}(n, \mathsf{i}_{\mathcal{A}}(\mathsf{i}_{\mathcal{A}}(\mathsf{h}_{\mathcal{A}}))) = \omega^{\omega} &>_{\mathsf{O}} \omega^{n+1} = \mathsf{i}_{\mathcal{A}}(\mathsf{C}_{\mathcal{A}}(n, \mathsf{h}_{\mathcal{A}})) && (13)\\
\mathsf{B}_{\mathcal{A}}(n, \mathsf{E}_{\mathcal{A}}(x)) = x + 1 &>_{\mathsf{O}} x = \mathsf{A}_{\mathcal{A}}(\mathsf{s}_{\mathcal{A}}(n), x) && (14)
\end{aligned}$$

Note that inequalities (10) – (13) use the fact that $\omega >_{\mathsf{O}} n$ holds for $n \in \mathbb{N}$. The compatibility of $\mathcal{A}$ with AC follows from the associativity and the commutativity of $\oplus$:

$$\begin{aligned} (x \mid_{\mathcal{A}} y) \mid_{\mathcal{A}} z = (x \oplus y) \oplus z &=_{\mathsf{O}} x \oplus (y \oplus z) = x \mid_{\mathcal{A}} (y \mid_{\mathcal{A}} z) \\ x \mid_{\mathcal{A}} y = x \oplus y &=_{\mathsf{O}} y \oplus x = x \mid_{\mathcal{A}} y \end{aligned}$$

Therefore, $\mathcal{A}$ is a quasi-model of $\mathcal{H}/\mathsf{AC}$. ◀

We now label A and B by the value of their second argument. Let $L_{\mathsf{A}} = L_{\mathsf{B}} = \mathbb{O}$ and $L_f = \varnothing$ for the other function symbols f, and define lab as follows:

$$\mathsf{lab}_{\mathsf{A}}(n, x) = \mathsf{lab}_{\mathsf{B}}(n, x) = x$$

The labeling (L, lab) results in the infinite rewrite system $\mathcal{H}_{\mathsf{lab}} \cup \mathcal{D}\mathsf{ec}$ with $\mathcal{H}_{\mathsf{lab}}$ consisting of the rewrite rules

$$\begin{aligned}
\mathsf{A}_{\omega}(n, \mathsf{i}(\mathsf{h})) &\xrightarrow{1} \mathsf{A}_1(\mathsf{s}(n), \mathsf{h}) & \mathsf{D}(n, \mathsf{i}(\mathsf{i}(x))) &\xrightarrow{8} \mathsf{i}(\mathsf{D}(n, \mathsf{i}(x))) \\
\mathsf{A}_{\omega^{v+1}}(n, \mathsf{i}(\mathsf{h} \mid x)) &\xrightarrow{2} \mathsf{A}_{\omega^v}(\mathsf{s}(n), \mathsf{i}(x)) & \mathsf{D}(n, \mathsf{i}(\mathsf{i}(x) \mid y)) &\xrightarrow{9} \mathsf{i}(\mathsf{D}(n, \mathsf{i}(x))) \mid y) \\
\mathsf{A}_{\omega^v}(n, \mathsf{i}(x)) &\xrightarrow{3} \mathsf{B}_{\omega^v}(n, \mathsf{D}(\mathsf{s}(n), \mathsf{i}(x))) & \mathsf{D}(n, \mathsf{i}(\mathsf{i}(\mathsf{h} \mid x) \mid y)) &\xrightarrow{10} \mathsf{i}(\mathsf{C}(n, \mathsf{i}(x)) \mid y) \\
\mathsf{C}(0, x) &\xrightarrow{4} \mathsf{E}(x) & \mathsf{D}(n, \mathsf{i}(\mathsf{i}(\mathsf{h} \mid x))) &\xrightarrow{11} \mathsf{i}(\mathsf{C}(n, \mathsf{i}(x))) \\
\mathsf{C}(\mathsf{s}(n), x) &\xrightarrow{5} x \mid \mathsf{C}(n, x) & \mathsf{D}(n, \mathsf{i}(\mathsf{i}(\mathsf{h}) \mid y)) &\xrightarrow{12} \mathsf{i}(\mathsf{C}(n, \mathsf{h}) \mid y) \\
\mathsf{i}(\mathsf{E}(x) \mid y) &\xrightarrow{6} \mathsf{E}(\mathsf{i}(x \mid y)) & \mathsf{D}(n, \mathsf{i}(\mathsf{i}(\mathsf{h}))) &\xrightarrow{13} \mathsf{i}(\mathsf{C}(n, \mathsf{h})) \\
\mathsf{i}(\mathsf{E}(x)) &\xrightarrow{7} \mathsf{E}(\mathsf{i}(x)) & \mathsf{B}_{v+1}(n, \mathsf{E}(x)) &\xrightarrow{14} \mathsf{A}_v(\mathsf{s}(n), x)
\end{aligned}$$

for all $v \in \mathbb{O}$ and $\mathcal{D}\mathsf{ec}$ consisting of the rewrite rules

$$\mathsf{A}_v(n, x) \to \mathsf{A}_w(n, x) \qquad \mathsf{B}_v(n, x) \to \mathsf{B}_w(n, x)$$

for all $v, w \in \mathbb{O}$ with $v > w$. According to Theorem 17, the AC termination of $\mathcal{H}$ on many-sorted terms follows from the AC termination of $\mathcal{H}_{\mathsf{lab}} \cup \mathcal{D}\mathsf{ec}$.

▶ **Corollary 19.** *If $\mathcal{H}_{\mathsf{lab}} \cup \mathcal{D}\mathsf{ec}$ is* AC *terminating then $\mathcal{H}$ is* AC *terminating on sorted terms.*

5 AC-RPO

In order to show AC termination of $\mathcal{H}_{\mathsf{lab}} \cup \mathcal{D}\mathsf{ec}$ we use the AC version of recursive path orders (AC-RPO), introduced by Rubio [20]. In this section we first recall the definition of AC-RPO, following the presentation in [23]. AC-RPO is a relation on terms constructed from a strict order $>$ on function symbols, called *precedence*. AC-RPO collects the arguments of successive occurrences of the same AC symbols in a multiset. This operation is captured by top-flattening. Let $\mathcal{F}_{\mathsf{AC}}$ be the set of AC symbols in $\mathcal{F}$. The *top-flattening* of a term t with respect to $f \in \mathcal{F}_{\mathsf{AC}}$ is the multiset $\triangledown_f(t)$ defined inductively as follows:

$$\triangledown_f(t) = \begin{cases} \triangledown_f(t_1) \uplus \triangledown_f(t_2) & \text{if } t = f(t_1, t_2) \\ \{t\} & \text{otherwise} \end{cases}$$

Multisets resulting from top-flattening are first compared by an embedding-like relation. Let t be a term with $\mathsf{root}(t) = f \in \mathcal{F}_{\mathsf{AC}}$ and $\triangledown_f(t) = \{t_1, \dots, t_n\}$. We write $t \rhd^f_{\mathsf{emb}} u$ for all terms u such that $\triangledown_f(u) = \{t_1, \dots, t_{i-1}, s_j, t_{i+1}, \dots, t_n\}$ for some $t_i = g(s_1, \dots, s_m)$ with $g \not> f$ and $1 \leqslant j \leqslant m$. Then terms in the multisets are grouped according to their forms, and compared in a sophisticated way. For this sake we define the following submultisets of a multiset T of terms:

$$T\restriction_{\mathcal{V}} = \{x \in T \mid x \in \mathcal{V}\} \qquad \begin{aligned} T\restriction^{>}_f &= \{g(t_1, \dots, t_n) \in T \setminus \mathcal{V} \mid g > f\} \\ T\restriction^{\not<}_f &= \{g(t_1, \dots, t_n) \in T \setminus \mathcal{V} \mid g \not< f\} \end{aligned}$$

Let $T = \{t_1, \dots, t_n\}$. By $\#(T)$ we denote the linear polynomial $\#(t_1) + \cdots + \#(t_n)$. Here $\#(t)$ is defined as follows:

$$\#(t) = \begin{cases} t & \text{if } t \text{ is a variable} \\ 1 & \text{otherwise} \end{cases}$$

Note that $\#(S) > \#(T)$ and $\#(S) \geqslant \#(T)$ denote comparisons of integer polynomials. After these preliminaries, we are ready to present the definition of AC-RPO.

▶ **Definition 20.** *Let $>$ be a precedence and let $\mathcal{F} \setminus \mathcal{F}_{\mathsf{AC}} = \mathcal{F}_{\mathsf{mul}} \uplus \mathcal{F}_{\mathsf{lex}}$. We define $>_{\mathsf{acrpo}}$ inductively as follows: $s >_{\mathsf{acrpo}} t$ if one of the following conditions holds:*

1. *$s = f(s_1, \dots, s_n)$ and $s_i \geqslant_{\mathsf{acrpo}} t$ for some $1 \leqslant i \leqslant n$,*
2. *$s = f(s_1, \dots, s_n)$, $t = g(t_1, \dots, t_m)$, $f > g$, and $s >_{\mathsf{acrpo}} t_j$ for all $1 \leqslant j \leqslant m$,*
3. *$s = f(s_1, \dots, s_n)$, $t = f(t_1, \dots, t_n)$, $f \notin \mathcal{F}_{\mathsf{AC}}$, $s >_{\mathsf{acrpo}} t_j$ for all $1 \leqslant j \leqslant n$, and either*
   (a) *$f \in \mathcal{F}_{\mathsf{lex}}$ and $(s_1, \dots, s_n) >^{\mathsf{lex}}_{\mathsf{acrpo}} (t_1, \dots, t_n)$, or*
   (b) *$f \in \mathcal{F}_{\mathsf{mul}}$ and $\{s_1, \dots, s_n\} >^{\mathsf{mul}}_{\mathsf{acrpo}} \{t_1, \dots, t_n\}$,*
4. *$s = f(s_1, s_2)$, $t = f(t_1, t_2)$, $f \in \mathcal{F}_{\mathsf{AC}}$, and $s' \geqslant_{\mathsf{acrpo}} t$ for some s' such that $s \rhd^f_{\mathsf{emb}} s'$,*
5. *$s = f(s_1, s_2)$, $t = f(t_1, t_2)$, $f \in \mathcal{F}_{\mathsf{AC}}$, $s >_{\mathsf{acrpo}} t'$ for all t' such that $t \rhd^f_{\mathsf{emb}} t'$, $S{\restriction}^{\not>}_f \uplus S{\restriction}_{\mathcal{V}} \geqslant^{\mathsf{mul}}_{\mathsf{acrpo},f} T{\restriction}^{\not>}_f \uplus T{\restriction}_{\mathcal{V}}$ for $S = \triangledown_f(s)$ and $T = \triangledown_f(t)$, and*
   (a) *$S{\restriction}^{>}_f >^{\mathsf{mul}}_{\mathsf{acrpo}} T{\restriction}^{>}_f$, or*
   (b) *$\#(S) > \#(T)$, or*
   (c) *$\#(S) \geqslant \#(T)$, and $S >^{\mathsf{mul}}_{\mathsf{acrpo}} T$.*

Here $s >_{\mathsf{acrpo},f} t$ means $s >_{\mathsf{acrpo}} t$ and if $\mathsf{root}(s) \not> f$ then $\mathsf{root}(s) \geqslant \mathsf{root}(t)$. The relation $=_{\mathsf{AC}}$ is used as preorder in $>^{\mathsf{lex}}_{\mathsf{acrpo}}$, $>^{\mathsf{mul}}_{\mathsf{acrpo}}$, and $\geqslant^{\mathsf{mul}}_{\mathsf{acrpo},f}$ as equivalence relation in $\geqslant_{\mathsf{acrpo}}$.

▶ **Theorem 21** ([20], Theorem 4)**.** *The relation $>_{\mathsf{acrpo}}$ is a* AC*-compatible rewrite order with the subterm property.*

As a consequence, $>_{\mathsf{acrpo}}$ is an AC-compatible reduction order when the underlying signature is finite. As noted in [20, Section 8.2], this also holds for infinite signatures, provided the precedence $>$ is well-founded. This is important because the signature of $\mathcal{H}_{\mathsf{lab}}$ is infinite. Below, we will formally prove the correctness of the extension, by adopting the approach of [16].

A strict order $>$ on a set A is a *partial well-order* if for every infinite sequence $a_0, a_1, \dots$ of elements in A there exist indices i and j such that $i < j$ and $a_i \leqslant a_j$. Well-founded total orders (*well-orders*) are partial well-orders. Given a partial well-order $>$ on $\mathcal{F}$, the *embedding* TRS $\mathcal{E}\mathsf{mb}(\mathcal{F}, >)$ consists of the rules $f(x_1, \dots, x_n) \to x_i$ for every n-ary function symbol and $1 \leqslant i \leqslant n$, together with the rules $f(x_1, \dots, x_n) \to g(x_{i_1}, \dots, x_{i_m})$ for all function symbols f and g with arities m and n such that $f > g$, and indices $1 \leqslant i_1 < i_2 < \cdots < i_m \leqslant n$. Here $x_1, \dots, x_n$ are pairwise distinct variables.

▶ **Theorem 22** ([16], Theorem 5.3)**.** *A rewrite order $\succ$ is well-founded if $\mathcal{E}\mathsf{mb}(\mathcal{F}, >) \subseteq \succ$ for some partial well-order $>$.*

▶ **Theorem 23.** *The relation $>_{\mathsf{acrpo}}$ is an* AC*-compatible reduction order for every well-founded precedence $>$.*

Proof. Let $>$ be a well-founded precedence. We only need to show well-foundedness of $>_{\mathsf{acrpo}}$ because the other properties follow by Theorem 21. Let $\sqsupset$ be an arbitrary well-order that contains $>$. Trivially, $\sqsupset$ is a partial well-order. The inclusion $\mathcal{E}\mathsf{mb}(\mathcal{F}, \sqsupset) \subseteq \sqsupset_{\mathsf{acrpo}}$ is easily verified. Hence the well-foundedness of $\sqsupset_{\mathsf{acrpo}}$ is obtained from Theorem 22. Since $> \subseteq \sqsupset$, the *incrementality* of AC-RPO [20, Lemma 22] yields $>_{\mathsf{acrpo}} \subseteq \sqsupset_{\mathsf{acrpo}}$. It follows that $>_{\mathsf{acrpo}}$ is well-founded. ◀

We show the termination of $\mathcal{H}_{\mathsf{lab}} \cup \mathcal{D}\mathsf{ec}$ by AC-RPO. To this end, we consider the following precedence $>$ on the labeled signature:

$$\begin{aligned}
\mathsf{A}_v &> \mathsf{A}_w && \text{for all } v, w \in \mathbb{O} \text{ with } v > w\\
\mathsf{B}_v &> \mathsf{B}_w && \text{for all } v, w \in \mathbb{O} \text{ with } v > w\\
\mathsf{B}_{v+1} &> \mathsf{A}_v > \mathsf{B}_v && \text{for all } v \in \mathbb{O}\\
\mathsf{B}_0 &> \mathsf{D} > \mathsf{C} > \mathsf{i} > \mathsf{E} > \mathsf{s} > |
\end{aligned}$$

Note that $>$ is well-founded.

▶ **Theorem 24.** $\mathcal{H}_{\mathsf{lab}} \cup \mathcal{D}\mathsf{ec} \subseteq >_{\mathsf{acrpo}}$

Proof. We show the compatibility verification for rules 3 and 6 of $\mathcal{H}_{\mathsf{lab}}$. The other rewrite rules are handled in a similar fashion. For rule 3 we have (the numbers next to the inference steps refer to the cases in Definition 20)

$$\dfrac{\mathsf{A}_v > \mathsf{B}_v \quad \dfrac{n \geqslant_{\mathsf{acrpo}} n}{\ell_3 >_{\mathsf{acrpo}} n}\,2 \quad \dfrac{\mathsf{A}_{\omega^v} > \mathsf{D} \quad \dfrac{\dfrac{n \geqslant_{\mathsf{acrpo}} n}{\ell_3 >_{\mathsf{acrpo}} n}\,1 \quad \mathsf{A}_{\omega^v} > \mathsf{s}}{\ell_3 >_{\mathsf{acrpo}} \mathsf{s}(n)}\,2 \quad \dfrac{\mathsf{i}(x) \geqslant_{\mathsf{acrpo}} \mathsf{i}(x)}{\ell_3 >_{\mathsf{acrpo}} \mathsf{i}(x)}\,1}{\ell_3 >_{\mathsf{acrpo}} \mathsf{D}(\mathsf{s}(n), \mathsf{i}(x))}\,2}{\ell_3 = \mathsf{A}_{\omega^v}(n, \mathsf{i}(x)) >_{\mathsf{acrpo}} \mathsf{B}_{\omega^v}(n, \mathsf{D}(\mathsf{s}(n), \mathsf{i}(x)))}$$

For rule 6 we have

$$\dfrac{\mathsf{i} > \mathsf{E} \quad \dfrac{\dfrac{\dfrac{\dfrac{x \geqslant_{\mathsf{acrpo}} x}{\mathsf{E}(x) >_{\mathsf{acrpo}} x}\,1}{\{\mathsf{E}(x), y\} \geqslant^{\mathsf{mul}}_{\mathsf{acrpo}} \{x, y\}} \quad \{\mathsf{E}(x)\} >^{\mathsf{mul}}_{\mathsf{acrpo}} \varnothing}\,5(\mathrm{a}) \quad \dfrac{\mathsf{E}(x) \mid y >_{\mathsf{acrpo}} x \mid y}{\mathsf{i}(\mathsf{E}(x) \mid y) >_{\mathsf{acrpo}} \mathsf{i}(x \mid y)}\,3(\mathrm{a})}{\mathsf{i}(\mathsf{E}(x) \mid y) >_{\mathsf{acrpo}} \mathsf{E}(\mathsf{i}(x \mid y))}\,2$$

The multiset comparisons in 5(a) correspond to $S\restriction_f^{\not\geqslant} \uplus S\restriction_{\mathcal{V}} \geqslant^{\mathsf{mul}}_{\mathsf{acrpo},f} T\restriction_f^{\not\geqslant} \uplus T\restriction_{\mathcal{V}}$ and $S\restriction_f^{>} >^{\mathsf{mul}}_{\mathsf{acrpo}} T\restriction_f^{>}$ for $f = |$, $S = \triangledown_f(\mathsf{E}(x) \mid y)$, and $T = \triangledown_f(x \mid y)$. These multisets are calculated as follows:

$$\begin{array}{llll}
S = \{\mathsf{E}(x), y\} & S\restriction_f^{>} = S\restriction_f^{\not\geqslant} = \{\mathsf{E}(x)\} & S\restriction_{\mathcal{V}} = \{y\} & S\restriction_f^{\not\geqslant} \uplus S\restriction_{\mathcal{V}} = \{\mathsf{E}(x), y\}\\
T = \{x, y\} & T\restriction_f^{>} = T\restriction_f^{\not\geqslant} = \varnothing & T\restriction_{\mathcal{V}} = \{x, y\} & T\restriction_f^{\not\geqslant} \uplus T\restriction_{\mathcal{V}} = \{x, y\}
\end{array}$$ ◀

From Theorems 4 and 24 we conclude that Hercules eventually beats Hydra in any battle. Theorems 24 and 6 in connection with Corollary 19 yield the AC termination of $\mathcal{H}$ on arbitrary terms.

6 Related Work

In an influential survey paper, Dershowitz and Jouannaud [4, p. 270] introduced a 5-rule rewrite system to simulate the Hydra Battle. The proposed rewrite system was later shown to be erroneous. A corrected version together with a detailed termination analysis has been given by Dershowitz and Moser [5], see also Moser [17]. Earlier, Touzet [21] presented an 11-rule rewrite system that encodes a specific battle with weakened Hydras (whose height is bounded by 4) and proved total termination by a semantic termination method. It is worth noting that our rewrite system $\mathcal{H}$ is not even simply terminating on unsorted terms. In fact, we have the following cyclic sequence with respect to $\mathcal{H} \cup \mathcal{E}\mathsf{mb}(\mathcal{F}, \varnothing)$:

$$\begin{aligned}\mathsf{A}(\mathsf{E}(\mathsf{i}(x)),\mathsf{i}(x)) &\xrightarrow{3} \mathsf{B}(\mathsf{E}(\mathsf{i}(x)),\mathsf{D}(\mathsf{s}(\mathsf{E}(\mathsf{i}(x))),\mathsf{i}(x))) \to^*_{\mathcal{E}\mathsf{mb}(\mathcal{F},\varnothing)} \mathsf{B}(\mathsf{E}(\mathsf{i}(x)),\mathsf{i}(x))\\ &\xrightarrow{14} \mathsf{A}(\mathsf{s}(\mathsf{E}(\mathsf{i}(x))),\mathsf{i}(x)) \to_{\mathcal{E}\mathsf{mb}(\mathcal{F},\varnothing)} \mathsf{A}(\mathsf{E}(\mathsf{i}(x)),\mathsf{i}(x))\end{aligned}$$

So the TRS $\mathcal{H}$ is not simply terminating (see [16, Lemma 4.6]).

The rewrite systems referred to above model the so-called *standard* battle, which corresponds to a specific strategy for Hercules. In this regard it is interesting to quote Kirby and Paris [11], who introduced the battle as an accessible example of an independence result for Peano arithmetic (P):

> A *strategy* is a function which determines for Hercules which head to chop off at each stage of any battle. It is not hard to find a reasonably fast *winning strategy* (i.e. a strategy which ensures that Hercules wins against any hydra). More surprisingly, Hercules cannot help winning:
>
> Theorem 2. (i) *Every strategy is a winning strategy.*
>
> [...]
>
> Theorem 2. (ii) *The statement "every recursive strategy is a winning strategy" is not provable from* P.

In a recent paper [6, Section 6], rules are presented to slay Hydras, independent of the strategy. These rules do not constitute a term rewrite system in the usual sense (they operate on terms with *sequence variables*). More importantly, the infinitely many rules do not faithfully represent the battle. Earlier, Ferreira and Zantema [7, Section 10] presented an infinite rewrite system to model the standard strategy and gave a direct ordinal interpretation to conclude its termination. In neither of the latter two papers stages of the battle are modeled.

7 Conclusion

We presented a new TRS encoding of the Battle of Hydra and Hercules. Unlike earlier encodings, it makes use of AC symbols. This allows to faithfully model any strategy of Hercules, as envisaged in the paper by Kirby and Paris [11] in which the Battle was first presented. To prove the termination of the encoding we employed many-sorted rewriting modulo AC and we extended semantic labeling modulo AC to many-sorted TRSs. The infinite TRS produced by semantic labelling was proved terminating by suitably instantiating AC-RPO.

The finite TRS $\mathcal{H}$ poses an interesting challenge for automatic termination tools. None of the tools (AProVE [8], muterm [1], NaTT [22]) competing in the "TRS Equational" category of last year's Termination Competition[1] succeeds on $\mathcal{H}/\mathsf{AC}$. This is not really surprising since most methods implemented in termination tool come with a multiple recursive upper bound on the derivation height (e.g. [10, 13, 18]). The tools even fail to prove termination of $\mathcal{H}$ without AC. The tool T$\top$T$_2$ [12] has support for ordinal interpretations [24] but also fails on $\mathcal{H}$.

Formalizing the techniques used in this paper in a proof assistant is an important task to ensure the correctness of the results. Interestingly, the informal paper [9] in which we announced our encoding also presents a termination proof, essentially extending a semantic

[1] https://termcomp.github.io/Y2022/

method of Touzet [21] and Zantema [27] to AC rewriting. Although we believe the non-trivial extension to be correct, its use in proving the AC termination of $\mathcal{H}$ has a critical mistake, which we recently discovered.

Another topic for future research is to investigate the scope of many-sorted semantic labeling. Can the termination of earlier encodings of the battle be established with many-sorted semantic labeling followed by some standard simplification order? Variants of the battle by Buchholz [2] and Lepper [14] are also of interest here.

References

1 Beatriz Alarcón, Raúl Gutiérrez, Salvador Lucas, and Rafael Navarro-Marset. Proving termination properties with MU-TERM. In *Proc. 13th Algebraic Methodology and Software Technology*, volume 6486 of *Lecture Notes in Computer Science*, pages 201–208, 2011. doi:10.1007/978-3-642-17796-5_12.

2 Wilfried Buchholz. An independence result for $(\pi^1_1 - \mathrm{CA}) + \mathrm{BI}$. *Annals of Pure and Applied Logic*, 33:131–155, 1987. doi:10.1016/0168-0072(87)90078-9.

3 Wilfried Buchholz. Another rewrite system for the standard Hydra battle. In *Proc. Mini-Workshop: Logic, Combinatorics and Independence Results*, volume 3(4) of *Oberwolfach Reports*, pages 3099–3102. European Mathematical Society, 2006.

4 Nachum Dershowitz and Jean-Pierre Jouannaud. *Rewrite Systems. Handbook of Theoretical Computer Science*, volume B: Formal Models and Semantics, pages 243–320. Elsevier, 1990.

5 Nachum Dershowitz and Georg Moser. The Hydra battle revisited. In *Rewriting, Computation and Proof, Essays Dedicated to Jean-Pierre Jouannaud on the Occasion of his 60th Birthday*, volume 4600 of *Lecture Notes in Computer Science*, pages 1–27, 2007. doi:10.1007/978-3-540-73147-4_1.

6 Jörg Endrullis, Jan Willem Klop, and Roy Overbeek. Star games and Hydras. *Logical Methods in Computer Science*, 17(2):20:1–20:32, 2021. doi:10.23638/LMCS-17(2:20)2021.

7 Maria C. F. Ferreira and Hans Zantema. Total termination of term rewriting. *Applicable Algebra in Engineering, Communication and Computing*, 7(2):133–162, 1996. doi:10.1007/BF01191381.

8 Jürgen Giesl, Cornelius Aschermann, Marc Brockschmidt, Fabian Emmes, Florian Frohn, Carsten Fuhs, Jera Hensel, Carsten Otto, Martin Plücker, Peter Schneider-Kamp, Thomas Ströder, Stephanie Swiderski, and René Thiemann. Analyzing program termination and complexity automatically with AProVE. *Journal of Automated Reasoning*, 58(1):3–31, 2017. doi:10.1007/s10817-016-9388-y.

9 Nao Hirokawa and Aart Middeldorp. Hydra battles and AC termination. In *Proc. 18th International Workshop on Termination*, pages 21–25, 2022.

10 Dieter Hofbauer. Termination proofs by multiset path orderings imply primitive recursive derivation lengths. *Theoretical Computer Science*, 105(1):129–140, 1992. doi:10.1016/0304-3975(92)90289-R.

11 Laurence Kirby and Jeff Paris. Accessible independency results for Peano arithmetic. *Bulletin of the London Mathematical Society*, 14:285–325, 1982. doi:10.1112/blms/14.4.285.

12 Martin Korp, Christian Sternagel, Harald Zankl, and Aart Middeldorp. Tyrolean Termination Tool 2. In *Proc. 20th International Conference on Rewriting Techniques and Applications*, volume 5595 of *Lecture Notes in Computer Science*, pages 295–304, 2009. doi:10.1007/978-3-642-02348-4_21.

13 Ingo Lepper. Derivation lengths and order types of Knuth–Bendix orders. *Theoretical Computer Science*, 269(1-2):433–450, 2001. doi:10.1016/S0304-3975(01)00015-9.

14 Ingo Lepper. Simply terminating rewrite systems with long derivations. *Archive for Mathematical Logic*, 43(1):1–18, 2004. doi:10.1007/s00153-003-0190-2.

15 Aart Middeldorp and Hitoshi Ohsaki. Type introduction for equational rewriting. *Acta Informatica*, 36(12):1007–1029, 2000. doi:10.1007/PL00013300.

16 Aart Middeldorp and Hans Zantema. Simple termination of rewrite systems. *Theoretical Computer Science*, 175(1):127–158, 1997. doi:10.1016/S0304-3975(96)00172-7.

17 Georg Moser. The Hydra battle and Cichon's principle. *Applicable Algebra in Engineering, Communication and Computing*, 20(2):133–158, 2009. doi:10.1007/s00200-009-0094-4.

18 Georg Moser and Andreas Schnabl. The derivational complexity induced by the dependency pair method. *Logical Methods in Computer Science*, 7(3), 2011. doi:10.2168/LMCS-7(3:1)2011.

19 Hitoshi Ohsaki, Aart Middeldorp, and Jürgen Giesl. Equational termination by semantic labelling. In *Proc. 14th EACSL Annual Conference on Computer Science Logic*, volume 1862 of *Lecture Notes in Computer Science*, pages 457–471, 2000. doi:10.1007/3-540-44622-2_31.

20 Albert Rubio. A fully syntactic AC-RPO. *Information and Computation*, 178(2):515–533, 2002. doi:10.1006/inco.2002.3158.

21 Hélène Touzet. Encoding the Hydra battle as a rewrite system. In *Proc. 23rd International Symposium on Mathematical Foundations of Computer Science*, volume 1450 of *Lecture Notes in Computer Science*, pages 267–276, 1998. doi:10.1007/BFb0055776.

22 Akihisa Yamada, Keiichirou Kusakari, and Toshiki Sakabe. Nagoya termination tool. In *Proc. 25th International Conference on Rewriting Techniques and Applications and 12th International Conference on Typed Lambda Calculi and Applications*, volume 8560 of *Lecture Notes in Computer Science*, pages 466–475, 2014. doi:10.1007/978-3-319-08918-8_32.

23 Akihisa Yamada, Sarah Winkler, Nao Hirokawa, and Aart Middeldorp. AC-KBO revisited. *Theory and Practice of Logic Programming*, 16(2):163–188, 2016. doi:10.1017/S1471068415000083.

24 Harald Zankl, Sarah Winkler, and Aart Middeldorp. Beyond polynomials and Peano arithmetic—Automation of elementary and ordinal interpretations. *Journal of Symbolic Computation*, 69:129–158, 2015. doi:10.1016/j.jsc.2014.09.033.

25 Hans Zantema. Termination of term rewriting: Interpretation and type elimination. *Journal of Symbolic Computation*, 17(1):23–50, 1994. doi:10.1006/jsco.1994.1003.

26 Hans Zantema. Termination of term rewriting by semantic labelling. *Fundamenta Informaticae*, 24:89–105, 1995. doi:10.3233/FI-1995-24124.

27 Hans Zantema. The termination hierarchy for term rewriting. *Applicable Algebra in Engineering, Communication and Computing*, 12(1-2):3–19, 2001. doi:10.1007/s002000100061.