ON TOOLS FOR COMPLETENESS
OF KLEENE ALGEBRA WITH HYPOTHESES

DAMIEN POUS ●[a], JURRIAAN ROT ●[b], AND JANA WAGEMAKER ●[c]

[a] CNRS, LIP, ENS de Lyon
  e-mail address: Damien.Pous@ens-lyon.fr

[b] Radboud University, Nijmegen
  e-mail address: Jurriaan.Rot@ru.nl

[c] Reykjavik University, Reykjavik
  e-mail address: janaw@ru.is

Abstract. In the literature on Kleene algebra, a number of variants have been proposed which impose additional structure specified by a theory, such as Kleene algebra with tests (KAT) and the recent Kleene algebra with observations (KAO), or make specific assumptions about certain constants, as for instance in NetKAT. Many of these variants fit within the unifying perspective offered by *Kleene algebra with hypotheses*, which comes with a canonical language model constructed from a given set of hypotheses. For the case of KAT, this model corresponds to the familiar interpretation of expressions as languages of guarded strings.

A relevant question therefore is whether Kleene algebra together with a given set of hypotheses is complete with respect to its canonical language model. In this paper, we revisit, combine and extend existing results on this question to obtain tools for proving completeness in a modular way.

We showcase these tools by giving new and modular proofs of completeness for KAT, KAO and NetKAT, and we prove completeness for new variants of KAT: KAT extended with a constant for the full relation, KAT extended with a converse operation, and a version of KAT where the collection of tests only forms a distributive lattice.

1. INTRODUCTION

Kleene algebras (KA) [Kle56, Con71] are algebraic structures involving an iteration operation, Kleene star, corresponding to reflexive-transitive closure in relational models and to language iteration in language models. Its axioms are complete w.r.t. relational models and language models [Koz91, Kro91, Bof90], and the resulting equational theory is decidable via automata algorithms (in fact, PSpace-complete [MS72, HRS76]).

These structures were later extended in order to deal with common programming constructs. For instance, Kleene algebras with tests (KAT) [Koz97], which combine Kleene algebra and Boolean algebra, make it possible to represent the control flow of while programs. Kleene star is used for while loops, and Boolean tests are used for the conditions of such loops, as well as the conditions in if-then-else statements. Again, the axioms of KAT are complete w.r.t. appropriate classes of models, and its equational theory remains in PSPACE. Proving so is non-trivial: Kozen's proof reduces completeness of KAT to completeness of KA, via a direct syntactic transformation on terms.

Another extension is Concurrent Kleene algebra (CKA) [HMSW11], where a binary operator for parallelism is added. The resulting theory is characterised by languages of pomsets rather than languages of words, and is EXPSPACE-complete [BPS17]. Trying to have both tests and concurrency turned out to be non-trivial, and called for yet another notion: Kleene algebras with observations (KAO) [KBR$^+$19], which are again complete w.r.t. appropriate models, and decidable.

When used in the context of program verification, e.g., in a proof assistant, such structures make it possible to write algebraic proofs of correctness, and to mechanise some of the steps: when two expressions e and f representing two programs happen to be provably equivalent in KA, KAT, or KAO, one does not need to provide a proof, one can simply call a certified decision procedure [BP10, KN12, Pou13]. However, this is often not enough [KP00, AK01, HK02]: most of the time, the expressions e and f are provably equal only under certain assumptions on their constituents. For instance, to prove that $(a + b)^*$ and a^*b^* are equal, one may have to use the additional assumption $ba = ab$. In other words, one would like to prove equations under some assumptions, to have algorithms for the Horn theory of Kleene algebra and its extensions rather than just their equational theories.

Unfortunately, those Horn theories are typically undecidable [Koz96, Koz02], even with rather restricted forms of hypotheses (e.g., commutation of specific pairs of letters, as in the above example). Nevertheless, important and useful classes of hypotheses can be 'eliminated', by reducing to the plain and decidable case of the equational theory. This is, for instance, the case for *Hoare hypotheses* [Koz00], of the shape $e = 0$, which allow to encode Hoare triples for partial correctness in KAT.

In some cases, one wants to exploit hypotheses about specific constituents (e.g, a and b in the above example). In other situations, one wants to exploit assumptions on the whole structure. For instance, in commutative Kleene algebra [Red64, Con71, Bru19], one assumes that the product is commutative everywhere.

Many of these extensions of Kleene algebra (KAT, KAO, commutative KA, specific hypotheses) fit into the generic framework of Kleene algebra with hypotheses [DKPP19], providing in each case a canonical model in terms of closed languages.

We show that we recover standard models in this way, and we provide tools to establish completeness and decidability of such extensions, in a modular way. The key notion is that of *reduction* from one set of hypotheses to another. We summarise existing reductions, provide new ones, and design a toolbox for combining those reductions together. We use this toolbox in order to obtain new and modular proofs of completeness for KAT, KAO and NetKAT [AFG$^+$14]; to prove completeness of KAT with a full element and KAT with a converse operation, two extensions that were formerly proposed for KA alone, respectively in [PW22] and [BÉS95, ÉB95]; and to prove completeness for the variant of KAT where tests are only assumed to form a distributive lattice.

Outline. We first recall Kleene algebra, Kleene algebra with hypotheses, and the key concept of reduction (Section 2). Then we design a first toolbox for reductions (Section 3): a collection of primitive reductions, together with lemmas allowing one to combine them together in a modular fashion. We exemplify those tools by using them to give a new proof of completeness for KAT (Section 4).

After this first example, we develop more systematic tools for reductions (Section 5): lemmas to combine more than two reductions at a time, and lemmas to prove the resulting side-conditions more efficiently. We finally demonstrate the flexibility of our approach in a series of examples of increasing complexity: KAO (Section 6), KAT with a full element (Section 7), KAT with converse (Section 8), KA with positive tests (KAPT—Section 9), and NetKAT (Section 10).

The first appendix contains an abstract theory of least closures in complete lattices, which we base our toolbox on (Appendix A). Two subsequent appendices contain proofs omitted from the main text (a direct soundness proof for Kleene algebra with hypotheses— Appendix B, and proofs related to guarded strings for KAT—Appendix C).

Differences with the conference version. This paper is a revised and extended version of an earlier paper in the proceedings of RaMiCS 2021 [PRW21]. Besides various improvements, and inclusion of most proofs, here are the main differences w.r.t. the conference version.

Section 3.3 is new, it provides a general technique to construct reductions via finite automata. We use it to establish some of the basic reductions from Section 3.4.

With the exception of Proposition 5.1, Section 5 on advanced tools is new (e.g., Proposition 5.3 or development on overlaps). This section makes it possible to present most examples using 'tables' (Tables 1 to 5), which we consider as a new contribution concerning methodology. Lemma 3.10, which makes it possible to obtain reductions for hypotheses of the shape $e \leq 1$ is also new; we need it to deal with NetKAT.

The examples of KAT with a full element, KAT with converse, and NetKAT, are new (Sections 7, 8 and 10); the example of KAPT (Section 9) is presented more uniformly, thanks to the new tools. Appendix A on the abstract theory of least closures is entirely new; it makes it possible to get rid of all proofs by transfinite induction that we used in the conference version. Appendix C was publicly available via the author version of [PRW21], but not officially published.

Preliminaries, notation. We write 1 for the empty word and uv for the concatenation of two words u, v. Given a set X, subsets of X form a complete lattice $\langle \mathcal{P}(X), \subseteq, \bigcup \rangle$. Given two functions f, g, we write $f \circ g$ for their composition: $(f \circ g)(x) = f(g(x))$. We write id for the identity function. Functions into a complete lattice, ordered pointwise, form a complete lattice where suprema are computed pointwise. We reuse the same notations on such lattices: for instance, given two functions $f, g \colon Y \to \mathcal{P}(X)$, we write $f \subseteq g$ when $\forall y \in Y, f(y) \subseteq g(y)$, or $f \cup g$ for the function mapping $y \in Y$ to $f(y) \cup g(y)$.

A function $f \colon \mathcal{P}(X) \to \mathcal{P}(X)$ is *monotone* if for all $S, S' \in \mathcal{P}(X)$, $S \subseteq S'$ implies $f(S) \subseteq f(S')$; it is a *closure* if it is monotone and satisfies id $\subseteq f$ and $f \circ f \subseteq f$. When f is a closure, we have $f \circ f = f$, and for all S, S', $S \subseteq f(S')$ iff $f(S) \subseteq f(S')$. These notions are generalised to arbitrary complete lattices in Appendix A.

2. Kleene algebra with hypotheses, closures, reductions

A *Kleene algebra* [Con71, Koz91] is a tuple $(K, +, \cdot, {}^*, 0, 1)$ such that $(K, +, \cdot, 0, 1)$ is an idempotent semiring, and * is a unary operator on K such that for all $x, y \in K$ the following axioms are satisfied:

$$1 + x \cdot x^* \leq x^* \qquad x + y \cdot z \leq z \Rightarrow y^* \cdot x \leq z \qquad x + y \cdot z \leq y \Rightarrow x \cdot z^* \leq y$$

Here, as later in the paper, we write $x \leq y$ as a shorthand for $x + y = y$. Given the idempotent semiring axioms, $\leq$ is a partial order in every Kleene algebra, and all operations are monotone w.r.t. that order. The seemingly missing law $1 + x^* \cdot x \leq x^*$ is derivable from these axioms.

Languages over some alphabet, as well as binary relations on a given set, are standard examples of Kleene algebras.

We let e, f range over regular expressions over an alphabet Σ, defined by:

$$e, f ::= e + f \mid e \cdot f \mid e^* \mid 0 \mid 1 \mid a \in \Sigma$$

We write $\mathcal{T}(\Sigma)$ for the set of such expressions, or simply $\mathcal{T}$ when the alphabet is clear from the context. We usually write ef for $e \cdot f$, and e^+ for $e \cdot e^*$. Given alphabets Σ and Γ, a function $h \colon \Sigma \to \mathcal{T}(\Gamma)$ extends uniquely to a homomorphism $h \colon \mathcal{T}(\Sigma) \to \mathcal{T}(\Gamma)$, referred to as the *homomorphism generated by h*. As usual, every regular expression e gives rise to a language $[\![e]\!] \in \mathcal{P}(\Sigma^*)$. Given regular expressions e, f, we write $\mathsf{KA} \vdash e = f$ when $e = f$ is derivable from the axioms of Kleene algebra. (Equivalently, when the equation $e = f$ holds universally, in all Kleene algebras.)

The central theorem of Kleene algebra is the following:

Theorem 2.1 (Soundness and Completeness of KA [Koz91, Kro91, Bof90])**.** *For all regular expressions $e, f \in \mathcal{T}$, we have:*

$$\mathsf{KA} \vdash e = f \quad \textit{if and only if} \quad [\![e]\!] = [\![f]\!] \ .$$

As a consequence, the equational theory of Kleene algebras is decidable.

2.1. **Hypotheses.** Our goal is to extend this result to the case where we have additional hypotheses on some of the letters of the alphabet, or axioms restricting the behaviour of certain operations. Those are represented by sets of *inequations*, i.e., pairs (e, f) of regular expressions written $e \leq f$ for the sake of clarity. Given a set H of such inequations, we write $\mathsf{KA}_H \vdash e \leq f$ when the inequation $e \leq f$ is derivable from the axioms of Kleene algebra and the hypotheses in H (similarly for equations). By extension, we write $\mathsf{KA}_H \vdash H'$ when $\mathsf{KA}_H \vdash e \leq f$ for all $e \leq f$ in H'. Since the ambient theory will systematically be KA, we will sometimes abbreviate $\mathsf{KA}_H \vdash H'$ as $H \vdash H'$, to alleviate notation.

Note that we consider letters of the alphabet as constants rather than variables. In particular, while we have $\mathsf{KA}_{ba \leq ab} \vdash (a+b)^* \leq a^* b^*$, we do not have $\mathsf{KA}_{ba \leq ab} \vdash (a+c)^* \leq a^* c^*$. Formally, we use a notion of derivation where there is no substitution rule, and where we have all instances of Kleene algebra axioms as axioms. When we want to consider hypotheses that are universally valid, it suffices to use all their instances. For example, to define commutative Kleene algebra, we simply use the infinite set $\{ef \leq fe \mid e, f \in \mathcal{T}\}$.

2.2. **Closures associated to hypotheses.** We associate a canonical language model to Kleene algebra with a set of hypotheses H, defined by *closure* under H [DKPP19].

For $u, v \in \Sigma^*$ and $L \subseteq \Sigma^*$, let $uLv \triangleq \{uxv \mid x \in L\}$.

Definition 2.2 (H-closure). Let H be a set of hypotheses and let $L \subseteq \Sigma^*$ be a language. The *H-closure of L*, denoted as $H^\star(L)$, is the smallest language containing L such that for all $e \leq f \in H$ and $u, v \in \Sigma^*$, if $u\,\llbracket f \rrbracket\, v \subseteq H^\star(L)$, then $u\,\llbracket e \rrbracket\, v \subseteq H^\star(L)$.

For every set of hypotheses H, $H^\star$ is indeed a closure (also see Section 3.1).

In many cases, H will consist of inequations whose members are words rather than arbitrary expressions. In that case, there is an intuitive string rewriting characterisation of the function $H^\star$. Given words u, v, we write $u \leftsquigarrow_H v$ when u can be obtained from v by replacing an occurrence as a subword of the right-hand side of an inequation of H by its left-hand side. Further write $\leftsquigarrow_H^*$ for the reflexive-transitive closure of $\leftsquigarrow_H$. Then we have $u \in H^\star(L)$ iff $u \leftsquigarrow_H^* v$ for some word $v \in L$.

Example 2.3. Let $H = \{ab \leq bc\}$. We have $baab \leftsquigarrow_H babc \leftsquigarrow_H bbcc$, whence $baab \in H^\star(\{bbcc\})$. For $L = \llbracket bc^* \rrbracket$, we have $H^\star(L) = \llbracket a^* bc^* \rrbracket$. $\qquad\square$

This notion of closure gives a *closed interpretation* of regular expressions, $H^\star \llbracket - \rrbracket$, for which KA_H is sound:

Theorem 2.4. $\mathsf{KA}_H \vdash e = f$ *implies* $H^\star \llbracket e \rrbracket = H^\star \llbracket f \rrbracket$.

Proof. This is basically [DKPP19, Theorem 2], which is proved there by constructing a model of KA_H. We provide a direct proof in Appendix B, by induction on the derivation. $\qquad\square$

In the sequel, we shall prove the converse implication, completeness, for specific choices of H: we say that KA_H is *complete* if for all expressions e, f:

$$H^\star \llbracket e \rrbracket = H^\star \llbracket f \rrbracket \quad \text{implies} \quad \mathsf{KA}_H \vdash e = f \,.$$

Remark 2.5. *Thanks to the presence of sum in the syntax of regular expressions (so that $\mathsf{KA}_H \vdash e \leq f$ iff $\mathsf{KA}_H \vdash e + f = f$), and to the fact that $H^\star$ is a closure, we have the following counterpart to Theorem 2.4 for soundness w.r.t inequations:*

$$\mathsf{KA}_H \vdash e \leq f \quad \text{implies} \quad \llbracket e \rrbracket \subseteq H^\star \llbracket f \rrbracket \,.$$

Similarly, KA_H is complete if and only if for all expressions e, f,

$$\llbracket e \rrbracket \subseteq H^\star \llbracket f \rrbracket \quad \text{implies} \quad \mathsf{KA}_H \vdash e \leq f \,. \qquad\square$$

We could hope that completeness always holds, notably because the notion of closure is invariant under inter-derivability of the considered hypotheses, as a consequence of the following lemma:

Lemma 2.6 [KBS+20, Lemma 4.10]. *Let H and H' be sets of hypotheses. If $H \vdash H'$ then $H'^\star \subseteq H^\star$.*

(There, $H'^\star \subseteq H^\star$ means pointwise inclusion of functions, i.e., for all L, $H'^\star(L) \subseteq H^\star(L)$.)

Another property witnessing the canonicity of the H-closed language interpretation is that it is sound and complete w.r.t. *$*$-continuous* models [DKPP19, Theorem 2].

Unfortunately, there are concrete instances for which KA_H is known to be incomplete. For instance, there is a finitely presented monoid (thus a finite set H_0 of equations) such that $\{(e, f) \mid H_0^\star \llbracket e \rrbracket = H_0^\star \llbracket f \rrbracket\}$ is not recursively enumerable [KM14, Theorem 1]. Since derivability in KA_H is recursively enumerable as soon as H is, KA_{H_0} cannot be complete.

For a more concrete counter-example, consider $H_1 \triangleq \{apb \leq p, \ u \leq a, u \leq p, u \leq b\}$, where a, p, b, u are four distinct letters. The closed language $H_1{}^\star[\![p]\!]$ contains all words of the shape $a^n p b^n$ for some $n \in \mathbb{N}$ thanks to the first inequation in H_1, and thus all words of the shape u^{2n+1} for some $n \in \mathbb{N}$ thanks to the other three inequations. It follows that $(uu)^* u \subseteq H_1{}^\star[\![p]\!]$. However, $(uu)^* u \leq p$ is not derivable in KA_{H_1}. The intuition is that the star induction rules in KA do not make it possible to reach the middle of a word. Formally, the action lattice provided by Kuznetsov in [Kuz18, Section 3] gives a counter-model: consider his Lemma 3.6 and interpret p as p, a as p/q, b as $p\backslash q$, and u as $p \wedge q \wedge a \wedge b$.

Before turning to techniques for proving completeness, let us describe the closed interpretation of regular expressions for two specific choices of hypotheses.

Example 2.7. Let us first consider *commutative Kleene algebra*, obtained using the set $\{ef \leq fe \mid e, f \in \mathcal{T}(\Sigma)\}$, as also mentioned in the introduction. Under Kleene algebra axioms, this set is equiderivable with its restriction to letters, $C = \{ab \leq ba \mid a, b \in \Sigma\}$ (a consequence of [AK01, Lemma 4.4]). The associated closure can be characterised as follows:

$$C^\star(L) = \{w \in \Sigma^* \mid \exists v \in L, \ |w|_x = |v|_x \text{ for all } x \in \Sigma\}$$

where $|w|_x$ denotes the number of occurrences of x in w. Thus, $w \in C^\star(L)$ if it is a permutation of some word in L.

This semantics matches the one used in [Con71] for commutative Kleene algebra: there, a function $[\![-]\!]_c : \mathcal{T}(\Sigma) \to \mathcal{P}(\mathbb{N}^\Sigma)$ interprets regular expressions as subsets of $\mathbb{N}^\Sigma$, whose elements are thought of as "commutative words": these assign to each letter the number of occurrences, but there is no order of letters. Let $q \colon \mathcal{P}(\Sigma^*) \to \mathcal{P}(\mathbb{N}^\Sigma)$, $q(L) = \{\lambda x.|w|_x \mid w \in L\}$; this map computes the *Parikh image* of a given language L, that is, the set of multisets representing occurrences of letters in words in L. Then the semantics is characterised by $[\![-]\!]_c = q \circ [\![-]\!]$.

One may observe that $[\![-]\!]_c = q \circ C^\star[\![-]\!]$, since $C^\star$ only adds words to a language which have the same number of occurrences of each letter as some word which is already there. Conversely, we have $C^\star[\![-]\!] = q' \circ [\![-]\!]_c$, where $q' \colon \mathcal{P}(\mathbb{N}^\Sigma) \to \mathcal{P}(\Sigma^*)$, $q'(L) = \{w \mid p \in L, \ \forall x \in \Sigma, |w|_x = p(x)\}$. As a consequence, we have $[\![e]\!]_c = [\![f]\!]_c$ iff $C^\star[\![e]\!] = C^\star[\![f]\!]$.

From there, we can easily deduce from the completeness result in [Con71, Chapter 11, Theorem 4], attributed to Pilling (see also [Bru19], which mentions the history of this result including the earlier proof by Redko [Red64]), that KA_C is complete.

Example 2.8. Let us now consider a single hypothesis: $D = \{ab \leq 0\}$ for some letters a and b. The D-closure of a language L consists of those words that either belong to L, or contain ab as a subword. As a consequence, we have $D^\star[\![e]\!] = D^\star[\![f]\!]$ if and only if $[\![e]\!]$ and $[\![f]\!]$ agree on all words not containing the pattern ab.

In this example, we can easily obtain decidability and completeness of KA_D. Indeed, consider the function $r \colon \mathcal{T}(\Sigma) \to \mathcal{T}(\Sigma)$, $r(e) \triangleq e + \Sigma^* ab \Sigma^*$. For all e, we have $\mathsf{KA}_D \vdash e = r(e)$, and $D^\star[\![e]\!] = [\![r(e)]\!]$. As a consequence, we have

$$D^\star[\![e]\!] = D^\star[\![f]\!]$$
$$\Leftrightarrow \qquad [\![r(e)]\!] = [\![r(f)]\!]$$
$$\Leftrightarrow \qquad \mathsf{KA} \vdash r(e) = r(f) \qquad\qquad \text{(Theorem 2.1)}$$
$$\Rightarrow \qquad \mathsf{KA}_D \vdash e = f$$

The first step above establishes decidability of the closed semantics; the following ones reduce the problem of completeness for KA_D to that for KA alone (which holds). By soundness (Theorem 2.4), the last line implies the first one, so that these conditions are all equivalent.

This second example exploits and illustrate a simple instance of the framework we design in the sequel to prove completeness of Kleene algebra with various sets of hypotheses.

2.3. **Reductions.** As illustrated above, the overall strategy is to reduce completeness of KA_H, for a given set of hypotheses H, to completeness of Kleene algebra without hypotheses. The core idea is to provide a map r from expressions to expressions, which incorporates the hypotheses H in the sense that $[\![r(e)]\!] = H^\star[\![e]\!]$, and such that $r(e)$ is provably equivalent to e under the hypotheses H. This idea leads to the unifying notion of *reduction*, developed in [KM14, DKPP19, KBS+20].

Definition 2.9 (Reduction). Assume $\Gamma \subseteq \Sigma$ and let H, H' be sets of hypotheses over Σ and Γ respectively. We say that H *reduces to* H' if

(1) $\mathsf{KA}_H \vdash H'$, and

there exists a map $r \colon \mathcal{T}(\Sigma) \to \mathcal{T}(\Gamma)$ such that for all $e \in \mathcal{T}(\Sigma)$,

(2) $\mathsf{KA}_H \vdash e = r(e)$, and
(3) $H^\star[\![e]\!] \cap \Gamma^* \subseteq H'^\star[\![r(e)]\!]$.

We often refer to the witnessing map r itself as a reduction. Note that condition (1) vanishes when $H' = \emptyset$ is empty. Thanks to the first two conditions, the third condition can be strengthened into an equality:

Lemma 2.10. *If r is a reduction from H to H' as in the above definition, then for all expressions e we have $H'^\star[\![r(e)]\!] = H^\star[\![e]\!] \cap \Gamma^*$.*

Proof. By item 3, it suffices to show $H'^\star[\![r(e)]\!] \subseteq H^\star[\![e]\!]$. We have

$$H'^\star[\![r(e)]\!] \subseteq H'^\star[\![H^\star[\![r(e)]\!]]\!]$$
$$= H'^\star[\![H^\star[\![e]\!]]\!] \qquad\qquad \text{(item 2 with Theorem 2.4)}$$
$$\subseteq H^\star[\![H^\star[\![e]\!]]\!] = H^\star[\![e]\!] \qquad\qquad \text{(item 1 with Lemma 2.6)} \quad \square$$

Generalising Example 2.8, we get the following key property of reductions:

Theorem 2.11. *Suppose H reduces to H'. If $\mathsf{KA}_{H'}$ is complete, then so is KA_H.*

Proof. Let r be the map for the reduction from H to H'. For all $e, f \in \mathcal{T}(\Sigma)$,

$$H^\star[\![e]\!] = H^\star[\![f]\!]$$
$$\Rightarrow \qquad H'^\star[\![r(e)]\!] = H'^\star[\![r(f)]\!] \qquad\qquad \text{(Lemma 2.10)}$$
$$\Rightarrow \qquad \mathsf{KA}_{H'} \vdash r(e) = r(f) \qquad\qquad \text{(completeness of } \mathsf{KA}_{H'})$$
$$\Rightarrow \qquad \mathsf{KA}_H \vdash r(e) = r(f) \qquad\qquad \text{(r a reduction, item 1)}$$
$$\Rightarrow \qquad \mathsf{KA}_H \vdash e = f \qquad\qquad \text{(r a reduction, item 2)} \quad \square$$

An important case is when $H' = \emptyset$: given a reduction from H to $\emptyset$, Theorem 2.11 gives completeness of KA_H, by completeness of KA. Such reductions to the empty set are what we ultimately aim for. However, in order to decompose reductions into small and elementary pieces, we need the extra generality allowed by our definition (e.g., so that reductions can be composed—Lemma 3.11).

While we focus on completeness in this paper, note that reductions can also be used to prove decidability:

Theorem 2.12. *If $\mathsf{KA}_{H'}$ is complete and decidable, and H reduces to H' via a computable function r, then KA_H is decidable.*

Proof. By soundness of KA_H (Theorem 2.4) and by inspecting the above proof of Theorem 2.11, we see that $\mathsf{KA}_H \vdash e = f$ iff $\mathsf{KA}_{H'} \vdash r(e) = r(f)$. ☐

Example 2.13. We consider KA together with a global "top element" $\mathbb{T}$ and the axiom $e \leq \mathbb{T}$. To make this precise in Kleene algebra with hypotheses, we assume an alphabet Σ with $\mathbb{T} \in \Sigma$, and take the set of hypotheses $H_{\mathbb{T}} = \{e \leq \mathbb{T} \mid e \in \mathcal{T}(\Sigma)\}$.

We claim that H reduces to $\emptyset$. The first condition is void since we target the empty set. For the two other conditions, define the unique homomorphism $r \colon \mathcal{T}(\Sigma) \to \mathcal{T}(\Sigma)$ such that $r(\mathbb{T}) = \Sigma^*$ (where we view Σ as the sum $a_1 + ... + a_n$ of all the letters in Σ) and $r(a) = a$ for $a \in \Sigma \setminus \{\mathbb{T}\}$.

Thanks to the hypotheses in H, we have $\mathsf{KA}_H \vdash \mathbb{T} = \Sigma^*$, from which we easily deduce the second condition ($\mathsf{KA}_H \vdash e = r(e)$ for all expressions e), by induction. Concerning the third condition, notice that the restriction to words over the smaller alphabet vanishes since the alphabet remains the same in that example. Then it suffices to observe that $H_{\mathbb{T}}^\star(L)$ contains those words obtained from a word $w \in L$ by replacing occurrences of $\mathbb{T}$ in w by arbitrary words in Σ^*.

This gives a first proof that $\mathsf{KA}_{H_{\mathbb{T}}}$ is complete, which we will revisit in Example 3.3. Note that this implies completeness w.r.t. validity of equations in all language models, where $\mathbb{T}$ is interpreted as the largest language: indeed, the closed semantics $H_{\mathbb{T}}^\star[\![-]\!]$ is generated by such a model.

Remark 2.14. *Reductions do not always exist. In Example 2.7, we discussed commutative KA as an instance of Kleene algebra with hypotheses H. While KA_C is complete in that case, there is no reduction from C to $\emptyset$, as $C^\star$ does not preserve regularity. Indeed, $C^\star[\![(ab)^*]\!] = \{w \mid |w|_a = |w|_b\}$ which is not regular. The completeness proof in [Con71, Bru19] is self-contained, and does not rely on completeness of KA.*

3. Tools for analysing closures and constructing reductions

To deal with concrete examples we still need tools to construct reductions. We provide such tools in this section. First, we give simpler conditions to obtain reductions in the special case where the underlying function is a homomorphism (Section 3.2). Second, we give a general technique to construct reductions via finite automata (Section 3.3). Third, we establish a few basic reductions for commonly encountered sets of hypotheses (Section 3.4). And fourth, we provide lemmas to compose reductions (Section 3.5).

Before doing so, we mention elementary properties of the functions $H^\star$ associated to sets H of hypotheses. Those properties will be useful first to obtain the aforementioned tools, and then later to deal with concrete examples.

3.1. **On closures w.r.t. sets of hypotheses.** Given a set H of hypotheses, the function $H^\star$ (Definition 2.2) actually is the least closure containing the following function:

$$H \colon \mathcal{P}(\Sigma^*) \to \mathcal{P}(\Sigma^*)$$
$$L \mapsto \bigcup \{u \, [\![e]\!] \, v \mid e \leq f \in H, \ u \, [\![f]\!] \, v \subseteq L\}$$

This function intuitively computes the words which are reachable from L in exactly one step. In particular, when H consists of inequations between words, we have $u \in H(L)$ iff $u \leftsquigarrow_H v$ for some $v \in L$. (Note that we slightly abuse notation by writing H both for a set of hypotheses and for the associated function.)

Thanks to the above characterisation, many properties of $H^\star$ can be obtained by analysing the simpler function H. To this end, we actually develop a general theory of least closures in Appendix A. We will state the relevant lemmas in the main text, delegating the corresponding proofs to that appendix.

Let us call a function between complete lattices *linear* when it preserves all joins, and *affine* when it preserves all non-empty joins (see Appendix A). Not all sets of hypotheses yield linear (or even affine) functions: consider for instance $\{1 \leq a + b\}$. However, an important class of hypotheses do:

Lemma 3.1. *If H is a set of hypotheses whose right-hand sides are words, then the functions H and $H^\star$ are linear.*

Proof. When H consists of hypotheses of the shape $e \leq w$ with w being a word, the definition of the function H simplifies to $H(L) = \bigcup \{u \llbracket e \rrbracket v \mid e \leq w \in H, \ uwv \in L\}$. We deduce

$$
H\left(\bigcup_{i \in I} L_i\right) = \bigcup \{u \llbracket e \rrbracket v \mid e \leq w \in H, \ uwv \in \bigcup_{i \in I} L_i\}
$$
$$
= \bigcup \{u \llbracket e \rrbracket v \mid e \leq w \in H, \ i \in I, \ uwv \in L_i\}
$$
$$
= \bigcup_{i \in I} \bigcup \{u \llbracket e \rrbracket v \mid e \leq w \in H, \ uwv \in L_i\} = \bigcup_{i \in I} H(L_i)
$$

Thus H is linear, and we deduce that so is $H^\star$ by Lemma A.10. $\qquad\square$

3.2. **Homomorphic reductions.** The following result from [KBS+20] (cf. Remark 3.5) simplifies the conditions of a reduction, assuming the underlying map r is a homomorphism.

Proposition 3.2. *Assume $\Gamma \subseteq \Sigma$ and let H, H' be sets of hypotheses over Σ and Γ respectively. If*

(1) $\mathsf{KA}_H \vdash H'$,

and there exists a homomorphism $r \colon \mathcal{T}(\Sigma) \to \mathcal{T}(\Gamma)$ such that:

(2) *for all $a \in \Sigma$, we have $\mathsf{KA}_H \vdash a = r(a)$,*
(3) *for all $a \in \Gamma$, we have $\mathsf{KA} \vdash a \leq r(a)$, and*
(4) *for all $e \leq f \in H$, we have $\mathsf{KA}_{H'} \vdash r(e) \leq r(f)$,*

then H reduces to H', with r being the witness of the reduction.

Conditions (3) and (4) are stated as derivability conditions, but this is only a matter of convenience: all we need are the corresponding inclusions of languages—closed by H' for (4).

We prove this proposition below; first we illustrate its usage by revisiting Example 2.13.

Example 3.3. The reduction we defined to deal with Example 2.13 is a homomorphism. Accordingly, we can use Proposition 3.2 to prove more easily that it is indeed a reduction from $H_{\mathbb{T}}$ to the empty set, without ever having to understand the closure $H_{\mathbb{T}}^\star$:

- the first condition is void since $H' = \emptyset$;
- the second and third condition hold trivially for all letters different from $\mathbb{T}$, on which r is the identity. For the letter $\mathbb{T}$, we have

$$\mathsf{KA} \vdash \mathbb{T} \le \Sigma^* = r(\mathbb{T}) \qquad\qquad (\text{because } \mathbb{T} \in \Sigma)$$

$$\mathsf{KA}_{H_{\mathbb{T}}} \vdash r(\mathbb{T}) \le \mathbb{T} \qquad\qquad (\text{by a single application of } H_{\mathbb{T}})$$

- the fourth condition holds by completeness of KA: $\mathsf{KA} \vdash r(e) \le \Sigma^* = r(\mathbb{T})$ for all expressions e since any language is contained in Σ^*, by definition. (Note that using completeness of KA is an overkill here.) □

We now turn to proving Proposition 3.2. We need the following auxiliary definition: for a map $r\colon \mathcal{T}(\Sigma) \to \mathcal{T}(\Gamma)$, define

$$\dot{r}\colon \mathcal{P}(\Sigma^*) \to \mathcal{P}(\Gamma^*)$$

$$L \mapsto \bigcup_{w \in L} [\![r(w)]\!] \ .$$

(Here words w are seen as regular expressions when fed to r.) The function $\dot{r}$ is linear. When r is a homomorphism, we have:

Fact 3.4. *If r is a homomorphism then $\dot{r}$ is a homomorphism and for all expressions $e \in \mathcal{T}(\Sigma)$ we have $\dot{r}([\![e]\!]) = [\![r(e)]\!]$.*

Proof of Proposition 3.2. First, using that r is a homomorphism, we extend by induction the assumptions (2) and (3) into:

(2') for all $e \in \mathcal{T}(\Sigma)$, $\mathsf{KA}_H \vdash e = r(e)$

(3') for all $e \in \mathcal{T}(\Gamma)$, $\mathsf{KA} \vdash e \le r(e)$

From (3') we deduce

(3") for all $L \subseteq \Gamma^*$, $L \subseteq \dot{r}(L)$

(Note the restriction to words in Γ^*.) Indeed, by soundness of KA and (3'), we have $[\![e]\!] \subseteq [\![r(e)]\!]$ for all $e \in \mathcal{T}(\Gamma)$, and in particular $w \in [\![r(w)]\!]$ for all words $w \in \Gamma^*$.

At this point we only have the third condition of a reduction left to show, i.e., that for all $e \in \mathcal{T}(\Sigma)$, $H^\star[\![e]\!] \cap \Gamma^* \subseteq H'^\star[\![r(e)]\!]$. We proceed as follows:

$$H^\star[\![e]\!] \cap \Gamma^* \subseteq \dot{r}(H^\star[\![e]\!] \cap \Gamma^*) \qquad\qquad (\text{by } (3"))$$

$$\subseteq \dot{r}(H^\star[\![e]\!]) \qquad\qquad (\dot{r} \text{ monotone})$$

$$\subseteq H'^\star(\dot{r}([\![e]\!])) \qquad\qquad (\dagger)$$

$$= H'^\star[\![r(e)]\!] \ . \qquad\qquad (\text{by Fact 3.4})$$

It remains to show $(\dagger)$ $\dot{r} \circ H^\star \subseteq H'^\star \circ \dot{r}$. Since $\dot{r}$ is linear, Proposition A.8 applies and it suffices to prove $(\ddagger)$ $\dot{r} \circ H \subseteq H'^\star \circ \dot{r}$. Let $L \subseteq \Sigma^*$. Again since $\dot{r}$ is linear, we have

$$\dot{r}(H(L)) = \bigcup_{\substack{e \le f \in H \\ u\,[\![f]\!]\,v \subseteq L}} \dot{r}(u\,[\![e]\!]\,v) \ .$$

Accordingly, let $e \le f \in H$ and let u, v be words such that $u\,[\![f]\!]\,v \subseteq L$. By assumption (4), we have a derivation $\mathsf{KA}_{H'} \vdash r(e) \le r(f)$, whence, by monotonicity, another derivation

$$\mathsf{KA}_{H'} \vdash r(u)r(e)r(v) \le r(u)r(f)r(v) \ . \qquad\qquad (3.1)$$

We deduce:

$$\dot{r}(u \llbracket e \rrbracket v) = \dot{r}(\llbracket uev \rrbracket)$$

$$= \llbracket r(uev) \rrbracket \qquad\qquad\qquad \text{(by Fact 3.4)}$$

$$= \llbracket r(u)r(e)r(v)) \rrbracket \qquad\qquad\qquad (r \text{ homomorphism})$$

$$\subseteq H'^{\star} \llbracket r(u)r(f)r(v)) \rrbracket \qquad\qquad \text{(by Theorem 2.4 applied to (3.1))}$$

$$= H'^{\star} \llbracket r(ufv) \rrbracket \qquad\qquad\qquad (r \text{ homomorphism})$$

$$= H'^{\star}(\dot{r}(\llbracket ufv \rrbracket)) \qquad\qquad\qquad \text{(by Fact 3.4)}$$

$$= H'^{\star}(\dot{r}(u \llbracket f \rrbracket v)) \subseteq H'^{\star}(\dot{r}(L)) . \qquad\qquad \text{(by monotonicity)}$$

Therefore, we get $\dot{r}(H(L)) \subseteq H'^{\star}(\dot{r}(L))$: we have proved (‡), which concludes the proof. □

Remark 3.5. *The idea to use two sets of hypotheses in Definition 2.9 is from* [KBS$^+$20]*, where reductions are defined slightly differently: the alphabet is fixed (that is, $\Sigma = \Gamma$), and the last condition is instead defined as $H^{\star}\llbracket e \rrbracket = H^{\star}\llbracket f \rrbracket \Rightarrow H'^{\star}\llbracket r(e) \rrbracket = H'^{\star}\llbracket r(f) \rrbracket$. An extra notion of* strong *reduction is then introduced, which coincides with our definition if $\Sigma = \Gamma$. By allowing a change of alphabet, we do not need to distinguish reductions and strong reductions. Proposition 3.2 is in* [KBS$^+$20, Lemma 4.23]*, adapted here to the case with two alphabets (this is taken care of in* loc. cit. *by assuming $H'^{\star}$ preserves languages over Γ).*

3.3. Automata-based reductions.
Another recipe to construct reductions consists in using finite automata (see, e.g., [ÉB95, KM14, DKPP19]).

Suppose, for instance, that we want to build a reduction from $H \triangleq \{a \leq bc\}$ to the empty set. Given an automaton for the language L, an automaton for the language $H^{\star}(L)$ is easily obtained by adding "shortcuts": we add an a-transition from state i to state j whenever there is a bc-labelled path from i to j. Using Kleene's theorem—the languages denoted by regular expressions are the languages accepted by finite automata—we define a function r on regular expressions accordingly: given an expression e, compute an automaton for its language, add shortcuts as above, and define $r(e)$ to be any expression for the resulting automaton. The first requirement of a reduction holds trivially since we target the empty set; the third one holds by construction; and since $\llbracket e \rrbracket \subseteq \llbracket r(e) \rrbracket$ (also by construction), we have $\mathsf{KA} \vdash e \leq r(e)$ by completeness of KA (Theorem 2.1). Therefore, it only remains to prove $\mathsf{KA}_H \vdash r(e) \leq e$. This last step is the most delicate one; it can be obtained using the following technique, inspired from [DDP18, Lemma 32].

Definition 3.6. Let H be a set of hypotheses. Given a (non-deterministic, finite) automaton $\mathcal{A}$, a labelling function $\mathbf{f}$ from states to regular expressions is *compatible (with $\mathcal{A}$)* if $\mathsf{KA}_H \vdash 1 \leq \mathbf{f}(i)$ whenever i is an accepting state, and $\mathsf{KA}_H \vdash a \cdot \mathbf{f}(j) \leq \mathbf{f}(i)$ whenever there is an a-transition from i to j (for a a letter, or 1 in case of an epsilon transition).

Any function associating to each state a regular expression for its language in the automaton is compatible (by completeness of KA); we call such labellings *canonical*.

Proposition 3.7. *Let H be a set of hypotheses, let $\mathcal{A}$ be an automaton, let $\mathbf{r}$ be a canonical labelling and let $\mathbf{f}$ be a compatible labelling. For all states i, we have $\mathsf{KA}_H \vdash \mathbf{r}(i) \leq \mathbf{f}(i)$.*

Proof. Thanks to completeness of KA, this is a direct consequence of [DDP18, Lemma 32] (this lemma only gives the statement for a specific canonical labelling).

A more direct proof can be obtained as follows, using the matricial representation of automata [Koz91]. Write the automaton $\mathcal{A}$ matricially as $\langle u, M, v \rangle$, where u is the row vector of initial states, M is the square matrix of transitions, and v is the column vector of accepting states. Accordingly, see the labellings $\mathbf{r}$ and $\mathbf{f}$ as column vectors. Recall that an expression for the language of $\mathcal{A}$ can be obtained by computing the product $u \cdot M^* \cdot v$ in the Kleene algebra of matrices. We have $\mathsf{KA} \vdash \mathbf{r} = M^* \cdot v$ by completeness of KA, and the compatibility of $\mathbf{f}$ can be restated as $\mathsf{KA}_H \vdash v \leq \mathbf{f}$ and $\mathsf{KA}_H \vdash M \cdot \mathbf{f} \leq \mathbf{f}$. By star induction on the left, we deduce $\mathsf{KA}_H \vdash \mathbf{r} = M^* \cdot v \leq \mathbf{f}$. □

This proposition makes it possible to finish our example with $H \triangleq \{a \leq bc\}$. Let e be a regular expression, let $\mathcal{A}$ be an automaton for its language, and let $\mathcal{A}'$ be the automaton obtained after adding shortcuts. Let $\mathbf{e}$ and $\mathbf{r}$ be canonical labellings of $\mathcal{A}$ and $\mathcal{A}'$, respectively. We have that $\mathbf{e}$ is compatible with $\mathcal{A}$, but also with $\mathcal{A}'$. Indeed, each new transition $i \xrightarrow{a} j$ in $\mathcal{A}'$ comes from two transitions $i \xrightarrow{b} k \xrightarrow{c} j$ in $\mathcal{A}$, and we have

$$\mathsf{KA}_H \vdash a \cdot \mathbf{e}(j) \leq b \cdot c \cdot \mathbf{e}(j) \qquad\qquad\qquad (\text{hypothesis in } H)$$
$$\leq b \cdot \mathbf{e}(k) \qquad\qquad\qquad (\mathbf{e} \text{ is compatible with } \mathcal{A})$$
$$\leq \mathbf{e}(i) \qquad\qquad\qquad (\mathbf{e} \text{ is compatible with } \mathcal{A})$$

Therefore, by Proposition 3.7, $\mathsf{KA}_H \vdash \mathbf{r} \leq \mathbf{e}$. It follows that $\mathsf{KA}_H \vdash r(e) = u \cdot \mathbf{r} \leq u \cdot \mathbf{e} = e$ (using the row vector u of initial states of $\mathcal{A}$).

3.4. **Basic reductions.** The following result collects several sets of hypotheses for which we have reductions to the empty set. These mostly come from the literature; we establish them using the tools from the previous two sections. They form basic building blocks used in the more complex reductions that we present in the examples below.

Lemma 3.8. *The following sets of hypotheses over Σ reduce to the empty set.*
  (i) $\{u_i \leq w_i \mid i \in I\}$ *with* $u_i, w_i \in \Sigma^*$ *and* $|u_i| \leq 1$ *for all* $i \in I$
  (ii) $\left\{ 1 = \sum_{a \in S_i} a \mid i \in I \right\}$ *with each* $S_i \subseteq \Sigma$ *finite*
  (iii) $\{e \leq 0\}$ *for* $e \in \mathcal{T}(\Sigma)$
  (iv) $\{ea \leq a\}$ *and* $\{ae \leq a\}$ *for* $a \in \Sigma$, $e \in \mathcal{T}(\Sigma \setminus \{a\})$
  (v) $\{aa \leq a\}$ *for* $a \in \Sigma$

Proof.
  (i) This is [KM14, Theorem 2]. (The result mentions equations, but in the proof only the relevant inequations are used.) Intuitively, the example from Section 3.3 can easily be generalised: starting from an automaton $\mathcal{A}$, add u_i-labelled shortcuts for all w_i-labelled paths in $\mathcal{A}$ to obtain an automaton $\mathcal{A}'$. There is a catch, however: if some of the u_i are letters appearing in some of the w_i, then $\mathcal{A}'$ is not necessarily closed: we may have to add shortcuts to $\mathcal{A}'$ and get a third automaton $\mathcal{A}''$, and so on. Fortunately, this process terminates since we never add new states to the automaton.
  (ii) This is [DKPP19, Theorem 4]. In the case of a single hypothesis $H = \{1 = a + b\}$, the proof given there can be rephrased as follows. Split H into two sets $H_1 = \{a \leq 1, \ b \leq 1\}$ and $H_2 = \{1 \leq a + b\}$. One can easily show that $H^\star = H_2^\star \circ H_1^\star$ (in fact this is an immediate consequence of Corollary 5.11 and Lemmas 3.15 and 3.17 below). By the previous item (i), H_1 reduces to the empty set, say via s. Let us try to construct a reduction r from H_2 to the empty set. We use for that the automata-based

technique from Section 3.3. A difficulty is that we need more room in the starting automaton to add shortcuts: since the right hand-side of the inequation in H_2 is a sum, we have to detect conjunctions.

Given an expression e and a deterministic finite automaton $\mathcal{A}_0$ for its language L, we first extend $\mathcal{A}_0$ into a (deterministic, finite) automaton $\mathcal{A}$ whose states recognise all intersections of residuals of L: if $\mathcal{A}_0$ has state-space X, initial state $i \in X$, transition function δ_0, and accepting states $F_0 \subseteq X$, then $\mathcal{A}$ has state-space $\mathcal{P}(X)$, initial state $\{i\}$, transition function $\delta(a, S) \triangleq \{\delta_0(a, x) \mid x \in S\}$, and accepting states $\mathcal{P}(F_0) \subseteq \mathcal{P}(X)$. By construction, the language of a state S in $\mathcal{A}$ is the intersection of the languages of the elements of S in $\mathcal{A}_0$, and $\mathcal{A}$ contains $\mathcal{A}_0$ via the singleton function.

Then we add shortcuts to $\mathcal{A}$ to obtain an automaton $\mathcal{A}'$: from every state S, we add an epsilon transition to the state

$$S' \triangleq \delta(a, S) \cup \delta(b, S)$$

(Recall that here, the union semantically yields a conjunction.) We let $r(e)$ be a regular expression for the language of $\mathcal{A}'$.

Let us prove the second condition of a reduction for r. We have $\mathsf{KA} \vdash e \leq r(e)$ by completeness and construction, since the language of $\mathcal{A}$ is contained in that of $\mathcal{A}'$. For the other inequation, we use Proposition 3.7. Let $\mathbf{e}$ and $\mathbf{r}$ be canonical labellings of $\mathcal{A}$ and $\mathcal{A}'$, respectively. Since we have $\mathsf{KA} \vdash e = \mathbf{e}(\{i\})$ and $\mathsf{KA} \vdash r(e) = \mathbf{r}(\{i\})$, it suffices to prove that $\mathbf{e}$, which is compatible with $\mathcal{A}$, is also compatible with $\mathcal{A}'$. Indeed, for each added epsilon transition from S to S', we have:

$$
\begin{aligned}
\mathsf{KA}_H \vdash 1 \cdot \mathbf{e}(S') &\leq (a + b) \cdot \mathbf{e}(S') && \text{(hypothesis in } H) \\
&= a \cdot \mathbf{e}(S') + b \cdot \mathbf{e}(S') && \text{(distributivity)} \\
&\leq a \cdot \mathbf{e}(\delta(a, S)) + b \cdot \mathbf{e}(\delta(b, S)) \\
&&& \mathllap{\text{(KA completeness with } [\![S']\!] = [\![\delta(a, S)]\!] \cap [\![\delta(b, S)]\!])} \\
&\leq \mathbf{e}(S) && \text{(compatibility of } \mathbf{e}, \text{ along } a \text{ and } b \text{ transitions)}
\end{aligned}
$$

Now consider the third condition for a reduction and define the following function on languages:

$$P_2 \colon L \mapsto \{u_0 u_1 \ldots u_n \mid u_0 \{a, b\} u_1 \ldots \{a, b\} u_n \subseteq L\}$$

Intuitively, $P_2(L)$ is the "one-parallel step" closure of L w.r.t. H_2, and the expression $r(e)$ we constructed via the automaton $\mathcal{A}'$ is such that:

$$[\![r(e)]\!] = P_2 [\![e]\!]$$

Unfortunately, in general, we only have $P_2(L) \subseteq H_2^\star(L)$. For instance, if $L = \{aa, ab, bb\}$, then the empty word does not belong to $P_2(L)$, although it belongs to $H_2^\star(L)$. Therefore, r is not a reduction from H_2 to the empty set. Still, if L is closed w.r.t. H_1 (i.e., $H_1(L) \subseteq L$), then $H_2^\star(L) = P_2(L)$. This makes it possible to conclude: $r \circ s$ is a reduction from H to the empty set: for all expressions e, we have

$$\mathsf{KA}_H \vdash r(s(e)) = s(e) = e, \text{ and}$$

$$H^\star[\![e]\!] = H_2^\star(H_1^\star[\![e]\!]) = P_2(H_1^\star[\![e]\!]) = P_2 [\![s(e)]\!] = [\![r(s(e))]\!] \ .$$

(iii) This is basically due to [Coh94], but since it is phrased differently there we include a proof.

Define $r \colon \mathcal{T}(\Sigma) \to \mathcal{T}(\Sigma)$ by $r(f) = f + \Sigma^* \cdot e \cdot \Sigma^*$ for $f \in \mathcal{T}(\Sigma)$, where Σ^* is seen as the expression $(\sum_{a \in \Sigma} a)^*$. We claim r witnesses a reduction. First, we have $\mathsf{KA}_H \vdash f \leq r(f)$ trivially, and $\mathsf{KA}_H \vdash r(f) \leq f$ follows from $e \leq 0 \in H$.

Second, we have to prove that $H^\star(\llbracket f \rrbracket) \subseteq \llbracket r(f) \rrbracket$. In fact, since the right-hand side of the considered hypothesis is 0, the one-step function associated to H is constant, and we have that for all languages L,

$$H^\star(L) = L \cup H(L) = L \cup \{u \llbracket e \rrbracket v \mid u, v \in \Sigma^*\} = L \cup \Sigma^* \cdot \llbracket e \rrbracket \cdot \Sigma^* .$$

In particular, for $L = \llbracket f \rrbracket$, we get $H^\star\llbracket f \rrbracket = \llbracket f + \Sigma^* e \Sigma^* \rrbracket = \llbracket r(f) \rrbracket$.

(iv) Hypotheses of a similar form are studied in the setting of Kleene algebra with tests in [HK02]. Consider the case $\{ea \leq a\}$. Define r as the unique homomorphism satisfying $r(a) = e^*a$, and $r(b) = b$ for all $b \in \Sigma \setminus \{a\}$. We use Proposition 3.2 to show that this witnesses a reduction.

The first two conditions are trivial for letters $b \in \Sigma$ with $a \neq b$, since r is the identity on those letters. For a, we have $\mathsf{KA} \vdash a \leq e^*a = r(a)$ easily from the KA axioms. Further, we get $\mathsf{KA}_H \vdash r(a) = e^*a \leq a$ by $ea \leq a$ and the (left) induction axiom for Kleene star in KA. Finally, we have to prove that $\mathsf{KA} \vdash r(ea) \leq r(a)$. We have $r(a) = e^*a$, and since e does not contain a, we have $r(e) = e$. Therefore, $r(ea) = r(e)r(a) = ee^*a$, and the required inequality follows since $\mathsf{KA} \vdash ee^* \leq e^*$.

(v) This is a variant of the previous item, but not an instance due to the condition that a cannot appear in e. The same homomorphism r (satisfying $r(a) = a^+$) nevertheless gives a reduction: the first three conditions of Proposition 3.2 are satisfied like in the previous item; for the fourth one, we have $\mathsf{KA} \vdash r(aa) = a^+a^+ \leq a^+ = r(a)$. $\qquad\square$

Note that Item (iii) above covers finite sets of hypotheses of the form $\{e_i \leq 0\}_{i \in I}$, as these can be encoded as the single hypothesis $\sum_{i \in I} e_i \leq 0$.

Remark 3.9. *We had wrongly announced in [PRW21, Lemma 3.6(ii)] a reduction to the empty set for $\{1 \leq \sum_{a \in S_i} a \mid i \in I\}$ (inequations rather than equations), attributing a proof to [DKPP19, Proposition 6]. We had forgotten the side condition in this proposition, which precisely prevents the function r given in the proof sketch of item (ii) above to be a reduction. Whether there exists a reduction to the empty set for such sets of hypotheses remains open.*

We conclude this subsection with a lemma allowing one to construct reductions for hypotheses of the shape $f \leq 1$ for f a regular expression. This is not always possible: consider for instance the hypothesis $P \triangleq \{ab \leq 1\}$ for two distinct letters a, b; then $P^\star\llbracket 1 \rrbracket$ consists of all well-parenthesised words (where a means opening a parenthesis, and b means closing one), which is not a regular language. In its special case where H is the empty set, Lemma 3.10 below shows a solution when we have a regular expression f' above f, whose language is closed under $f \leq 1$, and such that $f \leq 1 \vdash f' \leq 1$. These requirements are often too strong (as in the above example), which is why we allow for a generic target H. We use this lemma only for our most advanced example (NetKAT, in Section 10).

Lemma 3.10. *Let f be a regular expression and let H be a set of hypotheses. Set $H_f \triangleq \{f \leq 1\}$. If*
(1) $(H \cup H_f)^\star = H^\star \circ H_f^\star$
and there is a regular expression f' such that:
(2) $H, f \leq 1 \vdash f' \leq 1$

(3) $[\![f]\!] \cup H_f \, [\![f']\!] \subseteq [\![f']\!]$

then $H \cup H_f$ reduces to H.

Proof. Let $H_1 \triangleq H \cup H_f$. We trivially have $H_1 \vdash H$. We use the automata technique from Section 3.3 to build the reduction: given a regular expression e, compute an automaton for its language, add f'-labelled self-loops to all states, and read back a regular expression $r(e)$.

By assumption 2, and using Proposition 3.7 like in Section 3.3, we get $H_1 \vdash r(e) \leq e$. It follows that $H_1 \vdash r(e) = e$ by KA completeness, since $[\![e]\!] \subseteq [\![r(e)]\!]$. Therefore, it only remains to show the third condition of a reduction. By construction, we have

$$[\![r(e)]\!] = \big\{ u_0 v_0 \cdots v_{n-1} u_n \mid u_0 \cdots u_n \in [\![e]\!] \text{ and } \forall i, v_i \in [\![f']\!] \big\}$$

We prove that this language is H_f-closed ($H_f \, [\![r(e)]\!] \subseteq [\![r(e)]\!]$). Indeed, if we insert a word of $[\![f]\!]$ into a word of $[\![r(e)]\!]$, then it has to be either into one of the u_i in the above formula, or into one of the v_i. In both cases, we deduce by assumption 3 that the resulting word belongs to $[\![r(e)]\!]$.

The third condition of a reduction is $H_1{}^\star [\![e]\!] \subseteq H^\star [\![r(e)]\!]$. By general closure properties, it suffices to show that $H^\star [\![r(e)]\!]$ is H_f-closed ($H_f(H^\star [\![r(e)]\!]) \subseteq H^\star [\![r(e)]\!]$). In turn, since $H_f \circ H^\star \subseteq H^\star \circ H_f{}^\star$ by assumption 1, this follows from the fact that $[\![r(e)]\!]$ is H_f-closed, which we have just proved. □

3.5. **Composing reductions.** The previous subsection gives reductions for uniform sets of equations. However, in the examples we often start with a *collection* of hypotheses of different shapes, which we wish to reduce to the empty set. Therefore, we now discuss a few techniques for combining reductions.

First, reductions can be composed:

Lemma 3.11. *Let H_1, H_2 and H_3 be sets of hypotheses. If H_1 reduces to H_2 and H_2 reduces to H_3 then H_1 reduces to H_3.*

Proof. Let $r_1 \colon \mathcal{T}(\Sigma_1) \to \mathcal{T}(\Sigma_2)$ and $r_2 \colon \mathcal{T}(\Sigma_2) \to \mathcal{T}(\Sigma_3)$ be reductions from H_1 to H_2 and from H_2 to H_3, respectively. We show that $r_2 \circ r_1$ is a reduction from H_1 to H_3.

(1) We have $H_1 \vdash H_2$ and $H_2 \vdash H_3$, and therefore $H_1 \vdash H_3$.
(2) For $e \in \mathcal{T}(\Sigma_1)$, we have $H_1 \vdash e = r_1(e)$. Since $r_2(e) \in \mathcal{T}(\Sigma_2)$, we also have $H_2 \vdash r_1(e) = r_2(r_1(e))$. Since $H_1 \vdash H_2$, we deduce $H_1 \vdash e = r_1(e) = r_2(r_1(e))$.
(3) For $e \in \mathcal{T}(\Sigma_1)$, we have:

$$H_1{}^\star [\![e]\!] \cap \Sigma_3^* = (H_1{}^\star [\![e]\!] \cap \Sigma_2^*) \cap \Sigma_3^* \subseteq H_2{}^\star [\![r_1(e)]\!] \cap \Sigma_3^* \subseteq H_3{}^\star [\![r_2(r_1(e))]\!]$$

using that $\Sigma_3 \subseteq \Sigma_2$ for the first equality, and then that r_1 and r_2 are reductions. □

Second, the identity function gives non-trivial reductions under certain assumptions:

Lemma 3.12. *Let H, H' be sets of hypotheses over a common alphabet. If $H \vdash H'$ and $H \subseteq H'^\star$, then H reduces to H'.*

Proof. We have $H'^\star \subseteq H^\star$ by Lemma 2.6, whence $H^\star = H'^\star$ via the second assumption. Therefore, the identity map on terms fulfils the requirements of Definition 2.9. □

In particular, equiderivable sets of hypotheses reduce to each other:

Corollary 3.13. *Let H, H' be sets of hypotheses over a common alphabet. If $H \vdash H'$ and $H' \vdash H$ then H and H' reduce to each other.*

Proof. Thanks to Lemma 2.6, Lemma 3.12 applies in both directions.   □

Third, the following lemma makes it possible to obtain reductions for unions of hypotheses by considering those hypotheses separately.

Lemma 3.14. *Let H_1, H_2, H_1', H_2' be sets of hypotheses over a common alphabet Σ.*

If $\begin{cases} H_1 \text{ reduces to } H_1' \text{ and } H_2 \text{ reduces to } H_2' \\ (H_1 \cup H_2)^\star \subseteq H_2^\star \circ H_1^\star \\ H_2' \vdash H_1' \end{cases}$ then $H_1 \cup H_2$ reduces to $H_1' \cup H_2'$.

Proof. Let $H = H_1 \cup H_2$ and $H' = H_1' \cup H_2'$. For $i = 1, 2$, let r_i be the reduction from H_i to H_i'. We show that $r_2 \circ r_1$ is a reduction from H to H'.

(1) Since $H_i \vdash H_i'$ for $i = 1, 2$, we have $H \vdash H'$.

(2) For all expressions e, we have $H \vdash H_i \vdash e = r_i(e)$ for $i = 1, 2$. We deduce that for all expressions e, $H \vdash e = r_1(e) = r_2(r_1(e))$.

(3) For the third requirement, note that the restriction to the codomain alphabet is void since there is a single alphabet (so that we simply have $H_i^\star[\![e]\!] \subseteq H_i'^\star[\![r_i(e)]\!]$ for all i, e).

Also observe that by Lemma 2.6 and the assumptions $H_2 \vdash H_2' \vdash H_1'$, we have $H_1'^\star \subseteq H_2'^\star \subseteq H_2^\star$. From the first inclusion, we deduce (†) $H_2'^\star = H'^\star$; from the composed one, we obtain (‡) $H_2^\star H_1'^\star = H_2^\star$. We deduce

$$H^\star[\![e]\!] \subseteq H_2^\star(H_1^\star[\![e]\!]) \qquad\qquad \text{(by assumption)}$$
$$\subseteq H_2^\star(H_1'^\star[\![r_1(e)]\!]) \qquad\qquad (r_1 \text{ a reduction})$$
$$= H_2^\star[\![r_1(e)]\!] \qquad\qquad (\ddagger)$$
$$\subseteq H_2'^\star[\![r_2(r_1(e))]\!] \qquad\qquad (r_2 \text{ a reduction})$$
$$= H'^\star[\![r_2(r_1(e))]\!] \qquad\qquad (\dagger)$$

which concludes the proof.   □

In most cases, we use the above lemma with $H_1' = H_2'$, and the third condition is trivially satisfied. We now state a few results that make it possible to fulfil its second requirement. Those results hold in a much more general setting, they are proved in Appendix A.

Lemma 3.15. *Let H_1, H_2 be two sets of hypotheses over a common alphabet.*
We have $(H_1 \cup H_2)^\star = H_2^\star \circ H_1^\star$ if and only if $H_1^\star \circ H_2^\star \subseteq H_2^\star \circ H_1^\star$.

Lemma 3.16. *Let H_1, H_2 be two sets of hypotheses over a common alphabet.*
If $H_1 \circ H_2^\star \subseteq H_2^\star \circ H_1^\star$, then $H_1^\star \circ H_2^\star \subseteq H_2^\star \circ H_1^\star$.

Given a function f, we write $f^=$ for the function $f \cup \text{id}$ (i.e., $f^=(L) = f(L) \cup L$).

Lemma 3.17. *Let H_1, H_2 be two sets of hypotheses over a common alphabet.*
If H_1 is affine and either
(1) $H_1 \circ H_2 \subseteq H_2^\star \circ H_1^=$, *or*
(2) $H_1 \circ H_2 \subseteq H_2^= \circ H_1^\star$,
then $H_1^\star \circ H_2^\star \subseteq H_2^\star \circ H_1^\star$.

We conclude this section by returning to hypotheses of the form $e \leq 0$. We can indeed strengthen Lemma 3.8(iii) to Lemma 3.19 below, which gives a general treatment of hypotheses of the from $e \leq 0$: we can always get rid of finite sets of hypotheses of this form. A similar result, in terms of Horn formulas and in the context of KAT, is shown in [Har05].

Lemma 3.18. *Let $H_0 = \{e \leq 0\}$ for some term e. The set H_0 is affine, and for all sets H of hypotheses we have $H_0 \circ H \subseteq H_0$ and $(H_0 \cup H)^\star = H^\star \circ H_0^\star$.*

Proof. In fact the function H_0 is constant: $H_0(L) = \Sigma^* [\![e]\!] \Sigma^*$. Whence affineness and the first containment. The remaining equality follows from Lemmas 3.15 and 3.17. ◻

Lemma 3.19. *For any set of hypotheses H and any term e, $H \cup \{e \leq 0\}$ reduces to H.*

Proof. This is a direct application of Lemma 3.14: we combine the reduction from $\{e \leq 0\}$ to the empty set provided by Lemma 3.8(iii) with the identity reduction from H to itself. The second requirement holds by Lemma 3.18; the third requirement is void, since H_1' is chosen to be empty. ◻

4. Kleene algebra with tests

In this section we apply the machinery from the previous sections to obtain a modular completeness proof for Kleene algebra with tests [KS96].

A *Kleene algebra with tests (KAT)* is a Kleene algebra K containing a Boolean algebra L such that the meet of L coincides with the product of K, the join of L coincides with the sum of K, the top element of L is the multiplicative identity of K, and the bottom elements of K and L coincide.

Syntactically, we fix two finite sets Σ and Ω of primitive actions and primitive tests. We denote the set of Boolean expressions over alphabet Ω by $\mathcal{T}_{\mathsf{BA}}$:

$$\phi, \psi ::= \phi \vee \psi \mid \phi \wedge \psi \mid \neg\phi \mid \bot \mid \top \mid o \in \Omega$$

We write $\vdash_{\mathsf{BA}} \phi = \phi'$ when this equation is derivable from Boolean algebra axioms [BB70, DP90], and similarly for inequations.

We let α, β range over *atoms*: elements of the finite set $\mathsf{At} \triangleq 2^\Omega$. Those may be seen as valuations for Boolean expressions, or as complete conjunctions of literals: α is implicitly seen as the Boolean formula $\bigwedge_{\alpha(o)=1} o \wedge \bigwedge_{\alpha(o)=0} \neg o$. They form the atoms of the Boolean algebra generated by Ω. We write $\alpha \models \phi$ when ϕ holds under the valuation α. A key property of Boolean algebras is that for all atoms α and formulas ϕ, we have

$$\alpha \models \phi \Leftrightarrow \vdash_{\mathsf{BA}} \alpha \leq \phi \qquad \text{and} \qquad \vdash_{\mathsf{BA}} \phi = \bigvee_{\alpha \models \phi} \alpha$$

The KAT terms over alphabets Σ and Ω are the regular expressions over the alphabet $\Sigma + \mathcal{T}_{\mathsf{BA}}$: $\mathcal{T}_{\mathsf{KAT}} \triangleq \mathcal{T}(\Sigma + \mathcal{T}_{\mathsf{BA}})$. We write $\mathsf{KAT} \vdash e = f$ when this equation is derivable from the axioms of KAT, and similarly for inequations.

The standard interpretation of KAT associates to each term a language of guarded strings. A *guarded string* is a sequence of the form $\alpha_0 a_0 \alpha_1 a_1 \ldots a_{n-1} \alpha_n$ with $a_i \in \Sigma$ for all $i < n$, and $\alpha_i \in \mathsf{At}$ for all $i \leq n$. We write $\mathcal{GS}$ for the set $\mathsf{At} \times (\Sigma \times \mathsf{At})^*$ of such guarded strings. Now, the interpretation $\mathcal{G} \colon \mathcal{T}(\Sigma + \mathcal{T}_{\mathsf{BA}}) \to 2^{\mathcal{GS}}$ is defined as the homomorphic extension of the assignment $\mathcal{G}(a) = \{\alpha a \beta \mid \alpha, \beta \in \mathsf{At}\}$ for $a \in \Sigma$ and $\mathcal{G}(\phi) = \{\alpha \mid \alpha \models \phi\}$ for $\phi \in \mathcal{T}_{\mathsf{BA}}$, where for sequential composition of guarded strings the *coalesced product* is used. The coalesced product of guarded strings $u\alpha$ and βv is defined as $u\alpha v$ if $\alpha = \beta$ and undefined otherwise.

Theorem 4.1 [KS96, Theorem 8]. *For all $e, f \in \mathcal{T}_{\mathsf{KAT}}$, we have $\mathsf{KAT} \vdash e = f$ iff $\mathcal{G}(e) = \mathcal{G}(f)$.*

We now reprove this result using Kleene algebra with hypotheses. We start by defining the additional axioms of KAT as hypotheses.

Definition 4.2. We write bool for the set of all instances of Boolean algebra axioms over $\mathcal{T}_{\mathsf{BA}}$ and glue for the following set of hypotheses relating the Boolean algebra connectives to the Kleene algebra ones:

$$\mathsf{glue} \triangleq \{\phi \wedge \psi = \phi \cdot \psi, \ \phi \vee \psi = \phi + \psi \mid \phi, \psi \in \mathcal{T}_{\mathsf{BA}}\} \cup \{\bot = 0, \ \top = 1\}$$

We then define $\mathsf{kat} \triangleq \mathsf{bool} \cup \mathsf{glue}$.

(Note that all these equations are actually understood as double inequations.)

We prove completeness of $\mathsf{KA}_{\mathsf{kat}}$ in Section 4.2 below, by constructing a suitable reduction. Recall that this means completeness w.r.t. the interpretation $\mathsf{kat}^\star[\![-]\!]$ in terms of closed languages. Before proving completeness of $\mathsf{KA}_{\mathsf{kat}}$, we compare it to the classical completeness (Theorem 4.1). First note that $\mathsf{KA}_{\mathsf{kat}}$ contains the same axioms as Kleene algebra with tests, so that provability in $\mathsf{KA}_{\mathsf{kat}}$ and KAT coincide: $\mathsf{KA}_{\mathsf{kat}} \vdash e = f$ iff $\mathsf{KAT} \vdash e = f$. Comparing the interpretation $\mathsf{kat}^\star[\![-]\!]$ to the guarded string interpretation $\mathcal{G}$ is slightly more subtle, and is the focus of the next subsection.

4.1. **Relation to guarded string interpretation.** We prove here that the (standard) guarded string interpretation of KAT and the kat-closed language interpretation we use in the present paper both yield the same semantics. We only state the main lemmas we use to establish the correspondence, delegating detailed proofs to Appendix C.

The key step consists in characterising the words that are present in the closure of a language of guarded strings (Lemma 4.3 below). First observe that a guarded string may always be seen as a word over the alphabet $\Sigma + \mathcal{T}_{\mathsf{BA}}$. Conversely, a word over the alphabet $\Sigma + \mathcal{T}_{\mathsf{BA}}$ can always be uniquely decomposed as a sequence $\phi_0 a_0 \cdots \phi_{n-1} a_{n-1} \phi_n$ where $a_i \in \Sigma$ for all $i < n$ and each ϕ_i is a possibly empty sequence of Boolean expressions. We let ϕ range over such sequences, and we write $\overline{\phi}$ for the conjunction of the elements of ϕ.

Lemma 4.3. *Let L be a language of guarded strings. We have*

$$\phi_0 a_0 \cdots \phi_{n-1} a_{n-1} \phi_n \in \mathsf{kat}^\star(L)$$

$$\Leftrightarrow \qquad \forall \, (\alpha_i)_{i \leq n}, \quad (\forall i \leq n, \alpha_i \models \overline{\phi_i}) \quad \Rightarrow \quad \alpha_0 a_0 \cdots \alpha_{n-1} a_{n-1} \alpha_n \in L$$

Then we show that the kat-closures of $[\![e]\!]$ and $\mathcal{G}(e)$ coincide:

Lemma 4.4. *For all KAT expressions e, $\mathsf{kat}^\star[\![e]\!] = \mathsf{kat}^\star(\mathcal{G}(e))$.*

Let $\mathcal{GS}$ be the set of all guarded strings; we have:

Lemma 4.5. *For all KAT expressions e, $\mathcal{G}(e) = \mathsf{kat}^\star[\![e]\!] \cap \mathcal{GS}$.*

As an immediate consequence of Lemma 4.4 and Lemma 4.5, we can finally relate the guarded strings languages semantics to the kat-closed languages one:

Corollary 4.6. *Let $e, f \in \mathcal{T}_{\mathsf{KAT}}$. We have $\mathcal{G}(e) = \mathcal{G}(f) \Leftrightarrow \mathsf{kat}^\star[\![e]\!] = \mathsf{kat}^\star[\![f]\!]$.*

4.2. **Completeness.** To prove completeness of the kat-closed language model, we take the following steps:

(1) We reduce the hypotheses in kat to a simpler set of axioms: by putting the Boolean expressions into normal forms via the atoms, we can get rid of the hypotheses in bool. We do not remove the hypotheses in glue directly: we transform them into the following hypotheses about atoms:

$$\mathsf{atm}_0 \triangleq \{\alpha \cdot \beta \leq 0 \mid \alpha, \beta \in \mathsf{At},\ \alpha \neq \beta\}$$
$$\mathsf{atm}_1 \triangleq \{\alpha \leq 1 \mid \alpha \in \mathsf{At}\}$$
$$\mathsf{atm}_2 \triangleq \left\{1 \leq \sum_{\alpha \in \mathsf{At}} \alpha\right\}$$

We thus first show that kat reduces to $\mathsf{atm} \triangleq \mathsf{atm}_0 \cup \mathsf{atm}_1 \cup \mathsf{atm}_2$.

(2) Then we use results from Section 3 to construct a reduction from atm to the empty set, and thereby obtain completeness of $\mathsf{KA}_{\mathsf{kat}}$.

Let $r \colon \mathcal{T}(\Sigma + \mathcal{T}_{\mathsf{BA}}) \to \mathcal{T}(\Sigma + \mathsf{At})$ be the homomorphism defined by

$$r(x) \triangleq \begin{cases} a & x = a \in \Sigma \\ \sum_{\alpha \models \phi} \alpha & x = \phi \in \mathcal{T}_{\mathsf{BA}} \end{cases}$$

We show below that r yields a reduction from kat to atm, using Proposition 3.2.

Lemma 4.7. *We have* $\mathsf{atm} \vdash 1 = \sum_{\alpha \in \mathsf{At}} \alpha$, *and for all* α, $\mathsf{atm} \vdash \alpha\alpha = \alpha$.

Proof. The first equation comes from atm_1 and atm_2 For the second one,

$$\mathsf{atm} \vdash \alpha = \alpha \cdot 1 = \alpha \cdot \sum_{\beta \in \mathsf{At}} \beta = \sum_{\beta \in \mathsf{At}} \alpha \cdot \beta = \alpha \cdot \alpha$$

where we use atm_0 in the last step. $\qquad\square$

Lemma 4.8. *For all* $e \leq f \in \mathsf{kat}$, *we have that* $\mathsf{atm} \vdash r(e) \leq r(f)$.

Proof. We have five families of equations to consider, and thus ten families of inequations. We derive equalities directly, so that we only have to consider five cases:

• $\bot = 0$: we derive

$$\vdash r(\bot) = \sum_{\alpha \models \bot} \alpha = 0 = r(0)$$

• $\phi \vee \psi = \phi + \psi$: we derive

$$\vdash r(\phi + \psi) = r(\phi) + r(\psi) = \sum_{\alpha \models \phi} \alpha + \sum_{\alpha \models \psi} \alpha = \sum_{\alpha \models \phi \text{ or } \alpha \models \psi} \alpha = \sum_{\alpha \models \phi \vee \psi} \alpha = r(\phi \vee \psi)$$

• $\top = 1$: we derive

$$\mathsf{atm} \vdash r(\top) = \sum_{\alpha \models \top} \alpha = \sum_{\alpha \in \mathsf{At}} \alpha = 1 = r(1)$$

(using Lemma 4.7 for the last but one step)

- $\phi \wedge \psi = \phi \cdot \psi$: we derive

$$\mathsf{atm} \vdash r(\phi \cdot \psi) = r(\phi) \cdot r(\psi) = \sum_{\alpha \models \phi} \alpha \sum_{\beta \models \psi} \beta = \sum_{\alpha \models \phi,\ \beta \models \psi} \alpha\beta$$

$$= \sum_{\alpha \models \phi,\ \alpha \models \psi} \alpha\alpha \qquad\qquad (\mathsf{atm}_0)$$

$$= \sum_{\alpha \models \phi,\ \alpha \models \psi} \alpha \qquad\qquad (\text{Lemma 4.7})$$

$$= \sum_{\alpha \models \phi \wedge \psi} \alpha = r(\phi \wedge \psi)$$

- $\phi = \psi$, an instance of a Boolean algebra axiom: in this case, $\alpha \models \phi$ iff $\alpha \models \psi$ for all α, so that $r(\phi)$ and $r(\psi)$ are identical. $\qquad\square$

Lemma 4.9. *The homomorphism r yields a reduction from kat to atm.*

Proof. We use Proposition 3.2. We first need to show $\mathsf{kat} \vdash \mathsf{atm}$: for $\alpha, \beta \in \mathsf{At}$ with $\alpha \neq \beta$, we have the following derivations in $\mathsf{KA}_{\mathsf{kat}}$

$$\alpha \cdot \beta = \alpha \wedge \beta = \bot = 0 \qquad\qquad \alpha \leq \top = 1 \qquad\qquad 1 = \top = \bigvee_{\alpha \models \top} \alpha = \sum_{\alpha \in \mathsf{At}} \alpha$$

Now for $a \in \Sigma + \mathsf{At}$, we have $a = r(a)$ (syntactically): if $a = \mathsf{a} \in \Sigma$, then $r(\mathsf{a}) = \mathsf{a}$; if $a = \alpha \in \mathsf{At}$, then $r(\alpha) = \sum_{\alpha \models \alpha} \alpha = \alpha$. Condition 3 is thus satisfied, and it suffices to verify condition 2 for $\phi \in \mathcal{T}_{\mathsf{BA}}$. In this case, we have we have $\mathsf{kat} \vdash r(\phi) = \sum_{\alpha \models \phi} \alpha = \bigvee_{\alpha \models \phi} \alpha = \phi$. The last condition (4) was proven in Lemma 4.8. $\qquad\square$

We can finally conclude:

Theorem 4.10. *For all $e, f \in \mathcal{T}_{\mathsf{KAT}}$, $\mathsf{kat}^\star[\![e]\!] = \mathsf{kat}^\star[\![f]\!]$ implies $\mathsf{KA}_{\mathsf{kat}} \vdash e = f$.*

Proof. By completeness of KA (Theorem 2.1) and Theorem 2.11, it suffices to show that kat reduces to the empty set. By Lemma 4.9, kat reduces to atm, which reduces to $\mathsf{atm}_{1,2} \triangleq \mathsf{atm}_1 \cup \mathsf{atm}_2$ by Lemma 3.19, which reduces to the empty set by Lemma 3.8(ii). We conclude by Lemma 3.11. $\qquad\square$

5. MORE TOOLS

Before dealing with other examples, we define a few more tools for combining reductions. To alleviate notation in the sequel, we often omit the functional composition symbol ($\circ$), thus writing $H_1 H_2$ for the composition $H_1 \circ H_2$, or $H_n^\star \dots H_0^\star$ for $H_n^\star \circ \dots \circ H_0^\star$.

5.1. **Combining more than two reductions.** We start with a variant of Lemma 3.14 to compose more than two reductions, in the case they all share a common target. A similar lemma is formulated in the setting of bi-Kleene algebra [Kap20, Lemma 4.49].

Proposition 5.1. *Let $H_0, \dots, H_n, H'$ be sets of hypotheses over a common alphabet Σ. Let $H = \bigcup_{i \leq n} H_i$. If H_i reduces to H' for all i and $H^\star \subseteq H_n^\star \dots H_0^\star$, then H reduces to H'.*

Proof. The proof follows the same pattern as that of Lemma 3.14. Unfortunately, we cannot simply iterate that lemma as a black box: the assumption $H^\star = H_n^\star \ldots H_0^\star$ does not provide enough information about closures associated to subsets of $H_0, \ldots, H_n$.

Let $r_0, \ldots, r_n$ be the reductions from $H_0, \ldots, H_n$ to H'. We show that $r_n \circ \ldots \circ r_0$ is a reduction from H to H'.

(1) Since $H_0 \vdash H'$, we have $H \vdash H'$.
(2) By an easy induction on $k \leq n$, we obtain that for all e, $H \vdash e = r_k \ldots r_0(e)$; whence the second requirement for $k = n$ holds.
(3) For the third requirement, note that the restriction to the codomain alphabet is void since there is a single alphabet (so that we simply have $H_i^\star[\![e]\!] \subseteq H'^\star[\![r_i(e)]\!]$ for all i, e). We prove by induction on $k \leq n$ that for all e, $H_k^\star \ldots H_0^\star[\![e]\!] \subseteq H'^\star[\![r_k \ldots r_0(e)]\!]$.
   - for $k = 0$, the statement holds by assumption about r_0: $H_0^\star[\![e]\!] \subseteq H'^\star[\![r_0(e)]\!]$;
   - for $0 < k < n$, we have

$$H_k^\star H_{k-1}^\star \ldots H_0^\star[\![e]\!]$$
$$\subseteq H_k^\star H'^\star[\![r_{k-1} \ldots r_0(e)]\!] \tag{IH}$$
$$\subseteq H_k^\star H_k^\star[\![r_{k-1} \ldots r_0(e)]\!] \quad (H_k \vdash H' \text{ and Lemma 2.6})$$
$$= H_k^\star[\![r_{k-1} \ldots r_0(e)]\!] \quad (H_k^\star \text{ a closure})$$
$$\subseteq H'^\star[\![r_k r_{k-1} \ldots r_0(e)]\!] \quad (r_k \text{ is a reduction})$$

Together with $H^\star \subseteq H_n^\star \ldots H_0^\star$, the case $k = n$ finally gives $H^\star[\![e]\!] \subseteq H'^\star[\![r_n \ldots r_0(e)]\!]$, as required. □

5.2. **Organising closures.** The previous proposition makes it possible to combine reductions for various sets of hypotheses, provided the closure associated to the union of those hypotheses can be presented as a sequential composition of the individual closures.

The two results below help to obtain such presentations more easily. They are stronger (i.e., have weaker hypotheses) than similar results proposed in the past (e.g., [Kap20, Lemma 4.50], [PRW21, Lemma 3.10]) As before, these results are best presented at a much higher level of generality; they are proved in Appendix A.

Lemma 5.2. *Let $H_0, \ldots, H_n$ be sets of hypotheses; write $H_{<j}$ for $\bigcup_{i<j} H_i$. Suppose that for all $j \leq n$, we have $H_{<j}^\star H_j^\star \subseteq H_j^\star H_{<j}^\star$. Then $H_{<n+1}^\star = H_n^\star \ldots H_0^\star$.*

Proof idea. By iterating Lemma 3.15—see Proposition A.5. □

Proposition 5.3. *Let $H_0, \ldots, H_n$ be sets of hypotheses such that H_i is affine except possibly for H_n; write $H_{<j}$ for $\bigcup_{i<j} H_i$ as before.*

If for all $j \leq n$ we have either $\begin{cases} (1) \ \forall i < j, \ H_i H_j \subseteq H_j^\star H_{<j}^=, \ or \\ (2) \ \forall i < j, \ H_i H_j \subseteq H_j^= H_{<j}^\star, \end{cases}$ *then $H_{<n+1}^\star = H_n^\star \ldots H_0^\star$.*

Proof idea. By combining Lemma 5.2 and Lemma 3.17—see Corollary A.13. □

5.3. **Proving partial commutations.** The last proposition above is convenient as it makes it possible to analyse only compositions of single rewriting steps (i.e., functions of the shape $H_i H_j$). In the remainder of this section we provide additional tools to analyse such compositions (Proposition 5.10).

We first need to discuss an important property satisfied by all closures associated to sets of hypotheses.

Definition 5.4. A monotone function f on languages is *contextual* if for all words u, v and languages K, we have $u \cdot f(K) \cdot v \subseteq f(u \cdot K \cdot v)$.

Lemma 5.5. *The identity function is contextual, the composition of contextual functions is contextual, the union of contextual functions is contextual. Given a set H of hypotheses, the functions H and $H^\star$ are contextual.*

Proof. See Appendix A.2. □

For closures, we actually have:

Lemma 5.6. *Let H be a set of hypotheses. For all languages L, K we have*

$$H^\star(L) \cdot H^\star(K) \subseteq H^\star(L \cdot K).$$

Proof. See Lemma A.20. □

Note that such a property typically does not hold for single step functions H: take for instance $H = \{a \leq b\}$ and $L = K = \{b\}$, for which we have $H(L \cdot K) = H(\{bb\}) = \{ab, ba\}$ while $H(L) \cdot H(K) = H(\{b\}) \cdot H(\{b\}) = \{a\} \cdot \{a\} = \{aa\}$.

Also note that the converse inequality does not hold in general: take for instance $H = \{1 \leq aa\}$ and $L = K = \{a\}$, for which we have $H^\star(L \cdot K) = H^\star(\{aa\}) = \{1, aa\}$ while $H^\star(L) \cdot H^\star(K) = \{a\} \cdot \{a\} = \{aa\}$.

The above property actually makes it possible to construct models of KA_H [DKPP19], and to prove soundness (Theorem 2.4, cf. Appendix B).

Now we define a notion of *overlap* between words; together with contextuality, this will help to analyse all potential interactions between two sets of hypotheses.

Definition 5.7. An *overlap* of two words u, v is a tuple $\langle x, y, s, t \rangle$ of words such that $xuy = svt$ and either:
- x, t are empty and $|y| < |v|$; or
- y, s are empty and $|x| < |v|$; or
- x, y are empty and s, t are non-empty; or
- s, t are empty and x, y are non-empty.
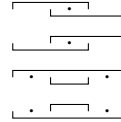
The idea is that overlaps of u, v correspond to the various ways u and v may overlap. Those situations are depicted on the right of each item. In each case the upper segment denotes u while the lower segment denotes v, and a dot indicates a non-empty word: we do forbid trivial overlaps (for instance, that the end of u overlaps with the beginning of v via the empty word does not count.)

Fact 5.8. *If $\langle x, y, s, t \rangle$ is an overlap of u, v then $\langle s, t, x, y \rangle$ is an overlap of v, u. Moreover,*
- *There is no overlap of 1 and a word of length smaller or equal to one.*
- *There is a single overlap of 1 and a word ab of size two: $\langle a, b, 1, 1 \rangle$.*
- *The only overlap between a letter and itself is $\langle 1, 1, 1, 1 \rangle$.*
- *There is no overlap of two distinct letters.*

- *There are at most two overlaps of a letter and a word of size two.*
- *There are at most three overlaps of words of size two.*

As an example of the last item, $\langle 1, c, a, 1 \rangle$ is the only overlap of ab, bc (provided a, b, c are pairwise distinct letters); and $\langle 1, 1, 1, 1 \rangle$, $\langle a, 1, 1, a \rangle$, and $\langle 1, a, a, 1 \rangle$ are the three overlaps between aa and itself.

The key property of overlaps is the following lemma.

Lemma 5.9. *For all words x, u, y, s, v, t such that $xuy = svt$, we have either:*

- $|xu| \leq |s|$, *or*
- $|sv| \leq |x|$, *or*
- *there are words l, r such that $x = lx'$, $s = ls'$, $y = y'r$, and $t = t'r$ for some overlap $\langle x', y', s', t' \rangle$ of u, v.*

Proposition 5.10. *Let H_i be a set of hypotheses whose RHS (right-hand side) are words. Let H_j be a set of hypotheses whose LHS (left-hand side) are words. Let g be a contextual function.*

If for all hypotheses $e \leq u$ of H_i, all hypotheses $v \leq f$ of H_j, and all overlaps $\langle x, y, s, t \rangle$ of u, v, we have $x \llbracket e \rrbracket y \subseteq g(s \llbracket f \rrbracket t)$, then $H_i H_j \subseteq H_j H_i \cup g$.

Proof. Suppose $w \in H_i(H_j(L))$. There is a hypothesis $e \leq u$ of H_i such that $w = xu_0y$ with $u_0 \in \llbracket e \rrbracket$ and $xuy \in H_j(L)$. In turn, there is a hypothesis $v \leq f$ of H_j such that $xuy = svt$ with $s \llbracket f \rrbracket t \subseteq L$. According to Lemma 5.9, there are three cases to consider.

- Either $|xu| \leq |s|$, in which case $s = xus'$ and $y = s'vt$ for some word s'. For all words $v_0 \in \llbracket f \rrbracket$, we have $xus'v_0t \in s \llbracket f \rrbracket t \subseteq L$, whence $xu_0s'v_0t \in x \llbracket e \rrbracket s'v_0t \subseteq H_i(L)$; in other words, $xu_0s' \llbracket f \rrbracket t \subseteq H_i(L)$. We deduce that $xu_0y = xu_0s'vt \in H_j(H_i(L))$.
- Or $|sv| \leq |x|$, in which case we proceed symmetrically.
- Or $x = lx'$, $s = ls'$, $y = y'r$, and $t = t'r$ for some words l, r and some overlap $\langle x', y', s', t' \rangle$ of u, v. We have $x' \llbracket e \rrbracket y' \subseteq g(s' \llbracket f \rrbracket t')$ by assumption, we conclude by contextuality of g:

$$xu_0y = lx'u_0y'r \in lx' \llbracket e \rrbracket y'r \subseteq l \cdot g(s' \llbracket f \rrbracket t') \cdot r \subseteq g(ls' \llbracket f \rrbracket t'r) = g(s \llbracket f \rrbracket t) \subseteq g(L) \qquad \square$$

Given two sets of hypotheses H_i, H_j, we say there is an overlap between H_i and H_j if there is any overlap between a RHS of H_i and an LHS of H_j.

Corollary 5.11. *Let H_i be a set of hypotheses whose RHS are words. Let H_j be a set of hypotheses whose LHS are words. If there are no overlaps between H_i and H_j, then we have $H_i H_j \subseteq H_j H_i$.*

In the sequel we will see examples where we have to combine more than two sets of hypotheses, and where there are some overlaps. In those cases, we will use Proposition 5.10 with a composite function g obtained from the various sets of hypotheses at hand: such functions will always be contextual by Lemma 5.5, and they will be contained in functions like $H_j^\star H_{<j}^=$ or $H_j^= H_{<j}^\star$, so that Proposition 5.3 may subsequently be applied.

Example 5.12. A typical use of the above proposition is the following. Let $H_0 = \{ac \leq cc\}$, $H_1 = \{aaa \leq ab\}$, and $H_2 = \{bc \leq ccc\}$ for three distinct letters a, b, c. To analyse the composition $H_1 H_2$, it suffices by Proposition 5.10 to consider the overlaps of ab and bc. There is only one such overlap, $\langle 1, c, a, 1 \rangle$, so that we have only to show $1 \llbracket e \rrbracket c \subseteq g(a \llbracket f \rrbracket 1)$, where $e = aaa$ is the LHS of H_1, $f = ccc$ is the RHS of H_2, and g is a function to be defined.

Simplifying both sides, we get to prove $aaac \in g(\{accc\})$. Since $aaac \leftsquigarrow_{H_0} aacc \leftsquigarrow_{H_0} accc$, we can take $g = H_0 H_0$ (or $g = H_0^\star$). We depict the overall situation as follows:

$$aaac \overset{H_1}{\leftsquigarrow} abc \overset{H_2}{\leftsquigarrow} accc$$
$$\underset{H_0}{\searrow\leftsquigarrow} aacc \underset{H_0}{\leftsquigarrow\nearrow}$$
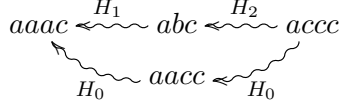
Thus we conclude that $H_1 H_2 \subseteq H_2 H_1 \cup H_0 H_0$, and we can weaken this bound to

$$H_1 H_2 \subseteq H_2 H_1 \cup H_0 H_0 \subseteq H_2 H_1 \cup H_0^\star \subseteq H_2^= (H_1 \cup H_0^\star) \subseteq H_2^= H_{<2}^\star \,.$$

5.4. **Universal hypotheses.** We prove one last generic result before moving to examples, which makes it possible to simplify the presentation of certain sets of hypotheses which are universally quantified on terms to their restriction where the quantification is only on words. This is useful in Sections 7 and 8.

We call a function h on regular expressions *monotone* if for all expressions e, f, $[\![e]\!] \subseteq [\![f]\!]$ implies $[\![h(e)]\!] \subseteq [\![h(f)]\!]$; we call it *linear* if for all expressions e, $[\![h(e)]\!] = \bigcup_{w \in [\![e]\!]} [\![h(w)]\!]$.

Proposition 5.13. *Let l, r be functions on regular expressions, with l linear and r monotone. Let $H \triangleq \{l(e) \leq r(e) \mid e \in \mathcal{T}(\Sigma)\}$ and $H' \triangleq \{l(w) \leq r(w) \mid w \in \Sigma^*\}$. We have $H = H'$, as functions on languages.*

Proof. We have $H' \subseteq H$ as sets and thus as functions, so that it suffices to prove the converse inclusion. Suppose $u \in H(L)$: $u \in x [\![l(e)]\!] y$ for some x, e, y such that $x [\![r(e)]\!] y \subseteq L$. By linearity of l, $u \in x [\![l(w)]\!] y$ for some word $w \in [\![e]\!]$. By monotonicity of r, we have $x [\![r(w)]\!] y \subseteq x [\![r(e)]\!] y$, whence $x [\![r(w)]\!] y \subseteq L$, and thus $u \in H'(L)$. $\square$

6. KLEENE ALGEBRA WITH OBSERVATIONS

A *Kleene algebra with Observations (KAO)* is a Kleene algebra which also contains a Boolean algebra, but the connection between the Boolean algebra and the Kleene algebra is different than for KAT: instead of having the axioms $\top = 1$ and $\phi \wedge \psi = \phi \cdot \psi$ for all $\phi, \psi \in \mathcal{T}_{\mathsf{BA}}$, we only have $\phi \wedge \psi \leq \phi \cdot \psi$ [KBR$^+$19]. This system was introduced to allow for concurrency and tests in a Kleene algebra framework, because associating $\phi \cdot \psi$ and $\phi \wedge \psi$ in a concurrent setting is no longer appropriate: $\phi \wedge \psi$ is one event, where we instantaneously test whether both ϕ and ψ are true, while $\phi \cdot \psi$ performs first the test ϕ, and then ψ, and possibly other things can happen between those tests in another parallel thread. Hence, the behaviour of $\phi \wedge \psi$ should be included in $\phi \cdot \psi$, but they are no longer equivalent.

Algebraically this constitutes a small change, and an ad-hoc completeness proof is in [KBR$^+$19]. Here we show how to obtain completeness within our framework. We also show how to add back the natural axiom $\top = 1$, which is not present in [KBR$^+$19], and thereby emphasise the modular aspect of the approach.

Note that even if we add the axiom $\top = 1$, in which case we have that $\phi \cdot \psi$ is below both ψ and ϕ, $\phi \cdot \psi$ and $\phi \wedge \psi$ do not collapse in this setting, because $\phi \cdot \psi$ need not be an element of the Boolean algebra. In particular, in contrast to KAT, we do not have $\alpha \cdot \beta = 0$ when α, β are distinct atoms.

Similar to KAT, we add the additional axioms of KAO to KA as hypotheses. The additional axioms of KAO are the axioms of Boolean algebra and the axioms specifying

the interaction between the two algebras. The KAO-terms are the same as the KAT-terms: regular expression over the alphabet $\Sigma + \mathcal{T}_{\mathsf{BA}}$.

Definition 6.1. We define the set of hypotheses $\mathsf{kao} \triangleq \mathsf{bool} \cup \mathsf{glue}'$, where

$$\mathsf{glue}' \triangleq \{\phi \wedge \psi \leq \phi \cdot \psi, \ \phi \vee \psi = \phi + \psi \mid \phi, \psi \in \mathcal{T}_{\mathsf{BA}}\} \cup \{\bot = 0\}$$

We prove completeness with respect to the kao-closed interpretation $(\mathsf{kao}^\star[\![-]\!])$. As shown below, this also implies completeness for the language model presented in [KBR$^+$19]. We take similar steps as for KAT:

(1) Reduce kao to a simpler set of axioms, $\mathsf{ctr} \triangleq \{\alpha \leq \alpha \cdot \alpha \mid \alpha \in \mathsf{At}\}$, where $\mathsf{At} = 2^\Omega$ is the same set of atoms as in Section 4.
(2) Use results from Section 3 to reduce ctr to the empty set.

For the first step, we use the same homomorphism r as for KAT.

Lemma 6.2. *For all $e \leq f \in \mathsf{kao}$, we have $\mathsf{ctr} \vdash r(e) \leq r(f)$.*

Proof. Similar to the proof of Lemma 4.8. First observe that in that proof, we needed the hypotheses of kat only for the $\top = 1$ case, which is not there, and for the $\phi \wedge \psi = \phi \cdot \psi$ case, which is now only an inequation, and which is dealt with as follows:

$$\mathsf{ctr} \vdash r(\phi \wedge \psi) = \sum_{\alpha \models \phi \wedge \psi} \alpha \leq \sum_{\alpha \models \phi \wedge \psi} \alpha \cdot \alpha \tag{ctr}$$

$$\leq \sum_{\alpha \models \phi, \ \beta \models \psi} \alpha \cdot \beta = \sum_{\alpha \models \phi} \alpha \cdot \sum_{\beta \models \psi} \beta = r(\phi) \cdot r(\psi) = r(\phi \cdot \psi) \qquad \square$$

Lemma 6.3. *The homomorphism r yields a reduction from kao to ctr.*

Proof. Like for Lemma 4.9, we use Proposition 3.2. We show $\mathsf{kao} \vdash \mathsf{ctr}$: for $\alpha \in \mathsf{At}$, we have $\mathsf{kao} \vdash \alpha = \alpha \wedge \alpha \leq \alpha \cdot \alpha$. The first and second condition about r are obtained like in the KAT case: the glueing equations for $\wedge$ were not necessary there. The third and last condition was proven in Lemma 6.2. $\qquad \square$

Theorem 6.4. *For all $e, f \in \mathcal{T}_{\mathsf{KAT}}$, $\mathsf{kao}^\star[\![e]\!] = \mathsf{kao}^\star[\![f]\!]$ implies $\mathsf{KA}_{\mathsf{kao}} \vdash e = f$.*

Proof. The set of hypotheses kao reduces to ctr (Lemma 6.3), which reduces to $\emptyset$ by Lemma 3.8(i), as both α and $\alpha \cdot \alpha$ are words and α is a word of length 1. $\qquad \square$

Note that the semantics defined in [KBR$^+$19] actually corresponds to $\mathsf{ctr}^\star[\![r(-)]\!]$ rather than $\mathsf{kao}^\star[\![-]\!]$. These semantics are nonetheless equivalent, kao reducing to ctr via r (the proof of Theorem 2.11 actually establishes that when H reduces to H' via r and $\mathsf{KA}_{H'}$ is complete, we have $H^\star[\![e]\!] = H^\star[\![f]\!]$ iff $H'^\star[\![r(e)]\!] = H'^\star[\![r(f)]\!]$ for all expressions e, f).

Because we set up KAO in a modular way, we can now easily extend it with the extra axiom $\top = 1$ (KA with "bounded observations"—KABO). Revisiting the proofs that r is a reduction from kat to atm and from kao to ctr, we can see that r is also a reduction from $\mathsf{kabo} \triangleq \mathsf{kao} \cup \{\top = 1\}$ to $\mathsf{atm}_{1,2} \cup \mathsf{ctr}$. Therefore, it suffices to find a reduction from $\mathsf{atm}_{1,2} \cup \mathsf{ctr}$ to the empty set. Thanks to Proposition 5.1, it suffices to decompose the closure $(\mathsf{atm}_{1,2} \cup \mathsf{ctr})^\star$ into a sequential composition of closures for which we do have reductions. We use Proposition 5.3 and Proposition 5.10 to obtain such a decomposition: by analysing the overlaps between the various components of $\mathsf{atm}_{1,2} \cup \mathsf{ctr}$, we can find how to order them sequentially.

Lemma 6.5. *We have*

(i) $\mathsf{atm}_1 \circ \mathsf{ctr} \subseteq \mathsf{ctr} \circ \mathsf{atm}_1$,
(ii) $\mathsf{atm}_1 \circ \mathsf{atm}_2 \subseteq \mathsf{atm}_2 \circ \mathsf{atm}_1$,
(iii) $\mathsf{ctr} \circ \mathsf{atm}_2 \subseteq \mathsf{atm}_2 \circ \mathsf{ctr} \cup \mathsf{ctr} \circ \mathsf{ctr}$.

Proof. For the first two items, we use Corollary 5.11: there are no overlaps between 1 and α, and no overlaps between 1 and itself. For the third item, we use Proposition 5.10 with the function $g = \mathsf{ctr} \circ \mathsf{ctr}$: the only overlap between $\alpha\alpha$ and 1 is $\langle 1, 1, \alpha, \alpha \rangle$, so that it suffices to show that $1\alpha1 \subseteq g(\alpha \left[\!\left[\sum_{\beta \in \mathsf{At}} \beta \right]\!\right] \alpha)$. We chose $\beta = \alpha$ in the sum: $\alpha\alpha\alpha \in \alpha \left[\!\left[\sum_{\beta \in \mathsf{At}} \beta \right]\!\right] \alpha$, and we check that $\alpha \leftsquigarrow_{\mathsf{ctr}} \alpha\alpha \leftsquigarrow_{\mathsf{ctr}} \alpha\alpha\alpha$ (so that $\alpha \in \mathsf{ctr}(\mathsf{ctr}(\alpha \left[\!\left[\sum_{\beta \in \mathsf{At}} \beta \right]\!\right] \alpha)))$. $\square$

Lemma 6.6. *We have* $(\mathsf{atm}_{1,2} \cup \mathsf{ctr})^{\star} = \mathsf{atm}_2{}^{\star} \circ \mathsf{ctr}^{\star} \circ \mathsf{atm}_1{}^{\star}$.

Proof. We use Proposition 5.3 with atm_1, ctr, and atm_2. The right-hand sides of ctr and atm_1 are words so that their closures are affine (Lemma 3.1). This is not the case for atm_2 which is fine because it is placed in last position. We finally have to show the commutation properties:

- there is nothing to show for the first set (atm_1);
- for the second set (ctr), we only have to bound $\mathsf{atm}_1 \circ \mathsf{ctr}$, and we may choose any of the two alternatives ((1) $\mathsf{ctr}^{\star}\mathsf{atm}_1^{=}$ or (2) $\mathsf{ctr}^{=}\mathsf{atm}_1{}^{\star}$) since Lemma 6.5(i) gives a bound ($\mathsf{ctr} \circ \mathsf{atm}_1$) which is below both of them;
- for the third set (atm_2), we have to bound both $\mathsf{atm}_1 \circ \mathsf{atm}_2$ and $\mathsf{ctr} \circ \mathsf{atm}_2$ by either (1) $\mathsf{atm}_2{}^{\star}(\mathsf{atm}_1 \cup \mathsf{ctr})^{=}$ or (2) $\mathsf{atm}_2^{=}(\mathsf{atm}_1 \cup \mathsf{ctr})^{\star}$. We have that $\mathsf{atm}_1 \circ \mathsf{atm}_2$ is below both by Lemma 6.5(ii); in contrast $\mathsf{ctr} \circ \mathsf{atm}_2$ is only (known to be) below the latter, by Lemma 6.5(iii); therefore we chose the latter option. $\square$

Unfortunately, we cannot directly apply Proposition 5.1 with this decomposition, because we do not have a reduction for atm_2 alone, but only one for $\mathsf{atm}_{1,2}$ (cf. Remark 3.9). We can nevertheless conclude just by assembling the results collected so far:

Theorem 6.7. *For all $e, f \in \mathcal{T}_{\mathsf{KAT}}$, if $\mathsf{kabo}^{\star}[\![e]\!] = \mathsf{kabo}^{\star}[\![f]\!]$ then $\mathsf{KA}_{\mathsf{kabo}} \vdash e = f$.*

Proof. From Lemma 6.6 we deduce

$$(\mathsf{atm}_{1,2} \cup \mathsf{ctr} \cup \mathsf{atm}_1)^{\star} = (\mathsf{atm}_2 \cup \mathsf{ctr} \cup \mathsf{atm}_1)^{\star} = \mathsf{atm}_2{}^{\star} \circ \mathsf{ctr}^{\star} \circ \mathsf{atm}_1{}^{\star} \subseteq \mathsf{atm}_{1,2}{}^{\star} \circ \mathsf{ctr}^{\star} \circ \mathsf{atm}_1{}^{\star}$$

(This is actually an equality since the right hand-side is contained in the left hand-side.) Thus we can apply Proposition 5.1 to atm_1, ctr, and $\mathsf{atm}_{1,2}$, for which we have reductions by Lemma 3.8. $\square$

The commutation properties used to combine the three sets of hypotheses ctr, atm_1 and atm_2 in the above proof can be presented in a table which makes it easier to see that Proposition 5.3 indeed applies (or to experiment when looking for a proof). This is Table 1.

There we use single letter shorthands for the three sets of hypotheses, which are recalled in the first column. In an entry (i, j), an expression g means we have $ij \subseteq ji \cup g$ and a dot indicates there is no overlap between i and j, so that $ij \subseteq ji$ by Corollary 5.11. The diagonal is filled with dashes only for the sake of readability. The last line collects, for each column j, an upper bound we may deduce for $(<j)j$; a dot indicates the strong bound $j^{=}(<j)^{=}$ (in both cases, $<j$ denotes the union of all functions appearing strictly before j).

			1	c	2
(atm$_1$)	$1:$	$\alpha \leq 1$	$-$	$.$	$.$
(ctr)	$c:$	$\alpha \leq \alpha\alpha$		$-$	cc
(atm$_2$)	$2:$	$1 \leq \sum \alpha$			$-$
				$.$	$2^{=}(<2)^{\star}$

Table 1: Summary of commutations for KABO.

The idea is that it suffices to cover all entries strictly above the diagonal, in such a way that the collected bound for each column (but the first one) fits into one of the two alternatives provided by Proposition 5.3.

Since dotted entries comply with both options, they pose no constraint: once they have been verified, they can be ignored. Other entries might be constraining: a column must comply to either (1) or (2) and cannot mix between them: for each column we have to make a common choice for all its rows. In Table 1 we see that entry $(c, 2)$ forces us to use the second option for column 2: with the order $1 < c < 2$, the function cc is below $2^{=}(<2)^{\star}$ (not using $2^{=}$ and entering the starred expression twice), but not below $2^{\star}(<2)^{=}$.

7. KAT with a full element

Zhang et al. observed in [ZdAG22] that extending KAT with a constant for the full relation makes it possible to model incorrectness logic [O'H20]. They left open the question of finding an axiomatisation which is complete w.r.t. relational models. We obtained a partial answer in [PW22]: for plain Kleene algebras (without tests), we get a complete axiomatisation by adding the following two axioms, where $\mathbb{F}$ is the new constant.

$$x \leq \mathbb{F} \tag{T}$$

$$x \leq x \cdot \mathbb{F} \cdot x \tag{F}$$

This completeness result was obtained in two steps: first we proved that with this new constant, the equational theory of relational models is characterised by languages closed under the above two axioms, and then we proved completeness of these axioms w.r.t. closed languages, by exhibiting a (non-trivial) reduction. In this section we show how to use this reduction as a black box and combine it with our reductions for KAT, so as to obtain completeness of KAT with the above two axioms (KATF).

Note that we use the symbol $\mathbb{F}$ here, unlike in [ZdAG22, PW22] where the symbol $\top$ is used: this makes it possible to distinguish this constant from the greatest element of the Boolean algebra of tests, and to emphasise that it should be understood as a *full* element rather than just a *top* element: axiomatically, a top element should only satisfy (T) (as was done in Example 2.13).

KATF terms ($\mathcal{T}_{\mathsf{KATF}}$) are regular expressions over the alphabet $\Sigma_{\mathsf{katf}} \triangleq \Sigma \uplus \mathcal{T}_{\mathsf{BA}} \uplus \{\mathbb{F}\}$, where $\mathcal{T}_{\mathsf{BA}}$ are the Boolean algebra terms as defined in Section 4. From now on we will often use commas to denote unions of named sets of hypotheses. As expected, we extend the set of hypotheses we used for KAT (kat) by setting $\mathsf{katf} \triangleq \mathsf{kat}, \mathsf{full}$, where

$$\mathsf{full} \triangleq \{e \leq \mathbb{F} \mid e \in \mathcal{T}_{\mathsf{KATF}}\} \cup \{e \leq e \cdot \mathbb{F} \cdot e \mid e \in \mathcal{T}_{\mathsf{KATF}}\}$$

The reduction r from kat to atm we defined in Section 4.2 actually extends to a reduction from katf to atm, full.

	0	1	t	f	2
0: $\quad \alpha\beta \leq 0\ (\alpha \neq \beta)$	−	..	..	..	..
1: $\quad\quad \alpha \leq 1$		−	t	f11	.
t: $\quad\quad w \leq \mathbb{F}$			−	ftt	.
f: $\quad\quad w \leq w\mathbb{F}w$				−	2f1
2: $\quad\quad 1 \leq \sum \alpha$					−
		.	.	f(<f)*	2(<2)*

Table 2: Summary of commutations for KATF.

Lemma 7.1. *The homomorphism r from Section 4.2 is a reduction from* katf *to* atm, full*.*

Proof. It suffices to adapt the proof of Lemma 4.9. In turn, this essentially means extending 4.8 to deal with the two new axioms:

- for $e \leq \mathbb{F}$, we have $\mathsf{full} \vdash r(e) \leq \mathbb{F} = r(\mathbb{F})$;
- for $e \leq e \cdot \mathbb{F} \cdot e$, we have $\mathsf{full} \vdash r(e) \leq r(e) \cdot \mathbb{F} \cdot r(e) = r(e \cdot \mathbb{F} \cdot e)$.

(In both cases using the fact that r is a homomorphism defined such that $r(\mathbb{F}) = \mathbb{F}$: r sees $\mathbb{F}$ as an arbitrary, non-interpreted, letter.) □

Let us now rephrase a central result of [PW22]:

Theorem 7.2 [PW22]**.** *There is a reduction from* full *to the empty set.*

Proof. This is not stated explicitly in that paper, but this is an implicit step in the proof of Theorem 4.16, which follows from Propositions 3.4, 4.7, and 4.14. □

As a consequence, since we have reductions for atm_0, atm_1 and $\mathsf{atm}_{1,2}$ to the empty set, we can follow the same strategy as before: it suffices to find how to organise the closures atm_0, atm_1, $\mathsf{atm}_{1,2}$ and full, so that we can use Proposition 5.1 and Proposition 5.3.

Since they are quantified over expressions, the inequations in full are not easy to work with. Fortunately, Proposition 5.13 gives us that $\mathsf{full}^\star = (\mathsf{t}, \mathsf{f})^\star$, where

$$\mathsf{t} \triangleq \{w \leq \mathbb{F} \mid w \in \Sigma^*_{\mathsf{katf}}\} \qquad\qquad \mathsf{f} \triangleq \{w \leq w \cdot \mathbb{F} \cdot w \mid w \in \Sigma^*_{\mathsf{katf}}\}$$

As a consequence, we can work with these simpler sets. From now on we simply write 0, 1 and 2 for atm_0, atm_1 and atm_2, and we use the following ordering:

$$0 < 1 < \mathsf{t} < \mathsf{f} < 2$$

Accordingly, we study overlaps to build Table 2. We use the same conventions as for Table 1, with the addition that we use double dotted entries to denote that Lemma 3.18 applies (in such a cell (i,j), we thus have $ij \subseteq i$, which is not constraining.) More formally, we prove:

Lemma 7.3. *We have the following inclusions of functions:*

(i) $0s \subseteq 0$ *for* $s \in \{1, \mathsf{t}, \mathsf{f}, 2\}$
(ii) $1\mathsf{t} \subseteq \mathsf{t}1 \cup \mathsf{t}$
(iii) $1\mathsf{f} \subseteq \mathsf{f}1 \cup \mathsf{f}11$
(iv) $\mathsf{tf} \subseteq \mathsf{ft} \cup \mathsf{ftt}$
(v) $12 \subseteq 21$
(vi) $\mathsf{t}2 \subseteq 2\mathsf{t}$
(vii) $\mathsf{f}2 \subseteq 2\mathsf{f} \cup 2\mathsf{f}1$

Proof. As before, we use Proposition 5.10, except for (i) which is just Lemma 3.18.

(ii) $1\mathsf{t} \subseteq \mathsf{t}1 \cup \mathsf{t}$: overlaps are of the form $\langle u, v, 1, 1\rangle$ which we solve as follows:

$$u\alpha v \xleftarrow{\ \ 1\ \ } uv \xleftarrow{\ \ \mathsf{t}\ \ } \mathbb{F}$$
$$u\alpha v \xleftarrow{\qquad \mathsf{t} \qquad} \mathbb{F}$$

(iii) $1\mathsf{f} \subseteq \mathsf{f}1 \cup \mathsf{f}11$: overlaps are of the form $\langle u, v, 1, 1\rangle$ which we solve as follows:

$$
\begin{array}{ccc}
u\alpha v & \xleftarrow{\ 1\ } uv \xleftarrow{\ \mathsf{f}\ } & uv\mathbb{F}uv \\
\mathsf{f}\uparrow & & \downarrow 1 \\
u\alpha v\mathbb{F}u\alpha v & \xleftarrow{\qquad 1 \qquad} & u\alpha v\mathbb{F}uv
\end{array}
$$

(iv) $\mathsf{t}\mathsf{f} \subseteq \mathsf{f}\mathsf{t} \cup \mathsf{f}\mathsf{t}\mathsf{t}$: overlaps are of the form $\langle u, w, 1, 1\rangle$ which we solve as follows:

$$
\begin{array}{ccc}
uvw & \xleftarrow{\ \mathsf{t}\ } u\mathbb{F}w \xleftarrow{\ \mathsf{f}\ } & u\mathbb{F}w\mathbb{F}u\mathbb{F}w \\
\mathsf{f}\uparrow & & \downarrow \mathsf{t} \\
uvw\mathbb{F}uvw & \xleftarrow{\qquad \mathsf{t} \qquad} & uvw\mathbb{F}u\mathbb{F}w
\end{array}
$$

(v) $12 \subseteq 21$: there are no overlaps.

(vi) $\mathsf{t}2 \subseteq 2\mathsf{t}$: there are no overlaps.

(vii) $\mathsf{f}2 \subseteq 2\mathsf{f} \cup 2\mathsf{f}1$: there are two kinds of overlaps:

- $\langle 1, 1, u, v\mathbb{F}uv\rangle$ which we intuitively solve as follows:

$$
\begin{array}{ccc}
uv & \xleftarrow{\ \mathsf{f}\ } uv\mathbb{F}uv \xleftarrow{\ 2\ } & u(\textstyle\sum \alpha)v\mathbb{F}uv \\
2\uparrow & & \downarrow 1 \\
u\beta v & \xleftarrow{\qquad \mathsf{f} \qquad} & u\beta v\mathbb{F}u\beta v
\end{array}
$$

Here the diagram is a bit sloppy: there is an implicit universal quantification on the atom β in the second line. Formally, we have to show $uv \in 2\mathsf{f}1(u(\sum \alpha)v\mathbb{F}uv)$. To this end, we prove that for all β, $u\beta v \in \mathsf{f}1(u(\sum \alpha)v\mathbb{F}uv)$. We fix such a β and we prove $u\beta v\mathbb{F}u\beta v \in 1(u(\sum \alpha)v\mathbb{F}uv)$. This follows from $u\beta v\mathbb{F}uv \in u(\sum \alpha)v\mathbb{F}uv$, which holds by choosing $\alpha = \beta$ in the sum.

- $\langle 1, 1, uv\mathbb{F}u, v\rangle$, which are handled symmetrically. $\square$

Thanks to these partial commutation properties, we obtain:

Lemma 7.4. *We have* $(0, 1, \mathsf{full}, 2)^\star = 2^\star\mathsf{f}^\star\mathsf{t}^\star 1^\star 0^\star = (1, 2)^\star\mathsf{full}^\star 1^\star 0^\star.$

Proof. We have $\mathsf{full} = \mathsf{t}, \mathsf{f}$ as functions by Proposition 5.13. Then we obtain:

$$(0, 1, \mathsf{t}, \mathsf{f}, 2)^\star = 2^\star\mathsf{f}^\star\mathsf{t}^\star 1^\star 0^\star \subseteq (1, 2)^\star(\mathsf{t}, \mathsf{f})^\star 1^\star 0^\star \subseteq (0, 1, \mathsf{t}, \mathsf{f}, 2)^\star$$

The equality is a direct application of Proposition 5.3: all requirements are provided in Table 2 (using alternative (2) for each column). The subsequent inclusions hold by general closure properties. $\square$

Theorem 7.5. *The set of hypotheses* katf *reduces to the empty set;* $\mathsf{KA}_{\mathsf{katf}}$ *is complete and decidable.*

(Note that we must apply Proposition 5.1 to $0, 1, \mathsf{full}$, and $(1, 2)$ rather than to $0, 1, \mathsf{t}, \mathsf{f}, 2$: while there is a reduction to the empty set for t alone—cf. Example 2.13, we do not know if there is such a reduction for f alone; similarly for 1 and 2.)

8. KAT with converse

Another important extension of Kleene algebra is Kleene algebra with converse (KA°), where a unary operation of *converse* is added ($\cdot^\circ$), to represent transposition of relations, or language reversal. One obtains a complete axiomatisation w.r.t. language models fairly easily: it suffices to state that converse is an involution that distribute over sums and products, reversing the arguments of the latter:

$$x^{\circ\circ} = x \qquad\qquad (x + y)^\circ = x^\circ + y^\circ \qquad\qquad (x \cdot y)^\circ = y^\circ \cdot x^\circ \qquad (\mathrm{I})$$

(That $0^\circ = 0$, $1^\circ = 1$, and $x^{*\circ} = x^{\circ*}$ follow.) Like with the constant $\mathbb{F}$, dealing with relational models requires more work because converse in relational models satisfies more laws than in language models. In particular, it satisfies the law

$$x \leq x \cdot x^\circ \cdot x \qquad (\mathrm{C})$$

Indeed, if a relation x contains a pair $\langle i, j \rangle$, then this pair also belongs to the relation $x \cdot x^\circ \cdot x$, by going from i to j, then back to i via x°, and then to j again.

Bloom, Bernátsky, Ésik, and Stefanescu have shown that adding this axiom actually yields a complete axiomatisation w.r.t. relational models [BÉS95, ÉB95]. We show in this section that we can integrate these axioms for converse together with the axioms for KAT, and prove completeness of KAT with converse (KAT°).

First of all, we need to get rid of the converse operation: our framework deals only with plain regular expressions. To this end, we use the laws (I) and their consequences mentioned below as left-to-right rewrite rules: this makes it possible to normalise every regular expression with converse into a regular expression on a duplicated alphabet.

To this end, let us setup some conventions on sets of shape $2X \triangleq X + X$ for some set X:

- we let bold letters range over their elements;
- we write ρ for the codiagonal surjection from $2X$ to X;
- we write $\mathsf{in}_l, \mathsf{in}_r$ for the two natural injections of X into $2X$;
- for $x \in X$, we simply write x for $\mathsf{in}_l(x)$; in contrast, we write $x^\bullet$ for $\mathsf{in}_r(x)$;
- we define an involution $\cdot^\circ$ on $2X$ by letting $x^\circ \triangleq x^\bullet$ and $x^{\bullet\circ} \triangleq x$ for all $x \in X$;

We extend the above involution $\cdot^\circ$ on $2X$ to an involution on regular expressions on the alphabet $2X$: e° is obtained from e by applying $\cdot^\circ$ on all letter leaves, and swapping the arguments of all products. (E.g., $(a \cdot b^* + c^\bullet)^\circ = b^{\bullet*} \cdot a^\bullet + c$.). This involution restricts to an involution on words over $2X$. For instance, we have $(ab^\bullet c)^\circ = c^\bullet b a^\bullet$.

Bloom, Bernátsky, Ésik, and Stefanescu first reduce the problem to a problem on (converse-free) regular expressions over the duplicated alphabet, and then they (implicitly) provide a reduction from (C) to the empty set (axioms (I) being dealt with by normalisation). To do so, we need to translate this axiom into a set of hypotheses over the duplicated alphabet:

$$\mathsf{cnv}_X \triangleq \{e \leq e \cdot e^\circ \cdot e \mid e \in \mathcal{T}(2X)\}$$

Theorem 8.1 [ÉB95]**.** *For all alphabets X, there is a reduction from cnv_X to the empty set.*

Now let us consider the combination of tests (from KAT) and converse, to obtain KAT°. There we also need to specify the behaviour of converse on tests (ϕ): we need the following law, which, unlike $1° = 1$ and $0° = 0$, is not derivable from (I).

$$\phi° = \phi \tag{I'}$$

Concerning terms, we move to regular expressions over the duplicated alphabet $2(\Sigma + \mathcal{T}_{\mathsf{BA}})$. According to the above axiom, we define:

$$\mathsf{i}_X \triangleq \{\phi° \leq \phi \mid \phi \in 2X\}$$

$$\mathsf{katc} \triangleq \mathsf{in}_l(\mathsf{kat}) \cup \mathsf{cnv}_{\Sigma+\mathcal{T}_{\mathsf{BA}}} \cup \mathsf{i}_{\mathcal{T}_{\mathsf{BA}}}$$

(Where $\phi°$ and ϕ in the definition of i_X are seen as one letter words over the alphabet $2(\Sigma + X)$, and where $\mathsf{in}_l(\mathsf{kat})$ denotes the injection via in_l of all inequations in kat (cf. Section 4), which are inequations between terms in $\mathcal{T}(\Sigma + \mathcal{T}_{\mathsf{BA}})$, into inequations between terms in $\mathcal{T}(2(\Sigma + \mathcal{T}_{\mathsf{BA}}))$.)

Recall the reduction $r\colon \mathcal{T}(\Sigma + \mathcal{T}_{\mathsf{BA}}) \to \mathcal{T}(\Sigma + \mathsf{At})$ from kat to atm we defined in Section 4.2, and let $s\colon \mathcal{T}(2(\Sigma + \mathcal{T}_{\mathsf{BA}})) \to \mathcal{T}(2(\Sigma + \mathsf{At}))$ be the homomorphism defined by

$$\begin{cases} s(\mathbf{a}) \triangleq \mathbf{a} & \mathbf{a} \in 2\Sigma \\ s(\phi) \triangleq s(\phi^\bullet) \triangleq r(\phi) + r(\phi)° & \phi \in \mathcal{T}_{\mathsf{BA}} \end{cases}$$

Following Section 4.2, we define the following sets of inequations over the alphabet $2(\Sigma + \mathsf{At})$:

- $\mathbf{0} \triangleq \{\alpha\beta \leq 0 \mid \alpha, \beta \in \mathsf{At}, \ \alpha \neq \beta\}$
- $\mathbf{1} \triangleq \{\boldsymbol{\alpha} \leq 1 \mid \boldsymbol{\alpha} \in 2\mathsf{At}\}$
- $\mathbf{2} \triangleq \{1 \leq \sum_{\alpha \in \mathsf{At}} \alpha\}$

Note that $\mathbf{0} = \mathsf{in}_l(\mathsf{atm}_0)$ and $\mathbf{2} = \mathsf{in}_l(\mathsf{atm}_2)$: these sets only deal with atoms on the left; in contrast, $\mathbf{1}$ is extended to deal with atoms on the right: $\mathbf{1} = \mathsf{in}_l(\mathsf{atm}_1), \mathsf{in}_r(\mathsf{atm}_1)$. We also set $\mathbf{1}' \triangleq \mathsf{in}_l(\mathsf{atm}_1)$, so that $\mathbf{1}', \mathbf{2}$ corresponds to $\mathsf{atm}_{1,2}$.

Lemma 8.2. *The homomorphism s is a reduction from* katc *to* $\mathbf{0}, \mathbf{1}, \mathbf{2}, \mathsf{cnv}_{\Sigma+\mathsf{At}}, \mathsf{i}_{\mathsf{At}}$.

Proof. We adapt the proof of Lemma 4.9. First observe that $\mathsf{i}_{\mathsf{At}} \vdash s(\phi) = r(\phi)$ for all $\phi \in \mathcal{T}_{\mathsf{BA}}$. This makes it possible to reuse Lemma 4.8, which we only need to extend to deal with the two new axioms:

- for $e \leq e \cdot e° \cdot e$, we first prove by induction on e that $\vdash s(e)° = s(e°)$, using that s is a homomorphism, and commutativity of sum for the base case of a formula $\phi \in 2\mathcal{T}_{\mathsf{BA}}$; then we deduce $\mathsf{cnv}_{\Sigma+\mathsf{At}}, \mathsf{i}_{\mathsf{At}} \vdash s(e) \leq s(e) \cdot s(e)° \cdot s(e) = s(e) \cdot s(e°) \cdot s(e) = s(e \cdot e° \cdot e)$.
- for $\phi° \leq \phi$ with $\phi \in 2\mathcal{T}_{\mathsf{BA}}$, we have $s(\phi°) = s(\phi)$, syntactically.

For $\mathbf{a} \in 2\Sigma$, we still have $\mathbf{a} = s(\mathbf{a})$ syntactically. We no longer have such a strong property for $\boldsymbol{\alpha} \in 2\mathsf{At}$, for which $s(\boldsymbol{\alpha}) = r(\alpha) + r(\alpha)° = \alpha + \alpha^\bullet$ with $\alpha = \rho(\boldsymbol{\alpha})$; but then $\vdash \boldsymbol{\alpha} \leq s(\boldsymbol{\alpha})$, and Condition 3 is satisfied. It remains to check Condition 2 for $\phi \in 2\mathcal{T}_{\mathsf{BA}}$; thanks to the first observation, we have $\mathsf{i}_{\mathsf{At}} \vdash s(\phi) = r(\phi)$ with $\phi = \rho(\phi)$, and we can use the fact that r satisfies Condition 2: $\mathsf{kat} \vdash \phi = r(\phi)$, so that $\mathsf{katc} \vdash \phi = \phi = r(\phi) = s(\phi)$. $\square$

We have reductions to the empty set for each set in $\mathbf{0}, \mathbf{1}, (\mathbf{1}', \mathbf{2}), \mathsf{cnv}_{\Sigma+\mathsf{At}}, \mathsf{i}_{\mathsf{At}}$: for $\mathsf{cnv}_{\Sigma+\mathsf{At}}$ this is Theorem 8.1, and the rest follows by Lemma 3.8. Therefore, it suffices to find how to organise their closures. Like for KATF, we first simplify $\mathsf{cnv}_{\Sigma+\mathsf{At}}$ into the following set, via Proposition 5.13:

$$\mathsf{c} \triangleq \{w \leq w \cdot w° \cdot w \mid w \in (2(\Sigma + \mathcal{T}_{\mathsf{BA}}))^*\}$$

	0	1	i	c	2
0 : $\quad \alpha\beta \leq 0 \ (\alpha \neq \beta)$	−	..	..	..	..
1 : $\quad \boldsymbol{\alpha} \leq 1$		−	.	c111	.
i : $\quad \boldsymbol{\alpha}^{\circ} \leq \boldsymbol{\alpha}$			−	ciii	.
c : $\quad w \leq ww^{\circ}w$				−	2ci=11
2 : $\quad 1 \leq \sum \alpha$					−
		.	.	c(<c)*	2(<2)*

Table 3: Summary of commutations for KAT°.

We also write i for i_{At}, and we use the following ordering:

$$0 < 1 < i < c < 2$$

Accordingly, we study overlaps to build Table 3. More formally, we prove:

Lemma 8.3. *We have the following inclusions of functions:*

(i) $1i \subseteq i1$
(ii) $1c \subseteq c1 \cup c111$
(iii) $ic \subseteq ci \cup ciii$
(iv) $12 \subseteq 21$
(v) $i2 \subseteq 2i$
(vi) $c2 \subseteq 2c \cup 2ci{=}11$

Proof. As before, we use Proposition 5.10.

(i) there are no overlaps.
(ii) $1c \subseteq c1 \cup c111$: overlaps are of the form $\langle u, v, 1, 1 \rangle$ which we solve as follows:

(Note that this case forces us to define 1 so that it is able to produce both kinds of atoms.)

(iii) $ic \subseteq ci \cup ciii$: overlaps are of the form $\langle u, v, 1, 1 \rangle$ which we solve as follows:

(iv) $12 \subseteq 21$: there are no overlaps.
(v) $i2 \subseteq 2i$: there are no overlaps.
(vi) $c2 \subseteq 2c \cup 2ci{=}11$: there are three kinds of overlaps:
- $\langle 1, 1, u, vv^{\circ}u^{\circ}uv \rangle$ which we intuitively solve as follows:

(Using the same implicit universal quantification on β in the bottom line, like in the proof of Lemma 7.3—restricted to atoms on the left.)

- $\langle 1, 1, uvv^\circ u^\circ u, v\rangle$, which is handled symmetrically.
- $\langle 1, 1, uvv^\circ, u^\circ uv\rangle$, for which we have

$$
\begin{array}{ccccc}
uv & \xleftarrow{\ \ \mathsf{c}\ \ } & uvv^\circ u^\circ uv & \xleftarrow{\ \ 2\ \ } & uvv^\circ(\sum \alpha)u^\circ uv \\
{\scriptstyle 2}\big\updownarrow & & & & \big\updownarrow{\scriptstyle 11} \\
u\beta v & \xleftarrow[\ \ \mathsf{c}\ \ ]{} u\beta vv^\circ\beta^\bullet u^\circ u\beta v & & \xleftarrow[\ \ \mathsf{i}\ \ ]{} & u\beta vv^\circ\beta u^\circ u\beta v
\end{array}
$$

(Again with an implicit universal quantification on left atom β in the bottom line.) Note that we need one more step than in the previous cases, to turn the left β produced from the sum into a $\beta^\bullet$. $\qquad\square$

Thanks to these partial commutation properties, we obtain:

Lemma 8.4. *We have* $(0, 1, \mathsf{i}, \mathsf{c}, 2)^\star = 2^\star \mathsf{c}^\star \mathsf{i}^\star 1^\star 0^\star = (1', 2)^\star \mathsf{c}^\star \mathsf{i}^\star 1^\star 0^\star$.

Proof. The first equality is a direct application of Proposition 5.3: all requirements are provided in Table 3 (using alternative (2) for all columns). The second one follows by general closure properties. $\qquad\square$

Recall that $(1', 2)$ corresponds to $\mathsf{atm}_{1,2}$, for which we have a reduction to the empty set (like for 0, 1, i and c). Therefore we can conclude:

Theorem 8.5. katc *reduces to the empty set;* $\mathsf{KA}_{\mathsf{katc}}$ *is complete and decidable.*

9. KLEENE ALGEBRA WITH POSITIVE TESTS

In KAT, tests are assumed to form a Boolean algebra. This is sometimes too strong; for instance, this prevents using the Coq tactic for KAT [Pou13] in situations where tests are only intuitionistic propositions. Here we study the structure obtained by assuming that they only form a distributive lattice. A *Kleene algebra with positive tests (KAPT)* is a Kleene algebra K containing a lattice L such that the meet of L coincides with the product of K, the join of L coincides with the sum of K, and all elements of L are below the multiplicative identity of K. (We discuss the variant where we have a bounded lattice at the end, see Remark 9.7). Since the product distributes over sums in K, L must be a distributive lattice. Also note that there might be elements of K below 1 that do not belong to L.

As before, we fix two finite sets Σ and Ω of primitive actions and primitive tests. We consider regular expressions over the alphabet $\Sigma + \mathcal{T}_{\mathsf{DL}}$, where $\mathcal{T}_{\mathsf{DL}}$ is the set of lattice expressions over Ω: expressions built from elements of Ω and two binary connectives $\vee, \wedge$.

We write dl for the set of all instances of distributive lattice axioms over $\mathcal{T}_{\mathsf{DL}}$ [DP90], and we set $\mathsf{kapt} \triangleq \mathsf{dl} \cup \mathsf{glue}''$ where

$$
\mathsf{glue}'' \triangleq \{\phi \wedge \psi = \phi \cdot \psi,\ \phi \vee \psi = \phi + \psi \mid \phi, \psi \in \mathcal{T}_{\mathsf{DL}}\} \cup \{\phi \leq 1 \mid \phi \in \mathcal{T}_{\mathsf{DL}}\}
$$

Like for Boolean algebras, the free distributive lattice over Ω is finite and can be described easily. An *atom* α is a non-empty subset of Ω, and we write At for the set of such atoms as before. However, while an atom $\{a, b\}$ of Boolean algebra was implicitly interpreted as the term $a \wedge b \wedge \neg c$ (when $\Omega = \{a, b, c\}$), the same atom in the context of distributive lattices is implicitly interpreted as the term $a \wedge b$—there are no negative literals

in distributive lattices. Again similarly to the case of Boolean algebras, the key property for atoms in distributive lattices is the following: for all atoms α and formulas ϕ, we have

$$\alpha \models \phi \iff \vdash_{\mathsf{DL}} \alpha \leq \phi \qquad \text{and} \qquad \vdash_{\mathsf{DL}} \phi = \bigvee_{\alpha \models \phi} \alpha$$

Like for KAT, such a property makes it possible to reduce kapt to the following set of equations on the alphabet $\Sigma + \mathsf{At}$.

$$\mathsf{atm}' \triangleq \{\alpha \cdot \beta = \alpha \cup \beta \mid \alpha, \beta \in \mathsf{At}\} \cup \{\alpha \leq 1 \mid \alpha \in \mathsf{At}\}$$

(Note that in the right-hand side of the first equation, $\alpha \cup \beta$ is a single atom, whose implicit interpretation is $\alpha \wedge \beta$.)

Lemma 9.1. *There is a reduction from* kapt *to* atm', *witnessed by the homomorphism* $r \colon \mathcal{T}(\Sigma + \mathcal{T}_{\mathsf{DL}}) \to \mathcal{T}(\Sigma + \mathsf{At})$ *defined by*

$$r(x) = \begin{cases} a & x = a \in \Sigma \\ \sum_{\alpha \models \phi} \alpha & x = \phi \in \mathcal{T}_{\mathsf{DL}} \end{cases}$$

As a consequence, in order to get decidability and completeness for KAPT (i.e., kapt), it suffices to reduce atm' to the empty set. Let us number the three kinds of inequations that appear in this set:

$$1 \triangleq \{\alpha \cup \beta \leq \alpha \cdot \beta \mid \alpha, \beta \in \mathsf{At}\} \qquad 2 \triangleq \{\alpha \cdot \beta \leq \alpha \cup \beta \mid \alpha, \beta \in \mathsf{At}\} \qquad 4 \triangleq \{\alpha \leq 1 \mid \alpha \in \mathsf{At}\}$$

We number the third set with 4 by anticipation: we will need the number 3 for another set of hypotheses later. Lemma 3.8(i) gives reductions to the empty set for 1 and 4, but so far we have no reduction for 2. We actually do not know if there is a reduction from 2 to the empty set. Instead, we establish a reduction from 2 together with 4 to 4 alone.

Lemma 9.2. *There is a reduction from 2,4 to 4, witnessed by the homomorphism* $r \colon \mathcal{T}(\Sigma + \mathsf{At}) \to \mathcal{T}(\Sigma + \mathsf{At})$ *defined by*

$$r(x) = \begin{cases} a & x = a \in \Sigma \\ \sum \left\{\alpha_1 \cdot \ldots \cdot \alpha_n \mid \alpha = \bigcup_{i \leq n} \alpha_i, \ i \neq j \Rightarrow \alpha_i \neq \alpha_j\right\} & x = \alpha \in \mathsf{At} \end{cases}$$

Proof. We use Proposition 3.2. The first condition is trivially satisfied since $2 \cup 4$ contains 4. For a letter $a \in \Sigma$, $r(a) = a$ so that second and third conditions are trivial for such letters, and we need to prove them only for atoms $\alpha \in \mathsf{At}$. $\vdash \alpha \leq r(\alpha)$ follows by using the singleton sequence α which is a term in the sum $r(\alpha)$. For the other condition, it thus suffices to show $2, 4 \vdash r(\alpha) \leq \alpha$, i.e., $2, 4 \vdash \alpha_1 \ldots \alpha_n \leq \alpha$ for all sequences $\alpha_1, \ldots, \alpha_n$ of pairwise distinct atoms whose union is α. This follows by $n - 1$ successive applications of inequations in 2. It remains to check the last condition of Proposition 3.2; we consider the two kinds of equations separately:

- $\alpha \cdot \beta \leq \alpha \cup \beta$: we have to derive $4 \vdash r(\alpha) \cdot r(\beta) \leq r(\alpha \cup \beta)$. By distributivity, this amounts to proving $\alpha_1 \ldots \alpha_n \beta_1 \ldots \beta_m \leq r(\alpha \cup \beta)$ for all sequences of pairwise distinct atoms $\alpha_1, \ldots, \alpha_n$ and $\beta_1, \ldots, \beta_m$ whose unions are α and β, respectively. The sequence $\alpha_1, \ldots, \alpha_n, \beta_1, \ldots, \beta_m$ almost yields a term in $r(\alpha \cup \beta)$: its union is $\alpha \cup \beta$, but it may contain duplicate entries. We simply remove such duplicates using inequations in 4 ($\gamma \leq 1$).
- $\alpha \leq 1$: we have to show $4 \vdash r(\alpha) \leq 1$. This follows by repeated applications of inequations in 4: all terms of the sum $r(\alpha)$ are below the identity. $\square$

	1	2	3	4	5
$1: \quad \alpha \cup \beta \leq \alpha\beta$	−	...	...	3	11
$2: \qquad \alpha\beta \leq \alpha \cup \beta$		−	332	44	.
$3: \quad \alpha \cup \beta \leq \alpha$			−	...	.
$4: \qquad\quad \alpha \leq 1$				−	.
$5: \qquad\quad 1 \leq \emptyset$					−
		.	$3^\star(<3)^=$	$4^\star(<4)^=$	$5^=(<5)^\star$

Table 4: Summary of commutations for KAPT (with 5 for KABPT).

(Note that the above reduction requires 4 in its target, and cannot be extended directly into a reduction from 1,2,4 to 4: $r(\alpha \cup \beta) \leq r(\alpha \cdot \beta)$ cannot be proved from 4—take $\alpha = \{a\}$, $\beta = \{b\}$, then ba is a term in $r(\alpha \cup \beta)$ which is not provably below $ab = r(\alpha \cdot \beta)$.)

Composed with the existing reduction from 4 to the empty set (Lemma 3.8(i)), we thus have a reduction from 2,4 to the empty set. It remains to combine this reduction to the one from 1 to the empty set (Lemma 3.8(i) again). To this end, we would like to use Proposition 5.1, which simply requires us to prove that the closure $\mathsf{atm}'^\star = (1,2,4)^\star$ is equal either to $1^\star(2,4)^\star$ or to $(2,4)^\star 1^\star$. Unfortunately, this is not the case. To see this, suppose we have two atomic tests a and b. For the first option, consider the singleton language $\{ab\}$ (a word consisting of two atoms); we have $ba \in (1,2,4)^\star \{ab\}$ (because $(a \wedge b) \in 1^\star \{ab\}$), and then using $2^\star$) but $ba \notin 1^\star(2,4)^\star \{ab\}$. For the second option, consider the singleton language $\{a\}$; we have $(a \wedge b) \in (1,2,4)^\star \{a\}$, because $ab \in 4^\star \{a\}$, but $(a \wedge b) \notin (2,4)^\star 1^\star \{a\}$ because $1^\star \{a\}$ is just $\{a\}$, and $(2,4)^\star$ does not make it possible to forge conjunctions.

In order to circumvent this difficulty, we use a fourth family of equations:

$$3 \triangleq \{\alpha \cup \beta \leq \alpha \mid \alpha, \beta \in \mathsf{At}\}$$

These axioms are immediate consequences of 1 and 4. Therefore, 1,2,4 reduces to 1,2,3,4. Moreover they consist of 'letter-letter' inequations, which are covered by Lemma 3.8(i): 3 reduces to the empty set. We shall further prove that $(1,2,3,4)^\star = 3^\star(2,4)^\star 1^\star$, so that Proposition 5.1 applies to obtain a reduction from 1,2,3,4 to the empty set.

Let us recall the five sets of hypotheses defined so far:
- $1 = \{\alpha \cup \beta \leq \alpha \cdot \beta \mid \alpha, \beta \in \mathsf{At}\}$,
- $2 = \{\alpha \cdot \beta \leq \alpha \cup \beta \mid \alpha, \beta \in \mathsf{At}\}$,
- $3 = \{\alpha \cup \beta \leq \alpha \mid \alpha, \beta \in \mathsf{At}\}$,
- $4 = \{\alpha \leq 1 \mid \alpha \in \mathsf{At}\}$.

We now prove the following partial commutations, which we summarise in Table 4. In addition to the conventions used for the previous tables, we mark with triple dots those entries (i,j) where there are overlaps but that we nevertheless have $ij \subseteq j^= i^=$ (a bound which is not constraining when we try to bound each column, nor if we try to reorder the lines and columns).

Lemma 9.3. *We have the following inclusions of functions:*
  (i) $12 \subseteq 21 \cup \mathrm{id}$
 (ii) $13 \subseteq 31$
(iii) $23 \subseteq 32 \cup 332$
 (iv) $14 \subseteq 41 \cup 3$
 (v) $24 \subseteq 42 \cup 44$

(vi) $34 \subseteq 43 \cup 4$

(vii) $43 \subseteq 34$

Proof. We use Proposition 5.10 so that it suffices to analyse overlaps.

(i) $12 \subseteq 21 \cup \mathrm{id}$: we must consider the overlaps of $\alpha\beta$ and $\gamma\delta$:
- when $\alpha\beta = \gamma\delta$, we have the full overlap $\langle 1, 1, 1, 1 \rangle$, and we check that

$$\alpha \cup \beta \overset{1}{\rightsquigarrow} \alpha\beta = \gamma\delta \overset{2}{\rightsquigarrow} \gamma \cup \delta$$

- when $\beta = \gamma$, we have the overlap $\langle 1, \delta, \alpha, 1 \rangle$, and we check that

$$(\alpha \cup \beta)\delta \overset{1}{\rightsquigarrow} \alpha\beta\delta = \alpha\gamma\delta \overset{2}{\rightsquigarrow} \alpha(\gamma \cup \delta)$$
$$\overset{}{\underset{2}{\rightsquigarrow}} \alpha \cup \beta \cup \delta = \alpha \cup \gamma \cup \delta \underset{1}{\overset{}{\rightsquigarrow}}$$

- when $\alpha = \delta$, we have the overlap $\langle \gamma, 1, 1, \beta \rangle$, which is handled symmetrically.

(ii) $13 \subseteq 31$: we must consider the overlaps of $\alpha\beta$ and $\gamma \cup \delta$.
- when $\beta = \gamma \cup \delta$ we have the overlap $\langle 1, 1, \alpha, 1 \rangle$, for which we have

$$\alpha \cup \gamma \cup \delta \overset{1}{\rightsquigarrow} \alpha(\gamma \cup \delta) \overset{3}{\rightsquigarrow} \alpha\gamma$$
$$\underset{3}{\rightsquigarrow} \alpha \cup \gamma \underset{1}{\rightsquigarrow}$$

- when $\alpha = \gamma \cup \delta$ we have the overlap $\langle 1, 1, 1, \beta \rangle$, which is handled symmetrically.

(iii) $23 \subseteq 32 \cup 332$: we must consider the overlaps of $\alpha \cup \beta$ and $\gamma \cup \delta$: there is only the full overlap, when $\alpha \cup \beta = \gamma \cup \delta$. We have

$$\alpha\beta \overset{2}{\rightsquigarrow} \alpha \cup \beta = \gamma \cup \delta \overset{3}{\rightsquigarrow} \gamma$$
$$3 \updownarrow \qquad\qquad\qquad\qquad \updownarrow 2$$
$$\alpha(\beta \cap \gamma) \underset{3}{\rightsquigarrow} (\alpha \cap \gamma)(\beta \cap \gamma)$$

(observing that $(\alpha \cap \gamma) \cup (\beta \cap \gamma) = \gamma$ for the first step, since $\alpha \cup \beta = \gamma \cup \delta$)

(iv) $14 \subseteq 41 \cup 3$: we must consider the overlaps of $\alpha\beta$ and γ.
- if $\gamma = \beta$ then we have the overlap $\langle 1, 1, \alpha, 1 \rangle$ for which we have

$$\alpha \cup \beta \overset{1}{\rightsquigarrow} \alpha\beta \overset{4}{\rightsquigarrow} \alpha$$
$$\underset{3}{\rightsquigarrow}$$

- the overlap when $\gamma = \alpha$ is handled symmetrically

(v) $24 \subseteq 42 \cup 44$: there is only the full overlap, for which we have

$$\alpha\beta \overset{2}{\rightsquigarrow} \alpha \cup \beta \overset{4}{\rightsquigarrow} 1$$
$$\underset{4}{\rightsquigarrow} \alpha \underset{4}{\rightsquigarrow}$$

(vi) $34 \subseteq 43 \cup 4$: there is only the full overlap, for which we have

$$\alpha \cup \beta \xleftarrow{\;3\;} \alpha \xleftarrow{\;4\;} 1$$

$$\xleftarrow{\;\;\;4\;\;\;}$$
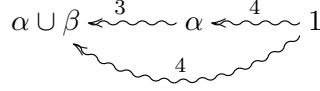
(vii) $43 \subseteq 34$: there are no overlaps between 1 and $\alpha \cup \beta$.   □

Thanks to these partial commutation properties, we obtain:

Lemma 9.4. *We have* $(1,2,3,4)^\star = 4^\star 3^\star 2^\star 1^\star = 3^\star (2,4)^\star 1^\star$.

Proof. We have

$$(1,2,3,4)^\star = 4^\star 3^\star 2^\star 1^\star \subseteq 3^\star 4^\star 2^\star 1^\star \subseteq 3^\star (2,4)^\star 1^\star \subseteq (1,2,3,4)^\star$$

The first equality is a direct application of Proposition 5.3: all requirements are provided in Table 4 (using alternative (1) for each column). The subsequent inclusion comes from Lemma 3.17 applied to 4 and 3, thanks to Lemma 9.4(vii). The remaining inclusions follow from basic properties of closures.   □

Remark 9.5. *We actually have* $(3,4)^\star = 3^\star 4^\star = 4^\star 3^\star$: *Lemma 3.17 can be applied in both directions with 3 and 4, thanks to items (vi) and (vii) in Lemma 9.4. This can be read directly on Table 4: when restricted to lines and columns 3,4, we only get dotted entries.*

However, we cannot place 4 before 3: the occurrence of 3 in entry (1,4) requires 3 to appear before 4 in order to validate column 4 when we apply Proposition 5.3 to 1,2,3,4.

We did proceed differently in the conference version of this article [PRW21], where 3 and 4 were swapped and where we did not have Proposition 5.3. This required us to iterate and combine the partial commutation properties from Lemma 9.4 manually, in a non-trivial way. We prefer the present proof, which is slightly more automatic.

Theorem 9.6. kapt *reduces to the empty set;* $\mathsf{KA_{kapt}}$ *is complete and decidable.*

Proof. kapt reduces to atm$'$ by Lemma 9.1, which in turn reduces to $1,2,3,4$ by Corollary 3.13. We see the latter as being composed of three sets of hypotheses, 1, 3, and $2,4$. All three of them reduce to the empty set: the first two by Lemma 3.8(i), and the third one by Lemma 9.2 and Lemma 3.8(i) again. These three reductions can be composed together by Proposition 5.1 and Lemma 9.4.   □

Remark 9.7. *The case of Kleene algebras containing a* bounded *distributive lattice, with extremal elements $\perp$ and $\top$ coinciding with 0 and 1, may be obtained as follows. Allow the empty atom $\emptyset$ in* At *(interpreted as $\top$), and add the inequation $5 \triangleq \{1 \leq \emptyset\}$ to* atm$'$. *Lemma 9.1 extends easily, and we have a reduction from 5 to the empty set (Lemma 3.8(i)). Therefore it suffices to find how to combine $5^\star$ with the other closures. We do so in the lemma below, so that we can conclude that the equational theory of Kleene algebras with a bounded distributive lattice is complete and decidable.*

Lemma 9.8. *We have* $(1,2,3,4,5)^\star = 5^\star 4^\star 3^\star 2^\star 1^\star = 5^\star (1,2,3,4)^\star$.

Proof. It suffices to prove the first equality: the second one follows from Lemma 9.4. To this end, it suffices to show that 5 somehow commutes over the four other functions. We do so by completing Table 4 and applying Proposition 5.3. Since the LHS of 5 is the empty

word, there are very few overlaps to consider: the only one if for bounding 15, where $\alpha\beta$ and the empty word overlap via $\langle 1, 1, \alpha, \beta \rangle$. In that case, we have

$$\alpha \cup \beta \xleftarrow{\hspace{0.3em}1\hspace{0.3em}} \alpha\beta \xleftarrow{\hspace{0.3em}5\hspace{0.3em}} \alpha\emptyset\beta$$
$$\underset{1}{\searrow} \quad \alpha\beta = \alpha(\emptyset \cup \beta) \underset{1}{\swarrow}$$

whence the 11 entry at position $(1, 5)$ in Table 4, thanks to Proposition 5.10. It is now easy to check that Proposition 5.3 applies to 1,2,3,4,5, using alternative (2) for the fifth column. $\qquad\square$

10. NetKAT

NetKAT is a framework for analysing network programs [AFG$^+$14]. It is a variant of KAT, adding an explicit alphabet of variable assignments (of the form $x \leftarrow n$) and tests of the form $x = n$, where n ranges over a finite domain of values. Here we work with *reduced* NetKAT (equivalent to NetKAT), where tests and assignments are replaced with more general complete assignments and tests. These complete tests are the atoms of a Boolean algebra. NetKAT in its reduced form is also treated as one of the motivating examples of Mamouras and Kozen's theory of Kleene algebra with equations [KM14].

Accordingly, we fix a finite set A and we work with the alphabet $A + P + \{\mathsf{dup}\}$ where $P \triangleq A$. We let α, β range over A and we call them *atoms*. We let p, q range over P and we call them *assignments*. Given an atom α, we write p_α for the corresponding assignment in P. Given an assignment p, we write α_p for the corresponding atom in A.

Following [AFG$^+$14, Figure 6], we define netkat as the following collection of equations:

$$pq = q \qquad p_\alpha\alpha = p_\alpha \qquad \alpha p_\alpha = \alpha \qquad \alpha\mathsf{dup} = \mathsf{dup}\alpha \qquad \textstyle\sum_{\alpha \in A} \alpha = 1 \qquad \alpha\beta = 0 \; (\alpha \neq \beta)$$

These equations are equivalent to $\mathsf{netkat}' \triangleq 0, 1, 2, 3, 4, 5, 6, 7$, where
- $0 \triangleq \{\alpha\beta, \alpha\mathsf{dup}\beta, p_\alpha\beta \leq 0 \mid \alpha, \beta \in A, \; \alpha \neq \beta\}$
- $1 \triangleq \{\alpha p_\alpha \leq 1 \mid \alpha \in A\}$
- $2 \triangleq \{pq \leq q \mid p, q \in P\}$
- $3 \triangleq \{\alpha \leq 1 \mid \alpha \in A\}$
- $4 \triangleq \{p_\alpha \leq p_\alpha\alpha \mid \alpha \in A\}$
- $5 \triangleq \{q \leq pq \mid p, q \in P\}$
- $6 \triangleq \{\alpha \leq \alpha p_\alpha \mid \alpha \in A\}$
- $7 \triangleq \{1 \leq \sum_{\alpha \in A} \alpha\}$

Note that 3 are 4 are redundant: they follows respectively from 1,6 and 0,7. We include them on purpose: they help getting appropriate partial commutations.

Lemma 10.1. *We have* $\mathsf{netkat}' \vdash \mathsf{netkat}$ *and* $\mathsf{netkat} \vdash \mathsf{netkat}'$.

Proof. First observe that:
- $\mathsf{netkat}' \vdash pq = q$ using 2,5;
- $\mathsf{netkat}' \vdash p_\alpha\alpha = p_\alpha$, using 3,4;
- $\mathsf{netkat}' \vdash \alpha p_\alpha \leq \alpha p_\alpha\alpha \leq \alpha \leq \alpha p_\alpha$, using 4,1,6;
- $\mathsf{netkat}' \vdash \alpha\mathsf{dup} \leq \alpha\mathsf{dup}\sum_\beta \beta = \sum_\beta \alpha\mathsf{dup}\beta = \alpha\mathsf{dup}\alpha \leq \mathsf{dup}\alpha$, using 7,0,3, and symmetrically, $\mathsf{netkat}' \vdash \mathsf{dup}\alpha \leq \alpha\mathsf{dup}$.

		0	1	2	3	4	5	6	7
0 :	$\alpha\beta, \alpha\mathsf{dup}\beta, p_\alpha\beta \le 0 \ (\alpha \neq \beta)$	−	..	..	..	..	..	..	..
1 :	$\alpha p_\alpha \le 1$		−	322	.	.	.	.	.
2 :	$pq \le q$			−	.	...	...	.	.
3 :	$\alpha \le 1$				−	.	.	.	.
4 :	$p_\alpha \le p_\alpha\alpha$					−	...	54	44
5 :	$q \le pq$						−	.	54
6 :	$\alpha \le \alpha p_\alpha$							−	66441
7 :	$1 \le \sum \alpha$								−
				×	.	.	.	$6^=(<6)^\star$	$7^=(<7)^\star$

Table 5: Summary of commutations for NetKAT.

Thus we have $\mathsf{netkat'} \vdash \mathsf{netkat}$. Conversely, $2, 4, 5, 6, 7$ already belong to netkat and $1, 3$ are almost immediate consequences (using the sum axiom of netkat). For 0, for all $\alpha \neq \beta$, we have: $\mathsf{netkat} \vdash \alpha\mathsf{dup}\beta = \mathsf{dup}\alpha\beta = \mathsf{dup}0 = 0$ and $\mathsf{netkat} \vdash p_\alpha\beta = p_\alpha\alpha\beta = p_\alpha 0 = 0$.   □

As a consequence, netkat and $\mathsf{netkat'}$ reduce to each other via Corollary 3.13.

Let us now study overlaps between the hypotheses of $\mathsf{netkat'}$ and build Table 5. As before for KATF and KAT°, the entries in the first line are all trivial by Lemma 3.18. For the other lines, the cases where there are overlaps are obtained as follows.

Proof of non-trivial entries in Table 5.

- 12: the only overlap is $\langle p, q, 1, 1 \rangle$, for which we have

$$
\begin{array}{ccccc}
p\alpha p_\alpha q & \overset{1}{\longleftarrow} & pq & \overset{2}{\longleftarrow} & q \\
{\scriptstyle 3}\big\uparrow & & & & \big\downarrow{\scriptstyle 2} \\
pp_\alpha q & & \overset{2}{\longleftarrow} & & p_\alpha q
\end{array}
$$

- 24: we only have the full overlap, for which we have

$$
\begin{array}{ccccc}
qp_\alpha & \overset{2}{\longleftarrow} & p_\alpha & \overset{4}{\longleftarrow} & p_\alpha\alpha \\
 & {\scriptstyle 4}\searrow & qp_\alpha\alpha & \overset{2}{\swarrow} &
\end{array}
$$

- 25: we only have the full overlap, for which we have

$$
\begin{array}{ccccc}
rp & \overset{2}{\longleftarrow} & p & \overset{5}{\longleftarrow} & qp \\
 & {\scriptstyle 5}\searrow & rqp & \overset{2}{\swarrow} &
\end{array}
$$

- 45: the only overlap is $\langle 1, 1, 1, \alpha \rangle$, for which we have

$$
\begin{array}{ccccc}
p_\alpha & \overset{4}{\longleftarrow} & p_\alpha\alpha & \overset{5}{\longleftarrow} & qp_\alpha\alpha \\
 & {\scriptstyle 5}\searrow & qp_\alpha & \overset{4}{\swarrow} &
\end{array}
$$

- 46: the only overlap is $\langle 1, 1, p_\alpha, 1 \rangle$, for which we have

$$p_\alpha \overset{4}{\longleftarrow\!\!\!\rightsquigarrow} p_\alpha \alpha \overset{6}{\longleftarrow\!\!\!\rightsquigarrow} p_\alpha \alpha p_\alpha$$
$$\overset{5}{\rightsquigarrow} p_\alpha p_\alpha \overset{4}{\longleftarrow}$$

- 47: the only overlap is $\langle 1, 1, p_\alpha, \alpha \rangle$, for which we have

$$p_\alpha \overset{4}{\longleftarrow\!\!\!\rightsquigarrow} p_\alpha \alpha \overset{7}{\longleftarrow\!\!\!\rightsquigarrow} p_\alpha (\textstyle\sum \beta) \alpha$$
$$\overset{4}{\rightsquigarrow} p_\alpha \alpha \overset{4}{\longleftarrow}$$

  (by choosing $\beta = \alpha$ in the sum to get $p_\alpha \alpha \alpha$)

- 57: the only overlap is $\langle 1, 1, p, q \rangle$, for which we have

$$q \overset{5}{\longleftarrow\!\!\!\rightsquigarrow} pq \overset{7}{\longleftarrow\!\!\!\rightsquigarrow} p(\textstyle\sum \beta)q$$
$$\overset{5}{\rightsquigarrow} pq \overset{4}{\longleftarrow}$$

  (by choosing $\beta = \alpha_p$ in the sum to get $p\alpha_p q$)

- 67: the only overlap is $\langle 1, 1, \alpha, p_\alpha \rangle$, for which we have

$$\alpha \overset{6}{\longleftarrow\!\!\!\rightsquigarrow} \alpha p_\alpha \overset{7}{\longleftarrow\!\!\!\rightsquigarrow} \alpha(\textstyle\sum \beta)p_\alpha$$
$$66 \big\updownarrow \qquad\qquad\qquad\qquad \big\downarrow 1$$
$$\alpha p_\alpha p_\alpha \overset{44}{\longleftarrow\!\!\!\!\!\!\!\!\!\!\!\!\!\!\!\!\!\!\!\rightsquigarrow} \alpha p_\alpha \alpha \alpha p_\alpha$$

  (by choosing $\beta = \alpha$ in the sum to get $\alpha \alpha p_\alpha$) $\qquad\qquad\qquad\qquad$ $\square$

At this point, we face several difficulties:

(a) we do not have a reduction from 1 to the empty set: these hypotheses are not covered by Lemma 3.8 (in fact such a reduction cannot exist: $1^\star$ does not preserve regularity);

(b) we do not have a reduction from 2 to the empty set, yet: while single hypotheses from 2 fit Lemma 3.8(iv)/(v), we need to combine the corresponding reductions to get a reduction for 2. (In contrast, note that sets 3,4,5,6 do not pose such a problem since Lemma 3.8(i) already deals with sets of inequations of the form letter below word.)

(c) the entry 322 at position $(1, 2)$ in Table 5 prevents us from using Proposition 5.3 directly: 322 is not below $2^=(<2)^\star$, nor below $2^\star(<2)^=$;

For (a), we will enrich 1 with other hypotheses (a subset of 2, and 3) in such a way that we recover a set of hypotheses whose closure preserve regularity, and construct a reduction via Lemma 3.10. We do so in Section 10.1. For (b), we will decompose 2 into smaller sets and use Proposition 5.1 to compose the basic reductions provided by Lemma 3.8. We do so in Section 10.2. For (c), we will deal with $(1, 2, 3)^\star$ 'manually' (Lemma 10.6), in such a way that we can use Proposition 5.3 on Table 5 by grouping (1,2,3) into a single set of hypotheses. We do so in Section 10.3, where we eventually assemble all the results to get completeness of NetKAT.

10.1. **A reduction including $\alpha p_\alpha \leq 1$.** Let us first provide a generic reduction, independently from NetKAT. We locally reuse notations 1,2,3 in the lemma below on purpose, just in the context of that lemma and its proof.

Lemma 10.2. *Let a, b be distinct letters and pose*

- $1 \triangleq \{ab \leq 1\}$
- $2 \triangleq \{bb \leq b\}$
- $3 \triangleq \{a \leq 1\}$

We have that 1,2,3 reduces to 2,3, which reduces to the empty set.

Proof. For the first reduction, we apply Lemma 3.10 to $f = ab$ and $H' = 2,3$, using $f' = a(a+b)^*$. For the two conditions about f':

(1) We have $2 \vdash b = b^+$ and $3 \vdash (a+b)^* \leq (1+b)^* = b^* \leq (a+b)^*$, from which we deduce $1, 2, 3 \vdash f' = a(a+b)^* = ab^* = a + ab^+ = a + ab \leq 1 + 1 = 1$.
(2) $H_1 \llbracket f' \rrbracket$ consists of words from $\llbracket f' \rrbracket$ where an occurrence of ab has been inserted; such words always start with an a so that they belong to $\llbracket f' \rrbracket$.

It remains to show the preliminary closure condition. Like for entries (1,2) and (1,3) in Table 5 analysing overlaps gives $12 \subseteq 21 \cup 322$ and $13 \subseteq 31$, whence $1(2, 3) \subseteq 21 \cup 322 \cup 31 \subseteq (2, 3)^\star 1^=$ (where we use additivity of 1 in the first step) and thus $(1, 2, 3)^\star = (2, 3)^\star 1^\star$ by Lemma 3.17(1), as required. We have thus proved that 1,2,3 reduces to 2,3.

For the second reduction, we have reductions to the empty set for 2 and 3 separately, by Lemma 3.8(v)/(i). We moreover have $23 \subseteq 32$: there are no overlaps. We conclude by Lemmas 3.14 and 3.17. $\square$

For $i = 1, 2, 3$, write i_α for i from the previous lemmas with $a = \alpha$ and $b = p_\alpha$. Write s_α for $1_\alpha, 2_\alpha, 3_\alpha$, and s for the union of all s_α. The set of hypotheses s almost corresponds to 1,2,3: it just misses $qp \leq p$ for all $p \neq q$.

Now recall that 0 consists of the netkat' hypotheses of the shape $e \leq 0$.

Lemma 10.3. $0, \mathsf{s}$ *reduces to the empty set.*

Proof. We have individual reductions to the empty set for 0 (by Lemma 3.19) and for each s_α (by Lemma 10.2). By Propositions 5.1 and 5.3, it suffices to check partial commutations. We have:

- for all α, $0\mathsf{s}_\alpha \subseteq \mathsf{s}_\alpha^= 0$ by Lemma 3.18;
- for all $\alpha \neq \beta$, $\mathsf{s}_\alpha \mathsf{s}_\beta \subseteq \mathsf{s}_\beta \mathsf{s}_\alpha \cup 0$; indeed we have
  - one overlap between 1_α and 1_β: $\langle \beta, p_\beta, 1, 1 \rangle$, which we can solve as follows:

$$\beta\alpha p_\alpha p_\beta \xleftarrow{\ \ 1_\alpha\ \ } \beta p_\beta \xleftarrow{\ \ 1_\beta\ \ } 1$$
$$\xleftarrow{\quad\ 0\ \quad}$$

  - one overlap between 1_α and 2_β: $\langle p_\beta, p_\beta, 1, 1 \rangle$, which we can solve as follows:

$$p_\beta\alpha p_\alpha p_\beta \xleftarrow{\ \ 1_\alpha\ \ } p_\beta p_\beta \xleftarrow{\ \ 2_\beta\ \ } p_\beta$$
$$\xleftarrow{\quad\ 0\ \quad}$$

– one overlap between 3_α and 1_β: $\langle \beta, p_\beta, 1, 1 \rangle$, which we can solve as follows:

$$\beta\alpha p_\beta \overset{3_\alpha}{\longleftarrow\!\!\rightsquigarrow} \beta p_\beta \overset{1_\beta}{\longleftarrow\!\!\rightsquigarrow} 1$$
$$\overset{0}{\longleftarrow\!\!\rightsquigarrow}$$

– one overlap between 3_α and 2_β: $\langle p_\beta, p_\beta, 1, 1 \rangle$, which we can solve as follows:

$$p_\beta\alpha p_\beta \overset{3_\alpha}{\longleftarrow\!\!\rightsquigarrow} p_\beta p_\beta \overset{2_\beta}{\longleftarrow\!\!\rightsquigarrow} p_\beta$$
$$\overset{0}{\longleftarrow\!\!\rightsquigarrow}$$

– no overlap between the remaining pairs.

We should thus place 0 first, followed by the s_α in any order. This concludes the proof. $\square$

10.2. **A single reduction for $\{qp \leq p \mid p, q \in P\}$.** Let us decompose the set of hypotheses 2 into several subsets: for $p \in P$, set

- $2_p \triangleq \{qp \leq p \mid q \in P\}$
- $2_{=p} \triangleq \{pp \leq p\}$
- $2_{\neq p} \triangleq \{qp \leq p \mid q \in P, q \neq p\}$

We have $2_p = 2_{=p}, 2_{\neq p}$ and 2 is the union of all 2_ps. Moreover, we have reductions to the empty set for each $2_{=p}$ by Lemma 3.8(v), and for each $2_{\neq p}$ by Lemma 3.8(iv) (by grouping the hypotheses in those latter sets into a single inequation $(\sum_{p \neq q} q)p \leq p$). We combine those elementary reductions in two steps.

Lemma 10.4. *For all $p \in P$, 2_p reduces to the empty set.*

Proof. There is a single overlap between $2_{=p}$ and $2_{\neq p}$, which we solve as follows:

$$qpp \overset{2_{=p}}{\longleftarrow\!\!\rightsquigarrow} qp \overset{2_{\neq p}}{\longleftarrow\!\!\rightsquigarrow} p$$
$$\overset{2_{\neq p}}{\searrow\!\!\rightsquigarrow} qp \overset{2_{=p}}{\swarrow}$$

Thus we have $2_{=p}2_{\neq p} \subseteq 2_{\neq p}2_{=p}$, and we conclude with Lemmas 3.14 and 3.17. $\square$

(Alternatively, it is also easy to build directly a homomorphic reduction from 2_p to the empty set by adapting the proof of Lemma 3.8(iv).)

Lemma 10.5. 2 *reduces to the empty set.*

Proof. For $p \neq q$, there is a single overlap between 2_q and 2_p, which we solve as follows:

$$rqp \overset{2_q}{\longleftarrow\!\!\rightsquigarrow} qp \overset{2_p}{\longleftarrow\!\!\rightsquigarrow} p$$
$$\overset{2_p}{\searrow\!\!\rightsquigarrow} rp \overset{2_p}{\swarrow}$$

Thus we have $2_q 2_p \subseteq 2_p 2_q \cup 2_p 2_p \subseteq 2_p^\star 2_q^=$, and we conclude with Propositions 5.1 and 5.3 by using any ordering of the 2_ps. $\square$

10.3. **Organising the closures.** We proceed in two steps to organise the closures.

Lemma 10.6. *We have* $(1, 2, 3)^\star = 3^\star 2^\star 1^\star$.

Proof. We follow the same path as in the proof of Lemma 10.2: from Table 5 we first deduce $1(2, 3) = 12 \cup 13 \subseteq 332 \cup 31 \subseteq (2, 3)^\star 1^=$ (since 1 is linear by Lemma 3.1 for the first step), whence $(1, 2, 3)^\star = (2, 3)^\star 1^\star$ by Lemma 3.17(1). Then we have $(2, 3)^\star = 3^\star 2^\star$ by Lemma 3.17, since $23 \subseteq 32$. The announced equality follows. □

Lemma 10.7. *We have* $\mathsf{netkat'}^\star = (0, \mathsf{s}, 2, 3, 4, 5, 6, 7)^\star = (3, 7)^\star 6^\star 5^\star 4^\star 3^\star 2^\star (0, \mathsf{s})^\star$.

Proof. We have
$$
\begin{aligned}
\mathsf{netkat'}^\star &= (0, 1, 2, 3, 4, 5, 6, 7)^\star \\
&= 7^\star 6^\star 5^\star 4^\star (1, 2, 3)^\star 0^\star &&\text{(by Proposition 5.3)} \\
&= 7^\star 6^\star 5^\star 4^\star 3^\star 2^\star 1^\star 0^\star &&\text{(by Lemma 10.6)} \\
&\subseteq (3, 7)^\star 6^\star 5^\star 4^\star 3^\star 2^\star (0, \mathsf{s})^\star \\
&\subseteq (0, \mathsf{s}, 2, 3, 4, 5, 6, 7)^\star = \mathsf{netkat'}^\star
\end{aligned}
$$

The first equality is by definition, the last one comes from the equality of the underlying sets (since $1, 2, 3 = \mathsf{s}, 2$). The last two inclusions follow from basic closure properties and $0, 1 \subseteq 0, \mathsf{s}$. The application of Proposition 5.3 is justified by Table 5, where we consider $1, 2, 3$ as a single set of hypotheses. This amounts to merging the corresponding lines and columns in the table, and we observe that all subsequent columns satisfy the requirements (choosing the second alternative for columns 6,7). □

At this point we can easily conclude.

Theorem 10.8. netkat *reduces to the empty set;* $\mathsf{KA}_{\mathsf{netkat}}$ *is complete and decidable.*

Proof. netkat reduces to $\mathsf{netkat'} = (0, \mathsf{s}, 2, 3, 4, 5, 6, 7)$, and we have reductions to the empty set for $0, \mathsf{s}$ by Lemma 10.3, for 2 by Lemma 10.5, and for 3, 4, 5, 6, and $(3, 7)$ by Lemma 3.8. Given Lemma 10.7, Proposition 5.1 applies. □

11. Related work

There is a range of papers on completeness and decidability of Kleene algebra together with specific forms of hypotheses, starting with [Coh94]. The general case of Kleene algebra with hypotheses, and reductions to prove completeness, has been studied recently in [KBS+20, DKPP19, KM14]. The current paper combines and extends these results, and thereby aims to provide a comprehensive overview and a showcase of how to apply these techniques to concrete case studies (KAT, KAO, NetKAT and the new theories KATF, KAT° and KAPT). Below, we discuss each of these recent works in more detail.

Kozen and Mamouras [KM14] consider restricted forms of hypotheses in terms of rewriting systems, and provide reductions for equations of the form $1 = w$ and $a = w$ (cf. Lemma 3.8). Their general results cover completeness results which instantiate to KAT and NetKAT. In fact, the assumptions made in their technical development are tailored towards these cases; for instance, their assumption $\alpha\beta \leq \perp$ (in Assumption 2) would have to be dropped to consider KAPT. The current paper focuses on generality and how to construct reductions in a modular way.

Doumane et al. [DKPP19] define the language interpretation of regular expressions for an arbitrary set of hypotheses via the notion of H-closure (Definition 2.2), and study

(un)decidability of the (in)equational theory of KA_H, i.e., the problem of checking $\mathsf{KA}_H \vdash e \leq f$, for various types of hypotheses H. In particular, they construct a reduction for hypotheses of the form $1 \leq \sum_{a \in S} a$ (cf. Lemma 3.8). A first step towards modularity may also be found in [DKPP19, Proposition 3].

Kappé et al. [KBS+20] study hypotheses on top of bi-Kleene algebra, where the canonical interpretation is based on pomset languages, and ultimately prove completeness of *concurrent Kleene algebra with observations*; many of the results there apply to the word case as well. We follow this paper for the basic definitions and results about reductions, with a small change in the actual definition of a reduction (Remark 3.5). Compositionality in the sense of Section 3.5 is treated in Kappé's PhD thesis [Kap20]. In the current paper we systematically investigate tools for combining reductions, based on lattice theory, proposing a number of new techniques (e.g. Lemma 3.17 and Sections 5.2 to 5.4). Further, we highlight the word case in this paper (as opposed to the pomset languages in concurrent Kleene algebra), by showcasing several examples.

12. CONCLUSIONS AND FUTURE WORK

We presented a general toolbox for proving completeness of Kleene algebra with hypotheses. While our examples demonstrate the rather wide applicability of our techniques, there are natural extensions that we have not covered here. For instance KAT+B! [GKM14] and KAO with a full element and/or converse seem to fit our framework. We hope we have provided enough tools and examples such that these theories can be investigated easily and in a principled way in the future.

Of course, there are Kleene algebra extensions that we cannot cover, a priori. For instance, for *action algebras* [Pra90] and *action lattices* [Koz94], it is not clear how to interpret the new operations as letters with additional structure. Finding reference models and proving completeness for such theories remains an important challenge.

There are also variations of Kleene algebra where some of the axioms weakened or removed, such as left-handed Kleene algebra [KS12, DDP18] (where one of the star-induction rules is removed) or Kleene algebra with abnormal termination [Mam17] (where the axiom $x0 = 0$ is no longer valid). In the former case, the equational theory remains the same, so that except for some of the basic reductions (e.g., Lemma 3.8(iv)), most of the framework we developed here can be reused. In the latter case instead, it is not clear how to proceed since the remaining axioms are no longer complete w.r.t. the language interpretation.

The general theory proposed here results in completeness with respect to a canonical language model, defined via language closure. In several instances however, there are other reference models for which we would like to obtain completeness. For instance, in the case of KAT, we provided a separate argument to relate the canonical model defined via closure to the model of guarded strings, which is a standard model of KAT. For NetKAT, there is a similar model of guarded strings, which we have not considered in our treatment. It would be interesting to identify a common pattern, and to try and develop generic tools that help transporting our results to such reference models.

Another direction of future work is that of decidability and complexity. Whenever our reduction technique yields a proof of completeness, and the reduction itself is computable (as it is in all examples), this immediately entails a decidability result, via decidability of KA (Theorem 2.12). However, in general this is far from an efficient procedure: computing the reduction itself can lead to a blow-up. Developing tools that help to obtain more efficient

decision procedures is left for future work. This problem may well be related to the problem discussed in the previous paragraph, on identifying suitable reference models.

References

[AFG⁺14] Carolyn Jane Anderson, Nate Foster, Arjun Guha, Jean-Baptiste Jeannin, Dexter Kozen, Cole Schlesinger, and David Walker. NetKAT: semantic foundations for networks. In *POPL*, pages 113–126. ACM, 2014. `doi:10.1145/2535838.2535862`.

[AK01] A. Angus and D. Kozen. Kleene algebra with tests and program schematology. Technical Report TR2001-1844, CS Dpt., Cornell University, July 2001. URL: `http://hdl.handle.net/1813/5831`.

[BB70] Garrett Birkhoff and Thomas C. Bartee. *Modern applied algebra*. McGraw-Hill, 1970.

[BÉS95] Stephen L. Bloom, Zoltán Ésik, and Gheorghe Stefanescu. Notes on equational theories of relations. *Algebra Universalis*, 33(1):98–126, 1995. `doi:10.1007/BF01190768`.

[Bof90] Maurice Boffa. Une remarque sur les systèmes complets d'identités rationnelles. *Informatique Théorique et Applications*, 24:419–428, 1990. URL: `http://archive.numdam.org/article/ITA_1990__24_4_419_0.pdf`.

[BP10] Thomas Braibant and Damien Pous. An efficient Coq tactic for deciding Kleene algebras. In *1st ITP*, volume 6172 of *LNCS*, pages 163–178. Springer, 2010. `doi:10.1007/978-3-642-14052-5_13`.

[BPS17] Paul Brunet, Damien Pous, and Georg Struth. On decidability of Concurrent Kleene Algebra. In *CONCUR*, volume 85 of *LIPIcs*, pages 24:1–24:16. Schloss Dagstuhl, 2017. `doi:10.4230/LIPIcs.CONCUR.2017.28`.

[Bru19] Paul Brunet. A note on commutative Kleene algebra. *CoRR*, abs/1910.14381, 2019. URL: `http://arxiv.org/abs/1910.14381`.

[Coh94] E. Cohen. Hypotheses in Kleene algebra. Technical report, Bellcore, N.J., 1994. URL: `http://www.researchgate.net/publication/2648968_Hypotheses_in_Kleene_Algebra`.

[Con71] J.H. Conway. *Regular Algebra and Finite Machines*. Chapman and Hall mathematics series. Chapman and Hall, 1971. Reedited by Dover Publications, Incorporated. URL: `https://books.google.nl/books?id=1KAXc5TpEV8C`.

[DDP18] Anupam Das, Amina Doumane, and Damien Pous. Left-handed completeness for Kleene algebra, via cyclic proofs. In *LPAR*, volume 57 of *EPiC Series in Computing*, pages 271–289. Easychair, 2018. `doi:10.29007/hzq3`.

[DKPP19] Amina Doumane, Denis Kuperberg, Pierre Pradic, and Damien Pous. Kleene algebra with hypotheses. In *FoSSaCS*, volume 11425 of *LNCS*, pages 207–223. Springer, 2019. `doi:10.1007/978-3-030-17127-8_12`.

[DP90] B. Davey and H. Priestley. *Introduction to Lattices and Order*. Cambridge University Press, 1990.

[ÉB95] Zoltán Ésik and Laszlo Bernátsky. Equational properties of Kleene algebras of relations with conversion. *T.C.S.*, 137(2):237–251, 1995. `doi:10.1016/0304-3975(94)00041-G`.

[GKM14] Niels Bjørn Bugge Grathwohl, Dexter Kozen, and Konstantinos Mamouras. KAT + B! In *CSL-LICS*. ACM, 2014. `doi:10.1145/2603088.2603095`.

[Har05] Christopher Hardin. Modularizing the elimination of r=0 in Kleene algebra. *LMCS*, 1(3), 2005. `doi:10.2168/LMCS-1(3:4)2005`.

[HK02] C. Hardin and D. Kozen. On the elimination of hypotheses in Kleene algebra with tests. Technical Report TR2002-1879, CS Dpt., Cornell University, October 2002. URL: `http://hdl.handle.net/1813/5855`.

[HMSW11] T. Hoare, B. Möller, G. Struth, and I. Wehrman. Concurrent Kleene algebra and its foundations. *Journal of Logic and Algebraic Programming*, 80(6):266–296, 2011. `doi:10.1016/j.jlap.2011.04.005`.

[HRS76]   Harry B. Hunt, Daniel J. Rosenkrantz, and Thomas G. Szymanski. On the equivalence, contain-
           ment, and covering problems for the regular and context-free languages. *Journal of Computer
           and System Sciences*, 12(2):222–268, 1976. `doi:10.1016/S0022-0000(76)80038-4`.
[Kap20]   Tobias Kappé. *Concurrent Kleene Algebra: Completeness and Decidability*. PhD thesis, UCL,
           London, 2020.
[KBR+19]   Tobias Kappé, Paul Brunet, Jurriaan Rot, Alexandra Silva, Jana Wagemaker, and Fabio Zanasi.
           Kleene algebra with observations. In *CONCUR*, pages 41:1–41:16, 2019. `doi:10.4230/LIPIcs.
           CONCUR.2019.41`.
[KBS+20]   Tobias Kappé, Paul Brunet, Alexandra Silva, Jana Wagemaker, and Fabio Zanasi. Concurrent
           Kleene algebra with observations: From hypotheses to completeness. In *FoSSaCS*, volume 12077
           of *LNCS*, pages 381–400. Springer, 2020. `doi:10.1007/978-3-030-45231-5_20`.
[Kle56]   S. C. Kleene. Representation of events in nerve nets and finite automata. In *Automata Studies*,
           pages 3–41. Princeton University Press, 1956. URL: `http://www.rand.org/pubs/research_
           memoranda/2008/RM704.pdf`.
[KM14]   Dexter Kozen and Konstantinos Mamouras. Kleene algebra with equations. In *ICALP*, volume
           8573 of *LNCS*, pages 280–292. Springer, 2014. `doi:10.1007/978-3-662-43951-7_24`.
[KN12]   A. Krauss and T. Nipkow. Proof pearl: Regular expression equivalence and relation algebra.
           Journal of Automated Reasoning, 49(1):95–106, 2012. `doi:10.1007/s10817-011-9223-4`.
[Koz91]   Dexter Kozen. A completeness theorem for Kleene Algebras and the algebra of regular events. In
           LICS, pages 214–225. IEEE Computer Society, 1991. `doi:10.1109/LICS.1991.151646`.
[Koz94]   D. Kozen. On action algebras. In *Logic and Inf. Flow*, pages 78–88. MIT Press, 1994.
[Koz96]   Dexter Kozen. Kleene algebra with tests and commutativity conditions. In Tiziana Margaria
           and Bernhard Steffen, editors, *TACAS*, volume 1055 of *LNCS*, pages 14–33. Springer, 1996.
           `doi:10.1007/3-540-61042-1\_35`.
[Koz97]   D. Kozen. Kleene algebra with tests. *Transactions on Programming Languages and Systems*,
           19(3):427–443, May 1997. `doi:10.1145/256167.256195`.
[Koz00]   D. Kozen. On Hoare logic and Kleene algebra with tests. *ACM Trans. Comput. Log.*, 1(1):60–76,
           2000. `doi:10.1145/343369.343378`.
[Koz02]   D. Kozen. On the complexity of reasoning in Kleene algebra. *I&C*, 179, 2002. `doi:10.1006/inco.
           2001.2960`.
[KP00]   D. Kozen and M.-C. Patron. Certification of compiler optimizations using Kleene algebra with
           tests. In *CL2000*, LNAI, pages 568–582. Springer, 2000. `doi:10.1007/3-540-44957-4_38`.
[Kro91]   D. Krob. Complete systems of B-rational identities. *T.C.S.*, 89(2):207–343, 1991. `doi:10.1016/
           0304-3975(91)90395-I`.
[KS96]   D. Kozen and F. Smith. Kleene algebra with tests: Completeness and decidability. In *CSL*, volume
           1258 of *LNCS*, pages 244–259. Springer, September 1996. `doi:10.1007/3-540-63172-0_43`.
[KS12]   Dexter Kozen and Alexandra Silva. Left-handed completeness. In *RAMiCS*, volume 7560 of
           LNCS, pages 162–178. Springer, 2012. `doi:10.1007/978-3-642-33314-9\_11`.
[Kuz18]   Stepan L. Kuznetsov. *-continuity vs. induction: Divide and conquer. In *Advances in Modal Logic*,
           pages 493–510. College Publications, 2018. URL: `http://www.aiml.net/volumes/volume12/
           Kuznetsov.pdf`.
[Mam17]   Konstantinos Mamouras. Equational theories of abnormal termination based on Kleene algebra. In
           FOSSACS, volume 10203 of *LNCS*, pages 88–105, 2017. `doi:10.1007/978-3-662-54458-7\_6`.
[MS72]   A.R. Meyer and L. J. Stockmeyer. The equivalence problem for regular expressions with squaring
           requires exponential space. In *SWAT*, pages 125–129. IEEE, 1972. `doi:10.1109/SWAT.1972.29`.
[O'H20]   Peter W. O'Hearn. Incorrectness logic. *Proc. ACM Program. Lang.*, 4(POPL):10:1–10:32, 2020.
           `doi:10.1145/3371078`.
[Pou13]   Damien Pous. Kleene Algebra with Tests and Coq tools for while programs. In *ITP*, volume
           7998 of *LNCS*, pages 180–196. Springer, 2013. `doi:10.1007/978-3-642-39634-2_15`.
[Pra90]   V. Pratt. Action logic and pure induction. In *JELIA*, volume 478 of *LNCS*, pages 97–120. Springer,
           1990. `doi:10.1007/BFb0018436`.
[PRW21]   Damien Pous, Jurriaan Rot, and Jana Wagemaker. On tools for completeness of Kleene algebra
           with hypotheses. In *RAMiCS*, volume 13027 of *LNCS*, pages 378–395. Springer, 2021. `doi:
           10.1007/978-3-030-88701-8_23`.

[PW22]   Damien Pous and Jana Wagemaker. Completeness theorems for Kleene algebra with top. In *CONCUR*, pages 26:1–26:18. Schloss Dagstuhl, 2022. `doi:10.4230/LIPIcs.CONCUR.2022.26`.

[Red64]   V.N. Redko. On the algebra of commutative events. *Ukrain. Mat*, (16):185–195, 1964.

[ZdAG22]   Cheng Zhang, Arthur Azevedo de Amorim, and Marco Gaboardi. On incorrectness logic and Kleene algebra with top and tests. *Proc. ACM Program. Lang.*, 6(POPL):1–30, 2022. `doi:10.1145/3498690`.

APPENDIX A. ON LEAST CLOSURES IN COMPLETE LATTICES

For a set of hypotheses H, the H-closure $H^\star$ is formally a closure operator: for all languages L we have $L \subseteq H^\star(L)$ and $H^\star H^\star(L) \subseteq H^\star(L)$. In this appendix, we use basic lattice theory [DP90] to establish various properties of such closure operators.

Let $\langle X, \leq, \bigvee \rangle$ be a complete lattice. We write $x + y$ for binary joins $\bigvee \{x, y\}$. We write 1 for the identity function on X. Given two functions f, g, we write fg for their composition: $fg(x) = f(g(x))$, and $f \leq g$ when $\forall x, f(x) \leq g(x)$. The finite iterations f^i of a function f are defined by induction on $i \in \mathbb{N}$: $f^0 = 1$ and $f^{i+1} = ff^i$. Functions on X, ordered pointwise as above, form a complete lattice where suprema are also computed pointwise.

A function f is *monotone* if $\forall x, y, x \leq y \Rightarrow f(x) \leq f(y)$. By Knaster-Tarski's theorem, every monotone function f admits a *least (pre)fixpoint* μf, which satisfies $f(\mu f) \leq \mu f$ and the *induction principle*: $\forall x, f(x) \leq x \Rightarrow \mu f \leq x$.

A.1. **Least closures.** A *closure* is a monotone function c such that $1 \leq c$ and $cc \leq c$.

Given a monotone function s and an element $x \in X$, we write $s^\star(x)$ for the least (pre)fixpoint of s above x: $s^\star(x) \triangleq \mu(\lambda y.x + s(y))$. This definition gives, for all x, y,

$$x + s(s^\star(x)) \leq s^\star(x) \tag{A.1}$$

$$x + s(y) \leq y \Rightarrow s^\star(x) \leq y \tag{A.2}$$

Equivalently we have that for all functions g, h,

$$1 + ss^\star \leq s^\star \tag{A.1'}$$

$$g + sh \leq h \Rightarrow s^\star g \leq h \tag{A.2'}$$

Lemma A.1. *If f, g are two monotone functions such that $f \leq g$, then $\mu f \leq \mu g$.*

Proof. Under the assumption, every pre-fixpoint of g is a pre-fixpoint of f. □

Proposition A.2. *For every monotone function s, $s^\star$ is the least closure above s.*

Proof. We first prove that $s^\star$ is a closure:
- monotonicity follows from Lemma A.1;
- $1 \leq s^\star$ follows from (A.1');
- $s^\star s^\star \leq s^\star$ follows from $s^\star + ss^\star \leq s^\star$ by (A.2'), which holds by (A.1');

Moreover, $s^\star$ is above s: we have $s \leq ss^\star \leq s^\star$ by $1 \leq s^\star$, monotonicity of s and (A.1').

Now, if c is a closure above s, then $s^\star \leq c$ follows from $1 + sc \leq c$ by (A.2'), which holds thanks to the assumption $s \leq c$ and the fact that c is a closure. □

Lemma A.3. *Let s, t be two monotone functions. If $s \leq t$ then $s^\star \leq t^\star$.*

Proof. Direct consequence of Lemma A.1. □

Proposition A.4. *If s, t are monotone functions such that $s^\star t^\star \leq t^\star s^\star$, then $(s+t)^\star = t^\star s^\star$.*

Proof. Set $c = t^\star s^\star$. First c is a closure: we have $1 \leq t^\star \leq t^\star s^\star$ and $t^\star s^\star t^\star s^\star \leq t^\star t^\star s^\star s^\star \leq t^\star s^\star$. Since c is above both s and t, we get $(s+t)^\star \leq c$ by Proposition A.2. Finally, $c = t^\star s^\star \leq (s+t)^\star (s+t)^\star \leq (s+t)^\star$ using Lemma A.3 twice for the first inequality. ☐

Proposition A.5. *Let $s_1, \ldots, s_n$ be monotone functions; write c_j for $s_j^\star$, and $c_{<j}$ for $\left(\bigvee_{1 \leq i < j} s_i\right)^\star$. Suppose that for all $1 < j \leq n$, we have $c_{<j} c_j \leq c_j c_{<j}$. Then $c_{<n+1} = c_n \ldots c_1$.*

(Note that for $n = 2$, the statement amounts to Proposition A.4.)

Proof. We prove by induction on j that for all $1 \leq j \leq n$, $c_{<j+1} = c_j \ldots c_1$. The case $j = 1$ is trivial. For the inductive case, suppose $1 < j \leq n$ and let $s = \bigvee_{1 \leq i < j} s_i$. By assumption, we have $s^\star s_j^\star = c_{<j}\, c_j \leq c_j\, c_{<j} = s_j^\star s^\star$. Therefore, we deduce

$$
\begin{aligned}
c_{<j+1} &= (s + s_j)^\star && \text{(by definition)}\\
&= s_j^\star s^\star && \text{(by Proposition A.4)}\\
&= c_j\, c_{<j} && \text{(by definition)}\\
&= c_j\, c_{j-1} \ldots c_1 && \text{(by induction)} \quad\square
\end{aligned}
$$

Proposition A.6. *Let s, f be monotone functions, and c a closure. We have:*

$$ sf \leq fc \Rightarrow s^\star f \leq fc $$

Proof. By (A.2'), it suffices to show $f + sfc \leq fc$. We have $f \leq fc$ by monotonicity of f and $1 \leq c$. We have $sfc \leq fcc \leq fc$ by assumption, monotonicity of f, and $cc \leq c$. ☐

A function f admits another function $f^\sharp$ as *upper adjoint* if $\forall x, y,\ f(x) \leq y \Leftrightarrow x \leq f^\sharp(y)$. Equivalently, for all functions g, h,

$$ fg \leq h \Leftrightarrow g \leq f^\sharp h \tag{A.3} $$

In such a situation, f and $f^\sharp$ are monotone, and we have

$$ f f^\sharp \leq 1 \leq f^\sharp f \tag{A.4} $$

A function f is *linear* if it preserves all joins: $\forall Y \subseteq X, f(\bigvee_{y \in Y} y) = \bigvee_{y \in Y} f(y)$; it is *affine* if it preserves all non-empty joins (i.e., $Y \neq \emptyset$ in the previous formula).

Linear functions are affine; affine functions are monotone; the set of linear (resp. affine) functions is closed under composition and arbitrary joins; all constant functions are affine, and only the least one is linear. Moreover, we have the following characterisations:

Lemma A.7.
  (i) *A function is linear if and only if it admits an upper adjoint.*
  (ii) *A function is affine if and only if it is the join of a constant and a linear function.*

Proof. The first item is [DP90, Proposition 7.34]. For the direct implication in the second one, let f be an affine function. Set $a = f(\bot)$, and define $f'(x) = \begin{cases} \bot & \text{if } x = \bot \\ f(x) & \text{otherwise} \end{cases}$.

We have $f(x) = a + f'(x)$ for all x, and f' is linear. The converse implication is immediate. ☐

We obtain a symmetrical version of Proposition A.6 by restricting to affine functions.

Proposition A.8. *Let s be a monotone function, f an affine function, and c a closure. We have:*

$$fs \le cf \Rightarrow fs^\star \le cf$$

Proof. We first prove the statement when f is linear. In such a case f has an upper adjoint $f^\sharp$, and we have:

$$
\begin{aligned}
fsf^\sharp cf &\le cff^\sharp cf && \text{(by assumption)} \\
&\le ccf && \text{(by (A.4))} \\
&\le cf && (c \text{ a closure})
\end{aligned}
$$

By (A.3), we deduce $sf^\sharp cf \le f^\sharp cf$. We also have $1 \le f^\sharp f \le f^\sharp cf$ by (A.4) and $1 \le c$. Therefore we get $s^\star \le f^\sharp cf$ by (A.2'), which means $fs^\star \le cf$ by (A.3).

Now, if f is only affine, then $f = x \mapsto a + f'(x)$ for some constant a and some linear function f', by Lemma A.7. The assumption $fs \le cf$ at bottom gives us $a \le c(a)$. Define $c'(x) = c(a + x)$. This function is a closure: we have $1 \le c \le c'$ and $c'(c'(x)) = c(a + c(a + x)) \le c(c(a) + c(a + x)) \le c(c(a + x)) \le c(a + x) = c'(x)$. Also observe that $cf = c'f'$. We have $f's \le fs \le cf = c'f'$ so that we can apply the linear version of the lemma to s, f' and c', to deduce $f's^\star \le c'f' = cf$. Combined with $a \le c(a)$, it follows that $fs^\star \le cf$. $\qquad\square$

Remark A.9. *Propositions A.6 and A.8 easily generalise to the case where s and c are functions on two distinct lattices X and Y, and f is a function between those lattices (from Y to X for Proposition A.6, and from X to Y for Proposition A.8. We actually use this more general form of the latter proposition in the proof of Proposition 3.2.*

Lemma A.10. *If s is affine, so is $s^\star$.*

Proof. As above, we first prove statement for linear functions s, using the characterisation of linear functions as those admitting an upper adjoint (Lemma A.7).

When given a monotone function g, let $g^\circ(y)$ be the greatest (post-)fixpoint of $\lambda x.y \wedge g(x)$. This construction is dual to the construction of the least closure above a monotone function, and yields the "largest coclosure" below g.

Assume that s admits t as upper adjoint; we show that $s^\star$ admits t° as upper adjoint. Suppose $s^\star(x) \le y$. Since $s(s^\star(x)) \le s^\star(x)$, the adjoint property (A.3) gives $s^\star(x) \le t(s^\star(x))$. Therefore, $s^\star(x)$ is a post-fixpoint of $\lambda x.y \wedge t(x)$, so that $s^\star(x) \le t^\circ(y)$. Since $x \le s^\star(x)$, we deduce $x \le t^\circ(y)$ by transitivity. The converse implication $(\forall x, y, \ x \le t^\circ(y) \Rightarrow s^\star(x) \le y)$ holds by duality.

Now, if s is only affine, then $s = a + f$ for some constant a and some linear function f, by Lemma A.7. We prove that $s^\star$ is affine by showing that $s^\star = f^\star(a) + f^\star$.

- We use (A.2') to show $s^\star \le f^\star(a) + f^\star$. Indeed, we have $1 \le f^\star \le f^\star(a) + f^\star$, and $s(f^\star(a) + f^\star) = a + f(f^\star(a)) + ff^\star \le f^\star(a) + f^\star$.
- For the converse inequation, we have $f^\star(a) \le s^\star(\bot)$ by (A.2), since $a \le s(\bot) \le s^\star(\bot)$ and $f(s^\star(\bot)) \le s(s^\star(\bot)) \le s^\star(\bot)$; and $f^\star \le s^\star$ follows from $f \le s$. $\qquad\square$

Given a function f, we write $f^=$ for the function $f + 1$ (i.e., $f^=(x) = f(x) + x$).

Lemma A.11. *For every monotone function s, we have $(s^=)^\star = s^\star$.*

Proof. Consequence of Proposition A.4, since $1^\star = 1$. $\qquad\square$

Proposition A.12. *Let s be affine and t be monotone.*
1. *If $st \leq t^\star s^=$ then $s^\star t^\star \leq t^\star s^\star$.*
2. *If $st \leq t^= s^\star$ then $s^\star t^\star \leq t^\star s^\star$.*

Proof. We first prove the statements where we remove the $\cdot^=$ in the assumptions:

1. assuming $st \leq t^\star s$, we deduce $st^\star \leq t^\star s$ by Proposition A.8 (s being affine, and $t^\star$ being a closure), whence $st^\star \leq t^\star s^\star$ since $s \leq s^\star$, and finally $s^\star t^\star \leq t^\star s^\star$ by Proposition A.6 ($s^\star$ being a closure).
2. assuming $st \leq ts^\star$, we deduce $s^\star t \leq ts^\star$ by Proposition A.6 ($s^\star$ being a closure), whence $s^\star t \leq t^\star s^\star$ since $t \leq t^\star$, and finally $s^\star t^\star \leq t^\star s^\star$ by Proposition A.8 ($s^\star$ being affine by Lemma A.10, and $t^\star$ being a closure).

Then we add the $\cdot^=$ back in the assumptions:

1. assume we have $st \leq t^\star s^=$; $s^=$ is affine, and we have $s^= t = st + t \leq t^\star s^= + t = t^\star s^=$. We can thus apply the first result above to $s^=$ and t, so as to obtain $(s^=)^\star t^\star \leq t^\star (s^=)^\star$.
2. assume we have $st \leq t^= s^\star$; we have $st^= = st + s$ since s is affine, and thus $st^= = st + s \leq t^= s^\star + s = t^= s^\star$. We can thus apply the second result above to s and $t^=$, so as to obtain $s^\star (t^=)^\star \leq (t^=)^\star s^\star$.

In both cases, we conclude by Lemma A.11. $\square$

Corollary A.13. *Let $s_1, \ldots, s_{n-1}$ be affine functions, and s_n a monotone function. Write c_j for $s_j^\star$, $s_{<j}$ for $\bigvee_{1 \leq i < j} s_i$, and $c_{<j}$ for $s_{<j}^\star$, like in Proposition A.5.*

If for all $1 < j \leq n$ we have either $\begin{cases} (1) \ \forall i < j, \ s_i s_j \leq c_j s_{<j}^=, \ or \\ (2) \ \forall i < j, \ s_i s_j \leq s_j^= c_{<j}. \end{cases}$ *then $c_{<n+1} = c_n \ldots c_1$.*

Proof. For all $1 < j \leq n$, we have that $s_{<j}$ is affine, and we deduce by summing the hypotheses that either $s_{<j}s_j \leq c_j s_{<j}^=$ or $s_{<j}s_j \leq s_j^= c_{<j}$; in both cases, Proposition A.12 applies so that we deduce $c_{<j}c_j \leq c_j c_{<j}$. We conclude with Proposition A.5. $\square$

A.2. **Contextual functions.** Let us assume that X actually is a *quantale*: a complete lattice which comes with a monoid $\langle X, \cdot, 1 \rangle$ whose multiplication ($\cdot$) distributes over all joins.

Definition A.14. A monotone function f on X is *contextual* if for all $x, y, z \in X$, we have $x \cdot f(y) \cdot z \leq f(x \cdot y \cdot z)$.

Lemma A.15. *Let c be a closure. The function c is contextual iff for all $x, y \in X$, we have $c(x) \cdot c(y) \leq c(x \cdot y)$.*

Proof. If c is a contextual closure, then $c(x) \cdot c(y) \leq c(c(x) \cdot y) \leq c(c(x \cdot y)) = c(x \cdot y)$ using contextuality twice. Conversely, we have $x \cdot c(y) \cdot z \leq c(x) \cdot c(y) \cdot c(z) \leq c(x \cdot y \cdot z)$ using extensivity (i.e., $1 \leq c$) twice, and then twice the assumption. $\square$

Fact A.16. *Contextual functions are closed under composition and arbitrary joins, and contain the identity function.*

Lemma A.17. *If s is contextual then so is $s^\star$.*

Proof. For $x, z \in X$, let $f_{x,z} : y \mapsto x \cdot y \cdot z$. A function s is contextual iff for all x, z, $f_{x,z}s \leq sf_{x,z}$. The functions $f_{x,z}$ are linear, so that we may apply Proposition A.8 to deduce that when s is contextual, we have $f_{x,z}s^\star \leq s^\star f_{x,z}$ for all x, z, i.e., $s^\star$ is contextual. $\square$

Now consider the quantale of languages.

Lemma A.18. *A monotone function f on languages is contextual iff for all words l, r and languages K, we have $l \cdot f(K) \cdot r \subseteq f(l \cdot K \cdot r)$.*

Proof. The left-to-right implication is trivial: take singleton languages $\{l\}$ and $\{r\}$ for x and z. For the converse implication, suppose that $u \in L \cdot f(K) \cdot R$: $u = lxr$ for words l, x, r respectively in $L, f(K)$ and R. We have in particular $lxr \in l \cdot f(K) \cdot r$. We deduce $lxr \in f(l \cdot K \cdot r)$ by assumption, whence $lxr \in f(L \cdot K \cdot R)$ by monotonicity. $\square$

Lemma A.19. *Let H be a set of hypotheses. The functions H and $H^\star$ are contextual.*

Proof. It suffices to prove that H is contextual by Lemma A.17, which is easy. $\square$

Lemma A.20. *Let H be a set of hypotheses. For all languages L, K we have*

$$H^\star(L) \cdot H^\star(K) \subseteq H^\star(L \cdot K).$$

Proof. By Lemmas A.15 and A.19. $\square$

Lemma A.21. *Let H be a set of hypotheses. For all languages L we have*

$$(H^\star(L))^* \subseteq H^\star(L^*).$$

Proof. It suffices to show $1 + H^\star(L) \cdot H^\star(L^*) \subseteq H^\star(L^*)$, which follows from Lemma 5.6. $\square$

Appendix B. Direct soundness proof

We give in this appendix a direct proof of soundness (Theorem 2.4, [DKPP19, Theorem 2]). It relies on the following lemma from [DKPP19].

Lemma B.1 [DKPP19, Lemma 2]**.** *Let H be a set of hypotheses. For all languages L, K, we have*

(1) $H^\star(L + K) = H^\star(H^\star(L) + H^\star(K))$,
(2) $H^\star(L \cdot K) = H^\star(H^\star(L) \cdot H^\star(K))$,
(3) $H^\star(L^*) = H^\star(H^\star(L)^*)$.

Proof. Via basic closure properties, Lemma A.20, and Lemma A.21. $\square$

Before proving soundness, let us also recall the following equivalence, which we use implicitly in the proof below: for all languages L, K,

$$H^\star(L + K) = H^\star(K) \quad \text{iff} \quad L \subseteq H^\star(K)$$

(This is useful to deal with inequations, since $e \leq f$ is defined as $e + f = f$.)

Theorem 2.4. $\mathsf{KA}_H \vdash e = f$ *implies* $H^\star[\![e]\!] = H^\star[\![f]\!]$.

Proof. By induction on the derivation.
- For KA equational axioms, we actually have $[\![e]\!] = [\![f]\!]$.
- For an inequation $e \leq f \in H$, we have $[\![e]\!] \subseteq H[\![f]\!]$ by definition, whence $[\![e]\!] \subseteq H^\star[\![f]\!]$.
- The reflexivity, symmetry, and transitivity rules are easy to handle.

- The contextuality rules are dealt with Lemma B.1. For instance, for $+$, suppose $\mathsf{KA}_H \vdash e + f = e' + f'$ is obtained from $\mathsf{KA}_H \vdash e = e'$ and $\mathsf{KA}_H \vdash f = f'$, so that $H^\star[\![e]\!] = H^\star[\![e']\!]$ and $H^\star[\![f]\!] = H^\star[\![f']\!]$ by induction. We derive:

$$H^\star[\![e + f]\!] = H^\star(H^\star[\![e]\!] + H^\star[\![f]\!]) \qquad\qquad \text{(Lemma B.1(1))}$$
$$= H^\star(H^\star[\![e']\!] + H^\star[\![f']\!]) \qquad\qquad \text{(IH)}$$
$$= H^\star[\![e' + f']\!] \qquad\qquad \text{(Lemma B.1(1))}$$

- It remains to deal with the two implications of KA. Suppose $\mathsf{KA}_H \vdash f^*e \le g$ is obtained from $\mathsf{KA}_H \vdash e + fg \le g$, so that $[\![e + fg]\!] \subseteq H^\star[\![g]\!]$ by induction. We have to show $[\![f^*e]\!] = [\![f]\!]^* [\![e]\!] \subseteq H^\star[\![g]\!]$, which we obtain from

$$[\![e]\!] + [\![f]\!] H^\star[\![g]\!] \subseteq [\![e]\!] + H^\star[\![f]\!] H^\star[\![g]\!]$$
$$\subseteq [\![e]\!] + H^\star[\![fg]\!] \qquad\qquad \text{(Lemma A.20)}$$
$$\subseteq H^\star[\![e + fg]\!] \subseteq H^\star[\![g]\!] \qquad\qquad \text{(IH)}$$

The other implication is dealt with symmetrically. □

Appendix C. Guarded string interpretation of $\mathsf{kat}^\star[\![-]\!]$

We give a detailed proof of Corollary 4.6 in this appendix, following the path sketched in Section 4.1. This result relates the (standard) guarded string interpretation of KAT to the kat-closed language interpretation we use in the present paper.

Lemma 4.3. *Let L be a language of guarded strings. We have*

$$\phi_0 a_0 \cdots \phi_{n-1} a_{n-1} \phi_n \in \mathsf{kat}^\star(L)$$
$$\Leftrightarrow \qquad \forall\, (\alpha_i)_{i \le n}, \quad (\forall i \le n, \alpha_i \models \overline{\phi_i}) \quad \Rightarrow \quad \alpha_0 a_0 \cdots \alpha_{n-1} a_{n-1} \alpha_n \in L$$

Proof. The proof for the left to right direction proceeds by induction on the closure—c.f. (A.2). In the base case, where the word already belongs to L, and thus is already a guarded string, the ϕ_i must all be single atoms. Since $\alpha \models \beta$ iff $\alpha = \beta$ for all atoms α, β, the condition is immediately satisfied.

Otherwise, we have $\phi_0 a_0 \cdots \phi_{n-1} a_{n-1} \phi_n \in u [\![e]\!] v$ for $u, v \in (\Sigma + \mathcal{T}_{\mathsf{BA}})^*$ and $e \le f \in \mathsf{kat}$, where guarded strings in $u [\![f]\!] v$ satisfy the property by induction. We proceed by a case distinction on the hypothesis in $e \le f \in \mathsf{kat}$.

- $e \le f \in \mathsf{bool}$: in this case $e = \phi$ and $f = \psi$ for some formulas ϕ, ψ such that $\vdash_{\mathsf{BA}} \phi = \psi$. Therefore, we have $\alpha \models \phi$ iff $\alpha \models \psi$ for all atoms α, so that the induction hypothesis on the unique word of $u\psi v$ is equivalent to the property we have to prove about the unique word of $u\phi v$.
- $e = \phi \wedge \psi$ and $f = \phi \cdot \psi$: there must be j such that $\phi_j = \phi(\phi \wedge \psi)\psi$ for $\phi, \psi \in \mathcal{T}_{\mathsf{BA}}{}^*$. The word of $u\phi\psi v$ has the same decomposition as $u(\phi \wedge \psi)v$, except that ϕ_j is replaced by $\phi\phi\psi\psi$. Since $\vdash_{\mathsf{BA}} \overline{\phi\phi\psi\psi} = \overline{\phi(\phi \wedge \psi)\psi}$, the induction hypothesis immediately applies.
- The case for $e = \phi \cdot \psi$ and $f = \phi \wedge \psi$ is handled similarly.
- $e = \phi \vee \psi$ and $f = \phi + \psi$: there must be j such that $\phi_j = \phi(\phi \vee \psi)\psi$ for $\phi, \psi \in \mathcal{T}_{\mathsf{BA}}{}^*$. We have $[\![f]\!] = \{\phi, \psi\}$ and the two words of $u [\![f]\!] v$ have the same decomposition as $u(\phi \vee \psi)v$, except that ϕ_j is replaced by $\phi\phi\psi$ or $\phi\psi\psi$. We use the induction hypothesis and the fact that $\alpha \models \phi \vee \psi$ iff $\alpha \models \phi$ or $\alpha \models \psi$.

- $e = \phi + \psi$ and $f = \phi \vee \psi$: there must be j such that w.l.o.g. $\phi_j = \phi(\phi)\psi$ for $\phi, \psi \in \mathcal{T}_{\mathsf{BA}}{}^*$. We have $[\![f]\!] = \{\phi \vee \psi\}$ and the word in $u\,[\![f]\!]\,v$ has the same decomposition as $u(\phi)v$, except that ϕ_j is replaced by $\phi(\phi \vee \psi)\psi$. We use the induction hypothesis and the fact that if $\alpha \models \phi$ then $\alpha \models \phi \vee \psi$.
- $e = \bot$ and $f = 0$: there must be j such that $\bot$ belongs to ϕ_j, so that there are no atoms α such that $\alpha \models \phi_j$: the condition is trivially satisfied.
- $e = 0$ and $f = \bot$: in this case $u\,[\![e]\!]\,v$ is empty so this case is trivially satisfied.
- $e = \top$ and $f = 1$: there is j, ϕ, ψ such that $\phi_j = \phi\top\psi$. The word of uv decomposes like $u\top v$, except that ϕ_j is replaced by $\phi\psi$. Since $\vdash_{\mathsf{BA}} \overline{\phi\psi} = \overline{\phi\top\psi}$, the induction hypothesis suffices to conclude.
- The case where $e = 1$ and $f = \top$ is handled similarly.

We now prove the right-to-left implication. From the assumptions we obtain

$$\left[\!\!\left[\sum_{\alpha \models \overline{\phi_0}} \alpha\right]\!\!\right] a_0 \ldots \left[\!\!\left[\sum_{\alpha \models \overline{\phi_{n-1}}} \alpha\right]\!\!\right] a_{n-1} \left[\!\!\left[\sum_{\alpha \models \overline{\phi_n}} \alpha\right]\!\!\right] \subseteq L$$

We deduce using the glueing inequations for $\vee$ and $\bot$ that

$$\left(\bigvee_{\alpha \models \overline{\phi_0}} \alpha\right) a_0 \ldots \left(\bigvee_{\alpha \models \overline{\phi_{n-1}}} \alpha\right) a_{n-1} \left(\bigvee_{\alpha \models \overline{\phi_n}} \alpha\right) \in \mathsf{kat}^\star(L)$$

Since $\vdash_{\mathsf{BA}} \phi = \bigvee_{\alpha \models \phi} \alpha$ for all ϕ, the inequations from bool yield

$$\overline{\phi_0} a_0 \ldots \overline{\phi_{n-1}} a_{n-1} \overline{\phi_n} \in \mathsf{kat}^\star(L)$$

We conclude with the glueing inequations for $\wedge$ and $\top$, which give

$$\phi_0 a_0 \ldots \phi_{n-1} a_{n-1} \phi_n \in \mathsf{kat}^\star(L) \qquad\qquad \square$$

Now we turn to showing that the kat-closures of $[\![e]\!]$ and $\mathcal{G}(e)$ coincide (Lemma 4.4), ultimately yielding Corollary 4.6. We first prove two lemmas about kat-closures of guarded string languages.

Lemma C.1. *For all guarded string languages L, K, $\mathsf{kat}^\star(L \cdot K) \subseteq \mathsf{kat}^\star(L \diamond K)$.*

Proof. We show that $L \cdot K \subseteq \mathsf{kat}^\star(L \diamond K)$, which is sufficient. Take $w \in L \cdot K$. Hence, $w = xy$ with $x \in L$ and $y \in K$. Because x and y are guarded strings, we know both of them begin and end in an atom. So we can write $x = x'\alpha$ and $y = \beta y'$ for some $\alpha, \beta \in \mathsf{At}$.

- If $\alpha = \beta$, then $x\alpha y \in L \diamond K$. In that case, we have $xy = x'\alpha\alpha y'$, but then, since x and y are guarded strings and α is the only atom s.t. $\alpha \models \alpha \wedge \alpha$, we get $xy \in \mathsf{kat}^\star(L \diamond K)$ by Lemma 4.3.
- Otherwise, if $\alpha \neq \beta$, then *a priori* $xy \notin L \diamond K$. However, since there are no atoms γ s.t. $\gamma \models \alpha \wedge \beta$, we have $xy \in \mathsf{kat}^\star(L \diamond K)$ by Lemma 4.3. $\qquad \square$

Given a guarded string language L, let us denote by $L^\diamond$ the language L iterated w.r.t. the coalesced product $\diamond$: $L^\diamond = \bigcup_n L^{\diamond n}$, where $L^{\diamond 0} = \{\alpha \mid \alpha \in \mathsf{At}\}$ and $L^{\diamond n+1} = L \diamond L^{\diamond n}$.

Lemma C.2. *For all guarded string languages L, $\mathsf{kat}^\star(L^*) \subseteq \mathsf{kat}^\star(L^\diamond)$.*

Proof. It suffices to prove $L^* \subseteq \mathsf{kat}^\star(L^\diamond)$, which in turn follows once we prove $\varepsilon \in \mathsf{kat}^\star(L^\diamond)$ and $L \cdot \mathsf{kat}^\star(L^\diamond) \subseteq \mathsf{kat}^\star(L^\diamond)$. The former follows from the fact that $\mathsf{kat} \vdash 1 \leq \sum_\alpha \alpha$, since $L^\diamond$ contains $L^{\diamond 0}$ and thus all atoms α. For the latter, we have

$$L \cdot \mathsf{kat}^\star(L^\diamond) \subseteq \mathsf{kat}^\star(L \cdot L^\diamond) \qquad\qquad (\mathsf{kat}^\star \text{ is contextual—Lemma 5.5})$$
$$\subseteq \mathsf{kat}^\star(L \diamond L^\diamond) \qquad\qquad\qquad\qquad (\text{Lemma C.1})$$
$$\subseteq \mathsf{kat}^\star(L^\diamond) \qquad\qquad\qquad\qquad\qquad\qquad \square$$

We can finally prove Lemma 4.4:

Lemma 4.4. *For all KAT expressions e, $\mathsf{kat}^\star[\![e]\!] = \mathsf{kat}^\star(\mathcal{G}(e))$.*

Proof. It suffices to prove that $[\![e]\!] \subseteq \mathsf{kat}^\star(\mathcal{G}(e))$ and $\mathcal{G}(e) \subseteq \mathsf{kat}^\star[\![e]\!]$. We prove both results via induction on e.

We start with $[\![e]\!] \subseteq \mathsf{kat}^\star(\mathcal{G}(e))$.

- 0: trivial as $[\![0]\!] = \emptyset$.
- 1: $[\![1]\!] = \{\varepsilon\}$ and $\mathcal{G}(1) = \{\alpha \mid \alpha \in \mathsf{At}\}$, therefore $\varepsilon \in \mathsf{kat}^\star(\mathcal{G}(1))$ by Lemma 4.3.
- $a \in \Sigma$: $[\![a]\!] = \{a\}$ and $\mathcal{G}(a) = \{\alpha a \beta \mid \alpha, \beta \in \mathsf{At}\}$, therefore $a \in \mathsf{kat}^\star(\mathcal{G}(a))$ by Lemma 4.3.
- $\phi \in \mathcal{T}_{\mathsf{BA}}$: $[\![\phi]\!] = \{\phi\}$ and $\mathcal{G}(\phi) = \{\alpha \mid \alpha \models \phi\}$, hence $\phi \in \mathsf{kat}^\star(\mathcal{G}(\phi))$ by Lemma 4.3.
- $e + f$: we have

$$[\![e + f]\!] = [\![e]\!] \cup [\![f]\!] \subseteq \mathsf{kat}^\star(\mathcal{G}(e)) \cup \mathsf{kat}^\star(\mathcal{G}(f)) \qquad\qquad (\text{IH})$$
$$\subseteq \mathsf{kat}^\star(\mathcal{G}(e) \cup \mathcal{G}(f)) = \mathsf{kat}^\star(\mathcal{G}(e + f)) \qquad (\mathsf{kat}^\star \text{ monotone})$$

- $e \cdot f$: we have

$$[\![e \cdot f]\!] = [\![e]\!] \cdot [\![f]\!] \subseteq \mathsf{kat}^\star(\mathcal{G}(e)) \cdot \mathsf{kat}^\star(\mathcal{G}(f)) \qquad\qquad (\text{IH})$$
$$\subseteq \mathsf{kat}^\star(\mathcal{G}(e) \cdot \mathcal{G}(f)) \qquad\qquad\qquad\qquad (\text{Lemma 5.6})$$
$$\subseteq \mathsf{kat}^\star(\mathcal{G}(e) \diamond \mathcal{G}(f)) = \mathsf{kat}^\star(\mathcal{G}(e \cdot f)) \qquad\quad (\text{Lemma C.1})$$

- e^*: we have

$$[\![e^*]\!] = [\![e]\!]^* \subseteq \mathsf{kat}^\star(\mathcal{G}(e))^* \qquad\qquad\qquad\qquad (\text{IH})$$
$$\subseteq \mathsf{kat}^\star(\mathcal{G}(e)^*) \qquad\qquad\qquad\qquad (\text{Lemma A.21})$$
$$\subseteq \mathsf{kat}^\star(\mathcal{G}(e)^\diamond) = \mathsf{kat}^\star(\mathcal{G}(e^*)) \qquad\quad (\text{Lemma C.2})$$

Next we prove $\mathcal{G}(e) \subseteq \mathsf{kat}^\star[\![e]\!]$:

- 0: trivial as $\mathcal{G}(0) = \emptyset$.
- 1: $\mathcal{G}(1) = \{\alpha \mid \alpha \in \mathsf{At}\}$; for all α, $\mathsf{kat} \vdash \alpha \leq 1$, whence $\alpha \in \mathsf{kat}^\star\{\varepsilon\} = \mathsf{kat}^\star[\![1]\!]$.
- $a \in \Sigma$: $\mathcal{G}(a) = \{\alpha a \beta \mid \alpha, \beta \in \mathsf{At}\}$; like above, for all α, β, $\alpha a \beta \in \mathsf{kat}^\star\{a\} = \mathsf{kat}^\star[\![a]\!]$.
- $\phi \in \mathcal{T}_{\mathsf{BA}}$: $\mathcal{G}(\phi) = \{\alpha \mid \alpha \models \phi\}$, for all α such that $\alpha \models \phi$, $\mathsf{kat} \vdash \alpha \leq \phi$, whence $\alpha \in \mathsf{kat}^\star\{\phi\} = \mathsf{kat}^\star[\![\phi]\!]$.
- $e + f$: we have

$$\mathcal{G}(e + f) = \mathcal{G}(e) \cup \mathcal{G}(f)$$
$$\subseteq \mathsf{kat}^\star[\![e]\!] \cup \mathsf{kat}^\star[\![f]\!] \qquad\qquad (\text{IH})$$
$$\subseteq \mathsf{kat}^\star([\![e]\!] \cup [\![f]\!]) \qquad\qquad (\mathsf{kat}^\star \text{ monotone})$$
$$= \mathsf{kat}^\star[\![e + f]\!])$$

- $e \cdot f$: suppose $w \in \mathcal{G}(e \cdot f)$, i.e., $w = x\alpha y$ for $x\alpha \in \mathcal{G}(e)$ and $\alpha y \in \mathcal{G}(f)$. Via the induction hypothesis we know that $x\alpha \in \mathsf{kat}^\star[\![e]\!]$ and $\alpha y \in \mathsf{kat}^\star[\![f]\!]$. Hence $x\alpha\alpha y \in \mathsf{kat}^\star[\![e]\!] \cdot \mathsf{kat}^\star[\![f]\!] \subseteq \mathsf{kat}^\star[\![e \cdot f]\!]$ via Lemma 5.6. As $\mathsf{kat} \vdash \alpha \leq \alpha\alpha$, we get that $x\alpha y \in \mathsf{kat}^\star[\![e \cdot f]\!]$.
- e^*: a similar argument works. $\qquad\square$

Recall that $\mathcal{GS}$ is the set of all guarded strings. We deduce from Lemma 4.3 that closing a guarded string language under kat does not add new guarded strings:

Lemma C.3. *For all guarded string language L, $L = \mathsf{kat}^\star(L) \cap \mathcal{GS}$.*

Proof. The left-to-right direction is trivial, as $L \subseteq \mathsf{kat}^\star(L)$ and all strings in L are guarded. For the right-to-left direction, given a guarded string $\alpha_0 a_0 \ldots a_{n-1}\alpha_n$ in $\mathsf{kat}^\star(L)$, we simply use Lemma 4.3 with the sequence of α_i themselves: we have $\alpha_i \models \alpha_i$ for all i, so that $\alpha_0 a_0 \ldots a_{n-1}\alpha_n$ actually belongs to L. $\qquad\square$

Lemma 4.5. *For all KAT expressions e, $\mathcal{G}(e) = \mathsf{kat}^\star[\![e]\!] \cap \mathcal{GS}$.*

Proof. Immediate from Lemma C.3 and Lemma 4.4. $\qquad\square$

Corollary 4.6. *Let $e, f \in \mathcal{T}_{\mathsf{KAT}}$. We have $\mathcal{G}(e) = \mathcal{G}(f) \Leftrightarrow \mathsf{kat}^\star[\![e]\!] = \mathsf{kat}^\star[\![f]\!]$.*