A cut-free cyclic proof system for kleene algebra

Das, Anupam; Pous, Damien

DOI:
10.1007/978-3-319-66902-1_16

A cut-free cyclic proof system for Kleene algebra*

Anupam Das and Damien Pous

Univ. Lyon, CNRS, ENS de Lyon, UCB Lyon 1, LIP, France

Abstract. We introduce a sound non-wellfounded proof system whose regular (or 'cyclic') proofs are complete for (in)equations between regular expressions. We achieve regularity by using *hypersequents* rather than usual sequents, with more structure in the succedent, and relying on the discreteness of *rational languages* to drive proof search. By inspection of the proof search space we extract a PSPACE bound for the system, which is optimal for deciding such (in)equations.

1 Introduction

Kleene algebra is a finite quasi-equational theory over regular expressions [11], which admits *formal languages* and *binary relations* as free models. Indeed, Krob and Kozen independently proved its completeness: every equation which is universally valid in one of those models, or equivalently, whose members denote the same rational language, is provable from the axioms of Kleene algebra [21] [28]. This theorem is important in practice since it shows that the equational theory of Kleene algebra is decidable, and actually PSPACE-complete: it reduces to the problem of comparing rational languages. Thanks to the model of binary relations, Kleene algebra and its extensions have been used to reason abstractly about program correctness [24,25,2,17,1]. The aforementioned decidability result actually made it possible to automate reasoning steps in proof assistants [5,26,31].

Following work in substructural logics about residuated lattices [29], Jipsen proposed a sequent system for Kleene algebra and asked whether the cut-rule is admissible in this system [19]—Buszkowski proved it is not [10]. Wurm recently proposed a different sequent system [34], but his cut-admissibility theorem does not hold [12, App. A]. Proofs in these two systems are finite and well-founded.

Palka proposed a sequent system for *star-continuous action lattices* [30], and thus in particular for Kleene algebra. She proved completeness and cut-elimination. Her system is wellfounded but relies on an 'ω-rule' for Kleene star with infinitely many premisses, in the traditional school of infinitary proof theory [33]. Doing so has the advantage of being simple, but it does not admit any reasonable notion

* An extended version of this abstract is available on HAL [12]. This work was supported by the European Research Council (ERC) under the Horizon 2020 programme (CoVeCe, grant agreement No 678157) and the LABEX MILYON (ANR-10-LABX-0070) of Université de Lyon, within the program "Investissements d'Avenir" (ANR-11-IDEX-0007)

$$\begin{tikzcd}
 & & \mathsf{HKA} \vdash^{\infty} e \to f \arrow[dl, Rightarrow, "\text{Thm. 9}"'] \\
\mathrm{KA} \vdash e \leq f \arrow[r, Leftrightarrow, "{[27],[21]}"] & \mathcal{L}(e) \subseteq \mathcal{L}(f) \arrow[dr, Rightarrow, "\text{Cor. 22}"] & \\
 & & \mathsf{HKA} \vdash^{\omega} e \to f \arrow[uu, Rightarrow]
\end{tikzcd}$$

Fig. 1. Algebraic and proof-theoretic views on rational language equivalence.

of 'finiteness': every proof is necessarily infinite. As a consequence, such a system cannot be used for proof search.

In similar lines of work, the ω-rule can often be restricted to only finitely many cases by some finite model property of the logic [7,8]. This indeed could be done for Palka's system, requiring an exponential number of cases, leading to rather large proofs and inefficient proof search. Such systems do not obey the subformula property and are only *weakly* sound, preserving validity rather than truth, making metalogical properties, such as interpolation, difficult to prove.

We introduce in this paper a calculus HKA for Kleene algebra whose non-wellfounded proofs we prove sound and complete (Sects. 5 and 6). This calculus is cut-free and admits the subformula property. We actually prove that its regular fragment—those proofs with potentially cyclic but finite dependency graphs—is complete. Our approach is related to other works on cyclic systems for logics, e.g. [13,15], but is more fine-grained proof theoretically. We give a diagrammatic summary of our contributions in Fig. 1, where we use the symbols $\vdash^{\omega}$ and $\vdash^{\infty}$ to distinguish between regular proofs and arbitrary, potentially infinite proofs resp.

Starting from Palka's system, a natural idea when looking for a regular system consists in replacing her infinitary rules for Kleene star by finitary ones, and allowing non-wellfounded proofs. Doing so, we obtain the calculus LKA described in Sect. 3: proofs that are well-founded but of infinite width in Palka's system become finitely branching but infinitely deep in LKA. These non-wellfounded proofs of LKA admit an elegant proof theory, but we show that its regular fragment is not complete: there are valid inequalities which require arbitrarily large sequents to appear in their proofs. We solve this problem by allowing slightly more structure in the succedents of sequents, moving to *hypersequents* to design the calculus HKA (Sect. 4). After showing completeness, inspection of the regular proofs of HKA yields an alternative proof that the equational theory of rational languages is in PSpace, without relying on automata-theoretic arguments (Sect. 7). We conclude this paper with some further comments and directions for future work (Sect. 8).

2 Kleene algebra

We consider *regular expressions* over a finite *alphabet* A:

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

$$
id\,\frac{}{e \to e} \qquad 0\text{-}l\,\frac{}{\Gamma, 0, \Delta \to e} \qquad 1\text{-}l\,\frac{\Gamma, \Delta \to e}{\Gamma, 1, \Delta \to e} \qquad 1\text{-}r\,\frac{}{\to 1}
$$

$$
\cdot\text{-}l\,\frac{\Gamma, e, f, \Delta \to g}{\Gamma, e \cdot f, \Delta \to g} \qquad +\text{-}l\,\frac{\Gamma, e, \Delta \to g \quad \Gamma, f, \Delta \to g}{\Gamma, e + f, \Delta \to g} \qquad *\text{-}l\,\frac{\Gamma, \Delta \to f \quad \Gamma, e, e^*, \Delta \to f}{\Gamma, e^*, \Delta \to f}
$$

$$
\cdot\text{-}r\,\frac{\Gamma \to e \quad \Delta \to f}{\Gamma, \Delta \to e \cdot f} \qquad +\text{-}r_i\,\frac{\Gamma \to e_i}{\Gamma \to e_1 + e_2}\, i \in \{1, 2\} \qquad *\text{-}r_1\,\frac{}{\to e^*} \qquad *\text{-}r_2\,\frac{\Gamma \to e \quad \Delta \to e^*}{\Gamma, \Delta \to e^*}
$$

Fig. 2. The rules of LKA.

Sometimes we simply write ef for $e \cdot f$. Each expression e denotes a rational language $\mathcal{L}(e) \subseteq A^*$, defined in the usual way [20]. A *Kleene algebra* is a tuple $(K, 0, 1, +, \cdot, *)$ where $(K, 0, 1, +, \cdot)$ is an idempotent semiring and:[1]

(a) $1 + xx^* \leq x^*$;
(b) if $xy \leq y$ then $x^*y \leq y$;
(c) if $yx \leq y$ then $yx^* \leq y$.

There are several equivalent variants of this definition [11]. Intuitively we have that x^*y (resp. yx^*) is the least fixpoint of $z \mapsto y + xz$ (resp. $z \mapsto y + zx$).

We write $\mathrm{KA} \vdash e \leq f$ if $e \leq f$ is provable from the axioms of Kleene Algebra, i.e. is true in all Kleene algebras (by completeness of first-order logic). We have the following completeness result, independently due to Kozen and Krob:

Theorem 1 ([21,28]). $\mathrm{KA} \vdash e \leq f$ *if and only if* $\mathcal{L}(e) \subseteq \mathcal{L}(f)$.

Formal languages, i.e. subsets of A^*, form a Kleene algebra, so the left-right implication is straightforward. The converse, completeness, is much harder.

3 An intrinsically non-regular system: LKA

A sequent is an expression $\Gamma \to e$, where Γ is a (possibly empty) list of regular expressions and e is a regular expression. For such a sequent we refer to Γ as the *antecedent* and e as the *succedent*. We say a sequent $e_1, \dots, e_n \to e$ is *valid* if $\mathcal{L}(e_1 \cdot \dots \cdot e_n) \subseteq \mathcal{L}(e)$, i.e. the comma is interpreted as sequential composition, and the sequent arrow as inclusion. We refer to expressions as 'formulae' when it is more natural proof theoretically, e.g. 'subformula' or 'principal formula'.

The rules of LKA are given in Fig. 2. Aside from the $*$-rules, these form a fragment of non-commutative intuitionistic linear logic [16],[2] or alternatively the Lambek calculus [29], restricted to the following connectives: multiplicative conjunction $(\cdot)$, multiplicative truth (1), additive disjunction $(+)$ and additive

[1] Here we write $x \leq y$ as a shorthand for $x + y = y$.
[2] This logic is non-commutative because there is no exchange rule, and intuitionistic since there is exactly one formula on the right-hand side.

falsity (0) (for which there is no right rule). The rules for Kleene star arise from the characterisation of e^* as a fixed point: $e^* = \mu x.(1+ex)$. In contrast, Palka [30] interprets Kleene star as an infinite sum $e^* = \Sigma_i e^i$, corresponding to $*$-*continuity* in a Kleene algebra, whence her left rule for Kleene star with infinitely many premisses and the infinitely many corresponding right rules.

The rules of LKA are *sound*: if each premiss of a rule is true in a Kleene algebra then so is its conclusion. LKA also has the *subformula property*: any expression in the premiss of a rule instance is a subformula of an expression in its conclusion. On the other hand, the usual finite well-founded proof system arising from these rules is not complete: there are valid sequents which conclude no finite proof tree of LKA rules, cf. Ex. 4 below. To obtain completeness, we consider *non-wellfounded* proofs. Intuitively, these are infinite trees built from the rules of LKA. More formally:

Definition 2. *A (binary, possibly infinite)* tree *is a prefix-closed subset of* $\{0,1\}^*$. *An* LKA-preproof *is a labelling* π *of a tree by sequents such that, for every node* v *with children* $v_1, \dots v_n$ $(n = 0, 1, 2)$*, the expression* $\dfrac{\pi(v_1) \ \cdots \ \pi(v_n)}{\pi(v)}$ *is an instance of an* LKA *rule. A preproof is* regular *if it has only finitely many distinct subtrees, i.e. it can be expressed as the infinite unfolding of a finite graph.*

Preproofs are not always sound (hence the terminology). Consider, for instance, the preproof on the right deriving a non-valid sequent, where we use the symbol $\bullet$ to indicate a circularity (i.e. to identify steps whose conclusions root the same subtree). Fortunately, we may rule out such behaviours by a simple fairness criterion:

$$\dfrac{\dfrac{}{\to 1}\,{\scriptstyle 1\text{-}r} \qquad \dfrac{\vdots}{a \to 1^*}\,{\scriptstyle *\text{-}r_2}\ \bullet}{a \to 1^*}\,{\scriptstyle *\text{-}r_2}\ \bullet$$

Definition 3. *A* proof *is a preproof that is* fair *for* $*$-l*, i.e. where every infinite branch contains infinitely many occurrences of* $*$-l*. We write* LKA $\vdash^\infty \Gamma \to e$ *if there is an* LKA *proof of* $\Gamma \to e$.

This criterion is somewhat simpler than the ones from other works, e.g. [15] and [13], which require a finer analysis of formula occurrences in infinite branches. However, for our purposes, the condition above suffices and, indeed, leads to a simpler correctness checking procedure for a preproof, cf. Sect. 7.

Example 4. Here is an (infinite but regular) proof of $a^*(b+c)^* \leq a^*(c+b)^*$ in LKA. The fairness criterion is satisfied since the only circularity goes through a $*$-l rule.

$$\dfrac{\dfrac{\dfrac{}{a^* \to a^*}\,{\scriptstyle id} \qquad \dfrac{\dfrac{}{\to (c+b)^*}\,{\scriptstyle *\text{-}r_1} \qquad \dfrac{\dfrac{\dfrac{\dfrac{}{b \to b}\,{\scriptstyle id}}{b \to c+b}\,{\scriptstyle +\text{-}r_2} \qquad \dfrac{\dfrac{}{c \to c}\,{\scriptstyle id}}{c \to c+b}\,{\scriptstyle +\text{-}r_1}}{b+c \to c+b}\,{\scriptstyle +\text{-}l} \qquad \dfrac{\vdots}{(b+c)^* \to (c+b)^*}\,{\scriptstyle *\text{-}l}\ \bullet}{b+c, (b+c)^* \to (c+b)^*}\,{\scriptstyle *\text{-}r_2}}{(b+c)^* \to (c+b)^*}\,{\scriptstyle *\text{-}l}\ \bullet}{a^*, (b+c)^* \to a^*(c+b)^*}\,{\scriptstyle \cdot\text{-}r}}{a^*(b+c)^* \to a^*(c+b)^*}\,{\scriptstyle \cdot\text{-}l}$$

Note that this sequent has no finite wellfounded proof in LKA.

Theorem 5 (Soundness). *If* $\mathsf{LKA} \vdash^{\infty} e_1, \dots, e_n \to e$ *then* $\mathcal{L}(e_1 \cdot \dots \cdot e_n) \subseteq \mathcal{L}(e)$.

Proof (idea). Similar to the proof we give in Sect. 5 for the system HKA. $\square$

While LKA satisfies the subformula property, the size and number of sequents occurring in a proof are not *a priori* bounded, due to the $*$-l rule. In fact, this system does not admit regular proofs for all valid sequents. An example is the inequality $aa^* \leq a^*a$, whose only proof in LKA is the following:

$$
{\scriptstyle \cdot\text{-}l}\,\dfrac{{\scriptstyle *\text{-}l}\,\dfrac{{\scriptstyle \cdot\text{-}r}\,\dfrac{{\scriptstyle *\text{-}r_1}\,\dfrac{}{\to a^*} \quad {\scriptstyle id}\,\dfrac{}{a \to a}}{a \to a^* a} \qquad {\scriptstyle *\text{-}l}\,\dfrac{{\scriptstyle \cdot\text{-}r}\,\dfrac{{\scriptstyle *\text{-}r_2}\,\dfrac{{\scriptstyle id}\,\dfrac{}{a \to a} \quad {\scriptstyle *\text{-}r_1}\,\dfrac{}{\to a^*}}{a \to a^*} \quad {\scriptstyle id}\,\dfrac{}{a \to a}}{a, a \to a^* a} \qquad {\scriptstyle *\text{-}l}\,\dfrac{\vdots}{a, a, a, a^* \to a^* a}}{a, a, a^* \to a^* a}}{a, a^* \to a^* a}}{aa^* \to a^* a} \tag{1}
$$

This proof necessarily contains all sequents of the form $a, \dots, a, a^* \to a^*a$. Even though it could arguably be 'described' in a finite way, this would require an external specification, contrary to the notion of regularity which simply allows cycles in the dependency graph of a proof. Indeed, only finitely many sequents occur in a regular proof, and so they are somewhat easier to reason about.

Many cases of non-regularity can be avoided by adding *symmetric* versions of the $*$ rules in LKA:

$$
{\scriptstyle *\text{-}l'}\,\dfrac{\Gamma, \Delta \to f \quad \Gamma, e^*, e, \Delta \to f}{\Gamma, e^*, \Delta \to f} \qquad {\scriptstyle *\text{-}r_2'}\,\dfrac{\Gamma \to e^* \quad \Delta \to e}{\Gamma, \Delta \to e^*} \tag{2}
$$

For instance, using these rules, it is not hard to see that the situation (1) above can be handled by a well-founded finite proof (see [12, App. B]).

However, adding the rules from (2) above does not always suffice for regularity. Consider the valid sequent $a^* \to (aa)^* + a(aa)^*$. It is not hard to see that any proof must contain a path of just $*$-l steps, since we are never able to apply a $+$-r step while there remains an a^* on the left. Thus it admits no regular proof in LKA, even with the rules from (2).

Similarly, consider the valid sequent $(a+b)^* \to a^*(ba^*)^*$. Any proof of this sequent, even with symmetric rules, must contain some path of sequents whose antecedents denote languages containing $a^m(a+b)^*a^n$, for sufficiently large m, n. Along such a path a $\cdot$-r step is never valid and so one is forced to apply $*$-l and $+$-l rules forever, again yielding a non-regular proof.

The next section presents a system where we can avoid these issues by reasoning *underneath* $\cdot$ and $+$ in the succedent, and thus arrive at a general completeness theorem for regular proofs. We come back to the problem cases discussed above at the end of the next section, in Ex. 23.

4 A calculus whose regular proofs are complete: HKA

We denote lists of formulae by Γ, Δ etc. as before. We will use X, Y, Z to vary over multisets of lists. A *hypersequent* is an expression $\Gamma \to X$, where Γ is a list and X is a multiset of lists. Henceforth we may simply say 'sequent' instead of 'hypersequent' when it is not ambiguous. We use the comma, ',', for both delimiting lists and denoting union of multisets, using angled brackets $\langle\cdot\rangle$ to distinguish lists in a multiset. Here is the general form of a hypersequent:

$$e_1, \dots, e_l \to \langle f_{11}, \dots, f_{1n_1}\rangle, \cdots, \langle f_{m1}, \dots, f_{mn_m}\rangle$$

We extend the notion of language of a regular expression to lists of expressions by setting $\mathcal{L}(\langle e_1, \dots, e_n\rangle) = \mathcal{L}(e_1 \cdot \dots \cdot e_n)$, and to multisets of such lists by setting $\mathcal{L}(\langle\Gamma_1\rangle, \dots, \langle\Gamma_n\rangle) = \bigcup_i \mathcal{L}(\Gamma_i)$. The hypersequent $\Gamma \to X$ is *valid* if $\mathcal{L}(\Gamma) \subseteq \mathcal{L}(X)$.

If $X = \langle\Delta_1\rangle, \dots, \langle\Delta_n\rangle$, we write $\langle\Sigma\rangle X$ for the set $\langle\Sigma, \Delta_1\rangle, \dots, \langle\Sigma, \Delta_n\rangle$. When Σ is a singleton $\langle e\rangle$ we simply write eX instead of $\langle e\rangle X$, as an abuse of notation.

The rules of HKA are given in Fig. 3. Notice that these rules satisfy the subformula property. The left logical rules are exactly those of LKA, lifted to hypersequents. The right logical rules slightly differ, to take advantage of the richer structure of the sequents. Weakening and contractions are allowed on the right of the sequents; the identity axiom from LKA is decomposed into an axiom for the empty lists, and a 'modal' rule (k).

Definition 6. Preproofs *and* proofs *of* HKA *are defined analogously to* LKA*; in particular proofs require fairness of $*$-l. We write* HKA $\vdash^{\omega}$ $\Gamma \to X$ *if* $\Gamma \to X$ *has a* regular *proof in* HKA*, i.e. one with only finitely many distinct subtrees.*

Remark 7 (Invertibility and cancellation). All rules of HKA except w and k are *strongly* invertible: truth of the conclusion implies truth of all premisses in any Kleene algebra. k is not strongly invertible, even in its atomic form, due to the possible existence of 0-divisors. It is however weakly invertible when e is atomic: the validity of the conclusion implies the validity of the premiss.[3] On the other hand, the non-invertibility of w turns out to be crucial for completeness, from a complexity theoretic point of view, cf. Sect. 7.

Example 8 (Atomic modal steps). We can reduce k steps to atomic form by regular derivations of HKA. This is proved by structural induction on the modal expression; the key case is for a $*$-formula, where non-wellfoundedness appears:

$$
{*\text{-}l}\ \dfrac{
{*\text{-}r,w}\ \dfrac{\Gamma \to X}{\Gamma \to e^* X}
\qquad
{*\text{-}r,w}\ \dfrac{
IH\ \dfrac{
{*\text{-}l}\ \dfrac{\vdots}{e^*, \Gamma \to e^* X}\ \bullet
}{e, e^*, \Gamma \to \langle e, e^*\rangle X}
}{e, e^*, \Gamma \to e^* X}
}{e^*, \Gamma \to e^* X}\ \bullet
$$

The derivation marked *IH* is obtained from the inductive hypothesis on e.

[3] Note that atomicity of e really is required for this, even in the usual rational language model. For instance, we have $\mathcal{L}(a^*ab) \subseteq \mathcal{L}(a^*b)$, but $\mathcal{L}(ab) \nsubseteq \mathcal{L}(b)$.

Left logical rules:

$$\text{0-}l\,\frac{}{\Gamma, 0, \Delta \rightarrow} \qquad \text{1-}l\,\frac{\Gamma, \Delta \rightarrow X}{\Gamma, 1, \Delta \rightarrow X} \qquad \cdot\text{-}l\,\frac{\Gamma, e, f, \Delta \rightarrow X}{\Gamma, e \cdot f, \Delta \rightarrow X}$$

$$+\text{-}l\,\frac{\Gamma, e, \Delta \rightarrow X \quad \Gamma, f, \Delta \rightarrow X}{\Gamma, e + f, \Delta \rightarrow X} \qquad *\text{-}l\,\frac{\Gamma, \Delta \rightarrow X \quad \Gamma, e, e^*, \Delta \rightarrow X}{\Gamma, e^*, \Delta \rightarrow X}$$

Right logical rules:

$$\text{1-}r\,\frac{\Gamma \rightarrow X, \langle \Delta, \Sigma \rangle}{\Gamma \rightarrow X, \langle \Delta, 1, \Sigma \rangle} \qquad \cdot\text{-}r\,\frac{\Gamma \rightarrow X, \langle \Delta, e, f, \Sigma \rangle}{\Gamma \rightarrow X, \langle \Delta, e \cdot f, \Sigma \rangle}$$

$$+\text{-}r\,\frac{\Gamma \rightarrow X, \langle \Delta, e, \Sigma \rangle, \langle \Delta, f, \Sigma \rangle}{\Gamma \rightarrow X, \langle \Delta, e + f, \Sigma \rangle} \qquad *\text{-}r\,\frac{\Gamma \rightarrow X, \langle \Delta, \Sigma \rangle, \langle \Delta, e, e^*, \Sigma \rangle}{\Gamma \rightarrow X, \langle \Delta, e^*, \Sigma \rangle}$$

Identity, modal and structural rules:

$$id\,\frac{}{\rightarrow \langle\ \rangle} \qquad k\,\frac{\Gamma \rightarrow X}{e, \Gamma \rightarrow eX} \qquad w\,\frac{\Gamma \rightarrow X}{\Gamma \rightarrow X, \langle \Delta \rangle} \qquad c\,\frac{\Gamma \rightarrow X, \langle \Delta \rangle, \langle \Delta \rangle}{\Gamma \rightarrow X, \langle \Delta \rangle}$$

Fig. 3. The rules of HKA.

5 Soundness

We now show that HKA proofs derive only valid sequents. Throughout this section and later, we use standard proof theoretic terminology about *ancestry* in proofs, e.g. from [9].

Theorem 9 (Soundness). *If* HKA $\vdash^\infty \Gamma \rightarrow X$*, then* $\mathcal{L}(\Gamma) \subseteq \mathcal{L}(X)$*.*

Before giving the proof, we need the following intermediate result.

Lemma 10. *If* HKA $\vdash^\infty \Gamma, e^*, \Delta \rightarrow X$ *then, for* $n \in \mathbb{N}$*,* HKA $\vdash^\infty \Gamma, e^n, \Delta \rightarrow X$.[4]

Proof. We define appropriate preproofs by induction on n. Replace every direct ancestor of e^* by e^n, adjusting origins as follows,

$$*\text{-}l\,\frac{\Gamma, \Delta \rightarrow X \quad \Gamma, e, e^*, \Delta \rightarrow X}{\Gamma, e^*, \Delta \rightarrow X} \qquad \mapsto \qquad \text{1-}l\,\frac{\Gamma, \Delta \rightarrow X}{\Gamma, 1, \Delta \rightarrow X} \quad \text{or} \quad \cdot\text{-}l\,\frac{\Gamma, e, e^{n-1}, \Delta \rightarrow X}{\Gamma, e^n, \Delta \rightarrow X}$$

when $n = 0$ or $n > 0$, respectively. In the latter case we appeal to the inductive hypothesis. Notice that, on branches where e^* is never principal, this is simply a global substitution of e^n for e^* everywhere along the branch. The preproof resulting from this entire construction is fair since every infinite branch will share a tail with a branch in the proof we began with. □

Now we define a measure with which Thm. 9 will be proved by induction.

[4] Strictly speaking, we should bracket e^n as $e(e(\cdots(ee)))$ and set e^0 to 1.

Definition 11 (Measure of a sequent). *The* $*$-height *of a regular expression* e*, denoted* $h_*(e)$*, is the maximum nesting of* $*$ *in its term tree. Formally:*

- $h_*(0) = h_*(1) = h_*(a) = 0$.
- $h_*(e \cdot f) = h_*(e + f) = \max(h_*(e), h_*(f))$.
- $h_*(e^*) = h_*(e) + 1$.

The weighted $*$-height *of a list* Γ *of expressions, denoted* $wh_*(\Gamma)$ *is the multiset* $\{h(e) : e \in \Gamma\}$*. We totally order such multisets under a well-known ordering [14]: for two multisets*[5] $N, M : \mathbb{N} \to \mathbb{N}$*, we set* $N < M$ *if for any* n *with* $N(n) > M(n)$ *there is a* $n' > n$ *s.t.* $N(n') < M(n')$.

Fact 12 *For every rule of* HKA *except* $*$-l*, the antecedent of each premiss has weighted* $*$*-height bounded by that of the antecedent of the conclusion.*

For the $*$-l rule also notice that, bottom-up, the maximum $*$-height of an expression in the antecedent does not increase. We now prove our soundness result:

Proof (of Thm. 9). Let π be an HKA proof of $\Sigma \to X$ and let us proceed by induction on the weighted $*$-height of the antecedent Σ. For each infinite branch of π take the least $*$-l step that occurs; their conclusions form a bar B through the infinite tree of π. Since π labels a binary tree, the prefix closure of B must be finite by König's Lemma and thus, if each of the sequents of B is valid then so is the conclusion of π by the soundness of well-founded HKA derivations.

Now, consider a subproof π' that derives a sequent in B. This sequent must have the form $\Gamma, f^*, \Delta \to Y$ where f^* is principal for the concluding $*$-l-step of π'. By construction and Fact 12 notice that $wh_*(\Gamma, f^*, \Delta) \leq wh_*(\Sigma)$. Now, by Lemma 10, π' can be transformed into proofs π'_n of $\Gamma, f^n, \Delta \to Y$ for each $n \in \mathbb{N}$. Since $wh_*(\Gamma, f^n, \Delta) < wh_*(\Sigma)$, each π'_n is sound by the inductive hypothesis. Finally, this means that $\Gamma, f^*, \Delta \to Y$ is valid, by definition of Kleene star for languages, and hence $\Sigma \to X$ is valid after all. $\square$

6 Completeness

Infinite non-wellfounded proofs are easily seen to be complete: bottom-up, simply apply left rules forever (they are invertible); the only normal forms of this procedure will have a finite word as the antecedent, whence we may perform the correct finite sequence of right steps to finish proof search.

In this section we give a more sophisticated argument showing that the *regular* fragment of HKA is complete: each valid inclusion has a finite circular proof.

6.1 A regular class of proofs

We first define a class of proofs which can be made regular in a systematic way.

[5] Here we construe multisets as mappings from elements to their multiplicity.

Definition 13. *A preproof is* leftmost *if the principal formula of every logical step is at the beginning of a list, either in the antecedent or the succedent.*

For regularity, the most useful property of a leftmost proof is the following:

Theorem 14. *A leftmost preproof contains only lists of length linear in the size of the end-sequent. Hence only finitely many lists occur in a leftmost preproof.*[6]

Before we can prove this, let us recall some basic notions regarding terms. An *occurrence* in e is a subformula of e together with its position in e. We often omit this positional information when it is unambiguous.

Definition 15 (Total order on occurrences). *Given a fixed term, we define a relation $\preccurlyeq$ on the occurrences in it as follows: $e \preccurlyeq f$ if f contains e, or if e and f are disjoint and e occurs to the left of f.*

Due to the tree structure of a term, any two occurrences in a term are either disjoint or one is contained in the other, so we have the following:

Proposition 16. *$\preccurlyeq$ is a total order on the occurrences in a term.*

In a preproof, let us identify every expression occurring as an occurrence of a term in the end-sequent in the natural way, due to the subformula property and via the usual notions of proof ancestry. In this way, we can meaningfully compare any two expressions in a preproof under $\preccurlyeq$. We have the following:

Lemma 17. *In any leftmost preproof every list is strictly increasing under $\preccurlyeq$.*

Now we can prove the bound on the size of lists in leftmost preproofs:

Proof (of Thm. 14). Every term in a preproof is an ancestor of an occurrence in a term of the end sequent, by the subformula property and usual notions of proof ancestry. Moreover, no occurrence can appear twice in the same list, otherwise we would contradict Lemma 17. □

We still do not quite have regularity, since in the succedent we may have multisets with arbitrarily many occurrences of the same list. Naturally, we appeal to the right structural rules to 'merge' occurrences in such a situation:

Corollary 18. *A leftmost preproof in* HKA *can be transformed into one of the same end-sequent that contains only finitely many distinct sequents.*

Proof. By Thm. 14 only finitely many distinct lists occur in a leftmost preproof. Thanks to contraction and weakening, we can always write succedents with at most two copies of each distinct list, of which there are only finitely many. □

It remains to show that we may place backpointers while preserving correctness:

Corollary 19. *A leftmost proof in* HKA *can be transformed into a regular proof with the same end-sequent.*

[6] *A priori*, this could still be exponentially many in the size of the end-sequent.

Proof. Assuming only finitely many distinct sequents occur, by Cor. 18 above, in each infinite branch some sequent occurs infinitely often, by the pigeonhole principle. This means that, due to fairness, for each infinite branch we may identify two instances of the same sequent with a $*$-l-step in between, whence we may correctly place a backpointer and preserve fairness. $\square$

6.2 Completeness of leftmost proofs

Thanks to Cor. 19, for completeness of the regular fragment of HKA it now suffices to show that any valid hypersequent admits a leftmost (possibly infinite) proof. We do so by providing a leftmost proof search strategy for which we need the following important result:

Lemma 20 (Productivity on the right). *Suppose there is a finite* HKA *derivation of right logical rules of the following format,*[7]

$$\dfrac{\begin{array}{c}\Gamma \to X, \langle e^*, \Delta\rangle \\ \Big| \, \pi \\ \Gamma \to Y, \langle \Delta \rangle, \langle e, e^*, \Delta\rangle\end{array}}{\Gamma \to Y, \langle e^*, \Delta\rangle}\ {*\text{-}r}$$

such that the list $\langle e^*, \Delta\rangle$ in the initial sequent is an ancestor of that from the end sequent. If the end sequent is valid, then so is $\Gamma \to X$.

Proof. Since all right logical rules of HKA are invertible, it suffices to show that $\langle e^*, \Delta\rangle$ in the initial sequent is redundant, i.e. that already $\mathcal{L}(X) \supseteq \mathcal{L}(\langle e^*, \Delta\rangle)$. For this, we appeal to soundness of fair preproofs, Thm. 9, and show that HKA proves the corresponding sequent: $e^*, \Delta \to X$.[8] We construct an appropriate proof π' bottom-up by induction on the length of π where, for each right logical rule in π, we apply the analogous left logical rule in π' along the appropriate branch. Each leaf of π' will be of the form $\Sigma \to X$, where Σ is a list occurring in the succedent of the premiss of π, by construction. Now, if $\Sigma \in X$ then we can conclude by weakening, k and identity; otherwise Σ is $\langle e^*, \Delta\rangle$, whence we can conclude by circularity. Notice that π' is fair due to the fact that the bottommost step is a $*$-l due to the analogous $*$-r beneath π. $\square$

We can now prove our main completeness result:

Theorem 21. *Every valid hypersequent has a leftmost proof in* HKA.

Proof. Construct a leftmost HKA preproof bottom-up as follows:

(i) Apply leftmost left logical rules as long as possible. After this any leaves will be valid, by invertibility of logical rules, and of the form:

$$\to X \quad \text{or} \quad a, \Gamma \to X$$

[7] Notice that right logical rules do not branch.

[8] This argument is akin to applying a cut, which is sound since we are only applying it once, and at the meta-level.

(ii) Apply leftmost right logical rules until the succedent contains only lists beginning with a $*$-term that have already been decomposed[9] or lists for which no leftmost right logical rule applies. This terminates after finitely many steps due to Thm. 14 and since only $*$-r can increase the length of a list in the succedent. All resulting leaves must be valid, again by invertibility.

(iii) Now we apply w to weaken any appropriate lists in the succedent that have already been decomposed. Leaves remain valid due to Lemma 20 and must be of the form:

$$\to (\langle\ \rangle,)\, \langle a_1, X_1\rangle, \ldots, \langle a_n, X_n\rangle \quad \text{or} \quad a, \Gamma \to (\langle\ \rangle,)\, \langle a_1, X_1\rangle, \ldots, \langle a_n, X_n\rangle$$

In the former case, since we have preserved validity going upwards, we must have that the empty list occurs in the succedent, whence we can close the branch by several w steps and id.
In the latter case, again since we have preserved validity going upwards, we must be able to weaken any list that begins with an a_i that is not a and preserve validity. Now any remaining leaves are of the form,

$$a, \Gamma \to aX$$

whence we can apply k and preserve validity by Rmk. 7. Now go back to (i) and repeat the entire procedure.

This procedure will produce a leftmost preproof that is fair since (ii) produces only finite well-founded derivations, and so any infinite branch must either eventually remain in the (i) or (iii) case. For the former, a $*$-l must occur infinitely often since the other left rules shorten the antecedent, and for the latter a k step occurs infinitely often, again meaning that a $*$-l step must occur infinitely often since k also shortens the antecedent. $\square$

Corollary 22. *If $\mathcal{L}(e) \subseteq \mathcal{L}(f)$ then* $\mathsf{HKA} \vdash^{\omega} e \to f$.

Proof. By Cor. 19 and Thm. 21. $\square$

Example 23. Let us see how the example issues for regularity for LKA we alluded to in Sect. 3 are resolved in HKA. In both cases we use variations of the strategy given in the proof above of Thm. 21.

$$
\dfrac{\dfrac{\dfrac{
 \dfrac{\dfrac{}{\to \langle\ \rangle}\,{\scriptstyle id}}{\to \langle (aa)^*\rangle, \langle a, (aa)^*\rangle}\,{\scriptstyle *\text{-}r,w}
 \qquad
 \dfrac{\dfrac{\dfrac{\dfrac{\vdots}{a^* \to \langle a, (aa)^*\rangle, \langle (aa)^*\rangle}\,{\scriptstyle *\text{-}l}\ \bullet}{a, a^* \to \langle a, a, (aa)^*\rangle, \langle a, (aa)^*\rangle}\,{\scriptstyle k}}{a, a^* \to \langle (aa)(aa)^*\rangle, \langle a, (aa)^*\rangle}\,{\scriptstyle 2\cdot\text{-}r}}{a, a^* \to \langle (aa)^*\rangle, \langle a, (aa)^*\rangle}\,{\scriptstyle *\text{-}r,w}
}{a^* \to \langle (aa)^*\rangle, \langle a, (aa)^*\rangle}\,{\scriptstyle *\text{-}l}\ \bullet}{a^* \to \langle (aa)(aa)^*\rangle, \langle a(aa)^*\rangle}\,{\scriptstyle \cdot\text{-}r}}{a^* \to \langle (aa)^* + a(aa)^*\rangle}\,{\scriptstyle +\text{-}r}
$$

[9] Here we mean in the sense that it is identical to a descendant, as in Lemma 20.

$$
\begin{prooftree}
\AxiomC{}
\RightLabel{}
\LeftLabel{id}
\UnaryInfC{$\to \langle\ \rangle$}
\LeftLabel{$*\text{-}r,w$}
\doubleLine
\UnaryInfC{$\to \langle (ba^*)^* \rangle$}
\LeftLabel{$*\text{-}r,w$}
\doubleLine
\UnaryInfC{$\to \langle a^*, (ba^*)^* \rangle$}
\AxiomC{$\vdots$}
\LeftLabel{$*\text{-}l$}
\RightLabel{$\bullet$}
\UnaryInfC{$(a+b)^* \to \langle a^*, (ba^*)^* \rangle$}
\LeftLabel{k}
\UnaryInfC{$a, (a+b)^* \to \langle a, a^*, (ba^*)^* \rangle$}
\LeftLabel{$*\text{-}r,w$}
\doubleLine
\UnaryInfC{$a, (a+b)^* \to \langle a^*, (ba^*)^* \rangle$}
\AxiomC{$\vdots$}
\LeftLabel{$*\text{-}l$}
\RightLabel{$\bullet$}
\UnaryInfC{$(a+b)^* \to \langle a^*, (ba^*)^* \rangle$}
\LeftLabel{k}
\UnaryInfC{$b, (a+b)^* \to \langle b, a^*, (ba^*)^* \rangle$}
\LeftLabel{$\cdot\text{-}r$}
\UnaryInfC{$b, (a+b)^* \to \langle ba^*, (ba^*)^* \rangle$}
\LeftLabel{$*\text{-}r,w$}
\doubleLine
\UnaryInfC{$b, (a+b)^* \to \langle (ba^*)^* \rangle$}
\LeftLabel{$*\text{-}r,wk$}
\doubleLine
\UnaryInfC{$b, (a+b)^* \to \langle a^*, (ba^*)^* \rangle$}
\LeftLabel{$+\text{-}l$}
\BinaryInfC{$a+b, (a+b)^* \to \langle a^*, (ba^*)^* \rangle$}
\LeftLabel{$*\text{-}l$}
\RightLabel{$\bullet$}
\BinaryInfC{$(a+b)^* \to \langle a^*, (ba^*)^* \rangle$}
\LeftLabel{$\cdot\text{-}r$}
\UnaryInfC{$(a+b)^* \to \langle a^*(ba^*)^* \rangle$}
\end{prooftree}
$$

Remark 24. Antimirov' *partial derivatives* [3] make it possible to build a non-deterministic automaton whose states are the regular expressions, and such that only finitely many states are reachable from a regular expression. The (finitely many) lists appearing in a leftmost proof, seen as regular expressions, are in sharp correspondence with the partial derivatives of the lists in its conclusion. As a consequence, the proof search procedure of Thm. 21 expresses at a very fine grained level the behaviour of certain coinductive algorithms for language inclusion (equivalence), that explore the reachable states of an Antimirov' automaton and try to build a (bi)simulation [18,4].

7 Complexity matters and algorithms for proof search

We present in this section a brief overview of the complexity theoretic aspects of proofs in our calculus HKA.

7.1 Checking validity of a regular preproof

When a preproof is given as a tree with backpointers, it is not difficult to see that checking validity is feasible (i.e. in polynomial time), since we may simply exhaust the paths of the tree, of which there are linearly many, to exclude the existence of a $*$-l-free loop. When the preproof is given as an arbitrary graph the problem is a little more subtle, but remains feasible. Construing sequents as nodes and inference steps as edges, let us delete any edge that corresponds to a $*$-l step. Notice that the original preproof was valid just if there are no infinite paths in the resulting graph, i.e. it is *acyclic*. This can be decided by computing its transitive closure, hence:

Proposition 25. *Validity of a regular* HKA*-preproof, given as an arbitrary directed graph, is polynomial-time decidable.*

Notice that this bound is lower than those for circular proofs in other systems, e.g. [6,15], since logics with more sophisticated fixed points and logical behaviour require a more general correctness criterion reducing to the inclusion of *Büchi automata*, a problem that is PSpace-complete.

7.2 Complexity of proof search

Proof search using HKA yields an optimal bound for deciding equations of Kleene algebra via the induced loop-checking procedure:

Proposition 26. *Proof search in* HKA *induces a* PSPACE *decision procedure for inequalities between regular expressions.*

Proof (sketch). For a leftmost proof we give a polynomial bound on the depth until a loop occurs. Notice that succedents only grow polynomially in depth and $*$-height, by inspection of HKA, and so this indeed yields a PSPACE bound.

Each time a k step is applied, bottom-up, it is on an atom occurrence that may not reoccur, unless we have already formed a loop, namely by unfolding the same $*$-expression, which by construction contains a $*$-l. Every other leftmost step decreases the size of the leftmost term in a list. Thus, any path in a leftmost proof will hit a loop within polynomially many steps. □

Notice that, while almost every step in HKA is invertible, it is the crucial applications of weakening in the procedure of Thm. 21, justified by Lemma 20, which requires proof search to operate in PSPACE rather than coNP. Indeed, it is the number of w steps along any proof path that allows search complexity to climb up the polynomial hierarchy. This cannot be uniformly bounded since deciding inequalities of regular expressions is known to be PSPACE-complete.

8 Conclusions and further work

We proposed a regular and cut-free hypersequent system HKA, which we proved sound and complete for rational language inclusion, and thus for Kleene algebra. We conclude with further comments and directions for future work.

8.1 Richer systems for theorem proving

Now that we have a completeness theorem for HKA, we could envisage enriching the system with more (sound) rules that might be more natural from the point of view of theorem proving. For instance, we might imagine alternative right logical rules for $+$ and $*$ as follows,

$$\frac{\Gamma \to X, \langle \Delta, e_i, \Sigma\rangle}{\Gamma \to X, \langle \Delta, e_1 + e_2, \Sigma\rangle} \quad \frac{\Gamma \to X, \langle \Delta, \Sigma\rangle}{\Gamma \to X, \langle \Delta, e^*, \Sigma\rangle} \quad \frac{\Gamma \to X, \langle \Delta, e, \Sigma\rangle}{\Gamma \to X, \langle \Delta, e^*, \Sigma\rangle} \quad \frac{\Gamma \to X, \langle \Delta, e^*, e^*, \Sigma\rangle}{\Gamma \to X, \langle \Delta, e^*, \Sigma\rangle}$$

Such systems are more expressive since they can encode not only the rules of HKA but also symmetric variants, e.g. unfolding $*$ to the right rather than the left.[10] An illustrative example is the inequality $a^*a \leq a^*$, which was one source of

[10] Notice that the $*$ rules here correspond in fact to an alternative fixed point definition of e^*: $\mu x.(1 + e + xx)$.

irregularity for LKA. Contrast the following two proofs, the left of which follows a leftmost strategy in HKA, the right of which uses the rules above and is acyclic:

$$
\begin{prooftree}
\AxiomC{}
\LeftLabel{id}
\UnaryInfC{$\to \langle\ \rangle$}
\LeftLabel{w}
\UnaryInfC{$\to \langle\ \rangle, \langle a, a^* \rangle$}
\LeftLabel{$*\text{-}r$}
\UnaryInfC{$\to a^*$}
\LeftLabel{k}
\UnaryInfC{$a \to \langle a, a^* \rangle$}
\LeftLabel{w}
\UnaryInfC{$a \to \langle\ \rangle, \langle a, a^* \rangle$}
\LeftLabel{$*\text{-}r$}
\UnaryInfC{$a \to \langle a^* \rangle$}
\AxiomC{$\vdots$}
\LeftLabel{$*\text{-}l$}
\RightLabel{$\bullet$}
\UnaryInfC{$a^*, a \to \langle a^* \rangle$}
\LeftLabel{k}
\UnaryInfC{$a, a^*, a \to \langle a, a^* \rangle$}
\LeftLabel{w}
\UnaryInfC{$a, a^*, a \to \langle\ \rangle, \langle a, a^* \rangle$}
\LeftLabel{$*\text{-}r$}
\UnaryInfC{$a, a^*, a \to \langle a^* \rangle$}
\LeftLabel{$*\text{-}l$}
\RightLabel{$\bullet$}
\BinaryInfC{$a^*, a \to \langle a^* \rangle$}
\LeftLabel{$\cdot\text{-}l$}
\UnaryInfC{$a^* a \to \langle a^* \rangle$}
\end{prooftree}
\qquad
\begin{prooftree}
\AxiomC{}
\LeftLabel{id}
\UnaryInfC{$\to \langle\ \rangle$}
\LeftLabel{k}
\UnaryInfC{$a \to \langle a \rangle$}
\LeftLabel{k}
\UnaryInfC{$a^*, a \to \langle a^*, a \rangle$}
\UnaryInfC{$a^*, a \to \langle a^*, a^* \rangle$}
\UnaryInfC{$a^*, a \to \langle a^* \rangle$}
\LeftLabel{$\cdot\text{-}l$}
\UnaryInfC{$a^* a \to \langle a^* \rangle$}
\end{prooftree}
$$

8.2 Extensions of Kleene algebra

Kleene algebra can be extended with operations such as *meet* [22], *residuals* [32], or *tests* [23]. One can thus ask whether we can obtain regular sequent systems for such extensions. Meets ($\cap$) and residuals ($\multimap$) correspond to additive conjunction and linear implications in (non-commutative) linear logic; they could easily be added to LKA (Palka actually includes them in her system [30]), but it is unclear how to add them to our hypersequent system while preserving regular cut-free completeness. An important difficulty here is that the free model for such structures is not the obvious language model.[11] In contrast, Kleene algebra with tests, whose free model is that of *guarded string* languages [23], could be handled using our approach. It would also be interesting to try adapt our systems to *ω-regular* expressions, which denote languages of *infinite words* and for which automaton models and notions of derivative are well-defined.

8.3 Cut-elimination

By completeness, any reasonable 'cut rule' is admissible in the regular fragment of HKA. A natural question is whether one can prove a direct cut-elimination result, using proof theoretic methods. There are several difficulties here: first one has to define a general enough notion of cut for the hypersequent system; second one has to come up with an appropriate correctness criterion for preproofs with cuts (fairness as in Dfn. 6 is not enough to guarantee soundness); finally, the regular system being complete, one would certainly like to prove that cut-elimination preserves regularity. Such a cut-elimination result would make it possible to interpret Kleene algebra proofs directly into HKA, without going through the free model (languages). This could be helpful to handle extensions of Kleene algebras whose free model is unknown, for instance with meet or with residuals.

[11] Notice also that while it would be natural to enrich the antecedent structure for $\cap$ as we did in succedents for $+$, there is a difficult asymmetry in that $x(y+z) = xy + xz$ but $x(y \cap z) \lneq xy \cap xz$.

8.4 Towards an alternative completeness result for KA

Conversely to the previous comments, an interesting question is whether our completeness result for the regular fragment of HKA can be used to obtain an *alternative* proof of the completeness of Kleene algebra, Thm. 1. Namely, can we prove directly that if $\mathsf{HKA} \vdash^{\omega} e \to f$ then $\mathrm{KA} \vdash e \leq f$, in a direct manner? We believe this is possible, by encoding cycles in a leftmost proof as specific instances of the 'induction' axioms (b) and (c) from Sect. 2.[12] For instance a loop in a regular derivation might be transformed as follows:

$$
\dfrac{f \to g \qquad \begin{array}{c} e^*, f \to g \\ \Big| \pi \\ e, e^*, f \to g \end{array}}{e^*, f \to g}\ {*\text{-}l}
\quad \rightsquigarrow \quad
\dfrac{f \to g \qquad \dfrac{\begin{array}{c} \dfrac{}{g \to g}\ id \\ \Big| \pi[g/(e^*,f)] \\ e, g \to g \end{array}}{e^*, g \to g}\ (b)}{e^*, f \to g}\ cut
$$

Generalising this idea into a full alternative proof of Kozen's and Krob's results is the subject of ongoing work.

References

1. C. J. Anderson, N. Foster, A. Guha, J.-B. Jeannin, D. Kozen, C. Schlesinger, and D. Walker. NetKAT: semantic foundations for networks. In *Proc. POPL*, pages 113–126. ACM, 2014.
2. A. Angus and D. Kozen. Kleene algebra with tests and program schematology. Technical Report TR2001-1844, CS Dpt., Cornell University, July 2001.
3. V. M. Antimirov. Partial derivatives of regular expressions and finite automaton constructions. *TCS*, 155(2):291–319, 1996.
4. F. Bonchi and D. Pous. Checking NFA equivalence with bisimulations up to congruence. In *Proc. POPL*, pages 457–468. ACM, 2013.
5. T. Braibant and D. Pous. An efficient Coq tactic for deciding Kleene algebras. In *Proc. 1st ITP*, volume 6172 of *LNCS*, pages 163–178. Springer, 2010.
6. J. Brotherston and A. Simpson. Sequent calculi for induction and infinite descent. *J. Log. and Comp.*, 21(6):1177–1216, 2011.
7. K. Brünnler and T. Studer. Syntactic cut-elimination for common knowledge. *Ann. Pure Appl. Logic*, 160(1):82–95, 2009.
8. K. Brünnler and T. Studer. Syntactic cut-elimination for a fragment of the modal mu-calculus. *Ann. Pure Appl. Logic*, 163(12):1838–1853, 2012.
9. S. R. Buss. An introduction to proof theory. *Handbook of proof theory*, 137:1–78, 1998.
10. W. Buszkowski. On action logic: Equational theories of action algebras. *J. Log. Comput.*, 17(1):199–217, 2007.
11. J. H. Conway. *Regular algebra and finite machines.* Chapman and Hall, 1971.

[12] Note that the broader problem of whether cyclic proofs can be simulated by 'inductive' proofs for a certain framework has no known general solution, cf. [6].

12. A. Das and D. Pous. A cut-free cyclic proof system for kleene algebra, 2017. Full version of this extended abstract, with appendix, available at `https://hal.archives-ouvertes.fr/hal-01558132/`.
13. C. Dax, M. Hofmann, and M. Lange. A proof system for the linear time μ-calculus. In *Proc. FSTTCS*, volume 4337 of *LNCS*, pages 273–284. Springer, 2006.
14. N. Dershowitz and Z. Manna. Proving termination with multiset orderings. *C. ACM*, 22(8):465–476, 1979.
15. A. Doumane, D. Baelde, L. Hirschi, and A. Saurin. Towards completeness via proof search in the linear time μ-calculus: The case of Büchi inclusions. In *Proc. LICS*, pages 377–386. ACM, 2016.
16. J.-Y. Girard. Linear logic. *TCS*, 50:1–102, 1987.
17. C. A. R. Hoare, B. Möller, G. Struth, and I. Wehrman. Concurrent Kleene Algebra. In *Proc. CONCUR*, volume 5710 of *LNCS*, pages 399–414. Springer, 2009.
18. J. E. Hopcroft and R. M. Karp. A linear algorithm for testing equivalence of finite automata. Technical Report 114, Cornell Univ., 1971.
19. P. Jipsen. From semirings to residuated Kleene lattices. *Studia Logica*, 76(2):291–303, 2004.
20. S. C. Kleene. Representation of events in nerve nets and finite automata. In *Automata Studies*, pages 3–41. Princeton University Press, 1956.
21. D. Kozen. A completeness theorem for Kleene algebras and the algebra of regular events. In *Proc. LICS*, pages 214–225. IEEE, 1991.
22. D. Kozen. On action algebras. In J. van Eijck and A. Visser, editors, *Logic and Information Flow*, pages 78–88. MIT Press, 1994.
23. D. Kozen. Kleene algebra with tests. *Transactions on Programming Languages and Systems*, 19(3):427–443, May 1997.
24. D. Kozen. On Hoare logic and Kleene algebra with tests. *ACM Trans. Comput. Log.*, 1(1):60–76, 2000.
25. D. Kozen and M.-C. Patron. Certification of compiler optimizations using Kleene algebra with tests. In *Proc. CL2000*, volume 1861 of *LNAI*, pages 568–582. Springer, 2000.
26. A. Krauss and T. Nipkow. Proof pearl: Regular expression equivalence and relation algebra. *JAR*, 49(1):95–106, 2012.
27. D. Krob. A Complete System of B-Rational Identities. In *Proc. ICALP*, volume 443 of *LNCS*, pages 60–73. Springer, 1990.
28. D. Krob. Complete systems of B-rational identities. *TCS*, 89(2):207–343, 1991.
29. J. Lambek. The mathematics of sentence structure. *The American Mathematical Monthly*, 65:154–170, 1958.
30. E. Palka. An infinitary sequent system for the equational theory of *-continuous action lattices. *Fundam. Inform.*, pages 295–309, 2007.
31. D. Pous. Kleene Algebra with Tests and Coq tools for while programs. In *Proc. ITP*, volume 7998 of *LNCS*, pages 180–196. Springer, 2013.
32. V. Pratt. Action logic and pure induction. In *Proc. JELIA*, volume 478 of *LNCS*, pages 97–120. Springer, 1990.
33. K. Schütte. *Proof Theory*. Grundlehren der mathematischen Wissenschaften 225. Springer Berlin Heidelberg, 1977. Translation of Beweistheorie, 1968.
34. C. Wurm. Kleene algebras, regular languages and substructural logics. In *Proc. GandALF*, EPTCS, pages 46–59, 2014.