Kleene Theorems for Lasso and ω-Languages

Mike Cruchten
The University of Sheffield

Abstract

Automata operating on pairs of words were introduced as an alternative way of approaching acceptance of regular ω-languages. Families of DFAs and lasso automata followed, and have been studied for their nice properties, giving rise to minimisation algorithms, a Myhill-Nerode theorem and language learning algorithms. For such a well-established class of automata, they are still missing a Kleene theorem.

In this paper, we introduce rational lasso languages and expressions, show a Kleene theorem for lasso languages and explore the connection between rational lasso and ω-expressions, which yields a Kleene theorem for ω-languages with respect to saturated lasso automata. For one direction of the Kleene theorems, we also provide a Brzozowski construction for lasso automata from rational lasso expressions.

1 Introduction

Lasso languages are languages of pairs of words, called lassos. Automata operating on lassos were introduced by [4] to devise an alternative way of accepting regular ω-languages. The idea is to use lassos as finite representatives of ultimately periodic words. As regular ω-languages are uniquely characterised by their ultimately periodic fragment, they are also uniquely characterised by the lasso language containing all the lassos representing some ultimately periodic word in the periodic fragment. With this idea at hand, they show that a regular ω-language can be accepted by a DFA with an extended alphabet. Their construction came with the hope of improving existing algorithms for deciding emptiness and language inclusion of regular ω-languages, which are prominently used in verification and model checking.

By combining ideas from [4, 11], [2] introduced families of DFAs (FDFAs) which are also automata operating on lassos. The authors give three kinds of FDFAs, of which the periodic FDFA resembles most closely the automaton by [4]. In their paper, they provide language learning algorithms to learn regular ω-languages in polynomial time in the size of the FDFA. Moreover, in [1], they investigate the complexity of certain operations and decision procedures on FDFAs, including the performance of Boolean operations and deciding emptiness and language inclusion, and show that these can all be performed in non-deterministic logarithmic space, validating the hopes of Calbrix et al.

An equivalent automaton to the FDFA is the lasso automaton defined in [5], where they give a Myhill-Nerode theorem and show that lasso automata can be minimised using partition refinement. It is known that minimisation can also be done using a double reverse powerset construction à la Brzozowski ([6]).

Although automata operating on lassos are well-established in many regards, they *still lack* a Kleene theorem. Our main goal is to establish a Kleene theorem for lasso languages with respect to lasso automata and to show how rational lasso and ω-expressions relate, which paves a way towards a Kleene theorem for ω-languages with respect to saturated lasso automata (Definition 2.7).

Our contributions are drawn as dashed arrows in Figure 1. We define rational lasso languages as those lasso languages which can be obtained from rational languages using rational lasso operations. Our first contribution is a Kleene theorem for lasso languages, i.e. we show that a lasso language is rational if and only if it is accepted by a finite lasso automaton (Theorem 5.7). For one direction, we

$$\begin{tikzcd}[column sep=large, row sep=huge]
\text{rational } \omega\text{-lang.} \arrow[r, dashed, "6.14"] \arrow[d, bend left, dashed, "6.15"] & \text{rational lasso lang.} \arrow[d, bend right, dashed, "4.8"'] \arrow[d, phantom, "5.7"] \\
\text{regular } \omega\text{-lang.} \arrow[u, bend left, "{[4]}"] \arrow[r, "{[4]}"'] & \text{regular lasso lang.} \arrow[u, bend right, dashed, "{5.4,5.6}"']
\end{tikzcd}$$

Figure 1: Diagram showing our main contributions as dashed arrows.

provide a Brzozowski construction which turns a rational lasso expression into a finite lasso automaton accepting the corresponding rational lasso language (Theorem 4.8). For the other direction, we show how to dissect a finite lasso automaton into several DFAs and prove that the lasso language accepted by the lasso automaton can be obtained from the rational languages corresponding to the DFAs by using the rational lasso operations (Proposition 5.4 and Corollary 5.6), following ideas by [4].

Secondly, we study the relationship between rational lasso and ω-expressions. We introduce a novel notion of a rational lasso expression *representing* a rational ω-expression (Definition 6.1). Intuitively, this notion expresses that the language semantics of either expression completely determines that of the other. We show that for any given rational ω-expression, we can construct a representing rational lasso expression in a syntactic manner provided we have access to two additional operations on rational expressions (cf. Proposition 6.14).

Our two contributions together with a result by [4] allow us to re-establish Kleene's theorem for ω-languages with respect to saturated lasso automata. Given a rational ω-language, we can turn it into a rational lasso expression (Proposition 6.14) and apply our Brzozowski construction (Theorem 4.8) to obtain the desired finite saturated lasso automaton, hence every rational ω-language is regular (Theorem 6.15). The other direction is given by [4], showing that every ω-language accepted by a finite saturated lasso automaton is rational.

Organisation of the paper Section 2 consists of the preliminaries. In Section 3 we introduce rational lasso expressions and languages together with an algebra. Section 4 is devoted to a Brzozowski construction for lasso automata. Section 5 shows how to obtain expressions from automata, and Section 6 investigates the connection between rational lasso and ω-expressions.

2 Preliminaries

We use Σ to denote a finite alphabet. The free monoid over Σ is $(\Sigma^*, \cdot, \varepsilon)$ which consists of finite words written u, v, w with concatenation $u \cdot v = uv$ and the empty word ε. We write Σ^ω for the collection of infinite words and Σ^{up} for the collection of ultimately periodic words, i.e. those of the form uv^ω. A *lasso* is a pair $(u, v) \in \Sigma^* \times \Sigma^+$ (which we abbreviate Σ^{*+}), with u the *spoke* and $v \neq \varepsilon$ the *loop*. We think of the lasso (u, v) as a representative for uv^ω.

We write U, V, W for languages of words and L, K for languages of infinite words (ω-languages) or of lassos, depending on context. The rational languages and ω-languages are defined as usual, so are their operations. A language (resp. ω-language) is regular if it is accepted by a DFA (resp. finite nondeterministic Büchi automaton). For an ω-language L, $\mathrm{UP}(L) = L \cap \Sigma^{\mathrm{up}}$ is its *ultimately periodic fragment*. We use $t, r, s \in \mathrm{Exp}$ for rational expressions and $[\![-]\!] : \mathrm{Exp} \to 2^{\Sigma^*}$ maps a rational expression to its corresponding rational language in the usual way. We write $t \in N$ if $\varepsilon \in [\![t]\!]$, i.e. t has the *empty* *word property.* We use $T \in \text{Exp}_\omega$ to denote rational ω-expressions, and $[\![-]\!]_\omega : \text{Exp}_\omega \to 2^{\Sigma^\omega}$ is the usual language map for rational ω-expressions.

The following definition introduces a rewrite system on lassos. The rewrite system alongside some of its properties (which are stated below) can be found in [6].

Definition 2.1. We define the following rewrite rules on Σ^{*+}:

$$\frac{a \in \Sigma \quad (ua, va)}{(u, av)}\,(\gamma_1) \qquad\qquad \frac{(u, v^k) \quad (k > 1)}{(u, v)}\,(\gamma_2)$$

We write $(u, v) \to_{\gamma_i} (u', v')$ if (u, v) rewrites to (u', v') in one step under γ_i and say that (u, v) γ_i-*reduces* to (u', v') (or that (u', v') γ_i-*expands* to (u, v)). We let $\to_\gamma = \to_{\gamma_1} \cup \to_{\gamma_2}$, denote by $\sim_\gamma$ the least equivalence relation including $\to_\gamma$ and say that two lassos are γ-*equivalent* if they are $\sim_\gamma$-related.

Proposition 2.2. *The relation $\to_\gamma$ is confluent and strongly normalising.*

It follows from this proposition that every lasso (u, v) has a unique *normal form.*

Proposition 2.3 (Lasso Representation Lemma). *Let $(u, v), (u', v') \in \Sigma^{*+}$. Then*

$$(u, v) \sim_\gamma (u', v') \iff uv^\omega = u'v'^\omega.$$

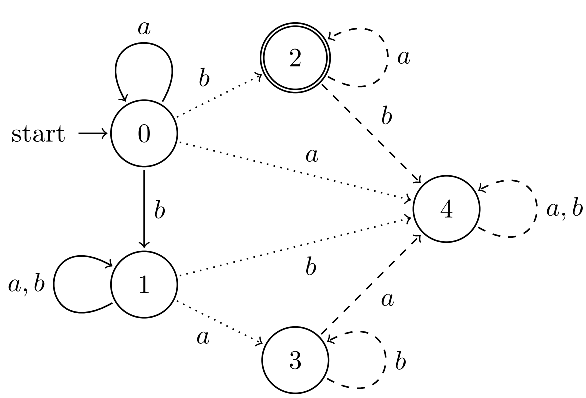
Definition 2.4 ([5]). A *lasso automaton* is a structure $\mathcal{A} = (X, Y, \overline{x}, \delta_1, \delta_2, \delta_3, F)$ where $\delta_1 \colon X \to X^\Sigma$, $\delta_2 \colon X \to Y^\Sigma$, $\delta_3 \colon Y \to Y^\Sigma$ and $F \subseteq Y$. The sets X and Y are called the *spoke* and *loop states.* The maps δ_1, δ_2 and δ_3 are called the *spoke*, *switch* and *loop transition* of $\mathcal{A}$. The set F denotes the *accepting states* and $\overline{x}$ is the *initial state* of $\mathcal{A}$.

For convenience, we also define the map $(\delta_2 : \delta_3) : X \uplus Y \to Y^\Sigma$ which is equal to δ_2 on X and equal to δ_3 on Y. The maps $\delta_1, (\delta_2 : \delta_3)$ and δ_3 can be extended from symbols to finite words in the usual way.

Definition 2.5 ([5]). A lasso $(u, v) \in \Sigma^{*+}$ is *accepted* by $\mathcal{A}$, if $(\delta_2 : \delta_3)(\delta_1(\overline{x}, u), v) \in F$. The *lasso language accepted by* $\mathcal{A}$ consists of all the lassos accepted by $\mathcal{A}$. A lasso language L is *regular* if it is accepted by a finite lasso automaton.

Example 2.6.
The diagram to the right depicts a lasso automaton. The spoke states are labelled 0 and 1. The states $2, 3$ and 4 are loop states. Spoke transitions are drawn as solid arrows, switch transitions as dotted arrows and loop transitions as dashed arrows. The initial state is indicated by the 'start' arrow and accepting states are drawn as double circles.
This lasso automaton accepts the language

$$\{(a^k, ba^j) \mid k, j \in \mathbb{N}\}.$$

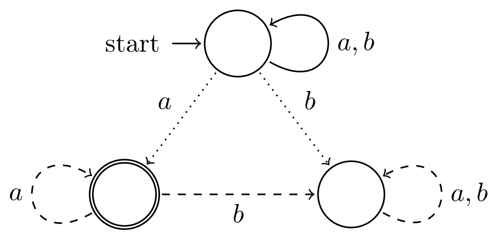


In [5], the authors define an Ω-automaton as a lasso automaton with special structural properties. We give an equivalent definition using saturation.

Definition 2.7. A lasso automaton $\mathcal{A}$ is *saturated* if for any two γ-equivalent lassos $(u_1, v_1), (u_2, v_2) \in \Sigma^{*+} : (u_1, v_1) \in L_\circ(\mathcal{A}) \iff (u_2, v_2) \in L_\circ(\mathcal{A})$. An Ω*-automaton* is a saturated lasso automaton.

Finite Ω-automata act as acceptors of regular ω-languages. A finite Ω-automaton $\mathcal{A}$ accepts the regular ω-language L if $L_\circ(\mathcal{A}) = \{(u, v) \mid uv^\omega \in L\}$ ([5]). Note that $L_\circ(\mathcal{A})$ is always $\sim_\gamma$-saturated (i.e. a union of $\sim_\gamma$ equivalence classes) for an Ω-automaton $\mathcal{A}$, and that $\{(u, v) \mid uv^\omega \in L\} = \{(u, v) \mid uv^\omega \in K\}$ implies $L = K$ for regular ω-languages K, L ([4]). The *regular ω-language accepted by a finite Ω-automaton* $\mathcal{A}$ is denoted $L_\omega(\mathcal{A})$.

Example 2.8.



The lasso automaton shown to the left is saturated. It accepts the regular lasso language $\{(u, a^k) \mid u \in \Sigma^*, k \geq 1\}$ and the regular ω-language $\{ua^\omega \mid u \in \Sigma^*\}$. Note that the automaton in Example 2.6 is not saturated as it accepts the lasso (ε, b) but it does not accept (b, b) although these two lassos are γ-equivalent.

3 Rational Lasso Expressions

This section introduces rational lasso expressions, languages and an algebra thereof which we show to be sound. A lasso language is *rational* if it is obtained from rational languages using the operations

$$U^\circ = \{(\varepsilon, u) \mid u \in U\}, \qquad U \cdot K = \{(uv, w) \mid u \in U, (v, w) \in K\}, \qquad K_1 \cup K_2,$$

where U is a rational language and K, K_1, K_2 are rational lasso languages.

From here on, we assume RA to be an arbitrary but fixed algebra of rational expressions (e.g. KA, the theory of Kleene Algebra [9]). We write $\vdash t = r$ if $t = r$ is deducible in RA and drop the turnstyle when it is clear from context. We write $t \leq r$ for $\vdash t + r = r$, as the $+$-reduct of an RA-algebra is a join-semilattice. Finally, for a formula φ, the *Iverson bracket* of φ is defined as

$$[\varphi] = \begin{cases} 1 & \text{if } \varphi \text{ is true,} \\ 0 & \text{otherwise.} \end{cases}$$

Definition 3.1. Let $t, r \in \text{Exp}$ with $r \notin N$. The set $\text{Exp}_\circ$ of *rational lasso expressions* is defined by the following grammar

$$\rho, \sigma ::= 0 \mid t \cdot \rho \mid \rho + \sigma \mid r^\circ.$$

We associate a rational lasso language to each rational lasso expression using the operations we defined at the start of this section.

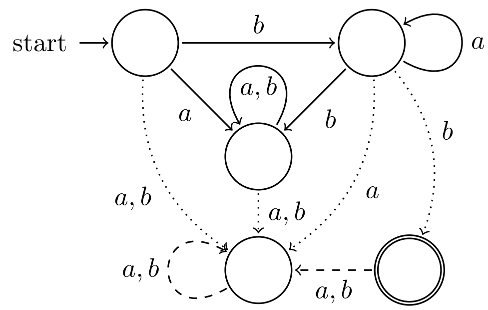
Definition 3.2. The *language semantics* for rational lasso expressions is given by the map $[\![-]\!]_\circ \colon \text{Exp}_\circ \to 2^{\Sigma^{*+}}$ defined recursively on rational lasso expressions:

$$[\![0]\!]_\circ = \emptyset, \qquad [\![t \cdot \rho]\!]_\circ = [\![t]\!] \cdot [\![\rho]\!]_\circ, \qquad [\![\rho + \sigma]\!]_\circ = [\![\rho]\!]_\circ \cup [\![\sigma]\!]_\circ, \qquad [\![r^\circ]\!]_\circ = [\![r]\!]^\circ.$$

Note how the lasso language semantics $[\![-]\!]_\circ$ extends the language semantics of rational expressions $[\![-]\!]$.

Example 3.3. Let $\Sigma = \{a, b\}$ and $\rho = b(a^*b^\circ)$. On the left we compute the associated rational lasso language $[\![\rho]\!]_\circ$. On the right we give a finite lasso automaton for $[\![\rho]\!]_\circ$ in particular showing that this rational lasso language is also regular.

$$\begin{aligned}
[\![\rho]\!]_\circ &= [\![b(a^*b^\circ)]\!]_\circ \\
&= [\![b(a^*b^\circ)]\!]_\circ \\
&= [\![b]\!][\![(a^*b^\circ)]\!]_\circ]\!]_\circ \\
&= \{b\}([\![a^*]\!][\![b^\circ]\!]_\circ) \\
&= \{b\}(\{a^k \mid k \geq 0\}[\![b]\!]^\circ) \\
&= \{b\}(\{a^k \mid k \geq 0\}\{(\varepsilon, b)\}) \\
&= \{b\}\{(a^k, b) \mid k \geq 0\} \\
&= \{(ba^k, b) \mid k \geq 0\}.
\end{aligned}$$



Next, we introduce an equational theory to reason about regular lasso expressions. This theory is sound with respect to the lasso language semantics which we require for the construction of a Brzozowski lasso automaton.

Definition 3.4. The *two-sorted equational theory LA of lasso algebras* extends the equational theory RA by the following axioms:

$$\begin{array}{ccc}
1 \cdot \rho = \rho & 0 \cdot \rho = 0 & \rho + \sigma = \sigma + \rho \\
(t + r) \cdot \rho = t \cdot \rho + r \cdot \rho & 0^\circ = 0 & (\rho + \sigma) + \tau = \rho + (\sigma + \tau) \\
t \cdot (\rho + \sigma) = t \cdot \rho + t \cdot \sigma & 0 + \rho = \rho & \rho + \rho = \rho \\
t \cdot (r \cdot \rho) = (t \cdot r) \cdot \rho & t \cdot 0 = 0 & (t + r)^\circ = t^\circ + r^\circ
\end{array}$$

with $t, r \in \mathrm{Exp}$ and $\rho, \sigma, \tau \in \mathrm{Exp}_\circ$. The axioms together with the laws for equality and the substitution of provably equivalent rational and rational lasso expressions gives us the deductive system **LA**. We write $\vdash_{\mathbf{LA}} \rho = \sigma$ when the equation $\rho = \sigma$ is deducible in LA. Whenever it is clear from context, we drop the turnstile $\vdash_{\mathbf{LA}}$.

Remark 3.5. We briefly highlight the differences between lasso and Wagner algebras ([12]). A Wagner algebra is a two-sorted algebra similar to the lasso algebra but having an operation $(-)^\omega$ instead of $(-)^\circ$. They are used to reason about rational ω-expressions and Wagner showed completeness of his axiomatisation with respect to the language semantics for rational ω-expressions. Wagner's axiomatisation looks very similar to that of a lasso algebra. However, the unary operations $(-)^\circ$ and $(-)^\omega$ satisfy different laws, the ones for $(-)^\omega$ being

$$(t \cdot r)^\omega = t \cdot (r \cdot t)^\omega \qquad \text{and} \qquad t^\omega = (t^+)^\omega.$$

Other than this, there are two more subtle differences that can be pointed out:

1. from the $(-)^\omega$-axioms, one can deduce that $0^\omega = 0$, this is not the case for lasso algebras (i.e. we need the axiom $0^\circ = 0$),

2. Wagner has an additional derivation rule which allows to solve equations of a particular type, such a rule is not given for lasso algebras.

The next proposition shows that the theory of lasso algebras is sound with respect to the language semantics for rational lasso expressions. We make no claim about its completeness.

Proposition 3.6 (Soundness). *Let $\rho, \sigma \in Exp_\circ$. Then $\vdash_{\mathbf{LA}} \rho = \sigma \implies [\![\rho]\!]_\circ = [\![\sigma]\!]_\circ$.*

Proof. We verify each law in Definition 3.4. The checks regarding substitution are easily done and omitted.

1. $1 \cdot \rho = \rho$: $[\![1 \cdot \rho]\!]_\circ = [\![1]\!] \cdot [\![\rho]\!]_\circ = \{\varepsilon\} \cdot [\![\rho]\!]_\circ = [\![\rho]\!]_\circ$.
2. $\rho + \sigma = \sigma + \rho$: $[\![\rho + \sigma]\!]_\circ = [\![\rho]\!]_\circ \cup [\![\sigma]\!]_\circ = [\![\sigma]\!]_\circ \cup [\![\rho]\!]_\circ = [\![\sigma + \rho]\!]_\circ$
3. $0^\circ = 0$: $[\![0^\circ]\!]_\circ = \{(\varepsilon, u) \mid u \in [\![0]\!]\} = \{(\varepsilon, u) \mid u \in \emptyset\} = \emptyset = [\![0]\!]_\circ$
4. $0 + \rho = \rho$: $[\![0 + \rho]\!]_\circ = [\![0]\!]_\circ \cup [\![\rho]\!]_\circ = \emptyset \cup [\![\rho]\!]_\circ = [\![\rho]\!]_\circ$
5. $\rho + \rho = \rho$: $[\![\rho + \rho]\!]_\circ = [\![\rho]\!]_\circ \cup [\![\rho]\!]_\circ = [\![\rho]\!]_\circ$
6. $t \cdot 0 = 0$: $[\![t \cdot 0]\!]_\circ = [\![t]\!] \cdot [\![0]\!]_\circ = [\![t]\!] \cdot \emptyset = \emptyset = [\![0]\!]_\circ$
7. $0 \cdot \rho = \rho$: $[\![0 \cdot \rho]\!]_\circ = [\![0]\!] \cdot [\![\rho]\!]_\circ = \emptyset \cdot [\![\rho]\!]_\circ = \emptyset = [\![0]\!]_\circ$.

For the equation 8. $t \cdot (r \cdot \rho) = (t \cdot r) \cdot \rho$ we get

$$[\![t \cdot (r \cdot \rho)]\!]_\circ = [\![t]\!] \cdot [\![r \cdot \rho]\!]_\circ = [\![t]\!] \cdot ([\![r]\!] \cdot [\![\rho]\!]_\circ) = ([\![t]\!] \cdot [\![r]\!]) \cdot [\![\rho]\!]_\circ = [\![(t \cdot r) \cdot \rho]\!]_\circ.$$

The remaining equations require a little bit more work.

9. $(t + r) \cdot \rho = t \cdot \rho + r \cdot \rho$

$$\begin{aligned} [\![(t + r) \cdot \rho]\!]_\circ &= [\![t + r]\!] \cdot [\![\rho]\!]_\circ = ([\![t]\!] \cup [\![r]\!]) \cdot [\![\rho]\!]_\circ \\ &= [\![t]\!] \cdot [\![\rho]\!]_\circ \cup [\![r]\!] \cdot [\![\rho]\!]_\circ = [\![t \cdot \rho]\!]_\circ \cup [\![r \cdot \rho]\!]_\circ \\ &= [\![t \cdot \rho + r \cdot \rho]\!]_\circ \end{aligned}$$

10. $(\rho + \sigma) + \tau = \rho + (\sigma + \tau)$

$$\begin{aligned} [\![(\rho + \sigma) + \tau]\!]_\circ &= [\![\rho + \sigma]\!]_\circ \cup [\![\tau]\!]_\circ = ([\![\rho]\!]_\circ \cup [\![\sigma]\!]_\circ) \cup [\![\tau]\!]_\circ \\ &= [\![\rho]\!]_\circ \cup ([\![\sigma]\!]_\circ \cup [\![\tau]\!]_\circ) = [\![\rho]\!]_\circ \cup [\![\sigma + \tau]\!]_\circ \\ &= [\![\rho + (\sigma + \tau)]\!]_\circ \end{aligned}$$

11. $t \cdot (\rho + \sigma) = t \cdot \rho + t \cdot \sigma$

$$\begin{aligned} [\![t \cdot (\rho + \sigma)]\!]_\circ &= [\![t]\!] \cdot [\![\rho + \sigma]\!]_\circ = [\![t]\!] \cdot ([\![\rho]\!]_\circ \cup [\![\sigma]\!]_\circ) \\ &= [\![t]\!] \cdot [\![\rho]\!]_\circ \cup [\![t]\!] \cdot [\![\sigma]\!]_\circ = [\![t \cdot \rho]\!]_\circ \cup [\![t \cdot \sigma]\!]_\circ \\ &= [\![t \cdot \rho + t \cdot \sigma]\!]_\circ \end{aligned}$$

12. $(t + r)^\circ = t^\circ + r^\circ$

$$\begin{aligned} [\![(t + r)^\circ]\!]_\circ &= \{(\varepsilon, u) \in \Sigma^{*+} \mid u \in [\![t + r]\!]\} \\ &= \{(\varepsilon, u) \in \Sigma^{*+} \mid u \in [\![t]\!] \cup [\![r]\!]\} \\ &= \{(\varepsilon, u) \in \Sigma^{*+} \mid u \in [\![t]\!]\} \cup \{(\varepsilon, u) \in \Sigma^{*+} \mid u \in [\![r]\!]\} \\ &= [\![t^\circ]\!]_\circ \cup [\![r^\circ]\!]_\circ \\ &= [\![t^\circ + r^\circ]\!]_\circ \end{aligned}$$

□

Analogously to the situation for rational and rational ω-expressions, each rational lasso expression is provably equivalent to a rational expression of the form $\sum_{i=1}^{n} t_i \cdot r_i^\circ$. Such a form is called a *disjunctive form* but note that a disjunctive form is not unique.

Proposition 3.7. *Let $\rho \in Exp_\circ$. Then there exists $n \in \mathbb{N}$ and $t_1, \ldots, t_n, r_1, \ldots, r_n \in Exp$ with $r_i \notin N$ for all $1 \leq i \leq n$ such that*

$$\vdash_{\mathbf{LA}} \rho = \sum_{i=1}^{n} t_i \cdot r_i^\circ.$$

Proof. We show this by structural induction on rational lasso expressions. For 0 we have that $0 = 0{\cdot}0^\circ$. For t° we have $t^\circ = 1 \cdot t^\circ$. For $r \cdot \rho$, we know by the induction hypothesis that we can find a finite number of pairs (t_i, r_i) such that $\rho = \sum_i (t_i, r_i)$. It follows that

$$r \cdot \rho = r \cdot \sum_i t_i \cdot r_i^\circ = \sum_i r \cdot (t_i \cdot r_i^\circ) = \sum_i (r \cdot t_i) \cdot r_i^\circ.$$

Finally, for $\rho + \sigma$ we can find two finite collections $\{(t_i, r_i)\}_{i \in I}, \{(t_j, r_j)\}_{j \in J}$ with $\rho = \sum_i (t_i, r_i)$ and $\sigma = \sum_j (t_j, r_j)$, then clearly $\rho + \sigma = \sum_{k \in I \cup J} (t_k, r_k)$ as required. $\square$

We use (t, r) as a shorthand for $t \cdot r^\circ$, which accentuates the distinction between the 'finite' and 'infinite' part more. It also allows for a more direct correspondence to the language semantics as

$$[\![(t, r)]\!]_\circ = \{(u, v) \in \Sigma^{*+} \mid u \in [\![t]\!], v \in [\![r]\!]\}.$$

4 Brzozowski Construction for Rational Lasso Expressions

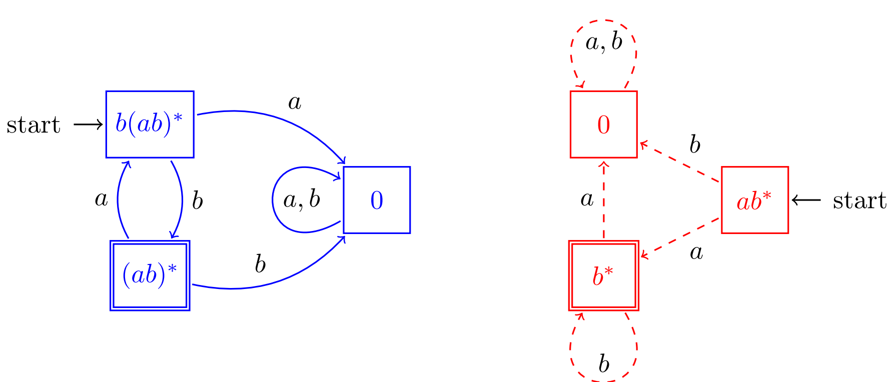
This section explores a construction method for lasso automata from rational lasso expressions. Given a rational lasso expression ρ, our aim is to build a finite lasso automaton $\mathcal{A}$ which accepts $[\![\rho]\!]_\circ$. This shows one direction of Kleene's Theorem, namely that every rational lasso language is regular.

We assume that the reader is familiar with the Brzozowski construction for deterministic finite automata (c.f. Appendix A and [3]).

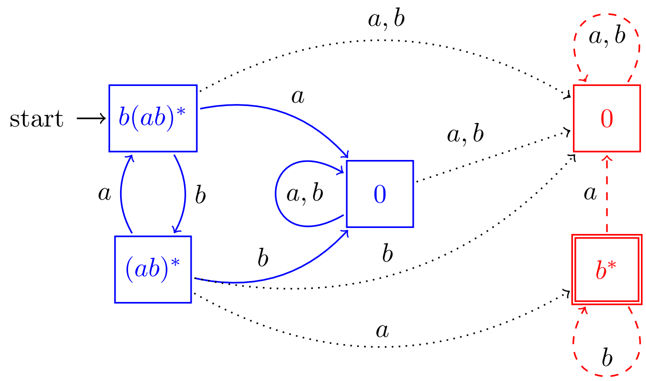
We write d for the *Brzozowski derivative* and N for the collection of expressions that have the *empty word property.* The automaton $\mathcal{B} = (\mathrm{Exp}, d, N)$ is the *Brzozowski automaton*, a deterministic automaton which has the property that for $t \in \mathrm{Exp}$, $L(\mathcal{B}, t) = [\![t]\!]$. This automaton is not necessarily finite but can be turned into a finite one by quotienting its state space by the equivalence relation $\sim_B$ defined in Appendix A. This equivalence relation is compatible with d and N, allowing us to define suitable $\widehat{d}$ and $\widehat{N}$ on equivalence classes. We obtain the automaton $\widehat{\mathcal{B}} = (\mathrm{Exp}/{\sim_B}, \widehat{d}, \widehat{N})$ with the property that for $t \in \mathrm{Exp}$, $L(\widehat{\mathcal{B}}, [t]_{\sim_B}) = [\![t]\!]$ and that there are only finitely many states reachable from $[t]_{\sim_B}$.

We start by looking at an example of the Brzozowski construction for lasso automata with the aim of giving the reader some intuition.

Example 4.1. Let $\rho = (b(ab)^*, ab^*)$. Then $(u, v) \in [\![\rho]\!]_\circ$ if and only if $u \in [\![b(ab)^*]\!]$ and $v \in [\![ab^*]\!]$. So the idea is to build one DFA for $b(ab)^*$ and one DFA for ab^*, which correspond to the spoke and loop part of the lasso automaton, and then link them. The construction of the DFAs is done using Brzozowski derivatives, which yields the following DFAs:



Note that we are allowed to transition from the first to the second DFA only after reading u, that is, only once we have reached an accepting state in the spoke DFA. In order to determine where to switch to, we think of our spoke state as the initial state of the loop DFA. From any other state in the spoke DFA (i.e. the non-accepting ones), attempting to transition just leads to a dead state. In the final lasso automaton the initial state of the second DFA is omitted as it is not reachable.



This example gives an intuition for the construction. We turn this intuition into a formal construction, first defining the derivatives and showing some of their properties, and then proving that the lasso automaton we obtain accepts the desired language.

Definition 4.2. Define the *spoke* and *switch Brzozowski derivatives* $d_1 \colon \mathrm{Exp}_\circ \to \mathrm{Exp}_\circ^\Sigma$ and $d_2 \colon \mathrm{Exp}_\circ \to \mathrm{Exp}^\Sigma$ recursively on the structure of rational lasso expressions as:

$$\begin{aligned} d_1(0, a) &= 0 & d_2(0, a) &= 0 \\ d_1(t^\circ, a) &= 0 & d_2(t^\circ, a) &= d(t, a) \\ d_1(\rho + \sigma, a) &= d_1(\rho, a) + d_1(\sigma, a) & d_2(\rho + \sigma, a) &= d_2(\rho, a) + d_2(\sigma, a) \\ d_1(r \cdot \rho, a) &= d(r, a) \cdot \rho + [r \in N] \cdot d_1(\rho, a) & d_2(r \cdot \rho, a) &= [r \in N] \cdot d_2(\rho, a) \end{aligned}$$

The next two propositions give a better understanding of the relationship between the derivatives we just defined, and the equational theory of lasso algebras we defined earlier.

Proposition 4.3. *Let $\rho, \sigma \in Exp_\circ$ with $\vdash_{\mathbf{LA}} \rho = \sigma$. Then for all $a \in \Sigma$:*

1. $\vdash d_2(\rho, a) = d_2(\sigma, a)$ *and*
2. $\vdash_{\mathbf{LA}} d_1(\rho, a) = d_1(\sigma, a)$.

Proof. In order to prove this claim, we have to check all the relevant equations from our theory and make sure that it works for substitution. We only treat some of the equations from the theory as this involves simple manipulations of equations. The checks for substitution are readily verified and hence omitted.

For both derivatives we show the claim for the equations $t \cdot (r \cdot \rho) = (t \cdot r) \cdot \rho$ and $(t + r)^\circ = t^\circ + r^\circ$. When treating the first equation, we need this additional fact:

$$\vdash [t \in N] \cdot [r \in N] = [t \cdot r \in N].$$

We start with the derivative d_1 and check the first equation:

$$\begin{aligned}
d_1(t\cdot(r\cdot\rho)) &= d(t,a)\cdot(r\cdot\rho) + [t\in N]\cdot d_1(r\cdot\rho,a) && \text{(defn. } d_1)\\
&= d(t,a)\cdot(r\cdot\rho) + [t\in N]\cdot(d(r,a)\cdot\rho\\
&\qquad + [r\in N]\cdot d_1(\rho,a)) && \text{(defn. } d_1)\\
&= d(t,a)\cdot(r\cdot\rho) + [t\in N]\cdot(d(r,a)\cdot\rho)\\
&\qquad + [t\in N]\cdot([r\in N]\cdot d_1(\rho,a)) && \text{(dist.)}\\
&= (d(t,a)\cdot r)\cdot\rho + ([t\in N]\cdot d(r,a))\cdot\rho\\
&\qquad + ([t\in N]\cdot[r\in N])\cdot d_1(\rho,a) && \text{(mixed assoc.)}\\
&= (d(t,a)\cdot r + [t\in N]\cdot d(r,a))\cdot\rho\\
&\qquad + [t\cdot r\in N]\cdot d_1(\rho,a) && \text{(dist. \& fact)}\\
&= d(t\cdot r,a)\cdot\rho + [t\cdot r\in N]\cdot d_1(\rho,a) && \text{(defn. } d)\\
&= d_1((t\cdot r)\cdot\rho,a) && \text{(defn. } d_1)
\end{aligned}$$

For the other equation we get

$$d_1((t+r)^\circ,a) = 0 = 0+0 = d_1(t^\circ,a)+d_1(r^\circ,a) = d_1(t^\circ+r^\circ,a),$$

using the definition of d_1 and $\vdash_{\mathbf{LA}} 0+0=0$.

Next we turn to d_2. Again, for the first equation we have

$$\begin{aligned}
d_2(t\cdot(r\cdot\rho)) &= [t\in N]\cdot d_2(r\cdot\rho,a) && \text{(defn. } d_2)\\
&= [t\in N]\cdot([r\in N]\cdot d_2(\rho,a)) && \text{(defn. } d_2)\\
&= ([t\in N]\cdot[r\in N])\cdot d_2(\rho,a) && \text{(mixed assoc.)}\\
&= [t\cdot r\in N]\cdot d_2(\rho,a) && \text{(fact)}\\
&= d_2((t\cdot r)\cdot\rho,a). && \text{(defn. } d_2)
\end{aligned}$$

Finally, for the second equation we obtain

$$\begin{aligned}
d_2((t+r)^\circ,a) &= d(t+r,a) && \text{(defn. } d_2)\\
&= d(t,a)+d(r,a) && \text{(defn. } d)\\
&= d_2(t^\circ,a)+d_2(r^\circ,a) && \text{(defn. } d_2)\\
&= d_2(t^\circ+r^\circ,a). && \text{(defn. } d_2)
\end{aligned}$$

□

Proposition 4.4 (Fundamental Theorem). *Let $\rho\in Exp_\circ$. Then*

$$\vdash_{\mathbf{LA}} \rho = \left(\sum_{a\in\Sigma} a\cdot d_1(\rho,a)\right) + \left(\sum_{a\in\Sigma} a\cdot d_2(\rho,a)\right)^\circ.$$

Proof. We show the fundamental theorem by structural induction on rational lasso expressions.

Case 0:

$$\begin{aligned}
\sum_{a\in\Sigma} a\cdot d_1(0,a) + \left(\sum_{a\in\Sigma} a\cdot d_2(0,a)\right)^\circ &= \sum_a a\cdot 0 + \left(\sum_a a\cdot 0\right)^\circ && \text{(defn.)}\\
&= \sum_a 0 + \left(\sum_a 0\right)^\circ && \text{(absorp.)}\\
&= 0+0^\circ && \text{(idem.)}
\end{aligned}$$

$$
\begin{aligned}
&= 0 + 0 && (0^\circ = 0)\\
&= 0 && \text{(idem.)}
\end{aligned}
$$

Case t°:

$$
\begin{aligned}
\sum_{a\in\Sigma} a \cdot d_1(t^\circ, a) + \left(\sum_{a\in\Sigma} a \cdot d_2(t^\circ, a)\right)^\circ &= \sum_a a \cdot 0 + \left(\sum_a a \cdot d(t, a)\right)^\circ && \text{(defn.)}\\
&= \sum_a 0 + t^\circ && \text{(absorp. \& Prop. A.3 } (t \notin N))\\
&= 0 + t^\circ && \text{(idem.)}\\
&= t^\circ
\end{aligned}
$$

Case $\rho + \sigma$:

$$
\begin{aligned}
&\sum_{a\in\Sigma} a \cdot d_1(\rho + \sigma, a) + \left(\sum_{a\in\Sigma} a \cdot d_2(\rho + \sigma, a)\right)^\circ\\
&= \sum_a a \cdot (d_1(\rho, a) + d_1(\sigma, a)) + \left(\sum_a a \cdot (d_2(\rho, a) + d_2(\sigma, a))\right)^\circ && \text{(defn.)}\\
&= \sum_a (a \cdot d_1(\rho, a) + a \cdot d_1(\sigma, a)) + \left(\sum_a (a \cdot d_2(\rho, a) + a \cdot d_2(\sigma, a))\right)^\circ && \text{(dist.)}\\
&= \sum_a a \cdot d_1(\rho, a) + \sum_a a \cdot d_1(\sigma, a) + \left(\sum_a a \cdot d_2(\rho, a) + \sum_a a \cdot d_2(\sigma, a)\right)^\circ && \text{(assoc.)}\\
&= \sum_a a \cdot d_1(\rho, a) + \sum_a a \cdot d_1(\sigma, a) + \left(\sum_a a \cdot d_2(\rho, a)\right)^\circ + \left(\sum_a a \cdot d_2(\sigma, a)\right)^\circ && ((t + r)^\circ = t^\circ + r^\circ)\\
&= \sum_a a \cdot d_1(\rho, a) + \left(\sum_a a \cdot d_2(\rho, a)\right)^\circ + \sum_a a \cdot d_1(\sigma, a) + \left(\sum_a a \cdot d_2(\sigma, a)\right)^\circ && \text{(comm.)}\\
&= \rho + \sigma && \text{(defn.)}
\end{aligned}
$$

Case $r \cdot \rho$:

$$
\begin{aligned}
&\sum_{a\in\Sigma} a \cdot d_1(r \cdot \rho, a) + \left(\sum_{a\in\Sigma} a \cdot d_2(r \cdot \rho, a)\right)^\circ\\
&= \sum_a a \cdot (d(r, a) \cdot \rho + [r \in N] \cdot d_1(\rho, a)) + \left(\sum_a a \cdot [r \in N] \cdot d_2(\rho, a)\right)^\circ && \text{(defn.)}\\
&= \sum_a a \cdot (d(r, a) \cdot \rho) + \sum_a a \cdot ([r \in N] \cdot d_1(\rho, a)) + \left(\sum_{a\in\Sigma} a \cdot [r \in N] \cdot d_2(\rho, a)\right)^\circ && \text{(assoc.)}\\
&= \sum_a (a \cdot d(r, a)) \cdot \rho + [r \in N] \cdot \sum_a a \cdot d_1(\rho, a) + [r \in N] \cdot \left(\sum_{a\in\Sigma} a \cdot d_2(\rho, a)\right)^\circ\\
&= \left(\sum_a a \cdot d(r, a)\right) \cdot \rho + [r \in N] \cdot \left(\sum_a a \cdot d_1(\rho, a) + \left(\sum_{a\in\Sigma} a \cdot d_2(\rho, a)\right)^\circ\right) && \text{(dist.)}\\
&= \left(\sum_a a \cdot d(r, a)\right) \cdot \rho + [r \in N] \cdot \rho && \text{(I.H.)}
\end{aligned}
$$

$$= \left(\left(\sum_a a \cdot d(r,a) \right) + [r \in N] \right) \cdot \rho \qquad \text{(dist.)}$$
$$= r \cdot \rho \qquad \text{(Prop. A.3)}$$

In this derivation we used the additional facts

$$\vdash t \cdot [r \in N] = [r \in N] \cdot t \qquad \text{and} \qquad \vdash_{\mathbf{LA}} ([r \in N] \cdot t)^\circ = [r \in N] \cdot t^\circ. \qquad \square$$

Using all the properties of rational lasso expressions and the derivatives, we obtain our first Brzozowski lasso automaton.

Proposition 4.5. *We call* $\mathcal{C} = (Exp_\circ, Exp, d_1, d_2, d, N)$ *the* Brzozowski lasso automaton*. If* $\rho \in Exp_\circ$*, then* $L_\circ(\mathcal{C}, \rho) = [\![\rho]\!]_\circ$*.*

Proof. We proceed by induction on the length of the spoke word. For the base case assume the spoke word to be the empty word, i.e. consider the lasso (ε, au). By using the Fundamental Theorem (Proposition 4.4) and Soundness (Proposition 3.6) we get:

$$\begin{aligned}
(\varepsilon, au) \in [\![\rho]\!]_\circ &\iff (\varepsilon, au) \in \left[\!\!\left[\sum_a a \cdot d_1(\rho, a) + \left(\sum_a a \cdot d_2(\rho, a) \right)^\circ \right]\!\!\right]_\circ \\
&\iff (\varepsilon, au) \in \left[\!\!\left[\left(\sum_a a \cdot d_2(\rho, a) \right)^\circ \right]\!\!\right]_\circ \\
&\iff au \in \left[\!\!\left[\sum_a a \cdot d_2(\rho, a) \right]\!\!\right] \\
&\iff au \in [\![a \cdot d_2(\rho, a)]\!] \\
&\iff u \in [\![d_2(\rho, a)]\!] \\
&\iff \varepsilon \in [\![d(d_2(\rho, a), u)]\!] \\
&\iff d(d_2(\rho, a), u) \in N \\
&\iff (\varepsilon, au) \in L_\circ(\mathcal{C}, \rho).
\end{aligned}$$

This establishes the base case. For the induction step consider the lasso (au, v).

$$\begin{aligned}
(au, v) \in [\![\rho]\!]_\circ &\iff (au, v) \in \left[\!\!\left[\sum_a a \cdot d_1(\rho, a) + \left(\sum_a a \cdot d_2(\rho, a) \right)^\circ \right]\!\!\right]_\circ \\
&\iff (au, v) \in \left[\!\!\left[\sum_a a \cdot d_1(\rho, a) \right]\!\!\right]_\circ \\
&\iff (au, v) \in [\![a \cdot d_1(\rho, a)]\!]_\circ \\
&\iff (u, v) \in [\![d_1(\rho, a)]\!]_\circ \\
&\iff (u, v) \in L_\circ(\mathcal{C}, d_1(\rho, a)) \\
&\iff (au, v) \in L_\circ(\mathcal{C}, \rho).
\end{aligned}$$

Here the induction hypothesis was used from the antepenultimate to the penultimate line. $\square$

The Brzozowski lasso automaton is not necessarily finite. The easiest fix to this is to assume that our rational lasso expressions are disjunctive forms (c.f. Proposition 3.7). To accommodate this assumption with regards to our Brzozowski lasso automaton, we slightly adapt the definition of d_1 so that $d_1(\rho, a)$ is again a disjunctive form for $a \in \Sigma$. This is done by defining

$$d_1(t \cdot s^\circ, a) := d(t, a) \cdot s^\circ.$$

It is important to point out that this does not jeopardise our earlier result (Proposition 4.5). Indeed without this change we would have

$$d_1(t \cdot s^\circ, a) = d(t,a) \cdot s^\circ + [t \in N] \cdot d(s^\circ, a) = d(t,a) \cdot s^\circ + [t \in N] \cdot 0,$$

which is provably equivalent to $d(t,a) \cdot s^\circ$. We ignore this technicality in pursuit of a cleaner presentation.

It should also be remarked, that with these changes d_1 only acts on the spoke expressions (where it acts just like a normal Brzozowski derivative), d_2 acts on the loop expression to give a rational expression (provided that the spoke expression has the empty word property) and finally d acts on the obtained rational expression. In this sense, we are really constructing two DAs using Brzozowski derivatives and linking them appropriately with d_2, as in Example 4.1.

In the remainder of this section, we quotient our Brzozowski lasso automaton by a suitable equivalence relation which respects the derivatives, and such that for any rational lasso expression, its collection of successors is finite. Using our assumption that rational lasso expressions are in a disjunctive form, the definition makes use of the equivalence relation $\sim_B$ (Definition A.5).

Definition 4.6. We define $\sim_C \subseteq \mathrm{Exp}_\circ^2$ to be the least equivalence relation such that

$$\sum_{i=1}^{n} t_i \cdot r_i^\circ \sim_C \sum_{i=1}^{n} t'_i \cdot (r'_i)^\circ \iff \forall\ 1 \leq i \leq n : t_i \sim_B t'_i \text{ and } r_i = r'_i.$$

Next, we show that defining our Brzozowski derivatives on the equivalence classes is well-defined.

Proposition 4.7. *The following two maps are well defined:*

$$\begin{aligned} \widehat{d_1} : \mathit{Exp}_\circ/{\sim_C} &\longrightarrow (\mathit{Exp}_\circ/{\sim_C})^\Sigma \\ [\rho]_{\sim_C} &\longmapsto \lambda a \in \Sigma.[d_1(\rho,a)]_{\sim_C} \end{aligned} \qquad \begin{aligned} \widehat{d_2} : \mathit{Exp}_\circ/{\sim_C} &\longrightarrow (\mathit{Exp}/{\sim_B})^\Sigma \\ [\rho]_{\sim_C} &\longmapsto \lambda a \in \Sigma.[d_2(\rho,a)]_{\sim_B} \end{aligned}$$

Proof. Let $\sum_{i=1}^n t_i \cdot s_i^\circ \sim_C \sum_{i=1}^n t'_i \cdot (s'_i)^\circ$. Then

$$d_1\left(\sum_{i=1}^{n} t_i \cdot s_i^\circ, a\right) = \sum_{i=1}^{n} d(t_i,a) \cdot s_i^\circ \qquad \text{and} \qquad d_1\left(\sum_{i=1}^{n} t'_i \cdot (s'_i)^\circ, a\right) = \sum_{i=1}^{n} d(t'_i,a) \cdot (s'_i)^\circ.$$

By assumption, $t_i \sim_B t'_i$, hence $d(t_i,a) \sim_B d(t'_i,a)$ by Lemma A.6. Furthermore, by assumption $s_i = s'_i$ and so

$$d_1\left(\sum_{i=1}^{n} t_i \cdot s_i^\circ, a\right) \sim_C d_1\left(\sum_{i=1}^{n} t_i \cdot s_i^\circ, a\right).$$

Next, by the definition of d_2,

$$d_2\left(\sum_{i=1}^{n} t_i \cdot s_i^\circ, a\right) = \sum_{i=1}[t_i \in N] \cdot d(s_i,a) \qquad \text{and} \qquad d_2\left(\sum_{i=1}^{n} t'_i \cdot (s'_i)^\circ, a\right) = \sum_{i=1}[t'_i \in N] \cdot d(s'_i,a).$$

By assumption $t_i \sim_B t'_i$ so that $[t_i \in N] = [t'_i \in N]$ by Lemma A.6. Moreover, $d(s_i,a) = d(s'_i,a)$ follows from $s_i = s'_i$, hence

$$d_2\left(\sum_{i=1}^{n} t_i \cdot s_i^\circ, a\right) = d_2\left(\sum_{i=1}^{n} t'_i \cdot (s'_i)^\circ, a\right)$$

and so they are in particular related by $\sim_B$. $\square$

The map sending a rational lasso expression to its equivalence class preserves the accepted language. Moreover, if $\rho \in \mathrm{Exp}_\circ$ then there are only finitely many states reachable from $[\rho]_{\sim_C}$. We get the following theorem, and it follows immediately that every rational lasso language is regular.

Theorem 4.8. *The lasso automaton* $\widehat{\mathcal{C}} = (Exp_\circ/\sim_C, Exp/\sim_B, \widehat{d_1}, \widehat{d_2}, \widehat{d}, \widehat{N})$ *satisfies for all* $\rho \in Exp_\circ$ *(ρ a disjunctive form):*

1. $L_\circ(\widehat{\mathcal{C}}, [\rho]_{\sim_C}) = [\![\rho]\!]_\circ$ *and*
2. *the set of states reachable from* $[\rho]_{\sim_C}$ *is finite.*

Proof. The first item follows as $\sim_C$ is a congruence compatible with the structure of the automaton, hence the map sending a state to its equivalence class preserves the accepted language.

The second item follows by the definition of $\sim_C$. More precisely, we defined it in terms of $\sim_B$ for which we already know that taking Brzozowski derivatives leads to finitely many reachable states. The fact that $\sim_C$ is given by equality on expressions underneath $(-)^\circ$ means that they do not contribute anything. $\square$

5 Rational Lasso Expressions from Finite Lasso Automata

The previous section gives us one direction of Kleene's theorem for lasso languages. We accomplish the other direction by slightly modifying a result by [4] which is interesting in its own right.

In [4], the authors show how to build the rational ω-language accepted by an Ω-automaton $\mathcal{A}$ from some rational languages defined on the basis of $\mathcal{A}$. This also shows that every regular ω-language is rational.

We first state the results (Lemma 5.1 and Theorem 5.2) by [4] and then follow it up with a similar result for lasso languages which shows that every regular lasso language is rational. The next lemma is of a technical nature.

Lemma 5.1 ([4]). *Let* U, V *be regular languages such that* $\varepsilon \notin V$, $UV^* = U$ *and* $V^+ = V$*. Then for all* $uv^\omega \in UV^\omega$*, there exist* $u' \in U$ *and* $v' \in V$ *such that* $uv^\omega = u'v'^\omega$.

Theorem 5.2 ([4]). *Let* $\mathcal{A} = (X, Y, \overline{x}, \delta_1, \delta_2, \delta_3, F)$ *be a finite* Ω*-automaton. For* $x \in X$ *and* $y \in F$ *we abbreviate*

$$S_x = \{u \in \Sigma^* \mid \delta_1(\overline{x}, u) = x\},$$
$$R_{x,y} = \{u \in \Sigma^+ \mid \delta_1(x, u) = x \text{ and } (\delta_2 : \delta_3)(x, u) = \delta_3(y, u) = y\}.$$

Then

$$L_\omega(\mathcal{A}) = \bigcup_{x \in X} \bigcup_{y \in F} S_x \cdot R^\omega_{x,y}.$$

The rational language S_x consists of all words that lead from the initial state $\overline{x}$ to x. As $R_{x,y}$ contains those words which at the same time bring us from x back to x via δ_1, from x to y via $(\delta_2 : \delta_3)$ and from y back to itself via δ_3, concatenating infinitely many such words traces in some sense paths which intersect a final state infinitely often. As such, a saturated lasso automaton can be seen as a special kind of Büchi automaton ([5]).

As S_x and $R_{x,y}$ are both rational languages, we can build rational expression from them and the next corollary follows immediately.

Corollary 5.3. *Given a finite* Ω*-automaton* $\mathcal{A}$*, we can construct an rational* ω*-expression* T *such that* $L_\omega(\mathcal{A}) = [\![T]\!]_\omega$.

Proof. The languages S_x and $R_{x,y}$ constructed in Theorem 5.2 are regular, hence we can find matching rational expressions which allow us to define a suitable rational ω-expression. $\square$

In order to obtain a similar result for lasso languages, we slightly modify the definition of $R_{x,y}$ from Theorem 5.2. This shows that every regular lasso language is rational.

Proposition 5.4. *Let $\mathcal{A} = (X, Y, \overline{x}, \delta_1, \delta_2, \delta_3, F)$ be a finite lasso automaton. For $x \in X$ and $y \in F$, let S_x be defined as in Theorem 5.2 and let*

$$R_{x,y} = \{u \in \Sigma^+ \mid (\delta_2 : \delta_3)(x, u) = y\}.$$

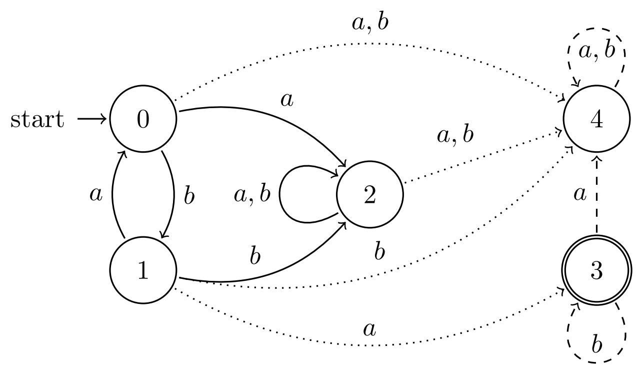
Then

$$L_\circ(\mathcal{A}) = \bigcup_{x \in X} \bigcup_{y \in F} S_x \cdot R^\circ_{x,y}.$$

Proof. If $(u, v) \in L_\circ(\mathcal{A})$, then $(\delta_2 : \delta_3)(\delta_1(\overline{x}, u), v) \in F$. Choose $x = \delta_1(\overline{x}, u)$ and $y = (\delta_2 : \delta_3)(x, v) \in F$, then $(u, v) \in S_x \cdot R^\circ_{x,y}$.

For the other inclusion, let $(u, v) \in S_x \cdot R^\circ_{x,y}$ for some $x \in X$ and $y \in F$. By definition, we have $x = \delta_1(\overline{x}, u)$ and $y = (\delta_2 : \delta_3)(x, v)$. Hence $(\delta_2 : \delta_3)(\delta_1(\overline{x}, u), v) \in F$ and $(u, v) \in L_\circ(\mathcal{A})$. $\square$

Example 5.5. In this example we extract a rational lasso expression from the lasso automaton in Example 4.1 shown below:



For each spoke state in $\{0, 1, 2\}$ we compute S_x, giving

$$S_0 = (ba)^*, \qquad S_1 = b(ab)^*, \qquad S_2 = (ba)^* a(a+b)^* + b(ab)^* b(a+b)^*.$$

Next we compute $R_{x,y}$, which results in

$$R_{0,3} = 0, \qquad R_{1,3} = ab^*, \qquad R_{2,3} = 0.$$

So the language accepted by the lasso automaton (which we call $\mathcal{A}$) is:

$$L_\circ(\mathcal{A}) = (ba)^* 0^\circ + b(ab)^*(ab^*)^\circ + [(ba)^* a(a+b)^* + b(ab)^* b(a+b)^*]\, 0^\circ.$$

Note that this is provably equivalent to $b(ab)^*(ab^*)^\circ$ which is exactly what we started out with in Example 4.1.

Corollary 5.6. *Given a finite lasso automaton $\mathcal{A}$, we can construct a rational lasso expression ρ such that $L_\circ(\mathcal{A}) = [\![\rho]\!]_\circ$.*

Proof. The proof is analogous to that of Corollary 5.3. $\square$

Corollary 5.6 and Theorem 4.8 give us a Kleene theorem for lasso languages.

Theorem 5.7. *A lasso language is regular if and only if it is rational.*

Proof. This theorem follows immediately from Corollary 5.6 and Theorem 4.8. □

6 Rational ω- and Lasso Expressions

In this section we explore the connection between rational lasso and ω-expressions, and define what it means for a rational lasso expression τ to represent a rational ω-expression T. Intuitively, this notion expresses that $[\![\tau]\!]_\circ$ and $[\![T]\!]_\omega$ completely determine each other. We show that for each T, we can construct τ in a syntactic way from T with the help of two additional operations on rational expressions. If we pair this with the Brzozowski construction, this shows that every rational ω-language is accepted by a finite Ω-automaton. By [4], we know that the converse also holds. This re-establishes the Kleene theorem for ω-languages with respect to Ω-automata. Additionally, we also obtain a more direct method for constructing Ω-automata from rational ω-expressions which has not been done before to the best of our knowledge.

Definition 6.1. Let $f\colon \Sigma^{*+} \to \Sigma^{\mathrm{up}}$ be given by $(u, v) \mapsto uv^\omega$. For a rational lasso expression $\tau \in \mathrm{Exp}_\circ$ and a rational ω-expression $T \in \mathrm{Exp}_\omega$, we say that

1. τ *weakly represents* T if $f\,[[\![\tau]\!]_\circ] = \mathrm{UP}([\![T]\!]_\omega)$ and
2. τ *represents* T if $[\![\tau]\!]_\circ = f^{-1}\left(\mathrm{UP}([\![T]\!]_\omega)\right)$.

Example 6.2. Let $T = (a + b)^* a^\omega$. Then $((a + b)^*, a)$ constitutes a weak representation but not a representation of T. This is seen from the following facts:

1. $f\,[[\![((a + b)^*, a)]\!]_\circ] = f[\{(u, a) \mid u \in \Sigma^*\}] = \{ua^\omega \mid u \in \Sigma^*\} = \mathrm{UP}([\![T]\!]_\omega)$,
2. $f(\varepsilon, aa) = a^\omega \in \mathrm{UP}([\![T]\!]_\omega)$ but $(\varepsilon, aa) \notin [\![((a + b)^*, a)]\!]_\circ$.

Remark 6.3. If τ represents T it also weakly represents T as f is surjective. Conversely, if τ weakly represents T, then it represents T if and only if $[\![\tau]\!]_\circ = f^{-1}(f[[\![\tau]\!]_\circ])$, that is if and only if $[\![\tau]\!]_\circ$ is $\sim_\gamma$-saturated. For the previous example, $((a + b)^*, a^+)$ is a weak representation of T and moreover its semantics is $\sim_\gamma$-saturated, hence it is a representation of T.

According to the last remark, in order to obtain a representation of T, we can start with a weak representation and then syntactically modify it such that its semantics is $\sim_\gamma$-saturated, all the while making sure that it stays a weak representation. This is our approach throughout this section. Constructing a weak representation is not a very difficult task, but our goal is to be a bit clever about this and try to construct a weak representation which is close to a representation. This hopefully reduces how much we have to syntactically modify it.

The following two properties of $(-)^\omega$ play an important motivational role.

$$u(vu)^\omega = (uv)^\omega \quad (1) \qquad\qquad (u^k)^\omega = u^\omega \quad (k \geq 1) \quad (2)$$

For a given $\rho \in \mathrm{Exp}_\circ$, $[\![\rho]\!]_\circ$ is $\sim_\gamma$-saturated if it is closed under γ-reduction and γ-expansion, which mimic Equations 1 and 2. So if we have $\rho \in \mathrm{Exp}_\circ$ whose semantics is not saturated, the idea is to add additional terms τ_i which, on a semantic level, add the necessary lassos. We show how to derive the tools we need for this endeavour using the two equations (but this can equivalently be done looking at the rewrite rules γ_1 and γ_2).

We start by looking at Equation 1, replacing the ultimately periodic words with lassos. From the left-to-right direction, the question we pose is: Assume $(u, vu) \in [\![\rho]\!]_\circ$, what τ do we have to add to

ρ such that $(\varepsilon, uv) \in [\![\rho + \tau]\!]_\circ$? We may simplify this a little bit and assume that $(u, vu) \in [\![tr^\circ]\!]_\circ$, in which case $u \in [\![t]\!]$ and $vu \in [\![r]\!]$. The idea is to split r into two rational expressions r_1, r_2 such that $v \in [\![r_1]\!]$ and $u \in [\![r_2]\!]$. In this manner we find a suitable τ to be $((r_2 \cap t)r_1)^\circ$. This relies on us being able to syntactically obtain such a split, and on the assumption that we have access to an additional operation on rational lasso expressions, namely $\cap$. We later introduce the *sequential splitting relation* which provides us with the splits in a syntactic way. For the right-to-left direction, with similar simplifications, we have $(\varepsilon, uv) \in [\![t^\circ]\!]_\circ$ and try to find a suitable τ to include (u, vu). We make anew use of splits, by splitting t into suitable t_1, t_2 with $u \in [\![t_1]\!]$ and $v \in [\![t_2]\!]$, in which case our candidate τ is $t_1(t_2t_1)^\circ$. To conclude, in order to mimic Equation 1, we use splits and $\cap$ (which we assume to have on top of the other rational operations).

Our approach for Equation 2 is going to be less involved, but requires also an additional operation which is non-standard. For a rational language U, the root of U is $\sqrt{U} = \{u \mid u^k \in U, k \geq 1\}$ ([10]). As $\sqrt{-}$ preserves rationality, we assume that we have an operation $\sqrt{-}$ on rational expressions. Let $t \in \mathrm{Exp}$. Then we can show that $(\varepsilon, u) \in [\![(\sqrt{t^+})^\circ]\!]_\circ \iff (\varepsilon, u^k) \in [\![(\sqrt{t^+})^\circ]\!]_\circ$. So we can take care of Equation 2 by introducing the transitive closure and the root underneath $(-)^\circ$. We remark that the order is important as $(-)^+$ and $\sqrt{-}$ do not commute.

There is one more subtle point that must be made. It should be clear that these various ideas give us everything we require (as we show later). What is maybe less clear is that they do not give us more than what we want. For instance taking a transitive closure t^+ now allows for the finite concatenation of arbitrary words $u_i \in [\![t]\!]$ and not just a finite concatenation of a single word u^k which is what we initially set out for. Fortunately, with all these ideas we can show that we do not get more than what we want. Taking the transitive closure is for instance permitted because $t^\omega = (t^+)^\omega$.

The next step is to formally define the idea of splits.

Definition 6.4 ([7, 8]). Let $\nabla : \mathrm{Exp} \to 2^{\mathrm{Exp} \times \mathrm{Exp}}$ be defined recursively on the structure of rational expressions as:

$$\nabla(0) = \emptyset \qquad \nabla(1) = \{(1,1)\} \qquad \nabla(a) = \{(1,a),(a,1)\}$$

$$\nabla(t + r) = \nabla(t) \cup \nabla(r) \qquad \nabla(t^*) = \{(t^* \cdot t_0, t_1 \cdot t^*) \mid (t_0, t_1) \in \nabla(t)\}$$

$$\nabla(t \cdot r) = \{(t_0, t_1 \cdot r) \mid (t_0, t_1) \in \nabla(t)\} \cup \{(t \cdot r_0, r_1) \mid (r_0, r_1) \in \nabla(r)\}$$

We write ∇_t for $\nabla(t)$ and call ∇_t the *sequential splitting relation of* t.

The following lemma establishes some properties of the splitting relation, of which points 1. and 2. are taken from [8].

Lemma 6.5. *Let $t \in Exp$. The sequential splitting relation ∇_t satisfies:*

1. *$|\nabla_t|$ is finite and $\forall (t_0, t_1) \in \nabla_t \colon t_0 \cdot t_1 \leq t$,*
2. *if $u \cdot v \in [\![t]\!]$, then there is a split $(t_0, t_1) \in \nabla_t$ such that $u \in [\![t_0]\!]$ and $v \in [\![t_1]\!]$,*
3. *if $(t_0, t_1) \in \nabla_t$ and $(r_0, r_1) \in \nabla_{t_1}$, then there exists $(s_0, s_1) \in \nabla_t$ such that $t_0 \cdot r_0 \leq s_0$ and $r_1 \leq s_1$.*

Proof. The first two items follow by [8], so we only focus on the third item which we show by structural induction on rational expressions. For $t = 0$, $\nabla_0 = \emptyset$ and so it trivially holds. For $t = 1$, $\nabla_1 = \{(1,1)\}$ and so there is only one choice for (t_0, t_1) and (r_0, r_1), namely the pair $(1,1)$. As $1 \cdot 1 \leq 1$ and $1 \leq 1$, we are done. For $t = a$, there are three cases. First, let $(t_0, t_1) = (a, 1)$, then $(r_0, r_1) = (1,1)$ and we have $a \cdot 1 \leq a$ and $1 \leq 1$. For the other two cases, let $(t_0, t_1) = (1, a)$. If $(r_0, r_1) = (1, a)$, then we are done as $1 \cdot 1 \leq 1$ and $a \leq a$. On the other hand, for $(r_0, r_1) = (a, 1)$ we have $1 \cdot a \leq a$ and $1 \leq 1$. This covers all the bases cases.

For the induction steps, we start with $t \cdot r$. There are two cases two consider, the first split can either be $(t_0, t_1 \cdot r)$ or $(t \cdot r_0, r_1)$. For the case $(t_0, t_1 \cdot r)$, there are again two subcases, namely the splits $(t_{10}, t_{11} \cdot r)$ and $(t_1 \cdot r_0, r_1)$ in $\nabla_{t_1 \cdot r}$. For the first subcase, we know that $(t_{10}, t_{11}) \in \nabla_{t_1}$, so by the induction hypothesis, there exists a split $(s_0, s_1) \in \nabla_t$ such that $t_0 \cdot t_{10} \leq s_0$ and $t_{11} \leq s_1$. It follows that $(s_0, s_1 \cdot r) \in \nabla_{t \cdot r}$ is a suitable candidate as $t_0 \cdot t_{10} \leq s_0$ and $t_{11} \cdot r \leq s_1 \cdot r$. For the second subcase, we have $(r_0, r_1) \in \nabla_r$, hence $(t \cdot r_0, r_1) \in \nabla_{t \cdot r}$ works as $t_0 \cdot t_1 \cdot r_0 \leq t \cdot r_0$ and $r_1 \leq r_1$. Finally, for the case $(t \cdot r_0, r_1)$, let $(r_{10}, r_{11}) \in \nabla_{r_1}$. Then, by the induction hypothesis, there exists a split $(s_0, s_1) \in \nabla_r$ with $r_0 \cdot r_{10} \leq s_0$ and $r_{11} \leq s_1$. Our candidate is $(t \cdot s_0, s_1)$ which works as $t \cdot r_0 \cdot r_{10} \leq t \cdot s_0$ and $r_{11} \leq s_1$.

The next induction step is $t_0 + t_1$. Let $(t, t') \in \nabla_{t_0 + t_1}$, then $(t, t') \in \nabla_{t_i}$ for some $i \in \{0, 1\}$. Next, take $(r_0, r_1) \in \nabla_{t'}$. By the induction hypothesis we find $(s_0, s_1) \in \nabla_{t_i}$ such that $t \cdot r_0 \leq s_0$ and $r_1 \leq s_1$. As $(s_0, s_1) \in \nabla_{t_i}$, we also have $(s_0, s_1) \in \nabla_{t_0 + t_1}$. Choosing (s_0, s_1) suffices to satisfy the claim.

For the last case, we consider the expression t^*. Let $(t^* \cdot t_0, t_1 \cdot t^*) \in \nabla_{t^*}$, where $(t_0, t_1) \in \nabla_t$. There are two cases, as the splits in $\nabla_{t_1 \cdot t^*}$ are either of the form $(t_1 \cdot t^* \cdot r_0, r_1 \cdot t^*)$ (with $(t^* \cdot r_0, r_1 \cdot t^*) \in \nabla_{t^*}$) or $(t_{10}, t_{11} \cdot t^*)$ (with $(t_{10}, t_{11}) \in \nabla_{t_1}$). For the first case, we have $(r_0, r_1) \in \nabla_t$ and so we choose as candidate the split $(t^* \cdot r_0, r_1 \cdot t^*) \in \nabla_{t^*}$. Then

$$t^* \cdot t_0 \cdot t_1 \cdot t^* \cdot r_0 \leq t^* \cdot t \cdot t^* \cdot r_0 \leq t^* \cdot r_0,$$

and also $r_1 \cdot t^* \leq r_1 \cdot t^*$. For the final case of $(t_{10}, t_{11} \cdot t^*)$, we can use the induction hypothesis to find $(s_0, s_1) \in \nabla_t$ with $t_0 \cdot t_{10} \leq s_0$ and $t_{11} \leq s_1$. Choosing $(t^* \cdot s_0, s_1 \cdot t^*)$ suffices as $t^* \cdot t_0 \cdot t_{10} \leq t^* \cdot s_0$ and $t_{11} \cdot t^* \leq s_1 \cdot t^*$. This finishes the proof. □

Example 6.6. For this example we explore the sequential splitting relation of a simple rational expression and briefly touch on the different properties outlined in Lemma 6.5. Let $t = b(a + b^*)$. Then

$$\nabla_t = \{(1, b(a + b^*)), (b, 1(a + b^*)), (b1, a), (ba, 1), (bb^*1, bb^*), (bb^*b, 1b^*), (b1, 1)\}.$$

If we choose a word in $[\![t]\!]$, say $bb \cdot b$, then we can find a corresponding split, in this case $(bb^*b, 1b^*)$ for example. Moreover, if we choose any split in ∇_t, say $(bb^*b, 1b^*)$, and take any split in ∇_{1b^*}, say $(1b^*1, bb^*)$, then we can find a split $(t_0, t_1) \in \nabla_t$ such that $bb^*b1b^*1 \leq t_0$ and $bb^* \leq t_1$, namely (bb^*1, bb^*).

The rest of the section takes the ingredients and ideas we have gathered so far and assembles them in the right order. We define a function h which we show to map a rational ω-expression to a weak representation. The most challenging part of the definition is that of t^ω. In its definition we use the sequential splitting relation to simulate one direction of Property 1, and also apply the idea of the transitive closure (by using Kleene stars) to obtain one direction of Property 2.

Definition 6.7. Let $h : \mathrm{Exp}_\omega \to \mathrm{Exp}_\circ$ be defined as

$$h(0) = 0 \qquad\qquad h(t^\omega) = \sum_{(t_0, t_1) \in \nabla_t} (t^* \cdot t_0, t_1 \cdot t^* \cdot t_0)$$

$$h(T_1 + T_2) = h(T_1) + h(T_2) \qquad\qquad h(t \cdot T) = t \cdot h(T)$$

Rational lasso expressions can only contain finite sums, so it is crucial in the definition of $h(t^\omega)$ that $|\nabla_t|$ is finite (Lemma 6.5).

Proposition 6.8. *Let $T \in Exp_\omega$. Then $h(T)$ weakly represents T.*

Proof. This is shown by structural induction on rational ω-expressions. The case $T = 0$ is trivial. For $T = t^\omega$, let first $uv^\omega \in [\![T]\!]_\omega$. Then

$$uv^\omega \in [\![t]\!]^*([\![t]\!]^+)^\omega$$

and by Lemma 5.1 there exist u', v' with $uv^\omega = u'v'^\omega$, $u' \in [\![t]\!]^*$ and $v' \in [\![t]\!]^+$. Let $v' = v_1 w$ with $v_1 \in [\![t]\!]$, which gives rise to a split $(t_0, t_1) \in \nabla_t$ with $\varepsilon \in [\![t_0]\!]$. It is quickly verified that

$$(u', v') \in [\![(t^* t_0, t_1 t^* t_0)]\!]_\circ \subseteq [\![h(t^\omega)]\!]_\circ.$$

Hence $uv^\omega = u'v'^\omega \in f[[\![h(t^\omega)]\!]_\circ]$. For the other direction, let $(u, v) \in [\![h(t^\omega)]\!]_\circ$, then there exists a split $(t_0, t_1) \in \nabla_t$ with $u \in [\![t^* t_0]\!]$ and $v \in [\![t_1 t^* t_0]\!]$. Now

$$[\![t^* t_0]\!] \cdot ([\![t_1 t^* t_0]\!])^\omega = [\![t^* t_0 t_1]\!] \cdot ([\![t^* t_0 t_1]\!])^\omega \subseteq [\![t^\omega]\!]_\omega,$$

hence $f(u, v) = uv^\omega \in [\![t^\omega]\!]_\omega$.

For the induction steps, we first consider the case $T_1 + T_2$. For the right to left inclusion, let $uv^\omega \in [\![T_1 + T_2]\!]_\omega$, then $uv^\omega \in [\![T_i]\!]_\omega$ for $i \in \{1, 2\}$. By the induction hypothesis, there exists some $(u', v') \in [\![h(T_i)]\!]_\circ$ with $(u, v) \sim_\gamma (u', v')$. Then

$$(u', v') \in [\![h(T_1)]\!] \cup [\![h(T_2)]\!]_\circ = [\![h(T_1) + h(T_2)]\!]_\circ = [\![h(T_1 + T_2)]\!]_\circ$$

as required. For the other inclusion, let $(u, v) \in [\![h(T_1 + T_2)]\!]_\circ$, then $(u, v) \in [\![h(T_i)]\!]_\circ$ for some $i \in \{1, 2\}$. By the induction hypothesis, $uv^\omega \in [\![T_i]\!]_\omega \subseteq [\![T_1 + T_2]\!]_\omega$.

The last case is that of $t \cdot T$. For the right to left inclusion, take $u_0 u_1 v^\omega \in [\![t \cdot T]\!]_\omega$ where $u_0 \in [\![t]\!]$ and $u_1 v^\omega \in [\![T]\!]_\omega$. By the induction hypothesis, there exists $(u', v') \in [\![h(T)]\!]_\circ$ such that $uv^\omega = u'v'^\omega$. It follows that

$$(u_0 u', v') \in [\![t \cdot h(T)]\!]_\circ = [\![h(t \cdot T)]\!]_\circ$$

and moreover, $u_0 u' v'^\omega = u_0 u_1 v^\omega$ and we are done. For the other inclusion, let $(u_0 u_1, v) \in [\![h(t \cdot T)]\!]_\circ$, where $u_0 \in [\![t]\!]$ and $(u_1, v) \in [\![h(T)]\!]_\circ$. By the induction hypothesis, $u_1 v^\omega \in [\![T]\!]_\omega$ and so $u_0 u_1 v^\omega \in [\![t \cdot T]\!]_\omega$ which concludes the proof. $\square$

Additionally to this function giving us a weak representation, it should not come as a surprise that the language semantics of the rational lasso expression we obtain is closed under γ-expansion. Indeed, the direction of both equations we included expand words, u^ω is expanded to $(u^k)^\omega$ and $(uv)^\omega$ to $u(vu)^\omega$.

Proposition 6.9. *Let $T \in Exp_\omega$. Then $[\![h(T)]\!]_\circ$ is closed under γ-expansion.*

Proof. We proceed by structural induction on rational ω-expressions. The first base case $T = 0$ is trivial as $[\![h(0)]\!]_\circ = \emptyset$. For the second, $T = t^\omega$. We show that $[\![h(t^\omega)]\!]_\circ$ is closed under γ_1- and γ_2-expansion. The result then follows as these commute. For γ_1-expansion, let $(ua, va) \to_{\gamma_1} (u, av)$ and $(u, av) \in [\![h(t^\omega)]\!]_\circ$. Then there exists a split $(t_0, t_1) \in \nabla_t$ such that $u \in [\![t^* \cdot t_0]\!]$ and $av \in [\![t_1 \cdot t^* \cdot t_0]\!]$. Hence we can find $v_1, \ldots, v_k \in \Sigma^*$ with $v_1 \in [\![t_1]\!]$, $v_2, \ldots, v_{k-1} \in [\![t^*]\!]$, $v_k \in [\![t_0]\!]$ and $v_1 v_2 \ldots v_k = av$.

We need to distinguish three subcases:

1. If $v_1 \neq \varepsilon$, then $v_1 = av'_1$ and this induces a split $(r_0, r_1) \in \nabla_{t_1}$, where $a \in [\![r_0]\!]$ and $v'_1 \in [\![r_1]\!]$. By Lemma 2.3, there exists a split $(s_0, s_1) \in \nabla_t$ such that $t_0 \cdot r_0 \leq s_0$ and $r_1 \leq s_1$. We now have that
$$ua \in [\![t^* \cdot t_0 \cdot r_0]\!] \subseteq [\![t^* \cdot s_0]\!] \qquad \text{and} \qquad va \in [\![r_1 \cdot t^* \cdot t_0 \cdot r_0]\!] \subseteq [\![s_1 \cdot t^* \cdot s_0]\!].$$
Hence $(ua, va) \in [\![h(t^\omega)]\!]_\circ$.

2. If $v_1 = \varepsilon$ and $k = 2$, we have that $v_2 = av$, and hence a split $(r_0, r_1) \in \nabla_{t_0}$ where $a \in [\![r_0]\!]$ and $v \in [\![r_1]\!]$. Moreover, as $v_1 = \varepsilon$, it follows that $1 \leq t_1$ and so $t_0 = t_0 \cdot 1 \leq t_0 \cdot t_1 \leq t$. Thus
$$ua \in [\![t^* \cdot t_0 \cdot r_0]\!] \subseteq [\![t^* \cdot e \cdot r_0]\!] \subseteq [\![t^* \cdot r_0]\!] \qquad \text{and} \qquad va \in [\![r_1 \cdot r_0]\!] \subseteq [\![r_1 \cdot t^* \cdot r_0]\!].$$
Hence $(ua, va) \in [\![h(t^\omega)]\!]_\circ$.

3. If $v_1 = \varepsilon$ and $k > 2$, then $v_2 = av_2'$ as $\varepsilon \notin [\![t]\!]$. This gives rise to a split $(r_0, r_1) \in \nabla_t$ with $a \in [\![r_0]\!]$ and $v_2' \in [\![r_1]\!]$. As in the previous case, we have $t_0 \leq t$ as $v_1 = \varepsilon$. This gives us that

$$\begin{aligned} ua &\in [\![t^* \cdot t_0 \cdot r_0]\!] \subseteq [\![t^* \cdot t \cdot r_0]\!] \subseteq [\![t^* \cdot r_0]\!], \\ va &\in [\![r_1 \cdot t^* \cdot t_0 \cdot r_0]\!] \subseteq [\![r_1 \cdot t^* \cdot t \cdot r_0]\!] \subseteq [\![r_1 \cdot t^* \cdot r_0]\!]. \end{aligned}$$

So $(ua, va) \in [\![h(t^\omega)]\!]_\circ$.

For γ_2-expansion, let $(u, v^k) \to_{\gamma_2} (u, v)$ $(k \geq 1)$ and $(u, v) \in [\![h(t^\omega)]\!]_\circ$. So there exists a split $(t_0, t_1) \in \nabla_t$ with $u \in [\![t^* \cdot t_0]\!]$ and $v \in [\![t_1 \cdot t^* \cdot t_0]\!]$. It suffices to show that $v^k \in [\![t_1 \cdot t^* \cdot t_0]\!]$. We have

$$v^k \in [\![t_1 \cdot t^* \cdot t_0]\!]^k = [\![(t_1 \cdot t^* \cdot t_0)^k]\!] \subseteq [\![t_1 \cdot t^* \cdot t_0]\!],$$

where the last step holds because

$$(t_1 \cdot t^* \cdot t_0) \cdot (t_1 \cdot t^* \cdot t_0) = t_1 \cdot t^* \cdot (t_0 \cdot t_1) \cdot t^* \cdot t_0 \leq t_1 \cdot t^* \cdot t \cdot t^* \cdot t_1 \leq t_1 \cdot t^* \cdot t_0,$$

as $t_0 \cdot t_1 \leq t$. This concludes the base cases.

For the first induction step, let $T = T_1 + T_2$. Let $(u, v) \to_\gamma (u', v')$ and $(u', v') \in [\![h(T_1 + T_2)]\!]_\circ$. As $[\![h(T_1 + T_2)]\!]_\circ = [\![h(T_1)]\!] \cup [\![h(T_2)]\!]_\circ$, $(u', v') \in [\![h(T_i)]\!]_\circ$ for $i \in \{1, 2\}$. By the induction hypothesis, $(u, v) \in [\![h(T_i)]\!]_\circ$ and so $(u, v) \in [\![h(T_1 + T_2)]\!]_\circ$.

For the remaining induction step, let $T = t \cdot T'$. We treat the case of γ_1- and γ_2-expansion separately. For γ_1-expansion, let $(ua, va) \to_{\gamma_1} (u, av)$ and $(u, av) \in [\![h(t \cdot T')]\!]_\circ$. As $[\![h(t \cdot T')]\!]_\circ = [\![t]\!] \cdot [\![h(T')]\!]_\circ$ there exist $u_0, u_1 \in \Sigma^*$ such that $u_0 u_1 = u$, $u_0 \in [\![t]\!]$ and $(u_1, av) \in [\![h(T')]\!]_\circ$. By the induction hypothesis, $(u_1 a, va) \in [\![h(T')]\!]_\circ$, and

$$(ua, va) = (u_0 u_1 a, va) \in [\![t]\!] \cdot [\![h(T')]\!]_\circ = [\![h(t \cdot T')]\!]_\circ.$$

Finally, for γ_2-expansion, let $(u, v^k) \to_{\gamma_2} (u, v)$ $(k \geq 1)$ and $(u, v) \in [\![h(t \cdot T')]\!]_\circ$. So there exist u_0, u_1 with $u_0 \in [\![t]\!]$, $(u_1, v) \in [\![h(T')]\!]_\circ$ and $u_0 u_1 = u$. By the induction hypothesis, $(u_1, v^k) \in [\![h(T')]\!]_\circ$ and so $(u, v^k) = (u_0 u_1, v^k) \in [\![h(t \cdot T')]\!]_\circ$. □

The next definition incorporates the remaining ingredients. We now apply the operations $\cap$ and $\sqrt{-}$ to get the other direction of both equations.

Definition 6.10. Let $\tau = \sum_{i=1}^n t_i \cdot r_i^\circ \in \mathrm{Exp}_\circ$. We define the map Γ as

$$\Gamma(\tau) = \sum_{i=1}^n \sum_{\substack{(t_0', t_1') \in \nabla_t \\ (s_0', s_1') \in \nabla_s}} t_0' \cdot \left(\sqrt{(t_1' \cap s_1') \cdot s_0'} \right)^\circ .$$

We follow up with a proposition which relates the semantics of τ to that of $\Gamma(\tau)$. From this we can derive a couple of corollaries which prove useful.

Proposition 6.11. *Let $\tau = \Sigma_{i=1}^n t_i \cdot r_i^\circ \in Exp_\circ$. Then*

$$(u, v) \in [\![\Gamma(\tau)]\!]_\circ \iff \exists k_1, k_2 \geq 0, \exists v_1, v_2 \in \Sigma^* : v = v_1 v_2 \text{ and } (uv^{k_1} v_1, v_2 v^{k_2 + k_1} v_1) \in [\![\tau]\!]_\circ.$$

Proof. This is shown by using the various definitions.

$$\begin{aligned}
(u,v) \in [\![\Gamma(\tau)]\!]_\circ \iff\ & \exists i, \exists (t'_0, t'_1) \in \nabla_{t_i}, \exists (s'_0, s'_1) \in \nabla_{s_i} : (u,v) \in \left[\!\!\left[t'_0 \cdot \left(\sqrt{(t'_1 \cap s'_1) \cdot s'_0} \right)^\circ \right]\!\!\right]_\circ \\
\iff\ & \exists i, \exists (t'_0, t'_1) \in \nabla_{t_i}, \exists (s'_0, s'_1) \in \nabla_{s_i}, \exists k \geq 1 : u \in [\![t'_0]\!], v^k \in [\![(t'_1 \cap s'_1) \cdot s'_0]\!] \\
\iff\ & \exists i, \exists (t'_0, t'_1) \in \nabla_{t_i}, \exists (s'_0, s'_1) \in \nabla_{s_i}, \exists k_1, k_2 \geq 0, \exists v_1, v_2 \in \Sigma^* : \\
& \quad v_1 v_2 = v, u \in [\![t'_0]\!], v^{k_1} v_1 \in [\![t'_1 \cap s'_1]\!], v_2 v^{k_2} \in [\![s'_0]\!] \\
\iff\ & \exists i, \exists k_1, k_2 \geq 0, \exists v_1, v_2 \in \Sigma^* : v_1 v_2 = v, u v^{k_1} v_1 \in [\![t_i]\!], v_2 v^{k_2 + k_1} v_1 \in [\![s_i]\!] \\
\iff\ & \exists k_1, k_2 \geq 0, \exists v_1, v_2 \in \Sigma^* : v_1 v_2 = v, (u v^{k_1} v_1, v_2 v^{k_2+k_1} v_1) \in [\![\tau]\!]_\circ. \qquad \square
\end{aligned}$$

Corollary 6.12. *Let $\tau \in Exp_\circ$. Then*

1. $\forall (u,v) \in [\![\Gamma(\tau)]\!]_\circ, \exists (u',v') \in [\![\tau]\!]_\circ \colon (u',v') \to_\gamma^* (u,v)$,
2. $\{uv^\omega \mid (u,v) \in [\![\tau]\!]_\circ\} = \{uv^\omega \mid (u,v) \in [\![\Gamma(\tau)]\!]_\circ\}$ *and*
3. $[\![\tau]\!]_\circ \subseteq [\![\Gamma(\tau)]\!]_\circ$.

Proof. Note that in Proposition 6.11 we have $(uv^{k_1}v_1, v_2 v^{k_2+k_1} v_1) \to_\gamma^* (u,v)$, hence item 1. follows. Moreover, this shows that for each lasso in $[\![\tau]\!]_\circ$ there exists a γ-equivalent one in $[\![\Gamma(\tau)]\!]_\circ$ and vice-versa, hence taking direct images via f on both sides yields the same set, giving us item 2. Finally, for $(u,v) \in [\![\tau]\!]_\circ$. Choose $k_1 = k_2 = 0$, $v_1 = \varepsilon$ and $v_2 = v$, so $(u,v) = (uv^{k_1}v_1, v_2 v^{k_2+k_1} v_1) \in [\![\tau]\!]_\circ$. By Proposition 6.11, $(u,v) = (u, v_1 v_2) \in [\![\Gamma(\tau)]\!]_\circ$ establishing item 3. $\square$

With the use of the previous corollaries, we show that if the language semantics of τ is closed under γ-expansion, then $[\![\Gamma(\tau)]\!]_\circ$ is $\sim_\gamma$-saturated. The requirement that $[\![\tau]\!]_\circ$ be closed under γ-expansion is not redundant. While property 1 does not have any restrictions on the length of words we can shift from the prefix underneath the $(-)^\omega$, $\cap$ does as it relies on the expression under the $(-)^\circ$.

Lemma 6.13. *Let $\tau \in Exp_\circ$. If $[\![\tau]\!]_\circ$ is closed under γ-expansion, then $[\![\Gamma(\tau)]\!]_\circ$ is $\sim_\gamma$-saturated.*

Lemma 6.13. First we remark that for $[\![\rho]\!]_\circ$ to be $\sim_\gamma$-saturated it is sufficient that it is closed under γ-reduction and γ-expansion. To see this, assume that it is closed under both, let $(u,v) \sim_\gamma (u',v')$ and assume $(u,v) \in [\![\rho]\!]_\circ$. We have to show that $(u',v') \in [\![\rho]\!]_\circ$. As $\to_\gamma$ is confluent and strongly normalising, both (u,v) and (u',v') reduce in 0 or multiple steps to some normal form (u'',v''). In particular, (u,v) reduces in 0 or multiple steps to (u'',v'') and (u'',v'') expands in 0 or multiple steps to (u',v'). As $[\![\rho]\!]_\circ$ is closed under both γ-reduction and γ-expansion it follows that (u'',v'') and (u',v') both belong to $[\![\rho]\!]_\circ$ as required. Hence, in order to show this lemma, it is enough to show that $[\![\Gamma(\tau)]\!]_\circ$ is closed under γ-reduction and γ-expansion.

We begin by showing that $[\![\Gamma(\tau)]\!]_\circ$ is closed under γ-reduction. We split the proof into two cases, one for each type of reduction. For γ_1-reduction we look at $(ua, va) \to_{\gamma_1} (u, av)$. Suppose $(ua, va) \in [\![\Gamma(\tau)]\!]_\circ$, we want to show that $(u, av) \in [\![\Gamma(\tau)]\!]_\circ$. By Proposition 6.11, we find k_1, k_2, v_1, v_2 such that

$$va = v_1 v_2 \qquad \text{and} \qquad (ua(va)^{k_1} v_1, v_2 (va)^{k_2+k_1} v_1) \in [\![\tau]\!]_\circ.$$

We proceed by case analysis on v_2. If $v_2 = \varepsilon$, then $v_1 = va$ and

$$(ua(va)^{k_1} va, (va)^{k_2+k_1} va) \in [\![\tau]\!]_\circ.$$

As $[\![\tau]\!]_\circ$ is closed under γ-expansion, we also have that

$$(u(av)^{k_1+1} a, v(av)^{(2\cdot k_2+k_1)+(k_1+1)} a) = (ua(va)^{k_1} va, ((va)^{k_2+k_1} va)^2) \in [\![\tau]\!]_\circ.$$

Using Proposition 6.11, we obtain $(u, av) \in [\![\Gamma(\tau)]\!]_\circ$. If $v_2 \neq \varepsilon$, we find v_2' with $v_2 = v_2' a$. Then

$$(u(av)^{k_1} av_1, v_2'(av)^{k_2+k_1} av_1) = (ua(va)^{k_1} v_1, v_2(va)^{k_2+k_1} v_1) \in [\![\tau]\!]_\circ.$$

Hence, by Proposition 6.11 and as $v_1 v_2' = v$ we again have $(u, av) \in [\![\Gamma(\tau)]\!]_\circ$.

For γ_2-reduction we look at $(u, v^k) \to_{\gamma_2} (u, v)$ where $k \geq 1$. Assume that $(u, v^k) \in [\![\Gamma(\tau)]\!]_\circ$. So by Proposition 6.11 there are k_1, k_2, v_1, v_2 with

$$v^k = v_1 v_2 \qquad \text{and} \qquad (u(v^k)^{k_1} v_1, v_2(v^k)^{k_2+k_1} v_1) \in [\![\tau]\!]_\circ.$$

As $v_1 v_2 = v^k$ we can find some w_1, w_2 and ℓ_1, ℓ_2 with $v^{\ell_1} w_1 = v_1$ and $w_2 v^{\ell_2} = v_2$. It follows that

$$(uv^{k \cdot k_1 + \ell_1} w_1, w_2 v^{\ell_2 + k \cdot (k_2+k_1) + \ell_1} w_1) = (u(v^k)^{k_1} v^{\ell_1} w_1, w_2 v^{\ell_2} (v^k)^{k_2+k_1} v^{\ell_1} w_1) \in [\![\tau]\!]_\circ.$$

Hence, by Proposition 6.11, $(u, v) = (u, w_1 w_2) \in [\![\Gamma(\tau)]\!]_\circ$. So $[\![\Gamma(\tau)]\!]_\circ$ is closed under γ-reduction.

Next we show that $[\![\Gamma(\tau)]\!]_\circ$ is closed under γ-expansion. Let $(u, v) \in [\![\Gamma(\tau)]\!]_\circ$ and $(u_1, v_1) \to_\gamma (u, v)$. We want to show that $(u_1, v_1) \in [\![\Gamma(\tau)]\!]_\circ$. By Corollary 3.1 there exists some $(u_2, v_2) \in [\![\tau]\!]_\circ$ with $(u_2, v_2) \to_\gamma^* (u, v)$. As $(u_1, v_1) \sim_\gamma (u_2, v_2)$, there exists a lasso (u_3, v_3) such that $(u_3, v_3) \to_\gamma^* (u_1, v_1)$ and $(u_3, v_3) \to_\gamma^* (u_2, v_2)$. As $[\![\tau]\!]_\circ$ is closed under γ-expansion, $(u_3, v_3) \in [\![\tau]\!]_\circ$. It follows by Corollary 3.3 that $(u_3, v_3) \in [\![\Gamma(\tau)]\!]_\circ$. Finally, as we have already shown that $[\![\Gamma(\tau)]\!]_\circ$ is closed under γ-reduction and as $(u_3, v_3) \to_\gamma^* (u_1, v_1)$, we obtain that $(u_1, v_1) \in [\![\Gamma(\tau)]\!]_\circ$. □

Proposition 6.14. *Let $T \in Exp_\omega$. Then $\Gamma(h(T))$ represents T.*

Proof. By Proposition 6.8, $h(T)$ weakly represents T, i.e. $\mathrm{UP}([\![T]\!]_\omega) = \{uv^\omega \mid (u, v) \in [\![h(T)]\!]_\circ\}$. It follows from Corollary 3.2 that $\Gamma(h(T))$ also weakly represents T, as

$$\mathrm{UP}([\![T]\!]_\omega) = \{uv^\omega \mid (u, v) \in [\![h(T)]\!]_\circ\} = \{uv^\omega \mid (u, v) \in [\![\Gamma(h(T))]\!]_\circ\}.$$

Furthermore, as $[\![h(T)]\!]_\circ$ is closed under γ-expansion (Proposition 6.9), $[\![\Gamma(h(T))]\!]_\circ$ is $\sim_\gamma$-saturated (Proposition 6.13) Hence, by Remark 6.3 $\Gamma(h(T))$ represents T. □

The last proposition together with the Brzozowski construction for lasso automata give us the main result of this section.

Theorem 6.15. *Every rational ω-language is accepted by a finite Ω-automaton.*

Proof. Let L be a rational ω-language and $T \in \mathrm{Exp}_\omega$ a corresponding rational ω-expression. Then $(\widehat{\mathcal{C}}, [\Gamma(h(T))]_{\sim_C})$ is a finite Ω-automaton with

$$L_\omega(\widehat{\mathcal{C}}, [\Gamma(h(T))]_{\sim_C}) = [\![T]\!]_\omega = L.$$ □

7 Conclusion

We have introduced rational lasso expressions and languages and shown a Kleene Theorem for lasso languages and ω-languages. In order to obtain these results we gave a Brzozowski construction for lasso automata. Moreover, we introduced the notion of representation and showed how to construct a representing rational lasso expression from a rational ω-expression. As a consequence, we obtained a construction method for converting rational ω-expressions to Ω-automata.

Our results present some interesting directions for future work. In [2], they introduce syntactic and recurring FDFAs and show that they can be up to exponentially smaller than periodic FDFAs. This raises the question whether from a given rational ω-expression, we can construct a rational lasso expression such that the Brzozowski construction yields a syntactic or recurring FDFA. Another line of work is the exploration of our results in a categorical setting. Finally, there is the question of complexity of our constructions, and investigating applications of our Brzozowski construction for Ω-automata.

Acknowledgements. The author would like to thank Harsh Beohar, Tobias Kappé, Georg Struth and Yde Venema.

References

[1] Dana Angluin, Udi Boker, and Dana Fisman. Families of DFAs as acceptors of omega-regular languages. In Piotr Faliszewski, Anca Muscholl, and Rolf Niedermeier, editors, *41st International Symposium on Mathematical Foundations of Computer Science, MFCS 2016, August 22-26, 2016 - Kraków, Poland*, volume 58 of *LIPIcs*, pages 11:1–11:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2016.

[2] Dana Angluin and Dana Fisman. Learning regular omega languages. *Theor. Comput. Sci.*, 650:57–72, 2016.

[3] Janusz A. Brzozowski. Derivatives of regular expressions. *J. ACM*, 11(4):481–494, 1964.

[4] Hugues Calbrix, Maurice Nivat, and Andreas Podelski. Ultimately periodic words of rational ω-languages. In Stephen D. Brookes, Michael G. Main, Austin Melton, Michael W. Mislove, and David A. Schmidt, editors, *Mathematical Foundations of Programming Semantics, 9th International Conference, New Orleans, LA, USA, April 7-10, 1993, Proceedings*, volume 802 of *Lecture Notes in Computer Science*, pages 554–566. Springer, 1993.

[5] Vincenzo Ciancia and Yde Venema. Omega-automata: A coalgebraic perspective on regular omega-languages. In Markus Roggenbach and Ana Sokolova, editors, *8th Conference on Algebra and Coalgebra in Computer Science (CALCO)*, volume 139 of *LIPIcs*, pages 5:1–5:18. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019.

[6] Mike Cruchten. Topics in Ω-automata – A journey through lassos, algebra, coalgebra and expressions. Master's thesis, The University of Amsterdam, June 2022.

[7] Nate Foster, Dexter Kozen, Matthew Milano, Alexandra Silva, and Laure Thompson. A coalgebraic decision procedure for NetKAT. In Sriram K. Rajamani and David Walker, editors, *Proceedings of the 42nd Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, pages 343–355. ACM, 2015.

[8] Tobias Kappé, Paul Brunet, Alexandra Silva, and Fabio Zanasi. Concurrent Kleene algebra: Free model and completeness. In Amal Ahmed, editor, *Proceedtings of the 27th European Symposium on Programming, ESOP 2018*, volume 10801 of *Lecture Notes in Computer Science*, pages 856–882. Springer, 2018.

[9] Dexter Kozen. A Completeness Theorem for Kleene Algebras and the Algebra of Regular Events. *Inf. Comput.*, 110(2):366–390, 1994.

[10] Bryan Krawetz, John Lawrence, and Jeffrey O. Shallit. State complexity and the monoid of transformations of a finite set. *Int. J. Found. Comput. Sci.*, 16(3):547–563, 2005.

[11] Oded Maler and Ludwig Staiger. On syntactic congruences for omega-languages. *Theor. Comput. Sci.*, 183(1):93–112, 1997.

[12] Klaus W. Wagner. Eine Axiomatisierung der Theorie der regulären Folgenmengen. *J. Inf. Process. Cybern.*, 12(7):337–354, 1976.

A The Brzozowski Construction for Deterministic Finite Automata

The results in this section are well-known ([3]).

Definition A.1. The *Brzozowski derivative* $d : \mathrm{Exp} \to \mathrm{Exp}^{\Sigma}$ is defined recursively on rational expressions.

$$\begin{array}{lll} d(0,a) = 0 & d(t+r,a) = d(t,a) + d(r,a) & d(b,a) = [b = a] \\ d(1,a) = 0 & d(t \cdot r,a) = d(t,a) \cdot r + [t \in N] \cdot d(r,a) & d(t^*,a) = d(t,a) \cdot t^* \end{array}$$

Proposition A.2. *The Brzozowski derivative preserves provable equality of terms, i.e. for rational expressions t, r with $\vdash t = r$ and for $a \in \Sigma$, we have $\vdash d(t,a) = d(r,a)$.*

Proposition A.3 (Fundamental Theorem). *If $t \in Exp$. Then*

$$\vdash t = [t \in N] + \sum_{a \in \Sigma} a \cdot d(t,a).$$

Proposition A.4. *The deterministic automaton $\mathcal{B} = (Exp, d, N)$ is called the* Brzozowski automaton *and for all $t \in Exp$ we have $L(\mathcal{B}, t) = [\![t]\!]$.*

Definition A.5. Let $\sim_B \subseteq \mathrm{Exp}^2$ be the least equivalence relation satisfying:

$$1 \cdot t \sim_B t \qquad 0 \cdot t \sim_B 0 \qquad t \sim_B t + t \qquad t + r \sim_B r + t \qquad (t + r) + g \sim_B t + (r + g).$$

We write $\sim$ whenever this does not lead to confusion.

Lemma A.6. *The equivalence relation $\sim$ is compatible with the Brzozowski derivative d and with the predicate N, i.e. if $t, r \in Exp$ and $t \sim r$, then*

1. *$t \in N \iff r \in N$ and*
2. *$\forall a \in \Sigma : d(t,a) \sim d(r,a)$.*

Corollary A.7. *The map $\widehat{d} : Exp/_{\sim} \to (Exp/_{\sim})^{\Sigma}$ given by $\widehat{d}([t]_{\sim}, a) = [d(t,a)]_{\sim}$ and the predicate $\widehat{N} \subseteq (Exp/_{\sim})^2$ defined by $[t]_{\sim} \in \widehat{N} \iff t \in N$ are both well-defined.*

Theorem A.8. *$\widehat{\mathcal{B}} = (Exp/_{\sim}, \widehat{d}, \widehat{N})$ is a deterministic automaton satisfying for all $t \in Exp$*

1. *$L(\widehat{\mathcal{B}}, [t]_{\sim}) = [\![t]\!]$ and*
2. *the set of states reachable from $[t]_{\sim}$ is finite.*