On the Positive Calculus of Relations with Transitive Closure

Damien Pous
Univ. Lyon, CNRS, ENS de Lyon, UCB Lyon 1, LIP
Damien.Pous@ens-lyon.fr

Abstract

Binary relations are such a basic object that they appear in many places in mathematics and computer science. For instance, when dealing with graphs, program semantics, or termination guarantees, binary relations are always used at some point.

In this survey, we focus on the relations themselves, and we consider algebraic and algorithmic questions. On the algebraic side, we want to understand and characterise the laws governing the behaviour of the following standard operations on relations: union, intersection, composition, converse, and reflexive-transitive closure. On the algorithmic side, we look for decision procedures for equality or inequality of relations.

After having formally defined the calculus of relations, we recall the existing results about two well-studied fragments of particular importance: Kleene algebras and allegories. Unifying those fragments yields a decidable theory whose axiomatisability remains an open problem.

2012 ACM Subject Classification Theory of computation → Formal languages and automata theory, Theory of computation → Equational logic and rewriting

Keywords and phrases Relation Algebra, Kleene Algebra, Allegories, Automata, Graphs

Digital Object Identifier 10.4230/LIPIcs.STACS.2018.3

Category Invited Talk

Funding This work has been supported by the European Research Council (ERC) under the European Union's Horizon 2020 programme (CoVeCe, grant agreement No 678157), and by the LABEX MILYON (ANR-10-LABX-0070) of Université de Lyon, within the program "Investissements d'Avenir" (ANR-11-IDEX-0007) operated by the French National Research Agency (ANR).

Acknowledgements I would like to thank Paul Brunet, with whom we obtained the few new results presented in this survey, as well as Arnaud Durand for his feedback on a preliminary version of this survey, which appeared in French as course notes for a research school (EJCIM'16) [20].

1 The calculus of relations

Given a set P, a *relation* on P is a set of pairs of elements from P. For instance, the usual order on natural numbers is a relation. In the sequel, relations are ranged over using letters R, S, their set is written $\mathcal{P}(P\times P)$, and we write $p\,R\,q$ for $\langle p,q\rangle \in R$.

The set of relations is equiped with a partial order, set-theoretic inclusion ($\subseteq$), and three binary operations: set-theoretic union, written $R + S$, set-theoretic intersection, written $R \cap S$, and relational composition:

$$R{\cdot}S \triangleq \{\langle p,q\rangle \mid \exists r \in P,\ p\,R\,r\ \wedge\ r\,S\,q\}\ .$$

35th Symposium on Theoretical Aspects of Computer Science (STACS 2018).
Editors: Rolf Niedermeier and Brigitte Vallée; Article No. 3; pp. 3:1–3:16
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

It also contains three specific relations: the empty relation, written 0, the full relation, written $\top$, and the identity relation:

$$1 \triangleq \{\langle p, p\rangle \mid p \in P\} \ .$$

Lastly, one can consider three unary operations: set-theoretic complement, written R^c, converse (or transpose), R°, and reflexive-transitive closure, $R^\star$, defined as follows:

$$R^c \triangleq \{\langle p, q\rangle \mid \neg p \, R \, q\} \ ,$$
$$R^\circ \triangleq \{\langle p, q\rangle \mid q \, R \, p\} \ ,$$
$$R^\star \triangleq \{\langle p, q\rangle \mid \exists p_0, \dots, p_n,\ p_0 = p \ \wedge\ p_n = q \ \wedge\ \forall i < n,\ p_i \, R \, p_{i+1}\} \ .$$

We restrict ourselves to this list of operations here, even though it is not exhaustive. These operations make it possibles to state many properties in a concise way, without mentioning the points related by the relations. Here are a few examples:

$1 \subseteq R$	R is reflexive: $\forall p \in P, p \, R \, p$
$R{\cdot}R \subseteq R$	R is transitive: $\forall pqr,\ p \, R \, r \wedge r \, R \, q \Rightarrow p \, R \, q$
$R{\cdot}R^\star \cap 1 = 0$	R is acyclic: $\forall p_0 \dots p_n,\ n{>}0,\ (\forall i,\ p_i \, R \, p_{i+1}) \Rightarrow p_0 \neq p_n$
$R^\circ{\cdot}S \subseteq S{\cdot}R^\circ$	R and S commute: $\forall pqr,\ r \, R \, p \wedge r \, S \, q \Rightarrow \exists t,\ q \, R \, t \wedge p \, S \, t$

▶ **Exercise 1.** *To which standard notions from rewriting theory correspond the inequations* $R^\circ{\cdot}R \subseteq R^\star{\cdot}R^{\circ\star}$ *and* $R^{\circ\star}{\cdot}R^\star \subseteq R^\star{\cdot}R^{\circ\star}$ *?*

Moreover, these operations satisfy many laws. Some of these laws are extremely simple (for instance, composition is associative, $(R \cdot R') \cdot R'' = R \cdot (R' \cdot R'')$; the empty relation is absorbs composition, $R{\cdot}0 = 0 = 0{\cdot}R$; reflexive-transitive closures are transitive, $R^\star{\cdot}R^\star \subseteq R^\star$). Others are much more complicated and counter-intuitive.

▶ **Exercise 2.** *Amongst the following equations and inequations, which ones are universally true? In each case, give a counter-example or a detailed proof.*

$$1 \cap R \subseteq R{\cdot}R \cap R{\cdot}R{\cdot}R \tag{1}$$
$$(R+S)^\star = R^\star{\cdot}(S{\cdot}R^\star)^\star \tag{2}$$
$$(R+S)^\star = ((1+R){\cdot}S)^\star \tag{3}$$
$$R{\cdot}(S \cap T) = R{\cdot}S \cap R{\cdot}T \tag{4}$$
$$R{\cdot}S \cap T \subseteq R{\cdot}(S \cap R^\circ{\cdot}T) \tag{5}$$
$$R{\cdot}S \cap T \subseteq (R \cap T{\cdot}S^\circ){\cdot}(S \cap R^\circ{\cdot}T) \tag{6}$$
$$(R \cap S{\cdot}\top){\cdot}T = R{\cdot}T \cap S{\cdot}\top \tag{7}$$

Two questions arise naturally:

1. is it possible to axiomatise the set of laws that are universally true, that is, to give a small number of elementary laws from which all valid laws follow?
2. is it possible to decide whether a law is valid or not?

When considering all the operations listed above, the answer is negative in both cases. Indeed, Monk proved that there cannot be a finite axiomatisation [16], and Tarski proved that the theory is actually undecidable [25, 24]. In both cases, reflexive-transitive closure is not necessary but the complement plays a crucial role. Thus we focus in the sequel on the *positive* fragments, where complement is excluded.

Now we setup the concepts and notation needed in the sequel.

Let Σ be a set, whose elements are denoted by letters a, b. *(Relational) expressions* are defined by the following grammar:

$$e, f, g ::= e + f \mid e \cap f \mid e \cdot f \mid e^\circ \mid e^\star \mid 0 \mid 1 \mid \top \mid a \qquad (a \in \Sigma) .$$

Given a set E and a function $\sigma : \Sigma \to \mathcal{P}(E{\times}E)$ mapping any letter from Σ to a relation on E, we define inductively the extension $\hat{\sigma}$ of σ to expressions:

$$\begin{aligned} \hat{\sigma}(e+f) &\triangleq \hat{\sigma}(e) + \hat{\sigma}(f) & \hat{\sigma}(e^\circ) &\triangleq \hat{\sigma}(e)^\circ & \hat{\sigma}(1) &\triangleq 1 \\ \hat{\sigma}(e \cap f) &\triangleq \hat{\sigma}(e) \cap \hat{\sigma}(f) & \hat{\sigma}(e^\star) &\triangleq \hat{\sigma}(e)^\star & \hat{\sigma}(\top) &\triangleq \top \\ \hat{\sigma}(e \cdot f) &\triangleq \hat{\sigma}(e) \cdot \hat{\sigma}(f) & \hat{\sigma}(0) &\triangleq 0 & \hat{\sigma}(a) &\triangleq \sigma(a) \end{aligned}$$

Given two expressions e and f, an equation is *valid*, written $\vDash e = f$, if for all set E and for all function $\sigma : \Sigma \to \mathcal{P}(E{\times}E)$, we have $\hat{\sigma}(e) = \hat{\sigma}(f)$. Intuitively, an equation is valid if it is universally true in relations, if it holds whatever the relations we use to interpret its variables.

Similarly, an inequation is valid, written $\vDash e \subseteq f$, if $\hat{\sigma}(e) \subseteq \hat{\sigma}(f)$ for all set E and function $\sigma : \Sigma \to \mathcal{P}(E{\times}E)$. Characterising valid equations is equivalent to characterising valid inequations, as shown in the following exercise.

▶ **Exercise 3.** *Let e, f be two expressions. We have $\vDash e = f$ iff $\vDash e \subseteq f$ and $\vDash f \subseteq e$. Show that $\vDash e \subseteq f$ iff $\vDash e + f = f$ iff $\vDash e \cap f = e$.*

2 The ideal fragment: Kleene algebra

In this section we remove from the syntax the operations of intersection and converse, as well as the constant $\top$. In other words, we restrict to regular expressions:

$$e, f, g ::= e + f \mid e \cdot f \mid e^\star \mid 0 \mid 1 \mid a \qquad (a \in \Sigma) .$$

we shall see that with such a restriction, the validity of an equation is decidable, and more precisely, PSPACE-complete.

2.1 Decidability

Let letter u, v range over finite words over the alphabet Σ, let ϵ denote the empty word, and uv the concatenation of two words u and v. A *language* is a set of words. We define inductively a function $[\cdot]$ associating a language to each expression:

$$\begin{aligned} [e+f] &\triangleq [e] \cup [f] & [0] &\triangleq \emptyset \\ [e \cdot f] &\triangleq \{uv \mid u \in [e], v \in [f]\} & [1] &\triangleq \{\epsilon\} \\ [e^\star] &\triangleq \{u_1 \dots u_n \mid \forall i, u_i \in [e]\} & [a] &\triangleq \{a\} \end{aligned}$$

The key result about this fragment of the calculus of relations is the following characterisation: an equation is valid for relations if and only if it corresponds to an equality of languages.

▶ **Theorem 4.** *For all regular expressions e, f, we have*

$$\vDash e = f \qquad \textit{iff} \qquad [e] = [f] .$$

We prove this theorem below. Its main consequence in practice is the decidability of the validity of equations: $[e]$ and $[f]$ are regular languages which we can easily represent using finite automata in order to compare them. This characterisation also gives us the precise complexity of the problem, as language equivalence of regular expression is PSPACE-complete [23].

Proof of Theorem 4. First we show the implication from left to right. Suppose $\vDash e = f$, we have to find a function σ from the alphabet Σ to a space of relations $\mathcal{P}(E\times E)$, such that $\hat{\sigma}(e) = \hat{\sigma}(f)$ entails $[e] = [f]$. Take $E = \Sigma^\star$, the set of words over Σ, and define σ as follows:

$$\begin{aligned} \sigma : \Sigma &\to \mathcal{P}(\Sigma^\star\times\Sigma^\star) \\ a &\mapsto \{\langle u, ua\rangle \mid u \in \Sigma^\star\} \end{aligned}$$

We will show that for all expression g, we have

$$\hat{\sigma}(g) = \{\langle u, uv\rangle \mid u \in \Sigma^\star, v \in [g]\} \ .$$

In particular, we will thus have $v \in [g]$ if and only if $\langle \epsilon, v\rangle \in \hat{\sigma}(g)$, so that $\hat{\sigma}(e) = \hat{\sigma}(f)$ entails $[e] = [f]$.
We proceed by induction on the expression g:

- $g = g' + g''$: we have

$$\begin{aligned} \hat{\sigma}(g) &= \hat{\sigma}(g') \cup \hat{\sigma}(g'') \\ &= \{\langle u, uv\rangle \mid u \in \Sigma^\star, v \in [g']\} \cup \{\langle u, uv\rangle \mid u \in \Sigma^\star, v \in [g'']\} && \text{(by induction)} \\ &= \{\langle u, uv\rangle \mid u \in \Sigma^\star, v \in [g'] \cup [g'']\} \\ &= \{\langle u, uv\rangle \mid u \in \Sigma^\star, v \in [g' + g'']\} \end{aligned}$$

- $g = g' \cdot g''$: we have

$$\begin{aligned} \hat{\sigma}(g) &= \hat{\sigma}(g') \cdot \hat{\sigma}(g'') \\ &= \{\langle u, uv\rangle \mid u \in \Sigma^\star, v \in [g']\} \cdot \{\langle u', u'w\rangle \mid u' \in \Sigma^\star, w \in [g'']\} && \text{(by induction)} \\ &= \{\langle u, uvw\rangle \mid u \in \Sigma^\star, v \in [g'], w \in [g'']\} \\ &= \{\langle u, uv\rangle \mid u \in \Sigma^\star, v \in [g' \cdot g'']\} \end{aligned}$$

- $g = g'^\star$: like in the previous point, we have

$$\begin{aligned} \hat{\sigma}(g) &= \hat{\sigma}(g')^\star \\ &= \{\langle u, uv\rangle \mid u \in \Sigma^\star, v \in [g']\}^\star && \text{(by induction)} \\ &= \{\langle u, uv\rangle \mid u \in \Sigma^\star, v \in [g'^\star]\} \end{aligned}$$

  (for the last step, we first show the following property, by induction on $k \in \mathbb{N}$: for all language $L \subseteq \Sigma^\star$, we have $\{\langle u, uv\rangle \mid u \in \Sigma^\star, v \in L\}^k = \{\langle u, uv\rangle \mid u \in \Sigma^\star, v \in L^k\}$).
- $g = 0$, $g = 1$, $g = a$: by unfolding definitions.

Now consider the converse implication. Fix a set E and a function $\sigma : \Sigma \to \mathcal{P}(E\times E)$; we have to show that $[e] = [f]$ entails $\hat{\sigma}(e) = \hat{\sigma}(f)$. This implication follows immediately from the following property, which we prove by induction on the expression g:

$$\hat{\sigma}(g) = \bigcup_{v\in[g]} \hat{\sigma}(v) \ .$$

(Note the slight abuse of notation in the term of the union: we apply the function $\hat{\sigma}$, expecting a regular expression, to a word v; we implicitely use the natural injection from words to expressions.)

- $g = g' + g''$: we have

$$\begin{aligned}
\hat\sigma(g) &= \hat\sigma(g') \cup \hat\sigma(g'') \\
&= \bigcup_{v\in[g']} \hat\sigma(v) \ \cup \bigcup_{v\in[g'']} \hat\sigma(v) && \text{(by induction)} \\
&= \bigcup_{v\in[g']\cup[g'']} \hat\sigma(v) \\
&= \bigcup_{v\in[g'+g'']} \hat\sigma(v)
\end{aligned}$$

- $g = g' \cdot g''$: we have

$$\begin{aligned}
\hat\sigma(g) &= \hat\sigma(g') \cdot \hat\sigma(g'') \\
&= \bigcup_{v\in[g']} \hat\sigma(v) \cdot \bigcup_{w\in[g'']} \hat\sigma(w) && \text{(by induction)} \\
&= \bigcup_{v\in[g'],\ w\in[g'']} \hat\sigma(v)\cdot\hat\sigma(w) && \text{(distributivity)} \\
&= \bigcup_{v\in[g'],\ w\in[g'']} \hat\sigma(v\cdot w) \\
&= \bigcup_{u\in[g'\cdot g'']} \hat\sigma(u)
\end{aligned}$$

- $g = g'^\star$: we have

$$\begin{aligned}
\hat\sigma(g) &= \hat\sigma(g')^\star \\
&= (\bigcup_{v\in[g']} \hat\sigma(v))^\star && \text{(by induction)} \\
&= \bigcup_{v_1\in[g'],\dots,v_n\in[g']} \hat\sigma(v_1)\cdot\dots\cdot\hat\sigma(v_n) \\
&= \bigcup_{v_1\in[g'],\dots,v_n\in[g']} \hat\sigma(v_1\cdot\dots\cdot v_n) \\
&= \bigcup_{u\in[g'^\star]} \hat\sigma(u)
\end{aligned}$$

- $g = 0$, $g = 1$, $g = a$: again, by unfolding definitions. ◀

Note that this proof leads to a similar characterisation for inequations: for all regular expressions e and f,

$$\models e \subseteq f \qquad \text{iff} \qquad [e] \subseteq [f] \ .$$

2.2 Axiomatisation

In 1956, Kleene asks for axiomatisations of the previous theory [7]: is it possible to find a small set of axioms (i.e., equations), from which follow all valid equations between regular expressions?

In the sixties, Salomaa gives two axiomatisations [22] which are not purely algebraic, and Redko proves that no finite equational axiomatisation can be complete [21]. Conway studies extensively this kind of questions in his monograph on regular algebra and finite

$$\left.\begin{aligned} e+(f+g)&=(e+f)+g\\ e+f&=f+e\\ e+0&=e\\ e+e&=e \end{aligned}\right\}\ \begin{array}{l}\langle +,0\rangle \text{ is a commutative}\\ \text{and idempotent monoid}\end{array}$$

$$\left.\begin{aligned} e\cdot(f\cdot g)&=(e\cdot f)\cdot g\\ e\cdot 1&=e\\ 1\cdot e&=e \end{aligned}\right\}\ \langle \cdot,1\rangle \text{ is a monoid}$$

$$\left.\begin{aligned} e\cdot(f+g)&=e\cdot f+e\cdot g\\ (e+f)\cdot g&=e\cdot g+f\cdot g\\ e\cdot 0&=0\\ 0\cdot e&=0 \end{aligned}\right\}\ \begin{array}{l}\text{distributivity between}\\ \text{the two monoids}\end{array}$$

$$\left.\begin{aligned} 1+e\cdot e^\star&=e^\star\\ e\cdot f\leq f\quad&\Rightarrow\quad e^\star\cdot f\leq f\\ f\cdot e\leq f\quad&\Rightarrow\quad f\cdot e^\star\leq f \end{aligned}\right\}\ \text{laws about Kleene star}$$

Figure 1 The axioms of Kleene algebra.

automata [7], but we have to wait for the nineties for new results: Krob and Kozen independently show that one can axiomatise this theory in a finite way, but using axioms that are not just equations, but implications between equations. (We move from varieties to quasi-varieties.)

Krob's proof is long and difficult [15], but it provides a complete picture: first he gives a purely equational axiomatisation, infinite but with more structure than Salomaa's axioms. Then he shows that those infinitely many axioms can be derived from various finite axiomatisations involving implications between equations.

On the contrary, Kozen goes straight to the point and focuses on a specific finite axiomatisation (with implications). His proof is not simple either, but much shorter [13, 14].

▶ **Theorem 5** (Kozen'91, Krob'91)**.** *For all regular expressions e, f, we have $[e] = [f]$ if and only if the equality $e = f$ is derivable from the axioms listed in Figure 1, where notation $e \leq f$ is a shorthand for $e + f = f$.*

These axioms can be decomposed into four groups: the first three correspond to the fact that we have an idempotent non-commutative semiring; the last group of axioms characterises the operation of reflexive-transitive closure, often called "Kleene star" in this context. This group is not entirely symmetric: the law $1 + e^\star \cdot e = e^\star$ is omitted as it can be derived from the other axioms. The last two axioms are implications; intuitively, they tell that if an expression f is invariant under composition with another expression e, then it is also invariant with $e^\star$. The expressive power of the axiomatisation mainly comes from those two implications: they make it possible to reason inductively on Kleene star, in a purely algebraic way.

One easily checks that each of these axioms is valid in the model of binary relations, but also when interpreting the expressions e, f, g as arbitrary languages. The converse implication from Theorem 5 follows from this remark: we prove only valid equations using those axioms.

The difficulty lies in the other implication: the completeness of these axioms, the fact that any valid equation might eventually be deduced from these axioms. We do not detail the proof here; a key step consists in showing that the set of matrices with coefficients in a Kleene algebra forms a new Kleene algebra (a Kleene algebra being a structure satisfying the axioms from Figure 1).

▶ **Exercise 6.** *Prove the following laws by using only Kleene algebras axioms:*

$$g + e{\cdot}f \leq f \Rightarrow e^\star{\cdot}g \leq f$$
$$g + f{\cdot}e \leq f \Rightarrow g{\cdot}e^\star \leq f$$
$$1 + e^\star{\cdot}e = e^\star$$
$$e{\cdot}f \leq g{\cdot}e \Rightarrow e{\cdot}f^\star \leq g^\star{\cdot}e$$
$$e{\cdot}f = g{\cdot}e \Rightarrow e{\cdot}f^\star = g^\star{\cdot}e$$
$$e{\cdot}(f{\cdot}e)^\star = (e{\cdot}f)^\star{\cdot}e$$
$$(e + f)^\star = e^\star{\cdot}(f^\star{\cdot}e)^\star$$

3 The strange fragment: allegories

Now consider a different fragment, where we only have composition, intersection, converse, and constants 1 and $\top$. For reasons to become clear in Section 4, we reuse letters u, v, w to denote the corresponding regular expressions, which we shall call *terms*:

$$u, v, w ::= u \cdot v \mid u \cap v \mid u^\circ \mid 1 \mid \top \mid a \qquad (a \in \Sigma) \ .$$

Modulo the presence of the constant $\top$, this fragment was studied by Andréka and Bredikhin [2], and by Freyd and Scedrov [11] under the name of (representable) *allegories*. We will see that one can decide the validity of inequations (and thus also equations) in this fragment, but that again, the corresponding theory is not finitely axiomatisable in a purely equational way.

3.1 Decidability

The key idea consists in characterising valid inequations by the existence of graph homomorphisms. More precisely, homomorphisms of directed and edge-labeled graphs with two distinguished vertices.

▶ **Definition 7** (Graph). A *graph* is a tuple $\langle V, E, \iota, o\rangle$, where V is a set of *vertices*, $E \subseteq V \times \Sigma \times V$ is a set of labelled edges, and $\iota, o \in V$ are two distinguished vertices, respectively called *input* and *output*.

We let letters G, H range over graphs and we define the following operations:

- $G \cdot H$ is the graph obtained by composing the two graphs in series, that is, by putting them one after the other and by merging the output of G with the input of H;
- $G \cap H$ is the graph obtained by composing the two graphs in parallel, that is, by putting them side by side and by merging their inputs and their outputs;
- G° is the graph obtained from G by exchanging input and output (without reversing edges);
- $\underline{1}$ is the graph without edges and with a single vertex $(\langle\{*\}, \emptyset, *, *\rangle)$;
- $\underline{\top}$ is the graph without edges and with two vertices, where input and output are distinct $(\langle\{\star, \bullet\}, \emptyset, \star, \bullet\rangle)$;

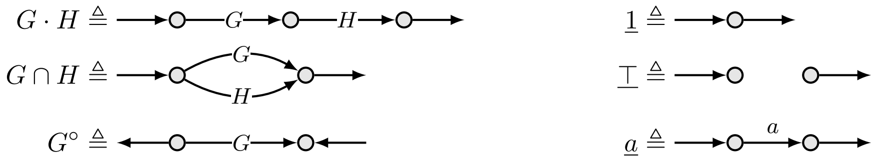


Figure 2 Operations on graphs.

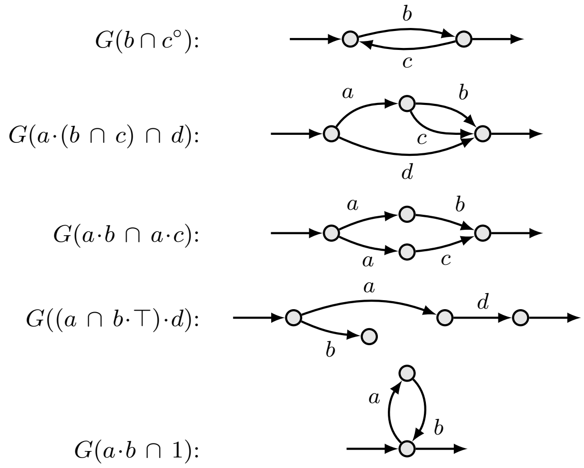


Figure 3 Graphs associated to some terms.

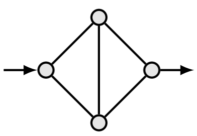
- for $a \in \Sigma$, $\underline{a}$ is the graph with two vertices and an edge labelled a from the input to the output $(\langle\{\star,\bullet\},\{\langle\star,a,\bullet\rangle\},\star,\bullet\rangle)$.

These operations are depicted on Figure 2; the input and the output of each graph is denoted using unlabelled arrows. These operations make it possible to associate a graph $G(u)$ to every term u, by structural induction:

$$G(u\cdot v) \triangleq G(u)\cdot G(v) \qquad G(1) \triangleq \underline{1}$$
$$G(u\cap v) \triangleq G(u)\cap G(v) \qquad G(\top) \triangleq \underline{\top}$$
$$G(u^\circ) \triangleq G(u)^\circ \qquad G(a) \triangleq \underline{a}$$

The graphs of a few terms are drawn on Figure 3. These are series-parallel graphs as long as we do not use converse and identity, that introduce loops in presence of intersection, nor the constant $\top$, that can disconnect some parts of the graphs.

Some graphs are not associated to any term. The canonical counter-example is the following one. (The labelling and the orientation of the five edges is irrelevant so that we omit this information.)



(8)

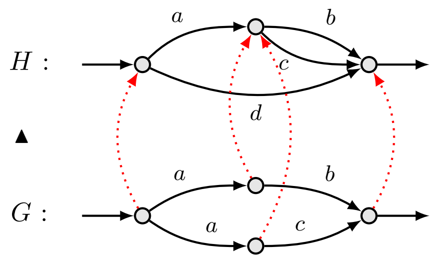


Figure 4 A graph homomorphism.

In fact, the graphs of terms are exactly the graphs of *treewidth* at most two; equivalently, they are the graphs excluding the complete graph with four vertices (K_4) as a minor[1] [9, 8].

One can compare graphs using homomorphisms:

▶ **Definition 8.** A *homomorphism* from the graph G to the graph H is a function from vertices of G to vertices of H that preserves labelled edges, input, and output. We write $H \blacktriangleleft G$ when there exists a homomorphism from G to H.

One easily checks that the relation $\blacktriangleleft$ is a preorder on graphs: it is reflexive and transitive.

As an example, the graph of $a{\cdot}(b \cap c) \cap d$ is smaller than that of $a{\cdot}b \cap a{\cdot}c$, thanks to the homomorphism depicted on Figure 4 using dotted arrows. Note that homomorphisms need not be injective or surjective, so that the preorder is completely unrelated to the sizes of the graphs: a graph may perfectly be smaller than another one, in the sense of the preorder, while having more vertices or edges (and vice-versa).

The nice property of the fragment considered here is the following characterisation: an inequation is valid for relations if and only if there exists a homorphism between the underlying graphs:

▶ **Theorem 9** ([2, Theorem 1], [11, page 208])**.** *For all terms u, v, we have*

$$\vDash u \subseteq v \qquad \textit{iff} \qquad G(u) \blacktriangleleft G(v) \ .$$

Graphs of terms being finite, one can look for a homomorphism between two such graphs in an exhaustive way, whence the decidability of the problem.

▶ **Exercise 10.** *Prove the laws* (1), (5), (6), *and* (7) *from Exercice 2, by using Theorem 9.*

We need a lemma in order to prove the theorem.

▶ **Lemma 11.** *Let u be a term, and let $G(u) = \langle V, E, \iota, o\rangle$ be its graph. Let S be a set and $\sigma : \Sigma \to \mathcal{P}(S{\times}S)$ an interpretation function. For all elements $i, j \in S$, we have $\langle i, j\rangle \in \hat\sigma(u)$ iff there exists a function $\phi : V \to S$ such that:*

$$\begin{cases} \phi(\iota) = i\,, \\ \phi(o) = j\,, \ \ et \\ \langle p, a, q\rangle \in E \ \Rightarrow \ \langle \phi(p), \phi(q)\rangle \in \sigma(a) \ . \end{cases}$$

Proof. We proceed by induction on u:

[1] In both cases, after adding a edge between the input and the output.

- $u = v \cdot w$: write $G(v) = \langle V_v, E_v, \iota_v, o_v\rangle$ and $G(w) = \langle V_w, E_w, \iota_w, o_w\rangle$. We have $\langle i, j\rangle \in \hat{\sigma}(u) = \hat{\sigma}(v)\cdot\hat{\sigma}(w)$ iff there exists $k \in S$ such that $\langle i, k\rangle \in \hat{\sigma}(v)$ and $\langle k, j\rangle \in \hat{\sigma}(w)$. By induction, this last property is equivalent to the existence of two functions $\phi_u : V_u \to S$ et $\phi_v : V_v \to S$ such that $\phi_u(\iota_u) = i$, $\phi_u(o_u) = k$, $\langle p, a, q\rangle \in E_u$ entails $\langle \phi_u(p), \phi_u(q)\rangle \in \sigma(a)$, $\phi_v(\iota_v) = k$, $\phi_v(o_v) = j$, and $\langle p, a, q\rangle \in E_v$ entails $\langle \phi_v(p), \phi_v(q)\rangle \in \sigma(a)$. By gluing back those two functions, we easily show the equivalence with the existence of a function from the graph $G(u) = G(v)\cdot G(w)$ satisfying the property from the statement.
- $u = v\cap w$: with the notations from the previous point, we have $\langle i, j\rangle \in \hat{\sigma}(u) = \hat{\sigma}(v)\cap\hat{\sigma}(w)$ iff $\langle i, j\rangle \in \hat{\sigma}(v)$ and $\langle i, j\rangle \in \hat{\sigma}(w)$. By induction, this conjunction is equivalent to the existence of two functions $\phi_u : V_u \to S$ and $\phi_v : V_v \to S$ such that $\phi_x(\iota_x) = i$, $\phi_x(o_x) = j$, and $\langle p, a, q\rangle \in E_x$ entails $\langle \phi_x(p), \phi_x(q)\rangle \in \sigma(a)$, for $x \in \{u, v\}$. As previously one easily shows the equivalence with the existence of a function from the graph $G(u) = G(v)\cap G(w)$ satisfying the property from the statement.
- $u = 1$: by definition, we have $\langle i, j\rangle \in \hat{\sigma}(u) = 1$ iff $i = j$, and the existence of a function ϕ satisfying the properties of the statement for the graph $\underline{1}$ is also equivalent to $i = j$.
- $u = \top$: by definition, $\langle i, j\rangle \in \hat{\sigma}(u) = \top$ is always true; and the existence of a function ϕ satisfying the properties of the statement for the graph $\underline{\top}$ is always guaranteed.
- $u = a$: $\hat{\sigma}(u) = \sigma(a)$ the existence of a function ϕ satisfying the properties of the statement for the graph $\underline{a}$ is equivalent to the membership of $\langle i, j\rangle$ to $\sigma(a)$. ◀

Proof of Theorem 9. Write $G(u) = \langle V, E, \iota, o\rangle$ et $G(v) = \langle V', E', \iota', o'\rangle$.

Start by the right-to-left implication: assume $G(u) ◀ G(v)$, i.e., a homomorphism γ from $G(v)$ to $G(u)$, and let us show $\vDash u \subseteq v$. Let S be a set and $\sigma : \Sigma \to \mathcal{P}(S\times S)$ an interpretation function; for all $\langle i, j\rangle \in \hat{\sigma}(u)$ (†), we have to show $\langle i, j\rangle \in \hat{\sigma}(v)$ (‡). Let $\phi : V \to S$ be the function given by Lemma 11 and assumption (†). By the same lemma, to prove (‡) it suffices to find a function $\psi : V' \to S$ satisfying $\psi(\iota') = i$, $\psi(o') = j$, and $\langle p', a, q'\rangle \in E'$ entails $\langle \psi(p'), \psi(q')\rangle \in \sigma(a)$. The composed function $\phi \circ \gamma$ is suitable.

Now let us show the direct implication. Suppose that $\vDash u \subseteq v$, we have to find a homomorphism from $G(v)$ to $G(u)$. Let σ be the following interpretation function:

$$\begin{aligned} &\sigma : \Sigma \to \mathcal{P}(V\times V) \\ &\quad a \mapsto \{\langle p, q\rangle \mid \langle p, a, q\rangle \in E\} \end{aligned}$$

By Lemma 11, using the identity function, we have $\langle \iota, o\rangle \in \hat{\sigma}(u)$. By assumption, we deduce $\langle \iota, o\rangle \in \hat{\sigma}(v)$, whence, by using Lemma 11 again, the existence of a function $\phi : V' \to V$ satisfying some properties. These properties precisely correspond to the fact that ϕ is a homomorphism from $G(v)$ to $G(u)$. ◀

3.2 Axiomatisation

Freyd and Scedrov define *allegories* [11] as structures satisfying the axioms from Figure 5[2]. First note that composition does not distribute over intersections: composition is monotone in its two arguments, which entails the following inequations but not their converses:

$$\begin{aligned} &e \cdot (f \cap g) \subseteq e{\cdot}f \,\cap\, e{\cdot}g \\ &(f \cap g) \cdot e \subseteq f{\cdot}e \,\cap\, g{\cdot}e \end{aligned}$$

[2] Up-to some details: they do not consider the constant $\top$, and they work in a categorical setting, where the various operations are typed.

$$\left.\begin{aligned} e\cap(f\cap g)&=(e\cap f)\cap g\\ e\cap f&=f\cap e\\ e\cap\top&=e\\ e\cap e&=e\end{aligned}\right\}\ \text{$\langle\cap,\top\rangle$ is a commutative and idempotent monoid}$$

$$\left.\begin{aligned} e\cdot(f\cdot g)&=(e\cdot f)\cdot g\\ e\cdot 1&=e\\ 1\cdot e&=e\end{aligned}\right\}\ \text{$\langle\cdot,1\rangle$ is a monoid}$$

$$\left.\begin{aligned} e\cdot(f\cap g)&\subseteq e\cdot f\\ (f\cap g)\cdot e&\subseteq f\cdot e\end{aligned}\right\}\ \text{composition is monotone}$$

$$\left.\begin{aligned} e^{\circ\circ}&=e\\ (e\cap f)^\circ&\subseteq e^\circ\\ (e\cdot f)^\circ&\subseteq f^\circ\cdot e^\circ\end{aligned}\right\}\ \text{converse is a monotone involution reversing composition}$$

$$e{\cdot}f\,\cap\,g\subseteq(e\,\cap\,g{\cdot}f^\circ)\cdot f\quad\}\ \text{modularity law}$$

Figure 5 Axioms of allegories.

One can also deduce from the axioms that converse reverses composition, distributes over intersections, and preserves constants 1 et $\top$:

$$(e\cap f)^\circ=e^\circ\cap f^\circ \qquad\qquad \top^\circ=\top$$
$$(e\cdot f)^\circ=f^\circ\cdot e^\circ \qquad\qquad 1^\circ=1$$

The last axiom in Figure 5 is uncommon. It is called *modularity law*, it is equivalent in presence of the other axioms to its symmetrical counterpart:

$$e{\cdot}f\,\cap\,g\subseteq e\cdot(f\,\cap\,e^\circ{\cdot}g)$$

It also admits as a consequence the following inequation, known as *Dedekind's inequality*:

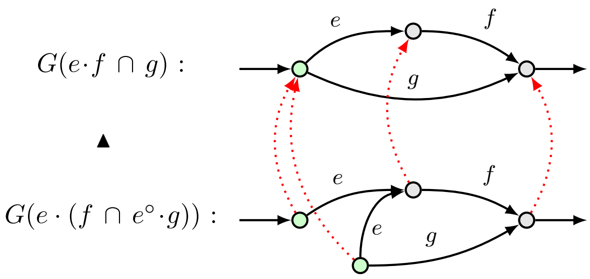
$$e{\cdot}f\,\cap\,g\subseteq(e\,\cap\,g{\cdot}f^\circ)\cdot(f\,\cap\,e^\circ{\cdot}g)$$

▶ **Exercise 12.** *Prove the six laws above from the axioms of Figure 5.*

Unfortunately, this finite and purely equational axiomatisation is not complete for relations: some valid equations are not consequences of the axioms. Freyd and Scedrov actually proved that there exists no finite equational axiomatisation. We give some intuitions about this result in the remainder of this section. Let us first check that the axiomatisation is sound:

▶ **Exercise 13.** *Prove that each axiom is valid by using Theorem 9: draw each graph and make explicit the homomorphisms corresponding to each inequation.*

When doing the above exercise, one can see that the only non-injective homomorphism is the one corresponding to the modularity law, and that this homomorphism equates exactly two vertices:



We actually have the following result:

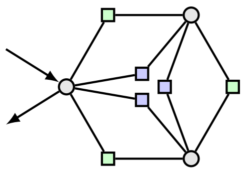
▶ **Claim 14.** *Let u and v be two terms. If there exists a homomorphism from $G(v)$ to $G(u)$ equating at most two vertices, then the inequality $u \subseteq v$ is a consequence of the axioms from Figure 5.*

Proof. Left to the reader by Freyd and Scedrov [11]. ◀

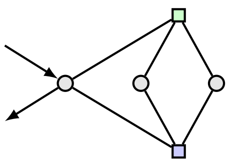
The converse does not hold: many inequations provable from the axioms correspond to homomorphisms equating arbitrarily many vertices (for instance, Dedekind's inequality, where two pairs of vertices are equated, or the inequation (1) from Exercice 2, where the five vertices of the right-hand side are equated).

Consider nevertheless an arbitrary homomorphism from the graph of a term v to that of a term u. This homomorphism can be decomposed in several ways into a sequence of homomorphisms each equating at most two vertices. One could thus believe that it suffices to use the claim 14 to obtain a sequence of provable inequations, leading to a proof of $u \subseteq v$ from the axioms and transitivity.

The problem is that the intermediate graphs appearing in these sequences of homomorphisms need not be graphs of terms (recall the graph (8)). Here is a counter-example; again, we do not label the edges nor we give their orientation as this information is irrelevant. Consider the following graph:



This graph corresponds to a term of the shape $1 \cap \Pi_{i=1,2,3}(a_i \cdot b_i \cap c_i \cdot d_i)$. If we equate the three inner, square, blue vertices, as well as the three outer, square, green vertices, we obtain the following graph:



This graph is associated to a term, of the shape $1 \cap i{\cdot}(e{\cdot}f \cap g{\cdot}h){\cdot}j$, so that the homomorphism implicitely considered corresponds to a valid inequation between two terms.

This homomorphism equates in one step two groups of three vertices. Now let us try to decompose it into a sequence of four morphisms equating each exactly two vertices. The first homomorphism must equate two blue vertices, or two green vertices. In both cases, we obtain a graph which is not the graph of any term.

By formalising this idea more precisely, one obtains a valid inequation which cannot be proved from the axioms, whence the incompletness of the axiomatisation. One can actually generalise the counter-example and show that every complete equational axiomatisation must contain axioms corresponding to homomorphisms equating arbitrarily many vertices, whence the impossibility for this axiomatisation to be finite [11, page 210].

Hodkinson and Mikulás further showed that there cannot be a finite first-order axiomatisation [12], and in particular a quasi-equational one like, e.g., for Kleene algebra. In contrast, we proved recently with Cosme-Llópez that the more restrictive theory of *isomorphism* (on graphs of terms) can be finitely axiomatised in a purely equational way [8].

4 Putting all together

Let us come back to the initial problem, that of the positive calculus of relations. We have seen that two fragments are decidable: the fragment corresponding to regular expressions $(+, \cdot, \cdot^\star, 0, 1)$, and that corresponding to allegories $(\cap, \cdot, \cdot^\circ, \top, 1)$. What happens when we take all operations?

First note that the function $[\cdot]$ associating a (regular) language to every regular expression can be extended to the operations of allegories:

$$\begin{aligned}
[e \cap f] &\triangleq [e] \cap [f] \\
[e^\circ] &\triangleq \{a_n \dots a_1 \mid a_1 \dots a_n \in [e]\} \\
[\top] &\triangleq \Sigma^\star
\end{aligned}$$

However, the characterisation obtained in Theorem 4 no longer works with these operations. Indeed, we have for instance

$$\begin{aligned}
[a \cap b] = \{a\} \cap \{b\} = \emptyset = [0] &\quad \text{but} \quad \not\models a \cap b = 0 \\
[a^\circ] = \{a\} = [a] &\quad \text{but} \quad \not\models a^\circ = a \\
[a] = \{a\} \not\subseteq \{aaa\} = [a \cdot a^\circ \cdot a] &\quad \text{but} \quad \models a \subseteq a \cdot a^\circ \cdot a \\
[\top \cdot a \cdot \top \cdot b \cdot \top] \neq [\top \cdot b \cdot \top \cdot a \cdot \top] &\quad \text{but} \quad \models \top \cdot a \cdot \top \cdot b \cdot \top = \top \cdot b \cdot \top \cdot a \cdot \top
\end{aligned}$$

To obtain a characterisation, we actually have to replace words (elements of $\Sigma^\star$) by graphs, and thus consider languages of graphs.

▶ **Definition 15.** The *language of graphs* of an expression e, written $\mathcal{G}(e)$, is defined as follows, by induction on e:

$$\begin{aligned}
\mathcal{G}(e + f) &\triangleq \mathcal{G}(e) \cup \mathcal{G}(f) & \mathcal{G}(0) &\triangleq \emptyset \\
\mathcal{G}(e \cap f) &\triangleq \{G \cap H \mid G \in \mathcal{G}(e),\ H \in \mathcal{G}(f)\} & \mathcal{G}(\top) &\triangleq \{\underline{\top}\} \\
\mathcal{G}(e \cdot f) &\triangleq \{G \cdot H \mid G \in \mathcal{G}(e),\ H \in \mathcal{G}(f)\} & \mathcal{G}(1) &\triangleq \{\underline{1}\} \\
\mathcal{G}(e^\star) &\triangleq \{G_1 \cdot \dots \cdot G_n \mid n \in \mathbb{N},\ \forall i \leq n, , G_i \in \mathcal{G}(e)\} & \mathcal{G}(a) &\triangleq \{\underline{a}\} \\
\mathcal{G}(e^\circ) &\triangleq \{G^\circ \mid G \in \mathcal{G}(e)\}
\end{aligned}$$

This definition properly generalises the usual notion of language: when the considered expression contains no intersection, no converse, and no constant $\top$, then the associated graphs are isomorphic to words: these are simple threads labelled by letters in Σ.

To generalise also allegories, we have to make use of graph homomorphisms. Given a set L of graphs, we write ${}^\blacktriangleleft L$ for is *downward closure* w.r.t. the preorder $(\blacktriangleleft)$:

$${}^\blacktriangleleft L \triangleq \{G \mid \exists H, G \blacktriangleleft H,\ H \in L\} \ .$$

We finally obtain the following characterisation:

▶ **Theorem 16.** *For all expressions e and f, we have*

$$\models e \subseteq f \qquad \textit{iff} \qquad \mathcal{G}(e) \subseteq {}^{\blacktriangleleft}\mathcal{G}(f) \ .$$

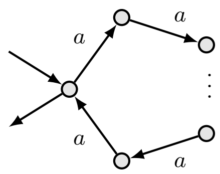
Proof. Similar to the proof of Theorem 4, using Theorem 9 (cf. [5, Theorem 6], adding the constant $\top$ is not problematic). ◀

This characterisation generalises both Theorem 4 and Theorem 9. If e and f are regular expressions, then all graphs in $\mathcal{G}(e)$ and $\mathcal{G}(f)$ are threads, and the unique possible homomorphism between two such graphs is the identity; whence $\mathcal{G}(e) \subseteq {}^{\blacktriangleleft}\mathcal{G}(f)$ iff $\mathcal{G}(e) \subseteq \mathcal{G}(f)$. If instead e and f are terms u and v, then $\mathcal{G}(e) = \{G(u)\}$ and $\mathcal{G}(f) = \{G(v)\}$, so that $\mathcal{G}(e) \subseteq {}^{\blacktriangleleft}\mathcal{G}(f)$ is equivalent to $G(u) \blacktriangleleft G(v)$.

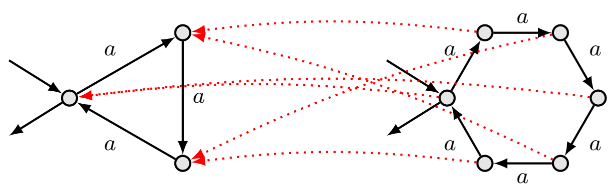
Note also that for all graph languages L, K, we have $L \subseteq {}^{\blacktriangleleft}K$ iff ${}^{\blacktriangleleft}L \subseteq {}^{\blacktriangleleft}K$. Valid equations are thus characterised as follows:

$$\models e = f \qquad \text{iff} \qquad {}^{\blacktriangleleft}\mathcal{G}(e) = {}^{\blacktriangleleft}\mathcal{G}(f) \ .$$

To illustrate this theorem, consider expressions $e \triangleq a^+ \cap 1$ and $f \triangleq (a{\cdot}a)^+ \cap 1$, where g^+ is a shorthand for $g{\cdot}g^\star$. The set of graphs $\mathcal{G}(e)$ is the set of non-trivial cycles labelled with a:



On the other side, $\mathcal{G}(f)$ is the set of non-trivial cycles of even length. Thus we immediately get $\mathcal{G}(f) \subseteq \mathcal{G}(e) \subseteq {}^{\blacktriangleleft}\mathcal{G}(e)$, whence $\models f \subseteq e$. The converse inequation is also valid: to each cycle from $\mathcal{G}(e)$, possibly of odd length, one can associate the cycle of double length, in $\mathcal{G}(f)$; indeed, there is a homomorphism from this cycle of double length into the shorter one:



▶ **Exercise 17.** *Use the same technique to prove the following laws:*

$$(a \cap b{\cdot}b)^\star \subseteq a^\star \cap b^\star$$
$$((a \cap b){\cdot}(1 \cap b){\cdot}(a \cap b))^\star \subseteq (a \cap b{\cdot}b)^\star$$
$$(a \cap b{\cdot}\top)^\star{\cdot}(1 \cap b{\cdot}\top) = (1 \cap \top{\cdot}b^\circ){\cdot}(a \cap \top{\cdot}b^\circ)^\star$$

Together with Paul Brunet [5], we proposed an automata model allowing us to recognise languages of graphs associated to expressions. This automata model takes inspiration from Petri nets [19, 17], which make it possible to explore richer structures than plain words. To each expression e, we associate what we call a *Petri automaton*, whose language is precisely ${}^{\blacktriangleleft}\mathcal{G}(e)$. Thanks to Theorem 16, the problem of validity of equations or inequations thus reduces to the problem of comparing Petri automata.

We solved this algorithmic problem only for a fragment of the calculus: we have to forbid converse and constants 1 and $\top$, and replace reflexive-transitive closure $\cdot^\star$ by transitive closure $\cdot^+$ (because reflexive-transitive closure implicitly contains the identity: we have $1 = 0^\star$). The corresponding equational theory was recently studied by Andréka, Mikulás, and Németi [1]: it coincide with that of languages over this signature. Under this restriction, the considered graphs are always acyclic, so that the automata become simpler to compare: we have shown that the problem of comparing these automata is EXPSPACE-complete [5].

Subsequently, Nakamura managed to prove that the problem remains in EXPSPACE in presence of converse and identity (but without $\top$, although his technique certainly applies) [18]. His solution consists in defining a notion of partial derivatives for graphs, similar to Antimirov' partial derivatives for regular expressions [3], and exploiting the fact that graph generated from a given expression have a bounded pathwidth [9].

5 Open questions

Is it possible to axiomatise the positive calculus of relations with transitive closure? For instance, do Kleene algebra axioms suffice when added to a complete axiomatisation of representable allegories? What about the fragment without converse, identity, and $\top$, studied by Andréka, Mikulás, and Németi [1]?

Note that intersection is the difficult operation: without intersection (and associated constant $\top$), we obtain *Kleene algebras with converse*, for which Bernátsky, Bloom, Ésik and Stefanescu have obtained decidability [4][3] and complete axiomatisability relatively to Kleene algebras: the following five axioms suffice when added to any complete axiomatisation of Kleene algebras (e.g., those from Figure 1) [10].

$$(e \cdot f)^\circ = f^\circ \cdot e^\circ \qquad e^{\circ\star} = e^{\star\circ} \qquad e \subseteq e \cdot e^\circ \cdot e$$
$$(e + f)^\circ = e^\circ + e^\circ \qquad e^{\circ\circ} = e$$

References

1 Hajnal Andréka, Szabolcs Mikulás, and István Németi. The equational theory of kleene lattices. *Theor. Comput. Sci.*, 412(52):7099–7108, 2011. doi:10.1016/j.tcs.2011.09.024.

2 H. Andréka and D.A. Bredikhin. The equational theory of union-free algebras of relations. *Algebra Universalis*, 33(4):516–532, 1995. doi:10.1007/BF01225472.

3 Valentin M. Antimirov. Partial derivatives of regular expressions and finite automaton constructions. *Theor. Comput. Sci.*, 155(2):291–319, 1996. doi:10.1016/0304-3975(95)00182-4.

4 Stephen L. Bloom, Zoltán Ésik, and Gheorghe Stefanescu. Notes on equational theories of relations. *algebra universalis*, 33(1):98–126, Mar 1995. doi:10.1007/BF01190768.

5 Paul Brunet and Damien Pous. Petri automata for kleene allegories. In *30th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2015, Kyoto, Japan, July 6-10, 2015*, pages 68–79. IEEE Computer Society, 2015. doi:10.1109/LICS.2015.17.

6 Paul Brunet and Damien Pous. Algorithms for kleene algebra with converse. *J. Log. Algebr. Meth. Program.*, 85(4):574–594, 2016. doi:10.1016/j.jlamp.2015.07.005.

7 J. H. Conway. *Regular algebra and finite machines*. Chapman and Hall, 1971.

[3] Which we refined to PSPACE-completeness with Brunet [6].

8 Enric Cosme-Llópez and Damien Pous. K4-free graphs as a free algebra. In Kim G. Larsen, Hans L. Bodlaender, and Jean-François Raskin, editors, *42nd International Symposium on Mathematical Foundations of Computer Science, MFCS 2017, August 21-25, 2017 - Aalborg, Denmark*, volume 83 of *LIPIcs*, pages 76:1–76:14. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2017. doi:10.4230/LIPIcs.MFCS.2017.76.

9 R. Diestel. *Graph Theory*. Graduate Texts in Mathematics. Springer Verlag, 2005.

10 Zoltán Ésik and L. Bernátsky. Equational properties of kleene algebras of relations with conversion. *Theor. Comput. Sci.*, 137(2):237–251, 1995. doi:10.1016/0304-3975(94)00041-G.

11 P. Freyd and A. Scedrov. *Categories, Allegories*. North Holland, 1990.

12 Ian Hodkinson and Szabolcs Mikulás. Axiomatizability of reducts of algebras of relations. *Algebra Universalis*, 43(2):127–156, 2000. doi:10.1007/s000120050150.

13 Dexter Kozen. A completeness theorem for kleene algebras and the algebra of regular events. In *Proceedings of the Sixth Annual Symposium on Logic in Computer Science (LICS '91), Amsterdam, The Netherlands, July 15-18, 1991*, pages 214–225. IEEE Computer Society, 1991. doi:10.1109/LICS.1991.151646.

14 Dexter Kozen. A completeness theorem for kleene algebras and the algebra of regular events. *Inf. Comput.*, 110(2):366–390, 1994. doi:10.1006/inco.1994.1037.

15 Daniel Krob. Complete systems of b-rational identities. *Theor. Comput. Sci.*, 89(2):207–343, 1991. doi:10.1016/0304-3975(91)90395-I.

16 Donald Monk. On representable relation algebras. *Michigan Math. J.*, 11(3):207–210, 09 1964. doi:10.1307/mmj/1028999131.

17 T. Murata. Petri nets: Properties, analysis and applications. *Proc. of the IEEE*, 77(4):541–580, Apr 1989. doi:10.1109/5.24143.

18 Yoshiki Nakamura. Partial derivatives on graphs for kleene allegories. In *32nd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2017, Reykjavik, Iceland, June 20-23, 2017*, pages 1–12. IEEE Computer Society, 2017. doi:10.1109/LICS.2017.8005132.

19 C. A. Petri. Fundamentals of a theory of asynchronous information flow. In *IFIP Congress*, pages 386–390, 1962.

20 Damien Pous. *Informatique Mathématique: une photographie en 2016*, chapter Algèbres de relations. E. Baudrier and L. Mazo (Eds), 2016. Course notes for the EJCIM research school.

21 V. Redko. On defining relations for the algebra of regular events. *Ukr. Mat. Z.*, 16:120–, 1964.

22 Arto Salomaa. Two complete axiom systems for the algebra of regular events. *J. ACM*, 13(1):158–169, 1966. doi:10.1145/321312.321326.

23 Larry J. Stockmeyer and Albert R. Meyer. Word problems requiring exponential time: Preliminary report. In Alfred V. Aho, Allan Borodin, Robert L. Constable, Robert W. Floyd, Michael A. Harrison, Richard M. Karp, and H. Raymond Strong, editors, *Proceedings of the 5th Annual ACM Symposium on Theory of Computing, April 30 - May 2, 1973, Austin, Texas, USA*, pages 1–9. ACM, 1973. doi:10.1145/800125.804029.

24 A. Tarski and S. Givant. *A Formalization of Set Theory without Variables*, volume 41 of *Colloquium Publications*. American Mathematical Society, Providence, Rhode Island, 1987.

25 Alfred Tarski. On the calculus of relations. *J. Symbolic logic*, 6:73–89, 1941.