Completeness Theorems for Behavioural Distances and Equivalences

Wojciech Krzysztof Różowski

A dissertation submitted in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

of

University College London.

Department of Computer Science

University College London

April 21, 2025

I, Wojciech Krzysztof Różowski, confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the work.

Abstract

My research is about stuff.

It begins with a study of some stuff, and then some other stuff and things.

There is a 300-word limit on your abstract.

Impact Statement

Outside academia: The discipline of formal verification enables making precise logical statements about computer systems to guarantee their correctness. This relies on the study of models of computation, which are mathematical objects representing the semantics of systems of interest. In theoretical computer science, it is customary to model computations as transition systems. This thesis is part of a larger research programme aimed at providing specification languages. for representing transition systems and studying formal systems for reasoning about the equivalence or similarity of systems represented by the syntax. One of the central kind of problems attached to this field of research are completeness problems, that concern showing that every semantic equivalence or similarity of transition systems can be witnessed through the means of axiomatic manipulation. Having a complete axiomatisation allows one to fully resort to syntactic reasoning, which is particularly amenable to implementation and thus desirable from the automated reasoning point of view.

Kleene Algebra (KA) [Koz94] is a central example of such specification language. KA is a foundation of tools relevant to industry, such as NetKAT [And+14], which enables reasoning about the behaviour of packet-passing Software-Defined Networks, or cf-GKAT [Zha+25], which can be used to certify the correctness of decompilation algorithms. This thesis particularly focuses on the semantic notions of behavioural distances and probabilistic language equivalence, where it makes its main contributions. Behavioural distances [BW01; Des+04] replace the strict notion of equivalence of states with a more liberal quantitative measure of dissimilarity. This is particularly desirable for stochastic or probabilistic systems, where a tiny observed perturbation would lead to inequivalence of states. Behavioural distances have been successfully applied to Markov Decision Processes within Reinforcement Learning (RL), with the key example being MICo (Matching under Independent Couplings) distance [Cas+21]. Additionally, probabilistic language equivalence, axiomatised in Chapter 4 of this thesis, underpins Apex [Kie+12], an automated equivalence checker for probabilistic programs.

Inside academia: This thesis makes original contributions within the field of theoretical computer science. The results in Chapter 2 are an adaptation of a concrete completeness result for behavioural distance of probabilistic transition systems to an instance of an abstract coalgebraic framework. Besides the basic examples provided recently by Lobbia et al [Lob+24], the content of Chapter 3 is the first work to propose a complete axiomatisation of a quantitative calculus of string diagrams through a systematic axiomatic foundation. Finally, Chapter 4 provides an alternative axiomatisation of language equivalence of generative probabilistic transition systems [GSS95] through a simple generalisation of Kleene's regular expressions. The completeness result makes use of recently developed theory of proper functors [Mil18] and provides further evidence that the use of coalgebras for proper functors provides a good abstraction for completeness theorems.

The entire material presented in this thesis has been accepted or is under review for several highly-ranked conferences in theoretical computer science. The results of this thesis have been disseminated to the computer science community through a series of seminar talks across the United Kingdom, the United States, and Germany.

Contents

List of Figures

Chapter 1

Introduction

One of the motivations for the mathematical study of the models of computation stems from the desire for precise and formal reasoning about the correctness of computer systems. These theoretical foundations enable formal verification experts to prove that systems deployed in safety-critical areas, such as avionics or health-care, behave as expected. In theoretical computer science it is customary to model computations as state transition systems, which are discrete models where a set of states is equipped with a notion of one-step observable behaviour, describing how the system evolves. Typical examples include finite automata, Kripke frames, and Markov chains among many others.

This central topic of this thesis are axiomatisations of behaviour of transition systems. By this we mean providing expression languages for representing the behaviour of transition systems and the study of formal systems for reasoning about equivalence or similarity of behaviours represented by expressions of the interest.

The interest in axiomatising behaviour of transition systems originates from the seminal work of Kleene on regular expressions [Kle51]. In his influential paper from 1951, Kleene introduced deterministic finite automata (DFAs), which are the fundamental model of sequential deterministic computations. Each state of a DFA can be associated with a formal language, a collection of strings that are accepted starting from a given state. This characterises an important class of formal languages, known as regular languages. Classically, two states are equivalent if they recognise the same language. In the same paper, Kleene proposed regular expressions, which are an algebraic specification language for DFAs and proved that both formalisms are equally expressive through a result known nowadays as Kleene's theorem. As an open problem, he left a completeness question: are there a finite number of rules that enable reasoning about language equivalence of regular expressions?

Shortly after Kleene's paper, Redko [Red64] demonstrated that one cannot use a finite number of equational axioms to axiomatise the language equivalence. But the search for axiomatisation made of more expressive rules continued. The first answer came in 1966 from Salomaa [Sal66], who presented two axiom systems. One was infinitary, and the other used finite equations along with an implicational rule encoding Arden's lemma [Ard61] for formal languages. While Salomaa's implicational axiomatisation later became a blueprint for inference systems for reasoning about semantic equivalence or similarity of transition systems, this formal system was not algebraic. Essentially, the implicational rule relied on the productivity side-condition called empty word property (EWP) that caused the resulting axiomatisation to be unsound under substitution of letters by arbitrary expressions. This problem has motivated several researchers including Conway [Con12], Krob [Kro90] and Boffa [Bof90] to pursue the problem of obtaining algebraic axiomatisation of language equivalence of DFAs, eventually leading to the celebrated completeness result of Kozen [Koz94]. The inference system of Kozen is known nowadays under the name Kleene Algebra (KA) and it forms a basis of several formal systems for equational reasoning about imperative programs [KS96], packet-passing software defined networks [And+14], and concurrent programs [Kap+18; Wag+19] among many others.

Besides DFAs, automata theorists have studied many variants of automata, including nondeterministic [RS59], weighted [Sch61] and probabilistic [Rab63] ones, usually focusing on the notion of language equivalence or inclusion. At the advent of process algebra in the 1980s, Milner and Park brought the concept of bisimilarity [Par81], a notion of equivalence finer than language equivalence, that was motivated by the needs of the study of concurrency theory and models such as labelled transition systems (LTSs). Essentially, language equivalence is a linear-time notion, as it hides the precise moment of resolving nondeterministic choice from the external observer. At the same time, bisimilarity allows for a more fine-grained comparison of behaviours by looking at the exact moment of resolving the nondeterministic choice.

Milner [Mil84] considered a variant of LTSs that he called charts and studied the associated problem of axiomatising the bisimilarity of charts. Interestingly, while the syntax of regular expressions can be used to specify behaviours of charts, it is not expressive, that is there exist behaviours that cannot be specified using Kleene's syntax. Instead, Milner proposed a more general language called the algebra of regular behaviours (ARB) featuring binders, action prefixing, and a recursion operator. The paper introducing ARB also provided a suitable generalisation of Salomaa's non-algebraic axiomatisation and demonstrated its soundness and completeness with respect to the bisimilarity of charts.

The completeness results of Salomaa, Kozen, and Milner mentioned above are prototypical instances of the vast strain of research that has been of particular interest to theoretical computer scientists for decades. Given a transition system model and an associated notion of semantic equivalence, having a complete axiomatisation allows one to reason about model behaviour through the syntactic manipulation of terms of the specification language, which is well-suited for implementation, automation, and formal reasoning. Each time when the needs of modelling computer systems result in a new transition system model or an associated notion of semantic equivalence, it is natural to ask about the complete axiomatisation.

This thesis provides contributions to the above outlined field of axiomatisations of behaviours of transition systems in two orthogonal directions.

1. The first part of the thesis is concerned with the study of formal systems for quantitative reasoning about behavioural distances, that replace conventional notions of behavioural equivalence with a quantitative measure of how close the behaviour of two states of transition systems is.

2. The second part focuses on probabilistic transition systems and presents a sound and complete axiomatisation of language equivalence of behaviours

specified through the syntax of Probabilistic Regular Expressions.

We now provide a brief outline of each of these directions.

1.1 Behavioural Distances

In many contexts, especially when dealing with probabilistic or quantitative models, focusing on exact equivalence of behaviours such as language equivalence or bisimilarity is too restrictive. A tiny perturbation in observed probability or weights of transition can deem two states inequivalent. Instead, it is often more meaningful to measure how far apart the behaviours of two states are.

This has motivated the development of behavioural distances, which endow the state spaces of transition systems with (pseudo)metric spaces quantifying the dissimilarity of states. In such a setting, states at distance zero are not necessarily the same, but rather equivalent with respect to some classical notion of behavioural equivalence. In a nutshell, equipping transition systems with such a notion of distance crucially relies on the possibility of lifting the distance between the states to the distance on the one-step observable behaviour of the transition system.

Behavioural distances first appeared in the context of probabilistic transition systems [Des+04; BW01], where one-step observable behaviour forms a probability distribution. In such a setting, in order to lift distances from the state space to one-step observable behaviour, one can rely on the classic Kantorovich lifting from transportation theory [Vil09].

More generally, behavioural distances are not limited to probabilistic or weighted systems; instead, they can be defined meaningfully for a variety of transition systems [Bal+18]. One of the simplest instances is deterministic finite automata, which can be equipped with a shortest-distinguishing-word distance [BKP18], where the longer the smallest word that can witness inequivalence of two states is, the closer the behaviour of compared states is. To illustrate that, given an alphabet $A = \{a\}$, we have that a state recognising the language $\{a, aa, aaa\}$ is closer to the one recognising $\{a, aa, aaa, aaa\}$ rather than $\{a\}$.

The study of axiomatisations of behavioural distances have mainly focused on concrete probabilistic cases [Bac+18a; Bac+18b; Bac+18c]. Axiomatisations of other important instances of behavioural distances are still underexplored. The main goal of the first part of this thesis is to initiate the study of axiomatisations and completeness problems for behavioural distances beyond the concrete probabilistic instances. Our starting point is the work of Bacci, Bacci, Larsen and Mardare [Bac+18a], who gave a sound and complete axiomatisation of branching-time behavioural distance of terms of a probabilistic process calculus.

1.2 Probabilistic Language Equivalence

In 1963, Rabin introduced probabilistic automata [Rab63]. This model captures the simple notion of randomised computation and acts as an acceptor for probabilistic languages. Under such semantics, each word over some fixed alphabet is associated with a weight from the unit interval capturing how likely the word is to be accepted. Throughout the years, Probabilistic Automata were deeply studied from an algorithmic point of view [Kie+11] that eventually enabled the development of practical verification tools for randomised programs [Kie+12].

In the process algebra community, Larsen and Skou [LS91] devised a notion of probabilistic bisimilarity, while Stark and Smolka [SS00] provided a probabilistic process calculus featuring binders and a recursion operator and gave a sound and complete axiomatisation of probabilistic bisimilarity of terms of their calculus. The later work of Silva and Sokolova [SS11] showed that one can extend Stark and Smolka's system with additional axioms characterising probabilistic language equivalence to obtain a complete axiomatisation of language equivalence.

While the result of Silva and Sokolova enables the use of the process algebraic syntax of Stark and Smolka for reasoning about probabilistic language equivalence, it is natural to ask if one could devise a simpler, binder-free specification language in the style of Kleene's Regular Expressions and provide a more streamlined axiomatisation in the style of Salomaa.

This problem is the central motivation for the second part of this thesis. One of the main inspirations for that comes from the probabilistic pattern matching community, where researchers already considered Regular Expression-like operations to specify probabilistic languages [Ros00]. They did so by replacing the union of languages and Kleene's star from the usual Regular Expression with their probabilistic counterparts, which respectively can be seen as a convex combination and a form of the Bernoulli process. At the same time, the precise connection of such syntaxes to the transition systems model was under-explored [Bee17] and the topic of axiomatisation was not tackled at all.

1.3 Coalgebra

Both behavioural distances and probabilistic language equivalence can be studied abstractly through the unifying framework of the universal coalgebra [Gum00; Rut00]. Coalgebras provide an abstract and uniform treatment of transition systems through the language of category theory. Generally speaking, transition systems can be seen as pairs consisting of a set of states and a transition function, mapping each state to its one-step behaviour. The coalgebraic outlook allows abstracting away the features of the one-step behaviour of the transition system, such as inputs, labels, nondeterminism, probability, and the like through the notion of a type, formally modelled as an endofunctor on the category of sets and functions. Given a type functor, one can uniformly instantiate abstract results concerning the transition systems of the interest.

In particular, each type of functor canonically determines a notion of behavioural equivalence of states. Under mild set-theoretic size constraints on the type functor, one can construct a final coalgebra, which provides a universal domain of behaviours of transition systems of interest. For example, the final coalgebra for the functor describing deterministic automata is isomorphic to the set of all formal languages over some alphabet [Rut00]. Concrete instances of coalgebraic behavioural equivalence usually capture variants of bisimilarity and coincide with the notions known for the literature such as bisimilarity of LTSs or probabilistic bisimilarity of Larsen and Skou [VR99].

Modelling finer notions of semantic equivalence can be phrased by changing the base category over which the type functor is defined to a more structured setting than sets and functions. For example, one of the ways to model probabilistic language equivalence in the language of coalgebra is to work with coalgebras for an appropriate type functor over the category of positive convex algebras [Sil+10; SS11]. In this category, the final coalgebra is precisely carried by the set of all probabilistic languages over some alphabet.

At the same time, the recent work on coalgebraic behavioural distances [Bal+18] provided a categorical generalisation of Kantorovich lifting to lifting endofunctors over the category of sets to the category of pseudometric spaces and nonexpansive maps between them. Final coalgebras for type functors obtained through such liftings come equipped with a pseudometric between behaviours. Such a coalgebraic outlook enables generalising the notions of behavioural distances beyond probabilistic transition systems and is extensively used in the first part of the thesis.

Using the theory of universal coalgebra for axiomatisation problems allows abstracting away the generic steps of completeness theorems and instantiating abstract categorical results to obtain concrete properties of transition systems of interest. For example, the recently developed theory of rational fixpoints [Mil10] for coalgebras for proper functors [Mil18] provides a useful generalisation of the notion of regular languages to coalgebraic generality. One of the concrete instances of such a theory enables characterising the analogue of regular languages in the case of probabilistic languages [SW18] and underpins key results in the second part of the thesis.

1.4 Overview of the thesis

Having outlined the scope and the main aims of this thesis, we summarise below the content of each chapter of the thesis and provide references to the main technical results. The description of each content chapter contains a table providing a high-level overview of the studied axiomatisation problem.

Chapter 2 presents a sound and complete axiomatisation of *shortest-distinguishing-word* distance between formal languages represented by regular expressions. The axiomatisation relies on a recently developed quantitative analogue of equational logic [MPP16], allowing manipulation of rational-indexed judgements of the form $e \equiv_r f$ meaning the distance between terms e and f is less or equal to r. The technical core of the chapter is dedicated to the completeness argument that draws techniques from order theory and Banach spaces to simplify the calculation of the behavioural distance to the point it can be then mimicked by axiomatic reasoning.

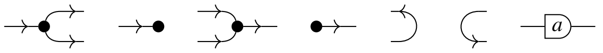
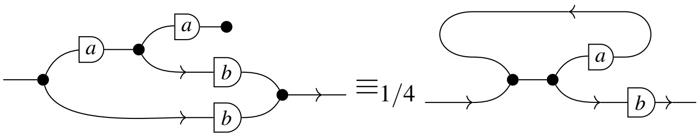
Summary of Chapter 2	
Model	deterministic finite automata (DFA)
Syntax	$e, f \in \mathsf{RExp} ::= 0 \mid 1 \mid a \in A \mid e + f \mid e \,; f \mid e^*$
Semantics	shortest-distinguishing-word distance of languages
Example fact	$a^* \equiv_{1/4} a + 1$
Soundness	Theorem 2.3.3
Completeness	Theorem 2.4.9

This chapter incorporates results from the following paper:

> Wojciech Różowski. "A Complete Quantitative Axiomatisation of Behavioural Distance of Regular Expressions". In: *51st International Colloquium on Automata, Languages, and Programming (ICALP 2024)*. Ed. by Karl Bringmann, Martin Grohe, Gabriele Puppis, and Ola Svensson. Vol. 297. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024, 149:1–149:20. ISBN: 978-3-95977-322-5

Chapter 3 describes a sound and complete axiomatisation of a behavioural metric for nondeterministic processes using Milner's charts [Mil84]—a model that generalises finite-state automata by incorporating variable outputs. Charts provide a compelling setting for studying behavioural distances because they shift the focus from language equivalence to bisimilarity.

To formalise this approach, we adopt string diagrams [Sel10; PZ23b] as our syntax of choice. String diagrams closely mirror the graphical structure of charts, while providing a rigorous formalism that supports inductive reasoning and compositional semantics. Unlike traditional algebraic syntaxes, which require additional mechanisms such as binders and substitution, string diagrams offer a variable-free representation where recursion naturally decomposes into simpler components. This makes them well-suited for reasoning about behavioural distances and aligns with broader efforts to axiomatise automata-theoretic equivalences through a unified diagrammatic framework [PZ23a; Ant+25].

Summary of Chapter 3	
Model	Milner's charts [Mil84]
Syntax	 $(a \in A)$
Semantics	bisimulation distance of regular behaviours
Example fact	
Soundness	**??**
Completeness	**??**

The findings presented in this chapter are the content of the following paper:

> Wojciech Różowski, Robin Piedeleu, Alexandra Silva, and Fabio Zanasi. "A Diagrammatic Axiomatisation of Behavioural Distance of Nondeterministic Processes". Under review. 2025

Chapter 4 introduces Probabilistic Regular Expressions (PRE), a probabilistic analogue of regular expressions denoting probabilistic languages in which every word is assigned a probability of being generated. PRE are formed through constants from an alphabet and regular operations of probabilistic choice, sequential composition, probabilistic Kleene star, identity and emptiness. We present and prove the completeness of an inference system for reasoning about probabilistic language equivalence of PRE based on Salomaa's axiomatisation of language equivalence of regular expressions. The technical core of the chapter is devoted to the completeness proof, which relies on technical tools from the theory of convex algebra [SW18], arising from the rich structure of probabilistic languages.

Summary of Chapter 4	
Model	generative probabilistic transition systems [GSS95]
Syntax	$e, f \in \mathsf{PExp} ::= 0 \mid 1 \mid a \in A \mid e \oplus_p f \mid e ; f \mid e^{[p]}$
Semantics	probabilistic language equivalence
Example fact	$a ; a^{[1/4]} \equiv a \oplus_{3/4} \left(a ; a^{[1/4]} ; a\right)$
Soundness	Theorem 4.4.20
Completeness	Theorem 4.5.22

The results described in this chapter were published in the paper referenced below:

> Wojciech Różowski and Alexandra Silva. “A Completeness Theorem for Probabilistic Regular Expressions”. In: *Proceedings of the 39th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2024, Tallinn, Estonia, July 8-11, 2024*. Ed. by Pawel Sobocinski, Ugo Dal Lago, and Javier Esparza. ACM, 2024, 66:1–66:14

Chapter 5 sketches directions for the future work and concludes this thesis.

Chapter 2

A Complete Quantitative Axiomatisation of Behavioural Distance of Regular Expressions

Deterministic automata have been traditionally studied through the point of view of language equivalence. Another perspective is given by the notion of *shortest-distinguishing-word* distance quantifying the dissimilarity of states. To illustrate that notion of distance, consider the following three deterministic finite automata:

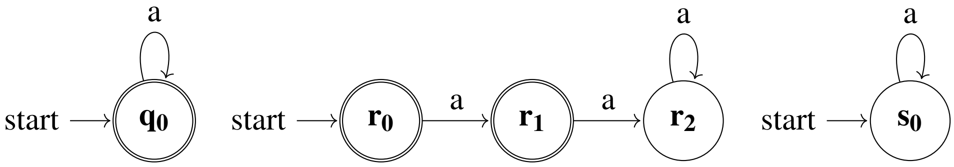


Figure 2.1: Three inequivalent DFAs

Neither of the above automata are language equivalent. Their languages are respectively: $\{\varepsilon, a, aa, aaa, \ldots\}$, $\{\varepsilon, a\}$, and $\emptyset$ (we use ε to denote the empty word). However, one could argue that the behaviour of the middle automaton is closer to the one on the left rather than the one on the right. In particular, languages of the left and middle automaton agree on all words of length less than two, while the left and right one disagree on all words.

One can make this idea precise, by providing a *shortest-distinguishing-word* metric $d_{\mathcal{P}(A^*)} \colon \mathcal{P}(A^*) \times \mathcal{P}(A^*) \to [0,1]$ on the set of all formal languages over some fixed alphabet A given by the following formula, where $\lambda \in \,]0,1[$ and $L,M \subseteq A^*$:

$$d_{\mathcal{L}}(L,M) = \begin{cases} \lambda^{|w|} & w \text{ is the shortest word that belongs to only one of } L \text{ and } M \\ 0 & \text{if } L = M \end{cases} \tag{2.1}$$

If we set $\lambda = \frac{1}{2}$, then

$$d_{\mathcal{P}(A^*)}(\{\varepsilon, a, aa, aaa, \ldots\}, \{\varepsilon, a\}) = \frac{1}{4} \quad \text{and} \quad d_{\mathcal{P}(A^*)}(\{\varepsilon, a, aa, aaa, \ldots\}, \emptyset) = 1$$

This allows us to formally state that the behaviour of the middle automaton is a better approximation of the left one, rather than the right one. Observe, that we excluded $\lambda = 0$ and $\lambda = 1$, as in both cases $d_{\mathcal{P}(A^*)}$ would become a pseudometric setting all languages to be at distance zero or one, without providing any quantitative information.

Equivalently, languages accepted by automata depicted on the Figure 2.1 can be represented using regular expressions a^*, $a+1$ and 0 respectively. To determine the distance between arbitrary regular expressions e and f one would have to construct corresponding deterministic finite automata and calculate (or approximate) the distance between their languages. Instead, as a main contribution of this chapter, we present a sound and complete quantitative inference system for reasoning about the shortest-distinguishing-word distance of languages denoted by regular expressions in question.

Since we are dealing with distances, rather than strict equality, we cannot rely on classical equational logic as a basis for our inference system. Instead, we rely on the quantitative analogue of equational logic [MPP16], which deals with the statements of the form $e \equiv_r f$, intuitively meaning *term e is within the distance of at most $r \in \mathbb{Q}$ from the term f*. While the existing work [Bac+18c; Bac+18a; Bac+18b] looked at quantitative axiomatisations of behavioural distance for probabilistic transition systems calculated through the Kantorovich lifting, which can be thought of as a special case of the abstract coalgebraic framework relying on lifting endofunctors to the category of pseudometric spaces [Bal+18], axiomatising behavioural distances for other kinds of transition systems have received little to no attention.

It turns out that the approach to completeness used in [Bac+18a] relies on properties which are not unique to distances obtained through the Kantorovich lifting and can be employed to give complete axiomatisations of behavioural distances for other kinds of transition systems obtained through the coalgebraic framework [Bal+18]. In this chapter, as a starting point, we look at one of the simplest instantiations of that abstract framework in the case of deterministic automata, yielding *shortest-distinguishing-word* distance.

Formally speaking, if $[\![-]\!] : \mathsf{RExp} \to \mathcal{P}(A^*)$ is the function taking regular expressions to their languages, then our inference system satisfies the following:

$$\vdash e \equiv_r f \iff d_{\mathcal{P}(A^*)}([\![e]\!], [\![f]\!]) \leq r$$

The rest of the chapter is organised is as follows:

In Section 2.1 we review basic definitions from the field of universal coalgebra [Rut00; Gum00] and automata theory. In particular, we recall the semantics of regular expressions through Brzozowski derivatives [Brz64]. Then, in order to talk about distances, we state basic definitions and properties surrounding (pseudo)metric spaces.

In Section 2.2 we recall the central notions of the abstract framework of coalgebraic behavioural metrics [Bal+18] and discuss its concrete instatiation to the concrete case of deterministic automata that yields shortest-distinguishing-word distance.

In Section 2.3 we introduce a quantitative inference system for reasoning about the shortest-distinguishing-word distance of regular expressions. We recall the definitions surrounding the quantitative equational theories [MPP16] from the literature. We then present the rules of our inference system, give soundness result and provide a discussion about the axioms. The interesting insight is that when relying on quantitative equational theories which contain an infinitary rule capturing the notion of convergence, there is no need for any fixpoint introduction rule. We illustrate this by axiomatically deriving Salomaa's fixpoint rule for regular expressions [Sal66].

The key result of our paper is contained in Section 2.4, where we prove completeness of our inference system. The heart of the argument relies on showing that the behavioural distance of regular expressions can be approximated from above using Kleene's fixpoint theorem, which can be then mimicked through the means of axiomatic reasoning. This part of the paper makes heavy use of the order-theoretic and Banach space structures carried by the sets of pseudometrics over a given set.

We conclude in Section 2.5, review related literature, and sketch directions for future work.

2.1 Preliminaries

In this section, we recall the main definitions and results from the literature that this and further chapters rely on. Throughout this thesis, we assume the familiarity of the reader with basic notions of category theory [AT11] and order theory [DP02]. Notation wise, given a category $\mathcal{C}$, we will write $\mathsf{Obj}(\mathcal{C})$ for the collection of its objects. For $X, Y \in \mathsf{Obj}(\mathcal{C})$, we will write $\mathcal{C}(X,Y)$ for the hom-object between objets X and Y. We will write $f \colon X \to Y$, to denote that f is a morphism from X to Y.

2.1.1 Coalgebra

Let $\mathcal{C}$ be a category. An $\mathcal{B}$-coalgebra is a pair $(X, \alpha \colon X \to \mathcal{B}X)$, where $X \in \mathsf{Obj}(\mathcal{C})$ and $\mathcal{B} \colon \mathcal{C} \to \mathcal{C}$ is an endofunctor on $\mathcal{C}$. We call $\mathcal{B}$ a *type functor* and refer to X and α as *state space* (or a *carrier*) and *transition structure* respectively. We will omit writing $\mathcal{B}$ when it is obvious from the context. A homomorphism $f \colon (X,\alpha) \to (Y,\beta)$ of coalgebras is an arrow $f \colon X \to Y$ in $\mathcal{C}$ making the following diagram commute:

$$\begin{tikzcd}
X \arrow[r, "f"] \arrow[d, "\alpha"'] & Y \arrow[d, "\beta"] \\
\mathcal{B}X \arrow[r, "\mathcal{B}f"'] & \mathcal{B}Y
\end{tikzcd}$$

$\mathcal{B}$-coalgebras and their homomorphisms form a category $\mathsf{Coalg}\,\mathcal{B}$.

Definition 2.1.1. We call a coalgebra $(\nu\mathcal{B}, t)$ *final* if for any coalgebra (X,α), there exists a unique homomorphism $\mathsf{beh}_\alpha \colon (X,\alpha) \to (\nu\mathcal{B}, t)$. A final coalgebra (if it exists) is precisely the final object in $\mathsf{Coalg}\,\mathcal{B}$.

If $\mathcal{C}$ is a concrete category, that is equipped with a faithful functor $\mathcal{U}\colon \mathcal{C} \to \mathsf{Set}$, one can define the notion of *behavioural equivalence*. All coalgebras considered in this thesis are defined over concrete categories.

Definition 2.1.2. Given $\mathcal{B}$-coalgebras (X,α) and (Y,β), and elements $x \in \mathcal{U}X$, $y \in \mathcal{U}Y$, we say that x is behaviourally equivalent to y (written $x \sim_{\mathsf{b}} y$), if there exists a third coalgebra (Z,γ) and $\mathcal{B}$-coalgebra homomorphisms $f\colon (X,\alpha) \to (Z,\gamma)$ and $g\colon (Y,\beta) \to (Z,\gamma)$, such that $\mathcal{U}f(x) = \mathcal{U}g(x)$.

For the remainder of this subsection, we will focus on properties of coalgebras for endofunctors over Set by setting $\mathcal{B}\colon \mathsf{Set} \to \mathsf{Set}$. For such coalgebras, one can phrase the notion of *coalgebraic bisimulation*.

Definition 2.1.3. Let (X,α) and (Y,β) be two coalgebras for the functor $\mathcal{B}\colon \mathsf{Set} \to \mathsf{Set}$. We call a relation $R \subseteq X \times Y$ a bisimulation if there exists a transition function $R \to \mathcal{B}R$ making the following diagram commute:

$$\begin{tikzcd}
X \arrow[d, "\alpha"] & R \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] \arrow[d] & Y \arrow[d, "\beta"] \\
\mathcal{B}X & \mathcal{B}R \arrow[l, "\mathcal{B}\pi_1"] \arrow[r, "\mathcal{B}\pi_2"'] & \mathcal{B}Y
\end{tikzcd}$$

In the above, $\pi_1\colon R \to X$ and $\pi_2\colon R \to Y$ are the canonical projection maps given by the product structure on $X \times Y$. Given $\langle x,y\rangle \in X \times Y$, we write $x \sim y$ if there exists a bisimulation R between (X,α) and (Y,γ), such that $\langle x,y\rangle \in R$. Moreover, constructing bisimulations is a sound technique for proving behavioural equivalence. It is also complete upon imposing a mild restriction on $\mathcal{B}$.

Lemma 2.1.4 ([Rut00, Theorem 9.3])**.** *We have that $x \sim y \implies x \sim_{\mathsf{b}} y$. The converse is true if $\mathcal{B}$ preserves weak pullbacks.*

Bisimulations and homomorphisms are related via the following lemma:

Lemma 2.1.5 ([Rut00, Theorem 2.5])**.** *Let (X,α) and (Y,β) be two coalgebras. A function $f\colon X \to Y$ is a homomorphism if and only if $G(f) = \{\langle x, f(x)\rangle \mid x \in X\} \subseteq X \times Y$ is a bisimulation.*

We call a bisimulation that is an equivalence relation a bisimulation equivalence. Forming a quotient using bisimulation equivalences can be used to construct quotient coalgebras.

Lemma 2.1.6 ([Rut00, Proposition 5.8]). *Let $R \subseteq X \times X$ be a bisimulation equivalence on a coalgebra (X,α). Let $[-]_R \colon X \to X/R$, be the canonical quotient map of R. Then, there is a unique transition structure $\overline{\alpha} \colon X/R \to \mathcal{B}X/R$ on X/R, that makes $[-]_R$ into a coalgebra homomorphism, thus making the following diagram commute:*

$$\begin{tikzcd}
X \arrow[r, "{[-]_R}"] \arrow[d, "\alpha"] & X/R \arrow[d, "\overline{\alpha}"] \\
\mathcal{B}X \arrow[r, "{\mathcal{B}[-]_R}"'] & \mathcal{B}X/R
\end{tikzcd}$$

Moreover, one can phrase the dual notion of subalgebras.

Definition 2.1.7. A coalgebra (X,α) is called a subcoalgebra of (Y,β), if $X \subseteq Y$ and the canonical inclusion map $i \colon X \hookrightarrow Y$ is a coalgebra homomorphism.

Upon imposing a mild restriction on $\mathcal{B}$, subcoalgebras carry a lattice structure.

Lemma 2.1.8 ([Rut00, Theorem 6.4.]). *If $\mathcal{B}$ preserves weak pullbacks, then the collection of all subcoalgebras of a system (Y,β) is a complete lattice. Least upper bounds and greatest lower bounds are respectively given by union and intersection of sets.*

Given a set $X \subseteq Y$, we will write $\langle X \rangle_{(Y,\beta)}$ for the least subcoalgebra of (Y,β) containing X. When (Y,β) is obvious from the context, we will omit writing it in the subscript. In the case when X is a singleton or a two-element set, we will lighten up the notation and respectively write $\langle x \rangle_{(Y,\beta)}$ and $\langle x,y \rangle_{(Y,\beta)}$ instead. Least subcoalgebras allow to characterise an important subcategory of coalgebras.

Definition 2.1.9. We call a coalgebra (X,α) locally finite if for all $x \in X$, we have that $\langle x \rangle_{(X,\alpha)}$ is finite.

We will write $\mathsf{Coalg}_{\mathsf{lf}}\,\mathcal{B}$ for the full subcategory of $\mathsf{Coalg}\,\mathcal{B}$ consisting only of locally finite coalgebras.

2.1.2 Deterministic automata

A deterministic automaton $\mathcal{M}$ with inputs in a finite alphabet A is a pair $(M, \langle o_M, t_M \rangle)$ consisting of a set of states M and a pair of functions $\langle o_M, t_M \rangle$, where $o_M \colon M \to \{0,1\}$ is the *output* function which determines whether a state m is final ($o_M(m) = 1$) or not ($o_M(m) = 0$), and $t : M \to M^A$ is the *transition* function, which, given an input letter a determines the next state. If the set M of states is finite, then we call an automaton $\mathcal{M}$ a deterministic finite automaton (DFA). We will frequently write m_a to denote $t_M(m)(a)$ and refer to m_a as the derivative of m for the input a. Definition of derivatives can be inductively extended to words $w \in A^*$. We will write ε to denote an empty word. We set $m_\varepsilon = m$ and $m_{aw'} = (m_a)_{w'}$ for $a \in A, w' \in A^*$.

Remark 2.1.10. Note that our definition of deterministic automaton slightly differs from the most common one in the literature, by not explicitly including the initial state. Instead of talking about the language of the automaton, we will talk about the languages of particular states of the automaton.

Given a state $m \in M$, we write $L_{\mathcal{M}}(m) \subseteq A^*$ for its language, which is formally defined by $L_{\mathcal{M}}(m) = \{w \in A^* \mid o(m_w) = 1\}$. Given two deterministic automata $(M, \langle o_M, t_M \rangle)$ and $(N, \langle o_N, t_N \rangle)$, a function $h \colon M \to N$ is a homomorphism if it preserves outputs and input derivatives, that is $o_N(h(m)) = o_M(m)$ and $h(m)_a = h(m_a)$. The set of all languages $\mathcal{P}(A^*)$ over an alphabet A can be made into a deterministic automaton $(\mathcal{P}(A^*), \langle o_L, t_L \rangle)$, where for $l \in \mathcal{P}(\Sigma^*)$ the output function is given by $o_L(l) = [\varepsilon \in l]$ and for all $a \in A$ the input derivative is defined to be $l_a = \{w \mid aw \in l\}$. This automaton is *final*, that is for any other automaton $\mathcal{M} = (M, \langle o_M, t_M \rangle)$ there exists a unique homomorphism from M to $\mathcal{P}(A^*)$, which is given by the map $L_{\mathcal{M}} \colon M \to \mathcal{P}(A^*)$ taking each state $m \in M$ to its language.

Remark 2.1.11. Deterministic automata are precisely coalgebras for the functor $\mathcal{H} \colon \mathsf{Set} \to \mathsf{Set}$ given by $\mathcal{H} = \{0,1\} \times (-)^A \colon \mathsf{Set} \to \mathsf{Set}$. The coalgebraic definition of homomorphism coincides with the definition of an automaton homomorphism stated above. The final coalgebra for that functor corresponds to the final automaton defined on the set $\mathcal{P}(A^*)$.

2.1.3 Regular expressions

We let e, f range over *regular expressions over* A generated by the following grammar:

$$e, f \in \mathsf{RExp} ::= 0 \mid 1 \mid a \in A \mid e + f \mid e\,; f \mid e^*$$

The standard interpretation of regular expressions $[\![-]\!] : \mathsf{RExp} \to \mathcal{P}(A^*)$ is inductively defined by the following equation:

$$[\![0]\!] = \emptyset \quad [\![1]\!] = \{\varepsilon\} \quad [\![a]\!] = \{a\} \quad [\![e+f]\!] = [\![e]\!] \cup [\![f]\!]$$
$$[\![e\,;f]\!] = [\![e]\!] \diamond [\![f]\!] \quad [\![e^*]\!] = [\![e]\!]^*$$

Given $L, M \subseteq A^*$, we define $L \diamond M = \{lm \mid l \in L, m \in M\}$, where mere juxtaposition denotes concatenation of words. L^* denotes the *asterate* of the language L defined as $L^* = \bigcup_{i \in \mathbb{N}} L^i$ with $L^0 = \{\varepsilon\}$ and $L^{n+1} = L \diamond L^n$.

2.1.4 Brzozowski derivatives

Kleene's theorem states that the formal languages accepted by DFA are in one-to-one correspondence with formal languages definable by regular expressions. One direction of this theorem involves constructing a DFA for an arbitrary regular expression. The most common way is via Thompson construction, ε-transition removal and determinisation. Instead, we recall a direct construction due to Brzozowski [Brz64], in which the set RExp of regular expressions is equipped with a structure of deterministic automaton $\mathcal{R} = (\mathsf{RExp}, \langle o_{\mathcal{R}}, t_{\mathcal{R}} \rangle)$ through so-called Brzozowski derivatives [Brz64]. The output derivative $o_{\mathcal{R}} : \mathsf{RExp} \to \{0, 1\}$ is defined inductively by the following

$$o_{\mathcal{R}}(0) = 0 \quad o_{\mathcal{R}}(1) = 1 \quad o_{\mathcal{R}}(a) = 0$$
$$o_{\mathcal{R}}(e+f) = o_{\mathcal{R}}(e) \vee o_{\mathcal{R}}(f) \quad o_{\mathcal{R}}(e\,;f) = o_{\mathcal{R}}(e) \wedge o_{\mathcal{R}}(f) \quad o_{\mathcal{R}}(e^*) = 1$$

for $a \in A$ and $e, f \in \mathsf{RExp}$. Similarly, the transition derivative $t_{\mathcal{R}} \colon \mathsf{RExp} \to A \to \mathsf{RExp}$ denoted $t_{\mathcal{R}}(e)(a) = (e)_a$ is defined by

$$(0)_a = 0 \quad (1)_a = 0 \quad (a')_a = \begin{cases} 1 & a = a' \\ 0 & a \neq a' \end{cases}$$

$$(e+f)_a = (e)_a + (f)_a \quad (e\,;f)_a = (e_a)\,;f + o_{\mathcal{R}}(e)\,;f \quad (e^*) = (e)_a\,;e^*$$

Semantics of regular expressions is well-behaved, that is the standard interpretation $[\![-]\!]$ assigning a language to each regular expression concides with the canonical language-assigning homomorphism from $\mathcal{R}$ to $\mathcal{L}$.

Lemma 2.1.12 ([Sil10, Theorem 3.1.4])**.** *For all $e \in \mathsf{RExp}$, $[\![e]\!] = L_{\mathcal{R}}(e)$*

Instead of looking at an infinite-state automaton defined on the state-space of all regular expressions, we can restrict ourselves to the subautomaton $\langle e \rangle_{\mathcal{R}}$ of $\mathcal{R}$ while obtaining the semantics of e.

Lemma 2.1.13. *For all $e \in \mathsf{RExp}$, $[\![e]\!] = L_{\langle e \rangle_{\mathcal{R}}}(e)$*

Proof. Let $i \colon \langle e \rangle_{\mathcal{R}} \hookrightarrow \mathsf{RExp}$ be the canonical inclusion homomorphism. Composing it with $L_{\mathcal{R}}$ a unique homomorphism from $\mathcal{R}$ into the final automaton $\mathcal{L}$ yields a homomorphism $L_{\mathcal{R}} \circ i$ from $\langle e \rangle_{\mathcal{R}}$ to the final automaton, which by finality is the same as $L_{\langle e \rangle_{\mathcal{R}}}$. Using Lemma 2.1.12 we can show the following:

$$[\![e]\!] = L_{\mathcal{R}}(e) = L_{\mathcal{R}}(i(e)) = L_{\langle e \rangle_{\mathcal{R}}}(e)$$

□

Unfortunately, for an arbitrary regular expression $e \in \mathsf{RExp}$, the automaton $\langle e \rangle_{\mathcal{R}}$ is not guaranteed to have a finite set of states. However, simplifying the transition derivatives by quotienting the expressions by associativity, commutativity and idempotence (ACI) guarantees a finite number of reachable states from any expression. Formally speaking, let $\doteq \subseteq \mathsf{RExp} \times \mathsf{RExp}$ be the least congruence relation closed under

1. $(e+f)+g \doteqdot e+(f+g)$ (Associativity)

2. $e+f \doteqdot f+e$ (Commutativity)

3. $e \doteqdot e+e$ (Idempotence)

for all $e, f, g \in \mathsf{RExp}$. We will write $\mathsf{RExp}/\doteqdot$ for the quotient of RExp by the relation $\doteqdot$ and $[-]_{\doteqdot} \colon \mathsf{RExp} \to \mathsf{RExp}/\doteqdot$ for the canonical map taking each expression $e \in \mathsf{RExp}$ into its equivalence class $[e]_{\doteqdot}$ modulo $\doteqdot$. It can be easily verified that $\doteqdot$ is a bisimulation and hence using Lemma 2.1.6, one can equip $\mathsf{RExp}/\doteqdot$ with a structure of deterministic automaton $\mathcal{Q} = (\mathsf{RExp}/\doteqdot, \langle o_{\mathcal{Q}}, t_{\mathcal{Q}} \rangle)$, where for all $e \in \mathsf{RExp}, a \in A$, $o_{\mathcal{Q}}([e]_{\doteqdot}) = o_{\mathcal{R}}(e)$ and $([e]_{\doteqdot})_a = [e_a]_{\doteqdot}$, which makes the quotient map $[-]_{\doteqdot} \colon \mathsf{RExp} \to \mathsf{RExp}/\doteqdot$ into an automaton homomorphism from the Brzozowski automaton $\mathcal{R}$ into $\mathcal{Q}$. This automaton enjoys the following property:

Lemma 2.1.14 ([Brz64, Theorem 4.3]). *For any $e \in \mathsf{RExp}$, the set $\langle e \rangle_{\mathcal{Q}} \subseteq \mathsf{RExp}/\doteqdot$ is finite.*

Through an identical line of reasoning to Lemma 2.1.12, we can show that:

Lemma 2.1.15. *For all $e \in \mathsf{RExp}$, $L_{\langle [e]_{\doteqdot} \rangle_{\mathcal{Q}}}([e]_{\doteqdot}) = [\![e]\!]$*

2.1.5 Pseudometric spaces

A 1-bounded *pseudometric* on a set X (or equivalently just a *pseudometric*) is a function $d \colon X \times X \to [0,1]$ satisfying

1. $d(x,x) = 0$ (Reflexivity)

2. $d(x,y) = d(y,x)$ (Symmetry)

3. $d(x,z) \leq d(x,y) + d(y,z)$ (Triangle inequality)

for all $x, y, z \in X$. If additionally $d(x,y) = 0$ implies $x = y$, d is called a (1-bounded) *metric*.

Definition 2.1.16. A pseudometric space is a pair (X,d), where X is a set and d is a pseudometric on X. We call a function $f \colon X \to Y$ between pseudometric spaces (X,d_1) and (Y,d_2) nonexpansive, if $d_2(f(x), f(y)) \leq d_1(x,y)$ for all $x, y \in X$. It is called isometry if it satisfies $d_Y(f(x), f(y)) = d_X(x,y)$.

Pseudometrics and nonexpansive functions form a category PMet. This category is bicomplete, i.e. has all limits and colimits [Bal+18, Theorem 3.8]. The categorical product in PMet is defined as follows:

Definition 2.1.17. Let (X,d_1) and (Y,d_2) be pseudometrics. We define $(X,d_1)\times(Y,d_2)=(X\times Y,d_{X\times Y})$, where $d_{X\times Y}(\langle x,y\rangle,\langle x',y'\rangle)=\max\{d_1(x,x'),d_2(y,y')\}$ for all $x,x'\in X$ and $y,y'\in Y$.

This can be easily extended to any n-tuple. We define 0-tuples to be given by $1_\bullet=(\{\bullet\},d_\bullet)$, the unique single point pseudometric space, where $d_\bullet(\bullet,\bullet)=0$. Given a function of multiple arguments, i.e. $X_1\to X_2\to Y$, we will call it nonexpansive, if it is nonexpansive as a function $f\colon(X_1,d_1)\times(X_2,d_2)\to(Y,d_Y)$.

Given a set X, we write D_X for the set of all pseudometrics on the set X. This set carries a partial order structure, given by

$$d_1\sqsubseteq d_2 \iff \forall x,y\in X.\, d_1(x,y)\leq d_2(x,y)$$

Lemma 2.1.18 ([Bal+18, Lemma 3.2])**.** *$(D_X,\sqsubseteq)$ is a complete lattice. The join of an arbitrary set of pseudometrics $D\subseteq D_X$ is taken pointwise, ie.* $(\sup D)(x,y)=\sup\{d(x,y)\mid d\in D\}$ *for $x,y\in X$. The meet of D is defined to be* $\inf D=\sup\{d\mid d\in D_X,\forall d'\in D,d\sqsubseteq d'\}$.

The top element of that lattice is given by the discrete pseudometric $\top\colon X\times X\to[0,1]$ such that $\top(x,y)=0$ if $x=y$, or $\top(x,y)=1$ otherwise.

Crucially for our completeness proof, if we are dealing with descending chains, that is sequences $\{d_i\}_{i\in\mathbb{N}}$, such that $d_i\sqsupseteq d_{i+1}$ for all $i\in\mathbb{N}$, then we can also calculate infima in the pointwise way.

Lemma 2.1.19. *Let $\{d_i\}_{i\in\mathbb{N}}$ be an infinite descending chain in the lattice $(D_X,\sqsubseteq)$ of pseudometrics over some fixed set X. Then* $(\inf\{d_i\mid i\in\mathbb{N}\})(x,y)=\inf\{d_i(x,y)\mid i\in\mathbb{N}\}$ *for any $x,y\in X$.*

Proof. It suffices to argue that $d(x,y) = \inf\{d_i(x,y) \mid i \in \mathbb{N}\}$ is a pseudometric. For reflexivity, observe that $d(x,x) = \inf\{d_i(x,x) \mid i \in \mathbb{N}\} = \inf\{0\} = 0$ for all $x \in X$.

For symmetry, we have that $d(x,y) = \inf\{d_i(x,y) \mid i \in \mathbb{N}\} = \inf\{d_i(y,x) \mid i \in \mathbb{N}\} = d(y,x)$ for any $x,y \in X$.

The only difficult case is triangle inequality. First, let $i,j \in \mathbb{N}$ and define $k = \max(i,j)$. Since $d_k \sqsubseteq d_i$ and $d_k \sqsubseteq d_j$, we have that $d_k(x,y) + d_k(y,z) \leq d_i(x,y) + d_j(y,z)$. Therefore $\inf\{d_l(x,y) + d_l(y,z) \mid l \in \mathbb{N}\}$ is a lower bound of $d_i(x,y) + d_j(y,z)$ for any $i,j \in \mathbb{N}$ and hence it is below the greatest lower bound, that is $\inf\{d_l(x,y) + d_l(y,z) \mid l \in \mathbb{N}\} \leq \inf\{d_i(x,y) + d_j(y,z) \mid i,j \in \mathbb{N}\}$. We can use that property to show that

$$\begin{aligned} d(x,y) &= \inf\{d_i(z,y) \mid i \in \mathbb{N}\} \\ &\leq \inf\{d_i(x,y) + d_i(y,z) \mid i \in \mathbb{N}\} \\ &\leq \inf\{d_i(x,y) + d_j(y,z) \mid i,j \in \mathbb{N}\} \\ &= \inf\{d_i(x,y) \mid i \in \mathbb{N}\} + \inf\{d_j(y,z) \mid j \in \mathbb{N}\} \\ &= d(x,y) + d(y,z) \end{aligned}$$

which completes the proof. □

Additionally, the set of pseudometrics can be equipped with a norm. We write $\overline{\mathbb{R}} = [-\infty, \infty]$ for the set of extended reals. For any set X, the set of functions $\overline{\mathbb{R}}^{X \times X}$, which is a superset of D_X, can be seen as a Banach space [Rud90] (complete normed vector space) by means of the sup-norm $\|d\| = \sup_{x,y \in X} |d(x,y)|$. This structure will implicitly underly some of the claims used as intermediate steps in the proof of completeness in this and next chapter.

2.2 Behavioural distance of deterministic automata

We now focus on defining a behavioural distance for deterministic automata through the abstract framework of coalgebraic behavioural distances [Bal+18]. We first recall the main definitions and then concretise the abstract results to the case of our interest.

2.2.1 Coalgebraic behavioural distances

In order to define a behavioural distance for $\mathcal{B}$-coalgebras for a functor $\mathcal{B}\colon \mathsf{Set} \to \mathsf{Set}$, we need to be able to *lift* the functor $\mathcal{B}$ describing the one-step dynamics of transition systems of interest to the category PMet of pseudometric spaces and nonexpansive functions. In terms of notation, we will write $\mathcal{U}\colon \mathsf{PMet} \to \mathsf{Set}$ for the canonical faithful functor taking each pseudometric space (X, d_X) to its underlying set X.

Definition 2.2.1. Let $\mathcal{B}\colon \mathsf{Set} \to \mathsf{Set}$ be a functor. We call a functor $\overline{\mathcal{B}}\colon \mathsf{PMet} \to \mathsf{PMet}$ a lifting of $\mathcal{B}$ if makes the following diagram commute:

$$\begin{tikzcd}
\mathsf{PMet} \arrow[r, "\overline{\mathcal{B}}"] \arrow[d, "\mathcal{U}"'] & \mathsf{PMet} \arrow[d, "\mathcal{U}"] \\
\mathsf{Set} \arrow[r, "\mathcal{B}"] & \mathsf{Set}
\end{tikzcd}$$

Given a pseudometric space (X, d), we will write $d^{\mathcal{B}}$ for the pseudometric $d^{\mathcal{B}}\colon \mathcal{B}X \times \mathcal{B}X \to [0,1]$ obtained by applying $\overline{\mathcal{B}}$ to (X, d).

We can use liftings to equip coalgebras with a notion of behavioural distance, through the following construction:

Lemma 2.2.2 ([Bal+18, Lemma 6.1])**.** *Let $\overline{\mathcal{B}}\colon \mathsf{PMet} \to \mathsf{PMet}$ be a lifting of a functor $\mathcal{B}\colon \mathsf{Set} \to \mathsf{Set}$ and let (X, α) be a $\mathcal{B}$-coalgebra. The mapping associating each pseudometric $d\colon X \times X \to [0, \top]$ with $d^{\mathcal{B}} \circ (\alpha \times \alpha)$ is a monotone mapping on the complete lattice $(D_X, \sqsubseteq)$ of pseudometrics over set X. By Knaster-Tarski fixpoint theorem, this mapping has a least fixpoint, that we will refer to as $d_\alpha\colon X \times X \to$* $[0,1]$*. Given a coalgebra (Y, β) and a homomorphism $f\colon (X, \alpha) \to (Y, \beta)$, we have that $f\colon (X, d_\alpha) \to (Y, d_\beta)$ is nonexpansive. If $\overline{\mathcal{B}}$ preserves isometries, then f is an isometry.*

If $\mathcal{B}\colon \mathsf{Set} \to \mathsf{Set}$ admits a final coalgebra $(\nu\mathcal{B}, t)$, then we can define behavioural distance on a coalgebra (X, α) to be the pseudometric space $\mathsf{bd}_\alpha\colon X \times X \to [0,1]$ given by $\mathsf{bd}_\alpha(x,y) = d_t(\mathsf{beh}_\alpha(x), \mathsf{beh}_\alpha(y))$ for all $x, y \in X$. Behavioural distances satisfy several desirable properties:

Lemma 2.2.3. *Let* $\overline{\mathcal{B}} \colon \mathsf{PMet} \to \mathsf{PMet}$ *be a lifting of a functor* $\mathcal{B} \colon \mathsf{Set} \to \mathsf{Set}$ *that admits a final coalgebra* $(\nu\mathcal{B}, t)$*. Given a coalgebra* (X, α) *and* $x, y \in X$*, the following facts hold:*

1. $x \sim_{\mathsf{b}} y \implies \mathsf{bd}_\alpha(x,y) = 0$

2. *If* $\overline{\mathcal{B}}$ *preserves metrics and* $\mathcal{B}$ *is finitary, then* $\mathsf{bd}_\alpha(x,y) = 0 \implies x \sim_{\mathsf{b}} y$

3. *If* $\overline{\mathcal{B}}$ *preserves isometries, then* $d_\alpha(x,y) = \mathsf{bd}(x,y)$

Proof. ① follows from [Bal+18, Lemma 6.6]. For ②, we have that if $\mathcal{B}$ is finitary, then $(\nu\mathcal{B}, t)$ can be obtained via the Adamek fixpoint theorem [AK95] and hence one can apply [Bal+18, Theorem 6.10]. Finally, ③ follows from [Bal+18, Theorem 6.7]. □

2.2.2 Behavioural distance of deterministic automata via functor lifting

It turns out that shortest-distinguishing-word metric (Equation (2.1)) can be obtained as an instance of the coalgebraic framework of behavioural distances [Bal+18, Example 6.5] using an appropriate lifting of the functor $\mathcal{H} = \{0,1\} \times (-)^A$ describing one-step behaviour of finite automata [Bal+18, Example 6.3]. That lifting is defined as follows; let $d \colon M \times M \to [0,1]$ be a pseudometric and let $\lambda \in]0,1[$ be a fixed *discount factor*. We can equip the set $\mathcal{H}X$ with a distance function given by

$$d^{\mathcal{H}}(\langle o_1, g_1\rangle, \langle o_2, g_2\rangle) = \max\{d_{\{0,1\}}(o_1,o_2), \lambda \cdot \max_{a\in A} d(g_1(a), g_2(a))\}$$

for all $\langle o_1, g_1\rangle, \langle o_2, g_2\rangle \in \mathcal{H}X$. The definition above involves $d_{\{0,1\}}$, the discrete metric on the set $\{0,1\}$. Intuitively, two one-step behaviours $\langle o_1, g_1\rangle, \langle o_2, g_2\rangle \in \{0,1\} \times M^A$ of a deterministic automaton with the set of states M are maximally apart if $o_1 \neq o_2$, that is, they disagree in their output behaviour. Otherwise, the distance is equal to a maximal distance $d(g_1(a), g_2(a))$ between reachable states for all letters $a \in A$ of the alphabet, discounted by the factor of λ.

The lifting defined above is particularly well-behaved, as it satisfies the following:

Proposition 2.2.4. *$d^{\mathcal{H}}$ preserves isometries and metrics.*

Proof. Preservation of isometries follows from [Bal+18, Theorem 5.23] and preservation of metrics follows from [Bal+18, Theorem 5.24]. □

Combining the statement above with Lemma 2.2.2 yields that for any deterministic automaton $\mathcal{M} := (M, \langle o_M, t_M \rangle)$, its behavioural distance $\mathsf{bd}_{\langle o_M, t_M \rangle}$ is a pseudometric space, whose values can be calculated as the least fixpoint of the monotone map $\Phi_{\langle o_M, t_M \rangle} : D_M \to D_M$ defined as

$$\Phi_{\langle o_M, t_M \rangle}(m_1, m_2) = d^{\mathcal{H}}(\langle o_M(m_1), t_M(m_1) \rangle, \langle o_M(m_2), t_M(m_2) \rangle)$$

for all $m_1, m_2 \in M$.

Moreover, because of the preservation of metrics and Lemma 2.2.3, we know that for the final deterministic automaton $\mathcal{L} = (\mathcal{P}(A^*), \langle o_L, t_L \rangle)$, the least fixpoint of $\Phi_{\langle o_L, t_L \rangle}$ is a metric space. This metric enjoys the following concrete characterisation:

Proposition 2.2.5 ([Bal+18, Example 6.5])**.** *For the final deterministic automaton $\mathcal{L} = (\mathcal{P}(A^*), \langle o_L, t_L \rangle)$, the least fixpoint of $\Phi_{\langle o_L, t_L \rangle}$ coincides with shortest-distinguishing-word metric.*

2.3 Quantitative Axiomatisation

In order to provide a quantitative inference system for reasoning about the behavioural distance of languages denoted by regular expressions, we first recall the definition of quantitative equational theories from the existing literature [MPP16; Bac+18a] following the notational conventions from [Bac+18a]. We then present our axiomatisation and demonstrate its soundness. The interesting thing about our axiomatisation is the lack of any fixpoint introduction rule. We show that in the case of quantitative analogue of equational logic [MPP16] containing the infinitary rule capturing the notion of convergence, we can use our axioms to derive Salomaa's fixpoint rule from his axiomatisation of language equivalence of regular expressions [Sal66].

2.3.1 Quantitative equational theories

Let Σ be an algebraic signature (in the sense of universal algebra [BS81]) consisting of operation symbols $f_n \in \Sigma$ of arity $n \in \mathbb{N}$. If we write X for the countable set of *metavariables*, then $\mathbb{T}(\Sigma, X)$ denotes a set of freely generated terms over X built from the signature Σ. As a notational convention, we will use letters $t, s, u, \ldots \in \mathbb{T}(\Sigma, X)$ to denote terms. By a *substitution* we mean a function of the type $\sigma \colon X \to \mathbb{T}(\Sigma, X)$ allowing to replace metavariables with terms. Each substitution can be inductively extended to terms in a unique way by setting $\sigma(f(t_1, \ldots, t_n)) = f(\sigma(t_1), \ldots, \sigma(t_n))$ for each operation symbol $f_n \in \Sigma$ from the signature. We will write $\mathcal{S}(\Sigma)$ for the set of all substitutions. Given two terms $t, s \in \mathbb{T}(\Sigma, X)$ and a nonnegative rational number $r \in \mathbb{Q}$ denoting the distance between the terms, we call $t \equiv_r s$ a *quantitative equation (of type* Σ*)*. Notation-wise, we will write $\mathcal{E}(\Sigma)$ to denote the set of all quantitative equations (of type Σ) and we will use the capital Greek letters $\Gamma, \Theta, \ldots \subseteq \mathcal{E}(\Sigma)$ to denote the subsets of $\mathcal{E}(\Sigma)$. By a *deducibility relation* we mean a binary relation denoted $\vdash \subseteq \mathcal{P}(\mathcal{E}(\Sigma)) \times \mathcal{E}(\Sigma)$. Similarly, to the classical equational logic, we will use the following notational shorthands:

$$\Gamma \vdash t \equiv_r s \iff (\Gamma, t \equiv_r s) \in \vdash \qquad \text{and} \qquad \vdash t \equiv_r s \iff \emptyset \vdash t \equiv_r s$$

Furthermore, following the usual notational conventions, we will write $\Gamma \vdash \Theta$ as a shorthand for the situation when $\Gamma \vdash t \equiv_r s$ holds for all $t \equiv_r s \in \Theta$. To call $\vdash$ a *quantitative deduction system (of type* Σ*)* it needs to satisfy the following rules of inference:

$$\begin{aligned}
&(\mathsf{Top}) && \vdash t \equiv_1 t\,, \\
&(\mathsf{Refl}) && \vdash t \equiv_0 t\,, \\
&(\mathsf{Symm}) && \{t \equiv_r s\} \vdash s \equiv_r t\,, \\
&(\mathsf{Triang}) && \{t \equiv_r u, u \equiv_{r'} s\} \vdash t \equiv_{r+r'} s\,, \\
&(\mathsf{Max}) && \{t \equiv_r s\} \vdash t \equiv_{r+r'} s\,, \text{ for all } r' > 0\,, \\
&(\mathsf{Cont}) && \{t \equiv_{r'} s \mid r' > r\} \vdash t \equiv_r s\,,
\end{aligned}$$

(NExp) $\{t_1 \equiv_r s_1, \dots, t_n \equiv_r s_n\} \vdash f(t_1, \dots, t_n) \equiv_r f(s_1, \dots, s_n)$, for all $f_n \in \Sigma$,

(Subst) If $\Gamma \vdash t \equiv_r s$, then $\sigma(\Gamma) \vdash \sigma(t) \equiv_r \sigma(s)$, for all $\sigma \in \mathcal{S}(\Sigma)$,

(Cut) If $\Gamma \vdash \Theta$ and $\Theta \vdash t \equiv_r s$, then $\Gamma \vdash t \equiv_r s$,

(Assum) If $t \equiv_r s \in \Gamma$, then $\Gamma \vdash t \equiv_r s$.

where $\sigma(\Gamma) = \{\sigma(t) \equiv_r \sigma(s) \mid t \equiv_r s \in \Gamma\}$. Finally, by a *quantitative equational theory* we mean a set $\mathcal{U}$ of universally quantified *quantitative inferences* $\{t_1 \equiv_{r_1} s_1, \dots, t_n \equiv_{r_n} s_n\} \vdash t \equiv_r s$, with *finitely many premises*, closed under $\vdash$-derivability.

2.3.2 Quantitative algebras

Quantitative equational theories lie on the syntactic part of the picture. On the semantic side, we have their models called *quantitative algebras*, defined as follows.

Definition 2.3.1 ([MPP16, Definition 3.1])**.** A quantitative algebra is a tuple $\mathcal{A} = (A, \Sigma^{\mathcal{A}}, d^{\mathcal{A}})$, such that $(A, \Sigma^{\mathcal{A}})$ is an algebra for the signature Σ and $(A, d^{\mathcal{A}})$ is a pseudometric such that for all operation symbols $f_n \in \Sigma$, for all $1 \leq i \leq n$, $a_i, b_i \in A$, $d^{\mathcal{A}}(a_i, b_i) \leq r$ implies $d^{\mathcal{A}}(f^{\mathcal{A}}(a_1, \dots, a_n), f^{\mathcal{A}}(b_1, \dots, b_n)) \leq r$.

Consider a quantitative algebra $\mathcal{A} = (A, \Sigma^{\mathcal{A}}, d^{\mathcal{A}})$. Given an assignment $\iota \colon X \to A$ of meta-variables from X to elements of carrier A, one can inductively extend it to Σ-terms $t \in \mathbb{T}(\Sigma, X)$ in a unique way. We will abuse the notation and just write $\iota(t)$ for the interpretation of the term t in quantitative algebra $\mathcal{A}$. We will say that $\mathcal{A}$ *satisfies* the quantitative inference $\Gamma \vdash t \equiv_r s$, written $\Gamma \models_{\mathcal{A}} t \equiv_r s$, if for any assignment of the meta-variables $\iota \colon X \to A$ it is the case that for all $t' \equiv_{r'} s' \in \Gamma$ we have that $d^{\mathcal{A}}(\iota(t'), \iota(s')) \leq r'$ implies $d^{\mathcal{A}}(\iota(t), \iota(s)) \leq r$. Finally, we say that a quantitative algebra $\mathcal{A}$ *satisfies* (or is a *model* of) the quantitative theory $\mathcal{U}$, if whenever $\Gamma \vdash t \equiv_r s \in \mathcal{U}$, then $\Gamma \models_{\mathcal{A}} t \equiv_r s$.

2.3.3 Quantitative algebra of regular expressions

From now on, let's focus on the signature $\Sigma^{\mathcal{B}} = \{0_0, 1_0, +_2, ;_2, (-)^*{}_1\} \cup \{a_0 \mid a \in A\}$, where A is a finite alphabet. This signature consists of all operations of regular expressions. We can easily interpret all those operations in the set RExp of all regular expressions, using trivial interpretation functions eg. $+^{\mathcal{B}}(e,f) = e+f$, which interpret the operations by simply constructing the appropriate terms. Formally speaking, we can do this because the set RExp is the carrier of initial algebra [BS81] (free algebra over the empty set of generators) for the signature Σ.

To make this algebra into a quantitative algebra, we first equip the set RExp with a pseudometric, given by

$$d^{\mathcal{B}}(e,f) = d_{\mathcal{P}(A^*)}(\llbracket e \rrbracket, \llbracket f \rrbracket) \qquad \text{for all } e,f \in \mathsf{RExp} \tag{2.2}$$

Recall that $d_{\mathcal{P}(A^*)}$ used in the definition above is a behavioural pseudometric on the final deterministic automaton carried by the set $\mathcal{P}(A^*)$ of all formal languages over an alphabet A. In other words, we define the distance between arbitrary expressions e and f to be the distance between formal languages $\llbracket e \rrbracket$ and $\llbracket f \rrbracket$ calculated through the shortest-distinguishing-word metric. It turns out, that in such a situation all the interpretation functions of Σ-algebra structure on RExp are nonexpansive with respect to the pseudometric defined above. In other words, we have that:

Lemma 2.3.2. $\mathcal{B} = (\mathsf{RExp}, \Sigma^{\mathcal{B}}, d^{\mathcal{B}})$ *is a quantitative algebra.*

Proof. Since $d_{\mathcal{P}(A^*)}$ is a pseudometric, then so is $d^{\mathcal{B}} = d_{\mathcal{P}(A^*)} \circ (\llbracket - \rrbracket \times \llbracket - \rrbracket)$. We now verify the nonexpansivity of interpretations of operations with non-zero arity. Let $e,f,g,h \in \mathsf{RExp}$, $d^{\mathcal{B}}(e,g) \leq r$ and $d^{\mathcal{B}}(f,h) \leq r$.

1. We show that $d^{\mathcal{B}}(e+f, g+h) \leq r$. In the case when $r = 0$, the proof simplifies to showing that if $\llbracket e \rrbracket = \llbracket g \rrbracket$ and $\llbracket f \rrbracket = \llbracket h \rrbracket$ then $\llbracket e+g \rrbracket = \llbracket g+h \rrbracket$, which holds immediately. For the remaining case, when $r > 0$, let $n = \lceil \log_{\lambda} r \rceil$.

   Observe that in such a case, we have that $d^{\mathcal{B}}(e,g) \leq \lambda^n$ and $d^{\mathcal{B}}(f,h) \leq \lambda^n$. Using it, we can deduce that $\llbracket e \rrbracket$ and $\llbracket g \rrbracket$ (and similarly $\llbracket f \rrbracket$ and $\llbracket h \rrbracket$) agree on all words of length strictly below n (because the shortest word for which they disagree is at least of length n). To put that formally:

$$\forall w \in A^*. \, |w| < n \implies (w \in \llbracket e \rrbracket \iff w \in \llbracket g \rrbracket) \wedge (w \in \llbracket f \rrbracket \iff w \in \llbracket h \rrbracket)$$

Let $w \in A^*$, such that $|w| < n$. We have that

$$\begin{aligned} w \in [\![e+f]\!] &\iff w \in [\![e]\!] \cup [\![f]\!] \iff (w \in [\![e]\!]) \vee (w \in [\![f]\!]) \\ &\iff (w \in [\![g]\!]) \vee (w \in [\![h]\!]) \qquad (|w| < n) \\ &\iff w \in [\![g+h]\!] \end{aligned}$$

And thus $[\![e+f]\!]$ and $[\![g+h]\!]$ agree on all words of the length below n and therefore $d^{\mathcal{B}}(e+f, g+h) \leq \lambda^n \leq r$.

2. The case for $r = 0$ holds immediately through the same line of reasoning as before, relying on well-definedness of $\diamond$ (concatenation) operation on formal languages. We focus on the remaining case, making the same simplification as before, that is we assume that $[\![e]\!]$ and $[\![g]\!]$ (as well as $[\![f]\!]$ and $[\![h]\!]$) agree on all word of length strictly below n). We show that $[\![e\,; f]\!]$ and $[\![g\,; h]\!]$ also agree on all words of the length strictly less than n. Let $w \in A^*$, such that $|w| < n$. We have that:

$$\begin{aligned} w \in [\![e\,; f]\!] &\iff w \in [\![e]\!] \diamond [\![f]\!] \\ &\iff (\exists u, v \in A^*. w = uv \wedge w \in [\![e]\!] \wedge v \in [\![f]\!]) \\ &\iff (\exists u, v \in A^*. w = uv \wedge w \in [\![g]\!] \wedge v \in [\![h]\!]) \\ &\qquad\qquad (\ |u| < n \text{ and } |v| < n) \\ &\iff w \in [\![g]\!] \diamond [\![h]\!] \iff w \in [\![g\,; h]\!] \end{aligned}$$

3. We use the same line of reasoning as before. Assume that $[\![e]\!]$ and $[\![g]\!]$ agree on all words of length below n. Let $w \in A^*$, such that $|w| < n$. We have the following:

$$\begin{aligned} w \in [\![e^*]\!] &\iff w \in [\![e]\!]^* \\ &\iff w = \varepsilon \vee (\exists k \geq 1. \exists u_1, \dots, u_k \in A^*. w = u_1 \dots u_k \\ &\qquad \wedge u_1 \in [\![e]\!] \wedge \cdots \wedge u_k \in [\![e]\!]) \end{aligned}$$

$$\begin{aligned}
&\iff w = \varepsilon \vee (\exists k \geq 1 . \exists u_1, \dots, u_k \in A^* . w = u_1 \dots u_k \\
&\qquad \wedge u_1 \in [\![g]\!] \wedge \dots \wedge u_k \in [\![g]\!]) \qquad (|u_1| < n, \dots, |u_k| < n) \\
&\iff w \in [\![g]\!]^* \iff w \in [\![g^*]\!]
\end{aligned}$$

□

In order to talk about the quantitative algebra $\mathcal{B}$ of the behavioural distance of regular expressions in an axiomatic way, we introduce the quantitative equational theory REG (Figure 2.2).

	Nondeterministic choice		**Sequential composition**
(SL1)	$\vdash e + e \equiv_0 e\,,$	(1S)	$\vdash 1 ; e \equiv_0 e\,,$
(SL2)	$\vdash e + f \equiv_0 f + e\,,$	(S)	$\vdash e ; (f ; g) \equiv_0 (e ; f) ; g\,,$
(SL3)	$\vdash (e + f) + g \equiv_0 e + (f + g)\,,$	(S1)	$\vdash e ; 1 \equiv_0 e\,,$
(SL4)	$\vdash e + 0 \equiv_0 e\,,$	(0S)	$\vdash 0 ; e \equiv_0 0\,,$
(SL5)	$\{e \equiv_r g, f \equiv_{r'} h\}$	(S0)	$\vdash e ; 0 \equiv_0 0\,,$
	$\vdash e + f \equiv_{\max(r,r')} g + h\,,$	(D1)	$\vdash e ; (f + g) \equiv_0 e ; f + e ; g\,,$
		(D2)	$\vdash (e + f) ; g \equiv_0 e ; g + f ; g\,,$
	Loops		**Behavioural pseudometric**
(Unroll)	$\vdash e^* \equiv_0 e ; e^* + 1\,,$	(λ-Pref)	$\{e \equiv_r f\} \vdash a ; e \equiv_{r'} a ; f\,,$
(Tight)	$\vdash (e + 1)^* \equiv_0 e^*\,,$		for $r' \geq \lambda \cdot r$

Figure 2.2: Axioms of the quantitative equational theory REG for $e, f, g \in$ RExp and $a \in A$.

The first group of axioms capture properties of the nondeterministic choice operator $+$ (SL1-SL5). The first four axioms (SL1-SL4) are the usual laws of semilattices with bottom element 0. (SL5) is a quantitative axiom allowing one to reason about distances between sums of expressions in terms of distances between expressions being summed. Moreover, (SL1-SL5) are axioms of so-called *Quantitative Semilattices with zero*, which have been shown to axiomatise the Hausdorff metric [MPP16].

The sequencing axioms (1S), (S1), (S) state that the set RExp of regular expressions has the structure of a monoid (with neutral element 1) with absorbent element 0 (0S), (S0). Additionally, (D1-D2) talk about interaction of the nondeterministic choice operator $+$ with sequential composition.

The loop axioms (Unroll) and (Tight) are directly inherited from Salomaa's axiomatisation of language equivalence of regular expressions [Sal66]. (Unroll) axiom associates loops with their intuitive behaviour of choosing, at each step, between successful termination and executing the loop body once. (Tight) states that the loop whose body might instantly terminate, causing the next loop iteration to be executed immediately is provably equivalent to a different loop, whose body does not contain immediate termination. Finally, (λ-Pref) captures the fact that prepending the same letter to arbitrary expressions shrinks the distance between them by the factor of $\lambda \in]0,1[$ (used in the definition of $d^{\mathcal{B}}$). This axiom is adapted from the axiomatisation of discounted probabilistic bisimilarity distance [Bac+18a]. Through a simple induction on the length of derivation, one can verify that indeed $\mathcal{B}$ is a model of the quantitative theory REG.

Theorem 2.3.3 (Soundness). *The quantitative algebra* $\mathcal{B} = (\mathsf{RExp}, \Sigma^{\mathcal{B}}, d^{\mathcal{B}})$ *is a model of the quantitative theory* REG. *In other words, for any* $e, f \in \mathsf{RExp}$ *and* $r \in \mathbb{Q}$, *if* $\Gamma \vdash e \equiv_r f \in \mathsf{REG}$, *then* $\Gamma \models_{\mathcal{B}} e \equiv_r f$

Proof. By the structural induction on the judgement $\Gamma \vdash e \equiv_r f \in \mathsf{REG}$. (Subst), (Cut) and (Assum) deduction rules from classical logic hold immediately. The soundness of (Top), (Refl), (Symm), (Triang), (Cont) and (Max) follows from the fact that $d^{\mathcal{B}}$ is a pseudometric. (NExp) follows from the fact that interpretations of symbols from the algebraic signature are nonexpansive (lemma 2.3.2). Recall that $d^{\mathcal{B}} = d_{\mathcal{P}(A^*)} \circ (\llbracket - \rrbracket \times \llbracket - \rrbracket)$.

Additionally, for all axioms in the form $\vdash e \equiv_0 f$ it suffices to show that $\llbracket e \rrbracket = \llbracket f \rrbracket$. (SL1-SL4), (1S), (S), (S1), (0S), (S0), (D1-D2), (Unroll) and (Tight) are taken from Salomaa's axiomatisation of language equivalence of regular expressions [Sal66] and thus both sides of those equations denote the same formal languages [Wag+19, Theorem 5.2]. For (λ-Pref) assume that the premise is satisfied in the model, that is $d_{\mathcal{P}(A^*)}(\llbracket e \rrbracket, \llbracket f \rrbracket) \leq r$. Let $r' \geq \lambda \cdot r$. We show the following:

$$\begin{aligned}
d^{\mathcal{B}}(a;e,a;f) &= d_{\mathcal{P}(A^*)}(\llbracket a;e \rrbracket, \llbracket a;f \rrbracket) && \text{(Equation (2.2))} \\
&= \Phi_{\langle o_L, t_L \rangle}(d_{\mathcal{P}(A^*)})(\llbracket a;e \rrbracket, \llbracket a;f \rrbracket) && (d_{\mathcal{P}(A^*)} \text{ is a fixpoint of } \Phi_{\langle o_L, t_L \rangle}) \\
&= \max\{d_{\{0,1\}}(o_L(a;e), o_L(a;e'))\lambda \cdot \max_{a' \in A} d_{\mathcal{P}(A^*)}(\llbracket a;e \rrbracket_{a'}, \llbracket a;f \rrbracket_{a'})\}
\end{aligned}$$

$$
\begin{aligned}
&= \lambda \cdot d_{\mathcal{P}(A^*)}(\llbracket e \rrbracket, \llbracket f \rrbracket) && \text{(Def. of final automaton)}\\
&\leq \lambda \cdot r \leq r'
\end{aligned}
$$

Finally, $(\mathsf{SL5})$ is derivable from other axioms; we included $(\mathsf{SL5})$ as an axiom to highlight the similarity of our inference system with axiomatisations of language equivalence of regular expressions [Sal66; Koz94] containing the axioms of semilattices with bottom. In the previous work [MPP16], $(\mathsf{SL1}\text{-}\mathsf{SL5})$ are precisely the axioms of *quantitative semilattices with zero* axiomatising the Hausdorff distance. If $r = \max(r, r')$ then $\{e \equiv_r g\} \vdash e \equiv_{\max(r,r')} g$ holds by (Assum). If $r < \max(r, r')$, then we can derive the quantitative judgement above using (Max). By a similar line of reasoning, we can show that $\{f \equiv_{r'} h\} \vdash f \equiv_{\max(r,r')} h$. Finally, using (Cut) and (NExp), we can show that $\{e \equiv_r g, f \equiv_{r'} h\} \vdash e + f \equiv_{\max(r,r')} g + h$ as desired. □

We now revisit the example from Figure 2.1. Recall that states marked as initial of the left and middle automata can be respectively represented as a^* and $a+1$. The shortest word distinguishing languages representing those expressions is aa. If we fix $\lambda = \frac{1}{2}$, then $d^{\mathcal{B}}(a^*, a+1) = d_{\mathcal{P}(A^*)}(\llbracket a^* \rrbracket, \llbracket a+1 \rrbracket) = \frac{1}{4} = \left(\frac{1}{2}\right)^{|aa|}$. We can derive this distance through the means of axiomatic reasoning using the quantitative equational theory REG in the following way:

Example 2.3.4.

$$
\begin{aligned}
&\vdash a^* \equiv_1 0 && (\mathsf{Top})\\
&\vdash a ; a^* \equiv_{\frac{1}{2}} a ; 0 && (\lambda\text{-}\mathsf{Pref})\\
&\vdash a ; a^* + 1 \equiv_{\frac{1}{2}} a ; 0 + 1 && (\vdash 1 \equiv_0 1 \text{ and } \mathsf{SL5})\\
&\vdash a^* \equiv_{\frac{1}{2}} 1 && (\mathsf{Triang, Unroll, S0 \text{ and } SL4})\\
&\vdash a ; a^* \equiv_{\frac{1}{4}} a ; 1 && (\lambda\text{-}\mathsf{Pref})\\
&\vdash a ; a^* + 1 \equiv_{\frac{1}{4}} a ; 1 + 1 && (\vdash 1 \equiv_0 1 \text{ and } \mathsf{SL5})\\
&\vdash a^* \equiv_{\frac{1}{4}} a + 1 && (\mathsf{Triang, Unroll \text{ and } S1})
\end{aligned}
$$

2.3.4 The lack of the fixpoint axiom

Traditionally, completeness of inference systems for behavioural equivalence of languages of expressions featuring recursive constructs such as Kleene star or μ-recursion [Mil84] rely crucially on fixpoint introduction rules. Those allow showing that an expression is provably equivalent to a looping construct if it exhibits some form of self-similarity, typically subject to productivity constraints. As an illustration, Salomaa's axiomatisation of language equivalence of regular expressions incorporates the following inference rule:

$$\frac{g \equiv e \,;g + f \qquad \varepsilon \notin [\![e]\!]}{g \equiv e^* \,; f} \tag{2.3}$$

The side condition on the right states that the loop body is *productive*, that is a deterministic automaton corresponding to an expression e cannot immediately reach acceptance without performing any transitions. This is simply equivalent to the language $[\![e]\!]$ not containing the empty word. It would be reasonable for one to expect REG to contain a similar rule to be complete, especially since it should be able to prove language equivalence of regular expressions (by proving that they are in distance zero from each other). Furthermore, all axioms of Salomaa except Equation (2.3) are contained in REG as rules for distance zero.

It turns out that in the presence of the infinitary continuity (Cont) rule of quantitative deduction systems and the $(\lambda\text{-}\mathsf{Pref})$ rule of REG, the Salomaa's inference rule (Equation (2.3)) becomes a derivable fact for distance zero. First of all, one can show that $(\lambda\text{-}\mathsf{Pref})$ can be generalised from prepending single letters to prepending any regular expression satisfying the side condition from Equation (2.3).

Lemma 2.3.5. *Let $e, f, g \in \mathsf{RExp}$, such that $\varepsilon \notin [\![e]\!]$. Then, $\{f \equiv_r g\} \vdash e\,; f \equiv_{r'} e\,; g$ is derivable using the axioms of REG for all $r' \geq \lambda \cdot r$.*

Proof. By induction on $e \in \mathsf{RExp}$. The cases when $e = 1$ and $e = (e_1)^*$ are not possible, because of the assumption that $\varepsilon \notin [\![e]\!]$.

$\boxed{e = 0}$ Because of the $(0S)$ axiom, we can derive that $e\,;f \equiv_0 0 \equiv_0 0\,;g \equiv_0 e\,;g$. We can show the desired conclusion, using (Max) axiom.

$\boxed{e = a}$ Holds immediately, because of $(\lambda\text{-}\mathsf{Pref})$ axiom.

$\boxed{e = e_1 + e_2}$ Because of the assumption, both $\varepsilon \notin [\![e_1]\!]$ and $\varepsilon \notin [\![e_2]\!]$. Using the induction hypothesis, we can derive that $\vdash e_1\,;f \equiv_{r'} e_1\,;g$ and $e_2\,;f \equiv_{r'} e_2\,;g$. We can apply the $(\mathsf{SL5})$ axiom to derive that $\vdash e_1\,;f + e_2\,;f \equiv_{r'} e_1\,;g + s_2\,;g$. Finally, we can apply the $(\mathsf{D2})$ axiom to both sides through (Triang) and derive $\vdash (e_1 + e_2)\,;f \equiv_{r'} (e_1 + e_2)\,;g$ as desired.

$\boxed{e = e_1\,;e_2}$ Because of the assumption, $\varepsilon \notin [\![e_1]\!]$ or $\varepsilon \notin [\![e_2]\!]$. First, let's consider the subcase when both $\varepsilon \notin [\![e_1]\!]$ and $\varepsilon \notin [\![e_2]\!]$. By induction hypothesis, we have that $\vdash e_2\,;f \equiv_{r'} e_2\,;g$. Since $\lambda \in \,]0,1[$, we have that $\lambda \cdot r' < r'$. Because of that, we can apply induction hypothesis again and obtain $\vdash e_1\,;e_2\,;f \equiv_{r'} e_1\,;e_2\,;g$. Now, let's consider the subcase when $\varepsilon \notin [\![e_1]\!]$, but $\varepsilon \in [\![e_2]\!]$. Using (NExp), we can obtain $\vdash e_2\,;f \equiv_r e_2\,;g$. Then, since $\varepsilon \notin [\![e_1]\!]$, we can apply the induction hypothesis and obtain $\vdash e_1\,;e_2\,;f \equiv_{r'} e_1\,;e_2\,;g$ as desired. The remaining subcase, when $\varepsilon \notin [\![e_2]\!]$ but $\varepsilon \in [\![e_1]\!]$ is symmetric and therefore omitted.

□

With the above lemma in hand, one can inductively show that if $g \equiv_0 e\,;g + f$ and $\varepsilon \notin [\![e]\!]$, then g gets arbitrarily close to $e^*\,;f$. Intuitively, the more we unroll the loop in $e^*\,;f$ using (Unroll) and the more we unroll the definition of g, then the closer both expressions become.

Lemma 2.3.6. *Let $e, f, g \in \mathsf{RExp}$, such that $\varepsilon \notin [\![e]\!]$ and let $n \in \mathbb{N}$. Then, $\{g \equiv_0 e\,;g + f\} \vdash g \equiv_r e^*\,;f$ is derivable using the axioms of* REG *for all $r \geq \lambda^n$.*

Proof. By induction. If $n = 0$, then using (Top), we can immediately conclude that $\vdash g \equiv_1 e^*\,;f$. Since by the assumption $r \geq \lambda^0 = 1$, we can apply (Max) and obtain $\vdash g \equiv_r e^*\,;f$.

For the inductive cases, we have that $r \geq \lambda^{n+1}$ and hence $r \cdot \lambda^{-1} \geq \lambda^n$. We cannot instantly apply the induction hypothesis, as $r \cdot \lambda^{-1}$ is not guaranteed to be rational. Instead, we will use (Cont) of quantitative deduction systems. Let r' be an arbitrary rational number strictly greater than r and let $\{r_n\}_{n\in\mathbb{N}}$ be any decreasing sequence of rationals that converges to λ^{-1}. Pick element r_N of that sequence that satisfies that $r' \geq r\cdot\lambda\cdot r_N$. We can always pick such an element, as $\{r_n\}_{n\in\mathbb{N}}$ gets arbitrarily close to λ^{-1}, so $\{\lambda\cdot r_n\}_{n\in\mathbb{N}}$ is a decreasing sequence that converges to 1 and additionally we have that $r' > r$, so $\frac{r'}{r} > 1$. From the definition of the limit, we know that there exists large enough $N\in\mathbb{N}$, such that $|\lambda\cdot r_N - 1| \leq \frac{r'}{r} - 1$. We can simplify the above relying on the fact that $\lambda\cdot r_n \geq 1$ for all $n\in\mathbb{N}$ and obtain that indeed $r' \geq r\cdot\lambda\cdot r_N$ as desired.

Since $r\cdot r_N \geq r\cdot\lambda^{-1} \geq \lambda^n$, we can apply induction hypothesis and obtain that $\vdash e \equiv_{r\cdot r_N} g^* ; f$. Since $\varepsilon \notin [\![e]\!]$, we can now use Lemma 2.3.5 to derive that $e ; g \equiv_{r'} e ; e^* ; f$. Since we have shown it for arbitrary $r' > r$, we can use (Cont) rule of the quantitative deduction systems and conclude that $\vdash e ; g \equiv_r e ; e^* ; f$, as desired.

Then, because of (Refl), we have that $\vdash f \equiv_0 f$. We can combine those two quantitative inferences using $(\mathsf{SL5})$ axiom in order to get $\vdash e ; g + f \equiv_r e ; e^* ; f + f$. By assumption, the left hand side satisfies that $\vdash g \equiv_0 e ; g + f$. Now, consider the right hand side of that quantitative inference:

$$\begin{aligned} \vdash e ; e^* ; f + f &\equiv_0 e ; e^* ; f + 1 ; f && (\mathsf{1S}) \\ &\equiv_0 (e ; e^* + 1) ; f && (\mathsf{D2}) \\ &\equiv_0 e^* ; f && (\mathsf{Unroll}) \end{aligned}$$

We can combine the reasoning above and conclude (using (Triang)) that $\vdash g \equiv_r e^* ; f$. □

Having the result above, we can now use the infinitary (Cont) rule capturing the limiting property of decreasing chain of overapproximations to the distance and show the derivability of Salomaa's inference rule.

Lemma 2.3.7. *Let $e, f, g \in \mathsf{RExp}$, such that $\varepsilon \notin [\![e]\!]$. Then, $\{g \equiv_0 e ; g + f\} \vdash g \equiv_0 e^* ; f$ is derivable using the axioms of REG.*

Proof. To deduce that $\vdash g \equiv_0 e^* ; f$ using (Cont) it suffices to show that $\vdash g \equiv_r e^* ; f$

for all $r > 0$. To do so, pick an arbitrary $r > 0$ and let $N = \lceil \log_\lambda r \rceil$. Observe that $\lambda^N = \lambda^{\lceil \log_\lambda r \rceil} \leq \lambda^{\log_\lambda r} = r$. Because of Lemma 2.3.6 we have that $\vdash g \equiv_r e^* ; f$, which completes the proof. □

2.4 Completeness

We now focus our attention on the central result of this paper, which is the completeness of REG with respect to the shortest-distinguishing-word metric on languages denoting regular expressions. We use the strategy from the proof of completeness of quantitative axiomatisation of probabilistic bisimilarity distance [Bac+18a]. It turns out that the results from [Bac+18a] rely on properties that are not unique to the Kantorovich/Wassertstein lifting and can be also established for instances of the abstract coalgebraic framework [Bal+18].

The heart of our argument relies on the fact that the distance between languages denoting regular expressions can be calculated in a simpler way than applying the Knaster-Tarski fixpoint theorem while looking at the infinite-state final automaton of all formal languages over some fixed alphabet.

In particular, regular expressions denote the behaviour of finite-state deterministic automata. Since automata homomorphisms are nonexpansive mappings, the distance between languages $[\![e]\!]$ and $[\![f]\!]$ of some arbitrary regular expressions $e, f \in \mathsf{RExp}$ is the same as the distance between states in some DFA whose languages corresponds to $[\![e]\!]$ and $[\![f]\!]$. To be precise, we will look at the finite subautomaton $\langle [e]_{\doteqdot}, [f]_{\doteqdot} \rangle_{\mathcal{Q}}$ of the $\doteqdot$ quotient of the Brzozowski automaton. The reason we care about deterministic finite automata is that it turns out that one can calculate the behavioural distance between two states through an iterative approximation from above, which can be also derived axiomatically using the (Cont) rule of quantitative deduction systems. We start by showing how this simplification works, and then we establish completeness.

2.4.1 Behavioural distance of finite-state automata

Consider a deterministic automaton $\mathcal{M} = (M, \langle o_M, t_M \rangle)$. The least fixpoint of a monotone endomap $\Phi_{\langle o_M, t_M \rangle} : D_M \to D_M$ on the complete lattice of pseudometrics on the set M results in $d_{\langle o_M,t_M\rangle}$. It is noteworthy that $\Phi_{\langle o_M,t_M\rangle}$ exhibits two generic properties. Firstly, $\Phi_{\langle o_M,t_M\rangle}$ behaves well within the Banach space structure defined by the supremum norm.

Lemma 2.4.1. *For any deterministic automaton $\mathcal{M} = (M, \langle o_M, t_M\rangle)$, $\Phi_{\langle o_M,t_M\rangle} : D_M \to D_M$ is contractive with respect to the supremum norm. In other words, for all $d, d' \in D_M$ we have that*

$$\|\Phi_{\langle o_M,t_M\rangle}(d') - \Phi_{\langle o_M,t_M\rangle}(d)\| \leq \lambda \cdot \|d' - d\|$$

Proof. We can safely assume that $d \sqsubseteq d'$, as other case will be symmetric. It sufices to show that for all $m, m' \in M$, $\Phi_{\langle o_M,t_M\rangle}(d')(m,m') - \Phi_{\langle o_M,t_M\rangle}(d)(m,m') \leq \|d' - d\|$. First, let's consider the case when $o_M(m) \neq o_M(m')$ and hence $d_{\{0,1\}}(m,m') = 1$. In such a scenario, it holds that

$$\Phi_{\langle o_M,t_M\rangle}(d')(m,m') - \Phi_{\langle o_M,t_M\rangle}(d)(m,m') = 0 \leq \lambda \cdot \|d' - d\|$$

From now on, we will assume that $o_M(m) = o_M(m)$ and hence $d_{\{0,1\}}(m,m') = 0$. We have the following:

$$\begin{aligned}
\Phi_{\langle o_M,t_M\rangle}(d')(m,m') - \Phi_{\langle o_M,t_M\rangle}(d)(m,m') &= \lambda \cdot \max_{a\in A} d'(m_a, m'_a) - \lambda \cdot \max_{a\in A} d(m_a, m'_a) \\
&= \lambda \cdot \left(\max_{a\in A} d'(m_a, m'_a) - \max_{a\in A} d(m_a, m'_a) \right) \\
&\leq \lambda \cdot \left(\max_{a\in A} \{ d'(m_a, m'_a) - d(m_a, m'_a) \} \right) \\
&\leq \lambda \cdot \sup_{n,n'\in M} \{ d'(n,n') - d(n,n') \} \\
&= \lambda \cdot \|d' - d\|
\end{aligned}$$

□

Secondly, contractivity of $\Phi_{\langle o_M,t_M\rangle}$ implies the following:

Corollary 2.4.2. *For any deterministic automaton $\mathcal{M} = (M, \langle o_M, t_M\rangle)$, $\Phi_{\langle o_M,t_M\rangle}$ has a unique fixed point.*

This means that if we want to calculate $d_{\langle o_M, t_M\rangle}$ it suffices to look at any fixpoint of $\Phi_{\langle o_M, t_M\rangle}$. This will enable a simpler characterisation, than the one given by the Knaster-Tarski fixpoint theorem. In particular, we will rely on the characterisation given by the Kleene's fixpoint theorem [San11, Theorem 2.8.5], which allows to obtain the greatest fixpoint of an endofunction on the lattice as the infimum of the decreasing sequence of finer approximations obtained by repeatedly applying the function to the top element of the lattice.

Theorem 2.4.3 (Kleene's fixpoint theorem)**.** *Let $(X, \sqsubseteq)$ be a complete lattice with a top element $\top$ and $f\colon X \to X$ an endofunction that is ω-cocontinuous or in other words for any decreasing chain $\{x_i\}_{i\in\mathbb{N}}$ it holds that*

$$\inf_{i\in\mathbb{N}}\{f(x_i)\} = f\left(\inf_{i\in\mathbb{N}}\{x_i\}\right)$$

Then, f possesses a greatest fixpoint, given by $\operatorname{gfp}(f) = \inf_{i\in\mathbb{N}}\{f^{(i)}(\top)\}$ where $f^{(n)}$ denotes n-fold self-composition of f given inductively by $f^{(0)}(x) = x$ and $f^{(n+1)}(x) = f^{(n+1)}(f(x))$ for all $x \in X$.

The theorem above requires the endomap to be ω-cocontinuous. Luckily, it is the case for $\Phi_{\langle o_M, t_M\rangle}$ if we restrict our attention to DFA. To show that, we directly follow the line of reasoning from [Bac+18a, Lemma 5.6] generalising the similar line of reasoning for ω-continuity from [Bre12, Theorem 1]. First, using Lemma 2.1.19 we show that decreasing chains of pseudometrics over a finite set converge to their infimum. That result is a minor re-adaptation of [Bre12, Theorem 1] implicitly used in [Bac+18a, Lemma 5.6].

Lemma 2.4.4. *Let $\{d_i\}_{i\in\mathbb{N}}$ be an infinite descending chain in the lattice $(D_X, \sqsubseteq)$, where X is a finite set. The sequence $\{d_i\}_{i\in\mathbb{N}}$ converges (in the sense of convergence in the Banach space) to $d(x,y) = \inf_{i\in\mathbb{N}} d_i(x,y)$.*

Proof. Let $r > 0$ and let $x, y \in X$. Since $d(x,y) = \inf_{i\in\mathbb{N}} d_i(x,y)$ there exists an index $m_{x,y} \in \mathbb{N}$ such that for all $n \geq m_{x,y}$, $|d_n(x,y) - d(x,y)| < r$. Now, let $N = \max\{m_{x,y} \mid x,y \in X\}$. This is well-defined because X is finite. Therefore, for all $n \geq N$ and $x,y \in X$, $|d_n(x,y) - d(x,y)| < r$ and hence $\|d_n - d\| < r$. □

We can now use the above to show the desired property, by re-adapting [Bre12, Theorem 1].

Lemma 2.4.5. *If $\mathcal{M} = (M, \langle o_M, t_M \rangle)$ is a deterministic finite automaton, then $\Phi_{\langle o_M, t_M \rangle}$ is ω-cocontinuous.*

Proof. By Lemma 2.4.4, the chain $\{d_i\}_{i \in \mathbb{N}}$ converges to $\inf_{i \in \mathbb{N}} d_i$. Since $\Phi_{\langle o_M, t_M \rangle}$ is contractive (Lemma 2.4.1) it is also continuous (in the sense of the Banach space continuity) and therefore $\{\Phi_{\langle o_M, t_M \rangle}(d_i)\}_{i \in \mathbb{N}}$ converges to $\Phi_{\langle o_M, t_M \rangle}(\inf_{i \in \mathbb{N}} d_i)$. Recall that $\Phi_{\langle o_M, t_M \rangle}$ is monotone, which makes $\{\Phi_{\langle o_M, t_M \rangle}(d_i)\}_{i \in \mathbb{N}}$ into a chain, which by Lemma 2.1.19 and Lemma 2.4.4 converges to $\inf_{i \in \mathbb{N}}\{\Phi_{\langle o_M, t_M \rangle}(d_i)\}$. Since limit points are unique, $\inf_{i \in \mathbb{N}}\{\Phi_{\langle o_M, t_M \rangle}(d_i)\} = \Phi_{\langle o_M, t_M \rangle}(\inf_{i \in \mathbb{N}} d_i)$. □

We can combine the preceding results and provide a straightforward characterisation of the distance between languages represented by arbitrary regular expressions, denoted as $e, f \in \mathsf{RExp}$. Utilising a simple argument based on Lemma 2.2.3, which asserts that automata homomorphisms are isometries, one can demonstrate that the distance between $\llbracket e \rrbracket$ and $\llbracket f \rrbracket$ in the final automaton is equivalent to the distance between $[e]_{\doteqdot}$ and $[f]_{\doteqdot}$ in $\langle [e]_{\doteqdot}, [f]_{\doteqdot} \rangle_{\mathcal{Q}}$. This is, the least subautomaton of $\mathcal{Q}$ that contains the derivatives (modulo $\doteqdot$) reachable from $[e]_{\doteqdot}$ and $[f]_{\doteqdot}$. Importantly, this automaton is finite (Lemma 2.1.14), allowing us to apply the Kleene's fixpoint theorem to calculate the distance.

Let $\Psi^{(0)}_{e,f}$ denote the discrete metric on the set $\langle [e]{\doteqdot}, [f]{\doteqdot} \rangle_{\mathcal{Q}}$ (the top element of the lattice of pseudometrics over that set). Define $\Psi^{(n+1)}_{e,f} = \Phi_{\langle [e]\doteqdot, [f]\doteqdot \rangle_{\mathcal{Q}}}\left(\Psi^{(n)}_{e,f}\right)$. Additionally, leveraging the fact that infima of decreasing chains are calculated pointwise (Lemma 2.1.19), we can conclude with the following:

Lemma 2.4.6. *For all $e, f \in \mathsf{RExp}$, the underlying pseudometric of the quantitative algebra $\mathcal{B}$ can be given by $d^{\mathcal{B}}(e, f) = \inf_{i \in \mathbb{N}}\left\{\Psi^{(i)}_{e,f}([e]_{\doteqdot}, [f]_{\doteqdot})\right\}$*

Proof. Recall that $d^{\mathcal{B}} = d_{\mathcal{P}(A^*)} \circ (\llbracket - \rrbracket \times \llbracket - \rrbracket)$. Moreover, the canonical quotient map $[-]_{\doteqdot} \colon \mathsf{RExp} \to \mathsf{RExp}/{\doteqdot}$ is an automaton homomorphism from $\mathcal{R}$ to $\mathcal{Q}$. Composing it with a language assigning homomorphism $L_{\mathcal{Q}} \colon \mathsf{RExp}/{\doteqdot} \to \mathcal{P}(A^*)$ yields an automaton homomorphism $L_{\mathcal{Q}} \circ [-]_{\doteqdot} \colon \mathsf{RExp} \to \mathcal{P}(A^*)$, which by finality must be the same as $L_{\mathcal{R}} \colon \mathsf{RExp} \to \mathcal{P}(A^*)$, and thus (by Lemma 2.1.12) the same as $[\![-]\!]$. Using the fact that automata homomorphisms are isometries (Lemma 2.2.3), we can derive the following:

$$\begin{aligned} d^{\mathcal{B}} &= d_{\mathcal{P}(A^*)} \circ ([\![-]\!] \times [\![-]\!]) \\ &= d_{\mathcal{P}(A^*)} \circ ((L_{\mathcal{Q}} \circ ([-]_{\doteqdot}) \times (L_{\mathcal{Q}} \circ ([-]_{\doteqdot})) \\ &= d_{\mathcal{P}(A^*)} \circ (L_{\mathcal{Q}} \times L_{\mathcal{Q}}) \circ ([-]_{\doteqdot} \times [-]_{\doteqdot}) \\ &= d_{\langle o_{\mathcal{Q}}, t_{\mathcal{Q}} \rangle} \circ ([-]_{\doteqdot} \times [-]_{\doteqdot}) && \text{(Lemma 2.2.3)} \end{aligned}$$

Additionally, since $\langle [e]_{\doteqdot}, [f]_{\doteqdot} \rangle_{\mathcal{Q}}$ is the subautomaton of $\mathcal{Q}$ containing all the derivatives (modulo $\doteqdot$) of e and f, the canonical inclusion map $\iota \colon \langle [e]_{\doteqdot}, [f]_{\doteqdot} \rangle_{\mathcal{Q}} \hookrightarrow \mathcal{Q}$ is a deterministic automaton homomorphism. Because $\iota([e]_{\doteqdot}) = [e]_{\doteqdot}$ and $\iota([f]_{\doteqdot}) = [f]_{\doteqdot}$, we can again use Lemma 2.2.3 to show that

$$d^{\mathcal{B}}(e,f) = d_{\langle o_{\mathcal{Q}}, t_{\mathcal{Q}} \rangle}([e]_{\doteqdot}, [f]_{\doteqdot}) = d_{\langle [e]_{\doteqdot}, [f]_{\doteqdot} \rangle_{\mathcal{Q}}}([e]_{\doteqdot}, [f]_{\doteqdot})$$

Because of the fact that $([e]_{\doteqdot}, [f]_{\doteqdot})$ has finitely many states (lemma 2.1.14) then by Corollary 2.4.2, Lemma 2.4.5 and Theorem 2.4.3 one can use the simplified iterative formula to calculate the behavioural pseudometric of $\langle [e]_{\doteqdot}, [f]_{\doteqdot} \rangle_{\mathcal{Q}}$. □

In simpler terms, we have demonstrated that the behavioural distance between a pair of arbitrary regular expressions can be calculated as the infimum of decreasing approximations of the actual distance from above.

Alternatively, one could calculate the same distance as the supremum of increasing approximations from below using the Kleene's fixpoint theorem for the least fixpoint. We chose the former approach because our proof of completeness relies on the (Cont) rule of quantitative deduction systems. This rule essentially states that to prove two terms are at a specific distance, we should be able to prove that for all approximations of that distance from above. This allows us to replicate the fixpoint calculation through axiomatic reasoning.

2.4.2 Completeness result

We start by recalling that regular expressions satisfy a certain decomposition property, stating that each expression can be reconstructed from its small-step semantics, up to $\equiv_0$. This property, often referred to as the fundamental theorem of regular expressions (in analogy with the fundamental theorem of calculus and following the terminology of Rutten [Rut00] and Silva [Sil10]) is useful in further steps of the proof of completeness. We will make use of the n-ary generalised sum operator, which is well defined because of (SL1-SL4) axioms of REG.

Theorem 2.4.7. *Fundamental Theorem For any $e \in \mathsf{RExp}$,*

$$\vdash e \equiv_0 \sum_{a \in A} a\,;(e)_a + o_{\mathcal{R}}(e)$$

is derivable using the axioms of REG.

Proof. See [Brz64, Theorem 4.4] or [Sal66, Lemma 4]. □

Let's now say that we are interested in the distance between some expressions $e, f \in \mathsf{RExp}$. As mentioned before, we will rely on $\langle [e]_{\doteq}, [f]_{\doteq} \rangle_{\mathcal{Q}}$, the least subautomaton of the $\doteq$ quotient of the Brzozowski automaton containing states reachable from $[e]_{\doteq}$ and $[f]_{\doteq}$. Recall that by Lemma 2.1.14 its state space is finite. It turns out that the approximations from above (from Lemma 2.4.6) to the distance between any pair of states in that automaton can be derived through the means of axiomatic reasoning.

Lemma 2.4.8. *Let $e, f \in \mathsf{RExp}$ be arbitrary regular expressions and let $[g]_{\doteq}, [h]_{\doteq} \in \langle [e]_{\doteq}, [f]_{\doteq} \rangle_{\mathcal{Q}}$. For all $i \in \mathbb{N}$, and $r \geq \Psi^{(i)}_{e,f}([g]_{\doteq}, [h]_{\doteq})$, one can derive $\vdash g \equiv_r h$ using the axioms of REG.*

Proof. We proceed by induction on i.

For the base case, observe that $\Psi^{(0)}_{e,f}$ is the discrete pseudometric on the set $\langle [e]_{\doteq}, [f]_{\doteq} \rangle_{\mathcal{Q}}$ such that $\Psi^{(0)}_{e,f}([g]_{\doteq}, [h]_{\doteq}) = 0$ if and only if $g \doteq h$, or otherwise $\Psi^{(0)}_{e,f}([g]_{\doteq}, [h]_{\doteq}) = 1$.

In the first case, we immediately have that $g \equiv_0 h$, because $\dot{\equiv}$ is contained in distance zero axioms of REG. In the latter case, we can just use (Top), to show that $g \equiv_1 h$. Then, in both cases, we can apply (Max) to obtain $\vdash g \equiv_r h$, since $r \geq \Psi^{(0)}_{e,f}([g]_{\dot{\equiv}}, [h]_{\dot{\equiv}})$.

For the induction step, let $i = j+1$ and derive the following:

$$
\begin{aligned}
& r \geq \Psi^{(j+1)}_{e,f}([g]_{\dot{\equiv}}, [h]_{\dot{\equiv}}) \iff r \geq \Phi_{\langle [e]_{\dot{\equiv}}, [f]_{\dot{\equiv}} \rangle_{\mathcal{Q}}} \left(\Psi^{(j)}_{e,f}\right)([g]_{\dot{\equiv}}, [h]_{\dot{\equiv}}) \\
& \hfill (\text{Def. of } \Psi^{j+1}_{e,f}) \\
& \iff r \geq \max\left\{ d_{\{0,1\}}(o_{\mathcal{Q}}([g]_{\dot{\equiv}}), o_{\mathcal{Q}}([h]_{\dot{\equiv}})), \lambda \cdot \max_{a \in A} \left\{ \Psi^{(j)}_{e,f}([g]_{\dot{\equiv}_a}, [h]_{\dot{\equiv}_a}) \right\} \right\} \\
& \hfill (\text{Def. of } \Phi) \\
& \iff r \geq \max\left\{ d_{\{0,1\}}\left(o_{\mathcal{R}}(g), o_{\mathcal{R}}(h)\right), \lambda \cdot \max_{a \in A} \left\{ \Phi^{(j)}_{\langle [e]_{\dot{\equiv}}, [f]_{\dot{\equiv}} \rangle_{\mathcal{Q}}}([(g)_a]_{\dot{\equiv}}, [(h)_a]_{\dot{\equiv}}) \right\} \right\} \\
& \hfill (\text{Def. of } \mathcal{Q}) \\
& \iff r \geq d_{\{0,1\}}(o_{\mathcal{R}}(g), o_{\mathcal{R}}(h)) \text{ and for all } a \in A,\ r \cdot \lambda^{-1} \geq \Psi^{(j)}_{e,f}([(g)_a]_{\dot{\equiv}}, [(h)_a]_{\dot{\equiv}})
\end{aligned}
$$

Firstly, since $d_{\{0,1\}}$ is the discrete pseudometric on the set $\{0,1\}$, we can use (Refl) or (Top) depending on whether $o_{\mathcal{R}}(g) = o_{\mathcal{R}}(h)$ and then apply (Max) to derive $\vdash o_{\mathcal{R}}(g) \equiv_r o_{\mathcal{R}}(h)$.

Let $a \in A$. We will show that $\vdash a\,;(g)_a \equiv_r a\,;(h)_a$. Since $r \cdot \lambda^{-1}$ is not guaranteed to be rational, we cannot immediately apply the induction hypothesis. Instead, we rely on (Cont) rule.

First, pick an arbitrary rational r' strictly greater than r and fix $\{r_n\}_{n \in \mathbb{N}}$ to be any decreasing sequence of rationals that converges to λ^{-1}. Let r_N be an element of that sequence such that $r' \geq \lambda \cdot r \cdot r_N$. It is always possible to pick such element because $\{\lambda \cdot r_n\}_{n \in \mathbb{N}}$ is a decreasing sequence that converges to 1 and $r' > r$.

Since $r \cdot r_N \geq r \cdot \lambda^{-1}$ and $r \cdot r_N \in \mathbb{Q}$, we can use the induction hypothesis and derive $\vdash (g)_a \equiv_{r \cdot r_N} (h_a)$. Then, by ($\lambda$-Pref) axiom we have that $\vdash a\,;(g)_a \equiv_{r'} a\,;(h)_a$. Since we have shown it for arbitrary $r' > r$, by (Cont) rule we have that $\vdash a\,;(g)_a \equiv_r a\,;(h)_a$. Using (SL5), we can combine all subexpressions involving the output and transition derivatives into the following:

$$\vdash \sum_{a\in A} a\,;(g)_a + o_{\mathcal{R}}(g) \equiv_r \sum_{a\in A} a\,;(h)_a + o_{\mathcal{R}}(h)$$

Since both sides are normal forms of g and h existing because of Theorem 2.4.7, we can apply (Triang) on both sides and obtain $\vdash g \equiv_r h$ thus completing the proof. □

At this point, we have done all the hard work, and establishing completeness involves a straightforward argument that utilises the (Cont) rule and the lemma above.

Theorem 2.4.9 (Completeness)**.** *For any $e, f \in \mathsf{RExp}$ and $r \in \mathbb{Q}$, if $\models_{\mathcal{B}} e \equiv_r f$, then $\vdash e \equiv_r f \in \mathsf{REG}$*

Proof. Assume that $\models_{\mathcal{B}} e \equiv_r f$, which by the definition of $\models_{\mathcal{B}}$ is equivalent to $d^{\mathcal{B}}(e,f) \leq r$. In order to use (Cont) axiom to derive $\vdash e \equiv_r f$, we need to be able to show $\vdash e \equiv_{r'} f$ for all $r' > r$. Because of iterative characterisation of $d^{\mathcal{B}}$ from Lemma 2.4.6, we have that $\inf_{i\in\mathbb{N}}\{\Psi^{(i)}_{e,f}([e]_{\doteq},[f]_{\doteq})\} < r'$. Since r' is strictly above the infimum of the descending chain of approximants, there exists a point $i \in \mathbb{N}$, such that $r' > \Psi^{(i)}_{e,f}([e]_{\doteq},[f]_{\doteq})$. We can show this by contradiction.

Assume that for all $i \in \mathbb{N}$, $r' \leq \Psi^{(i)}_{e,f}([e]_{\doteq},[f]_{\doteq})$. This would make r' into the lower bound of the chain $\left\{\Psi^{(i)}_{e,f}([e]_{\doteq},[f]_{\doteq})\right\}_{i\in\mathbb{N}}$ and in such a case r' would be less than or equal to the infimum of that chain, which by assumption is less than or equal to r. By transitivity, we could obtain $r' \leq r$. Since $r' > r$, by antisymmetry we could derive that $r' = r$, which would lead to the contradiction.

Using the fact shown above, we can use Lemma 2.4.8 to obtain $\vdash e \equiv_{r'} f \in \mathsf{REG}$ for any $r' > r$, which completes the proof. □

2.5 Discussion

We have presented a sound and complete axiomatisation of the shortest-distinguishing word distance between languages representing regular expressions through a quantitative analogue of equational logic [MPP16]. Outside of the coalgebra community, the shortest-distinguishing word distance and its variants also appear in the model checking [Kwi90] and the automata learning [FHS22] literature.

Early works on axiomatising behavioural distances relied on ad-hoc inference systems. The earliest example of such a system was presented by Larsen, Fahrenberg and Thrane [LFT11], who focused on the directed simulation distance of streams of elements equipped with a metric space structure. Later work of D'Argenio, Gebler and Lee [DGL14] studied systematic axiomatisations of behavioural between processes featuring both probability and nondeterminism definable through the PGSOS rule format. It is important to note that the inference system of D'Argenio et al contained a powerful rule internalising the definition of Kantorovich lifting as an inference rule.

The introduction of quantitative equational theories [MPP16] made more principled approaches possible. Bacci, Mardare, Panangaden and Plotkin [Bac+18c] axiomatised bisimilarity metric of Markov processes [Des+04] and in the later work similarly considered behavioural distance of Mealy machines and Markov decision processes [Bac+24]. Those results crucially hinged on the quantitative generalisations of results from universal algebra, such as the notion of the tensor of algebraic theories. It is worth noting that specification languages used in those axiomatisations did not feature separate primitives for introducing recursion.

An alternative approach was proposed by Bacci, Bacci, Larsen and Mardare [Bac+18a], who used a mild relaxation of quantitative equational theories and gave a sound and complete axiomatisation of bisimulation distance between terms of probabilistic process calculus of Stark and Smolka [SS00] and later adapted their results to a coarser notion of total variation distance between infinite traces [Bac+18b].

We have followed the strategy for proving completeness from [Bac+18a]. The interesting insight about that strategy is that it relies on properties that are not exclusive to distances obtained through the Kantorovich/Wasserstein lifting and can be established for notions of behavioural distance for other kinds of transition systems stemming from the coalgebraic framework. In particular, one needs to show that the monotone map on the lattice of pseudometrics used in defining the distance of finite-state systems is nonexpansive with respect to the sup norm (and hence ω-cocontinuous) and has a unique fixpoint, thus allowing to characterise the behavioural distance as the greatest fixpoint obtained through the Kleene's fixpoint theorem. This point of view allows one to reconstruct the fixpoint calculation in terms of axiomatic manipulation involving the (Cont) rule, eventually leading to completeness.

We have additionally observed that in the presence of the infinitary (Cont) rule and the (λ-Pref) axiom, there is no need for a fixpoint introduction rule, which is commonplace in axiomatisations of language equivalence regular expressions but also other work on process calculi. Inrestingly, the previous work on axiomatising a discounted probabilistic bisimilarity distance from [Bac+18a] includes both (λ-Pref) and the fixpoint introduction rule, but its proof of completeness [Bac+18a, Theorem 6.4] does not involve the fixpoint introduction rule at any point. We are highly confident that in the case of that axiomatisation, the fixpoint introduction rule could be derived from other axioms in a similar fashion to the way we derived Salomaa's rule for introducing the Kleene star [Sal66]. Additionally, we are interested in how much this argument relates to the recent study of fixpoints in quantitative equational theories [MPP21].

In this chapter, we have focused on the simplest and most intuitive instantiation of the coalgebraic framework in the case of deterministic automata, but the natural next step would be to generalise our results to a wider class of transition systems. A good starting point could be to consider coalgebras for *polynomial* endofunctors, in the fashion of the framework of *Kleene Coalgebra* [Sil10]. Alternatively, it would be interesting to look at recent work on a family of process algebras parametric on an equational theory representing the branching constructs [Sch+22] and study its generalisations to quantitative equational theories. A related and interesting avenue for future work are equational axiomatisations of behavioural equivalence of Guarded Kleene Algebra with Tests (GKAT) [Smo+20; Sch+22] and its probabilistic extension (ProbGKAT) [Róż+23], whose completeness results rely on a powerful uniqueness of solutions axiom (UA). The soundness of UA in both cases is shown through an involved argument relying on equipping the transition systems giving the operational semantics with a form of behavioural distance and showing that recursive specifications describing finite-state systems correspond to certain contractive mappings. It may be more sensible, particularly for ProbGKAT to consider quantitative axiomatisations in the first place and give the proofs of completeness through the pattern explored in this chapter.

Chapter 3

A Diagrammatic Approach to Behavioural Distance of Nondeterministic Processes

3.1 Preliminaries

3.1.1 Charts

Fix a set $V = \{v_1, v_2, \dots\}$ of *variables* and Σ of *letters* respectively. A prechart is a triple (Q, E, D), where Q is a set of states, $D \subseteq Q \times \Sigma \times Q$ a finite labelled transition relation and $E \subseteq Q \times V$ is a finite output relation. Precharts can be thought as a generalisation of nondeterministic automata, where instead of acceptance, we deal with the notion of outputs. Moreover, when D and E are clear from the context, we will write $q \xrightarrow{a} q' \iff (q, a, q') \in D$ and $q \rhd v \iff (q, v) \in E$. A chart C is a quadruple (Q, s, D, E), where (Q, D, E) is a prechart and $s \in Q$ is a distinguished start node. We call a chart finite if Q is finite.

Definition 3.1.1 (Strong Bisimulation)**.** Let $C_i = (Q_i, D_i, E_i)$, $i \in \{1, 2\}$ be precharts. A bisimulation between C_1 and C_2 is a relation $R \subseteq Q_1 \times Q_2$, such that ① if $(q_1, q_2) \in R$, then $E(q_1) = E(q_2)$, ② if $(q_1, q_2) \in R$ and $q_1 \xrightarrow{a} q_1'$, then there exists $q_2' \in Q_2$, such that $q_2 \xrightarrow{a} q_2'$ and $(q_1', q_2') \in R$ and symmetrically. If C_1 and C_2 are charts, we say that they are bisimilar (denoted $C_1 \sim C_2$) if there exists a bisimulation between their underlying precharts that relates their start nodes.

Using the above definition, we can also define the following:

Definition 3.1.2 (Prechart homomorphism)**.** Let $C_i = (Q_i, D_i, E_i)$, $i \in \{1,2\}$ be precharts. We call a function $f\colon Q_1 \to Q_2$ a prechart homomorphism if the graph of f, given by $G(f) = \{(q, f(q)) \mid q \in Q_1\}$ is a bisimulation between C_1 and C_2.

In other words, prechart homomorphisms preserve and reflect transitions. Given a chart $C = (Q, s, D, E)$ we say that a variable $v \in V$ is *live* in C if there exists a path of transitions $s \xrightarrow{a_1} \cdots \xrightarrow{a_n} s' \rhd v$ or call it *dead* otherwise. It can be easily observed that bisimulations and homomorphisms preserve the liveness of variables.

Given a prechart (Q, E, D), we can equivalently see it as a pair (Q, β), where β is a combined transition function $Q \to \mathcal{P}_\omega(\Sigma \times Q + V)$, where $\mathcal{P}_\omega$ denotes a finite powerset. Such transition function β takes each state $q \in Q$, to the set $\beta(q) = D(q) \cup E(q)$ of possible successors, that include labelled transitions and variable outputs.

In other words, precharts are coalgebras for the functor $\mathcal{L}\colon \mathsf{Set} \to \mathsf{Set}$, given by $\mathcal{L} = \Sigma \times (-) + V$. Bisimulations and homomorphisms of $\mathcal{L}$-coalgebras are captured concretely by strong bisimulation of precharts and their homomorphisms. Because of this, we will interchangeably use terms prechart and $\mathcal{L}$-coalgebra. Moreover, $\mathcal{L}$ preserves weak pullbacks and hence $\sim$ is an equivalence relation that captures behavioural equivalence of $\mathcal{L}$-coalgebras. For more details of coalgebraic treatment of precharts, we direct an interested reader to [SRS21].

3.1.2 Algebra of regular behaviours

To define charts, Milner proposed a specification language called an *algebra of regular behaviours* (ARB). The syntax of ARB is given by the following:

$$e, f \in \mathsf{MExp} ::= 0 \mid v \in V \mid a.e \mid e + f \mid \mu v.e$$

where $V = \{v_1, v_2, \dots\}$ and Σ be sets of *variables* and *letters* respectively. Given an expression f containing a variable v, we say that v is *free* in f, if it appears outside of the scope of the $\mu v.e$ operator or say that it is *bound* otherwise. Given an expression $e \in \mathsf{MExp}$, we write $\mathsf{fv}(e) \subseteq V$ for the set of its free variables.

Definition 3.1.3 ([Mil84]). Given vectors $\vec{v}$ of binders and $\vec{e}$ of expressions of the same size, we define a syntactic substitution operator $[\vec{e}/\vec{v}]\colon \mathsf{MExp} \to \mathsf{MExp}$ by the following

$$v[\vec{e}/\vec{v}] = \begin{cases} \vec{e}_i & \text{if } v = \vec{v}_i \\ v & \text{otherwise} \end{cases} \qquad (a.e)[\vec{e}/\vec{v}] = a.(e[\vec{e}/\vec{v}])$$

$$(e+f)[\vec{e}/\vec{v}] = e[\vec{e}/\vec{v}] + f[\vec{e}/\vec{v}]$$

$$(\mu w.e)[\vec{e}/\vec{v}] = \begin{cases} \mu w.(e[\vec{e}/\vec{v}]) & \text{if } w \text{ is not in } \vec{v} \text{ nor free in } \vec{e} \\ \mu w.(e[z/w][\vec{e}/\vec{v}]) & \text{otherwise for some } z \text{ not in } \vec{v} \text{ nor free in } \vec{e} \end{cases}$$

We can now define operational semantics of ARB, by equipping its syntax with a prechart structure.

Definition 3.1.4 ([Mil84]). Let $(\mathsf{MExp}, \partial)$ be a prechart whose transition function (called *derivative*) is a least one satisfying the following inference rules

$$\frac{e \xrightarrow{a} e'}{a.e \xrightarrow{a} e'} \qquad \frac{}{v \rhd v} \qquad \frac{e \xrightarrow{a} e'}{e+f \xrightarrow{a} e'} \qquad \frac{f \xrightarrow{a} f'}{e+f \xrightarrow{a} f'}$$

$$\frac{e \rhd v}{e+f \rhd v} \qquad \frac{f \rhd v}{e+f \rhd v} \qquad \frac{e \rhd v \quad v \neq w}{\mu w.e \rhd v} \qquad \frac{e \xrightarrow{a} e'}{\mu v.e \xrightarrow{a} e'[\mu v.e/v]}$$

Remark 3.1.5 ([Sew95]). The last rule (that defines the transition behaviour of the μ recursion operator) can be replaced by the following:

$$\frac{e[\mu v.e/v] \xrightarrow{a} e'}{\mu v.e \xrightarrow{a} e'}$$

Remark 3.1.6 ([Mil84, Proposition 5.4.]). Syntactic substitution can be described operationally using the following rules

$$\frac{e \rhd v \quad f \xrightarrow{a} f'}{e[f/v] \xrightarrow{a} f'} \qquad \frac{e \xrightarrow{a} e'}{e[f/v] \xrightarrow{a} e'[f/v]}$$

$$\frac{e \rhd w \quad w \neq v}{e[f/v] \rhd w} \qquad \frac{e \rhd v \quad f \rhd w}{e[f/v] \rhd w}$$

The syntactic prechart defined above is locally finite.

Lemma 3.1.7 ([Mil84, Proposition 5.1]). *For all $e \in \mathsf{MExp}$, $\langle e \rangle_\partial$ is finite.*

We can use Lemma 2.1.6 and construct a following quotient $\mathcal{L}$-coalgebra.

Lemma 3.1.8. *We can equip $\mathsf{MExp}/\sim$ with a transition function $\overline{\partial}$ given by*

$$\frac{e \xrightarrow{a}_\partial e'}{[e]_\sim \xrightarrow{a}_{\overline{\partial}} [e']_\sim} \qquad \frac{e \rhd_\partial v_i}{[e]_\sim \rhd_{\overline{\partial}} v_i}$$

This map is a unique transition function on $\mathsf{MExp}/\sim$ that makes the quotient map $[-]_\sim \colon \mathsf{MExp} \to \mathsf{MExp}/\sim$ into prechart homomorphism.

We will refer to the elements of the set $\mathsf{MExp}/\sim$ as *regular behaviours*.

It turns out that all operations from the syntax are compositional with respect to bisimilarity, and hence can be unambiguously lifted to the quotient prechart defined above.

Lemma 3.1.9 ([Sew95, Proposition 7]). *$\sim$ is a congruence on MExp with respect to all operations of the algebra of regular behaviours.*

From now on, we will overload the notation and simply write e for the equivalence class $[e]_\sim$. Using the definitions state above, we can show the following technical lemma, that we will use when constructing a semantic category for our string diagrammatic syntax.

Lemma 3.1.10. *For all $e, f_1, \ldots, f_m, g_1, \ldots, g_m \in \mathsf{MExp}$ and vectors $\vec{v} = (v_{i_1}, \ldots, v_{i_m}), \vec{w} = (v_{j_1}, \ldots, v_{j_n})$, such that all free variables of e are contained in $\vec{v}$ and all free variables of $\vec{f}$ are contained in $\vec{w}$, we have that*

$$(e[\vec{f}/\vec{v}])[\vec{g}/\vec{w}] = e[(f_1[\vec{g}/\vec{w}], \ldots, f_m[\vec{g}/\vec{w}])/\vec{v}]$$

Proof. We define a relation $R \subseteq \mathsf{MExp} \times \mathsf{MExp}$, given by the following:

$$\begin{aligned} R = \mathrm{Id} \cup \{ ((e[\vec{f}/\vec{v}])[\vec{g}/\vec{w}], e[(f_1[\vec{g}/\vec{w}], \ldots, f_m[\vec{g}/\vec{w}])/\vec{v}]) \\ \mid e, f_1, \ldots, f_m, g_1, \ldots, g_m \in \mathsf{MExp}, \vec{v} = (v_{i_1}, \ldots, v_{i_m}), \vec{w} = (v_{j_1}, \ldots, v_{j_n}), \\ \mathsf{fv}(e) \subseteq \vec{v}, \mathsf{fv}(\vec{f}) \subseteq \vec{w} \} \end{aligned}$$

We claim that R is a bisimulation. For pairs $(e,e) \in R$, the conditions of bisimulation are immediately satisfied.

For the remaining pairs, assume that $(e[\vec{f}/\vec{v}])[\vec{g}/\vec{w}] \triangleright u$. In such a case at least one of the following is true:

- $e[\vec{f}/\vec{v}] \triangleright u$
- $e[\vec{f}/\vec{v}] \triangleright w_j$ for $w_j \in \vec{w}$ and $g_j \triangleright u$

Since all free variables of e are contained in $\vec{v}$ and all free variables of $\vec{f}$ are contained in $\vec{w}$, we have that all free variables of $e[\vec{f}/\vec{v}]$ are also contained in $\vec{w}$, which makes the first case impossible. Through a similar line of reasoning, we can deduce that $e \triangleright v_i$ for some $v_i \in \vec{v}$ and $f_i \triangleright w_j$. Since $g_j \triangleright u$, we have that $f_i[\vec{g}/\vec{w}] \triangleright u$. Finally, we have that $e[(f_1[\vec{g}/\vec{w}], \ldots, f_n[\vec{g}/\vec{w}]), \vec{v}] \triangleright u$.

Assume that $(e[\vec{f}/\vec{v}])[\vec{g}/\vec{w}] \xrightarrow{a} h$. Then, at least one of the following is true:

- $e[\vec{f}/\vec{v}] \triangleright w_j$ and $g_j \xrightarrow{a} h$.
- $h = h'[\vec{g}/\vec{w}]$, such that $e[\vec{f}/\vec{v}] \xrightarrow{a} h'$

In the first case, through a similar line of reasoning as before, we can conclude that $e \triangleright v_i$ for some $v_i \in \vec{v}$ and $f_i \triangleright w_j$. Hence, $f_i[\vec{g}/\vec{w}] \xrightarrow{a} h$. Finally, we can deduce that $e[(f_1[\vec{g}/\vec{w}], \ldots, f_m[\vec{g}/\vec{w}])/\vec{v}] \xrightarrow{a} h$. Obviously, $(h,h) \in R$.

In the second case, we have that $e[\vec{f}/\vec{v}] \xrightarrow{a} h'$. There are two subcases, that need to be considered

- $e \triangleright v_i$ and $f_i \xrightarrow{a} h'$
- $h' = h''[\vec{f}/\vec{w}]$ and $e \xrightarrow{a} h''$

In the first subcase, we have that $f_i[\vec{g}/\vec{w}] \xrightarrow{a} h'[\vec{g}/\vec{w}]$ and hence

$$e[(f_1[\vec{g}/\vec{w}],\ldots,f_m[\vec{g}/\vec{w}])/\vec{v}] \xrightarrow{a} h'[\vec{g}/\vec{w}]$$

or equivalently $e[(f_1[\vec{g}/\vec{w}],\ldots,f_m[\vec{g}/\vec{w}])/\vec{v}] \xrightarrow{a} h$. As before, of course $(h,h) \in R$.

Finally, moving on to the second subcase, we have that $e \xrightarrow{a} h''$ and hence

$$e[(f_1[\vec{g}/\vec{w}],\ldots,f_m[\vec{g}/\vec{w}])/\vec{v}] \xrightarrow{a} h''[(f_1[\vec{g}/\vec{w}],\ldots,f_m[\vec{g}/\vec{w}])/\vec{v}]$$

Recall that $(e[\vec{f}/\vec{v}])[\vec{g}/\vec{w}] \xrightarrow{a} h$ and $h = (h''[\vec{f}/\vec{v}])[\vec{g}/\vec{v}]$. Both of those reachable expressions are actually in the relation R. The remaining conditions of bisimulation, can be shown via a symmetric argument. □

3.1.3 Behavioural distance of precharts

Given a pseudometric space defined on a state-space of a prechart, we can *lift* it to the set of possible transitions through the following construction.

Definition 3.1.11 (Transitions lifting)**.** Let (X,d) be a pseudometric space. We write $d^{\uparrow}$ for the pseudometric on $\Sigma \times X + V$ defined by $d^{\uparrow}(m,n) = \frac{1}{2}d(x,y)$ if $m = (a,x)$ and $n = (a,y)$, $d^{\uparrow}(m,n) = 0$ if $m = n$ or $d^{\uparrow}(m,n) = 1$ otherwise.

This lifting admits the following:

Lemma 3.1.12. *$(-)^{\uparrow} \colon D_X \to D_{\Sigma\times X+V}$, the lifitng for $\Sigma \times (-) + V$ is contractive with respect to the metric induced by the sup norm. Namely, $\|d^{\uparrow} - d'^{\uparrow}\| \leq \frac{1}{2}\|d - d'\|$*

Proof. For the sake of simplicity, assume that $d' \sqsubseteq d$, and hence $d'^{\uparrow} \sqsubseteq d^{\uparrow}$. It suffices that we show that for all $u,w \in \Sigma \times X + V$, we have that $d^{\uparrow}(u,w) - d'^{\uparrow}(u,w) \leq \frac{1}{2}\|d - d'\|$. Recall that in all cases, except when $u = (a,x)$ and $w = (a,y)$ for some $a \in \Sigma$ and $x,y \in X$, $d^{\uparrow}(u,w) = d'^{\uparrow}(u,w)$ and hence $d^{\uparrow}(u,w) - d'^{\uparrow}(u,w) = 0 \leq \frac{1}{2}\|d - d'\|$. In the remaining case, we have that

$$\begin{aligned} d^{\uparrow}((a,x),(a,y)) - d'^{\uparrow}((a,x),(a,y)) &= \frac{1}{2}d(x,y) - \frac{1}{2}d'(x,y) \\ &\leq \frac{1}{2}\|d' - d\| \end{aligned}$$

which completes the proof. □

Similarly, we can lift distances over X to distances between elements of $\mathcal{P}_\omega(X)$.

Definition 3.1.13 (Hausdorff lifting)**.** Let (X,d) be a pseudometric space. We can equip $\mathcal{P}_\omega(X)$ with a distance function

$$\mathcal{H}(d)(X,Y) = \max\{\sup_{x\in X}\inf_{y\in Y} d(x,y), \sup_{y\in Y}\inf_{x\in X} d(y,x)\}$$

making $(\mathcal{P}_\omega(X), \mathcal{H}(d))$ into a pseudometric.

Moreover, Hausdorff lifting can be equivalently characterised via the notion of *relational couplings*.

Remark 3.1.14 ([Bal+18, Example 5.31]). Let (X,d) be a pseudometric space and let $A,B \in \mathcal{P}_\omega(X)$. Let $\Gamma(A,B)$ denote the set of relational couplings of A and B, namely elements $R \in \mathcal{P}_\omega(A \times B)$, such that $\pi_1(R) = A$ and $\pi_2(R) = B$. The Hausdorff distance between A and B can be alternatively presented as:

$$\mathcal{H}(d)(A,B) = \inf\left\{ \sup_{(x,y)\in R} d(x,y) \mid R \in \Gamma(A,B) \right\}$$

Hausdorff lifting satisfies the following property:

Lemma 3.1.15 ([van12])**.** *Hausdorff lifting $\mathcal{H}\colon D_X \to D_{\mathcal{P}_\omega(X)}$ is nonexpansive with respect to the metric induced by the sup norm. Namely,*

$$\|\mathcal{H}(d) - \mathcal{H}(d')\| \leq \|d - d'\|$$

Given a prechart (Q,β), whose state-space is equipped with a pseudometric d_Q, we can define a new pseudometric $\Phi_\beta(d_Q)$ that calculates the distance between any pair $q_1, q_2 \in Q$ of states, by lifting d_Q to the set $\mathcal{P}_\omega(\Sigma \times Q + V)$ and comparing $\beta(q_1)$ with $\beta(q_2)$, namely $\Phi_\beta(d_Q)(q_1,q_2) = \mathcal{H}\left(d_Q^{\uparrow}\right)(\beta(q_1), \beta(q_2))$. This is used to define the *behavioural distance*.

Theorem 3.1.16. *Let (Q,β) be a prechart. Then, the following properties hold:* ① *$d_Q \mapsto \Phi_\beta(d_Q)$ is a monotone mapping on the lattice D_Q,* ② *Φ_β has a least fixpoint bd_β,* ③ *$x \sim y \implies \mathsf{bd}_\beta(x,y) = 0$ and* ④ *a homomorphism $f\colon Q \to R$ between precharts (Q,β) and (R,γ) is an isometry between (Q,bd_β) and (R,bd_γ).*

Proof. $\mathcal{H}$ and $(-)^\uparrow$ are liftings for the functors $\mathcal{P}_\omega$ and $\Sigma \times (-) + V$ respectively, that preserve isometries [Bal+18, Theorem 5.8]. The rest follows from Lemma 2.2.2 and Lemma 2.2.3 □

The monotone map used to define the behavioural distance satisfies the following:

Lemma 3.1.17. *$\Phi_\beta\colon D_X \to D_X$ is contractive with respect to the metric induced by the sup norm, namely*

$$\|\Phi_\beta(d) - \Phi_\beta(d')\| \leq \frac{1}{2}\|d - d'\|$$

Proof. For the sake of simplicity, assume that $d' \sqsubseteq d$ and hence $\Phi_\beta(d') \sqsubseteq \Phi_\beta(d)$. It suffices to show that for all $x, y \in X$, we have that $\mathcal{H}(d^\uparrow)(\beta(x),\beta(y)) - \mathcal{H}(d'^\uparrow)(\beta(x),\beta(y)) \leq \frac{1}{2}\|d - d'\|$. We can combine the previous results and for arbitrary $x, y \in X$ obtain the following

$$\begin{aligned}
\mathcal{H}(d^\uparrow)(\beta(x),\beta(y)) - \mathcal{H}(d'^\uparrow)(\beta(x),\beta(y)) &\leq \|\mathcal{H}(d^\uparrow) - \mathcal{H}(d'^\uparrow)\| \\
&\leq \|d^\uparrow - d'^\uparrow\| \\
&\leq \frac{1}{2}\|d - d'\|
\end{aligned}$$

□

As a consequence, we have the following corollary:

Corollary 3.1.18. *Φ_β has a unique fixpoint.*

Additionally, through an identical argument to Lemma 2.4.5, we can show the following:

Lemma 3.1.19. *For a finite prechart (X,β), Φ_β is cocontinuous.*

Since Φ_β has a unique fixpoint, we can use Theorem 2.4.3 to calculate behavioural distance of states of finite precharts.

Lemma 3.1.20. *Let (Q,β) be a finite prechart. The behavioural distance between any pair $q_1,q_2 \in Q$ of states can be calculated by $\mathsf{bd}_\beta(q_1,q_2) = \inf_{p\in\mathbb{N}}\left\{\Phi_\beta^{(p)}(q_1,q_2)\right\}$, where $\Phi_\beta^{(0)}$ is a discrete pseudometric and for any $p \in \mathbb{N}$, we define $\Phi_\beta^{(p+1)} = \Phi_\beta\left(\Phi_\beta^{(p)}\right)$.*

The characterisation described above can be extended to any locally finite prechart.

Corollary 3.1.21. *For any locally finite prechart (X,β), the distance between $x,y \in X$, can be calculated by:*

$$\mathsf{bd}_\beta(x,y) = \inf_{i\in\mathbb{N}}\left(\Phi_\beta^{(i)}(x,y)\right)$$

Proof. Since (X,β) is locally finite, then its subprechart $(\langle x,y\rangle_\beta, \beta|_{\langle x,y\rangle_\beta})$ is finite. Since homomorphisms are ismometries, calculating distance between x and y in (X,β) is the same as calculating it in $(\langle x,y\rangle_\beta, \beta|_{\langle x,y\rangle_\beta})$. Because of Lemma 3.1.19, Φ_β is cocontinuous (when restricted to $\langle x,y\rangle_{\beta|_{\langle x,y\rangle_\beta}}$) and hence we can employ Kleene fixpoint theorem. Since the infima in the lattice of pseudometrics can be calculated pointwise ([Róż24, Lemma 6]), we have that

$$\mathsf{bd}_\beta(x,y) = \mathsf{bd}_{\beta|_{\langle x,y\rangle_\beta}}(x,y) = \inf_{i\in I}\left(\Phi^{(i)}_{\beta|_{\langle x,y\rangle_\beta}}(x,y)\right)$$

Since $\beta|_{\langle x,y\rangle_\beta}$ is a restriction of β to $\langle x,y\rangle_\beta$ and each $\Phi^{(i)}_{\beta|_{\langle x,y\rangle_\beta}}$ makes only use of the states in $\langle x,y\rangle_\beta$, we can rewrite the above as

$$\mathsf{bd}_\beta(x,y) = \inf_{i\in\mathbb{N}}\left(\Phi_\beta^{(i)}(x,y)\right)$$

as desired. □

3.1.4 Conway Theories

Let $\mathcal{C}$ be a category, whose objects are natural numbers and 0 is the initial object. We will write 0_n for the unique map $0_n\colon 0 \to n$. Additionally, we assume that $\mathcal{C}$ is equipped with all finite coproducts, where binary coproduct is given by addition, i.e. $n \oplus m := n + m$. Given $f\colon k \to m$ and $g\colon l \to m$, we will write $[f,g] : k+l \to m$ for the mediating map from the universal property of the coproduct that makes the following diagram commute:

$$\begin{tikzcd}
k \arrow[dr, "\mathsf{inl}_{k+l}"] \arrow[ddr, "f"'] & & l \arrow[dl, "\mathsf{inr}_{k+l}"'] \arrow[ddl, "g"] \\
& k+l \arrow[d, dashed, "{[f,g]}"] & \\
& m &
\end{tikzcd}$$

For every $n \in N$, we can define a codiagonal $\nabla_n\colon n+n \to n$, given by $\nabla_n := [\mathsf{id}_n, \mathsf{id}_n]$.

Given $f\colon k \to l$ and $g\colon m \to n$, we can define their *separated sum* $f \oplus g\colon k + m \to l + n$, given by the unique mediating arrow in the following diagram

$$\begin{tikzcd}
k \arrow[r, "\mathsf{inl}_{k,m}"] \arrow[d, "f"'] & k+m \arrow[d, dashed, "f\oplus g"] & m \arrow[l, "\mathsf{inr}_{k,m}"'] \arrow[d, "g"] \\
l \arrow[r, "\mathsf{inl}_{l,n}"'] & l+n & n \arrow[l, "\mathsf{inr}_{l,n}"]
\end{tikzcd}$$

Remark 3.1.22. Because of the universal properties of coproduct and initial object, the following identities hold:

1. $[f,[g,h]] = [[f,g],h]$
2. $[0_n, f] = f = [f, 0_n]$
3. $[f,g]\,;h = [f;h, g;h]$
4. $(f \oplus g) \oplus h = f \oplus (g \oplus h)$

5. $0_n \oplus f = f = f \oplus 0_n$

6. $(f \oplus g);(h \oplus i) = f;h \oplus g;i$

Because of the above remark, we can unambiguously define an n-ary version of $[-,-]$ and hence we can view every map $f\colon m \to n$, as $f = [f_1, \dots, f_m]$.

Observe that under the assumptions listed above $\mathcal{C}$ is equipped with all finite coproducts (since it has an initial object and binary coproducts), and hence $(\mathcal{C}, \oplus, 0)$ is a cocartesian strict symmetric monoidal category.

We call $\mathcal{C}$ a *preiteration theory* if for every morphism $f\colon n \to p+n$, there exists a morphism $f^{\dagger}_{n,p}\colon n \to p$ called *dagger*. We will often omit the subscripts and write $f^{\dagger}$, when n and m are clear from the context. Note that the definition does not impose any conditions on the dagger. However, for $f\colon 0 \to p$, when always we have that $f^{\dagger}_{0,p} = 0_p$.

Definition 3.1.23 ([Ési99, Definition 3.1])**.** A Conway Theory is a preiteration theory, in which the following conditions are satisfied:

- **(Scalar parameter identity)**

$$(f;(g \oplus \mathsf{id}_1))^{\dagger} = f^{\dagger};g$$

for all $f\colon 1 \to p+1$, $g\colon p \to q$.

- **(Scalar composition identity)**

$$(f;[\mathsf{id}_p \oplus 0_1, g])^{\dagger} = f;\left[\mathsf{id}_p, (g;[\mathsf{id}_p \oplus 0_1, f])^{\dagger}\right]$$

for all $f, g\colon 1 \to p+1$.

- **(Scalar double dagger identity)**

$$f^{\dagger\dagger} = (f;(\mathsf{id}_p \oplus \nabla_1))^{\dagger}$$

for all $f\colon 1 \to p+2$.

- **(Scalar pairing identity)**

  $$[f,g]^\dagger = \left[f^\dagger;\left[\mathsf{id}_p,h^\dagger\right],h^\dagger\right]$$

  for all $f\colon n \to p+1+n$, $g\colon 1 \to p+1+n$ where

  $$h = g;\left[\mathsf{id}_{p+1},f^\dagger\right]\colon 1 \to p+1$$

Remark 3.1.24 ([Ési99, Remark 3.2]). Note that in order to define a Conway theory it suffices to define $f^\dagger\colon 1 \to p$ for all $f\colon 1 \to p+1$ that satisfies first three axioms of Definition 3.1.23 and use **scalar pairing identity** to inductively define $(-)^\dagger$.

3.1.5 Trace-fixpoint correspondence

It turns out, that having a category $\mathcal{C}$ with finite coproducts and equipped with a dagger operator satisfying the axioms of Conway Theories is synonymous with $\mathcal{C}$ being traced symmetric monoidal category. This is captured by the following theorem that was independently proved by Hasegawa [Has97] and Haghverdi [Hag00]. The formulation of Hasegawa is phrased dually via the setting of products and cartesian categories.

Theorem 3.1.25 ([Hag00, Proposition 3.1.9])**.** *For any category with finite coproducts, to give a Conway operator is to give a trace (where finite coproducts are taken as the monoidal structure).*

That bijective correspondence is concretely given by the following:

$$\frac{f\colon n \to p+n}{f^\dagger = \mathsf{Tr}^n_{n,p}(\nabla_n;f)\colon n \to p} \qquad \frac{g\colon p+n \to q+n}{\mathsf{Tr}^n_{p,q}(g) = \mathsf{inl}_{p,n};(g;\left[\mathsf{id}_q,\mathsf{inr}_{q+p,n}\right])^\dagger\colon p \to q}$$

3.1.6 Int construction

Given a traced symmetric monoidal category $(\mathcal{C},\otimes,I)$, we can construct a compact closed category $\mathbf{Int}(\mathcal{C})$. The objects of $\mathbf{Int}(\mathcal{C})$ are the pairs (A^+,A^-) of objects of $\mathcal{C}$. Morphisms f from (A^+,A^-) to (B^+,B^-) are the morphisms $f\colon A^+\otimes B^- \to$

$A^- \otimes B^+$ of $\mathcal{C}$. The identity of any object (A^+,A^-) is given by the symmetry of $\mathcal{C}$, namely $\mathsf{id}_{(A^+,A^-)} = \sigma_{A^+,A^-}$. The composition $f;g\colon (A^+,A^-) \to (C^+,C^-)$ of morphisms $f\colon (A^+,A^-) \to (B^+,B^-)$ and $g\colon (B^+,B^-) \to (C^+,C^-)$ is defined as $\mathsf{Tr}^{B^-\otimes B^+}_{A^+\otimes C^-,A^-\otimes C^+}(\alpha;(f\otimes g);\beta)$, where

$$\alpha = (\mathsf{id}_{A^+} \otimes \sigma_{C^-,B^-} \otimes \mathsf{id}_{B^+});(\mathsf{id}_{A^+} \otimes \mathsf{id}_{B^-} \otimes \sigma_{C^-,B^+})$$

$$\beta = (\mathsf{id}_{A^-} \otimes \mathsf{id}_{B^+} \otimes \sigma_{B^-,C^+});(\mathsf{id}_{A^-} \otimes \sigma_{B^+,C^+} \otimes \mathsf{id}_{B^-});(\mathsf{id}_{A^-} \otimes \mathsf{id}_{C^+} \otimes \sigma_{B^+,B^-})$$

Int$(\mathcal{C})$ is equipped with the monoidal structure. The tensor product of (A^+,A^-) and (B^+,B^-) is given by taking the tensor product of $\mathcal{C}$ pointwise, namely $(A^+ \otimes B^+,A^- \otimes B^-)$. The unit of that monoidal product is given by (I,I), where I is the unit of the monoidal product on $\mathcal{C}$. The tensor product $f \otimes g\colon (A^+ \otimes C^+,B^- \otimes D^-) \to (A^- \otimes C^-,B^+ \otimes D^+)$ of $f\colon (A^+,A^-) \to (B^+,B^-)$ and $g\colon (C^+,C^-) \to (C^+,C^-) \to (D^+,D^-)$ is given by the following:

$$f \otimes g = (\mathsf{id}_{A^+} \otimes \sigma_{C^+,B^-} \otimes \mathsf{id}_{D^-});(f \otimes g);(\mathsf{id}_{A^-} \otimes \sigma_{B^+,C^-} \otimes \mathsf{id}_{D^+})$$

The dual $(A^+,A^-)^*$ of (A^+,A^-) is given by exchanging the components, that is by (A^-,A^+). Then, the unit $\eta_{(A^+,A^-)}\colon (I,I) \to (A^+,A^-) \otimes (A^+,A^-)^*$ is a morphism $\sigma_{A^-,A^+}\colon A^- \otimes A^+ \to A^+ \otimes A^-$. The counit $\varepsilon_{(A^+,A^-)}\colon (A^+,A^-)^* \otimes (A^+,A^-) \to (I,I)$ can be similarly given by $\sigma_{A^-,A^+}\colon A^- \otimes A^+ \to A^+ \otimes A^-$ in $\mathcal{C}$.

Int$(\mathcal{C})$ is equipped with a canonical trace, which takes a morphism

$$f\colon (A^+,A^-) \otimes (U^+,U^-) \to (B^+,B^-) \otimes (U^+,U^-)$$

to the map given by the following:

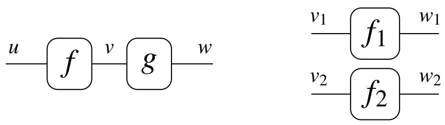
$$\left(\mathsf{id}_{(A^+,A-)} \otimes \eta_{(U^+,U-)}\right);\left(f \otimes \mathsf{id}_{(U^+,U^-)^*}\right);\left(\mathsf{id}_{(B^+,B^-)} \otimes \varepsilon_{(U^+,U-)}\right)$$

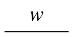
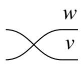
3.2 Monoidal Syntax

We adopt the diagrammatic syntax for NFA that has appeared in a number of previous papers [PZ23a; Ant+25]. We refer the reader to Selinger's classic survey [Sel10], or to Piedeleu and Zanasi's recent text for a more gentle introduction to the language of string diagrams [PZ23b].

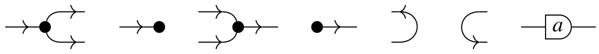
This syntax is formalised as a product and permutation category, or prop, a structure which generalises algebraic theories. Formally, a *prop* is a strict symmetric monoidal category (SMC) whose objects are words over a set of generators and whose monoidal product $\oplus$ is given by concatenation. More specifically, our syntax is the free prop $\mathsf{T}_{\mathcal{S}}$ over the signature $\mathcal{S} = (\mathcal{O}, \mathcal{M})$, given by a set $\mathcal{O}$ of generating objects and a set $\mathcal{M}$ of generating morphisms $g : v \to w$, with $v, w \in \mathcal{O}^*$ (we use ε to denote the empty word). Morphisms of $\mathsf{T}_{\mathcal{S}}$ can be combined in two different ways, using the composition operation $(-);(-) : \mathsf{T}_{\mathcal{S}}(u,v) \times \mathsf{T}_{\mathcal{S}}(v,w) \to \mathsf{T}_{\mathcal{S}}(u,w)$ or the monoidal product $(-) \oplus (-) : \mathsf{T}_{\mathcal{S}}(v_1,w_1) \times \mathsf{T}_{\mathcal{S}}(v_2,w_2) \to \mathsf{T}_{\mathcal{S}}(v_1v_2, w_1w_2)$. We also have distinguished constants: identities $\mathsf{id}_w : w \to w$, which are the unit for composition, and symmetries $\sigma_w^v : vw \to wv$, to reorder the letters of a given object. In summary, morphisms of $\mathsf{T}_{\mathcal{S}}$ can be described as terms of the $(\mathcal{O}^*, \mathcal{O}^*)$-sorted syntax generated from the constants $\mathcal{M} + \{\mathsf{id}_w : w \in \mathcal{O}^*\} + \{\sigma_w^v : v, w \in \mathcal{O}^*\}$ using the operations ; and $\oplus$, *quotiented* by the axioms of SMCs. However, the terms of this syntax are very cumbersome to work with.

We adopt a more convenient way to represent morphisms of $\mathsf{T}_{\mathcal{S}}$, using the graphical notation of *string diagrams*. In this view, a morphism $f : v \to w$ of $\mathsf{T}_{\mathcal{S}}$ is depicted as a f-labelled box with a v-labelled on the left and a w-labelled wire on the right. The operations of composition and monoidal product are represented by connecting two boxes horizontally and juxtaposing two boxes vertically, respectively:

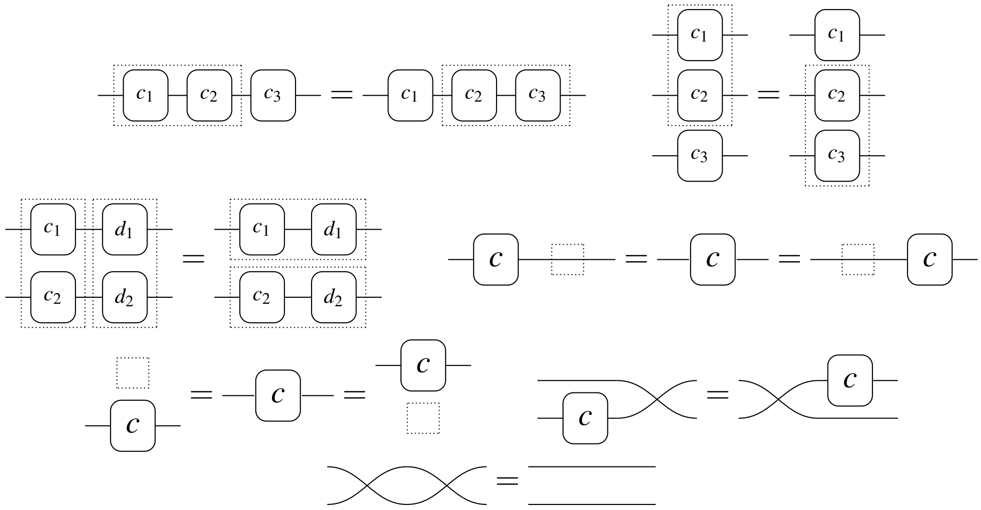


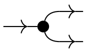
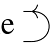
Wires  represent identities, the wire crossing  represents the symmetry σ_w^v, and the empty diagram the identity $\mathsf{id}_\varepsilon : \varepsilon \to \varepsilon$.

Definition 3.2.1. We call Syn the free prop over the signature given by

- two generating objects ◀ ("left") and ▶ ("right"), with their identity morphisms depicted respectively as —‹— and —›—;
- generating morphisms  $(a \in \Sigma)$.

Morphisms of Syn are thus vertical and horizontal composites of the generators above, potentially including wire crossings and identity wires, *up to* the laws of symmetric monoidal categories, listed below:



The direction of the arrows on the wires denotes the corresponding object: for example,  represents an operation of type ▶→▶▶, while  has type ◀▶→ ε. Note that, when we have n parallel wires of the same type, say ▶, we depict them as a single directed wire labelled by a natural number label, as —›—^{n}. We call *inputs* the incoming wires of a diagram, and *outputs* its outgoing wires; formally, the inputs (resp. outputs) of $f : v \to w$ are the set of positions of the word v which are ▶ (resp. ◀) and the position of w which are ◀ (resp. ▶).

3.3 Monoidal semantics

In order to interpret the string diagrams described in Section 3.2, we construct an appropriate semantic universe out of regular behaviours. As much as the technical development makes use of category theory, we will keep the description of the formalism high-level. We will write V_n for the set $V_n = \{v_1, \dots, v_n\} \subseteq V$ and $\mathsf{MExp}/{\equiv}(n)$ for the set of all regular behaviours whose live variables are contained in the set V_n. For any $m, n \in \mathbb{N}$, we will write $\mathsf{RegBeh}(m,n)$ for the set of m-tuples of elements of $\mathsf{MExp}/{\equiv}(n)$.

For every $n \in N$, we define an identity map $\mathsf{id}_n \in \mathsf{RegBeh}(n,n)$ as $\mathsf{id}_n = \vec{v}_n = (v_1, \dots, v_n)$. When n is clear from the context, we will abuse the notation and simply write $\vec{v}$ instead.

Given $f \in \mathsf{RegBeh}(m,n)$ and $g \in \mathsf{RegBeh}(n,p)$, we can define their sequential composition $f;g \in \mathsf{RegBeh}(m,p)$ to be given by $(f_1[\vec{g}/\vec{v}], \dots, f_m[\vec{g}/\vec{v}])$, where $\vec{v} = (v_1, \dots, v_n)$. It turns out that sequential composition is associative, with identity being a neutral element when composed both on the left and right.

Lemma 3.3.1. *Let $f\colon m \to n$, $g\colon n \to p$, $h\colon p \to q$. We have that:*

1. $(f;g);h = f;(g;h)$
2. $\mathsf{id}_m; f = f$
3. $f; \mathsf{id}_n = f$

Proof. We respectively prove each of the properties.

1. $$\begin{aligned}(f;g);h &= (f_1[\vec{g}/\vec{v}], \dots, f_m[\vec{g}/\vec{v}]);h \\ &= \Big((f_1[\vec{g}/\vec{v}])[\vec{h}, \vec{v}], \dots, (f_m[\vec{g}/\vec{v}])[\vec{h}, \vec{v}]\Big) \\ &= \Big(f_1[(g_1[\vec{h}/\vec{v}], \dots, g_n[\vec{h}/\vec{v}])/\vec{v}], \dots, f_m[(g_1[\vec{h}/\vec{v}], \dots, g_n[\vec{h}/\vec{v}])/\vec{v}]\Big) \\ &\qquad\qquad\qquad \text{(Lemma 3.1.10)} \\ &= \Big(f_1[(\overrightarrow{g;h})/\vec{v}], \dots f_m[(\overrightarrow{g;h})/\vec{v}]\Big) \\ &= f;(g;h)\end{aligned}$$

2. $id_m; f = (v_1[\vec{f}/\vec{v}], \dots, v_m[\vec{f}/\vec{v}]) = (f_1, \dots, f_m) = f$

3. $f;id_n = (f_1[\vec{v}/\vec{v}],\dots,f_m[\vec{v}/\vec{v}]]) = (f_1,\dots,f_m) = f$

□

Because of the above, we can define a category RegBeh, whose objects are natural numbers and morphisms $f\colon m \to n$ are elements $f \in \mathsf{RegBeh}(m,n)$.

For every $n \in \mathbb{N}$, there is a unique element $0_n \in \mathsf{RegBeh}(0,n)$ given by the empty tuple.

Lemma 3.3.2. 0 *is the initial object of* RegBeh.

Proof. For any $n \in \mathbb{N}$, the unique universal arrow is given by 0_n, which immediately completes the proof. □

Moreover, RegBeh can be equipped with binary coproducts, which is defined on objects as addition.

Lemma 3.3.3. RegBeh *has binary coproducts. In particular, given $k,l \in \mathbb{N}$, the inclusions $\mathsf{inl}_{k,l}\colon k \to k+l$ and $\mathsf{inr}_{k,l}\colon l \to k+l$ are given by $\mathsf{inl}_{k,l} = (v_1,\dots,v_k)$ and $\mathsf{inr}_{k,l} = (v_{k+1},\dots,v_{k+l})$ respectively, while the mediating map is given by pairing.*

Proof. Let $f\colon k \to m$ and $g\colon l \to m$. We can safely assume that $f = (f_1,\dots,f_k)$ and $g = (g_1,\dots,g_l)$. Recall that $\mathsf{inl}_{k,l} = (v_1,\dots,v_k)$ and $\mathsf{inr}_{k,l} = (v_{k+1},\dots,v_{k+l})$. For the existence proof, define $[f,g] : k+l \to m$ as a $k+l$-tuple $(f_1,\dots,f_k,g_1,\dots,g_l)$. We show that that the coproduct diagram commutes. We start from the left triangular subdiagram.

$$\begin{tikzcd}
k \arrow[dr, "\mathsf{inl}_{k+l}"] \arrow[ddr, "f"'] & & l \arrow[dl, "\mathsf{inr}_{k+l}"'] \arrow[ddl, "g"] \\
& k+l \arrow[d, dashed, "{[f,g]}"] & \\
& m &
\end{tikzcd}$$

$$\begin{aligned}
\mathsf{inl}_{k,l};[f,g] &= (v_1,\dots,v_k);(f_1,\dots f_k,g_1,\dots,g_l) \\
&= (f_1,\dots,f_k)
\end{aligned}$$

$$= f$$

Similarly, for the right subdiagram, we have that:

$$\begin{aligned}
\mathsf{inr}_{k,l};[f,g] &= (v_{k+1},\dots,v_{k+l});(f_1,\dots f_k,g_1,\dots,g_l) \\
&= (g_1,\dots,g_l) \\
&= g
\end{aligned}$$

For the uniqueness proof, assume that there exists a map $h\colon k+l \to m$, which makes the coproduct diagram commute. We can safely assume that $h = (h_1,\dots,h_{k+l})$. Since $\mathsf{inl}_{k,l};h = f$ and $\mathsf{inr}_{k,l};h = g$, we have that:

$$\begin{aligned}
(f_1,\dots,f_k) &= f \\
&= \mathsf{inl}_{k,l};h \\
&= (v_1,\dots,v_k);(h_1,\dots,h_{k+l}) \\
&= (h_1,\dots,h_k)
\end{aligned}$$

Similarly, we have that:

$$\begin{aligned}
(g_1,\dots,g_l) &= f \\
&= \mathsf{inr}_{k,l};h \\
&= (v_{k+1},\dots,v);(h_1,\dots,h_{k+l}) \\
&= (h_{k+1},\dots,h_{k+l})
\end{aligned}$$

Hence, $h = (f_1,\dots,f_k,g_1,\dots,g_l) = [f,g]$ as desired. □

Chapter 4

Completeness Theorem for Probabilistic Regular Expressions

Kleene [Kle51] introduced regular expressions and proved that these denote exactly the languages accepted by deterministic finite automata. In his seminal paper, Kleene left open a completeness question: are there a finite number of rules that enable reasoning about language equivalence of regular expressions? Since then, the pursuit of inference systems for equational reasoning about the equivalence of regular expressions has been subject of extensive study [Sal66; Kro90; Bof90; Koz94]. The first proposal is due to Salomaa [Sal66], who introduced a non-algebraic axiomatisation of regular expressions and proved its completeness.

Deterministic automata are a particular type of transition system: simply put, an automaton is an object with a finite set of states and a *deterministic* transition function that assigns every state and every action of the input alphabet *exactly* one next state. By varying the type of transition function one gets different systems: e.g. if the function assigns to every state and every action of the input alphabet *a set* of the next states, the resulting system is said to be non-deterministic; if the transition function assigns the next state based on any sort of *probability distribution* then the system is said to be probabilistic. Probabilistic systems appear in a range of applications, including modelling randomised algorithms, cryptographic protocols, and probabilistic programs. In this chapter, we focus on generative probabilistic transition systems (GPTS) with explicit termination [GSS95], and study the questions that Kleene and Salomaa answered for deterministic automata.

Our motivation to look at probabilistic extensions of regular expressions and axiomatic reasoning is two-fold: first, regular expressions and extensions thereof have been used in the verification of uninterpreted imperative programs, including network policies [KP00; KK05; And+14]; second, reasoning about exact behaviour of probabilistic imperative programs is subtle [Che+22], in particular in the presence of loops. By studying the semantics and axiomatisations of regular expressions featuring probabilistic primitives, we want to enable axiomatic reasoning for randomised programs and provide a basis to develop further verification techniques.

We start by introducing the syntax of Probabilistic Regular Expressions (PRE), inspired by work from the probabilistic pattern matching literature [Ros00]. PRE are formed through constants from an alphabet and *regular* operations of probabilistic choice, sequential composition, probabilistic Kleene star, identity and emptiness. We define the probabilistic analogue of *language semantics* of PRE as exactly the behaviours of GPTS. We achieve this by endowing PRE with operational semantics in the form of GPTS via a construction reminiscent of Antimirov derivatives of regular expressions [Ant96]. We also give a converse construction, allowing us to describe languages accepted by finite-state GPTS in terms of PRE, thus establishing an analogue of Kleene's theorem.

The main contribution of this chapter is presenting an inference system for reasoning about probabilistic language equivalence of PRE and proving its completeness. The technical core of this chapter is devoted to the completeness proof, which relies on technical tools convex algebra, arising from the rich structure of probabilistic languages. While being in the spirit of classic results from automata theory, our development relies on the more abstract approach enabled via the theory of universal coalgebra [Rut00]. As much as a concrete completeness proof ought to be possible, our choice to use the coalgebraic approach was fueled by wanting to reuse recent abstract results on the algebraic structure of probabilistic languages. A concrete proof would have to deal with fixpoints of probabilistic languages and would therefore require highly combinatorial and syntactic proofs about these. Instead, we reuse a range of hard results on convex algebras and fixpoints that Milius [Mil18], Sokolova and Woracek [SW15; SW18] proved in the last 5 years. In particular, we rely on the theory of rational fixpoints (for proper functors [Mil18]), which can be seen as a categorical generalisation of regular languages.

Our completeness proof provides further evidence that the use of coalgebras over proper functors provides a good abstraction for completeness theorems, where general steps can be abstracted away leaving as a domain-specific task to achieve completeness a construction to syntactically build solutions to systems of equations. Proving the uniqueness of such solutions is ultimately the most challenging step in the proof. By leveraging the theory of proper functors, our proof of completeness, which depends on establishing an abstract universal property, boils down to an argument that can be viewed as a natural extension of the work by Salomaa [Sal66] and Brzozowski [Brz64] from the 1960s.

The remainder of this chapter is organised as follows:

In Section 4.1, we introduce Probabilistic Regular Expressions (PRE), an analogue of Kleene's regular expressions denoting probabilistic languages and propose an inference system for reasoning about language equivalence of PRE.

Then, in Section 4.2, we elaborate on the main theoretical preliminaries for the technical development of this chapter. The coalgebraic approach to language semantics of GPTS is described in Section 4.3. In the remainder of that section, we provide a small-step semantics of PRE through an analogue of Antimirov derivatives endowing expressions with a structure of Generative Probabilistic Transition Systems (GPTS).

The technical core of this chapter is located in Section 4.4 and Section 4.5, where we obtain soundness and completeness results for our axiomatisation. Due to our use of proper functors, the proof boils down to a generalisation of a known proof of Salomaa for regular expressions [Sal66] exposing the connection to a classical result. We also obtain an analogue of Kleene's theorem allowing the conversion of finite-state GPTS to expressions through an analogue of Brzozowski's method [Brz64].

We conclude the chapter in Section 4.6, where we survey related work and sketch some areas for future work.

4.1 Overview

In this section, we will introduce the syntax and the language semantics of probabilistic regular expressions (PRE), as well as a candidate inference system to reason about the equivalence of PRE.

4.1.1 Syntax

Given a finite alphabet A, the syntax of PRE is given by:

$$e, f \in \mathsf{PExp} ::= 0 \mid 1 \mid a \in A \mid e \oplus_p f \mid e \,;f \mid e^{[p]} \qquad p \in [0,1]$$

We denote the expressions that immediately abort and successfully terminate by 0 and 1 respectively. For every letter $a \in A$ in the alphabet, there is a corresponding expression representing an atomic action. Given two expressions $e, f \in \mathsf{PExp}$ and $p \in [0,1]$, probabilistic choice $e \oplus_p f$ denotes an expression that performs e with probability p and performs f with probability $1-p$. One can think of $\oplus_p$ as the probabilistic analogue of the plus operator $(e+f)$ in Kleene's regular expressions. $e\,;f$ represents sequential composition, while $e^{[p]}$ is a probabilistic analogue of Kleene star: it successfully terminates with probability $1-p$ or with probability p performs e and then iterates $e^{[p]}$ again. In terms of the notational convention, the sequential composition (;) has higher precedence than the probabilistic choice $(\oplus_p)$.

Example 4.1.1. The expression $a\,;a^{[\frac{1}{4}]}$ first performs action a with probability 1 and then enters a loop which successfully terminates with probability $\frac{3}{4}$ or performs action a with probability $\frac{1}{4}$ and then repeats the loop again. Intuitively, if we think of the action a as observable, the expression above denotes a probability associated with a non-empty sequence of a's. For example, the sequence aaa would be observed with probability $1 \cdot (1/4)^2 \cdot 3/4 = 3/64$.

4.1.2 Language semantics

PRE denote probabilistic languages $A^* \to [0,1]$. For instance, the expression 0 denotes a function that assigns 0 to every word, whereas 1 and a respectively assign probability 1 to the empty word and the word containing a single letter a from the alphabet. The probabilistic choice $e \oplus_p f$ denotes a language in which the probability of each word is the total sum of its probability in e scaled by p and its probability in f scaled by $1-p$. Describing the semantics of sequential composition and loops inductively is more involved. In particular, the semantics of loops would require a fixpoint calculation, which does not have as clear and straightforward (closed-form) formula, as the asterate of regular languages. Instead, we take an *operational approach*, and we formally define the language semantics of PRE in Section 4.3 through a small-step operational semantics, using a specific type of probabilistic transition system, which we introduce next.

4.1.3 Generative probabilistic transition systems

A GPTS consists of a set of states Q and a transition function that maps each state $q \in Q$ to finitely many distinct outgoing arrows of the form:

- *successful termination* with probability t (denoted $q \overset{t}{\Rightarrow} \checkmark$), or
- to another state r, via an *a-labelled transition*, with probability $s \in [0,1]$ (denoted $q \xrightarrow{a|s} r$).

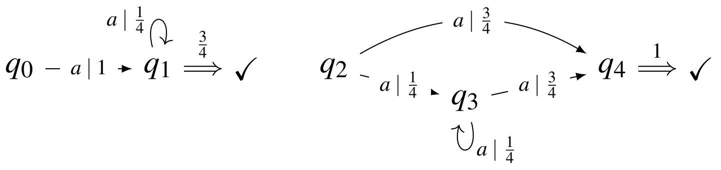
We require that, for each state, the total sum of probabilities appearing on outgoing arrows sums up to less or equal to one. The remaining probability mass is used to model unsuccessful termination, hence the state with no outgoing arrows can be thought of as exposing deadlock behaviour.

Given a word $w \in A^*$ the probability of it being generated by a state $q \in Q$ (denoted $\mathsf{Lang}(q)(w) \in [0,1]$) is defined inductively:

$$\mathsf{Lang}(q)(\varepsilon) = t \quad \text{if } q \overset{t}{\Rightarrow} \checkmark \qquad\qquad \mathsf{Lang}(q)(av) = \sum_{q \xrightarrow{a|s} r} s \cdot \mathsf{Lang}(r)(v) \tag{4.1}$$

We say that two states q and q' are *language equivalent* if for all words $w \in A^*$, we have that $\mathsf{Lang}(q)(w) = \mathsf{Lang}(q')(w)$.

Example 4.1.2. Consider the following GPTS:



States q_0 and q_2 both assign probability 0 to the empty word ε and each word a^{n+1} is mapped to the probability $\left(\frac{1}{4}\right)^n \cdot \frac{3}{4}$. Later, we show that the languages generated by states q_0 and q_2 can be specified using expressions $a\,;a^{[\frac{1}{4}]}$ and $a \oplus_{\frac{3}{4}} (a\,;a^{[\frac{1}{4}]}\,;a)$ respectively.

In Section 4.3, we will associate to each PRE e an operational semantics or, more precisely, a state q_e in a GPTS. The language semantics $[\![e]\!]$ of e will then be the language $\mathsf{Lang}(q_e)\colon A^* \to [0,1]$ generated by q_e. Two PRE e and f are language equivalent if $\mathsf{Lang}(q_e) = \mathsf{Lang}(q_f)$. One of our main goals is to present a complete inference system to reason about language equivalence. In a nutshell, we want to present a system of (quasi-)equations of the form $e \equiv f$ such that:

$$e \equiv f \Leftrightarrow [\![e]\!] = [\![g]\!] \Leftrightarrow \mathsf{Lang}(e) = \mathsf{Lang}(f)$$

Such an inference system will have to contain rules to reason about all constructs of PRE, including probabilistic choice and loops. We describe next the system, with some intuition for the inclusion of each group of rules.

4.1.4 Axiomatisation of language equivalence of PRE

We define $\equiv \subseteq \mathsf{PExp} \times \mathsf{PExp}$ to be the least congruence relation closed under the axioms shown on Figure 4.1. We will show in Section 4.5 that these axioms are complete with respect to language semantics.

The first group of axioms capture properties of the probabilistic choice operator $\oplus_p$ (C1-C4) and its interaction with sequential composition (D1-D2). Intuitively, (C1-C4) are the analogue of the semilattice axioms governing the behaviour of

Probabilistic choice

(C1) $e \oplus_p e \equiv e$

(C2) $e \oplus_1 f \equiv e$

(C3) $e \oplus_p f \equiv f \oplus_{1-p} e$

(C4) $(e \oplus_p f) \oplus_q g \equiv e \oplus_{pq} \left(f \oplus_{\frac{(1-p)q}{1-pq}} g \right)$

Sequential composition

(1S) $1 ; e \equiv e ,$

(S) $e ; (f ; g) \equiv (e ; f) ; g ,$

(S1) $e ; 1 \equiv e ,$

(0S) $0 ; e \equiv 0 ,$

(S0) $e ; 0 \equiv 0 ,$

(D1) $e ; (f \oplus_p g) \equiv e ; f \oplus_p e ; g$

(D2) $(e \oplus_p f) ; g \equiv e ; g \oplus_p f ; g$

Loops

(Unroll) $e^{[p]} \equiv e ; e^{[p]} \oplus_p 1$

(Tight) $(e \oplus_p 1)^{[q]} \equiv e^{\left[\frac{pq}{1-(1-p)q}\right]}$

(Div) $1^{[1]} \equiv 0$

(Unique) $\dfrac{g \equiv e ; g \oplus_p f \qquad \boxed{E(e) = 0}}{g \equiv e^{[p]} ; f}$

Termination condition: $E \colon \mathsf{PExp} \to [0,1]$

$$\boxed{\begin{array}{ccc} E(1) = 1 & E(0) = E(a) = 0 & E(e \oplus_p f) = pE(e) + (1-p)E(f) \\ E(e ; f) = E(e)E(f) & & E\left(e^{[p]}\right) = \begin{cases} 0 & E(e) = 1 \wedge p = 1 \\ \frac{1-p}{1-pE(e)} & \text{otherwise} \end{cases} \end{array}}$$

Figure 4.1: Axioms for language equivalence of PRE. The rules involving the division of probabilities are defined only when the denominator is non-zero. The function $E(-)$ provides a termination side condition to the (Unique) fixpoint axiom.

$+$ in regular expressions. These four axioms are reminiscent of the axioms of barycentric algebras [Sto49]. (D1) and (D2) are *right and left distributivity* rules of $\oplus$ over ;. The sequencing axioms (1S), (S1), (S) state PRE have the structure of a monoid (with neutral element 1 with absorbent element 0 – see axioms (0S), (S0). The loop axioms contain respectively *unrolling*, *tightening*, and *divergency* axioms plus a *unique fixpoint* rule. The (Unroll) axiom associates loops with their intuitive behaviour of choosing, at each step, probabilistically between successful termination and executing the loop body once. (Tight) and (Div) are the probabilistic analogues of the identity $(e+1)^* \equiv e^*$ from regular expressions. In the case of PRE, we need two axioms: (Tight) states that the probabilistic loop whose body might instantly terminate, causing the next loop iteration to be executed immediately is provably equivalent to a different loop, whose body does not contain immediate termination; (Div) takes care of the edge case of a no-exit loop and identifies it with failure. Finally, the unique fixpoint rule is a re-adaptation of the analogous axiom from Salomaa's axiomatisation and provides a partial converse to the loop unrolling axiom, given the loop body is productive – i.e. cannot immediately terminate. This productivity property is formally written using the side condition $E(e) = 0$, which can be thought of as the probabilistic analogue of empty word property from Salomaa's axiomatisation. Consider an expression $a^{\left[\frac{1}{2}\right]} ; (b \oplus_{\frac{1}{2}} 1)$. The only way it can accept the empty word is to leave the loop with the probability of $\frac{1}{2}$ and then perform 1, which also can happen with probability $\frac{1}{2}$. In other words, $[\![a^{\left[\frac{1}{2}\right]} ; (b \oplus_{\frac{1}{2}} 1)]\!](\varepsilon) = \frac{1}{4}$. A simple calculation allows to verify that $E(a^{\left[\frac{1}{2}\right]} ; (b \oplus_{\frac{1}{2}} 1)) = \frac{1}{4}$.

Example 4.1.3. We revisit the expressions from Example 4.1.2 and show their equivalence via axiomatic reasoning.

$$\begin{aligned}
a ; a^{\left[\frac{1}{4}\right]} &\equiv a ; (a^{\left[\frac{1}{4}\right]} ; a \oplus_{\frac{1}{4}} 1) && (\dagger)\\
&\equiv (a ; a^{\left[\frac{1}{4}\right]} ; a) \oplus_{\frac{1}{4}} a ; 1 && (\mathsf{D2})\\
&\equiv (a ; a^{\left[\frac{1}{4}\right]} ; a) \oplus_{\frac{1}{4}} a && (\mathsf{S1})\\
&\equiv a \oplus_{\frac{3}{4}} (a ; a^{\left[\frac{1}{4}\right]} ; a) && (\mathsf{C3})
\end{aligned}$$

The † step of the proof above relies on the equivalence $e^{[p]} ; e \oplus_p 1 \equiv e$ derivable from other axioms under the assumption $E(e) = 0$ through a following line of reasoning:

$$\begin{aligned}
e^{[p]} ; e \oplus_p 1 &\equiv (e ; e^{[p]} \oplus_p 1) ; e \oplus_p 1 && (\mathsf{Unroll})\\
&\equiv (e ; e^{[p]} ; e \oplus_p 1 ; e) \oplus_p 1 && (\mathsf{D1})\\
&\equiv (e ; e^{[p]} ; e \oplus_p e) \oplus_p 1 && (\mathsf{1S})
\end{aligned}$$

$$\equiv (e\,;e^{[p]}\,;e \oplus_p e\,;1) \oplus_p 1 \tag{S1}$$

$$\equiv e\,;(e^{[p]}\,;e \oplus_p 1) \oplus_p 1 \tag{D2}$$

Since $E(e) = 0$, we then have: $e^{[p]}\,;e \oplus_p 1 \overset{\text{(Unique)}}{\equiv} e^{[p]}\,;1 \overset{\text{(S1)}}{\equiv} e^{[p]}$.

4.2 Preliminaries

In this section, we review the main preliminaries for the technical development outlined in the subsequent sections.

4.2.1 Locally finitely presentable categories

In this chapter, we will rely on notions associated with the theory of locally finitely presentable categories [AR94], that allows to generalise the notion of *finiteness* to more structured categories than just Set.

$\mathcal{D}$ is a filtered category, if every finite subcategory $\mathcal{D}_0 \hookrightarrow \mathcal{D}$ has a cocone in $\mathcal{D}$. A filtered colimit is a colimit of the diagram $\mathcal{D} \to \mathcal{C}$, where $\mathcal{D}$ is a filtered category. A directed colimit is a colimit of the diagram $\mathcal{D} \to \mathcal{C}$, where $\mathcal{D}$ is a directed poset. We call a functor *finitary* if it preserves filtered colimits. An object C is *finitely presentable (fp)* if the representable functor $\mathcal{C}(C,-)\colon \mathcal{C} \to \mathsf{Set}$ preserves filtered colimits. Similarly, an object C is *finitely generated (fg)* if the representable functor $\mathcal{C}(C,-)\colon \mathcal{C} \to \mathsf{Set}$ preserves directed colimits of monomorphisms. Importantly, every finitely presentable object is finitely generated, but the converse does not hold in general.

Definition 4.2.1. A category $\mathcal{C}$ is *locally finitely presentable (lfp)* if it is cocomplete and there exists a set of finitely presentable objects, such that every object of $\mathcal{C}$ is a filtered colimit of objects from that set.

Set is the prototypical example of a locally finitely presentable category, where finitely presentable objects are precisely finite sets.

4.2.2 Monads and their algebras

A monad (over the category Set) is a triple $\mathbf{T} = (T, \mu, \eta)$ consisting of a functor $T\colon \mathsf{Set} \to \mathsf{Set}$ and two natural transformations: a unit $\eta\colon \mathsf{Id} \Rightarrow T$ and multiplication

$\mu\colon T^2 \Rightarrow T$ satisfying $\mu \circ \eta_T = \mathsf{id}_T = \mu \circ T\eta$ and $\mu \circ \mu_T = \mu \circ T\mu$ A **T**-algebra (also called an Eilenberg-Moore algebra) for a monad T is a pair (X,h) consisting of a set $X \in \mathcal{O}(\mathcal{C})$, called carrier, and a function $h\colon TX \to X$ such that $h \circ \mu_X = h \circ Th$ and $h \circ \eta_X = \mathsf{id}_X$. A **T**-algebra homomorphism between two T-algebras (X,h) and (Y,k) is a function $f\colon X \to Y$ satisfying $k \circ Tf = f \circ h$.

T-algebras and **T**-homomorphisms form a category $\mathsf{Set}^{\mathbf{T}}$. There is a canonical forgetful functor $\mathcal{U}\colon \mathsf{Set}^{\mathbf{T}} \to \mathsf{Set}$ that takes each **T**-algebra to its carrier. This functor has a left adjoint $X \mapsto (TX, \mu_X\colon T^2X \to T)$, mapping each set to its free **T**-algebra. If X is finite, then we call (TX, μ_X) free finitely generated.

Given a function $f\colon X \to Y$, where Y is a carrier of a **T**-algebra (Y,h), there is a unique homomorphism $f^\sharp\colon (TX,\mu_X) \to (Y,h)$ satisfying $f^\sharp \circ \eta_X = f$ that is explicitly given by $f^\sharp = h \circ Tf$.

4.2.3 Generalised determinisation

Language acceptance of nondeterministic automata (NDA) can be captured via determinisation. NDA can be viewed as coalgebras for the functor $N = 2 \times \mathcal{P}_\omega{}^A$, where $\mathcal{P}_\omega$ is the finite powerset monad. Determinisation converts a NDA $(X, \beta\colon X \to 2 \times \mathcal{P}_\omega X^A)$ into a deterministic automaton $\left(\mathcal{P}_\omega X, \beta^\sharp\colon \mathcal{P}_\omega X \to 2 \times (\mathcal{P}_\omega X)^A\right)$, where for $A \subseteq X$, we define $\beta^\sharp(A) = \bigcup_{x \in A} \beta(a)$. Additionally, $\beta^\sharp$ satisfies $\beta^\sharp(\{x\}) = \beta(x)$ for all $x \in X$. A language of the state $x \in X$ of NDA, is given by the language accepted by the state $\{x\}$ in the determinised automaton $(\mathcal{P}_\omega X, \beta^\sharp)$.

This construction can be generalised [Sil+10] to HT-coalgebras, where $T\colon \mathsf{Set} \to \mathsf{Set}$ is an underlying functor of finitary monad **T** and $H\colon \mathsf{Set} \to \mathsf{Set}$ an endofunctor that admits a final coalgebra that can be lifted to the functor $\overline{H}\colon \mathsf{Set}^{\mathbf{T}} \to \mathsf{Set}^{\mathbf{T}}$. Liftings of functors $H\colon \mathsf{Set} \to \mathsf{Set}$ to $\overline{H}\colon \mathsf{Set}^{\mathbf{T}} \to \mathsf{Set}^{\mathbf{T}}$, are in one-to-one correspondence with distributive laws of the monad T over the functor H [JSS15], which are natural transformations $\rho\colon TH \Rightarrow HT$ satisfying $H\eta_X = \rho_X \circ \eta_{HX}$ and $H\mu_X \circ \rho_{TX} \circ T\rho_X = \rho_X \circ \mu_{HX}$. In particular, given a **T**-algebra $(X, k\colon TX \to X)$, we can equip HX, with a **T**-algebra structure, given by the following composition of maps:

$$THX \xrightarrow{\rho_X} HTX \xrightarrow{Ht} HX$$

Generalised determinisation turns HT-coalgebras $(X, \beta\colon X \to HTX)$ into H-coalgebras $(TX, \beta^\sharp\colon TX \to HTX)$, where $\beta^\sharp$ is the unique extension arising from the free-forgetful adjunction between Set and $\mathrm{Set}^{\mathbf{T}}$. The language of a state $x \in X$ is given by $\mathsf{beh}_{\beta^\sharp} \circ \eta_X\colon X \to \nu H$, where η is the unit of the monad $\mathbf{T}$. Since $\beta^\sharp\colon TX \to HTX$ can be seen as a $\mathbf{T}$-algebra homomorphism $(TX, \mu_X) \to \overline{H}(TX, \mu_X)$, each determinisation $(TX, \beta^\sharp)$ can be viewed as an $\overline{H}$-coalgebra $((TX, \mu_X), \beta^\sharp)$. The carrier of the final H-coalgebra can be canonically equipped with $\mathbf{T}$-algebra structure, yielding the final $\overline{H}$-coalgebra. In such a case, the unique final homomorphism from any determinisation (viewed as an H-coalgebra) is precisely an underlying function of the final $\overline{H}$-coalgebra homomorphism.

4.2.4 Subdistribution monad

A function $\nu\colon X \to [0,1]$ is called a subprobability distribution or subdistribution, if it satisfies $\sum_{x \in X} \nu(x) \leq 1$. A subdistribution ν is *finitely supported* if the set $\mathrm{supp}(\nu) = \{x \in X \mid \nu(x) > 0\}$ is finite. We use $\mathcal{D}X$ to denote the set of finitely supported subprobability distributions on X. The weight of a subdistribution $\nu\colon X \to [0,1]$ is a total probability of its support:

$$|\nu| = \sum_{x \in X} \nu(x)$$

Given $\nu \in \mathcal{D}X$ and $Y \subseteq X$, we will write $\nu[Y] = \sum_{x \in Y} \nu(x)$. This sum is well-defined as only finitely many summands have non-zero probability.

Given $x \in X$, its *Dirac* is a subdistribution δ_x which is given by $\delta_x(y) = 1$ only if $x = y$, and 0 otherwise. We will moreover write $\mathbb{0} \in \mathcal{D}X$ for a subdistribution with an empty support. It is defined as $\mathbb{0}(x) = 0$ for all $x \in X$. When $\nu_1, \nu_2\colon X \to [0,1]$ are subprobability distributions and $p \in [0,1]$, we write $p\nu_1 + (1-p)\nu_2$ for the convex combination of ν_1 and ν_2, which is the probability distribution given by

$$(p\nu_1 + (1-p)\nu_2)(x) = p\nu_1(x) + (1-p)\nu_2(x)$$

for all $x \in X$. Note that this operation preserves finite support.

$\mathcal{D}$ is in fact a functor on the category Set, which maps each set X to $\mathcal{D}X$ and maps each arrow $f\colon X \to Y$ to the function $\mathcal{DF}\colon \mathcal{D}X \to \mathcal{D}Y$ given by

$$\mathcal{DF}(\nu)(x) = \sum_{y \in f^{-1}(x)} \nu(y)$$

Moreover, $\mathcal{D}$ also carries a monad structure with unit $\eta_X(x) = \delta_x$ and multiplication $\mu_X(\Phi)(x) = \sum_{\varphi \in \mathcal{D}X} \Phi(\varphi)\varphi(x)$ for $\Phi \in \mathcal{D}^2X$. Using the free-forgetful adjunction between Set and category of $\mathcal{D}$-algebras, given $f\colon X \to \mathcal{D}Y$, there exists a unique map $f^\sharp\colon \mathcal{D}X \to \mathcal{D}Y$ satisfying $f = f^\sharp \circ \delta$ called the *convex extension of* f, and explicitly given by $f^\sharp(\nu)(y) = \sum_{x \in X} \nu(x)f(x)(y)$.

4.2.5 Positive convex algebras

By Σ_{PCA} we denote a signature given by

$$\Sigma_{\mathsf{PCA}} = \left\{ \boxplus_{i \in I} p_i \cdot (-)_i \mid I \text{ finite}, \forall i \in I.\, p_i \in [0,1], \sum_{i \in I} p_i \leq 1 \right\}$$

A positive convex algebra is a an algebra for the signature Σ_{PCA}, that is a pair $\mathcal{A} = \left(X, \Sigma^{\mathcal{A}}_{\mathsf{PCA}}\right)$, where X is the carrier set and $\Sigma^{\mathcal{A}}_{\mathsf{PCA}}$ is a set of interpretation functions $\boxplus_{i \in I} p_i \cdot (-)_i\colon X^{|I|} \to X$ satisfying the axioms:

1. (Projection) $\boxplus_{i \in I} p_i \cdot x_i = x_j$ if $p_j = 1$

2. (Barycenter) $\boxplus_{i \in I} p_i \cdot \left(\boxplus_{j \in J} q_{i,j} \cdot x_j\right) = \boxplus_{j \in J} \left(\sum_{i \in I} p_i q_{i,j}\right) \cdot x_j$

In terms of notation, we denote the unary sum by $p_0 \cdot x_0$. Throughout this chapter we will we abuse the notation by writing

$$\left(\boxplus_{i \in I} p_i \cdot e_i\right) \boxplus \left(\boxplus_{i \in J} q_j \cdot f_j\right)$$

for a single sum $\boxplus_{k \in I+J} r_k \cdot g_k$, where $r_k = p_k$ and $g_k = e_k$ for $k \in I$ and similarly $r_k = q_k$ and $g_k = f_k$ for $k \in J$. Note that this is well-defined only if $\sum_{i \in I} p_i + \sum_{j \in J} r_j \leq 1$.

The signature of positive convex algebras can be alternatively presented as a family of binary operations, in the following way:

Proposition 4.2.2 ([BSS17, Proposition 7])**.** *If X is a set equipped with a binary operation $\boxplus_p : X \times X \to X$ for each $p \in [0,1]$ and a constant $0_{\boxplus} \in X$ satisfying for all $x, y, z \in X$ (when defined) the following:*

$$x \boxplus_p x = x \qquad x \boxplus_1 y = x \qquad x \boxplus_p y = y \boxplus_{1-p} x$$
$$(x \boxplus_p y) \boxplus_q z = x \boxplus_{pq} \left(y \boxplus_{\frac{(1-p)q}{1-pq}} z \right)$$

then X carries the structure of a positive convex algebra. The interpretation of $\boxplus_{i \in I} p_i \cdot (-)_i$ is defined inductively by the following

$$\mathop{\boxplus}_{i \in I} p_i \cdot x_i = \begin{cases} 0_{\boxplus} & \textit{if } I = \emptyset \\ x_0 & \textit{if } p_0 = 1 \\ x_n \boxplus_{p_k} \left(\boxplus_{i \in I \setminus \{k\}} \frac{p_i}{1-p_k} \cdot x_i \right) & \textit{otherwise, for some } k \in I \end{cases}$$

Below we state several properties of positive convex algebras, that we will use throughout this chapter.

Proposition 4.2.3. *Let I be a finite indexed set, and let $\{p_i\}_{i \in I}$ and $\{x_i\}_{i \in I}$ be indexed collections of elements of $[0,1]$ and X respectively. Then, in any positive convex algebra, the following statements hold:*

1. $$\mathop{\boxplus}_{i \in I} p_i \cdot x_i = \mathop{\boxplus}_{x \in \bigcup_{i \in I} \{x_i\}} \left(\sum_{x_i = x} p_i \right) \cdot x$$

2. *Let $=_R \subseteq X \times X$ be a congruence relation, with $[-]_R \colon X \to X/{=_R}$ being its canonical quotient map. Then,*

   $$\mathop{\boxplus}_{i \in I} p_i \cdot x_i =_R \mathop{\boxplus}_{[x]_R \in \bigcup_{i \in I} \{[x_i]_R\}} \left(\sum_{x_i =_R x} p_i \right) \cdot x$$

3. *All terms $\boxplus_{i \in I} 0 \cdot x_i$ coincide and are all provably equivalent to the empty convex sum.*

4. *If $J \subseteq I$ and $\{i \in I \mid p_i \neq 0\} \subseteq J$, then*

$$\mathop{\boxplus}_{i\in I} p_i \cdot x_i = \mathop{\boxplus}_{j\in J} p_j \cdot x_j$$

5. *Let $\sigma \colon I \to I$ be a permutation of the index set I. Then, we have that*

$$\mathop{\boxplus}_{i\in I} p_i \cdot x_i = \mathop{\boxplus}_{i\in I} p_{\sigma(i)} \cdot x_{\sigma(i)}$$

Proof. We write $[\Phi]$ to denote Iverson bracket, which is defined to be 1 if Φ is true and 0 otherwise.

For ① we have that

$$\begin{aligned}
\mathop{\boxplus}_{i\in I} p_i \cdot x_i &= \mathop{\boxplus}_{i\in I} p_i \cdot \left(\mathop{\boxplus}_{x\in\cup_{i\in I}\{x_i\}} [x_i = x] \cdot x \right) && \text{(Projection axiom)}\\
&= \mathop{\boxplus}_{x\in\cup_{i\in I}\{x_i\}} \left(\sum_{i\in I} p_i [x_i = x] \right) \cdot x && \text{(Barycenter axiom)}\\
&= \mathop{\boxplus}_{x\in\cup_{i\in I}\{x_i\}} \left(\sum_{x_i = x} p_i \right) \cdot x
\end{aligned}$$

② can be shown by picking a representative for each equivalence class and then using ①. For ③, by [SW15, Lemma 3.4] we know that all terms $\boxplus_{i\in I} 0 \cdot x_i$ coincide. To see that they are provably equivalent to the empty convex sum, observe that

$$\begin{aligned}
\mathop{\boxplus}_{i\in I} 0 \cdot x_i &= \mathop{\boxplus}_{i\in I} 0 \cdot \left(\mathop{\boxplus}_{j\in\emptyset} p_j \cdot y_j \right) && \text{([SW15, Lemma 3.4])}\\
&= \mathop{\boxplus}_{j\in\emptyset} 0 \cdot y_j && \text{(Barycenter axiom)}
\end{aligned}$$

Finally, ④ follows from [SW15, Lemma 3.4], while ⑤ was proved in [Dob08, Proposition 3.1]. □

Lemma 4.2.4. *Let I, J be finite index sets, $\{p_i\}_{i\in I}$, $\{q_{i,j}\}_{(i,j)\in I\times J}$ and $\{x_{i,j}\}_{(i,j)\in I\times J}$*

indexed collections such that for all $i \in I$ and $j \in J$, $p_i, q_{i,j} \in [0,1]$ and $x_{i,j} \in X$. If X carries PCA *structure, then:*

$$\bigboxplus_{i\in I} p_i \cdot \left(\bigboxplus_{j\in J} q_{i,j} \cdot x_{i,j} \right) = \bigboxplus_{(i,j)\in I\times J} p_i q_{i,j} \cdot x_{i,j}$$

Proof.

$$\begin{aligned}
\bigboxplus_{i\in I} p_i \cdot \left(\bigboxplus_{j\in J} q_{i,j} \cdot x_{i,j} \right) &= \bigboxplus_{i\in I} p_i \cdot \left(\bigboxplus_{(k,j)\in\{i\}\times J} q_{k,j} \cdot x_{k,j} \right) \\
&= \bigboxplus_{i\in I} p_i \cdot \left(\bigboxplus_{(k,j)\in I\times J} [k=i] q_{k,j} \cdot x_{k,j} \right) && \text{(Proposition 4.2.3)} \\
&= \bigboxplus_{(k,j)\in I\times J} \left(\sum_{i\in I} p_i [k=i] q_{k,j} \right) \cdot x_{k,j} && \text{(Barycenter axiom)} \\
&= \bigboxplus_{(k,j)\in I\times J} p_k q_{k,j} \cdot x_{k,j} \\
&= \bigboxplus_{(i,j)\in I\times J} p_i q_{i,j} \cdot x_{i,j} && \text{(Renaming indices)}
\end{aligned}$$

□

Lemma 4.2.5. *Let I be a finite index set, $\{p_i\}_{i\in I}$ and $\{q_i\}_{i\in I}$ indexed collections such that $p_i, q_i \in [0,1]$ for all $i \in I$, $\sum_{i\in I} p_i + \sum_{i\in I} q_i \leq 1$ and let $\{x_i\}_{i\in I}$ and $\{y_i\}_{i\in I}$ indexed collection such that $x_i, y_i \in X$ for all $i \in I$. If X carries* PCA *structure, then:*

$$\left(\bigboxplus_{i\in I} p_i \cdot x_i \right) \boxplus \left(\bigboxplus_{i\in I} q_i \cdot y_i \right) = \bigboxplus_{i\in I} (p_i + q_i) \cdot \left(\frac{p_i}{p_i+q_i} \cdot x_i \boxplus \frac{q_i}{p_i+q_i} \cdot y_i \right)$$

Proof. Let $J = \{0,1\}$. Define indexed collections $\{r_{i,j}\}_{(i,j)\in I\times J}$ and $\{z_{i,j}\}_{(i,j)\in I\times J}$, such that $r_{i,0} = \frac{p_i}{p_i+q_i}$ and $z_{i,0} = x_i$ and $r_{i,1} = \frac{q_i}{p_i+q_i}$ and $z_{i,1} = x_i$. We have the following:

$$
\begin{aligned}
\mathop{\boxplus}\limits_{i\in I}(p_i+q_i)\cdot\left(\frac{p_i}{p_i+q_i}\cdot x_i\boxplus\frac{q_i}{p_i+q_i}\cdot y_i\right) &= \mathop{\boxplus}\limits_{i\in I}(p_i+q_i)\cdot\left(\mathop{\boxplus}\limits_{j\in J} r_{i,j}\cdot z_{i,j}\right)\\
&= \mathop{\boxplus}\limits_{(i,j)\in I\times J}(p_i+q_i)r_{i_j}\cdot z_{i,j}\\
&\qquad\qquad\text{(Lemma 4.2.4)}\\
&= \left(\mathop{\boxplus}\limits_{i\in I} p_i\cdot x_i\right)\boxplus\left(\mathop{\boxplus}\limits_{i\in I} q_i\cdot y_i\right)
\end{aligned}
$$

□

Speaking more abstractly, positive convex algebras and their homomorphisms (in the sense of homomorphisms of algebras for the signature from universal algebra) form a category, that we will call PCA. This category can be seen as a concrete presentation of an Eilenberg-Moore algebra for the subdistribution monad.

Theorem 4.2.6. *There is an isomorphism of categories between* PCA *and* $\mathsf{Set}^{\mathcal{D}}$. *Given a set* X *equipped with a positive convex algebra structure, we can define a map* $h\colon \mathcal{D}X \to X$, *given by*

$$h(\nu) = \mathop{\boxplus}\limits_{x\in\mathrm{supp}(\nu)} \nu(x)\cdot x$$

for all $\nu\in\mathcal{D}X$, *making* (X,h) *into an algebra for the monad* $\mathcal{D}$. *Equivalently, given a* $\mathcal{D}$*-algebra* (X,h), *one can define*

$$\mathop{\boxplus}\limits_{i\in I} p_i\cdot x_i = h\left(\sum_{i\in I} p_i\cdot\delta_{x_i}\right)$$

for all finite I *and indexed collections* $\{p_i\}_{i\in I}$, $\{x_i\}_{i\in I}$, *such that* $\sum_{i\in I} p_i \le 1$ *and for all* $i\in I$, $x_i\in X$. *This equips the set* X *with a positive convex algebra structure.*

Proof. See [Jac10, Theorem 4] or [Dob08, Proposition 5.3]. □

Moreover, PCA as a category enjoys the following property:

Theorem 4.2.7 ([SW15])**.** *In* PCA *finitely presented and finitely generated objects coincide.*

4.2.6 Rational fixpoint

The completeness claim presented in this chapter will rely on the universal property of the rational fixpoint [AMV06; Mil10], which provides a convenient notion of a domain representing *finite* behaviours of structured transition systems, by relying on the theory of locally finitely presentable categories.

Let $\mathcal{B}\colon \mathcal{C} \to \mathcal{C}$ be a finitary functor. We will write $\mathsf{Coalg}_{\mathsf{fp}}\,\mathcal{B}$ for the subcategory of $\mathsf{Coalg}\,\mathcal{B}$ consisting only of $\mathcal{B}$-coalgebras with finitely presentable carrier. The *rational fixpoint* is defined as

$$(\rho\mathcal{B}, r) = \operatorname{colim}(\mathsf{Coalg}_{\mathsf{fp}}\,\mathcal{B} \hookrightarrow \mathsf{Coalg}\,\mathcal{B})$$

In other words, $(\rho\mathcal{B}, r)$ is colimit of the inclusion functor from the subcategory of coalgebras with finitely presentable carriers. We call it a fixpoint, as the map $r\colon \rho\mathcal{B} \to \mathcal{B}(\rho\mathcal{B})$ is an isomorphism [AMV06].

Under some restrictions on underlying category and the type functor of coalgebras, we have the following result:

Theorem 4.2.8 ([MPW20, Corollary 3.10, Theorem 3.12])**.** *If finitely presentable and finitely generated objects coincide in* $\mathcal{C}$ *and* $\mathcal{B}\colon \mathcal{C} \to \mathcal{C}$ *is a finitary endofunctor preserving non-empty monomorphisms, then rational fixpoint is fully abstract, that is,* $(\rho B, r)$ *is a subcoalgebra of the final coalgebra* $(\mathcal{B}, t)$*.*

The requirement of preserving non-empty monomorphisms is quite weak and is satisfied by any lifting of a Set endofunctor to the category of Eilenberg-Moore algebras.

Lemma 4.2.9. *Let* $H\colon \mathsf{Set} \to \mathsf{Set}$ *be an endofunctor and let* $\mathbf{T}$ *be a finitary monad on* Set*. Then, the lifting* $\overline{H}\colon \mathsf{Set}^{\mathbf{T}} \to \mathsf{Set}^{\mathbf{T}}$ *preserves non-empty monomorphisms.*

Proof. Follows from [Gum00, Corollary 3.16] and [MPW20, Lemma 2.4]. □

4.3 Operational semantics

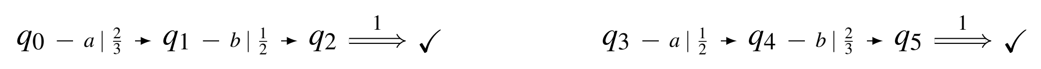
In this section, we begin by describing a coalgebraic approach to modelling the probabilistic language semantics of GPTS. Building on this, we introduce an operational semantics for PRE, drawing inspiration from Antimirov's partial derivatives for NFAs [Ant96].

4.3.1 Language semantics of GPTS

Let $\mathcal{F}\colon \mathsf{Set} \to \mathsf{Set}$ be an endofunctor given by $\mathcal{F} = \{\checkmark\} + A \times (-)$. GPTS are precisely $\mathcal{DF}$-coalgebras, that is pairs (X, β), where X is a set of states and $\beta\colon X \to \mathcal{D}(\{\checkmark\} + A \times X)$ is a transition structure. Because of this, we will interchangeably use terms "$\mathcal{DF}$-coalgebra" and "GPTS".

The functor $\mathcal{DF}$ admits a final coalgebra, but unfortunately it is not carried by the set of probabilistic languages, that is $[0,1]^{A^*}$, because the canonical semantics of $\mathcal{DF}$-coalgebras happens to correspond to the more restrictive notion of probabilistic bisimilarity (also known as Larsen-Skou bisimilarity [LS91]). Probabilistic bisimilarity is a branching-time notion of equivalence, requiring observable behaviour of compared states to be equivalent at every step, while probabilistic language equivalence is a more liberal notion comparing sequences of observable behaviour. In general, if two states are bisimilar, then they are language equivalent, but the converse does not hold.

Example 4.3.1. Consider the following GPTS:



States q_0 and q_3 are language equivalent because they both accept the string *ab* with the probability $\frac{1}{3}$, but are not bisimilar, because the state q_0 can make a transition with the probability $\frac{2}{3}$, while q_3 can perform an a transition with probability $\frac{1}{2}$.

A similar situation happens when looking at nondeterministic automata through the lenses of universal coalgebra, where again the canonical notion of equivalence is the one of bisimilarity. A known remedy is the powerset construction from classic automata theory, which converts a nondeterministic automaton to a deterministic automaton, whose states are sets of states of the original nondeterministic automaton we have started from. In such a case, the nondeterministic branching structure is factored into the state space of the determinised automaton. The language of an arbitrary state of the nondeterministic automaton corresponds to the language of the singleton set containing that state in the determinised automaton.

Generalised determinisation extends the above idea to HT-coalgebras, where $T\colon \mathsf{Set} \to \mathsf{Set}$ is an underlying functor of a finitary monad $\mathbf{T}$ and $H\colon \mathsf{Set} \to \mathsf{Set}$ is a functor that can be lifted to category $\mathsf{Set}^{\mathbf{T}}$ of $\mathbf{T}$-algebras. Generalised determinisation provides a uniform treatment of language semantics of variety of transition systems, where the final H-coalgebra provides a notion of language. Unfortunately, $\mathcal{DF}$-coalgebras do not fit immediately to this picture. Luckily, each such $\mathcal{DF}$-coalgebra can be seen as a special case of a more general kind of transition system, known as reactive probabilistic transition systems (RPTS) [GSS95] or Rabin probabilistic automata [Rab63].

RPTS can be intuitively viewed as a probabilistic counterpart of nondeterministic automata and they can be determinised to obtain probabilistic language semantics. In an RPTS, each state x is mapped to a pair $\langle o_x, n_x \rangle$, where $o \in [0,1]$ is the acceptance probability of state x and $n_x\colon A \to \mathcal{D}(X)$ is the next-state function, which takes a letter $a \in A$ and returns the subprobability distribution over successor states. Formally speaking, let $\mathcal{G}\colon \mathsf{Set} \to \mathsf{Set}$ be an endofunctor $\mathcal{G} = [0,1] \times (-)^A$. RPTS are precisely $\mathcal{GD}$-coalgebras, that is pairs (X, β), where X is a set of states and $\beta\colon X \to [0,1] \times \mathcal{D}(X)^A$ is a transition function. Following the convention outlined before, we will use terms "$\mathcal{GD}$-coalgebras" and "RPTS" interchangeably.

$\mathcal{GD}$-coalgebras fit into framework of generalised determinisation [SS11]. In particular, there exists a distributive law $\rho\colon \mathcal{DG} \Rightarrow \mathcal{GD}$ of the monad $\mathcal{D}$ over the functor $\mathcal{G}$ that allows to lift $\mathcal{G}\colon \mathsf{Set} \to \mathsf{Set}$ to $\overline{\mathcal{G}}\colon \mathsf{PCA} \to \mathsf{PCA}$. Speaking in concrete terms, if the set X is equipped with a convex sum operation $\boxplus_{i \in I} p_i \cdot (-)$, then so is $[0,1] \times X^A$. Let $\{\langle o_i, t_i \rangle\}_{i \in I}$ be an indexed collection of elements of $[0,1] \times X^A$.

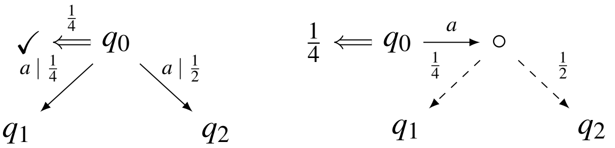
Then, we can define

$$\mathop{\boxplus}_{i\in I} p_i \cdot \langle o_i, t_i \rangle = \left\langle \sum_{i\in I} p_i \cdot o_i, \lambda a. \mathop{\boxplus}_{i\in I} p_i \cdot t_i(a) \right\rangle \tag{4.2}$$

The final coalgebra for the functor $\mathcal{G}$ is precisely carried by the set $[0,1]$ of probabilistic languages.

In order to talk about language semantics of $\mathcal{DF}$-coalgebras, we first provide an informal intuition that each $\mathcal{DF}$-coalgebra can be seen as a special case of a $\mathcal{GD}$-coalgebra.

Example 4.3.2. Fragment of a GPTS (on the left) and of the corresponding RPTS (on the right).



In the corresponding RPTS state q_0 accepts with probability $\frac{1}{4}$ and given input a it transitions to subprobability distribution that has $\frac{1}{4}$ probability of going to to q_1 and $\frac{1}{2}$ probability of going to q_2.

We can make the above intuition formal. Let X be a set, and let $\zeta \in \mathcal{DF}X$. Define a function $\gamma_X \colon \mathcal{DF}X \to \mathcal{GD}X$, given by

$$\gamma_X(\zeta) = \langle \zeta(\checkmark), \lambda a. \lambda x. \zeta(a,x) \rangle$$

Such functions define components of the natural transformation.

Proposition 4.3.3. *[SS11] $\gamma \colon \mathcal{DF} \Rightarrow \mathcal{GD}$ is a natural transformation with injective components.*

We now have all ingredients to specify language semantics of $\mathcal{DF}$-coalgebras. Given a $\mathcal{DF}$-coalgebra (X, β), one can use the natural transformation γ and obtain $\mathcal{GD}$-coalgebra $(X, \gamma_X \circ \beta)$. Since $\mathcal{G}$ can be lifted to PCA, we can obtain $\mathcal{G}$-coalgebra $\big(\mathcal{D}X, (\gamma_X \circ \beta)^\sharp\big)$. Note that this coalgebra carries an additional algebra structure and its transition map is a PCA homomorphism, thus making $\big((\mathcal{D}X,\mu_X),(\gamma_X\circ\beta)^\sharp\big)$ into a $\overline{\mathcal{G}}$-coalgebra. The resulting language semantics of (X,β) are given by the map $\mathsf{Lang}_{(X,\beta)}\colon X\to[0,1]^A$ explicitly given by

$$\mathsf{Lang}_{(X,\beta)}=\mathsf{beh}_{(\gamma_X\circ\beta)^\sharp}\circ\eta_X$$

where $\eta\colon X\to\mathcal{D}X$ is a unit of the monad $\mathcal{D}$ taking each state $x\in X$ to its Dirac $\delta_x\in\mathcal{D}X$.

This can be summarised by the following commutative diagram:

$$\begin{tikzcd}
X \arrow[r, "\eta_X"] \arrow[d, "\beta"] & \mathcal{D}X \arrow[r, dashed, "\mathsf{beh}_{(\gamma_X\circ\beta)^\sharp}"] \arrow[ddl, dashed, "(\gamma_X\circ\beta)^\sharp"] & {[0,1]^{A^*}} \arrow[dd, "t"] \\
\mathcal{D}\mathcal{F}X \arrow[d, "\gamma_X"] & & \\
\mathcal{G}\mathcal{D}X \arrow[rr, dashed, "\mathcal{G}\mathsf{beh}_{(\gamma_X\circ\beta)^\sharp}"] & & \mathcal{G}\left([0,1]^{A^*}\right)
\end{tikzcd}$$

The language semantics defined above coincide with the explicit definition of Lang we gave in Equation (4.1) (this is a consequence of a result in [SS11]).

Moreover, the natural transformation $\gamma\colon\mathcal{D}\mathcal{F}\Rightarrow\mathcal{G}\mathcal{D}$ interacts well with the distributive law $\rho\colon\mathcal{D}\mathcal{G}\Rightarrow\mathcal{G}\mathcal{D}$, making the following diagram commute:

$$\begin{tikzcd}
\mathcal{D}^2\mathcal{F} \arrow[rr, "\mu_{\mathcal{G}}"] \arrow[d, "\mathcal{D}\gamma"'] & & \mathcal{D}\mathcal{F} \arrow[d, "\gamma"] \\
\mathcal{D}\mathcal{G}\mathcal{D} \arrow[r, "\rho_{\mathcal{D}}"'] & \mathcal{G}\mathcal{D}^2 \arrow[r, "\mathcal{G}\mu"'] & \mathcal{G}\mathcal{D}
\end{tikzcd}$$

The above is a consequence of $\gamma\colon\mathcal{D}\mathcal{F}\Rightarrow\mathcal{G}\mathcal{D}$ being so-called extension law – for more details see [JSS15, Section 7.2]. Using this fact, we can show the following:

Proposition 4.3.4. *For any $\mathcal{D}\mathcal{F}$-coalgebra (X,β), the following diagram commutes:*

$$\begin{tikzcd}
\mathcal{D}X \arrow[r, "\mathcal{D}\beta"] \arrow[rrr, bend left, "(\gamma_X\circ\beta)^\sharp"] & \mathcal{D}^2\mathcal{F}X \arrow[r, "\mu_{\mathcal{F}X}"] & \mathcal{D}\mathcal{F}X \arrow[r, "\gamma_X"] & \mathcal{G}\mathcal{D}X
\end{tikzcd}$$

Proof. We first argue that $(\gamma_X \circ \beta)^\sharp \circ \eta_X = \gamma_X \circ \mu_{\mathcal{F}X} \circ \mathcal{D}\beta \circ \eta_X$

$$\begin{aligned} \eta_X \circ \mathcal{D}\beta \circ \mu_{\mathcal{F}X} \circ \gamma_X &= \beta \circ \eta_{\mathcal{D}\mathcal{F}X} \circ \mu_{\mathcal{F}X} \circ \gamma_X && (\eta \text{ is natural}) \\ &= \beta \circ \gamma_X && (\text{Monad laws}) \\ &= (\beta \circ \gamma_X)^\sharp \circ \eta_X && (\text{Kleisli extension}) \end{aligned}$$

Then, we argue that $\gamma_X \circ \mu_{\mathcal{F}X} \circ \mathcal{D}\beta$ is a PCA homomorphism from the free PCA (X, μ_X) to $\overline{\mathcal{G}}(X, \mu_X)$, by checking the commutativity of the diagram below.

$$\begin{tikzcd}
\mathcal{D}^2X \arrow[r, "\mathcal{D}^2\beta"] \arrow[dd, "\mu_X"'] & \mathcal{D}^3\mathcal{F}X \arrow[r, "\mathcal{D}\mu_{\mathcal{F}X}"] \arrow[dd, "\mu_{\mathcal{D}\mathcal{F}X}"] & \mathcal{D}^2\mathcal{F}X \arrow[r, "\mathcal{D}\gamma_X"] \arrow[dd, "\mu_{\mathcal{D}\mathcal{F}X}"'] & \mathcal{D}\mathcal{F}\mathcal{D}X \arrow[d, "\rho_{\mathcal{D}X}"] \\
 & & & \mathcal{G}^2\mathcal{D}X \arrow[d, "\mathcal{G}\mu_X"] \\
\mathcal{D}X \arrow[r, "\mathcal{D}\beta"'] & \mathcal{D}^2\mathcal{F}X \arrow[r, "\mu_{\mathcal{F}X}"] & \mathcal{D}\mathcal{F}X \arrow[r, "\gamma_X"] & \mathcal{G}\mathcal{D}X
\end{tikzcd}$$

The left diagram commutes because μ is natural, while the middle one commutes because of μ being a multiplication map of the monad. Finally, the commutativity of the rightmost subdiagram is guaranteed by γ being an extension law (see discussion above).

Since, $\mu_{\mathcal{F}X} \circ \mathcal{D}\beta \circ \eta_X$ is a PCA homomorphism that factorises through η in the same way as $(\gamma_X \circ \beta)^\sharp$, we have that $(\gamma_X \circ \beta)^\sharp = \gamma_X \circ \mu_{\mathcal{F}X} \circ \mathcal{D}\beta$. □

4.3.2 Antimirov derivatives

We now equip PExp with a $\mathcal{D}\mathcal{F}$-coalgebra structure, that is, we define a function $\partial : \mathsf{PExp} \to \mathcal{D}(1 + A \times \mathsf{PExp})$. We refer to ∂ as the *Antimirov derivative*, as it is reminiscent of the analogous construction for regular expressions and nondeterminisic automata due to Antimirov [Ant96]. Given $a \in A$, $e, f \in \mathsf{PExp}$ and $p \in [0,1]$ we define:

$$\partial(0) = \mathbb{0} \qquad \partial(1) = \delta_{\checkmark} \qquad \partial(a) = \delta_{(a,1)}$$
$$\partial(e \oplus_p f) = p\partial(e) + (1-p)\partial(f)$$

The expression $\mathsf{0}$ is mapped to the empty subdistribution, intuitively representing a deadlock. On the other hand, the expression 1 represents immediate acceptance, that is it transitions to $\checkmark$ with probability 1. For any letter $a \in A$ in the alphabet, the expression a performs a-labelled transition to 1 with probability 1. The outgoing transitions of the probabilistic choice $e \oplus_p f$ consist of the outgoing transitions of e with probabilities scaled by p and the outgoing transitions of f scaled by $1-p$.

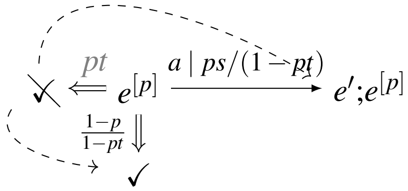
The definition of $\partial(e;f)$ is slightly more involved. We need to factor in the possibility that e may accept with some probability t, in which case the outgoing transitions of f contribute to the outgoing transitions of $e\,;f$. Formally, $\partial(e\,;f) = \partial(e) \lhd f$ where for any $f \in \mathsf{PExp}$ the operation $(- \lhd f) : \mathcal{D}\mathcal{F}\mathsf{PExp} \to \mathcal{D}\mathcal{F}\mathsf{PExp}$ is given by $(- \lhd f) = {c_f}^\sharp$, the convex extension of $c_f : 1 + A \times \mathsf{PExp} \to \mathcal{D}(1 + A \times \mathsf{PExp})$ given below on the left.

$$c_f(x) = \begin{cases} \partial(f) & x = \checkmark \\ \delta_{(a,e';f)} & x = (a,e') \end{cases}$$

$$\begin{tikzcd} \partial(f)\ \cancel{\checkmark} & e\,;f \arrow[l, Rightarrow, "t"'] \arrow[d, "a \mid s"'] \\ & e'\,;f \end{tikzcd}$$

Intuitively, c_f reroutes the transitions coming out of e: acceptance (the first case) is replaced by the behaviour of f, and the probability mass of transitioning to e' (the second case) is reassigned to $e\,;f$. A pictorial representation of the effect on the derivatives of $e\,;f$ is given above on the right. Here, we assume that $\partial(e)$ can perform a a-transition to e' with probability s; we make the same assumption in the informal descriptions of derivatives for the loops, below.

For loops, we require $\partial\left(e^{[p]}\right)$ to be the least subdistribution satisfying $\partial\left(e^{[p]}\right) = p\partial(e) \lhd e^{[p]} + (1-p)\partial(\checkmark)$. In the case when $\partial(e)(\checkmark) \neq 0$, the above becomes a fixpoint equation (as in such a case, the unrolling of the definition of $(- \lhd e^{[p]})$ involves $\partial\left(e[p]\right)$). We can define $\partial\left(e^{[p]}\right)$ as a closed form, but we need to consider two cases. If $\partial(e)(\checkmark) = 1$ and $p = 1$, then the loop body is constantly executed, but the inner expression e does not perform any labelled transitions. We identify such divergent loops with deadlock behaviour and hence $\partial(e^{[p]})(x) = 0$. Otherwise, we look at $\partial(e)$ to build $\partial\left(e^{[p]}\right)$. First, we make sure that the loop may be skipped with probability $1-p$. Next, we modify the branches that perform labelled transitions by adding $e^{[p]}$ to be executed next. The remaining mass is $p\partial(e)(\checkmark)$, the probability that we will enter the loop and immediately exit it without performing any labelled transitions. We discard this possibility and redistribute it among the remaining branches. As before, we provide an informal visual depiction of the probabilistic loop semantics below, using the same conventions as before. The crossed-out checkmark along with the dashed lines denotes the redistribution of probability mass described above.



Formally speaking, the definition of $\partial\left(e^{[p]}\right)$ can be given by the following:

$$\partial\left(e^{[p]}\right)(x) = \begin{cases} \frac{1-p}{1-p\partial(e)(\checkmark)} & x=\checkmark \\ \frac{p\partial(e)(a,e')}{1-p\partial(e)(\checkmark)} & x=(a,(e';e^{[p]})) \\ 0 & \text{otherwise} \end{cases}$$

Having defined the Antimirov transition system, one can observe that the termination operator $E(-)\colon \mathsf{PExp} \to [0,1]$ precisely captures the probability of an expression transitioning to $\checkmark$ (successful termination) when viewed as a state in the Antimirov GPTS.

Lemma 4.3.5. *For all $e \in \mathsf{PExp}$ it holds that $E(e) = \partial(e)(\checkmark)$.*

Proof. By structural induction. The base cases $E(0) = 0 = \partial(0)(\checkmark)$, $E(1) = 1 = \partial(1)(\checkmark)$ and $E(a) = 0 = \partial(a)(\checkmark)$ hold immediately. For the inductive steps, we have the following:

$\boxed{\text{Probabilistic choice}}$

$$\begin{aligned} E(e \oplus_p f) &= pE(e) + (1-p)E(f) \\ &= p\partial(e)(\checkmark) + (1-p)\partial(f)(\checkmark) \\ &= \partial(e \oplus_p f)(\checkmark) \end{aligned}$$

$\boxed{\text{Sequential compositon}}$

$$\begin{aligned} E(e\,;f) &= E(e)E(f) \\ &= \partial(e)(\checkmark)\partial(f)(\checkmark) \\ &= (\partial(e) \lhd f)(\checkmark) \\ &= \partial(e\,;f)(\checkmark) \end{aligned}$$

$\boxed{\text{Loops}}$ First, we consider the case when $\partial(e)(\checkmark) = 1$ and the loop probability is 1. By induction hypothesis, also $E(e) = 1$ and hence $E\left(e^{[1]}\right) = \partial\left(e^{[1]}\right)(\checkmark)$. Otherwise, we have the following:

$$\begin{aligned} E(e^{[p]}) &= \frac{1-p}{1-pE(e)} \\ &= \frac{1-p}{1-p\partial(e)(\checkmark)} \\ &= \partial\left(e^{[p]}\right)(\checkmark) \end{aligned}$$

□

Given an expression $e \in \mathsf{PExp}$, we write $\langle e \rangle \subseteq \mathsf{PExp}$ for the set of states reachable from e by repeatedly applying ∂. It turns out that the operational semantics of every PRE can be always described by a finite-state GPTS given by $(\langle e \rangle, \partial)$.

Lemma 4.3.6. *For all $e \in \mathsf{PExp}$, the set $\langle e \rangle$ is finite. In fact, the number of of states is bounded above by $\#(-) \colon \mathsf{PExp} \to \mathbb{N}$, where $\#(-)$ is defined recursively by:*

$$\#(0) = \#(1) = 1 \quad \#(a) = 2 \quad \#(e \oplus_p f) = \#(e) + \#(f)$$

$$\#(e\,;f) = \#(e) + \#(f) \quad \#(e^{[p]}) = \#(e) + 1$$

Proof. We adapt the analogous proof for GKAT [Sch+21].

For any $e \in \mathsf{PExp}$, let $|\langle e \rangle|$ be the cardinality of the carrier set of the least subcoalgebra of $(\mathsf{PExp}, \partial)$ containing e. We show by induction that for all $e \in \mathsf{PExp}$ it holds that $|\langle e \rangle| \leq \#(e)$.

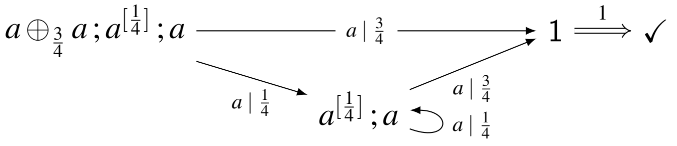
For the base cases, observe that for 0 and 1 the subcoalgebra has exactly one state. Hence, $\#(\mathsf{0}) = 1 = |\langle \mathsf{0} \rangle|$. Similarly, we have $\#(\mathsf{1}) = 1 = |\langle \mathsf{1} \rangle|$. For $a \in A$, we have two states; the initial state, which transitions with probability 1 on a to the state which outputs ✓ with probability 1.

For the inductive cases, assume that $|\langle e \rangle| \leq \#(e)$, $|\langle f \rangle| \leq \#(f)$ and $p \in [0,1]$.

- Every derivative of $e \oplus_p f$ is either a derivative of e or f and hence $|\langle e \oplus_p f \rangle| \leq |\langle e \rangle| + |\langle f \rangle| = \#(e) + \#(f) = \#(e \oplus_p f)$.
- In the case of $e\,;f$, every derivative of this expression is either a derivative of f or some derivative of e followed by f. Hence, $|\langle e\,;f \rangle| = |\langle e \rangle \times \{f\}| + |\langle f \rangle| \leq \#(e) + \#(f) = \#(e\,;f)$.
- For the probabilistic loop case, observe that every derivative of $e^{[p]}$ is a derivative of e followed by $e^{[p]}$ or it is the state that outputs ✓ with probability 1. It can be easily observed,that $|\langle e^{[p]} \rangle| \leq |\langle e \rangle| + 1 = \#(e) = \#(e^{[p]})$.

□

Example 4.3.7. Operational semantics of the expression $e = a \oplus_{\frac{3}{4}} a; a^{[\frac{1}{4}]}; a$ correspond to the following GPTS:



One can observe that the transition system above for e is isomorphic to the one starting in q_2 in Example 4.1.2.

Given the finite-state GPTS $(\langle e\rangle, \partial)$ associated with an expression $e \in \mathsf{PExp}$ we can define the language semantics of e as the probabilistic language $[\![e]\!] \in [0,1]^{A^*}$ generated by the state e in the GPTS $(\langle e\rangle, \partial)$.

4.3.3 Roadmap to soundness and completeness

The central aim of this chapter is to show that the axioms in Figure 4.1 are sound and complete to reason about probabilistic language equivalence of PRE, that is:

$$\begin{tikzcd}
e \equiv f \arrow[r, Rightarrow, shift right, "\text{Soundness}"'] & {[\![e]\!] = [\![f]\!]} \arrow[l, Rightarrow, shift right, "\text{Completeness}"']
\end{tikzcd}$$

We now sketch the roadmap on how we will prove these two results to ease the flow into the upcoming technical sections. Perhaps not surprisingly, the completeness direction is the most involved.

The heart of both arguments will rely on arguing that the semantics map $[\![-]\!] \colon \mathsf{PExp} \to [0,1]^{A^*}$ assigning a probabilistic language to each expression can be seen as the following composition of maps:

$$\begin{tikzcd}
\mathsf{PExp} \arrow[r, "{[-]}"] \arrow[rr, bend right=15, "{[\![-]\!]}"'] & \mathsf{PExp}/{\equiv} \arrow[r, "\mathsf{beh}_d"] & {[0,1]^{A^*}}
\end{tikzcd}$$

The tehcnical core of both arguments will rely on equipping $\mathsf{PExp}/{\equiv}$ with a structure of a $\mathcal{G}$-coalgebra, possesing additional well-behaved PCA structure making it into a $\overline{\mathcal{G}}$-coalgebra. In the picture above $[-] \colon \mathsf{PExp} \to \mathsf{PExp}/{\equiv}$ is a quotient map taking expressions to their equivalence class modulo the axioms of $\equiv$, while $\mathsf{beh}_d \colon \mathsf{PExp}/{\equiv} \to [0,1]^{A^*}$ is a final $\mathcal{G}$-coalgebra homomorphism taking each equivalence class to the corresponding probabilistic language. In such a case, soundness follows as a sequence of three steps:

$$e \equiv f \Rightarrow [e] = [f] \Rightarrow \mathsf{beh}_d([e]) = \mathsf{beh}_d([f]) \Rightarrow [\![e]\!] = [\![f]\!] \tag{4.3}$$

In general, obtaining the appropriate transition system structure on $\mathsf{PExp}/_{\equiv}$ needs a couple of intermediate steps, which then lead to soundness:

1. We first prove the soundness of a subset of the axioms of Figure 4.1: omitting (S0) and (D2) yields a sound inference system, which we call $\equiv_b$, with respect to a finer equivalence–probabilistic bisimilarity as defined by Larsen and Skou [LS91] (Lemma 4.4.1). As a consequence, there exists a deterministic transition system structure on the set $\mathcal{D}\mathsf{PExp}/{\equiv_b}$, such that $\mathcal{D}[-]_{\equiv_b} \colon \mathcal{D}\mathsf{PExp} \to \mathcal{D}\mathsf{PExp}/{\equiv_b}$ is a $\mathcal{G}$-coalgebra homomorphism (Lemma 4.4.2).

2. We then prove that the set of expressions modulo the bisimilarity axioms, that is $\mathsf{PExp}/_{\equiv_b}$, has the structure of a positive convex algebra $\alpha_{\equiv_b} \colon \mathcal{D}\mathsf{PExp}/{\equiv_b} \to \mathsf{PExp}/{\equiv_b}$ (Lemma 4.4.10). This allows us to flatten a distribution over equivalence classes into a single equivalence class. This proof makes use of a *fundamental theorem* decomposing expressions (Theorem 4.4.3). Additionally, we obtain that the coarser quotient $\mathsf{PExp}/{\equiv}$ also has a PCA structure (Lemma 4.4.12), that will become handy in the proof of completeness.

3. With the above result, we equip the set $\mathsf{PExp}/{\equiv_b}$ with a $\mathcal{G}$-coalgebra structure and show that the positive convex algebra structure map on $\mathsf{PExp}/{\equiv_b}$ is also a $\mathcal{G}$-coalgebra homomorphism from $\mathcal{D}\mathsf{PExp}/{\equiv_b}$ into it (Lemma 4.4.13).

4. Through a simple argument (this step encapsulates the key part of the soundness argument), we show that there exists a unique deterministic transition system structure on the coarser quotient, that is $\mathsf{PExp}/{\equiv}$, that makes further identification using axioms (S0) and (D2) (denoted $[-]_{\equiv} \colon \mathsf{PExp}/{\equiv_b} \to \mathsf{PExp}/{\equiv}$) into a $\mathcal{G}$-coalgebra homomorphism (Lemma 4.4.14). We compose all homomorphisms into a map of the type $\mathcal{D}\mathsf{PExp} \to \mathsf{PExp}/{\equiv}$ and show the correspondence of the probabilistic language of the state $[e]$ in the above mentioned $\mathcal{G}$-coalgebra with the one of δ_e in the determinisation of Antimirov GPTS (Lemma 4.4.19), thus establishing soundness (Theorem 4.4.20).

As much as our proof of soundness is not a straightforward inductive argument like in ordinary regular expressions, it immediately sets up the stage for the completeness argument. To obtain completeness we want to reverse all implications in Equation (4.3)–and they all are easily reversible except $[e] = [f] \Rightarrow \mathsf{beh}_d([e]) = \mathsf{beh}_d([f])$. To obtain this reverse implication we will need to show that beh_d is *injective*. We will do this, by showing that the (algebraically structured) coalgebra on $\mathsf{PExp}/{\equiv}$ satisfies a universal property of the rational fixpoint, that generalises the idea of regular languages representing finite-state deterministic automata.

As we will see in Section 4.5, determinising a finite-state GPTS can lead to an infinite state $\mathcal{G}$-coalgebra. Instead, we will rely on the theory of locally finitely presentable categories to characterise finite behaviour. It turns out, each determinisation of a finite-state GPTS carries a structure of a positive convex algebra, that is *free finitely generated.* Thanks to the work of Milius [Mil18] and Sokolova & Woracek [SW18] on *proper functors*, we will see that establishing that $\mathsf{PExp}/{\equiv}$ is isomorphic to the rational fixpoint boils down to showing uniqueness of homomorphisms from determinisations of finite-state GPTS. We will reduce this problem to converting GPTS to language equivalent expressions through the means of axiomatic manipulation using a procedure reminiscent of Brzozowski's equation solving method [Brz64] for converting DFAs to regular expressions. As a corollary, we will obtain an analogue of (one direction of) Kleene's theorem for GPTS and PRE. To sum up, the completeness result is obtained in 4 steps:

1. We show that the structure map of $\mathcal{G}$-coalgebra on $\mathsf{PExp}/{\equiv}$ constructed in previous step is in fact a PCA homomorphism, thus making it into a $\overline{\mathcal{G}}$-coalgebra (Lemma 4.5.1).

2. We show that determinisations of GPTS, as well as $\overline{\mathcal{G}}$-coalgebra structure on $\mathsf{PExp}/{\equiv}$, are precisely coalgebras for the functor $\hat{\mathcal{G}} \colon \mathsf{PCA} \to \mathsf{PCA}$ that is proper and a subfunctor of $\overline{\mathcal{G}}$.

3. Following a traditional pattern in completeness proofs [Sal66; Bac76; Mil84], we represent GPTS as *left-affine* systems of equations within the calculus and

show that these systems have *unique* solutions up to provable equivalence (Theorem 4.5.14).

4. We then show that these solutions are in 1-1 correspondence with well-behaved maps from $\hat{\mathcal{G}}$-coalgebras obtained from determinising finite-state GPTS into the $\hat{\mathcal{G}}$-coalgebra on $\mathsf{PExp}/{\equiv}$ (Lemma 4.5.5).

5. Finally, we use this correspondence together with an abstract categorical argument to show that $\hat{\mathcal{G}}$-coalgebra structure on $\mathsf{PExp}/{\equiv}$ has a universal property of the rational fixpoint (Corollary 4.5.19) that eventually implies injectivity of beh_d, establishing completeness (Theorem 4.5.22).

4.4 Soundness

We are now ready to execute the roadmap to soundness described in Section 4.3.3.

4.4.1 Step 1: Soundness with respect to bisimilarity

We first check that a subset of axioms generating $\equiv$ is sound with respect to bisimilarity of $\mathcal{DF}$-coalgebras, which is a coarser notion of equivalence than probabilistic language equivalence. Let $\equiv_b \subseteq \mathsf{PExp} \times \mathsf{PExp}$ denote the least congruence relation closed under the axioms on Figure 4.1 except (S0) and (D2). We will use the following notation for the quotient maps associated with $\equiv$ and $\equiv_b$:

$$\begin{tikzcd}
\mathsf{PExp} \arrow[r, "{[-]_{\equiv_b}}"'] \arrow[rr, bend left, "{[-]}"] & \mathsf{PExp}/{\equiv_b} \arrow[r, "{[-]_{\equiv}}"'] & \mathsf{PExp}/{\equiv}
\end{tikzcd}$$

A straightforward induction on the length derivation of $\equiv_b$ allows us to show that this relation is a bisimulation equivalence on $(\mathsf{PExp}, \partial)$. As mentioned before, in the case of GPTS this notion corresponds to bisimulation equivalences in the sense of Larsen and Skou [LS91]. Due to readability concerns, the proof of that result is delegated to Appendix A.

Lemma 4.4.1. *The relation* $\equiv_b \subseteq \mathsf{PExp} \times \mathsf{PExp}$ *is a bisimulation equivalence.*

As a consequence of Lemma 4.4.1 and Lemma 2.1.6, there exists a unique coalgebra structure $[\partial]_{\equiv_b} \colon \mathsf{PExp}/{\equiv_b} \to \mathcal{D}\mathcal{F}\mathsf{PExp}/{\equiv_b}$, which makes the quotient map $[-]_{\equiv_b} \colon \mathsf{PExp} \to \mathsf{PExp}/{\equiv_b}$ into a $\mathcal{D}\mathcal{F}$-coalgebra homomorphism from $(\mathsf{PExp}, \partial)$ to $(\mathsf{PExp}/{\equiv_b}, [\partial]_{\equiv_b})$. It turns out, that upon converting those $\mathcal{D}\mathcal{F}$-coalgebras to $\mathcal{G}\mathcal{D}$-coalgebras using the natural transformation $\rho \colon \mathcal{D}\mathcal{F} \to \mathcal{G}\mathcal{D}$ and determinising them, $\mathcal{D}[-]_{\equiv_b} \colon \mathcal{D}\mathsf{PExp} \to \mathcal{D}\mathsf{PExp}/{\equiv_b}$ becomes a homomorphism between the determinisations.

Lemma 4.4.2. *$\mathcal{D}[-]_{\equiv_b} \colon \mathcal{D}\mathsf{PExp} \to \mathcal{D}\mathsf{PExp}/{\equiv_b}$ is a $\mathcal{G}$-coalgebra homomorphism from $(\mathcal{D}\mathsf{PExp}, (\rho_{\mathsf{PExp}} \circ \partial)^\sharp)$ to $(\mathcal{D}\mathsf{PExp}/{\equiv_b}, (\rho_{\mathsf{PExp}/\equiv_b} \circ [\partial]_{\equiv_b})^\sharp)$. In other words, the following diagram commutes:*

$$\begin{tikzcd}
 & \mathcal{D}\mathsf{PExp} \arrow[rr, "\mathcal{D}[-]_{\equiv_b}"] \arrow[dddl, "(\gamma_{\mathsf{PExp}} \circ \partial)^\sharp"] & & \mathcal{D}\mathsf{PExp}/{\equiv_b} \arrow[dddl, "(\gamma_{\mathsf{PExp}/\equiv_b} \circ [\partial]_{\equiv_b})^\sharp"] \\
\mathsf{PExp} \arrow[ur, "\eta_{\mathsf{PExp}}"] \arrow[rr, "{[-]_{\equiv_b}}"'] \arrow[d, "\partial"'] & & \mathsf{PExp}/{\equiv_b} \arrow[ur, "\eta_{\mathsf{PExp}/\equiv_b}"] \arrow[d, "{[\partial]_{\equiv_b}}"'] & \\
\mathcal{D}\mathcal{F}\mathsf{PExp} \arrow[d, "\gamma_{\mathsf{PExp}}"'] & & \mathcal{D}\mathcal{F}\mathsf{PExp}/{\equiv_b} \arrow[d, "\gamma_{\mathsf{PExp}/\equiv_b}"'] & \\
\mathcal{G}\mathcal{D}\mathsf{PExp} \arrow[rr, "\mathcal{G}\mathcal{D}[-]_{\equiv_b}"] & & \mathcal{G}\mathcal{D}\mathsf{PExp}/{\equiv_b} &
\end{tikzcd}$$

Proof. Front face of the diagram commutes because $[-]_{\equiv_b}$ is $\mathcal{D}\mathcal{F}$-coalgebra homomorphism and because of [Rut00, Theorem 15.1]. The sides of the diagram commute because of the free-forgetful adjunction between Set and PCA. Finally, the commutativity of the square at the back of the diagram above follows from [Sil+10, Theorem 4.1]. □

4.4.2 Step 2a: Fundamental theorem

We show that every PRE is provably equivalent (modulo the axioms of $\equiv_b$) to a decomposition involving sub-expressions obtained in its small-step semantics. This property, often referred to as the fundamental theorem (in analogy with the fundamental theorem of calculus) is useful in proving soundness. In order to encode elements of $\mathcal{F}\mathsf{PExp}$ using the syntax of PExp, we define a function $\mathrm{exp} \colon \mathcal{F}\mathsf{PExp} \to$

PExp, given by $\exp(\checkmark) = 1$ and $\exp(a, e') = a; e'$ for all $a \in A$ and $e' \in \mathsf{PExp}$. To be able to syntactically express finitely supported subdistributions, we will use n-ary convex sum of elements of PExp obeying the axioms of positive convex algebras, that exists because of Proposition 4.2.2 and axioms (C1-C4). Using it, we can state the following:

Theorem 4.4.3. *For all $e \in \mathsf{PExp}$ we have that*

$$e \equiv_b \bigoplus_{d \in \operatorname{supp}(\partial(e))} \partial(e)(d) \cdot \exp(d)$$

Before we give the proof of the result above, we start by establishing a couple of intermediate results. Firstly, we show that the binary probabilistic choice satisfies the following identities:

Lemma 4.4.4. *The following facts are derivable in $\equiv_b$*

1. $e \oplus_p (f \oplus_q g) \equiv_b \left(e \oplus_{\frac{p}{1-(1-p)(1-q)}} f\right) \oplus_{1-(1-p)(1-q)} g$
2. $(e \oplus_p f) \oplus_q (g \oplus_p h) \equiv_b (e \oplus_q g) \oplus_p (f \oplus_q h)$

Proof. For ①, let $k = \frac{p}{1-(1-p)(1-q)}$ and $l = 1-(1-p)(1-q)$. We derive the following:

$$\begin{aligned}
e \oplus_p (f \oplus_q g) &\equiv_b (f \oplus_q g) \oplus_{1-p} e && (\mathsf{C3})\\
&\equiv_b (g \oplus_{1-q} f) \oplus_{1-p} e && (\mathsf{C3})\\
&\equiv_b g \oplus_{1-l} (f \oplus_{1-k} e) && (\mathsf{C4})\\
&\equiv_b (f \oplus_{1-k} e) \oplus_l g && (\mathsf{C3})\\
&\equiv_b (e \oplus_k f) \oplus_l g && (\mathsf{C3})
\end{aligned}$$

For ② we show the following:

$$(e \oplus_p f) \oplus_q (g \oplus_p h) \equiv_b e \oplus_{pq} \left(f \oplus_{\frac{1-pq}{1-pq}} (g \oplus_p h)\right) \qquad (\mathsf{C4})$$

$$\equiv_b e \oplus_{pq} \left((g \oplus_p h) \oplus_{\frac{1-q}{1-pq}} f \right) \tag{C3}$$

$$\equiv_b e \oplus_{pq} \left(g \oplus_{\frac{p(1-q)}{1-pq}} (h \oplus_{1-q} f) \right) \tag{C4}$$

$$\equiv_b e \oplus_{pq} \left(g \oplus_{\frac{p(1-q)}{1-pq}} (f \oplus_q h) \right) \tag{C3}$$

$$\equiv_b (e \oplus_q g) \oplus_p (f \oplus_q h) \tag{\textcircled{1}}$$

□

Then, we argue that any non-empty n-ary convex sum can be expressed as a binary probabilistic choice and an $(n-1)$-nary convex sum.

Lemma 4.4.5. *Let $\{p_i\}_{i \in I}$ and $\{e_i\}_{i \in I}$ be non-empty collections indexed by a finite set I, such that for all $i \in I$, $p_i \in [0,1]$ and $e_i \in \mathsf{PExp}$. For any $j \in I$ it holds that:*

$$\bigoplus_{i \in I} p_i \cdot e_i \equiv_b e_j \oplus_{p_j} \left(\bigoplus_{i \in I \setminus \{j\}} \frac{p_i}{1-p_j} \cdot e_i \right)$$

Proof. In the edge case, when $p_j = 1$ (and therefore $I = \{j\}$) we have that $\bigoplus_{i \in I} p_i \cdot e_i \equiv_b e_j$ and therefore

$$e_j \equiv_b e_j \oplus_1 0 \tag{C2}$$

$$\equiv_b e_j \oplus_{p_j} \left(\bigoplus_{i \in \emptyset} \frac{p_i}{1-p_j} \cdot e_i \right) \tag{Def. of empty n-ary convex sum}$$

$$\equiv_b e_j \oplus_{p_j} \left(\bigoplus_{i \in I \setminus \{j\}} \frac{p_i}{1-p_j} \cdot e_i \right) \tag{$I = \{j\}$}$$

Note that despite the fact that $p_j = 1$, the n-ary sum on the right is well-defined as it ranges over an empty index set and thus division by zero never happens. The remaining case, when $p_j \neq 1$ holds because of Proposition 4.2.2. □

Using the above result, we can also split the normal form used in Theorem 4.4.3 into two parts; one describing acceptance and one describing labelled transitions.

Lemma 4.4.6. *For all $e \in \mathsf{PExp}$,*

$$\bigoplus_{d \in \operatorname{supp}(\partial(e))} \partial(e)(d) \cdot \exp(d) \equiv_b 1 \oplus_{\partial(e)(\checkmark)} \left(\bigoplus_{d \in \operatorname{supp}(\partial(e)) \setminus \{\checkmark\}} \frac{\partial(e)(d)}{1 - \partial(e)(\checkmark)} \cdot \exp(d) \right)$$

Proof. If $\operatorname{supp}(\partial(e)) = \emptyset$, then

$$\begin{aligned}
\bigoplus_{d \in \operatorname{supp}(\partial(e))} \partial(e)(d) \cdot \exp(d) &\equiv_b 0 && \text{(Def. of empty } n\text{-ary convex sum)} \\
&\equiv_b 0 \oplus_1 1 && (\mathsf{C2}) \\
&\equiv_b 1 \oplus_0 0 && (\mathsf{C3}) \\
&\equiv_b 1 \oplus_{\partial(e)(\checkmark)} 0 && (\partial(e)(\checkmark) = 0) \\
&\equiv_b 1 \oplus_{\partial(e)(\checkmark)} \left(\bigoplus_{d \in \emptyset} \frac{\partial(e)(d)}{1 - \partial(e)(\checkmark)} \cdot \exp(d) \right) \\
&\equiv_b 1 \oplus_{\partial(e)(\checkmark)} \left(\bigoplus_{d \in \operatorname{supp}(\partial(e)(\checkmark)) \setminus \{\checkmark\}} \frac{\partial(e)(d)}{1 - \partial(e)(\checkmark)} \cdot \exp(d) \right)
\end{aligned}$$

The remaining case when $\operatorname{supp}(\partial(e)) \neq 0$ holds by Lemma 4.4.5 and the fact that $\exp(\checkmark) = 1$. □

Finally, we generalise the axiom ($\mathsf{D1}$) to n-ary convex sums.

Lemma 4.4.7. *Let $f \in \mathsf{PExp}$, I be a finite index set and let $\{p_i\}_{i \in I}$ and $\{e_i\}_{i \in I}$ indexed collections of probabilities and expressions respectively. Then,*

$$\left(\bigoplus_{i \in I} p_i \cdot e_i \right) ; f \equiv_b \bigoplus_{i \in I} p_i \cdot e_i ; f$$

Proof. By induction. If $I = \emptyset$, then using ($\mathsf{0S}$) we can show that

$$\left(\bigoplus_{i \in I} p_i \cdot e_i \right) ; f \equiv_b 0 ; f \equiv_b 0 \equiv_b \bigoplus_{i \in I} p_i \cdot e_i ; f$$

If there exists $j \in I$, such that $p_j = 1$, then

$$\left(\bigoplus_{i\in I} p_i \cdot e_i\right) ; f \equiv_b e_j ; f \equiv_b \left(\bigoplus_{i\in I} p_i \cdot e_i ; f\right)$$

Finally, for the induction step, we have that

$$\begin{aligned}
\left(\bigoplus_{i\in I} p_i \cdot e_i\right) ; f &\equiv_b \left(e_j \oplus_{p_j} \left(\bigoplus_{i\in I\setminus\{j\}} \frac{p_i}{1-p_j} \cdot e_i\right)\right) ; f \\
&\equiv_b e_j ; f \oplus_{p_j} \left(\bigoplus_{i\in I\setminus\{j\}} \frac{p_i}{1-p_j} \cdot e_i\right) ; f && \text{(D1)} \\
&\equiv_b e_j ; f \oplus_{p_j} \left(\bigoplus_{i\in I\setminus\{j\}} \frac{p_i}{1-p_j} \cdot e_i ; f\right) && \text{(Induction hypothesis)} \\
&\equiv_b \left(\bigoplus_{i\in I} p_i \cdot e_i ; f\right)
\end{aligned}$$

□

We now have all the ingredients to show the fundamental theorem.

Proof of Theorem 4.4.3. We proceed by the structural induction on $e \in \mathsf{PExp}$. For the base cases, we have the following:

$\boxed{e = 0}$

$$\begin{aligned}
0 &\equiv_b \bigoplus_{d\in\emptyset} \partial(0)(d) \cdot \exp(d) && \text{(Proposition 4.2.3)} \\
&\equiv_b \bigoplus_{d\in \mathrm{supp}(\partial(0))} \partial(0)(d) \cdot \exp(d) && (\mathrm{supp}(\partial(0)) = \emptyset)
\end{aligned}$$

$\boxed{e = 1}$

$$\begin{aligned}
1 &\equiv_b \exp(\checkmark) && \text{(Def. of exp)} \\
&\equiv_b \bigoplus_{d\in \mathrm{supp}(\partial(1))} \partial(1)(d) \cdot \exp(d) && (\partial(1) = \delta_{\checkmark})
\end{aligned}$$

$$\boxed{e = a}$$

$$\begin{aligned}
a &\equiv_b a\,;1 && \text{(S1)}\\
&\equiv_b \exp((a,\checkmark)) && \text{(Def. of exp)}\\
&\equiv_b \bigoplus_{d\in\mathrm{supp}(\partial(a))} \partial(a)(d)\cdot\exp(d) && (\partial(a) = \delta_{(a,\checkmark)})
\end{aligned}$$

We now move on to inductive steps.

$$\boxed{e = f \oplus_p g}$$

$$\begin{aligned}
f \oplus_p g &\equiv_b \left(\bigoplus_{d\in\mathrm{supp}(\partial(f))} \partial(f)(d)\cdot\exp(d)\right) \oplus_p \left(\bigoplus_{d\in\mathrm{supp}(\partial(g))} \partial(g)(d)\cdot\exp(g)\right)\\
&\hspace{20em}\text{(Induction hypothesis)}\\
&\equiv_b \left(\bigoplus_{d\in\mathrm{supp}(\partial(f\oplus_p g))} \partial(f)(d)\cdot\exp(d)\right) \oplus_p \left(\bigoplus_{d\in\mathrm{supp}(\partial(f\oplus_p g))} \partial(g)(d)\cdot\exp(g)\right)\\
&\hspace{20em}\text{(Proposition 4.2.3)}\\
&\equiv_b \bigoplus_{d\in\mathrm{supp}(\partial(f\oplus_p g))} (p\partial(f)(d) + (1-p)\partial(g)(d))\cdot\exp(d)\\
&\hspace{20em}\text{(Barycenter axiom)}\\
&\equiv_b \bigoplus_{d\in\mathrm{supp}(\partial(f\oplus_p g))} \partial(f\oplus_p g)(d)\cdot\exp(d) \qquad \text{(Def. of } \partial\text{)}
\end{aligned}$$

$$\boxed{e = f\,;g}$$

$$\begin{aligned}
f\,;g &\equiv_b \left(\bigoplus_{d\in\mathrm{supp}(\partial(f))} \partial(f)(d)\cdot\exp(d)\right);g && \text{(Induction hypothesis)}\\
&\equiv_b \left(1 \oplus_{\partial(f)(\checkmark)} \left(\bigoplus_{d\in\mathrm{supp}(\partial(f))\setminus\{\checkmark\}} \frac{\partial(f)(d)}{1-\partial(f)(\checkmark)}\cdot\exp(d)\right)\right);g && \text{(Lemma 4.4.6)}\\
&\equiv_b \left(1\,;g \oplus_{\partial(f)(\checkmark)} \left(\bigoplus_{d\in\mathrm{supp}(\partial(f))\setminus\{\checkmark\}} \frac{\partial(f)(d)}{1-\partial(f)(\checkmark)}\cdot\exp(d)\right);g\right) && \text{(D1)}\\
&\equiv_b g \oplus_{\partial(f)(\checkmark)} \left(\bigoplus_{d\in\mathrm{supp}(\partial(f))\setminus\{\checkmark\}} \frac{\partial(f)(d)}{1-\partial(f)(\checkmark)}\cdot\exp(d)\,;g\right)\\
&&& \text{(Lemma 4.4.7 and 1S)}
\end{aligned}$$

$$\equiv_b \left(\bigoplus_{d \in \mathrm{supp}(\partial(g))} \partial(g)(d) \cdot \exp(d) \right)$$

$$\oplus_{\partial(f)(\checkmark)} \left(\bigoplus_{d \in \mathrm{supp}(\partial(f)) \setminus \{\checkmark\}} \frac{\partial(f)(d)}{1 - \partial(f)(\checkmark)} \cdot \exp(d) \,;g \right)$$

(Induction hypothesis)

$$\equiv_b \left(\bigoplus_{d \in \mathrm{supp}(\partial(f;g))} \partial(g)(d) \cdot \exp(d) \right)$$

$$\oplus_{\partial(f)(\checkmark)} \left(\bigoplus_{d \in \mathrm{supp}(\partial(f)) \setminus \{\checkmark\}} \frac{\partial(f)(d)}{1 - \partial(f)(\checkmark)} \cdot \exp(d) \,;g \right)$$

(Proposition 4.2.3)

Now, we simplify the subexpression on the right part of the convex sum. Define $n\colon F\mathsf{PExp} \to [0,1]$ to be:

$$n(d) = \begin{cases} \partial(f)(a, f') & d = (a, f' \,; g) \\ 0 & \text{otherwise} \end{cases}$$

By applying Proposition 4.2.3 and the preceding definition, it follows that:

$$\bigoplus_{d \in \mathrm{supp}(\partial(f)) \setminus \{\checkmark\}} \frac{\partial(f)(d)}{1 - \partial(f)(\checkmark)} \cdot \exp(d) \,; g \equiv_b \bigoplus_{d \in \mathrm{supp}(\partial(f;g))} \frac{n(d)}{1 - \partial(f)(\checkmark)} \cdot \exp(d)$$

By combining this with the previous derivation and applying the barycenter axiom, we can conclude that

$$f \,; g \equiv_b \bigoplus_{d \in \mathrm{supp}(\partial(f;g))} (\partial(f)(\checkmark)\partial(g)(d) + n(d)) \cdot \exp(d)$$

Combining it with the previous derivation, using the barycenter axiom, we can show that:

$$f \,; g \equiv_b \bigoplus_{d \in \mathrm{supp}(\partial(f;g))} (\partial(f)(\checkmark)\partial(g)(d) + n(d)) \cdot \exp(d)$$

Observe that for $d = (a, f' ; g)$, we obtain

$$\partial(f)(\checkmark)\partial(g)(d) + n(d) = \partial(f)(\checkmark)\partial(g)(a, f' ; g) + \partial(f)(a, f') = \partial(f ; g)(d)$$

When $d = \checkmark$, it follows that

$$\partial(f)(\checkmark)\partial(g)(d) + n(d) = \partial(f)(\checkmark)\partial(g)(d) = \partial(f ; g)(d)$$

In all remaining cases, both functions assign the value 0 to d. Consequently, we conclude that

$$f ; g \equiv_b \bigoplus_{d \in \mathrm{supp}(\partial(f;g))} \partial(f ; g)(d) \cdot \exp(d)$$

which establishes the desired result for this case.

$\boxed{e = f^{[p]}}$

We begin by considering the case where $\partial(f)(\checkmark) = 1$ and $p = 1$.

$$\begin{aligned} f^{[p]} &\equiv_b \left(\bigoplus_{d \in \mathrm{supp}(\partial(f))} \partial(f)(d) \cdot \exp(d) \right)^{[1]} && \text{(Induction hypothesis)} \\ &\equiv_b 1^{[1]} && (\partial(f)(\checkmark) = 1) \\ &\equiv_b 0 && \text{(Div)} \\ &\equiv_b \bigoplus_{d \in \mathrm{supp}\left(\partial\left(f^{[1]}\right)\right)} \partial\left(f^{[1]}\right)(d) \cdot \exp(d) \end{aligned}$$

Otherwise, we first apply the (Tight) axiom to the loop body as follows:

$$\begin{aligned} f^{[p]} &\equiv_b \left(\bigoplus_{d \in \mathrm{supp}(\partial(f))} \partial(f)(d) \cdot \exp(d) \right)^{[p]} && \text{(Induction hypothesis)} \\ &\equiv_b \left(1 \oplus_{\partial(f)(\checkmark)} \left(\bigoplus_{d \in \mathrm{supp}(\partial(f)) \setminus \{\checkmark\}} \frac{\partial(f)(d)}{1 - \partial(f)(\checkmark)} \cdot \exp(d) \right) \right)^{[p]} && \text{(Lemma 4.4.6)} \end{aligned}$$

$$\equiv_b \left(\left(\bigoplus_{d\in\mathrm{supp}(\partial(f))\setminus\{\checkmark\}} \frac{\partial(f)(d)}{1-\partial(f)(\checkmark)}\cdot \exp(d)\right) \oplus_{1-\partial(f)(\checkmark)} 1\right)^{[p]} \tag{C3}$$

$$\equiv_b \left(\bigoplus_{d\in\mathrm{supp}(\partial(f))\setminus\{\checkmark\}} \frac{\partial(f)(d)}{1-\partial(f)(\checkmark)}\cdot \exp(d)\right)^{\left[\frac{(1-\partial(f)(\checkmark))p}{1-p\partial(f)(\checkmark)}\right]} \tag{Tight}$$

For convenience, we denote by $g^{[r]}$ the expression obtained from the preceding derivation. We proceed by applying the (Unroll) axiom.

$$g^{[r]} \equiv_b \left(\bigoplus_{d\in\mathrm{supp}(\partial(f))\setminus\{\checkmark\}} \frac{\partial(f)(d)}{1-\partial(f)(\checkmark)}\cdot \exp(d)\right) ; g^{[r]} \oplus_r 1 \tag{Unroll}$$

$$\equiv_b \left(\bigoplus_{d\in\mathrm{supp}(\partial(f))\setminus\{\checkmark\}} \frac{\partial(f)(d)}{1-\partial(f)(\checkmark)}\cdot \exp(d)\right) ; f^{[p]} \oplus_r 1 \tag{$f^{[p]} \equiv_b g^{[r]}$}$$

$$\equiv_b \left(\bigoplus_{d\in\mathrm{supp}(\partial(f))\setminus\{\checkmark\}} \frac{\partial(f)(d)}{1-\partial(f)(\checkmark)}\cdot \exp(d) ; f^{[p]}\right) \oplus_r 1 \tag{Lemma 4.4.7}$$

$$\equiv_b \left(\bigoplus_{d\in\mathrm{supp}(\partial(f))\setminus\{\checkmark\}} \frac{\partial(f)(d)}{1-\partial(f)(\checkmark)}\cdot \exp(d) ; f^{[p]}\right)$$
$$\oplus_r \left(\bigoplus_{d\in\mathrm{supp}\left(\partial\left(f^{[p]}\right)\right)} \delta_{\checkmark}(d)\cdot \exp(d)\right)$$

Next, we simplify the left-hand side of the binary convex sum. Define $n\colon F\mathsf{PExp} \to [0,1]$ as follows

$$n(d) = \begin{cases} \partial(f)(a,f') & d = \left(a, f' ; f^{[p]}\right) \\ 0 & \text{otherwise} \end{cases}$$

By applying Proposition 4.2.3 and the definition above, we obtain

$$\bigoplus_{d\in\mathrm{supp}(\partial(f))\setminus\{\checkmark\}} \frac{\partial(f)(d)}{1-\partial(f)(\checkmark)}\cdot \exp(d) ; f^{[p]} \equiv_b \bigoplus_{d\in\mathrm{supp}\left(\partial\left(f^{[p]}\right)\right)} \frac{n(d)}{1-\partial(f)(\checkmark)}\cdot \exp(d)$$

By combining the above with the previous derivation and applying the barycenter axiom, we obtain:

$$f^{[p]} \equiv_b \bigoplus_{d \in \mathrm{supp}\left(\partial\left(f^{[p]}\right)\right)} \left(\frac{pn(d)}{1-p\partial(f)(\checkmark)} + \frac{1-p\delta_{\checkmark}(d)}{1-\partial(f)(\checkmark)p}\right) \cdot \exp(d)$$

Observe that for $d = (a, f'\,; g)$, we obtain

$$\frac{pn(d)}{1-p\partial(f)(\checkmark)} + \frac{1-p\delta_{\checkmark}(d)}{1-p\partial(f)(\checkmark)} = \frac{p\partial(f)(a,f')}{1-p\partial(f)(\checkmark)} = \partial\left(f^{[p]}\right)(d)$$

When $d = \checkmark$, it follows that

$$\frac{pn(d)}{1-p\partial(f)(\checkmark)} + \frac{1-p\delta_{\checkmark}(d)}{1-p\partial(f)(\checkmark)} = \frac{1-p}{1-p\partial(f)(\checkmark)} = \partial\left(f^{[p]}\right)(d)$$

In all remaining cases, we have that

$$\frac{pn(d)}{1-p\partial(f)(\checkmark)} + \frac{1-p\delta_{\checkmark}(d)}{1-p\partial(f)(\checkmark)} = 0 = \partial\left(f^{[p]}\right)(d)$$

Thus, we obtain the following:

$$f^{[p]} \equiv_b \bigoplus_{d \in \mathrm{supp}\left(\partial\left(f^{[p]}\right)\right)} \partial\left(f^{[p]}\right)(d) \cdot \exp(d)$$

This establishes the desired result.

□

A direct corollary of the result established above is that every loop is provably equivalent to a loop whose body does not assign any probability to transitions to ✓.

Corollary 4.4.8 (Productive loop)**.** *Let* $e \in \mathsf{PExp}$ *and* $p \in [0,1]$. *We have that* $e^{[p]} \equiv_b f^{[r]}$ *for some* $f \in \mathsf{PExp}$ *and* $r \in [0,1]$*, such that* $E(f) = 0$.

Proof. If $\partial(e)(\checkmark) = 1$ and $p = 1$, then it follows that:

$$e^{[1]} \equiv_b \left(\bigoplus_{d \in \mathrm{supp}(\partial(e))} \partial(e)(d) \cdot \exp(d)\right)^{[1]} \qquad \text{(Theorem 4.4.3)}$$

$$
\begin{aligned}
&\equiv_b 1^{[1]} \\
&\equiv_b 0 && \text{(Div)} \\
&\equiv_b 0\,;0 \oplus_1 1 && \text{(0S and C2)} \\
&\equiv_b 0^{[1]} && \text{(Unique fixpoint rule and } E(0)=0\text{)}
\end{aligned}
$$

Therefore, $e^{[1]} = 0^{[1]}$. In this case, it follows that $E(0) = 0$. In the remaining cases, we obtain the following:

$$
\begin{aligned}
e^{[p]} &\equiv_b \left(\bigoplus_{d \in \mathrm{supp}(\partial(e))} \partial(e)(d) \cdot \exp(d) \right)^{[p]} && \text{(Theorem 4.4.3)} \\
&\equiv_b \left(1 \oplus_{\partial(e)(\checkmark)} \left(\bigoplus_{d \in \mathrm{supp}(\partial(e)) \setminus \{\checkmark\}} \frac{\partial(e)(d)}{1-\partial(e)(\checkmark)} \cdot \exp(d) \right) \right)^{[p]} && \text{(Lemma 4.4.6)} \\
&\equiv_b \left(\left(\bigoplus_{d \in \mathrm{supp}(\partial(e)) \setminus \{\checkmark\}} \frac{\partial(e)(d)}{1-\partial(e)(\checkmark)} \cdot \exp(d) \right) \oplus_{1-\partial(e)(\checkmark)} 1 \right)^{[p]} && \text{(C3)} \\
&\equiv_b \left(\left(\bigoplus_{(a,e') \in \mathrm{supp}(\partial(e))} \frac{\partial(e)(a,e')}{1-\partial(e)(\checkmark)} \cdot a\,;e' \right) \oplus_{1-\partial(e)(\checkmark)} 1 \right)^{[p]} && \text{(Def. of exp)} \\
&\equiv_b \left(\bigoplus_{(a,e') \in \mathrm{supp}(\partial(e))} \frac{\partial(e)(a,e')}{1-\partial(e)(\checkmark)} \cdot a\,;e' \right)^{\left[\frac{p(1-\partial(e)(\checkmark))}{1-p\partial(e)(\checkmark)}\right]} && \text{(Tight)}
\end{aligned}
$$

Observe that the body of the loop above is an n-ary probabilistic sum involving terms of the form $a\,;e'$ (where $a \in A$, $e' \in \mathsf{PExp}$), for which $E(a\,;e') = 0$. By examining the definition of the n-ary sum (Proposition 4.2.2) and the termination operator $E(-)$ (Figure 4.1), we immediately conclude that the loop body is mapped to 0 by $E(-)$, which completes the proof. □

4.4.3 Step 2b: Algebra structure

Then, we equip the set $\mathsf{PExp}/{\equiv_b}$ with a PCA structure. To do so, we first observe that as a consequence of Theorem 4.4.3, we have that $[\partial]_{\equiv_b} \colon \mathsf{PExp}/{\equiv_b} \to \mathcal{D}\mathcal{F}\mathsf{PExp}/{\equiv_b}$ is an isomorphism.

Corollary 4.4.9. *The structure map* $[\partial]_{\equiv_b} : \mathsf{PExp}/{\equiv_b} \to \mathcal{DF}\mathsf{PExp}/{\equiv_b}$ *of the quotient coalgebra* $(\mathsf{PExp}/{\equiv_b}, [\partial]_{\equiv_b})$ *is an isomorphism.*

Proof. Given $\nu \in \mathcal{DF}\mathsf{PExp}/{\equiv_b}$ define a function $[\partial]^{-1}_{\equiv_b} \colon \mathcal{DF}\mathsf{PExp}/{\equiv_b} \to \mathsf{PExp}/{\equiv_b}$ as follows:

$$[\partial]^{-1}_{\equiv_b}(\nu) = \left[\nu(\checkmark) \cdot 1 \oplus \left(\bigoplus_{(a,[e']_{\equiv_b}) \in \mathrm{supp}(\nu)} \nu(a, [e']_{\equiv_b}) \cdot a \mathbin{;} e' \right)\right]_{\equiv_b}$$

First, observe that for arbitrary $\nu \in \mathcal{DF}\mathsf{PExp}/{\equiv_b}$ we have that:

$$\begin{aligned}
&\left([\partial]_{\equiv_b} \circ [\partial]^{-1}_{\equiv_b}\right)(\nu)(\checkmark) \\
&= ([\partial]_{\equiv_b} \circ [-]_{\equiv_b}) \left(\nu(\checkmark) \cdot 1 \oplus \left(\bigoplus_{(a,[e']_{\equiv_b}) \in \mathrm{supp}(\nu)} \nu(a, [e']_{\equiv_b}) \cdot a \mathbin{;} e' \right)\right)(\checkmark) \\
&= (\mathcal{DF}[-]_{\equiv_b} \circ \partial) \left(\nu(\checkmark) \cdot 1 \oplus \left(\bigoplus_{(a,[e']_{\equiv_b}) \in \mathrm{supp}(\nu)} \nu(a, [e']_{\equiv_b}) \cdot a \mathbin{;} e' \right)\right)(\checkmark) \\
&\qquad\qquad ([-]_{\equiv} \text{ is a } \mathcal{DF}\text{-coalgebra homomorphism}) \\
&= \partial \left(\nu(\checkmark) \cdot 1 \oplus \bigoplus_{(a,[e']_{\equiv_b}) \in \mathrm{supp}(\nu)} \nu(a, [e']_{\equiv_b}) \cdot a \mathbin{;} e' \right)(\checkmark) \\
&= \nu(\checkmark) \qquad\qquad (\text{Def. of } \partial)
\end{aligned}$$

Similarly, for any $(b, [f']_{\equiv_b}) \in \mathrm{supp}(\nu)$, it follows that:

$$\begin{aligned}
&([\partial]_{\equiv_b} \circ [\partial]^{-1}_{\equiv_b})(\nu)(b, [f']_{\equiv_b}) \\
&= ([\partial]_{\equiv_b} \circ [-]_{\equiv_b}) \left(\nu(\checkmark) \cdot 1 \oplus \left(\bigoplus_{(a,[e']_{\equiv_b}) \in \mathrm{supp}(\nu)} \nu(a, [e']_{\equiv_b}) \cdot a \mathbin{;} e' \right)\right)(b, [f']_{\equiv_b}) \\
&= (\mathcal{DF}[-]_{\equiv_b} \circ \partial) \left(\nu(\checkmark) \cdot 1 \oplus \left(\bigoplus_{(a,[e']_{\equiv_b}) \in \mathrm{supp}(\nu)} \nu(a, [e']_{\equiv_b}) \cdot a \mathbin{;} e' \right)\right)(b, [f']_{\equiv_b}) \\
&\qquad\qquad ([-]_{\equiv} \text{ is a } \mathcal{DF}\text{-coalgebra homomorphism}) \\
&= \sum_{g \equiv f'} \partial \left(\nu(\checkmark) \cdot 1 \oplus \left(\bigoplus_{(a,[e']_{\equiv_b}) \in \mathrm{supp}(\nu)} \nu(a, [e']_{\equiv_b}) \cdot a \mathbin{;} e' \right)\right)(b, g) \\
&= \nu(b, [f']_{\equiv_b}) \qquad\qquad (\text{Def. of } \partial)
\end{aligned}$$

For the second part of the proof, let $e \in \mathsf{PExp}$. As a consequence of Theorem 4.4.3, it follows that:

$$\begin{aligned}
e &\equiv_b \bigoplus_{d \in \operatorname{supp}(\partial(e))} \partial(e)(d) \cdot \exp(d) && \text{(Theorem 4.4.3)} \\
&\equiv_b \partial(e)(\checkmark) \cdot 1 \oplus \left(\bigoplus_{(a,e') \in \operatorname{supp}(\partial(e))} \partial(e)(a,e') \cdot a \,; e' \right) \\
&\equiv_b \partial(e)(\checkmark) \cdot 1 \oplus \left(\bigoplus_{(a,[e']_{\equiv_b}) \in A \times \mathsf{PExp}/\equiv_b} \left(\sum_{g \equiv e'} \partial(e)(a,g) \right) \cdot a \,; e' \right) && \text{(Proposition 4.2.3)}
\end{aligned}$$

Next, observe that:

$$\begin{aligned}
&([\partial]^{-1}_{\equiv_b} \circ [\partial]_{\equiv_b})[e]_{\equiv_b} = ([\partial]^{-1}_{\equiv_b} \circ \mathcal{DF}[-]_{\equiv_b} \circ \partial)(e) \\
&\qquad\qquad \text{($[-]_{\equiv}$ is a $\mathcal{DF}$-coalgebra homomorphism)} \\
&= \left[\partial(e)(\checkmark) \cdot 1 \oplus \left(\bigoplus_{(a,[e']_{\equiv_b}) \in \operatorname{supp}((\mathcal{DF}[-]_{\equiv_b} \circ \partial)(e))} (\mathcal{DF}[-]_{\equiv_b} \circ \partial)(e)(a,[e']_{\equiv_b})) \cdot a \,; e' \right) \right]_{\equiv_b} \\
&= \left[\partial(e)(\checkmark) \cdot 1 \oplus \left(\bigoplus_{(a,[e']_{\equiv_b}) \in A \times \mathsf{PExp}/\equiv_b} (\mathcal{DF}[-]_{\equiv_b} \circ \partial)(e)(a,[e']_{\equiv_b})) \cdot a \,; e' \right) \right]_{\equiv_b} \\
&\qquad\qquad \text{(Proposition 4.2.3)} \\
&= \left[\partial(e)(\checkmark) \cdot 1 \oplus \left(\bigoplus_{(a,[e']_{\equiv_b}) \in A \times \mathsf{PExp}/\equiv_b} \left(\sum_{g \equiv e'} \partial(e)(a,g) \right) \cdot a \,; e' \right) \right]_{\equiv_b} && \text{(Def. of $[\partial]_{\equiv_b}$)} \\
&= [e]_{\equiv_b} && \text{(Derivation above)}
\end{aligned}$$

This completes the proof. □

The result above allows to define a map $\alpha_{\equiv_b} : \mathcal{D}\mathsf{PExp}/\equiv_b \to \mathsf{PExp}/\equiv_b$ as the following composition of morphisms:

$$\mathcal{D}\mathsf{PExp}/\equiv_b \xrightarrow{\mathcal{D}[\partial]_{\equiv_b}} \mathcal{DDF}\mathsf{PExp}/\equiv_b \xrightarrow{\mu_{\mathcal{F}\mathsf{PExp}/\equiv}} \mathcal{DF}\mathsf{PExp}/\equiv_b \xrightarrow{[\partial]_{\equiv_b}{}^{-1}} \mathsf{PExp}/\equiv_b$$

In fact, this map equips the set $\mathsf{PExp}/\equiv_b$ with a positive convex algebra structure.

Lemma 4.4.10. $(\mathsf{PExp}/\equiv_b, \alpha_{\equiv_b})$ *is an Eilenberg-Moore algebra for the finitely* *supported subdistribution monad.*

Proof. We first verify that $\alpha_{\equiv_b} \circ \eta_{\mathsf{PExp}/\equiv_b} = \mathsf{id}_{\mathsf{PExp}/\equiv_b}$.

$$\begin{aligned}
\alpha_{\equiv_b} \circ \eta_X &= [\partial]^{-1}_{\equiv_b} \circ \mu_{\mathcal{F}\mathsf{PExp}/\equiv_b} \circ \mathcal{D}[\partial]_{\equiv_b} \circ \eta_{\mathsf{PExp}/\equiv_b} && \text{(Def. of } \alpha_{\equiv_b}) \\
&= [\partial]^{-1}_{\equiv_b} \circ \mu_{\mathcal{F}\mathsf{PExp}/\equiv_b} \circ \eta_{\mathcal{D}\mathcal{F}\mathsf{PExp}/\equiv_b} \circ [\partial]_{\equiv_b} && (\eta \text{ is natural}) \\
&= [\partial]^{-1}_{\equiv_b} \circ [\partial]_{\equiv_b} && \text{(Monad laws)} \\
&= \mathsf{id}_{\mathsf{PExp}/\equiv_b} && \text{(Corollary 4.4.9)}
\end{aligned}$$

Then, we show that $\alpha_{\equiv_b} \circ \mathcal{D}\alpha_{\equiv_b} = \alpha_{\equiv_b} \circ \mu_{\mathsf{PExp}/\equiv_b}$

$$\begin{aligned}
\alpha_{\equiv_b} \circ \mathcal{D}\alpha_{\equiv_b} &= [\partial]^{-1}_{\equiv_b} \circ \mu_{\mathcal{F}\mathsf{PExp}/\equiv_b} \circ \mathcal{D}[\partial]_{\equiv_b} \circ \mathcal{D}[\partial]^{-1}_{\equiv_b} \circ \mathcal{D}\mu_{\mathcal{F}\mathsf{PExp}/\equiv_b} \circ \mathcal{D}^2[\partial]_{\equiv_b} \\
& && \text{(Def. of } \alpha_{\equiv_b}) \\
&= [\partial]^{-1}_{\equiv_b} \circ \mu_{\mathcal{F}\mathsf{PExp}/\equiv_b} \circ \mathcal{D}\mu_{\mathcal{F}\mathsf{PExp}/\equiv_b} \circ \mathcal{D}^2[\partial]_{\equiv_b} && \text{(Corollary 4.4.9)} \\
&= [\partial]^{-1}_{\equiv_b} \circ \mu_{\mathcal{F}\mathsf{PExp}/\equiv_b} \circ \mu_{\mathcal{D}\mathcal{F}\mathsf{PExp}/\equiv_b} \circ \mathcal{D}^2[\partial]_{\equiv_b} && \text{(Monad laws)} \\
&= [\partial]^{-1}_{\equiv_b} \circ \mu_{\mathcal{F}\mathsf{PExp}/\equiv_b} \circ \mathcal{D}[\partial]_{\equiv_b} \circ \mu_{\mathsf{PExp}/\equiv_b} && (\mu \text{ is natural}) \\
&= \alpha_{\equiv_b} \circ \mu_{\mathsf{PExp}/\equiv_b} && \text{(Def. of } \alpha_{\equiv_b})
\end{aligned}$$

□

Moreover, using the isomorphism between PCA and $\mathsf{Set}^{\mathcal{D}}$ one can calculate the concrete formula for PCA structure on $\mathsf{PExp}/\equiv_b$.

Lemma 4.4.11. *The* PCA *structure on* $\mathsf{PExp}/\equiv_b$ *is concretely given by:*

$$\boxplus_{i \in I} p_i \cdot [e_i]_{\equiv_b} = \left[\bigoplus_{i \in I} p_i \cdot e_i\right]_{\equiv_b}$$

Proof.

$$\begin{aligned}
\boxplus_{i \in I} p_i \cdot [e_i]_{\equiv_b} &= \alpha_{\equiv_b}\left(\sum_{i \in I} p_i \delta_{[e_i]_{\equiv_b}}\right) && \text{(Theorem 4.2.6)} \\
&= [\partial]^{-1}_{\equiv_b} \circ \mu_{\mathcal{F}\mathsf{PExp}/\equiv_b} \circ \mathcal{D}[\partial]_{\equiv_b}\left(\sum_{i \in I} p_i \delta_{[e_i]_{\equiv_b}}\right) && \text{(Def. of } \alpha_{\equiv_b})
\end{aligned}$$

$$
\begin{aligned}
&= [\partial]^{-1}_{\equiv_b} \circ \mu_{\mathcal{F}\mathsf{PExp}/\equiv_b}\left(\sum_{i\in I} p_i \delta_{[\partial]_{\equiv_b}([e_i]_{\equiv_b})}\right) \\
&= [\partial]^{-1}_{\equiv_b}\left(\sum_{i\in I} p_i [\partial]_{\equiv_b}([e_i]_{\equiv_b})\right) \\
&= [\partial]^{-1}_{\equiv_b}\left(\sum_{i\in I} p_i (\mathcal{D}\mathcal{F}[-]_{\equiv_b} \circ \partial)(e_i)\right) && \text{($[-]_{\equiv}$ is a $\mathcal{D}\mathcal{F}$-coalgebra homomorphism)} \\
&= \left[\left(\sum_{i\in I} p_i \partial(e_i)(\checkmark)\right) \cdot 1 \oplus \left(\bigoplus_{(a,[e']_{\equiv_b})\in A\times\mathsf{PExp}/\equiv_b} \left(\sum_{i\in I} p_i \partial(e_i)(a,[e']_{\equiv_b})\right)\right) \cdot a\,; e'\right]_{\equiv_b} \\
& && \text{(Def. of $[\partial]^{-1}_{\equiv_b}$ and Proposition 4.2.3)} \\
&= \left[\bigoplus_{i\in I} p_i \cdot \left(\partial(e_i)(\checkmark)\cdot 1 \oplus \left(\bigoplus_{(a,[e']_{\equiv_b})\in A\times\mathsf{PExp}/\equiv_b} \partial(e_i)(a,[e']_{\equiv_b})\cdot a\,; e'\right)\right)\right]_{\equiv_b} \\
& && \text{(Barycenter axiom)} \\
&= \left[\bigoplus_{i\in I} p_i \cdot \left(\partial(e_i)(\checkmark)\cdot 1 \oplus \left(\bigoplus_{(a,[e']_{\equiv_b})\in \mathsf{supp}(\partial(e_i))} \partial(e_i)(a,[e']_{\equiv_b})\cdot a\,; e'\right)\right)\right]_{\equiv_b} \\
& && \text{(Proposition 4.2.3)} \\
&= \left[\bigoplus_{i\in I} p_i \cdot e_i\right]_{\equiv_b} && \text{(Theorem 4.4.3)}
\end{aligned}
$$

□

We can also equip the coarser quotient, that is $\mathsf{PExp}/\equiv$, with a PCA structure.

Lemma 4.4.12. *The set* $\mathsf{PExp}/\equiv$ *can be equipped with a positive convex algebra structure, given by the following:*

$$\boxplus_{i\in I} p_i \cdot [e_i] = \left[\bigoplus_{i\in I} p_i \cdot e_i\right]$$

Moreover, $[-]_{\equiv} \colon \mathsf{PExp}/\equiv_b \to \mathsf{PExp}/\equiv$ *is a* PCA *homomorphism.*

Proof. The positive convex algebra structure on $\mathsf{PExp}/\equiv$ is well-defined, because $\equiv$ is a congruence and the definition on n-ary probabilistic choice (Proposition 4.2.2).

To show that $[-]_{\equiv}$ is a PCA homomorphism, we argue the following:

$$\begin{aligned}
\left[\boxplus_{i\in I} p_i \cdot [e_i]_{\equiv_b}\right]_{\equiv} &= \left[\left[\bigoplus_{i\in I} p_i \cdot e_i\right]_{\equiv_b}\right]_{\equiv} && \text{(Lemma 4.4.11)}\\
&= \left[\bigoplus_{i\in I} p_i \cdot e_i\right] \\
&= \boxplus_{i\in I} p_i \cdot [e_i] && \text{(Def. of } \mathsf{PCA} \text{ structure on } \mathsf{PExp}/\equiv)\\
&= \boxplus_{i\in I}^{n} p_i \left[[e_i]_{\equiv_b}\right]_{\equiv}
\end{aligned}$$

□

4.4.4 Step 3: Coalgebra structure

Having established the necessary algebraic structure, we move on to showing how we can equip the quotient $\mathsf{PExp}/\equiv$ with a structure of coalgebra for the functor $\mathcal{G}\colon \mathsf{Set} \to \mathsf{Set}$. First, we focus on the $\mathcal{G}$-coalgebra structure $c\colon \mathsf{PExp}/{\equiv_b} \to \mathcal{G}\mathsf{PExp}/{\equiv_b}$ on the finer quotient $\mathsf{PExp}/{\equiv_b}$, defined as the following composition of maps:

$$\begin{tikzcd}
\mathsf{PExp}/{\equiv_b} \arrow[r, "{[\partial]_{\equiv_b}}"] \arrow[rrr, bend right=10, "c"'] & \mathcal{D}\mathcal{F}\mathsf{PExp}/{\equiv_b} \arrow[r, "\gamma_{\mathsf{PExp}/\equiv_b}"] & \mathcal{G}\mathcal{D}\mathsf{PExp}/{\equiv_b} \arrow[r, "\mathcal{G}\alpha_{\equiv_b}"] & \mathcal{G}\mathsf{PExp}/{\equiv_b}
\end{tikzcd}$$

Formally, we equip a quotient $\mathsf{PExp}/{\equiv_b}$ with $\mathcal{D}\mathcal{F}$-coalgebra structure, which exists due to soundness of $\equiv_b$ with respect to bisimilarity. Then, we transform it into a $\mathcal{G}\mathcal{D}$-coalgebra using the natural transformation $\gamma\colon \mathcal{D}\mathcal{F} \Rightarrow \mathcal{G}\mathcal{D}$. Rather than determinising this coalgebra directly, we instead flatten each reachable subdistribution using the algebra map $\alpha_{\equiv_b}\colon \mathcal{D}\mathcal{F}\mathsf{PExp}/{\equiv_b} \to \mathsf{PExp}/{\equiv_b}$, thereby inducing a $\mathcal{G}$-coalgebra structure. This construction is closely related to the determinisation of $\left(\mathsf{PExp}/{\equiv_b}, \gamma_{\mathsf{PExp}/\equiv_b} \circ [\partial]_{\equiv_b}\right)$. In particular, we have the following result

Lemma 4.4.13. *The* PCA *structure map* $\alpha_{\equiv_b}\colon \mathcal{D}\mathsf{PExp}/{\equiv_b} \to \mathsf{PExp}/{\equiv_b}$ *is a* $\mathcal{G}$-

coalgebra homomorphism of the following type:

$$\alpha_{\equiv_b}\colon \left(\mathcal{D}\mathsf{PExp}/{\equiv_b}, (\gamma_{\mathsf{PExp}/\equiv_b} \circ [\partial]_{\equiv_b})^\sharp\right) \to (\mathsf{PExp}/{\equiv_b}, c)$$

Proof. We show that the following diagram commutes:

$$\begin{tikzcd}
\mathcal{D}\mathsf{PExp}/{\equiv_b} \arrow[rr, "\alpha_{\equiv_b}"] \arrow[dr, "\mathcal{D}[\partial]_{\equiv_b}"] \arrow[ddd, "(\gamma_{\mathsf{PExp}/\equiv_b}\circ[\partial]_{\equiv_b})^\sharp"'] & & \mathsf{PExp}/{\equiv_b} \arrow[d, "{[\partial]_{\equiv_b}}"'] \\
& \mathcal{D}\mathcal{D}\mathcal{F}\mathsf{PExp}/{\equiv_b} \arrow[r, "\mu_{\mathcal{F}\mathsf{PExp}/\equiv_b}"] & \mathcal{D}\mathcal{F}\mathsf{PExp}/{\equiv_b} \arrow[d, "\gamma_{\mathsf{PExp}/\equiv_0}"'] \\
& & \mathcal{G}\mathcal{D}\mathsf{PExp}/{\equiv_b} \arrow[d, "\mathcal{G}\alpha_{\equiv_b}"'] \\
\mathcal{G}\mathcal{D}\mathsf{PExp}/{\equiv_b} \arrow[rr, "\mathcal{G}\alpha_{\equiv_b}"] & & \mathcal{G}\mathsf{PExp}/{\equiv_b}
\end{tikzcd}$$

For the top right square, we have the following:

$$\begin{aligned}
[\partial]_{\equiv_b} \circ \alpha_{\equiv_b} &= [\partial]_{\equiv_b} \circ [\partial]_{\equiv_b}^{-1} \circ \mu_{\mathcal{F}\mathsf{PExp}/\equiv_b} \circ \mathcal{D}[\partial]_{\equiv_b} && (\text{Def. of } \alpha_{\equiv_b}) \\
&= \mu_{\mathcal{F}\mathsf{PExp}/\equiv_b} \circ \mathcal{D}[\partial]_{\equiv_b} && (\text{Corollary 4.4.9})
\end{aligned}$$

The commutativity of the bottom hexagon diagram follows directly from Proposition 4.3.4. □

We can utilise the aformentioned $\mathcal{G}$-coalgebra structure on $\mathsf{PExp}/{\equiv_b}$ to induce a corresponding $\mathcal{G}$-coalgebra structure on the coarser quotient $\mathsf{PExp}/{\equiv}$.

Lemma 4.4.14. *There exists a unique $\mathcal{G}$-coalgebra structure $d\colon \mathsf{PExp}/{\equiv} \to$*

$\mathcal{G}\mathsf{PExp}/{\equiv}$ *such that the following diagram commutes:*

$$\begin{tikzcd}
\mathsf{PExp} \arrow[r, "{[-]_{\equiv_b}}"] \arrow[d, "\partial"'] & \mathsf{PExp}/{\equiv_b} \arrow[r, "{[-]_{\equiv}}"'] \arrow[d, "{[\partial]_{\equiv_b}}"'] & \mathsf{PExp}/{\equiv} \arrow[ddd, dashed, "d"] \\
\mathcal{D}\mathcal{F}\mathsf{PExp} \arrow[r, "{\mathcal{D}\mathcal{F}[-]_{\equiv_b}}"'] & \mathcal{D}\mathcal{F}\mathsf{PExp}/{\equiv_b} \arrow[d, "\gamma_{\mathsf{PExp}/\equiv_b}"'] & \\
& \mathcal{G}\mathcal{D}\mathsf{PExp}/{\equiv_b} \arrow[d, "G\alpha_{\equiv_b}"'] & \\
& \mathcal{G}\mathsf{PExp}/{\equiv_b} \arrow[r, "{\mathcal{G}[-]_{\equiv}}"'] & \mathcal{G}\mathsf{PExp}/{\equiv}
\end{tikzcd}$$

The above lemma encapsulates the key step of the soundness proof. To establish the existence of $\mathcal{G}$-coalgebra structure on $\mathsf{PExp}/{\equiv}$, we will rely on diagonal fill-in property [Gum00, Lemma 3.17] to show the existence of the $\mathcal{G}$-coalgebra structure on $\mathsf{PExp}/{\equiv}$. Consequently, it suffices to demonstrate for all $e, f \in \mathsf{PExp}$ the following:

$$e \equiv f \implies \mathcal{G}[-]_{\equiv} \circ c(e) = \mathcal{G}[-]_{\equiv} \circ c(f) \tag{4.4}$$

We begin by demonstrating that the map on the right-hand side of the equation above can be expressed explicitly.

Lemma 4.4.15. *For all $o \in [0,1]$ and indexed collections $\{p_i\}_{i\in I}$, $\{a_i\}_{i\in I}$, and $\{e_i\}_{i\in I}$, such that $p_i \in [0,1]$, $a_i \in A$ and $e_i \in \mathsf{PExp}$ for all $i \in I$ and $\sum_{i\in I} p_i \leq 1-o$, we have that:*

$$(\mathcal{G}[-]_{\equiv} \circ c)\left(\left[o \cdot 1 \oplus \left(\bigoplus_{i\in I} p_i \cdot a_i \,; e_i\right)\right]_{\equiv_b}\right) = \left\langle o, \lambda a. \left[\bigoplus_{a_i = a} p_i \cdot e_i\right]\right\rangle$$

Proof. We first observe the following:

$$\begin{aligned}
c &= \mathcal{G}[-]_{\equiv} \circ \mathcal{G}\alpha_{\equiv_b} \circ \gamma_{\mathsf{PExp}/\equiv_b} \circ [\partial]_{\equiv_b} \circ [-]_{\equiv_b} && \text{(Def. of } c\text{)}\\
&= \mathcal{G}\alpha_{\equiv} \circ \mathcal{G}\mathcal{D}[-]_{\equiv} \circ \gamma_{\mathsf{PExp}/\equiv_b} \circ [\partial]_{\equiv_b} \circ [-]_{\equiv_b} && \text{(Lemma 4.4.12)}\\
&= \mathcal{G}\alpha_{\equiv} \circ \mathcal{G}\mathcal{D}[-]_{\equiv} \circ \gamma_{\mathsf{PExp}/\equiv_b} \circ \mathcal{D}\mathcal{F}[-]_{\equiv_b} \circ \partial \\
& && \text{(}[-]_{\equiv_b} \text{ is a } \mathcal{D}\mathcal{F}\text{-coalgebra homomorphism)}
\end{aligned}$$

$$
\begin{aligned}
&= \mathcal{G}\alpha_{\equiv} \circ \mathcal{G}\mathcal{D}[-]_{\equiv} \circ \mathcal{G}\mathcal{D}[-]_{\equiv_b} \circ \gamma_{\mathsf{PExp}} \circ \partial && \text{(Proposition 4.3.3)}\\
&= \mathcal{G}\alpha_{\equiv} \circ \mathcal{G}\mathcal{D}[-] \circ \gamma_{\mathsf{PExp}} \circ \partial && \text{(Definition of } [-])\\
&= \mathcal{G}\alpha_{\equiv} \circ \gamma_{\mathsf{PExp}/\equiv} \circ \mathcal{D}\mathcal{F}[-] \circ \partial && \text{(Proposition 4.3.3)}
\end{aligned}
$$

Using the above reasoning, we can obtain the following:

$$
\begin{aligned}
&\mathcal{G}\alpha_{\equiv} \circ \gamma_{\mathsf{PExp}/\equiv} \circ \mathcal{D}\mathcal{F}[-] \circ \partial \left(o \cdot 1 \oplus \left(\bigoplus_{i \in I} p_i \cdot a_i \,; e_i \right)\right)\\
&= \mathcal{G}\alpha_{\equiv} \circ \gamma_{\mathsf{PExp}/\equiv} \circ \mathcal{D}\mathcal{F}[-] \left(o\delta_{\checkmark} + \left(\sum_{i \in I} p_i \delta_{(a_i, e_i)} \right)\right) && \text{(Def. of } \partial)\\
&= \mathcal{G}\alpha_{\equiv} \circ \gamma_{\mathsf{PExp}/\equiv} \left(o\delta_{\checkmark} + \left(\sum_{i \in I} p_i \delta_{(a_i, [e_i])} \right)\right)\\
&= \mathcal{G}\alpha_{\equiv} \left\langle o, \lambda a. \sum_{a_i = a} p_i \delta_{[e_i]} \right\rangle && \text{(Def. of } \gamma)\\
&= \left\langle o, \lambda a. \left[\bigoplus_{a_i = a} p_i \cdot e \right] \right\rangle && \text{(Theorem 4.2.6)}
\end{aligned}
$$

□

Since the proof of Lemma 4.4.14 equires passing through the quotient $\mathsf{PExp}/{\equiv_b}$, which identifies the expressions modulo the axioms of the finer relation $\equiv_b$, emains only to verify the soundness of axioms (S0) and (D2) present in the finer relation $\equiv$.

Before proceeding, we first show that (S0) and (D2) can be reformulated in a simpler yet equally expressive form, which simplifies checking their soundness. Define the relation $\doteq \subseteq \mathsf{PExp} \times \mathsf{PExp}$ as follows: Let $\doteq \subseteq \mathsf{PExp} \times \mathsf{PExp}$ be a relation defined by the following

1. If $e \equiv_b f$, then $e \doteq f$

2. $a\,; (e \oplus_p f) \doteq a\,; e \oplus_p a\,; f$

3. $a\,; \mathsf{0} \doteq \mathsf{0}$

for all $e, f \in \mathsf{PExp}$, $p \in [0,1]$ and $a \in A$.

It can be easily seen that both relations are equal.

Lemma 4.4.16. *For all $e, f \in \mathsf{PExp}$, $e \equiv f$ if and only if $e \doteqdot f$*

Proof. We split the proof into two cases.

$\boxed{e \doteqdot f \implies e \equiv f}$

This implication holds immediately. If $e \equiv_b f$, then $e \equiv f$. The remaining rules stating that $a\,;(e \oplus_p f) \doteqdot a\,;e \oplus_p a\,;f$ and $a\,;\mathsf{0} \doteqdot \mathsf{0}$ are special cases of (D2) and (S0) axioms of $\equiv$ specialised to single letters of the alphabet.

$\boxed{e \equiv f \implies e \doteqdot f}$

Axioms of $\equiv$ are either axioms of $\equiv_b$, which are already contained in $\doteqdot$ or are instances of (D2)/(S0) axioms. It suffices to show that latter two are derivable in $\doteqdot$. First, we show by induction that for all $e \in \mathsf{PExp}$, $e\,;\mathsf{0} \doteqdot \mathsf{0}$.

$\boxed{e = \mathsf{0}}$

$$\begin{aligned} e\,;\mathsf{0} &\doteqdot \mathsf{0}\,;\mathsf{0} && (e = \mathsf{0}) \\ &\doteqdot \mathsf{0} && (\mathsf{0S}) \end{aligned}$$

$\boxed{e = 1}$

$$\begin{aligned} e\,;\mathsf{0} &\doteqdot 1\,;\mathsf{0} && (e = 1) \\ &\doteqdot \mathsf{0} && (\mathsf{1S}) \end{aligned}$$

$\boxed{e = a}$

$$\begin{aligned} e\,;\mathsf{0} &\doteqdot a\,;\mathsf{0} && (e = a) \\ &\doteqdot \mathsf{0} && (\text{Def. of } \doteqdot) \end{aligned}$$

$\boxed{e = f \oplus_p g}$

$$\begin{aligned} e\,;0 &\doteqdot (f \oplus_p g)\,;0 && (e = f \oplus_p g) \\ &\doteqdot f\,;0 \oplus_p g\,;0 && (\mathsf{D1}) \\ &\doteqdot 0 \oplus_p 0 && \text{(Induction hypothesis)} \\ &\doteqdot 0 && (\mathsf{C1}) \end{aligned}$$

$\boxed{e = f\,;g}$

$$\begin{aligned} e\,;0 &\doteqdot (f\,;g)\,;0 && (e = f\,;g) \\ &\doteqdot f\,;(g\,;0) && (\mathsf{S}) \\ &\doteqdot f\,;0 && \text{(Induction hypothesis)} \\ &\doteqdot 0 && \text{(Induction hypothesis)} \end{aligned}$$

$\boxed{e = f^{[p]}}$

First, by Corollary 4.4.8 we know that $f^{[p]} \doteqdot g^{[r]}$, such that $E(g) = 0$.

$$\begin{aligned} 0 &\doteqdot 0 \oplus_r 0 && (\mathsf{C1}) \\ &\doteqdot g\,;0 \oplus_r 0 && \text{(Induction hypothesis)} \end{aligned}$$

Since $E(g) = 0$, we can use unique fixpoint axiom and obtain:

$$\begin{aligned} 0 &\doteqdot g^{[r]}\,;0 && (\mathsf{Unique}) \\ &\doteqdot f^{[p]}\,;0 && \text{(Corollary 4.4.8)} \end{aligned}$$

Secondly, we show by induction that for all $e, f, g \in \mathsf{PExp}$ and $p \in [0,1]$ we have that $e\,;(f \oplus_p g) \doteqdot e\,;f \oplus_p e\,;g$.

$\boxed{e = \mathsf{0}}$

$$\begin{aligned}
e\,;(f \oplus_p g) &\doteq \mathsf{0}\,;(f \oplus_p g) && (e = \mathsf{0})\\
&\doteq \mathsf{0} && (\mathsf{0S})\\
&\doteq \mathsf{0} \oplus_p \mathsf{0} && (\mathsf{C1})\\
&\doteq \mathsf{0}\,; f \oplus_p \mathsf{0}\,; g && (\mathsf{0S})
\end{aligned}$$

$\boxed{e = \mathsf{1}}$

$$\begin{aligned}
e\,;(f \oplus_p g) &\doteq \mathsf{1}\,;(f \oplus_p g) && (e = \mathsf{1})\\
&\doteq f \oplus_p g && (\mathsf{1S})\\
&\doteq \mathsf{1}\,; f \oplus_p \mathsf{1}\,; g && (\mathsf{1S})
\end{aligned}$$

$\boxed{e = a}$

$$\begin{aligned}
e\,;(f \oplus_p g) &\doteq a\,;(f \oplus_p g) && (e = a)\\
&\doteq a\,; f \oplus_p a\,; g && (\text{Def. of } \doteq)
\end{aligned}$$

$\boxed{e = h \oplus_r i}$

$$\begin{aligned}
e\,;(f \oplus_p g) &\doteq (g \oplus_r h)\,;(f \oplus_p g) && (e = h \oplus_r i)\\
&\doteq h\,;(f \oplus_p g) \oplus_r i\,;(f \oplus_p g) && (\mathsf{D1})\\
&\doteq (h\,; f \oplus_p h\,; g) \oplus_r (i\,; f \oplus_p i\,; g) && (\text{Induction hypothesis})\\
&\doteq (h\,; f \oplus_r i\,; f) \oplus_p (h\,; g \oplus_r i\,; g) && (\text{Lemma 4.4.4})\\
&\doteq (h \oplus_r i)\,; f \oplus_p (h \oplus_r i)\,; g && (\mathsf{D1})
\end{aligned}$$

$\boxed{e = h\,;i}$

$$\begin{aligned} e\,;(f \oplus_p g) &\doteq (h\,;i)\,;(f \oplus_p g) && (e = h\,;i)\\ &\doteq h\,;(i\,;(f \oplus_p g)) && \textsf{(S)}\\ &\doteq h\,;(i\,;f \oplus_p i\,;g) && \text{(Induction hypothesis)}\\ &\doteq (h\,;(i\,;f) \oplus_p h\,;(i\,;g)) && \text{(Induction hypothesis)}\\ &\doteq (h\,;i)\,;f \oplus_p (h\,;i)\,;g && \textsf{(S)} \end{aligned}$$

$\boxed{e = h^{[r]}}$

First, by Corollary 4.4.8 we know that $h^{[r]} \doteq i^{[q]}$, such that $E(i) = 0$.

Next, we derive the following:

$$\begin{aligned} i^{[q]}\,;f \oplus_p i^{[q]}\,;g &\doteq (i\,;i^{[q]} \oplus_q 1)\,;f \oplus_p (i\,;i^{[q]} \oplus_q 1)\,;g && \textsf{(Unroll)}\\ &\doteq (i\,;i^{[q]}\,;f \oplus_q 1\,;f) \oplus_p (i\,;i^{[q]}\,;g \oplus_q 1\,;g) && \textsf{(D1)}\\ &\doteq (i\,;i^{[q]}\,;f \oplus_q f) \oplus_p (i\,;i^{[q]}\,;g \oplus_q g) && \textsf{(0S)}\\ &\doteq (i\,;i^{[q]}\,;f \oplus_p i\,;i^{[q]}\,;g) \oplus_q (f \oplus_p g) && \text{(Lemma 4.4.4)}\\ &\doteq i\,;(i^{[q]}\,;f \oplus_p i^{[q]}\,;g) \oplus_q (f \oplus_p g)\\ & && \text{(Induction hypothesis)} \end{aligned}$$

Since $E(i) = 0$, we can use (Unique) rule to derive

$$i^{[q]}\,;f \oplus_p i^{[q]}\,;g \doteq i^{[q]}\,;(f \oplus_p g)$$

Since $h^{[r]} \doteq i^{[q]}$, we have that

$$h^{[r]}\,;f \oplus_p h^{[r]}\,;g \doteq h^{[r]}\,;(f \oplus_p g)$$

This completes the proof.

□

We now have all the ingredients to obtain the desired result.

Proof of Lemma 4.4.14. Assume $e \equiv f$. Because of Lemma 4.4.16, we have that $e \doteqdot f$. We will argue that $(\mathcal{G}[-]_{\equiv} \circ c)\,([e]_{\equiv_b}) = (\mathcal{G}[-]_{\equiv} \circ c)\,([f]_{\equiv_b})$. Because of the definition of $\doteqdot$ we have only three cases to consider.

$\boxed{e \equiv_b f}$

In this case, it follows immediately that $[e]_{\equiv_b} = [f]_{\equiv_\beta}$, which implies Equation (4.4).

$\boxed{b\,;(g \oplus_p h) \doteqdot b\,; g \oplus_p b\,; h}$

Applying $\mathcal{G}[-]_{\equiv} \circ c$ and using Lemma 4.4.15 to both sides immediately gives

$$(\mathcal{G}[-]_{\equiv} \circ c)[b\,;(g \oplus_p h)]_{\equiv} = \langle 0, s\rangle = (\mathcal{G}[-]_{\equiv} \circ c)[b\,; g \oplus_p b\,; h]_{\equiv}$$

where $s\colon A \to \mathsf{PExp}/{\equiv}$ is a function given by

$$s(a) = \begin{cases} [g \oplus_p h] & \text{if } a = b \\ [0] & \text{otherwise} \end{cases}$$

$\boxed{b\,; 0 \doteqdot 0}$

Once again, we apply Lemma 4.4.15 and obtain the following:

$$(\mathcal{G}[-]_{\equiv} \circ c)[b\,;0]_{\equiv} = \langle 0, \lambda a.\,[0]\rangle = (\mathcal{G}[-]_{\equiv} \circ c)[0]_{\equiv}$$

□

As a direct corollary of Lemma 4.4.15 and the result established above, we obtain a concrete characterisation of the map d.

Corollary 4.4.17. *For all* $[o \cdot 1 \bigoplus_{i \in I} p_i \cdot a_i\,; e_i] \in \mathsf{PExp}/{\equiv}$*, we have that*

$$d\left(\left[o \cdot 1 \bigoplus_{i \in I} p_i \cdot a_i\,; e_i\right]\right) = \left\langle o, \lambda a.\left[\bigoplus_{a_i = a} p_i \cdot e_i\right]\right\rangle$$

4.4.5 Step 4: Soundness result

Through a simple finality argument, we can show that the unique $\mathcal{G}$-coalgebra homomorphism from the determinisation of the Antimirov transition system, can be viewed as the following composition of homomorphisms, which we have obtained in the earlier steps.

Lemma 4.4.18. *The following diagram commutes:*

$$\begin{tikzcd}
\mathcal{D}\mathsf{PExp} \arrow[r, "\mathcal{D}[-]_{\equiv_b}"] \arrow[rrrr, bend right, "\mathsf{beh}_{(\gamma_{\mathsf{PExp}} \circ \partial)^{\sharp}}"'] & \mathcal{D}\mathsf{PExp}/{\equiv_b} \arrow[r, "\alpha_{\equiv_b}"] & \mathsf{PExp}/{\equiv_b} \arrow[r, "{[-]_{\equiv}}"] & \mathsf{PExp}/{\equiv} \arrow[r, dashed, "\mathsf{beh}_d"] & {[0,1]^{A^*}}
\end{tikzcd}$$

Proof. By combining Lemma 4.4.2, Lemma 4.4.13, and Lemma 4.4.14, we conclude that $\mathcal{D}[-]_{\equiv_b} \circ \alpha_{\equiv_b} \circ [-]_{\equiv_b}$ is a $\mathcal{G}$-coalgebra homomorphism from $\big(\mathcal{D}\mathsf{PExp}, (\gamma_{\mathsf{PExp}} \circ \partial)^{\sharp}\big)$ to $(\mathsf{PExp}/{\equiv}, d)$. Composing with a final map beh_d into it, we obtain a $\mathcal{G}$-coalgebra homomorphism rom $\big(\mathcal{D}\mathsf{PExp}, (\gamma_{\mathsf{PExp}} \circ \partial)^{\sharp}\big)$ into the final coalgebra $\Big([0,1]^{A^*}, t\Big)$, which, by finality must be equal to $\mathsf{beh}_{(\gamma_{\mathsf{PExp}} \circ \partial)^{\sharp}}$. □

Since the language-assigning map relies on the homomorphism described above, we can show the following:

Lemma 4.4.19. *The function $[\![-]\!] : \mathsf{PExp} \to [0,1]^{A^*}$ assigning each expression to its semantics satisfies: $[\![-]\!] = \mathsf{beh}_d \circ [-]$.*

Proof.

$$\begin{aligned}
[\![-]\!] &= \dagger(\gamma_{\mathsf{PExp}} \circ \partial)^{\sharp} \circ \eta_{\mathsf{PExp}} && (\text{Def. of } [\![-]\!]) \\
&= \mathsf{beh}_d \circ [-]_{\equiv} \circ \alpha_{\equiv_b} \circ \mathcal{D}[-]_{\equiv_b} \circ \eta_{\mathsf{PExp}} && (\text{Lemma 4.4.18}) \\
&= \mathsf{beh}_d \circ [-]_{\equiv} \circ \alpha_{\equiv_b} \circ \eta_{\mathsf{PExp}/\equiv_b} \circ [-]_{\equiv_b} && (\eta \text{ is natural}) \\
&= \mathsf{beh}_d \circ [-]_{\equiv} \circ [-]_{\equiv_b} && (\alpha_{\equiv} \text{ is an Eilenberg-Moore algebra}) \\
&= \mathsf{beh}_d \circ [-] && ([-] = [-]_{\equiv} \circ [-]_{\equiv_b})
\end{aligned}$$

□

We can now immediately conclude that provably equivalent expressions are mapped to the same probabilistic languages, thus establishing soundness.

Theorem 4.4.20 (Soundness)**.** *For all* $e, f \in \mathsf{PExp}$*, if* $e \equiv f$ *then* $[\![e]\!] = [\![f]\!]$.

4.5 Completeness

The completeness proof will follow a pattern of earlier work of Jacobs [Jac06], Silva [Sil10] and Milius [Mil10] and show that the coalgebra structure on the $\mathsf{PExp}/\equiv$ is isomorphic to the subcoalgebra of an appropriate final coalgebra, ie. the unique final coalgebra homomorphism from $\mathsf{PExp}/\equiv$ is injective. The intuition comes from the coalgebraic modelling of deterministic automata studied in the work of Jacobs [Jac06]. In such a case, the final coalgebra is simply the automaton structure on the set of all formal languages, while the final homomorphism is given by the map taking a state of the automaton to a language it denotes. Restricting the attention to finite-state automata only yields *regular languages*. The set of regular languages can be equipped with an automaton structure, in a way that inclusion map into the final automaton on the set of all formal languages becomes a homomorphism. In such a case, Kozen's completeness proof of Kleene Algebra [Koz94] can be seen as showing isomorphism of the automaton of regular languages and the automaton structure on the regular expressions modulo KA axioms.

Unfortunately, we cannot immediately rely on the identical pattern. Our semantics relies on determinising GPTS, but unfortunately, determinising a finite-state GPTS can yield $\mathcal{G}$-coalgebras with infinite carriers. For example, determinising a single-state GPTS would yield a $\mathcal{G}$-coalgebra over the set of subdistributions over a singleton set, which is infinite. Luckily, all $\mathcal{G}$-coalgebras we work with have additional algebraic structure. This algebraic structure will allow us to rely on the generalisations of the concept of finiteness beyond the category of sets, offered by the theory of locally finite presentable categories [AR94]. Being equipped with those abstract lenses, one can immediately see that the earlier mentioned infinite set of all subdistributions over a singleton set is also a free PCA generated by a single element and thus finitely presentable [SW15].

In particular, we will work with a *rational fixpoint*; a generalisation of the idea of subcoalgebra of regular languages to coalgebras for finitary functors $\mathcal{B} : \mathcal{A} \to \mathcal{A}$ over locally finitely presentable categories. The rational fixpoint provides a semantic domain for the behaviour of coalgebras whose carriers are finitely presentable in the same way as regular languages provide a semantic domain for all finite-state deterministic automata. The completeness proof will essentially rely on establishing that the coalgebra structure on $\mathsf{PExp}/{\equiv}$ satisfies the universal property of the rational fixpoint.

We now move on to showing completeness through the steps described in Section 4.3.3.

4.5.1 Step 1: Algebra structure

Throughout the soundness proof, we have shown that the semantics of any expression $e \in \mathsf{PExp}$ can also be seen as the language of the state corresponding to the equivalence class $[e]$ in the deterministic transition system ($\mathcal{G}$-coalgebra) defined on the set $\mathsf{PExp}/{\equiv}$. The completeness proof will rely on establishing that this coalgebra possesses a universal property of rational fixpoint, that will imply completeness. A first step in arguing so is observing that the coalgebra $(\mathsf{PExp}/{\equiv}, d)$ interacts well with an algebra structure on $\mathsf{PExp}/{\equiv}$ defined in Lemma 4.4.12. Thanks to the fact that we can lift $\mathcal{G} \colon \mathsf{Set} \to \mathsf{Set}$ to $\overline{\mathcal{G}} \colon \mathsf{PCA} \to \mathsf{PCA}$, the set $\mathcal{G}\mathsf{PExp}/{\equiv}$ also carries the algebra structure. In such a setting, the transition function $d \colon \mathsf{PExp}/{\equiv} \to \mathcal{G}\mathsf{PExp}/{\equiv}$ becomes an algebra homomorphism.

Lemma 4.5.1. *$d \colon \mathsf{PExp}/{\equiv} \to \mathcal{G}\mathsf{PExp}/{\equiv}$ is a PCA homomorphism*

$$d \colon (\mathsf{PExp}/{\equiv}, \alpha_{\equiv}) \to \overline{\mathcal{G}}(\mathsf{PExp}/{\equiv}, \alpha_{\equiv})$$

Proof. We need to show that $d\left(\boxplus_{i \in I} p_i \cdot [e_i]\right) = \boxplus_{i \in I} p_i \cdot d([e_i])$. As a consequence of Theorem 4.4.3, we can safely assume that $e_i \equiv q^i \cdot 1 \oplus \bigoplus_{j \in J} r^i_j \cdot a^i_j \,; e^i_j$ for all $i \in I$

and hence $d([e_i]) = \left\langle q^i, \lambda a. \left[\bigoplus_{a=a^i_j} r^i_j \cdot e^i_j\right]\right\rangle$. We show the following:

$$\begin{aligned}
d\left(\boxplus_{i\in I} p_i \cdot [e_i]\right) &= d\left(\left[\bigoplus_{i\in I} p_i \cdot e_i\right]\right) && \text{(Lemma 4.4.12)}\\
&= d\left(\left[\bigoplus_{i\in I} p_i \cdot \left(q^i \cdot 1 \oplus \bigoplus_{j\in J} r^i_j \cdot a^i_j \,;\, e^i_j\right)\right]\right) && \text{(Def. of } e_i\text{)}\\
&= d\left(\bigoplus_{i\in I} p_i q^i \cdot 1 \oplus \bigoplus_{(i,j)\in I\times J} p_i r^i_j \cdot a^i_j \,;\, e^i_j\right) && \text{(Lemma 4.2.4)}\\
&= \left\langle \sum_{i\in I} p_i q^i, \lambda a. \left[\bigoplus_{a=a^i_j} p_i r^i_j \cdot e^i_j\right]\right\rangle && \text{(Corollary 4.4.17)}\\
&= \left\langle \sum_{i\in I} p_i q^i, \lambda a. \left[\bigoplus_{i\in I} p_i \cdot \left(\bigoplus_{a=a^i_j} r^i_j \cdot e^i_j\right)\right]\right\rangle && \text{(Lemma 4.2.4)}\\
&= \left\langle \sum_{i\in I} p_i q^i, \lambda a. \boxplus_{i\in I} p_i \cdot \left[\bigoplus_{a=a^i_j} r^i_j \cdot e^i_j\right]\right\rangle && \text{(Lemma 4.4.12)}\\
&= \boxplus_{i\in I} p_i \left\langle q^i, \lambda a. \left[\bigoplus_{a=a^i_j} r^i_j \cdot e^i_j\right]\right\rangle && \text{(Def. of } \overline{\mathcal{G}}\text{)}\\
&= \boxplus_{i\in I} p_i \cdot d(e_i) && \text{(Corollary 4.4.17)}
\end{aligned}$$

□

As a consequence of the result showed above, we have that

$$\left(\mathsf{PExp}/{\equiv}, d\colon\ (\mathsf{PExp}/{\equiv}, \alpha_{\equiv}) \to \overline{\mathcal{G}}\,(\mathsf{PExp}/{\equiv}, \alpha_{\equiv})\right)$$

is a $\overline{\mathcal{G}}$-coalgebra.

4.5.2 Step 2: Proper functors

It can be easily noticed that the generalised determinisation of coalgebras with finite carriers corresponds to algebraically structured coalgebras of a particular, well-behaved kind. Namely, their carriers are algebras which are free finitely generated. We write $\mathsf{Coalg}_{\mathsf{free}}\,\mathcal{B}$ for the subcategory of $\mathsf{Coalg}\,\mathcal{B}$ consisting only of $\mathcal{B}$-coalgebras with free finitely generated carriers. The recent work of Milius [Mil18] characterised *proper* functors, for which in order to establish that some $\mathcal{B}$-coalgebra is isomorphic to the rational fixpoint it will suffice to look at coalgebras with free finitely generated carriers. To put that formally:

Theorem 4.5.2 ([Mil18, Corollary 5.9]). *Let $\mathcal{B}\colon \mathsf{Set}^{\mathbf{T}} \to \mathsf{Set}^{\mathbf{T}}$ be a proper functor. Then a $\mathcal{B}$-coalgebra (R,r) is isomorphic to the rational fixpoint if (R,r) is locally finitely presentable and for every $\mathcal{B}$-coalgebra (TX,c) in $\mathsf{Coalg}_{\mathsf{free}}\mathcal{B}$ there exists a unique homomorphism from TX to R.*

As much as $\overline{\mathcal{G}}\colon \mathsf{PCA} \to \mathsf{PCA}$ is known to be proper [SW18], not every $\overline{\mathcal{G}}$-coalgebra with a free finitely generated carrier corresponds to a determinisation of some (converted) GPTS. This is simply too general, as some $\overline{\mathcal{G}}$-coalgebras with free finitely generated carriers might be determinisations of RPTS not corresponding to any GPTS. To circumvent that, instead of looking at all coalgebras for the functor $\overline{\mathcal{G}}\colon \mathsf{PCA} \to \mathsf{PCA}$, we can restrict our attention in a way that will exclude determinisations of RPTS not corresponding to any GPTS. To do so, define a functor $\hat{\mathcal{G}}\colon \mathsf{PCA} \to \mathsf{PCA}$. Given a positive convex algebra $\mathbb{X}$ defined on a set X, we define:

$$\begin{aligned}\hat{\mathcal{G}}\mathbb{X} = &\{(o,f) \in [0,1] \times X^A \mid \forall a \in A.\, \exists p_a^i \in [0,1], x_a^i \in X.\\ &f(a) = \boxplus_{i \in I} p_a^i x_a^i \text{ and } \sum_{a\in A}\sum_{i\in I} p_a^i \leq 1-o\}\end{aligned}$$

The PCA structure on $\hat{\mathcal{G}}\mathbb{X}$, as well as the action of $\hat{\mathcal{G}}$ on arrows is defined to be the same as in the case of $\mathcal{G}$. It can be immediately observed that $\hat{\mathcal{G}}$ is a subfunctor of $\overline{\mathcal{G}}$.

Remark 4.5.3. Given that $\overline{\mathcal{G}}$ preserves non-empty monomorphisms (Lemma 4.2.9) and $\hat{\mathcal{G}}$ coincides with $\overline{\mathcal{G}}$ on arrows, it follows that $\hat{\mathcal{G}}$ also preserves non-empty monomorphisms.

Whenever the algebra structure is clear from the context, we write $\hat{\mathcal{G}}X$ for $\hat{\mathcal{G}}\mathbb{X}$. Most importantly for us, thanks to the result of Sokolova and Woracek, we know that $\hat{\mathcal{G}}$ is also proper [SW18]. We can now see the following correspondence:

Lemma 4.5.4. *$\mathcal{DF}$-coalgebras with finite carriers are in one-to-one correspondence with $\hat{\mathcal{G}}$-coalgebras with free finitely generated carriers.*

$$\frac{\beta\colon X \to \mathcal{DF}X \textit{ in } \mathsf{Set}}{\xi : (\mathcal{D}X, \mu_X) \to \hat{\mathcal{G}}(\mathcal{D}X, \mu_X) \textit{ in } \mathsf{PCA}}$$

In other words, every coalgebra structure map $\xi\colon (\mathcal{D}X, \mu_X) \to \hat{\mathcal{G}}(\mathcal{D}X, \mu_X)$ is given by $\xi = (\gamma_X \circ \beta)^\sharp$ for some unique $\beta\colon X \to \mathcal{DF}X$.

Proof. Since X is finite, we can assume that $X = \{s_i\}_{i\in I}$ for some finite set I. Because of the free-forgetful adjunction between PCA and Set, we have the following correspondence of maps:

$$\frac{\zeta\colon X \to \mathcal{GD}X \text{ on Set}}{\xi\colon (\mathcal{D}X, \mu_X) \to \mathcal{G}(\mathcal{D}X, \mu_X) \text{ on } \mathsf{PCA}}$$

First, we show that for all $x \in X$, we have that $\gamma_X \circ \beta(x) \in \hat{\mathcal{G}}\mathcal{D}X$. Pick an arbitrary $x \in X$. For every $a \in A$ define $p_i^a = \beta(x)(a, s_i)$. This implies that $(\pi_2 \circ \gamma_X \circ \beta)(x)(a)(x_i) = p_i^a$ if $x_i \in S$. Therefore, we have

$$(\gamma_X \circ \beta)(x) = \left\langle \beta(x)(\checkmark), \lambda a. \sum_{i\in I} p_i^a \delta_{x_i^a} \right\rangle$$

Using the isomorphism between PCA and $\mathsf{Set}^{\mathcal{D}}$, we can reformulate the above as

$$(\gamma_X \circ \beta)(x) = \left\langle \beta(x)(\checkmark), \lambda a. \bigboxplus_{i\in I} p_i^a \cdot x_i^a \right\rangle$$

where $\boxplus$ denotes the structure map of PCA isomorphic to $(\mathcal{D}X, \mu_X)$, the free Eilenberg-Moore algebra generated by X. From the well-definedness of $\beta(x)$ it follows that:

$$\sum_{a\in A} \sum_{i\in I} p_i^a \leq 1 - \beta(x)(\checkmark)$$

which establishes that $\gamma_X \circ \beta(x) \in \hat{\mathcal{G}}\mathcal{D}X$.

For the converse, observe that every arrow $\xi\colon (\mathcal{D}X,\mu_X) \to \hat{\mathcal{G}}\mathcal{D}(\mathcal{D}X,\mu_X)$ in PCA arises as an extension of $\zeta\colon X \to \mathcal{U}\hat{\mathcal{G}}(\mathcal{D}X,\mu_X)$. Furthermore, any such ζ can be expressed as composition $\gamma_X \circ \beta$, where $\beta(x)(\checkmark) = \pi_1 \circ \zeta(x)$ and $\beta(x)(a,x') = \pi_2 \circ \zeta(x)(a)(x')$. The fact that $\zeta(x) \in \hat{\mathcal{G}}\mathcal{D}X$ ensures that $\beta(x)$ is a well-defined subdistribution. Finally, since γ_X is injective (Proposition 4.3.3), β is unique. □

Next, we argue that the coalgebra structure on $\mathsf{PExp}/{\equiv}$, which is at the centre of attention of the completeness proof, happens also to be a $\hat{\mathcal{G}}$-coalgebra.

Lemma 4.5.5. *$((\mathsf{PExp}/{\equiv},\alpha_{\equiv}),d)$ is a $\hat{\mathcal{G}}$-coalgebra.*

Before we prove the lemma above, we establish two intermediate results. First, we show the lemma allowing to establish that the coalgebra structure on $\mathsf{PExp}/{\equiv}$ defined in Section 4.4.4 is also a $\hat{\mathcal{G}}$-coalgebra.

Lemma 4.5.6. *Let (X,α) be a PCA. Then for every $\zeta \in \mathcal{D}\mathcal{F}X$ we have that $\mathcal{G}\alpha \circ \gamma_X(\zeta) \in \hat{\mathcal{G}}(X,\alpha)$.*

Proof. Let $\zeta \in \mathcal{D}\mathcal{F}X$. Recall that $\gamma_X(\zeta) = \langle \zeta(\checkmark), \lambda a.\, \lambda x.\, \zeta(a,x)\rangle$. Let $S = \{x \in X \mid \exists a \in A.\, (a,x) \in \mathrm{supp}(\zeta)\}$. Without the loss of generality, we can assume that $S = \{s_i\}_{i\in I}$ for some finite set I. For every $a \in A$ define $p_i^a = \zeta(a,s_i)$. This implies that $(\pi_2 \circ \gamma_X)(\zeta)(a)(x_i) = p_i^a$ if $x_i \in S$ or $(\pi_2 \circ \gamma_X)(\zeta)(a)(x_i) = 0$ otherwise. Therefore, we have the following:

$$(\mathcal{G}\alpha \circ \gamma_X)(\zeta) = \left\langle \zeta(\checkmark), \lambda a.\, \boxplus_{i\in I} p_i^a \cdot s_i \right\rangle$$

Finally, since $\zeta \in \mathcal{D}\mathcal{F}X$ we have that $\sum_{a\in A}\sum_{i\in I} p_i^a \leq 1 - \zeta(\checkmark)$ which proves that indeed the image of $G\alpha \circ \gamma_X$ belongs to $\hat{\mathcal{G}}(X,\alpha)$. □

Next, we show the following preservation result.

Lemma 4.5.7. *$\hat{\mathcal{G}}$-coalgebras are closed under surjective $\overline{\mathcal{G}}$-coalgebra homomorphisms.*

Proof. Let $((X,\alpha_X),\beta_X)$ be a $\hat{\mathcal{G}}$-coalgebra, $((Y,\alpha_Y),\beta_Y)$ be a $\overline{\mathcal{G}}$-coalgebra and let $e\colon X \to Y$ be a surjective $\overline{\mathcal{G}}$-homomorphism $e\colon ((X,\alpha_X),\beta_X) \to ((Y,\alpha_Y),\beta_Y)$. We need to show that for all $y \in Y$, $\beta_Y(y) \in \mathcal{U}\hat{\mathcal{G}}(Y, \alpha_Y)$. Pick an arbitrary $y \in Y$. Since $e\colon X \to Y$ is surjective, we know that $y = e(x)$. Let $\beta_X(x) = \langle o, f \rangle$. We have that:

$$\beta_Y(y) = (\beta_Y \circ e)(x) = (\mathcal{G}e \circ \beta_X)(x) = \langle o, e \circ f \rangle$$

Since $\beta_X(x) \in \hat{\mathcal{G}}X$, we have that for all $a \in A$ there exist $p^i_a \in [0,1]$ and $x^i_a \in X$ such that $f(a) = \boxplus_{i \in I} p^i_a x^i_a$ and $\sum_a \sum_{i \in I} p^i_a \leq 1 - o$. Let $e(x^i_a) = y^i_a$. For all $a \in A$, we have that:

$$(e \circ f)(a) = e\left(\boxplus_{i \in I} p^i_a \cdot x^i_a\right) = \boxplus_{i \in I} p^i_a \cdot e(x^i_a) = \boxplus_{i \in I} p^i_a \cdot y^i_a$$

Therefore, for all $a \in A$ there exist $p^i_a \in [0,1]$ and $y^i_a \in X$ such that $f(a) = \boxplus_{i \in I} p^i_a y^i_a$ and $\sum_a \sum_{i \in I} p^i_a \leq 1 - o$, which completes the proof. □

We are ready to prove the desired result.

Proof of Lemma 4.5.5. Recall that $((\mathsf{PExp}/{\equiv}, \alpha_{\equiv}), d)$ is a quotient coalgebra of

$$(\mathsf{PExp}/{\equiv_b}, \mathcal{G}\alpha_{\equiv_b} \circ \gamma_{\mathsf{PExp}/\equiv_b} \circ [\partial]_{\equiv_b})$$

which by Lemma 4.5.6 is a $\hat{\mathcal{G}}$-coalgebra. Because of Lemma 4.5.7 so is $(\mathsf{PExp}/{\equiv}, d)$. □

Moreover, when regarded as $\hat{\mathcal{G}}$-coalgebra, $((\mathsf{PExp}/{\equiv}, \alpha_{\equiv}), d)$ forms a fixpoint of the functor, meaning d is an isomorphism.

Lemma 4.5.8. *$d\colon (\mathsf{PExp}, \alpha_{\equiv}) \to \hat{\mathcal{G}}(\mathsf{PExp}, \alpha_{\equiv})$ is an isomorphism*

Proof. We construct a map $d^{-1}\colon \hat{\mathcal{G}}(\mathsf{PExp}/{\equiv}, \alpha_{\equiv}) \to (\mathsf{PExp}/{\equiv}, \alpha_{\equiv})$ and show that $d \circ d^{-1} = \mathsf{id} = d^{-1} \circ d$. Given that the forgetful functor $\mathcal{U}\colon \mathsf{PCA} \to \mathsf{Set}$ is conservative (reflects isomorphisms), this will immediately imply that d^{-1} is a PCA homomorphism. Given $\langle o, f \rangle \in \hat{\mathcal{G}}X$, such that for any $a \in A$, and $f(a) = \boxplus_{i \in I} p^i_a [e^i_a]$

for some $p_a^i \in [0,1]$, and $[e_a^i] \in \mathsf{PExp}/\equiv$, we define the following:

$$d^{-1}\langle o, f\rangle = \left[o \cdot 1 \oplus \left(\bigoplus_{(a,i)\in A\times I} p_a^i \cdot a \,; e_a^i \right) \right]$$

The expression inside the brackets is well defined as $\sum_a \sum_{i\in I} p_a^i \leq 1 - o$. To show that d^{-1} is well-defined, assume that we have $g \colon A \to \mathsf{PExp}/\equiv$, such that $f(a) = g(a) = \boxplus_{i\in I} q_a^i [h_a^i]$ for all $a \in A$ and some $q_a^i \in [0,1]$ and $[h_a^i] \in \mathsf{PExp}/\equiv$. To begin, due to the definition of the PCA structure on $\mathsf{PExp}/\equiv$ (Lemma 4.4.12), we have that:

$$\left[\bigoplus_{i\in I} p_a^i \cdot e_a^i \right] = \boxplus_{i\in I} p_a^i [e_a^i] = \boxplus_{i\in I} q_a^i [h_a^i] = \left[\bigoplus_{i\in I} q_a^i \cdot h_a^i \right]$$

Using the above, we can show:

$$\begin{aligned} d^{-1}\langle o, f\rangle &= \left[o \cdot 1 \oplus \left(\bigoplus_{(a,i)\in A\times I} p_a^i \cdot a \,; e_a^i \right) \right] \\ &= \left[o \cdot 1 \oplus \left(\bigoplus_{(a,i)\in A\times I} q_a^i \cdot a \,; h_a^i \right) \right] && (\equiv \text{ is a congruence}) \\ &= d^{-1}\langle o, g\rangle \end{aligned}$$

This establishes the well-definedness of d^{-1}. To verify one side of the isomorphism, consider the following:

$$\begin{aligned} (d \circ d^{-1})(\langle o, f\rangle) &= d\left(\left[o \cdot 1 \oplus \left(\bigoplus_{(a,i)\in A\times I} p_a^i \cdot a \,; e_a^i \right) \right] \right) && (\text{Def. of } d^{-1}) \\ &= \left\langle o, \lambda a. \left[\bigoplus_{i\in I} p_a^i \cdot e_a^i \right] \right\rangle && (\text{Corollary 4.4.17}) \\ &= \left\langle o, \boxplus_{i\in I} p_a^i \cdot [e_a^i] \right\rangle && (\text{Lemma 4.4.12}) \\ &= \langle o, f\rangle \end{aligned}$$

For the converse direction, let$[e] \in \mathsf{PExp}/{\equiv}$. By Theorem 4.4.3, we have that:

$$e \equiv o \cdot 1 \oplus \left(\bigoplus_{(a,i) \in A \times I} p_a^i \cdot a \,; e_a^i \right)$$

Next, note that:

$$\begin{aligned}
(d^{-1} \circ d)([e]) &= d^{-1} \left\langle o, \lambda a. \left[\bigoplus_{i \in I} p_a^i \cdot e_a^i \right] \right\rangle && \text{(Corollary 4.4.17)} \\
&= d^{-1} \left\langle o, \lambda a. \boxplus_{i \in I} p_a^i \cdot \left[e_a^i \right] \right\rangle && \text{(Lemma 4.4.12)} \\
&= \left[o \cdot 1 \oplus \left(\bigoplus_{(a,i) \in A \times I} p_a^i \cdot a \,; e_a^i \right) \right] && \text{(Def. of } d^{-1}\text{)} \\
&= d([e])
\end{aligned}$$

This completes the proof. □

4.5.3 Step 3: Systems of equations

In order to establish that $\mathsf{PExp}/{\equiv}$ is isomorphic to the rational fixpoint (which is the property that will eventually imply completeness), we will show the satisfaction of conditions of Theorem 4.5.2. One of the required things we need to show is that the determinisation of an arbitrary finite-state GPTS admits a unique homomorphism to the coalgebra carried by $\mathsf{PExp}/{\equiv}$. In other words, we need to convert states of an arbitrary finite-state GPTS to language equivalent expressions in a way which is unique up to the axioms of $\equiv$. This can be thought of as an abstract reformulation of one direction of the Kleene theorem to the case of PRE. To make that possible, we give a construction inspired by Brzozowski's equation solving method [Brz64] of converting a DFA to the corresponding regular expression. We start by stating the necessary definitions.

Definition 4.5.9. A left-affine system on finite non-empty set Q of unknowns is a quadruple

$$\mathcal{S} = (M\colon Q \times Q \to \mathsf{PExp}, p\colon Q \times Q \to [0,1], b\colon Q \to \mathsf{PExp}, r\colon Q \to [0,1])$$

such that for all $q, q' \in Q$, $\sum_{q' \in Q} p_{q,q'} + r_q = 1$ and $E(M_{q,q'}) = 0$.

Definition 4.5.10. Let $\equiv_c \subseteq \mathsf{PExp} \times \mathsf{PExp}$ be a congruence relation. A map $h\colon Q \to \mathsf{PExp}$ is $\equiv_b$-solution if for all $q \in Q$ we have that:

$$h(q) \equiv_b \left(\bigoplus_{q' \in Q} p_{q,q'} \cdot M_{q,q'} \, ; h(q') \right) \oplus r_q \cdot b_q$$

Definition 4.5.11. A system representing the finite-state $\mathcal{DF}$-coalgebra (X, β) is given by $\mathcal{S}(\beta) = \langle M^\beta, p^\beta, b^\beta, r^\beta \rangle$ where for all $x, x' \in X$ we have:

$$p^\beta_{x,x'} = \sum_{a' \in A} \beta(x)(a', x') \qquad r^\beta_x = 1 - \sum_{(a',x') \in \mathsf{supp}(\beta(x))} \beta(x)(a', x')$$

$$M^\beta_{x,x'} = \begin{cases} \bigoplus_{a \in A} \frac{\beta(x)(a,x')}{p^\beta_{x,x'}} \cdot a & \text{if } p^\beta_{x,x'} \neq 0 \\ \mathsf{0} & \text{otherwise} \end{cases} \qquad b^\beta_x = \begin{cases} \frac{\beta(x)(\checkmark)}{r^\beta_x} \cdot 1 & \text{if } r^\beta_x \neq 0 \\ \mathsf{0} & \text{otherwise} \end{cases}$$

The original Brzozowski's equation-solving method is purely semantic, as it crucially relies on Arden's rule [Ard61] by providing solutions up to the language equivalence to the systems of equations. As much as it would be enough for an analogue of the one direction of Kleene's theorem, for the purposes of our completeness argument, we need to argue something stronger. Namely, we show that we can uniquely solve each system purely through the means of syntactic manipulation using the axioms of $\equiv$. This is where the main complexity of the completeness proof is located. We show this property, by re-adapting the key result of Salomaa [Sal66] to the systems of equations of our interest.

The proof of the uniqueness of solutions theorem will proceeds by induction on the size of the systems of equations. We first show that systems with only one unknown can be solved via the (Unique) fixpoint axiom. Systems of the size $n+1$ can be reduced to systems of size n, by solving for one of the unknowns and substituting the obtained equation with n unknowns to the remaining n equations. We note that this reduction step is highly reliant on axioms (D2) and (S0). In particular, we will rely on the following property to generalise left distributivity to arbitrary n-ary convex sums.

Lemma 4.5.12. *Let $f \in \mathsf{PExp}$, I be a finite index set and let $\{p_i\}_{i\in I}$ and $\{e_i\}_{i\in I}$ be indexed collections of probabilities and expressions, respectively. Then:*

$$\left(\bigoplus_{i\in I} p_i \cdot e_i\right) ; f \equiv_b \bigoplus_{i\in I} p_i \cdot e_i ; f$$

Proof. By induction. If $I = \emptyset$, then:

$$\begin{aligned} f ; \left(\bigoplus_{i\in I} p_i \cdot e_i\right) &\equiv f ; 0 \\ &\equiv 0 && \text{(S0)} \\ &\equiv \bigoplus_{i\in I} p_i \cdot f ; e_i && (I = \emptyset) \end{aligned}$$

If there exists $j \in I$ such that $p_j = 1$, then:

$$\begin{aligned} f ; \left(\bigoplus_{i\in I} p_i \cdot e_i\right) &\equiv_b f ; e_j \\ &\equiv \left(\bigoplus_{i\in I} p_i \cdot f ; e_i\right) \end{aligned}$$

Finally, for the induction step, we obtain the following:

$$\begin{aligned} f ; \left(\bigoplus_{i\in I} p_i \cdot e_i\right) &\equiv f ; \left(e_j \oplus_{p_j} \left(\bigoplus_{i\in I\setminus\{j\}} \frac{p_i}{1-p_j} \cdot e_i\right)\right) \\ &\equiv f ; e_j \oplus_{p_j} f ; \left(\bigoplus_{i\in I\setminus\{j\}} \frac{p_i}{1-p_j} \cdot e_i\right) && (\textbf{D1}) \\ &\equiv f ; e_j \oplus_{p_j} \left(\bigoplus_{i\in I\setminus\{j\}} \frac{p_i}{1-p_j} \cdot f ; e_i\right) && \text{(Induction hypothesis)} \end{aligned}$$

$$\equiv \left(\bigoplus_{i \in I} p_i \cdot f \,; e_i \right)$$

□

Before proceeding, we demonstrate how to solve a system of equations with two unknowns.

Example 4.5.13. Consider the transition system from the left-hand side of Example 4.1.2. A map $h\colon \{q_0, q_1\} \to \mathsf{PExp}$ is an $\equiv$-solution to the system associated with that transition system, if and only if:

$$h(q_0) \equiv a \,; h(q_1) \qquad h(q_1) \equiv a \,; h(q_1) \oplus_{\frac{1}{4}} 1$$

Since $E(a) = 0$, we can apply the (Unique) axiom and $e \,; 1 \equiv e$ to the equation on the right to deduce that $h(q_1) \equiv a^{\left[\frac{1}{4}\right]}$. Substituting it into the left equation yields $h(q_0) \equiv a \,; a^{\left[\frac{1}{4}\right]}$.

Finally, we proceed to the central result.

Theorem 4.5.14. *Every left-affine system of equations admits a unique $\equiv$-solution.*

Proof. We will write $\mathcal{M} = (M, p, b, r)$ for an arbitrary left-affine system of equations on some finite set Q. Since Q is finite and non-empty, we can safely assume that $Q = \{q_1, \dots q_n\}$ for some positive $n \in \mathbb{N}$. We proceed by induction on n.

$\boxed{\text{Base case}}$ If $Q = \{q_1\}$, then we set $h(q_1) = {M_{1,1}}^{[p_{1,1}]} \,; b_1$. To see that it is indeed a $\equiv$-solution, observe the following:

$$\begin{aligned}
h(q_1) &= {M_{1,1}}^{[p_{1,1}]} \,; b_1 && \text{(Def. of } h\text{)}\\
&\equiv \left(M_{1,1} \,; {M_{1,1}}^{[p_{1,1}]} \oplus_{p_{1,1}} 1 \right) ; b_1 && (\mathsf{Unroll})\\
&\equiv M_{1,1} \,; {M_{1,1}}^{[p_{1,1}]} \,; b_1 \oplus_{p_{1,1}} b_1 && (\mathsf{D1})\\
&\equiv M_{1,1} \,; h(q_1) \oplus_{p_{1,1}} b_1 && \text{(Def. of } h\text{)}\\
&\equiv p_{1,1} \cdot (M_{1,1} \,; h(q_1)) \oplus (1 - p_{1,1}) \cdot b_1 &&\\
&\equiv p_{1,1} \cdot (M_{1,1} \,; h(q_1)) \oplus r_1 \cdot b_1 && (r_1 = 1 - p_{1,1})
\end{aligned}$$

Given an another $\equiv$-solution $g\colon Q \to \mathsf{PExp}$, we have that:

$$\begin{aligned}
g(q_1) &\equiv p_{1,1} \cdot (M_{1,1} \,;g(q_1)) \oplus r_1 \cdot b_1 \\
&\equiv p_{1,1} \cdot (M_{1,1} \,;g(q_1)) \oplus (1 - p_{1,1}) \cdot b_1 && (r_1 = 1 - p_{1,1}) \\
&\equiv M_{1,1} \,; g(q_1) \oplus_{p_{1,1}} b_1 \\
&\equiv {M_{1,1}}^{[p_{1,1}]} \,; b_1 && ((\mathsf{Unique}) \text{ and } E(M_{1,1}) = 0) \\
&\equiv h(q_1)
\end{aligned}$$

$\boxed{\text{Induction step}}$ Assume that all systems of the size n admit a unique $\equiv$-solution. We begin by demonstrating that the problem of finding a $\equiv$-solution for a system with $n+1$ unknowns can be reduced to finding $\equiv$-solution to the system with n unknowns. Let $Q = \{q_1, \dots, q_{n+1}\}$. For $h\colon Q \to \mathsf{PExp}$ to be a $\equiv$-solution to the system $\mathcal{M}$ of size $n+1$, it must satisfy the following equivalence:

$$h(q_{n+1}) \equiv \left(\bigoplus_{i=1}^{n+1} p_{n+1,i} \cdot M_{n+1,i} \,; h(q_i) \right) \oplus r_{n+1} \cdot b_{n+1}$$

We can expand the $(n+1)$-ary sum in the equation above using Lemma 4.4.5 and obtain the following:

$$\begin{aligned}
h(q_{n+1}) \equiv M_{n+1,n+1} \,; h(q_{n+1}) &\oplus_{p_{n+1,n+1}} \\
&\left(\left(\bigoplus_{i=1}^{n} \frac{p_{n+1,i}}{1 - p_{n+1,n+1}} \cdot M_{n+1,i} \,; h(q_i) \right) \oplus \frac{r_{n+1}}{1 - p_{n+1,n+1}} \cdot b_{n+1} \right)
\end{aligned}$$

Since $E(M_{n+1,n+1}) = 0$, we can apply the (Unique) axiom to derive the following:

$$h(q_{n+1}) \equiv M_{n+1,n+1}^{[p_{n+1,n+1}]} \,; \left(\left(\bigoplus_{i=1}^{n} \frac{p_{n+1,i}}{1 - p_{n+1,n+1}} \cdot M_{n+1,i} \,; h(q_i) \right) \oplus \frac{r_{n+1}}{1 - p_{n+1,n+1}} \cdot b_{n+1} \right)$$

Observe that the above expression depends only on $h(q_1), \dots h(q_n)$.). We now substitute the equation for $h(q_{n+1})$ into the equations for $h(q_1), \dots, h(q_n)$. Before proceeding, recall that for any q_j, such that $j \leq n$, we have:

$$h(q_j) \equiv p_{j,n+1} \cdot M_{j,n+1} \,;h(q_{n+1}) \oplus \left(\bigoplus_{i=1}^{n} p_{j,i} \cdot M_{j,i} \,;h(q_i) \right) \oplus r_j \cdot b_j$$

Substituting $h(q_{n+1})$ now yields the following:

$$\begin{aligned} h(q_j) \equiv & \left(\bigoplus_{i=1}^{n} p_{j,i} \cdot M_{j,i} \,;h(q_i) \right) \oplus r_j \cdot b_j \\ & \oplus p_{j,n+1} \cdot M_{j,n+1} \,;M_{n+1,n+1}^{[p_{n+1,n+1}]} \,; \left(\left(\bigoplus_{i=1}^{n} \frac{p_{n+1,i}}{1 - p_{n+1,n+1}} \cdot M_{n+1,i} \,;h(q_i) \right) \oplus \frac{r_{n+1}}{1 - p_{n+1,n+1}} \cdot b_{n+1} \right) \end{aligned}$$

Applying Lemma 4.5.12, we can rewrite the above as:

$$\begin{aligned} h(q_j) \equiv & \left(\bigoplus_{i=1}^{n} p_{j,i} \cdot M_{j,i} \,;h(q_i) \right) \oplus r_j \cdot b_j \\ & \oplus \left(\bigoplus_{i=1}^{n} \frac{p_{j,n+1} p_{n+1,i}}{1 - p_{n+1,n+1}} \cdot M_{j,n+1} \,;M_{n+1,n+1}^{[p_{n+1,n+1}]} \,;M_{n+1,i} \,;h(q_i) \right) \\ & \oplus \frac{p_{j,n+1} r_{n+1}}{1 - p_{n+1,n+1}} \cdot \left(M_{j,n+1} \,;M_{n+1,n+1}^{[p_{n+1,n+1}]} \,;b_{n+1} \right) \end{aligned}$$

To simplify the expression above, we introduce the following shorthand notation fo $i, j \in \{1, \ldots, n\}$:

$$s_{j,i} = p_{j,i} + \frac{p_{j,n+1} p_{n+1,i}}{1 - p_{n+1,n+1}}$$

$$N_{j,i} = \frac{p_{j,i}}{s_{j,i}} \cdot M_{j,i} \oplus \frac{p_{j,n+1} p_{n+1,i}}{s_{j,i}(1 - p_{n+1,n+1})} \cdot \left(M_{j,n+1} \,;M_{n+1,n+1}^{[p_{n+1,n+1}]} \,;M_{n+1,i} \right)$$

$$t_j = r_j + \frac{p_{j,n+1} r_{n+1}}{p_{n+1,n+1}}$$

$$c_j = \frac{r_j}{t_j} \cdot b_j \oplus \frac{p_{j,n+1} r_{n+1}}{t_j(1 - p_{n+1,n+1})} \cdot M_{j,n+1} \,;M_{n+1,n+1}^{[p_{n+1,n+1}]} \,;b_{n+1}$$

Note that $E(N_{j,i}) = 0$ for all $i, j \in \{1, \ldots, n\}$. Now, applying Lemma 4.2.5 and

Lemma 4.4.7, we obtain the following for all $j \in \{1,\dots,n\}$:

$$h(q_j) = \left(\bigoplus_{i=1}^{n} s_{j,i} \cdot N_{j,i} \,; h(q_i)\right) \oplus t_j \cdot c_j$$

In other words, the restriction of h to $\{q_1,\dots,q_n\}$ must be a $\equiv$-solution to the left-affine system $\mathcal{T} = (N,s,c,t)$, which, by the induction hypothesis, has a unique solution. This solution can be extended to the entire system $\mathcal{S}$, by defining $h(q_n)$ as follows:

$$h(q_{n+1}) \equiv M_{n+1,n+1}^{[p_{n+1,n+1}]} \,; \left(\left(\bigoplus_{i=1}^{n} \frac{p_{n+1,i}}{1-p_{n+1,n+1}} \cdot M_{n+1,i} \,; h(q_i)\right) \oplus \frac{r_{n+1}}{1-p_{n+1,n+1}} \cdot b_{n+1}\right)$$

By applying the (Unroll) axiom and reversing the axiomatic manipulations outlined above, it can be shown that this is indeed a $\equiv$-solution to $\mathcal{S}$.

To prove the $\equiv$-uniqueness of h, assume that $g \colon Q \to \mathsf{PExp}$, is another $\equiv$-solution to the system $\mathcal{S}$. Because of (Unique) axiom, we have that:

$$g(q_{n+1}) \equiv M_{n+1,n+1}^{[p_{n+1,n+1}]} \,; \left(\left(\bigoplus_{i=1}^{n} \frac{p_{n+1,i}}{1-p_{n+1,n+1}} \cdot M_{n+1,i} \,; g(q_i)\right) \oplus \frac{r_{n+1}}{1-p_{n+1,n+1}} \cdot b_{n+1}\right)$$

Substituting this into the equations for $h(q_1),\dots,h(q_n)$ and following the same steps as before leads to the requirement that, for all $j \in \{1,\dots,n\}$, we have:

$$g(q_j) = \left(\bigoplus_{i=1}^{n} s_{j,i} \cdot N_{j,i} \,; g(q_i)\right) \oplus t_j \cdot c_j$$

By the induction hypothesis, the left-affine system of equations $\mathcal{T}$ admits a unique $\equiv$-solution. Therefore for all $j \in \{1,\dots,n\}$, we have that $g(q_j) \equiv h(q_j)$. Consequently, we have that:

$$\begin{aligned} g(q_{n+1}) &\equiv M_{n+1,n+1}^{[p_{n+1,n+1}]} \,; \left(\left(\bigoplus_{i=1}^{n} \frac{p_{n+1,i}}{1-p_{n+1,n+1}} \cdot M_{n+1,i} \,; g(q_i)\right) \oplus \frac{r_{n+1}}{1-p_{n+1,n+1}} \cdot b_{n+1}\right) \\ &\equiv M_{n+1,n+1}^{[p_{n+1,n+1}]} \,; \left(\left(\bigoplus_{i=1}^{n} \frac{p_{n+1,i}}{1-p_{n+1,n+1}} \cdot M_{n+1,i} \,; h(q_i)\right) \oplus \frac{r_{n+1}}{1-p_{n+1,n+1}} \cdot b_{n+1}\right) \end{aligned}$$

$$\equiv h(q_{n+1})$$

This completes the proof. □

4.5.4 Step 3: Correspondence of solutions and homomorphisms

We are not done yet, as in the last step we only proved properties of systems of equations and their solutions, while our main interest is in appropriate $\hat{\mathcal{G}}$-coalgebras and their homomorphisms. As desired, it turns out that $\equiv$-solutions are in one-to-one correspondence with $\hat{\mathcal{G}}$-coalgebra homomorphisms from determinisations of (converted) finite state $\mathcal{DF}$-coalgebras to the coalgebra structure on $\mathsf{PExp}/{\equiv}$.

Lemma 4.5.15. *For a finite set X, we have the following one-to-one correspondence:*

$$\frac{\hat{\mathcal{G}}\text{-coalgebra homomorphisms } m\colon ((\mathcal{D}X,\mu_X),(\gamma_X\circ\beta)^\sharp)\to((\mathsf{PExp}/{\equiv},\alpha_{\equiv}),d)}{\equiv\text{-solutions } h\colon X\to\mathsf{PExp} \text{ to a system } \mathcal{S}(\beta) \text{ associated with } \mathcal{D}F\text{-coalgebra } (X,\beta)}$$

Before diving into the main argument, we establish the following helper lemma, which provides a concrete characterization of $\equiv$-solutions to systems associated with $\mathcal{DF}$-coalgebras.

Lemma 4.5.16. *Let (X,β) be a finite-state $\mathcal{DF}$-coalgebra. A map $h\colon X\to\mathsf{PExp}$ is a $\equiv$-solution to the system $\mathcal{S}(\beta)$ if and only if for all $x\in X$, we have that:*

$$h(x)\equiv\left(\bigoplus_{(a,x')\in A\times X}\beta(x)(a,x')\cdot a\,;h(x')\right)\oplus\beta(x)(\checkmark)\cdot 1$$

Proof. Fix an arbitrary $x\in X$. Recall that, if $p^{\beta}_{x,x'}=0$, then $M^{\beta}_{x,x'}=0\equiv\bigoplus_{a\in A}0\cdot a$. Because of that, we can safely assume that $M^{\beta}_{x,x'}$ can be always written out in the following form:

$$M^{\beta}_{x,x'}\equiv\bigoplus_{a\in A}s^{a}_{x,x'}\cdot a$$

where $s^a_{x,x'} \in [0,1]$. By definition of $\equiv$-solution, we the following:

$$\begin{aligned}
h(x) &\equiv \left(\bigoplus_{x' \in X} p^{\beta}_{x,x'} \cdot M^{\beta}_{x,x'} \,;h(x') \right) \oplus r^{\beta}_{x} \cdot b^{\beta}_{x} \\
&\equiv \left(\bigoplus_{x' \in X} p^{\beta}_{x,x'} \cdot \left(\bigoplus_{a \in A} s^{a}_{x,x'} \cdot a \right) ;h(x') \right) \oplus r^{\beta}_{x} \cdot b^{\beta}_{x} \\
&\equiv \left(\bigoplus_{x' \in X} p^{\beta}_{x,x'} \cdot \left(\bigoplus_{a \in A} s^{a}_{x,x'} \cdot a \,;h(x') \right) \right) \oplus r^{\beta}_{x} \cdot b^{\beta}_{x} && \text{(Lemma 4.4.7)} \\
&\equiv \left(\bigoplus_{(a,x') \in A \times X} p^{\beta}_{x,x'} s^{a}_{x,x'} \cdot a \,;h(x') \right) \oplus \beta(x)(\checkmark) \cdot 1 && \text{(Lemma 4.2.4)} \\
&\equiv \left(\bigoplus_{(a,x') \in A \times X} \beta(x)(a,x') \cdot a \,;h(x') \right) \oplus \beta(x)(\checkmark) \cdot 1
\end{aligned}$$

The last step of the proof relies on the observation that for both cases of how $M^{\beta}_{x,x'}$ is defined, we have that $p^{\beta}_{x,x'} s^{a}_{x,x'} = \beta(x)(a,x')$. Similarly, in the passage between the third and fourth line, we have used the observation that: $r^{\beta}_{x} \cdot b^{\beta}_{x} \equiv \beta(x)(\checkmark) \cdot 1$ for both possible cases. □

We are now ready to prove the main claim.

Proof of Lemma 4.5.15. First, assume that $h \colon X \to \mathsf{PExp}$ is a solution to the system $\mathcal{S}(\beta)$ associated with $\mathcal{DF}$-coalgebra (X,β). We show that $([-] \circ h)^{\sharp} \colon \mathcal{D}X \to \mathsf{PExp}/{\equiv}$ is a $\hat{\mathcal{G}}$-coalgebra homomorphism. In other words, we will claim the commutativity of the diagram below.

$$\begin{tikzcd}
X \arrow[dr, "\eta_X"'] \arrow[drr, "{[-]\circ h}"] \arrow[dd, "\beta"'] & & \\
& \mathcal{D}X \arrow[r, "{([-]\circ h)^{\sharp}}"'] \arrow[ddl, "{(\gamma_X \circ \beta)^{\sharp}}"] & \mathsf{PExp}/{\equiv} \arrow[dd, "d"] \\
\mathcal{DF}X \arrow[d, "\gamma_X"'] & & \\
\hat{\mathcal{G}}\mathcal{D}X \arrow[rr, "{\hat{\mathcal{G}}([-]\circ h)^{\sharp}}"] & & \hat{\mathcal{G}}\mathsf{PExp}/{\equiv}
\end{tikzcd}$$

Because of Lemma 4.5.16, we have that for all $x \in X$:

$$h(x) \equiv \left(\bigoplus_{(a,x') \in A \times X} \beta(x)(a,x') \cdot a \, ; h(x') \right) \oplus \beta(x)(\checkmark) \cdot 1$$

Let $\nu \in \mathcal{D}X$. The convex extension of the map $[-] \circ h \colon X \to \mathsf{PExp}/{\equiv}$ is given by the following:

$$([-] \circ h)^{\sharp}(\nu) = \left[\bigoplus_{x \in \mathrm{supp}(\nu)} \nu(x) \cdot h(x) \right]$$

Similarly, given $\beta : X \to \mathcal{D}\mathcal{F}X$ we have that:

$$(\gamma_X \circ \beta)^{\sharp}(\nu) = \left\langle \sum_{x \in \mathrm{supp}(\nu)} \nu(x)\beta(x)(\checkmark), \lambda a. \lambda x'. \sum_{x \in \mathrm{supp}(\nu)} \nu(x)\beta(x)(a,x') \right\rangle$$

For any distribution $\nu \in \mathcal{D}X$, we have the following:

$$\begin{aligned}
& d \circ ([-] \circ h)^{\sharp}(\nu) \\
& = d \left[\bigoplus_{x \in \mathrm{supp}(\nu)} \nu(x) \cdot \left(\left(\bigoplus_{(a,x') \in A \times X} \beta(x)(a,x') \cdot a \, ; h(x') \right) \oplus \beta(x)(\checkmark) \cdot 1 \right) \right] \\
& = d \left[\left(\bigoplus_{(a,x') \in A \times X} \left(\sum_{x \in \mathrm{supp}(\nu)} \nu(x)\beta(x)(a,x'). \right) \cdot a \, ; h(x') \right) \right. \\
& \qquad \left. \oplus \left(\sum_{x \in \mathrm{supp}(\nu)} \nu(x)\beta(x)(\checkmark) \right) \cdot 1 \right] && \text{(Barycenter axiom)} \\
& = \left\langle \sum_{x \in \mathrm{supp}(\nu)} \nu(x)\beta(x)(\checkmark), \lambda a. \left[\bigoplus_{x' \in X} \left(\sum_{x \in \mathrm{supp}(\nu(x))} \nu(x)\beta(x)(a,x') \right) \cdot h(x') \right] \right\rangle
\end{aligned}$$

Now, consider $\hat{\mathcal{G}}([-] \circ h)^{\sharp} \circ (\gamma_X \circ \beta)^{\sharp}(\nu)$. We have the following:

$$\begin{aligned}
& \hat{\mathcal{G}}([-] \circ h)^{\sharp} \circ (\gamma_X \circ \beta)^{\sharp}(\nu) \\
& = \hat{\mathcal{G}}([-] \circ h)^{\sharp} \left\langle \sum_{x \in \mathrm{supp}(\nu)} \nu(x)\beta(x)(\checkmark), \lambda a. \lambda x'. \sum_{x \in \mathrm{supp}(\nu)} \nu(x)\beta(x)(a,x') \right\rangle \\
& = \left\langle \sum_{x \in \mathrm{supp}(\nu)} \nu(x)\beta(x)(\checkmark), \lambda a. ([-] \circ h)^{\sharp} \left(\lambda x'. \sum_{x \in \mathrm{supp}(\nu)} \nu(x)\beta(x)(a,x') \right) \right\rangle
\end{aligned}$$

$$= \left\langle \sum_{x \in \mathrm{supp}(\nu)} \nu(x)\beta(x)(\checkmark), \lambda a. \left[\bigoplus_{x' \in X} \left(\sum_{x \in \mathrm{supp}(\nu)} \nu(x)\beta(x)(a,x') \right) \cdot h(x') \right] \right\rangle$$

Hence, we obtain $d \circ ([-] \circ h)^\sharp(\nu) = \hat{\mathcal{G}}([-] \circ h)^\sharp \circ (\gamma_X \circ \beta)^\sharp$, thus demonstrating that $([-] \circ h)^\sharp$ is indeed a $\hat{\mathcal{G}}$-coalgebra homomorphism.

For the converse, let $((\mathcal{D}X, \mu_X), (\gamma_X \circ \beta)^\sharp)$ be a $\hat{\mathcal{G}}$-coalgebra and let $m \colon \mathcal{D}X \to \mathsf{PExp}/\!\equiv$ be a $\hat{\mathcal{G}}$-coalgebra homomorphism from $((\mathcal{D}X, \mu_X), (\gamma_X \circ \beta)^\sharp)$ to $((\mathsf{PExp}/\!\equiv, \alpha_\equiv), d)$. Recall that m arises uniquely as a convex extension of some map $\overline{h} \colon X \to \mathsf{PExp}/\!\equiv$. This map can be factored as $\overline{h} = [-] \circ h$, for some $h \to \mathsf{PExp}$. Observe that any two such factorisations determine the same $\equiv$-solution. In particular, let $s \colon \mathsf{PExp}/\!\equiv \to \mathsf{PExp}$ be a section of $[-] \colon \mathsf{PExp} \to \mathsf{PExp}/\!\equiv$ and define $h = s \circ \overline{h}$.

Since m is a homomorphism, the inner square in the diagram above commutes. Moreover, as the triangle diagrams also commute, it follows that the outer diagram commutes as well. Recall Lemma 4.5.8, which states that $d \colon \mathsf{PExp}/\!\equiv \to \hat{\mathcal{G}}\mathsf{PExp}/\!\equiv$ is an isomorphism. Consequently, for all $x \in X$ we have the following:

$$\begin{aligned}
([-] \circ h)(x) &= (d^{-1} \circ \hat{\mathcal{G}}([-] \circ h)^\sharp \circ \gamma_x \circ \beta)(x) \\
&= (d^{-1} \circ \hat{\mathcal{G}}([-] \circ h)^\sharp) \langle \beta(x)(\checkmark), \lambda a. \lambda x'. \beta(x)(a,x') \rangle \\
&= d^{-1} \left\langle \beta(x)(\checkmark), \lambda a. ([-] \circ h)^\sharp(\lambda x'. \beta(x)(a,x')) \right\rangle \\
&= d^{-1} \left\langle \beta(x)(\checkmark), \lambda a. \left[\bigoplus_{x' \in X} \beta(x)(a,x') \cdot h(x') \right] \right\rangle \\
&= d^{-1} \left\langle \beta(x)(\checkmark), \lambda a. \boxplus_{x' \in X} \beta(x)(a,x') \cdot [h(x')] \right\rangle && \text{(Lemma 4.4.12)} \\
&= \left[\left(\bigoplus_{(a,x') \in A \times X} \beta(x)(a,x') \cdot a \,; h(x') \right) \beta(x)(\checkmark) \cdot 1 \right] && \text{(Def. of } d^{-1})
\end{aligned}$$

□

Aside from the completeness argument, the above result also gives us an analogue of (one direction of) Kleene's theorem for $\mathcal{DF}$-coalgebras as a corollary. The other direction, converting PRE to finite-state $\mathcal{DF}$-coalgebras is given by the

Antimirov construction, described in Section 4.3.

Corollary 4.5.17. *Let (X,β) be a finite-state $\mathcal{DF}$-coalgebra. For every state $x \in X$, there exists an expression $e_x \in \mathsf{PExp}$, such that the probabilistic language denoted by x is the same as $[\![e_x]\!]$.*

Proof. Let $h\colon X \to \mathsf{PExp}$ be the solution to the system $\mathcal{S}(\beta)$ associated with a $\mathcal{DF}$-coalgebra (X,β) existing because of Theorem 4.5.14. For each $x \in X$, set $e_x = h(x)$. Recall that because of Lemma 4.5.15, $([-] \circ h)^\sharp \colon \mathcal{D}X \to \mathsf{PExp}/{\equiv}$ is a $\hat{\mathcal{G}}$-coalgebra homomorphism from $(\mathcal{D}X, (\gamma_X \circ \beta)^\sharp)$ to $(\mathsf{PExp}/{\equiv}, d)$. For any $x \in X$, we have the following:

$$\begin{aligned}
[\![e_x]\!] &= [\![h(x)]\!] && \text{(Def. of } e_x\text{)}\\
&= \mathsf{beh}_d([h(x)]) && \text{(Lemma 4.4.19)}\\
&= \mathsf{beh}_d \circ ([-]\circ h)^\sharp(\eta_X(x)) && \text{(Free-forgetful adjunction)}\\
&= \mathsf{beh}_{(\gamma_X\circ\beta)^\sharp}(\eta_X(x)) \qquad = \mathsf{Lang}_{(X,\beta)}(x) && \text{(}([-]\circ h)^\sharp \text{ is a homomorphism)}
\end{aligned}$$

□

4.5.5 Step 4: Establish the universal property

The only remaining piece allowing us to use Theorem 4.5.2 is to claim that the $\hat{\mathcal{G}}$-coalgebra is locally finitely presentable. We indirectly rely on finiteness of Antimirov derivatives shown in Lemma 4.3.6.

Lemma 4.5.18. *$((\mathsf{PExp}/{\equiv}, \alpha_{\equiv}), d)$ is locally finitely presentable $\hat{\mathcal{G}}$-coalgebra.*

Proof. We establish the simpler conditions of [Mil10, Definition III.7]. Recall that in PCA locally presentable and locally generated objects coincide [SW15]. Because of that, it suffices to show that every finitely generated subalgebra is contained in finitely generated subcoalgebra. Let (Y, α_Y) be a finitely generated subalgebra of $(\mathsf{PExp}/{\equiv}, \alpha_{\equiv})$ generated by $[e_1], \dots, [e_n] \in \mathsf{PExp}/{\equiv}$ where $1 \leq i \leq n$. We will construct a finitely generated subalgebra (Z, α_Y) of $(\mathsf{PExp}/{\equiv}, \alpha_{\equiv})$ such that $[e_i] \in Z$ (hence containing (Y, α_Y) as subalgebra) that is subcoalgebra as well.

Recall that given an expression $e \in \mathsf{PExp}$, we write $\langle e \rangle_\partial \subseteq \mathsf{PExp}$ for the set of all expressions reachable via Antimirov derivative from e. By Lemma 4.3.6 that set is finite. Let (Z, α_Z) be a subalgebra of $(\mathsf{PExp}/{\equiv}, \alpha_{\equiv})$ generated by the following set

$$\left\{ [a\,;e'] \mid a \in A, e' \in \bigcup_{i=1}^{i \leq n} \langle e_i \rangle_\partial \right\} \cup \{1\}$$

Note that this set is finite, as A is finite and there are only finitely many expressions e_i, each with finitely many derivatives. We proceed to showing that Z is closed under the transitions of the coalgebra structure d. Pick an element $z \in Z$, given by the following:

$$z = p \cdot [1] \boxplus \bigboxplus_{j \in J} p_j \cdot [a_j\,;e'_j]$$

Note that:

$$d([a_k\,;e'_j]) = \langle 0, f_j \rangle \text{ with } f_j(a) = [e'_j] \text{ if } a = a_j \text{ or otherwise } f(a) = [0]$$
$$d([1]) = \langle 1, f \rangle \text{ with } f(a) = [0] \text{ for all } a \in A$$

Therefore, we can conclude that:

$$d(z) = \left\langle p, \lambda a. \left[\bigoplus_{a_j = a} p_j \cdot e'_j \right] \right\rangle$$

As a consequence of Theorem 4.4.3, for all $j \in J$ we have that $[e'_j] = \left[q_j \cdot 1 \oplus \bigoplus_{k \in K} p_{j,k} \cdot a_{j,k}\,; e'_{j,k} \right]$, where $[e'_{i,k}] \in \bigcup_{i=1}^{i \leq n} \langle e_i \rangle_\partial$ and hence $[a_{j,k}\,; e'_{j,k}] \in Z$ for all $k \in K$. To complete the proof, we will argue that $d(z) \in \hat{\mathcal{G}}(Z, \alpha_Z)$. Observe the following:

$$\begin{aligned} d(y) &= \left\langle p, \lambda a. \left[\bigoplus_{a_j = a} p_j \cdot \left(q_j \cdot 1 \oplus \bigoplus_{k \in K} p_{j,k} \cdot a_{j,k}\,; e'_{j,k} \right) \right] \right\rangle \\ &= \left\langle p, \lambda a. \left[\sum_{a_j = a} p_j q_j \cdot 1 \oplus \bigoplus_{\substack{(j,k) \in J \times K \\ a_j = a}} p_j p_{i,k} \cdot a_{j,k}\,; e'_{j,k} \right] \right\rangle && \text{(Lemma 4.2.4)} \end{aligned}$$

$$= \left\langle p, \lambda a. \left(\sum_{a_j = a} p_j q_j \cdot [1] \boxplus \bigboxplus_{\substack{(j,k) \in J \times K \\ a_j = a}} p_j p_{i,j} \cdot \left[a_{j,k} \,;\, e'_{j,k} \right] \right) \right\rangle$$

(Lemma 4.4.12)

Now, it remains to observe the following for all $a \in A$:

$$\sum_{\substack{j,k \in J \times K \\ a_j = a}} p_j(p_{i,k} + q_j) \leq \sum_{j \in J} p_j \leq 1 - p$$

Hence $d(y) \in \hat{\mathcal{G}}(Z, \alpha_Z)$, as desired. □

We are now ready to obtain the following result.

Corollary 4.5.19. $((\mathsf{PExp}/{\equiv}, \alpha_{\equiv}), d)$ *is isomorphic to the rational fixpoint of the functor* $\hat{\mathcal{G}}$ *and is a subcoalgebra of the final* $\hat{\mathcal{G}}$*-coalgebra.*

Proof. Follows from Lemma 4.5.4, Theorem 4.5.14, Lemma 4.5.15 and Lemma 4.5.18. Since in PCA finitely presentable and finitely generated objects coincide (Theorem 4.2.7) and $\hat{\mathcal{G}}$ preserves non-empty monomorphisms (Remark 4.5.3), we have that the rational fixpoint of $\hat{\mathcal{G}}$ is fully abstract (Theorem 4.2.8). □

The only thing is to connect the final $\hat{\mathcal{G}}$-coalgebra with the final $\overline{\mathcal{G}}$-coalgebra, which is carried by $[0,1]^{A^*}$.

Lemma 4.5.20. *The final* $\hat{\mathcal{G}}$*-coalgebra is a subcoalgebra of the final* $\overline{\mathcal{G}}$*-coalgebra.*

Proof. Let $\nu\hat{\mathcal{G}}$ be the final $\hat{\mathcal{G}}$-coalgebra and $\nu\overline{\mathcal{G}}$ be the final $\overline{\mathcal{G}}$-coalgera. Since $\nu\hat{\mathcal{G}}$ can be seen as a $\overline{\mathcal{G}}$-coalgebra, there is a unique $\overline{\mathcal{G}}$-coalgebra homomorphism $\mathsf{beh}_{\nu\hat{\mathcal{G}}} \colon \nu\hat{\mathcal{G}} \to \nu\overline{\mathcal{G}}$. Since $\mathcal{D} \colon \mathsf{Set} \to \mathsf{Set}$ preserves epimorphisms [Gum00, Corollary 3.16], we have that epi-mono factorisations in Set carry to epi-mono factorisations in PCA [Wiß22, Proposition 3.7]. Moreover, since $\overline{\mathcal{G}}$ preserves non-empty monomorphisms (Lemma 4.2.9), we can further lift epi-mono factorisations in PCA to epi-mono factorisations in $\mathsf{Coalg}\,\overline{\mathcal{G}}$ [MPW20, Lemma 2.5]. Because of this, we can factorise $\mathsf{beh}_{\nu\overline{\mathcal{G}}}$ in the following way:

$$\begin{tikzcd}
\nu\hat{\mathcal{G}} \arrow[r, tail, two heads, "e"] \arrow[rr, bend right, "\mathsf{beh}_{\nu\hat{\mathcal{G}}}"'] & Q \arrow[r, hook, "m"] & \nu\overline{\mathcal{G}}
\end{tikzcd}$$

In the above, Q is a $\overline{\mathcal{G}}$-coalgebra, $e\colon \nu\hat{\mathcal{G}} \to Q$ a surjective $\overline{\mathcal{G}}$-coalgebra homomorphism, and $m\colon Q \to \nu\overline{\mathcal{G}}$ an injective $\overline{\mathcal{G}}$-coalgebra homomorphism. We will argue that $e\colon \nu\hat{\mathcal{G}} \to Q$ is an isomorphism.

First, we can use Lemma 4.5.7 to show that Q is a $\hat{\mathcal{G}}$-coalgebra and $e\colon \nu\hat{G} \to Q$ is a $\hat{\mathcal{G}}$-coalgebra homomorphism. Because of this there exists a unique map $\mathsf{beh}_Q\colon Q \to \nu\hat{\mathcal{G}}$ that is a $\hat{\mathcal{G}}$-coalgebra homomorphism. Then, $\mathsf{beh}_Q \circ e\colon \nu\hat{\mathcal{G}} \to \nu\hat{\mathcal{G}}$ is a $\hat{\mathcal{G}}$-homomorphism that by finality of $\nu\hat{\mathcal{G}}$ must be equal to $\mathsf{id}_{\nu\hat{\mathcal{G}}}$.

To see that $e \circ \mathsf{beh}_Q\colon Q \to Q$ is an identity, observe that by finality of $\nu\overline{\mathcal{G}}$, the following two maps must be equal:

$$m \circ e \circ \mathsf{beh}_Q = m \circ \mathsf{id}_Q$$

Since $m\colon Q \to \nu\overline{\mathcal{G}}$ is monic, we can cancel it on the left and obtain $e \circ \mathsf{beh}_Q = \mathsf{id}_Q$, as desired. Since e is an isomorphism, we have that $\mathsf{beh}_{\nu\hat{\mathcal{G}}}$ is injective, which completes the proof. □

A direct consequence of the above is the following:

Corollary 4.5.21. *The map* $\mathsf{beh}_d\colon \mathsf{PExp}/{\equiv} \to [0,1]^{A^*}$ *is injective.*

Proof. Recall that beh_d is a unique $\overline{\mathcal{G}}$-coalgebra homomorphism from $((\mathsf{PExp}/{\equiv}, \alpha_{\equiv}), d)$ to the final $\overline{\mathcal{G}}$-coalgebra carried by the set $[0,1]^{A^*}$. Since $((\mathsf{PExp}/{\equiv}, \alpha_{\equiv}), d)$ is the rational fixpoint of the functor of $\hat{\mathcal{G}}$, beh_d can be factorised as follows:

$$\begin{tikzcd}
\mathsf{PExp}/{\equiv} \arrow[r, hook] \arrow[rr, bend right, "\mathsf{beh}_d"'] & \nu\hat{\mathcal{G}} \arrow[r, hook] & {[0,1]^{A^*}}
\end{tikzcd}$$

Due to Corollary 4.5.19 and Lemma 4.5.20, the maps involved in the above factorisations are injective, which implies that beh_d is also injective. □

At this point, showing completeness becomes straightforward.

Theorem 4.5.22. *Let $e, f \in \mathsf{PExp}$. If $[\![e]\!] = [\![f]\!]$, then $e \equiv f$.*

Proof. We have the following:

$$\begin{aligned} [\![e]\!] = [\![f]\!] &\iff \mathsf{beh}_d([e]) = \mathsf{beh}_d([f]) && \text{(Lemma 4.4.19)} \\ &\implies [e] = [f] && (\mathsf{beh}_d \text{ is injective}) \\ &\iff e \equiv f \end{aligned}$$

□

4.6 Discussion

In this chapter, we introduced probabilistic regular expressions (PRE), a probabilistic counterpart to Kleene's regular expressions. As the main technical contribution, we presented a Salomaa-style inference system for reasoning about probabilistic language equivalence of expressions and proved it sound and complete. Additionally, we gave a two-way correspondence between languages denoted by PRE and finite-state Generative Probabilistic Transition Systems. Our approach is coalgebraic and enabled us to reuse several recently proved results on fixpoints of functors and convex algebras. This abstract outlook guided the choice of the right formalisms and enabled us to isolate the key results we needed to prove to achieve completeness while at the same time reusing existing results and avoiding repeating complicated combinatorial proofs. The key technical lemma, on uniqueness of solutions to certain systems of equations, is a generalisation of automata-theoretic constructions from the 60s further exposing the bridge between our probabilistic generalisation and the classical deterministic counterpart.

4.6.1 Related work

Probabilistic process algebras and their axiomatisations have been widely studied [BS01; SS00; MOW03; Ber22] with syntaxes featuring action prefixing and least fixed point operators instead of the regular operations of sequential composition and probabilistic loops. This line of research focussed on probabilistic bisimulation, while probabilistic language equivalence, which we focus on, stems from automata theory, e.g. the work of Rabin on probabilistic automata [Rab63]. Language equivalence of Rabin automata has been studied from an algorithmic point of view [Kie+11; Kie+12].

Stochastic Regular Expressions (SRE) [Ros00; Bee17; GPG18], which were one of the main inspirations for this chapter, can also be used to specify probabilistic languages. The syntax of SRE features probabilistic Kleene star and n-ary probabilistic choice, however, it does not include 0 and 1. The primary context of that line of research was around genetic programming in probabilistic pattern matching, and the topic of axiomatisation was simply not tackled.

PRE can be thought of as a fragment of ProbGKAT [Róż+23], a probabilistic extension of a strictly deterministic fragment of Kleene Algebra with Tests, that was studied only under the finer notion of bisimulation equivalence. The completeness ProbGKAT was obtained through a different approach to ours, as it relied on a powerful axiom scheme to solve systems of equations.

Our soundness result, as well as semantics via generalised determinisation, were inspired by the work of Silva and Sokolova [SS11], who introduced a two-sorted process calculus for reasoning about probabilistic language equivalence of GPTS. Unlike PRE, their language syntactically excludes the possibility of introducing recursion over terms which might immediately terminate. Moreover, contrary to our completeness argument, their result hinges on the subset of axioms being complete with respect to bisimilarity, similarly to the complete axiomatisation of trace congruence of LTS due to Rabinovich [Rab93]. The use of coalgebra to model trace/language semantics is a well-studied topic [JSS15; RJL21] and other approaches besides generalised determinisation [Sil+10; BSS17] included the use of Kleisli categories [HJS07] and coalgebraic modal logic [KR15]. We build on the vast line of work on coalgebraic completeness theorems [Jac06; Sil10; SRS21; Mil10; BMS13], coalgebraic semantics of probabilistic systems [VR99; Sok05] and fixpoints of the functors [MPW20; Mil18; SW18].

4.6.2 Future work

A first natural direction is exploring whether one could obtain an *algebraic* axiomatisation of PRE. Similarly to Salomaa's system, our axiomatisation is unsound under substitution of letters by arbitrary expressions in the case of the termination operator used to give side condition to the unique fixpoint axiom. We are interested if one could give an alternative inference system in the style of Kozen's axiomatisation [Koz94], in which the Kleene star is the least fixpoint wrt the natural order induced by the $+$ operation and thus not requiring the side condition to introduce loops. In the case of PRE, the challenge is that there is no obvious way of defining a natural order on PRE in terms of $\oplus_p$ operation.

Additionally, it is interesting to ask if the finer relation $\equiv_b$ is complete with respect to the probabilistic bisimilarity. One could view it as a probabilistic analogue of the problem of completeness of Kleene Algebra modulo bisimilarity posed by Milner [Mil84], which was recently answered positively by Grabmayer [Gra22].

Finally, an interesting direction would be to study a more robust notion of *language distance* [Bal+18] of GPTS by extending our axiomatisation to a system based on quantitative equational logic [MPP16]. Similar results were already obtained in the case of probabilistic process calculus of Stark and Smolka [SS00] for bisimulation distance [Bac+18a] and distance between infinite traces [Bac+18b].

Chapter 5

Conclusions

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Appendix A

Omitted proofs from Chapter 4

This chapter contains the omitted proofs from Chapter 4, in particular the soundness of $\equiv_b$ with respect to bisimilarity of $\mathcal{DF}$-coalgebras and the intermediate results leading to it.

A.1 Couplings of subdistributions

Below, we recall the notions surrounding couplings of (sub)probability distributions, which we use to concretely characterise bisimulations of $\mathcal{DF}$-coalgebras.

Definition A.1.1 ([Hsu17, Definition 2.1.2])**.** Given two subdistributions ν_1, ν_2 over X and Y respectively, a subdistribution ν over $X \times Y$ is called coupling if:

1. For all $x \in X$, $\nu_1(x) = \nu[\{x\} \times Y]$

2. For all $y \in Y$, $\nu_2(y) = \nu[X \times \{y\}]$

It can be straightforwardly observed that a coupling ν of (ν_1, ν_2) is finitely supported if and only if both ν_1 and ν_2 are finitely supported.

Definition A.1.2 ([Hsu17, Definition 2.1.7])**.** Let ν_1, ν_2 be subdistributions over X and Y respectively and let $R \subseteq X \times Y$ be a relation. A subdistribution ν over $X \times Y$ is a *witness* for the *R-lifting* of (ν_1, ν_2) if:

1. ν is a coupling for (ν_1, ν_2)

2. $\text{supp}(\nu) \subseteq R$

If there exists ν satisfying these two conditions, we say ν_1 and ν_2 are related by the *lifting* of R and write $\langle \nu_1, \nu_2 \rangle \in \mathsf{Lift}(R)$. It can be immediately observed that $\langle \nu_1, \nu_2 \rangle \in \mathsf{Lift}(R)$ implies $\langle \nu_2, \nu_1 \rangle \in \mathsf{Lift}(R^{-1})$.

Given a relation $R \subseteq X \times Y$ and a set $B \subseteq X$, we will write $R(B) \subseteq Y$ for the set given by $R(B) = \{y \in Y \mid \langle x, y \rangle \in R \text{ and } x \in B\}$. We will write $R^{-1} \subseteq Y \times X$ for the converse relation given by $R^{-1} = \{\langle y, x \rangle \in Y \times X \mid \langle x, y \rangle \in R\}$.

Theorem A.1.3 ([Hsu17, Theorem 2.1.11])**.** *Let ν_1, ν_2 be subdistributions over X and Y respectively and let $R \subseteq X \times Y$ be a relation. Then $\langle \nu_1, \nu_2 \rangle \in \mathsf{Lift}(R)$ implies $\nu_1[B] \le \nu_2[R(B)]$ for every subset $B \subseteq X$. The converse holds if ν_1 and ν_2 have equal weight.*

Lemma A.1.4. *Let ν_1, ν_2 be subdistributions over X and Y respectively and let $R \subseteq X \times Y$ be a relation. $\langle \nu_1, \nu_2 \rangle \in \mathsf{Lift}(R)$ if and only if:*

1. *For all $B \subseteq X$, $\nu_1[B] \le \nu_2[R(B)]$*
2. *For all $C \subseteq Y$, $\nu_2[C] \le \nu_1[R^{-1}(C)]$*

Proof. Assume that $\langle \nu_1, \nu_2 \rangle \in \mathsf{Lift}(R)$. Recall, that in such a case $\langle \nu_2, \nu_1 \rangle \in \mathsf{Lift}(R^{-1})$. Applying Theorem A.1.3 yields ① and ② respectively.

For the converse, assume ① and ② do hold. We have the following:

$$\begin{aligned} |\nu_1| &= \nu_1[X] \le \nu_2[R(X)] && (\textcircled{1}) \\ &\le \nu_2[Y] && (R(X) \subseteq Y) \\ &= |\nu_2| \end{aligned}$$

By a symmetric reasoning involving ②, we can show that $|\nu_2| \le |\nu_1|$ and therefore $|\nu_1| = |\nu_2|$. Since condition ① holds, we can use Theorem A.1.3 to conclude that $\langle \nu_1, \nu_2 \rangle \in \mathsf{Lift}(R)$. □

A.2 Relation lifting

Definition A.2.1 ([Sok05, Definition 3.6.1])**.** Let $R \subseteq X \times Y$ be a relation, and $\mathcal{B}$ a Set endofunctor. The relation R can be lifted to relation $\mathsf{Rel}(\mathcal{B})(R) \subseteq \mathcal{B}X \times \mathcal{B}Y$

defined by

$$\langle x,y\rangle \in \mathsf{Rel}(\mathcal{B})(R) \iff \exists z \in \mathcal{B}R \text{ such that } \mathcal{B}\pi_1(z) = x \text{ and } \mathcal{B}\pi_2(z) = y$$

Lemma A.2.2 ([Sok05, Lemma 3.6.4])**.** *Let $\mathcal{B}\colon \mathsf{Set} \to \mathsf{Set}$ be an arbitrary endofunctor on* Set*. A relation $R \subseteq X \times Y$ is a bisimulation between the $\mathcal{B}$-coalgebras (X,β) and (Y,γ) if and only if:*

$$\langle x,y\rangle \in R \implies \langle \beta(x), \gamma(y)\rangle \in \mathsf{Rel}(\mathcal{B})(R)$$

Lemma A.2.3. *For any $R \subseteq X \times Y$*

1. $\mathsf{Rel}(\mathcal{F})(R) = \{\langle \checkmark, \checkmark\rangle\} \cup \{\langle a,x\rangle, \langle a,y\rangle) \mid a \in A, \langle x,y\rangle \in R\}$

2. $\mathsf{Rel}(\mathcal{D}\mathcal{F})(R) = \mathsf{Lift}(\mathsf{Rel}(\mathcal{F})(R))$

Proof. Using the inductive definition of relation liftings from [Sok05, Lemma 3.6.7]. □

Lemma A.2.4. *Let $\nu_1 \in \mathcal{D}\mathcal{F}X$, $\nu_2 \in \mathcal{D}\mathcal{F}Y$ and let $R \subseteq X \times Y$. The necessary and sufficient conditions for $\langle \nu_1, \nu_2\rangle \in \mathsf{Rel}(\mathcal{D}\mathcal{F})(R)$ are:*

1. $\nu_1(\checkmark) = \nu_2(\checkmark)$

2. For all $B \subseteq X$ and all $a \in A$, $\nu_1[\{a\} \times B] \leq \nu_2[\{a\} \times R(B)]$

3. For all $C \subseteq Y$ and all $a \in A$, $\nu_2[\{a\} \times C] \leq \nu_1[\{a\} \times R^{-1}(C)]$

Proof. Assume that $(\nu_1, \nu_2) \in \mathsf{Rel}(\mathcal{D}F)(R)$. By Lemma A.2.3, it is equivalent to ν_1 and ν_2 being related by the lifting of $\mathsf{Rel}(\mathcal{F})(R)$. First, observe that:

1. $\mathsf{Rel}(\mathcal{F})(R)(\{\checkmark\}) = \{\checkmark\}$

2. $\mathsf{Rel}(\mathcal{F})(R)(\{a\} \times B) = \{a\} \times R(B)$ for all $a \in A$, and $B \subseteq X$

First, we have the following:

$$\nu_1(\checkmark) = \nu_1[\{\checkmark\}]$$

$$
\begin{aligned}
&\leq \nu_2[R(\{\checkmark\})] && \text{(Lemma A.1.4)}\\
&\leq \nu_2[\{\checkmark\}]\\
&\leq \nu_1[R^{-1}(\{\checkmark\})]\\
&\leq \nu_1(\checkmark)
\end{aligned}
$$

which proves $\nu_1(\checkmark) = \nu_2(\checkmark)$, establishing ①. Now, pick an arbitrary $a \in A$ and $B \subseteq X$. We have that:

$$
\begin{aligned}
\nu_1[\{a\} \times B] &\leq \nu_2[\mathsf{Rel}(\mathcal{F})(R)(\{a\} \times B)] && \text{(Lemma A.1.4)}\\
&= \nu_2[\{a\} \times R(B)]
\end{aligned}
$$

which proves ②. Symmetric reasoning involving R^{-1} allows to prove ③.

For the converse, assume that conditions ①, ② and ③ hold. Let $M \subseteq \mathcal{F}X$. We can partition M in the following way:

$$M = \{o \mid o \in \{\checkmark\} \cap M\} \cup \bigcup_{a \in A} \{a\} \times \{x \mid (a,x) \in M\}$$

We have the following:

$$
\begin{aligned}
\nu_1[B] &= \nu_1[\{o \mid o \in \{\checkmark\} \cap M\}] + \sum_{a \in A} \nu_1[\{a\} \times \{x \mid (a,x) \in M\}]\\
&= [\checkmark \in M]\ \nu_1(\checkmark) + \sum_{a \in A} \nu_1[\{a\} \times \{x \mid (a,x) \in M\}]\\
&\leq [\checkmark \in M]\ \nu_2(\checkmark) + \sum_{a \in A} \nu_2[\{a\} \times R(\{x \mid (a,x) \in M\})] && \text{(① and ②)}\\
&= \nu_2[\{o \mid o \in \{\checkmark\} \cap M\}] + \sum_{a \in A} \nu_2[\{a\} \times R(\{x \mid (a,x) \in M\})]\\
&= \nu_2[\{o \mid o \in \{\checkmark\} \cap M\}] + \nu_2[\mathsf{Rel}(\mathcal{F})(R)((A \times X) \cap M)]\\
&= \nu_2[\mathsf{Rel}(\mathcal{F})(R)(M)]
\end{aligned}
$$

Recall that relation liftings preserve inverse relations [HJ04] and therefore $\mathsf{Rel}(\mathcal{F})(R)^{-1} = \mathsf{Rel}(\mathcal{F})(R^{-1})$. A similar reasoning to the one before allows us to conclude that for all $N \subseteq \mathcal{F}Y$ we have that $\nu_2[N] \leq \nu_1[\mathsf{Rel}(\mathcal{F})(R)^{-1}(N)]$. Finally, we can apply Lemma A.1.4 to conclude that $\langle \nu_1, \nu_2 \rangle \in \mathsf{Lift}(\mathsf{Rel}(\mathcal{F})(R))$. □

Lemma A.2.5. *A relation $R \subseteq X \times Y$ is a bisimulation between $\mathcal{D}F$-coalgebras (X, β) and (Y, γ) if and only if for all $(x, y) \in R$, we have that:*

1. $\beta(x)(\checkmark) = \gamma(y)(\checkmark)$

2. *For all $a \in A$, and for all $B \subseteq X$, we have that:*

$$\beta(x)[\{a\} \times B] \leq \gamma(y)[\{a\} \times R(B)]$$

3. *For all $a \in A$, and for all $C \subseteq Y$, we have that:*

$$\gamma(y)[\{a\} \times C] \leq \gamma(y)[\{a\} \times R^{-1}(C)]$$

Proof. Consequence of Lemma A.2.2 and Lemma A.2.4. □

Lemma A.2.6. *Let (X, β) be a $\mathcal{D}F$-coalgebra, $R \subseteq X \times X$ be an equivalence relation, $\langle x, y \rangle \in R$ and $a \in A$. We have that:*

1. *For all $G \subseteq X$, $\beta(x)[\{a\} \times G] \leq \beta(y)[\{a\} \times R(G)]$*

2. *For all $H \subseteq X$, $\beta(y)[\{a\} \times H] \leq \beta(x)[\{a\} \times R^{-1}(H)]$*

if and only if $\beta(x)[\{a\} \times Q] = \beta(y)[\{a\} \times Q]$ for all equivalence classes $Q \in X/R$.

Proof. First assume that ① and ② do hold. We have that:

$$\begin{aligned} \beta(x)[\{a\} \times Q] &\leq \beta(y)[\{a\} \times R(Q)] \\ &= \beta(y)[\{a\} \times Q] && (R \text{ is an equivalence relation}) \end{aligned}$$

Since $R = R^{-1}$ we can employ the symmetric reasoning and show that $\beta(y)[\{a\} \times Q] \leq \beta(x)[\{a\} \times Q]$, which allows us to conclude that $\beta(x)[\{a\} \times Q] = \beta(y)[\{a\} \times Q]$.

For the converse, let $G \subseteq X$ be an arbitrary set. Let G/R be the quotient of G by the relation R and let X/R be the quotient of X by R. Observe that G/R is a partition of G and X/R is a partition of X.

For each equivalence class $P \in G/R$, there exists an equivalence class $Q_P \in X/R$, such that $P \subseteq Q_P = R(P)$, which implies that $\beta(x)[\{a\} \times P] \leq \beta(x)[\{a\} \times Q_P]$. Using additivity of subprobabilities of disjoint sets we can conclude the following:

$$\begin{aligned}
\beta(x)[\{a\} \times G] &= \beta(x)\left[\{a\} \times \bigcup_{P \in G/R} P\right] \\
&= \sum_{P \in G/R} \beta(x)[\{a\} \times P] \\
&\leq \sum_{P \in G/R} \beta(y)[\{a\} \times Q_P] \\
&= \beta(y)\left[\{a\} \times \bigcup_{P \in G/R} Q_P\right] \\
&= \beta(y)\left[\{a\} \times \bigcup_{P \in G/R} R(P)\right] \\
&= \beta(y)[\{a\} \times R(G)]
\end{aligned}$$

We can show ② via symmetric line of reasoning to the one above. □

A.3 Soundness argument

Given a set $Q \subseteq \mathsf{PExp}$ and an expression $f \in \mathsf{PExp}$, we will write:

$$Q/f = \{e \in \mathsf{PExp} \mid e \,; f \in Q\}$$

Lemma A.3.1. *If $R \subseteq \mathsf{PExp} \times \mathsf{PExp}$ is a congruence relation and $(e, f) \in R$ then for all $G \subseteq \mathsf{PExp}$, $R(G/e) \subseteq R(G)/f$.*

Proof. If $g \in R(G/e)$, then there exists some $h \in G/e$ such that $(h, g) \in R$. Since $h \in G/e$, also $h \,; e \in G$. Because R is a congruence relation, we have that $(h \,; e, g \,; f) \in R$

and hence $g\,;f \in R(G)$, which in turn implies that $g \in G/f$. □

Lemma A.3.2. *Let $e, f \in \mathsf{PExp}$ and let $Q \in \mathsf{PExp}/{\equiv_b}$. Now $(Q/f)/e = Q/e\,;f$*

Proof. Let $g \in Q$; we derive as follows

$$\begin{aligned} g \in (Q/f)/e &\iff g\,;e \in Q/f \iff (g\,;e)\,;f \in Q \\ &\iff g\,;(e\,;f) \in Q \iff g \in Q/e\,;f \end{aligned}$$

Here, the second to last step follows by associativity (S). □

Lemma A.3.3. *Let $e, f \in \mathsf{PExp}$, $R \subseteq \mathsf{PExp} \times \mathsf{PExp}$ a congruence relation and let $Q \in \mathsf{PExp}/R$. If $\langle e, f\rangle \in R$, then $Q/e = Q/f$*

Proof.

$$g \in Q/e \iff g\,;e \in Q \iff g\,;f \in Q \iff g \in Q/f$$

□

Lemma A.3.4. *For all $e, f \in \mathsf{PExp}$, $a \in A$, $G \subseteq \mathsf{PExp}$ we have that:*

$$\partial(e\,;f)[\{a\} \times G] = \partial(e)(\checkmark)\partial(f)[\{a\} \times G] + \partial(e)[\{a\} \times G/f]$$

Proof. A straightforward calculation using the definition of the Antimirov derivative.

$$\begin{aligned} \partial(e\,;f)[\{a\} \times G] &= \sum_{g \in G} \partial(e\,;f)(a,g) \\ &= \sum_{g \in G} \partial(e)(\checkmark)\partial(f)(a,g) + \sum_{g;f \in G} \partial(e)(a,g) \\ &= \partial(e)(\checkmark) \sum_{g \in G} \partial(f)(a,g) + \sum_{g;f \in G} \partial(e)(a,g) \\ &= \partial(e)(\checkmark)\partial(f)[\{a\} \times G] + \partial(e)[\{a\} \times G/f] \end{aligned}$$

□

Lemma A.3.5. *Let $e \in \mathsf{PExp}$, $r \in [0,1]$, $G \subseteq \mathsf{PExp}$ and $r\partial(e)(\checkmark) \neq 1$. We have that:*

$$\partial\left(e^{[r]}\right)[\{a\} \times G] = \frac{r\partial(e)[\{a\} \times G/e^{[r]}]}{1 - r\partial(e)(\checkmark)}$$

Proof. A straightforward calculation using the definition of the Antimirov derivative.

$$\begin{aligned}
\partial\left(e^{[r]}\right)[\{a\} \times G] &= \sum_{g \in G} \partial\left(e^{[r]}\right)(a,g) \\
&= \sum_{g; e^{[r]} \in G} \frac{r\partial(e)(a,g)}{1 - r\partial(e)(\checkmark)} \\
&= \frac{r\partial(e)[\{a\} \times G/e^{[r]}]}{1 - r\partial(e)(\checkmark)}
\end{aligned}$$

□

Lemma 4.4.1. *The relation $\equiv_b \subseteq \mathsf{PExp} \times \mathsf{PExp}$ is a bisimulation equivalence.*

Proof. We proceed by structural induction on the length derivation of $\equiv_b$, we show that all conditions of *Lemma A*.2.5 are satisfied. In most of the cases, we will rely on simpler coinditions from Lemma A.2.6. For the first few cases, we will rely on even simpler characterisation. In particular, observe that if for some $e, f \in \mathsf{PExp}$, $\partial(e) = \partial(f)$, then immediately $\partial(e)(\checkmark) = \partial(f)(\checkmark)$ and for all $a \in A$, $Q \in \mathsf{PExp}/{\equiv_b}$ we have that $\partial(e)[\{a\} \times Q] = \partial(f)[\{a\} \times Q]$. In other words, equality of subdistributions obtained by applying the coalgebra structure map to some pair states implies that they are bisimilar.

$\boxed{e \oplus_1 f \equiv_b e}$ For all $e, f \in \mathsf{PExp}$, $x \in \mathcal{F}\mathsf{PExp}$ we have that:

$$\partial(e \oplus_1 f)(x) = 1\partial(e)(x) + 0\partial(f)(x) = \partial(e)(x)$$

Since $\partial(e \oplus_1 f) = \partial(e)$, then $e \oplus_1 f$ and e are bisimilar.

$\boxed{e \oplus_p f \equiv_b f \oplus_{1-p} e}$ For all $e, f \in \mathsf{PExp}$, $p \in [0,1]$ and $x \in \mathcal{F}\mathsf{PExp}$ we have that

$$\partial(e \oplus_p f)(x) = p\partial(e)(x) + (1-p)\partial(f)$$

$$= (1-p)\partial(f)(x) + (1-(1-p))\partial(e)(x)$$
$$= \partial(f \oplus_{1-p} e)(x)$$

Since $\partial(e \oplus_p f) = \partial(f \oplus_{1-p} e)$, then $e \oplus_p f$ and $f \oplus_{1-p} e$ are bisimilar.

$\boxed{(e \oplus_p f) \oplus_q g \equiv_b e \oplus_{pq} \left(f \oplus_{\frac{(1-p)q}{1-pq}} g\right)}$ For all $e, f, g \in \mathsf{PExp}$, $p, q \in [0,1]$ such that $pq \neq 1$ and for all $x \in \mathcal{F}\mathsf{PExp}$ we have the following:

$$\begin{aligned}
\partial\left((e \oplus_p f) \oplus_q g\right)(x) &= q\partial(e \oplus_p f)(x) + (1-q)\partial(g)(x) \\
&= pq\partial(e)(x) + (1-p)q\partial(f)(x) + (1-q)\partial(g)(x) \\
&= pq\partial(e)(x) \\
&\qquad + (1-pq)\left(\frac{(1-p)q}{1-pq}\partial(f)(x) + \frac{1-q}{1-pq}\partial(g)(x)\right) \\
&= pq\partial(e)(x) + (1-pq)\partial\left(f \oplus_{\frac{(1-p)q}{1-pq}} g\right)(x) \\
&= \partial\left(e \oplus_{pq}\left(f \oplus_{\frac{(1-p)q}{1-pq}} g\right)\right)(x)
\end{aligned}$$

Since $\partial\left((e \oplus_p f) \oplus_q g\right) = \partial\left(e \oplus_{pq}\left(f \oplus_{\frac{(1-p)q}{1-pq}} g\right)\right)$, then $(e \oplus_p f) \oplus_q g$ and $e \oplus_{pq}\left(f \oplus_{\frac{(1-p)q}{1-pq}} g\right)$ are bisimilar.

$\boxed{e\,;1 \equiv_b e}$ For all $e \in \mathsf{PExp}$, we have that

$$\partial(e\,;1)(\checkmark) = \partial(e)(\checkmark)\delta_{\checkmark}(\checkmark) = \partial(e)(\checkmark)$$

For all $a \in A$ and $Q \in \mathsf{PExp}/{\equiv_b}$ we have that:

$$\partial(e\,;1)[\{a\} \times Q] = \partial(e)[\{a\} \times Q/1] + \partial(e)(\checkmark)\partial(1)[\{a\} \times Q]$$

(Lemma A.3.4)

$$= \partial(e)[\{a\} \times Q/1]$$
$$= \sum_{q;1 \in Q} \partial(e)(a,q)$$

$$= \sum_{q \in Q} \partial(e)(a,q) \qquad \textbf{(S1)}$$

$$= \partial(e)[\{a\} \times Q]$$

$\boxed{1\,;e \equiv_b e}$ For all $e \in \mathsf{PExp}$, we have that:

$$\partial(1\,;e)(\checkmark) = \delta_{\checkmark}(\checkmark)\partial(e)(\checkmark) = \partial(e)(\checkmark)$$

For all $a \in A$ and $Q \in \mathsf{PExp}/{\equiv_b}$ we have that:

$$\partial(1\,;e)[\{a\} \times Q] = \partial(1)[\{a\} \times Q/e] + \partial(1)(\checkmark)\partial(e)[\{a\} \times Q] \qquad \text{(Lemma A.3.4)}$$

$$= \partial(e)[\{a\} \times Q]$$

$\boxed{0\,;e \equiv_b 0}$ For all $e \in \mathsf{PExp}$ we have that:

$$\partial(0\,;e)(\checkmark) = \partial(0)(\checkmark)\partial(e)(\checkmark) = 0 = \partial(0)(\checkmark)$$

For all $a \in A$ and $Q \in \mathsf{PExp}/{\equiv_b}$ we have that:

$$\begin{aligned}\partial(0\,;e)[\{a\} \times Q] &= \partial(0)[\{a\} \times Q/e] + \partial(0)(\checkmark)\partial(e)[\{a\} \times Q] \\ &= 0 = \partial(0)[\{a\} \times Q]\end{aligned}$$

$\boxed{e\,;(f\,;g) \equiv_b (e\,;f)\,;g}$ For all $e,f,g \in \mathsf{PExp}$ we have that:

$$\begin{aligned}\partial(e\,;(f\,;g))(\checkmark) &= \partial(e)(\checkmark)\partial(f\,;g)(\checkmark) \\ &= \partial(e)(\checkmark)\partial(f)(\checkmark)\partial(g)(\checkmark) \\ &= \partial(e\,;f)(\checkmark)\partial(g)(\checkmark) \\ &= \partial((e\,;f)\,;g)(\checkmark)\end{aligned}$$

For all $a \in A$ and $Q \in \mathsf{PExp}/{\equiv_b}$ we have that:

$$\begin{aligned}
\partial(e\,;(f\,;g))[\{a\}\times Q] &= \partial(e)[\{a\}\times Q/f\,;g] + \partial(e)(\checkmark)\partial(f\,;g)[\{a\}\times Q] \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{(Lemma A.3.4)} \\
&= \partial(e)[\{a\}\times Q/g/f] + \partial(e)(\checkmark)\partial(f)[\{a\}\times Q/g] \\
&\qquad + \partial(e)(\checkmark)\partial(f)(\checkmark)\partial(g)[\{a\}\times Q] \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{(Lemma A.3.2)} \\
&= \partial(e\,;f)[\{a\}\times Q/f] + \partial(e\,;f)(\checkmark)\partial(g)[\{a\}\times Q] \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{(Lemma A.3.4)} \\
&= \partial((e\,;f)\,;g)[\{a\}\times Q] \qquad\qquad \text{(Lemma A.3.4)}
\end{aligned}$$

$\boxed{(e \oplus_p f)\,;g \equiv_b e\,;g \oplus_p f\,;g}$ For all $e, f, g \in \mathsf{PExp}$ and $p \in [0,1]$ we have that:

$$\begin{aligned}
\partial((e\oplus_p f)\,;g)(\checkmark) &= \partial(e\oplus_p f)(\checkmark)\partial(g)(\checkmark) \\
&= p\partial(e)(\checkmark)\partial(g)(\checkmark) + (1-p)\partial(f)(\checkmark)\partial(g)(\checkmark) \\
&= p\partial(e\,;g)(\checkmark) + (1-p)\partial(f\,;g)(\checkmark) \\
&= \partial(e\,;g\oplus_p f\,;g)(\checkmark)
\end{aligned}$$

For all $a \in A$ and $Q \in \mathsf{PExp}/{\equiv_b}$ we have that:

$$\begin{aligned}
&\partial((e\oplus_p f)\,;g)[\{a\}\times Q] \\
&\quad= \partial(e\oplus_p f)[\{a\}\times Q/g] + \partial(e\oplus_p f)(\checkmark)\partial(g)[\{a\}\times Q] \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{(Lemma A.3.4)} \\
&\quad= p\partial(e)[\{a\}\times Q] + p\partial(e)(\checkmark)[\{a\}\times Q] \\
&\qquad (1-p)\partial(f)[\{a\}\times Q] + (1-p)\partial(f)(\checkmark)[\{a\}\times Q] \\
&\quad= p\partial(e\,;g)[\{a\}\times Q] + (1-p)\partial(f\,;g)[\{a\}\times Q] \qquad \text{(Lemma A.3.4)} \\
&\quad= \partial(e\,;g\oplus_p f\,;g)[\{a\}\times Q]
\end{aligned}$$

$\boxed{e^{[p]} \equiv_b e\,;e^{[p]} \oplus_p 1}$ Let $e \in \mathsf{PExp}$ and $p \in [0,1]$. We distinguish two subcases.

If $\partial(e)(\checkmark) = 1$ and $p = 1$, then:

$$\partial(e^{[p]})(\checkmark) = 0 = \partial(e)(\checkmark)\partial(e^{[p]})(\checkmark) = \partial(e\,;e^{[p]} \oplus_p 1)(\checkmark)$$

For all $a \in A$ and $Q \in \mathsf{PExp}/{\equiv_b}$ we have that:

$$\begin{aligned}
\partial(e^{[p]})[\{a\} \times Q] = 0 & \\
&= \partial(e)[\{a\} \times Q] + \partial(e^{[p]})[\{a\} \times Q] \\
&= \partial(e)[\{a\} \times Q/e^{[p]}] + \partial(e)(\checkmark)\partial(e^{[p]})[\{a\} \times Q] \\
&= \partial(e\,;e^{[p]})[\{a\} \times Q] \\
&= \partial(e\,;e^{[p]} \oplus_p 1)[\{a\} \times Q]
\end{aligned}$$

From now on, we can safely assume that $p\partial(e)(\checkmark) \neq 1$. We have that:

$$\begin{aligned}
\partial(e^{[p]})(\checkmark) &= \frac{1-p}{1-p\partial(e)(\checkmark)} \\
&= \frac{(1-p)(1+p\partial(e)(\checkmark)-p\partial(e)(\checkmark))}{1-p\partial(e)(\checkmark)} \\
&= \frac{(1-p)(p\partial(e)(\checkmark))}{1-p\partial(e)(\checkmark)} + \frac{(1-p)(1-p\partial(e)(\checkmark))}{1-p\partial(e)(\checkmark)} \\
&= p\partial(e)(\checkmark)\frac{1-p}{1-p\partial(e)(\checkmark)} + (1-p) \\
&= p\partial(e)(\checkmark)\partial(e^{[p]})(\checkmark) + (1-p)
\end{aligned}$$

Let $a \in A$ and $Q \in \mathsf{PExp}/{\equiv_b}$. We have that:

$$\begin{aligned}
\partial(e^{[p]})[\{a\} \times Q] &= \frac{p\partial(e)[\{a\} \times Q/e^{[p]}]}{1-p\partial(e)(\checkmark)} && \text{(Lemma A.3.5)} \\
&= p\partial(e)[\{a\} \times Q/e^{[p]}]\frac{1}{1-p\partial(e)(\checkmark)} \\
&= p\partial(e)[\{a\} \times Q/e^{[p]}]\frac{1-p\partial(e)(\checkmark)+p\partial(e)(\checkmark)}{1-p\partial(e)(\checkmark)} \\
&= p\partial(e)[\{a\} \times Q/e^{[p]}]\left(1+\frac{p\partial(e)(\checkmark)}{1-p\partial(e)(\checkmark)}\right)
\end{aligned}$$

$$
\begin{aligned}
&= p\partial(e)[\{a\}\times Q/e^{[p]}] + p\partial(e)(\checkmark)\frac{p\partial(e)[\{a\}\times Q/e^{[p]}]}{1-p\partial(e)(\checkmark)}\\
&= p\partial(e)[\{a\}\times Q/e^{[p]}] + p\partial(e)(\checkmark)\partial(e^{[p]})[\{a\}\times Q]\\
&\hspace{20em}\text{(Lemma A.3.5)}\\
&= p\partial(e\,;e^{[p]})[\{a\}\times Q] \qquad\qquad \text{(Lemma A.3.4)}\\
&= \partial(e\,;e^{[p]}\oplus_p 1)[\{a\}\times Q]
\end{aligned}
$$

$$\boxed{(e\oplus_p 1)^{[q]} \equiv_b e^{\left[\frac{pq}{1-(1-p)q}\right]}}$$

Let $e\in\mathsf{PExp}$ and let $p,q\in[0,1]$ such that $(1-p)q\neq 1$. Observe, that in such a situation $\frac{pq}{1-(1-p)q}\neq 1$. First, consider the following:

$$
\begin{aligned}
\partial\left((e\oplus_p 1)^{[q]}\right)(\checkmark) &= \frac{1-q}{1-q\partial(e\oplus_p 1)(\checkmark)}\\
&= \frac{1-q}{1-q(1-p)-pq\partial(e)(\checkmark)}\\
&= \frac{1-q}{(1-q(1-p))\left(1-\frac{pq}{1-q(1-p)}\partial(e)(\checkmark)\right)}\\
&= \frac{\frac{1-q}{1-q(1-p)}}{1-\frac{pq}{1-q(1-p)}\partial(e)(\checkmark)}\\
&= \frac{1-\frac{pq}{1-q(1-p)}}{1-\frac{pq}{1-q(1-p)}\partial(e)(\checkmark)}\\
&= \partial\left(e^{\left[\frac{pq}{1-(1-p)q}\right]}\right)(\checkmark)
\end{aligned}
$$

For all $a\in A$ and $Q\in\mathsf{PExp}/{\equiv_b}$ we have that the following holds:

$$
\begin{aligned}
\partial\left((e\oplus_p 1)^{[q]}\right)[\{a\}\times Q] &= \frac{q\partial(e\oplus_p 1)[\{a\}\times Q/(e\oplus_p 1)^{[q]}]}{1-q\partial(e\oplus_p 1)(\checkmark)}\\
&\hspace{20em}\text{(Lemma A.3.5)}\\
&= \frac{pq\partial(e)[\{a\}\times Q/(e\oplus_p 1)^{[q]}]}{1-(1-p)q-pq\partial(e)(\checkmark)}\\
&= \frac{pq\partial(e)[\{a\}\times Q/(e\oplus_p 1)^{[q]}]}{(1-(1-p)q)\left(1-\frac{pq}{1-(1-p)q}\partial(e)(\checkmark)\right)}
\end{aligned}
$$

$$= \frac{\frac{pq}{1-(1-p)q}\partial(e)[\{a\}\times Q/(e\oplus_p 1)^{[q]}]}{(1-(1-p)q)\left(1-\frac{pq}{1-(1-p)q}\partial(e)(\checkmark)\right)}$$

$$= \frac{\frac{pq}{1-(1-p)q}\partial(e)[\{a\}\times Q/e^{\left[\frac{pq}{1-(1-p)q}\right]}]}{(1-(1-p)q)\left(1-\frac{pq}{1-(1-p)q}\partial(e)(\checkmark)\right)}$$

(Lemma A.3.3)

$$= \partial\left(e^{\left[\frac{pq}{1-(1-p)q}\right]}\right)[\{a\}\times Q]$$

$\boxed{\text{From } g \equiv_b e\,;g\oplus_p f \text{ and } E(g)=0 \text{ derive } g\equiv_b e^{[p]}\,;f}$ Let $e,f,g \in \mathsf{PExp}$, such that $g \equiv e\,;g\oplus_p f$ and $E(e)=0$. Recall that by Lemma 4.3.5, we have that $\partial(e)(\checkmark)=0$. First, observe that:

$$\begin{aligned}
\partial(g)(\checkmark) &= \partial(e\,;g\oplus_p f)(\checkmark) && \text{(Induction hypothesis)}\\
&= p\partial(e)(\checkmark)\partial(g)(\checkmark)+(1-p)\partial(f)(\checkmark)\\
&= (1-p)\partial(f)(\checkmark)\\
&= \frac{1-p}{1-p\partial(e)(\checkmark)}\partial(f)(\checkmark)\\
&= \partial(e^{[p]})(\checkmark)\partial(f)(\checkmark)\\
&= \partial(e^{[p]}\,;f)(\checkmark)
\end{aligned}$$

For all $a\in A$ and $Q\in \mathsf{PExp}/{\equiv_b}$ we have that:

$$\begin{aligned}
\partial(g)[\{a\}\times Q] &= \partial(e\,;g\oplus_p f)[\{a\}\times Q]\\
&\qquad\qquad\qquad\qquad \text{(Induction hypothesis)}\\
&= p\partial(e\,;g)[\{a\}\times Q]+(1-p)\partial(f)[\{a\}\times Q]\\
&= p\partial(e)[\{a\}\times Q/g]+p\partial(e)(\checkmark)\partial(g)[\{a\}\times Q]\\
&\quad +(1-p)\partial(f)[\{a\}\times Q]\\
&= p\partial(e)[\{a\}\times Q/g]+(1-p)\partial(f)[\{a\}\times Q]\\
&= p\partial(e)[\{a\}\times Q/e^{[p]}\,;f]+(1-p)\partial(f)[\{a\}\times Q]\\
&\qquad\qquad\qquad\qquad \text{(Lemma A.3.3)}
\end{aligned}$$

$$
\begin{aligned}
&= p\partial(e)[\{a\} \times (Q/f)/e^{[p]}] + (1-p)\partial(f)[\{a\} \times Q] && \text{(Lemma A.3.2)} \\
&= \frac{p\partial(e)[\{a\} \times (Q/f)/e^{[p]}]}{1 - p\partial(e)(\checkmark)} + \frac{1-p}{1-p\partial(e)(\checkmark)}\partial(f)[\{a\} \times Q] \\
&= \partial(e^{[p]})[\{a\} \times Q/f] + \partial(e^{[p]})(\checkmark)\partial(f)[\{a\} \times Q] && \text{(Lemma A.3.5)} \\
&= \partial(e^{[p]} \,;f)[\{a\} \times Q] && \text{(Lemma A.3.4)}
\end{aligned}
$$

$\boxed{\text{reflexivity, transitivity and symmetry}}$ We omit the proof, as it is trivial.

$\boxed{\text{From } e \equiv_b g \text{ and } f \equiv_b h \text{ derive that } e \oplus_p f \equiv_b g \oplus_p h}$ Let $e, f, g, h \in \mathsf{PExp}$, such that $e \equiv_b g$ and $f \equiv_b h$. We have that

$$
\begin{aligned}
\partial(e \oplus_p f)(\checkmark) &= p\partial(e)(\checkmark) + (1-p)\partial(f)(\checkmark) \\
&= p\partial(g)(\checkmark) + (1-p)\partial(h)(\checkmark) && \text{(Induction hypothesis)} \\
&= \partial(g \oplus_p h)(\checkmark)
\end{aligned}
$$

For all $a \in A$ and $Q \in \mathsf{PExp}/{\equiv_b}$ we have that:

$$
\begin{aligned}
\partial(e \oplus_p f)[\{a\} \times Q] &= p\partial(e)[\{a\} \times Q] + (1-p)\partial(f)[\{a\} \times Q] \\
&= p\partial(g)[\{a\} \times Q] + (1-p)\partial(h)[\{a\} \times Q] && \text{(Induction hypothesis)} \\
&= \partial(g \oplus_p h)[\{a\} \times Q]
\end{aligned}
$$

$\boxed{\text{From } e \equiv_b g \text{ and } f \equiv_b h \text{ derive that } e \,; f \equiv_b g \,; h}$ Let $e, f, g, h \in \mathsf{PExp}$, such that $e \equiv_b g$ and $f \equiv_b h$. We have that:

$$
\begin{aligned}
\partial(e \,; f)(\checkmark) &= \partial(e)(\checkmark)\partial(f)(\checkmark) \\
&= \partial(g)(\checkmark)\partial(h)(\checkmark) && \text{(Induction hypothesis)} \\
&= \partial(g \,; h)(\checkmark)
\end{aligned}
$$

For all $a \in A$ and $G \subseteq \mathsf{PExp}$, we have that:

$$\begin{aligned}
\partial(e\,;f)[\{a\} \times G] &= \partial(e)[\{a\} \times G/f] + \partial(e)(\checkmark)\partial(f)[\{a\} \times G] && \text{(Lemma A.3.4)}\\
&\leq \partial(g)[\{a\} \times R(G/f)] + \partial(g)(\checkmark)\partial(h)[\{a\} \times R(G)]\\
&\leq \partial(g)[\{a\} \times R(G)/h] + \partial(g)(\checkmark)\partial(h)[\{a\} \times R(G)]\\
& && \text{(Lemma A.3.1)}\\
&= \partial(g\,;h)[\{a\} \times R(G)] && \text{(Lemma A.3.4)}
\end{aligned}$$

Condition that $\partial(g\,;h)[\{a\} \times G] \leq \partial(e\,;f)[\{a\} \times R^{-1}(G)]$ can be shown by a symmetric argument.

$\boxed{\text{From } e \equiv_b f \text{ derive } e^{[p]} \equiv_b f^{[p]}}$ Let $e, f \in \mathsf{PExp}$ such that $e \equiv_b f$. We distinguish two subcases. First, consider the situation when $p = 1$ and $\partial(e)(\checkmark) = 1$. Observe that by induction hypothesis we have that $\partial(f)(\checkmark) = 1$. We have that $\partial(e^{[p]}) = \mathbb{0} = \partial(f^{[p]})$. Hence, $e^{[p]}$ and $f^{[p]}$ are bisimilar.

From now on, we can safely assume that $p\partial(e)(\checkmark) \neq 1$ and $p\partial(f)(\checkmark) \neq 1$. We have that:

$$\begin{aligned}
\partial\left(e^{[p]}\right)(\checkmark) &= \frac{1-p}{1-p\partial(e)(\checkmark)}\\
&= \frac{1-p}{1-p\partial(f)(\checkmark)} && \text{(Induction hypothesis)}\\
&= \partial\left(f^{[p]}\right)(\checkmark)
\end{aligned}$$

For all $a \in A$ and $G \subseteq \mathsf{PExp}$ we have that:

$$\begin{aligned}
\partial\left(e^{[p]}\right)[\{a\} \times G] &= \frac{p\partial(e)[\{a\} \times G/e^{[p]}]}{1-p\partial(e)(\checkmark)} && \text{(Lemma A.3.5)}\\
&\leq \frac{p\partial(f)[\{a\} \times R(G/e^{[p]})]}{1-p\partial(f)(\checkmark)} && \text{(Induction hypothesis)}\\
&\leq \frac{p\partial(f)[\{a\} \times R(G)/f^{[p]}]}{1-p\partial(f)(\checkmark)} && \text{(Lemma A.3.1)}\\
&= \partial\left(e^{[p]}\right)[\{a\} \times R(G)]
\end{aligned}$$

Condition that $\partial\left(f^{[p]}\right)[\{a\} \times G] \leq \partial\left(e^{[p]}\right)[\{a\} \times R^{-1}(G)]$ can be shown by a symmetric argument. □

Bibliography

[AK95] Jiří Adámek and Václav Koubek. “On the greatest fixed point of a set functor”. In: *Theoretical Computer Science* 150.1 (1995), pp. 57–75. ISSN: 0304-3975 (cit. on p. 33).

[AMV06] Jiří Adámek, Stefan Milius, and Jiri Velebil. “Iterative algebras at work”. In: *Math. Struct. Comput. Sci.* 16.6 (2006), pp. 1085–1131 (cit. on p. 90).

[And+14] Carolyn Jane Anderson, Nate Foster, Arjun Guha, Jean-Baptiste Jeannin, Dexter Kozen, Cole Schlesinger, and David Walker. “NetKAT: semantic foundations for networks”. In: *POPL*. 2014, pp. 113–126 (cit. on pp. 4, 11, 75).

[Ant+25] Thibaut Antoine, Robin Piedeleu, Alexandra Silva, and Fabio Zanasi. “A Complete Diagrammatic Calculus for Automata Simulation”. In: *33rd EACSL Annual Conference on Computer Science Logic, CSL 2025, February 10-14, 2025, Amsterdam, Netherlands*. Ed. by Jörg Endrullis and Sylvain Schmitz. Vol. 326. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2025, 27:1–27:22 (cit. on pp. 18, 69).

[Ant96] Valentin Antimirov. “Partial derivatives of regular expressions and finite automaton constructions”. In: *Theoretical Computer Science* 155.2 (1996), pp. 291–319. ISSN: 0304-3975 (cit. on pp. 75, 91, 95).

[AR94] J. Adamek and J. Rosicky. *Locally Presentable and Accessible Categories*. London Mathematical Society Lecture Note Series. Cambridge University Press, 1994 (cit. on pp. 82, 129).

[Ard61] Dean N. Arden. "Delayed-logic and finite-state machines". In: *2nd Annual Symposium on Switching Circuit Theory and Logical Design (SWCT 1961).* 1961, pp. 133–151 (cit. on pp. 11, 138).

[AT11] Samson Abramsky and Nikos Tzevelekos. "Introduction to Categories and Categorical Logic". In: *CoRR* abs/1102.1313 (2011). arXiv: 1102.1313 (cit. on p. 23).

[Bac+18a] Giorgio Bacci, Giovanni Bacci, Kim G. Larsen, and Radu Mardare. "A Complete Quantitative Deduction System for the Bisimilarity Distance on Markov Chains". In: *Log. Methods Comput. Sci.* 14.4 (2018) (cit. on pp. 14, 21, 22, 34, 40, 45, 47, 53, 54, 154).

[Bac+18b] Giorgio Bacci, Giovanni Bacci, Kim G. Larsen, and Radu Mardare. "Complete Axiomatization for the Total Variation Distance of Markov Chains". In: *Proceedings of the Thirty-Fourth Conference on the Mathematical Foundations of Programming Semantics, MFPS 2018, Dalhousie University, Halifax, Canada, June 6-9, 2018*. Ed. by Sam Staton. Vol. 341. Electronic Notes in Theoretical Computer Science. Elsevier, 2018, pp. 27–39 (cit. on pp. 14, 21, 53, 154).

[Bac+18c] Giorgio Bacci, Radu Mardare, Prakash Panangaden, and Gordon D. Plotkin. "An Algebraic Theory of Markov Processes". In: *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2018, Oxford, UK, July 09-12, 2018*. Ed. by Anuj Dawar and Erich Grädel. ACM, 2018, pp. 679–688 (cit. on pp. 14, 21, 53).

[Bac+24] Giorgio Bacci, Radu Mardare, Prakash Panangaden, and Gordon Plotkin. "Sum and Tensor of Quantitative Effects". In: *Logical Methods in Computer Science* Volume 20, Issue 4, 9 (Oct. 2024). ISSN: 1860-5974 (cit. on p. 53).

[Bac76] Roland Carl Backhouse. "Closure algorithms and the star-height problem of regular languages". PhD thesis. Imperial College London, UK, 1976 (cit. on p. 102).

[Bal+18] Paolo Baldan, Filippo Bonchi, Henning Kerstan, and Barbara König. “Coalgebraic Behavioral Metrics”. In: *Log. Methods Comput. Sci.* 14.3 (2018) (cit. on pp. 13, 16, 22, 30–34, 45, 62, 63, 154).

[Bee17] Tobias Beeh. “Transformations between Markov Chains and Stochastic Regular Expressions”. MA thesis. University of Stuttgart, 2017 (cit. on pp. 15, 153).

[Ber22] Marco Bernardo. “Probabilistic Trace and Testing Semantics: The Importance of Being Coherent”. In: *Found. Trends Program. Lang.* 7.4 (2022), pp. 244–332 (cit. on p. 152).

[BKP18] Filippo Bonchi, Barbara König, and Daniela Petrisan. “Up-To Techniques for Behavioural Metrics via Fibrations”. In: *29th International Conference on Concurrency Theory, CONCUR 2018, September 4-7, 2018, Beijing, China*. Ed. by Sven Schewe and Lijun Zhang. Vol. 118. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018, 17:1–17:17 (cit. on p. 13).

[BMS13] Marcello M. Bonsangue, Stefan Milius, and Alexandra Silva. “Sound and Complete Axiomatizations of Coalgebraic Language Equivalence”. In: *ACM Trans. Comput. Log.* 14.1 (2013), 7:1–7:52 (cit. on p. 153).

[Bof90] Maurice Boffa. “Une remarque sur les systèmes complets d’identités rationnelles”. In: *RAIRO Theor. Informatics Appl.* 24 (1990), pp. 419–423 (cit. on pp. 11, 74).

[Bre12] Franck van Breugel. “On behavioural pseudometrics and closure ordinals”. In: *Inf. Process. Lett.* 112.19 (2012), pp. 715–718 (cit. on pp. 47, 48).

[Brz64] Janusz A. Brzozowski. “Derivatives of Regular Expressions”. In: *J. ACM* 11.4 (1964), pp. 481–494 (cit. on pp. 22, 27, 29, 50, 76, 102, 137).

[BS01] Emanuele Bandini and Roberto Segala. “Axiomatizations for Probabilistic Bisimulation”. In: *Automata, Languages and Programming, 28th International Colloquium, ICALP 2001, Crete, Greece, July 8-12, 2001, Proceedings*. Ed. by Fernando Orejas, Paul G. Spirakis, and Jan van Leeuwen. Vol. 2076. Lecture Notes in Computer Science. Springer, 2001, pp. 370–381 (cit. on p. 152).

[BS81] Stanley Burris and H P Sankappanavar. *A Course in Universal Algebra*. en. Lecture Notes in Statistics. New York, NY: Springer, Nov. 1981 (cit. on pp. 35, 37).

[BSS17] Filippo Bonchi, Alexandra Silva, and Ana Sokolova. “The Power of Convex Algebras”. In: *28th International Conference on Concurrency Theory (CONCUR 2017)*. Ed. by Roland Meyer and Uwe Nestmann. Vol. 85. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2017, 23:1–23:18. ISBN: 978-3-95977-048-4 (cit. on pp. 86, 153).

[BW01] Franck van Breugel and James Worrell. “Towards Quantitative Verification of Probabilistic Transition Systems”. In: *Automata, Languages and Programming, 28th International Colloquium, ICALP 2001, Crete, Greece, July 8-12, 2001, Proceedings*. Ed. by Fernando Orejas, Paul G. Spirakis, and Jan van Leeuwen. Vol. 2076. Lecture Notes in Computer Science. Springer, 2001, pp. 421–432 (cit. on pp. 4, 13).

[Cas+21] Pablo Samuel Castro, Tyler Kastner, Prakash Panangaden, and Mark Rowland. “MICo: Improved representations via sampling-based state similarity for Markov decision processes”. In: *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*. Ed. by Marc’Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan. 2021, pp. 30113–30126 (cit. on p. 5).

[Che+22] Mingshuai Chen, Joost-Pieter Katoen, Lutz Klinkenberg, and Tobias Winkler. "Does a Program Yield the Right Distribution? - Verifying Probabilistic Programs via Generating Functions". In: *Computer Aided Verification - 34th International Conference, CAV 2022, Haifa, Israel, August 7-10, 2022, Proceedings, Part I*. Ed. by Sharon Shoham and Yakir Vizel. Vol. 13371. Lecture Notes in Computer Science. Springer, 2022, pp. 79–101 (cit. on p. 75).

[Con12] J.H. Conway. *Regular Algebra and Finite Machines*. Chapman and Hall mathematics series. Dover Publications, Incorporated, 2012. ISBN: 9780486485836 (cit. on p. 11).

[Des+04] Josée Desharnais, Vineet Gupta, Radha Jagadeesan, and Prakash Panangaden. "Metrics for labelled Markov processes". In: *Theor. Comput. Sci.* 318.3 (2004), pp. 323–354 (cit. on pp. 4, 13, 53).

[DGL14] Pedro R. D'Argenio, Daniel Gebler, and Matias David Lee. "Axiomatizing Bisimulation Equivalences and Metrics from Probabilistic SOS Rules". In: *Foundations of Software Science and Computation Structures - 17th International Conference, FOSSACS 2014, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2014, Grenoble, France, April 5-13, 2014, Proceedings*. Ed. by Anca Muscholl. Vol. 8412. Lecture Notes in Computer Science. Springer, 2014, pp. 289–303 (cit. on p. 53).

[Dob08] Ernst-Erich Doberkat. "Erratum and Addendum: Eilenberg-Moore algebras for stochastic relations". In: *Inf. Comput.* 206.12 (2008), pp. 1476–1484 (cit. on pp. 87, 89).

[DP02] B. A. Davey and H. A. Priestley. *Introduction to Lattices and Order*. 2nd ed. Cambridge University Press, 2002 (cit. on p. 23).

[Ési99] Z Ésik. "Group Axioms for Iteration". In: *Information and Computation* 148.2 (1999), pp. 131–180. ISSN: 0890-5401 (cit. on pp. 66, 67).

[FHS22] Tiago Ferreira, Gerco van Heerdt, and Alexandra Silva. “Tree-Based Adaptive Model Learning”. In: *A Journey from Process Algebra via Timed Automata to Model Learning - Essays Dedicated to Frits Vaandrager on the Occasion of His 60th Birthday*. Ed. by Nils Jansen, Mariëlle Stoelinga, and Petra van den Bos. Vol. 13560. Lecture Notes in Computer Science. Springer, 2022, pp. 164–179 (cit. on p. 53).

[GPG18] Sinem Getir, Esteban Pavese, and Lars Grunske. “Formal Semantics for Probabilistic Verification of Stochastic Regular Expressions”. In: *Proceedings of the 27th International Workshop on Concurrency, Specification and Programming, Berlin, Germany, September 24-26, 2018*. Ed. by Bernd-Holger Schlingloff and Samira Akili. Vol. 2240. CEUR Workshop Proceedings. CEUR-WS.org, 2018 (cit. on p. 153).

[Gra22] Clemens Armin Grabmayer. “Milner’s Proof System for Regular Expressions Modulo Bisimilarity is Complete: Crystallization: Near-Collapsing Process Graph Interpretations of Regular Expressions”. In: *LICS ’22: 37th Annual ACM/IEEE Symposium on Logic in Computer Science, Haifa, Israel, August 2 - 5, 2022*. Ed. by Christel Baier and Dana Fisman. ACM, 2022, 34:1–34:13 (cit. on p. 154).

[GSS95] Rob J. van Glabbeek, Scott A. Smolka, and Bernhard Steffen. “Reactive, Generative and Stratified Models of Probabilistic Processes”. In: *Inf. Comput.* 121.1 (1995), pp. 59–80 (cit. on pp. 5, 19, 74, 92).

[Gum00] H. Peter Gumm. *Elements Of The General Theory Of Coalgebras*. 2000 (cit. on pp. 15, 22, 90, 121, 150).

[Hag00] Esfandiar Haghverdi. “A categorical approach to linear logic, geometry of proofs and full completeness”. PhD thesis. University of Ottawa, Canada, 2000 (cit. on p. 67).

[Has97] Masahito Hasegawa. “Models of sharing graphs : a categorical semantics of let and letrec”. PhD thesis. University of Edinburgh, UK, 1997 (cit. on p. 67).

[HJ04] Jesse Hughes and Bart Jacobs. "Simulations in coalgebra". In: *Theor. Comput. Sci.* 327.1-2 (2004), pp. 71–108 (cit. on p. 159).

[HJS07] Ichiro Hasuo, Bart Jacobs, and Ana Sokolova. "Generic Trace Semantics via Coinduction". In: *Log. Methods Comput. Sci.* 3.4 (2007) (cit. on p. 153).

[Hsu17] Justin Hsu. "Probabilistic Couplings for Probabilistic Reasoning". PhD thesis. University of Pennsylvania, 2017 (cit. on pp. 156, 157).

[Jac06] Bart Jacobs. "A Bialgebraic Review of Deterministic Automata, Regular Expressions and Languages". In: *Algebra, Meaning, and Computation, Essays Dedicated to Joseph A. Goguen on the Occasion of His 65th Birthday*. Ed. by Kokichi Futatsugi, Jean-Pierre Jouannaud, and José Meseguer. Vol. 4060. Lecture Notes in Computer Science. Springer, 2006, pp. 375–404 (cit. on pp. 129, 153).

[Jac10] Bart Jacobs. "Convexity, Duality and Effects". In: *Theoretical Computer Science - 6th IFIP TC 1/WG 2.2 International Conference, TCS 2010, Held as Part of WCC 2010, Brisbane, Australia, September 20-23, 2010. Proceedings*. Ed. by Cristian S. Calude and Vladimiro Sassone. Vol. 323. IFIP Advances in Information and Communication Technology. Springer, 2010, pp. 1–19 (cit. on p. 89).

[JSS15] Bart Jacobs, Alexandra Silva, and Ana Sokolova. "Trace semantics via determinization". In: *J. Comput. Syst. Sci.* 81.5 (2015), pp. 859–879 (cit. on pp. 83, 94, 153).

[Kap+18] Tobias Kappé, Paul Brunet, Alexandra Silva, and Fabio Zanasi. "Concurrent Kleene Algebra: Free Model and Completeness". In: *ESOP*. Vol. 10801. Lecture Notes in Computer Science. Springer, 2018, pp. 856–882 (cit. on p. 11).

[Kie+11] Stefan Kiefer, Andrzej S. Murawski, Joël Ouaknine, Björn Wachter, and James Worrell. "Language Equivalence for Probabilistic Automata". In: *Computer Aided Verification - 23rd International Conference, CAV*

2011, Snowbird, UT, USA, July 14-20, 2011. Proceedings. Ed. by Ganesh Gopalakrishnan and Shaz Qadeer. Vol. 6806. Lecture Notes in Computer Science. Springer, 2011, pp. 526–540 (cit. on pp. 14, 153).

[Kie+12] Stefan Kiefer, Andrzej S. Murawski, Joël Ouaknine, Björn Wachter, and James Worrell. "APEX: An Analyzer for Open Probabilistic Programs". In: *Computer Aided Verification - 24th International Conference, CAV 2012, Berkeley, CA, USA, July 7-13, 2012 Proceedings*. Ed. by P. Madhusudan and Sanjit A. Seshia. Vol. 7358. Lecture Notes in Computer Science. Springer, 2012, pp. 693–698 (cit. on pp. 5, 14, 153).

[KK05] Lucja Kot and Dexter Kozen. "Kleene Algebra and Bytecode Verification". In: *Proceedings of the First Workshop on Bytecode Semantics, Verification, Analysis and Transformation, Bytecode@ETAPS 2005, Edinburgh, UK, April 9, 2005*. Ed. by Fausto Spoto. Vol. 141. Electronic Notes in Theoretical Computer Science 1. Elsevier, 2005, pp. 221–236 (cit. on p. 75).

[Kle51] S.C. Kleene. *Representation of Events in Nerve Nets and Finite Automata*. Memorandum (Rand Corporation). Rand Corporation, 1951 (cit. on pp. 10, 74).

[Koz94] Dexter Kozen. "A Completeness Theorem for Kleene Algebras and the Algebra of Regular Events". In: *Inf. Comput.* 110.2 (1994), pp. 366–390 (cit. on pp. 4, 11, 41, 74, 129, 154).

[KP00] Dexter Kozen and Maria-Christina Patron. "Certification of Compiler Optimizations Using Kleene Algebra with Tests". In: *Computational Logic - CL 2000, First International Conference, London, UK, 24-28 July, 2000, Proceedings*. Ed. by John W. Lloyd, Veronica Dahl, Ulrich Furbach, Manfred Kerber, Kung-Kiu Lau, Catuscia Palamidessi, Luis Moniz Pereira, Yehoshua Sagiv, and Peter J. Stuckey. Vol. 1861. Lecture Notes in Computer Science. Springer, 2000, pp. 568–582 (cit. on p. 75).

[KR15] Bartek Klin and Jurriaan Rot. “Coalgebraic Trace Semantics via Forgetful Logics”. In: *Foundations of Software Science and Computation Structures - 18th International Conference, FoSSaCS 2015, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2015, London, UK, April 11-18, 2015. Proceedings*. Ed. by Andrew M. Pitts. Vol. 9034. Lecture Notes in Computer Science. Springer, 2015, pp. 151–166 (cit. on p. 153).

[Kro90] Daniel Krob. “A Complete System of B-Rational Identities”. In: *Automata, Languages and Programming, 17th International Colloquium, ICALP90, Warwick University, England, UK, July 16-20, 1990, Proceedings*. Ed. by Mike Paterson. Vol. 443. Lecture Notes in Computer Science. Springer, 1990, pp. 60–73 (cit. on pp. 11, 74).

[KS96] Dexter Kozen and Frederick Smith. “Kleene Algebra with Tests: Completeness and Decidability”. In: *CSL*. Vol. 1258. Lecture Notes in Computer Science. Springer, 1996, pp. 244–259 (cit. on p. 11).

[Kwi90] Marta Z. Kwiatkowska. “A Metric for Traces”. In: *Inf. Process. Lett.* 35.3 (1990), pp. 129–135 (cit. on p. 53).

[LFT11] Kim G. Larsen, Uli Fahrenberg, and Claus R. Thrane. “Metrics for weighted transition systems: Axiomatization and complexity”. In: *Theor. Comput. Sci.* 412.28 (2011), pp. 3358–3369 (cit. on p. 53).

[Lob+24] Gabriele Lobbia, Wojciech Różowski, Ralph Sarkis, and Fabio Zanasi. “Quantitative Monoidal Algebra: Axiomatising Distance with String Diagrams”. In: *CoRR* abs/2410.09229 (2024). arXiv: `2410.09229` (cit. on p. 5).

[LS91] Kim Guldstrand Larsen and Arne Skou. “Bisimulation through Probabilistic Testing”. In: *Inf. Comput.* 94.1 (1991), pp. 1–28 (cit. on pp. 14, 91, 101, 103).

[Mil10] Stefan Milius. “A Sound and Complete Calculus for Finite Stream Circuits”. In: *Proceedings of the 25th Annual IEEE Symposium on Logic in Computer Science, LICS 2010, 11-14 July 2010, Edinburgh, United Kingdom*. IEEE Computer Society, 2010, pp. 421–430 (cit. on pp. 16, 90, 129, 148, 153).

[Mil18] Stefan Milius. “Proper Functors and Fixed Points for Finite Behaviour”. In: *Log. Methods Comput. Sci.* 14.3 (2018) (cit. on pp. 5, 16, 76, 102, 132, 153).

[Mil84] Robin Milner. “A Complete Inference System for a Class of Regular Behaviours”. In: *J. Comput. Syst. Sci.* 28.3 (1984), pp. 439–466 (cit. on pp. 12, 17, 18, 42, 58, 59, 102, 154).

[MOW03] Michael W. Mislove, Joël Ouaknine, and James Worrell. “Axioms for Probability and Nondeterminism”. In: *Proceedings of the 10th International Workshop on Expressiveness in Concurrency, EXPRESS 2003, Marseille, France, September 2, 2003*. Ed. by Flavio Corradini and Uwe Nestmann. Vol. 96. Electronic Notes in Theoretical Computer Science. Elsevier, 2003, pp. 7–28 (cit. on p. 152).

[MPP16] Radu Mardare, Prakash Panangaden, and Gordon D. Plotkin. “Quantitative Algebraic Reasoning”. In: *Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science, LICS ’16, New York, NY, USA, July 5-8, 2016*. Ed. by Martin Grohe, Eric Koskinen, and Natarajan Shankar. ACM, 2016, pp. 700–709 (cit. on pp. 17, 21, 22, 34, 36, 39, 41, 52, 53, 154).

[MPP21] Radu Mardare, Prakash Panangaden, and Gordon D. Plotkin. “Fixed-Points for Quantitative Equational Logics”. In: *36th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2021, Rome, Italy, June 29 - July 2, 2021*. IEEE, 2021, pp. 1–13 (cit. on p. 54).

[MPW20] Stefan Milius, Dirk Pattinson, and Thorsten Wißmann. “A new foundation for finitary corecursion and iterative algebras”. In: *Inf. Comput.* 271 (2020), p. 104456 (cit. on pp. 90, 150, 153).

[Par81] David Park. “Concurrency and automata on infinite sequences”. In: *Theoretical Computer Science*. Ed. by Peter Deussen. Berlin, Heidelberg: Springer Berlin Heidelberg, 1981, pp. 167–183. ISBN: 978-3-540-38561-5 (cit. on p. 11).

[PZ23a] Robin Piedeleu and Fabio Zanasi. “A Finite Axiomatisation of Finite-State Automata Using String Diagrams”. In: *Log. Methods Comput. Sci.* 19.1 (2023) (cit. on pp. 18, 69).

[PZ23b] Robin Piedeleu and Fabio Zanasi. “An Introduction to String Diagrams for Computer Scientists”. In: *CoRR* abs/2305.08768 (2023). arXiv: `2305.08768` (cit. on pp. 17, 69).

[Rab63] Michael O. Rabin. “Probabilistic automata”. In: *Information and Control* 6.3 (1963), pp. 230–245. ISSN: 0019-9958 (cit. on pp. 11, 14, 92, 153).

[Rab93] Alexander Moshe Rabinovich. “A Complete Axiomatisation for Trace Congruence of Finite State Behaviors”. In: *Mathematical Foundations of Programming Semantics, 9th International Conference, New Orleans, LA, USA, April 7-10, 1993, Proceedings*. Ed. by Stephen D. Brookes, Michael G. Main, Austin Melton, Michael W. Mislove, and David A. Schmidt. Vol. 802. Lecture Notes in Computer Science. Springer, 1993, pp. 530–543 (cit. on p. 153).

[Red64] V. N. Redko. “On defining relations for the algebra of regular events”. In: *Ukrainskii Matematicheskii Zhurnal* 16 (1 1964), pp. 120–126 (cit. on p. 11).

[RJL21] Jurriaan Rot, Bart Jacobs, and Paul Blain Levy. “Steps and traces”. In: *J. Log. Comput.* 31.6 (2021), pp. 1482–1525 (cit. on p. 153).

[Ros00] Brian J. Ross. “Probabilistic Pattern Matching and the Evolution of Stochastic Regular Expressions”. In: *Appl. Intell.* 13.3 (2000), pp. 285–300 (cit. on pp. 15, 75, 153).

[Róż+23] Wojciech Różowski, Tobias Kappé, Dexter Kozen, Todd Schmid, and Alexandra Silva. “Probabilistic Guarded KAT Modulo Bisimilarity: Completeness and Complexity”. In: *50th International Colloquium on Automata, Languages, and Programming, ICALP 2023, July 10-14, 2023, Paderborn, Germany*. Ed. by Kousha Etessami, Uriel Feige, and Gabriele Puppis. Vol. 261. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023, 136:1–136:20 (cit. on pp. 54, 153).

[Róż+25] Wojciech Różowski, Robin Piedeleu, Alexandra Silva, and Fabio Zanasi. “A Diagrammatic Axiomatisation of Behavioural Distance of Nondeterministic Processes”. Under review. 2025 (cit. on p. 18).

[Róż24] Wojciech Różowski. “A Complete Quantitative Axiomatisation of Behavioural Distance of Regular Expressions”. In: *51st International Colloquium on Automata, Languages, and Programming (ICALP 2024)*. Ed. by Karl Bringmann, Martin Grohe, Gabriele Puppis, and Ola Svensson. Vol. 297. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024, 149:1–149:20. ISBN: 978-3-95977-322-5 (cit. on pp. 17, 64).

[RS24] Wojciech Różowski and Alexandra Silva. “A Completeness Theorem for Probabilistic Regular Expressions”. In: *Proceedings of the 39th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2024, Tallinn, Estonia, July 8-11, 2024*. Ed. by Pawel Sobocinski, Ugo Dal Lago, and Javier Esparza. ACM, 2024, 66:1–66:14 (cit. on p. 19).

[RS59] M. O. Rabin and D. Scott. “Finite Automata and Their Decision Problems”. In: *IBM Journal of Research and Development* 3.2 (1959), pp. 114–125 (cit. on p. 11).

[Rud90] Walter Rudin. *Functional Analysis*. en. 2nd ed. International Series in Pure & Applied Mathematics. Maidenhead, England: McGraw Hill Higher Education, Oct. 1990 (cit. on p. 31).

[Rut00] J.J.M.M. Rutten. "Universal coalgebra: a theory of systems". In: *Theoretical Computer Science* 249.1 (2000). Modern Algebra, pp. 3–80. ISSN: 0304-3975 (cit. on pp. 15, 22, 24, 25, 50, 75, 104).

[Sal66] Arto Salomaa. "Two Complete Axiom Systems for the Algebra of Regular Events". In: *J. ACM* 13.1 (Jan. 1966), pp. 158–169. ISSN: 0004-5411 (cit. on pp. 11, 23, 34, 39–41, 50, 54, 74, 76, 102, 138).

[San11] Davide Sangiorgi. "Coinduction and the duality with induction". In: *Introduction to Bisimulation and Coinduction*. Cambridge University Press, 2011, pp. 28–88 (cit. on p. 47).

[Sch+21] Todd Schmid, Tobias Kappé, Dexter Kozen, and Alexandra Silva. "Guarded Kleene Algebra with Tests: Coequations, Coinduction, and Completeness". In: *48th International Colloquium on Automata, Languages, and Programming, ICALP 2021, July 12-16, 2021, Glasgow, Scotland (Virtual Conference)*. Ed. by Nikhil Bansal, Emanuela Merelli, and James Worrell. Vol. 198. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021, 142:1–142:14 (cit. on p. 99).

[Sch+22] Todd Schmid, Wojciech Różowski, Alexandra Silva, and Jurriaan Rot. "Processes Parametrised by an Algebraic Theory". In: *49th International Colloquium on Automata, Languages, and Programming, ICALP 2022, July 4-8, 2022, Paris, France*. Ed. by Mikolaj Bojanczyk, Emanuela Merelli, and David P. Woodruff. Vol. 229. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022, 132:1–132:20 (cit. on p. 54).

[Sch61] M.P. Schützenberger. "On the definition of a family of automata". In: *Information and Control* 4.2 (1961), pp. 245–270. ISSN: 0019-9958 (cit. on p. 11).

[Sel10] P. Selinger. “A Survey of Graphical Languages for Monoidal Categories”. In: *New Structures for Physics*. Springer Berlin Heidelberg, 2010, pp. 289–355. ISBN: 9783642128219 (cit. on pp. 17, 69).

[Sew95] Peter Michael Sewell. “The Algebra of Finite State Processes”. PhD thesis. University of Edinburgh, 1995 (cit. on pp. 58, 59).

[Sil+10] Alexandra Silva, Filippo Bonchi, Marcello M. Bonsangue, and Jan J. M. M. Rutten. “Generalizing the powerset construction, coalgebraically”. In: *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2010, December 15-18, 2010, Chennai, India*. Ed. by Kamal Lodaya and Meena Mahajan. Vol. 8. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2010, pp. 272–283 (cit. on pp. 16, 83, 104, 153).

[Sil10] A.M Silva. “Kleene coalgebra”. PhD thesis. Radboud Universiteit Nijmegen, 2010 (cit. on pp. 28, 50, 54, 129, 153).

[Smo+20] Steffen Smolka, Nate Foster, Justin Hsu, Tobias Kappé, Dexter Kozen, and Alexandra Silva. “Guarded Kleene algebra with tests: verification of uninterpreted programs in nearly linear time”. In: *Proc. ACM Program. Lang.* 4.POPL (2020), 61:1–61:28 (cit. on p. 54).

[Sok05] Ana Sokolova. “Coalgebraic analysis of probabilistic systems”. PhD thesis. Technische Universiteit Eindhoven, 2005 (cit. on pp. 153, 157, 158).

[SRS21] Todd Schmid, Jurriaan Rot, and Alexandra Silva. “On Star Expressions and Coalgebraic Completeness Theorems”. In: *Proceedings 37th Conference on Mathematical Foundations of Programming Semantics, MFPS 2021, Hybrid: Salzburg, Austria and Online, 30th August - 2nd September, 2021*. Ed. by Ana Sokolova. Vol. 351. EPTCS. 2021, pp. 242–259 (cit. on pp. 57, 153).

[SS00] Eugene W. Stark and Scott A. Smolka. “A complete axiom system for finite-state probabilistic processes”. In: *Proof, Language, and Interaction, Essays in Honour of Robin Milner*. Ed. by Gordon D. Plotkin, Colin Stirling, and Mads Tofte. The MIT Press, 2000, pp. 571–596 (cit. on pp. 14, 53, 152, 154).

[SS11] Alexandra Silva and Ana Sokolova. “Sound and Complete Axiomatization of Trace Semantics for Probabilistic Systems”. In: *Twenty-seventh Conference on the Mathematical Foundations of Programming Semantics, MFPS 2011, Pittsburgh, PA, USA, May 25-28, 2011*. Ed. by Michael W. Mislove and Joël Ouaknine. Vol. 276. Electronic Notes in Theoretical Computer Science. Elsevier, 2011, pp. 291–311 (cit. on pp. 14, 16, 92–94, 153).

[Sto49] Michael H. Stone. “Postulates for the barycentric calculus”. In: *Annali di Matematica Pura ed Applicata* 29 (1949), pp. 25–30 (cit. on p. 80).

[SW15] Ana Sokolova and Harald Woracek. “Congruences of convex algebras”. In: *Journal of Pure and Applied Algebra* 219.8 (2015), pp. 3110–3148. ISSN: 0022-4049 (cit. on pp. 76, 87, 90, 129, 148).

[SW18] Ana Sokolova and Harald Woracek. “Proper Semirings and Proper Convex Functors”. In: *Foundations of Software Science and Computation Structures - 21st International Conference, FOSSACS 2018, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2018, Thessaloniki, Greece, April 14-20, 2018, Proceedings*. Ed. by Christel Baier and Ugo Dal Lago. Vol. 10803. Lecture Notes in Computer Science. Springer, 2018, pp. 331–347 (cit. on pp. 16, 18, 76, 102, 132, 153).

[van12] Franck van Breugel. “On behavioural pseudometrics and closure ordinals”. In: *Information Processing Letters* 112.19 (2012), pp. 715–718. ISSN: 0020-0190 (cit. on p. 62).

[Vil09] Cédric Villani. *Optimal Transport*. Springer Berlin Heidelberg, 2009. ISBN: 9783540710509 (cit. on p. 13).

[VR99] Erik P. de Vink and Jan J. M. M. Rutten. “Bisimulation for Probabilistic Transition Systems: A Coalgebraic Approach”. In: *Theor. Comput. Sci.* 221.1-2 (1999), pp. 271–293 (cit. on pp. 15, 153).

[Wag+19] Jana Wagemaker, Marcello M. Bonsangue, Tobias Kappé, Jurriaan Rot, and Alexandra Silva. “Completeness and Incompleteness of Synchronous Kleene Algebra”. In: *MPC*. 2019, pp. 385–413 (cit. on pp. 11, 40).

[Wiß22] Thorsten Wißmann. “Minimality Notions via Factorization Systems and Examples”. In: *Log. Methods Comput. Sci.* 18.3 (2022) (cit. on p. 150).

[Zha+25] Cheng Zhang, Tobias Kappé, David E. Narváez, and Nico Naus. “CF-GKAT: Efficient Validation of Control-Flow Transformations”. In: *Proc. ACM Program. Lang.* 9.POPL (2025), pp. 600–626 (cit. on p. 4).