Minimal and Canonical Quotients for Simulation Equivalences

Eduardo Costa Martins
Eindhoven University of Technology, The Netherlands

Tim A. C. Willemse
Eindhoven University of Technology, The Netherlands

Abstract

Quotients have only been studied for a handful of equivalences in the linear time-branching time spectrum, for which there are results pertaining to canonicity and minimality. We extend these results to weak simulation equivalence and coupled similarity, two closely related equivalences induced by simulation preorders. We describe abstract procedures for transforming an LTS into a unique representative of its equivalence class, and for transforming an LTS into an equivalent state- and transition-minimal LTS. Moreover, we show the minimisation problem is NP-complete.

2012 ACM Subject Classification Theory of computation → Concurrency; Theory of computation → Problems, reductions and completeness

Keywords and phrases Coupled Similarity, Weak Simulation, Minimisation

Digital Object Identifier 10.4230/LIPIcs.CONCUR.2026.26

Related Version *Technical Report*: `https://arxiv.org/abs/2606.21450` [17]

Funding The Cynergy4MIE project is supported by the Chips Joint Undertaking and its members, including the top-up funding by National Authorities under Grant Agreement No 101140226.

1 Introduction

A Labelled Transition System is a mathematical model for capturing how a system performs actions to evolve from one configuration to another. LTSs are typically used as a target model for higher-level languages such as Process Algebras or Petri nets, providing these languages with a formal semantic model. Concepts such as behavioural equivalence are often studied at the level of LTSs. For instance, in [23] and [24], van Glabbeek introduced the linear time-branching time spectrum of behavioural equivalences for systems with and without unobservable actions (τ-steps), respectively.

Given an equivalence relation on LTSs, one common transformation is to reduce the state space whilst preserving semantic equivalence. A common reduction technique is quotienting, which involves the merging of states into equivalence classes, and adding transitions between equivalence classes according to a certain schema. Quotienting is interesting for a variety of reasons. It is often used in compositional and modular reasoning, thereby helping to combat the infamous state space explosion problem in practice; this is used in the CADP toolset [9], and, to a lesser extent, also in the FDR toolset [10] and in the mCRL2 toolset [16, 5]. Moreover, quotienting also offers a more abstract view on a system's behaviour by removing distinctions in the LTS that are not semantically important.

In [14], quotienting functions were found for all behavioural equivalences studied in [23]. Of the behavioural equivalences studied in [24], quotienting functions have been found for (divergence preserving) weak bisimilarity [4], (divergence preserving) branching bisimilarity [8], and trace equivalence, where the latter is a well-known result from automata theory.

It seems natural to assume that quotienting works for all the behavioural equivalences, but this is not known, nor is there – to the best of our knowledge – any argument for an underlying reason as to why it should. For instance, even though quotienting functions exist for the most common behavioural equivalences, the quotients themselves differ in a number of ways. For bisimilarity, the quotient is *canonical* (i.e. unique up to isomorphism), and *minimal* in the number of states and transitions. For branching bisimilarity, the same holds if τ-loops are removed afterwards [8]. For weak bisimilarity, so-called duplicate transitions need to be removed as well [8]. For these equivalences:

- the minimal LTS is unique;
- it can be computed in polynomial time [11, 12]; and
- the quotient is state minimal if unreachable states are removed.

Yet, for trace equivalence, this is not the case. Whilst there is a unique and minimal *deterministic* LTS for a given equivalence class, there may be multiple distinct and minimal LTSs if non-determinism is permitted. Additionally, the quotient LTS may not be state minimal, and the computation of a minimal LTS is PSPACE-complete. Thus, there are few recurring patterns amongst quotients for different equivalences.

In this paper, we study the canonicity and minimality for quotienting for coupled simulation equivalence [22] and weak simulation equivalence [18, 24]. First, we generalise the notion of duplicate transitions from the setting of weak bisimilarity. Then, we introduce transformations based on the τ-laws for weak simulation equivalence and for coupled similarity, namely $\tau.(x + y) = x + y$ and $\tau.(\tau.x + y) = \tau.x + y$. Applying the τ-laws from left to right is deterministic, and provides the ingredients for obtaining a canonical quotient. Applying them from right to left is non-deterministic, and provides the means for a minimal quotient.

When applying the τ-laws from right to left, there is an optimal choice that provides the largest reduction in size. However, finding a minimal quotient can be as hard as finding such an optimal choice, which itself is as hard as finding a minimal set cover, which is NP-complete, and minimisation turns out to be NP-complete, too. The procedure we give for computing a minimal quotient relies on some solver for the set cover problem. This way, we can leverage state of the art algorithms for the set cover problem in order to minimise state spaces. Moreover, the instances of the set cover problem that need to be solved are typically small.

Related Work. In [14] it is shown that a quotienting function exists for every equivalence in [23]. It is shown in [15] that this is because quotienting preserves and reflects a class of modal formulas that characterise most of the equivalences in [23]. A canonical and minimal quotient for simulation equivalence (without internal τ-steps) in the context of Kripke structures was found in [6]. The results from [6] extend to LTSs using translations found in [20]. As for the equivalences in [24], in [4] it is shown how LTSs can be quotiented w.r.t. weak bisimilarity by solving the relational coarsest partition problem. It was shown in [7] that by removing so-called *redundant* transitions from a quotiented LTS, the resulting LTS is unique and minimal w.r.t. weak bisimilarity. The results from [7] were generalised to branching bisimilarity and their divergence-preserving variants in [8].

Outline. Section 2 contains preliminaries. In Section 3 we show that the $\forall$-quotient is sound for both equivalences, though not necessarily state- or transition-minimal, nor canonical. In Section 4, we describe a procedure for a canonical quotient, which is not necessarily minimal. In Section 5 we describe a procedure for a minimal quotient. This procedure can be modified so that the output is also canonical. Lastly, in Section 6 we conclude our findings. Proofs can be found in the accompanying technical report [17].

2 Preliminaries

We recall Labelled Transition Systems, discuss the two notions of equivalence considered in this paper, and introduce the notational conventions we use throughout the paper.

▶ **Definition 2.1** (Labelled Transition System). *A* labelled transition system *(LTS) is a 4-tuple $L = \langle S, \mathcal{A}_\tau, \rightarrow, \iota\rangle$ with:*
- *S a set of states;*
- *$\mathcal{A}_\tau$ a set of actions of which τ, the unobservable action, is an element;*
- *$\rightarrow \subseteq S \times \mathcal{A}_\tau \times S$ a transition relation; and*
- *$\iota \in S$ is the initial state.*

Henceforth, we only consider transition systems with a finite number of states and transitions. We fix the set of actions $\mathcal{A}_\tau$, and write $\mathcal{A}$ to denote $\mathcal{A}_\tau \setminus \{\tau\}$. Elements of $\mathcal{A}$ are called *visible*, whereas τ is *invisible*. We use the following notation:
- $p \xrightarrow{\alpha} q$ denotes $(p, \alpha, q) \in \rightarrow$,
- $p \xrightarrow{X} q$, for $X \subseteq \mathcal{A}_\tau$, denotes $p \xrightarrow{\alpha} q$ for each $\alpha \in X$,
- $p \xrightarrow{\alpha}$ means there exists some $q \in S$ such that $p \xrightarrow{\alpha} q$,
- $p \stackrel{\alpha}{\nrightarrow} q$ denotes $(p, \alpha, q) \notin \rightarrow$.

Moreover, for a transition $p \xrightarrow{\alpha} q$, we call p and q the *source* and *target* states of the transition, respectively.

For an LTS, $L = \langle S, \mathcal{A}_\tau, \rightarrow, \iota\rangle$, we may remove or add transitions as follows. Let $Y \subseteq S \times \mathcal{A}_\tau \times S$. Then the LTSs $L \cup Y$ and $L \setminus Y$ are defined as follows:
- $L \cup Y \coloneqq \langle S, \mathcal{A}_\tau, \rightarrow \cup Y, \iota\rangle$; and similarly
- $L \setminus Y \coloneqq \langle S, \mathcal{A}_\tau, \rightarrow \setminus Y, \iota\rangle$.

We frequently consider LTSs with some states in common. Say there are LTSs $L, \overline{L}, L_i$ that have some states, p, q, in common. To distinguish transitions in each LTS, we use the following notational conventions:
- $p \xrightarrow{\alpha} q$ as usual means "(p, α, q) is a transition of L",
- $\bar{p} \xrightarrow{\alpha} \bar{q}$ is shorthand for "(p, α, q) is a transition of $\overline{L}$", and
- $p_i \xrightarrow{\alpha} q_i$ is shorthand for "(p, α, q) is a transition of L_i".

Here, we assume that the names $\bar{p}, p_i, \bar{q}, q_i$ are not part of the LTSs.

The main behavioural equivalences we discuss in this paper are weak simulation equivalence and coupled similarity. Whilst not the main focus, weak bisimilarity is relevant as well. These equivalences are defined using *weak* transitions:
- $p \stackrel{\tau}{\Longrightarrow} q$ iff q is reachable from p using zero or more τ-transitions, and
- $p \stackrel{a}{\Longrightarrow} q$ iff $\exists p', q' : p \stackrel{\tau}{\Longrightarrow} p' \xrightarrow{a} q' \stackrel{\tau}{\Longrightarrow} q$.

On the other hand, transitions of the form $p \xrightarrow{\alpha} q$ with $\alpha \in \mathcal{A}_\tau$ are called *strong* transitions.

▶ **Definition 2.2** (Weak Simulation). *Let $L = \langle S, \mathcal{A}_\tau, \rightarrow, \iota\rangle$ be an LTS. A binary relation $\mathcal{R}$ on S is a weak* simulation *on L iff for all $(p, q) \in \mathcal{R}, \alpha \in \mathcal{A}_\tau, p' \in S$, the* simulation condition *holds:*

$$p \xrightarrow{\alpha} p' \implies \exists q' \in S : q \stackrel{\alpha}{\Longrightarrow} q' \wedge p'\mathcal{R}q' \quad .$$

We write $p \sqsubseteq_S q$ iff there exists a weak simulation $\mathcal{R}$ such that $p\mathcal{R}q$, and say that q simulates *p.*

Moreover, a symmetric weak simulation is called a weak bisimulation*. We write $p \leftrightarroweq_w q$ iff there exists a weak bisimulation $\mathcal{R}$ such that $p\mathcal{R}q$, and say that p and q are* weakly bisimilar*.*

▶ **Definition 2.3** (Coupled Simulation). *Let $L = \langle S, \mathcal{A}_\tau, \rightarrow, \iota\rangle$ be an LTS. A relation $\mathcal{R}$ on S is a* coupled simulation *on L iff it is a weak simulation, and moreover for all $(p, q) \in \mathcal{R}$, the* coupling condition *holds:*

$$\exists q' \in S : q \stackrel{\tau}{\Longrightarrow} q' \wedge q'\mathcal{R}p \quad .$$

We write $p \sqsubseteq_{CS} q$ iff there exists a coupled simulation $\mathcal{R}$ such that $p\mathcal{R}q$, and say that q coupled simulates *p.*

Throughout this paper, we use the fact that $\sqsubseteq_S$ is the greatest weak simulation, and $\sqsubseteq_{CS}$ is the greatest coupled simulation [2].

In this paper, we provide a number of results that apply to both preorders and their induced equivalences. To avoid repetition, we fix $x \in \{S, CS\}$ so that $\sqsubseteq_x$ describes the weak simulation preorder, or the coupled similarity preorder. An x-simulation is a weak simulation if $x = S$, and a coupled simulation if $x = CS$. For states p, q, and LTSs L and L' with initial states ι and ι', respectively, we write:

- $p \equiv_x q$ iff $p \sqsubseteq_x q$ and $q \sqsubseteq_x p$,
- $p \sqsubset_x q$ iff $p \sqsubseteq_x q$, but not $q \sqsubseteq_x p$,
- $p \sqsupseteq_x q$ and $p \sqsupset_x q$ iff $q \sqsubseteq_x p$ and $q \sqsubset_x p$, respectively,
- $L \equiv_x L'$ iff $\iota \equiv_x \iota'$ in the disjoint union of L and L',
- $[p]_x$ for the $\equiv_x$ equivalence class of p, and
- $p \sqsupset_x$ to denote the set $\{q \,|\, p \sqsupset_x q\}$.

When we say equivalent, we mean equivalent with respect to $\equiv_x$. The examples in this paper work for both $\equiv_S$ (weak simulation equivalence) and $\equiv_{CS}$ (coupled similarity) unless stated otherwise. An *inert* transition is one of the form $p \xrightarrow{\tau} q$ such that p and q are equivalent.

Quotienting describes the merging of equivalent states in an LTS in such a way that the resulting LTS is again equivalent to the starting LTS.

▶ **Definition 2.4** (Quotient). *An LTS, $L = \langle S, \mathcal{A}_\tau, \rightarrow, \iota\rangle$, is a* quotient *LTS iff for all states $p, q \in S$, $p \equiv_x q$ implies $p = q$. A function $\mathcal{Q}$ mapping one LTS to another is a* quotienting function *iff for all LTSs L, $\mathcal{Q}(L) \equiv_x L$ and $\mathcal{Q}(L)$ is a quotient LTS. We say that $\mathcal{Q}(L)$ is a quotient of L.*

We use isomorphism to describe uniqueness of a quotient.

▶ **Definition 2.5** (Isomorphism). *Let $L = \langle S, \mathcal{A}_\tau, \rightarrow, \iota\rangle$ and $L' = \langle S', \mathcal{A}_\tau, \rightarrow', \iota'\rangle$ be two LTSs. We say L and L' are* isomorphic*, written $L \sim L'$, iff there is a bijection $f : S \rightarrow S'$ such that:*

- *$f(\iota) = \iota'$; and*
- *$(p, \alpha, q) \in \rightarrow$ iff $(f(p), \alpha, f(q)) \in \rightarrow'$.*

Let $\mathcal{Q}$ be some quotienting function and L some LTS. We say $\mathcal{Q}(L)$ is canonical *or* unique *iff $\mathcal{Q}(L) \sim \mathcal{Q}(L')$ for all $L' \equiv_x L$.*

As is typical when discussing quotients [6, 8], we define the size of an LTS using a lexicographic ordering on states and transitions, thereby prioritising a reduction in the number of states over a reduction in the number of transitions.

▶ **Definition 2.6** (Size). *Let $L = \langle S, \mathcal{A}_\tau, \rightarrow, \iota\rangle$ and $L' = \langle S', \mathcal{A}_\tau, \rightarrow', \iota'\rangle$ be two LTSs. We write $|L| \leq |L'|$ iff:*

- *$|S| < |S'|$; or*
- *$|S| = |S'|$ and $|\rightarrow| \leq |\rightarrow'|$.*

We say L is minimal *(for $\equiv_x$) iff $|L| \leq |L'|$ for all $L' \equiv_x L$.*

3 Universal Quotient and Redundant Transitions

In this section, we explore existing concepts for quotients, and extend them to the context of $\equiv_x$. We do so with the (false) hopes that these concepts are sufficient to define canonical and minimal quotients for $\equiv_x$. We start with the universal quotient, or $\forall$-quotient, which is a common transformation [6] for various behavioural equivalences involving the merging of equivalent states, see Figure 1. It gets its name from the $\forall$-quantifier that appears in the definition of the transition relation.

▶ **Definition 3.1** ($\forall$-Quotient). *Let $L = \langle S, \mathcal{A}_\tau, \rightarrow, \iota\rangle$ be an LTS. The* $\forall$-quotient *of L is an LTS, $L_{/\equiv} = \langle S_{/\equiv}, \mathcal{A}_\tau, \rightarrow_{/\equiv}, [\iota]_x\rangle$, where:*

- $S_{/\equiv} = \{[p]_x \mid p \in S\}$; *and*
- $\rightarrow_{/\equiv} = \{([p]_x, \alpha, [q]_x) \mid \forall p' \in [p]_x, \exists q' \in [q]_x : p' \stackrel{\alpha}{\Longrightarrow} q' \wedge (\alpha = \tau \implies p' \not\equiv_x q')\}$.

▶ Remark 3.2. The transition relation for the $\forall$-quotient is defined such that $L_{/\equiv}$ has no inert transitions. Successive transformations we define require the absence of such inert transitions, and also do not introduce inert transitions.

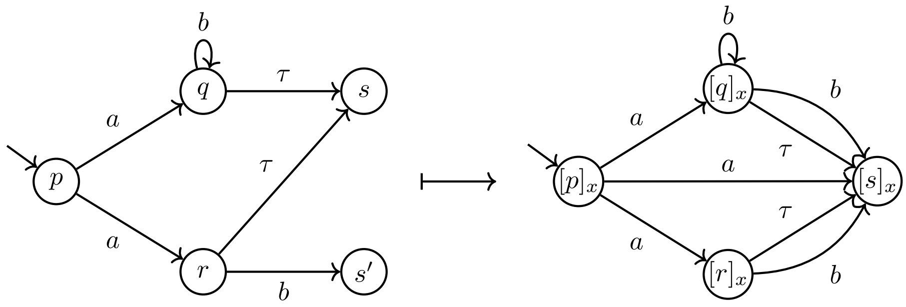

Figure 1 The $\forall$-quotient maps the left LTS to the right LTS, where $s' \in [s]_x$.

That the $\forall$-quotient *is* indeed a quotienting function is stated in Proposition 3.7, the proof of which relies on the following lemmas. These require the following definitions, where $\alpha \in \mathcal{A}_\tau$ and X is a set of states, in the context of an LTS $L = \langle S, \mathcal{A}_\tau, \rightarrow, \iota\rangle$.

$$G_\alpha(p) = \{p' \mid p \stackrel{\alpha}{\Longrightarrow} p'\}$$
$$\mathsf{max}(X) = \{p \in X \mid \forall q \in X : p \sqsubseteq_x q \implies q \sqsubseteq_x p\}$$
$$\mathsf{min}(X) = \{p \in X \mid \forall q \in X : q \sqsubseteq_x p \implies p \sqsubseteq_x q\}$$

We remark that whenever X is non-empty and finite, then $\mathsf{max}(X)$ and $\mathsf{min}(X)$ are non-empty.

▶ **Lemma 3.3.** *If $p \stackrel{\tau}{\Longrightarrow} q$, then $q \sqsubseteq_x p$.*

▶ **Lemma 3.4.** *If $p \sqsubseteq_x q$ and $p \stackrel{\alpha}{\Longrightarrow} p'$, then there exists q' such that $q \stackrel{\alpha}{\Longrightarrow} q'$ and $p' \sqsubseteq_x q'$.*

▶ **Lemma 3.5.** *Let $p \in S, \alpha \in \mathcal{A}_\tau$. Then, for all $q \equiv_x p$ and all $p' \in \mathsf{max}(G_\alpha(p))$, there is some $q' \in S$ such that $q \stackrel{\alpha}{\Longrightarrow} q'$ and $p' \equiv_x q'$.*

▶ **Lemma 3.6.** *Let $p \in S$ and suppose $x = CS$. Then, for all $q \in S$ such that $p \sqsubseteq q$, and all $p' \in \mathsf{min}(G_\tau(p))$, there is some $q' \in S$ such that $q \overset{\tau}{\Longrightarrow} q'$ and $p' \equiv_x q'$.*

▶ **Proposition 3.7** (∀-Quotient)**.** *The function mapping an LTS L to its ∀-quotient, $L_{/\equiv}$, is a quotienting function. Moreover, any state p of L is equivalent to the state $[p]_x$ of $L_{/\equiv}$.*

If an LTS has at least two equivalent states, then the ∀-quotient guarantees a reduction in the size of the LTS. However, if no two states are equivalent, then the ∀-quotient essentially computes the transitive closure of the given LTS. That is, it adds a $p \xrightarrow{\alpha} q$ transition whenever $p \overset{\alpha}{\Longrightarrow} q$, potentially creating a *duplicate transition* and increasing the size of the LTS.

▶ **Definition 3.8** (Duplicate Transitions [8])**.** *Let $L = \langle S, \mathcal{A}_\tau, \rightarrow, \iota\rangle$ be an LTS without inert transitions. We say a transition $p \xrightarrow{\alpha} q$ of L* duplicates *$p' \xrightarrow{\alpha} q'$ iff:*
- $p \overset{\tau}{\Longrightarrow} p'$,
- $q' \overset{\tau}{\Longrightarrow} q$, *and*
- $(p, q) \neq (p', q')$.

We say $p \xrightarrow{\alpha} q$ is a duplicate *if there is some $p' \xrightarrow{\alpha} q'$ that $p \xrightarrow{\alpha} q$ duplicates.*

Conversely, if some $p \xrightarrow{\alpha} q$ transition is a duplicate, then p can reach q with a weak α-transition without using the duplicate $p \xrightarrow{\alpha} q$ transition, which can be removed. Consider the LTS, L, on the right in Figure 1, and the LTS, L', on the left in Figure 2. The two LTSs are equivalent, and computing the ∀-quotient of L' results in (an LTS isomorphic to) the larger LTS L. Conversely, removing all duplicate transitions of L results in (an LTS isomorphic to) the smaller LTS L'.

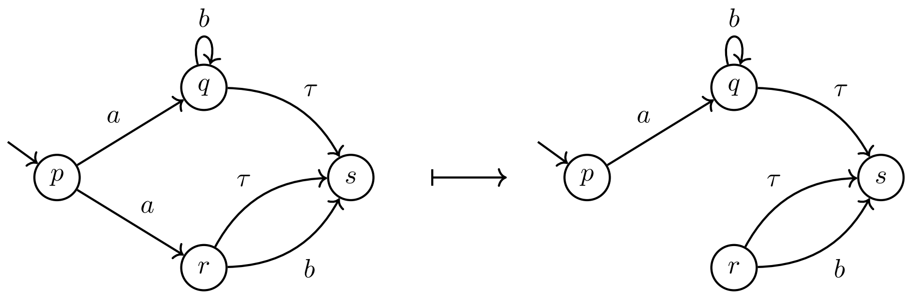

Figure 2 Two equivalent LTSs illustrating the need for a stronger notion of duplicate transitions.

Removing all duplicate transitions after quotienting results in an LTS no larger than the input LTS – we prove a slightly different result in Proposition 3.13. For weak bisimilarity, this procedure describes a canonical, minimal LTS [7, 8]. However, this is not the case for coupled or weak simulation equivalence, as there may be removable transitions that are not included in the notion of duplicate transitions. Consider the LTSs in Figure 2. The $p \xrightarrow{a} r$ transition may be removed since $p \xrightarrow{a} q$ and $r \sqsubset_x q$. Consequently, r may be removed since it is unreachable. The $p \xrightarrow{a} r$ transition is what we call a *covered transition*. Removing covered transitions is similar to removing duplicate transitions, except that we do not require a weak transition to the same state as the covered transition, but simply to some greater state, see Figure 3. The concept of a covered transition resembles that of a "little brother" in the context of simulation equivalence for Kripke structures [13, 6].

▶ **Definition 3.9** (Covered Transitions). *Let $L = \langle S, \mathcal{A}_\tau, \to, \iota \rangle$ be an LTS without inert transitions. We say a transition $p \xrightarrow{\alpha} q$ of L is* covered by *$p' \xrightarrow{\alpha} q'$ iff:*
- $p \overset{\tau}{\Longrightarrow} p'$,
- $q \sqsubseteq q'$, *and*
- $(p, q) \neq (p', q')$.

We say $p \xrightarrow{\alpha} q$ is covered *if there is some $p' \xrightarrow{\alpha} q'$ that covers $p \xrightarrow{\alpha} q$.*

Figure 3 Comparing duplicate and covered transitions. The $p \xrightarrow{\alpha} q$ transition on the left (right) is duplicated (covered) by the $p' \xrightarrow{\alpha} q'$ transition.

▶ **Lemma 3.10.** *Let $L = \langle S, \mathcal{A}_\tau, \to, \iota \rangle$ be an LTS without inert transitions, and $p \xrightarrow{\alpha} q$ some transition of L covered by $p' \xrightarrow{\alpha} q'$. Let $s, t \in S$ be arbitrary, then:*

$$s \overset{\tau}{\Longrightarrow} p' \xrightarrow{\alpha} q' \overset{\tau}{\Longrightarrow} t \text{ in } L \iff s \overset{\tau}{\Longrightarrow} p' \xrightarrow{\alpha} q' \overset{\tau}{\Longrightarrow} t \text{ in } L \setminus \{(p, \alpha, q)\} \quad .$$

▶ **Corollary 3.11** (Removing Covered Transitions). *Let $L = \langle S, \mathcal{A}_\tau, \to, \iota \rangle$ be an LTS without inert transitions, and let $p \xrightarrow{\alpha} q$ be some covered transition of L. Let $\overline{L} \coloneqq L \setminus \{(p, \alpha, q)\}$. Then $L \equiv_x \overline{L}$. Moreover, for all $s \in S$, it holds that s in L is equivalent to s in $\overline{L}$.*

The following lemma states that a covered transition remains covered after the removal of some other covered transition. Therefore, one may identify all the covered transitions of some LTS, L, remove them all at once, and the resulting LTS is equivalent. Moreover, since removing a covered transition does not introduce new covered transitions, the resulting LTS has no covered transitions.

▶ **Lemma 3.12.** *Let $L = \langle S, \mathcal{A}_\tau, \to, \iota \rangle$ be an LTS without inert transitions. Let $p \xrightarrow{\alpha} q$ and $s \xrightarrow{\beta} t$ be two distinct covered transitions of L. Let $\overline{L} \coloneqq L \setminus \{(p, \alpha, q)\}$, then $s \xrightarrow{\beta} t$ is a covered transition of $\overline{L}$.*

By computing the $\forall$-quotient of an LTS, and then removing all covered transitions, we obtain a reduction.

▶ **Proposition 3.13.** *Let $L = \langle S, \mathcal{A}_\tau, \to, \iota \rangle$ be an LTS. Let $\overline{L} = \langle \overline{S}, \mathcal{A}_\tau, \overline{\to}, \overline{\iota} \rangle$ be the LTS obtained from L by computing its $\forall$-quotient and subsequently removing all covered transitions. Then $|\overline{L}| \leq |L|$.*

Surprisingly, even with a generalised notion of duplicate transitions and the subsequent removal of unreachable states, we do not have a canonical or minimal quotient. Consider the LTSs in Figure 4. They are both equivalent, yet we cannot reduce the number of states of the left LTS using the transformations listed so far. Therefore, this procedure does not lead to a minimal quotient. Similarly, the transformations listed so far do not introduce new states, so we do not have a canonical quotient since this would require reducing the number of states of the left LTS, or increasing the number of states of the right LTS.

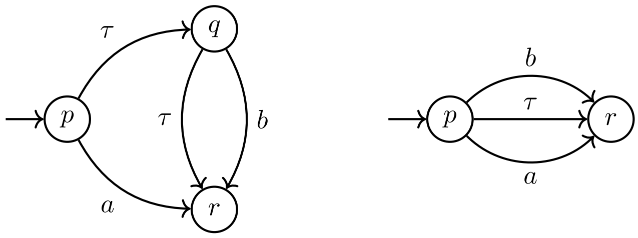

Figure 4 The two LTSs are equivalent, but with the transformations introduced so far we cannot reduce the size of the left LTS, or increase the number of states with the right LTS.

Moreover, the problem of finding a canonical quotient cannot necessarily be solved by finding a minimal quotient, for there are classes of LTSs that do not have a unique minimal LTS. For example, the two LTSs in Figure 5 are both minimal in size, so a canonical minimal quotient would need to be able to distinguish between minimal LTSs.

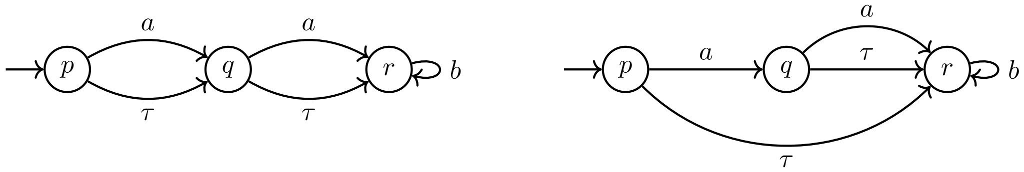

Figure 5 The two LTSs are equivalent and minimal. In particular, we cannot map between them using the transformations listed so far.

4 Desaturation and Canonicity

The previous sections showed that transformations resembling those found in the literature are insufficient to obtain a canonical quotient. It turns out that with the introduction of one more transformation, τ-desaturation, we *are* able to describe a quotient that is canonical. This transformation is based on the axioms $\tau.x = x$ and $\tau.(\tau.x + y) = \tau.x + y$ for weak simulation equivalence and coupled simulation, respectively [1, 19]. They describe the removal of a τ-transition followed by the saturation of the source state with the outgoing transitions of the target state, see Figure 6. For coupled similarity, this requires that the target state also has an outgoing τ-transition.

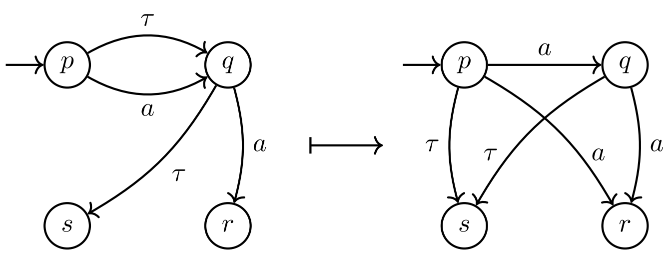

Figure 6 τ-Desaturation maps the left LTS to the right LTS by removing the $p \xrightarrow{\tau} q$ transition and adding $p \xrightarrow{\alpha} q'$ for each $q \xrightarrow{\alpha} q'$. For coupled similarity, we require that q has an outgoing τ-transition.

▶ **Proposition 4.1** (τ-Desaturation). *Let $L = \langle S, \mathcal{A}_\tau, \rightarrow, \iota\rangle$ be an LTS without inert transitions. Let $p, q \in S$ be such that $p \xrightarrow{\tau} q$. Assume $x = CS$ implies $q \xrightarrow{\tau} r$ for some $r \in S$. Define:*

$$\overline{L} \coloneqq (L \cup \{(p, \alpha, q') \mid q \xrightarrow{\alpha} q'\}) \setminus \{(p, \tau, q)\} \quad .$$

Then $\overline{L} \equiv_x L$. Moreover, for all $s \in S$, it holds that s in L is equivalent to s in $\overline{L}$.

Since τ-desaturation may introduce new τ-transitions, it is not immediately clear whether or not the repeated application of τ-desaturation terminates. The following proposition states that it does. As a consequence, we may obtain an LTS with no τ-transitions in the case of weak simulation equivalence, or no *consecutive* τ-transitions in the case of coupled simulation. We call such an LTS *desaturated*.

▶ **Proposition 4.2.** *Let $L = \langle S, \mathcal{A}_\tau, \rightarrow, \iota\rangle$ be an LTS without inert transitions. The repeated application of τ-desaturation terminates.*

Reconsider the LTSs in Figure 4. Applying τ-desaturation on the left LTS to remove the $p \xrightarrow{\tau} q$ transition results in the state q being unreachable. This unreachable state can subsequently be removed, and as such we obtain the LTS on the right of Figure 4. Likewise for the LTSs in Figure 5, we can apply τ-desaturation on the left LTS to remove the $p \xrightarrow{\tau} q$ transition. One of the added transitions is a $p \xrightarrow{a} r$ transition, but this transition is covered (since $p \xrightarrow{\tau} q \xrightarrow{a} r$) and thus it can be removed, resulting in the LTS on the right of Figure 5.

We now have the necessary transformations to obtain a canonical quotient for $\equiv_x$. We say an LTS is *reduced* if:

- it is desaturated;
- it has no covered transitions;
- it has no unreachable states;
- it has no inert transitions; and
- no two states are equivalent.

It turns out that whenever two equivalent LTSs are reduced, they are also isomorphic, as claimed by Theorem 4.4. The proof of this theorem uses the following observation to show that x-similarity and weak bisimilarity coincide on reduced LTSs:

▶ **Lemma 4.3.** *Let $L = \langle S, \mathcal{A}_\tau, \rightarrow, \iota\rangle$ be a desaturated LTS with no covered transitions. Let $p \xrightarrow{\alpha} q$ be some transition of L and let $p' \equiv_x p$. Then there exists some state $q' \equiv_x q$ such that $p' \overset{\alpha}{\Longrightarrow} q'$.*

▶ **Theorem 4.4.** *Let $L = \langle S, \mathcal{A}_\tau, \rightarrow, \iota\rangle$ and $\overline{L} = \langle \overline{S}, \mathcal{A}_\tau, \overline{\rightarrow}, \bar{\iota}\rangle$ be two reduced LTSs. Then $L \equiv_x \overline{L}$ iff $L \sim \overline{L}$.*

Proof. It is clear $L \sim \overline{L}$ implies $L \equiv_x \overline{L}$. To show that $L \equiv_x \overline{L}$ implies $L \sim \overline{L}$, we use Lemma 21 from [8] which states that if:

1. $L \leftrightarroweq_w \overline{L}$, and
2. L and $\overline{L}$ are state minimal w.r.t. $\leftrightarroweq_w$, and
3. L and $\overline{L}$ have no duplicate transitions,

then $L \sim \overline{L}$. For (1), it immediately follows by Lemma 4.3 that

$$\mathcal{R} \coloneqq \{(p, \bar{q}) \in S \times \overline{S} \mid p \equiv_x \bar{q}\} \cup \{(\bar{p}, q) \in \overline{S} \times S \mid \bar{p} \equiv_x q\}$$

is a weak bisimulation. For (2), state minimality for weak bisimilarity requires that no two states are weakly bisimilar, and all states are reachable [8]. Since $\leftrightarroweq_w \subseteq \equiv_x$ [24], and no two states are equivalent w.r.t. $\equiv_x$, it holds that L and $\overline{L}$ are state minimal w.r.t. $\leftrightarroweq_w$. For (3), note that L and $\overline{L}$ have no covered transitions, so it suffices to show that all duplicate transitions are covered. Consider some $p \xrightarrow{\alpha} q$ that duplicates $p' \xrightarrow{\alpha} q'$. Then by Lemma 3.3, it holds that $q \sqsubseteq_x q'$, and therefore $p \xrightarrow{\alpha} q$ is covered by $p' \xrightarrow{\alpha} q'$. Hence, $p \xrightarrow{\alpha} q$ is covered. ◀

Algorithm 1 computes a reduced LTS equivalent to the input LTS. The order of operations is important: it is essential to first compute the $\forall$-quotient in order to eliminate inert transitions and ensure the correctness of the remaining transformations; we need to desaturate before removing covered transitions, since covered transitions may be introduced by desaturation; and we need to remove covered transitions before removing unreachable states, since states may become unreachable by removing covered transitions.

Algorithm 1 A quotienting function that yields a canonical quotient.

---

procedure CanQuotient(L)
  $L_1 \coloneqq L_{/\equiv}$
  $L_2 \coloneqq$ Desaturate L_1
  $L_3 \coloneqq$ Remove covered transitions from L_2
  $L_4 \coloneqq$ Remove unreachable states from L_3
  return L_4

This algorithm can be implemented to run in polynomial time since $\sqsubseteq_x$ can be computed in polynomial time [3], desaturation needs to look at no more than two consecutive transitions, and likewise the removal of covered transitions needs to look at no more than two consecutive transitions due to the preceding desaturation step.

5 Saturation and Minimality

The procedure described in the previous section does not necessarily yield a *minimal* quotient. This is because τ-desaturation may cause a strict increase in the number of transitions, as shown in Figure 7. Yet, τ-desaturation is needed because it reduces the number of incoming transitions to a state, and if this number is reduced to zero, the size of the LTS can be reduced by removing the state.

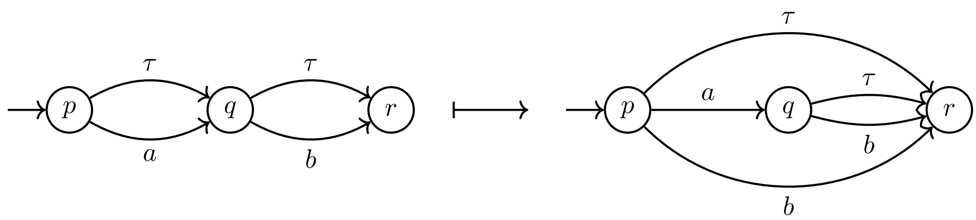

Figure 7 τ-Desaturation maps the left LTS to the right LTS.

In this section, we provide a procedure for obtaining a minimal quotient for $\equiv_x$, the key to which is the inverse operation of τ-desaturation. Consider Figure 7: to go from the right LTS to the left LTS, observe that one may reintroduce the $p \xrightarrow{\tau} q$ transition, and then remove the covered $p \xrightarrow{\tau} r$ and $p \xrightarrow{b} r$ transitions. The introduction of the $p \xrightarrow{\tau} q$ is known as *τ-saturation*, and this transformation is sound since $q \sqsubseteq_x p$.

▶ **Proposition 5.1** (τ-Saturation)**.** *Let $L = \langle S, \mathcal{A}_\tau, \to, \iota\rangle$ be an LTS. Let $p, q \in S$ be such that $q \sqsubseteq_x p$, and define $\overline{L} \coloneqq L \cup \{(p, \tau, q)\}$. Then $L \equiv_x \overline{L}$. Moreover, for all $s \in S$, it holds that s in L is equivalent to s in $\overline{L}$.*

A reduction is then given by performing τ-saturation whenever doing so would result in covered transitions that can be removed. We say a transition $p \xrightarrow{\alpha} q$ of an LTS, L, is *coverable* if it can be made covered by a τ-saturation step, i.e. if it is an element of:

$$\mathsf{Coverable}(p) \coloneqq \{(p, \alpha, q) \mid \exists p' \sqsubset_x p : p \xrightarrow{\alpha} q \text{ is covered in } L \cup \{(p, \tau, p')\}\} \quad .$$

▶ Remark 5.2. In the definition above, we require $p' \sqsubset_x p$ rather than $p' \sqsubseteq_x p$ since (p, τ, p') cannot be inert.

The existential quantifier in this definition strongly suggests that there may not be a unique, deterministic way to remove coverable transitions, for there may be several τ-saturation steps that can be used to cover a transition. This raises the following question: in case we are given such a choice, is one more optimal than the other?

▶ **Example 5.3.** In this example, we show that this choice matters. Consider the LTS in Figure 8. Here, it holds that each $s_i \sqsubset_x u$, and consequently we may τ-saturate by adding $u \xrightarrow{\tau} s_i$. Note that if we add a $u \xrightarrow{\tau} s_i$ transition, then each $u \xrightarrow{\alpha} \perp$ transition for which $\alpha \in S_i$ becomes covered, and therefore may be removed. Since each $u \xrightarrow{\alpha} \perp$ transition is such that α is in some S_i, we may remove every such transition using τ-saturation.

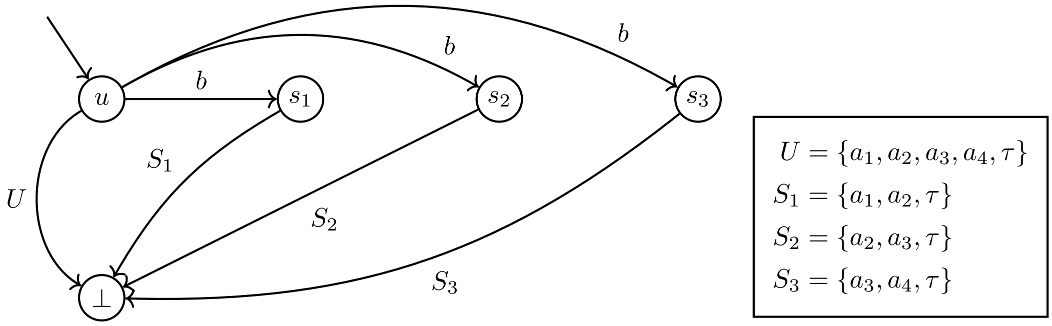

Figure 8 Example showing that τ-saturation must be performed carefully.

To demonstrate that the order of τ-saturation is important, suppose we add a $u \xrightarrow{\tau} s_2$ transition. Then we may remove the $u \xrightarrow{\alpha} \perp$ transition for each $\alpha \in S_2$. So one transition was added, three were removed, and we are left with a $u \xrightarrow{a_1} \perp$ transition, and a $u \xrightarrow{a_4} \perp$ transition. The net effect is the removal of two transitions. At this point, τ-saturation will not reduce the overall number of transitions: we could add a $u \xrightarrow{\tau} s_1$ transition, but only one transition would be removable, and likewise for s_3.

If instead of adding a $u \xrightarrow{\tau} s_2$ transition, we add a $u \xrightarrow{\tau} s_1$ and $u \xrightarrow{\tau} s_3$, then each $u \xrightarrow{\alpha} \perp$ transition becomes covered, so we may remove five transitions at the cost of adding two. This reduction performs better than when we added the $u \xrightarrow{\tau} s_2$ transition, so we conclude that the choice of τ-saturation is important.

Whereas $\mathsf{Coverable}(p)$ describes the set of transitions that can be covered using *some* τ-saturation step, we define $\mathsf{CoverableBy}((p, \tau, p'))$ as the set of transitions that are covered by a *given* τ-saturation step.

$$\mathsf{CoverableBy}((p, \tau, p')) \coloneqq \{(p, \alpha, q) \mid p \xrightarrow{\alpha} q \text{ is covered in } L \cup \{(p, \tau, p')\}\} \quad .$$

For a given state p and set $Y \subseteq \{(p, \alpha, q) \mid p \xrightarrow{\alpha} q\}$, we say a set, $X \subseteq \{(p, \tau, p') \mid p' \sqsubset_x p\}$, is a *cover* of Y iff $\bigcup_{(p,\tau,p') \in X} \mathsf{CoverableBy}((p, \tau, p')) \supseteq Y$, and say X is a cover of p iff X is a cover of $\mathsf{Coverable}(p)$. What the previous example highlights is that some covers are smaller than others, and thus result in better reductions. As suggested by this example (and the terminology we use), the problem of finding a minimal cover is NP-complete[1], by reduction to and from the set cover problem.

[1] Typically, the notion of "completeness" is reserved for decision problems. We extend the notion to optimisation problems here since, in this case, there exist polynomial-time reductions between the decision and optimisation variants.

▶ **Definition 5.4** (Set Cover Problem). *An instance of the set cover problem is a pair $\langle U, \mathcal{S}\rangle$ where $\bigcup_{S\in\mathcal{S}} S = U$. We call U the* universe *and $\mathcal{S}$ the* collection*. A cover of U is some subset $\mathcal{C}$ of $\mathcal{S}$ such that $\bigcup_{S\in\mathcal{C}} S = U$. A solution to the problem is some minimal cover of U.*

By considering $\langle \mathsf{Coverable}(p), \{\mathsf{CoverableBy}((p,\tau,p')) \mid p' \sqsubset_x p\}\rangle$ as an instance of the set cover problem, it is not too hard to see that an algorithm for solving the set cover problem can be used to find a minimal cover for some state p. Likewise, given an instance $\langle U, \mathcal{S}\rangle$ of the set cover problem, one may construct an LTS similar to the one Figure 8, such that a minimal cover of the initial state corresponds to a minimal set cover of the given instance. It turns out that minimisation w.r.t. $\equiv_x$ is NP-hard for similar reasons.

▶ **Theorem 5.5.** *Minimisation w.r.t. $\equiv_x$ is NP-hard.*

Algorithm 2 describes a procedure for computing a minimal quotient, as claimed by Corollary 5.10. First, note that Line 3 is superfluous since L_5 is reduced, and hence $L_5 \sim \textsc{CanQuotient}(L)$ holds by Theorem 4.4. Thus, Algorithm 2 essentially computes $\textsc{CanQuotient}(L)$, and then covers each state optimally. As a consequence, an efficient algorithm for the set cover problem allows us to efficiently minimise LTSs w.r.t. $\equiv_x$.

The reason for including the superfluous Line 3 is because it simplifies the minimality proof, due to Proposition 3.13. As a consequence, we need only show that $|L_7| \leq |L_2|$ to show that $|\textsc{MinQuotient}(L)| \leq |L|$. Then, minimality follows from the fact that L_5 is a canonical representative of its $\equiv_x$-equivalence class.

Algorithm 2 A quotienting function that yields a minimal quotient.

1: **procedure** $\textsc{MinQuotient}(L)$
2: $\quad L_1 \coloneqq L_{/\equiv}$
3: $\quad L_2 \coloneqq$ Remove covered transitions from L_1
4: $\quad L_3 \coloneqq$ Desaturate L_2 $\qquad \Big\}\ \textsc{CanQuotient}(L)$
5: $\quad L_4 \coloneqq$ Remove covered transitions from L_3
6: $\quad L_5 \coloneqq$ Remove unreachable states from L_4
7: $\quad$ **for all** $p \in S$ **do**
8: $\qquad X_p \coloneqq$ be a minimal cover of p
9: $\quad L_6 \coloneqq L_5 \cup \bigcup_{p\in S} X_p$
10: $\quad L_7 \coloneqq$ Remove covered transitions from L_6
11: $\quad$ **return** L_7

To reason about an LTS, L, throughout different stages of Algorithm 2, we shall henceforth use L_i for $2 \leq i \leq 7$ to denote the LTS in Algorithm 2 indicated by L_i. Likewise, we shall parameterise Coverable and CoverableBy by i as follows:

$$\mathsf{Coverable}_i(p) \coloneqq \{(p,\alpha,q) \mid \exists p' \sqsubset_x p : p_i \xrightarrow{\alpha} q_i \text{ is covered in } L_i \cup \{(p,\tau,p')\}\}$$
$$\mathsf{CoverableBy}_i((p,\tau,p')) \coloneqq \{(p,\alpha,q) \mid p_i \xrightarrow{\alpha} q_i \text{ is covered in } L_i \cup \{(p,\tau,p')\}\} \quad .$$

After Line 2, the transformations used preserve $\sqsubseteq_x$, and therefore we do not find the need to parameterise $\sqsubseteq_x$ by i. We shall use the three LTSs in Figure 9 to demonstrate various claims, and illustrate why $|L_7| \leq |L_2|$.

Firstly, observe the middle LTS. This LTS is reduced, so it is unique. Note that the $p \xrightarrow{a} r$ and $p \xrightarrow{b} r$ transitions are coverable. The former can be covered by adding a $p \xrightarrow{\tau} q$ transition, and the latter can be covered by adding a $p \xrightarrow{\tau} s$ transition. Hence, we indicate these pairs of transitions using dashed and dotted transitions.

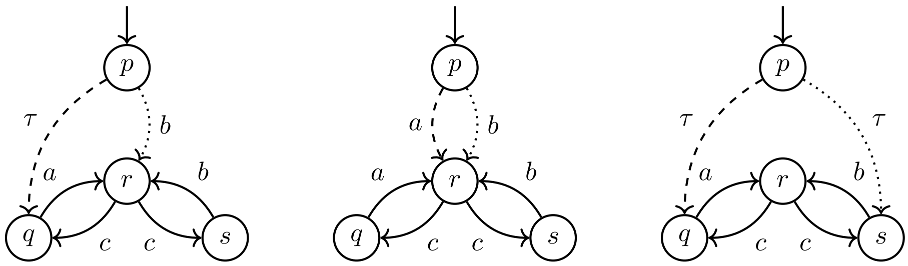

Figure 9 Three weak simulation equivalent LTSs used to demonstrate various claims. They are not coupled similar. Some transitions are dashed or dotted for later reference.

Let L be some LTS such that L_2 is given by the left LTS of Figure 9. Then L_5 is given by the middle LTS. The $p \xrightarrow{a} r$ transition was added by the τ-desaturation step that removed the $p \xrightarrow{\tau} q$ transition. Reintroducing the transition allows us to remove the $p \xrightarrow{a} r$ transition again. Generally speaking, whenever there is a coverable transition of L_5 that is not a transition of L_2, then there is some τ-desaturation step that can be reverted in order to remove this transition. This is given by Corollary 5.7.

▶ **Lemma 5.6.** *Let $L = \langle S, \mathcal{A}_\tau, \rightarrow, \iota \rangle$ be an LTS without inert transitions. Let $\overline{L}$ be obtained from L by some number of τ-desaturation steps. Suppose $p \xrightarrow{\alpha} q$ in $\overline{L}$, but not in L. Then there is some state $p' \neq p$ such that $p \xrightarrow{\tau} p' \xRightarrow{\alpha} q$ in L, but $p \xrightarrow{\tau}\!\!\!\!\!/\;\, p'$ in $\overline{L}$.*

▶ **Corollary 5.7.** *Let $L = \langle S, \mathcal{A}_\tau, \rightarrow, \iota \rangle$ be an LTS such that L_2 and L_7 have the same set of states. Let $Z_p = \{(p, \tau, p') \mid p_2 \xrightarrow{\tau} p'_2 \wedge p_3 \xrightarrow{\tau}\!\!\!\!\!/\;\, p'_3\}$. Suppose $p_5 \xrightarrow{\alpha} q_5$ and $p_2 \xrightarrow{\alpha}\!\!\!\!\!/\;\, q_2$. Then there exists some $(p, \tau, p') \in Z_p$ such that $(p, \alpha, q) \in \mathsf{CoverableBy}_5((p, \tau, p'))$.*

Thus, we can use the set Z_p defined in Corollary 5.7 to cover and remove part of $\mathsf{Coverable}_5(p)$. When we remove transitions in this way, the resulting LTS is no larger than L. However, not all coverable transitions get removed. In our previous example, the $p \xrightarrow{b} r$ transition does not get removed. This transition can be removed by extending Z_p with the $p \xrightarrow{\tau} s$ transition, and we obtain the LTS on the right of Figure 9. In general, the set Z_p can always be extended in this way in order to cover p.

▶ **Lemma 5.8.** *Let $L = \langle S, \mathcal{A}_\tau, \rightarrow, \iota \rangle$ be an LTS such that L_2 and L_7 have the same set of states, and let:*

$$\begin{aligned} &Y_p \textit{ be a cover of } \{(p, \alpha, q) \in \mathsf{Coverable}_5(p) \mid p_2 \xrightarrow{\alpha} q_2\} \\ &Z_p = \{(p, \tau, p') \mid p_2 \xrightarrow{\tau} p'_2 \wedge p_3 \xrightarrow{\tau}\!\!\!\!\!/\;\, p'_3\} \quad . \end{aligned}$$

Then $Y_p \cup Z_p$ is a cover of p in L_5.

If the set Y_p in Lemma 5.8 is minimal, then Y_p is no larger than the subset of $\mathsf{Coverable}_5(p)$ covered by elements of Y_p. Thus, removing transitions using Y_p does not increase the size of the LTS. Since this was also the case for Z_p, we can use $Y_p \cup Z_p$ in place of X_p on Line 8, and this guarantees that $|L_2| \leq |L_7|$. Then, since X_p is no larger than $Y_p \cup Z_p$, and it covers all the same transitions, we also guarantee $|L_2| \leq |L_7|$ by using X_p.

▶ **Theorem 5.9.** *Let L be an LTS. Then $|L_7| \leq |L|$.*

Proof. As previously discussed, we need only show that $|L_7| \leq |L_2|$. This requires showing that if L_2 and L_7 have an equal number of states, then L_7 has at most as many transitions as L_2. Suppose L_2 and L_7 have the same number of states. Then none of the transformations after Line 3 modify the set of states, and hence L_2 and L_7 have the same set of states. For an arbitrary state, p, let X_p be the set defined on Line 8, let Y_p be a *minimal* cover of $\hat{Y}_p = \{(p, \alpha, q) \in \mathsf{Coverable}_5(p) \mid p_2 \xrightarrow{\alpha} q_2\}$, and let $Z_p = \{(p, \tau, p') \mid p_2 \xrightarrow{\tau} p_2' \wedge p_3 \not\xrightarrow{\tau} p_3'\}$. Figure 10 depicts an injection, f, from the transitions of L_7 to the transitions of L_2, with the following form:

$$f((p, \alpha, q)) = \begin{cases} (p, \alpha, q) & \text{if } (p, \alpha, q) \notin X_p \\ \mathsf{rep}_p \circ g_p((p, \alpha, q)) & \text{if } (p, \alpha, q) \in X_p \end{cases}$$

$$\mathsf{rep}_p((p, \alpha, q)) = \begin{cases} (p, \alpha, q) & \text{if } (p, \alpha, q) \in Z_p \setminus Y_p \\ h_p((p, \alpha, q)) & \text{if } (p, \alpha, q) \in Y_p \end{cases} .$$

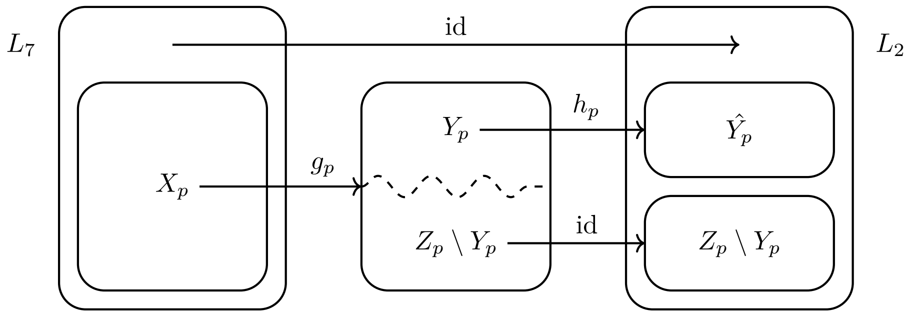

Figure 10 Structure of an injection from transitions of L_7 to transitions of L_2. Each arrow represents an injection, and id is the identity function.

By Lemma 5.8, $Y_p \cup Z_p$ is a cover of p_5. Since X_p is a minimal cover of p_5, there is an injection from X_p to $Y_p \cup Z_p$. Let g_p be such an injection. Now we show that there is an injection from Y_p to $\hat{Y}_p$. Consider some $(p, \tau, p') \in Y_p$, and suppose for the sake of contradiction that for each $(p, \alpha, q) \in \mathsf{CoverableBy}_5((p, \tau, p')) \cap \hat{Y}$, there is some $(p, \tau, p'') \in Y_p$ such that $p' \neq p''$ and $(p, \alpha, q) \in \mathsf{CoverableBy}_5((p, \tau, p'')) \cap \hat{Y}$. Then $Y_p \setminus \{(p, \tau, p')\}$ is a smaller cover of $\hat{Y}$, contradicting our choice of Y_p. Therefore, we may construct an injection from Y_p to $\hat{Y}_p$ by mapping each $(p, \tau, p') \in Y_p$ to some $(p, \alpha, q) \in \hat{Y}$ that is uniquely removed by (p, τ, p'). Let h_p be such an injection, thereby concluding the construction of f.

Now we show that rep_p is an injection. Since both h_p and the identity are injections, it suffices to show that $\hat{Y}_p \cap (Z_p \setminus Y_p) = \emptyset$. By definition of $\hat{Y}_p$, each of its transitions is a transition of L_5. By definition of Z_p, each of its transitions is not a transition of L_3, and hence not a transition of L_5. Therefore, these two sets do not intersect.

Now we show that f is an injection. Note that f maps a transition with source p to another transition with source p. Therefore, we may fix p and show that:

1. $f((p, \alpha, q)) \in \hat{Y}_p$ implies $(p, \alpha, q) \in X_p$, and
2. $f((p, \alpha, q)) \in Z_p \setminus Y_p$ implies $(p, \alpha, q) \in X_p$.

For (1), take $f((p, \alpha, q)) \in \hat{Y}_p$ and assume for the sake of contradiction that $(p, \alpha, q) \notin X_p$. Then $f((p, \alpha, q)) = (p, \alpha, q)$, and (p, α, q) is a transition of both L_2 and L_7. However, $\hat{Y}_p$ is a set of removable transitions in L_5, and therefore no transition of $\hat{Y}_p$ is a transition of L_7. This contradicts $p_7 \xrightarrow{\alpha} q_7$.

For (2), take $f((p,\alpha,q)) \in Z_p \setminus Y_p$ and assume for the sake of contradiction that $(p,\alpha,q) \notin X_p$. Then $f((p,\alpha,q)) = (p,\alpha,q)$, and (p,α,q) is a transition of both L_2 and L_7. However, Z_p is a set of transitions that do not belong to L_3, which means the (p,α,q) transition was added on Line 9, and therefore $(p,\alpha,q) \in X_p$. This contradicts our assumption that $(p,\alpha,q) \notin X_p$.

Thus, f is an injection from transitions of L_7 to L_2, which was what we wanted. ◀

▶ **Corollary 5.10.** *Algorithm 2 computes a minimal quotient.*

As discussed previously, finding minimal covers of states is NP-complete, so Algorithm 2 cannot be implemented efficiently unless $P = NP$. However, this procedure allows us to leverage state of the art algorithms for the set cover problem. Moreover, the instances of the set cover problem that need to be solved are typically small. Recall that finding a minimal cover of some state p corresponds to finding a minimal set cover for the instance $\langle \mathsf{Coverable}(p), \{\mathsf{CoverableBy}((p,\tau,p')) \mid p' \sqsubset_x p\}\rangle$. The size of $\mathsf{Coverable}(p)$ is bounded by the out-degree of p since, in the worst case, every outgoing transition of p is coverable. Thus, the size of the set that needs to be covered is small. Furthermore, by the following proposition it suffices to consider all $p' \in \max(p \sqsupset_x)$, rather than all $p' \sqsubset_x p$, both in the construction of the universe, $\mathsf{Coverable}(p)$, and in the construction of the collection, $\{\mathsf{CoverableBy}((p,\tau,p')) \mid p' \sqsubset_x p\}$. Lastly, note that $\mathsf{CoverableBy}((p,\tau,p'))$ may be empty, so the size of the collection may be reduced even further.

▶ **Proposition 5.11.** *Let $L = \langle S, \mathcal{A}_\tau, \rightarrow, \iota\rangle$ be an LTS without inert or covered transitions, and let $p, p' \in S$ be such that $p' \sqsubset_x p$. Then $\mathsf{CoverableBy}((p,\tau,p')) \subseteq \mathsf{CoverableBy}((p,\tau,p''))$ holds for all p'' satisfying $p' \sqsubseteq_x p'' \sqsubset_x p$.*

The output of Algorithm 2 is – generally speaking – not canonical since on Line 8, there may be multiple minimal covers to choose from. To show how this step can be made deterministic, take some LTS L, and suppose the states of L_5 can be ordered $p_1, \ldots, p_n$. A cover for some p_i can be written as a binary array of length n, where the jth bit equals 1 iff $p_i \xrightarrow{\tau} p_j$ belongs to the cover. Then one can deterministically assign a minimal cover to p_i by choosing the cover that is lexicographically minimal w.r.t. the chosen order. Since L_5 is canonical, we only need some algorithm that returns the “same” ordering for isomorphic LTSs. Specifically, if $L_5' \sim L_5$ and the states of L_5' are ordered $q_1, \ldots, q_n$, then $f : p_i \mapsto q_i$ is an isomorphism.

Algorithms for canonical forms of graphs work by computing such an ordering, and labelling vertices accordingly. It is shown in [21] how this can be done for LTSs, assuming a universal ordering on the set of actions. In practice, this is a fair assumption since one could order actions by their unicode representations. Current algorithms for computing canonical forms have an exponential worst case complexity, so one may choose to forego this step.

6 Conclusion

We investigated the problem of finding canonical and minimal quotients for coupled and weak simulation equivalence, two closely related notions of equivalence for LTSs. While both equivalences turn out to admit canonical quotients, we showed that these are not necessarily minimal. Furthermore, somewhat to our surprise, for both equivalences, the problem of computing a minimal quotient turns out to be NP-complete. While this is worse than for strong, weak and branching bisimilarity, for which quotients can be computed in polynomial time, it is better than for (weak) trace equivalence, for which the problem is PSPACE-complete.

The quotienting function we describe for obtaining a minimal quotient essentially computes a canonical quotient and subsequently reverses some of the operations that ultimately led to an increase in size. Assuming a universal ordering on the set of actions, computing the canonical quotient as an intermediary step allows the quotienting function to compute a canonical minimal quotient, since we may take advantage of canonical labelling algorithms.

Interestingly, our approach relied little on specific properties of the equivalences in question. For example, the proof of Theorem 4.4 relies mainly on Lemma 4.3, which shows that equivalent LTSs that are desaturated and have no covered transitions are weakly bisimilar. Afterwards, we use a previously established fact for canonicity of weakly bisimilar LTSs.

Our notion of τ-desaturation depends on the τ-law for the given equivalence. It happens to be that coupled and weak simulation equivalence have similar τ-laws, which allows the results in this paper to generalise well to both equivalences. However, we cannot reasonably expect this to be the case for the remainder of the linear time-branching time spectrum. Therefore, it would be worthwhile to investigate how the τ-laws for different equivalences can be used to describe canonical and minimal quotients, as they do here. In particular, it would be interesting to see whether there is a relation between the spectrum and the computational complexity of minimising with respect to a given equivalence, and identify the inflection points across this spectrum.

References

1 Luca Aceto, David de Frutos Escrig, Carlos Gregorio-Rodríguez, and Anna Ingolfsdottir. Axiomatizing weak simulation semantics over BCCSP. *Theoretical Computer Science*, 537:42–71, June 2014. doi:10.1016/j.tcs.2013.03.013.

2 Benjamin Bisping and Luisa Montanari. Coupled similarity and contrasimilarity, and how to compute them. *Arch. Formal Proofs*, 2023, August 2023. URL: https://www.isa-afp.org/entries/Coupledsim_Contrasim.html.

3 Benjamin Bisping and Uwe Nestmann. Computing Coupled Similarity. In Tomáš Vojnar and Lijun Zhang, editors, *Tools and Algorithms for the Construction and Analysis of Systems*, pages 244–261, Cham, 2019. Springer International Publishing. doi:10.1007/978-3-030-17462-0_14.

4 Tommaso Bolognesi and Scott A. Smolka. Fundamental results for the verification of observational equivalence: A survey. In Harry Rudin and Colin H. West, editors, *Protocol Specification, Testing and Verification VII, Proceedings of the IFIP WG6.1 Seventh International Conference on Protocol Specification, Testing and Verification, Zurich, Switzerland, 5-8 May, 1987*, pages 165–179. North-Holland, January 1987. Pages: 179.

5 Mark Bouwman, Maurice Laveaux, Bas Luttik, and Tim A. C. Willemse. Decompositional branching bisimulation minimisation of monolithic processes. In *FACS*, volume 13712 of *Lecture Notes in Computer Science*, pages 161–182. Springer, 2022. doi:10.1007/978-3-031-20872-0_10.

6 Doron Bustan and Orna Grumberg. Simulation Based Minimization. In David McAllester, editor, *Automated Deduction - CADE-17*, pages 255–270, Berlin, Heidelberg, 2000. Springer. doi:10.1007/10721959_20.

7 Jaana Eloranta. Minimizing the number of transitions with respect to observation equivalence. *BIT Numerical Mathematics*, 31(4):576–590, December 1991. doi:10.1007/BF01933173.

8 Jaana Eloranta, Martti Tienari, and Antti Valmari. Essential transitions to bisimulation equivalences. *Theoretical Computer Science*, 179(1):397–419, June 1997. doi:10.1016/S0304-3975(96)00281-2.

9 Hubert Garavel, Frédéric Lang, Radu Mateescu, and Wendelin Serwe. CADP 2011: a toolbox for the construction and analysis of distributed processes. *Int. J. Softw. Tools Technol. Transf.*, 15(2):89–107, 2013. doi:10.1007/S10009-012-0244-Z.

10 Thomas Gibson-Robinson, Philip J. Armstrong, Alexandre Boulgakov, and A. W. Roscoe. FDR3: a parallel refinement checker for CSP. *Int. J. Softw. Tools Technol. Transf.*, 18(2):149–167, 2016. doi:10.1007/S10009-015-0377-Y.

11 David N. Jansen, Jan Friso Groote, Jeroen J. A. Keiren, and Anton Wijs. An O(m log n) algorithm for branching bisimilarity on labelled transition systems. In Armin Biere and David Parker, editors, *Tools and Algorithms for the Construction and Analysis of Systems*, pages 3–20, Cham, 2020. Springer International Publishing. doi:10.1007/978-3-030-45237-7_1.

12 Paris C. Kanellakis and Scott A. Smolka. CCS expressions, finite state processes, and three problems of equivalence. *Information and Computation*, 86(1):43–68, May 1990. doi:10.1016/0890-5401(90)90025-D.

13 Antonín Kucera and Richard Mayr. Simulation preorder on simple process algebras. In Jirí Wiedermann, Peter van Emde Boas, and Mogens Nielsen, editors, *Automata, Languages and Programming, 26th International Colloquium, ICALP'99, Prague, Czech Republic, July 11-15, 1999, Proceedings*, Lecture Notes in Computer Science, pages 503–512. Springer, 1999. doi:10.1007/3-540-48523-6_47.

14 Antonín Kučera. On finite representations of infinite-state behaviours. *Information Processing Letters*, 70(1):23–30, April 1999. doi:10.1016/S0020-0190(99)00029-0.

15 Antonín Kučera and Javier Esparza. A Logical Viewpoint on Process-algebraic Quotients. *Journal of Logic and Computation*, 13(6):863–880, December 2003. doi:10.1093/logcom/13.6.863.

16 Maurice Laveaux and Tim A. C. Willemse. Decomposing monolithic processes in a process algebra with multi-actions. *J. Log. Algebraic Methods Program.*, 132:100858, 2023. doi:10.1016/J.JLAMP.2023.100858.

17 Eduardo Costa Martins and Tim Willemse. Minimal and canonical quotients for simulation equivalences. *CoRR*, abs/2606.21450, 2026. doi:10.48550/arXiv.2606.21450.

18 David Park. Concurrency and automata on infinite sequences. In Peter Deussen, editor, *Theoretical Computer Science*, pages 167–183, Berlin, Heidelberg, 1981. Springer. doi:10.1007/BFb0017309.

19 Joachim Parrow and Peter Sjödin. The complete axiomatization of cs-congruence. In Patrice Enjalbert, Ernst W. Mayr, and Klaus W. Wagner, editors, *STACS 94, 11th Annual Symposium on Theoretical Aspects of Computer Science, Caen, France, February 24-26, 1994, Proceedings*, volume 775 of *Lecture Notes in Computer Science*, pages 557–568, Berlin, Heidelberg, 1994. Springer. doi:10.1007/3-540-57785-8_171.

20 Michel A. Reniers, Rob Schoren, and Tim A.C. Willemse. Results on Embeddings Between State-Based and Event-Based Systems. *The Computer Journal*, 57(1):73–92, January 2014. doi:10.1093/comjnl/bxs156.

21 Edd Turner, Michael Leuschel, Corinna Spermann, and Michael Butler. Symmetry reduced model checking for b. In *First Joint IEEE/IFIP Symposium on Theoretical Aspects of Software Engineering (TASE '07)*, pages 25–34, 2007. doi:10.1109/TASE.2007.50.

22 Irek Ulidowski. Equivalences on observable processes. In *[1992] Proceedings of the Seventh Annual IEEE Symposium on Logic in Computer Science*, pages 148–159, 1992. doi:10.1109/LICS.1992.185529.

23 Rob J. van Glabbeek. The linear time – Branching time spectrum. In J. C. M. Baeten and J. W. Klop, editors, *CONCUR '90 Theories of Concurrency: Unification and Extension*, pages 278–297, Berlin, Heidelberg, 1990. Springer. doi:10.1007/BFb0039066.

24 Rob J. van Glabbeek. The linear time – Branching time spectrum II. In Eike Best, editor, *CONCUR'93*, pages 66–81, Berlin, Heidelberg, 1993. Springer. doi:10.1007/3-540-57208-2_6.