# Weak Bisimulation Up to Elaboration

Damien Pous

# Weak Bisimulation up to Elaboration*

Damien Pous

ENS Lyon

**Abstract** We study the use of the elaboration preorder (due to Arun-Kumar and Natarajan) in the framework of up-to techniques for weak bisimulation. We show that elaboration yields a correct technique that encompasses the commonly used up to expansion technique. We also define a theory of up-to techniques for elaboration that in particular validates an elaboration up to elaboration technique, while it is known that weak bisimulation up to weak bisimilarity is unsound. In this sense, the resulting setting improves over previous works in terms of modularity. Our results are obtained using nontrivial proofs that exploit termination arguments. In particular, we need the termination of internal computations for the up-to techniques to be correct. We show how this condition can be relaxed to some extent in order to handle processes exhibiting infinite internal behaviour.

## Introduction

Weak bisimilarity ($\approx$) is a commonly used behavioural equivalence for the analysis of concurrent systems. *Weak* here means distinguishing between visible actions of a system, that express interactions with its environment, and $\tau$ *transitions*, that are treated as internal moves, and hence unobservable. To prove a weak bisimilarity result, one usually exhibits a relation $\mathcal{R}$ between states of the systems being compared, and shows that $\mathcal{R}$ is a weak bisimulation relation (we shall often simply use 'bisimilarity' and 'bisimulation' in the sequel, and refer explicitly to the strong version of these relations when necessary).

The crux of a bisimulation proof is often the study of silent transitions, as this part of the proof expresses the fact that internal calculations do not introduce unexpected behaviours. Typically, this is where it is shown that an optimisation is valid, that an encoding is fully abstract, or that some invariant about a data structure is preserved. Because one has to take into account all possible silent transitions, this makes bisimulation relations grow a lot, although, intuitively, many of the $\tau$ transitions being examined are irrelevant from the point of view of the overall behaviour of the system.

Several *up-to techniques* have been proposed to alleviate the task of bisimulation proofs. The idea of up-to techniques is to manipulate functions from

* Author's version of the paper published by Springer in Proc. CONCUR'06, available at `http://dx.doi.org/10.1007/11817949_26`. This work has been supported by the french initiatives "ACI GEOCAL" and "ANR ARASSIA, projet ModyFiable"

relations to relations, that compute a form of closure. These functions are used in the bisimulation game as shown on the diagram on the left below:

$$\begin{array}{ccc} P & \mathcal{R} & Q \\ {\scriptstyle\alpha}\downarrow & & \Downarrow\widehat{\alpha} \\ P' & \mathcal{F}(\mathcal{R}) & Q' \end{array} \qquad \begin{array}{l} \mathcal{U} : \mathcal{R} \mapsto \mathcal{R} \cup \approx \\ \mathcal{W} : \mathcal{R} \mapsto \approx \mathcal{R} \approx \end{array} \qquad \begin{array}{l} \mathcal{X} : \mathcal{R} \mapsto \succsim \mathcal{R} \approx \\ \mathcal{E} : \mathcal{R} \mapsto \succsim\!\!\!\!\approx \mathcal{R} \approx \end{array}$$

When the symmetric candidate relation $\mathcal{R}$ contains a pair $\langle P, Q\rangle$, and $P$ does a transition to $P'$ along an action $\alpha$, $Q$ has to perform the same action, modulo some internal computation ($\tau$ transitions), to yield a process $Q'$. The point is that the resulting pair $\langle P', Q'\rangle$ has to belong to $\mathcal{F}(\mathcal{R})$ instead of $\mathcal{R}$ (bisimulation is obtained by taking the identity function for $\mathcal{F}$).

For example, if we take for $\mathcal{F}$ the function $\mathcal{U}$ above, we can use known facts about $\approx$ when examining the transitions of processes related by $\mathcal{R}$. More interestingly, function $\mathcal{W}$ allows one to apply known bisimilarity laws to transform $P'$ and $Q'$ in order to obtain a pair belonging to $\mathcal{R}$. Unfortunately, the technique given by $\mathcal{W}$ is unsound, as shown by the following standard counterexample (written in CCS): consider a process $P$ which is not bisimilar to 0, and define $\mathcal{R} \triangleq \{\langle \tau.P, 0\rangle\}$. Since $\tau.P \approx P$, we can use $\mathcal{W}$ to repeatedly undo the silent transition $\tau.P \xrightarrow{\tau} P$, so that in the game of weak bisimulation up to weak bisimilarity, we never explore the actual behaviour of $P$.

$$\begin{array}{ccccccccc} \tau.P & \mathcal{R} & 0 & & & & & & \\ {\scriptstyle\tau}\downarrow & & & & & & & & \\ P & \approx & \tau.P & \mathcal{R} & 0 & & & & \\ & & {\scriptstyle\tau}\downarrow & & & & & & \\ & & P & \approx & \tau.P & \mathcal{R} & 0 & & \\ & & & & {\scriptstyle\tau}\downarrow & & & & \\ & & & & & \ldots & & & \end{array}$$

To address this difficulty, [10] introduces *expansion* ($\succsim$), a behavioural preorder included in weak bisimilarity, that leads to the up-to technique given by function $\mathcal{X}$ defined above. Unlike $\mathcal{W}$, $\mathcal{X}$ yields a correct proof technique, because expansion expresses a kind of efficiency constraint: intuitively, if $P \succsim Q$, then $Q$ is 'faster' than $P$, in the sense that $P$ and $Q$ exhibit the same behaviour, but $Q$ has to require less silent transitions to do so (we define $\succsim$ formally below). Since $P \succsim \tau.P$ does not hold, $\mathcal{X}$ rules out the above counterexample.

However, as experience shows [5,7], there are cases where reasoning up to expansion does not suffice, because the silent moves one would like to factor out in a bisimulation proof are not contained in expansion. Typically, this occurs when the 'faster process' has to spend some time at certain points to do some internal bookkeeping, for instance to update a data structure. To go beyond expansion, we have proposed in [7] a general and, at least to some extent, modular theory of up-to techniques for weak bisimulation. [7] introduces a notion of *controlled relation*, which guarantees that a given relation can be used in place of expansion. Several sufficient conditions for a relation to be controlled are given, among which, most notably, a criterion based on a termination property that prevents

the existence of what we call 'infinite ladders' like depicted on the diagram above (which shows an infinite $\xrightarrow{\tau}\approx$ ladder).

Nevertheless, the resulting setting lacks flexibility, essentially because the property of being a controlled relation is not stable by union. This prevents the incremental construction of bisimulation proofs, and thus represents a drawback in terms of modularity: in this setting, extending a proof requires an involved knowledge of the up-to techniques at work, in order to check that relations remain controlled along the extension (we discuss this at the end of Sect. 3).

In this paper, we focus on the *elaboration* preorder, which has been introduced in [2]. Elaboration, written $\gtrsim$, is somehow the dual of expansion: informally, $P \gtrsim Q$ means that $P$ performs *at least as many silent transitions* as $Q$, while exhibiting the same behaviour. Elaboration strictly contains expansion, and is in some sense very close to $\approx$. The focus in [2] is on congruence properties of $\gtrsim$ in the setting of CCS, and on the axiomatisation of this relation.

The first result we establish is that $\gtrsim$ yields a correct up-to technique for bisimulation when the system is *terminating*, that is, when it does not exhibit infinite sequences of silent transitions. Rather remarkably, this result cannot be derived by a simple diagram chasing (as is the case for the up to expansion technique). The proof relies instead on the approach of [7], the termination hypothesis being used to derive the absence of infinite 'ladders'.

Our second contribution is to show that $\gtrsim$ supports the development of a modular theory of up-to techniques, along the lines of the treatment of up-to techniques for strong bisimulation presented in [9]. This represents a significant step forward w.r.t. [7] in terms of modularity, notably because the *up to transitivity* proof technique, given by $\mathcal{T} : \mathcal{R} \mapsto \mathcal{R}^*$, is shown to be correct for elaboration (under the previous termination hypothesis). We devote particular attention to this important result: when applicable to reason about a coinductively defined relation $\simeq$, $\mathcal{T}$ provides the powerful techniques given by $\mathcal{R} \mapsto (\mathcal{R} \cup \simeq)^\star$, or the more restrictive (but more commonly used) $\mathcal{R} \mapsto \simeq \mathcal{R} \simeq$. As we show in the paper, an application of the resulting framework is the study of an *up to polyadic contexts* proof technique (a polyadic context is a context with possibly many holes in it). Establishing directly the correctness of this technique can be really tedious, while correctness of $\mathcal{T}$ allows one to derive a modular proof that boils down to correctness in the – simpler – monadic case.

Although it can be argued that the termination of silent transitions is realistic in many systems (typically, when silent moves are used to update the internal representation of a data structure), some programming techniques may be the source of deliberate infinite internal behaviours, such as busy waiting loops. In order to be able to handle some of these situations, we move to a setting where silent transitions are decomposed into two kinds of internal moves, respectively called the *progressive* and *non-progressive* silent transitions (as in [4]). Only progressive silent transitions are supposed to be terminating. We show that under this relaxed hypothesis, the previous results can be adapted, by validating an 'up to progressive elaboration' technique for bisimulation, and showing the correctness of progressive elaboration up to transitivity. While being similar to

the proofs of the results above, establishing the properties for non-terminating systems involves rather intricate usages of well-founded induction. Beyond this technical aspect, we believe that the general proof pattern adopted in this paper exposes an interesting application of rewriting techniques to concurrency.

*Outline of the paper.* In Sect. 1, we introduce our notations and briefly recall the results of [7] that will be used in the sequel. In Sect. 2 we define the elaboration preorder, and establish correctness of the up to elaboration proof technique when silent transitions of the system are terminating. We develop in Sect. 3 a theory of up-to techniques for elaboration, and draw a comparison with existing techniques. We extend these results to non-terminating systems in Sect. 4, and give final remarks in Sect. 5.

# 1 Preliminaries

## 1.1 Labelled Transition Systems, Definitions

We consider labelled transition systems (LTS) $\langle \mathcal{P}, \mathcal{L}, \rightarrow \rangle$, with domain $\mathcal{P}$, *labels* or *actions* in $\mathcal{L}$ and transition relation $\rightarrow \subseteq \mathcal{P} \times \mathcal{L} \times \mathcal{P}$. The elements of $\mathcal{P}$ are called *processes* and are denoted by $P, Q$. Except in Sect. 4, $\mathcal{L}$ will always implicitly contain a distinguished *silent action*, noted $\tau$. We let $\alpha, \beta$ (resp. $a, b$) range over actions, in $\mathcal{L}$ (resp. *visible actions*, in $\mathcal{L}\backslash\{\tau\}$). Some examples will be given using the syntax of CCS, which we suppose known to the reader.

We let $\mathcal{R}, \mathcal{S}, \mathcal{B}$ range over binary relations (simply called *relations* in the sequel) between processes. We denote respectively by $\mathcal{R}^+, \mathcal{R}^=, \mathcal{R}^\star$ the transitive, reflexive, transitive and reflexive closures of a relation $\mathcal{R}$. $P\mathcal{R}Q$ means $\langle P, Q \rangle \in \mathcal{R}$. The composition of two relations $\mathcal{R}$ and $\mathcal{S}$, written $\mathcal{RS}$, is defined by $\mathcal{RS} \triangleq \{\langle P, Q \rangle / P\mathcal{R}T \text{ and } T\mathcal{S}Q \text{ for some process } T\}$. We also define the inverse of a relation: $\mathcal{R}^{-1} \triangleq \{\langle P, Q \rangle / Q\mathcal{R}P\}$. $\mathcal{I}$ is the identity relation, defined by $\mathcal{I} \triangleq \{\langle P, P \rangle / P \in \mathcal{P}\}$. We say that $\mathcal{R}$ *contains* $\mathcal{S}$ (alternatively, that $\mathcal{S}$ is contained in $\mathcal{R}$), written $\mathcal{S} \subseteq \mathcal{R}$, if $P\mathcal{S}Q$ implies $P\mathcal{R}Q$. Given an action $\alpha$, the set of transitions along $\alpha$ induces a relation denoted by $\xrightarrow{\alpha}$: $\xrightarrow{\alpha} \triangleq \{\langle P, Q \rangle / \langle P, \alpha, Q \rangle \in \rightarrow\}$. Its inverse is written using a reversed arrow: $\xleftarrow{a} = (\xrightarrow{a})^{-1}$, and similarly for other forms of arrows, defined below. Finally, we call *function* a mapping from relations to relations.

**Definition 1.1 (Termination).** *A relation* $\mathcal{R}$ terminates *if there is no infinite sequence* $(P_i)_{i \in \mathbb{N}}$ *such that* $\forall i, P_i \mathcal{R} P_{i+1}$.

Such terminating relations are also called *Nœtherian* in the literature. They lead to the powerful technique of proof by *well-founded induction* on which we heavily rely in the sequel. We will also make use of *lexicographic inductions*, that is, inductions based on lexicographic orders. In our case, such orders will always consist of the product of a terminating relation $\mathcal{R}$ with the standard ordering of natural numbers: $\langle P, n \rangle \succ \langle Q, m \rangle$ iff $P\mathcal{R}Q$ or $(P = Q \text{ and } n > m)$.

The definitions of behavioural equivalences and preorders will make use of the following *weak transition* relations.

**Definition 1.2 (Weak transitions).**

$$\stackrel{\widehat{\alpha}}{\rightarrow} \triangleq \begin{cases} \stackrel{\tau}{\rightarrow}^{=} & \text{if } \alpha = \tau \\ \stackrel{a}{\rightarrow} & \text{if } \alpha = a \in \mathcal{L}\backslash\{\tau\} \end{cases} \qquad \begin{array}{l} \stackrel{\alpha}{\Rightarrow} \triangleq \stackrel{\tau}{\rightarrow}^{\star} \stackrel{\alpha}{\rightarrow} \stackrel{\tau}{\rightarrow}^{\star} \\ \stackrel{\widehat{\alpha}}{\Rightarrow} \triangleq \stackrel{\tau}{\rightarrow}^{\star} \stackrel{\widehat{\alpha}}{\rightarrow} \stackrel{\tau}{\rightarrow}^{\star} \end{array}$$

We can remark the following properties: $\stackrel{\widehat{\tau}}{\Rightarrow} = \stackrel{\tau}{\rightarrow}^{\star}$, $\stackrel{\tau}{\Rightarrow} = \stackrel{\tau}{\rightarrow}^{+}$, $\stackrel{\widehat{a}}{\Rightarrow} = \stackrel{a}{\Rightarrow}$ (note in particular the difference between $\stackrel{\widehat{\tau}}{\Rightarrow}$ and $\stackrel{\tau}{\Rightarrow}$).

**Definition 1.3 (Evolution of relations).** *Let $\alpha$ be an action and $\mathcal{R}, \mathcal{S}$ two relations. We say that $\mathcal{R}$* $\alpha$-evolves *to $\mathcal{S}$ if whenever $P\mathcal{R}Q$, $P \stackrel{\alpha}{\rightarrow} P'$ implies $Q \stackrel{\widehat{\alpha}}{\Rightarrow} Q'$ and $P'\mathcal{S}Q'$ for some $Q'$. Given two relations $\mathcal{R}$ and $\mathcal{S}$, we say that:*

- *$\mathcal{R}$* evolves to *$\mathcal{S}$ if $\mathcal{R}$ $\alpha$-evolves to $\mathcal{S}$ for all $\alpha \in \mathcal{L}$,*
- *$\mathcal{R}$* evolves silently *to $\mathcal{S}$ if $\mathcal{R}$ $\tau$-evolves to $\mathcal{S}$,*
- *$\mathcal{R}$* evolves visibly *to $\mathcal{S}$ if $\mathcal{R}$ $a$-evolves to $\mathcal{S}$ for all $a \in \mathcal{L}\backslash\{\tau\}$.*

**Definition 1.4 (Bisimulation, Bisimilarity).** *Let $\mathcal{R}$ be a relation. $\mathcal{R}$ is a* bisimulation *if it is symmetric and evolves to itself.* Bisimilarity*, denoted by $\approx$, is defined as the union of all bisimulations.*

### 1.2 Existing Up-to Techniques for Bisimulation

The following lemma will be useful in the sequel. It states correctness of reasoning up to transitivity on visible actions.

**Lemma 1.5.** *Let $\mathcal{R}$ be a relation. If $\mathcal{R}$ evolves silently to itself, and visibly to $\mathcal{R}^\star$, then $\mathcal{R}^\star$ evolves to itself.*

*Proof.* By two successive inductions, we show that for any $n$, $\mathcal{R}^n$ evolves silently to itself, and $\mathcal{R}^n$ evolves visibly to $\mathcal{R}^\star$ ($\mathcal{R}^n$ is the composition of $\mathcal{R}$ with itself, $n$ times). □

Some important up-to techniques for bisimulation are given by the two following results which are simple reformulations of [7, Theorems 2.6 and 3.6].

**Theorem 1.6.** *Let $\mathcal{R}$ be a symmetric relation. If $\mathcal{R}$ evolves silently to $\succcurlyeq \mathcal{R} \approx$ and visibly to $\mathcal{R}^\star$, then $\mathcal{R}$ is contained in bisimilarity.*

**Theorem 1.7.** *Let $\mathcal{B}$ be a relation contained in bisimilarity, evolving to $\mathcal{B}^\star$, and such that $\mathcal{B}^+ \stackrel{\tau}{\Rightarrow}$ terminates. If $\mathcal{R}$ is a symmetric relation that evolves silently to $\mathcal{B}^\star\mathcal{R}\approx$ and visibly to $\mathcal{R}^\star$, then $\mathcal{R}$ is contained in bisimilarity.*

In both cases, visible and silent transitions are treated differently, and up to transitivity is allowed on visible actions only. The difference between these two results is in the up-to technique that is allowed after a silent action: in the first case, one uses the compression preorder, written $\succcurlyeq$ ($\succcurlyeq$ will be defined in Sect. 2.1). This result is essentially already present in [10,11], without the transitivity on visible actions. In the second case, the up-to technique is given

by a relation $\mathcal{B}$, which has to satisfy a termination property. In [7], the actual requirement for $\mathcal{B}$ is to be a *controlled relation* [7, Definition 3.1], and it is shown that the conditions in the above theorem are sufficient for $\mathcal{B}$ to be controlled.

The compression, used in Theorem 1.6, is not as involved as the sufficient condition expressed by Theorem 1.7. On the other hand, as will be discussed in Sect. 3, the technique given by the former theorem is more amenable to the incremental development of proofs than the setting of the latter.

## 2 Elaboration

### 2.1 Definition and Basic Properties

We now define elaboration, that has been introduced in the setting of CCS in [2].

**Definition 2.1 (Elaboration relation, Elaboration).** *A relation $\mathcal{R}$ is an* elaboration relation *(in short, an* elaboration*) if whenever $P\mathcal{R}Q$:*

*(i) if $P \stackrel{\alpha}{\to} P'$, then $Q \stackrel{\widehat{\alpha}}{\Rightarrow} Q'$ with $P'\mathcal{R}Q'$,*
*(ii) if $Q \stackrel{\alpha}{\to} Q'$, then $P \stackrel{\alpha}{\Rightarrow} P'$ with $P'\mathcal{R}Q'$.*

Elaboration*, denoted by $\succcurlyeq$, is the union of all elaboration relations.*

Note that [2] uses a reversed version of the symbol for elaboration – we adopted this choice to follow the convention in other papers about up-to techniques and behavioural preorders, notably [10].

The intuition behind elaboration is that if $P \succcurlyeq Q$, then $P$ is able to always be *at least as slow* as $Q$, as expressed by clause $(ii)$. In relation to this, we may remark that divergences blur the difference between elaboration and bisimilarity: if $P \approx Q$, then $P|!\tau \succcurlyeq Q$. This observation suggests that elaboration is a coarse relation, rather close to $\approx$ (see also Prop. 2.3 below). Moreover, if we consider the bisimilarity defined by using clause $(ii)$ on both sides, we obtain *progressing bisimulation* [6]. On CCS agents, the latter equivalence (which is contained in $\succcurlyeq$) coincides with the greatest weak bisimulation that is a congruence.

To draw a comparison between $\succcurlyeq$ and other behavioural preorders, we recall the definition of *expansion* [10,11] (called *efficiency preorder* in [1]). A slightly coarser definition of expansion appears in [3,7], here we call it *compression* in order to avoid confusions. The difference has consequences as far as up-to techniques are concerned, as will be explained in Sect. 3.

**Definition 2.2 (Expansion, Compression).**

Expansion*, denoted by $\succsim$, is the largest relation such that whenever $P \succsim Q$,*
- *if $P \stackrel{\alpha}{\to} P'$, then $Q \stackrel{\widehat{\alpha}}{\to} Q'$ with $P' \succsim Q'$,*
- *if $Q \stackrel{\alpha}{\to} Q'$, then $P \stackrel{\alpha}{\Rightarrow} P'$ with $P' \succsim Q'$.*

Compression*, denoted by $\succcurlyeq$, is the largest relation such that whenever $P \succcurlyeq Q$,*
- *if $P \stackrel{\alpha}{\to} P'$, then $Q \stackrel{\widehat{\alpha}}{\to} Q'$ with $P' \succcurlyeq Q'$,*

– *if* $Q \xrightarrow{\alpha} Q'$, *then* $P \stackrel{\widehat{\alpha}}{\Rightarrow} P'$ *with* $P' \succcurlyeq Q'$.

In contrast with $\gtrsim$, $P \succsim Q$ intuitively captures the fact that $Q$ is able to be always faster than $P$ (and similarly for $P \succcurlyeq Q$).

**Proposition 2.3.** *In any LTS, we have* $\sim \subset \succsim \subset \gtrsim \subset \approx$ *and* $\succsim \subset \succcurlyeq \subset \approx$. *Moreover, in CCS,* $a|\tau \begin{array}{c}\not\succsim\\ \gtrsim\end{array} \tau.a$ *and* $a \begin{array}{c}\succsim\\ \not\gtrsim\end{array} a|\tau$.

As shown by the examples above, elaboration and compression are not comparable in general. These examples can be used to make the same observation with *almost weak bisimulation* [10] or *relaxed expansion* [7] instead of compression.

### 2.2 Bisimulation up to Elaboration

In order for elaboration to yield a correct up-to technique, we need a termination hypothesis, for which we introduce the following terminology.

**Definition 2.4 ($\alpha$-terminating LTS).** *Let* $\mathbb{S} = \langle \mathcal{P}, \mathcal{L}, \rightarrow \rangle$ *be an LTS, and* $\alpha \in \mathcal{L}$ *a label of* $\mathbb{S}$. *We say that* $\mathbb{S}$ *is* $\alpha$-terminating *if* $\stackrel{\alpha}{\Rightarrow}$ *terminates.*

**Lemma 2.5.** *Let* $\alpha$ *be an action and* $\mathcal{R}$ *a relation such that* $\mathcal{R}\stackrel{\alpha}{\Rightarrow} \subseteq \stackrel{\alpha}{\Rightarrow}\mathcal{R}$. *If* $\stackrel{\alpha}{\Rightarrow}$ *terminates, then so does* $\mathcal{R}\stackrel{\alpha}{\Rightarrow}$.

*Proof.* First we prove $\varphi : \forall n, \mathcal{R}^n \stackrel{\alpha}{\Rightarrow} \subseteq \stackrel{\alpha}{\Rightarrow}\mathcal{R}^n$. Then, suppose that $\mathcal{R}\stackrel{\alpha}{\Rightarrow}$ does not terminate: there exists an infinite sequence $(Q_i)_{i\geq 0}$ such that $Q_i \ \mathcal{R} \stackrel{\alpha}{\Rightarrow} Q_{i+1}$. Using $\varphi$, we can define by induction an infinite sequence $(P_i)_{i\geq 0}$ such that $P_i \stackrel{\alpha}{\Rightarrow} P_{i+1}$ and $P_i\mathcal{R}^iQ_i$. This sequence is contradictory with the termination of $\stackrel{\alpha}{\Rightarrow}$. □

**Theorem 2.6 (Bisimilarity up to Elaboration).** *In a* $\tau$*-terminating LTS, any symmetric relation* $\mathcal{R}$ *that evolves silently to* $\gtrsim \mathcal{R} \approx$ *and visibly to* $\mathcal{R}^\star$ *is contained in bisimilarity.*

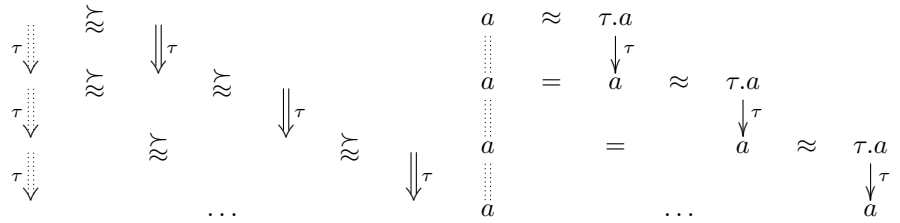
*Proof.* We show easily $\gtrsim\stackrel{\tau}{\Rightarrow} \subseteq \stackrel{\tau}{\Rightarrow}\gtrsim$, so that $\gtrsim\stackrel{\tau}{\Rightarrow}$ terminates by Lemma 2.5. Then we check that $\gtrsim$ and $\mathcal{R}$ satisfy the hypotheses of Theorem 1.7. □

We make some comments about this result and its proof.

We have $\stackrel{\tau}{\Rightarrow} = \stackrel{\tau}{\rightarrow}^+$, so that the $\tau$-termination is actually the termination of $\stackrel{\tau}{\rightarrow}$ (a property called *convergence* in [4]). Without this hypothesis, up to elaboration fails to be correct: in CCS, we have $!\tau|a \gtrsim !\tau|\tau.a$, and hence the (symmetric) relation $\mathcal{R} = \{\langle !\tau|\tau.a, 0\rangle; \langle 0, !\tau|\tau.a\rangle\}$ evolves to $\gtrsim \mathcal{R}$, but $\mathcal{R} \not\subseteq \approx$. We show in Sect. 4 how to relax the $\tau$-termination requirement in some cases.

Theorem 2.6 is an application of the results proved in [7] – summed up in Theorem 1.7 – that exploit the termination of *ladders* (that is, sequences of processes related by $\mathcal{B}^+\stackrel{\tau}{\Rightarrow}$). Remarkably, we are able to require here a termination property that does no longer involve the relation of interest ($\gtrsim$). This is achieved by using the right-to-left part of the elaboration game: as shown in the proof of Lemma 2.5, and depicted on the left diagram below, we use this part of the

elaboration game in order to transform any infinite ladder into an infinite sequence of $\tau$-transitions, that would contradict the $\tau$-termination hypothesis. By contrast, when considering $\approx$ instead of $\gtrsim$, the same argument does not hold, as shown on the right diagram, which recasts the counterexample seen in the introduction: in a bisimulation game, the left hand side process is allowed not to move and hence an infinite ladder may yield a finite sequence of $\tau$-moves.




We can moreover remark that Lemma 2.5 actually entails that $\gtrsim$ can be used in the general setting proposed in [7] (it is a *controlled relation* – cf. [7]). In particular, in systems where $\gtrsim$ is a precongruence, up to elaboration can be combined with the 'up to context' technique, yielding a powerful tool for bisimulation proofs.

## 3 Up-to Techniques for Elaboration

We now present some techniques that can be used to establish elaboration results, which in turn can be used for bisimulation proofs, by Theorem 2.6. We develop a theory of up-to techniques for elaboration along the lines of the study of up-to techniques for strong bisimulation in [9].

**Definition 3.1 (Progression).** *Let $\mathcal{R}, \mathcal{S}$ be two relations. We say that $\mathcal{R}$* progresses *to $\mathcal{S}$, denoted by $\mathcal{R} \rightsquigarrow \mathcal{S}$, if whenever $P\mathcal{R}Q$,*

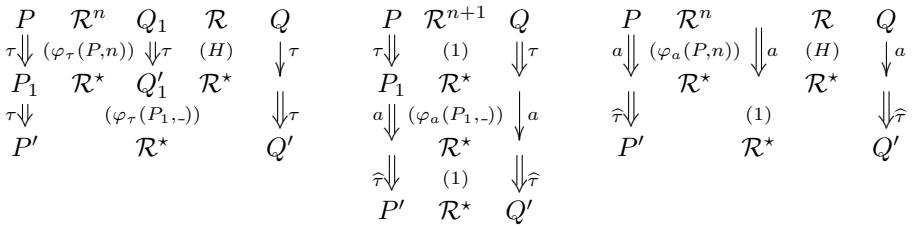
- *if $P \xrightarrow{\alpha} P'$, then $Q \stackrel{\widehat{\alpha}}{\Rightarrow} Q'$ with $P'\mathcal{S}Q'$,*
- *if $Q \xrightarrow{\alpha} Q'$, then $P \stackrel{\alpha}{\Rightarrow} P'$ with $P'\mathcal{S}Q'$.*

This notion of progression is the counterpart of evolution (Definition 1.3) where an 'elaboration game' replaces the 'simulation game'. In particular, $\mathcal{R}$ is an elaboration iff $\mathcal{R}$ progresses to itself.

First we show that like strong bisimilarity, elaboration validates the powerful up to transitivity technique. As a corollary, elaboration up to elaboration is a correct technique: this means in particular that the elaboration preorder does not suffer from the irregularities of weak bisimilarity.

**Theorem 3.2 (Elaboration up to transitivity).** *In a $\tau$-terminating LTS, if $\mathcal{R}$ is a relation that progresses to $\mathcal{R}^\star$, then $\mathcal{R}$ is contained in elaboration.*

*Proof.* We show that $\mathcal{R}^\star$ is an elaboration relation. For $\alpha \in \mathcal{L}$, let $\varphi_\alpha(P,n)$ denote the predicate: "for any $Q'$ such that $P\mathcal{R}^n \stackrel{\alpha}{\Rightarrow} Q'$, $P \stackrel{\alpha}{\Rightarrow} \mathcal{R}^\star Q'$". We prove $\mathcal{R}^\star \stackrel{\tau}{\Rightarrow} \subseteq \stackrel{\tau}{\Rightarrow} \mathcal{R}^\star$ (1) by a lexicographic induction based on the termination of $\stackrel{\tau}{\Rightarrow}$, with the predicate $\varphi_\tau$. The argument for the non-trivial case is sketched on the left diagram below:




Then we prove $\mathcal{R}^\star \stackrel{a}{\Rightarrow} \subseteq \stackrel{a}{\Rightarrow} \mathcal{R}^\star$ (2) by a second lexicographic induction with the predicate $\varphi_a$. The two diagrams on the right above give the interesting cases. Finally, by applying Lemma 2.5 to $\mathcal{R}^\star$ and (1), we obtain the termination of $\mathcal{R}^\star \stackrel{\tau}{\Rightarrow}$, that leads to $\stackrel{\alpha}{\Leftarrow} \mathcal{R}^\star \subseteq \mathcal{R}^\star \stackrel{\widehat{\alpha}}{\Leftarrow}$ using [7, Theorem 3.12]. □

We now introduce a class of functions corresponding to correct up-to techniques, that enjoys nice compositional properties.

**Definition 3.3 (Safe function).** *A function $\mathcal{F}$ is* safe *if for any relations $\mathcal{R}$ and $\mathcal{S}$,*

$$\text{if } \begin{cases} \mathcal{R} \subseteq \mathcal{S} \\ \mathcal{R} \rightsquigarrow \mathcal{S}^\star \end{cases} \quad \text{then } \begin{cases} \mathcal{F}(\mathcal{R}) \subseteq \mathcal{F}(\mathcal{S}) \\ \mathcal{F}(\mathcal{R}) \rightsquigarrow \mathcal{F}(\mathcal{S})^\star \end{cases}$$

This definition corresponds to [9, Definition 2.5]. The main difference is that we consider progressions to the reflexive transitive closures of relations. As shown in the following theorem, using Theorem 3.2, this makes it possible to use safe functions 'up to transitivity'.

**Theorem 3.4 (Correctness of safe functions).** *Let $\mathcal{F}$ be a safe function. In a $\tau$-terminating LTS, if a relation $\mathcal{R}$ progresses to $\mathcal{F}(\mathcal{R})^\star$, then it is contained in elaboration.*

*Proof.* Let $\mathcal{R}_0 = \mathcal{R}$, $\mathcal{R}_{n+1} = \mathcal{R}_n \cup \mathcal{F}(\mathcal{R}_n)$, $\mathcal{R}_\omega = \bigcup_n \mathcal{R}_n$. We show by induction $\forall n$, $\mathcal{R}_n \rightsquigarrow \mathcal{R}^\star_{n+1}$. Hence $\mathcal{R}_\omega \rightsquigarrow \mathcal{R}^\star_\omega$, and finally $\mathcal{R}_\omega \subseteq \gtrsim$ using Theorem 3.2. □

The main point of safe functions is that they can be combined in a modular way: given two safe functions $\mathcal{F}$ and $\mathcal{G}$, their union $\mathcal{F} \cup \mathcal{G} : \mathcal{R} \mapsto \mathcal{F}(\mathcal{R}) \cup \mathcal{G}(\mathcal{R})$ and their functional composition $\mathcal{F} \circ \mathcal{G} : \mathcal{R} \mapsto \mathcal{F}(\mathcal{G}(\mathcal{R}))$ are safe. Hence, we can define correct up-to techniques incrementally (see for example the proof of Corollary 3.7). By contrast with [9], composing functions using the *chaining operator* $\mathcal{F}^\frown\mathcal{G} : \mathcal{R} \mapsto \mathcal{F}(\mathcal{R})\mathcal{G}(\mathcal{R})$ does not preserve safety, essentially for the same reasons as in the weak bisimilarity case [11] (in particular, $\tau$-termination does not help). However, chaining can be 'emulated' since we are allowed to use safe functions up to transitivity: instead of $\mathcal{F}^\frown\mathcal{G}$, we can work with $(\mathcal{F} \cup \mathcal{G})^\star$, which we believe provides enough flexibility for actual elaboration proofs.

*Elaboration up to context.* We further enrich the set of up-to techniques for elaboration with an up to context technique. We call *context* a mapping from processes to processes (like in [7], we adopt an approach that allows us to abstract over the details of the underlying syntax). We denote by $C[P]$ the application of a context $C$ to a process $P$. In the following technical definition, both $\xrightarrow{\epsilon}$ and $\stackrel{\epsilon}{\Rightarrow}$ are synonyms for the identity relation $\mathcal{I}$ (we suppose $\epsilon \notin \mathcal{L}$).

**Definition 3.5 (Faithfulness).** *Let $\mathcal{C}$ be a family of contexts. We say that $\mathcal{C}$ is* faithful *if for all $C \in \mathcal{C}$, whenever $C[P] \xrightarrow{\alpha} R$, there are $C' \in \mathcal{C}$, $P' \in \mathcal{P}$ and $\delta \in \mathcal{L} \cup \{\epsilon\}$ such that $R = C'[P']$ and $P \xrightarrow{\delta} P'$, and for any $Q, Q'$ such that $Q \stackrel{\delta}{\Rightarrow} Q'$, $C[Q] \stackrel{\alpha}{\Rightarrow} C'[Q']$.*

This is the direct adaptation to the weak case of the notion of faithfulness found in [9]. In CCS *non-degenerate* contexts [11] are faithful; in the $\pi$-calculus, *non-input guarded* contexts are faithful. The following proposition shows that these families of contexts yield correct up-to techniques for elaboration. The proof is very similar to the proof of the corresponding result in [11].

**Proposition 3.6 (Safety of faithful families of contexts).** *Let $\mathcal{C}$ be a faithful family of contexts; the following* closure up to $\mathcal{C}$ *function is safe:*

$$\widetilde{\mathcal{C}} : \mathcal{R} \mapsto \{\langle C[P], C[Q]\rangle \ / \ C \in \mathcal{C} \text{ and } P\mathcal{R}Q\} \ .$$

The following corollary sums up all previous results, yielding a powerful up-to technique for elaboration. It appears that the theory of up-to techniques for elaboration is as smooth as that for strong bisimilarity. Also notice that while we considered only *monadic* contexts in Prop. 3.6, Theorem 3.4 allows us to use $\widetilde{\mathcal{C}}$ transitively, thus validating the up to *polyadic* contexts technique.

**Corollary 3.7 (Elaboration up to context and transitivity).** *Let $\mathcal{C}$ be a faithful family of contexts and $\mathcal{R}$ a relation. If $\xrightarrow{\tau}$ terminates and $\mathcal{R}$ progresses to $(\widetilde{\mathcal{C}}(\mathcal{R}) \cup \gtrsim)^\star$, then $\mathcal{R}$ is contained in elaboration.*

*Proof.* The functions $\mathcal{R} \mapsto \gtrsim$ and $\widetilde{\mathcal{C}}$ are safe, hence so is $\mathcal{R} \mapsto \mathcal{C}(\mathcal{R}) \cup \gtrsim$. □

*Up to deterministic transitions.* Let us finally mention a corollary of Theorem 3.2, that extends a technique which has been introduced in [3, Chap. 4] in the setting of barbed bisimilarity. Together with Theorem 2.6, this result gives the possibility, when $\xrightarrow{\tau}$ is terminating and deterministic, to normalise processes w.r.t. $\xrightarrow{\tau}$ along a bisimulation proof. Notice that [3] does not suppose $\tau$-termination, but requires the stronger commutation hypothesis $\stackrel{\alpha}{\leftarrow}\stackrel{\tau}{\rightarrow} \subseteq \stackrel{\widehat{\tau}}{\rightarrow}\stackrel{\widehat{\alpha}}{\leftarrow}$.

**Corollary 3.8.** *If $\xrightarrow{\tau}$ terminates and for all $\alpha \in \mathcal{L}$, $\stackrel{\alpha}{\leftarrow}\stackrel{\tau}{\rightarrow} \subseteq \stackrel{\widehat{\tau}}{\Rightarrow}\stackrel{\widehat{\alpha}}{\Leftarrow}$, then $\xrightarrow{\tau} \subseteq \gtrsim$.*

*Proof.* We remark that $\xrightarrow{\tau}\xrightarrow{\alpha} \subseteq \stackrel{\alpha}{\Rightarrow} \subseteq \stackrel{\alpha}{\Rightarrow}\stackrel{\widehat{\tau}}{\Rightarrow}$, so that relation $\xrightarrow{\tau}$ satisfies the requirements of Theorem 3.2, and hence is an elaboration up to transitivity. □

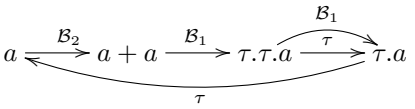
**On Modularity Properties of Up-to Techniques.** Introducing the up to elaboration proof technique enriches the existing landscape of up-to techniques for bisimulation. We have seen that this behavioural preorder enjoys nice properties, allowing one to develop elaboration proofs in an incremental and modular fashion. We now study other up-to techniques from this point of view.

*On the use of compression.* As shown in [3,7], compression also yields a correct up-to technique. By Proposition 2.3 above, elaboration and compression are not comparable. The following example in CCS shows that they are neither compatible, in the sense that they cannot be used in the same bisimulation proof. Let $P = \tau.\tau.a$ and $Q = \tau.(\tau.\tau|a)$; we have $P \xrightarrow{\tau} \tau.a \succcurlyeq Q \xrightarrow{\tau} \tau.\tau|a \gtrapprox P$ so that the symmetric relation $\mathcal{R} = \{\langle P, 0\rangle; \langle Q, 0\rangle; \langle 0, P\rangle; \langle 0, Q\rangle\}$ evolves to $(\succcurlyeq \cup \gtrapprox)\mathcal{R}$, but obviously $\mathcal{R} \not\subseteq \approx$.

Another observation we can make about compression is that unlike elaboration, compression result cannot be proved up to transitivity, even when the LTS is $\tau$-terminating. Indeed, the relation $\{\langle 0, \tau.a\rangle; \langle \tau.a, a\rangle; \langle 0, 0\rangle; \langle a, a\rangle\}$ over finite CCS processes is a 'compression up to transitivity', but it is clearly not contained in bisimilarity, and thus neither in compression.

*Incrementality in the setting of [7].* Stability by union for up-to techniques provides a form of modularity, since it allows one to extend an existing proof by simply adding new behavioural laws. This property is immediate for coinductively defined relations such as $\gtrapprox$, $\gtrsim$ or $\succcurlyeq$. On the contrary, the setting of [7] lacks this facility: in order to extend a bisimulation proof up to $\mathcal{B}_1^*$ using a relation $\mathcal{B}_2$ ($\mathcal{B}_1$ and $\mathcal{B}_2$ are supposed to satisfy the hypotheses of Theorem 1.7), one needs to prove the termination of $(\mathcal{B}_1 \cup \mathcal{B}_2)^+ \stackrel{\tau}{\Rightarrow}$, which involves some knowledge about $\mathcal{B}_1$. To illustrate the difficulties, consider the following example in CCS:

$$\mathcal{B}_1 = \{(a + a, \tau.\tau.a), (\tau.\tau.a, \tau.a)\} \qquad \mathcal{B}_2 = \{(a, a + a)\}$$




These relations satisfy the required property: for $i \in \{1, 2\}$, $\mathcal{B}_i$ evolves to $\mathcal{B}_i^\star$ and $\mathcal{B}_i^+ \stackrel{\tau}{\Rightarrow}$ terminates. But $(\mathcal{B}_1 \cup \mathcal{B}_2)^+$ contains the pair $\langle a, \tau.a\rangle$, and hence $\mathcal{B}_1 \cup \mathcal{B}_2$ does not qualify to apply Theorem 1.7. We return to this question in Sect. 5.

## 4 The Case of Non-Terminating Systems

We now show how the results from the two previous sections can be adapted to cases where the $\tau$-termination assumption is not satisfied. Before moving to the formal definitions, we make a few remarks on the $\tau$-termination requirement. It should be noticed that for the up to elaboration technique to be applicable, *the whole LTS* does not necessarily need to be $\tau$-terminating. What we need is rather a transition closed subset of (pairs of) processes for which this condition holds. For instance, we might want to represent a system in CCS, a calculus where divergences are of course expressible, but the processes used for the modelling do not exhibit $\tau$-divergences.

If, on the contrary, the system we would like to reason about does contain divergences, a first approach could be to 'tag' non-terminating silent moves and treat these as visible. However, such visible transitions must be mapped to some visible actions on the other side of the elaboration game, in order to play these in one-to-one correspondence. This of course might be too demanding in some cases, typically when divergences arise because implementing a given behaviour introduces some loops (that are not present in the original specification). In order to address such situations, we adopt an approach from [4], which consists in isolating a subset of the $\tau$ transitions that are terminating, while still treating all $\tau$ moves as silent.

In the following we consider a LTS where silent moves are split into two special actions: $\{\tau_>, \tau_=\} \subseteq \mathcal{L}$. Transitions $\xrightarrow{\tau_>}$ and $\xrightarrow{\tau_=}$ will respectively be called *progressive* and *non-progressive* silent transitions. *Silent transitions*, written $\xrightarrow{\tau}$, are *defined* by $\xrightarrow{\tau} \triangleq \xrightarrow{\tau_>} \cup \xrightarrow{\tau_=}$. Coherently, $a, b$ will range over $\mathcal{L} \setminus \{\tau_>, \tau_=\}$. We recall our notations for weak transitions (Definition 1.2) below.

$$\stackrel{\widehat{\tau}}{\Rightarrow} = \xrightarrow{\tau}{}^{\star} \qquad \stackrel{\widehat{a}}{\Rightarrow} = \stackrel{a}{\Rightarrow} = \xrightarrow{\tau}{}^{\star} \xrightarrow{a} \xrightarrow{\tau}{}^{\star} \qquad \stackrel{\tau_>}{\Rightarrow} = \xrightarrow{\tau}{}^{\star} \xrightarrow{\tau_>} \xrightarrow{\tau}{}^{\star}$$

In this setting the notions of bisimulation and bisimilarity ignore the distinction between the two kinds of silent transitions (in particular, these relations do not coincide with what we would obtain by treating $\tau_=$ as visible actions). The definition of elaboration is adapted so as to control progressive transitions only:

**Definition 4.1 ($\tau_>$-Elaboration).** $\tau_>$-Elaboration*, denoted by* $\gtrsim^{>}_{\approx}$*, is the largest relation such that whenever* $P \gtrsim^{>}_{\approx} Q$*,*

*(i) if* $P \xrightarrow{\alpha} P'$ *then* $Q \stackrel{\widehat{\alpha}}{\Rightarrow} Q'$ *with* $P' \gtrsim^{>}_{\approx} Q'$*, for any* $\alpha \in \mathcal{L}$*,*
*(ii) if* $Q \xrightarrow{\alpha} Q'$ *then* $P \stackrel{\widehat{\alpha}}{\Rightarrow} P'$ *with* $P' \gtrsim^{>}_{\approx} Q'$*, for any* $\alpha \neq \tau_>$*,*
*(iii) if* $Q \xrightarrow{\tau_>} Q'$ *then* $P \stackrel{\tau_>}{\Rightarrow} P'$ *with* $P' \gtrsim^{>}_{\approx} Q'$*.*

$\tau_>$-expansion is the 'progressive elaboration' we alluded to in the introduction. Clause $(i)$ corresponds to bisimulation, while when playing from right to left, we ensure that progressive silent transitions are 'preserved' $(iii)$. We can easily check that $\gtrsim^{>}_{\approx}$ is a preorder, and that we have $\sim \subset \gtrsim^{>}_{\approx} \subset \approx$.

This adaptation leads to the following theorem, where the termination hypothesis concerns progressive silent transitions. As expected, up to transitivity is allowed on visible transitions $(i)$, and up to $\tau_>$-elaboration is supported only on progressive silent transitions $(ii)$. Clause $(iii)$ for non-progressive transitions does not allow up-to reasoning on the left of $\mathcal{R}$. We show in [8] how to relax this condition by using an adapted version of expansion. We omit this development here for the sake of simplicity.

**Theorem 4.2 (Bisimulation up to $\tau_>$-Elaboration).** *Let* $\mathcal{R}$ *be a symmetric relation. If the following conditions hold whenever* $P\mathcal{R}Q$*:*

*(i) if* $P \stackrel{a}{\rightarrow} P'$ *then* $Q \stackrel{a}{\Rightarrow} Q'$ *with* $P'\mathcal{R}^{\star}Q'$*,*
*(ii) if* $P \stackrel{\tau_>}{\rightarrow} P'$ *then* $Q \stackrel{\widehat{\tau}}{\Rightarrow} Q'$ *with* $P' \gtrsim^{>}_{\approx} \mathcal{R} \approx Q'$*, and*

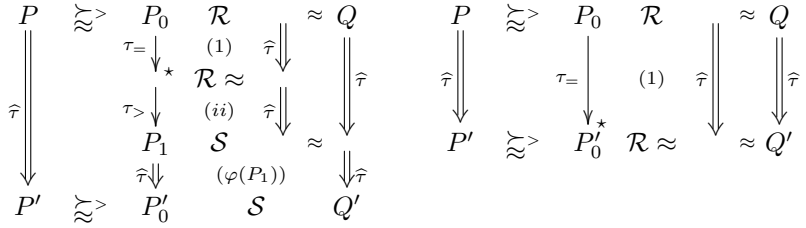
*(iii) if* $P \stackrel{\tau_=}{\Rightarrow} P'$ *then* $Q \stackrel{\widehat{\tau}}{\Rightarrow} Q'$ *with* $P'\mathcal{R} \approx Q'$,

*and the LTS is* $\tau_>$*-terminating then* $\mathcal{R}$ *is contained in bisimilarity.*

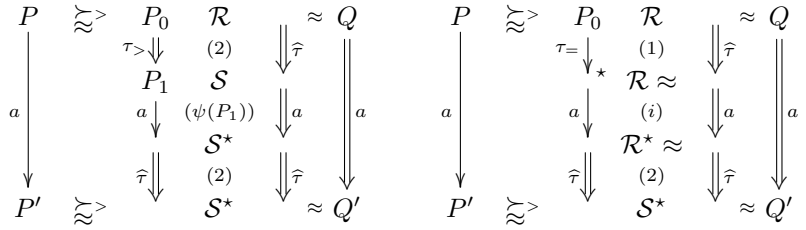
*Proof.* We first prove the termination of $\gtrsim^{>} \stackrel{\tau_>}{\Rightarrow}$ using Lemma 2.5. Then we show that the symmetric relation $(\mathcal{R} \cup \approx)^\star$ is a bisimulation. Let $\mathcal{S} = \gtrsim^{>} \mathcal{R} \approx$; we remark that $(\mathcal{R} \cup \approx)^\star = \approx \mathcal{S}^\star$, so that it is sufficient to show that $\mathcal{S}^\star$ evolves to itself. This is established by proving successively the following inclusions:

$$(1) \quad \stackrel{\tau_=}{\Leftarrow}{}^\star \mathcal{R} \subseteq \mathcal{R} \approx \stackrel{\widehat{\tau}}{\Leftarrow} \qquad (2) \quad \stackrel{\tau}{\Leftarrow} \mathcal{S} \subseteq \mathcal{S} \stackrel{\widehat{\tau}}{\Leftarrow} \qquad (3) \quad \stackrel{a}{\leftarrow} \mathcal{S} \subseteq \mathcal{S}^\star \stackrel{a}{\Leftarrow}$$

We obtain (1) from (*iii*) and a simple induction over the sequence $\stackrel{\tau_=}{\Rightarrow}{}^\star$. We prove (2) by well-founded induction using the termination of $\gtrsim^{>} \stackrel{\tau_>}{\Rightarrow}$ and the predicate $\varphi(P)$: "for any $P', Q$ such that $P \stackrel{\widehat{\tau}}{\Rightarrow} P'$ and $P\mathcal{S}Q$, we have $P'\mathcal{S} \stackrel{\widehat{\tau}}{\Leftarrow} Q$". This leads to the diagrams below, where we reason by cases according to whether there is a progressive silent transition between $P_0$ and $P_0'$ or not. In the former case, $P \gtrsim^{>} \stackrel{\tau}{\Rightarrow} P_1$ so that $\varphi(P_1)$ holds. Otherwise, we just use (1).




Then we prove (3) by well-founded induction using the termination of $\gtrsim^{>} \stackrel{\tau_>}{\Rightarrow}$ and the predicate $\psi(P)$: "for any $P', Q$ such that $P \stackrel{a}{\rightarrow} P'$ and $P\mathcal{S}Q$, we have $P'\mathcal{S}^\star \stackrel{a}{\Leftarrow} Q$". As depicted in the following diagrams, if there is a progressive silent transition transition between $P_0$ and $P_1$, we use the induction hypothesis, otherwise, (1) is sufficient to close the diagram.




Finally, we apply Lemma 1.5 with (2) and (3) so that $\mathcal{S}^\star$ evolves to itself. □

We now show that $\tau_>$-elaboration validates the powerful up to transitivity proof technique on visible and progressive silent actions.

**Theorem 4.3 ($\tau_>$-Elaboration up to Transitivity).** *Let* $\mathcal{R}$ *be a relation. If the following conditions hold whenever* $P\mathcal{R}Q$*:*

*(i) if* $P \stackrel{\alpha}{\rightarrow} P'$ *then* $Q \stackrel{\widehat{\alpha}}{\Rightarrow} Q'$ *with* $P'\mathcal{R}^\star Q'$*, for any* $\alpha \neq \tau_=$,

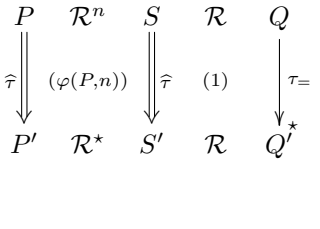
*(ii)* *if* $P \stackrel{\tau_=}{\to} P'$ *then* $Q \stackrel{\widehat{\tau}}{\Rightarrow} Q'$ *with* $P'\mathcal{R}Q'$,
*(iii)* *if* $Q \stackrel{\alpha}{\to} Q'$ *then* $P \stackrel{\alpha}{\Rightarrow} P'$ *with* $P'\mathcal{R}^\star Q'$, *for any* $\alpha \neq \tau_=$, *and*
*(iv)* *if* $Q \stackrel{\tau_=}{\to} Q'$ *then* $P \stackrel{\widehat{\tau}}{\Rightarrow} P'$ *with* $P'\mathcal{R}Q'$,

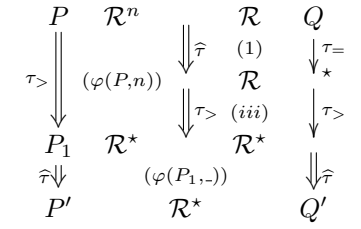
*and the LTS is* $\tau_>$*-terminating then* $\mathcal{R}$ *is contained in* $\tau_>$*-elaboration.*

*Proof.* We show that $\mathcal{R}^\star$ is a $\tau_>$-elaboration relation by successively establishing the following properties.

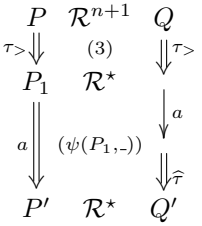
$$\mathcal{R} \stackrel{\tau_=}{\to}{}^\star \subseteq \stackrel{\widehat{\tau}}{\Rightarrow} \mathcal{R} \qquad (1)$$
$$\mathcal{R}^\star \stackrel{\tau_=}{\to}{}^\star \subseteq \stackrel{\widehat{\tau}}{\Rightarrow} \mathcal{R}^\star \qquad (2)$$
$$\mathcal{R}^\star \stackrel{\tau_>}{\Rightarrow} \subseteq \stackrel{\tau_>}{\Rightarrow} \mathcal{R}^\star \qquad (3)$$
$$\mathcal{R}^\star \stackrel{a}{\Rightarrow} \subseteq \stackrel{a}{\Rightarrow} \mathcal{R}^\star \qquad (4)$$
$$\mathcal{R}^\star \stackrel{\tau_>}{\Rightarrow} \text{ terminates} \qquad (5)$$
$$\stackrel{\tau_=}{\Leftarrow}{}^\star \mathcal{R} \subseteq \mathcal{R} \stackrel{\widehat{\tau}}{\Leftarrow} \qquad (6)$$
$$\stackrel{\tau}{\Leftarrow} \mathcal{R}^\star \subseteq \mathcal{R}^\star \stackrel{\widehat{\tau}}{\Leftarrow} \qquad (7)$$
$$\stackrel{a}{\Leftarrow} \mathcal{R}^\star \subseteq \mathcal{R}^\star \stackrel{\widehat{a}}{\Leftarrow} \qquad (8)$$

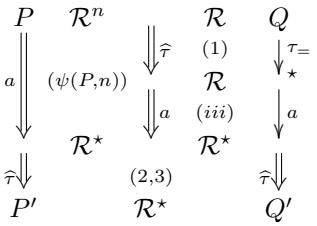
A simple induction and $(iv)$ yields (1), we prove (2) and (3) simultaneously by a lexicographic induction, using the termination of $\stackrel{\tau}{\Rightarrow}$ and $\varphi(P,n)$: "for any $Q, Q'$ such that $P\mathcal{R}^nQ$, if $Q \stackrel{\tau_=}{\to}{}^\star Q'$ then $P \stackrel{\widehat{\tau}}{\Rightarrow} \mathcal{R}^\star Q'$, and if $Q \stackrel{\tau_>}{\Rightarrow} Q'$ then $P \stackrel{\tau_>}{\Rightarrow} \mathcal{R}^\star Q'$". The non-trivial cases are respectively depicted on the two diagrams below.



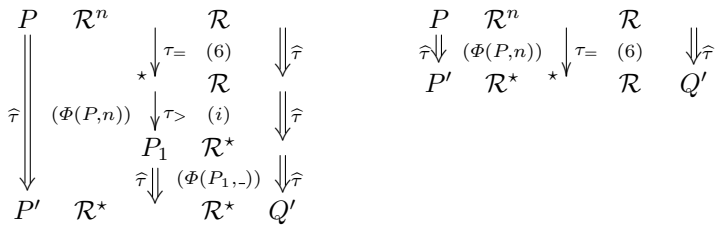




We prove (4) with another lexicographic induction, with the predicate $\psi(P,n)$: "for any $Q'$ if $P\mathcal{R}^n \stackrel{a}{\Rightarrow} Q'$ then $P \stackrel{a}{\Rightarrow} \mathcal{R}^\star Q'$". Depending on the existence of a progressive silent transition before the visible action of the transition $Q \stackrel{a}{\Rightarrow} Q'$, we close the diagrams as depicted below.




We obtain (5) by applying Lemma 2.5 to $\mathcal{R}^\star$ and (3). A simple induction and $(ii)$ give (6). We show (7) with a lexicographic induction using the termination of $\mathcal{R}^\star \stackrel{\tau_>}{\Rightarrow}$ and the predicate $\Phi(P,n)$: "if $P\mathcal{R}^nQ$ and $P \stackrel{\alpha}{\Rightarrow} P'$ then $P'\mathcal{R}^\star \stackrel{\widehat{\alpha}}{\Leftarrow} Q$".




The proof of (8) follows the lines of (7). □

## 5 Concluding Remarks

We have proposed the new up to elaboration proof technique for bisimulation as an alternative to existing approaches. The proofs in this paper demonstrate how nontrivial termination arguments can be used to validate sophisticated proof techniques for bisimulation.

We have argued that up to elaboration offers advantages with respect to existing up-to techniques, in terms of expressiveness, flexibility or modularity. Our hope is that this technique can help addressing more complex weak bisimulation proofs. That it could be the case is suggested by the mathematical elegance of the framework we obtain, which opens the way for modular and incremental construction of proofs. This should nevertheless be confirmed by actual experiments in the study of systems involving manipulation of large bisimulation relations.

Several results in this paper suggest directions for future investigations. To enhance further our framework, it would be interesting to study how to integrate different kinds of methods in order to guarantee $\tau$-termination, which is necessary for the results in Sect. 2. A possible approach would be to provide a measure together with the LTS, or to adopt syntactical criteria when the LTS is given by a calculus (a process algebra). Another interesting idea in this direction is given by type systems for termination. In Sect. 4, we proposed a way to handle the case of non terminating systems. We can however think of other approaches; in particular, we would like to study LTS where non-termination of $\xrightarrow{\tau}$ comes from cycles only, or where any state has a finite number of derivatives.

Finally, we would like to have a better understanding of the main problem of the setting of [7] (to which this paper proposes an alternative solution), namely the fact that controlled relations are not stable by union. An interesting direction would be to look for connections with the question of termination of the union of terminating rewrite systems, that has been widely studied in rewriting theory.

*Acknowledgements.* We are very thankful to Daniel Hirschkoff for his numerous comments and suggestions, and his great help during the redaction process. We would also like to thank an anonymous referee for pointing out an incorrect proof.

## References


1. S. Arun-Kumar and M. Hennessy. An Efficiency Preorder for Processes. *Acta Informatica*, 29(9):737–760, 1992.
2. S. Arun-Kumar and V. Natarajan. Conformance: A Precongruence Close to Bisimilarity. In *Proc. Struct. in Concurrency Theory*, pages 55–68. Springer Verlag, 1995.
3. C. Fournet. *The Join-Calculus: a Calculus for Distributed Mobile Programming*. PhD thesis, Ecole Polytechnique, 1998.
4. J. Groote and M. Reniers. Algebraic Process Verification. In *Handbook of Process Algebra*, pages 1151–1208. Elsevier, 2001.
5. D. Hirschkoff, D. Pous, and D. Sangiorgi. A Correct Abstract Machine for Safe Ambients. In *Proc. COORD '05*, volume 3454 of *LNCS*. Springer Verlag, 2005.
6. U. Montanari and V. Sassone. Dynamic Congruence vs. Progressing Bisimulation for CCS. *Fundamenta Informaticae*, 16(1):171–199, 1992.
7. D. Pous. Up-to Techniques for Weak Bisimulation. In *Proc. 32th ICALP*, volume 3580 of *LNCS*, pages 730–741. Springer Verlag, 2005.
8. D. Pous. Weak Bisimulation up to Elaboration. Long version of this paper, with full proofs – available from `http://perso.ens-lyon.fr/damien.pous/upto`, 2006.
9. D. Sangiorgi. On the Bisimulation Proof Method. *Journal of Mathematical Structures in Computer Science*, 8:447–479, 1998.
10. D. Sangiorgi and R. Milner. The problem of "Weak Bisimulation up to". In *Proc. 3rd CONCUR*, volume 630 of *LNCS*, pages 32–46. Springer Verlag, 1992.
11. D. Sangiorgi and D. Walker. *The $\pi$-calculus: a Theory of Mobile Processes*. Cambridge University Press, 2001.