Theoretical Computer Science 612 (2016) 29–44

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Unique parallel decomposition in branching and weak bisimulation semantics

Bas Luttik

ARTICLE INFO

Article history:
Received 4 March 2015
Received in revised form 23 July 2015
Accepted 9 October 2015
Available online 21 October 2015
Communicated by R. van Glabbeek

Keywords:
Unique parallel decomposition
Branching bisimilarity
Weak bisimilarity
Process calculus

ABSTRACT

We consider the property of unique parallel decomposition modulo branching and weak bisimilarity. First, we show that normed behaviours always have parallel decompositions, but that these are not necessarily unique. Then, we establish that finite behaviours have unique parallel decompositions. We derive the latter result from a general theorem about unique decompositions in partial commutative monoids.

1. Introduction

A recurring question in process theory is to what extent the behaviours definable in a certain process calculus admit a unique decomposition into indecomposable parallel components. Milner and Moller [22] were the first to address the question. They proved a unique parallel decomposition theorem for a simple process calculus, which allows the specification of finite behaviour up to strong bisimilarity and includes parallel composition in the form of pure interleaving without interaction between the components. They also presented counterexamples showing that unique parallel decomposition may fail in process calculi in which it is possible to specify infinite behaviour, or in which certain coarser notions of behavioural equivalence are used.

Moller proved several more unique parallel decomposition results in his dissertation [23], replacing interleaving parallel composition by CCS parallel composition, and then also considering weak bisimilarity. These results were established with subsequent refinements of an ingenious proof technique attributed to Milner. Christensen, in his dissertation [7], further refined the proof technique to make it work for the *normed* behaviours recursively definable modulo strong bisimilarity, and for *all* behaviours recursively definable modulo distributed bisimilarity.

With each successive refinement of Milner's proof technique, the technical details became more complicated, but the general idea of the proof remained the same. In [18] we made an attempt to isolate the deep insights from the technical details, by identifying a sufficient condition on partial commutative monoids that facilitates an abstract version of Milner's proof technique. To concisely present the sufficient condition, we have put forward the notion of *decomposition order*; it is established in [18], by means of an abstract version of Milner's technique, that if a partial commutative monoid can be endowed with a decomposition order, then it has unique decomposition.

Application of the general result of [18] in commutative monoids of behaviour is often straightforward: a well-founded order naturally induced on behaviour by (a terminating fragment of) the transition relation typically satisfies the properties of a decomposition order. All the aforementioned unique parallel decomposition results can be directly obtained in this

E-mail address: s.p.luttik@tue.nl.

http://dx.doi.org/10.1016/j.tcs.2015.10.013
0304-3975/

way, except Moller's result that finite behaviours modulo weak bisimilarity have unique decomposition. It turns out that a decomposition order cannot straightforwardly be obtained from the transition relation if certain transitions are deemed unobservable by the behavioural equivalence under consideration.

In this article, we address the question of how to establish unique parallel decomposition in settings with a notion of unobservable behaviour. Our main contribution will be an adaptation of the general result in [18] to make it suitable for establishing unique parallel decomposition also in settings with a notion of unobservable behaviour. To illustrate the result, we shall apply it to establish unique parallel decomposition for finite behaviour modulo branching or weak bisimilarity. We shall also show, by means of a counterexample, that unique parallel decomposition fails for infinite behaviours modulo branching and weak bisimilarity, even if only a very limited form of infinite behaviour is considered (normed behaviour definable in a process calculus with prefix iteration).

A positive answer to the unique parallel decomposition question seems to be primarily of theoretical interest, as a tool for proving other theoretical properties about process calculi. For instance, Moller's proofs in [24,25] that PA and CCS cannot be finitely axiomatised without auxiliary operations and Hirshfeld and Jerrum's proof in [16] that bisimilarity is decidable for normed PA both rely on unique parallel decomposition. When parallel composition cannot be eliminated from terms by means of axioms, then unique parallel decomposition is generally used to find appropriate normal forms in completeness proofs for equational axiomatisations [1–3,11,15]. In [17], a unique parallel decomposition result serves as a stepping stone for proving complete axiomatisation and decidability results in the context of a higher-order process calculus.

There is an intimate relationship between unique parallel decomposition and of cancellation with respect to parallel composition; the properties are in most circumstances equivalent. In [6], cancellation with respect parallel composition was first proved and exploited to prove the completeness of an axiomatisation of distributed bisimilarity.

Unique parallel decomposition could be of practical interest too, e.g., to devise methods for finding the maximally parallel implementation of a behaviour [8], or for improving verification methods [14]. In [10], unique parallel decomposition results are established for the Applied π-calculus, as a tool in the comparison of different security notions in the context of electronic voting.

This article is organised as follows. In Section 2 we introduce the process calculus that we shall use to illustrate our theory of unique decomposition. There, we also present counterexamples to the effect that infinite behaviours in general may not have a decomposition, and normed behaviours may have more than one decomposition. In Section 3 we recap the theory of decomposition put forward in [18] and discuss why it is not readily applicable to establish unique parallel decomposition for finite behaviours modulo branching and weak bisimilarity. In Section 4 we adapt the theory of [18] to make it suitable for proving unique parallel decomposition results in process calculi with a notion of unobservability. In Section 5 we apply the theorem from Section 4, showing that bounded behaviours have a unique parallel decomposition both modulo branching and weak bisimilarity. We end the article in Section 6 with a short conclusion.

An extended abstract of this article appeared as [19].

2. Processes up to branching and weak bisimilarity

We define a simple language of process expressions together with an operational semantics, and notions of branching and weak bisimilarity. We shall then investigate to what extent process expressions modulo branching or weak bisimilarity admit parallel decompositions. We shall present examples of process expressions without a decomposition, and of normed process expressions with two distinct decompositions.

Syntax We fix a set $\mathcal{A}$ of *actions*, and declare a special action τ that we assume is not in $\mathcal{A}$. We denote by $\mathcal{A}_\tau$ the set $\mathcal{A} \cup \{\tau\}$, and we let a range over $\mathcal{A}$ and α over $\mathcal{A}_\tau$. The set $\mathcal{P}$ of *process expressions* is generated by the following grammar:

$$P ::= \mathbf{0} \mid \alpha.P \mid P + P \mid P \parallel P \mid \alpha^* P \quad (\alpha \in \mathcal{A}_\tau).$$

The language above is BCCS (the core of Milner's CCS [20]) extended with a construction $\_ \parallel \_$ to express interleaving parallelism and the prefix iteration construction $\alpha^*\_$ to specify a restricted form of infinite behaviour. We include only a very basic notion of parallel composition in our calculus, but note that this is just to simplify the presentation. Our unique decomposition theory extends straightforwardly to more intricate notions of parallel composition, e.g., modelling some form of communication between components. To be able to omit some parentheses when writing process expressions, we adopt the conventions that $\alpha.$ and α^* bind stronger, and that $+$ binds weaker than all the other operations.

Operational semantics and branching and weak bisimilarity We define on $\mathcal{P}$ binary relations $\xrightarrow{\alpha}$ $(\alpha \in \mathcal{A}_\tau)$ by means of the operational rules in Table 1. We denote by $\twoheadrightarrow$ the reflexive–transitive closure of $\xrightarrow{\tau}$, i.e., $P \twoheadrightarrow P'$ if there exist $P_0, \ldots, P_n$ $(n \geq 0)$ such that $P = P_0 \xrightarrow{\tau} \cdots \xrightarrow{\tau} P_n = P'$. Furthermore, we shall write $P \xrightarrow{(\alpha)} P'$ if $P \xrightarrow{\alpha} P'$ or $\alpha = \tau$ and $P = P'$.

Definition 1 *(Branching bisimilarity [13]).* A symmetric binary relation $\mathcal{R}$ on $\mathcal{P}$ is a *branching bisimulation* if for all $P, Q \in \mathcal{P}$ such that $P \mathrel{\mathcal{R}} Q$ and for all $\alpha \in \mathcal{A}_\tau$ it holds that

if $P \xrightarrow{\alpha} P'$ for some $P' \in \mathcal{P}$, then there exist $Q'', Q' \in \mathcal{P}$ such that $Q \twoheadrightarrow Q'' \xrightarrow{(\alpha)} Q'$ and $P \mathrel{\mathcal{R}} Q''$ and $P' \mathrel{\mathcal{R}} Q'$.