Timing the Untimed: Terminating Successfully While Being Conservative

J.C.M. Baeten, M.R. Mousavi, and M.A. Reniers

Department of Computer Science,
Eindhoven University of Technology (TU/e),
NL-5600 MB Eindhoven, The Netherlands

Abstract. There have been several timed extensions of ACP-style process algebras with successful termination. None of them, to our knowledge, are equationally conservative (ground-)extensions of ACP with successful termination. Here, we point out some design decisions which were the possible causes of this misfortune and by taking different decisions, we propose a spectrum of timed process algebras ordered by equational conservativity ordering.

1 The Untimed Past

The term "process algebra" was coined by Jan Bergstra and Jan Willem Klop in [8] to denote an algebraic approach to concurrency theory. Their process algebra had uniform atomic actions a_i for $i \in I$ (with I some index set), sequential composition $\_ \cdot \_$, choice (alternative composition) $\_ + \_$ and left merge $\_ \mathbin{⌊\!⌊} \_$ as the basic composition operators.[1]

Much of the core theory of [8] remained intact in the course of more than 20 years of developments in the ACP-school (for Algebra of Communicating Processes) of process algebra. Their theory has however been subject to a number of, rather important, extensions and improvements. Next, we list some of the developments that are most relevant to the subject matter of this paper.

1. A major improvement over the process algebra of [8] was combining the concepts of *communication and concurrency* in the *Algebra of Communicating Processes (ACP)* which was proposed by Bergstra and Klop in [9, 10]. In the process algebra of [8], parallel composition $x \,||\, y$ was a shorthand as defined below.
$$x \,||\, y \doteq (x \mathbin{⌊\!⌊} y) + (y \mathbin{⌊\!⌊} x)$$
There was no possibility for the parallel components to communicate or synchronize. The situation was improved in [9, 10] by introducing a (total) communication function, defining a communication merge operator $|$ and raising the parallel composition operator $||$ to a basic composition operator in the algebra, rather than a defined term.

[1] Sequential composition was called "concatenation" and choice was called "union" in [8].

A. Middeldorp et al. (Eds.): Processes... (Klop Festschrift), LNCS 3838, pp. 251–279, 2005.

2. Another major improvement has been the addition of *identity elements*. Bergstra and Klop in [8] did study the addition of a constant 0 which is an identity element for both nondeterministic choice and sequential composition but then they ruled out this option by observing that the addition of 0 leads to the following counter-intuitive equality:
$$x \cdot y = (x + 0) \cdot y = (x \cdot y) + (0 \cdot y) = (x \cdot y) + y$$
The above equality states that the sequential composition may forget about its first argument which is indeed pathological. A couple of years had to pass to reveal that, as in ordinary rings, two process constants ϵ and δ can be used to give $\_ \cdot \_$ and $\_ + \_$ their identity elements, respectively [16, 22]. (Note that unlike in rings, left-distributivity of choice over sequential composition is still prohibited in the extended process algebra.) Hence, the process algebra PA_δ^ϵ of [22] had two extra constants ϵ and δ. A different proposal for the interplay of ϵ and parallel composition was formulated in [4, 7]. There, a new unary function symbol $\surd(\_)$ is added to the signature in order to capture the possibility of termination for complex terms. ACP of [9, 10] had δ as an identity element for choice but lacked ϵ. In both [16, 22], ϵ is added to ACP resulting in ACP^ϵ. The constant ϵ denotes termination, whereas the action constant encompasses both the action execution and the termination afterwards.
3. The third improvement concerning the subject matter of this paper was the addition of *quantitative time*. Baeten and Bergstra, in [2], proposed a real-time-stamped extension of ACP. In [3], they extend ACP with discrete time using prefix operators $\sigma_{rel}.\_$ and $\sigma_{abs}.\_$ for relative and absolute timing, respectively.

Vereijken tried to extend the result of the first and second improvements with the third aspect in Chapter 6 of his Ph.D. thesis [20]. There, he introduced $ACP_{drt,\epsilon} - ID$ as a discrete time extension of ACP^ϵ (here, $-ID$ denotes the absence of an immediate deadlock constant). However, as it turns out, the above three extensions do not match perfectly: while the extensions in each direction can be interpreted as a conservative one, there is no conservativity result for the extension of ACP^ϵ with timing. In the next section, we review design decisions on the way to timing untimed process algebras. Among the design decisions, we try to find possible cause(s) for this misfortune and will try to improve the situation by redesigning the extensions. This way, we may deviate from the commonly accepted principles of ACP, as we see appropriate. The result will be a lattice of process theories ordered by equational conservativity ordering.

2 Timing the Untimed

The following design decisions have to be taken in order to extend an untimed process algebra with timing information:

1. Delayable vs. urgent actions: When extending an untimed process algebra with timing, a natural question is how to deal with the timing behavior