Bisimulation on Speed: Lower Time Bounds

Gerald Lüttgen[1,*] and Walter Vogler[2]

[1] Department of Computer Science, University of York, York YO10 5DD, U.K.,
`luettgen@cs.york.ac.uk`
[2] Institut für Informatik, Universität Augsburg, D–86135 Augsburg, Germany,
`vogler@informatik.uni-augsburg.de`

Abstract. More than a decade ago, Moller and Tofts published their seminal work on relating processes that are annotated with lower time bounds, with respect to speed. Their paper has left open many questions concerning the semantic theory for their suggested bisimulation–based faster–than preorder, the MT–preorder, which have not been addressed since. The encountered difficulties concern a general compositionality result, a complete axiom system for finite processes, and a convincing intuitive justification of the MT–preorder.

This paper solves these difficulties by developing and employing novel tools for reasoning in discrete–time process algebra, in particular a general commutation lemma relating the sequencing of action and clock transitions. Most importantly, it is proved that the MT–preorder is fully–abstract with respect to a natural amortized preorder that uses a simple bookkeeping mechanism for deciding whether one process is faster than another. Together these results reveal the intuitive roots of the MT–preorder as a faster–than relation, while testifying to its semantic elegance. This lifts some of the barriers that have so far hampered progress in semantic theories for comparing the speed of processes.

1 Introduction

Over the past two decades, the field of process algebra [7] has proved successful for modeling and reasoning about the communication behavior of concurrent processes. Early process algebras, such as Milner's CCS [18] and Hoare's CSP [15], have been augmented to capture other important system aspects as well, including timing behavior [6]. Many variants of *timed process algebra* that employ either discrete or continuous notions of time have been proposed, whose semantic theories are usually based on the well–studied concepts of bisimulation [19], failures [22], or testing [14].

While several approaches for comparing the efficiency of processes have been introduced in the literature [4, 21], theories for comparing timed processes with respect to *speed* are seeded very sparsely. The most seminal paper in the latter category was published over a decade ago [20]. In this paper, the authors

---

Moller and Tofts argue that a *faster–than relation* on processes can only exist for those process–algebraic settings where the passage of time cannot preempt behavior, and especially not for settings involving timeout operators. For a timeout–less fragment of TCCS [19], Moller and Tofts then introduced a compositional faster–than preorder based on strong bisimulation [18], and discussed some of its underlying algebraic laws. Despite the paper's originality, the work is lacking regarding three important aspects. First, the advocated preorder is not intuitively justified but appears to be an ad–hoc remedy for a compositionality problem. Second, the framework possesses technical weaknesses. For example, Moller and Tofts only managed to prove compositionality of their preorder for the class of regular processes, and their proposed laws for characterizing their preorder are incomplete. Third, no semantic theory that abstracts from internal computation, in the sense of observation equivalence [18], is presented in [20].

The aim of this paper is to put the faster–than preorder of Moller and Tofts, or *MT–preorder* for short, on solid semantic grounds and to highlight its intuitive roots, thereby testifying to the elegance of Moller and Tofts' approach. Technically, we add to Milner's CCS a discrete–time clock prefixing operator "σ.", interpreted as *lower time bound*. Intuitively, process P in $\sigma.P$ is only activated after the ticking of the abstract clock σ, i.e., after one time unit. The nesting of σ–prefixes then allows the specification of arbitrary delays (written as prefix (n) with $n \in \mathbb{N}$ in [20]), which results in a process algebra equivalent to the fragment of TCCS studied by Moller and Tofts. We refer to this algebra as *Timed Asynchronous Communicating Systems with lower time bounds*, or TACS$^{\text{LT}}$. As our first main result we prove that the MT–preorder is compositional and *fully–abstract* with respect to a natural *amortized* preorder that uses a simple bookkeeping mechanism for deciding whether one process is faster than another. The intuition behind this amortized preorder is that the faster process must execute each action no later than the slower process does, while both processes must be functionally equivalent in the usual sense of strong bisimulation. To obtain this result we also establish some powerful semantic tools for reasoning within discrete–time process algebra, in particular a general *commutation lemma* relating the sequencing of action and clock transitions. As our second main result we provide a sound and complete *axiomatization* of the MT–preorder for the class of finite processes. This includes the provision of a simple *expansion law*, which Moller and Tofts had claimed could not exist. The twist is that this expansion law is only valid for finite processes, but interestingly not for arbitrary recursive processes. As our third and final main result we introduce a notion of a *weak MT–preorder* — a task that turns out to be far more challenging than in other bisimulation–based process–algebraic settings.

Our results shed light on the nature of the MT–preorder and overcome the technical difficulties experienced by Moller and Tofts, thereby completing, generalizing, and strengthening their results and providing groundwork for advancing semantic theories that compare processes with respect to speed. This paper also complements our previous work on bisimulation–based faster–than relations for timed process algebra with *upper time bounds* [17]. Indeed, several ideas

and technical concepts can be carried over from the upper–time–bounds setting of [17] to the lower–time–bounds setting presented here.

Due to lack of space, all proofs had to be omitted; they will be available under http://www.informatik.uni-augsburg.de/skripts/techreports/ as technical report 2004-1.

2 Timed Asynchronous Communicating Systems

Our process algebra TACS$^{\text{LT}}$ conservatively extends Milner's CCS [18] by permitting the specification of *lower time bounds* for the execution of actions and processes. These will then be used to compare processes with respect to speed. Syntactically, TACS$^{\text{LT}}$ includes a *clock prefixing operator* "$\sigma.$", taken from Hennessy and Regan's TPL [14]. Semantically, it adopts a concept of global, discrete time in which processes are *lazy* and can always let time pass. For example, $\sigma.P$ must wait for *at least* one time unit before it can start executing process P.

Syntax. The syntax of TACS$^{\text{LT}}$ is identical to the one in [17], where we considered a faster–than preorder that relates processes on the basis of upper rather than lower time bounds. Formally, let Λ be a countably infinite set of actions not including the distinguished unobservable, *internal* action τ. With every $a \in \Lambda$ we associate a *complementary action* $\overline{a}$. We define $\overline{\Lambda} =_{\text{df}} \{\overline{a} \,|\, a \in \Lambda\}$ and take $\mathcal{A}$ to denote the set $\Lambda \cup \overline{\Lambda} \cup \{\tau\}$. Complementation is lifted to $\Lambda \cup \overline{\Lambda}$ by defining $\overline{\overline{a}} =_{\text{df}} a$. As in CCS [18], an action a communicates with its complement $\overline{a}$ to produce the internal action τ. We let $a, b, \ldots$ range over $\Lambda \cup \overline{\Lambda}$, $\alpha, \beta, \ldots$ over $\mathcal{A}$, and represent clock ticks by σ. The syntax of TACS$^{\text{LT}}$ is defined as follows:

$$P \ ::= \ \mathbf{0} \ | \ x \ | \ \alpha.P \ | \ \sigma.P \ | \ P + P \ | \ P|P \ | \ P \setminus L \ | \ P[f] \ | \ \mu x.P$$

where x is a *variable* taken from a countably infinite set $\mathcal{V}$ of variables, $L \subseteq \mathcal{A} \setminus \{\tau\}$ is a *restriction set*, and $f : \mathcal{A} \to \mathcal{A}$ is a *finite relabeling*. A finite relabeling satisfies the properties $f(\tau) = \tau$, $f(\overline{a}) = \overline{f(a)}$, and $|\{\alpha \,|\, f(\alpha) \neq \alpha\}| < \infty$. The set of all terms is abbreviated by $\widehat{\mathcal{P}}$, and we define $\overline{L} =_{\text{df}} \{\overline{a} \,|\, a \in L\}$. Moreover, we use the standard definition for *open* and *closed* terms. A variable is called *guarded* in a term if each occurrence of the variable is within the scope of an action or clock prefix. Moreover, we require for terms of the form $\mu x.P$ that x is guarded in P. We refer to closed and guarded terms as *processes*, with the set of all processes written as $\mathcal{P}$, and write $\equiv$ for syntactic equality.

Semantics. The *operational semantics* of a TACS$^{\text{LT}}$ term $P \in \widehat{\mathcal{P}}$ is given by a labeled transition system $\langle \widehat{\mathcal{P}}, \mathcal{A} \cup \{\sigma\}, \longrightarrow, P \rangle$, where $\widehat{\mathcal{P}}$ is the set of states, $\mathcal{A} \cup \{\sigma\}$ the alphabet, $\longrightarrow \subseteq \widehat{\mathcal{P}} \times (\mathcal{A} \cup \{\sigma\}) \times \widehat{\mathcal{P}}$ the transition relation, and P the start state. Transitions labeled with an action α are called *action transitions* that, like in CCS, are local handshake communications in which two processes may synchronize to take a joint state change together. Transitions labeled with the clock symbol σ are called *clock transitions* representing a recurrent global synchronization that encodes the progress of time.

The operational semantics for action and clock transitions can be defined via the *structural operational rules* shown in Tables 1 and 2, resp. As usual, we write $P \xrightarrow{\gamma} P'$ instead of $\langle P, \gamma, P' \rangle \in \longrightarrow$, for $\gamma \in \mathcal{A} \cup \{\sigma\}$, and say that P *may engage in γ and thereafter behave like P'*. Sometimes it is also convenient to write (i) $P \xrightarrow{\gamma}$ for $\exists P'. P \xrightarrow{\gamma} P'$, (ii) $\xrightarrow{\sigma}{}^{k}$ for $k \in \mathbb{N}$ consecutive clock transitions, with $\mathbb{N}$ including 0, and (iii) $P \xrightarrow{w} P'$, where either $w = \epsilon$ and $P \equiv P'$, or $w = \gamma w'$ for some $\gamma \in \mathcal{A} \cup \{\sigma\}$ and $w' \in (\mathcal{A} \cup \{\sigma\})^{*}$, and $\exists \hat{P}. P \xrightarrow{\gamma} \hat{P} \xrightarrow{w'} P'$.

Table 1. Operational semantics for TACS$^{\text{LT}}$ (action transitions)

Act $\quad \dfrac{-}{\alpha.P \xrightarrow{\alpha} P} \qquad$ Rel $\quad \dfrac{P \xrightarrow{\alpha} P'}{P[f] \xrightarrow{f(\alpha)} P'[f]} \qquad$ Rec $\quad \dfrac{P \xrightarrow{\alpha} P'}{\mu x.P \xrightarrow{\alpha} P'[\mu x.P/x]}$

Sum1 $\dfrac{P \xrightarrow{\alpha} P'}{P + Q \xrightarrow{\alpha} P'} \qquad$ Sum2 $\dfrac{Q \xrightarrow{\alpha} Q'}{P + Q \xrightarrow{\alpha} Q'} \qquad$ Res $\quad \dfrac{P \xrightarrow{\alpha} P'}{P \setminus L \xrightarrow{\alpha} P' \setminus L} \;\; \alpha \notin L \cup \overline{L}$

Com1 $\dfrac{P \xrightarrow{\alpha} P'}{P|Q \xrightarrow{\alpha} P'|Q} \qquad$ Com2 $\dfrac{Q \xrightarrow{\alpha} Q'}{P|Q \xrightarrow{\alpha} P|Q'} \qquad$ Com3 $\dfrac{P \xrightarrow{a} P' \quad Q \xrightarrow{\overline{a}} Q'}{P|Q \xrightarrow{\tau} P'|Q'}$

The *action–prefix* term $\alpha.P$ may engage in action α and then behave like P. It may also *idle*, i.e., engage in a clock transition to itself, as process $\mathbf{0}$ does. The *clock–prefix* term $\sigma.P$ can engage in a clock transition to P and ensures that there is a delay of *at least* one time unit before P is activated. The *summation operator* $+$ denotes nondeterministic choice: $P + Q$ may behave like P or Q; according to the deterministic nature of time, a clock transition cannot resolve choices. The *restriction operator* $\setminus L$ prohibits the execution of actions in $L \cup \overline{L}$ and, thus, permits the scoping of actions. $P[f]$ behaves exactly as P with actions renamed by the *relabeling* f. The term $P|Q$ stands for the *parallel composition* of P and Q according to an interleaving semantics with synchronized communication on complementary actions, resulting in the internal action τ. Again, time has to proceed equally on both sides of the operator, i.e., deterministically. Finally, $\mu x. P$ denotes *recursion*, it behaves as a distinguished solution to the equation $x = P$. The rules for action transitions are the same as for CCS, with the exception of the new clock–prefix operator and the rule for recursion; however, the latter is equivalent to the standard CCS rule over guarded terms [5].

The operational semantics for TACS$^{\text{LT}}$ possesses several important properties [14]. Firstly, any process can perform a clock transition due to our adoption of a lazy nil–process $\mathbf{0}$ and a lazy prefix operator. Secondly, the semantics is *time–deterministic*, i.e., progress of time does not resolve choices. Formally, $P \xrightarrow{\sigma} P'$ and $P \xrightarrow{\sigma} P''$ implies $P' \equiv P''$, for all $P, P', P'' \in \widehat{\mathcal{P}}$, which can easily be proved via induction on the structure of P.

Table 2. Operational semantics for $\mathrm{TACS^{LT}}$ (clock transitions)

tNil	$\dfrac{-}{\mathbf{0} \xrightarrow{\sigma} \mathbf{0}}$	tRec	$\dfrac{P \xrightarrow{\sigma} P'}{\mu x.P \xrightarrow{\sigma} P'[\mu x.P/x]}$	tRes	$\dfrac{P \xrightarrow{\sigma} P'}{P \setminus L \xrightarrow{\sigma} P' \setminus L}$
tAct	$\dfrac{-}{\alpha.P \xrightarrow{\sigma} \alpha.P}$	tSum	$\dfrac{P \xrightarrow{\sigma} P' \quad Q \xrightarrow{\sigma} Q'}{P + Q \xrightarrow{\sigma} P' + Q'}$	tRel	$\dfrac{P \xrightarrow{\sigma} P'}{P[f] \xrightarrow{\sigma} P'[f]}$
tPre	$\dfrac{-}{\sigma.P \xrightarrow{\sigma} P}$	tCom	$\dfrac{P \xrightarrow{\sigma} P' \quad Q \xrightarrow{\sigma} Q'}{P|Q \xrightarrow{\sigma} P'|Q'}$		

3 The Moller–Tofts Preorder

This section first recalls the faster–than preorder introduced by Moller and Tofts in [20], to which we refer as *Moller–Tofts preorder*, or MT–preorder for short. As the section's main contribution, we prove the compositionality of this preorder for arbitrary processes, which has only been conjectured by Moller and Tofts. Indeed, the compositionality proof offered in [20] is restricted to processes that do not have any parallel operators inside the scope of a recursion. The key for proving compositionality in the general setting is a nontrivial *commutation lemma* that considers what happens when adjacent action and clock transitions are transposed. This lemma also plays an important role when obtaining the full–abstraction result presented in the next section.

Definition 1 (MT–preorder [20]). A relation $\mathcal{R} \subseteq \mathcal{P} \times \mathcal{P}$ is an *MT–relation* if, for all $\langle P, Q \rangle \in \mathcal{R}$ and $\alpha \in \mathcal{A}$:

1. $P \xrightarrow{\alpha} P'$ implies $\exists Q', k, P''. Q \xrightarrow{\sigma}{}^k \xrightarrow{\alpha} Q'$, $P' \xrightarrow{\sigma}{}^k P''$, and $\langle P'', Q' \rangle \in \mathcal{R}$.
2. $Q \xrightarrow{\alpha} Q'$ implies $\exists P'. P \xrightarrow{\alpha} P'$ and $\langle P', Q' \rangle \in \mathcal{R}$.
3. $P \xrightarrow{\sigma} P'$ implies $\exists Q'. Q \xrightarrow{\sigma} Q'$ and $\langle P', Q' \rangle \in \mathcal{R}$.
4. $Q \xrightarrow{\sigma} Q'$ implies $\exists P'. P \xrightarrow{\sigma} P'$ and $\langle P', Q' \rangle \in \mathcal{R}$.

We write $P \sqsupseteq_{\mathrm{mt}} Q$ if $\langle P, Q \rangle \in \mathcal{R}$ for some MT–relation $\mathcal{R}$, and call $\sqsupseteq_{\mathrm{mt}}$ the *MT–preorder*.

Technically, all conditions of this definition, with the exception of the first one, are identical to the ones of *temporal strong bisimulation* (cf., e.g., [8]). Intuitively, the weaker first condition states that, if the faster process P can perform an action, then the slower process Q must not match this action right away, but can perform an arbitrary number k of time steps before doing so. However, delaying k time steps may make the resulting process Q' faster than P'. To account for this, Moller and Tofts suggest that P' performs k time steps of its own, resulting in process P'' that should then be faster than Q'. To see the necessity for this, consider the processes $a.\mathbf{0}|\sigma.b.\mathbf{0}$ and $\sigma.a.\mathbf{0}|\sigma.b.\mathbf{0}$, for which a sensible faster–than

preorder should clearly identify the former process as the faster one. Here, the a–transition of the former process to $\mathbf{0}|\sigma.b.\mathbf{0}$ can only be matched by the latter process after a delay of one time unit, leading to $\mathbf{0}|b.\mathbf{0}$. However, $\mathbf{0}|\sigma.b.\mathbf{0}$ is not faster than $\mathbf{0}|b.\mathbf{0}$, but only if it has delayed a time unit as well. *Forcing* the faster process to match the delay of the slower one *immediately* seems arbitrary and restrictive. Nevertheless, we will show in the next section that this is not the case and that there is a very natural explanation for this.

It is easy to see that $\sqsupseteq_{\mathrm{mt}}$ is indeed a preorder, i.e., it is reflexive and transitive, and that it is the largest MT–relation. Moreover, if one studies CCS process terms only, i.e., $\mathrm{TACS}^{\mathrm{LT}}$ processes not containing any clock prefix operator, then two processes are related in the MT–preorder if and only if they are strongly bisimular. This is because here all clock transitions are idling transitions, i.e., σ–loops; vice versa, every process can idle due to the laziness property. Hence, CCS is a sub–calculus of $\mathrm{TACS}^{\mathrm{LT}}$.

Theorem 2 (Precongruence). *The MT–preorder $\sqsupseteq_{mt}$ is a precongruence for all $\mathrm{TACS}^{\mathrm{LT}}$ operators.*

The only difficult and non–standard part of the proof concerns compositionality regarding parallel composition and is based on the following *commutation lemma*.

Lemma 3 (Commutation). *Let $P, P' \in \mathcal{P}$ and $w \in (\mathcal{A} \cup \{\sigma\})^*$. If $P \xrightarrow{w} \xrightarrow{\sigma}{}^k P'$, for $k \in \mathbb{N}$, then $\exists P''. P \xrightarrow{\sigma}{}^k \xrightarrow{w} P''$ and $P' \sqsupseteq_{mt} P''$.*

The commutation lemma states that a delay, i.e., one or more clock transitions, after a given sequence of transitions can also be made before this sequence. Moreover, the earlier a delay is performed, the slower the resulting process is. The proof of this lemma is non–trivial and requires the introduction of a simple syntactic faster–than relation on process terms that essentially encodes the syntactic implications of our intuition that any term P should be faster than $\sigma.P$.

Definition 4. The relation $\succ \subseteq \widehat{\mathcal{P}} \times \widehat{\mathcal{P}}$ is defined as the smallest relation satisfying the following properties, for all $P, P', Q, Q' \in \widehat{\mathcal{P}}$.

$$
\begin{array}{llll}
\text{Always:} & (1) \ P \succ P & (2) & P \succ \sigma.P \\
\text{If } P' \succ P \text{ and } Q' \succ Q: & (3) \ P'|Q' \succ P|Q & (4) & P' + Q' \succ P + Q \\
& (5) \ P' \setminus L \succ P \setminus L & (6) & P'[f] \succ P[f] \\
\text{If } P' \succ P \text{ and } x \text{ guarded in } P: & (7) \ P'[\mu x.\, P/x] \succ \mu x.\, P
\end{array}
$$

Observe that relation $\succ$ is not transitive and that it is also defined for open terms. It is interesting to note that $\succ$ is adopted from [17], where we studied bisimulation–based faster–than relations for *upper* time bounds.

Important properties are: $\succ_{|\mathcal{P} \times \mathcal{P}}$ is an MT–relation, i.e. a syntactically faster process is also semantically faster; the process resulting from a clock-transition is syntactically and hence semantically faster than the initial process.

4 The MT–Preorder is Fully–Abstract

While the MT–preorder is algebraically appealing due to its precongruence property, it does not necessarily seem to be a natural choice for defining a faster–than relation. As mentioned earlier, Def. 1 requires that differences in delays between processes must be accounted for within one step of matching, and hence not all the future behaviour of P' in Part 1 is considered. In the following we explore an alternative *amortized* view of faster–than, where the differences can be smoothened out over several steps. Technically, we will prove that the MT–preorder is fully–abstract with respect to this amortized preorder, which demonstrates that the MT–preorder has indeed very intuitive roots.

Definition 5 (Amortized faster–than preorder). A family $(\mathcal{R}_i)_{i\in\mathbb{N}}$ of relations over $\mathcal{P}$ is a *family of faster–than relations* if, for all $i \in \mathbb{N}$, $\langle P, Q\rangle \in \mathcal{R}_i$, and $\alpha \in \mathcal{A}$:

1. $P \xrightarrow{\alpha} P'$ implies $\exists Q', k.\, Q \xrightarrow{\sigma}^k \xrightarrow{\alpha} Q'$ and $\langle P', Q'\rangle \in \mathcal{R}_{i+k}$.
2. $Q \xrightarrow{\alpha} Q'$ implies $\exists P', k{\leq}i.\, P \xrightarrow{\sigma}^k \xrightarrow{\alpha} P'$ and $\langle P', Q'\rangle \in \mathcal{R}_{i-k}$.
3. $P \xrightarrow{\sigma} P'$ implies $\exists Q', k{\geq}0.\, k \geq 1 - i$, $Q \xrightarrow{\sigma}^k Q'$, and $\langle P', Q'\rangle \in \mathcal{R}_{i-1+k}$.
4. $Q \xrightarrow{\sigma} Q'$ implies $\exists P', k{\geq}0.\, k \leq i + 1$, $P \xrightarrow{\sigma}^k P'$, and $\langle P', Q'\rangle \in \mathcal{R}_{i+1-k}$.

We write $P \sqsupseteq_i Q$ if $\langle P, Q\rangle \in \mathcal{R}_i$ for some family of faster–than relations $(\mathcal{R}_i)_{i\in\mathbb{N}}$, and call $\sqsupseteq_0$ the *amortized faster–than preorder*.

This definition reflects our intuition that processes that perform delays later along execution paths are faster than functionally equivalent ones that perform delays earlier; this is because the former processes are executing actions at earlier absolute times (as measured from the start of the processes) than the latter ones. Def. 5 formalizes this intuition as follows: $P \sqsupseteq_i Q$ means that Q, or rather some predecessor of Q, has already performed i clock transitions that were not matched by P; therefore, P has a credit of i clock transitions that it might perform later without a match by Q (cf. Part (3) for $k = 0$). Any extra delays of the slower process when matching an action or clock transition of the faster process, increase credit i accordingly (cf. Parts (1) and (3) for $k > 1$). Vice versa, an action or clock transition of the slower process does not necessarily have to be matched directly by the faster one: the latter may delay up to as many clock transitions as are allowed by the current credit i (cf. Parts (2) and (4)).

Processes $P =_{\text{df}} c.a.\sigma.b.\mathbf{0} + c.a.b.\mathbf{0}$ and $Q =_{\text{df}} c.a.b.\mathbf{0}$ exhibit the difference to the MT–preorder. The family of faster–than relations defined by $\mathcal{R}_0 =_{\text{df}} \{\langle P, Q\rangle\} \cup \{\langle R, R\rangle \mid R \in \mathcal{P}\}$, $R_1 =_{\text{df}} \{\langle a.\sigma.b.\mathbf{0}, a.b.\mathbf{0}\rangle, \langle \sigma.b.\mathbf{0}, b.\mathbf{0}\rangle, \langle b.\mathbf{0}, b.\mathbf{0}\rangle, \langle \mathbf{0}, \mathbf{0}\rangle\}$, and $\mathcal{R}_i =_{\text{df}} \emptyset$, for $i > 1$, testifies to $P \sqsupseteq_0 Q$; note that $P \xrightarrow{c} a.\sigma.b.\mathbf{0}$ is matched by $Q \xrightarrow{\sigma} \xrightarrow{c} a.b.\mathbf{0}$. However, we do not have $P \sqsupseteq_{\text{mt}} Q$. The step $P \xrightarrow{c} a.\sigma.b.\mathbf{0}$ could only be matched by $Q \xrightarrow{\sigma}^k \xrightarrow{c} a.b.\mathbf{0}$ for some $k \in \mathbb{N}$. Since $a.\sigma.b.\mathbf{0} \xrightarrow{\sigma}^k a.\sigma.b.\mathbf{0}$, for any k, this would require $a.\sigma.b.\mathbf{0} \sqsupseteq_{\text{mt}} a.b.\mathbf{0}$, which is clearly wrong.

It can be shown that the amortized faster–than preorder is indeed a preorder and that $(\sqsupseteq_i)_{i\in\mathbb{N}}$ is the (componentwise) largest family of faster–than relations.

However, there is an important shortcoming: $\succsim_0$ is not preserved under parallel composition. Consider the processes P and Q above, where $P \succsim_0 Q$. For $R =_{\mathrm{df}} \mu x.(\sigma.d.\mathbf{0} \mid \sigma.x)$, where d is a 'fresh' action not occurring in the sorts of P and Q, one may show that $P \mid R \not\succsim_0 Q \mid R$. The reason for this is as follows. Transition $P \mid R \xrightarrow{c} a.\sigma.b.\mathbf{0} \mid R$ would need to be matched by a sequence of transitions $Q \mid R \xrightarrow{\sigma}{}^k \xrightarrow{c} a.b.\mathbf{0} \mid d.\mathbf{0} \mid \cdots \mid d.\mathbf{0} \mid R$, for some $k \in \mathbb{N}$ and k parallel components $d.\mathbf{0}$, such that $a.\sigma.b.\mathbf{0} \mid R \succsim_k a.b.\mathbf{0} \mid d.\mathbf{0} \mid \cdots \mid d.\mathbf{0} \mid R$ would hold. Now, let the latter process engage in all d–computations of the k components $d.\mathbf{0}$. Since d is a fresh action, these can only be matched by unfolding k–times process R in $a.\sigma.b.\mathbf{0} \mid R$ and executing k clock transitions and k d–transitions. Thus, $a.\sigma.b.\mathbf{0} \mid R \succsim_0 a.b.\mathbf{0} \mid R$ would follow necessarily, i.e., no credit remains. While the right–hand process can now engage in the sequence $a.b$, the left–hand process can only match action a, but not also action b due to the lack of credit.

To address this compositionality problem of $\succsim_0$ we refine its definition.

Definition 6 (Amortized faster–than precongruence). A family $(\mathcal{R}_i)_{i \in \mathbb{N}}$ of relations over $\mathcal{P}$ is a *precongruence family* if, for all $i \in \mathbb{N}$, $\langle P, Q \rangle \in \mathcal{R}_i$, and $\alpha \in \mathcal{A}$:

1. $P \xrightarrow{\alpha} P'$ implies $\exists Q', k. Q \xrightarrow{\sigma}{}^k \xrightarrow{\alpha} Q'$ and $\langle P', Q' \rangle \in \mathcal{R}_{i+k}$.
2. $Q \xrightarrow{\alpha} Q'$ implies $\exists P', k \leq i. P \xrightarrow{\sigma}{}^k \xrightarrow{\alpha} P'$ and $\langle P', Q' \rangle \in \mathcal{R}_{i-k}$.
3. $P \xrightarrow{\sigma} P'$ implies (a) $i > 0$ and $\langle P', Q \rangle \in \mathcal{R}_{i-1}$, or
   (b) $i = 0$ and $\exists Q'. Q \xrightarrow{\sigma} Q'$ and $\langle P', Q' \rangle \in \mathcal{R}_i$.
4. $Q \xrightarrow{\sigma} Q'$ implies $\langle P, Q' \rangle \in \mathcal{R}_{i+1}$.

We write $P \succsim_i Q$ if $\langle P, Q \rangle \in \mathcal{R}_i$ for some precongruence family $(\mathcal{R}_i)_{i \in \mathbb{N}}$ and call $\succsim_0$ the *amortized faster–than precongruence*.

One can show that this amortized faster–than precongruence is indeed a preorder and that $(\succsim_i)_{i \in \mathbb{N}}$ is the (componentwise) largest family of faster–than relations. This preorder's definition is identical to the one of the amortized faster–than preorder, with the exception that a delay of the faster process now always results in consuming any available credit, while any delay of the slower process results in increasing the credit available to the faster one. As a consequence, it is easy to see that the amortized faster–than precongruence refines the amortized faster–than preorder, i.e., $\succsim_0 \subseteq \succsim_0$.

Theorem 7 (Coincidence). *The preorders $\succsim_0$ and $\succsim_{mt}$ coincide.*

As a consequence, $\succsim_0$ is not only a preorder but indeed a precongruence, since $\succsim_{mt}$ is a precongruence. Note, however, that the relations $\succsim_i$, for $i > 0$, are not precongruences; for example, $\sigma.b.\mathbf{0} \succsim_1 b.\mathbf{0}$ but not $a.\sigma.b.\mathbf{0} \succsim_1 a.b.\mathbf{0}$ due to Def. 6(3).

Theorem 8 (Full abstraction). *The preorder $\succsim_0$ is the* largest precongruence *contained in $\succsim_0$.*

Intuitively, Thms. 7 and 8 show that the MT–preorder rests on a very natural, amortized view of the notion of faster–than. Henceforth, we will call $\succsim_{mt} = \succsim_0$ the *strong faster–than precongruence*.

5 Axiomatizing the Moller–Tofts Preorder

We give a sound and complete axiomatization of our strong faster–than precongruence $\sqsupseteq_{\mathrm{mt}}$ for the class of *finite* processes, which do not contain any recursion operator. This allows one to compare our semantic theory for a calculus with lower time bounds, with the one developed for a calculus with upper time bounds presented in [17], as well as with the CCS theory of *strong bisimulation* [18].

Table 3. Axiom system for finite processes

(A1)	$t + u \;=\; u + t$	(D1)	$\mathbf{0}[f] \;=\; \mathbf{0}$	
(A2)	$t + (u + v) \;=\; (t + u) + v$	(D2)	$(\alpha.t)[f] \;=\; f(\alpha).(t[f])$	
(A3)	$t + t \;=\; t$	(D3)	$(\sigma.t)[f] \;=\; \sigma.(t[f])$	
(A4)	$t + \mathbf{0} \;=\; t$	(D4)	$(t + u)[f] \;=\; t[f] + u[f]$	

(P3)	$t + \sigma.t \;=\; t$	(C1)	$\mathbf{0} \setminus L \;=\; \mathbf{0}$	
(P4)	$\sigma.(t + u) \;=\; \sigma.t + \sigma.u$	(C2)	$(\alpha.t) \setminus L \;=\; \mathbf{0}$	$\alpha \in L \cup \overline{L}$
(P5)	$t \;\sqsupseteq\; \sigma.t$	(C3)	$(\alpha.t) \setminus L \;=\; \alpha.(t \setminus L)$	$\alpha \notin L \cup \overline{L}$
		(C4)	$(\sigma.t) \setminus L \;=\; \sigma.(t \setminus L)$	
(P6)	$\alpha.t \;=\; \alpha.\sigma.t + \alpha.t$	(C5)	$(t + u) \setminus L \;=\; (t \setminus L) + (u \setminus L)$	

Let $\quad t \equiv \sum_{i \in I} \alpha_i.t_i \; [\, + \sigma.t_\sigma \,]$ and $u \equiv \sum_{j \in J} \beta_j.u_j \; [\, + \sigma.u_\sigma \,]$.

(E) $\quad t|u = \sum_{i \in I} \alpha_i.(t_i|u) + \sum_{j \in J} \beta_j.(t|u_j) + \sum_{\alpha_i = \overline{\beta}_j} \tau.(t_i|u_j) +$

$$
\begin{cases}
\mathbf{0} & \text{if both } \sigma.t_\sigma,\ \sigma.u_\sigma \text{ are absent} \\
\sigma.(((\sum_{i \in I} \alpha_i.t_i) + t_\sigma)\,|\,(\sum_{j \in J} \beta_j.u_j)) & \text{if only } \sigma.u_\sigma \text{ is absent} \\
\sigma.((\sum_{i \in I} \alpha_i.t_i)\,|\,((\sum_{j \in J} \beta_j.u_j) + u_\sigma)) & \text{if only } \sigma.t_\sigma \text{ is absent} \\
\sigma.(((\sum_{i \in I} \alpha_i.t_i) + t_\sigma)\,|\,((\sum_{j \in J} \beta_j.u_j) + u_\sigma)) & \text{otherwise}
\end{cases}
$$

The axioms for our MT–precongruence are shown in Table 3, where a term in square brackets is optional. Moreover, $\sum$ is the indexed version of $+$, and we adopt the convention that the sum over the empty index set is identified with process $\mathbf{0}$. Any axiom of the form $t = u$ should be read as two axioms $t \sqsupseteq u$ and $u \sqsupseteq t$. We write $\vdash t \sqsupseteq u$ if $t \sqsupseteq u$ can be derived from the axioms.

Axioms (A1)–(A4), (D1)–(D4), and (C1)–(C5) are exactly the ones for strong bisimulation in CCS [18]. Hence, the semantic theory of our calculus is distinguished from the one for strong bisimulation by the additional Axioms (P3)–(P6) and the refined expansion law (E). Further, it is distinguished from the one for the faster–than preorder for upper time bounds [17] by leaving out Axioms (P1) and (P2) related to enforcing upper time bounds, and by adding Axiom (P6). Intuitively, this added axiom states that inserting a delay within a path of a process does not alter the speed of the process, as long as there exists a functionally equivalent path without delay; this shows that our theory concentrates on best–case behavior by ignoring the slower summand that has the optional delay. Axiom (P6) generalizes to

$$\text{(P6')} \quad \alpha.P \;=\; \alpha.\sigma^k.P + \alpha.P\,,$$

for any $k \in \mathbb{N}$, by repeated application; here, "$\sigma^k.$" stands for k nested clock prefixes. Axiom (P3) is similar in spirit to Axiom (P6) but cannot be derived from the other axioms. Axiom (P4) is a standard axiom in timed process algebras and testifies to the fact that time is a deterministic concept and does not resolve choices. Finally, Axiom (P5) encodes our elementary intuition of clock prefixes and speed within TACS$^{\text{LT}}$, namely that any process t is faster than process $\sigma.t$, which *must* delay the execution of t by one clock tick.

The correctness of our axioms relative to $\sqsupseteq_{\mathrm{mt}}$ can be established as usual [18]. Note that all axioms, with the exception of the Expansion Axiom (E) and Axiom (P3), are sound for arbitrary processes, not only for finite ones. It should be noted here that the axioms presented in [20] do not completely correspond with the MT–preorder, as has also been noted by Moller and Tofts since the publication of their paper in 1991 [*priv. commun.*]. For example, $a.\sigma.b.\mathbf{0} + a.b.\mathbf{0}$ is as fast as $a.b.\mathbf{0}$, which does not seem to be derivable from the axioms in [20]. In our theory, this example is a simple instantiation of Axiom (P6).

Moller and Tofts claim in [20] that the "standard" expansion law [18] for faster–than relations based on lower time bounds does not hold, even for finite processes. While this observation is true for arbitrary processes, it is incorrect for finite ones. As a simple example we have $a.\mathbf{0} \mid \sigma.b.\mathbf{0} = a.(\mathbf{0}|\sigma.b.\mathbf{0})+\sigma.(a.\mathbf{0}|b.\mathbf{0})$, contrary to the claims in [20].

The proof for the completeness of our axiomatization is based on the following notion of *normal form.*

Definition 9 (Normal form). A finite process t is *in normal form* if

$$ t \;\equiv\; \sum_{i \in I} \alpha_i.t_i \;\; [\, + \sigma.t_\sigma \,] \,, $$

where (i) I denotes a finite index set, (ii) $\alpha_i \in \mathcal{A}$ for all $i \in I$, (iii) all the t_i are in normal form, and (iv) the subterm in brackets is optional and, if it exists, t_σ is in normal form $\sum_{j \in J} \beta_j.u_j \; [\, + \sigma.u_\sigma \,]$ and $\forall i \in I \, \exists j \in J. \; \alpha_i.t_i \equiv \beta_j.u_j$.

Theorem 10 (Correctness & completeness). *For finite processes t and u we have:* $\vdash t \sqsupseteq u$ *if and only if* $t \sqsupseteq_{mt} u$.

6 Example

This section applies our semantic theory to a simple example dealing with two implementations of a two–place storage in terms of *two cells* and a *buffer*, respectively (cf., [18]). For simplifying the presentation we specify recursion via recursive process equations in the style of Milner [18], instead of using our recursion operator. The *two–cells* system is defined as the parallel composition of two one–place cells $C_0 \stackrel{\text{def}}{=} in.C_1$, where $C_1 \stackrel{\text{def}}{=} \sigma.\overline{out}.C_0$. The two–place buffer B_0 is given by the process equations $B_0 \stackrel{\text{def}}{=} in.B_1$, $B_1 \stackrel{\text{def}}{=} \sigma.\overline{out}.B_0 + in.B_2$ and $B_2 \stackrel{\text{def}}{=} \sigma.\overline{out}.B_1$. As is reflected by the σ–prefixes in front of the $\overline{out}$–prefixes, both cells C_0 and the two–place buffer B_0 have to delay at least one time unit until they can offer

a communication on port $\overline{out}$. Intuitively, one would expect the two cell system to be strictly faster, since if both cells are full, then both data items stored may be output after a delay of only one time unit, while the buffer requires a delay of at least two time units until it may release the second data item.

As desired, our semantic theory for $\mathrm{TACS^{LT}}$ relates $C_0 \,|\, C_0$ and B_0. Formally, this may be witnessed by the MT–relation given below, in which we employ the abbreviations $C_1' =_{\mathrm{df}} \overline{out}.C_0$, $B_1' =_{\mathrm{df}} \overline{out}.B_0 + in.B_2$, and $B_2' =_{\mathrm{df}} \overline{out}.B_1$.

$$\begin{array}{cccc}
\langle\, C_0 \,|\, C_0 \,,\, B_0 \,\rangle & \langle\, C_1 \,|\, C_0 \,,\, B_1 \,\rangle & \langle\, C_0 \,|\, C_1 \,,\, B_1 \,\rangle & \langle\, C_1' \,|\, C_0 \,,\, B_1' \,\rangle \\
\langle\, C_0 \,|\, C_1' \,,\, B_1' \,\rangle & \langle\, C_1 \,|\, C_1 \,,\, B_2 \,\rangle & \langle\, C_1' \,|\, C_1 \,,\, B_2 \,\rangle & \langle\, C_1 \,|\, C_1' \,,\, B_2 \,\rangle \\
\langle\, C_1' \,|\, C_1' \,,\, B_2' \,\rangle & \langle\, C_1' \,|\, C_0 \,,\, B_1 \,\rangle & \langle\, C_0 \,|\, C_1' \,,\, B_1 \,\rangle &
\end{array}$$

It is easy to check, by referring to Def. 1, that this relation is indeed an MT–relation, whence $C_0 \,|\, C_0 \sqsupseteq_{\mathrm{mt}} B_0$. Vice versa, $B_0 \sqsupseteq_{\mathrm{mt}} C_0 \,|\, C_0$ does *not* hold according to Def. 1, since $C_0 \,|\, C_0$ can engage in the transition sequence $C_0 \,|\, C_0 \xrightarrow{in} \xrightarrow{in} \xrightarrow{\sigma} \xrightarrow{\overline{out}} \xrightarrow{\overline{out}}$, which cannot be matched by B_0. Thus, the two–cells system is faster than the two–place buffer in all contexts, although functionally equivalent, which matches our intuition mentioned above.

Another example, which compares the speeds of different forms of mail delivery and originates in [20], can be adapted from our earlier work on faster–than relations for processes with upper time bounds [17]. This adaptation only requires one to interpret σ–prefixes as lower time bounds instead of upper time bounds. The axiomatic reasoning may then proceed as in [17], which only employs axioms that are part of the axiom system presented in Sec. 5, too.

7 Abstracting from Internal Computation

As usual in process algebra, one wishes to coarsen a semantic theory by abstracting from the internal action τ, which is supposed to be hidden from an external observer. While doing this is usually quite straightforward for CCS–based calculi [18], it turns out to be highly non–trivial here, which we guess may be the reason why it has not been attempted by Moller and Tofts in [20].

We start off by defining a weak version of our reference preorder, the amortized faster–than preorder, which requires us to introduce the following auxiliary notations. For any action α we define $\hat{\alpha} =_{\mathrm{df}} \epsilon$, if $\alpha = \tau$, and $\hat{\alpha} =_{\mathrm{df}} \alpha$, otherwise. Further, we let $\xLongrightarrow{\epsilon} =_{\mathrm{df}} \xrightarrow{\tau}{}^*$ and write $P \xLongrightarrow{\gamma} Q$, where $\gamma \in \mathcal{A} \cup \{\sigma\}$, if there exist R and S such that $P \xLongrightarrow{\epsilon} R \xrightarrow{\gamma} S \xLongrightarrow{\epsilon} Q$. We also let $\xLongrightarrow{\sigma}{}^0$ stand for $\xLongrightarrow{\epsilon}$.

Definition 11 (Weak amortized faster–than preorder). A family $(\mathcal{R}_i)_{i \in \mathbb{N}}$ of relations over $\mathcal{P}$ is a *family of weak faster–than relations* if, for all $i \in \mathbb{N}$, $\langle P, Q \rangle \in \mathcal{R}_i$, and $\alpha \in \mathcal{A}$:

1. $P \xrightarrow{\alpha} P'$ implies $\exists Q', k, k'.\, Q \xLongrightarrow{\sigma}{}^k \xLongrightarrow{\hat{\alpha}} \xLongrightarrow{\sigma}{}^{k'} Q'$ and $\langle P', Q' \rangle \in \mathcal{R}_{i+k+k'}$.
2. $Q \xrightarrow{\alpha} Q'$ implies $\exists P', k, k'.\, k+k' \leq i$, $P \xLongrightarrow{\sigma}{}^k \xLongrightarrow{\hat{\alpha}} \xLongrightarrow{\sigma}{}^{k'} P'$ and
$$\langle P', Q' \rangle \in \mathcal{R}_{i-k-k'}.$$

3. $P \xrightarrow{\sigma} P'$ implies $\exists Q', k{\geq}0.\ k \geq 1{-}i,\ Q \xRightarrow{\sigma}{}^{k} Q'$, and $\langle P', Q' \rangle \in \mathcal{R}_{i-1+k}$.

4. $Q \xrightarrow{\sigma} Q'$ implies $\exists P', k{\geq}0.\ k \leq i{+}1,\ P \xRightarrow{\sigma}{}^{k} P'$, and $\langle P', Q' \rangle \in \mathcal{R}_{i+1-k}$.

We write $P \mathrel{\underset{\approx}{\sqsupseteq}}_i Q$ if $\langle P, Q \rangle \in \mathcal{R}_i$ for some family of weak faster–than relations $(\mathcal{R}_i)_{i \in \mathbb{N}}$, and call $\mathrel{\underset{\approx}{\sqsupseteq}}_0$ the *weak amortized faster–than preorder*.

Relation $\mathrel{\underset{\approx}{\sqsupseteq}}_0$ is indeed a preorder; while reflexivity is obvious, establishing transitivity is simple but nontrivial. The best way of proving transitivity is by showing that $R_k =_{\mathrm{df}} \{ \mathrel{\underset{\approx}{\sqsupseteq}}_i \circ \mathrel{\underset{\approx}{\sqsupseteq}}_j \mid i{+}j = k \}$, for $k \in \mathbb{N}$, is a family of weak faster–than relations. Moreover, one may check that $(\mathrel{\underset{\approx}{\sqsupseteq}}_i)_{i \in \mathbb{N}}$ is the (componentwise) largest family of weak faster–than relations.

Our weakening of the amortized faster–than preorder might appear surprising at first sight, due to the presence of $\xRightarrow{\sigma}{}^{k'}$ trailing weak action transitions on the right–hand side of the definition. As usual for weak bisimilarity, one may have a number of internal transitions before and after a matching action transition, and to get to these trailing internal transitions one may need to pass further clock transitions.

As in the strong case, it is easy to see that $\mathrel{\underset{\approx}{\sqsupseteq}}_0$ is not a precongruence, even not for parallel composition. For reasons we cannot discuss for lack of space, we define the following weak variant of the MT–preorder. (Observe the requirements $P' \xrightarrow{\sigma}{}^{k+k'} P''$ in (1) and $P \xrightarrow{\sigma} P'$ in (4).)

Definition 12 (Weak MT–preorder). A relation $\mathcal{R} \subseteq \mathcal{P} \times \mathcal{P}$ is a *weak MT–relation* if, for all $\langle P, Q \rangle \in \mathcal{R}$ and $\alpha \in \mathcal{A}$:

1. $P \xrightarrow{\alpha} P'$ implies $\exists Q', k, P'', k'.\ Q \xRightarrow{\sigma}{}^{k} \xRightarrow{\hat{\alpha}} \xRightarrow{\sigma}{}^{k'} Q',\ P' \xrightarrow{\sigma}{}^{k+k'} P''$, and $\langle P'', Q' \rangle \in \mathcal{R}$.

2. $Q \xrightarrow{\alpha} Q'$ implies $\exists P'.\ P \xRightarrow{\hat{\alpha}} P'$ and $\langle P', Q' \rangle \in \mathcal{R}$.

3. $P \xrightarrow{\sigma} P'$ implies $\exists Q'.\ Q \xrightarrow{\sigma} Q'$ and $\langle P', Q' \rangle \in \mathcal{R}$.

4. $Q \xrightarrow{\sigma} Q'$ implies $\exists P'.\ P \xrightarrow{\sigma} P'$ and $\langle P', Q' \rangle \in \mathcal{R}$.

We write $P \mathrel{\underset{\approx}{\sqsupseteq}}_{\mathrm{mt}} Q$ if $\langle P, Q \rangle \in \mathcal{R}$ for some weak MT–relation $\mathcal{R}$, and call $\mathrel{\underset{\approx}{\sqsupseteq}}_{\mathrm{mt}}$ the *weak MT–preorder*.

Relation $\mathrel{\underset{\approx}{\sqsupseteq}}_{\mathrm{mt}}$ is a preorder, but the proof of transitivity is difficult. It is obvious from Defs. 1 and 12 that the MT–preorder $\mathrel{\sqsupseteq}_{\mathrm{mt}}$ is a weak MT-relation and hence included in $\mathrel{\underset{\approx}{\sqsupseteq}}_{\mathrm{mt}}$. Further, $\mathrel{\underset{\approx}{\sqsupseteq}}_{\mathrm{mt}}$ is included in the weak amortized faster–than preorder $\mathrel{\underset{\approx}{\sqsupseteq}}_0$, since one can prove that $\mathcal{R}_i =_{\mathrm{df}} \{ \langle P, Q \rangle \mid P \xrightarrow{\sigma}{}^{i} P' \mathrel{\underset{\approx}{\sqsupseteq}}_{\mathrm{mt}} Q \}$ is a family of weak faster–than relations.

Proposition 13. *The weak MT–preorder $\mathrel{\underset{\approx}{\sqsupseteq}}_{mt}$ is compositional for all* $\mathrm{TACS}^{\mathrm{LT}}$ *operators except summation.*

As expected for a CCS–based process calculus, $\mathrel{\underset{\approx}{\sqsupseteq}}_{\mathrm{mt}}$ is not a precongruence for summation, but the summation fix used for other bisimulation–based timed process algebras proves adequate for $\mathrm{TACS}^{\mathrm{LT}}$, too.

Definition 14 (Weak MT–precongruence). A relation $\mathcal{R} \subseteq \mathcal{P} \times \mathcal{P}$ is a *weak MT–precongruence relation* if, for all $\langle P, Q \rangle \in \mathcal{R}$ and $\alpha \in \mathcal{A}$:

1. $P \xrightarrow{\alpha} P'$ implies $\exists Q', k, P'', k'. \, Q \xRightarrow{\sigma}{}^{k} \xRightarrow{\alpha} \xRightarrow{\sigma}{}^{k'} Q', \, P' \xrightarrow{\sigma}{}^{k+k'} P''$, and
$P'' \mathrel{\overset{\sqsupseteq}{\approx}}_{\mathrm{mt}} Q'$.
2. $Q \xrightarrow{\alpha} Q'$ implies $\exists P'. \, P \xRightarrow{\alpha} P'$ and $P' \mathrel{\overset{\sqsupseteq}{\approx}}_{\mathrm{mt}} Q'$.
3. $P \xrightarrow{\sigma} P'$ implies $\exists Q'. \, Q \xrightarrow{\sigma} Q'$ and $\langle P', Q' \rangle \in \mathcal{R}$.
4. $Q \xrightarrow{\sigma} Q'$ implies $\exists P'. \, P \xrightarrow{\sigma} P'$ and $\langle P', Q' \rangle \in \mathcal{R}$.

We write $P \mathrel{\overset{\sqsupseteq}{\approx}}_{\mathrm{mt}} Q$ if $\langle P, Q \rangle \in \mathcal{R}$ for some weak MT–precongruence relation $\mathcal{R}$, and call $\mathrel{\overset{\sqsupseteq}{\approx}}_{\mathrm{mt}}$ the *weak MT–precongruence*.

Again, $\mathrel{\overset{\sqsupseteq}{\approx}}_{\mathrm{mt}}$ is a preorder and the largest weak MT–precongruence relation. It is worth pointing out that the strong faster–than precongruence $\mathrel{\overset{\sqsupseteq}{\sim}}_{\mathrm{mt}}$ is contained in the weak faster–than precongruence $\mathrel{\overset{\sqsupseteq}{\approx}}_{\mathrm{mt}}$, which follows by inspecting the respective definitions. The recursive definition of the weak MT–precongruence employed in (3) and (4) above reflects the fact that clock transitions do not resolve choices.

Theorem 15. $\mathrel{\overset{\sqsupseteq}{\approx}}_{mt}$ *is the largest precongruence contained in* $\mathrel{\overset{\sqsupseteq}{\approx}}_{mt}$.

It remains an open question whether the weak MT–precongruence is also the largest precongruence contained in the weak amortized faster–than preorder. Our attempts of finding a suitable context for proving this full–abstraction result have been unsuccessful so far. Nevertheless we believe in the validity of such a result and are optimistic to identify a simpler formulation of the weak MT–preorder, referring to fewer internal computation steps, from which the desired context may be derived.

8   Related Work

Although there is a wealth of literature on timed process algebras [6], little work has been done in developing theories for relating processes with respect to speed. The approaches closest to ours are obviously the one by Moller and Tofts regarding processes equipped with lower time bounds [20], and our own one regarding processes equipped with upper time bounds [17]. As these have been referred to and discussed throughout, we refrain from repetitions here.

The probably best–known related work focuses on comparing *process efficiency* rather than process speed. Arun–Kumar and Hennessy [3, 4] have developed a bisimulation–based theory for untimed processes that is based on counting internal actions, which was later carried over to De Nicola and Hennessy's testing framework [12] by Natarajan and Cleaveland [21]. In these theories, runs of parallel processes are seen to be the interleaved runs of their component processes. Consequently, e.g., $(\tau.a.\mathbf{0} \mid \tau.\overline{a}.b.\mathbf{0}) \setminus \{a\}$ is as efficient as $\tau.\tau.\tau.b.\mathbf{0}$, whereas, in our setting, $(\sigma.a.\mathbf{0} \mid \sigma.\overline{a}.b.\mathbf{0}) \setminus \{a\}$ is strictly faster than $\sigma.\sigma.\tau.b.\mathbf{0}$.

The sparse work on comparing process speeds largely concentrated on worst–case timing behavior on the basis of upper time bounds. Research by Vogler et al. [16, 23] originally employed the concurrency–theoretic framework of Petri nets and testing semantics; it has only recently been carried over to the process algebra PAFAS [11] and is discussed in [17]. Simultaneously, Corradini et al. [10] pursued a different idea for relating processes with respect to speed, which is known as the *ill–timed–but–well–caused* approach [2, 13]. This approach allows system components to attach local time stamps to actions. Since actions may occur as in an untimed process algebra, local time stamps may decrease within a sequence of actions which is exhibited by several processes running in parallel. The presence of these "ill–timed" runs makes it difficult to relate the faster–than preorder of Corradini et al. to the one of Moller and Tofts; a modified approach that restricts attention to "well–timed" behaviour might allow some meaningful result.

9   Conclusions and Future Work

In previous work [17], the authors investigated bisimulation–based preorders that relate the speeds of asynchronous processes relative to given upper time bounds, specifying when actions have to be executed at the latest. The present paper considered the case of lower time bounds, specifying when actions may be executed at the earliest, by revisiting the seminal approach of Moller and Tofts [20]. Although Moller and Tofts' work was published more than a decade ago and the first one to introduce a faster–than relation in timed process algebra, it was never followed up in the literature – except for [1] where characteristic formulae are provided. One reason for this might be the absence of strong theoretical results, including the absence of a compositionality result for arbitrary processes, of a full–abstraction result, and of a complete axiomatization for finite processes, as well as the bleak picture drawn in [20] for achieving such results elegantly.

This paper established these nontrivial results by introducing some novel process–algebraic techniques, including a commutation lemma between action and clock transitions. In particular, we proved a full–abstraction theorem with respect to a very intuitive amortized preorder that uses bookkeeping for deciding whether one process is faster than another. In addition, an expansion law was established for finite processes, which paved the way for a sound and complete axiomatization of the Moller–Tofts preorder. This not only testifies to the nature of the MT–preorder but also highlights its importance among the sparse related work in the field. Last, but not least, a variant of the MT–preorder that abstracts from internal, unobservable actions was studied.

Future work should proceed along three directions. First, we wish to complete the theory for our weak MT–precongruence by establishing the conjectured full–abstraction result. Second, the developed preorders should be implemented in a formal verification tool, such as the *Concurrency Workbench NC* [9]. Third, we intend to integrate our theory for lower time bounds with our earlier work on

upper time bounds [17], thereby exploring the appropriateness of our faster–than approaches for settings with restricted asynchrony.

References

[1] L. Aceto, A. Ingólfsdóttir, M. L. Pedersen, and J. Poulsen. Characteristic formulae for timed automata. *RAIRO, Theoretical Informatics and Applications*, 34:565–584, 2000.

[2] L. Aceto and D. Murphy. Timing and causality in process algebra. *Acta Inform.*, 33(4):317–350, 1996.

[3] S. Arun-Kumar and M.C.B. Hennessy. An efficiency preorder for processes. *Acta Inform.*, 29(8):737–760, 1992.

[4] S. Arun-Kumar and V. Natarajan. Conformance: A precongruence close to bisimilarity. In *STRICT '95*, Workshops in Comp., pp. 55–68. Springer-Verlag, 1995.

[5] E. Badouel and P. Darondeau. On guarded recursion. *TCS*, 82(2):403–408, 1991.

[6] J.C.M. Baeten and C.A. Middelburg. Process algebra with timing: Real time and discrete time. In Bergstra et al. [7], ch. 10, pp. 627–684.

[7] J.A. Bergstra, A. Ponse, and S.A. Smolka, eds. *Handbook of Process Algebra*. Elsevier Science, 2001.

[8] R. Cleaveland, G. Lüttgen, and M. Mendler. An algebraic theory of multiple clocks. In *CONCUR '97*, vol. 1243 of *LNCS*, pp. 166–180. Springer-Verlag, 1997.

[9] R. Cleaveland and S. Sims. The NCSU Concurrency Workbench. In *CAV '96*, vol. 1102 of *LNCS*, pp. 394–397. Springer-Verlag, 1996.

[10] F. Corradini, R. Gorrieri, and M. Roccetti. Performance preorder and competitive equivalence. *Acta Inform.*, 34(11):805–835, 1997.

[11] F. Corradini, W. Vogler, and L. Jenner. Comparing the worst-case efficiency of asynchronous systems with PAFAS. *Acta Informatica*, 38:735–792, 2002.

[12] R. De Nicola and M.C.B. Hennessy. Testing equivalences for processes. *TCS*, 34(1-2):83–133, 1984.

[13] R. Gorrieri, M. Roccetti, and E. Stancampiano. A theory of processes with durational actions. *TCS*, 140(1):73–94, 1995.

[14] M.C.B. Hennessy and T. Regan. A process algebra for timed systems. *Inform. and Comp.*, 117(2):221–239, 1995.

[15] C.A.R. Hoare. *Communicating Sequential Processes*. Prentice Hall, 1985.

[16] L. Jenner and W. Vogler. Fast asynchronous systems in dense time. *TCS*, 254(1-2):379–422, 2001.

[17] G. Lüttgen and W. Vogler. A faster-than relation for asynchronous processes. In *CONCUR 2001*, vol. 2154 of *LNCS*, pp. 262–276. Springer-Verlag, 2001. Full version to appear in Inform. and Comp. under the title *Bisimulation on Speed: Worst–Case Efficiency*.

[18] R. Milner. *Communication and Concurrency*. Prentice Hall, 1989.

[19] F. Moller and C. Tofts. A temporal calculus of communicating systems. In *CONCUR '90*, vol. 458 of *LNCS*, pp. 401–415. Springer-Verlag, 1990.

[20] F. Moller and C. Tofts. Relating processes with respect to speed. In *CONCUR '91*, vol. 527 of *LNCS*, pp. 424–438. Springer-Verlag, 1991.

[21] V. Natarajan and R. Cleaveland. An algebraic theory of process efficiency. In *LICS '96*, pp. 63–72. IEEE Computer Society Press, 1996.

[22] S. Schneider. An operational semantics for timed CSP. *Inform. and Comp.*, 116(2):193–213, 1995.

[23] W. Vogler. Faster asynchronous systems. In *CONCUR '95*, vol. 962 of *LNCS*, pp. 299–312. Springer-Verlag, 1995.