

A Process Algebra with Distributed Priorities

R. Cleaveland* G. Lüttgen† V. Natarajan*
e-mail: {rance,luetngen,nvaidhy}@eos.ncsu.edu

Technical Report TR-96-02
Department of Computer Science
North Carolina State University
Raleigh, NC 27695-8206, USA
March 1996

Abstract

This paper presents a process algebra for distributed systems in which some actions may take precedence over others. In contrast with existing approaches to priorities, our algebra only allows actions to take priority over others at the same “location” and therefore captures a notion of *localized preemption*. Using Park’s and Milner’s notion of strong bisimulation as a basis, we develop a behavioral congruence and axiomatize it for finite processes; we also derive an associated observational congruence. Simple examples are given that highlight the utility of the theory.

Keywords: process algebras, distributed systems, priorities, bisimulation, localized preemption, locations, axiomatization.

*Research supported by NSF/DARPA grant CCR-9014775, NSF grant CCR-9120995, ONR Young Investigator Award N00014-92-J-1582, NSF Young Investigator Award CCR-9257963, NSF grant CCR-9402807, and AFOSR grant F49620-95-1-0508.

†Research support provided by the German Academic Exchange Service under grant D/95/09026 (Doktorandenstipendium HSP II/ AUFE).

1 Introduction

Process algebras [5, 18, 19, 22] provide widely studied frameworks for modeling and verifying concurrent systems [12, 15]. Such theories typically consist of a simple language with a well-defined operational semantics given in terms of labeled transition systems; a behavioral equivalence is then used to relate implementations and specifications, which are both given as terms in the language. In order to facilitate compositional reasoning, in which systems are verified on the basis of the behavior of their components, researchers have devoted great attention on the definition of behavioral congruences which allow the substitution of “equals for equals” inside larger systems. Traditional process algebras focus on modeling the potential nondeterminism that concurrent processes may exhibit; approaches have also been suggested for introducing sensitivity to other aspects of system behavior, including priority [3, 9, 10, 13, 16, 20, 26] and concurrency [1, 8, 24]. The latter work presents theories in which concurrency is treated as a primitive notion that is not reducible to nondeterminism, while the former enables the modeling of systems in which some system transitions (e.g. interrupts) may take precedence over others.

In this paper, we develop an algebraic theory of action priority for *distributed* systems. As in existing work, our aim is to model systems in which some transitions have precedence over others. Our point of departure is that the priority scheme should be localized within individual sites in the system; actions should only be able to preempt actions being performed at the “same location.” This constraint reflects an essential intuition about distributed systems, which is that the execution of a process on one processor should not affect the behavior of a process on another processor unless the designer explicitly builds in an interaction (e.g. synchronization) between them. Technically, we begin with a theory of priority that includes a notion of global precedence [13, 26] and show how its semantics may be altered using ideas from true concurrency [1, 8] to localize capabilities for preemption. We then define a strong congruence for this language, axiomatize it for finite processes, and derive an observational congruence. Those congruences are the coarsest ones contained in the strong bisimulation and observational equivalence adapted from [23], respectively. We also provide logical characterizations of our behavioral relations.

Organization of the Paper. In the next section, we present a generic example illustrating the need for local preemption in modeling systems. The following three sections present our language and derive the technical results discussed above, while Section 6 presents an example showing the application of our theory. Section 7 discusses related work, and the last section presents our conclusions and directions for future work. An appendix contains proofs of the more challenging results.

2 Motivating Example

The example depicted in Figure 1 motivates the necessity of considering a local notion of preemption when dealing with priorities in distributed systems. It consists of two sites, **Site1** and **Site2**, e.g. two computers, that are connected via the network **Network**. Each site runs an application, **Application1** and **Application2**, respectively, which may send or receive information from the application at the other site via its (interrupt-)handler, **Handler1** or **Handler2**. A handler delivers the message to the network or receives a message for its site from the network and notifies the application by sending an interrupt. Now, we have the following intuitive requirements which the semantics of our language CCS^{prio} should satisfy, in order to reflect the behavior of the system correctly. First, the interrupt of a handler should preempt the normal work of

the application at its site, i.e. the application should immediately react on an interrupt request. Second, both sites should be able to perform internal computations only involving one specific site autonomously. Especially, `Handler1` should not be able to preempt the work of `Handler2`, and vice versa. Whereas the first requirement is satisfied in traditional approaches to priorities in process algebra, the second requirement is violated, since e.g. the following situation may occur with respect to the traditional semantics [13, 26]. If the priority of `Application2` is higher than the priority of `Application1`, then `Application1` is blocked until `Application2` is terminated or until it has to wait for a message from the other site. In general, priorities of actions which are local to one site should be incomparable with priorities arising from another site. However, a global agreement of the meaning of priority values has to exist between communicating processes.

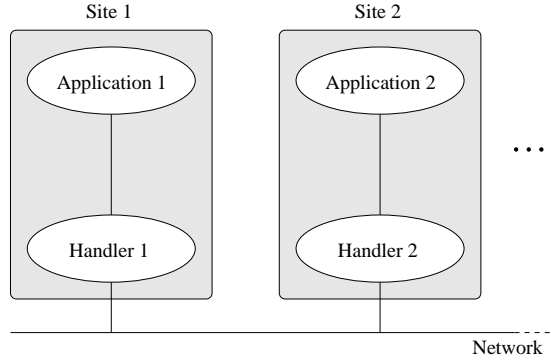


Figure 1: Standard distributed system

The notion of global preemption in traditional approaches to priorities eliminates many execution paths of the real system in the model. Therefore, safety properties verified in the model may be violated in the real system. This is often an intolerable side-effect which we eliminate in this paper by introducing a notion of *localized* preemption.

3 Syntax and Semantics of CCS^{prio}

In this section, we define the syntax and semantics of our language CCS^{prio} which is based on CCS [22].

3.1 Syntax of CCS^{prio}

The syntax of CCS^{prio} differs from CCS by the structure of the action set which exhibit a priority scheme. For the sake of simplicity, we restrict ourselves to a two-level priority scheme. However, all results presented in this paper can be generalized to multi-level priority schemes in a straightforward fashion. Intuitively, actions represent potential synchronizations that a process may be willing to engage in with its environment. Given a choice between a synchronization on a high priority action and one on a low priority action, a process should choose the former.

Formally, let Λ be a countable set of action labels, not including the so called *internal* or *silent* action τ . For every *input action* $a \in \Lambda$, there exists a *complementary action* \bar{a} , the corresponding *output action*. Further, let $\bar{\Lambda} =_{\text{df}} \{\bar{a} \mid a \in \Lambda\}$, and let us denote the set of all actions $\Lambda \cup \bar{\Lambda} \cup \{\tau\}$, where $\tau \notin \Lambda$, by A . Intuitively, an action indicates that a process is willing to perform a synchronization on the *port* associated with the action name, i.e. action a means that the process wants to receive a message from port a whereas \bar{a} means that the process wants to send a message

via port a . The action τ either indicates an internal action of a process or the synchronization of two processes on some port in order to communicate with each other. Finally, we let a, b, \dots range over Λ and α, β, \dots over A .

In order to define *prioritized actions*, let $\underline{\Lambda}$ be a countable set of prioritized action labels. Then $\underline{A} =_{\text{df}} \underline{\Lambda} \cup \overline{\underline{\Lambda}} \cup \{\underline{\tau}\}$ is the set of *prioritized actions* where $\underline{\tau}$ is the prioritized *internal* or *silent* action. We use $\mathcal{A} =_{\text{df}} A \cup \underline{A}$ to denote the set of all actions.

In the remainder of the paper, let $\underline{a}, \underline{b}, \dots$ range over $\underline{\Lambda}$, the symbols $\underline{\alpha}, \underline{\beta}, \dots$ over \underline{A} , and γ, δ over A . Additionally, we extend $\bar{\cdot}$ by $\overline{\overline{\gamma}} = \gamma$, and if $L \subseteq \mathcal{A} \setminus \{\tau, \underline{\tau}\}$ then $\overline{L} =_{\text{df}} \{\overline{\gamma} \mid \gamma \in L\}$. A function f on \mathcal{A} is called a *relabeling* if it satisfies $f(\overline{a}) = \overline{f(a)}$, $f(\underline{a}) = \underline{f(a)}$, $f(\tau) = \tau$, and $f(\underline{\tau}) = \underline{\tau}$. Moreover, f preserves priorities, i.e. $f(\Lambda) \subseteq \Lambda$ and $f(\underline{\Lambda}) \subseteq \underline{\Lambda}$, and the set $\{\gamma \mid f(\gamma) \neq \gamma\}$ is finite.

Further, we use the standard definitions for *sort* of a process, *free* and *bound variables*, *open* and *closed terms*, *guarded recursion*, and *contexts*. We refer to closed and guarded terms as *processes* and denote syntactic equality by \equiv . Let P, Q, R, \dots range over the set \mathcal{P} of processes. The syntax of our language is defined by the following BNF where f is a relabeling, $L \subseteq \mathcal{A} \setminus \{\tau, \underline{\tau}\}$, and C is a process constant.

$$\begin{array}{lclclclcl}
 P & ::= & \mathbf{0} & \mid & \gamma.P & \mid & P + P & \mid & P|P & \mid \\
 & & P[f] & \mid & P \setminus L & \mid & C \stackrel{\text{def}}{=} P & & &
 \end{array}$$

3.2 Comparability Relation

In this section, we present some notions and intuition which are necessary to understand the operational semantics of CCS^{prio} that is given in the next section. In order to define a semantics dealing with a notion of localized preemption we have to take locations of actions into account. Therefore, we are formally introducing locations. A location of an action in a process is a string which *addresses* the subterm of the process from which the action is performed. Our address scheme follows the convention that addresses are written down from the most specific to the most general information, from left to right (cf. [24]).

Let $\mathcal{A}_{\text{loc}} =_{\text{df}} \{L, R, l, r\}$ be the *location alphabet*, and let $\mathcal{L}oc$ denote the set of all words over \mathcal{A}_{loc} concatenated with the special symbol \bullet to the left, i.e. $\mathcal{L}oc$ is the set of all *locations*. As usual, \cdot denotes the concatenation operator as e.g. in $\bullet \cdot L \cdot l \in \mathcal{L}oc$. Further, we abbreviate $\{m \cdot \zeta \mid m \in M\}$ by $M \cdot \zeta$ for $M \subseteq \mathcal{L}oc$ and $\zeta \in \mathcal{A}_{\text{loc}}$. Our semantics of CCS^{prio} is defined in a way that an action is associated with the location \bullet if that action can be performed right here, whereas l (r) requires to explore the left (right) side of the summation operator, and L (R) directs us to the left (right) side of the parallel operator, respectively. For example, the process $(a \bullet \mathbf{0} \mid b \bullet \mathbf{0}) + c \bullet \mathbf{0}$ can perform action a from location $\bullet \cdot L \cdot l$, action b from location $\bullet \cdot R \cdot l$, and action c from location $\bullet \cdot r$. The location \bullet is the ‘root’ of all locations. For simplicity, we often write m instead of $\bullet \cdot m$ for $m \in \mathcal{L}oc$.

As mentioned in the introduction, we want to adopt the view that processes on different sides of the parallel operator are (logically) executed on different processors, i.e. at different locations. Thus, priorities on different sides of the parallel operator are distributed and, therefore, should be incomparable. However, processes on different sides of the summation operator, which models nondeterministic choice, are scheduled on a single processor, i.e. they should be comparable. We formalize this intuition in the following comparability relation on locations which is adopted from [17].

Definition 3.1 *The comparability relation \bowtie on locations is the smallest reflexive and symmetric subset of $\mathcal{Loc} \times \mathcal{Loc}$ such that for all $v, w \in \mathcal{Loc}$.*

1. $(v \cdot l, w \cdot r) \in \bowtie$, and
2. $(v, w) \in \bowtie$ implies $(v \cdot \zeta, w \cdot \zeta) \in \bowtie$ for $\zeta \in \mathcal{A}_{loc}$.

We write $v \bowtie w$ instead of $(v, w) \in \bowtie$.

Note that the comparability relation is not transitive, e.g. we have $L \cdot l \bowtie r$ and $r \bowtie R \cdot l$ but $L \cdot l \not\bowtie R \cdot l$ since $L \not\bowtie R$. Considering our example $(a \cdot \mathbf{0} \mid b \cdot \mathbf{0}) + c \cdot \mathbf{0}$ above, the locations of the actions a and c and the locations of the actions b and c are comparable since they are just on different sides of the summation operator. In contrast, the locations of the actions a and b are incomparable since they are on different sides of the parallel operator.

In the following, let m, n, o, \dots range over \mathcal{Loc} . Moreover, $[m]$ denotes the set $\{o \in \mathcal{Loc} \mid o \bowtie m\}$, and $[[m, n]]$, where $\langle m, n \rangle \in \mathcal{Loc} \times \mathcal{Loc}$, is defined by $[m] \cup [n]$. For the sake of brevity, we write \mathcal{Loc} for the set $\mathcal{Loc} \cup (\mathcal{Loc} \times \mathcal{Loc})$. Considering also pairs of locations is necessary because communications in a CCS-based framework take place between two processes offering complementary actions. The result of a communication is an internal action which, therefore, is assigned the two locations of the complementary actions. Finally, we define $\langle m, n \rangle \cdot \zeta =_{\text{def}} \langle m \cdot \zeta, n \cdot \zeta \rangle$ where $m, n \in \mathcal{Loc}$ and $\zeta \in \mathcal{A}_{loc}$.

3.3 Semantics of CCS^{prio}

The (operational) semantics of a CCS^{prio} process $P \in \mathcal{P}$ is given by a labeled transition system $\langle \mathcal{P}, \mathcal{A}, \longrightarrow, P \rangle$ where \mathcal{P} is the set of states, \mathcal{A} the alphabet, \longrightarrow the transition relation, and P the start state. The transition relation $\longrightarrow \subseteq \mathcal{P} \times (\mathcal{Loc} \times \mathcal{A}) \times \mathcal{P}$ is defined in Table 2 using Plotkin-style [29] operational rules. We write $P \xrightarrow{m, \gamma} P'$ instead of $\langle P, m, \gamma, P' \rangle \in \longrightarrow$, and $P \xrightarrow{m, \epsilon} P'$ if $P \equiv P'$. We say that P may engage in action γ offered from location m and thereafter behaves like process P' . Moreover, if $\gamma \in \underline{\mathcal{A}}$ then we abbreviate $P \xrightarrow{m, \gamma} P'$ by $P \xrightarrow{\gamma} P'$ since it turns out that the location m is not important when reasoning about prioritized transitions, i.e. transitions labeled by a prioritized action.

The presentation of the operational rules requires *prioritized initial action sets* which are defined as the least relations satisfying the rules in Table 1. Intuitively, $\underline{\mathcal{I}}_m(P)$ denotes the set of all prioritized initial actions of P from location m . Note that those sets are either empty or contain exactly one initial transition. $\underline{\mathcal{I}}_m(P) = \emptyset$ means that either m is not a location of P or P wants to perform an unprioritized action at location m . Additionally, let us denote the set of all prioritized initial actions of process P from locations $M \subseteq \mathcal{Loc}$ by $\underline{\mathcal{I}}_M(P)$, and the set of all prioritized initial actions of process P by $\underline{\mathcal{I}}(P)$. We also define analogous initial action sets ignoring internal actions and denote them by $\underline{\underline{\mathcal{I}}}_m(P)$, $\underline{\underline{\mathcal{I}}}_M(P)$, and $\underline{\underline{\mathcal{I}}}(P)$, respectively.

Note that the set of actions is defined independently from the transition relation \longrightarrow . Therefore, \longrightarrow is well-defined although its definition contains negative premises (side conditions) [7]. The side conditions of the operational semantic rules guarantee that a process does not perform an unprioritized action if it can engage in a prioritized synchronization or internal computation, i.e. a $\underline{\tau}$ -transition, from a comparable location. Therefore, $\underline{\tau}$ -actions have preemptive power over unprioritized actions. The reason that prioritized visible actions do *not* have priority over unprioritized actions is that visible actions only indicate the potential of a synchronization, i.e. the potential of progress, whereas internal actions describe *real* progress in our model.

Table 1: Initial action sets

$\underline{L}_m(C) = \text{df } \underline{L}_m(P)$ where $C \stackrel{\text{def}}{=} P$	$\underline{L}_\bullet(\underline{\alpha}.P) = \text{df } \{\underline{\alpha}\}$
$\underline{L}_{m.l}(P + Q) = \text{df } \underline{L}_m(P)$	$\underline{L}_{n.r}(P + Q) = \text{df } \underline{L}_n(Q)$
$\underline{L}_m(P[f]) = \text{df } \{f(\underline{\alpha}) \mid \underline{\alpha} \in \underline{L}_m(P)\}$	$\underline{L}_m(P \setminus L) = \text{df } \underline{L}_m(P) \setminus (L \cup \bar{L})$
$\underline{L}_{m.L}(P Q) = \text{df } \underline{L}_m(P)$	$\underline{L}_{n.R}(P Q) = \text{df } \underline{L}_n(Q)$
$\underline{L}_{(m.L, n.R)}(P Q) = \text{df } \underline{L}_m(P) \cup \underline{L}_n(Q) \cup \{\underline{\tau} \mid \underline{L}_m(P) \cap \bar{\underline{L}}_n(Q) \neq \emptyset\}$	
$\underline{L}_M(P) = \text{df } \bigcup \{\underline{L}_m(P) \mid m \in M\}$	$\underline{\mathbb{L}}_M(P) = \text{df } \underline{L}_M(P) \setminus \{\underline{\tau}\}$
$\underline{\mathbb{L}}(P) = \text{df } \bigcup \{\underline{L}_{(m,n)}(P) \mid m, n \in \text{Loc}\}$	$\underline{\mathbb{L}}(P) = \text{df } \underline{\mathbb{L}}(P) \setminus \{\underline{\tau}\}$

The semantics of CCS^{prio} according to prioritized transitions is the same as the usual CCS semantics. The difference arises by the side conditions in the rules for unprioritized transitions. The process $\gamma.P$ may engage in action γ and then behave like P . The *summation operator* $+$ denotes *nondeterministic choice*. The process $P + Q$ may behave like process P (Q) if Q (P) does not preempt unprioritized actions by performing a $\underline{\tau}$ -action. Consequently, priorities arising from different sides of the summation operator are comparable. The *restriction operator* $\setminus L$ prohibits the execution of actions in $L \cup \bar{L}$. Thus, the restriction operator permits the *scoping* of actions. $P[f]$ behaves exactly as the process P where the actions are renamed with respect to the relabeling f . The process $P|Q$ 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 . Since locations on different sides of a parallel operator are incomparable, $\underline{\tau}$'s arising from a location of P (Q) cannot preempt the execution of an action, even an unprioritized one, of Q (P). Only if P (Q) engages in a prioritized synchronization with Q (P), then unprioritized actions of P and Q are preempted. Finally, $C \stackrel{\text{def}}{=} P$ denotes a *constant definition*, i.e. C is a recursively defined process which behaves as a distinguished solution of the equation $C = P$. Note that the side conditions of summation and parallel composition are conform with our comparability relation. Moreover, the operational semantics possesses the following important properties.

Lemma 3.2 (Associativity & Commutativity)

The summation and the parallel operator of CCS^{prio} are associative and commutative in the sense that corresponding transition systems are identical up to renaming of states.

4 Prioritized Strong Bisimulation

In this section, we present an equivalence relation for CCS^{prio} processes which is based on bisimulation [28]. Our aim is to characterize the largest congruence contained in the standard strong bisimulation [22].

Table 2: Operational semantics for CCS^{prio}

<u>Act</u>	$\frac{-}{\alpha.P \xrightarrow{\alpha} P}$	Act	$\frac{-}{\alpha.P \xrightarrow{\bullet, \alpha} P}$
<u>Sum1</u>	$\frac{P \xrightarrow{\alpha} P'}{P + Q \xrightarrow{\alpha} P'}$	Sum1	$\frac{P \xrightarrow{m, \alpha} P'}{P + Q \xrightarrow{m, l, \alpha} P'} \quad \tau \notin \mathbb{I}(Q)$
<u>Sum2</u>	$\frac{Q \xrightarrow{\alpha} Q'}{P + Q \xrightarrow{\alpha} Q'}$	Sum2	$\frac{Q \xrightarrow{n, \alpha} Q'}{P + Q \xrightarrow{n, r, \alpha} Q'} \quad \tau \notin \mathbb{I}(P)$
<u>Rel</u>	$\frac{P \xrightarrow{\alpha} P'}{P[f] \xrightarrow{f(\alpha)} P'[f]}$	Rel	$\frac{P \xrightarrow{m, \alpha} P'}{P[f] \xrightarrow{m, f(\alpha)} P'[f]}$
<u>Res</u>	$\frac{P \xrightarrow{\alpha} P'}{P \setminus L \xrightarrow{\alpha} P' \setminus L} \quad \alpha \notin L \cup \bar{L}$	Res	$\frac{P \xrightarrow{m, \alpha} P'}{P \setminus L \xrightarrow{m, \alpha} P' \setminus L} \quad \alpha \notin L \cup \bar{L}$
<u>Com1</u>	$\frac{P \xrightarrow{\alpha} P'}{P Q \xrightarrow{\alpha} P' Q}$	Com1	$\frac{P \xrightarrow{m, \alpha} P'}{P Q \xrightarrow{m, L, \alpha} P' Q} \quad \mathbb{I}_{[m]}(P) \cap \bar{\mathbb{I}}(Q) = \emptyset$
<u>Com2</u>	$\frac{Q \xrightarrow{\alpha} Q'}{P Q \xrightarrow{\alpha} P Q'}$	Com2	$\frac{Q \xrightarrow{n, \alpha} Q'}{P Q \xrightarrow{n, R, \alpha} P Q'} \quad \mathbb{I}_{[n]}(Q) \cap \bar{\mathbb{I}}(P) = \emptyset$
<u>Com3</u>	$\frac{P \xrightarrow{a} P' \quad Q \xrightarrow{\bar{a}} Q'}{P Q \xrightarrow{\tau} P' Q'}$	Com3	$\frac{P \xrightarrow{m, a} P' \quad Q \xrightarrow{n, \bar{a}} Q'}{P Q \xrightarrow{\langle m, L, n, R \rangle, \tau} P' Q'} \quad \begin{array}{l} \mathbb{I}_{[m]}(P) \cap \bar{\mathbb{I}}(Q) = \emptyset \text{ and} \\ \mathbb{I}_{[n]}(Q) \cap \bar{\mathbb{I}}(P) = \emptyset \end{array}$
<u>Con</u>	$\frac{P \xrightarrow{\alpha} P'}{C \xrightarrow{\alpha} P'} \quad C \stackrel{\text{def}}{=} P$	Con	$\frac{P \xrightarrow{m, \alpha} P'}{C \xrightarrow{m, \alpha} P'} \quad C \stackrel{\text{def}}{=} P$

Definition 4.1 (Prioritized Strong Bisimulation)

A symmetric relation $\mathcal{R} \subseteq \mathcal{P} \times \mathcal{P}$ is called prioritized strong bisimulation if for every $\langle P, Q \rangle \in \mathcal{R}$, $\gamma \in \mathcal{A}$, and $m \in \text{Loc}$ the following condition holds.

$$P \xrightarrow{m, \gamma} P' \text{ implies } \exists Q', n. Q \xrightarrow{n, \gamma} Q' \text{ and } \langle P', Q' \rangle \in \mathcal{R} .$$

We write $P \simeq Q$ if there exists a prioritized strong bisimulation \mathcal{R} such that $\langle P, Q \rangle \in \mathcal{R}$.

It is straightforward to establish that \simeq is the *largest* prioritized strong bisimulation and that \simeq is an equivalence relation. Unfortunately, \simeq is *not* a congruence which is a necessary requirement for an equivalence to be suitable for compositional reasoning. The lack of compositionality is demonstrated by the following example which presents the traditional view of process algebras that “parallelism = nondeterminism”. We have

$$a.\underline{b}.\mathbf{0} + \underline{b}.a.\mathbf{0} \simeq a.\mathbf{0} | \underline{b}.\mathbf{0}$$

but

$$(a.\underline{b}.\mathbf{0} + \underline{b}.a.\mathbf{0}) | \bar{\underline{b}}.\mathbf{0} \not\approx (a.\mathbf{0} | \underline{b}.\mathbf{0}) | \bar{\underline{b}}.\mathbf{0}$$

since the latter can do an a -transition while the corresponding a -transition of the former process is preempted because the right process in the summation can engage in a τ -transition.

The above observation is not surprising since the distribution of processes has influence on the preemption of transitions and, consequently, on the bisimulation. Thus, in order to find the largest congruence relation \simeq^+ contained in \simeq we have to take the *local* preemption of processes into account. However, it turns out that the locations themselves are not important. In the following, we define prioritized strong congruence \simeq^+ , and we show that it is indeed the largest congruence contained in \simeq .

Definition 4.2 (Prioritized Strong Congruence)

A symmetric relation $\mathcal{R} \subseteq \mathcal{P} \times \mathcal{P}$ is a prioritized strong congruence if for every $\langle P, Q \rangle \in \mathcal{R}$, $\alpha \in A$, $\underline{\alpha} \in \underline{A}$, and $m \in \text{Loc}$ the following conditions hold.

1. $P \xrightarrow{\underline{\alpha}} P'$ implies $\exists Q'. Q \xrightarrow{\underline{\alpha}} Q'$ and $\langle P', Q' \rangle \in \mathcal{R}$.
2. $P \xrightarrow{m, \alpha} P'$ implies $\exists Q', n. Q \xrightarrow{n, \alpha} Q'$, $\mathbb{I}_{[n]}(Q) \subseteq \mathbb{I}_{[m]}(P)$, and $\langle P', Q' \rangle \in \mathcal{R}$.

We write $P \simeq^+ Q$ if there exists a prioritized strong congruence \mathcal{R} such that $\langle P, Q \rangle \in \mathcal{R}$.

The difference between this definition and the definition of \simeq is the additional requirement concerning the initial action sets, parameterized with the appropriate location, in the condition for unprioritized transitions. Intuitively, the prioritized initial action set of a process with respect to some location, and not the location itself, is a measure of the preemptive power of the process relative to the specific location. Thus, the second condition of Definition 4.2 states that an unprioritized action α from some location m of the process P has to be matched by the same action from some location n of Q . Moreover, the preemptive power of Q with respect to n is at most as strong as the preemptive power of P with respect to m .

Proposition 4.3 *The relation \simeq^+ is a congruence, i.e. for all CCS^{prio} contexts $C[X]$ we have: $P \simeq^+ Q$ implies $C[P] \simeq^+ C[Q]$.*

The next theorem states the main result of this section.

Theorem 4.4 *The congruence \simeq^+ is the largest congruence contained in \simeq .*

4.1 Axiomatization of \simeq^+

In this section, we give an axiomatization of \simeq^+ for *finite* processes, i.e. processes that do not contain recursion. In order to develop the axiomatization, we need to add a new summation operator \oplus to the process algebra CCS^{prio} . This operator is called *distributed summation* and its semantics is similar to $+$ but priorities on different sides of the operator are considered as incomparable.

Definition 4.5 (Distributed Summation)

The semantics of the new binary operator \oplus on processes is defined by the following semantic rules.

$$\begin{array}{ll}
\underline{\text{iSum1}} \frac{P \xrightarrow{\alpha} P'}{P \oplus Q \xrightarrow{\alpha} P'} & \text{iSum1} \frac{P \xrightarrow{m,\alpha} P'}{P \oplus Q \xrightarrow{m,L,\alpha} P'} \\
\underline{\text{iSum2}} \frac{Q \xrightarrow{\alpha} Q'}{P \oplus Q \xrightarrow{\alpha} Q'} & \text{iSum2} \frac{Q \xrightarrow{n,\alpha} Q'}{P \oplus Q \xrightarrow{n,R,\alpha} Q'}
\end{array}$$

Table 3: Axiomatization of \simeq^+ (Part I)

(A1)	$x + y = y + x$	
(A2)	$x + (y + z) = (x + y) + z$	
(A3)	$x + x = x$	
(A4)	$x + \mathbf{0} = x$	
(iA1)	$x \oplus y = y \oplus x$	
(iA2)	$x \oplus (y \oplus z) = (x \oplus y) \oplus z$	
(iA3)	$x \oplus x = x$	
(iA4)	$x \oplus \mathbf{0} = x$	
(P)	$\underline{\tau}.x + \alpha.y = \underline{\tau}.x$	
(E)	$P \equiv \bigoplus_i \sum_j \gamma_{ij}.P_{ij}$ and $Q \equiv \bigoplus_k \sum_l \delta_{kl}.Q_{kl}$ implies $P \mid Q =$ $\bigoplus_i \sum_j (\gamma_{ij}.(P_{ij} \mid Q) + \sum_k \sum_l \{\tau.(P_{ij} \mid Q_{kl}) \mid \gamma_{ij} = \bar{\delta}_{kl}, \gamma_{ij}, \delta_{kl} \in A\}$ $\quad + \sum_k \sum_l \{\underline{\tau}.(P_{ij} \mid Q_{kl}) \mid \gamma_{ij} = \bar{\delta}_{kl}, \gamma_{ij}, \delta_{kl} \in \underline{A}\}) \oplus$ $\bigoplus_k \sum_l (\delta_{kl}.(P \mid P_{kl}) + \sum_i \sum_j \{\tau.(P_{ij} \mid Q_{kl}) \mid \gamma_{ij} = \bar{\delta}_{kl}, \gamma_{ij}, \delta_{kl} \in A\}$ $\quad + \sum_i \sum_j \{\underline{\tau}.(P_{ij} \mid Q_{kl}) \mid \gamma_{ij} = \bar{\delta}_{kl}, \gamma_{ij}, \delta_{kl} \in \underline{A}\})$	
(Res1)	$\mathbf{0} \setminus L = \mathbf{0}$	
(Res2)	$(\gamma.x) \setminus L = \mathbf{0}$	$(\gamma \in L \cup \bar{L})$
(Res3)	$(\gamma.x) \setminus L = \gamma.(x \setminus L)$	$(\gamma \notin L \cup \bar{L})$
(Res4)	$(x + y) \setminus L = (x \setminus L) + (y \setminus L)$	
(iRes4)	$(x \oplus y) \setminus L = (x \setminus L) \oplus (y \setminus L)$	
(Rel1)	$\mathbf{0}[f] = \mathbf{0}$	
(Rel2)	$(\gamma.x)[f] = f(\gamma).(x[f])$	
(Rel3)	$(x + y)[f] = x[f] + y[f]$	
(iRel3)	$(x \oplus y)[f] = x[f] \oplus y[f]$	

Now, we turn to the axioms for prioritized strong congruence. We write $\vdash_E P = Q$ if P can be rewritten to Q by using the axioms in the Tables 3 and 4. The Axioms (S2) and (S3) involve side conditions. The relation \sqsubseteq_i is the precongruence on finite processes generated from the axioms presented in Table 5 using the laws of inequational reasoning. The axioms in Table 3 are basically those presented in [13] augmented with the corresponding axioms for the incomparable summation operator. Moreover, the expansion axiom has been adapted for our algebra (cf. Axiom (E)). The axioms in Table 4 are new and show how we may ‘restructure’ locations. They deal with the *distributivity* of the summation operators (Axioms (D1), (D2), and (D3)), the *interchangeability* of the summation operators (Axioms (Ic1) and (Ic2)), and the *saturation* of locations (Axioms (S1), (S2), and (S3)), respectively.

Table 4: Axiomatization of \simeq^+ (Part II)

(D1)	$(x \oplus \alpha.y) + \beta.z = (x + \beta.z) \oplus \alpha.y$	
(D2)	$(x \oplus \underline{\alpha}.y) + \beta.z = x \oplus (x + \underline{\alpha}.y + \beta.z)$	
(D3)	$(x \oplus y) + \underline{\alpha}.z = (x + \underline{\alpha}.z) \oplus (y + \underline{\alpha}.z)$	
(Ic1)	$\underline{\alpha}.x \oplus \underline{\beta}.y = \underline{\alpha}.x + \underline{\beta}.y$	
(Ic2)	$(\underline{\alpha}.x + y) = (\underline{\alpha}.x + y) \oplus \underline{\alpha}.x$	
(S1)	$(x + \underline{\alpha}.y) \oplus (x' + \underline{\alpha}.y') = (x + \underline{\alpha}.y + \underline{\alpha}.y') \oplus (x' + \underline{\alpha}.y + \underline{\alpha}.y')$	
(S2)	$(x + \underline{\alpha}.z) \oplus (y + \underline{\alpha}.z) = (x + \underline{\alpha}.z) \oplus y$	($\vdash x \sqsubseteq_i y$)
(S3)	$x \oplus y = x + y$	($\vdash x =_i y$)

Table 5: Axiomatization of \sqsubseteq_i

(iC1)	$\underline{\alpha}.x \sqsubseteq_i \underline{\alpha}.y$
(iC2)	$\mathbf{0} \sqsubseteq_i \nu.x \quad \nu \in \mathcal{A} \setminus \{\underline{\tau}\}$
(iC3)	$\alpha.x \sqsubseteq_i \mathbf{0}$

Lemma 4.6 *Let $\vdash P \sqsubseteq_i Q$ for some processes $P, Q \in \mathcal{P}$. Then, $\mathbb{I}(P) \subseteq \mathbb{I}(Q)$ holds. Moreover, $\underline{\tau} \in \mathbb{I}(P)$ if and only if $\underline{\tau} \in \mathbb{I}(Q)$.*

We write $\vdash P =_i Q$ iff $\vdash P \sqsubseteq_i Q$ and $\vdash Q \sqsubseteq_i P$. Considering the meaning of the side conditions as made precise in Lemma 4.6, it is immediately clear that the Axioms (S2) and (S3) are sound.

In order to prove our axiomatization complete, we define a notion of *normal form* of processes which is based on the following definition.

Definition 4.7 (Summation Form)

A process $P \in \mathcal{P}$ is called in summation form if it has the form $P \equiv \bigoplus_{i=1}^m \sum_{j=1}^{n_i} \gamma_{ij}.P_{ij}$ where $m, n_i \in \mathbb{N}$ and the processes P_{ij} are again in summation form. Per definition, $\mathbf{0}$ is in summation form.

Intuitively, P is distributed throughout m incomparable locations which themselves consist of n_i comparable locations, $1 \leq i \leq m$. Now, we are able to define *normal forms*.

Definition 4.8 (Normal Form)

Let $P \equiv \bigoplus_{i=1}^m \sum_{j=1}^{n_i} \gamma_{ij}.P_{ij}$ be in summation form. We define $\underline{\gamma}_{i} =_{\text{df}} \{\gamma_{ij} \mid 1 \leq j \leq n_i\} \cap \underline{A}$. The process P is said to be in normal form if the following properties hold.*

1. $\emptyset \subset L \subseteq \mathbb{I}(P)$ implies $\exists i. \underline{\gamma}_{i*} = L$.
2. $\gamma_{ij} = \underline{\tau}$ and $\gamma_{kl} \in A$ imply $i \neq k$.
3. $\gamma_{ij} = \gamma_{kl} = \underline{\alpha}$ implies $\exists j'. P_{ij'} \equiv P_{kl}$ and $\gamma_{ij'} = \underline{\alpha}$.

4. $i \neq k$ implies $\underline{\gamma}_{i*} \neq \underline{\gamma}_{k*}$.

5. $\gamma_{ij}.P_{ij} \equiv \gamma_{kl}.P_{kl}$, $\gamma_{ij} \in A$, and $i \neq k$ imply $\underline{\gamma}_{i*} \not\subseteq \underline{\gamma}_{j*}$.

The following proposition plays a key role for the completeness proof of our axiomatization.

Proposition 4.9 *If P is a finite process, then there exists a normal form N such that $\vdash_E N = P$.*

Rewriting a process in its normal form means to restructure its locations. After this is done, standard techniques used in CCS (cf. [23]) can be applied in order to show our axiomatization complete.

The following theorem presents the main result of this section, namely that the axioms in the Tables 3, 4, and 5 equationally characterize \simeq^+ .

Theorem 4.10 (Soundness & Completeness)

For finite processes $P, Q \in \mathcal{P}$ we have: $\vdash_E P = Q$ iff $P \simeq^+ Q$.

4.2 Logical Characterization of \simeq^+

In this section, we provide a logical characterization of \simeq^+ . We adapt the well-known Hennessy-Milner Logic [23] by changing the semantics of the modal operators.

The syntax of the logic we use is defined by the following BNF where $L \subseteq \underline{A} \setminus \{\tau\}$.

$$\Phi ::= tt \mid \neg\Phi \mid \Phi \wedge \Psi \mid \langle \underline{\alpha} \rangle \Phi \mid \langle \alpha, L \rangle \Phi$$

The set of all formulae is denoted by \mathcal{F} and ranged over by Φ, Ψ, \dots . For notational convenience, we also introduce the following dual operators: $tt =_{\text{df}} \neg ff$, $\Phi \vee \Psi =_{\text{df}} \neg(\neg\Phi \wedge \neg\Psi)$, $[\underline{\alpha}]\Phi =_{\text{df}} \neg\langle \underline{\alpha} \rangle(\neg\Phi)$, and $[\alpha, L]\Phi =_{\text{df}} \neg\langle \alpha, L \rangle(\neg\Phi)$.

We define the satisfaction relation $\models \subseteq \mathcal{P} \times \mathcal{F}$ between processes and formulae inductively on the structure of formulae.

$$\begin{aligned} P &\models tt \\ P &\models \neg\Phi \quad \text{if} \quad \text{not } P \models \Phi \\ P &\models \Phi \wedge \Psi \quad \text{if} \quad P \models \Phi \text{ and } P \models \Psi \\ P &\models \langle \underline{\alpha} \rangle \Phi \quad \text{if} \quad \exists P' \in \mathcal{P}. P \xrightarrow{\underline{\alpha}} P' \text{ and } P' \models \Phi \\ P &\models \langle \alpha, L \rangle \Phi \quad \text{if} \quad \exists P' \in \mathcal{P}, m \in \text{Loc}. P \xrightarrow{m, \alpha} P', \mathbb{I}_{[m]}(P) \subseteq L, \text{ and } P' \models \Phi \end{aligned}$$

Intuitively, P satisfies $\langle \alpha, L \rangle \Phi$ if P possesses an α -transition from some location m to a process satisfying Φ . Moreover, the preemptive power of P with respect to m may be at most L .

The above defined logic characterizes prioritized strong congruence.

Theorem 4.11 (Characterization of \simeq^+)

Let $P, Q \in \mathcal{P}$. We have $P \simeq^+ Q$ if and only if $\{\Phi \in \mathcal{F} \mid P \models \Phi\} = \{\Phi \in \mathcal{F} \mid Q \models \Phi\}$.

5 Prioritized Observational Congruence

The semantic congruence developed in the previous section is too strong for verifying systems in practice. The prioritized strong congruence requires that two equivalent systems have to match exactly each others transitions, even those labeled with internal actions. Therefore, we want to abstract from internal actions and develop a semantic congruence from the point of view of an external observer. Our approach follows the lines of [26, 22]. We start off with the definition of a naive prioritized weak bisimulation which abstracts from internal actions. This relation is an adaption of observational equivalence [23].

Definition 5.1 (Naive Weak Transition Relation)

We define:

1. $\hat{\gamma} =_{\text{df}} \epsilon$ if $\gamma \in \{\tau, \tau\}$ and $\hat{\gamma} =_{\text{df}} \gamma$, otherwise.
2. $\xRightarrow{\epsilon}_{\times} =_{\text{df}} (\{ \xrightarrow{m, \tau} \mid m \in \text{Loc} \} \cup \{ \xrightarrow{\tau} \})^*$
3. $\xRightarrow{\alpha}_{\times} =_{\text{df}} \xRightarrow{\epsilon}_{\times} \circ \xrightarrow{\alpha} \circ \xRightarrow{\epsilon}_{\times}$
4. $\xRightarrow{m, \alpha}_{\times} =_{\text{df}} \xRightarrow{\epsilon}_{\times} \circ \xrightarrow{m, \alpha} \circ \xRightarrow{\epsilon}_{\times}$

Now, we define the *naive prioritized weak bisimulation* as follows.

Definition 5.2 (Naive Prioritized Weak Bisimulation)

A symmetric relation $\mathcal{R} \subseteq \mathcal{P} \times \mathcal{P}$ is a naive prioritized weak bisimulation if for every $\langle P, Q \rangle \in \mathcal{R}$, $\gamma \in \mathcal{A}$, and $m \in \text{Loc}$ the following condition holds.

$$P \xrightarrow{m, \gamma} P' \text{ implies } \exists Q', n. Q \xRightarrow{n, \hat{\gamma}}_{\times} Q' \text{ and } \langle P', Q' \rangle \in \mathcal{R} .$$

We write $P \approx_{\times} Q$ if there exists a naive prioritized weak bisimulation \mathcal{R} such that $\langle P, Q \rangle \in \mathcal{R}$.

Since no initial action sets are considered, it is easy to see that \approx_{\times} is no congruence. In order to get closer to our goal to define a prioritized observational congruence, we redefine the weak transition relation.

Definition 5.3 (Prioritized Weak Transition Relation)

We introduce the following notations where $L, M \subseteq \mathcal{A} \setminus \{\tau\}$.

1. $\hat{\tau} =_{\text{df}} \epsilon$ and $\hat{a} =_{\text{df}} a$.
 $\hat{\tau} =_{\text{df}} \epsilon$ and $\hat{a} =_{\text{df}} a$.
2. $P \xrightarrow{m, \alpha}_L P'$ iff $P \xrightarrow{m, \alpha} P'$ and $\mathbb{I}_{[m]}(P) \subseteq L$.
3. $\xRightarrow{\epsilon} =_{\text{df}} (\{ \xrightarrow{\tau} \} \cup \{ \xrightarrow{m, \tau} \mid m \in \text{Loc} \})^*$
4. $\xRightarrow{\alpha} =_{\text{df}} \xRightarrow{\epsilon} \circ \xrightarrow{\alpha} \circ \xRightarrow{\epsilon}$
5. $\xRightarrow{\epsilon}_L =_{\text{df}} (\{ \xrightarrow{\tau} \} \cup \{ \xrightarrow{m, \tau}_L \mid m \in \text{Loc} \})^*$
6. $P \xrightarrow{m, \alpha}_{L, M} P'$ iff $\exists P'', P'''. P \xrightarrow{\epsilon}_L P'' \xrightarrow{m, \alpha}_L P''' \xRightarrow{\epsilon} P'$ and $\mathbb{I}(P'') \subseteq M$.

Intuitively, $P \xrightarrow[L]{m, \alpha} P'$ means that P can evolve to P' by performing action α from location m and the preemptive power of P at location m is at most L . Recall that the prioritized initial action set of a process (with respect to a location) is a measure for its preemptive power. Actually, there are two slightly different perspectives of preemption which are encoded in the sets L and M in the definition of $P \xrightarrow[L, M]{m, \alpha} P'$, respectively. Whereas L is concerned with the influence of the environment, i.e. a parallel context, on actions performed on the path from P to P''' , the set M reflects the impact of P'' on potential synchronization partners (cf. Rule Com3). Note that the definition of $P \xrightarrow[L]{\epsilon} P'$ reflects our intuition that internal actions, and, therefore, their locations are unobservable. Additionally, a parallel context of P is not influenced by internal actions performed by P since priorities arising from different sides of the parallel operator are incomparable. Therefore, the parameter M is missing in the definition of the relation $\xrightarrow[L]{\epsilon}$.

Definition 5.4 (Prioritized Weak Bisimulation)

A symmetric relation $\mathcal{R} \subseteq \mathcal{P} \times \mathcal{P}$ is a prioritized weak bisimulation if for every $\langle P, Q \rangle \in \mathcal{R}$, $\alpha \in A$, $\underline{\alpha} \in \underline{A}$, and $m \in \text{Loc}$ the following conditions hold.

1. $\exists Q', Q''. Q \xrightarrow{\epsilon} Q'' \xrightarrow{\epsilon} Q'$, $\mathbb{I}(Q'') \subseteq \mathbb{I}(P)$, and $\langle P, Q' \rangle \in \mathcal{R}$.
2. $P \xrightarrow{\alpha} P'$ implies $\exists Q'. Q \xrightarrow{\hat{\alpha}} Q'$ and $\langle P', Q' \rangle \in \mathcal{R}$.
3. $P \xrightarrow[L, M]{m, \alpha} P'$ implies $\exists Q', n. Q \xrightarrow[L, M]{n, \hat{\alpha}} Q'$, $L = \mathbb{I}_{[m]}(P)$, $M = \mathbb{I}(P)$, and $\langle P', Q' \rangle \in \mathcal{R}$.

We write $P \underline{\cong} Q$ if there exists a prioritized weak bisimulation \mathcal{R} such that $\langle P, Q \rangle \in \mathcal{R}$.

From this definition, we may directly conclude that $\underline{\cong}$ is the *largest* prioritized weak bisimulation, and that \cong is an equivalence relation.

The first condition of Definition 5.4 guarantees that prioritized weak bisimulation is compositional with respect to the parallel operator. Its necessity is best illustrated by the following example. The processes $P \stackrel{\text{def}}{=} \underline{\tau}. \underline{a}. \mathbf{0}$ and $Q \stackrel{\text{def}}{=} \underline{a}. \mathbf{0}$ would be considered as equivalent if the first condition would be absent. However, the parallel context $C[X] \stackrel{\text{def}}{=} X | (\underline{b}. \mathbf{0} + b. \mathbf{0})$ is able to distinguish them since $C[P]$ can perform a b -transition while $C[Q]$ cannot perform a weak b -transition. This example is surprising because the transition relation of CCS^{prio} with respect to prioritized transitions is exactly defined as in CCS.

Proposition 5.5 *The equivalence relation $\underline{\cong}$ is a congruence relation with respect to all CCS^{prio} operators except the summation operator $+$, the distributed summation operator \oplus , and recursion.*

In contrast to [26], the summation fix presented in [23] is not sufficient in order to achieve a congruence relation. E.g., let $C \stackrel{\text{def}}{=} \underline{\tau}. D$ and $D \stackrel{\text{def}}{=} \tau. C$. Now, define $P \stackrel{\text{def}}{=} \tau. C$ and $Q \stackrel{\text{def}}{=} \underline{\tau}. D$. By Definition 5.4 we may observe $P \underline{\cong} Q$, but $P + a. \mathbf{0} \not\underline{\cong} Q + a. \mathbf{0}$ since the former can perform an a -action whereas the latter cannot. It turns out that we have to require that observationally congruent processes must have the same initial actions. This requirement is stronger than the first condition of Definition 5.4.

Definition 5.6 (Prioritized Observational Congruence)

We define $P \underline{\cong}^+ Q$ if for all $\alpha \in A$, $\underline{\alpha} \in \underline{A}$, and $m \in \text{Loc}$ the following conditions hold.

1. $\underline{\mathbb{I}}(P) = \underline{\mathbb{I}}(Q)$
2. $P \xrightarrow{\alpha} P'$ implies $\exists Q'. Q \xrightarrow{\alpha} Q'$ and $P' \cong Q'$.
3. $P \xrightarrow{m,\alpha} P'$ implies $\exists Q', n. Q \xrightarrow{L,M}^{n,\alpha} Q'$, $L = \underline{\mathbb{I}}_{[m]}(P)$, $M = \underline{\mathbb{I}}(P)$, and $P' \cong Q'$.
4. $Q \xrightarrow{\alpha} Q'$ implies $\exists P'. P \xrightarrow{\alpha} P'$ and $P' \cong Q'$.
5. $Q \xrightarrow{m,\alpha} Q'$ implies $\exists P', n. P \xrightarrow{L,M}^{n,\alpha} P'$, $L = \underline{\mathbb{I}}_{[m]}(Q)$, $M = \underline{\mathbb{I}}(Q)$, and $P' \cong Q'$.

Now, we are able to present the main result of this section.

Theorem 5.7 *The relation \cong^+ is the largest congruence contained in \approx_\times .*

We want to conclude this section with a remark on the logical characterization of \cong . Defining a suitable logic can be done by replacing the $\langle \alpha, L \rangle$ operators of the logic presented in Section 4.2 by new operators $\langle\langle \alpha, L, M \rangle\rangle$ for $M \subseteq \underline{A} \setminus \{\tau\}$ where a process $P \in \mathcal{P}$ satisfies the formula $\langle\langle \alpha, L, M \rangle\rangle \Phi$ if there exists a process $P' \in \mathcal{P}$ and a location $m \in \mathcal{Loc}$ such that $P \xrightarrow{L,M}^{m,\alpha} P'$ and $P' \models \Phi$. The operators $\langle \underline{\alpha} \rangle$ have also to be replaced by operators $\langle\langle \underline{\alpha} \rangle\rangle$ where $P \models \langle\langle \underline{\alpha} \rangle\rangle \Phi$ if there exists a process $P' \in \mathcal{P}$ such that $P \xrightarrow{\alpha} P'$ and $P' \models \Phi$. Moreover, the logic has to be extended by new operators $\uparrow_L \Phi$ in order to match the first requirement of Definition 5.4 where $P \models \uparrow_L \Phi$ if $\exists P', P''. P \xrightarrow{\epsilon} P'' \xrightarrow{\epsilon} P'$, $\underline{\mathbb{I}}(P'') \subseteq L$, and $P' \models \Phi$. With these definitions, a characterization of \cong can be done along the lines of [23].

6 Example

In this section, we demonstrate the usefulness of CCS^{prio} for the verification of distributed systems by a typical example which involves an architecture scheme that is realized in many of today's computers.

Our example system consists of some application which receives and writes data from two memory benches (cf. Figure 2). In order to improve the efficiency in a computer system each bench is connected to a direct-memory-access (DMA) controller.

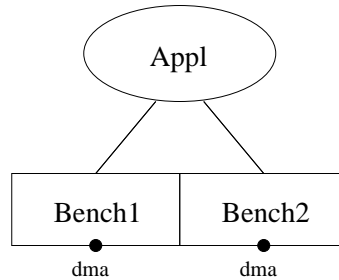


Figure 2: Example system

To overcome the low speed of most memory modules, the application **Appl** works alternately with each memory bench. We model **Appl** in CCS^{prio} by

$$\text{Appl} \stackrel{\text{def}}{=} \underline{\text{fetch1}}.\underline{\text{fetch2}}.\text{Appl} .$$

Each memory bench **Bench1** and **Bench2** is continuously able to serve the application or to allow the external DMA controller to access the memory (action **dma**). However, if a memory bench has to decide between both activities, then it chooses the former since the progress of the application is considered as more important. Consequently, we define

$$\mathbf{Bench1} \stackrel{\text{def}}{=} \overline{\mathbf{fetch1}}.\mathbf{Bench1} + \mathbf{dma}.\mathbf{Bench1}$$

and

$$\mathbf{Bench2} \stackrel{\text{def}}{=} \overline{\mathbf{fetch2}}.\mathbf{Bench2} + \mathbf{dma}.\mathbf{Bench2} .$$

Now, the overall system **Sys** is given by

$$\mathbf{Sys} \stackrel{\text{def}}{=} (\mathbf{App1} \mid \mathbf{Bench1} \mid \mathbf{Bench2}) \setminus \{\mathbf{fetch1}, \mathbf{fetch2}\} .$$

Since the application uses the memory cells alternately, the DMA is expected to be allowed to access the free memory bench. Therefore, the specification is simply $\mathbf{Spec} \stackrel{\text{def}}{=} \mathbf{dma}.\mathbf{Spec} .$

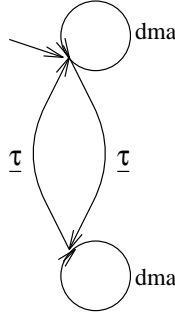


Figure 3: CCS^{prio} semantics of the example system

The CCS^{prio} semantics of **Sys** is given in Figure 3 where we abstract from the locations of the unprioritized action **dma**. It is easy to see that the symmetric closure of

$$\{ \langle \mathbf{Spec}, \mathbf{Sys} \rangle, \langle \mathbf{Spec}, (\overline{\mathbf{fetch2}}.\mathbf{App1} \mid \mathbf{Bench1} \mid \mathbf{Bench2}) \setminus \{\mathbf{fetch1}, \mathbf{fetch2}\} \rangle \}$$

is a prioritized weak bisimulation. Note that Condition (1) of Definition 5.4 is trivially satisfied since **Spec** and **Sys** do not contain any visible prioritized actions. Therefore, we obtain $\mathbf{Spec} \cong \mathbf{Sys}$ as expected. However, in the traditional approach [13, 26] the **dma**-loops in the labeled transition system of **Sys** would be missing, and **Sys** would not be observationally equivalent to **Spec**. In other words, the traditional approach would not allow a memory bench to engage in a communication, e.g. with a DMA controller, at any time although the application does not need the bench at all moments. This example shows that traditional approaches to priorities in process algebras which consider a global notion of preemption have undesired side-effects and are often not useful for modeling and verifying distributed systems.

7 Discussion and Related Work

Several approaches have been made in order to extend traditional process algebras with priorities. They differ in their objectives which priority aspects of real systems or programming languages to model. The objectives arise from modeling *interrupt systems* [3], from programming language constructs like the **PRIALT**-construct in **occam** [10, 20], and from efforts to combine real-time approaches with priorities [9, 16].

Another approach for extending the process algebra CCS [23] has been proposed in [13] where priorities are assigned to actions in a *globally dynamic* way, i.e. in one state of a system action α may have priority over action β while the situation may be converse in another state of the system. For that process algebra a complete semantic theory has been developed in an analogous fashion to [23] which includes congruences based on strong and weak bisimulation and their axiomatic characterizations [26].

The difference of most above mentioned process algebras with respect to their expressiveness of priority constraints has been studied in [31].

Our process algebra CCS^{prio} is based on the approach in [13, 26] where we adopt all design decisions except the notion of *global* preemption. Therefore, CCS^{prio} has the following characteristics. Only transitions labeled by complementary actions with the same priority may engage in a synchronization. As in [13], we consider actions with different priorities as *different* channels. This is sufficient for most cases occurring in practice [14] and avoids that priorities values have to be adjusted in case of communication. Existing approaches to priorities often take the sum of the priorities of the sending and receiving action as the priority of the resulting synchronization action [16, 17]. Thus, prioritized α -actions may always have priority over unprioritized α -actions, even if the system is composed with another concurrent system, since relative priorities are preserved. However, that approach, mostly considered in CSP-based frameworks, is inconvenient. The more parallel components participate in a synchronization, the higher the priority of a synchronized action becomes. Other adjustments, e.g. taking the minimum or maximum priority of the communicating actions, are also possible and well motivated for some examples.

The strong relation of CCS^{prio} to the process algebra proposed in [13, 26] can be made precise by the following fact. If we globalize preemption in our framework by defining $[m] =_{\text{af}} \text{Loc}$ for all $m \in \text{Loc}$, our operational semantics and our behavioral relations reduce to the corresponding notions presented in [13, 26].

For a comparison with our work it is of importance that all the above mentioned traditional approaches are provided with a semantics which deals with *global* preemption. In contrast, we consider a notion of *localized* preemption. This idea is also presented in [17], where a CSP-based language is extended with priorities. However, this process algebra suffers from a complicated semantics, especially for the hiding operator. The authors only conjecture that their strong bisimulation is a congruence. They do not provide an axiomatization for their equivalence and do not present a theory for observational congruence.

We want to remark on our notion of strong and weak bisimulation. Since our semantic theory reflects local preemption, locations are implicitly occurring in our semantic equivalences. However, in contrast to [1, 8], where CCS has been syntactically extended by locations and a bisimulation called *observational equivalence* respecting localities has been developed, locations are not explicitly considered in our bisimulations. Our objective is not to observe locations but to observe local preemption which is necessary for causal reasoning in process algebras with priorities. However, if we abstract from priorities, our *prioritized observational equivalence* lies in between observational equivalence [23] and location equivalence.

8 Conclusions and Future Work

We have presented a process algebra with distributed priorities, called CCS^{prio} , which is more suitable for modeling and verifying distributed systems than traditional approaches. The key idea for this algebra is to take the distribution of the considered system, i.e. the location of

actions, into account for defining a notion of *localized* preemption. We have developed a semantic theory for this algebra and have shown its suitability by an example which models a direct-memory-access scheme. However, it remains to show how our prioritized bisimulations can be computed efficiently before implementing CCS^{prio} in an automated verification tool [15]. In order to apply standard algorithms [21, 27] the bisimulations have to be characterized using a transition relation that is not parameterized with prioritized initial action sets. Moreover, developing an axiomatization of prioritized observational congruence would help for comparing it with other behavioral relations such as location equivalence [1, 8].

We have seen that the introduction of priorities in process algebras provides a motivation for considering the distribution of systems and, hereby, gives a strong argument for rejecting the view “parallelism = nondeterminism” when dealing with priorities. Having done the first step towards a non-interleaving semantics, it is worth investigating how our notion of *localized* preemption goes along with traditional non-interleaving semantics like *transition systems with independence*, *asynchronous transition systems* [33] or Petri Net semantics [6, 32]. For the former model, a bisimulation has already been proposed that coincides with location equivalence for special classes of processes [25]. Because of the relationship between our prioritized weak bisimulation and location equivalence mentioned above, this direction of research is especially interesting.

References

- [1] L. Aceto. A static view of localities. *Formal Aspects of Computing*, 6(2):201–222, 1994.
- [2] J. Leach Albert, B. Monien, and M. Rodríguez Artalejo, editors. *Automata, Languages and Programming (ICALP '91)*, volume 510 of *Lecture Notes in Computer Science*, Madrid, July 1991. Springer-Verlag.
- [3] J.C.M. Baeten, J.A. Bergstra, and J.W. Klop. Syntax and defining equations for an interrupt mechanism in process algebra. *Fundamenta Informaticae IX*, pages 127–168, 1986.
- [4] J.C.M. Baeten and J.W. Klop, editors. *CONCUR '90*, volume 458 of *Lecture Notes in Computer Science*, Amsterdam, August 1990. Springer-Verlag.
- [5] J.C.M. Baeten and W.P. Weijland. *Process Algebra*, volume 18 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press, Cambridge, England, 1990.
- [6] E. Best and M. Koutny. Petri net semantics of priority systems. *Theoretical Computer Science*, 96:175–215, 1992.
- [7] R.N. Bol and J.F. Groote. The meaning of negative premises in transition system specifications. In Albert et al. [2], pages 481–494.
- [8] G. Boudol, I. Castellani, M. Hennessy, and A. Kiehn. Observing localities. *Theoretical Computer Science*, 114(1):31–61, June 1993.
- [9] P. Brémonte-Grégoire, I. Lee, and R. Gerber. ACSR: An algebra of communicating shared resources with dense time and priorities. In E. Best, editor, *CONCUR '93*, volume 715 of *Lecture Notes in Computer Science*, pages 417–431, Hildesheim, Germany, August 1993. Springer-Verlag.
- [10] J. Camilleri and G. Winskel. CCS with priority choice. In *Sixth Annual Symposium on Logic in Computer Science (LICS '91)*, pages 246–255, Amsterdam, July 1991. IEEE Computer Society Press.
- [11] R. Cleaveland, editor. *CONCUR '92*, volume 630 of *Lecture Notes in Computer Science*, Stony Brook, New York, August 1992. Springer-Verlag.
- [12] R. Cleaveland. Analyzing concurrent systems using the Concurrency Workbench. In P.E. Lauer, editor, *Functional Programming, Concurrency, Simulation and Automated Reasoning*, volume 693 of *Lecture Notes in Computer Science*, pages 129–144. Springer-Verlag, 1993.
- [13] R. Cleaveland and M.C.B. Hennessy. Priorities in process algebra. *Information and Computation*, 87(1/2):58–77, July/August 1990.

- [14] R. Cleaveland, G. Lüttgen, V. Natarajan, and S. Sims. Priorities for modeling and verifying distributed systems. In T. Margaria and B. Steffen, editors, *Second International Workshop on Tools and Algorithms for the Construction and Analysis of Systems (TACAS '96)*, volume 1055 of *Lecture Notes in Computer Science*, pages 278–297, Passau, Germany, March 1996. Springer-Verlag.
- [15] R. Cleaveland, J. Parrow, and B. Steffen. The Concurrency Workbench: A semantics-based tool for the verification of finite-state systems. *ACM Transactions on Programming Languages and Systems*, 15(1):36–72, January 1993.
- [16] R. Gerber and I. Lee. CCSR: A calculus for communicating shared resources. In Baeten and Klop [4], pages 263–277.
- [17] H. Hansson and F. Orava. A process calculus with incomparable priorities. In *Proceedings of the North American Process Algebra Workshop*, Workshops in Computing, pages 43–64, Stony Brook, New York, August 1992. Springer-Verlag.
- [18] M.C.B. Hennessy. *Algebraic Theory of Processes*. MIT Press, Boston, 1988.
- [19] C.A.R. Hoare. *Communicating Sequential Processes*. Prentice-Hall, London, 1985.
- [20] C.-T. Jensen. *Prioritized and Independent Actions in Distributed Computer Systems*. PhD thesis, Aarhus University, August 1994.
- [21] P. Kanellakis and S.A. Smolka. CCS expressions, finite state processes, and three problems of equivalence. *Information and Computation*, 86(1):43–68, May 1990.
- [22] R. Milner. *A Calculus of Communicating Systems*, volume 92 of *Lecture Notes in Computer Science*. Springer-Verlag, Berlin, 1980.
- [23] R. Milner. *Communication and Concurrency*. Prentice-Hall, London, 1989.
- [24] U. Montanari and D. Yankelevich. A parametric approach to localities. In W. Kuich, editor, *Automata, Languages and Programming (ICALP '92)*, volume 623 of *Lecture Notes in Computer Science*, pages 617–628, Vienna, July 1992. Springer-Verlag.
- [25] M. Mukund and M. Nielsen. CCS, locations and asynchronous transition systems. In *Proceedings of FST & TCS '92*, volume 652 of *Lecture Notes in Computer Science*, pages 328–341. Springer-Verlag, 1992.
- [26] V. Natarajan, L. Christoff, I. Christoff, and R. Cleaveland. Priorities and abstraction in process algebra. In P.S. Thiagarajan, editor, *Foundations of Software Technology and Theoretical Computer Science*, volume 880 of *Lecture Notes in Computer Science*, pages 217–230, Madras, India, December 1994. Springer-Verlag.
- [27] R. Paige and R.E. Tarjan. Three partition refinement algorithms. *SIAM Journal of Computing*, 16(6):973–989, December 1987.
- [28] D.M.R. Park. Concurrency and automata on infinite sequences. In *Proceedings of 5th G.I. Conference on Theoretical Computer Science*, volume 104 of *Lecture Notes in Computer Science*, pages 167–183. Springer-Verlag, 1980.
- [29] G.D. Plotkin. A structural approach to operational semantics. Technical Report DAIMI-FN-19, Computer Science Department, Aarhus University, Denmark, 1981.
- [30] D. Sangiorgi and R. Milner. The problem of ‘weak bisimulation up to’. In Cleaveland [11], pages 32–46.
- [31] S.A. Smolka and B. Steffen. Priority as extremal probability. In Baeten and Klop [4], pages 456–466.
- [32] R. van Glabbeek and F. Vaandrager. Petri net models for algebraic theories of concurrency. In J.W. Bakker, A.J. Nijman, and P.C. Treleaven, editors, *Proceedings PARLE conference, Eindhoven, Vol. II (Parallel Languages)*, volume 259 of *Lecture Notes in Computer Science*, pages 224–242. Springer-Verlag, 1987.
- [33] G. Winskel and M. Nielsen. Models for concurrency. In S. Abramsky, D.M. Gabbay, and T.S.E. Maibaum, editors, *Handbook of Logic Computer Science*, volume 4, pages 1–148. Oxford Science Publications, 1995.

A Proofs

In the appendix, we prove the most interesting cases of the above propositions and theorems. It includes especially the *largest congruence* proofs of the *prioritized strong congruence* \simeq^+ and the *prioritized weak congruence* $\underline{\simeq}^+$ for CCS^{prio} . Moreover, we prove the completeness of the axiomatization of \simeq^+ . Due to tedious case distinctions the proofs are most often not complete. However, we give all non-standard parts in full detail. The other parts follow exactly along the lines of the corresponding proofs in [23].

A.1 Proof of Proposition 4.3

Here, we present an outline of the proof that \simeq^+ is a congruence. The most cases of the proof are standard (cf. [22]) or especially long; e.g. the compositionality proof of \simeq^+ with respect to the recursion operator requires the introduction of *prioritized strong bisimulation up to* similar to the according notion presented in [22].

In the remainder, we restrict ourselves to prove the compositionality of \simeq^+ with respect to summation and parallel composition. The following lemma is useful for the congruence proofs of \simeq^+ and $\underline{\simeq}^+$ with respect to summation. Therefore, we present it slightly more general than it is necessary for proving Proposition 4.3.

Lemma A.1 *Let $P, Q \in \mathcal{P}$ and $m, n \in \text{Loc}$. Then the following properties hold for all $R \in \mathcal{P}$ and $o \in \text{Loc}$.*

1. $P \simeq^+ Q$ implies $\mathbb{I}(P) = \mathbb{I}(Q)$
2. $\mathbb{I}_{[n]}(Q) \subseteq \mathbb{I}_{[m]}(P)$ implies $\mathbb{I}_{[n \cdot l]}(Q + R) \subseteq \mathbb{I}_{[m \cdot l]}(P + R)$
3. $\mathbb{I}(Q) \subseteq \mathbb{I}(P)$ implies $\mathbb{I}_{[o \cdot r]}(Q + R) \subseteq \mathbb{I}_{[o \cdot r]}(P + R)$
4. $\mathbb{I}(Q) \subseteq \mathbb{I}(P)$ implies $\mathbb{I}(Q + R) \subseteq \mathbb{I}(P + R)$.

Proof:

1. This part follows directly from the definition of \simeq^+ .
- 2.

$$\begin{aligned}
 & \mathbb{I}_{[n \cdot l]}(Q + R) \\
 \text{(Def. 3.1)} \quad &= \mathbb{I}_{[n]}(Q) \cup \mathbb{I}(R) \\
 \text{(premise)} \quad &\subseteq \mathbb{I}_{[m]}(P) \cup \mathbb{I}(R) \\
 \text{(Def. 3.1)} \quad &= \mathbb{I}_{[m \cdot l]}(P + R)
 \end{aligned}$$

- 3.

$$\begin{aligned}
 & \mathbb{I}_{[o \cdot r]}(Q + R) \\
 \text{(Def. 3.1)} \quad &= \mathbb{I}(Q) \cup \mathbb{I}_{[o]}(R) \\
 \text{(premise)} \quad &\subseteq \mathbb{I}(P) \cup \mathbb{I}_{[o]}(R) \\
 \text{(Def. 3.1)} \quad &= \mathbb{I}_{[o \cdot r]}(P + R)
 \end{aligned}$$

4.

$$\begin{aligned}
& \underline{\mathbb{I}}(Q + R) \\
(\text{def. of } \underline{\mathbb{I}}(\cdot)) &= \underline{\mathbb{I}}(Q) \cup \underline{\mathbb{I}}(R) \\
(\text{premise}) &\subseteq \underline{\mathbb{I}}(P) \cup \underline{\mathbb{I}}(R) \\
(\text{def. of } \underline{\mathbb{I}}(\cdot)) &= \underline{\mathbb{I}}(P + R)
\end{aligned}$$

□

Now, we are able to prove the following proposition.

Proposition A.2 $P \simeq^+ Q$ implies $P + R \simeq^+ Q + R$ for all processes R .

Proof: According to the definition of \simeq^+ it is sufficient to prove that

$$\mathcal{R} =_{\text{df}} \{ \langle P + R, Q + R \rangle \mid P \simeq^+ Q \} \cup \text{Id}$$

is a prioritized strong congruence where Id is the identity relation on \mathcal{P} .

Let $P + R \xrightarrow{p, \alpha} V$ for some $V \in \mathcal{P}$ and $p \in \mathcal{L}oc$. We have to show the existence of some $W \in \mathcal{P}$ and $q \in \mathcal{L}oc$ such that $Q + R \xrightarrow{q, \alpha} W$ and $\langle V, W \rangle \in \mathcal{R}$. Therefore, consider the following case distinction according to the operational rules for summation.

1. *Case.* $V \equiv P'$ and $p = m \cdot l$.

Here, we conclude

$$\begin{aligned}
& P + R \xrightarrow{m \cdot l, \alpha} P' \\
(\text{Sum1}) & \text{ implies } P \xrightarrow{m, \alpha} P' \text{ and } \tau \notin \underline{\mathbb{I}}(R) \\
(P \simeq^+ Q) & \text{ implies } \exists Q', n. Q \xrightarrow{n, \alpha} Q' \text{ and } P' \simeq^+ Q' \text{ and} \\
& \underline{\mathbb{I}}_{[n]}(Q) \subseteq \underline{\mathbb{I}}_{[m]}(P) \text{ and } \tau \notin \underline{\mathbb{I}}(R) \\
(\text{Sum1, def. of } \mathcal{R}, & \text{ implies } \exists Q', n. Q + R \xrightarrow{n \cdot l, \alpha} Q' \text{ and } \langle P', Q' \rangle \in \mathcal{R}, \text{ and} \\
\text{La. A.1(2)}) & \underline{\mathbb{I}}_{[n \cdot l]}(Q + R) \subseteq \underline{\mathbb{I}}_{[m \cdot l]}(P + R) .
\end{aligned}$$

Now, we obtain our claim by choosing $W =_{\text{df}} Q'$ and $q =_{\text{df}} n \cdot l$.

2. *Case.* $V \equiv R'$ and $p = o \cdot r$.

In this case, we observe

$$\begin{aligned}
& P + R \xrightarrow{o \cdot r, \alpha} R' \\
(\text{Sum2}) & \text{ implies } R \xrightarrow{o, \alpha} R' \text{ and } \tau \notin \underline{\mathbb{I}}(P) \\
(P \simeq^+ Q) & \text{ implies } R \xrightarrow{o, \alpha} R' \text{ and } \tau \notin \underline{\mathbb{I}}(Q) \\
(\text{Sum2, def. of } \mathcal{R}, & \text{ implies } Q + R \xrightarrow{o \cdot r, \alpha} R' \text{ and } \langle R', R' \rangle \in \mathcal{R}, \text{ and} \\
\text{La. A.1(1) and (3)}) & \underline{\mathbb{I}}_{[o \cdot r]}(Q + R) = \underline{\mathbb{I}}_{[o \cdot r]}(P + R) .
\end{aligned}$$

By choosing $W =_{\text{df}} R'$ and $q =_{\text{df}} o \cdot r$, we have finished the proof.

The above argument is sufficient to prove the second condition of the definition of \simeq^+ . The first condition is standard since the semantics of the summation operator with respect to prioritized transitions coincides with that of CCS. \square

For proving the compositionality of \simeq^+ with respect to parallel composition we need the following lemma.

Lemma A.3 *Let $P, Q, R \in \mathcal{P}$. Then the following properties hold for all $m, n, o \in \text{Loc}$.*

1. $P \simeq^+ Q$ implies $\mathbb{I}(P) = \mathbb{I}(Q)$.
2. $\mathbb{I}_{[n]}(Q) \subseteq \mathbb{I}_{[m]}(P)$ implies $\mathbb{I}_{[n \cdot L]}(Q \mid R) \subseteq \mathbb{I}_{[m \cdot L]}(P \mid R)$.
3. $\mathbb{I}_{[o \cdot R]}(Q \mid R) = \mathbb{I}_{[o \cdot R]}(P \mid R)$
4. $\mathbb{I}_{[n]}(Q) \subseteq \mathbb{I}_{[m]}(P)$ implies $\mathbb{I}_{[\langle n \cdot L, o \cdot R \rangle]}(Q \mid R) \subseteq \mathbb{I}_{[\langle m \cdot L, o \cdot R \rangle]}(P \mid R)$.
5. $\mathbb{I}(Q) \subseteq \mathbb{I}(P)$ implies $\mathbb{I}(Q \mid R) \subseteq \mathbb{I}(P \mid R)$.

Proof:

1. This property is an immediate consequence of Lemma A.1(1).
- 2.

$$\begin{aligned} & \mathbb{I}_{[n \cdot L]}(Q \mid R) \\ (\text{Def. 3.1}) &= \mathbb{I}_{[n]}(Q) \\ (\text{premise}) &\subseteq \mathbb{I}_{[m]}(P) \\ (\text{Def. 3.1}) &= \mathbb{I}_{[m \cdot L]}(P \mid R) \end{aligned}$$

- 3.

$$\begin{aligned} & \mathbb{I}_{[o \cdot R]}(Q \mid R) \\ (\text{Def. 3.1}) &= \mathbb{I}_{[o]}(R) \\ (\text{Def. 3.1}) &= \mathbb{I}_{[o \cdot R]}(P \mid R) \end{aligned}$$

- 4.

$$\begin{aligned} & \mathbb{I}_{[\langle n \cdot L, o \cdot R \rangle]}(Q \mid R) \\ (\text{Def. 3.1}) &= \mathbb{I}_{[n]}(Q) \cup \mathbb{I}_{[o]}(R) \\ (\text{premise}) &\subseteq \mathbb{I}_{[m]}(P) \cup \mathbb{I}_{[o]}(R) \\ (\text{Def. 3.1}) &= \mathbb{I}_{[\langle m \cdot L, o \cdot R \rangle]}(P \mid R) \end{aligned}$$

5.

$$\begin{aligned}
& \underline{\mathbb{I}}(Q \mid R) \\
(\text{def. of } \underline{\mathbb{I}}(\cdot)) &= \underline{\mathbb{I}}(Q) \cup \underline{\mathbb{I}}(R) \\
(\text{premise}) &\subseteq \underline{\mathbb{I}}(P) \cup \underline{\mathbb{I}}(R) \\
(\text{def. of } \underline{\mathbb{I}}(\cdot)) &= \underline{\mathbb{I}}(P \mid R)
\end{aligned}$$

□

Now, we can prove the following.

Proposition A.4 $P \simeq^+ Q$ implies $P \mid R \simeq^+ Q \mid R$ for all processes $R \in \mathcal{P}$.

Proof: According to the definition of \simeq^+ it is sufficient to prove that

$$\mathcal{R} =_{\text{df}} \{ \langle P \mid R, Q \mid R \rangle \mid P \simeq^+ Q \}$$

is a prioritized strong congruence.

Let $P \mid R \xrightarrow{p, \alpha} V$ for some $V \in \mathcal{P}$ and $p \in \mathcal{Loc}$. We have to show the existence of some $W \in \mathcal{P}$ and $q \in \mathcal{Loc}$ such that $Q \mid R \xrightarrow{q, \alpha} W$, $\underline{\mathbb{I}}_{[q]}(Q \mid R) \subseteq \underline{\mathbb{I}}_{[p]}(P \mid R)$, and $\langle V, W \rangle \in \mathcal{R}$. Consider the following case distinction according to the operational rules for parallel composition.

1. *Case.* $V \equiv P' \mid R$ and $p = m \cdot L$. We conclude

$$\begin{aligned}
& P \mid R \xrightarrow{m \cdot L, \alpha} P' \mid R \\
(\text{Com1}) & \text{ implies } P \xrightarrow{m, \alpha} P' \text{ and } \underline{\mathbb{I}}_{[m]}(P) \cap \overline{\mathbb{I}}(R) = \emptyset \\
(P \simeq^+ Q) & \text{ implies } \exists Q', n. Q \xrightarrow{n, \alpha} Q' \text{ and } P' \simeq^+ Q' \text{ and} \\
& \underline{\mathbb{I}}_{[n]}(Q) \subseteq \underline{\mathbb{I}}_{[m]}(P) \text{ and} \\
& \underline{\mathbb{I}}_{[m]}(P) \cap \overline{\mathbb{I}}(R) = \emptyset \\
(\text{set theory}) & \text{ implies } \exists Q', n. Q \xrightarrow{n, \alpha} Q' \text{ and } P' \simeq^+ Q' \text{ and} \\
& \underline{\mathbb{I}}_{[n]}(Q) \subseteq \underline{\mathbb{I}}_{[m]}(P) \text{ and} \\
& \underline{\mathbb{I}}_{[n]}(Q) \cap \overline{\mathbb{I}}(R) = \emptyset \\
(\text{Com1,} & \text{ implies } \exists Q' \mid R, n. Q \mid R \xrightarrow{n \cdot L, \alpha} Q' \mid R \text{ and} \\
\text{def. of } \mathcal{R}, & \langle P' \mid R, Q' \mid R \rangle \in \mathcal{R} \text{ and} \\
\text{La. A.3(2)}) & \underline{\mathbb{I}}_{[n \cdot L]}(Q \mid R) \subseteq \underline{\mathbb{I}}_{[m \cdot L]}(P \mid R) .
\end{aligned}$$

By choosing $W =_{\text{df}} Q' \mid R$ and $q =_{\text{df}} n \cdot L$, we are done.

2. *Case.* $V \equiv P \mid R'$ and $p = o \cdot R$.

Here, we observe

$$\begin{aligned}
& P \mid R \xrightarrow{o \cdot R, \alpha} P \mid R' \\
\text{(Com2)} \quad & \text{implies } R \xrightarrow{o, \alpha} R' \text{ and } \underline{\mathbb{I}}_{[o]}(R) \cap \overline{\mathbb{I}}(P) = \emptyset \\
\text{(Lemma A.3(1))} \quad & \text{implies } R \xrightarrow{o, \alpha} R' \text{ and } \underline{\mathbb{I}}_{[o]}(R) \cap \overline{\mathbb{I}}(Q) = \emptyset \\
\text{(Com2,} & \text{implies } Q \mid R \xrightarrow{o \cdot R, \alpha} Q \mid R' \text{ and} \\
\text{def. of } \mathcal{R}, & \langle P \mid R', Q \mid R' \rangle \in \mathcal{R} \text{ and} \\
\text{La. A.3(3))} & \underline{\mathbb{I}}_{[o \cdot R]}(P \mid R) = \underline{\mathbb{I}}_{[o \cdot R]}(Q \mid R) .
\end{aligned}$$

Our claim follows by choosing $W =_{\text{df}} Q \mid R'$ and $q =_{\text{df}} o \cdot R$.

3. *Case.* $V \equiv P' \mid R'$ and $p = \langle m \cdot L, o \cdot R \rangle$.

In this case, we establish

$$\begin{aligned}
& P \mid R \xrightarrow{\langle m \cdot L, o \cdot R \rangle, \tau} P' \mid R' \\
\text{(Com3)} \quad & \text{implies } P \xrightarrow{m, a} P' \text{ and } \underline{\mathbb{I}}_{[m]}(P) \cap \overline{\mathbb{I}}(R) = \emptyset \text{ and} \\
& R \xrightarrow{o, \bar{a}} R' \text{ and } \underline{\mathbb{I}}_{[o]}(R) \cap \overline{\mathbb{I}}(P) = \emptyset \\
(P \simeq^+ Q) \quad & \text{implies } \exists Q', n. Q \xrightarrow{n, a} Q' \text{ and } P' \simeq^+ Q' \text{ and} \\
& \underline{\mathbb{I}}_{[n]}(Q) \subseteq \underline{\mathbb{I}}_{[m]}(P) \text{ and} \\
& \underline{\mathbb{I}}_{[m]}(P) \cap \overline{\mathbb{I}}(R) = \emptyset \text{ and } R \xrightarrow{o, \bar{a}} R' \text{ and} \\
& \underline{\mathbb{I}}_{[o]}(R) \cap \overline{\mathbb{I}}(P) = \emptyset \\
\text{(set theory \& La. A.3(1))} \quad & \text{implies } \exists Q', n. Q \xrightarrow{n, a} Q' \text{ and } \underline{\mathbb{I}}_{[n]}(Q) \subseteq \underline{\mathbb{I}}_{[m]}(P) \text{ and} \\
& \underline{\mathbb{I}}_{[n]}(Q) \cap \overline{\mathbb{I}}(R) = \emptyset \text{ and } R \xrightarrow{o, \bar{a}} R' \text{ and} \\
& \underline{\mathbb{I}}_{[o]}(R) \cap \overline{\mathbb{I}}(Q) = \emptyset \text{ and } P' \simeq^+ Q' \\
\text{(Com3,} & \text{implies } \exists Q' \mid R', n. Q \mid R \xrightarrow{\langle n \cdot L, o \cdot R \rangle, \tau} Q' \mid R' \\
\text{def. of } \mathcal{R}, & \text{and } \langle P' \mid R', Q' \mid R' \rangle \in \mathcal{R} \text{ and} \\
\text{La. A.3(4))} & \underline{\mathbb{I}}_{[\langle n \cdot L, o \cdot R \rangle]}(Q \mid R) \subseteq \underline{\mathbb{I}}_{[\langle m \cdot L, o \cdot R \rangle]}(P \mid R) .
\end{aligned}$$

According to the definition of \mathcal{R} and by defining $W =_{\text{df}} Q' \mid R'$ and $q =_{\text{df}} \langle n \cdot L, o \cdot R \rangle$, we have finished the proof.

Proving the first condition of Definition 4.2 is standard since the semantics of the parallel operator with respect to prioritized transitions coincides with those of CCS. \square

A.2 Proof of Theorem 4.4

In order for the processes used in the proof of Theorem 4.4 to be well-defined, we present the following lemma.

Lemma A.5 (Finite Sorts)

Let $P \in \mathcal{P}$ be a CCS^{prio} process. Then the unprioritized sort of P , i.e. the set of unprioritized

actions occurring in the semantics of P , and the prioritized sort of P , i.e. the set of prioritized actions occurring in the semantics of P , are finite. We denote the unprioritized sort of P by $\mathcal{S}(P)$ and the prioritized sort of P by $\underline{\mathcal{S}}(P)$.

This observation is an immediate consequence of the fact that process terms are finite, and relabelings f satisfy the condition $|\{\gamma \mid f(\gamma) \neq \gamma\}| < \infty$.

Now, we prove Theorem 4.4 which states that the congruence \simeq^+ is the *largest* congruence contained in \simeq .

Proof: The largest congruence contained in \simeq is characterized as the relation $\simeq_a \subseteq \mathcal{P} \times \mathcal{P}$ which satisfies for all $P, Q \in \mathcal{P}$:

$$P \simeq_a Q \text{ iff } \forall \text{CCS}^{\text{prio}}\text{-contexts } C. C[P] \simeq C[Q] .$$

Since \simeq^+ is a congruence, we know that $\simeq^+ \subseteq \simeq_a$. In order to prove the inclusion $\simeq_a \subseteq \simeq^+$, it suffices to show that

$$\mathcal{R} =_{\text{df}} \{\langle P, Q \rangle \mid C_{PQ}[P] \simeq C_{PQ}[Q]\}$$

is a prioritized strong congruence for some CCS^{prio} -context $C[X]$. For our purposes we define $C_{PQ}[X] =_{\text{df}} X \mid H_{PQ}$ for $P, Q \in \mathcal{P}$ where

$$H_{PQ} \stackrel{\text{def}}{=} \sum_{L \subseteq \underline{\mathcal{S}}(P) \cup \underline{\mathcal{S}}(Q)} \tau.(\underline{d}_L.H_{PQ} + D_L)$$

and

$$D_L =_{\text{df}} \sum_{\underline{c} \in L} \underline{c}.0 .$$

Note that H_{PQ} is well-defined because of Lemma A.5. We assume that $\underline{d}_L \notin \mathcal{S}(P) \cup \underline{\mathcal{S}}(P) \cup \mathcal{S}(Q) \cup \underline{\mathcal{S}}(Q)$. Such \underline{d}_L 's exist because the sort and the prioritized sort of a process are always finite according to Lemma A.5. Finally, we want to remark that the context C_{PQ} is adapted from [26].

Now, let $P, Q \in \mathcal{P}$ satisfying $C_{PQ}[P] \simeq C_{PQ}[Q]$ and $P \xrightarrow{m, \alpha} P'$. Therefore, $C_{PQ}[P]$ can engage in the transitions illustrated in the left hand side of Figure 4 where $L = \{\underline{c} \mid \underline{c} \in (\underline{\mathcal{S}}(P) \cup \underline{\mathcal{S}}(Q)) \setminus \underline{\mathbb{I}}_{[m]}(P)\}$. Since $C_{PQ}[P] \simeq C_{PQ}[Q]$, the process $C_{PQ}[Q]$ has to match each step.

In order to be able to match the first step, $C_{PQ}[Q]$ has to choose exactly the same branch of H_{PQ} yielding to the process $\underline{d}_L.H_{PQ} + D_L$, because only this process is able to execute the distinguished action \underline{d}_L . For matching the second step, the process Q must be able to perform an α -transition from some location $n \in \text{Loc}$. According to our semantics for the parallel composition, the condition

$$\underline{\mathbb{I}}_{[n]}(Q) \cap \overline{\mathbb{I}}(\underline{d}_L.H_{PQ} + D_L) = \emptyset$$

must be satisfied. Because of the choice of L , this implies

$$\underline{\mathbb{I}}_{[n]}(Q) \subseteq \underline{\mathbb{I}}_{[m]}(P) .$$

Figure 4 shows the existence of some $Q' \in \mathcal{P}$ satisfying $C_{PQ}[P'] \simeq C_{PQ}[Q']$. Since $\underline{\mathcal{S}}(P') \subseteq \underline{\mathcal{S}}(P)$ and $\underline{\mathcal{S}}(Q') \subseteq \underline{\mathcal{S}}(Q)$ it follows that $C_{P'Q'}[P'] \simeq C_{P'Q'}[Q']$ as desired.

The case where P performs a prioritized transition needs no special attention since the Definitions 4.1 and 4.2 are identical in this case. Summarizing, we have shown that all conditions of Definition 4.2 are satisfied, and we may conclude that \mathcal{R} is a prioritized strong congruence. Hence, $P \simeq^+ Q$, which completes the proof. \square

$$\begin{array}{ccc}
C_{PQ}[P] \equiv P \mid H_{PQ} & \simeq & Q \mid H_{PQ} \equiv C_{PQ}[Q] \\
\downarrow \tau & & \downarrow \tau \\
P \mid \underline{d}_L.H_{PQ} + D_L & \simeq & Q \mid \underline{d}_L.H_{PQ} + D_L \\
\downarrow m \cdot L, \alpha & & \downarrow n \cdot L, \alpha \\
P' \mid \underline{d}_L.H_{PQ} + D_L & \simeq & Q' \mid \underline{d}_L.H_{PQ} + D_L \\
\downarrow \underline{d}_L & & \downarrow \underline{d}_L \\
C_{PQ}[P'] \equiv P' \mid H_{PQ} & \simeq & Q' \mid H_{PQ} \equiv C_{PQ}[Q']
\end{array}$$

Figure 4: Largest congruence proof - illustration

A.3 Proof of Theorem 4.10

For the soundness of all the axioms, which were presented in the Tables 3, 4, and 5, except Axioms (S2) and (S3) we observe that those axioms are true when replacing $=$ by \simeq^+ . The Axioms (S2) and (S3) involve a side condition. Their soundness follows immediately by considering Lemma 4.6 that can be proven correct by reasoning on the structure of finite process terms.

The key idea for the completeness proof of the axiomatization is to abstract from locations or their prioritized initial action sets by rewriting finite processes into a special normal form. Our normal form as defined in Definition 4.8, is not only a *summation form* (cf. [23] and Definition 4.7), but also let us reduce the completeness proof to the completeness proof of the axiomatization of strong bisimulation for CCS as presented in [23].

We first argue that every finite process can be rewritten in its normal form as defined in Definition 4.8 (cf. Proposition 4.9). This normal form possesses an incomparable summand for each subset of its prioritized initial actions. Moreover, terms prefixed by an unprioritized action are copied to each distributed summand which already has another term prefixed by the same prioritized action. Finally, if there exists a term prefixed by an unprioritized action in two incomparable summands and the initial action set of one summand is included in that of the other, then the prefixed term is deleted in the summand with the larger prioritized initial action set.

Let $P \in \mathcal{P}$. First, we transform P into its summation form S (see Definition 4.7). This procedure corresponds to that in CCS and uses Axioms (A1)–(A4), (iA1)–(iA4), (E), (Res1)–(Res3), (iRes3), (Rel1)–(Rel3), and (iRel3). The formal proof that every finite process can be rewritten into summation form can be done by induction on the *depth* of P which is defined as in [23]. Next, we rewrite S into a term N which satisfies the conditions stated in Definition 4.8, i.e N is in normal form. We achieve Condition (1) by using the *Exchange Axioms* (Ic1) and (Ic2) together

with the *Distributive Axioms* (D1), (D2) and (D3). Moreover, we use Axiom (P) to establish Condition (2) and *Saturation Axiom* (S1) to establish Condition (3). By using Axioms (S2), (S3), (Ic1) and (Ic2) we obtain Conditions (4) and (5). Also here, an induction argument on the depth of S is necessary to show formally the existence of such a normal form N which is equal to S in E . Combining both steps, the validity of Proposition 4.9 follows.

In order to prove the completeness part of Theorem 4.10, let P and Q be finite processes satisfying $P \simeq^+ Q$ which are already given in normal form. We show that P and Q are identical up to (incomparable) associativity, commutativity, and idempotence such that they can be rewritten as syntactically equal terms by using Axioms (A1), (A2), (A3), (iA1), (iA2), and (A3). Formally, we reason by induction on the maximum depth of the processes P and Q . The induction base, i.e. both processes correspond to $\mathbf{0}$, is trivial. For the induction step, the key is the following lemma. Its correctness follows by the conditions satisfied by terms in normal form.

Lemma A.6

Let $P \equiv \bigoplus_{i=1}^m P_i$ where $P_i =_{\text{df}} \sum_{j=1}^{n_i} \gamma_{ij} \cdot P_{ij}$ and $Q \equiv \bigoplus_{k=1}^{m'} Q_k$ where $Q_k =_{\text{df}} \sum_{l=1}^{n'_k} \delta_{kl} \cdot Q_{kl}$ be in normal form satisfying $P \simeq^+ Q$. Then we have:

$$\forall 1 \leq i \leq m \exists 1 \leq k \leq m'. P_i \simeq^+ Q_k$$

and

$$\forall 1 \leq k \leq m' \exists 1 \leq i \leq m. Q_k \simeq^+ P_i.$$

Using the above lemma we finally get terms with only one location “on top level”, i.e. locations do not play a role any longer. In this case, prioritized strong bisimulation reduces on top level to standard strong bisimulation. For those terms we can apply the standard technique presented in [23] together with our induction hypothesis in order to show that equivalent terms of those forms are equal with respect to our equations. Note that the axioms in [23] are included in our set of axioms.

Now, we are almost done. Perhaps, we additionally have to apply the Axioms (A1), (A2), (A3), (iA1), (iA2), and (iA3) to rewrite P and Q such that they are syntactically equal. Summarizing, we have shown that $\vdash_E P = Q$, as desired.

A.4 Proof of Theorem 4.11

Most proof parts for Theorem 4.11, which states that our logic characterizes prioritized strong bisimulation, are similar to the corresponding ones presented in [23].

First, we define a characterization of prioritized strong congruence for finite-branching transition systems over \mathcal{A} . A transition system $\langle \mathcal{P}, \mathcal{A}, \longrightarrow, P \rangle$ is called *finite-branching* if the set $\{P'' \mid P' \xrightarrow{m, \gamma} P'', \gamma \in \mathcal{A}, m \in \text{Loc}\}$ is finite for all reachable states P' of the process P .

Definition A.7 Let $\langle \mathcal{P}, \mathcal{A}, \longrightarrow, P \rangle$ and $\langle \mathcal{P}, \mathcal{A}, \longrightarrow, Q \rangle$ be finite-branching transition systems. We define $\simeq^+_{\mathbf{0}} = \mathcal{P} \times \mathcal{P}$ and $P \simeq^+_{k+1} Q$ for some $k \in \mathbb{N}$ if the following properties hold:

1. $P \xrightarrow{\alpha} P'$ implies $\exists Q'. Q \xrightarrow{\alpha} Q'$ and $P' \simeq^+_k Q'$.
2. $P \xrightarrow{m, \alpha} P'$ implies $\exists Q', n. Q \xrightarrow{n, \alpha} Q', \mathbb{I}_{[n]}(Q) \subseteq \mathbb{I}_{[m]}(P)$, and $P' \simeq^+_k Q'$.
3. $Q \xrightarrow{\alpha} Q'$ implies $\exists P'. P \xrightarrow{\alpha} P'$ and $P' \simeq^+_k Q'$.

4. $Q \xrightarrow{m,\alpha} Q'$ implies $\exists P', n. P \xrightarrow{n,\alpha} P', \mathbb{I}_{[n]}(P) \subseteq \mathbb{I}_{[m]}(Q)$, and $P' \simeq^+_k Q'$.

The proof of the next proposition follows the lines in [23]. Note that for all processes in CCS^{prio} their corresponding transition systems are finite-branching.

Proposition A.8 *Let $P, Q \in \mathcal{P}$. We have $P \simeq^+ Q$ if and only if $P \simeq^+_i Q$ for all $i \in \mathbb{N}$.*

Now we are able to prove Theorem 4.11. By using Proposition A.8 it is sufficient to establish the following two lemmata.

Lemma A.9 *Let $P, Q \in \mathcal{P}$, $i \in \mathbb{N}$, and $\Phi \in \mathcal{F}$ such that $P \simeq^+_i Q$ and $P \models \Phi$. Then $Q \models \Phi$ holds.*

Proof: We prove the lemma by induction on i where the induction step is divided into several cases according to the structure of Φ . The only non-standard case is $\Phi = \langle \alpha, L \rangle \Psi$ for $\alpha \in A$ and $L \subseteq \underline{A} \setminus \{\tau\}$. By definition of \models we conclude the existence of a process $P' \in \mathcal{P}$ and a location $m \in \text{Loc}$ such that $P \xrightarrow{m,\alpha} P'$, $\mathbb{I}_{[m]}(P) \subseteq L$, and $P' \models \Psi$. Since $P \simeq^+_i Q$ we also know of the existence of some $Q' \in \mathcal{P}$ and $n \in \text{Loc}$ such that $Q \xrightarrow{n,\alpha} Q'$, $\mathbb{I}_{[n]}(Q) \subseteq \mathbb{I}_{[m]}(P)$, and $P' \simeq^+_{i-1} Q'$. We conclude that $\mathbb{I}_{[n]}(Q) \subseteq L$ and, by the induction hypothesis, that $Q' \models \Psi$. Therefore, $Q \models \langle \alpha, L \rangle \Psi$, as desired. \square

Lemma A.10 *Let $P, Q \in \mathcal{P}$ and $i \in \mathbb{N}$ such that $P \not\simeq^+_i Q$ holds. There exists a formula $\Phi \in \mathcal{F}$ such that $P \models \Phi$ but $Q \not\models \Phi$.*

Proof: Again, we prove this lemma by induction on i . The induction base is trivial since the premise $P \not\simeq^+_0 Q$ does not hold. Now, let $i > 0$ and $P \not\simeq^+_i Q$. We have to find a formula $\Phi \in \mathcal{F}$ such that $P \models \Phi$ and $Q \not\models \Phi$. Since $P \not\simeq^+_i Q$ we have $P \xrightarrow{m,\gamma} P'$ for some $\gamma \in A$, $m \in \text{Loc}$, and $P' \in \mathcal{P}$ which cannot be matched by a step of Q . The case where $\gamma \in \underline{A}$ follows the standard lines. Now, let $\gamma = \alpha \in A$, i.e. we know that whenever $Q \xrightarrow{n,\alpha} Q'$ and $\mathbb{I}_{[n]}(Q) \subseteq \mathbb{I}_{[m]}(P)$ then $P' \not\simeq^+_{i-1} Q'$. Let $\{Q' \mid Q \xrightarrow{n,\alpha} Q' \text{ and } \mathbb{I}_{[n]}(Q) \subseteq \mathbb{I}_{[m]}(P)\} = \{Q_j \mid j \in J\}$ for some index set J . By induction hypothesis we conclude the existence of formulae Ψ_j , for $j \in J$, such that $P' \models \Psi_j$ and $Q_j \not\models \Psi_j$. Now, define $\Phi =_{\text{df}} \langle \alpha, L \rangle \bigwedge_{j \in J} \Psi_j$ where $L =_{\text{df}} \mathbb{I}_{[m]}(P)$. Because of the choice of L , it is easy to see that $P \models \Phi$. Since no α -derivative of Q from location n with $\mathbb{I}_{[n]}(Q) \subseteq L$ satisfies $\bigwedge_{j \in J} \Psi_j$, we have $Q \not\models \Phi$. \square

A.5 Proof of Proposition 5.5

In this section, we present an outline of the proof that \cong is a congruence for all operators except summation and recursion. The most interesting case is the compositionality with respect to parallel composition which we give in full detail.

Proposition A.11 *$P \cong Q$ implies $P \mid R \cong Q \mid R$ for all processes R .*

Proof: According to the definition of \cong it is sufficient to prove that

$$\mathcal{R} =_{\text{df}} \{ \langle P \mid R, Q \mid R \rangle \mid P \cong Q \}$$

is a prioritized weak bisimulation.

Let $P \mid R \xrightarrow{p, \alpha} V$ for some $V \in \mathcal{P}$ and $p \in \mathcal{Loc}$. We have to show the existence of some $W \in \mathcal{P}$ and $q \in \mathcal{Loc}$ such that $Q \mid R \xrightarrow[q, \alpha]{L, M} W$ and $\langle V, W \rangle \in \mathcal{R}$ where $L = \underline{\mathbb{I}}_{[p]}(P \mid R)$ and $M = \underline{\mathbb{I}}(P \mid R)$. Consider the following case distinction according to the operational rules for parallel composition.

1. *Case.* $P \xrightarrow{m, \alpha} P'$ and $\underline{\mathbb{I}}_{[m]}(P) \cap \underline{\mathbb{I}}(R) = \emptyset$, i.e. $V \equiv P' \mid R$ and $p = m \cdot L$

Since $P \cong Q$ there exists a process $Q' \in \mathcal{P}$ and a location $n \in \mathcal{Loc}$ satisfying $Q \xrightarrow[n, \hat{\alpha}]{L', M'} Q'$, $L' = \underline{\mathbb{I}}_{[m]}(P)$, $M' = \underline{\mathbb{I}}(P)$, and $P' \cong Q'$.

According to the definition of the prioritized weak transition relation, we have the following situation:

$$\exists s, t \in \mathbb{N} \forall 0 \leq i < s \forall 0 < j < t \exists Q_i, Q_{s+j} \in \mathcal{P} \exists n_i, n_{s+j} \in \mathcal{Loc}$$

where $Q_0 \equiv Q, Q_{s+t} \equiv Q'$ such that

- (a) $Q_i \xrightarrow[n_i, \tau]{L'} Q_{i+1}$ or $Q_i \xrightarrow{\tau} Q_{i+1}$,
- (b) $Q_s \xrightarrow[n, \hat{\alpha}]{L'} Q_{s+1}$ and $\underline{\mathbb{I}}(Q_s) \subseteq M'$, and
- (c) $Q_{s+j} \xrightarrow[n_{s+j}, \tau]{\emptyset} Q_{s+j+1}$ or $Q_{s+j} \xrightarrow{\tau} Q_{s+j+1}$.

Considering the parallel composition with the process R , we obtain the following results.

- (a) If $Q_i \xrightarrow[n_i, \tau]{L'} Q_{i+1}$, then we have $Q_i \mid R \xrightarrow[n_i \cdot L, \tau]{L} Q_{i+1} \mid R$ by Rule Com1 where $L = \underline{\mathbb{I}}_{[m \cdot L]}(P \mid R)$. This is because
 - i. $L = L'$ (by Def. 3.1),
 - ii. $(\underline{\mathbb{I}}_{[n_i]}(Q_i) \cap \underline{\mathbb{I}}(R)) \subseteq (\underline{\mathbb{I}}_{[m]}(P) \cap \underline{\mathbb{I}}(R)) = \emptyset$ (since $\underline{\mathbb{I}}_{[n_i]}(Q_i) \subseteq \underline{\mathbb{I}}_{[m]}(P)$), and
 - iii. $\underline{\mathbb{I}}_{[n_i \cdot L]}(Q_i \mid R) \subseteq \underline{\mathbb{I}}_{[m \cdot L]}(P \mid R)$ (by Lemma A.3(2)).

If $Q_i \xrightarrow{\tau} Q_{i+1}$, then we obtain $Q_i \mid R \xrightarrow{\tau} Q_{i+1} \mid R$ by Rule Com1.

- (b) Here, we conclude $Q_s \mid R \xrightarrow[n, \hat{\alpha}]{L} Q_{s+1} \mid R$ where L is defined as above, since
 - i. $(\underline{\mathbb{I}}_{[n]}(Q_s) \cap \underline{\mathbb{I}}(R)) \subseteq (\underline{\mathbb{I}}_{[m]}(P) \cap \underline{\mathbb{I}}(R)) = \emptyset$ (because $\underline{\mathbb{I}}_{[n]}(Q_s) \subseteq \underline{\mathbb{I}}_{[m]}(P)$),
 - ii. $\underline{\mathbb{I}}_{[n \cdot L]}(Q_s \mid R) \subseteq \underline{\mathbb{I}}_{[m \cdot L]}(P \mid R)$ (by Lemma A.3(2)).

Moreover, $\underline{\mathbb{I}}(Q_s \mid R) \subseteq M$ for $M = \underline{\mathbb{I}}(P \mid R)$ (by Lemma A.3(5)).

- (c) If $Q_{s+j} \xrightarrow[n_{s+j}, \tau]{\emptyset} Q_{s+j+1}$, then $Q_{s+j} \mid R \xrightarrow[n_{s+j} \cdot L, \tau]{\emptyset} Q_{s+j+1} \mid R$ holds by Rule Com1. The reason for this is, that
 - i. $(\underline{\mathbb{I}}_{[n_{s+j}]}(Q_{s+j}) \cap \underline{\mathbb{I}}(R)) \subseteq (\emptyset \cap \underline{\mathbb{I}}(R)) = \emptyset$ (since $\underline{\mathbb{I}}_{[n_{s+j}]}(Q_{s+j}) \subseteq \emptyset$), and
 - ii. $\underline{\mathbb{I}}_{[n_{s+j} \cdot L]}(Q_{s+j} \mid R) \subseteq \emptyset$ (by Lemma A.3(2)).

If $Q_{s+j} \xrightarrow{\tau} Q_{s+j+1}$, then we may immediately conclude $Q_{s+j} \mid R \xrightarrow{\tau} Q_{s+j+1} \mid R$ by Rule Com1.

According to the definition of the prioritized weak transition relation, we have shown that $Q \mid R \xrightarrow[q, \alpha]{n \cdot L, \hat{\alpha}} Q_{s+t} \mid R$. By choosing $W =_{\text{df}} Q_{s+t} \mid R$ and $q =_{\text{df}} n \cdot L$, and since $\langle P \mid R, Q_{s+t} \mid R \rangle \in \mathcal{R}$, we are done.

2. *Case.* $R \xrightarrow{o,\alpha} R'$ and $\underline{\mathbb{I}}_{[o]}(R) \cap \overline{\mathbb{I}}(P) = \emptyset$, i.e. $V \equiv P \mid R'$ and $p = o \cdot R$

Since $P \cong Q$ there exists processes $Q', Q'' \in \mathcal{P}$ such that $Q \xrightarrow{\epsilon} Q'' \xrightarrow{\epsilon} Q'$, $\underline{\mathbb{I}}(Q'') \subseteq \underline{\mathbb{I}}(P)$, and $P \cong Q'$.

According to the definition of the prioritized weak transition relation, we have the following situation:

$$\exists s, t \in \mathbb{N} \forall 0 \leq i < s \forall 0 < j < t \exists Q_i, Q_{s+j} \in \mathcal{P} \exists n_i, n_{s+j} \in \mathcal{L}oc$$

where $Q_0 \equiv Q$, $Q_s \equiv Q''$, $Q_s \equiv Q_{s+1}$, and $Q_{s+t} \equiv Q'$ such that

- (a) $Q_i \xrightarrow[n_i, \tau]{\tau} Q_{i+1}$ or $Q_i \xrightarrow{\tau} Q_{i+1}$,
- (b) $\underline{\mathbb{I}}(Q_s) \subseteq \underline{\mathbb{I}}(P)$, and
- (c) $Q_{s+j} \xrightarrow[n_{s+j}, \tau]{\tau} Q_{s+j+1}$ or $Q_{s+j} \xrightarrow{\tau} Q_{s+j+1}$.

Adding the parallel context R , we obtain the following.

- (a) If $Q_i \xrightarrow[n_i, \tau]{\tau} Q_{i+1}$, then $Q_i \mid R \xrightarrow[L]{n_i \cdot L, \tau} Q_{i+1} \mid R$ holds for all prioritized action sets L . The reason for this is beside Rule Com1, that
 - i. $(\underline{\mathbb{I}}_{[n_i]}(Q_i) \cap \overline{\mathbb{I}}(R)) \subseteq (\emptyset \cap \overline{\mathbb{I}}(R)) = \emptyset$ (since $\underline{\mathbb{I}}_{[n_i]}(Q_i) \subseteq \emptyset$), and
 - ii. $\underline{\mathbb{I}}_{[n_i \cdot L]}(Q_i \mid R) \subseteq \emptyset \subseteq L$ (because $\underline{\mathbb{I}}_{[n_i]}(Q_i) \subseteq \emptyset$).

If $Q_i \xrightarrow{\tau} Q_{i+1}$, then we may immediately conclude $Q_i \mid R \xrightarrow{\tau} Q_{i+1} \mid R$ by Rule Com1.

- (b) For $L = \underline{\mathbb{I}}_{[o \cdot R]}(P \mid R)$ and $M = \underline{\mathbb{I}}(P \mid R)$, we have $Q_s \mid R \xrightarrow[L]{m \cdot R, \alpha} Q_{s+1} \mid R'$ by Rule Com2 and
 - i. $\underline{\mathbb{I}}_{[o \cdot R]}(Q_s \mid R) \subseteq L$ (by Lemma A.3(3) and the definition of L),
 - ii. $(\underline{\mathbb{I}}_{[o]}(R) \cap \overline{\mathbb{I}}(Q_s)) \subseteq (\underline{\mathbb{I}}_{[o]}(R) \cap \overline{\mathbb{I}}(P)) = \emptyset$ (because $\underline{\mathbb{I}}(Q_s) \subseteq \underline{\mathbb{I}}(P)$), and $\underline{\mathbb{I}}(Q_s \mid R) \subseteq M$ (by Lemma A.3(5) and the definition of M).

- (c) If $Q_{s+j} \xrightarrow[n_{s+j}, \tau]{\tau} Q_{s+j+1}$, then $Q_{s+j} \mid R' \xrightarrow[n_{s+j}, \tau]{n_{s+j} \cdot L, \tau} Q_{s+j+1} \mid R'$ holds by Rule Com1 and
 - i. $(\underline{\mathbb{I}}_{[n_{s+j}]}(Q_{s+j}) \cap \overline{\mathbb{I}}(R')) \subseteq (\emptyset \cap \overline{\mathbb{I}}(R')) = \emptyset$ (since $\underline{\mathbb{I}}_{[n_{s+j}]}(Q_{s+j}) \subseteq \emptyset$),
 - ii. $\underline{\mathbb{I}}_{[n_{s+j} \cdot L]}(Q_{s+j} \mid R') \subseteq \emptyset$ (because $\underline{\mathbb{I}}_{[n_{s+j}]}(Q_{s+j}) \subseteq \emptyset$).

If $Q_{s+j} \xrightarrow{\tau} Q_{s+j+1}$, then we may immediately conclude $Q_{s+j} \mid R' \xrightarrow{\tau} Q_{s+j+1} \mid R'$ by Rule Com1.

According to the definition of the prioritized weak transition relation, we have shown that $Q \mid R \xrightarrow[L, M]{o \cdot R, \hat{\alpha}} Q_{s+t} \mid R'$. By choosing $W =_{\text{df}} Q_{s+t} \mid R' \equiv Q' \mid R'$ and $q =_{\text{df}} o \cdot R$ we have finished this case, because $\langle P \mid R', Q' \mid R' \rangle \in \mathcal{R}$ holds according to the definition of \mathcal{R} .

3. *Case.* $P \xrightarrow{m, a} P'$, $R \xrightarrow{o, \bar{\alpha}} R'$, $\underline{\mathbb{I}}_{[m]}(P) \cap \overline{\mathbb{I}}(R) = \emptyset$, and $\underline{\mathbb{I}}_{[o]}(R) \cap \overline{\mathbb{I}}(P) = \emptyset$, i.e. $V \equiv P' \mid R'$ and $p = \langle m \cdot L, o \cdot R \rangle$

Since $P \cong Q$ there exists $Q' \in \mathcal{P}$ and $n \in \mathcal{L}oc$ satisfying $Q \xrightarrow[L', M']{n, \hat{\alpha}} Q'$, $L' = \underline{\mathbb{I}}_{[m]}(P)$, $M' = \underline{\mathbb{I}}(P)$, and $P' \cong Q'$.

According to the definition of the prioritized weak transition relation, we have the following situation:

$$\exists s, t \in \mathbb{N} \forall 0 \leq i < s \forall 1 < j < t \exists Q_i, Q_{s+j} \in \mathcal{P} \exists n_i, n_{s+j} \in \mathcal{L}oc$$

where $Q_0 \equiv Q$, $Q_{s+t} \equiv Q'$ such that

- (a) $Q_i \xrightarrow[L']{n_i, \tau} Q_{i+1}$ or $Q_i \xrightarrow{\tau} Q_{i+1}$,
- (b) $Q_s \xrightarrow[L']{n, a} Q_{s+1}$ and $\underline{\mathbb{I}}(Q_s) \subseteq M'$, and
- (c) $Q_{s+j} \xrightarrow[\emptyset]{n_{s+j}, \tau} Q_{s+j+1}$ or $Q_{s+j} \xrightarrow{\tau} Q_{s+j+1}$.

Now, we add the parallel context R .

- (a) Here, we can argue similar to Case (1) since $\underline{\mathbb{I}}_{[m]}(P) \subseteq \underline{\mathbb{I}}_{[\langle m \cdot L, \circ \cdot R \rangle]}(P | R)$ (cf. Def. 3.1).
- (b) The transition $Q_s | R \xrightarrow[L, M]{\langle n \cdot L, \circ \cdot R \rangle, \tau} Q_{s+1} | R'$ where $L = \underline{\mathbb{I}}_{[\langle m \cdot L, \circ \cdot R \rangle]}(P | R)$ and $M = \underline{\mathbb{I}}(P | R)$ is possible, because we have beside Rule Com3 that
 - i. $(\underline{\mathbb{I}}_{[n]}(Q) \cap \overline{\mathbb{I}}(R)) \subseteq (\underline{\mathbb{I}}_{[m]}(P) \cap \overline{\mathbb{I}}(R)) = \emptyset$ (since $\underline{\mathbb{I}}_{[n]}(Q) \subseteq \underline{\mathbb{I}}_{[m]}(P)$),
 - ii. $(\underline{\mathbb{I}}_{[\circ]}(R) \cap \overline{\mathbb{I}}(Q)) \subseteq (\underline{\mathbb{I}}_{[\circ]}(R) \cap \overline{\mathbb{I}}(P)) = \emptyset$ (because $\underline{\mathbb{I}}(Q) \subseteq \underline{\mathbb{I}}(P)$),
 - iii. $\underline{\mathbb{I}}_{[\langle n \cdot L, \circ \cdot R \rangle]}(Q | R) \subseteq L$ (by Lemma A.3(4) and the definition of L), and
 - iv. $\underline{\mathbb{I}}(Q | R) \subseteq M$ (by Lemma A.3(5) and the definition of M).
- (c) Here, we can follow the argumentation along the lines of the second case since $\underline{\mathbb{I}}_{[m]}(P) \subseteq \underline{\mathbb{I}}_{[\langle m \cdot L, \circ \cdot R \rangle]}(P | R)$ (cf. Def. 3.1).

According to the definition of the prioritized weak transition relation, we have shown that $Q | R \xrightarrow[L, M]{\langle n \cdot L, \circ \cdot R \rangle, \tau} Q_{s+t} | R'$. Thus, $Q | R \xrightarrow[L]{\epsilon} Q_{s+t} | R'$. By choosing $W =_{\text{df}} Q_{s+t} | R' \equiv Q' | R'$ and because $\langle P' | R', Q' | R' \rangle \in \mathcal{R}$, we are done.

For establishing the first condition of Definition 5.4 we argue as follows.

Since $P \approx Q$ there exists $Q', Q'' \in \mathcal{P}$ such that $Q \xrightarrow{\epsilon} Q'' \xrightarrow{\epsilon} Q'$, $\underline{\mathbb{I}}(Q'') \subseteq \underline{\mathbb{I}}(P)$, and $P \approx Q'$.

According to the definition of the prioritized weak transition relation, we have the following situation:

$$\exists s, t \in \mathbb{N} \forall 0 \leq i < s \forall 0 \leq j < t \exists Q_i, Q_{s+j} \in \mathcal{P} \exists n_i, n_{s+j} \in \mathcal{Loc}$$

where $Q_0 \equiv Q$, $Q_s \equiv Q''$, and $Q_{s+t} \equiv Q'$ such that

- (a) $Q_i \xrightarrow[\emptyset]{n_i, \tau} Q_{i+1}$ or $Q_i \xrightarrow{\tau} Q_{i+1}$,
- (b) $\underline{\mathbb{I}}(Q_s) \subseteq \underline{\mathbb{I}}(P)$, and
- (c) $Q_{s+j} \xrightarrow[\emptyset]{n_{s+j}, \tau} Q_{s+j+1}$ or $Q_{s+j} \xrightarrow{\tau} Q_{s+j+1}$.

It follows, that

- (a) $Q_i | R \xrightarrow[\emptyset]{n_i \cdot L, \tau} Q_{i+1} | R$ or $Q_i | R \xrightarrow{\tau} Q_{i+1} | R$ by the same argument as in Case (1),
- (b) $\underline{\mathbb{I}}(Q_s | R) \subseteq \underline{\mathbb{I}}(P | R)$ by Lemma A.3(5), and
- (c) $Q_{s+j} | R \xrightarrow[\emptyset]{n_{s+j} \cdot L, \tau} Q_{s+j+1} | R$ or $Q_{s+j} | R \xrightarrow{\tau} Q_{s+j+1} | R$ analogue to Case (1).

Moreover, we have $\langle P | R, Q' | R \rangle \in \mathcal{R}$ according to the definition of \mathcal{R} .

The second condition of Definition 5.4 is standard since the semantics for the parallel operator with respect to prioritized transitions coincides with those of CCS. \square

A.6 Proof of Theorem 5.7

In this section, we prove Theorem 5.7 which states that $\underline{\cong}^+$ is the *largest* congruence contained in \approx_\times . First, we show that $\underline{\cong}^+$ is a congruence. Because we have not made use of matching a τ -transition by an ‘idling’ ϵ -transition in the proof of Proposition 5.5, and because of the definition of initial action sets, we obtain immediately that $\underline{\cong}^+$ is compositional for all operators except summation and recursion. We want to remark, that the compositionality proof of $\underline{\cong}^+$ with respect to the distributed summation operator \oplus is similar to the one for the usual summation in CCS. In the following, we present the compositionality result of $\underline{\cong}^+$ with respect to summation and recursion.

Proposition A.12 $P \underline{\cong}^+ Q$ implies $P + R \underline{\cong}^+ Q + R$ for all processes R .

Proof: Let $P, Q \in \mathcal{P}$ satisfying $P \underline{\cong}^+ Q$. According to the definition of $\underline{\cong}^+$ it is sufficient to prove that

$$P + R \xrightarrow{p,\alpha} V \text{ implies } \exists W, q. Q + R \xrightarrow{q,\alpha}_{L,M} W, L = \underline{\mathbb{I}}_{[p]}(P + R), M = \underline{\mathbb{I}}(P + R), \text{ and } V \underline{\cong} W$$

holds, since Property (1) of Definition 5.6 follows directly by Definition 5.6 for $P \underline{\cong}^+ Q$ and Definition 3.1, and the Properties (2) and (4) are analogue to those of *observational congruence* in CCS.

It remains to show the existence of some $W \in \mathcal{P}$ and $q \in \mathcal{Loc}$ such that $Q + R \xrightarrow{q,\alpha}_{L,M} W$ and $V \underline{\cong} W$. Therefore, consider the following case distinction according to the operational rules for summation.

1. *Case.* $P \xrightarrow{m,\alpha} P'$ and $\tau \notin \underline{\mathbb{I}}(R)$, i.e. $p = m \cdot l$ and $V \equiv P'$. Since $P \underline{\cong}^+ Q$ we know

$$\exists Q', n. Q \xrightarrow{n,\alpha}_{L',M'} Q', L' = \underline{\mathbb{I}}_{[m]}(P), M' = \underline{\mathbb{I}}(P), \text{ and } P' \underline{\cong} Q'.$$

According to the definition of the prioritized weak transition relation, we distinguish the following cases where $Q'' \in \mathcal{P}$ and $k \in \mathcal{Loc}$.

- (a) $Q \xrightarrow{n,\alpha}_{L'} Q'' \xrightarrow{\epsilon} Q'$. Therefore, we have $\underline{\mathbb{I}}_{[n]}(Q) \subseteq L'$.
- (b) $Q \xrightarrow{k,\tau}_{L'} Q'' \xrightarrow{n,\alpha}_{L',M'} Q'$. Here, we have $\underline{\mathbb{I}}_{[k]}(Q) \subseteq L'$.
- (c) $Q \xrightarrow{\tau} Q'' \xrightarrow{n,\alpha}_{L',M'} Q'$.

Now, let us define $L =_{\text{df}} \underline{\mathbb{I}}_{[m \cdot l]}(P + R)$ and $M =_{\text{df}} \underline{\mathbb{I}}(P + R)$. The first case implies that

$$Q + R \xrightarrow{n \cdot l, \alpha}_L Q'$$

by Rule Sum1, the premise $\underline{\mathbb{I}}_{[n]}(Q) \subseteq L'$, and Lemma A.1(2). Moreover, $\underline{\mathbb{I}}(Q + R) \subseteq \underline{\mathbb{I}}(P + R)$ holds by the fact $\underline{\mathbb{I}}(P) = \underline{\mathbb{I}}(Q)$ and Lemmata A.3(1) and A.3(5). Therefore, we have

$$Q + R \xrightarrow{n \cdot l, \alpha}_{L,M} Q'' \xrightarrow{\epsilon} Q' \text{ and } P' \underline{\cong} Q'$$

by the definition of the prioritized weak transition relation. Hence, $Q + R \xrightarrow{n \cdot l, \alpha}_{L,M} Q'$ and $P' \underline{\cong} Q'$. We conclude for this case by choosing $W =_{\text{df}} Q'$ and $q =_{\text{df}} n \cdot l$.

The second case is similar to the first one, whereas the last case implies by Rule Sum1 that

$$Q + R \xrightarrow{\tau} Q'' \xrightarrow[L', M']{n, \alpha} Q' .$$

Because of $L' \subseteq L$ and $M' \subseteq M$, we obtain $Q + R \xrightarrow[L, M]{n, \alpha} Q'$ and $P' \cong Q'$. By choosing $W =_{\text{df}} Q'$ and $q =_{\text{df}} n$ we are done.

2. *Case.* $R \xrightarrow{o, \alpha} R'$ and $\tau \notin \underline{\mathbb{I}}(P)$, i.e. $V \equiv R'$ and $p = o \cdot r$.

Since $P \cong^+ Q$ and, therefore, by Definition 5.6 $\underline{\mathbb{I}}(P) = \underline{\mathbb{I}}(Q)$, we have

- (a) $Q + R \xrightarrow{o, r, \alpha} R'$ (because $\tau \notin \underline{\mathbb{I}}(P) = \underline{\mathbb{I}}(Q)$),
- (b) $\underline{\mathbb{I}}_{[o, r]}(Q + R) \subseteq \underline{\mathbb{I}}_{[o, r]}(P + R)$ (by Lemmata A.3(2) and A.1(3)), and
- (c) $\underline{\mathbb{I}}(Q + R) \subseteq \underline{\mathbb{I}}(P + R)$ (by Lemmata A.3(2) and A.1(4)).

According to the symmetry of \cong and the definition of the prioritized weak transition relation we have $R' \cong R'$ and $Q + R \xrightarrow[L, M]{o, r, \alpha} R'$ where $L =_{\text{df}} \underline{\mathbb{I}}_{[o, r]}(P + R)$ and $M =_{\text{df}} \underline{\mathbb{I}}(P + R)$. By choosing $W =_{\text{df}} R'$ and $q =_{\text{df}} o \cdot r$, we have finished the proof. □

In order to show that \cong^+ is compositional with respect to recursion, we need to define a notion of *prioritized weak bisimulation up to \cong* (cf. [30]). With this definition, the proof is completely standard [23].

Definition A.13 (Prioritized Weak Bisimulation up to \cong)

A relation $R \subseteq \mathcal{P} \times \mathcal{P}$ is a prioritized weak bisimulation up to \cong if for every $\langle P, Q \rangle \in R$, $\alpha \in A$, $\underline{\alpha} \in \underline{A}$, and $m \in \text{Loc}$ the following conditions hold.

1. $\underline{\mathbb{I}}(P) = \underline{\mathbb{I}}(Q)$
2. $P \xrightarrow{\underline{\alpha}} P'$ implies $\exists Q'. Q \xrightarrow{\alpha} Q'$ and $P' R \cong Q'$.
3. $P \xrightarrow{m, \alpha} P'$ implies $\exists Q', n. Q \xrightarrow[L, M]{n, \alpha} Q'$, $L = \underline{\mathbb{I}}_{[m]}(P)$, $M = \underline{\mathbb{I}}(P)$, and $P' R \cong Q'$.
4. $Q \xrightarrow{\underline{\alpha}} Q'$ implies $\exists P'. P \xrightarrow{\alpha} P'$ and $P' \cong R Q'$.
5. $Q \xrightarrow{m, \alpha} Q'$ implies $\exists P', n. P \xrightarrow[L, M]{n, \alpha} P'$, $L = \underline{\mathbb{I}}_{[m]}(Q)$, $M = \underline{\mathbb{I}}(Q)$, and $P' \cong R Q'$.

As expected, this notion satisfies the property that, if R is a *prioritized weak bisimulation up to \cong* , then $R \subseteq \cong$.

Up to now, we only know that \cong^+ is a congruence. However, it is relatively easy to establish the following result, which turn out to be very useful for the proof of Theorem 5.7.

Proposition A.14 *The congruence \cong^+ is the largest congruence contained in \cong .*

Proof: Since the relation \cong^+ is a congruence it is sufficient to show that for all CCS^{prio} -contexts C and processes $P, Q \in \mathcal{P}$ satisfying $C[P] \cong C[Q]$ we have $P \cong^+ Q$. Moreover, we may restrict ourselves to a subset of CCS^{prio} -contexts. For this proof, we choose the single context

$C[X] =_{\text{df}} (\underline{c} \cdot \mathbf{0} + d \cdot \mathbf{0}) + X$ where $\underline{c} \notin \underline{\mathcal{S}}(P) \cup \underline{\mathcal{S}}(Q)$ and $d \notin \mathcal{S}(P) \cup \mathcal{S}(Q)$. Note that such action \underline{c} and d exist by Lemma A.5.

Assume that $\underline{\tau} \notin \underline{\mathbb{I}}(P)$ and, therefore, $C[P] \xrightarrow{r,l,d} \mathbf{0}$. Since $P \cong Q$ and $d \notin \mathcal{S}(Q)$ we necessarily have $C[Q] \xrightarrow{r,l,d}_{L,M} \mathbf{0}$ and $\mathbf{0} \cong \mathbf{0}$ where $L = M = \underline{\mathbb{I}}(P) \cup \{\underline{c}\}$. But this requires $\tau \notin \underline{\mathbb{I}}(Q)$ and $\underline{\mathbb{I}}(Q) \subseteq \underline{\mathbb{I}}(P)$. Thus, we have $\underline{\mathbb{I}}(Q) \subseteq \underline{\mathbb{I}}(P)$.

Let $C[P] \xrightarrow{m,\alpha} P'$. Since $P \cong Q$ there exists a process $Q' \in \mathcal{P}$ and a location $n \in \mathcal{L}oc$ satisfying $C[Q] \xrightarrow{n,\alpha}_{L,M} Q'$, $L = M = \underline{\mathbb{I}}(P) \cup \{\underline{c}\}$, and $P' \cong Q'$. We know that $Q' \neq C[Q]$ because $P' \cong Q'$ and P' is *not* capable of performing a \underline{c} -transition. Therefore, the matching step is necessary, even if $\alpha = \tau$.

Finally, let $C[P] \xrightarrow{\alpha} P'$. The same argumentation as above leads to the conclusion that $C[Q]$ must perform at least one step to simulate a step of $C[P]$.

Since also the symmetric properties hold, all conditions of Definition 5.6 are satisfied, and we obtain $P \cong^+ Q$ as desired. \square

Now, we are able to prove Theorem 5.7. This proof is going to be the hardest proof in the paper. It uses the following result from universal algebra.

Fact A.15 *Let X and Y be equivalence relations. Then the largest congruence X^+ in X exists. X^+ is characterized by $X^+ = \{\langle P, Q \rangle \mid \forall \text{CCSPrio-contexts } C. \langle C[P], C[Q] \rangle \in X\}$. Moreover, if $X^+ \subseteq Y \subseteq X$ then $X^+ = Y^+$.*

In our case, we choose $X = \approx_{\times}$ and $Y = \cong$. The inclusion $\cong \subseteq \approx_{\times}$ follows immediately from the definition of the naive and the prioritized weak transition relation. In order to apply Fact A.15, we have to establish $\approx_{\times}^+ \subseteq \cong$. This inclusion turns out to be difficult to show. Therefore, we define the equivalence relation

$$\cong_a =_{\text{df}} \{\langle P, Q \rangle \mid C_{PQ}[P] \approx_{\times} C_{PQ}[Q]\} .$$

Here, using the abbreviation $\underline{\mathcal{S}} =_{\text{df}} \underline{\mathcal{S}}(P) \cup \underline{\mathcal{S}}(Q)$, we define

$$C_{PQ}[X] =_{\text{df}} X \mid H_{PQ}$$

and

$$H_{PQ} \stackrel{\text{def}}{=} \underline{c} \cdot \mathbf{0} + D_{\underline{\mathcal{S}}} + \sum_{\substack{L, M \subseteq \underline{\mathcal{S}}(P) \cup \underline{\mathcal{S}}(Q), \\ b \in \Lambda}} \underline{\tau} \cdot \left(\begin{array}{l} \underline{d}_{L, M, b} \cdot H_{PQ} + \\ D_L + e \cdot H_{PQ} + \\ \bar{b} \cdot (\underline{f} \cdot H_{PQ} + D_{\underline{\mathcal{S}}}) \end{array} \right) \oplus D_M).$$

Note that H_{PQ} is well-defined because of Lemma A.5. Moreover, the processes D_L and D_M are defined as in the proof of Theorem 4.4, and the actions $\underline{c}, \underline{d}_{L, M, b}, e, \underline{f}$ are supposed to be ‘fresh’ actions, i.e. they are not in the unprioritized or prioritized sort of the processes P and Q under consideration. By Fact A.15, we may immediately conclude that $\approx_{\times}^+ \subseteq \cong_a$. The other necessary inclusion is established by the following proposition.

Proposition A.16 *The inclusion $\cong_a \subseteq \cong$ holds.*

Proof: Let $P, Q \in \mathcal{P}$ satisfying $P \cong_a Q$, i.e. $C_{PQ}[P] \approx_{\times} C_{PQ}[Q]$ by definition of \cong_a . In the following, we consider four situations, each illustrated by an accompanying diagram. The double

arrows in the diagrams symbolize the naive weak transition relation. The left hand side of each diagram displays a sequence of transitions which we choose. According to the definition of \approx_\times , matching sequences exist which mimic each step by a corresponding naive weak transition. From these transitions, we may extract additional conditions which have to be satisfied according to the semantics of CCS^{prio} . Those conditions are sufficient to conclude that $\underline{\approx}_a$ is a prioritized weak bisimulation. Hence, $P \underline{\approx} Q$, as desired.

Situation 1

Let $P \xrightarrow{m,\tau} P'$ for some process $P' \in \mathcal{P}$ and some location $m \in \text{Loc}$.

Choose for the first step the transition $C_{PQ}[P] \xrightarrow{\tau} P \mid H_{LM}$ where $H_{LM} = \text{df}(\underline{d}_{L,M,b} \cdot H_{PQ} + D_L + e \cdot H_{PQ} + \bar{b} \cdot (\underline{f} \cdot H_{PQ} + D_{\underline{s}})) \oplus D_M$ for some $b \notin \mathcal{S}(P) \cup \mathcal{S}(Q)$, $M = \emptyset$, and $L = \text{df}\{\underline{c} \mid \underline{c} \in (\underline{\mathcal{S}}(P) \cup \underline{\mathcal{S}}(Q)) \setminus \underline{\mathbb{I}}_{[m]}(P)\}$.

Since $C_{PQ}[P] \approx_\times C_{PQ}[Q]$, we have $C_{PQ}[Q] \xrightarrow{\epsilon} W$ for some $W \in \mathcal{P}$. We know that H_{PQ} has to perform a τ -transition to H_{LM} since the action $\underline{d}_{L,M,b}$ is unique. However, Q may be able to perform some prioritized or unprioritized τ -transition to a state $\bar{Q} \in \mathcal{P}$, i.e. we have the following situation:

$$\exists s, t \in \mathbb{N} \forall 0 \leq i < s \forall 0 < j < t \exists Q_i, Q_{s+j} \in \mathcal{P}. Q_0 \equiv Q, Q_{s+t} \equiv \bar{Q}$$

and

1. $Q_i \mid H_{PQ} \xrightarrow{n_i \cdot L, \tau} Q_{i+1} \mid H_{PQ}$ or $Q_i \mid H_{PQ} \xrightarrow{\tau} Q_{i+1} \mid H_{PQ}$,
2. $Q_s \mid H_{PQ} \xrightarrow{n_s \cdot L, \tau} Q_{s+1} \mid H_{LM}$ or $Q_s \mid H_{PQ} \xrightarrow{\tau} Q_{s+1} \mid H_{LM}$, and
3. $Q_{s+j} \mid H_{LM} \xrightarrow{n_{s+j} \cdot L, \tau} Q_{s+j+1} \mid H_{LM}$ or $Q_{s+j} \mid H_{LM} \xrightarrow{\tau} Q_{s+j+1} \mid H_{LM}$.

According to the definition of our semantics, the following conditions must be satisfied if (unprioritized) τ -transitions are participating.

1. $\underline{\mathbb{I}}_{[n_i]}(Q_i) \cap \underline{\mathbb{I}}(H_{PQ}) = \emptyset$, which implies $\underline{\mathbb{I}}_{[n_i]}(Q_i) = \emptyset$,
2. $\underline{\mathbb{I}}_{[n_s]}(Q_s) \cap \underline{\mathbb{I}}(H_{PQ}) = \emptyset$, which implies $\underline{\mathbb{I}}_{[n_s]}(Q_s) = \emptyset$, and
3. $\underline{\mathbb{I}}_{[n_{s+j}]}(Q_{s+j}) \cap \underline{\mathbb{I}}(H_{LM}) = \emptyset$, which implies $\underline{\mathbb{I}}_{[n_{s+j}]}(Q_{s+j}) \subseteq \underline{\mathbb{I}}_{[m]}(P)$ by our choice of L and M .

In the second step, let $P \mid H_{LM} \xrightarrow{m \cdot L, \tau} P' \mid H_{LM}$. Note that this transition is possible according to the semantics of CCS^{prio} and the choice of L and M . Since $P \mid H_{LM} \approx_\times \bar{Q} \mid H_{LM}$, there exists some $\bar{W} \in \mathcal{P}$ such that $\bar{Q} \mid H_{LM} \xrightarrow{\epsilon} \bar{W}$ and $P' \mid H_{LM} \approx_\times \bar{W}$. Now, observe that H_{LM} has no τ -transitions. Moreover, a communication involving H_{LM} leads to a state which has the distinguished action \underline{f} enabled and, therefore, cannot be equivalent to $P' \mid H_{LM}$. This leads to the following two possible cases:

1. $\bar{W} \equiv C_{PQ}[\bar{Q}]$, and we define $\bar{Q}' = \text{df} \bar{Q}$, or
2. $\bar{W} \equiv \bar{Q}' \mid H_{LM}$ for some $\bar{Q}' \in \mathcal{P}$ with $\bar{Q} \xrightarrow{\epsilon} \bar{Q}'$.

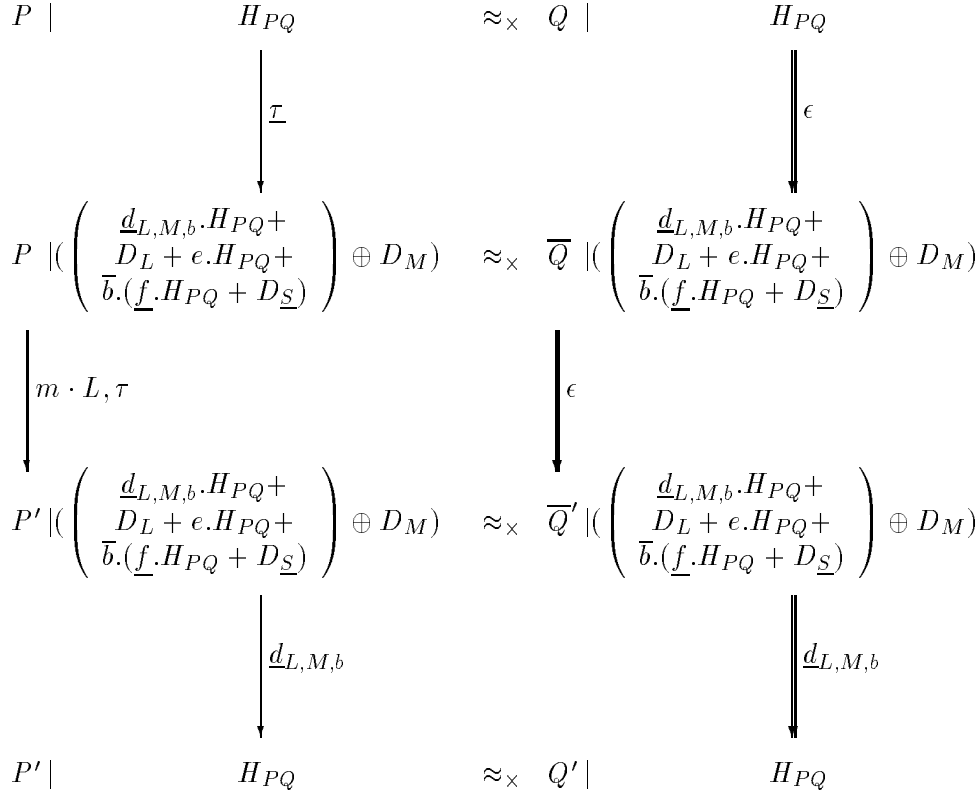


Figure 5: Largest congruence proof - illustration of Situation (1)

The first case corresponds to an ‘idling’ and needs no special attention. In the second case, we have the following situation according to the definition of the naive weak transition relation:

$$\exists \bar{s} \in \mathbb{N} \forall 0 \leq \bar{i} < \bar{s} \exists \bar{Q}_{\bar{i}} \in \mathcal{P}. \bar{Q}_0 \equiv \bar{Q}, \bar{Q}_{\bar{s}} \equiv \bar{Q}' \text{ such that } \bar{Q}_{\bar{i}} \xrightarrow{m_{\bar{i}}, \tau} \bar{Q}_{\bar{i}+1} \text{ or } \bar{Q}_{\bar{i}} \xrightarrow{\tau} \bar{Q}_{\bar{i}+1} .$$

If $\bar{Q}_{\bar{i}} \mid H_{LM} \xrightarrow{m_{\bar{i}}, L, \tau} \bar{Q}_{\bar{i}+1} \mid H_{LM}$, then we have $\mathbb{I}_{[m_{\bar{i}}]}(\bar{Q}_{\bar{i}}) \cap \mathbb{I}(H_{LM}) = \emptyset$ according to the semantics of CCS^{prio} . Because of the choice of L and M , we conclude $\mathbb{I}_{[m_{\bar{i}}]}(\bar{Q}_{\bar{i}}) \subseteq L'$ for $L' = \mathbb{I}_{[m]}(P)$, i.e. $\bar{Q}_{\bar{i}} \xrightarrow{m_{\bar{i}}, \tau} \bar{Q}_{\bar{i}+1}$. Additionally, we have $P' \mid H_{LM} \approx_{\times} \bar{Q}' \mid H_{LM}$.

Finally, let $P' \mid H_{LM} \xrightarrow{\underline{d}_{L,M,b}} C_{PQ}[P']$. In order for $\bar{Q}' \mid H_{LM}$ to match this step with a naive weak $\underline{d}_{L,M,b}$ -transition to a state $W' \in \mathcal{P}$ satisfying $C_{PQ}[P'] \approx_{\times} W'$, the process H_{LM} has to perform a $\underline{d}_{L,M,b}$ -transition to the process H_{PQ} . Moreover, H_{PQ} cannot perform a τ -transition because of the distinguished action \underline{c} . However, \bar{Q} may be able to perform a sequence of prioritized and unprioritized τ -transitions to some process $Q' \in \mathcal{P}$, i.e.

$$\exists s', t' \in \mathbb{N} \forall 0 \leq i' < s' \forall 0 < j' < t' \exists \bar{Q}'_{i'}, \bar{Q}'_{s'+j'} \in \mathcal{P}. \bar{Q}'_0 \equiv \bar{Q}', \bar{Q}'_{s'+j'} \equiv Q'$$

and

1. $\bar{Q}'_{i'} \mid H_{LM} \xrightarrow{n_{i'}, L, \tau} \bar{Q}'_{i'+1} \mid H_{LM}$ or $\bar{Q}'_{i'} \mid H_{LM} \xrightarrow{\tau} \bar{Q}'_{i'+1} \mid H_{LM}$,
2. $\bar{Q}'_{s'} \mid H_{LM} \xrightarrow{\underline{d}_{L,M,b}} \bar{Q}'_{s'} \mid H_{PQ}$, and

$$3. \overline{Q}'_{s'+j'} | H_{PQ} \xrightarrow{n_{s'+j'} \cdot L, \tau} \overline{Q}'_{s'+j'+1} | H_{PQ} \text{ or } \overline{Q}'_{s'+j'} | H_{PQ} \xrightarrow{\tau} \overline{Q}'_{s'+j'+1} | H_{PQ} .$$

According to the definition of our semantics, the following conditions must be satisfied for (un-prioritized) τ -transitions.

1. $\mathbb{I}_{[n_{i'}]}(\overline{Q}'_{i'}) \cap \mathbb{I}(H_{LM}) = \emptyset$, which implies $\mathbb{I}_{[n_{i'}]}(\overline{Q}'_{i'}) \subseteq \mathbb{I}_{[m]}(P)$, and
2. $\mathbb{I}_{[n_{s'+j'}]}(\overline{Q}'_{s'+j'}) \cap \mathbb{I}(H_{PQ}) = \emptyset$, which implies $\mathbb{I}_{[n_{s'+j'}]}(\overline{Q}'_{s'+j'}) = \emptyset$.

Moreover, we have $W' \equiv C_{PQ}[Q']$. Summarizing, we have shown that $Q \xrightarrow[L']{\epsilon} Q'$, according to the definition of the prioritized weak transition relation. Since $C_{PQ}[P'] \approx_{\times} C_{PQ}[Q']$, $\mathcal{S}(P') \subseteq \mathcal{S}(P)$, and $\mathcal{S}(Q') \subseteq \mathcal{S}(Q)$, we also have $C_{P'Q'}[P'] \approx_{\times} C_{P'Q'}[Q']$. Therefore, $P' \approx_a Q'$.

Situation 2

Let $P \xrightarrow{m,a} P'$ for some process $P' \in \mathcal{P}$, some location $m \in \mathcal{Loc}$, and some action $a \in \Lambda$. As illustrated in Figure 6, we let $C_{PQ}[P]$ perform a τ -transition to the process $P | H_{LM}$, where $H_{LM} = \text{df}(\underline{d}_{L,M,b} \cdot H_{PQ} + D_L + e \cdot H_{PQ} + \overline{b} \cdot (\underline{f} \cdot H_{PQ} + D_{\underline{S}})) \oplus D_M$ for $b = a$, $L = \{\underline{\tau} | \underline{c} \in (\mathcal{S}(P) \cup \mathcal{S}(Q)) \setminus \mathbb{I}(P)\}$, and $M = \{\underline{\tau} | \underline{c} \in (\mathcal{S}(P) \cup \mathcal{S}(Q)) \setminus \mathbb{I}_{[m]}(P)\} \setminus L$. Now, $P | H_{LM}$ can perform a τ -transition from location $\langle m \cdot L, o \cdot L \cdot R \rangle$ to $P' | (\underline{f} \cdot H_{PQ} + D_{\underline{S}})$ according to the semantics of CCS^{prio} . Here, the location $o \in \{l, r\}^*$ needs not to be specified more precisely since priorities on different sides of the summation operator are comparable.

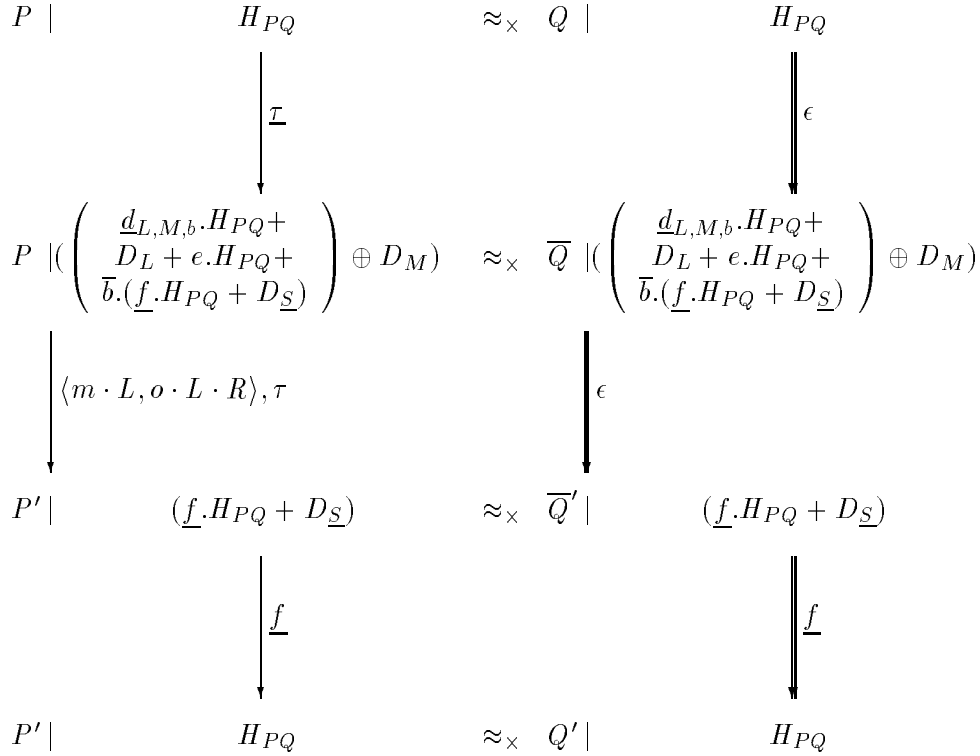


Figure 6: Largest congruence proof - illustration of Situation (2)

Consider the first step. Since $C_{PQ}[P] \approx_{\times} C_{PQ}[P]$, we have $C_{PQ} \xRightarrow{\epsilon}_{\times} W$ for some $W \in \mathcal{P}$. We know that H_{PQ} has to perform a τ -transition to H_{LM} since the action $\underline{d}_{L,M,b}$ is unique. However, Q may be able to perform some prioritized or unprioritized τ -transitions to some state $\overline{Q} \in \mathcal{P}$, i.e. we have the following situation:

$$\exists s, t \in \mathbb{N} \forall 0 \leq i < s \forall 0 < j < t \exists Q_i, Q_{s+j} \in \mathcal{P}. Q_0 \equiv Q, Q_{s+t} \equiv \overline{Q}$$

and

1. $Q_i | H_{PQ} \xrightarrow{n_i \cdot L, \tau} Q_{i+1} | H_{PQ}$ or $Q_i | H_{PQ} \xrightarrow{\tau} Q_{i+1} | H_{PQ}$,
2. $Q_s | H_{PQ} \xrightarrow{n_s \cdot L, \tau} Q_{s+1} | H_{LM}$ or $Q_s | H_{PQ} \xrightarrow{\tau} Q_{s+1} | H_{LM}$, and
3. $Q_{s+j} | H_{LM} \xrightarrow{n_{s+j} \cdot L, \tau} Q_{s+j+1} | H_{LM}$ or $Q_{s+j} | H_{LM} \xrightarrow{\tau} Q_{s+j+1} | H_{LM}$.

According to the definition of our semantics, the following conditions must be satisfied if τ -transitions are involved.

1. $\underline{\mathbb{I}}_{[n_i]}(Q_i) \cap \overline{\mathbb{I}}(H_{PQ}) = \emptyset$, which implies $\underline{\mathbb{I}}_{[n_i]}(Q_i) = \emptyset$,
2. $\underline{\mathbb{I}}_{[n_s]}(Q_s) \cap \overline{\mathbb{I}}(H_{PQ}) = \emptyset$, which implies $\underline{\mathbb{I}}_{[n_s]}(Q_s) = \emptyset$, and
3. $\underline{\mathbb{I}}_{[n_{s+j}]}(Q_{s+j}) \cap \overline{\mathbb{I}}(H_{LM}) = \emptyset$, which implies $\underline{\mathbb{I}}_{[n_{s+j}]}(Q_{s+j}) \subseteq \underline{\mathbb{I}}_{[m]}(P)$.

Now, we take a closer look at the second step. Since $P | H_{LM} \approx_{\times} \overline{Q} | H_{LM}$, we know that some $\overline{W}' \in \mathcal{P}$ exist such that $\overline{Q} | H_{LM} \xRightarrow{\epsilon}_{\times} \overline{W}'$ and $C_{PQ}[P'] \approx_{\times} \overline{W}'$. We first argue that the matching transition must arise from a communication between \overline{Q} and H_{LM} . First, H_{LM} must perform an \overline{a} -transition to $\underline{f}.H_{PQ} + D_{\underline{S}}$ since only that process can engage in the distinguished \underline{f} -transition. Because we have to match a τ -transition, we may conclude that $\overline{Q} \xrightarrow{a}_{\times} \overline{Q}'$ for some $\overline{Q}' \in \mathcal{P}$, i.e.

$$\exists \overline{s}, \overline{t} \in \mathbb{N} \forall 0 \leq \overline{i} < \overline{s} \forall 0 < \overline{j} < \overline{t}. \exists \overline{Q}_{\overline{i}}, \overline{Q}_{\overline{s}+\overline{j}} \in \mathcal{P}. \overline{Q}_0 \equiv \overline{Q}, \overline{Q}_{\overline{s}+\overline{t}} \equiv \overline{Q}'$$

such that

1. $\overline{Q}_{\overline{i}} | H_{LM} \xrightarrow{n_{\overline{i}} \cdot L, \tau} \overline{Q}_{\overline{i}+1} | H_{LM}$ or $\overline{Q}_{\overline{i}} | H_{LM} \xrightarrow{\tau} \overline{Q}_{\overline{i}+1} | H_{LM}$,
2. $\overline{Q}_{\overline{s}} | H_{LM} \xrightarrow{\langle n_{\overline{s}} \cdot L, o \cdot L \cdot R \rangle, \tau} \overline{Q}_{\overline{s}+1} | (\underline{f}.H_{PQ} + D_{\underline{S}})$, and
3. $\overline{Q}_{\overline{s}+\overline{j}} | (\underline{f}.H_{PQ} + D_{\underline{S}}) \xrightarrow{n_{\overline{s}+\overline{j}} \cdot L, \tau} \overline{Q}_{\overline{s}+\overline{j}+1} | (\underline{f}.H_{PQ} + D_{\underline{S}})$ or $\overline{Q}_{\overline{s}+\overline{j}} | (\underline{f}.H_{PQ} + D_{\underline{S}}) \xrightarrow{\tau} \overline{Q}_{\overline{s}+\overline{j}+1} | (\underline{f}.H_{PQ} + D_{\underline{S}})$.

Moreover, we conclude that $\overline{W}' \equiv \overline{Q}' | (\underline{f}.H_{PQ} + D_{\underline{S}})$. According to the definition of our semantics, the following conditions must be satisfied if not a τ -transition is chosen.

1. $\underline{\mathbb{I}}_{[n_{\overline{i}}]}(\overline{Q}_{\overline{i}}) \cap \overline{\mathbb{I}}(H_{LM}) = \emptyset$, which implies $\underline{\mathbb{I}}_{[n_{\overline{i}}]}(\overline{Q}_{\overline{i}}) \subseteq \underline{\mathbb{I}}_{[m]}(P)$,
2. $\underline{\mathbb{I}}_{[n_{\overline{s}}]}(\overline{Q}_{\overline{s}}) \cap \overline{\mathbb{I}}(H_{LM}) = \emptyset$ and $\underline{\mathbb{I}}_{[o]}(H_{LM}) \cap \overline{\mathbb{I}}(\overline{Q}_{\overline{s}}) = \emptyset$, which implies $\underline{\mathbb{I}}_{[n_{\overline{s}}]}(\overline{Q}_{\overline{s}}) \subseteq \underline{\mathbb{I}}_{[m]}(P)$ and $\underline{\mathbb{I}}(\overline{Q}_{\overline{s}}) \subseteq \underline{\mathbb{I}}(P)$, and
3. $\underline{\mathbb{I}}_{[n_{\overline{s}+\overline{j}}]}(\overline{Q}_{\overline{s}+\overline{j}}) \cap \overline{\mathbb{I}}(\underline{f}.H_{PQ} + D_{\underline{S}}) = \emptyset$, which implies $\underline{\mathbb{I}}_{[n_{\overline{s}+\overline{j}}]}(\overline{Q}_{\overline{s}+\overline{j}}) = \emptyset$.

Finally, let $P' | (\underline{f}.H_{PQ} + D_{\underline{S}}) \xrightarrow{\underline{f}} C_{PQ}[P']$. Since $P' | (\underline{f}.H_{PQ} + D_{\underline{S}}) \approx_{\times} \overline{Q}' | (\underline{f}.H_{PQ} + D_{\underline{S}})$, we know that some $W' \in \mathcal{P}$ exist such that $\overline{Q}' | (\underline{f}.H_{PQ} + D_{\underline{S}}) \xrightarrow{\underline{f}}_{\times} W'$ and $C_{PQ}[P'] \approx_{\times} W'$. Since \underline{f} and \underline{c} are distinguished actions, we know that $\underline{f}.H_{PQ} + D_{\underline{S}}$ has to perform its \underline{f} action to the process H_{PQ} . However, \overline{Q}' may be able to perform some prioritized or unprioritized τ -transitions to some state $Q' \in \mathcal{P}$, i.e. we have the following situation:

$$\exists s', t' \in \mathbb{N} \forall 0 \leq i' < s' \forall 0 < j' < t' \exists \overline{Q}'_{i'}, \overline{Q}'_{s'+j'} \in \mathcal{P}. \overline{Q}'_0 \equiv \overline{Q}', \overline{Q}'_{s'+t'} \equiv Q'$$

and

1. $\overline{Q}'_{i'} | (\underline{f}.H_{PQ} + D_{\underline{S}}) \xrightarrow{n_{i'} \cdot L, \tau} \overline{Q}'_{i'+1} | (\underline{f}.H_{PQ} + D_{\underline{S}})$ or $\overline{Q}'_{i'} | (\underline{f}.H_{PQ} + D_{\underline{S}}) \xrightarrow{\tau} \overline{Q}'_{i'+1} | (\underline{f}.H_{PQ} + D_{\underline{S}})$,
2. $\overline{Q}'_{s'} | (\underline{f}.H_{PQ} + D_{\underline{S}}) \xrightarrow{\underline{f}} \overline{Q}'_{s'+1} | H_{PQ}$, and
3. $\overline{Q}'_{s'+j'} | H_{PQ} \xrightarrow{n_{s'+j'} \cdot L, \tau} \overline{Q}'_{s'+j'+1} | H_{PQ}$ or $\overline{Q}'_{s'+j'} | H_{PQ} \xrightarrow{\tau} \overline{Q}'_{s'+j'+1} | H_{PQ}$.

According to the definition of our semantics, the following conditions must be satisfied if τ -transitions are involved.

1. $\underline{\mathbb{I}}_{[n_{i'}]}(\overline{Q}'_{i'}) \cap \underline{\mathbb{I}}((\underline{f}.H_{PQ} + D_{\underline{S}})) = \emptyset$, which implies $\underline{\mathbb{I}}_{[n_{i'}]}(\overline{Q}'_{i'}) = \emptyset$, and
2. $\underline{\mathbb{I}}_{[n_{s'+j'}]}(\overline{Q}'_{s'+j'}) \cap \underline{\mathbb{I}}(H_{PQ}) = \emptyset$, which implies $\underline{\mathbb{I}}_{[n_{s'+j'}]}(\overline{Q}'_{s'+j'}) = \emptyset$.

According to the definition of the prioritized weak transition relation we have shown that $Q \xrightarrow{a}_{L', M'} Q'$ where $L' = \underline{\mathbb{I}}_{[m]}(P)$ and $M' = \underline{\mathbb{I}}(P)$. Since $C_{PQ}[P'] \approx_{\times} C_{PQ}[Q']$, $\underline{\mathcal{S}}(P') \subseteq \underline{\mathcal{S}}(P)$, and $\underline{\mathcal{S}}(Q') \subseteq \underline{\mathcal{S}}(Q)$, also $C_{P'Q'}[P'] \approx_{\times} C_{P'Q'}[Q']$ holds, i.e. $P' \approx_a Q'$ and this part of the proof is finished.

After discussing the previous situations in fully detail, we omit the details of similar parts for the remaining situations.

Situation 3

In this situation, let $P \xrightarrow{\underline{a}} P'$ for some process $P' \in \mathcal{P}$ and some action $\underline{a} \in \underline{A}$. We first choose the transition $C_{PQ}[P] \xrightarrow{\underline{a}} P | H_{LM}$ where $H_{LM} = \text{df}(\underline{d}_{L, M, b}.H_{PQ} + D_L + e.H_{PQ} + \overline{b}.(\underline{f}.H_{PQ} + D_{\underline{S}})) \oplus D_M$ for some $b \notin \underline{\mathcal{S}}(P) \cup \underline{\mathcal{S}}(Q)$, $M = \emptyset$, and $L = \text{df} \underline{S}$. Analogue to Situation (1), but considering $L = \underline{S}$, we may conclude the existence of some process $\overline{Q} \in \mathcal{P}$ such that $C_{PQ}[Q] \xrightarrow{\underline{a}} \overline{Q} | H_{LM}$ and $P | H_{LM} \approx_{\times} \overline{Q} | H_{LM}$.

In the second step, let $P | H_{LM} \xrightarrow{\underline{a}} P' | H_{LM}$. Since $P | H_{LM} \approx_{\times} \overline{Q} | H_{LM}$, there exists some $\overline{W} \in \mathcal{P}$ such that $\overline{Q} | H_{LM} \xrightarrow{\underline{a}}_{\times} \overline{W}$ and $C_{PQ}[P'] \approx_{\times} \overline{W}$. Because of the distinguished action $\underline{d}_{L, M, b}$, we are in the following situation:

$$\exists s, t \in \mathbb{N} \forall 0 \leq i < s \forall 0 < j < t. \exists Q_i, Q_{s+j} \in \mathcal{P}. Q_0 \equiv \overline{Q}, Q_{s+t} \equiv \overline{Q}'$$

such that

1. $Q_i | H_{LM} \xrightarrow{n_i \cdot L, \tau} Q_{i+1} | H_{LM}$ or $Q_i | H_{LM} \xrightarrow{\tau} Q_{i+1} | H_{LM}$,

2. $Q_s \mid H_{LM} \xrightarrow{\underline{\alpha}} Q_{s+1} \mid H_{PQ}$ or (if $\underline{\alpha} = \underline{\tau}$) $Q_s \equiv Q_{s+1}$, and
3. $Q_{s+j} \mid H_{PQ} \xrightarrow{n_{s+j} \cdot L, \tau} Q_{s+j+1} \mid H_{PQ}$ or $Q_{s+j} \mid H_{PQ} \xrightarrow{\underline{\tau}} Q_{s+j+1} \mid H_{PQ}$.

According to the definition of our semantics, the following conditions must be satisfied for unprioritized τ -transitions.

1. $\underline{\mathbb{I}}_{[n_i]}(Q_i) \cap \overline{\mathbb{I}}(H_{LM}) = \emptyset$, which implies $\underline{\mathbb{I}}_{[n_i]}(Q_i) = \emptyset$,
2. $\underline{\mathbb{I}}_{[n_{s+j}]}(Q_{s+j}) \cap \overline{\mathbb{I}}(H_{PQ}) = \emptyset$, which implies $\underline{\mathbb{I}}_{[n_{s+j}]}(Q_{s+j}) = \emptyset$.

Therefore, we have $\overline{Q} \xRightarrow{\underline{\tau}} \overline{Q}'$ and $\overline{W} \equiv \overline{Q}' \mid H_{LM}$.

$$\begin{array}{ccc}
P \mid & H_{PQ} & \approx_{\times} Q \mid & H_{PQ} \\
& \downarrow \underline{\tau} & & \downarrow \epsilon \\
P \mid \left(\begin{array}{c} \underline{d}_{L,M,b} \cdot H_{PQ} + \\ D_L + e \cdot H_{PQ} + \\ \overline{b} \cdot (\underline{f} \cdot H_{PQ} + D_{\underline{S}}) \end{array} \right) \oplus D_M & \approx_{\times} & \overline{Q} \mid \left(\begin{array}{c} \underline{d}_{L,M,b} \cdot H_{PQ} + \\ D_L + e \cdot H_{PQ} + \\ \overline{b} \cdot (\underline{f} \cdot H_{PQ} + D_{\underline{S}}) \end{array} \right) \oplus D_M \\
& \downarrow \underline{\alpha} & & \downarrow \hat{\underline{\alpha}} \\
P' \mid \left(\begin{array}{c} \underline{d}_{L,M,b} \cdot H_{PQ} + \\ D_L + e \cdot H_{PQ} + \\ \overline{b} \cdot (\underline{f} \cdot H_{PQ} + D_{\underline{S}}) \end{array} \right) \oplus D_M & \approx_{\times} & \overline{Q}' \mid \left(\begin{array}{c} \underline{d}_{L,M,b} \cdot H_{PQ} + \\ D_L + e \cdot H_{PQ} + \\ \overline{b} \cdot (\underline{f} \cdot H_{PQ} + D_{\underline{S}}) \end{array} \right) \oplus D_M \\
& \downarrow \underline{d}_{L,M,b} & & \downarrow \underline{d}_{L,M,b} \\
P' \mid & H_{PQ} & \approx_{\times} Q' \mid & H_{PQ}
\end{array}$$

Figure 7: Largest congruence proof - illustration of Situation (3)

Finally, let $P' \mid H_{LM} \xrightarrow{\underline{d}_{L,M,b}} C_{PQ}[P']$. Similar to Situation (1), we can show the existence of some process $Q' \in \mathcal{P}$ satisfying $\overline{Q}' \xRightarrow{\underline{\epsilon}} Q'$ and $C_{PQ}[P'] \approx_{\times} C_{PQ}[Q']$.

Altogether, we have shown that $Q \xRightarrow{\underline{\alpha}} Q'$. Since $C_{PQ}[P'] \approx_{\times} C_{PQ}[Q']$, $\underline{\mathcal{S}}(P') \subseteq \underline{\mathcal{S}}(P)$, and $\underline{\mathcal{S}}(Q') \subseteq \underline{\mathcal{S}}(Q)$, we can also establish $C_{P'Q'}[P'] \approx_{\times} C_{P'Q'}[Q']$. Therefore, $P' \xrightarrow{\underline{\alpha}} Q'$ holds.

Situation 4

Here, we are going to establish Condition (1) of Definition 5.4. We choose the transition sequence displayed in Figure 8, where $L =_{\text{df}} \underline{S} \setminus \overline{\mathbb{I}}(P)$, $M =_{\text{df}} \underline{S}$, and $b \notin \underline{\mathcal{S}}(P) \cup \underline{\mathcal{S}}(Q)$. As before, H_{LM}

denotes the process $(\underline{d}_{L,M,b}.H_{PQ} + D_L + \epsilon.H_{PQ} + \bar{b}.(\underline{f}.H_{PQ} + D_{\underline{S}})) \oplus D_M$. Observe that in each step in Figure 8 the prioritized initial actions in our context are always the complete universe \underline{S} . This is important in order to conclude the following property. Whenever the left parallel component at the right side does an unprioritized τ -transition, the corresponding initial action set is empty. However, the choice of L and the use of the *incomparable* summation operator in the context ensure that the ϵ -transition on the left side in Figure 8 is possible.

Therefore, the first two and the last two steps in Figure 8 are slight modifications of the proofs in the preceding situations. The interesting part of this situation is the matching of the ϵ -transition. Let $Q'' \in \mathcal{P}$ be the process such that $\bar{Q} \xrightarrow{\epsilon} \times Q'' \xrightarrow{\epsilon} \times Q'$ and $Q'' \mid H_{LM} \xrightarrow{o \cdot L \cdot R, \epsilon} W$ for some $W \in \mathcal{P}$. The location $o \in \{l, r\}^*$ needs not to be specified more precisely since priorities on different sides of the summation operator are comparable. According to the semantics of CCS^{prio} we have the constraint $\underline{\mathbb{I}}_{[o \cdot L]}(H_{LM}) \cap \bar{\mathbb{I}}(Q'') = \emptyset$ which implies $\bar{\mathbb{I}}(Q'') \subseteq \bar{\mathbb{I}}(P)$.

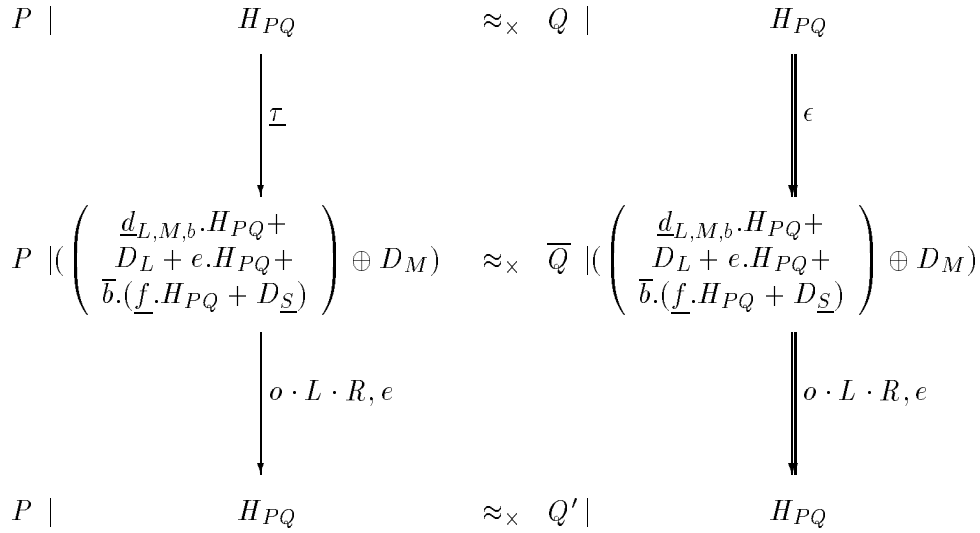


Figure 8: Largest congruence proof - illustration of Situation (4)

Considering also the similarity to the situations discussed above, we conclude that $Q \xrightarrow{\epsilon} Q'' \xrightarrow{\epsilon} Q'$, $\bar{\mathbb{I}}(Q'') \subseteq \bar{\mathbb{I}}(P)$, and $C_{P'Q'}[P'] \approx_{\times} C_{P'Q'}[Q']$, i.e. $P' \underline{\approx}_a Q'$.

Summarizing, we have shown according to Definition 5.4 that $\underline{\approx}_a$ is a prioritized weak bisimulation. Therefore, $P \underline{\approx} Q$, as desired. \square

Considering the proof of Theorem 5.7 we are in the situation illustrated in Figure 9, where an arrow from some relation R_1 to a relation R_2 means that $R_1 \subseteq R_2$.

Combining Propositions A.14, A.16 and the observations made before Proposition A.16, our Main Theorem 5.7 is now a simple corollary because of Fact A.15.

Corollary A.17 *The congruence $\underline{\approx}^+$ is the largest congruence contained in \approx_{\times} .*

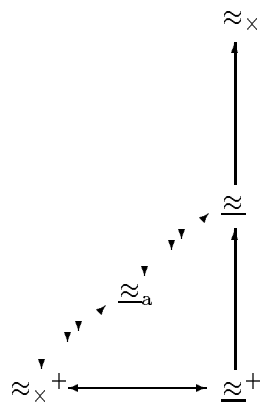


Figure 9: Situation in the proof of Theorem 5.7