Rooted Divergence-Preserving Branching Bisimilarity is a Congruence: A Simpler Proof

David N. Jansen[1,2](✉) and Xinxin Liu[1,2,3](✉)

[1] State Key Laboratory of Computer Science, Institute of Software, Chinese Academy of Sciences, Beijing, China
{dnjansen,xinxin}@ios.ac.cn
[2] University of Chinese Academy of Sciences, Beijing, China
[3] Southwest University, Chongqing, China

Abstract. Van Glabbeek, Luttik and Spanink proved in 2020 [3] that rooted divergence-preserving branching bisimilarity is a congruence for the process specification language consisting of inaction, action prefix, choice, and recursion. In this article we show the same result by using an alternative characterization of bisimulation, so that the heavy notion of bisimulation up to can be spared and a shorter proof obtained.

Keywords: Branching bisimulation · Divergence preservation · Congruence

1 Introduction

Already in the first publication mentioned on his webpage [13], Vaandrager included an extensive discussion of *fairness* in process algebra. He defines this notion as "a certain option is not discarded infinitely often." In particular, this holds for probabilistic choice: if a discrete random experiment is started infinitely often, almost surely every outcome is chosen. In the publication, this was used to prove progress properties of a certain communication protocol where failures occur with a (low but not further specified) probability. Vaandrager admitted that "in reality certain choices are fair, other choices are unfair" (for example, non-probabilistic choices) but decided to postpone the introduction of unfairness. In this article, we are looking at one consequence of unfairness, namely divergence.

We are working on the basis of Milner's CCS to describe and compare behaviours of interacting processes [10]. We mainly consider the subset of *finite-state* processes. They are composed from actions by action prefix, choice and recursion. Full CCS also includes parallel composition and two operators supporting the latter (restriction and relabelling). The recursion operator uses process variables to describe a behaviour X satisfying $X = E$, where X may appear in the process expression E again.

While CCS differs from ACP, the process algebra used by Vaandrager in several publications including [13], we are confident that the superficial differences are less important than the "close relationships between the various process algebras" in terms of the "semantical reality" (using Vaandrager's words from [5]).

N. Jansen et al. (Eds.): Vaandrager Festschrift 2022, LNCS 13560, pp. 358–370, 2022.
https://doi.org/10.1007/978-3-031-15629-8_19

Often, processes are compared by means of bisimilarities, notions to find processes that have equivalent behaviours. For process expressions considered in this paper, bisimilarities can be defined in two ways: an *algebraic* definition defines bisimilarity on expressions that do not contain free process variables first and then extends to all expressions through substitutions; an *operational* definition defines bisimilarity directly on all expressions, including process variables. Generally, the two definitions lead to the same relation, and we find both definitions in the literature, depending on what is easier to use in the context.

In branching and weak bisimulation, it is agreed that internal activity (denoted with the special action symbol τ in CCS) should be regarded as invisible to the behaviour comparison; but what about divergent internal activity $\tau.\tau.\tau\ldots$? Depending on the property that the process is required to satisfy, divergence may be a relevant distinction or not: divergent behaviour—if we do not assume fairness—may delay a required visible behaviour or termination indefinitely.

Branching and weak bisimilarity differ slightly in how they treat internal choice; we will concentrate on branching bisimilarity, like Vaandrager [6] did. Branching and weak bisimilarity are not congruences for the CCS operators. One normally corrects that by a rootedness condition: an *initial* invisible step is inequivalent to doing nothing. The resulting relations are called branching (behaviour) congruence and weak (behaviour) congruence, respectively. The advantage is: it is much easier to reason about a congruence using an equational axiomatisation. Several sound and complete axiomatisations for these congruences (and their divergence-preserving variants) exist [2,7,8,11].

However, in order to do so, one needs to prove that this rootedness condition actually suffices to turn bisimilarity into a congruence. Milner [10] already proved that rooted weak bisimilarity is a congruence using the algebraic definition of weak bisimilarity. Van Glabbeek [2] claims that the proof for branching bisimilarity proceeds similarly. These proofs have to proceed in two steps, following the algebraic definition: first, congruence is proven for all closed processes, and then the proof is extended to the open processes. The divergence-preserving variants were proven congruences later: For rooted divergence-preserving weak bisimilarity, a proof along these lines is found in [8]. A detailed proof that rooted divergence-preserving branching bisimilarity is a congruence has recently appeared as [3]. In all the proofs mentioned above, a general technique called bisimulation up to was used, whose soundness often needs lengthy justification.

This contribution shows that using the operational definition of rooted divergence-preserving branching bisimilarity, one can achieve a shorter proof without resorting to bisimulation up to.

2 Finite-State CCS and Branching Bisimulation

Let $\mathcal{V}$ be an infinite set of *variables,* $\mathcal{A}$ an infinite set of *visible actions,* τ the *invisible action* or *silent move* ($\tau \notin \mathcal{A}$). We write $\mathcal{A}_\tau$ for $\mathcal{A} \cup \{\tau\}$. The *(finite-state process) expressions* are defined by the BNF grammar (for $a \in \mathcal{A}_\tau$ and $X \in \mathcal{V}$):

$$E \quad ::= \quad \mathbf{0} \mid X \mid a.E \mid E + E \mid \mu X.E$$

We denote the set of expressions with $\mathcal{E}$. Informally, the expressions mean:

Inaction: $\mathbf{0}$ is not capable of any action.
Prefix: $a.E$ first performs action a and afterwards behaves as E.
Non-deterministic Choice: $E + F$ can behave either as E or as F.
Recursion: $\mu X.E$ behaves as E, except that whenever X is reached in an execution, then it behaves as $\mu X.E$ again.

We define the *free variables* of an expression as follows:

$$fv(\mathbf{0}) = \varnothing \qquad fv(X) = \{X\} \qquad fv(a.E) = fv(E)$$
$$fv(E + F) = fv(E) \cup fv(F) \qquad fv(\mu X.E) = fv(E) \setminus \{X\}$$

A *closed expression* or *process* is an expression $P \in \mathcal{E}$ with $fv(P) = \varnothing$. The set of all closed expressions is denoted $\mathcal{P}$, and we use P to range over $\mathcal{P}$.

We write $E\{F/X\}$ for the expression obtained by capture-free substitution of F for free occurrences of X in E. We write $E \equiv F$ when E and F are syntactically identical.

We define the semantics of expressions operationally by a transition relation $\longrightarrow$ and a binary relation $\rhd$ between expressions and variables.

Definition 1. *The transition relation* $\longrightarrow \subseteq \mathcal{E} \times \mathcal{A}_\tau \times \mathcal{E}$ *(written* $E \xrightarrow{a} E'$*) is the smallest relation that satisfies:*

1. $a.E \xrightarrow{a} E$.
2. *If* $E \xrightarrow{a} E'$ *then* $E + F \xrightarrow{a} E'$ *and* $F + E \xrightarrow{a} E'$.
3. *If* $E\{\mu X.E/X\} \xrightarrow{a} E'$ *then* $\mu X.E \xrightarrow{a} E'$.

We also write $\Longrightarrow$ *for* $(\xrightarrow{\tau})^*$ *(the transitive-reflexive closure of* $\xrightarrow{\tau}$*) and* $\overset{a}{\Longrightarrow}$ *for* $\Longrightarrow\xrightarrow{a}\Longrightarrow$.

The relation $\rhd \subseteq \mathcal{E} \times \mathcal{V}$ *(written* $E \rhd X$*) is the smallest relation that satisfies:*

1. $X \rhd X$.
2. *If* $E \rhd X$ *then* $E + F \rhd X$ *and* $F + E \rhd X$.
3. *If* $E\{\mu Y.E/Y\} \rhd X$ *then* $\mu Y.E \rhd X$.

Lemma 2. *Let* $E, F, H \in \mathcal{E}$, $a \in \mathcal{A}_\tau$, *and* $X \in \mathcal{V}$. *Then*

1. *If* $H \rhd X$ *and* $E \xrightarrow{a} F$ *then* $H\{E/X\} \xrightarrow{a} F$.
2. *If* $H \xrightarrow{a} H'$, *then* $H\{E/X\} \xrightarrow{a} H'\{E/X\}$.
3. *If* $H\{E/X\} \xrightarrow{a} F$, *then either* $H \rhd X$ *and* $E \xrightarrow{a} F$, *or there is* H' *such that* $H \xrightarrow{a} H'$ *and* $F \equiv H'\{E/X\}$.
4. *If* $H\{E/X\} \rhd Y$, *then either* $H \rhd X$ *and* $E \rhd Y$, *or* $H \rhd Y$.

Proof. This is Lemma 4 in [2], adapted to our notation. □

Lemma 3. *Let $E, F \in \mathcal{E}$, $a \in \mathcal{A}_\tau$, and $X, W \in \mathcal{V}$.*

1. *$\mu X.E \xrightarrow{a} F$ iff there is $E' \in \mathcal{E}$ such that $F \equiv E'\{\mu X.E/X\}$ and $E \xrightarrow{a} E'$.*
2. *$\mu X.E \rhd W$ iff $E \rhd W$ and $W \neq X$.*

Proof. Claim 1 is Lemma 6 in [8]. The proof of Claim 2 is straightforward. □

The following way to define bisimulation is modelled after [11].

Definition 4 (Operational Definition). *A binary relation $R \subseteq \mathcal{E} \times \mathcal{E}$ is a* divergence-preserving branching bisimulation *if R is symmetric and satisfies the following conditions, for every $\langle E, F\rangle \in R$:*

1. ***Simulation of Actions:** Whenever $E \xrightarrow{a} E'$, then either $a = \tau$ and there exists F' such that $F \Longrightarrow F'$ and $\langle E, F'\rangle \in R$ and $\langle E', F'\rangle \in R$, or there exist F', F'' such that $F \Longrightarrow F' \xrightarrow{a} F''$ and $\langle E, F'\rangle \in R$ and $\langle E', F''\rangle \in R$.*
2. ***Simulation of Variables:** Whenever $E \rhd X$, then there exists F' such that $F \Longrightarrow F' \rhd X$ and $\langle E, F'\rangle \in R$.*
3. ***Simulation of Divergence:** Whenever $E \xrightarrow{\tau} E_1 \xrightarrow{\tau} E_2 \xrightarrow{\tau} \cdots$ is an infinite τ-run from E, then there exist E_i on the τ-run and F' such that $F \overset{\tau}{\Longrightarrow} F'$ and $\langle E_i, F'\rangle \in R$.*

Two expressions $E, F \in \mathcal{E}$ are divergence-preserving branching bisimilar *(written $E \approx_{\mathrm{b}}^{\Delta} F$) if there exists a divergence-preserving branching bisimulation containing the pair $\langle E, F\rangle$.*

Definition 2.1 of [3] defines divergence-preserving branching bisimulation for closed expressions only and uses a different clause, denoted (D), instead of our Clause 3 in Definition 4. The two clauses are equivalent for closed expressions according to Proposition 3.1 in [4]. In other words, when applied to processes, Definition 4 results in the same divergence-preserving bisimilarity as Definition 2.1 in [3]. Finally, [4] also proves that $\approx_{\mathrm{b}}^{\Delta}$ is an equivalence relation, and that it is the largest divergence-preserving branching bisimulation.

3 Congruence for Finite-State Processes

As mentioned earlier, branching bisimulation is not a congruence for the CCS operators. In particular, $a.\mathbf{0} \approx_{\mathrm{b}}^{\Delta} \tau.a.\mathbf{0}$ but $a.\mathbf{0} + b.\mathbf{0} \not\approx_{\mathrm{b}}^{\Delta} \tau.a.\mathbf{0} + b.\mathbf{0}$. One normally corrects that by a rootedness condition: an initial invisible step is inequivalent to doing nothing. The resulting relation is called a branching (behaviour) congruence. The advantage is: it is much easier to reason about a congruence using an equational axiomatisation.

Definition 5. *Two process expressions $E, F \in \mathcal{E}$ are* rooted divergence-preserving branching bisimilar *or* divergence-preserving branching congruent *(written $E =_{\mathrm{b}}^{\Delta} F$) if they satisfy:*

1. ***Simulation of Actions:*** *Whenever $E \xrightarrow{a} E'$, then there exists F' such that $F \xrightarrow{a} F'$ and $E' \approx^{\Delta}_{\mathrm{b}} F'$; whenever $F \xrightarrow{a} F'$, then there exists E' such that $E \xrightarrow{a} E'$ and $E' \approx^{\Delta}_{\mathrm{b}} F'$.*
2. ***Simulation of Variables:*** *$E \rhd X$ if and only if $F \rhd X$.*

Lemma 6. *Let $E \in \mathcal{E}$, $X \in \mathcal{V}$. Then $\mu X.E =^{\Delta}_{\mathrm{b}} E\{\mu X.E/X\}$.*

Proof. Immediately follows from the operational semantics of Definition 1. □

Lemma 7. *Let E_0, E, F_0, F be expressions, X and Y be variables. If $E_0 \approx^{\Delta}_{\mathrm{b}} F_0$ and $E =^{\Delta}_{\mathrm{b}} F$, then $E_0\{\mu X.E/Y\} \approx^{\Delta}_{\mathrm{b}} F_0\{\mu X.F/Y\}$.*

Proof. For the given E, F with $E =^{\Delta}_{\mathrm{b}} F$, construct the binary relation S:

$$S = \{\langle G\{\mu X.E/Z\}, H\{\mu X.F/Z\}\rangle \mid G, H \in \mathcal{E}, Z \in \mathcal{V}, \text{ and } G \approx^{\Delta}_{\mathrm{b}} H\}.$$

We show that $S \cup S^{-1}$ is a divergence-preserving branching bisimulation. When this is done, since $E_0 \approx^{\Delta}_{\mathrm{b}} F_0$, so $\langle E_0\{\mu X.E/Y\}, F_0\{\mu X.F/Y\}\rangle \in S$, we obtain $E_0\{\mu X.E/Y\} \approx^{\Delta}_{\mathrm{b}} F_0\{\mu X.F/Y\}$.

It is obvious that $S \cup S^{-1}$ is symmetric. To show that $S \cup S^{-1}$ is a divergence-preserving branching bisimulation, let $\langle C, D\rangle \in S \cup S^{-1}$. We need to check the conditions of Definition 4. If $\langle C, D\rangle \in S$, then according to the construction of S, there exist $G \approx^{\Delta}_{\mathrm{b}} H$ such that $C \equiv G\{\mu X.E/Z\}$ and $D \equiv H\{\mu X.F/Z\}$.

1. **Simulation of Actions.**
   Suppose that $G\{\mu X.E/Z\} \xrightarrow{a} G'$. We distinguish cases following Lemma 2, Claim 3:

   Case 1.1: $G \xrightarrow{a} G_1$ with $G' \equiv G_1\{\mu X.E/Z\}$.

   We further distinguish cases on how H simulates transition $G \xrightarrow{a} G_1$:

   Case 1.1.1: $a = \tau$, there is H_1 with $H \Longrightarrow H_1$ and $G \approx^{\Delta}_{\mathrm{b}} H_1 \approx^{\Delta}_{\mathrm{b}} G_1$.
   Then $H\{\mu X.F/Z\} \Longrightarrow H_1\{\mu X.F/Z\}$, and by the definition S contains the two pairs $\langle G\{\mu X.E/Z\}, H_1\{\mu X.F/Z\}\rangle$ and $\langle G_1\{\mu X.E/Z\}, H_1\{\mu X.F/Z\}\rangle$.

   Case 1.1.2: There are H_1, H_2 with $H \Longrightarrow H_1 \xrightarrow{a} H_2$ and $G \approx^{\Delta}_{\mathrm{b}} H_1$, $G_1 \approx^{\Delta}_{\mathrm{b}} H_2$.
   Then $H\{\mu X.F/Z\} \Longrightarrow H_1\{\mu X.F/Z\} \xrightarrow{a} H_2\{\mu X.F/Z\}$, and by the definition S contains the two pairs $\langle G\{\mu X.E/Z\}, H_1\{\mu X.F/Z\}\rangle$ and $\langle G_1\{\mu X.E/Z\}, H_2\{\mu X.F/Z\}\rangle$.

   Case 1.2: $G \rhd Z$ and $\mu X.E \xrightarrow{a} G'$.
   According to Claim 1 of Lemma 3, there is E_1 such that $E \xrightarrow{a} E_1$ and $G' \equiv E_1\{\mu X.E/X\}$. Since $E =^{\Delta}_{\mathrm{b}} F$, there exists F_1 with $F \xrightarrow{a} F_1$ and $E_1 \approx^{\Delta}_{\mathrm{b}} F_1$, thus by Claim 2 of Lemma 2 $F\{\mu X.F/X\} \xrightarrow{a} F_1\{\mu X.F/X\}$ and $\langle E_1\{\mu X.E/X\}, F_1\{\mu X.F/X\}\rangle \in S$. On the other hand, since $\approx^{\Delta}_{\mathrm{b}}$ is a divergence-preserving branching bisimulation, there is H_1 such that $H \Longrightarrow H_1 \rhd Z$ and $G \approx^{\Delta}_{\mathrm{b}} H_1$. Thus $H\{\mu X.F/Z\} \Longrightarrow H_1\{\mu X.F/Z\} \xrightarrow{a} F_1\{\mu X.F/X\}$ and $\langle G\{\mu X.E/X\}, H_1\{\mu X.F/X\}\rangle \in S$.

   In all cases, we have found a matching transition for $G\{\mu X.E/X\} \xrightarrow{a} G'$.

2. **Simulation of Variables.**
   Suppose $G\{\mu X.E/Z\} \rhd W$. Then according to Lemma 2, Claim 4, either $G \rhd Z$ and $\mu X.E \rhd W$, or $G \rhd W$, and it is routine to show that in both cases there is H_1 with $H\{\mu X.F/Z\} \Longrightarrow H_1\{\mu X.F/Z\} \rhd W$ and $\langle G\{\mu X.E/Z\}, H_1\{\mu X.F/Z\}\rangle \in S$.
3. **Simulation of Divergence.**
   Suppose $G\{\mu X.E/Z\} \equiv G_0 \xrightarrow{\tau} G_1 \xrightarrow{\tau} G_2 \xrightarrow{\tau} \cdots$ is an infinite τ-run from $G\{\mu X.E/Z\}$. Then we distinguish cases by Lemma 2, Claim 3, according to whether $\mu X.E$ ever participates in this infinite τ-run.
   Case 3.1: $\mu X.E$ does not participate in the infinite τ-run.
   Then there is an infinite τ-run $G \xrightarrow{\tau} G'_1 \xrightarrow{\tau} G'_2 \xrightarrow{\tau} \cdots$ such that $G_i \equiv G'_i\{\mu X.E/Z\}$ for $i = 1, 2, \ldots$. Since $\approx_{\mathrm{b}}^{\Delta}$ is a divergence-preserving branching bisimulation, with $G \approx_{\mathrm{b}}^{\Delta} H$ there must be H' and m such that $H \overset{\tau}{\Longrightarrow} H'$ and $G'_m \approx_{\mathrm{b}}^{\Delta} H'$, thus $H\{\mu X.F/Z\} \overset{\tau}{\Longrightarrow} H'\{\mu X.F/Z\}$ and $\langle G'_m\{\mu X.E/Z\}, H'\{\mu X.F/Z\}\rangle \in S$.
   Case 3.2: $\mu X.E$ does participate in the infinite τ-run.
   Then we can find the smallest k such that $G_i \equiv G'_i\{\mu X.E/Z\}$ for $i = 0, \ldots, k$ and $G'_k \rhd Z$ and $\mu X.E \xrightarrow{\tau} G_{k+1}$. By Claim 1 of Lemma 3 there is E' such that $E \xrightarrow{\tau} E'$ and $G_{k+1} \equiv E'\{\mu X.E/X\}$. Then, since $E =_{\mathrm{b}}^{\Delta} F$ there is F' such that $F \xrightarrow{\tau} F'$ with $E' \approx_{\mathrm{b}}^{\Delta} F'$. So $F\{\mu X.F/X\} \xrightarrow{\tau} F'\{\mu X.F/X\}$ and $\langle E'\{\mu X.E/X\}, F'\{\mu X.F/X\}\rangle \in S$. On the other hand we have $G \approx_{\mathrm{b}}^{\Delta} H$, and for the τ-run from G to G'_k, there must be H_1 such that $H \Longrightarrow H_1$ and $G'_k \approx_{\mathrm{b}}^{\Delta} H_1$. Since $G'_k \rhd Z$ there must be H' such that $H_1 \Longrightarrow H' \rhd Z$ and $G'_k \approx_{\mathrm{b}}^{\Delta} H'$. Thus we get $H\{\mu X.F/Z\} \Longrightarrow H'\{\mu X.F/Z\} \xrightarrow{\tau} F'\{\mu X.F/X\}$, and so we have found $F'\{\mu X.F/X\}$ such that $H\{\mu X.F/X\} \overset{\tau}{\Longrightarrow} F'\{\mu X.F/X\}$ and $\langle G_{k+1}, F'\{\mu X.F/X\}\rangle \in S$.

If $\langle C, D\rangle \in S^{-1}$, then $\langle D, C\rangle \in S$, and according to the construction of S, there exist $G \approx_{\mathrm{b}}^{\Delta} H$ such that $D \equiv G\{\mu X.E/Z\}$ and $C \equiv H\{\mu X.F/Z\}$. We can reason as above to show that $\langle C, D\rangle$ also satisfies the conditions of simulation of actions, variables and divergence. This establishes that $S \cup S^{-1}$ is a divergence-preserving branching bisimulation. □

Note that in the proof (Case 3.2), we require congruence $E =_{\mathrm{b}}^{\Delta} F$ (and not only $E \approx_{\mathrm{b}}^{\Delta} F$) to ensure that if E_0 has an infinite run involving $\mu X.E$, then F_0 can take at least one τ-step using $\mu X.F$. This avoids the wrong conclusion from $\tau.X \approx_{\mathrm{b}}^{\Delta} X$ to $\mu X.\tau.X \overset{?}{\approx}_{\mathrm{b}}^{\Delta} \mu X.X$.

With the preparation of Lemma 7, we are in the position to present our main result, which is to provide a more direct proof of the congruence of $=_{\mathrm{b}}^{\Delta}$.

Theorem 8. *$=_{\mathrm{b}}^{\Delta}$ is a congruence on $\mathcal{E}$, i.e. if $E =_{\mathrm{b}}^{\Delta} F$ then $a.E =_{\mathrm{b}}^{\Delta} a.F$, $E + D =_{\mathrm{b}}^{\Delta} F + D$, $D + E =_{\mathrm{b}}^{\Delta} D + F$, and $\mu X.E =_{\mathrm{b}}^{\Delta} \mu X.F$ for arbitrary $a \in \mathcal{A}_\tau$, $D \in \mathcal{E}$, and $X \in \mathcal{V}$.*

Proof. We assume $E =_{\mathrm{b}}^{\Delta} F$ and prove $\mu X.E =_{\mathrm{b}}^{\Delta} \mu X.F$, all other constructions are simple.

Assume $\mu X.E \xrightarrow{a} E'$. We need to prove that this transition can be simulated by some strong transition $\mu X.F \xrightarrow{a} F'$. By Claim 1 of Lemma 3, there exists E'' such that $E \xrightarrow{a} E''$ and $E' \equiv E''\{\mu X.E/X\}$. Because $E =^{\Delta}_{\mathrm{b}} F$, there also exists F'' such that $F \xrightarrow{a} F''$ and $E'' \approx^{\Delta}_{\mathrm{b}} F''$, and by Lemma 7 we then get $E''\{\mu X.E/X\} \approx^{\Delta}_{\mathrm{b}} F''\{\mu X.F/X\}$. Therefore, $\mu X.F \xrightarrow{a} F''\{\mu X.F/X\}$ is the transition simulating $\mu X.E \xrightarrow{a} E'$. The converse statement (a transition $\mu X.F \xrightarrow{a} F'$ can be simulated by $\mu X.E$) is proven analogously.

Clause 2 of Definition 5 is proven similarly, using Claim 2 of Lemma 3. □

Now we have arrived at the conclusion that $=^{\Delta}_{\mathrm{b}}$ is a congruence. But since we used an operational definition of $=^{\Delta}_{\mathrm{b}}$, we need to argue that it is the same relation discussed in [3]. We will make such an argument in the next section. In the rest of this section we will prove some properties of $\approx^{\Delta}_{\mathrm{b}}$ and $=^{\Delta}_{\mathrm{b}}$ which will be used to support the argument in the next section.

Lemma 9. *Let $E, F, G \in \mathcal{E}$ be expressions, X be a variable. If $E =^{\Delta}_{\mathrm{b}} F$, then*

1. $G\{E/X\} =^{\Delta}_{\mathrm{b}} G\{F/X\}$;
2. $E\{G/X\} =^{\Delta}_{\mathrm{b}} F\{G/X\}$.

Proof. $G\{E/X\} =^{\Delta}_{\mathrm{b}} G\{F/X\}$ is easily proved by routine induction on the structure of G, using Theorem 8.

To prove $E\{G/X\} =^{\Delta}_{\mathrm{b}} F\{G/X\}$, we first prove the following fact: If K, H are expressions such that $K \approx^{\Delta}_{\mathrm{b}} H$, then $K\{G/X\} \approx^{\Delta}_{\mathrm{b}} H\{G/X\}$. To see the fact, let Y be a variable such that $Y \notin fv(G)$, by Lemma 6 $\mu Y.G =^{\Delta}_{\mathrm{b}} G\{\mu Y.G/Y\}$, and $G\{\mu Y.G/Y\} \equiv G$ since Y does not occur freely in G. Then

$$\begin{aligned} K\{G/X\} &=^{\Delta}_{\mathrm{b}} K\{\mu Y.G/X\} && \text{(Claim 1 above)} \\ &\approx^{\Delta}_{\mathrm{b}} H\{\mu Y.G/X\} && \text{(Lemma 7)} \\ &=^{\Delta}_{\mathrm{b}} H\{G/X\} && \text{(Claim 1 above)} \end{aligned}$$

With this fact, suppose $E =^{\Delta}_{\mathrm{b}} F$, it is easy to prove that $E\{G/X\} =^{\Delta}_{\mathrm{b}} F\{G/X\}$ by analysing the transitions from $E\{G/X\}$ and $F\{G/X\}$ using Lemma 2. □

We define the *sort* (of visible actions) of a process expression as follows:

$$\begin{array}{rlrl} sort(\mathbf{0}) = \varnothing & & sort(E+F) = sort(E) \cup sort(F) & \\ sort(X) = \varnothing & & sort(\mu X.E) = sort(E) & \\ sort(\tau.E) = sort(E) & & sort(a.E) = sort(E) \cup \{a\} & \text{(if } a \neq \tau\text{)} \end{array}$$

We will write $sort(E, F)$ for $sort(E) \cup sort(F)$.

The following lemma is formally very similar to Lemma 20 in [8], but as that paper uses the algebraic definition and concerns (variants of) weak bisimulation, we cannot copy their proof.

Lemma 10. *Let $E, F \in \mathcal{E}$ be expressions, X be a variable, a be a visible action, $a \notin sort(E, F)$. If $E\{a.\mathbf{0}/X\} =^{\Delta}_{\mathrm{b}} F\{a.\mathbf{0}/X\}$ then $E =^{\Delta}_{\mathrm{b}} F$.*

Proof. First we construct the following binary relation S:

$$S = \{\langle G, H\rangle \mid G, H \in \mathcal{E}, a \notin \mathit{sort}(G, H),\ \text{and}\ G\{a.\mathbf{0}/X\} \approx^{\triangle}_{\mathrm{b}} H\{a.\mathbf{0}/X\}\}.$$

It is routine to show that S is a divergence-preserving branching bisimulation (note that S is symmetric).

Now suppose E, F be expressions, X be a variable, a be a visible action, $a \notin \mathit{sort}(E, F)$, and $E\{a.\mathbf{0}/X\} =^{\triangle}_{\mathrm{b}} F\{a.\mathbf{0}/X\}$. To prove that $E =^{\triangle}_{\mathrm{b}} F$, assume $E \xrightarrow{b} E'$. We need to show that this transition can be simulated by a strong transition $F \xrightarrow{b} F'$. Note that by Claim 2 of Lemma 2, $E \xrightarrow{b} E'$ implies $E\{a.\mathbf{0}/X\} \xrightarrow{b} E'\{a.\mathbf{0}/X\}$, since $E\{a.\mathbf{0}/X\} =^{\triangle}_{\mathrm{b}} F\{a.\mathbf{0}/X\}$, it follows that there is F'' such that $F\{a.\mathbf{0}/X\} \xrightarrow{b} F''$ and $E'\{a.\mathbf{0}/X\} \approx^{\triangle}_{\mathrm{b}} F''$, then by Claim 3 of Lemma 2 there exists F' such that $F \xrightarrow{b} F'$ and $F'' \equiv F'\{a.\mathbf{0}/X\}$ (since $a \notin \mathit{sort}(E, F)$, b cannot be a), so $\langle E', F'\rangle \in S$, thus $E' \approx^{\triangle}_{\mathrm{b}} F'$. Therefore, $F \xrightarrow{a} F'$ is the transition simulating $E \xrightarrow{b} E'$. Clause 2 of Definition 5 is proven similarly, using Claim 4 of Lemma 2. □

Theorem 11. *Let $E, F \in \mathcal{E}$ be expressions, $\{X_1, \dots, X_n\}$ be a set of variables. Then $E =^{\triangle}_{\mathrm{b}} F$ if and only if for arbitrary processes $P_1, \dots, P_n \in \mathcal{P}$ it holds that $E\{P_1/X_1, \dots, P_n/X_n\} =^{\triangle}_{\mathrm{b}} F\{P_1/X_1, \dots, P_n/X_n\}$.*

Proof. We prove the theorem by induction on n. If $n = 0$, then it holds vacuously. Assume the claim holds for n, i.e. $E' =^{\triangle}_{\mathrm{b}} F'$ if and only if for arbitrary processes $P_1, \dots, P_n \in \mathcal{P}$ we have $E'\{P_1/X_1, \dots, P_n/X_n\} =^{\triangle}_{\mathrm{b}} F'\{P_1/X_1, \dots, P_n/X_n\}$. We prove that the claim holds for $n + 1$. For the only if direction, suppose $E =^{\triangle}_{\mathrm{b}} F$ and $P_1, \dots, P_{n+1} \in \mathcal{P}$, we will show

$$E\{P_1/X_1, \dots, P_{n+1}/X_{n+1}\} =^{\triangle}_{\mathrm{b}} F\{P_1/X_1, \dots, P_{n+1}/X_{n+1}\}.$$

In this case, since $E =^{\triangle}_{\mathrm{b}} F$, by Claim 2 of Lemma 9 $E\{P_{n+1}/X_{n+1}\} =^{\triangle}_{\mathrm{b}} F\{P_{n+1}/X_{n+1}\}$, then by the ind. hyp. $E\{P_{n+1}/X_{n+1}\}\{P_1/X_1, \dots, P_n/X_n\} =^{\triangle}_{\mathrm{b}} F\{P_{n+1}/X_{n+1}\}\{P_1/X_1, \dots, P_n/X_n\}$, so

$$\begin{aligned} E\{P_1/X_1, \dots, P_{n+1}/X_{n+1}\} &\equiv E\{P_{n+1}/X_{n+1}\}\{P_1/X_1, \dots, P_n/X_n\} \\ &=^{\triangle}_{\mathrm{b}} F\{P_{n+1}/X_{n+1}\}\{P_1/X_1, \dots, P_n/X_n\} \quad \text{(IH)} \\ &\equiv F\{P_1/X_1, \dots, P_{n+1}/X_{n+1}\}. \end{aligned}$$

Hence the only if direction. For the if direction, we will prove $E =^{\triangle}_{\mathrm{b}} F$ under the assumption that $E\{P_1/X_1, \dots, P_{n+1}/X_{n+1}\} =^{\triangle}_{\mathrm{b}} F\{P_1/X_1, \dots, P_{n+1}/X_{n+1}\}$ for all $P_1, \dots, P_{n+1} \in \mathcal{P}$. In this case, choose $a \notin \mathit{sort}(E, F)$, then for all $P_1, \dots, P_n \in \mathcal{P}$ we have

$$\begin{aligned} E\{a.\mathbf{0}/X_{n+1}\}\{P_1/X_1, \dots, P_n/X_n\} &\equiv E\{P_1/X_1, \dots, P_n/X_n, a.\mathbf{0}/X_{n+1}\} \\ &=^{\triangle}_{\mathrm{b}} F\{P_1/X_1, \dots, P_n/X_n, a.\mathbf{0}/X_{n+1}\} \\ &\equiv F\{a.\mathbf{0}/X_{n+1}\}\{P_1/X_1, \dots, P_n/X_n\}. \end{aligned}$$

Then by the induction hypothesis $E\{a.\mathbf{0}/X_{n+1}\} =^{\triangle}_{\mathrm{b}} F\{a.\mathbf{0}/X_{n+1}\}$. Now by Lemma 10 $E =^{\triangle}_{\mathrm{b}} F$. □

4 Comparison with the Congruence Proof for a Traditional Definition

Here, we compare our proof with the recent proof by Glabbeek, Luttik and Spanink [3], who define divergence-preserving branching bisimulation on closed process expressions only and extend it to open process expressions through substitutions. This more traditional approach is often used to prove that some rooted bisimilarity is a congruence; it follows the basic idea of Milner [9,10]. He was the first to use the so-called "up-to technique": A relation R is called a *weak bisimulation up to* $\approx$ if $P \; R \; Q$ and $P \overset{a}{\Longrightarrow} P'$ imply that there exists some Q' with $Q \overset{a}{\Longrightarrow} Q'$ and $P' \approx R \approx Q'$. It can be shown that if R is a weak bisimulation up to $\approx$ then $\approx R \approx$ is a weak bisimulation.

The important step to prove that some behavioural congruence $=$ is a congruence under recursion, i.e. $E = F$ implies $\mu X.E = \mu X.F$, is: one shows that the symmetric closure of relation $R_{E,F} = \{\langle G\{\mu X.E/X\}, G\{\mu X.F/X\}\rangle \mid G \in \mathcal{E}$ and $fv(G) \subseteq \{X\}\}$ is a bisimulation up to $\approx$ (or, sometimes, a slighty stronger relation).

Glabbeek, in [2], stated that the proof of [10] can be adapted to (non-divergence-preserving) branching bisimilarity. However, the case of divergence-preserving branching bisimilarity was only handled by Glabbeek et al. [3] recently. The latter paper used the bisimulation-up-to technique to prove that rooted divergence-preserving branching bisimilarity is a congruence. They had to vary the relation $R_{E,F}$ for the proof.

More in detail, rooted (divergence-preserving) weak or branching bisimilarity is defined in [3,8,10] by applying Definition 5 only to closed process expressions. for open process expressions, [3] sets:

Definition 12 (Algebraic Definition of Divergence-Preserving Branching Congruence). *Given two expressions $E, F \in \mathcal{E}$ and a vector of variables $\langle X_1, \ldots, X_n\rangle$ that covers their free variables (i.e. $fv(E) \cup fv(F) \subseteq \{X_1, \ldots, X_n\}$), E and F are* algebraically rooted divergence-preserving branching bisimilar *or* algebraically divergence-preserving branching congruent *(written $E =_{\mathrm{a}}^{\Delta} F$) if for arbitrary processes $P_1, \ldots, P_n \in \mathcal{P}$ we have*

$$E\{P_1/X_1, \ldots, P_n/X_n\} =_{\mathrm{b}}^{\Delta} F\{P_1/X_1, \ldots, P_n/X_n\}.$$

The two definitions lead to the same relation on finite-state expressions, i.e. for two expressions E, F it holds that $E =_{\mathrm{a}}^{\Delta} F$ if and only if $E =_{\mathrm{b}}^{\Delta} F$. Here is a direct proof using Theorem 11: Let $E, F \in \mathcal{E}$ and $fv(E) \cup fv(E) = \{X_1, \ldots, X_n\}$, then $E =_{\mathrm{a}}^{\Delta} F$ iff (by Definition 12) for arbitrary processes $P_1, \ldots, P_n \in \mathcal{P}$, we have $E\{P_1/X_1, \ldots, P_n/X_n\} =_{\mathrm{b}}^{\Delta} F\{P_1/X_1, \ldots, P_n/X_n\}$ iff (by Theorem 11) $E =_{\mathrm{b}}^{\Delta} F$. Similar proofs are found e.g. in [2] as Propositions 2[1] and 3. Van Glabbeek writes, however, "defining $\approx_{\mathrm{b}}^{\Delta}$ on open process expressions ... does not carry over to full CCS."

Then, [3] introduces their variant of "bisimulation up to $\approx_{\mathrm{b}}^{\Delta}$":

[1] Proposition 2 in [2] claims more than we need. An error in another part of the proof has been pointed out and corrected in [1].

Definition 13 ([3], Definition 3.10). *The symmetric relation R on $\mathcal{P}$ is a* rooted divergence-preserving branching bisimulation up to $\approx_{\mathrm{b}}^{\Delta}$ *if it satisfies the conditions:*

1. ***Root Condition:*** *If $P \xrightarrow{a} P'$, then there exists Q' such that $Q \xrightarrow{a} Q'$ and $P' \approx_{\mathrm{b}}^{\Delta} R \approx_{\mathrm{b}}^{\Delta} Q'$.*
2. ***Simulation of Actions:*** *If $P \Longrightarrow P'' \xrightarrow{\hat{a}} P'$, then there exist Q'' and Q such that $Q \Longrightarrow Q'' \xrightarrow{\hat{a}} Q'$ and $P'' \approx_{\mathrm{b}}^{\Delta} R \approx_{\mathrm{b}}^{\Delta} Q''$ and $P' \approx_{\mathrm{b}}^{\Delta} R \approx_{\mathrm{b}}^{\Delta} Q'$. (Here, $P'' \xrightarrow{\hat{a}} P'$ means: either $P'' \xrightarrow{a} P'$, or $a = \tau$ and $P'' \equiv P'$.)*
3. ***Simulation of Divergence:*** *If there exists an infinite sequence of closed process expressions $(P_k)_{k \in \omega}$ such that $P = P_0$ and $P_k \xrightarrow{\tau} P_{k+1}$ for all $k \in \omega$, then there also exists an infinite sequence of closed process expressions $(Q_\ell)_{\ell \in \omega}$ and a mapping $\sigma : \omega \to \omega$ such that $Q = Q_0$ and $Q_\ell \xrightarrow{\tau} Q_{\ell+1}$ and $P_{\sigma(\ell)} \approx_{\mathrm{b}}^{\Delta} R \approx_{\mathrm{b}}^{\Delta} Q_\ell$ for all $\ell \in \omega$.*

While they define simulation of divergence differently, it can be shown that this condition is equivalent to our usual condition (see the explanation after Definition 4). Also note that $\approx_{\mathrm{b}}^{\Delta}$ is only used to relate closed expressions, where there is no difference between an operational and an algebraic definition.

They then go on to prove that if R is a rooted divergence-preserving branching bisimulation up to $\approx_{\mathrm{b}}^{\Delta}$ then $R \subseteq =_{\mathrm{b}}^{\Delta}$ (note that all the relations are between processes). Then comes the hardest part: they spend almost three pages to prove that the relation $R_{E,F}$ is a rooted divergence-preserving branching bisimulation up to $\approx_{\mathrm{b}}^{\Delta}$ if $fv(E) \cup fv(F) \subseteq \{X\}$. After that, they can use this property to prove quickly that $=_{\mathrm{a}}^{\Delta}$ is indeed a congruence under $\mu X.\_$ for all process expressions.

In comparison, our proof is much shorter. Still, the hard work of [3] is not completely lost: we will see shortly that our method does not extend to further CCS operators. The reason that we can achieve a shorter proof is that the operational definition of bisimulation (Definition 4) allows us to discuss operational behaviour directly on expressions containing free variables, so that we can construct the relation

$$S = \{\langle G\{\mu X.E/Z\}, H\{\mu X.F/Z\}\rangle \mid G, H \in \mathcal{E}, Z \in \mathcal{V}, \text{ and } G \approx_{\mathrm{b}}^{\Delta} H\}$$

in the proof of Lemma 7 and prove $S \cup S^{-1}$ to be a divergence-preserving branching bisimulation, instead of constructing the less powerful (more strict)

$$R_{E,F} = \{\langle G\{\mu X.E/X\}, G\{\mu X.F/X\}\rangle \mid G \in \mathcal{E} \text{ and } fv(G) \subseteq \{X\}\}$$

and then having to use the bisimulation-up-to technique. In this way we successfully avoided the complication relating to this technique. It is interesting to note the similarities and differences between the constructions of S and $R_{E,F}$. In fact, [3] already used a construction similar to S in the proof of Lemma 3.6, which corresponds to Claim 2 of Lemma 9 of this paper. In other words, following the idea of this paper, the proof of Lemma 3.6 in [3] can be generalized to a proof that $\mu X.\_$ preserves $=_{\mathrm{a}}^{\Delta}$, thus avoiding the use of the bisimulation-up-to technique altogether.

5 Extending the Proof to Full CCS?

Thus far we did only look at finite-state CCS; however, there are operators for parallelism as well. Full CCS has the following grammar:

$$E \quad ::= \quad \mathbf{0} \mid X \mid a.E \mid E + E \mid \mu X.E \mid E|F \mid E\backslash H \mid E[f]$$

where $a \in Act := \mathcal{A} \cup \overline{\mathcal{A}} \cup \{\tau\}$, $X \in \mathcal{V}$ as above, $H \subseteq \mathcal{A}$, and $f : \mathcal{A} \to Act$. We denote the set of process expressions in full CCS with $\mathcal{E}_{\mathrm{par}}$. Informally, the new expressions mean:

Actions with Overline $\overline{a}$: Actions a and $\overline{a}$ can synchronize in parallel processes. We set $\overline{\overline{a}} = a$ and $\overline{\tau} = \tau$.

Parallel Composition: $E|F$ interleaves the behaviours of E and F. Additionally, if $E \xrightarrow{a} E'$ and $F \xrightarrow{\overline{a}} F'$ for some $a \in \mathcal{A} \cup \overline{\mathcal{A}}$, the parallel composition has the behaviour $E|F \xrightarrow{\tau} E'|F'$. This models a synchronisation between the processes.

Restriction: $E\backslash H$ can do all behaviours of E except the actions in H. This operator is used to forbid $E|F$ from taking certain steps without synchronisation.

Relabelling: Whenever E can do action a or $\overline{a}$, then $E[f]$ can do action $f(a)$ or $\overline{f(a)}$, respectively, instead.

The transition relation, which is the first part of Definition 1, can easily be extended to include these constructs:

Definition 14. *The transition relation $\longrightarrow \subseteq \mathcal{E}_{par} \times Act \times \mathcal{E}_{par}$ (written $E \xrightarrow{a} E'$) is the smallest relation that satisfies the clauses given in Definition 1 and:*

1. *If $E \xrightarrow{a} E'$, then $E|F \xrightarrow{a} E'|F$ and $F|E \xrightarrow{a} F|E'$.*
2. *If $E \xrightarrow{a} E'$ and $F \xrightarrow{\overline{a}} F'$ for some $a \in \mathcal{A} \cup \overline{\mathcal{A}}$, then $E|F \xrightarrow{\tau} E'|F'$.*
3. *If $E \xrightarrow{a} E'$ and $a, \overline{a} \notin H$, then $E\backslash H \xrightarrow{a} E'\backslash H$.*
4. *If $E \xrightarrow{a} E'$, then $E[f] \xrightarrow{f(a)} E'[f]$. If $E \xrightarrow{\overline{a}} E'$, then $E[f] \xrightarrow{\overline{f(a)}} E'[f]$. If $E \xrightarrow{\tau} E'$, then $E[f] \xrightarrow{\tau} E'[f]$.*

However, the relation $\rhd$, which is the second part of Definition 1, does not convey enough information to form the basis of a correct operational definition of bisimulation. In particular, one would like to define $X|X \rhd X$, but how can this relation then distinguish expression $X \in \mathcal{E}$ from $X|X \in \mathcal{E}_{\mathrm{par}}$? Perhaps van Glabbeek's notation of [2,3] can help; they write $E \xrightarrow{X} \mathbf{0}$ instead of our $E \rhd X$, and this notation could be extended to something like $E|E \xrightarrow{X} E|\mathbf{0} \xrightarrow{X} \mathbf{0}|\mathbf{0}$. In any case, this would also require extending Claim 3 of Lemma 2: The expression $E = a.X|X$ cannot do a τ step, but $E\{\overline{a}.\mathbf{0}/X\}$ can. Similarly, E is not divergent, but $E\{\mu Z.\overline{a}.a.Z/X\}$ is, even though neither E nor $\mu Z.\overline{a}.a.Z$ can do τ steps. The proof of our central Lemma 7 would need a corresponding extension.

Even if that may be possible, other difficulties remain: van Glabbeek's notation is not informative enough to describe a restriction like $X\backslash\{a\}|\overline{a}.\mathbf{0}$ or a relabelling like $X[a \mapsto \overline{a}]|X$. We could not find an obvious extension of the above proof method to full CCS. We are working on an extension [12], using the bisimulation-up-to technique.

Weak bisimulation (not divergence-preserving) has been shown to be a congruence for all operators except +, and rooted weak bisimulation is a congruence for all operators [10]. However, these proofs use the bisimulation-up-to technique.

We assume that Robin Milner was aware of (some of) these difficulties and therefore chose to switch between the two definitions and the two methods: in 1986 [9] he used what we called the algebraic definition to prove congruence for full CCS, while his article [11] used the operational definition with the smaller set of operators (to prove the completeness of his axiomatisation of rooted weak bisimilarity). The remark of van Glabbeek, cited above on page 9, also suggests that van Glabbeek was aware of the discrepancies. Perhaps the long-term goal of proving the congruence property for all CCS operators motivated him and the other authors of [3] to pursue the longer path.

References

1. Basten, T.: Branching bisimilarity is an equivalence indeed! Inf. Proc. Lett. **58**(3), 141–147 (1996). https://doi.org/10.1016/0020-0190(96)00034-8
2. van Glabbeek, R.J.: A complete axiomatization for branching bisimulation congruence of finite-state behaviours. In: Borzyszkowski, A.M., Sokołowski, S. (eds.) Mathematical Foundations of Computer Science 1993. LNCS, vol. 711, pp. 473–484. Springer, Berlin (1993). https://doi.org/10.1007/3-540-57182-5_39
3. van Glabbeek, R., Luttik, B., Spaninks, L.: Rooted divergence-preserving branching bisimilarity is a congruence. Log. Meth. Comput. Sci. **16**(3), 14:1–14:16 (2020). https://doi.org/10.23638/LMCS-16(3:14)2020
4. van Glabbeek, R., Luttik, B., Trčka, N.: Branching bisimilarity with explicit divergence. Fundam. Inform. **93**(4), 371–392 (2009). https://doi.org/10.3233/FI-2009-109
5. van Glabbeek, R., Vaandrager, F.: Modular specification of process algebras. Theor. Comput. Sci. **113**(2), 293–348 (1993). https://doi.org/10.1016/0304-3975(93)90006-F
6. Groote, J.F., Vaandrager, F.: An efficient algorithm for branching bisimulation and stuttering equivalence. In: Paterson, M.S. (ed.) Automata, Languages and Programming. LNCS, vol. 443, pp. 626–638. Springer, New York (1990). https://doi.org/10.1007/BFb0032063
7. Liu, X., Yu, T.: A complete axiomatisation for divergence preserving branching congruence of finite-state behaviours. In: 2021 36th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS), pp. 1–13. IEEE, [s.l.] (2021). https://doi.org/10.1109/LICS52264.2021.9470647
8. Lohrey, M., D'Argenio, P.R., Hermanns, H.: Axiomatising divergence. Inf. Comput. **203**(2), 115–144 (2005). https://doi.org/10.1016/j.ic.2005.05.007
9. Milner, R.: Lectures on a calculus for communicating systems. In: Broy, M. (ed.) Control Flow and Data Flow: Concepts of Distributed Programming, pp. 205–228. Springer, Berlin (1986). https://doi.org/10.1007/978-3-642-82921-5_5

10. Milner, R.: Communication and Concurrency. Prentice-Hall, New York (1989)
11. Milner, R.: A complete axiomatisation for observational congruence of finite-state behaviours. Inf. Comput. **81**(2), 227–247 (1989). https://doi.org/10.1016/0890-5401(89)90070-9
12. Sun, Q., Jansen, D.N., Liu, X., Zhang, W.: Divergence-preserving congruences for CCS. Manuscript under submission (2022)
13. Vaandrager, F.W.: Verification of two communication protocols by means of process algebra. Report CS-R6808, CWI, Amsterdam (1986). https://ir.cwi.nl/pub/6298