

The Saga of Finite Equational Bases over BCCSP

Taolue Chen¹ and Wan Fokkink²

¹ DACS & FMT, University of Twente, The Netherlands

² Department of Theoretical Computer Science, Vrije Universiteit Amsterdam, The Netherlands

Abstract. Van Glabbeek (1990) presented the “linear time – branching time spectrum”, a lattice of behavioral semantics over labeled transition systems ordered by inclusion. He studied these semantics in the setting of the basic process algebra BCCSP, and tried to give finite complete axiomatizations for them. Obtaining such axiomatizations in concurrency theory often turns out to be difficult, which had raised a host of open questions that were the subject of intensive research in recent years. All of these questions are settled over BCCSP now, either positively by giving a finite complete axiomatization, or negatively by proving that such an axiomatization does not exist. This essay reports on these results.

1 Introduction

Labeled transition systems (LTSs) constitute a fundamental model of concurrent computation which is widely used in light of its flexibility and applicability. They model processes by explicitly describing their states and transitions from state to state, together with the actions that produce them. Several notions of behavioral equivalences have been proposed, with the aim to identify those states of LTSs that afford the same observations. The lack of consensus on what constitutes an appropriate notion of observable behavior for reactive systems has led to a large number of proposals for behavioral equivalences for concurrent processes.

Van Glabbeek [22,23] presented the linear time – branching time spectrum of behavioral *pre-orders* and *equivalences* for finitely branching, concrete, sequential processes, which is a lattice of known behavioral preorders and equivalences over LTSs, ordered by inclusion. The semantics in the spectrum are based on *simulation* notions or on *decorated traces*. **Fig. 1** depicts the linear time – branching time spectrum³, where an arrow from one semantics to another means that the source of the arrow is finer than the target.

To give further insight into the identifications made by the respective behavioral equivalences in his spectrum, van Glabbeek [22,23] studied them in the setting of the process algebra BCCSP, which contains only the basic process algebraic operators from CCS and CSP, but is sufficiently powerful to express all finite synchronization trees. In particular, he associated with every behavioral equivalence in his spectrum a sound equational *axiomatization*, i.e., a collection of equations of behaviorally equivalent BCCSP terms. Most of the axiomatizations were also shown to be “complete” in the sense that whenever two *closed* BCCSP terms (i.e., terms containing no occurrence of variables) are behaviorally equivalent, then the axiomatization admits a derivation in equational logic of the corresponding equation.

In general, axiomatizations arise from the desire of isolating the features that are common to a collection of algebraic structures, namely, their semantics models. One requires that a set of axioms is *sound* (i.e., if two behaviors can be equated, then they are semantically related), and one desires that it is *complete* (i.e., if two behaviors are semantically related, then they can be equated). Having defined a semantic model for a process algebra in terms of LTSs, it is natural to

³Note that the *completed simulation* and *impossible futures* semantics were missing in the original spectrum [22,23], but are included here.

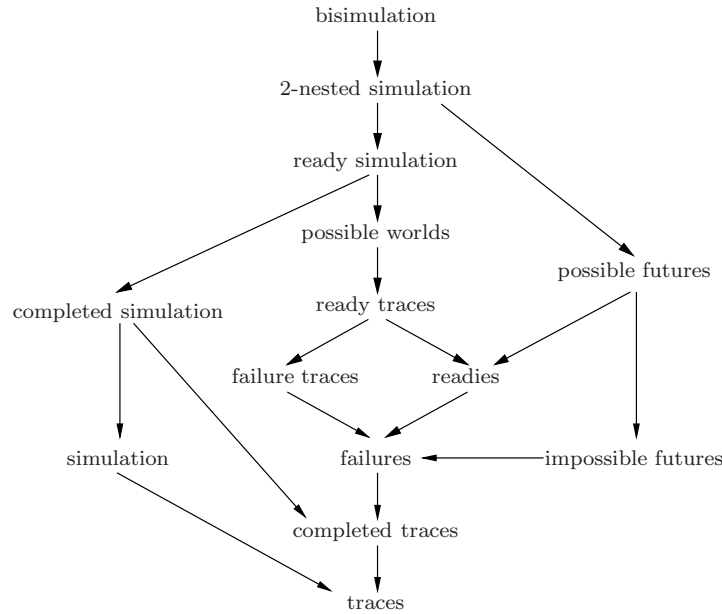


Fig. 1. The linear time – branching time spectrum

study its *(in)equational theory*, that is, the collection of (in)equations that are valid in the given model. The key questions here are:

- Are there reasonably informative *sound* and *complete* axiomatizations for the chosen semantic model?
- Does the algebra of LTSs modulo the chosen notion of behavioral semantics afford a *finite* (in)equational axiomatization?

A sound and complete axiomatization of a behavioral congruence (resp. precongruence) yields a purely syntactic characterization, independent of LTSs and of the actual details of the definition of the chosen behavioral equivalence (resp. preorder), of the semantics of the process algebra. This bridge between syntax and semantics plays an important role in both the theory and the practice of process algebras. From the theoretical perspective, complete axiomatizations of behavioral preorders or equivalences capture the essence of different notions of semantics for processes in terms of a basic collection of identities, and this often allows one to compare semantics which might have been defined in very different styles and frameworks. From the point of view of practice, these proof systems can be used to perform system verifications in a purely syntactic way using general purpose theorem provers or proof checkers, and form the basis of purpose-built axiomatic verification tools like, e.g. PAM [15]. Hence a positive answer to the first basic question raised above is therefore not just theoretically pleasing, but has potential practical applications.

In literature, different forms of completeness are often considered. For BCCSP, an axiomatization is said to be *complete* if any two behaviorally equivalent BCCSP terms (not just the closed ones) can be equated; completeness for *closed* terms only we shall henceforth refer to as *ground-completeness*. The notion of completeness of an axiomatization also relates the *proof-theoretic* notion of derivability using the rules of equational logic with the *model-theoretic* one of “validity in a model”. From a proof-theoretic perspective, a useful property of a ground-complete axiomatization is that whenever all closed instances of an equation can be derived from it, then the

equation itself can also be derived from it; this property is generally referred to as ω -*completeness*. For theorem proving applications, it is particularly convenient if an axiomatization is ω -complete, because it means that proofs by (structural) induction can be avoided in favor of purely equational reasoning; see [14]. In [12] it was argued that ω -completeness is desirable for the partial evaluation of programs. It turns out that completeness and ω -completeness are closely related properties of an axiomatization. Indeed, in the setting of BCCSP, it is not hard to show that any complete axiomatization is also ω -complete. Conversely any ω -complete axiomatization is, by definition, complete.

In universal algebra, a complete axiomatization is referred to as a *basis* for the equational theory of the algebra it axiomatizes. The existence of a finite basis for an equational theory is a classic topic of study in universal algebra (see, e.g., [18]), dating back to Lyndon [16]. Murskii [21] proved that “almost all” *finite* algebras (namely all quasi-primal ones) are finitely based, while in [20] he presented an example of a three-element algebra that has no finite basis. Henkin [13] showed that the algebra of naturals with addition and multiplication is finitely based, while Gurevič [11] showed that after adding exponentiation the algebra is no longer finitely based. McKenzie [17] settled Tarski’s Finite Basis Problem in the negative, by showing that the general question whether a finite algebra is finitely based is undecidable.

Process algebra, as a branch of universal algebra and equational logic, naturally features the study of results pertaining to the existence or non-existence of finite bases for algebras modulo given semantics. Many of the existing axiomatizations of behavioral semantics over expressive process algebras studied in concurrency theory are powerful enough to prove all of the valid equalities between *closed* terms; they are ground-complete, but *not* ω -complete. In fact, obtaining ω -complete axiomatizations in concurrency theory often turns out to be a difficult question, even in the setting of simple languages like BCCSP. This has raised a host of open questions that have been the subject of intensive investigation by process algebraists in recent years. Fortunately, these questions are finally settled for all the semantics in the linear time – branching time spectrum in the setting of BCCSP, either positively by giving a finite sound and ground-complete axiomatization that is ω -complete, or negatively by proving that such a finite basis for the equational theory does not exist. In this essay, we report on these positive and negative results. We hope that this will contribute to their dissemination in our research community and stimulate further investigations.

This essay is organized as follows: Section 2 gives some preliminaries, in particular, the linear time – branching time spectrum and process algebra BCCSP; Section 3 reports on results on axiomatizations of behavioral semantics in the spectrum over the language BCCSP; Section 4 gives a summary.

2 Preliminaries

2.1 The linear time – branching time spectrum

A *labeled transition system* consists of a set of states S , with typical element s , and a transition relation $\rightarrow \subseteq S \times L \times S$, where L is a set of labels ranged over by a . We write $s \xrightarrow{a} s'$ if the triple (s, a, s') is an element of \rightarrow . The set $\mathcal{I}(s)$ consists of those labels a for which there exists s' such that $s \xrightarrow{a} s'$. Let $a_1 \cdots a_k$ be a sequence of labels; we write $s \xrightarrow{a_1 \cdots a_k} s'$ if there are states s_0, \dots, s_k such that $s = s_0 \xrightarrow{a_1} \cdots \xrightarrow{a_k} s_k = s'$.

First we define six semantics based on decorated versions of execution traces.

Definition 1 (Decorated Traces). *Assume a labeled transition system.*

- A sequence $a_1 \cdots a_k$, with $k \geq 0$, is a trace of a state s if there is a state s' such that $s \xrightarrow{a_1 \cdots a_k} s'$. It is a completed trace of s if moreover $\mathcal{I}(s') = \emptyset$.

- A pair $(a_1 \cdots a_k, B)$, with $k \geq 0$ and $B \subseteq A$, is a ready pair of a state s_0 if there is a sequence of transitions $s_0 \xrightarrow{a_1} \cdots \xrightarrow{a_k} s_k$ with $\mathcal{I}(s_k) = B$. It is a failure pair of s_0 if there is such a sequence with $\mathcal{I}(s_k) \cap B = \emptyset$.
- A sequence $B_0 a_1 B_1 \dots a_k B_k$, with $k \geq 0$ and $B_0, \dots, B_k \subseteq A$, is a ready trace of a state s_0 if there is a sequence of transitions $s_0 \xrightarrow{a_1} \cdots \xrightarrow{a_k} s_k$ with $\mathcal{I}(s_i) = B_i$ for $i = 0, \dots, k$. It is a failure trace of s_0 if there is such a sequence with $\mathcal{I}(s_i) \cap B_i = \emptyset$ for $i = 0, \dots, k$.

We write $s \lesssim_{\square} s'$ with $\square \in \{\text{T, CT, R, F, RT, FT}\}$ if the traces, completed traces, ready pairs, failure pairs, ready traces, or failure traces, respectively, of s are included in those of s' .

Next we define five semantics based on simulation.

Definition 2 (Simulations). Assume an A -labeled transition system.

- A binary relation \mathcal{R} on states is a simulation if $s_0 \mathcal{R} s_1$ and $s_0 \xrightarrow{a} s'_0$ imply $s_1 \xrightarrow{a} s'_1$ for some state s'_1 with $s'_0 \mathcal{R} s'_1$.
- A simulation \mathcal{R} is a completed simulation if $s_0 \mathcal{R} s_1$ and $\mathcal{I}(s_0) = \emptyset$ imply $\mathcal{I}(s_1) = \emptyset$.
- A simulation \mathcal{R} is a ready simulation if $s_0 \mathcal{R} s_1$ and $a \notin \mathcal{I}(s_0)$ imply $a \notin \mathcal{I}(s_1)$.
- A simulation \mathcal{R} is a 2-nested simulation if \mathcal{R}^{-1} is included in a simulation.
- A bisimulation is a symmetric simulation.

We write $s \lesssim_{\square} s'$ with $\square \in \{\text{S, CS, RS, 2N}\}$ if there exists a simulation, completed simulation, ready simulation or 2-nested simulation \mathcal{R} , respectively, with $s \mathcal{R} s'$. We write $s \rightleftharpoons s'$ if there exists a bisimulation \mathcal{R} with $s \mathcal{R} s'$.

Finally, we define three semantics based on (im)possible futures and on possible worlds.

Definition 3 ((Im)Possible futures/worlds). Assume an A -labeled transition system.

- A pair $(a_1 \cdots a_k, X)$, with $n \geq 0$ and $X \subseteq A^*$ is a possible future of a state s_0 if there is a sequence of transitions $s_0 \xrightarrow{a_1} \cdots \xrightarrow{a_k} s_k$ where X is the set of traces of s_k .
- A pair $(a_1 \cdots a_k, X)$, with $k \geq 0$ and $X \subseteq A^*$, is an impossible future of a state s_0 if there is a sequence of transitions $s_0 \xrightarrow{a_1} \cdots \xrightarrow{a_k} s_k$ for some state s_k with $\mathcal{T}(s_k) \cap X = \emptyset$.
- A state s is deterministic if for each $a \in \mathcal{I}(s)$ there is exactly one state s_0 such that $s \xrightarrow{a} s_0$, and moreover s_0 is deterministic. A state s is a possible world of a state s_0 if s is deterministic and $s \mathcal{R} s_0$ for some ready simulation \mathcal{R} .

We write $s \lesssim_{\square} s'$ with $\square \in \{\text{PF, IF, PW}\}$ if the possible futures, impossible futures or the possible worlds, respectively, of s are included in those of s' .

In general, we write $s \simeq_{\square} s'$ if both $s \lesssim_{\square} s'$ and $s' \lesssim_{\square} s$ for $\square \in \{\text{T, CT, R, F, RT, FT, S, CS, RS, 2N, PF, IF, PW}\}$.

2.2 BCCSP

BCCSP is a basic process algebra for expressing finite process behavior. Its signature consists of the constant $\mathbf{0}$, the binary operator $+$, and unary prefix operators $a-$, where a ranges over a nonempty set A of actions, called the *alphabet*, with typical elements a, b, c . Intuitively, closed BCCSP terms, denoted p, q, r , represent finite process behaviors, where $\mathbf{0}$ does not exhibit any behavior, $p + q$ offers a choice between the behaviors of p and q , and ap executes action a to transform into p . This intuition is captured by the transition rules below, in which a ranges over A . They give rise to A -labeled transitions between BCCSP terms.

$$\frac{}{ax \xrightarrow{a} x} \quad \frac{x \xrightarrow{a} x'}{x + y \xrightarrow{a} x'} \quad \frac{y \xrightarrow{a} y'}{x + y \xrightarrow{a} y'}$$

We also assume a countably infinite set V of variables; x, y, z denote elements of V , and X, Y, Z denote finite subsets of V . Open BCCSP terms, which may contain variables from V , are denoted t, u, v, w . A term t is called a *prefix* if $t = at'$ for some $a \in A$ and for some term t' .

The preorders \lesssim in the linear time – branching time spectrum are all *precongruences* with respect to BCCSP, meaning that $p_1 \lesssim q_1$ and $p_2 \lesssim q_2$ imply $p_1 + p_2 \lesssim q_1 + q_2$ and $ap_1 \lesssim aq_1$ for $a \in A$. Likewise, the equivalences in the spectrum are all *congruences* with respect to BCCSP.

A (closed) substitution, denoted ρ, σ, τ , maps variables in V to (closed) BCCSP terms. For open BCCSP terms t and u , and a preorder \lesssim (or equivalence \simeq) on closed BCCSP terms, we define $t \lesssim u$ (or $t \simeq u$) if $\rho(t) \lesssim \rho(u)$ (resp. $\rho(t) \simeq \rho(u)$) for all closed substitutions ρ .

An *equational axiomatization* is a collection of equations $t \approx u$, and an *inequational axiomatization* is a collection of inequations $t \preccurlyeq u$. The (in)equations in an axiomatization E are referred to as *axioms*. If E is an equational axiomatization, we write $E \vdash t \approx u$ if the equation $t \approx u$ is derivable from the axioms in E using the rules of equational logic (reflexivity, symmetry, transitivity, substitution, and closure under BCCSP contexts):

$$\frac{}{t \approx t} \quad \frac{t \approx u}{u \approx t} \quad \frac{t \approx u \quad u \approx v}{t \approx v} \quad \frac{t \approx u}{\rho(t) \approx \rho(u)} \quad \frac{t \approx u}{at \approx au} \quad \frac{t_1 \approx u_1 \quad t_2 \approx u_2}{t_1 + t_2 \approx u_1 + u_2}$$

For the derivation of an inequation $t \preccurlyeq u$ from an inequational axiomatization E of inequations, denoted $E \vdash t \preccurlyeq u$, the second rule, for symmetry, is omitted.

An axiomatization E is *sound* modulo \lesssim (or \simeq) if for any open BCCSP terms t, u , from $E \vdash t \preccurlyeq u$ (or $E \vdash t \approx u$) it follows that $\rho(t) \lesssim \rho(u)$ (or $\rho(t) \simeq \rho(u)$) for all closed substitutions ρ . E is *ground-complete* modulo \lesssim (or \simeq) if $p \lesssim q$ (or $p \simeq q$) implies $E \vdash p \preccurlyeq q$ (or $E \vdash p \approx q$), for all closed BCCSP terms p and q ; it is *complete* modulo \lesssim (or \simeq) if $p \lesssim q$ (or $p \simeq q$) implies $E \vdash p \preccurlyeq q$ (or $E \vdash p \approx q$) for all BCCSP terms p and q . Finally, E is ω -*complete* if for any open BCCSP terms t and u with $E \vdash \rho(t) \preccurlyeq \rho(u)$ (or $E \vdash \rho(t) \approx \rho(u)$) for all closed substitutions ρ , we have $E \vdash t \preccurlyeq u$ (or $E \vdash t \approx u$). A preorder \lesssim or an equivalence \simeq is said to be *finitely based* if there exists a finite axiomatization E that is sound and complete modulo \lesssim or \simeq .

Let $\{t_1, \dots, t_n\}$ be a finite set of terms; we use summation $\sum\{t_1, \dots, t_n\}$ to denote $t_1 + \dots + t_n$, adopting the convention that the summation of the empty set denotes $\mathbf{0}$. Furthermore, we write $a^n t$ to denote the term obtained from t by prefixing it n times with a , i.e., $a^0 t = t$ and $a^{n+1} t = a(a^n t)$. When writing terms, we adopt as binding convention that $_ + _$ and summation bind weaker than $a_$. With abuse of notation, we often let a finite set X denote the term $\sum_{x \in X} x$.

3 Positive and Negative Results for BCCSP

In this section we will survey positive and negative results on the existence of a finite basis for the equational (resp. inequational) theories of BCCSP modulo the equivalences (resp. preorders) in the spectrum above. The axiomatizations that we will present for the different semantics in the spectrum were mostly taken from [23]. Note that in case of an infinite alphabet, occurrences of action names in axioms are interpreted as variables, as otherwise most of the axiomatizations mentioned in this introduction would be infinite.

As the readers might see, except for bisimulation, the semantics considered in this essay have a natural formulation as a *preorder* relation, while the corresponding *equivalence* is defined as the kernel of the preorder. Recently, Aceto, Fokkink and Ingólfssdóttir [1] gave an algorithm that, given a sound and ground-complete axiomatization for BCCSP modulo a preorder no finer than ready simulation, produces a sound and ground-complete axiomatization for BCCSP modulo the corresponding equivalence. Moreover, if the original axiomatization for the preorder is ω -complete, then so is the resulting axiomatization for the equivalence. So for the positive result regarding a

semantics, the stronger result is obtained by considering the preorder. (The result for the equivalence can be read as a corollary.) On the other hand, the negative results become more general if they are proved for the equivalence relations. We note that, as we will see soon, the condition “no finer than ready simulation” of the algorithm is essential. This also suggests that, for 2-nested simulation, possible futures and impossible futures semantics which fail this condition, the results for preorders and equivalences must be stated separately.

3.1 Bisimulation

The core axioms in the following table are sound and ground-complete for BCCSP modulo bisimulation. Moller [19] proved using normal forms that this axiomatization is ω -complete; Groote provided an alternative proof of this result in [10] using his inverted substitutions technique.

A1	$x + y$	\approx	$y + x$
A2	$(x + y) + z$	\approx	$x + (y + z)$
A3	$x + x$	\approx	x
A6	$x + \mathbf{0}$	\approx	x

Table 1. The axioms for bisimulation.

3.2 2-Nested Simulation

Chen and Fokkink [5] proved that BCCSP modulo any semantics no coarser than impossible futures equivalence and no finer than 2-nested simulation equivalence does not possess a finite sound and ground-complete axiomatization. (Note that possible futures equivalence is within this semantics range.) This improves a result due to Aceto, Fokkink, van Glabbeek and Ingolfsdottir [2] which covers 2-nested simulation and possible future equivalences only. The cornerstone for this negative result is the following infinite family of equations: for $m \geq 0$,

$$aa^{2m}\mathbf{0} + a(a^m\mathbf{0} + a^{2m}\mathbf{0}) \approx a(a^m\mathbf{0} + a^{2m}\mathbf{0}) .$$

These equations are sound modulo 2-nested simulation equivalence. However, any finite axiomatization for BCCSP which is sound modulo impossible futures equivalence cannot derive all of them.

For 2-nested simulation preorder, Aceto, Fokkink, van Glabbeek and Ingolfsdottir [2] proved that it lacks a finite, sound, ground-complete axiomatization as well. The infinite family of inequations that they used to prove this negative result is, for $m \geq 0$,

$$a^{2m} \not\approx a^{2m} + a^m .$$

These inequations are sound modulo 2-nested simulation preorder. However, any finite axiomatization for BCCSP which is sound modulo 2-nested simulation preorder cannot derive all of them.

3.3 Possible Futures

Aceto, Fokkink, van Glabbeek and Ingolfsdottir [2] proved that BCCSP modulo possible futures preorder does not possess a finite, sound, ground-complete axiomatization. The infinite family of inequations that they used to prove this negative result is, for $m \geq 0$,

$$a(a^m + a^{2m}) + aa^{3m} \not\approx a(a^m + a^{3m}) + aa^{2m} .$$

These equations are sound modulo possible futures preorder. However, any finite axiomatization for BCCSP which is sound modulo possible futures preorder cannot derive all of them.

As for possible futures equivalence, the aforementioned negative result (cf. the previous section) implies that BCCSP modulo possible futures equivalence does not possess a finite, sound, ground-complete axiomatization.

3.4 Impossible Futures

Chen and Fokkink [5] provided a sound and ground-complete axiomatization for BCCSP modulo impossible futures *preorder*. This is obtained by extending the four core axioms with two extra axioms:

$$\begin{aligned} a(x + y) &\preceq ax + ay \text{ ,} \\ a(x + y) + ax + a(y + z) &\approx ax + a(y + z) \text{ .} \end{aligned}$$

This result is quite surprising since the aforementioned negative result (cf. the section on 2-nested simulation) implies that BCCSP modulo impossible futures *equivalence* does *not* have a finite, sound, ground-complete axiomatization.

When A is infinite, Groote's technique of inverted substitutions can be applied to show that the aforementioned ground-complete axiomatization is ω -complete. When A is finite, Chen and Fokkink [5] proved that BCCSP modulo impossible futures preorder does not possess a finite sound and ω -complete axiomatization. The infinite family of inequations that they used to prove this negative result is defined in the following way: in case of $|A| = 1$, for $m \geq 0$,

$$a^m x \preceq a^m x + x \text{ ,}$$

while in case of $2 \leq |A| < \infty$,

$$a(a^m x) + a(a^m x + x) + \sum_{b \in A} a(a^m x + a^m b \mathbf{0}) \preceq a(a^m x + x) + \sum_{b \in A} a(a^m x + a^m b \mathbf{0}) \text{ .}$$

To the best of our knowledge, impossible futures semantics is the first (and up to now, the only) example that affords a finite, ground-complete axiomatization for BCCSP modulo the *preorder*, while missing a finite, ground-complete axiomatization for BCCSP modulo the *equivalence*. This fact suggests that, for instance, if one wants to show $p \simeq_{\text{IF}} q$ in general, one has to resort to deriving $p \preceq_{\text{IF}} q$ and $q \preceq_{\text{IF}} p$ separately, instead of proving it directly.

It is worth pointing out that this result does not contradict the algorithm [1] mentioned at the beginning of this section, since that algorithm only applies to semantics that are at least as coarse as *ready simulation semantics*. Since impossible futures semantics is incomparable to ready simulation semantics, it falls outside the scope of [1]. Interestingly, this result yields that no such algorithm exists for certain semantics incomparable with (or finer than) ready simulation.

3.5 Ready Simulation

Van Glabbeek gave a finite axiomatization that is sound and ground-complete for BCCSP modulo ready simulation preorder. It consists of four core axioms and the following axiom:

$$ax \preceq ax + ay \text{ ,}$$

where a ranges over A . For ready simulation equivalence, he only presented a conditional axiom: $\mathcal{I}(x) = \mathcal{I}(y) \Rightarrow a(x + y) \approx a(x + y) + ay$. Blom, Fokkink and Nain [3] showed that a sound and

ground-complete finite equational axiomatization for BCCSP modulo ready simulation equivalence does exist. It can be obtained by extending the four core axioms with

$$a(bx + by + z) \approx a(bx + by + z) + a(bx + z) ,$$

where a, b range over A .

When A is infinite, Groote's technique of inverted substitutions can be applied to show that these axiomatizations are ω -complete. When A is finite, Chen, Fokkink, and Nain [7] proved that BCCSP modulo ready simulation equivalence does not possess a finite sound and ω -complete axiomatization. (Hence neither does ready simulation preorder.) The infinite family of equations that they used to prove this negative result is, for $n > 0$,

$$a^n x + a^n \mathbf{0} + \sum_{b \in A} a^n (x + b\mathbf{0}) \approx a^n \mathbf{0} + \sum_{b \in A} a^n (x + b\mathbf{0}) .$$

These equations are sound modulo ready simulation equivalence. However, they cannot be derived from any finite axiomatization for BCCSP sound which is sound modulo ready simulation equivalence. When $|A| = 1$, ready simulation equivalence (resp. preorder) coincides with completed trace equivalence (resp. preorder), and we will see that in this case a finite basis does exist.

3.6 Completed Simulation

Van Glabbeek gave a finite axiomatization that is sound and ground-complete for BCCSP modulo completed simulation preorder. It consists of four core axioms together with

$$ax \preceq ax + y ,$$

where a range over A . It follows that

$$a(bx + y + z) \approx a(bx + y + z) + a(bx + z)$$

suffices to obtain a finite, sound and ground-complete axiomatization for the completed simulation equivalence.

When $|A| > 1$, Chen, Fokkink and Nain [7] proved that BCCSP modulo completed simulation equivalence does not possess a finite sound and ω -complete axiomatization. (Hence neither does ready simulation preorder.) The infinite family of equations that they used to prove this negative result is, for $n \geq 0$,

$$a^n x + a^n \mathbf{0} + a^n (x + y) \approx a^n \mathbf{0} + a^n (x + y) .$$

These equations are sound modulo completed simulation equivalence. However, they cannot be derived from any finite axiomatization for BCCSP which is sound modulo completed simulation equivalence. When $|A| = 1$, completed simulation equivalence (resp. preorder) coincides with completed trace equivalence (resp. preorder), and we will see that in this case a finite basis does exist.

It is worth mentioning that completed simulation is the only semantics in the linear time – branching time spectrum that in case of an *infinite* alphabet has a finite sound and ground-complete axiomatization for BCCSP, but no finite ω -complete axiomatization.

3.7 Simulation

A sound and ground-complete axiomatization for BCCSP modulo simulation preorder is obtained by extending the four core axioms with

$$x \preceq x + y .$$

It follows that the following axiom

$$a(x + y) \approx a(x + y) + ay \text{ ,}$$

suffices to obtain a sound and ground-complete axiomatization for BCCSP modulo simulation equivalence. When A is infinite, Groote's technique of inverted substitutions can be applied to show that these axiomatizations are ω -complete. When $1 < |A| < \infty$, Chen and Fokkink [4] proved that BCCSP modulo simulation equivalence does not possess a finite sound and ω -complete axiomatization. (Hence neither does simulation preorder.) The infinite family of equations that they used to prove this negative result is, for $n \geq 0$,

$$a(x + \Psi_n) + \sum_{\theta \in A^n} a(x + \Psi_n^\theta) + a\Phi_n \approx \sum_{\theta \in A^n} a(x + \Psi_n^\theta) + a\Phi_n \text{ .}$$

Here the Φ_n are defined inductively as follows:

$$\begin{cases} \Phi_0 &= \mathbf{0} \\ \Phi_{n+1} &= \sum_{b \in A} b\Phi_n \end{cases}$$

Moreover, the Ψ_n and Ψ_n^θ are defined by:

$$\begin{aligned} \Psi_n &= \sum_{b_1 \dots b_n \in A^n} b_1 \dots b_n \mathbf{0} \\ \Psi_n^\theta &= \sum_{b_1 \dots b_n \in A^n \setminus \{\theta\}} b_1 \dots b_n \mathbf{0} \quad \text{for } \theta \in A^n \text{ .} \end{aligned}$$

These equations are sound modulo simulation equivalence. However, any finite axiomatization for BCCSP which is sound modulo simulation equivalence cannot derive all of them. When $|A| = 1$, simulation equivalence (resp. preorder) coincides with trace equivalence (resp. preorder), and we will see that in this case a finite basis does exist.

3.8 Possible Worlds

A sound and ground-complete axiomatization for BCCSP modulo possible worlds preorder is obtained by extending the four core axioms with

$$\begin{aligned} ax &\preceq ax + ay \text{ ,} \\ a(bx + by + z) &\preceq a(bx + z) + a(by + z) \text{ .} \end{aligned}$$

It follows that the following axiom

$$a(bx + by + z) \approx a(bx + z) + a(by + z)$$

suffices to obtain a sound, ground-complete axiomatization for BCCSP modulo possible worlds equivalence.

When A is infinite, Groote's technique of inverted substitutions can be applied to show that these axiomatizations are ω -complete. Fokkink and Nain [8] showed that when $1 < |A| < \infty$, BCCSP modulo any semantics no coarser than ready pair equivalence and no finer than possible worlds equivalence does not possess a finite basis. (Note that ready traces equivalence is within this semantic range.) Their proof of this negative result, which uses cover equations and applies

the compactness theorem to the equational theory for terms of depth 1, is based on the following infinite family of equations:

$$\begin{aligned}
& a\left(\sum_{i=1}^{|A|-1} x_i\right) + \sum_{j=1}^{|A|-1} a\left(\sum_{i=1}^{j-1} x_i + \sum_{i=j+1}^n x_i\right) + \sum_{j=|A|}^n a\left(\sum_{i=1}^{|A|-1} x_i + x_j + y_j\right) \approx \\
& a\left(\sum_{i=1}^{|A|-1} x_i\right) + \sum_{j=1}^{|A|-1} a\left(\sum_{i=1}^{j-1} x_i + \sum_{i=j+1}^n x_i\right) + \sum_{j=|A|}^n a\left(\sum_{i=1}^{|A|-1} x_i + x_j + y_j\right) + a\left(\sum_{i=1}^n x_i\right).
\end{aligned}$$

These equations are sound modulo possible worlds equivalence for $n \geq |A|$. However, any finite axiomatization that is sound for BCCSP modulo ready pairs equivalence cannot derive them all. When $|A| = 1$, possible worlds equivalence (resp. preorder) coincides with completed trace equivalence (resp. preorder), and we will see that in this case a finite basis does exist.

3.9 Ready Traces

Van Glabbeek presented a conditional axiom for ready trace equivalence: $\mathcal{I}(x) = \mathcal{I}(y) \Rightarrow ax + ay \approx a(x + y)$. Blom, Fokkink and Nain [3] showed that when A is finite, a sound and ground-complete finite equational axiomatization for BCCSP modulo ready traces exists. For the ready traces preorder, it can be obtained by extending the four core axioms with

$$\begin{aligned}
& ax \preceq ax + ay \text{ ,} \\
& a\left(\sum_{i=1}^{|A|} (b_i x_i + b_i y_i) + z\right) \preceq a\left(\sum_{i=1}^{|A|} b_i x_i + z\right) + a\left(\sum_{i=1}^{|A|} b_i y_i + z\right) .
\end{aligned}$$

It follows that the following axiom

$$a\left(\sum_{i=1}^{|A|} (b_i x_i + b_i y_i) + z\right) \approx a\left(\sum_{i=1}^{|A|} b_i x_i + z\right) + a\left(\sum_{i=1}^{|A|} b_i y_i + z\right)$$

suffices to obtain a finite, sound and ground-complete axiomatization for BCCSP modulo ready traces equivalence.

When A is infinite, Blom, Fokkink and Nain showed using the compactness theorem that a finite sound and ground-complete axiomatization does not exist. Their proof is based on the following equations, for $n > 0$:

$$a\left(\sum_{i=1}^n (b_i c\mathbf{0} + b_i d\mathbf{0})\right) \approx a\left(\sum_{i=1}^n b_i c\mathbf{0}\right) + a\left(\sum_{i=1}^n b_i d\mathbf{0}\right) .$$

When $1 < |A| < \infty$, the aforementioned negative result from [8] (cf. the section on possible worlds) implies that BCCSP modulo ready traces does not possess a finite basis. When $|A| = 1$, ready trace equivalence (resp. preorder) coincides with completed trace equivalence (resp. preorder), and we will see that in this case a finite ω -complete axiomatization does exist.

3.10 Failure Traces

Van Glabbeek presented a conditional axiom for failure traces (the same one as for ready traces). Blom, Fokkink and Nain [3] showed using normal forms that a sound and ground-complete finite

equational axiomatization for BCCSP modulo failure traces equivalence exists. For failure traces preorder, it is obtained by extending the four core axioms with

$$\begin{aligned} ax &\preceq ax + ay \text{ ,} \\ a(x + y) &\preceq ax + ay \text{ .} \end{aligned}$$

It follows that the following axioms

$$\begin{aligned} a(bx + by + z) &\approx a(bx + by + z) + a(by + z) \text{ ,} \\ ax + ay &\approx ax + ay + a(x + y) \text{ .} \end{aligned}$$

suffice to obtain a finite, sound and ground-complete axiomatization for BCCSP modulo failure traces equivalence.

When A is infinite, Groote's technique of inverted substitutions can be applied to show that these axiomatizations are ω -complete. When $1 < |A| < \infty$, Chen, Fokkink and Luttik [6] showed that BCCSP modulo failure traces equivalence does not possess a finite sound and ω -complete axiomatization. (Hence neither does failure traces preorder.) The infinite family of equations that they used to prove this negative result is, for $n \geq 0$,

$$a^{n+1}x + a(a^n x + x) + a \sum_{b \in A \setminus \{a\}} a^n(b\mathbf{0} + x) \approx a(a^n x + x) + a \sum_{b \in A \setminus \{a\}} a^n(b\mathbf{0} + x) \text{ .}$$

These equations are sound modulo failure traces equivalence. However, any finite axiomatization for BCCSP which is sound modulo failure traces equivalence cannot derive all of them. When $|A| = 1$, failure trace equivalence (resp. preorder) coincides with completed trace equivalence (resp. preorder), and we will see that in this case a finite basis does exist.

3.11 Ready Pairs

A sound and ground-complete axiomatization for BCCSP modulo ready pairs preorder is obtained by extending the four core axioms with

$$\begin{aligned} ax &\preceq ax + ay \text{ ,} \\ a(bx + by + z) &\preceq a(bx + z) + a(by + w) \text{ .} \end{aligned}$$

It follows that the following axiom

$$a(bx + z) + a(by + w) \approx a(bx + by + z) + a(by + w)$$

suffices to obtain a finite sound and ground-complete axiomatization for BCCSP modulo ready pairs equivalence.

When A is infinite, Groote's technique of inverted substitutions can be applied to show that these axiomatizations are ω -complete. When $1 < |A| < \infty$, the aforementioned negative result from [8] (cf. the section on possible worlds) implies that BCCSP modulo ready pairs equivalence does not possess a finite basis. (Hence neither does ready pairs preorder.) When $|A| = 1$, ready pairs equivalence (resp. preorder) coincides with completed trace equivalence (resp. preorder), and we will see that in this case a finite basis does exist.

3.12 Failure Pairs

A sound and ground-complete axiomatization for BCCSP modulo failure pairs preorder is obtained by extending the four core axioms with

$$a(x + y) \preceq ax + a(y + z) .$$

It follows that the following two axioms

$$\begin{aligned} a(bx + by + z) &\approx a(bx + by + z) + a(bx + z) , \\ ax + a(y + z) &\approx ax + a(y + z) + a(x + y) \end{aligned}$$

suffice to obtain a finite, sound and ground-complete axiomatization for BCCSP modulo failure pairs equivalence. Fokkink and Nain [9] proved using cover equations that when A is infinite, these axiomatizations are ω -complete. They also proved that when A is finite, one extra axiom is needed to obtain an ω -complete axiomatization. For failure pairs preorder, it is formulated as

$$\sum_{i=1}^{|A|} a_i x_i \preceq \sum_{i=1}^{|A|} a_i x_i + y ,$$

while for failure pairs equivalence, it is formulated as

$$a\left(\sum_{i=1}^{|A|} b_i x_i + y + z\right) \approx a\left(\sum_{i=1}^{|A|} b_i x_i + y + z\right) + a\left(\sum_{i=1}^{|A|} b_i x_i + y\right) .$$

3.13 Completed Traces

A sound and ground-complete axiomatization for BCCSP modulo completed traces preorder is obtained by extending the four core axioms with

$$\begin{aligned} ax &\preceq ax + y , \\ a(bw + cx + y + z) &\preceq a(bw + y) + a(cx + z) . \end{aligned}$$

It follows that the following axiom

$$a(bw + y) + a(cx + z) \approx a(bw + cx + y + z)$$

suffices to obtain a sound and ground-complete axiomatization for BCCSP modulo completed traces equivalence.

Groote [10] proved using normal forms that in order to obtain an ω -complete axiomatization, one extra axiom is needed. For completed traces preorder, it is formulated as

$$a(x + y) \preceq ax + a(y + z) ,$$

while for complete traces equivalence, it is formulated as

$$ax + a(y + z) \approx ax + a(y + z) + a(x + y) .$$

3.14 Traces

A sound and ground-complete axiomatization for BCCSP modulo traces preorder is obtained by extending the four core axioms with

$$\begin{aligned} x &\preceq x + y , \\ a(x + y) &\preceq ax + ay . \end{aligned}$$

It follows that the following axiom

$$ax + ay \approx a(x + y)$$

suffices to obtain a sound and ground-complete axiomatization for BCCSP modulo traces equivalence.

Groote [10] proved using normal forms that these axiomatizations are ω -complete when $|A| > 1$. When $|A| = 1$, it is not hard to see that one extra axiom suffices to make the axiomatization ω -complete. For traces preorder, it is formulated as

$$x \preceq ax ,$$

while for traces equivalence, it is formulated as

$$ax + x \approx ax .$$

4 Conclusion

The questions whether a semantics in the linear time – branching time spectrum is finitely based over BCCSP have been settled completely. We summarize the results here:

- For most of the semantics in the linear time – branching time spectrum, corresponding preorders and equivalences share the same axiomatizability properties. The only exception is the impossible futures semantics. **Tab. 2** presents an overview regarding this semantics, where + means that the axiomatization exists, – means that there is no such axiomatization.

	ground-comp.	ω -comp.	
	$1 \leq A \leq \infty$	$ A = \infty$	$1 \leq A < \infty$
preorder	+	+	–
equivalence	–	–	–

Table 2. Axiomatizability of impossible futures for BCCSP

- BCCSP has a finite sound and ground-complete axiomatization for most of the semantics in the linear time – branching time spectrum. Only for 2-nested simulation and possible futures, and for ready traces in case of an infinite alphabet, such an axiomatization does not exist. **Tab. 3** presents an overview, where we distinguish between an infinite alphabet and a finite alphabet.
- Regarding ω -completeness, matters are more mixed, especially when $1 < |A| < \infty$. **Tab. 4** presents an overview, where we distinguish among an infinite alphabet, a finite alphabet with more than one element, and a singleton alphabet.

	$ A < \infty$	$ A = \infty$
bisimulation	+	+
2-nested simulation	-	-
possible futures	-	-
ready simulation	+	+
completed simulation	+	+
simulation	+	+
possible worlds	+	+
ready traces	+	-
failure traces	+	+
readies	+	+
failures	+	+
completed traces	+	+
traces	+	+

Table 3. The existence of ground-complete axiomatizations for BCCSP

	$ A = 1$	$1 < A < \infty$	$ A = \infty$
bisimulation	+	+	+
2-nested simulation	-	-	-
possible futures	-	-	-
ready simulation	+	-	+
completed simulation	+	-	-
simulation	+	-	+
possible worlds	+	-	+
ready traces	+	-	-
failure traces	+	-	+
readies	+	-	+
failures	+	+	+
completed traces	+	+	+
traces	+	+	+

Table 4. The existence of ω -complete axiomatizations for BCCSP

Acknowledgement

The work reported in this essay is mostly fruits of joint work with Luca Aceto, Rob van Glabbeek, Anna Ingólfssdóttir, Bas Luttik and Sumit Nain. We would like to thank them for collaborations.

References

1. L. Aceto, W. Fokkink, and A. Ingólfssdóttir. Ready to preorder: Get your BCCSP axiomatization for free! In T. Mossakowski, U. Montanari, and M. Haveraaen, editors, *CALCO*, volume 4624 of *Lecture Notes in Computer Science*, pages 65–79. Springer, 2007.
2. L. Aceto, W. Fokkink, R. van Glabbeek, and A. Ingólfssdóttir. Nested semantics over finite trees are equationally hard. 191(2):203–232, June 2004.
3. S. Blom, W. Fokkink, and S. Nain. On the axiomatizability of ready traces, ready simulation, and failure traces. In J. C. M. Baeten, J. K. Lenstra, J. Parrow, and G. J. Woeginger, editors, *ICALP*, volume 2719 of *Lecture Notes in Computer Science*, pages 109–118. Springer, 2003.
4. T. Chen and W. Fokkink. On finite alphabets and infinite bases III: Simulation. In C. Baier and H. Hermanns, editors, *CONCUR*, volume 4137 of *Lecture Notes in Computer Science*, pages 421–434. Springer, 2006.

5. T. Chen and W. Fokkink. On the axiomatizability of impossible futures: Preorder versus equivalence. In *LICS*, pages 156–165. IEEE Computer Society, 2008.
6. T. Chen, W. Fokkink, B. Luttik, and S. Nain. On finite alphabets and infinite bases. *Inf. Comput.*, 206(5):492–519, 2008.
7. T. Chen, W. Fokkink, and S. Nain. On finite alphabets and infinite bases II: Completed and ready simulation. In L. Aceto and A. Ingólfssdóttir, editors, *FoSSaCS*, volume 3921 of *Lecture Notes in Computer Science*, pages 1–15. Springer, 2006.
8. W. Fokkink and S. Nain. On finite alphabets and infinite bases: From ready pairs to possible worlds. In I. Walukiewicz, editor, *FoSSaCS*, volume 2987 of *Lecture Notes in Computer Science*, pages 182–194. Springer, 2004.
9. W. Fokkink and S. Nain. A finite basis for failure semantics. In L. Caires, G. F. Italiano, L. Monteiro, C. Palamidessi, and M. Yung, editors, *ICALP*, volume 3580 of *Lecture Notes in Computer Science*, pages 755–765. Springer, 2005.
10. J. F. Groote. A new strategy for proving ω -completeness applied to process algebra. In J. C. M. Baeten and J. W. Klop, editors, *CONCUR*, volume 458 of *Lecture Notes in Computer Science*, pages 314–331. Springer, 1990.
11. R. Gurevic. Equational theory of positive numbers with exponentiation is not finitely axiomatizable. *Ann. Pure Appl. Logic*, 49(1):1–30, 1990.
12. J. Heering. Partial evaluation and ω -completeness of algebraic specifications. *Theor. Comput. Sci.*, 43:149–167, 1986.
13. L. Henkin. The logic of equality. *American Mathematical Monthly*, 84(8):597–612, 1977.
14. A. Lazrek, P. Lescanne, and J.-J. Thiel. Tools for proving inductive equalities, relative completeness, and ω -completeness. *Inf. Comput.*, 84(1):47–70, 1990.
15. H. Lin. PAM: A process algebra manipulator. *Formal Methods in System Design*, 7(3):243–259, 1995.
16. R. Lyndon. Identities in two-valued calculi. *Transactions of the American Mathematical Society*, 71:457–465, 1951.
17. R. McKenzie. Tarski’s finite basis problem is undecidable. *Journal of Algebra and Computation*, 6(1):49–104, 1996.
18. R. McKenzie, G. McNulty, and W. Taylor. *Algebras, Varieties, Lattices*. Wadsworth & Brooks/Cole, 1987.
19. F. Moller. *Axioms for Concurrency*. PhD thesis, Department of Computer Science, University of Edinburgh, July 1989. Report CST-59-89. Also published as ECS-LFCS-89-84.
20. V. Murskiĭ. The existence in the three-valued logic of a closed class with a finite basis having no finite complete system of identities. *Doklady Akademii Nauk SSSR*, 163:815–818, 1965. In Russian.
21. V. Murskiĭ. The existence of a finite basis of identities, and other properties of “almost all” finite algebras. *Problemy Kibernetiki*, 30:43–56, 1975. In Russian.
22. R. J. van Glabbeek. The linear time-branching time spectrum (extended abstract). In J. C. M. Baeten and J. W. Klop, editors, *CONCUR*, volume 458 of *Lecture Notes in Computer Science*, pages 278–297. Springer, 1990.
23. R. J. van Glabbeek. The linear time-branching time spectrum I. In J. Bergstra, A. Ponse, and S. Smolka, editors, *Handbook of Process Algebra*, pages 3–99. Elsevier Science, 2001.