

# Proving Non-finite Axiomatizability Results in Process Algebra via Reductions

Luca Aceto  
ICE-TCS and Reykjavik University

18 January 2008

Joint on-going work with Wan Fokkink (Vrije Universiteit Amsterdam, NL), Anna Ingolfsdottir (Reykjavik University) and MohammadReza Mousavi (Eindhoven University of Technology, NL).

# General Intellectual Framework for this Work

## My Tenet

(Theoretical) Computer Science and its forefather Mathematical Logic are pretty much unique in their development of mathematical methodology for proving negative results.

## Christos Papadimitriou's Viewpoint

Negative results are **the only possible** self-contained theoretical results in Computer Science.

Successful exploratory theoretical research is bound to produce predominantly negative results. (From "Database metatheory: Asking the big queries")

How are negative results established?

# A General-Purpose Tool: Reductions!

## A “Proof Strategy”

- “Reduce” a problem known to be “hard” to your purportedly “difficult” problem.
- A very common proof technique in computability theory and complexity theory, among others.

## Two Classic Examples:

- To prove undecidability: Show that your problem is (at least) as difficult as the halting problem for Turing machines or any other known undecidable problem.
- To prove  $\mathcal{CC}$ -hardness: Show that your problem is (at least) as difficult as a problem known to be hard for your favourite complexity class  $\mathcal{CC}$ .

# A General-Purpose Tool: Reductions!

## A “Proof Strategy”

- “Reduce” a problem known to be “hard” to your purportedly “difficult” problem.
- A very common proof technique in computability theory and complexity theory, among others.

## Two Classic Examples:

- To prove undecidability: Show that your problem is (at least) as difficult as the halting problem for Turing machines or any other known undecidable problem.
- To prove  $\mathcal{CC}$ -hardness: Show that your problem is (at least) as difficult as a problem known to be hard for your favourite complexity class  $\mathcal{CC}$ .

# The Role of Equalities Between Programs

**Motto:** In Computer Science, we use formal languages to communicate with machines and describe expected properties of computations.

**Fact of Life:** We often need to know when two syntactically different descriptions are describing the “same thing”. Examples?

- Optimization in compilers.
- Program analysis/partial evaluation.
- Correctness: Is **SPEC**ification equivalent to **IMP**lementation?

**Tenet:** Equational logic can be used to capture “valid” equivalences.

# Finite, Complete Axiomatizations

## The Challenge (Phrased in Terms of Precongruences)

Given some algebraic **signature**  $\Sigma$ , and some **precongruence**  $\lesssim$  over (closed) terms

*Is there a **finite** set  $\mathcal{E}$  of  $\Sigma$ -inequations  $s \leq t$  such that*

$$t \lesssim u \Leftrightarrow \mathcal{E} \vdash t \leq u$$

*for all (**closed**)  $\Sigma$ -terms  $t, u$ ?*

$\mathcal{E}$  is called a **sound** and (**ground-**)**complete** axiomatization.

# Why is This an Interesting Game?

## Answer 1

The axiomatic method is a very powerful method of scientific analysis, so studying its power in Computer Science must be interesting!... **And I like it!**

## Answer 2

An equational axiomatization

- 1 tells ye all ye need to know about your notion of program equivalence;
- 2 allows you to relate it to other types of program equivalence by simply looking at laws;
- 3 may form the basis for program verification tools based on theorem proving technology.

# Why is This an Interesting Game?

## Answer 1

The axiomatic method is a very powerful method of scientific analysis, so studying its power in Computer Science must be interesting!... **And I like it!**

## Answer 2

An equational axiomatization

- 1 tells ye all ye need to know about your notion of program equivalence;
- 2 allows you to relate it to other types of program equivalence by simply looking at laws;
- 3 may form the basis for program verification tools based on theorem proving technology.

# The Cold Shower

## Main General Technical Message of the Talk

The life of a concurrency theorist is equationally hard.

- In many situations, the collection of valid (in)equivalences **cannot** be “captured” by means of a finite collection of (in)equations. This holds true even for **very simple** languages!
- **Reductions, suitably defined, can be used to prove some of these results!**

And now for a little technical content!

# Outline

- 2 Preliminaries
- 3 Outline of the Method
- 4 Some Applications
- 5 A Negative Result About a Negative Meta-Theorem
- 6 Conclusions

# A Cute Subset of CCS

## Syntax: Grammar for Closed Terms

$$P ::= 0 \mid a.P \mid P + P \mid P \parallel P$$

Open terms may contain occurrences of variables  $x, y, z \dots$

## Operational Semantics

$$\frac{}{a.x \xrightarrow{a} x} \quad \frac{x_0 \xrightarrow{a} y_0}{x_0 + x_1 \xrightarrow{a} y_0} \quad \frac{x_1 \xrightarrow{a} y_1}{x_0 + x_1 \xrightarrow{a} y_1}$$

$$\frac{x_0 \xrightarrow{a} y_0}{x_0 \parallel x_1 \xrightarrow{a} y_0 \parallel x_1} \quad \frac{x_1 \xrightarrow{a} y_1}{x_0 \parallel x_1 \xrightarrow{a} x_0 \parallel y_1}$$

# CCS: Behavioural Semantics

## Bisimulation and Bisimilarity

- A bisimulation relation  $R$  is a symmetric relation such that:

*if  $p R q$  then, for each  $p'$ ,  $p \xrightarrow{a} p'$  implies  $q \xrightarrow{a} q'$  for some  $q'$  such that  $p' R q'$ .*

- $p \underline{\leftrightarrow} q$  when there exists a bisimulation  $R$  such that  $p R q$ ;
- $s \underline{\leftrightarrow} t$  when  $\sigma(s) \underline{\leftrightarrow} \sigma(t)$ , for each substitution  $\sigma$  mapping variables to closed terms.
- Theorem:  $\underline{\leftrightarrow}$  is a **congruence** over CCS.

Question: Does  $\underline{\leftrightarrow}$  afford a finite equational axiomatization over CCS?

# Axiomatizations for CCS

## CCS without Parallel: Yes!

$$A0 \quad x + y = y + x$$

$$A1 \quad (x + y) + z = x + (y + z)$$

$$A2 \quad x + x = x$$

$$A3 \quad 0 + x = x$$

## Parallel?

$$a \parallel a \iff a.a$$

$$a \parallel (a + a^2) \iff a.(a + a^2) + a^2 + a^3$$

$$a \parallel (a + a^2 + a^3) \iff a.(a + a^2 + a^3) + a^2 + a^3 + a^4$$

$$\vdots$$

# Axiomatizations for CCS

## CCS without Parallel: Yes!

$$A0 \quad x + y = y + x$$

$$A1 \quad (x + y) + z = x + (y + z)$$

$$A2 \quad x + x = x$$

$$A3 \quad 0 + x = x$$

## Parallel?

$$a \parallel a \iff a.a$$

$$a \parallel (a + a^2) \iff a.(a + a^2) + a^2 + a^3$$

$$a \parallel (a + a^2 + a^3) \iff a.(a + a^2 + a^3) + a^2 + a^3 + a^4$$

$$\vdots$$

# Moller's Cold Shower

## CCS with Parallel

Faron Moller [LICS 90 and Ph.D. Thesis]: **CCS** modulo **bisimilarity** **does not afford** a finite axiomatization.

## Moller's Proof (Put Informally)

Let  $E$  be a **finite and sound** axiom system for CCS. Then, for “all sufficiently large”  $m$ , using  $E$  one cannot prove the valid equality

$$a \parallel \sum_{i=1}^m a^i \quad = \quad a.(\sum_{i=1}^m a^i) + \sum_{i=2}^{m+1} a^i$$

This will be the “mother yeast” for **all** the applications of our reduction method to follow.

# Moller's Cold Shower

## CCS with Parallel

Faron Moller [LICS 90 and Ph.D. Thesis]: **CCS** modulo **bisimilarity** **does not afford** a finite axiomatization.

## Moller's Proof (Put Informally)

Let  $E$  be a **finite and sound** axiom system for CCS. Then, for “all sufficiently large”  $m$ , using  $E$  one cannot prove the valid equality

$$a \parallel \sum_{i=1}^m a^i = a.(\sum_{i=1}^m a^i) + \sum_{i=2}^{m+1} a^i$$

This will be the “mother yeast” for **all** the applications of our reduction method to follow.



# Proving Other Non-finite Axiomatizability Results

## Beyond CCS and Bisimilarity

Most other non-finite axiomatizability proofs are as **difficult** and **delicate** as Moller's. These proofs are highly **language dependent**.

## Goal: Some Method in the Madness

- 1 Take the **calculus** of your choice (with signature  $\Sigma_e$ );
- 2 Take the behavioral **preorder** of your choice  $\lesssim_e$ ;
- 3 Give a **well-behaved reduction** from  $\mathcal{T}(\Sigma_e)$  to **CCS** modulo bisimilarity (or any other non-finitely axiomatizable calculus).
- 4 AFIM meta-theorem **guarantees non-finite axiomatizability of  $\lesssim_e$  over  $\mathcal{T}(\Sigma_e)$ !**

# Reductions Defined

A mapping  $\hat{\_} : \mathcal{T}(\Sigma_e) \rightarrow \mathcal{T}(\Sigma_o)$  is a **reduction** from  $\Sigma_e$  to  $\Sigma_o$  when:

- ①  $t \lesssim_e u \Rightarrow \hat{t} \lesssim_o \hat{u}$ , for all  $t, u \in \mathcal{T}(\Sigma_e)$ , and
- ②  $E \vdash t \leq u \Rightarrow \hat{E} \vdash \hat{t} \leq \hat{u}$ , for each axiom system  $E$ .

**Notation:**  $\hat{E} \doteq \{\hat{t} \leq \hat{u} \mid t \leq u \in E\}$ .

## Question

Can we avoid checking condition 2 above?

# Structural Mapping

$\widehat{\cdot} : \mathcal{T}(\Sigma_e) \rightarrow \mathcal{T}(\Sigma_o)$  is **structural**, when  $\widehat{\sigma}(t) \equiv \widehat{\sigma}(\widehat{t})$ , where

$$\widehat{\sigma}(x) = \widehat{\sigma}(x) \text{ for each variable } x .$$

**Remark:** Actually, we give some syntactic conditions implying the above property.

## Lemma

If  $\widehat{\cdot}$  is structural, then

$$E \vdash t \leq u \Rightarrow \widehat{E} \vdash \widehat{t} \leq \widehat{u} ,$$

for each axiom system  $E$ .

# The Last Ingredient: $E$ -reflecting Reductions

Given an axiom system  $E$  on  $\mathcal{T}(\Sigma_o)$ , a **reduction**  $\hat{\cdot}$  is  **$E$ -reflecting**, when for each  $t \leq u \in E$ , there exists an inequation  $t' \leq u'$  over  $\mathcal{T}(\Sigma_e)$  that is sound w.r.t.  $\lesssim_e$  such that  $\hat{t}' \equiv t$  and  $\hat{u}' \equiv u$ .

## Main Theorem

Assume that there are

- a sound set of axioms  $E$  w.r.t.  $\lesssim_o$  that is **not provable** from any **finite, sound axiom system over  $\mathcal{T}(\Sigma_o)$** , and
- an  **$E$ -reflecting reduction** from  $\Sigma_e$  to  $\Sigma_o$ .

Then  $\lesssim_e$  affords **no finite axiomatization** over  $\mathcal{T}(\Sigma_e)$ .

**Crunch Question:** Can this be used?

# Basic Theory (Recap)

## Syntax

$$P ::= 0 \mid a.P \mid P + P \mid P \parallel P$$

## Moller's Equations

One cannot prove the following set of axioms, which are sound in CCS/  $\leftrightarrow$  :

$$\mathcal{M} = \{a \parallel \sum_{i=1}^m a^i = a.(\sum_{i=1}^m a^i) + \sum_{i=2}^{m+1} a^i \mid m \geq 1\}$$

# TACS<sup>UT</sup>: Syntax

## Syntax ( $\Sigma_e$ )

$$P ::= 0 \mid \underline{\mu}.P \mid \underline{\epsilon(1)}.P \mid P + P \mid P \parallel P$$

where  $\mu$  is an action taken from a set  $A \cup \bar{A} \cup \{\tau\}$ .

## Urgent Initial Actions

$\mathcal{U}(p)$  is the set of *urgent initial actions* of  $p$ . For example,

$$\mathcal{U}(\underline{\mu}.p) = \{\mu\} \quad \mathcal{U}(\underline{\epsilon(1)}.p) = \emptyset$$

$$\mathcal{U}(p \parallel q) = \mathcal{U}(p) \cup \mathcal{U}(q) \cup \{\tau \mid \mathcal{U}(p) \cap \overline{\mathcal{U}(q)} \neq \emptyset\}$$

# TACS<sup>UT</sup>: Operational Semantics

## Operational Semantics: Sample Rules

$$\frac{}{0 \xrightarrow{\epsilon(1)} 0}$$

$$\frac{}{\underline{\epsilon(1)}.x \xrightarrow{\epsilon(1)} x}$$

$$\frac{x \xrightarrow{\mu} y}{\underline{\epsilon(1)}.x \xrightarrow{\mu} y}$$

$$\frac{x_0 \xrightarrow{\epsilon(1)} y_0 \quad x_1 \xrightarrow{\epsilon(1)} y_1}{x_0 \parallel x_1 \xrightarrow{\epsilon(1)} y_0 \parallel y_1} \quad \tau \notin \mathcal{U}(x_0 \parallel x_1)$$

$$\frac{}{\underline{\mu}.x \xrightarrow{\mu} x}$$

$$\frac{}{\underline{\alpha}.x \xrightarrow{\epsilon(1)} \underline{\alpha}.x} \quad \alpha \neq \tau$$

$$\frac{x_0 \xrightarrow{\epsilon(1)} y_0 \quad x_1 \xrightarrow{\epsilon(1)} y_1}{x_0 + x_1 \xrightarrow{\epsilon(1)} y_0 + y_1}$$

# The Faster-Than Preorder

## Faster-Than

The **faster-than preorder** is the largest relation  $\sqsupseteq$  satisfying the following transfer properties for all  $p, q$  such that  $p \sqsupseteq q$ :

- 1  $\forall_{p'} p \xrightarrow{a} p' \Rightarrow \exists_{q'} q \xrightarrow{a} q' \wedge p' \sqsupseteq q'$ ,
- 2  $\forall_{q'} q \xrightarrow{a} q' \Rightarrow \exists_{p'} p \xrightarrow{a} p' \wedge p' \sqsupseteq q'$  and
- 3  $\forall_{p'} p \xrightarrow{\epsilon(1)} p' \Rightarrow \mathcal{U}(q) \subseteq \mathcal{U}(p) \wedge \exists_{q'} (q \xrightarrow{\epsilon(1)} q' \wedge p' \sqsupseteq q')$ .

**Examples:**  $\underline{\epsilon(1)}.a.0 \not\sqsupseteq a.0$ , but  $a.0 \sqsupseteq \underline{\epsilon(1)}.a.0$ . Intuitively,  $p \sqsupseteq q$  means that  $\underline{p}$  is “at least as fast as”  $\underline{q}$ .

**Theorem:**  $\sqsupseteq$  is **not** finitely axiomatizable over  $TACS^{UT}$ .

# The Faster-Than Preorder

## Faster-Than

The **faster-than preorder** is the largest relation  $\sqsupseteq$  satisfying the following transfer properties for all  $p, q$  such that  $p \sqsupseteq q$ :

- 1  $\forall_{p'} p \xrightarrow{a} p' \Rightarrow \exists_{q'} q \xrightarrow{a} q' \wedge p' \sqsupseteq q'$ ,
- 2  $\forall_{q'} q \xrightarrow{a} q' \Rightarrow \exists_{p'} p \xrightarrow{a} p' \wedge p' \sqsupseteq q'$  and
- 3  $\forall_{p'} p \xrightarrow{\epsilon(1)} p' \Rightarrow \mathcal{U}(q) \subseteq \mathcal{U}(p) \wedge \exists_{q'} (q \xrightarrow{\epsilon(1)} q' \wedge p' \sqsupseteq q')$ .

**Examples:**  $\underline{\epsilon(1)}.a.0 \not\sqsupseteq a.0$ , but  $a.0 \sqsupseteq \underline{\epsilon(1)}.a.0$ . Intuitively,  $p \sqsupseteq q$  means that  $\underline{p}$  is “at least as fast as”  $\underline{q}$ .

**Theorem:**  $\sqsupseteq$  is **not** finitely axiomatizable over  $TACS^{UT}$ .

# Proof Using Reduction from $TACS^{UT}$ to CCS

## The Reduction from $TACS^{UT}$ to CCS

$$\widehat{0} = 0$$

$$\widehat{x} = x$$

$$\widehat{a.t} = a.\widehat{t}$$

$$\widehat{\mu.t} = 0 \text{ for } \mu \neq a$$

$$\widehat{\epsilon(1).t} = \widehat{t}$$

$$\widehat{t+u} = \widehat{t} + \widehat{u}$$

$$\widehat{t \parallel u} = \widehat{t} \parallel \widehat{u}$$

# Meta-Theorem: Applied to $TACS^{UT}$

## Reductions at Work

We observe that:

- 1  $t \sqsubseteq u \Rightarrow \hat{t} \leftrightarrow \hat{u}$ ,
- 2  $\hat{\_}$  is structural,
- 3 all Moller's axioms in  $\mathcal{M}$  are sound with respect to  $\sqsubseteq$  (after underlining all  $a$ 's) and
- 4  $\hat{\_}$  maps the underlined version of a CCS term  $p$  to  $p$ .

Thus  $TACS^{UT} / \sqsubseteq$  affords no finite axiomatization.

Was this a lucky break?

# Meta-Theorem: Applied to $TACS^{UT}$

## Reductions at Work

We observe that:

- 1  $t \sqsubseteq u \Rightarrow \hat{t} \leftrightarrow \hat{u}$ ,
- 2  $\hat{\_}$  is structural,
- 3 all Moller's axioms in  $\mathcal{M}$  are sound with respect to  $\sqsubseteq$  (after underlining all  $a$ 's) and
- 4  $\hat{\_}$  maps the underlined version of a CCS term  $p$  to  $p$ .

Thus  $TACS^{UT} / \sqsubseteq$  affords no finite axiomatization.

Was this a lucky break?

## Other Examples

No lucky break. We have similar results for the following algebras.

- 1 Discrete-Timed CCS modulo Timed Bisimulation
- 2 Temporal Calculus of Communicating Systems modulo Timed Bisimulation
- 3  $TACS^{LT}$  modulo MT-preorder
- 4 TACS modulo Urgent Bisimulation
- 5 IMC modulo Markovian Bisimulation

**First (Admittedly Biased) Conclusion:** The reduction method seems to be widely applicable.

**Question:** But how applicable are reductions to CCS?

## Free Lunch? (Starter)

CCS<sub>Ω</sub>: Syntax

$$P ::= 0 \mid \Omega \mid a.P \mid P + P \mid P \parallel P$$

CCS<sub>Ω</sub>: Semantics (Transitions + Convergence Predicate)

Transition semantics: just like CCS (thus, no transition for  $\Omega$ ).

$$\frac{}{0 \downarrow} \quad \frac{}{a.p \downarrow} \quad \frac{p \downarrow \quad q \downarrow}{p + q \downarrow} \quad \frac{p \downarrow \quad q \downarrow}{p \parallel q \downarrow}$$

N.B.  $p \downarrow$ , for each CCS process  $p$ .

## Free Lunch? (Main Course)

## Prebisimilarity

The relation  $\Xi_{pre}$  is the largest relation satisfying the following clauses whenever  $p \approx_{pre} q$ :

- ① if  $p \xrightarrow{a} p'$  then  $q \xrightarrow{a} q'$  for some  $q'$  such that  $p' \Xi_{pre} q'$ ;
- ② if  $p \downarrow$ , then
  - ①  $q \downarrow$  and
  - ② if  $q \xrightarrow{a} q'$  then  $p \xrightarrow{a} p'$  for some  $p'$  such that  $p' \Xi_{pre} q'$ .

N.B.  $\Omega \Xi_{pre} p$ , for each  $p$ .

**Theorem (AFIM):**  $\text{CCS}_\Omega$  modulo  $\approx_{pre}$  has no finite axiomatization.

# Free Lunch? (Gammel Dansk Bitter Dram)

## No Free Lunch (at least with Moller)!

- Suppose  $\hat{-}$  is a **reduction** from  $\text{CCS}_\Omega$  to  $\text{CCS}$ ;
- $\hat{\Omega} \leftrightarrow \hat{p}$  for each  $p$  because  $\Omega \stackrel{E}{\sim}_{pre} p$ ;
- Thus,  $\hat{-}$  is a **constant** function modulo  $\leftrightarrow$ ;
- For  $m \neq n$  it does not hold that  $a \parallel \sum_{i=1}^m a^i \leftrightarrow a \parallel \sum_{i=1}^n a^i$ ;
- Hence,  $\hat{-}$  cannot be  **$\mathcal{M}$ -reflecting**.

**Remark:** In fact,  $\hat{-}$  cannot reflect any “non-trivial” set of equations.

# Conclusions

## Done

- 1 A **reduction** method for proving non-finite axiomatizability with general **algebraic conditions**;
- 2 applied to **many examples** from the literature (leading to novel results);
- 3 investigated some of the **limitations** of the method.

## To Be Done

- 1 Finding **language-based concrete sufficient criteria** for the applicability of the method. **SOS meta-theory** is a promising candidate: e.g., the link with **conservative** and orthogonal extensions.
- 2 Removing the **limitation** by taking “**partial reductions**”.

# That's All Folks!

Thank you! Any questions?