

A grayscale photograph of a large, multi-story brick building with many windows, likely a university building. The building has a modern glass-enclosed section on the right side. The text is overlaid on the image.

Formal Methods for Reversible Concurrent Calculi: an Introduction

Gabriele Cecilia

Research Colloquium presentation

Augusta University, 28 August 2025

A brief personal introduction



Gabriele Cecilia

A brief personal introduction



Gabriele Cecilia



Italy

A brief personal introduction

- B.S. in Mathematics at University of Milan, Italy, 2019

A brief personal introduction

- B.S. in Mathematics at University of Milan, Italy, 2019
- M.S. in Mathematics at University of Milan, Italy, 2024

Thesis title: *Formalizing the Operational Semantics of the π -calculus: a Solution to the Concurrent Calculi Formalization Benchmark (part 2)*

Thesis advisor: Dr. Alberto Momigiano

A brief personal introduction

- B.S. in Mathematics at University of Milan, Italy, 2019
- M.S. in Mathematics at University of Milan, Italy, 2024

Thesis title: *Formalizing the Operational Semantics of the π -calculus: a Solution to the Concurrent Calculi Formalization Benchmark (part 2)*

Thesis advisor: Dr. Alberto Momigiano

→ *A Beluga Formalization of the Harmony Lemma in the π -Calculus*, LFMTTP '24.

A brief personal introduction

- B.S. in Mathematics at University of Milan, Italy, 2019
- M.S. in Mathematics at University of Milan, Italy, 2024

Thesis title: *Formalizing the Operational Semantics of the π -calculus: a Solution to the Concurrent Calculi Formalization Benchmark (part 2)*

Thesis advisor: Dr. Alberto Momigiano

→ *A Beluga Formalization of the Harmony Lemma in the π -Calculus*, LFMTTP '24.

- Ph.D. student in Computer and Cyber Sciences at Augusta University

Table of Contents

- ▶ Background notions
- ▶ Motivation and applications
- ▶ State of the art and open problems
- ▶ Current and future work

My research project



Clément Aubert

My research project



Clément Aubert



The screenshot shows the NSF website with the following elements:

- Header:** NSF U.S. National Science Foundation
- Navigation:** Find Funding, How to Apply, Manage Your Award, Focus Areas
- Awards Section:**
 - Awards** (with a graphic of a network)
 - [Search Awards](#)
 - [Recent Awards](#)
 - [Presidential and Honorary Awards](#)
- Award Abstract # 2242786**
 - SHF:Small:Concurrency In Reversible Computations**

NSF Org:	CCF Division of Computing and Communicati
Recipient:	AUGUSTA UNIVERSITY RESEARCH INSTITUTE
Initial Amendment Date:	February 15, 2023
Latest Amendment Date:	February 15, 2023

My research project



Clément Aubert



U.S. National Science Foundation

[Find Funding](#) ▾ [How to Apply](#) ▾ [Manage Your Award](#) ▾ [Focus Areas](#) ▾

Awards

[Search Awards](#)

[Recent Awards](#)

[Presidential and Honorary Awards](#)

Award Abstract # 2242786

SHF:Small:Concurrency In Reversible Computations

NSF Org:	CCF Division of Computing and Communicati
Recipient:	AUGUSTA UNIVERSITY RESEARCH INSTITUTE
Initial Amendment Date:	February 15, 2023
Latest Amendment Date:	February 15, 2023

Formal Methods for Reversible Concurrent Calculi

Concurrent Calculi

Concurrency: simultaneous execution of multiple operations in the same environment

Toy example: two clients and two servers

Concurrent Calculi

Concurrency: simultaneous execution of multiple operations in the same environment

Toy example: two clients and two servers

Concurrent Calculi: abstract models for concurrent systems
Processes, channels, transition or reduction rules

Concurrent Calculi

Concurrency: simultaneous execution of multiple operations in the same environment

Toy example: two clients and two servers

Concurrent Calculi: abstract models for concurrent systems

Processes, channels, transition or reduction rules

$\text{Client}_1 \stackrel{\text{def}}{=} \overline{\text{req}}.\text{resp}.\text{Client}_1$

Concurrent Calculi

Concurrency: simultaneous execution of multiple operations in the same environment

Toy example: two clients and two servers

Concurrent Calculi: abstract models for concurrent systems

Processes, channels, transition or reduction rules

$$\text{Client}_1 \stackrel{\text{def}}{=} \overline{\text{req}}.\text{resp}.\text{Client}_1$$
$$\text{Server}_1 \stackrel{\text{def}}{=} \text{req}.\overline{\text{resp}}.\text{Server}_1$$

Concurrent Calculi

Concurrency: simultaneous execution of multiple operations in the same environment

Toy example: two clients and two servers

Concurrent Calculi: abstract models for concurrent systems

Processes, channels, transition or reduction rules

$$\text{Client}_1 \stackrel{\text{def}}{=} \overline{\text{req}}.\text{resp}.\text{Client}_1$$
$$\text{Server}_1 \stackrel{\text{def}}{=} \text{req}.\overline{\text{resp}}.\text{Server}_1$$
$$\text{Client}_1 \mid \text{Client}_2 \mid \text{Server}_1 \mid \text{Server}_2$$

Reversible Concurrent Calculi

Reversible Concurrent Calculi:

abstract models for concurrent systems in which **any action can be undone**

Reversible Concurrent Calculi

Reversible Concurrent Calculi:

abstract models for concurrent systems in which **any action can be undone**

Example: $a \mid b[k] \xrightarrow{a[m]} a[m] \mid b[k] \xrightarrow{\text{wavy } b[k]} a[m] \mid b$

Reversible Concurrent Calculi

Reversible Concurrent Calculi:

abstract models for concurrent systems in which **any action can be undone**

Example: $a \mid b[k] \xrightarrow{a[m]} a[m] \mid b[k] \xrightarrow{\text{wavy } b[k]} a[m] \mid b$

What does this mean?

Reversibility - Examples



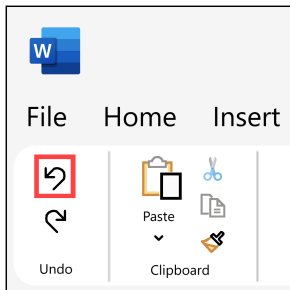
Reversibility - Examples



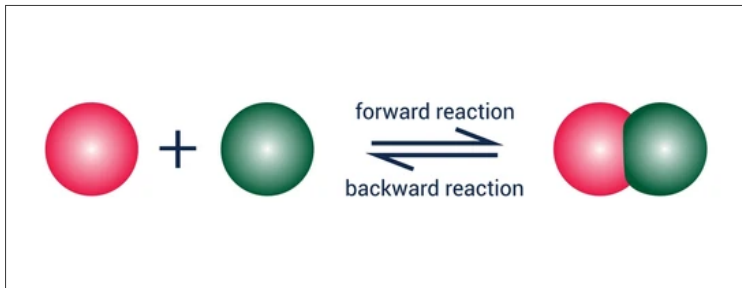
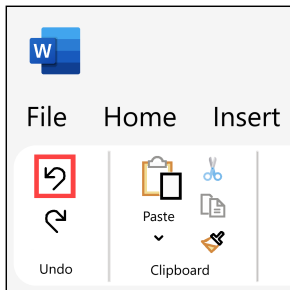
Reversibility - Examples



Examples of Reversibility



Examples of Reversibility



Formal methods

How one line of code caused a \$60 million loss

60,000 people lost full phone service, half of AT&T's network was down, and 500 airline flights were delayed

NOV 13, 2023

On January 15th, 1990, AT&T's New Jersey operations center detected a widespread system malfunction, shown by a plethora of red warnings on their network display.

Despite attempts to rectify the situation, the network remained compromised for 9 hours, leading to a 50% failure rate in call connections.

AT&T lost over **\$60 million** as a result with over **60,000** of Americans left with fully disconnected phones.



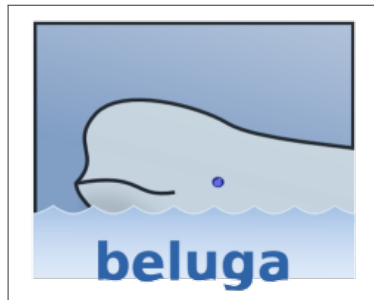
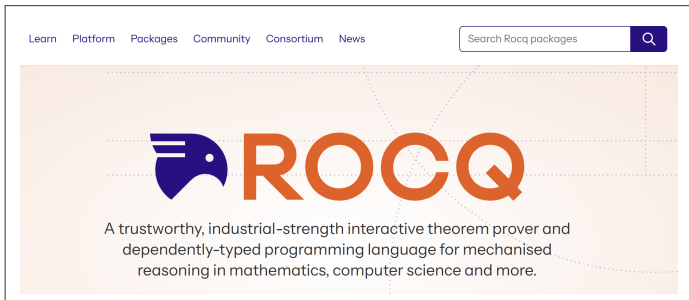
How a single line of code brought down a half-billion euro rocket launch

It's Tuesday, June 4th, 1996, and the European Space Agency is set to launch its new Ariane 5 rocket for the first time. This is the culmination of a decade of design, testing and a budget spending billions of euros.

Formal methods: mathematics to verify that a program meets its specifications

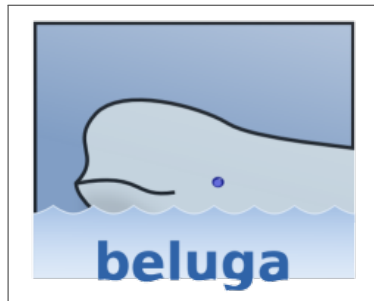
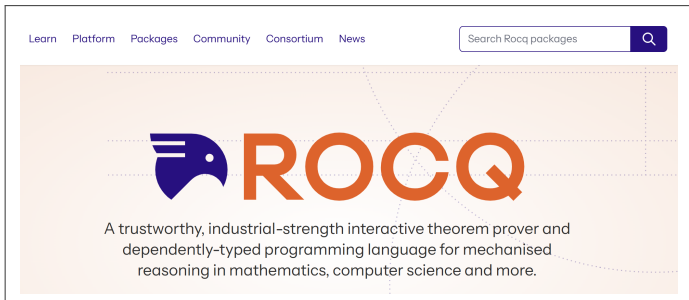
Proof assistants and formalization

Application: specify (programming) languages and study their properties



Proof assistants and formalization

Application: specify (programming) languages and study their properties



Formalization of a language

Putting the pieces together...

Formal Methods for Reversible Concurrent Calculi:

Studying and improving reversible concurrent calculi, e.g. by formalizing their properties with proof assistants

Table of Contents

► Background notions

► Motivation and applications

► State of the art and open problems

► Current and future work

Why studying reversible concurrent calculi?

- Theoretical foundations for the development of reversible computers

Why studying reversible concurrent calculi?

- Theoretical foundations for the development of reversible computers
- Faithful and efficient representation of concrete systems

Why studying reversible concurrent calculi?

- Theoretical foundations for the development of reversible computers
- Faithful and efficient representation of concrete systems
- Reversibility is the “right” environment to treat properties like causality

Why formalizing them?

- Deeper understanding of reversible concurrent calculi

Why formalizing them?

- Deeper understanding of reversible concurrent calculi
- Verification of the correctness of definitions and proofs

Why formalizing them?

- Deeper understanding of reversible concurrent calculi
- Verification of the correctness of definitions and proofs
- (It is fun)

What are the applications of my research?

- Concurrent calculi: uncover privacy flaws in e-passports

Breaking Unlinkability of the ICAO 9303 Standard for e-Passports using Bisimilarity

Ihor Filimonov, Ross Horne, Sjouke Mauw, and Zach Smith

Computer Science and Communications, University of Luxembourg

Abstract. We clear up confusion surrounding privacy claims about the ICAO 9303 standard for e-passports. The ICAO 9303 standard includes a Basic Access Control (BAC) protocol that should protect the user from being traced from one session to another. While it is well known that there are attacks on BAC, allowing an attacker to link multiple uses of the same passport, due to differences in implementation; there still remains confusion about whether there is an attack on unlinkability directly on the BAC protocol as specified in the ICAO 9303 standard.

What are the applications of my research?

- Reversibility:

Landauer's principle

 17 languages ▼

Article [Talk](#)

[Read](#) [Edit](#) [View history](#) [Tools](#) ▼

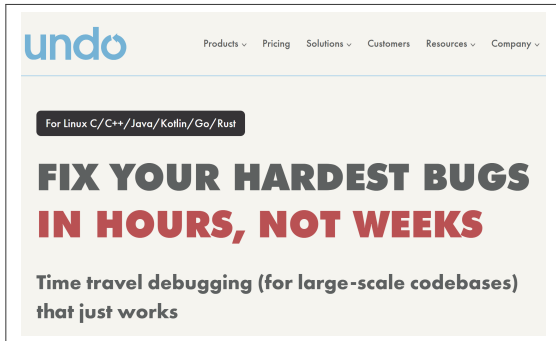
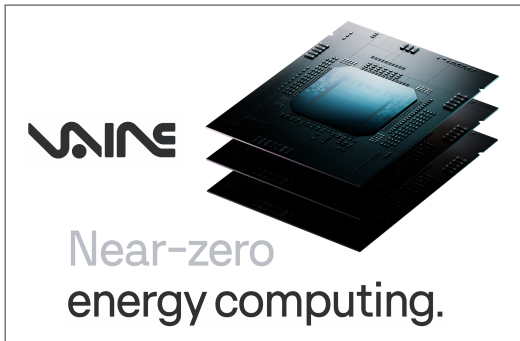
From Wikipedia, the free encyclopedia

Not to be confused with [Landau principle](#).

Landauer's principle is a [physical principle](#) pertaining to a lower [theoretical](#) limit of [energy consumption](#) of [computation](#). It holds that [an irreversible change in information stored in a computer](#), such as merging two computational paths, [dissipates a minimum amount of heat to its surroundings](#).^[1] It is hypothesized that [energy consumption below this lower bound would require the development of reversible computing](#).

What are the applications of my research?

- Reversibility: hardware, debugging



What are the applications of my research?

- Reversible concurrent calculi: eventually, reversible computers

Table of Contents

- ▶ Background notions
- ▶ Motivation and applications
- ▶ State of the art and open problems
- ▶ Current and future work

Concurrent calculi, today

Concurrent calculi: well-established, plenty of features

- CCS: *A Calculus of Communicating Systems*, Milner, 1980.
- π -calculus: *A Calculus of Mobile Processes*, Milner et al., 1992.

Concurrent calculi, today

Concurrent calculi: well-established, plenty of features

- CCS: *A Calculus of Communicating Systems*, Milner, 1980.
- π -calculus: *A Calculus of Mobile Processes*, Milner et al., 1992.

$$P ::= 0 \mid \bar{x}y.P \mid x(y).P \mid \tau.P \mid (\nu x)P \mid !P \mid P_1|P_2 \mid P_1 + P_2 \mid [x = y]P \mid [x \neq y]P$$

For an overview of their features, check *π -calculus in Coinductive Type Theory*, Honsell et al., 2001 or *An Introduction to the π -Calculus*, Parrow, 2001

Reversible concurrent calculi, today

Reversible concurrent calculi: more recent, less features

- CCSK: *Reversing algebraic process calculi*, Phillips & Ulidowski, 2007.
- CCSK^P: *The Correctness of Concurrencies in (Reversible) Concurrent Calculi*, Aubert, 2024.

Reversible concurrent calculi, today

Reversible concurrent calculi: more recent, less features

- CCSK: *Reversing algebraic process calculi*, Phillips & Ulidowski, 2007.
- CCSK^P: *The Correctness of Concurrencies in (Reversible) Concurrent Calculi*, Aubert, 2024.

$X, Y ::=$	$\mathbf{0}$	(Inactive)		$\alpha.X$	(Prefix)
	$\alpha[k].X$	(Keyed prefix)		$X + Y$	(Sum)
	$X \mid Y$	(Parallel composition)		$X \setminus a$	(Restriction)

Example of open research problem

How to define replication or recursion in reversible concurrent calculi?

Well-foundedness has to be respected: *there is no infinite reverse computation*


Concurrent Calculi Formalizations, today

[illegible]

Concurrent Calculi Formalization Benchmark, Carbone et al., 2024:
set of problems to clarify, compare and advance the state-of-the-art

Reversible Concurrent Calculi Formalizations, today

A Formalization of the Reversible Concurrent Calculus CCSK^P in Beluga

Gabriele Cecilia 

School of Computer & Cyber Sciences,
Augusta University, Augusta, USA

gcecilia@augusta.edu

Reversible concurrent calculi are abstract models for concurrent systems in which any action can potentially be undone. Over the last few decades, different formalisms have been developed and their mathematical properties have been explored; however, none have been machine-checked within a proof assistant. This paper presents the first Beluga formalization of the Calculus of Communicating Systems with Keys and Proof labels (CCSK^P), a reversible extension of CCS. Beyond the syntax and semantics of the calculus, the encoding covers state-of-the-art results regarding three relations over proof labels – namely, dependence, independence and connectivity – which offer new insights into the notions of causality and concurrency of events. As is often the case with formalizations, our encoding introduces adjustments to the informal proof and makes explicit details which were previously only sketched, some of which reveal to be less straightforward than initially assumed. We believe this work lays the foundations for future reversible concurrent calculi formalizations.

A Formalization of the Reversible Concurrent Calculus CCSKP in Beluga, 2025

Reversible Concurrent Calculi Formalizations, today

A Formalization of the Reversible Concurrent Calculus CCSK^P in Beluga

Gabriele Cecilia ☺

School of Computer & Cyber Sciences,
Augusta University, Augusta, USA

gcecilia@augusta.edu

Reversible concurrent calculi are abstract models for concurrent systems in which any action can potentially be undone. Over the last few decades, different formalisms have been developed and their mathematical properties have been explored; however, none have been machine-checked within a proof assistant. This paper presents the first Beluga formalization of the Calculus of Communicating Systems with Keys and Proof labels (CCSK^P), a reversible extension of CCS. Beyond the syntax and semantics of the calculus, the encoding covers state-of-the-art results regarding three relations over proof labels – namely, dependence, independence and connectivity – which offer new insights into the notions of causality and concurrency of events. As is often the case with formalizations, our encoding introduces adjustments to the informal proof and makes explicit details which were previously only sketched, some of which reveal to be less straightforward than initially assumed. We believe this work lays the foundations for future reversible concurrent calculi formalizations.

Open problem: formalization of reversible concurrent calculi

Table of Contents

- ▶ Background notions
- ▶ Motivation and applications
- ▶ State of the art and open problems
- ▶ Current and future work

Current work

A Formalization of the Reversible Concurrent Calculus CCSKP in Beluga:
sections 3-4 of *Independence and Causality in the Reversible Concurrent Setting*,
Aubert et al., 2025.

Mechanized definitions and proofs: syntax, semantics, complementarity of
dependence and independence, characterization of connectivity

Current work

A Formalization of the Reversible Concurrent Calculus CCSKP in Beluga:
sections 3-4 of *Independence and Causality in the Reversible Concurrent Setting*,
Aubert et al., 2025.

Mechanized definitions and proofs: syntax, semantics, complementarity of
dependence and independence, characterization of connectivity

→ **Journal paper**, extending both theoretical and formalized results

Mechanized definitions and proofs: bijection between CCSK^P and CCSK , square
property, well-foundedness, ...

Future work

- *An Axiomatic Theory for Reversible Computation*, Lanese et al., 2024:
List of axioms characterizing well-behaved reversible concurrent calculi
→ Formalization of such results
- Improve reversible concurrent calculi, e.g. by adding replication/recursion

Thank you for listening!
Any questions?

Slides available at:
<https://gabrielececilia.github.io/>