Quantitativity and Fixed Points in the Infinitary Relational Semantics of Linear Logic

Pierre Clairambault* Charles Grellois†

May 2025 (work in progress)

Abstract

Grellois and Melliès introduced in 2015 an infinitary variant of the relational model of linear logic [GM15a], in which the exponential can build multisets with finite or countable multiplicities. Their aim was to use such models for higher-order model-checking, in which denotational models need to capture finite programs generating finite width trees with branches of at most countable length, generated by infinitary recursion. While the traditional, finitary relational semantics admits a unique, uniform fixed point operator, they showed that the (countably) infinitary variant has several distinct fixed points. In this talk, we explain that there was an error in the paper: the fixed point they give, which reflects the traditional quantitative interpretation of the relational semantics of linear logic, is not the greatest fixed point as they claimed. We also discuss consequences, and relations between the greatest fixed point and this intermediate, quantitative fixed point.

1 Introduction

Higher-order model-checking is an approach to the verification of functional programs, in which these programs are modeled as trees of *traces*, of finite width but with branches that can be of countable depth. The main historical problem, that stayed open for a couple of decades, was to decide whether such a tree satisfies a given formula of monadic second-order logic (MSO). Ong proved in 2006 [Ong06] that this problem is decidable, by adapting game semantics to analyse the finite lamba-term with (typed) recursion that generates the tree, and which is the only finite representation we have at hand. This adaptation is such that properties over this infinite tree computed by infinitary evaluation can actually be inferred by static analysis from the finite λY-term that produces it.

This result has since attracted quite a lot of attention, leading to numerous alternate proofs both in the automata community [HMOS08, CS12], in the types community [KO09], and in the semantics community [SW15, GM15b, Gre16, Wal19]. The importance of this quest for decidability in higher-order model-checking probably lies in that it led, in each of these approaches, to new theoretical developments. In the linear logic-based approach of Grellois and Melliès, revisiting the decidability result has among others led to introducing an infinitary (at most countable) variant of the traditional relational semantics of linear logic, and two fixpoints over them (plus the coloured fixpoint operator of [GM15a], which is outside the scope of this talk).

In this talk, we will explain that there was an error in Grellois' and Melliès' article, which hid an important aspect of the infinitary relational semantics: the existence of a *quantitative* fixpoint operator, lying strictly in between the *finitary* least fixed point and the *infinitary* greatest fixpoint. The latter suffers from a form of "non-productivity" from the environment side, which affects the parametrized version of the fixed point operator: a denotation computed using the greatest fixed point to model recursion can contain information about resources that will *not* be used in a computation. A denotation that claims to use some resource *later on* can claim it forever, without actually doing it. This is reminiscent of what justified the introduction of the *parity condition* in logic and in verification.

*CNRS, Aix Marseille Univ, LIS, Marseille, France

†School of Computer Science, University of Sheffield, Sheffield, UK

2 The Infinitary Relational Semantics of Linear Logic

Here we recall the main ingredients of the definition of the infinitary relational semantics from [GM15a], see also [Gre16] for proof details. The main changes lie in the cardinality of the sets under study, and on the definition of the exponential functor. Note that a proper treatment of enumerations of the elements of multisets, as we commonly do in the finitary case, requires more care, see the original paper for details.

Definition 1. 1. The category $\underline{Rel}$ has the sets A, B of cardinality below the cardinality of the reals as objects, and binary relations $f \subseteq A \times B$ between A and B as morphisms $A \to B$.

2. A countable multiset of elements over a set A is a function $A \to \mathbb{N} \cup \{\omega\}$, where ω represents the countable multiplicity[1].

3. The infinitary exponential $\lightning A$ of a set A maps it to the set of countable multisets over it.

Tensor product, cartesian product... are obtained precisely as in the traditional relational semantics.

Proposition 1 ([GM15a])**.** *The infinitary relational semantics is a (Seely) model of linear logic.*

3 Three Fixed Point Operators

Grellois and Melliès, motivated by their quest for applications to higher-order model-checking, introduced fixed point operators in the infinitary relational semantics in a way that was mimicking the behaviour of a tree automaton, but lifted to elements of the semantics rather than on mere labels from a tree signature. Such an higher-order lifting of the behaviour of a tree automaton is key to most proofs of decidability of higher-order model-checking: the only finite representation we have of the infinite tree we want to analyse in the finite λY-term that generates it.

For the sake of clarity, and of conciseness, we take a more informal approach to the definition of two of the fixed points from [GM15a]. We will start with a generic scheme, and then explain how it gives birth to two different fixed point operators, before discussing case of the greatest fixed point.

Let $f : \lightning X \otimes \lightning A \multimap A$ be a morphism in $\underline{Rel}$. We see each element $((m_X, m_A), a) \in f$ as a tree whose root is labeled with a, and with one leaf by element of m_X and of m_A – we enumerate them as leaves, skipping here the details of how to do this enumeration when it involves a concatenation of infinite sequences.

We then define a glueing of such trees: a leaf labeled by an element of A can be replaced by a tree (coming from f) whose root has the same label.

We say that a tree is a (finitary; infinitary) *Y-witness* over f when:

1. it comes directly from f, or from glueing of elements coming from f (with branches of finite length for finitary witnesses; infinitary witnesses may have countable branches),

2. and none of its leaves is labeled by an element of A.

These trees represent compositions of elements of the relation f, iterated until no further composition is possible, and with a restriction to finite depth for finitary Y-witnesses. They represent operationally how fixpoints are computed.

We define parametrized fixed point operators that map f to a morphism $\lightning X \multimap A$. The elements (m, a) of the resulting relating are the ones such that:

1. there exists a Y-witness over f whose root is labelled with a,

2. and m is obtained as the (possibly countable) multiset sum of all the leaves of this Y-witness.

If we restrict to *finitary* Y-witnesses for f, we obtain its least fixed point. If we use infinitary Y-witnesses, it was wrongly claimed in [GM15a] that we obtained the greatest fixed point of $\underline{\text{Rel}}$.

[1] Since we restricted the objects of $\underline{Rel}$ to finite-or-countable sets, we do not need an adaptation of the notion of finite support to the infinitary case here.

Indeed, if we consider the definition of the greatest fixed point induced by the Tarski-Knaster theorem, we can for instance prove that:

$$([\star], \star) \quad \in \quad [\![Y\,(\lambda F.\lambda x.(b\,(F\,x)))]\!]$$

with $b : o \to o$ a constant.

However, this recursive term fully evaluates to the infinite tree with only one branch b^ω, which *does not* use the variable x. The traditional quantitative intuition from the finitary relational semantics is therefore violated here. On the other hand, the fixed point operator based on infinitary Y-witnesses does *not* allow this denotation if used to interpret the above recursive term.

In other words, the infinitary relational semantics of linear logic admits at least three distinct fixed point operators:

- the least fixed point, in the spirit of induction,
- the greatest fixed point, in the spirit of coinduction,
- and this *quantitative* fixed point operator based on Y-witness trees.

In higher-order model-checking, other fixed points are at play, mixing the least and the quantitative fixed point operator within a same fixed point, see for instance [GM15a]. Since then, some approaches such as [Wal19] have introduced an encoding that allows to restrict to independent uses of both the least and the quantitative fixed points[2], achieving the same goal without creating more technical difficulties. We restricted our attention to these three fixed points for this reason.

4 Perspectives

The existence of such a fixpoint operator of a quantitative flavour, lying strictly in between the least and greatest fixpoints of the infinitary relational model, opens several significant questions that the authors are investigating at the moment:

1. Can one collapse the greatest fixed point to the quantitative one? It seems like the quantitative fixed point can be obtained from the greatest fixed point by removing the denotations introducing elements that are not accessed during the computation.
2. Study the notion of *co-productivity* that is at play here: while productivity models the fact that a program should always output an extra part of the result of an infinitary computation after finitely many reduction steps, co-productivity would ensure that each resource is used after a finite amount of steps. Sound criteria for productivity could be adapted via a form of duality to provide sound criteria for co-productivity.
3. The authors work together on an analysis of higher-order model-checking based on contemporary game semantics. Strategies, being quantitative by essence, seem to relate to the quantitative fixed point, while in the finitary semantics they would relate to the greatest fixed point when used to fill a similar purpose.

The talk will follow the structure of this extended abstract, so as to introduce the audience to the infinitary relational semantics; to introduce these three fixed point operators and emphasize on their differences; and will open the aforementioned questions, hoping for fruitful discussions along these lines during the workshop.

References

[CS12] Arnaud Carayol and Olivier Serre. Collapsible pushdown automata and labeled recursion schemes: Equivalence, safety and effective selection. In *Proceedings of the 27th Annual IEEE Symposium on Logic in Computer Science, LICS 2012, Dubrovnik, Croatia, June 25-28, 2012*, pages 165–174. IEEE Computer Society, 2012.

[2] Walukiewicz (and former work by Salvati and Walukiewicz) use finitary semantics, in which they mix least and greatest fixed point. However, doing the same in the infinitary relational setting would require considering the quantitative fixpoint rather than the greatest, to achieve similar purposes.

[GM15a] Charles Grellois and Paul-André Melliès. An infinitary model of linear logic. In Andrew M. Pitts, editor, *FoSSaCS 2015 Proceedings*, volume 9034 of *LNCS*, pages 41–55. Springer, 2015.

[GM15b] Charles Grellois and Paul-André Melliès. Relational semantics of linear logic and higher-order model checking. In Stephan Kreutzer, editor, *24th EACSL Annual Conference on Computer Science Logic, CSL 2015, September 7-10, 2015, Berlin, Germany*, volume 41 of *LIPIcs*, pages 260–276. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2015.

[Gre16] Charles Grellois. *Semantics of linear logic and higher-order model-checking. (Sémantique de la logique linéaire et "model-checking" d'ordre supérieur).* PhD thesis, Paris Diderot University, France, 2016.

[HMOS08] Matthew Hague, Andrzej S. Murawski, C.-H. Luke Ong, and Olivier Serre. Collapsible pushdown automata and recursion schemes. In *Proceedings of the Twenty-Third Annual IEEE Symposium on Logic in Computer Science, LICS 2008, 24-27 June 2008, Pittsburgh, PA, USA*, pages 452–461. IEEE Computer Society, 2008.

[KO09] Naoki Kobayashi and C.-H. Luke Ong. A type system equivalent to the modal mu-calculus model checking of higher-order recursion schemes. In *Proceedings of the 24th Annual IEEE Symposium on Logic in Computer Science, LICS 2009, 11-14 August 2009, Los Angeles, CA, USA*, pages 179–188. IEEE Computer Society, 2009.

[Ong06] C.-H. Luke Ong. On model-checking trees generated by higher-order recursion schemes. In *21th IEEE Symposium on Logic in Computer Science (LICS 2006), 12-15 August 2006, Seattle, WA, USA, Proceedings*, pages 81–90. IEEE Computer Society, 2006.

[SW15] Sylvain Salvati and Igor Walukiewicz. Using models to model-check recursive schemes. *Log. Methods Comput. Sci.*, 11(2), 2015.

[Wal19] Igor Walukiewicz. Lambda y-calculus with priorities. In *34th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2019, Vancouver, BC, Canada, June 24-27, 2019*, pages 1–13. IEEE, 2019.