Enhanced Induction in Behavioural Relations

Davide Sangiorgi
University of Bologna, Italy
INRIA, Sophia Antipolis, France

Abstract

We outline an attempt at transporting the well-known theory of enhancements for the *coinduction* proof method, widely used on behavioural relations such as bisimilarity, onto the realms of *inductive* behaviour relations, i.e., relations defined from inductive observables, and discuss relevant literature.

2012 ACM Subject Classification Theory of computation → Program semantics

Keywords and phrases coinduction, induction, semantics, behavioural relations

Digital Object Identifier 10.4230/LIPIcs.CSL.2023.4

Category Invited Talk

Related Version *Full Version*: https://dl.acm.org/doi/10.1145/3498679

1 Discussion

In this paper we discuss, informally, an attempt [35] at transporting the well-known theory of enhancements for the *coinduction* proof method, widely used on behavioural relations such as bisimilarity, onto the realms of *inductive* behaviour relations, i.e., relations defined from inductive observables. We also comment on the relevant literature. We refer to [35] for more details.

Behavioural relations (equalities, preorders) represent a one of the most basic elements for reasoning on programs or systems, because any transformation or property that we wish to prove is supposed to be in agreement with the behavioural relation adopted. A number of proposals for behavioural preorders and equalities have been made in the literature; see, e.g, van Glabbeek's spectrum [11,12]. Among the notions so formulated, bisimilarity has emerged as one of the most studied and used [19, 21]. While introduced in Concurrency Theory, bisimilarity has spread to other areas of Computer Science, as well as to other domains such as Mathematics and Cognitive Science.

Bisimilarity is the union of all bisimulations. A bisimulation is a relation on the terms of a language that is invariant under the observables of the language (i.e., what can be observed of the terms). Thus the definition itself immediately leads to a well-established proof technique: to prove two terms bisimilar, find a bisimulation relation containing the two terms as a pair. Furthermore, such a proof method can be enhanced, with the goal of making it more effective (easier to use, both in paper proofs and in tools for automated or semi-automated analysis) and more broadly applicable. Examples of enhancements are "up-to context", "up-to transitive closure", "up-to bisimilarity", "up-to environment", and so on [29]. Theories of enhancements have been proposed [25, 26, 31]. Most important, these theories allow one to *combine* enhancements so to obtain, for free, the soundness of more complex enhancements. Bisimilarity and the bisimulation proof method are instances of a coinductive definition and of the coinduction proof method. Analogously the bisimulation enhancements can be lifted to coinduction; see [28] for a presentation that follows fixed-point theory. Abstract formulations of the meaning of coinductive enhancements have also been given using category theory. The main technical tools are final semantics, coalgebras, spans of coalgebra homomorphisms, fibrations, and corecursion schemes. See, e.g., [4, 27, 30] (based on earlier works such as [3, 15, 17, 18, 37]), and [2, 5, 6, 16]. Enhancements of corecursion

31st EACSL Annual Conference on Computer Science Logic (CSL 2023).
Editors: Bartek Klin and Elaine Pimentel; Article No. 4; pp. 4:1–4:6
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

schemes may also be examined using the generalised powerset construction [36]. For more details on coinductive enhancements, we refer to the technical survey [28] and to the historical review [29].

The bisimulation proof method and its enhancements are a major reason for the success of bisimilarity. Sometimes the enhancements seem essential to be able to carry out a proof: defining a full bisimulation, with all needed pairs, can be considerably hard, let alone carrying out the whole proof. This is frequent in languages for name mobility such as the π-calculus and its many dialects, and in languages including higher-order features such as λ-calculi. In these languages bisimilarity is hardly ever applied without enhancements.

As a behavioural equivalence, however, bisimilarity has also been criticised. One of the main arguments is that it may be regarded as too fine, discriminating processes that an external observer could not tell apart. For instance, in the CSP community failure equivalence [7] is used in place of bisimilarity. Another argument against bisimilarity is that it does not have a natural associated preorder. For instance, similarity – the "one-way" bisimilarity – or variants of it do not yield bisimilarity as their induced equivalence [33]. (Further, similarity as a behavioural preorder is often inadequate because it does not respect deadlocks.) Various inductive behavioural relations, both preorders and equivalences, have been put forward and studied that improve on such limitations: examples are preorders based on traces, failures, ready sets, refusals, may and must testing, ready and failure traces, e.g., [1, 7–9, 12, 13, 20, 22–24]. We call them "inductive" because resulting from inductively-defined observables, usually enriched forms of traces. Correspondingly, we sometimes call "inductive" the resulting enhancements.

At the heart of theories of bisimulation enhancements such as [25, 26, 31] is the notion of *progression*. A progression from a relation $\mathcal{R}$ to a relation $\mathcal{S}$, written $\mathcal{R} \rightarrowtail \mathcal{S}$, indicates that pairs of processes in $\mathcal{R}$ can match each other's actions and their derivatives (i.e., the processes resulting from performing such actions) are in $\mathcal{S}$. The progressions that are considered are of the form $\mathcal{R} \rightarrowtail \mathcal{F}(\mathcal{R})$, where $\mathcal{F}$ is a function on relations. Conditions on functions on relations guarantee *soundness* of the progressions, meaning that if $\mathcal{R} \rightarrowtail \mathcal{F}(\mathcal{R})$ then $\mathcal{R}$ only includes pairs of bisimilar processes. These are functorial-like conditions such as *respectfulness* [31] and *compatibility* [25]. Such conditions can then be extended to higher-order functions, also called *constructors*, so to be able to combine sound functions, that is, to derive sophisticated sound functions – and hence sophisticated proof techniques for bisimilarity – from simpler ones. As an example, "up-to bisimilarity" can be combined with "up-to context" yielding "up-to bisimilarity and context", in which one is allowed, in the bisimulation game, both to rewrite the derivative processes into bisimilar ones, and to remove, in the resulting terms, a common context. In fact, in this way even "up-to bisimilarity" is derivable from simpler functions, namely the identity function and the constant function mapping every relation onto bisimilarity itself [31].

To investigate the enhancements in an inductive setting, in [35] the observables for the inductive behavioural preorders and equivalences are described by means of *modal formulas*. The operators include the "diamond" $\langle \mu \rangle . \theta$, to detect the possibility of performing the action μ, a (possibly infinitary) 'and', to permit multiple observations, and a set of atomic observables. (Without the atomic observables, it is the positive fragment of Hennessy-Milner logic [14].) Progressions are maintained as the basic schema for studying enhancements. In fact, one-way progressions, called semi-progressions and written $\mathcal{R} \rightarrowtail\!\!\rightharpoonup \mathcal{S}$, are employed, aiming to capture also preorders (besides equivalences). The meaning of soundness, and the conditions on functions to guarantee soundness, are however modified. Moreover, everything is parametrised on a preorder, say $\precsim$, as the theory is supposed to be applicable to different preorders. Thus, in such enhancements, a function $\mathcal{F}$ is *sound for* $\precsim$ if $\mathcal{R} \rightarrowtail \mathcal{F}(\mathcal{R})$ implies that $\mathcal{R}$ is included in the given preorder $\precsim$; and similarly for equivalences. The crux of this theory of inductive enhancements is the condition for functional soundness that should permit composition of enhancements. The condition hinges on the inductive definition of the observables. A weight is associated to each observable, intuitively expressing the depth of the nesting of actions in the behaviour of a process that may have be looked at when checking that observable. This yields a stratification of the preorder $\precsim$; at stage n of the preorder, $\precsim_n$, only the observables of weight less than or equal to n are taken into account. The condition for functional soundness, called *weight-preservation*, requires that if $\mathcal{R}$ complies with $\precsim_n$, i.e., $\mathcal{R} \subseteq \precsim_n$, then also $\mathcal{F}(\mathcal{R})$ should comply, i.e., $\mathcal{F}(\mathcal{R}) \subseteq \precsim_n$. In the case of equivalences, rather than preorders, one adds analogous converse requirements on pairs of related processes.

Common basic functions and constructors in the literature about bisimulation enhancements are shown to be weight-preserving, and may therefore be used also in inductive enhancements. Examples of basic functions and constructors are function composition, union, chaining (that gives us relational composition), and the context-closure function These closure properties allow one to derive sophisticated sound functions (and hence sophisticated proof techniques) from simpler ones. Examples of derived functions are the transitive-closure function, the closure under context and $\precsim$ (the analogous of the "up-to context and bisimilarity" enhancement for bisimilarity).

The inductive preorders and equivalences considered in [35] are the best-known relations in the literature, following [11, 12]. They include the trace, failure, failure trace, ready, and ready trace preorders (other preorders, like may and must testing and refusal, coincide with some of these, under mild conditions on the transitions performed by the processes), and their induced equivalences. In all cases, the soundness of the above functions and constructors is usually straightforward, the only exception being the context-closure function. As the theory of inductive enhancement is parametrised on a preorder, proofs can sometimes be made *parametric* on such a preorder, so to make them valid for a number of preorders. See [35] for examples.

The paper [35] develops its work for ordinary Labeled Transition Systems (LTSs). A theory is proposed both for strong semantics, where all actions are equally visible, and for weak semantics, in which a special action denoting internal activity may be partly or completely ignored. It is well-known that theories of weak coinductive enhancements tend to be rather more involved than the "strong" theories. For instance, a useful constructor, chaining, is sound only in the strong theories; to compensate for this, auxiliary relations such as *expansion* [32] and *contraction* [34] have been introduced. Similar issues show up in the inductive enhancements. In addition, some of the weak behavioural relations make use of state predicates such as *stability*, which do not appear in the strong case. Some of the technical solutions that are adopted for the inductive setting are inspired from those used in the coinductive setting, others are specific to the weight-based conditions for induction mentioned above. For instance, different forms of weak weight are considered (e.g., distinguishing the contribution of internal and visible actions), and their relative advantages and disadvantages are examined.

Another approach at transferring the coinductive enhancements to an inductive setting is based on the techniques of unique solutions of equations [10, 34]. Such proof techniques (for weak bisimilarity) employ equations as well as special inequations called *contractions*. The techniques give one the power of some bisimulation enhancements such as "up-to context", and can be transferred to relations such as trace equivalence and trace preorder. In general the techniques seem to have a limited applicability to preorders: they can only be used to show that a given process is related to the "syntactic solution of an equation", that is, the process whose syntactic definition is the equation itself.

2 Further work

We comment some directions of work to be explored. First, it would be interesting to see if and how the theory of inductive enhancements can be formulated in a more abstract setting, e.g., fixed-point theory or category theory. Proposals along these lines exist for the coinductive enhancements. Their meaning in the inductive setting is unclear, both because the observables are inductive and because of the coinductive flavour of the semi-progressions at the heart of the theory. Also, it is unclear how the theory could be lifted to a probabilistic setting and labelled Markov processes.

A powerful enhancement is up-to context. The paper [35] uses first-order LTSs and examines a a CCS-like language. Here an objective would be to examine general conditions that guarantee its soundness, more precisely the weight-preserving property. Such conditions could, for instance, look at the format of the rules defining the operational behaviour of the operators of the language. In coinduction, up-to-context has been shown very effective in *higher-order languages*, such as λ-calculi or languages enriched with functional features, and *nominal languages* such as the π-calculus. The objective could thus be extended towards the transfer of the inductive enhancements to these classes of languages.

Finally, it would be interesting to see if the theory can be lifted to behavioural relations that make use of both inductive and coinductive observables. Examples of observables that are naturally defined coinductively are infinite traces and divergence. Divergence in particular often appears in definition of *weak* behavioural relations (e.g., failure semantics and must testing).

References

1. Jos C. M. Baeten, Jan A. Bergstra, and Jan Willem Klop. Ready-trace semantics for concrete process algebra with the priority operator. *Comput. J.*, 30(6):498–506, 1987.
2. F. Bartels. Generalised coinduction. *Math. Struct. in Computer Science*, 13(2):321–348, 2003. `doi:10.1017/S0960129502003900`.
3. F. Bartels. *On generalised coinduction and probabilistic specification formats*. PhD thesis, CWI, Amsterdam, April 2004.
4. Henning Basold, Damien Pous, and Jurriaan Rot. Monoidal company for accessible functors. In *Proc. CALCO*, volume 72 of *LIPIcs*. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2017. `doi:10.4230/LIPIcs.CALCO.2017.5`.
5. Filippo Bonchi, Barbara König, and Daniela Petrisan. Up-to techniques for behavioural metrics via fibrations. In Sven Schewe and Lijun Zhang, editors, *CONCUR'18,*, volume 118 of *LIPIcs*, pages 17:1–17:17. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2018. `doi:10.4230/LIPIcs.CONCUR.2018.17`.
6. Filippo Bonchi, Daniela Petrisan, Damien Pous, and Jurriaan Rot. A general account of coinduction up-to. *Acta Inf.*, 54(2):127–190, 2017. `doi:10.1007/s00236-016-0271-4`.
7. Stephen D. Brookes, C. A. R. Hoare, and A. W. Roscoe. A theory of communicating sequential processes. *J. ACM*, 31(3):560–599, 1984. `doi:db/journals/jacm/BrookesHR84.html,10.1145/828.833`.
8. Stephen D. Brookes and A. W. Roscoe. An improved failures model for communicating processes. In Stephen D. Brookes, A. W. Roscoe, and Glynn Winskel, editors, *Seminar on Concurrency*, volume 197 of *Lecture Notes in Computer Science*, pages 281–305. Springer Verlag, 1984.
9. R. De Nicola and R. Hennessy. Testing equivalences for processes. *Theoretical Computer Science*, 34:83–133, 1984.

10 Adrien Durier, Daniel Hirschkoff, and Davide Sangiorgi. Divergence and unique solution of equations. *Logical Methods in Computer Science*, 15(3), 2019. doi:10.23638/LMCS-15(3:12)2019.

11 R.J. van Glabbeek. The linear time—branching time spectrum II (the semantics of sequential systems with silent moves). In E. Best, editor, *Proc. CONCUR '93*, volume 715. Springer Verlag, 1993. doi:10.1007/3-540-57208-2_6.

12 R.J. van Glabbeek. The linear time—branching time spectrum I. In A. Ponse J. Bergstra and S. Smolka, editors, *Handbook of Process Algebra*, pages 3–99. Elsevier, 2001. doi:10.1016/b978-044482830-9/50019-9.

13 M. Hennessy. *Algebraic Theory of Processes*. The MIT Press, Cambridge, Mass., 1988.

14 M. Hennessy and R. Milner. Algebraic laws for nondeterminism and concurrency. *Journal of the ACM*, 32:137–161, 1985.

15 Bart Jacobs. Distributive laws for the coinductive solution of recursive equations. *Information and Computation*, 204(4):561–587, 2006. doi:10.1016/j.ic.2005.03.006.

16 Marina Lenisa. From set-theoretic coinduction to coalgebraic coinduction: some results, some problems. *Electronical Notes in Computer Science*, 19:2–22, 1999. doi:10.1016/S1571-0661(05)80265-8.

17 Marina Lenisa, John Power, and Hiroshi Watanabe. Distributivity for endofunctors, pointed and co-pointed endofunctors, monads and comonads. *Electronical Notes in Computer Science*, 33:230–260, 2000. doi:10.1016/S1571-0661(05)80350-0.

18 S. Milius, L. S. Moss, and D. Schwencke. Abstract GSOS rules and a modular treatment of recursive definitions. *Logical Methods in Computer Science*, 9(3), 2013. doi:10.2168/LMCS-9(3:28)2013.

19 R. Milner. *Communication and Concurrency*. Prentice Hall, 1989.

20 Ernst-Rüdiger Olderog and C. A. R. Hoare. Specification-oriented semantics for communicating processes. *Acta Inf.*, 23(1):9–66, 1986.

21 D. Park. A new equivalence notion for communicating systems. In G. Maurer, editor, *Bulletin EATCS*, volume 14, pages 78–80, 1981. Abstract of the talk presented at the Second Workshop on the Semantics of Programming Languages, Bad Honnef, March 16–20 1981. Abstracts collected in the Bulletin by B. Mayoh.

22 Iain Phillips. Refusal testing. *Theor. Comput. Sci.*, 50:241–284, 1987. A preliminary version in Proc. ICALP'86, Lecture Notes in Computer Science 226, Springer Verlag.

23 A. Pnueli. Linear and branching structures in the semantics and logics of reactive systems. In W. Brauer, editor, *12th ICALP*, volume 194 of *Lecture Notes in Computer Science*, pages 15–32. Springer Verlag, 1985.

24 Lucia Pomello. Some equivalence notions for concurrent systems. an overview. In Grzegorz Rozenberg, editor, *Advances in Petri Nets 1985*, volume 222 of *Lecture Notes in Computer Science*, pages 381–400. Springer, 1985. doi:10.1007/BFb0016202.

25 D. Pous. Complete lattices and up-to techniques. In *Proc. APLAS '07*, volume 4807 of *Lecture Notes in Computer Science*, pages 351–366. Springer Verlag, 2007. doi:10.1007/978-3-540-76637-7_24.

26 Damien Pous. Coinduction all the way up. In *Proc. LICS*, pages 307–316. ACM, 2016. doi:10.1145/2933575.2934564.

27 Damien Pous and Jurriaan Rot. Companions, Codensity, and Causality. In *Proc. FoSSaCS*, volume 10203 of *Lecture Notes in Computer Science*, pages 106–123. Springer Verlag, 2017. doi:10.1007/978-3-662-54458-7_7.

28 Damien Pous and Davide Sangiorgi. Enhancements of the bisimulation proof method. In Davide Sangiorgi and Jan Rutten, editors, *Advanced Topics in Bisimulation and Coinduction*. Cambridge University Press, 2012.

29 Damien Pous and Davide Sangiorgi. Bisimulation and coinduction enhancements: A historical perspective. *Formal Asp. Comput.*, 31(6):733–749, 2019. doi:10.1007/s00165-019-00497-w.

30 Jurriaan Rot, Filippo Bonchi, Marcello M. Bonsangue, Damien Pous, Jan Rutten, and Alexandra Silva. Enhanced coalgebraic bisimulation. *Mathematical Structures in Computer Science*, 27(7):1236–1264, 2017. doi:10.1017/S0960129515000523.

31 D. Sangiorgi. On the bisimulation proof method. *Journal of Mathematical Structures in Computer Science*, 8:447–479, 1998. doi:10.1017/S0960129598002527.

32 D. Sangiorgi and R. Milner. The problem of "Weak Bisimulation up to". In W.R. Cleveland, editor, *Proc. CONCUR '92*, volume 630 of *Lecture Notes in Computer Science*, pages 32–46. Springer Verlag, 1992. doi:10.1007/BFb0084781.

33 Davide Sangiorgi. *Introduction to Bisimulation and Coinduction*. Cambridge University Press, 2012. doi:10.1017/CBO9780511777110.

34 Davide Sangiorgi. Equations, contractions, and unique solutions. *ACM Trans. Comput. Log.*, 18(1):4:1–4:30, 2017. doi:10.1145/2971339.

35 Davide Sangiorgi. From enhanced coinduction towards enhanced induction. *Proc. ACM Program. Lang.*, 6(POPL):1–29, 2022. doi:10.1145/3498679.

36 A. Silva, F. Bonchi, M. Bonsangue, and J. Rutten. Generalizing the powerset construction, coalgebraically. In *FSTTCS*, LIPIcs, pages 272–283. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2010. doi:10.4230/LIPIcs.FSTTCS.2010.272.

37 Tarmo Uustalu, Varmo Vene, and Alberto Pardo. Recursion schemes from comonads. *Nord. J. Comput.*, 8(3):366–390, 2001. URL: http://www.cs.helsinki.fi/njc/References/uustaluvp2001:366.html.