Towards a Coinductive Characterization of Computational Indistinguishability

(Extended Abstract)

Alberto Cappai and Ugo Dal Lago
Università di Bologna & INRIA
{cappai,dallago}@cs.unibo.it

***Abstract*—Computational indistinguishability (CI in the following) is one of the most central concepts in modern cryptography, and many other definitions (e.g. pseudorandomness, security of cryptographic schemes) can be formulated in terms of CI. We present the results of a study directed towards giving a direct and precise characterization of computational indistinguishability in an higher-order functional language for polynomial time computability, in which tools from implicit computational complexity and coinduction both play a central role.**

Contrarily to formal cryptography as embodied by the Dolev-Yao model, computational cryptography is founded around the concepts of a probabilistic polynomial time algorithm and of a negligible (and in general non-null) probability of "error". This makes the task of proving primitives and protocols secure much harder than in formal security. The literature offers at least two ways to overcome the problem:

- On the one hand, one can prove that formal security is, usually with some restriction, *sound* with respect to computational security. This approach, pioneered by Abadi and Rogaway [1], has lead to a number of studies about the computational soundness of various logics and formal techniques (e.g. [2], [3])
- On the other, one can go towards *computer-assisted cryptographic proofs* [4], where the burden of evaluating (and ultimately bounding) the probability of success of any adversary is alleviated by providing a (semi)-automatic framework for game-based proofs [5]. A recent example of this approach is EasyCrypt.

This work aims at pointing to a "third way": computational indistinguishability of two programs can be proved by coinductive methodologies akin to applicative bisimulation [6], instantiated on programming languages in which *only* polytime programs can be written [7]. This allows to transparently restrict the class of adversaries which need to be taken into account, to the adversaries running in polynomial time, but also of allowing to get rid of the universal quantification over all adversaries (or distinguishers, or contexts) which is inherent to all definitions of CI. In other words, we are following the path traced by [8], but with a special emphasis on higher-order computation and full-abstraction results.

What makes this goal more likely to be achieved now than in the past are recent advances in the fields of implicit computational complexity (ICC in the following) and bisimulation techniques. More specifically, recent work in ICC aims at characterizing *probabilistic* complexity classes, and takes the form of λ-calculi with probabilistic choice in which every (first-order) program is guaranteed to take at most polynomial time when executed [9]. Coinductive techniques, which are well-known to work well in higher-order deterministic or nondeterministic settings, have been recently adapted to *probabilistic* λ*-calculi* [10], obtaining sometime unexpected results [11]. The main of idea behind this work, then, consists in combining these two ingredients.

The contributions we will talk about (which cannot be all described here) can be summarized as follows:

- We introduce RSLR, a typed λ-calculus for probabilistic polynomial time computation, and a notion of *probabilistic applicative bisimulation* for it, following [9], [10].
- We prove the latter sound with respect to a notion of *exact* context equivalence, in which the context interact with terms of arbitrary types, but the probability of getting any observation must be exactly the same. We also discuss the issue of full-abstraction for the obtained calculus.
- We hint at how exact context equivalence can be relaxed into a notion of *approximate* context equivalence, this way capturing CI.

References

[1] M. Abadi and P. Rogaway, "Reconciling two views of cryptography (the computational soundness of formal encryption)," *J. Cryptology*, vol. 20, no. 3, p. 395, 2007.

[2] P. Adão, G. Bana, J. Herzog, and A. Scedrov, "Soundness and completeness of formal encryption: The cases of key cycles and partial information leakage," *Journal of Computer Security*, vol. 17, no. 5, pp. 737–797, 2009.

[3] M. Hajiabadi and B. M. Kapron, "Computational soundness of coinductive symbolic security under active attacks," in *TCC*, ser. LNCS, vol. 7785, 2013, pp. 539–558.

[4] V. Shoup, "Sequences of games: a tool for taming complexity in security proofs," *IACR Cryptology ePrint Archive*, vol. 2004, p. 332, 2004.

[5] G. Barthe, B. Grégoire, and S. Z. Béguelin, "Formal certification of code-based cryptographic proofs," in *POPL*, 2009, pp. 90–101.

[6] S. Abramsky, "The Lazy λ-Calculus," in *Research Topics in Functional Programming*, D. Turner, Ed. Addison Wesley, 1990, pp. 65–117.

[7] S. Bellantoni and S. A. Cook, "A new recursion-theoretic characterization of the polytime functions (extended abstract)," in *STOC*, 1992, pp. 283–293.

[8] J. C. Mitchell, A. Ramanathan, A. Scedrov, and V. Teague, "A probabilistic polynomial-time process calculus for the analysis of cryptographic protocols," *Theor. Comput. Sci.*, vol. 353, no. 1-3, pp. 118–164, 2006.

[9] U. Dal Lago and P. P. Toldin, "A higher-order characterization of probabilistic polynomial time," in *FOPARA*, ser. LNCS, vol. 7177, 2011, pp. 1–18.

[10] U. Dal Lago, D. Sangiorgi, and M. Alberti, "On coinductive equivalences for higher-order probabilistic functional programs," in *POPL*, 2014, pp. 297–308.

[11] R. Crubillé and U. Dal Lago, "On probabilistic applicative bisimulation and call-by-value -calculi," in *ESOP*, ser. LNCS, vol. 8410, 2014, pp. 209–228.