Internship Proposal
Revisiting Contextual Equivalences for Process Algebras

Keywords	Process Algebra, Concurrency, Testing Equivalences, Process Semantics
Location	Univ. Paris Est Creteil, LACL, F-94010 Creteil, France, or Augusta University, GA, USA
Advisors	- Clément Aubert, School of Computer and Cyber Sciences, Augusta University
	- Daniele Varacca, LACL, Université Paris Est - Créteil
Preferred Skills	Interest in formal languages and specification, curiosity for distributed computation, abstract and logic reasoning

Context. Process algebras (π-calculus, CCS, Ambiant calculus, etc.) are an abstraction of concurrent systems useful to study, specify and verify distributed behaviours. Those abstractions led to very concrete and practical results, such as the verification of security protocols [1] or the identification of security breaches using bisimilarity [2].

They generally value a compact presentation of their operators (e.g. parallel composition, non-determistic sum, recursion, etc.) and object of study (processes, contexts and relations) that enables and facilitates logical and inductive reasoning about the systems they model.

However, as recently argued by the advisors [3], they generally tend to re-introduce a granularity in their object of study "under the hood": while the definitions remain supposedly the same through the development of the study, many adjustments are performed by need or taste without always being clearly discussed. The advisors suggested a new conceptual frame by offering a finer division between what they call systems, processes and tests, that allows to recast previous results in a cleaner manner, and that draws new perspectives.

Goals. The student will engage in the next research steps with their advisors, that offer to embrace one of the two following applications of their initial discussion:

1. Featherweight Java [4] is an elegant semantics for the programming language Java where our division is present, but where contextual equivalences could be introduced and studied using our distinction between process (i.e., "snippet of codes"), completed processes (i.e., "standalone" methods) and systems (i.e., interaction between `main` methods and libraries). The goal would be to narrow the gap between idealized algebra and concrete programming language by developing a theory of observational equivalences in the former for the latter.
2. The theory of monitors [5] offers an interesting instanciation of this new framework, but could be diversified along multiple axis: removing the need to quantify over traces in its equivalences or replacing the monitors' silent transitions with a structural equivalence could refine this theory and illustrate its potential for a more usable theory of testing preorders.

Perks. Both Dr. Aubert and Dr. Varacca have an history of involving undergraduate, graduate and post-graduate students in their research, and can tune the level and nature of their engagement in the student's research based on their needs and tastes. Regular discussion with the second advisor will open up the possibility of future collaborations and of visit to Augusta University, Augusta, USA, or Créteil (near Paris), France. Even if the advisors have a precise strategy in mind, they remain open to suggested deviations from this program based on mutual interest.

References.

[1]   B. Blanchet, Modeling and verifying security protocols with the applied pi calculus and ProVerif, Foundations and Trends in Privacy and Security. 1 (2016) 1–135. https://doi.org/10.1561/3300000004.

[2]   I. Filimonov, R. Horne, S. Mauw, Z. Smith, Breaking unlinkability of the ICAO 9303 standard for e-passports using bisimilarity, in: K. Sako, S.A. Schneider, P.Y.A. Ryan (Eds.), Computer Security - ESORICS 2019 - 24th European Symposium on Research in Computer Security, Luxembourg, September 23-27, 2019, Proceedings, Part I, Springer, 2019: pp. 577–594. https://doi.org/10.1007/978-3-030-29959-0_28.

[3]   C. Aubert, D. Varacca, Processes, systems & tests: Defining contextual equivalences, in: J. Lange, A. Mavridou, L. Safina, A. Scalas (Eds.), Proceedings 14th Interaction and Concurrency Experience, Online, 18th June 2021, Open Publishing Association, 2021: pp. 1–21. https://doi.org/10.4204/EPTCS.347.1.

[4]   A. Igarashi, B.C. Pierce, P. Wadler, Featherweight Java: A minimal core calculus for Java and GJ, ACM Trans. Program. Lang. Syst. 23 (2001) 396–450. https://doi.org/10.1145/503502.503505.

[5]   A. Francalanza, A theory of monitors, Inf. Comput. 281 (2021) 104704. https://doi.org/10.1016/j.ic.2021.104704.