Wagner: Formal Models of Real-Time Digital Signal Processing

Internship proposal — 2018

Centre de recherche en informatique, MINES ParisTech

The MINES ParisTech Centre de recherche en informatique (CRI) offers an internship to explore formally verified models of digital signal processing. The starting point is WAGNER [1], a functional synchronous language (see for instance [2]) tailored to mechanical verification and formal study of efficient DSP models). WAGNER originated in the ANR FEEVER project as an operational model of the FAUST DSP language. WAGNER can also be seen as a more formal version of recent developments such as the MathWorks Audio System Toolbox.

We have developed a WAGNER compiler, written in OCaml and intended for real-world use, and a formal model of the language using Coq and the Mathematical Components Library, together with proofs about relevant properties such as linearity (in the Linear Time-Invariant, or LTI, sense). Internship goals are open and we encourage the students to get in touch to discuss about the project; we believe that this internship is a good opportunity to get knowledgeable about synchronous programming language semantics, abstract machines, and medium-to-advanced mathematics formalization in Coq. Some ideas for work are as follows:

Semantics WAGNER has a type system based on co-effects and a simple notion of type polarity, a step-indexed operational semantics, and an "efficient" abstract machine, which is related to the original semantics using a logical relation, whereas type-soundness is proven using realizability. There are many possibilities to modify the current framework here to better account for higher order (using better notions of polarity), to modify WAGNER's feedback operator to use a Nakano-style fixpoint, to relax causality, or to interpret programs as convergent sequences.

Mechanized Semantics The mechanized WAGNER model has undergone quite a few iterations. Possibilities of work here are to improve/complete the abstract-machine proof, develop a canonicity theorem for programs, or develop some LTI theory in Coq.

Type Inference and Compilation The current WAGNER compiler is capable of performing some basic type inference and code generation. Work topics here are to improve the type inference strategy by adding bidirectional type checking, evolving the current abstract machine implementation using continuation-passing and defunctionalization, or port parts of the industrial-level FAUST library to WAGNER.

Desirable Requisites: We recommend familiarity with Coq and/or OCaml, and with the basic notions of functional programming theory for the more theoretical-oriented parts; knowledge of DSP is a plus.

Compensation: MINES ParisTech internships may be funded, typically along the lines fixed by law (about 400 euros per month).

Contact: Emilio J. Gallego Arias(`e@x80.org`),
Pierre Jouvelot (`pierre.jouvelot@mines-paristech.fr`).

References

[1] E. J. Gallego Arias. *Experiments in Certified Digital Audio Processing*. 1h Seminar. `http://www.cri.ensmp.fr/people/gallego/slides/SpecFun-04-2016/`. 2016.

[2] A. Guatto. "A Synchronous Functional Language with Integer Clocks". PhD thesis. École Normale Supérieure Paris, 2016.