Tutorial proposal

Administrative matters

Names and affiliations of the organisers:

Sonia Marin
IT-University of Copenhagen
sonm@itu.dk
http://www.lix.polytechnique.fr/~smarin

Giselle Reis
Carnegie Mellon University in Qatar
giselle@cmu.edu
http://www.gisellereis.com

Iliano Cervesato
Carnegie Mellon University
iliano@cmu.edu
https://www.cs.cmu.edu/~iliano

Title: Formalising concurrent computation: CLF, Celf, and applications.

Proposed event duration: We request half a day for the tutorial, preferentially on Monday September 2nd, due to other teaching duties.

Format: We will run a short and accessible tutorial, that we hope will be attractive for the Tableaux/FroCoS community at large.

Estimate of the number of participants: Based on our own experience of attending Tableaux/FroCoS tutorials, we expect around 10 to 15 persons in the audience.

Past and similar events: There has previously been two related tutorials on *Mechanizing Metatheory with LF and Twelf*, one associated with PoPL 2009 and the other with CADE 2013. The tutorial proposed here will be similar in the introduction to the basics of the logical framework discipline, but will focus on the specifics of CLF and the Celf tool.

Topic and relevance to Tableaux/FroCoS

This tutorial aims at introducing the Tableaux/FroCoS community to the Concurrent Logical Framework (CLF) and the Celf tool, as well as some of their applications.

What is a logical framework and what is it useful for? A logical framework is a meta-language to specify and reason about deductive systems, as found in logic or programming language theory. The resulting specifications are executable and help reason about languages in a clear and simple way

What is CLF in particular? CLF is a logical framework for specifying and implementing concurrent systems. It is an extension of the LF framework with (*i*) *linear types* to support representation of stateful computation and (*ii*) a *monad* to support representation of concurrency. As a result, term equality in this language is not only defined modulo α-renaming but also modulo `let`-floating, which allows CLF signatures to be interpreted as concurrent

logic programs.

What is Celf? Celf is an implementation of CLF, as Twelf is for LF. It allows the user to specify a concurrent system and provides them with type checking, type reconstruction and proof search. However, it does not yet include the meta-theoretic capabilities that Twelf has; this is under current investigation.

What are the possible applications? We will illustrate the CLF approach and the usage of Celf in a step-by-step manner by going from a simple encoding of producer/consumer Petri nets to the more involved example of the concurrent λ-calculus, showing how the various features of the framework combine to allow clear formalisation of these concurrent systems.

Relevance to Tableaux/FroCoS: We hope for this tutorial to be accessible and relevant for both the Tableaux and the FroCoS communities. On the one hand, CLF is the result of a long line of work on theoretical aspects of formal reasoning systems. It combines concepts probably more familiar to the Tableaux attendants of dependent types, linear logic and modalities but in an effective way, to represent stateful and concurrent computation. On the other, it can be used as a tool in which formal systems can be specified in a clean and elegant way. The added linearity and concurrency aspects makes it useful for a set of formal systems, which require complex representations in other frameworks, hopefully making it appealing to the FroCoS audience