TESSLA—A Temporal Stream-based Specification Language

March 2, 2016

1 Introduction

Outline

1. purpose, motivation
   - online processing of data
   - monitoring of trace properties, specifically execution traces of programs
   - Functional reactive programming as related concept
2. What you describe with a TeSSLa specification
   - input, output streams
   - application of functions, composition of function
   -
3. Modelling data in terms of streams
   - timing model ($\mathbb{R}$,$\mathbb{N}$,$\mathbb{Q}$,...), restrictions to streams with discrete set of time stamps (event streams) or peace-wise constant streams (continuous stream)
   - continuous streams and event streams
4. Functions on streams and desired properties (in general)
   - small examples
   - causality, statefulness, time invariance
   - (composition lemmata)
5. TeSSLa syntax
   - base grammar with functions and type annotations
   - syntactical extensions: infix operators, named arguments, the "on"

6. Types
    - Generic types
    - Coercion
7. Functional semantics of operators, small examples
8. Larger example/case study
    - producer/consumer, ring buffer, ...