Goal-Directed Abstract Interpretation and Event-Driven Frameworks*

Bor-Yuh Evan Chang[1,2][0000−0002−1954−0774]

[1] University of Colorado Boulder, USA
[2] Amazon**, USA
evan.chang@colorado.edu

Abstract. Static analysis is typically about computing a global over-approximation of a program's behavior from its source code. But what if most of the program code is missing or unknown to the analyzer? What if even where the program starts is unknown? This fundamentally thorny situation arises when attempting to analyze interactive applications (apps) developed against modern, event-driven software frameworks.

Rich event-driven software frameworks enable software engineers to create complex applications on sophisticated computing platforms (e.g., smartphones with a broad range of sensors and rich interactivity) with relatively little code by simply implementing callbacks to respond to events. But developing apps against them is also notoriously difficult. To create apps that behave as expected, developers must follow the complex and opaque asynchronous programming protocols imposed by the framework. So what makes static analysis of apps hard is essentially what makes programming them hard: the specification of the programming protocol is unclear and the possible control flow between callbacks is largely unknown.

While the typical workaround to perform static analysis with an unknown framework implementation is to either assume it to be arbitrary or attempt to eagerly specify all possible callback control flow, this solution can be too pessimistic to prove properties of interest or too burdensome and tricky to get right. In this talk, I argue for a rethinking of how to analyze app code in the context of an unknown framework implementation. In particular, I present some benefits from taking a goal-directed or backward-from-error formulation to prove just the assertions of interest and from designing semantics, program logics, specification logics, and abstract domains to reason about the app-framework boundary in

* I would like to especially thank the following for making significant contributions to the research described in this talk: Ph.D. students Shawn Meier, Benno Stein, and Sam Blackshear; postdoc Sergio Mover; and collaborators Manu Sridharan and Gowtham Kaki. The University of Colorado Programming Languages and Verification (CUPLV) Group has offered the essential community with insightful discussions to conduct this work. This research was supported in part by NSF awards CCF-1055066, CCF-1619282, CCF-2008369 and DARPA award FA8750-14-2-0263.

** Bor-Yuh Evan Chang holds concurrent appointments at the University of Colorado Boulder and as an Amazon Scholar. This talk describes work performed at the University of Colorado Boulder and is not associated with Amazon.

a first-class manner. What follows are hopefully lines of work that make analyzing modern interactive applications more targeted, more compositional, and ultimately more trustworthy.

Keywords: goal-directed verification · backwards abstract interpretation · event-driven framework modeling