Introduction to the Special Issue on ESOP 2018

The 27th European Symposium on Programming (ESOP) took place from April 16 to 19, 2018, in Thessaloniki, Greece. ESOP is one of the European Joint Conferences on Theory and Practice of Software (ETAPS). It is devoted to fundamental issues in the specification, design, analysis, and implementation of programming languages and systems. Thirty-six papers were chosen to appear at the conference. The program committee selected an outstanding few and invited their authors to submit to this special issue of *TOPLAS*. This issue includes the four accepted papers, each of which demonstrates the quality and breadth of the conference. These four papers make significant contributions on a wide range of topics.

In “[Modular Product Programs](#),” Eilers, Müller, and Hitz present a novel technique for producing product programs that embed multiple executions into a single program. This turns the task of verifying a hyperproperty into verifying a standard trace property, enabling modular verification of hyperproperties using standard verification infrastructure.

In “[Behavioral Equivalence via Modalities for Algebraic Effects](#),” Simpson and Voorneveld show how to reason about equivalence of programs with algebraic effects via a new logic with modalities, which are predicates that characterize effect-specific observational properties of programs. They identify two key conditions these modalities must satisfy for soundness of the logic and show that these conditions hold for a number of algebraic effects.

In “[Consistent Subtyping for All](#),” Xie, Bi, Oliveira, and Schrijvers present a gradually typed calculus for predicative implicit higher-rank polymorphism that uses a new notion of consistent subtyping. Consistent subtyping is used in some gradual type systems to validate type conversions, and here the authors give a generalization of consistent subtyping that makes it suitable for polymorphism.

In “[Reasoning about a Machine with Local Capabilities: Provably Safe Stack and Return Pointer Management](#),” Skorstengaard, Devriese, and Birkedal present a secure calling convention for a capability-based machine similar to CHERI, using the machine’s local capabilities to ensure control-flow integrity in addition to data integrity of stack frames. To reason about the guarantees offered by capabilities, they set up a step-indexed Kripke logical relation for a simple assembly language with such capabilities and use it to verify control-flow correctness and local-state encapsulation.

I thank the authors and reviewers for all their effort in producing and reviewing these papers. I would also like to gratefully acknowledge the support of the *TOPLAS* editor-in-chief, Andrew Myers, the *TOPLAS* editorial staff, and the ESOP 2018 program committee for their help in selecting the invited papers.

Amal Ahmed
ESOP 2018 PC Chair
Northeastern University
Guest Editor