

Abstract of the Keynote: Partial Behavior Modeling

Marsha Chechik

Department of Computer Science

University of Toronto

chechik@cs.toronto.edu

Although software behavior modeling and analysis has been shown to be successful in uncovering subtle requirements and design errors, adoption by practitioners has been slow. One of the reasons for this is that traditional approaches to behavior models are required to be complete descriptions of the system behavior up to some level of abstraction, i.e., the transition system is assumed to completely describe the system behavior with respect to a fixed alphabet of actions. This completeness assumption is limiting in the context of software development process best practices which include iterative development, adoption of use-case and scenario-based techniques and viewpoint or stakeholder-based analysis; practices which require modeling and analysis in the presence of partial information about system behavior.

We believe that there is much to be gained by shifting the focus in software engineering from traditional behavior models to partial behavior models, i.e. operational descriptions that are capable of distinguishing known behavior (both required and proscribed) from unknown behavior that is yet to be elicited. Our overall aim is to develop the foundations, techniques and tools that will enable the automated construction of partial behavior models from multiple sources of partial specifications, the provision of early feedback through automated partial behavior model analysis, and the support for incremental, iterative elaboration of behavior models.

In this talk, I highlighted some of the results obtained in this line of research, joint work with Sebastian Uchitel and many students at the University of Toronto, University of Buenos Aires, and Imperial College London.