ABSTRACT

Mobile cloud computing, social media, cyberphysical systems, and the internet of things, are examples of increasingly important applications requiring scalable concurrency. The Actor model facilitates programming large-scale concurrent applications. Not surprisingly, Actor languages and frameworks have been widely adopted in industry to address scalability. Although this has significantly reduced programming errors, developing complex concurrent systems and reasoning about their properties can nevertheless be challenging and error prone. A key source of complexity is the interactions between actors. I will describe our work in programming languages defining new constructs to address this complexity. Specifically, we have developed ways of expressing multiparty session types and synchronization constraints which capture interactions. I will then briefly discuss three promising techniques to formally reason about Actor systems. First, inferring the concurrency structure of an actor program can facilitate targeted test generation. Second, predictive runtime verification can flag safety violations in future potential executions. Finally, statistical methods can improve confidence that a system obeys certain probabilistic properties. I will conclude by discussing open problems and promising research directions.

BIO

Dr. Gul Agha is Professor Emeritus of Computer Science at the University of Illinois at Urbana-Champaign and Interim CEO of Embedor Technologies. Dr. Agha is a Fellow of the ACM and of the IEEE, and a recipient the 2019 ACM Sigsoft Impact Paper Award. He has served as Editor-in-Chief of IEEE Concurrency: Parallel, Distributed and Mobile Computing, and of ACM Computing Surveys. Dr. Agha is best known for his formalization of the Actor model, used in languages and frameworks such as Erlang, Scala/Akka, and Orleans, and to develop scalable applications such as Twitter, LinkedIn, and Facebook Chat. Dr. Agha’s other contributions include development of Statistical Model Checking, which has applications in biological systems and cyberphysical systems among others: Concolic Testing for programs with dynamic memory and concurrency, a method incorporated in industrial software testing tools such as KLEE, Microsoft SAGE, and S2E; Euclidean model checking for reasoning about the evolution of probability distributions and for synthesizing controllers; the application of computational learning to program verification; logical methods for automated decentralized, predictive runtime verification; and distributed algorithms for wireless sensor networks (WSNs). Dr. Agha co-founded Embedor Technologies which is applying WSNs to continually monitor the structural health of bridges, buildings and large machinery. For example, Embedor’s technology was used to monitor the world largest Ferris wheel during construction, and will be used to continuously monitor it during operation.

Faculty Host: Nalini Venkatasubramanian