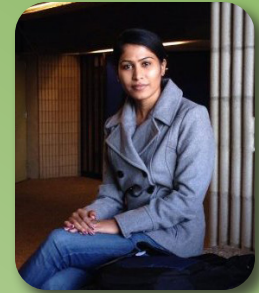


*The College of Computing is pleased to present a lecture by Computer Science faculty candidate*

# Vidhya Nagaraju

Friday, March 20, 2020

3:00 pm | Rekhi 214



## Software Reliability Engineering: Algorithms and Tools

**W**hile there are many software reliability models, there are relatively few tools to automatically apply these models. Moreover, these tools are over two decades old and are difficult or impossible to configure on modern operating systems, even with a virtual machine. To overcome this technology gap, Nagaraju is developing an open source software reliability tool for the software and system engineering community.

A key challenge posed by such a project is the stability of the underlying model fitting algorithms, which must ensure that the parameter estimates of a model are indeed those that best characterize the data. If such model fitting is not achieved, users who lack knowledge of the underlying mathematics may inadvertently use inaccurate predictions. This is potentially dangerous if the model underestimates important measures such as the number of faults remaining or overestimates the mean time to failure (MTTF).

To improve the robustness of the model fitting process, expectation conditional maximization (ECM) algorithms have been developed to compute the maximum likelihood estimates of nonhomogeneous Poisson process (NHPP) software reliability models.

Nagaraju 's talk will present an implicit ECM algorithm, which eliminates computationally intensive integration from the update rules of the ECM algorithm, thereby achieving a speedup of between 200 and 400 times that of explicit ECM algorithms. The enhanced performance and stability of these algorithms will ultimately benefit the software and system engineering communities that use the open source software reliability tool.

**Vidhyashree Nagaraju** is a Ph.D. candidate in the ECE department at the University of Massachusetts Dartmouth, where she received a M.S. in computer engineering in 2015.



Michigan Tech  
College of Computing