Preconditions, Postconditions, and Certifying Algorithms

Ross McConnell
Colorado State University

A certifying algorithm is one that produces a certificate with each output that proves that it has not been compromised by an implementation bug.

An example is an algorithm for recognizing interval graphs that either returns an interval model, proving that the given graph is an interval graph, or that points out an instance of one of the well-known forbidden subgraphs for the class. This sidesteps the problem of proving that the implementation is bug-free; it shows only that no bug has compromised the output for the given instance.

The talk will explore a novel type of algorithm that produces a certificate that one of the following has occurred:

1. A correct output was given;

2. A precondition on the input was violated.

It is not necessary that the certificate provide a simple or efficient means of checking which of these two events occurred. In other words, the certificate proves only a logical implication: the precondition is met implies the output is correct. The advantage of these is that they are often easier to come up with, and they can be chained together as steps of certifying algorithms that have no preconditions.

The technique holds promise in developing certifying algorithms for problems that have no obvious certificates of some possible outputs, such as recognition of circular-arc graphs.