Test Generation Using the Program Dependency Graph

Jose Sieira
February 16, 2010
Control Flow and Dependency Graphs

When used in conjunction with Data Dependency Graph, build test data to exercise selected predicates.

Fig. 1. A control flow graph and its control dependence subgraph.

Beats me to the punch. :(  

- Defines metrics:  
  - Ease of Forcing Execution of Component C  

"Which path to C is easier to take?"

Tool Support for Improving Test Coverage  

Susan Horwitz (horwitz@cs.wisc.edu)

University of Wisconsin and GrammaTech, Inc.
Must-Execute-Set Metric:
The number of untested components that must execute if predicate P executes and evaluates to v, and might not execute otherwise.

Improved-Ease-Set Metric:
The total amount by which the ease metrics of untested components are guaranteed to be lowered if predicate P executes and evaluates to v.
CodeSurfer by GrammaTech

- CodeSurfer generates a System Dependence Graph (AKA Program Dependence Graph)
- SDG data is accessible via an API.
- Academic version can only be requested by faculty.
- Horwitz proposes the use of CodeSurfer.

Further Work to be Done

- Horwitz tool has not been implemented so no real data exists to evaluate her contentions.
- "How well does the Ease-of-Execution metric predict the actual effort required to force a given predicate to evaluate to a given value?"
- "Can control and flow dependences help with the problem of automatic test input generation?"