Using a Goal-driven Approach to Generate Test Cases for GUIs

By Atif Memon, Martha E. Pollack, Mary Lou Soffa

This paper describes the use of planning, an AI technique, to automatically generate test cases for a GUI. "Given a set of operator, an initial state and a goal state, a planner produces a sequence of the operators that will change the initial state to the goal state." The paper argues that instead of trying to design test cases for every possible state of a GUI, it is more effective and simpler to specify the goals a user might want to accomplish and have a tool that can automate the sequence of events that would need to occur to meet those goals. The test case generation system developed by the authors, takes these goals as input in the form of initial and end states. Their system, using the Interference Progression Planner, then generates sequences of actions to get between the two states, which serve as the test cases for the GUI.

This is definitely one of the better ideas I have read so far on GUI test case generation. The only issue I would have with it is that it depends on the test case writer to setup all the different goal states, which may lead to some scenarios inadvertently being left out. Nevertheless, I think it is a powerful thing to be able to specify getting from one state of the GUI to another and having all the necessary tests cases generated for such a transition.

Bib reference:

```
@inproceedings{
 author = {Atif M. Memon and Martha E. Pollack and Mary Lou Soffa},
 title = {Using a goal-driven approach to generate test cases for GUIs},
 booktitle = {ICSE '99: Proceedings of the 21st international conference on Software engineering},
 year = {1999},
 isbn = {1-58113-074-0},
 pages = {257--266},
 location = {Los Angeles, California, United States},
 doi = {http://doi.acm.org/10.1145/302405.302632},
 publisher = {ACM},
 address = {New York, NY, USA}
}
```