Implementing and comparing impact algorithms

Szabolcs Faragó

As software systems are growing, in parallel the code analysis techniques are spreading as well. The aim of these techniques is to help the work of software developers or testers. As it is hard to understand complex systems, the size of the software usually affects the length of testing.

The aim of impact analysis is to determine how the methods of a program affect other methods. Impact set for a method, or a method set includes the methods affected by the given method/method set. Impact analysis has two modes:

- static impact analysis
- dynamic impact analysis

Both methods have advantages and disadvantages.

Static impact analysis algorithms return with much precise result because these analyze the whole source code. Because of this, the static algorithms are slower than other techniques.

Dynamic impact analysis is a new research area. It determines the impact set to one execution of the program. This method is faster than static methods, but it is not as precise as the old analysis techniques.

The impact algorithms have been used in more areas, mostly in help for testing, but during developing as well.

During my PhD work I have worked in a project team, where we have researched static and dynamic techniques, and we have tested the precision and the speed of analysis algorithms. We have written implementations for static and dynamic algorithms in Java language. Additionally, we have executed several tests on these implementations, and have compared the results. The results have proved that static algorithms are much precise than dynamic algorithms but slower too.

In my presentation I would like to introduce the algorithms, which we have implemented and tested, and I would also like to present the results of tests.