# Comparison on Static Slicing of C and Binary Programs

**Ákos Kiss**

Program slicing is a technique developed for automatically decomposing programs by analysing its control and data flow. Different slicing methods have been intensively studied in recent decades and many applications have been proposed, including maintenance, reverse engineering, testing and debugging. A (backward) slice consists of those parts of a program that potentially affect a set of variables at a specific program point, called the slicing criterion. Static slicing computes slices using static analysis only, without making any assumption regarding the input of the sliced program.

The slicing of programs written in a high-level language has been widely studied in the literature, but the slicing of binary executable programs got attention only in the near past [1]. Since the slicing of binaries is a relatively new topic no previously published work deals with comparing the slices computed from the source code form and the binary executable form of the same program.

In this paper we present our observations comparing slices of C programs and their binary counterparts using a publicly available C slicer and our prototype binary slicer.

## References

[1] Ákos Kiss, Judit Jász, Gábor Lehotai, and Tibor Gyimóthy. Interprocedural static slicing of binary executables. In *Proceedings of the Third IEEE International Workshop on Source Code Analysis and Manipulation (SCAM 2003)*, pages 118–127, September 2003.