Language Support for High-level Worst-case Execution Time Estimation

Áron Baráth and Zoltán Porkoláb

There are many ways to implement one specific task, but some of them are faster than the others. In general we can specify an acceptable complexity that we can validate based on the source code or the object file. However, the validation requires external tool, which cause undesired dependency from a third-party utility. A better solution when the compiler provides language support for checking the expected worst-case execution time for each functions.

Validating the expected worst-case execution time should be an aspect of the correctness of the program, because it guarantees high-level correctness at design time. Therefore, the complexity of program will not exceed an expected value. It is important to note that, our goal is to give an expected upper limit for the complexity at *design time*.

In this paper, we first define the method to estimate the worst-case execution time of a function. This is a structural algorithm, which analyze e.g. loops, function calls and assignments, and detect correlations between elements to provide as accurate estimation as possible. We introduce our experimental language, which can perform worst-case execution time analysis, and the programmer can annotate the function with the expected time-complexity. This is a C-like language with strict syntax, strong static type-system, an powerful compile-time checks. The compiler can validate the given value with the estimated worst-case execution time, and gives an error when the complexity exceeds the expected value. Furthermore, the accepted values for the time-complexity will be used at runtime in the dynamic program loader. It is used to ensure that, the proper function is loaded with the correct computed worst-case execution time value.

References

[1] Wilhelm, Reinhard, et al. "The worst-case execution-time problem – overview of methods and survey of tools." ACM Transactions on Embedded Computing Systems (TECS) 7.3 (2008): 36.

[2] Huynh, Bach Khoa, Lei Ju, and Abhik Roychoudhury. "Scope-aware data cache analysis for WCET estimation." Real-Time and Embedded Technology and Applications Symposium (RTAS), 2011 17th IEEE. IEEE, 2011.