Infinite loop detection

Reference No: B76069

CHALLENGE

The tremendous rise in **real-world implementations of cyber-physical systems,** for example in the context of the industrial internet of things (IIOT) and advanced driver-assistance systems (ADAS), is accompanied by an explosion in the number of sensors continuously monitoring the physical environment. These sensors are often implemented as embedded systems with some initial processing taking place at the sensor, leading to the name intelligent sensors. **The constant availability of these sensors is of the utmost importance.** Thus, infinite loop bugs are both an issue of safety and of security, as a **program that runs into an infinite loop becomes unresponsive,** a phenomenon exploited in a denial of service attack.

While the problem of finding infinite loops in programs is as old as computing itself, the potential consequences of not being able to perform in-line detection of infinite loops become increasingly dire, as the proliferation of cyber-physical systems increases. Furthermore, these embedded systems are often **strongly constraint in their computational complexity in order to limit fabrication cost and power consumption.**

INNOVATION

This invention presents a new algorithm for automated detection of infinite loops. It has **sufficiently low computational complexity to be applied continuously at runtime.** It does not need constraint solver, program source code or hash values over program states. The algorithm is based on an autocorrelation measure, commonly used in many areas of applied statistics, here calculated on a program execution's branch target address sequence (1). The **algorithm can be implemented in hardware or software.** Modification of existing source code is not required.

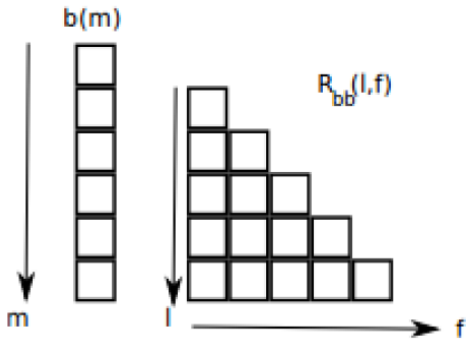
COMMERCIAL OPPORTUNITIES

The invented method can find application in all types of embedded systems, especially when heightened security and safety concerns are combined with constraints on the locally available computational resources. Some use cases include:

- IIOT, i.e. industry 4.0
- Autonomous vehicles

DEVELOPMENT STATUS

Functionality of the tool is evaluated with infinite loop bug test cases from the Juliet test suite for program analyzers. Applicability of the algorithm to production software is demonstrated by using the tool to detect previously known infinite loop bugs in cgit, Avahi and PHP.



$$R_{bb}(l,f,m) = \begin{cases} R_{bb}(l,f,m-1) & \text{for } f \not\equiv m \pmod{l} \\ R_{bb}(l,f,m-1)+1 & \text{for } f \equiv m \pmod{l} \\ & \quad \wedge\, b(m) = b(m-l) \\ 0 & \text{for } f \equiv m \pmod{l} \\ & \quad \wedge\, b(m) \neq b(m-l) \end{cases}$$

Figure: Triangular coefficient matrix (left). Correlation detects revisited branches with constant branchsequence length (right)

REFERENCE:

1. Ibing, A., J. Kirsch, and L. Panny. Autocorrelation-Based Detection of Infinite Loops at Runtime. In IEEE Int. Conf. Dependable, Autonomic and Secure Computing, August 2016

BayPAT

Technische Universität München

Technology from
TECHNISCHE UNIVERSITÄT MÜNCHEN

IP rights:
filed in 2016
US (pending)
DE (pending)

Contact:
Dr. Immo N. Söllner
+49 (0) 89 5480177-17
isoellner@baypat.de

Bayerische Patentallianz GmbH
Prinzregentenstr. 52
80538 München
www.baypat.de