

Smart bug-checking for software

September 14 2020



Thomas Neele. Credit: Eindhoven University of Technology

Computers and software are more important than ever. In systems such as cars, airplanes and medical devices, it is critical to implement software without major flaws, or 'bugs.' Eindhoven University of Technology Ph.D. candidate Thomas Neele developed three techniques for smarter and faster bug checking, based on the model checking

method.

Model checking is one of the most rigorous techniques to check [software](#). It looks at all possible things a software system can do, and the 'states' it can be in, to check if it works as required. The challenge is that software often consists of many parts that work in parallel. That can cause an explosion in the number of states that need to be investigated, making [model](#) checking costly and perhaps even unmanageable. Neele explored new ways to address this problem by reducing the number of states that need to be checked.

Keeping the goal of the software in mind

The three new reduction techniques Neele developed have one thing in common: they take the requirement of the states in mind. This extra piece of information means it is easier to see which states don't need checking. To be able to look at the system's behavior and requirement at the same time, Neele first developed a new, structured way to show the behavior-requirement combination.

The first technique reduces the number of states by grouping similar states together. This even makes it possible to work with an infinite number of states. The second approach checks whether the parts that work in parallel sometimes perform a task independently. In that case, it is not necessary to investigate all possible states. The last technique checks whether certain data elements are relevant and removes them if they're not.

Making software safer, faster

In the future, these ideas will help to reduce the development cost and time-to-market of safety-critical software. This can make high-tech

health care more affordable and thus more accessible to everyone. Furthermore, Neele's techniques can help eliminate dangerous bugs from systems such as airplanes. Currently, the application of Neele's ideas, and model checking in general, requires a lot of expertise. To enable widespread adoption, it's necessary to develop a fully-automated system that decides which approach is most suitable for a particular application.

More information: Reductions for Parity Games and Model Checking: research.tue.nl/files/160672035/20200916_Neele.pdf

Provided by Eindhoven University of Technology

Citation: Smart bug-checking for software (2020, September 14) retrieved 2 May 2024 from <https://techxplore.com/news/2020-09-smart-bug-checking-software.html>

This document is subject to copyright. Apart from any fair dealing for the purpose of private study or research, no part may be reproduced without the written permission. The content is provided for information purposes only.
