SpotSDC: An Information Visualization System To Analyze Silent Data Corruption

Zhimin Li
Scientific Computing and Imaging Institute
Salt Lake City, Utah
zhimin@sci.utah.edu

Harshitha Menon
Lawrence Livermore National Laboratory
Livermore, California
harshitha@llnl.gov

Yarden Livnat
Scientific Computing and Imaging Institute
Salt Lake City, Utah
yarden@sci.utah.edu

Kathryn Mohror
Lawrence Livermore National Laboratory
Livermore, California
mohror1@llnl.gov

Valerio Pascucci
Scientific Computing and Imaging Institute
Salt Lake City, Utah
pascucci@sci.utah.edu

ABSTRACT

The increasing complexity of high-performance computation makes the transient error more observable and the silent data corruption it causes threaten the computation's reliability. Understanding silent data corruption's impact on a program become a concern in computation community. In this study, we design SpotSDC a visualization system to analyze programs' resiliency to silent data corruption. SpotSDC enables users to compare a program's different code region's resiliency to transient errors and study how key variables' each bit in a program may influence the program's outcome. SpotSDC also provides a propagation visualization allows users to see how a transient error propagate through the program during the execution.

KEYWORDS

Fault Tolerance, Information Visualization, Transient Error

ACM Reference Format:
Zhimin Li, Harshitha Menon, Yarden Livnat, Kathryn Mohror, and Valerio Pascucci. 2018. SpotSDC: An Information Visualization System To Analyze Silent Data Corruption. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, Article 4, 2 pages. https://doi.org/10.475/123_4

1 INTRODUCTION

Large-scale computation and simulation are prevalent in different fields to help solve critical problems. However, the aggressive scaling in high-performance computation makes the computation process susceptible to the transient error result in an unreliable computation outcome. A transient error may silently corrupt the program's data and change program's output. The previous[] studies show that the transient error becomes more and more observable in modern computation system. How to develop more reliability program against it becomes a concern in the computation community.

During the program execution, a transient error may crash a program if it corrupts a pointer variable or vanishes during the program execution because of a program's resiliency property. The unfavorable cases are the error silently propagate through the program and change the output without warning. Studying the silent data corruption's impact on a program, and how a transient error propagates through it will help researchers develop a more robust program.

As far as we know, limited researches are working on using visualization techniques to analyze silent data corruption's impact. In this study, we design SpotSDC, a visualization system that helps to study the silent data corruption's impact on a program.

2 SPOTSDC SYSTEM

SpotSDC provides three visualizations, program tree visualization, propagation visualization and source code visualization. It loosely follows Schneiderman mantra:"Overview first, zoom, filter, details on demand" allow users to see how the transient errors' impact on the program and zoom into detail about how an error on program's specific region will spread during the execution.

2.1 Program Tree View

Program Tree View(Figure 1) provides a feasible interface that enables a user to study the impact of the transient error on a program's different level. For example, SpotSDC allows Users to see how transient error's impact on each function or each variable. As figure 1 shows, the visualization gives an overview of how frequently each line of the conjugate gradient program may be attacked. How each bit flipped occur in the different location will impact the program's outcome.

2.2 Propagation View

Propagation View (Figure 2) shows how a transient error will propagate through the program. A user enables to see the overview of program's different region's relative error change over time and use a visual lens technique to examine specific time region's error change.

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 123-4567-24-567/08/06.
https://doi.org/10.475/123_4

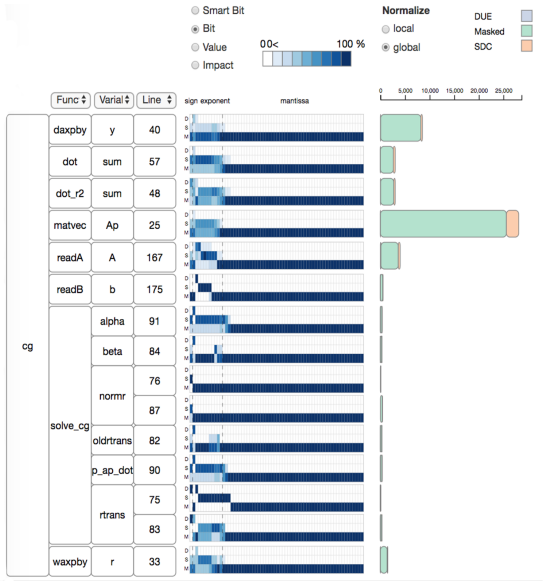

Figure 1: Program Tree Visualization

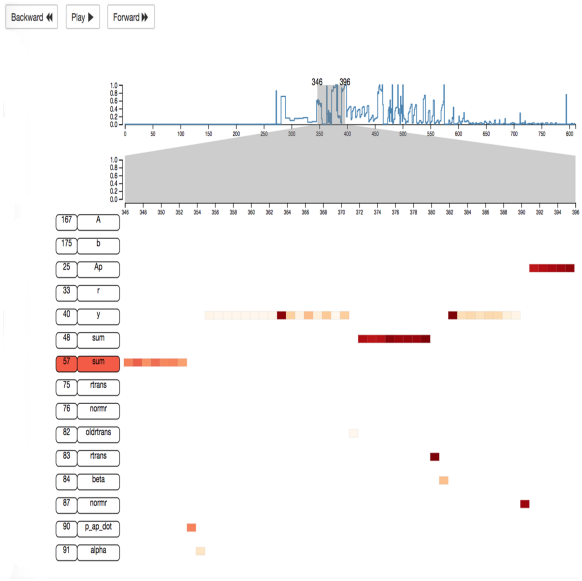

Figure 2: Error Propagation Visualization

2.3 Source Code View

Source code view (Figure 3) is on the side of program tree view or propagation view for users to reference some interesting features observed in the other two visualizations. An annotation visualization place on the top of the source code to show how SDC's impact in different iteration on this line or the SDC ratio[1] on this line.

3 CONCLUSIONS

SpotSDC is a visualization system to help users to understand the silent data corruption, which caused by transient error, impact

Figure 3: Source code view

on a program. It enables users to study different code region's resilient property by examining different input error scale or each bit impact on the program's outcome. It allows users to see how error propagates through the program's key variable and help users build better intuition about the resiliency property of a program.

ACKNOWLEDGMENTS

The authors would like to thank Shusen Liu, Dan Maljovec for their suggestion during the program development.This material is based upon work supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research under Award Number(s) DE- SC0014098, program manager Lucy Nowell. This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344 (LLNL-CONF-XXXX).

REFERENCES

[1] Harshitha Menon and Kathryn Mohror. 2018. D is CV ar: discovering critical variables using algorithmic differentiation for transient faults. In *Proceedings of the 23rd ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming*. ACM, 195–206.