The Nature of the Times to Flight Software Failure during Space Missions

Javier Alonso, Michael Grottke, Allen P. Nikora and Kishor S. Trivedi

Abstract: The growing complexity of mission-critical space mission software makes it prone to suffer failures during operations. The success of space missions depends on the ability of the systems to deal with software failures, or to avoid them in the first place. In order to develop more effective mitigation techniques, it is necessary to understand the nature of the failures and the underlying software faults.

Based on their characteristics, software faults can be classified into Bohrbugs, non-aging-related Mandelbugs, and aging-related bugs. Each type of fault requires different kinds of mitigation techniques. While Bohrbugs are usually easy to fix during development or testing, this is not the case for non-aging-related Mandelbugs and aging-related bugs due to their inherent complexity. Systems need mechanisms like software restart, software replication or software rejuvenation to deal with failures caused by these faults during the operational phase.

In a previous study, we classified space mission flight software faults into the three above-mentioned categories based on problems reported during operations. That study concentrated on the percentages of the bugs of each type and the variation of these percentages within and across different missions. This paper extends that work by exploring the nature of the times to software failure, due to Bohrbugs and non-aging-related Mandelbugs for eight JPL/NASA missions. We start by applying trend tests to the times to failure to check if there is any reliability growth (or decay) for each type of failure. For those times to failure sequences with no trend, we fit distributions to the data sets and carry out goodness-of-fit tests. The results will be used to guide the development of improved operational failure mitigation techniques, thereby increasing the reliability of space mission software.