National Aeronautics and Space Administration

Supporting NASA's Open-Source Diagnostics and Prognostics Software

Megan Lim
NASA Ames Research Center | UC Berkeley

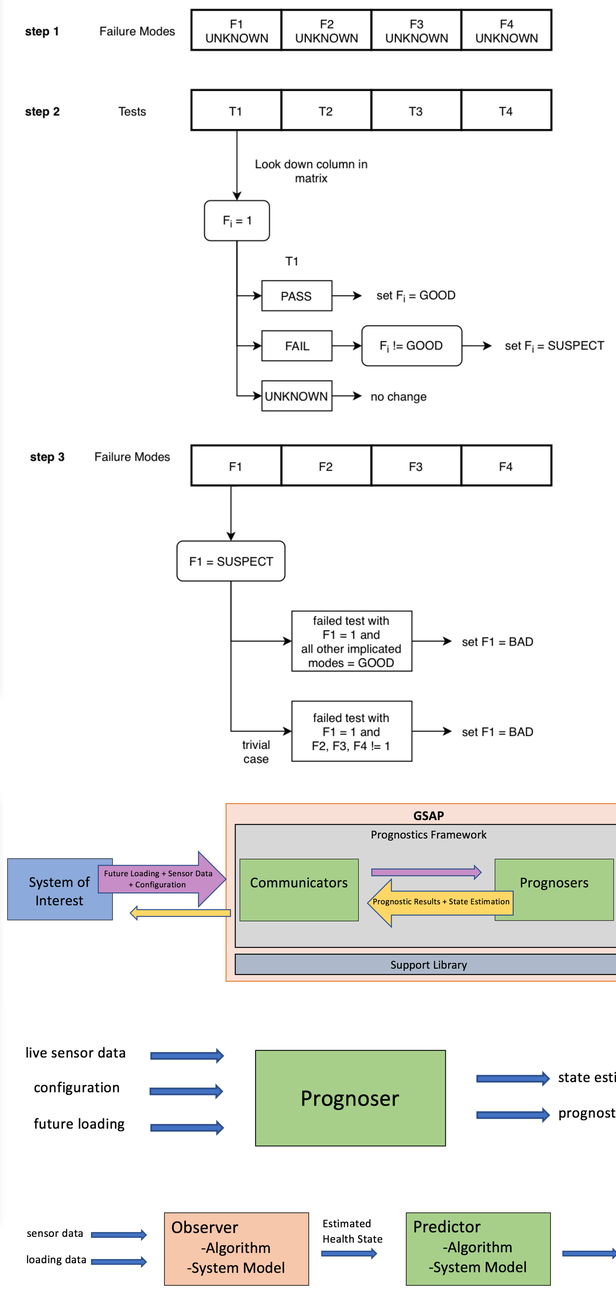
ABSTRACT

This summer, I worked with the Diagnostics and Prognostics Research Group, supporting their open-source software. Diagnostics and prognostics each contribute to the investigation of NASA system health management technologies. Essentially, Diagnostics serves to discover and identify the faults of a system, while Prognostics' function is to evaluate the evolution of these faults to determine this system's Remaining Useful Life (RUL). Algorithms have been and are currently being developed in each of these divisions. Metrics are then used to evaluate the performance and accuracy of these algorithms. I first worked on Diagnostic Reasoner (DR), which is software created by NASA to detect and identify the failure or failures that have occurred within a system. I did technical writing on the DR software user's manual, that serves to teach software users how to implement the DR algorithm. Completing this user's manual allowed DR to continue through the NASA software release process. I also developed and compiled an example system model in the C programming language to include in the DR software release. For Prognostics, I did technical writing on the Github wiki documentation for Generic Software Architecture for Prognostics (GSAP) and then the Prognostics Metrics Library.

GSAP WIKI DOCUMENTATION

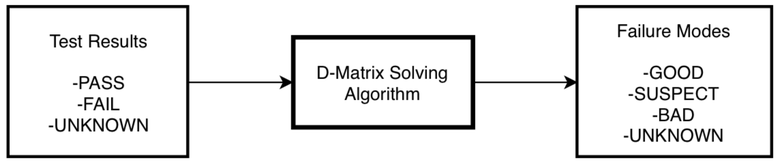
Generic Software Architecture for Prognostics (GSAP) is a framework that makes applying prognostics more efficient by implementing many of the common elements across prognostic applications. The standard interface of the GSAP framework enables adaptability of prognostic algorithms and models to systems of interest.

Example Model: Battery
Algorithms: Unscented Kalman Filter, Particle Filter, Monte Carlo Predictor



DIAGNOSTIC REASONER (DR)

For Diagnostics, I first worked on the Diagnostic Reasoner (DR) manual that served to teach software users how to implement the DR algorithm. The D-matrix, a part of DR, is utilized to store the relationships between tests and failures modes in a compact way. The algorithm then steps through the matrix table and assigns descriptive values to the failure modes. It's applicability to many systems allows it to diagnose varied technologies depending on interest.



PROGNOSTICS METRICS

After Diagnostics determines the failures associated with a particular system of interest, Prognostics then evaluates the evolution of these failures and Remaining Useful Life (RUL) of the system. The purpose of Metrics is to evaluate Prognostics performance. The overarching goal is to create standardized methodology for this performance evaluation.

Methods: Prognostic Horizon, α-λ Performance, Relative Accuracy, Convergence

ACKNOWLEDGEMENTS

I would like to thank Adam Sweet for his support and guidance as my mentor throughout this internship and for taking the time to explain what a Kalman Filter is. Many thanks to the Diagnostics and Prognostics Research Group for their help as well.