FAVPRO: NRC'S 21st CENTURY REACTOR PRESSURE VESSEL PROBABILISTIC FRACTURE ANALYSIS TOOL

Christopher Ulmer[1]*,Christopher Nellis[1], and Patrick Raynaud[1]

[1]*U.S. Nuclear Regulatory Commission, Washington, DC, USA*

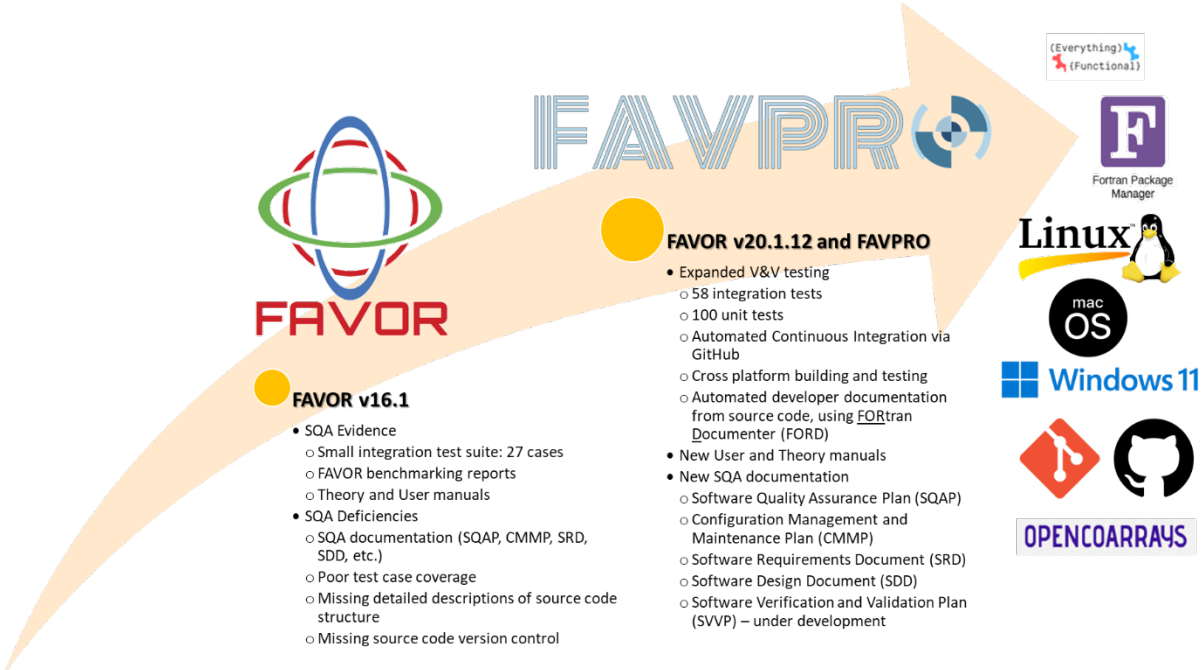
Abstract
Planned and unplanned transients in the operation of a nuclear reactor place stresses on the reactor pressure vessel (RPV) that affect its structural integrity. FAVPRO: Fracture Analysis of Vessels - Probabilistic, is a modern, parallel, object-oriented software tool that can provide high-confidence probabilistic assessments of reactor pressure vessel integrity for any population of flaws and any number of transients, with the goal of risk-informing technical and regulatory decision making. FAVPRO is the successor to the NRC's FAVOR code with higher solution speeds and new features like automated testing and documentation. The updated tool also includes new embrittlement models and other improvements to align with new fracture mechanics standards.

1. Introduction

In 2020, the U.S. NRC embarked on a project to modernize its legacy reactor pressure vessel integrity assessment tool FAVOR (Fracture Analysis of Vessels - Oak Ridge), version 16.1. This modernization effort consisted of two main tasks: the first task was refactoring the legacy Fortran 77/90/95 source code, and the second task was implementing state-of-practice software quality assurance (SQA) and verification and validation (V&V) practices and documentation.

2. Results

To create FAVPRO, the FAVOR code was completely rewritten and refactored, from legacy Fortran 77/90/95 to state-of-the-art Fortran 2018. Fortran 2018 provides all the functionality of a modern programming language, including object-orientation and parallelization, which were used to the extensively in the development of FAVPRO.



FAVPRO leverages Git and GitHub for version control and continuous integration. A large suite of unit and integration tests was developed and incorporated into the automated testing of the software, which has

been expanded from just Windows to also include Linux and macOS. FAVPRO also leverages a number of open-source libraries to handle unit testing, to build and test the code, to implement parallel execution, and to manage the input and output of the code via the JSON format. The automation of testing and resulting documentation provides a robust software testing record that enhances confidence in FAVPRO's results. In addition to FAVPRO's continuous integration testing, automatic code documentation is generated and available to the code developers via the FORD (Fortran Documentation) tool, providing up-to-date accurate information on the software structure to developers at all times. This developer documentation is accompanied by improved and updated User and Theory manuals, as well as a full suite of SQA documents that meets the intent of modern SQA standards.

In addition to enhanced performance and user-experience, FAVPRO brings new functionality to the analyst, such as new embrittlement models (including ASTM-E900), new stress-intensity factor solutions that follow the latest ASME standards, a new as-found-flaw population modeling option, and a new adaptive time step solver that provides higher solution speed in most cases.

The new FAVPRO code is a modular software tool whose modern source code represents a robust and resilient foundation that can be built upon to add new models, new probabilistic functionality, new materials, and new physical models to adapt to the rapidly evolving nuclear technology landscape.

3. Conclusions

FAVPRO is a 21st century probabilistic fracture mechanics code targeting applications in the RPV integrity arena. It leverages modern software development practices and modern features of the Fortran language to deliver highly parallel performance, state-of-practice I/O, enhanced software quality assurance and testing, and a highly adaptable and modular structure to allow for future expansion and adaptation to the changing nuclear environment.

Acknowledgements

Thanks to Damian Rouson, Brad Richardson, Karla Morris, Kate Rasmussen, and Giovanni Facco of Archaeologic Inc. for their expertise in modern Fortran and software archaeology. Thanks to Izaak Beekman of ParaTools for his expertise in parallel programming and MPI implementations. Thanks to Terry Dickson, Marvin Smith, and Andy Dyszel of Numark Associates Inc. for their expertise in RPV integrity, software QA, and the legacy FAVOR code.