Detection of Model Conflicts in the MYRRHA Reactor Design

<text>
The primary objective of this investigation is to characterize and quantify potential model conflicts within the MYRRHA Reactor Design. This assessment leverages a multi-faceted approach incorporating simulation, formal verification, and rigorous testing protocols. Specifically, the methodology involves establishing a baseline model representing the core functional parameters of the reactor's various subsystems. We then systematically introduce perturbations – modifications to specific model parameters – within a controlled environment. The analysis proceeds in stages, starting with a preliminary assessment of potential instability based on established modeling principles and preliminary simulations. Subsequent iterations involve deeper, more complex modeling utilizing probabilistic methods to estimate the impact of these perturbations on the overall system behavior. A crucial element of this process is the utilization of a "Conflict Detection Engine" (CDE) – a custom-built tool – designed to flag scenarios exhibiting statistically significant deviations from expected operational parameters. These flagged scenarios trigger a rigorous validation process. This validation process incorporates a series of stress tests, including scenario simulations mimicking real-world operational conditions, and component-level checks utilizing simulation-based verification. The CDE's output is then correlated with established conflict resolution strategies, aiming to propose mitigations and re-calibration of individual model parameters to eliminate identified conflicts. The effectiveness of the validation process is continuously monitored and adjusted based on the results obtained during the iterative testing phases. This ongoing monitoring ensures the model remains robust and accurately represents the reactor's operational constraints.

Background: MYRRHA Reactor Architecture

The MYRRHA Reactor Architecture utilizes a multi-layered design incorporating a core, containment system, and control mechanisms. The system is built upon a complex web of interconnected models, including the Reactor Control Model (RCM), the Thermal Management Model (TM), the Structural Integrity Model (SIG), and the Anomaly Detection Model (ADM). These models operate in conjunction, and potential conflicts could arise from inconsistencies or incompatible data. This system utilizes a 'simulation-driven' design and incorporates a robust feedback loop.

The Conflict Detection Engine (CDE) – a custom-built tool – is designed to flag scenarios exhibiting statistically significant deviations from expected operational parameters. These flagged scenarios trigger a rigorous validation process. This process incorporates a series of stress tests, including scenario simulations mimicking real-world operational conditions, and component-level checks utilizing simulation-based verification. The CDE's output is then correlated with established conflict resolution strategies, aiming to propose mitigations and re-calibration of individual model parameters to eliminate identified conflicts. The effectiveness of the validation process is continuously monitored and adjusted based on the results obtained during the iterative testing phases. This ongoing monitoring ensures the model remains robust and accurately represents the reactor’s operational constraints.

Conflict Identification Methodology

Conflict Identification Methodology

Our methodology utilizes a layered approach combining static analysis, dynamic simulations, and anomaly detection. Static analysis focuses on reviewing code and configuration files for potential inconsistencies – emphasizing data flow and input validation. Dynamic simulations, particularly utilizing the ‘Chaos Engine’, are employed to evaluate the system’s behavior under various scenarios and identify potential conflict points. Anomalous behavior identified by the ADM is flagged for further investigation. A ‘Conflict Score’ is calculated based on the severity of each identified conflict, prioritizing those with the highest potential impact.

We have developed a ‘Conflict Matrix’ that visually represents the interplay of each model and identifies critical areas requiring more scrutiny.

We’re incorporating version control and automated testing to minimize manual intervention.