STEVENS INSTITUTE OF TECHNOLOGY 1870

Ph.D. DISSERTATION DEFENSE

Candidate:	Chenhao Wei
Degree:	Doctor of Philosophy
School/Department:	Charles V. Schaefer, Jr. School of Engineering and Science / System Engineering
Date:	Dec 3nd 2025
Time/Location:	10 a.m. / Babbio 541A
Title:	Enforcing Test Case Quality from the AAA Perspective
Chairperson:	Dr. Lu Xiao, Department of System Engineering, School of Engineering & Sciences
Committee Members:	Dr. Xueqing Liu, Department of System Engineering, School of Engineering & Sciences Dr. Eman Alomar, Department of Computer Science, School of Engineering & Sciences Dr. Ying Wang, Department of System Engineering, School of Engineering & Sciences

ABSTRACT

The Arrange–Act–Assert (AAA) pattern is a widely recommended structure for unit tests and is believed to improve readability, maintainability, and debugging efficiency. Despite its popularity across educational, industrial, and open-source communities, several gaps remain. First, there is no large-scale, automated method for reliably identifying AAA structures in real-world test suites. Second, the prevalence, characteristics, and root causes of deviations from the AAA pattern have not been systematically studied in either open-source or industrial contexts. Third, existing tools for test quality analysis lack explicit AAA awareness and cannot support automated improvement of AAA-related design issues. This dissertation makes four contributions toward addressing these gaps.

The first contribution is an automated AAA tagging approach that mimics developer reasoning by combining semantic, syntactic, and contextual features. This tagging capability establishes the foundation for scalable and fine-grained analysis of AAA structures at the statement level, enabling systematic study of real-life test design.

The second contribution is the first comprehensive empirical investigation of AAA usage and deviations in open-source unit tests. By examining test structure, recurring AAA anti-patterns, and developer perspectives on addressing these issues, this study provides an evidence-based characterization of how AAA is practiced in the open-source ecosystem and why structural inconsistencies arise.

The third contribution extends AAA analysis to the industrial domain through a collaboration with Envestnet Inc. This study adapts the AAA tagging approach for domain-specific codebases and examines how AAA-related issues manifest in large-scale commercial systems. Through joint analysis and developer interviews, it reveals the practical constraints, trade-offs, and organizational factors that shape AAA practices in industry.

The fourth contribution is an automated AAA-aware detection and refactoring framework that leverages large language models to identify AAA issues and generate structurally consistent repairs. This framework demonstrates the feasibility of automated AAA improvement by integrating structural analysis, rule-guided reasoning, and behavior-preserving refactoring strategies.

Taken together, these four contributions provide the first end-to-end body of knowledge on AAA-based test design quality, spanning automated analysis, empirical understanding, industrial validation, and automated improvement. This dissertation establishes AAA as a rigorous and actionable lens for advancing the clarity, reliability, and maintainability of unit tests.