AI Usage Documentation

Conversation 1: Rankine Validation System

ME 401: Engineering Systems and Applications

Submitted by:
Caleb Hottes
calebhottes@u.boisestate.edu

Boise State University
Department of Mechanical and Biomedical Engineering

February 13, 2026

This memo documents AI-assisted development activity for the Rankine optimization validation workflow.

EXECUTIVE SUMMARY

This document records AI usage for the conversation where the Rankine optimization workflow was upgraded with automated physics validation, solver-constraint bug fixes, and report output integration. The AI assistant inspected the existing codebase, reproduced the specific SciPy constraint failure, generated a decision-complete implementation plan, then implemented and tested the requested changes. The resulting code added a reusable validation module, connected both optimization methods to validation checks, produced a console validation summary, and wrote a machine-readable JSON validation report. The assistant also corrected the differential evolution constraint API mismatch, retained plotting behavior, and verified compatibility with the existing command-line analysis runner.

1. PRIMARY PROMPT

1.1 User Prompt (Major Conversation 1)

I need you to buld a validation script for a rankine cycle optimization. The ranking validation file already exists with a comment in it. Please fill out this file so that it will perform a validation on a given optimization to verify that it obeys the laws of phisycics. The ranking optimization file is the main file. I want you to modify it so that when you run it it will automatically do each optimization method and then pass the results somehow to the vlaidation script which will create a validation report. Then make the ranking optimization script output both its outputs and the validation reports and retain the current plotting capabilities.

Finally there is another task. There is currently a bug with the constraints in the rankine optimization script, when its run it says the constraints are not recognized. I have band-aid fixed it by commening out the constriants on line 268 but I want you to fix this. Refer to the scipy docs as a starting point. Do not get sidetracked by any of the other possible ways to expand things that are hardcoded into the cod efiles

I am going to attach some additional files that My instructors have

`give me as context for this job. Do not write the report. the napkin engineering document is my draft of the final report that gives a lot of context in how this code is currently being used, but this report will change signifanctly.`

2. DESIRED OUTCOME

2.1 Requested Deliverables

- Fill and implement `rankine_validation.py`.
- Integrate validation into `rankine_optimization.py`.
- Fix differential evolution constraints error.
- Preserve existing plot outputs and workflow behavior.
- Generate a validation report artifact.

3. MESSAGE HISTORY AND STATUS UPDATES

3.1 Conversation Trace

- Assistant status update: file inspection started for optimization and validation modules.
- Assistant status update: identified likely SciPy constraints API mismatch between SLSQP and differential evolution.
- Assistant status update: verified installed SciPy signatures and reproduced the exact runtime error (`constraint of an unknown type is passed`).
- Assistant status update: requested decisions on report format and strictness of validation checks.
- Assistant provided full proposed plan block.
- User approved implementation.
- Assistant status update: implemented validation module functions and integrated optimization pipeline changes.
- Assistant status update: executed regression tests for optimization script and CLI runner.
- Assistant status update: verified JSON report creation and failure-path behavior.

- Assistant final summary delivered with modified files and test results.

3.2 Plan History (From Proposed Plan)

- Planned creation of `validate_rankine_solution()`, `print_validation_report()`, and `build_validation_report()`.
- Planned DE constraint fix using `NonlinearConstraint` while preserving SLSQP dict constraints.
- Planned correction of turbine efficiency typo in quality constraint.
- Planned end-of-run validation summary and JSON report output.
- Planned non-regression checks for plotting and CLI integration.

4. RESULTS AND DISCUSSION

4.1 Code Changes Achieved

- Implemented validation module with tolerance-based checks: energy balance, efficiency consistency, sign checks, efficiency bounds, superheat margin, quality margin, and pressure sanity.
- Added console `VALIDATION SUMMARY` and per-method detailed check output.
- Added JSON report generation to `rankine_validation_report.json`.
- Re-enabled and fixed differential evolution constraints using: `NonlinearConstraint(constraint_0, inf)` and `NonlinearConstraint(constraint_quality_turbine_exit, 0, inf)`.
- Preserved contour and sensitivity plotting behavior.

4.2 Verification and Debug Notes

- Constraint bug was reproduced first, then fixed and validated by rerunning optimization.
- CLI compatibility was verified with `run_rankine_analysis.py --mode optimize --no-plots`.
- Validation failure-path behavior was checked to confirm non-crashing output with explicit failure flags.

5. AI-ASSISTED ENGINEERING REFLECTION

5.1 What AI Did Well

- Rapidly isolated root cause of a solver API mismatch.
- Converted requirements into concrete interfaces and tests.
- Kept implementation scoped to requested behavior without unrelated expansion.

5.2 What Required Human Direction

- Validation strictness and report-format preferences.
- Acceptance boundaries (no extra architecture refactors).

6. CONCLUSION

The first major AI conversation produced the requested Rankine validation system, fixed the SciPy constraints bug, preserved plotting, and added reproducible validation reporting suitable for subsequent report-writing tasks.