Reflection Report for Inverted Pendulum Control Systems (IPCS)

Morteza Mirzaei

1 Changes in Response to Feedback

Feedback from reviewers, including domain experts, secondary reviewers, and the instructor, has been diligently incorporated into the documentation.

1.1 SRS

The Software Requirements Specification (SRS) has been meticulously updated based on reviewer feedback. The changes made are summarized below:

- Addition of theoretical underpinnings for the definition of underlying acceleration.
- Inclusion of derivations for two general definitions.
- Consolidation of two instance models into a single one.
- Enhancement of specificity and completeness in both functional and non-functional requirements.

For a detailed overview of these modifications, refer to the pull request [#29].

1.2 VnV Plan

The Verification and Validation (VnV) Plan has been updated to address reviewer comments effectively. Key changes include:

- Addition of supplementary tests for integration testing.
- Clarification on scenarios where a test should fail.
- Removal of test `test-vis-3`.

For a comprehensive view of these adjustments, consult the pull request [#30].

1.3 Design and Design Documentation

The design and accompanying documentation have been refined to align with reviewer recommendations. Notable enhancements comprise:

- Introduction of a main module to oversee the entire system.
- Creation of a Manager abstract class, from which GUI and File Output modules inherit.
- Rectification of input and output types in various modules.
- Consistent naming conventions across files.

To delve deeper into these changes, review the pull request [#34].

2 Design Iteration

The journey from the initial design to the final implementation witnessed significant evolution and refinement. Two major iterations stand out:

1. Addition of the constraint $\frac{m}{5} \leq m \leq M$: Although initially deemed unnecessary, this constraint proved crucial for the proper functioning of controller hyperparameters.
2. Refinement of the simulation formula: Initially neglecting torque and friction in the simulation, we adopted a formula discovered in a Matlab code for simulating the inverted pendulum. This decision facilitated easier implementation by translating the Matlab code into Python.