

CSCI 1951L Capstone Reflection

Code Review

Link to Repo:

Once the assumptions about how to use the existing AMM code's helper functions to access the eth and token reserves were made clear, the coding portion of the assignment became straightforward. For the forward direction the eth reserve was used to substitute for X, the primary trade asset. For the reverse direction, the token reserve was used to substitute for the primary trade asset. Because of the simplifications UNISWAP allowed us to make (working with a constant product AMM where $k = 1$), the formulas for divergence loss, linear and angular slippage, and load were straightforward and simple to implement, not requiring any additional helper functions or code. In order to implement the view function for angular slippage, reliance upon the derivation of equation 9 was crucial. Borrowing the `atanSmall` helper function from Macroverse's math util file, implementing angular slippage became a difference between two arctangent functions. Aside from this nontrivial view function, the others relied on sequences of chained multiplication, addition, and division between the eth/tokens sold, and the respective reserve (either eth or token).

Project Difficulties

The main difficulties that had to be dealt with during the course of this capstone are related to solidity as a language and understanding the existing AMM repository. Once solidity was introduced and enough time was spent understanding how an AMM should work, and specifically how UNISWAP is implemented, the project became much less difficult.

Project Ambiguities

Initial attempts at starting the capstone were difficult, as attempting to understand AMM theory prior to encountering it in the course was not feasible. While the general premise of what an AMM is, what its purpose is, and what the use cases for an AMM are, were straightforward and easy to understand, delving deeper into exactly how an AMM is built required the capstone's timeline to be synched with the course material.

Project Highlights

Professor Herlihy's eagerness to meet and discuss the project's expectations, outlining very clearly what the deliverable should include, as well as making it evident what the final submission should entail made it such that this project was not stressful. In effect, it allowed me to spend more time grappling with the derivations, motivations, and theory behind the view functions, making it a more rewarding project.

Project Suggestions and Conclusion:

At first, I was under the impression the capstone project would require implementing the entire AMM. This would be a feasible project if the capstone were introduced at the beginning of the class and there was TA support for allowing students to work on the capstone in concurrence with the regular course material. However, I feel this may not be the most optimal suggestion, since students that have not had exposure to either Golang or Solidity may be overwhelmed. If the goal of the capstone project is to keep it as such, where students implement these 8 view functions for trades, then it would be worth adding an additional requirement where students do a paper reading, and write responses to the paper, in a sort of collaborative lab session. Then the students could be tasked with giving a sort of analysis, similar to how the whitepaper is formatted, that should include demonstration of knowledge regarding the derivations of the view function formulas, so that students do not merely read the end of the paper, implement the simple functions, and fail to interact with the content in a significant manner. Overall, this project's scope is very feasible, and increasing the scope may or may not make the project too overwhelming, however, I feel the most

crucial aspects of this capstone lie in being able to draw upon all of the course material, read and understand a new system, implemented in an industry specific, niche language, and produce meaningful additions, all the while understanding exactly why the view function formulas work as they have been proposed. This last bit is most crucial, and further additions to the capstone requirements should focus on engaging the students more with the paper and the fundamental theories, as opposed to the coding element.