Algorithmic work completed

- After switching to the scenario generation project, I started exploring the use of LLMs for generating simulation scenarios. I started by exploring the Scenic language, which focuses on generating scenarios for autonomous driving. The language has a complex grammar and syntax which is compiled to Python code. It has support for various simulators, the most comprehensive of which is CARLA. I was able to get simple scenarios generated by LLMs to run in CARLA, but more complex scenarios (i.e. overtaking maneuvers) were unsuccessful. Currently, the language only supports generating actors and behaviors to be placed on 2-dimensional maps, which might limit its applicability to other domains or more complex autonomous driving scenarios. For example, the language does not support generating the roadmaps for vehicles and pedestrians (i.e. OpenDRIVE files) for simulators like CARLA.
- I envision scenario generation as a two-part problem: generating the world/map and static objects for the environment, and generating the dynamic objects and their behaviors for a given world. I think a general, LLM-based architecture for scenario generation is possible by targeting the world and actor generation in composable stages. Figure 1 illustrates my proposed architecture. Here, a structured, text-based, representation of a scenario can be obtained from an LLM by prompting it with a natural language description and a schema to structure the world and actors. Then, parsers can be implemented to convert the structured representation into specific formats required by different simulators. Moreover, the world definition can be extracted separately from the actors, allowing for more flexibility and adaptability to different domains by simplifying future scenario generation processes, as shown in Figure 2.

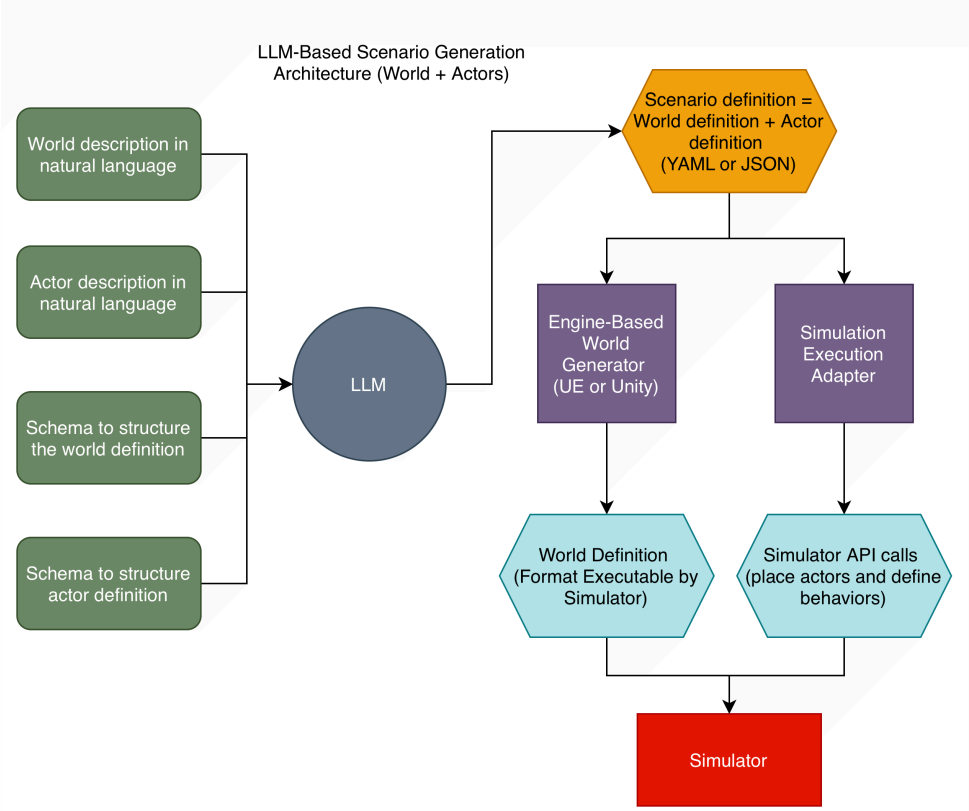

Figure 1: My proposed architecture for LLM-based scenario generation (world + actors)

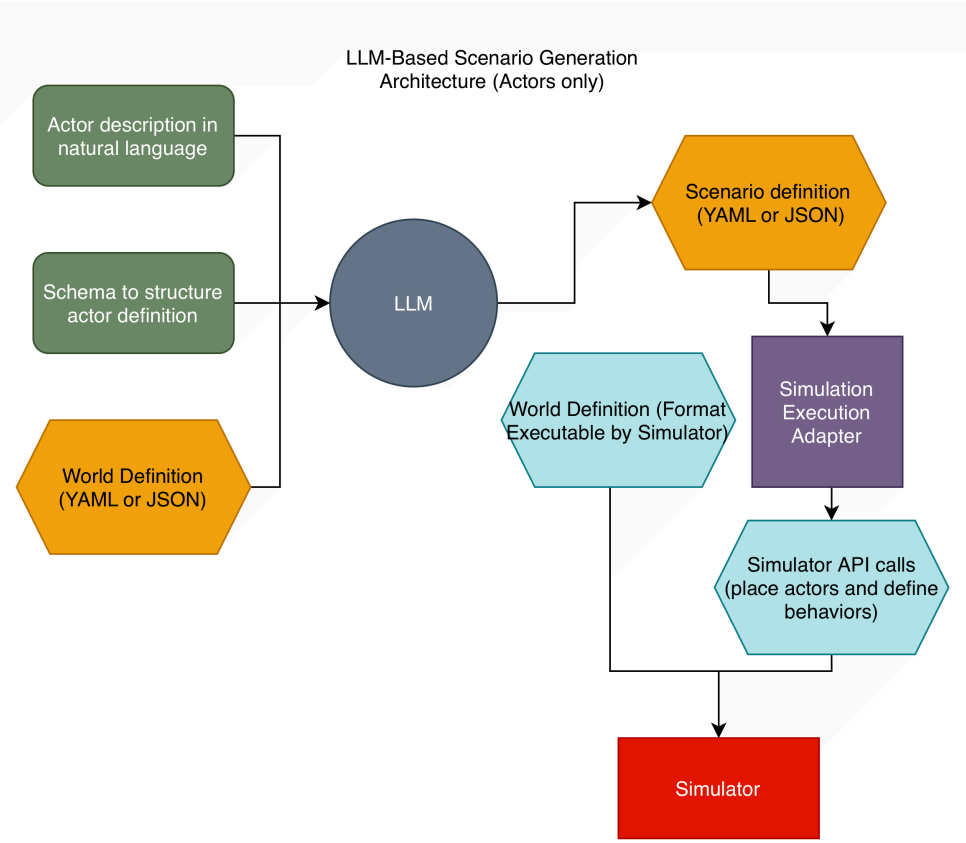

Figure 2: My proposed architecture for LLM-based scenario generation (actors only)

System work completed

- My systems work for the scenario generation project throughout this rotation consists of: a survey over recent scenario generation methods, including my initial effort at developing a schema for generating flight scenarios, desigining my proposed scenario generation architecture (described in the previous section), and creating LLM-generated scenarios using Scenic and rendering them in CARLA. The scenic code can be found in this repository's "scenic" directory, but the full virtual environment is needed to run the code.
- I also had some initial work for the CPS attack platform project, which involved trying to setup a MATLAB/Simulink environment for simulating EMI attacks on drones. This project (found in the "cps-attack-platform" directory) includes my efforts at integrating a drone model from Simulink with an IMU sensor model for eventually simulating EMI signal injection attacks on the drone. After refocusing from the cps-attack-platform project to trying to assess the practical feasibility of RF attacks on CPS, I created a survey over recent works in EMI signal injection attacks on CPS and tried to identify the physics behind each attack. I also attempted to identify a possible RF attack that could be quick to replicate with a HackRF module, but was unsuccessful.

How can the group improve to make CSPL more enjoyable while maintaining productivity?

- I think I could have benefitted from more frequent or detailed discussions to reduce ambiguity about my research directions. For example, My initial rotation project plan was to build a CPS attack simulation platform using MATLAB/Simulink. At first, I thought this idea and the implementation plan was well-received. However, after spending a week trying to implement the

plan, we had a meeting and decided that simulation would be insufficient for the project, prompting me to pivot to a different idea.

Summary of My Rotation

Week 1 (08/25-08/31)

- Identified possible direction for rotation project (adversarial robustness of CPS) and presented on an initial paper in the area

Week 2 (09/01-09/07)

- Further work on literature review over CPS sensor attacks and identifying a concrete rotatino project.

Week 3 (09/08-09/14)

- Discussed with Shanghao about a possible project for the rotation. We came to trying to simulate IMU/GPS spoofing attacks on drones using MATLAB/Simulink
- Developed implementation plan for building a CPS attack simulation platform using MATLAB/Simulink
- Started trying to setup initial components of the platform by integrating a drone model from Simulink

Week 4 (09/15-09/21)

- Continued working on building a CPS attack simulation platform using MATLAB/Simulink
- Started trying to setup an IMU sensor model and integrating it with a drone in Simulink for simulating EMI signal injection attacks on CPS
- Pivoted away from designing the MATLAB/Simulink platform to focus on assessing practical feasibility of RF attacks on CPS

Week 5 (09/22-09/28)

- Surveyed recent works in EMI signal injection attacks on CPS and tried to identify the physics behind each attack (RF attack survey)
- Attempted to idenfiy a possible RF attack that could be quick to replicate with a HackRF module.

Week 6 (09/29-10/05)

- Pivoted away from analyzing practical feasibility of RF attacks on CPS to exploring use of LLMs for scenario generation in autonomous driving
- Cloned Scenic repository and review documentation
- Looked at possibility of fine-tuning LLMs on Scenic code generation. Deemed unlikely to be successful due to limited amount of training data.
- Got LLM agent to create simple driving scenarios using Scenic
- Setup CARLA simulator to render scenarios generated by LLMs using Scenic

Week 7 (10/06-10/12)

- Rendered simple driving scenarios in CARLA using Scenic code generated by LLMs
- Attempted to get LLMs to generate more complex scenarios (i.e. overtaking maneuvers) but was unsuccessful

- Investigated possibility of using LLMs to generate the roadmaps for vehicles and pedestrians (i.e. OpenDRIVE files) for simulators like CARLA - deemed to be too time consuming for little gain
- Investigated recent works in LLM-based scenario generation for autonomous driving (LLM scenario generation survey)
- Met with VBS4 representatives to discuss applicability of their simulator/SDK for combat-focused adversarial sticker and scenario generation

Week 8 (10/13-10/17)

- Reviewed code from similar works and started identifying methods to adapt these ideas to other problem domains (i.e. flight, robotics, combat)
- Proposed Architectures for LLM-Based Scenario Generation outside of autonomous driving domain
- Created this wrap-up document