Statement of Work

NASA: Optimization to Surface planning and Scheduling

Sponsor Mentor: Robert Morris
Student Team Members:

- Ritwik Rajendra
- Hsiang Yu (Heron) Yang
- Anshu Rajendra

Faculty Advisor: Corina Pasearanu

Code and Documentation: https://github.com/heronyang/airport-simulation (Private repo, request for permission)

Overview

The overall objective of the practicum is to demonstrate the use of probabilistic modeling and analysis tools to develop and implement scalable control and optimization algorithms to improve surface operations at large airports. This work builds upon and expands existing research being conducted at Ames Research Center in the following areas:

- The use of autonomous vehicles for taxiing aircraft and other logistics problems;
- The use of probabilistic programming to model complex data;
- The application of planning and scheduling technologies to airport surface logistics; and
- The development of future architectures for ground traffic monitoring and control at NASA Ames Research Center.

Objectives

- To create an airport simulation tool that works for SFO airport
- To add uncertainties into simulation and to analysis the performance of different scheduling algorithms
- To explore different auto-scheduling methods and to compare with current FCFS method.

Motivation

Congestion at key airports as the area in which the traffic capacity problem is most prominent. Capacity gain requires construction of new runways and expansion of surface area for taxiing. Capacity gain means more complexity in Air Traffic Control (ATC) operations, increasing risk of human error, operator workload, and the potential for inefficiencies in operations. Capacity gain also means potentially harmful effects on the environment, such as more noise and air pollution, and higher fuel costs for taxiing aircraft. The practical difficulties of increasing capacity through airport expansion introduce the desire for enhanced airport ground movement efficiency by the intelligent use of the existing resources. The problem to be solved is a logistics problem: the coordinated movement of humans and machines to accomplish a complex task: getting arriving aircraft to gates and departing aircraft in the air. Safety and efficiency in operations are the primary goals. Efficiency means smart planning and scheduling and also being able to use data to make smart predictions.

Scope of work

The research team will be deployed to work on three areas:

- Using probabilistic programming to develop realistic simulations of airport surface activity from operational data supplied by NASA;
- Applying the predictive capabilities of the probabilistic models to improve algorithms for surface movement planning and scheduling; and
- Conduct experiments in simulation using a system developed at NASA Ames to demonstrate the value of real-world operational data in the development of optimization algorithms to solve the surface logistics problem.

Period of Performance

Reading research papers on the Problem statement

Status	**Achieved**
Objective	To read 3-5 papers in the airport scheduling problem domain and understand the various components of airport scheduling system
Activity dates	*Planned:* 08/31-09/10 *Actual:* 08/31-09/10
Progress	This activity started in the first week of semester and was finished successfully. Each of us read 2 papers in the domain and gained insight into the various parts of an airport scheduling system.
Outputs created	We created a wiki page with questions related to all the papers we read and had detailed discussions around the current state-of-the-art approaches.

Understanding the current architecture - ASSET simulator and player

Status	**Achieved**
Objective	To understand the architecture of current ASSET simulator and player that we have been provided and how it works.
Activity dates	*Planned:* 09/10-09/20 *Actual:* 09/10-09/20
Progress	We took this time to understand the current architecture of the ASSET simulator and player that has been applied to solve the problem by our stakeholders. Again we documented our questions about the current design and had discussions with our stakeholders.
Outputs created	We were able to reproduce the current working system and functionality on our own system and documented and clarified questions around the current design.

Create potential subtasks and identify area of improvements

Status	**Achieved**
Objective	To identify area of improvements in the current architecture and come up with tangible goals for the project
Activity dates	*Planned:* 09/15-09/25 *Actual:* 09/15-09/25
Progress	After looking at the current architecture and with preliminary background investigation, we narrowed down issues with the design and what might need improvement and possible approaches for the same.
Outputs created	We were able to segregate the project goals into three segregated components which could be worked upon independently and concurrently and documented the same on our Project wiki page.

Start with basic implementation on the project goals

Status	**In progress**
Objective	To start with first draft implementation on the decided goals
Activity dates	*Planned:* 09/25 - Ongoing *Actual:* Ongoing
Progress	Each of us have individual subtasks for the project and as a starting point, we want to implement basic functionality for the target goals which we can improve upon iteratively.
Outputs created	An asset player basic functionality has been implemented and is now on the team github. We also discussed about another goal in detail in a recent meeting to identify sub-goals.

Deliverable Schedule

We meet regularly on a weekly basis with our mentors and faculty advisor to discuss our progress and also iterate over the architectural design decisions. We maintain an active meeting minutes on Github wiki updating it with relevant details/updates about our project. Below is the detailed project plan we have come up in agreement with our mentors :

Indicator	**Dates**	**Status**
Identification of project goals	08/31-09/10	**Achieved**
Reproducing current working simulation on local environment	09/10-09/20	**Achieved**

Review current literature	09/15-09/25	**Achieved**
Review current architecture	09/15-09/25	**Achieved**
Clean current code	09/25 -10/10	**In Progress**
Develop first draft for identified goals	09/30-10/20	**In Progress**
Analyze and iteratively improve	10/20-11/20	**Not started**
Experiment/Test	11/15-11/25	**Not started**
Visualize Experiment Results	11/25-12/07	**Not started**

Methodology

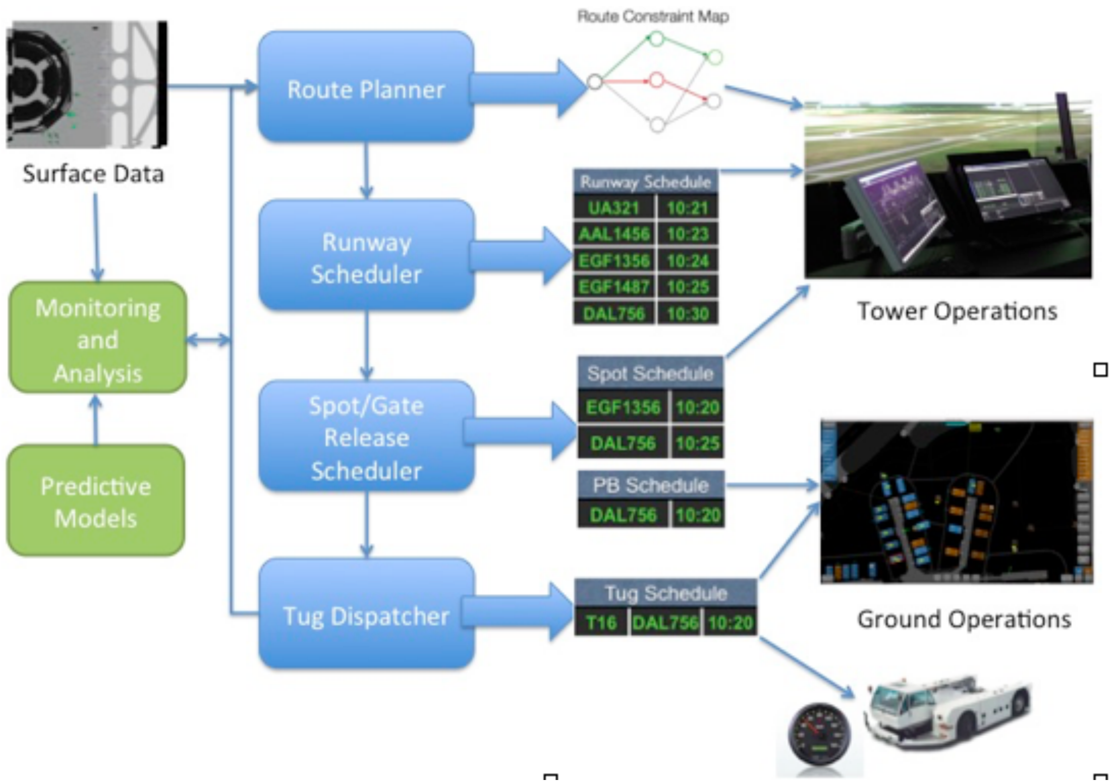
Planning and scheduling

In terms of planning and scheduling, the focus will be on two areas:

- Applying recent advances in Multi-agent Path Finding (MAPF) algorithms to generate optimal routes for autonomous towing vehicles; and
- Enhancing predictive surface planning models by integrating the probabilistic models that incorporate uncertainty.

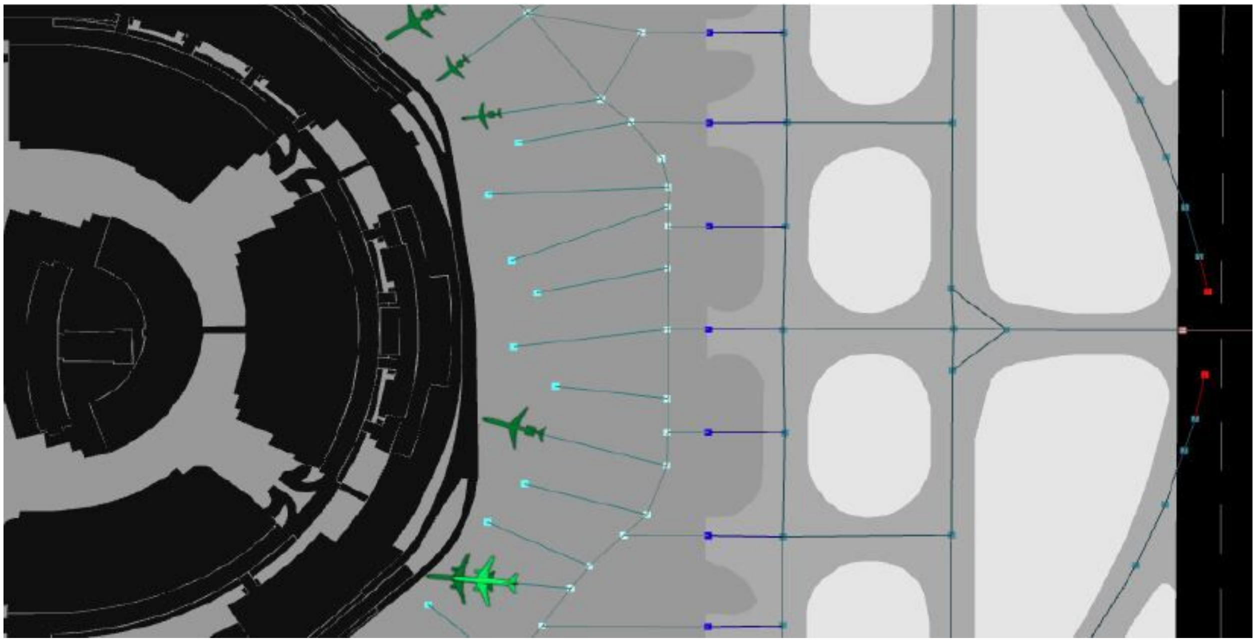
Surface movement planning starts by generating routes for arriving and departing aircraft over a certain planning horizon. Here MAPF algorithms will be applied with guidance provided by surveillance data and predictive models. Routes are then refined through a process of assigning times for push-backs, spot release, and runway queue management, as well as tug assignment. we will investigate different ways in which decisions made by these modules will benefit by querying the analysis tool to make decisions. To collect statistics related to these metrics, we will utilize a fast time Python-based simulator called ASSET (Airport Surface Simulator and Evaluation Tool) developed at NASA Ames. ASSET is based on the SARDA framework for scheduling, but with reduced capabilities that allows for rapid prototyping of route planning and scheduling algorithms.

Simulation and Visualization



To collect statistics related to these metrics, we utilize the ASSET tool mentioned above. ASSET contains three components: a scheduler, a simulator, and visualizer and analysis tools. The inputs to the simulator include a graphical model of an airport; a model of aircraft (including wing span, length and average taxi speed); and a scenario, a list of departure and arrivals for different aircraft, and the times at which they enter the surface system. The simulator, in conjunction with the scheduler, outputs the surface track information (i.e. the flow of traffic) over time. The simulator also models the 'intent' of the towing vehicles by automatically enforcing the separation constraints and other rules governing safe surface traverse.

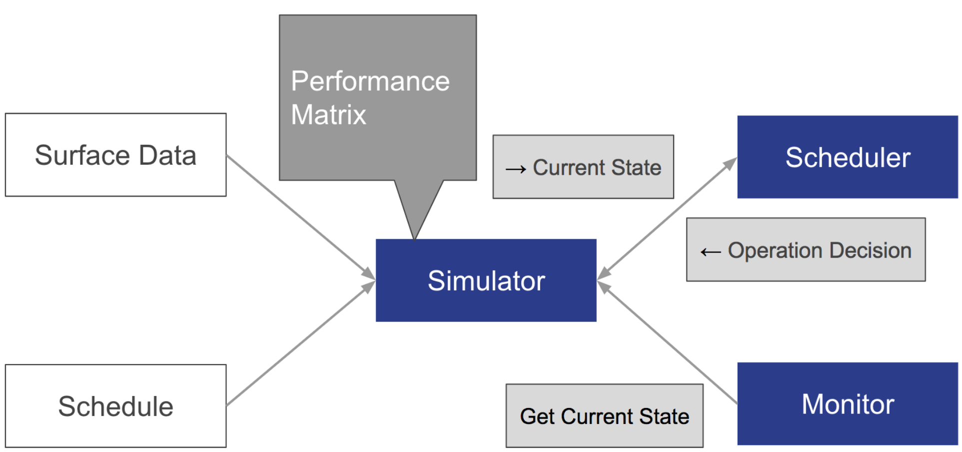
The ASSET visualizer reads simulator output and displays the progress of the scenario on the airport surface. The evaluation tool reads the simulator output into an SQL database, from which statistical inferences can be made and plotted, relevant to the four metrics listed above.



In addition, we will build a separate simulation for San Francisco Airport asides from ASSET, which will then be used for evaluating the updated schedulers and uncertainty experiments. The goal of building another simulation is

- to design an airport simulation works for multiple airports;
- to works with an external scheduler in runtime, and be able to show performances, and
- to be extensible for adding new constrains, features in the future.

All components of the SFO simulation will be written in Python3 where the graphical interface will apply PyQt5. Therefore, it's easy to maintain in the future as it only uses one programming language and all parts locate in the same repo. The surface data will be retrieved and modeled from Google Map and OpenStreetMap.



Uncertainty

Uncertainty with airport surface operations comes from three sources: human decision-making, mechanical problems, and changes to the environment. A passenger on a departing flight might take too long finding an overhead bin for his bag. This might delay the pushback time for his flight. Mechanical problems to the aircraft or towing vehicles might delay or cancel a flight. Changes to weather conditions (an afternoon summer thunderstorm) typically delay all incoming and outgoing flights.

One way of managing the uncertainty by anticipating all future states of the world. At the other extreme of perfect prediction is a purely reactive approach, where you can't anticipate anything about the future. This requires continuously examining the global state of the world (airport surface), comparing it with the current plan, and updating the plan on the fly. Specifically, to check at each time frame whether the world conforms to the current plan and revise the plan accordingly.

These extremes illustrate that there are two ways of dealing with uncertainty: during planning and during operations. The first can be called model-based or proactive planning; the second can be called sensor-based (model-free) and reactive. Between these extremes there will be hybrid approaches that combine models and sensing. This is the most reasonable approach, because models will always be approximations of the behavior of complex logistic systems.

Risk management

To avoid the risk of dependency of different modules on each other, we plan to carry out the work on path planning and scheduling using the existing DFW airport model, while the SFO model is being developed concurrently. Likewise, the work on modeling and uncertainty will be done independently from the other two modules. As a result, if any specific module does not produce satisfactory results, it will not have a negative impact on the others, and we will still have results delivered from the successful modules. We also plan to perform error analysis in case any of our work does not yield positive output, so that we can determine the root cause and make recommendations for future work.

Proposed Budget

We do not anticipate any expenditure for the project as all the resources being used are open source.

Acceptance of Deliverables/Evaluation criteria for project success

The goal of the proposed practicum is to demonstrate the use of probabilistic modeling and analysis tools to develop and implement scalable control and optimization algorithms to improve surface operations at large airports. The practicum team will use NASA operational data for Dallas Fort Worth and Charlotte airports to build predictive models that will form the basis of a predictive analysis tool. In parallel, the team will expand the planning capabilities of the ASSET scheduler by implementing the path finding algorithm in Python for use in route planning.

In evaluating the overall system, four performance metrics will be used:

- Efficiency, in the form of maximizing throughput;
- Reducing complexity of operations, primarily in the form of workload for flight crew, tower personnel or ground crew;
- Safety, specifically the ability to maintain separation constraints and avoiding potentially dangerous events such as runway incursions; and
- Environmental and economic benefits through reduced fuel emissions and reduced maintenance costs through less wear on airplane engines.

Using the ASSET simulator, the overall outcome of the project is the ability to measure the improvement in system performance with and without the use of probabilistic analysis. The deliverables for this project will be the enhanced ASSET system and a detailed NASA technical report summarizing the overall effort.