Welcome to the WSC 2025 Simulation Challenge! This year, our focus is on port operations, presenting a simplified yet authentic representation of real-world port activities. This material is designed to help participants thoroughly understand the system, and grasp the competition requirements in a clearer way.

The Origin of the Simulation Challenge

Since its inception in 2022 as the "Case Study Competition," **the Winter Simulation Conference (WSC) Simulation Challenge** has consistently emphasized the vital role of simulation in bridging the gap between academia and industry. This initiative is rooted in fostering interdisciplinary collaboration, driving innovation, and addressing real-world challenges through advanced simulation technologies.

Below are the official competition pages from the past three years:

- 2024 Simulation Challenge
- 2023 Simulation Challenge
- 2022 Case Study Competition

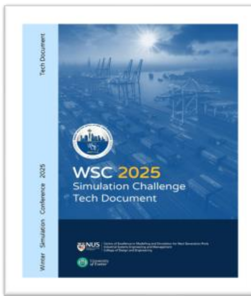
The homepage of this year's competition:

- 2025 Simulation Challenge

We will first discuss the preparation materials provided to participants and how to best utilize these resources. Following this, we'll offer a concise overview of the challenge and outline participant expectations. Lastly, we'll summarize the simulation code structure.

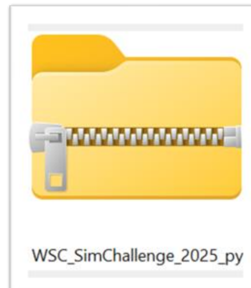
Let's begin with the preparation materials.

Preparations



Download Tech Document

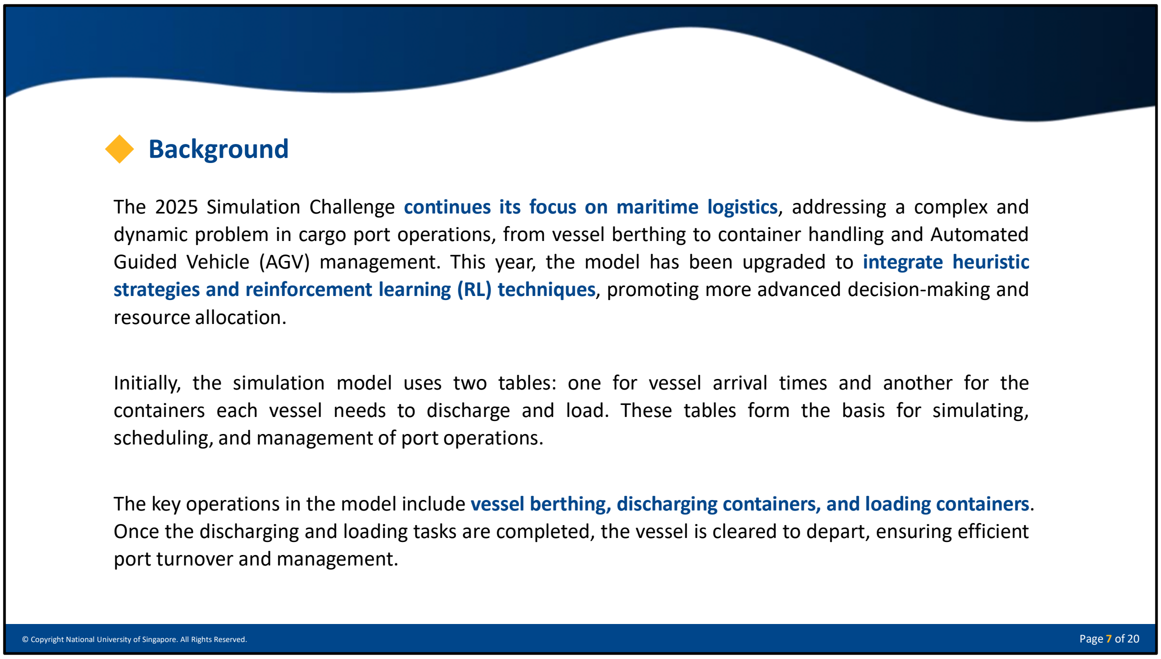
- PDF Reader
- English Version



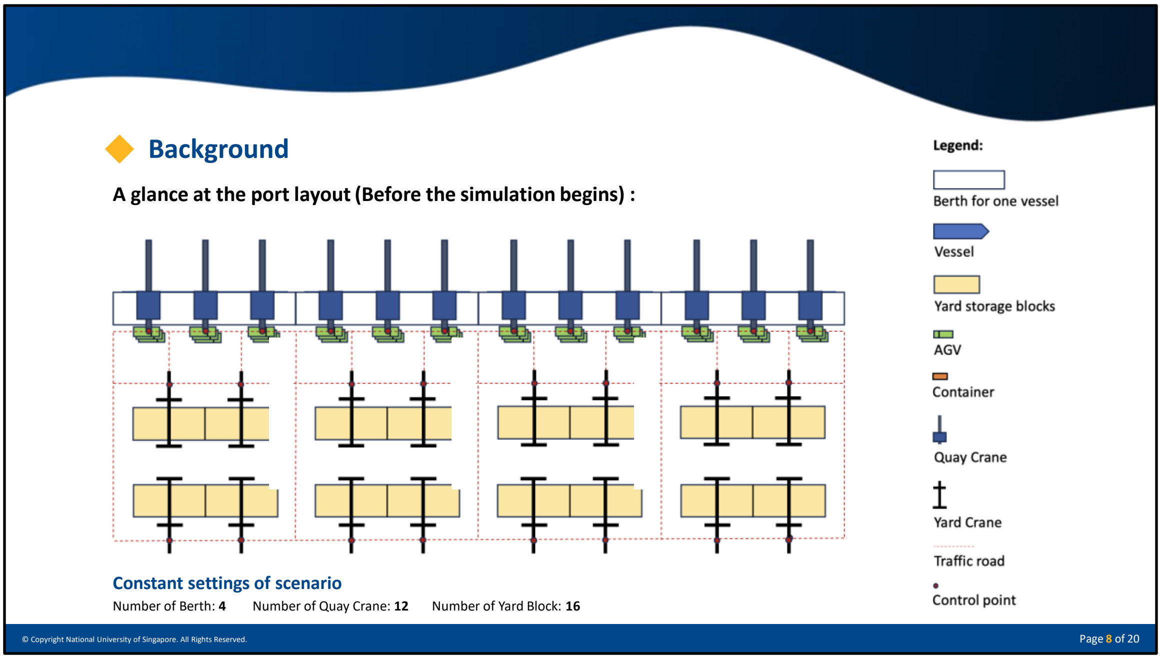
Download Source Code Package

- Decompress Package
- #python

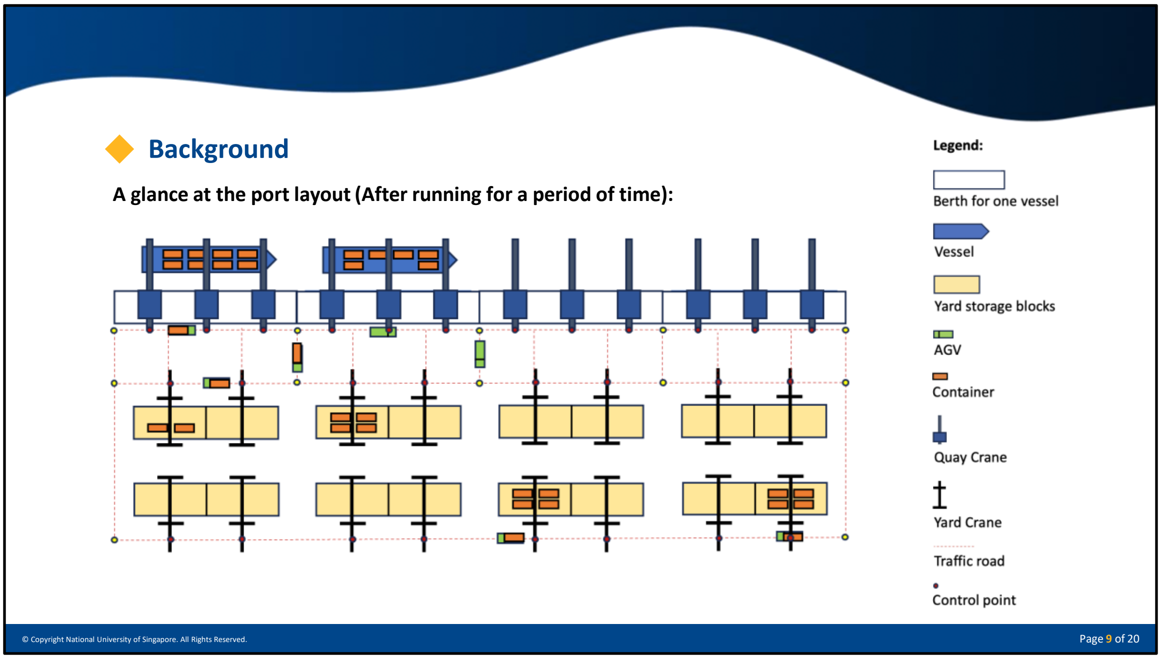
Next, we briefly explain the problem.



Our model simplifies real-world port operations, retaining core functionalities to closely reflect actual scenarios. This approach enables participants to concentrate on key operational challenges, making the competition accessible even for those new to this field.



The figure above depicts the port layout at the beginning of the simulation, before vessels or containers arrive. AGVs are initially stationed uniformly at each quay crane's control points. While fixed parameters—such as the number of berths, quay cranes, and yard blocks—remain constant across all four competition rounds, variable elements like yard storage capacity, vessel arrival frequency, and AGV numbers may differ in each round.

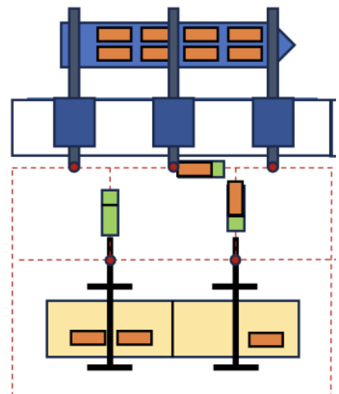


This diagram illustrates the layout of the port after the simulation has been running for some time, showing vessels docked at the berths, each assigned three quay cranes. AGVs are either discharging containers from the vessels onto the yard blocks or loading containers from the yard blocks onto the vessels. In the decision module and assumptions sections, we provide a more detailed introduction to the areas available for optimization by the participants and some operational rules.

Background

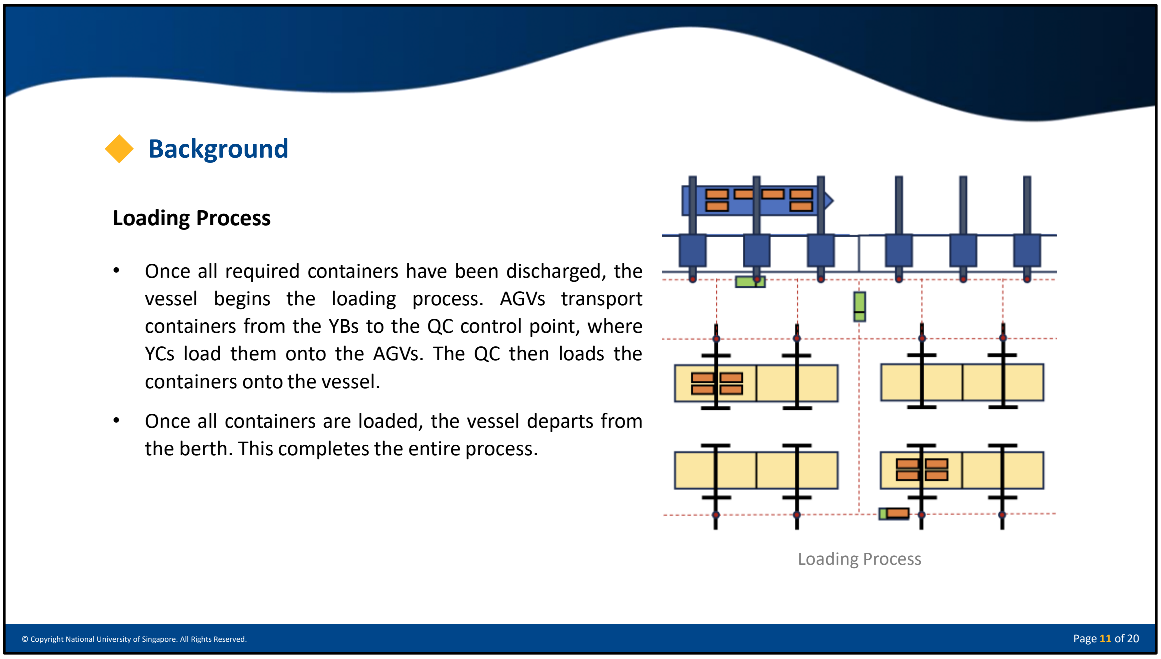
Discharging Process

- At the start of the simulation, AGVs are stationed at control points near each quay crane. As vessels arrive at the waiting area, they dock if idle berths are available; otherwise, they continue waiting.
- Once a vessel docks, quay cranes (QCs) begin discharging containers, which AGVs transport to yard block (YB) control points. Yard cranes (YCs) then transfer the containers from the AGVs to the YBs for stacking.



Discharging Process

Next, we will introduce the detailed processes of discharging and loading.



Importantly, vessels must complete all discharging tasks before loading can commence; this ensures all containers scheduled for unloading are removed first.

Decision Module

Key Areas for Decision Module Enhancement are in:

- **Anchorage Departure Sequencing**: Prioritizing vessel departure from the designated waiting area.
- **Berth allocation**: Allocate berths to vessels, considering arrivals and departures.
- **Yard block allocation**: Identify yard blocks to store containers discharged from vessels.
- **AGV assignment**: Assign AGVs to perform discharging and loading tasks.

The existing default settings in the code:

- Vessels are dispatched to berths following their queue order within the anchorage.
- Vessels are sequentially allocated to Berths. (i.e., from Berth 1 to Berth 4)
- Containers from the discharging vessel are allocated to the closest available YB.
- AGVs will be evenly assigned to each QC all the time.

Participants can modify the provided decision modules to customize their strategies.

Assumptions

To simplify to problem, we made some assumptions

- A berth can only accommodate one vessel.
- Each berth has three quay cranes, and quay cranes cannot be moved to other berths.
- Each yard block has only one yard crane, and yard cranes cannot be moved to other yard blocks.
- There is only one type of vessels.
- All AGVs operate at a constant speed.
- There is no congestion or collision between AGVs.
- The positions of all quay cranes and yard cranes do not change during operation. The execution of each task occurs at the control point.
- After completing the task, the AGV will stay at the control point and wait for the next task.

Objective Function

The objective function is to minimize the average total time (including waiting time and service time) of the total number of arriving vessels. The mathematical formulation of this function is as follows:

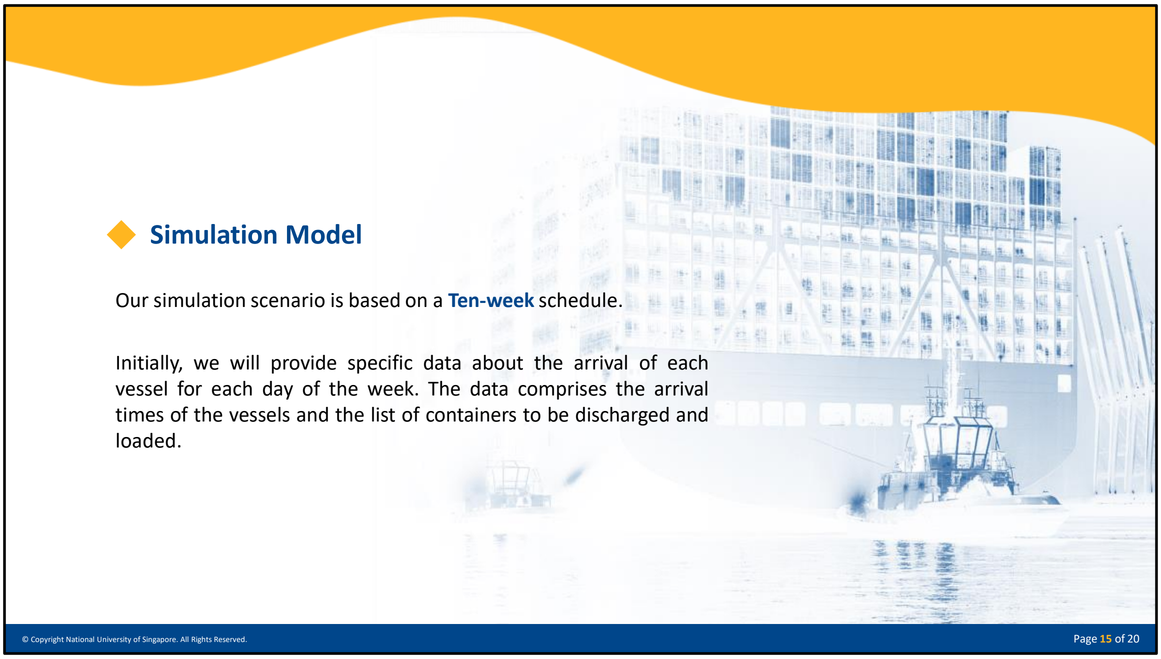
$$\text{Minimize} \quad S = \frac{1}{n}\sum_{i=1}^{n}\left(t_i^{de} - t_i^{ar}\right)$$

Where:

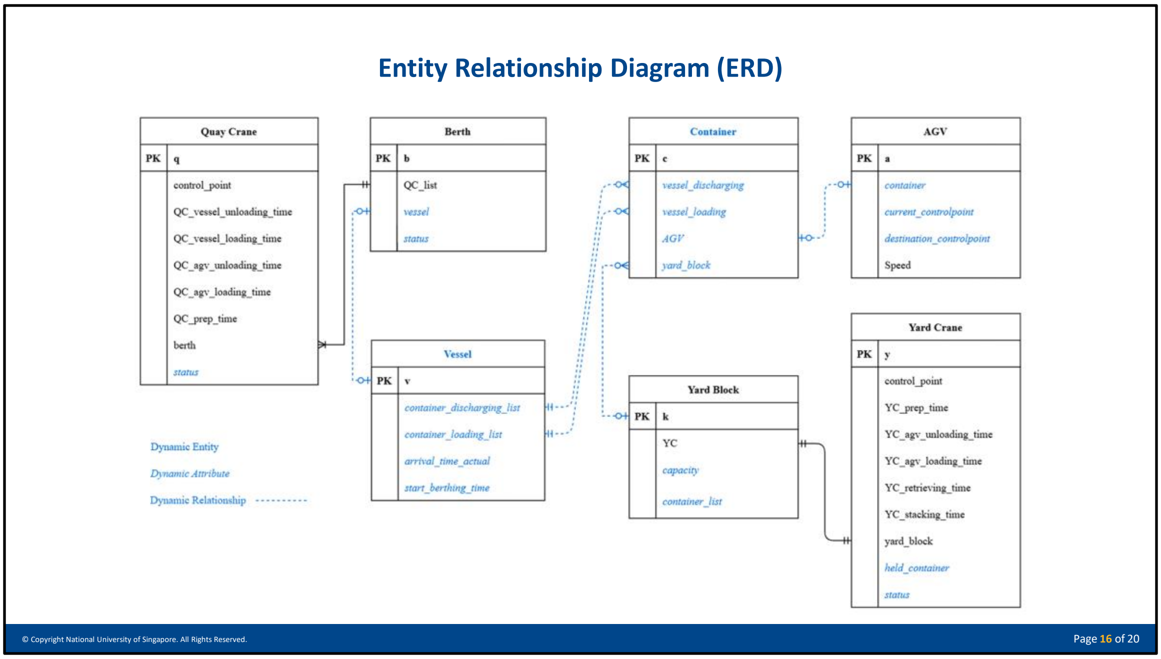
- **S** is the objective function representing the average total time of the vessels.
- n is the total number of vessels.
- t_i^{de} is the time when vessel i departs from the berth.
- t_i^{ar} is the time when vessel i arrives at the anchorage.

Our objective function, described mathematically in this slide, aims to minimize the average total time (including waiting time and service time) of the total number of arriving vessels.

We encourage all participants to approach these challenges enthusiastically. Remember, every intricate problem presents an opportunity for an exciting and engaging experience. You may utilize simulation, optimization, machine learning, or hybrid methods to achieve optimal results.

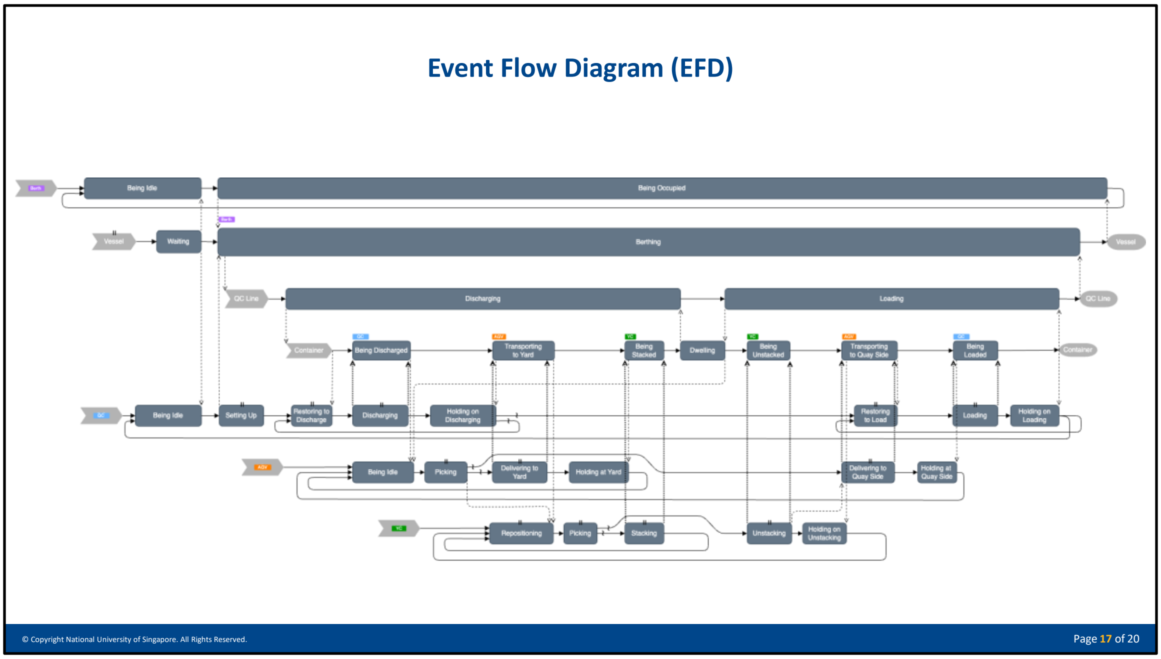


Please note that during the first round of competition, the simulation will run for ten weeks, with identical input data (such as vessel arrival times and transshipment details) for each week. Initially, only discharging operations occur, allowing containers to accumulate in yard blocks. Loading processes begin from the second week. Any changes to these conditions for subsequent rounds will be communicated promptly.



The Entity Relationship Diagram above displays **Seven** entities: Quay Crane, Berth, Vessel, Container, AGV, Yard Block, and Yard Crane.

Each entity has attributes, with static (i.e., unchanging) attributes shown in **black** and dynamic attributes shown in **blue** (subject to change over time). For detailed definitions for these attributes, please refer to the Tech Document.



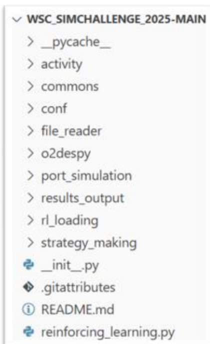
The Event Flow Diagram outlines the sequence of activities for each entity. The Tech Document breaks down this diagram into manageable sections, detailing how entities progress through various activities and interact with each other.

03 Summary

- Structure of Simulation System
- A demo Experiment

Structure of Simulation System

Port Operation System Structure

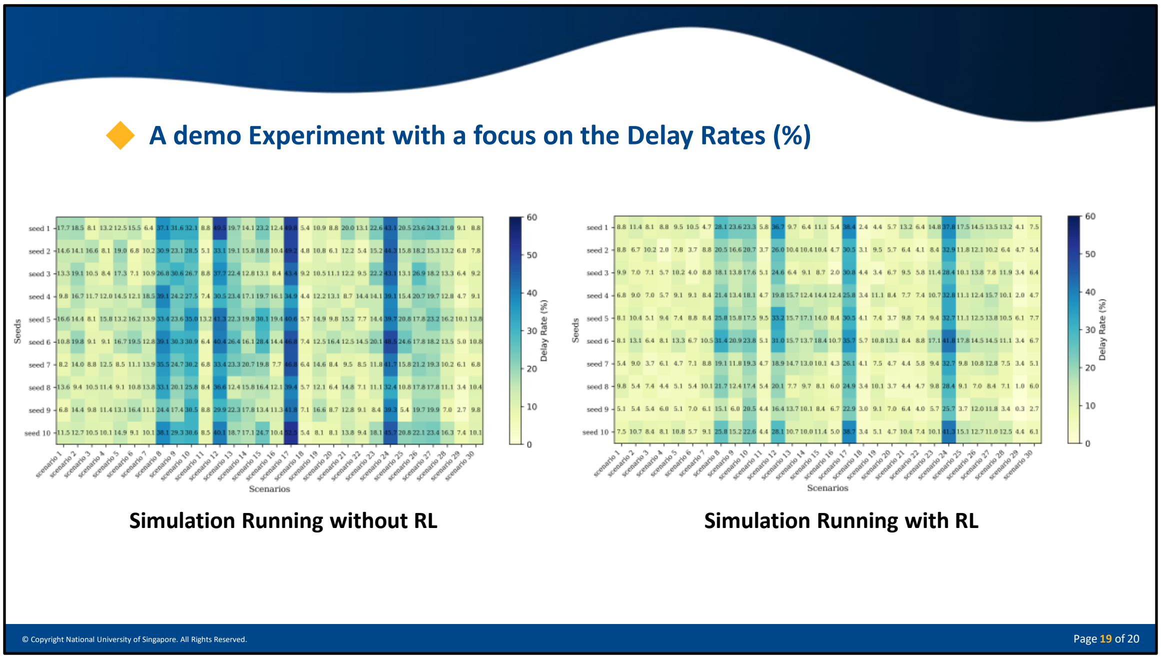


Folder Name	Files Contained
conf	QC_controlpoint.csv; transhipment.csv (0-29); vessel_arrival_time.csv (0-29); YC_controlpoint.csv
port_simulation	agv.py; berth.py; container.py; control_point.py; qc.py; qc_line.py; vessel.py; yc.py; port_sim_model.py
strategy_making	decision_maker_heuristic.py; decision_maker_learning.py; default.py

The above table lists the content of each folder in the provided simulation code. The **conf** folder contains the scenario files (CSV documents). Please note that the existing CSV files are for Round 1 and when the competition progresses to Round 2 and 3, we will provide corresponding scenario files, respectively.

The **port_simulation** folder contains the code related to entities and the simulation model.

The **strategy_making** folder contains the default.py, decision_maker_heuristic.py and decision_maker_learning.py files.



This is a demo showing the role of reinforcement learning (RL).

As you can clearly see in the comparison, the simulation on the right, which incorporates our RL model, maintains a significantly lower delay rate across nearly all scenarios.

This visual representation strongly supports the effectiveness of our RL approach in optimizing performance.

Thank you for participating in the WSC Simulation Challenge 2025. Best of luck!