Middle East Technical University
Northern Cyprus Campus

System Simulation

Project Progress Report - CNG 476

Submitted By:
Brandon Simoko - 2700870
Muhammad Arslan Ul Haq - 2692507
Computer Eng. Department

Submitted To:
Assist. Prof. Dr. Toaha Raza Khan
Comp. Eng. Department

May 5, 2025

Smart Courier Service

1 Introduction

This progress report outlines the current status and structure of our smart courier service simulation project. The goal of the project is to simulate a basic courier delivery system within a small city using discrete-event simulation techniques. The system models parcel movement, vehicle routing, and node communication, where all sub-hubs communicate directly with the central hub. This report explains how key simulation concepts such as entities, events, and system state are implemented in our model.

2 Progress Outline

To this point we have managed to identify the key components in our system design which include the entities and their attributes, activities, events, network architecture for data transmission between nodes along with the type of simulation we aim to realize. In addition, we have managed to code all essential probability distribution functions which are to be utilized within the system along with helper functions. It is worth noting that our system is still in the early stages thus more components will be added with time. All function definitions mentioned in this paper are available in our repository on github, link provided here `https://github.com/branrx/CNG476.git`. Regarding, actual simulation with OMNET++ and FLORA, we are still to integrate the said framework as we are yet to see an introduction of the said frameworks in class.

3 Simulation Concepts and their Implementation in our System

To better understand how our courier simulation system functions, we describe below how key concepts of discrete-event simulation are applied in our project.

3.1 Model

Our model represents a smart courier system operating within a small city. It includes a central hub and multiple sub-hubs (nodes), connected by delivery routes. Parcels are transported from source to destination using delivery vehicles. The model simulates real-world courier operations such as parcel scanning, movement, and vehicle routing.

3.2 System

The system is a network of courier nodes (central and sub-hubs), parcels, and delivery vehicles. These entities interact over time, parcels are handed in, scanned, transported between nodes and delivered.

3.3 System State

The system state at any point includes:

- The number of parcels at each node.
- The location of each delivery vehicle.

- The delivery status of each parcel (e.g., in-transit, delivered)
- The current time in simulation.

3.4 Entities

3.4.1 Main node

The central station which monitors other sub nodes, its keeps tracks of system analytics within the network.

Attributes

- Number of trucks, truck status (in transit or idle)
- Number of packages at each location
- Number of packages in transit

3.4.2 Sub Nodes

A point where parcels are handed in or delivered.

Attributes

- **C**urrent parcel count.
- **C**onnected nodes.

3.4.3 Parcels

Attributes

- **D**estination
- **W**eight
- **D**elivery status

3.4.4 Delivery trucks

Attributes

- **D**estination, where it's going.

3.5 List

We use the following:

- A **queue** at each node to hold parcels waiting for pickup.
- A **vehicle assignment list** to track which vehicle is heading to which node.

3.6 Events

Events in our system include:

- A parcel being handed in at a node.
- A vehicle arriving at a node.
- A parcel being delivered.
- A network failure.

3.6.1 Package arrival

This describes the arrival of a package at a node within the network. We utilizes the inverse exponential distribution function to determine time of arrival, by the use of function with definition provided here 1 and inverse poisson distribution function to determine the number of packages arriving at the said time, which makes use of the function defined here 2.

inverse exponential distribution function definition

```
double inverseExponential(float count, float t, double u)
```

inverse poisson distribution function definition

```
int poissonInverse(float lambda, int n)
```

3.6.2 Package Delivery

This describes the arrival of a package at destination node. Either success or failure, thus it uses the binomial distribution to generate a value. Where $u<0.5$ failure, $u \geq 0.5$ success. For this we utilized the function defined here 3.

3.6.3 Network failure

Since our system relies on a star network for communication between nodes, to acknowledge node statuses, (package counts, package delivery status), we aim to simulate communication failure between main and sub nodes. Function definition is provided in 3.

binomial function definition

```
long double binomial(int n, int k, float p)
```

3.7 Event List (FEL)

We maintain a **Future Event List (FEL)** with scheduled parcel hand-ins, vehicle movements, and delivery completions, all ordered by simulated time.

3.8 Activity (Unconditional Wait)

These are processes which happen over a period of time, therefore to model this we have to simulate amount of time the process takes until it triggers an event. For example, a truck traveling between nodes is an activity, and it arriving at the intended destination node is the event triggered after the said event.

3.9 Arrival of delivery vehicle at a node

To facilitate transportation of parcels between nodes, our system relies on physical vehicles travelling between nodes in predefined routes. We simulate the time it takes for a truck to travel between nodes and with this we can establish the expected time of arrival of a truck at a node. Average time taken between each pair of nodes is initialized at system start up and varies across the system depending on the distance between nodes. Thus we simulate the time taken by a truck to travel between nodes and then trigger arrival of the truck as an event. We use the exponential distribution defined here 1 to model the time taken for truck to travel between nodes.

3.10 Time to repair

This describes the time taken for a network failure to be resolved. We model this using the exponential distribution, and the mean value is randomly decided at system initialization with function definition provided here 1.

Examples:

1. Time taken by a vehicle to travel from one node to another (known when the vehicle starts moving).
2. Parcel scanning time at a node (fixed short duration).

3.11 Delay (Conditional Wait)

This an undefined amount of time in which the clock progresses whilst some entities within the system wait for an event. Examples:

1. A parcel waiting at a node until a vehicle becomes available for pickup.
2. A vehicle waiting until it gets assigned a route based on parcel load.

3.12 Clock

We use a **discrete simulation clock** that moves forward in discrete steps, based on the next scheduled event in the FEL. Time is updated as events are processed. For simplicity, the clock shall tick hourly as we will be running simulations on a daily basis. This is intended to simulate a working day, starting from 8 am to 5 pm, as package arrivals and deliveries are only expected within this time period.

3.13 Network Architecture

We intend to create a star network data for data communication between nodes. Given we have a single main node, it will be linked to all other nodes which increases the networks robustness and also reducing complexity of data hoping through multiple nodes as was with the mesh network proposed prior.

4 Conclusion

In this progress report, we have outlined the foundational structure of our smart courier service simulation using discrete-event simulation concepts along with the achievements obtained thus far. By clearly defining the model, system components, and simulation events, we have established a simple yet effective framework for tracking parcels and vehicle movements across the city. Moving forward, we will focus on refining our simulation logic, incorporating additional parameters (if needed), and evaluating system performance using FLORA and OMNET++ to enhance system reliability and simulation precision.