Wojciech Rościszewski-Wojtanowski
ID 140062

POLITECHNIKA POZNAŃSKA – POZNAŃ UNIVERSITY OF TECHNOLOGY

Wydział Informatyki i Telekomunikacji – Faculty of Computing and Telecommunications

Techniki symulacyjne – Simulation Techniques

Project Task: Truck Company – Zadanie Projektowe: Firma Ciężarowa

CONSULTANT PROFESSOR

Dr. Paweł Sroka.

Poznań, 2020.

1. Copyright

2. Abstract

Presents the state-of-art techniques used in environment modelling, simulation-based evaluation of discrete-event systems, clock advance mechanisms, activity scanning, random number generators and all other examples of simulation models, methodology of computer simulation and source code documentation with all strictly speaking.

3. Table of Contents

4. Table of Figures

5. Introduction

Task:

- Simulate a transport company, where k trucks with capacity of u units transport cargo goods between the headquarters and n regional depots.
- Goods arrive in batches of size r units, where r is a random variable following normal distribution with mean μr and variance σr 2 (but the lowest possible size is 0.1 unit and the highest is 10 units).
- In the headquarters the cargo batch destination depot dn is a random variable following uniform distribution with equal probability for each depot, while the cargo batch destination for all cargo generated in depots is the headquarters.
- The time interval between the arrival (generation) of two consecutive cargo batches is a random variable with exponential distribution and average chq and cd for the headquarters and each depot, respectively.
- Cargo batches wait in queues at the headquarters and the depots until they are taken by the next truck going to the destination point of a cargo batch, where the truck destination at the headquarters is selected with strategy S, while for the depots the destination is always the headquarters.
- After the destination is determined the cargo batches destined to this point are loaded in first-in first-out order (FIFO) until the next one exceeds the truck capacity.
- Only full batch can be loaded, assuming there is enough space in the truck. In the headquarters there are M loading/unloading platforms, while in each depot there are N loading/unloading platforms, with single platform able to accommodate one truck.
- Trucks upon arrival enter a free platform unless all of them are occupied – in such a case they wait in queue. Then they are unloaded and later loaded, with the loading and unloading time for each cargo batch determined as r*Tl and r*Tu, respectively.
- Delivered cargo batches leave the system. Trucks leave the platform according to strategy P.
- The traveling time for each truck between any depot and the headquarters follows normal distribution with average μt and variance σt 2.

6. Simulation Parameters

At a glance:

Parameters	D2
K Truck	12
U Capacity of truck	12
N Number of regional depots	8
(normal distribution with mean and variance) **(μ_r and $\sigma_r{}^2$)**	(2.5, 0.82)
c_{hq}	3.1
c_d	102
(*M*, *N*) Loading/unloading (HQ plat, Deports plat)	(3,2)
T_l	0.65
T_u	0.55
(μ_t, $\sigma_t{}^2$)	(22.0, 1.82)

Figure 1 Parameters

Strategy:

	Destination selection strategy S:
S3	Destination depot is selected as the one with the highest number of queued cargo units destined to this point.
	Trucks departure strategy P:
P3	If the truck is fully loaded right after unloading (more cargo batches queued that truck can accommodate) it leaves immediately. Otherwise, the truck status is checked every 2.0 time units and the decision on the leave is made according to uniform probability distribution with the probability of immediate departure equal to the fraction of truck capacity that is already loaded (e.g. if the truck is loaded in 55% the probability of departure is 0.55).

Figure 2 Strategy

Method:

	MT Simulation Method
M1	Activity scanning

Figure 3 Simulation Method

7. Simulation Model Scheme

In the below please seek a simulation model scheme that presents the steps of mostly each object found in the simulation task, my parameters should satisfy this model. We should consider each item such as a truck, loading depot, unloading depot as objects to which we assign certain events, I believe I can present the structure of how such events should be carried out.

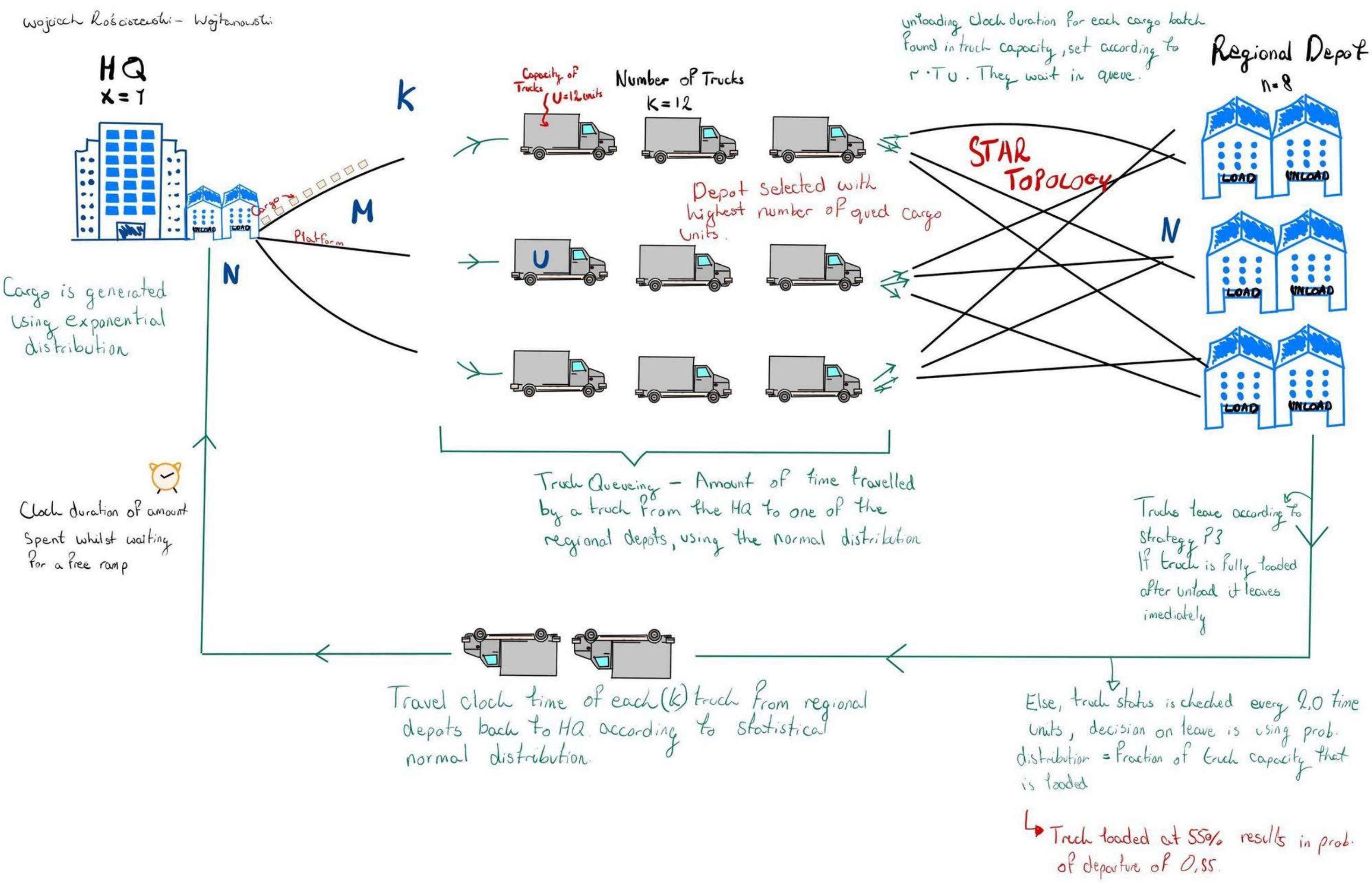

Figure 4 Simulation Method

In the below please seek a block diagram, which presents the layout of each event found in the simulation.

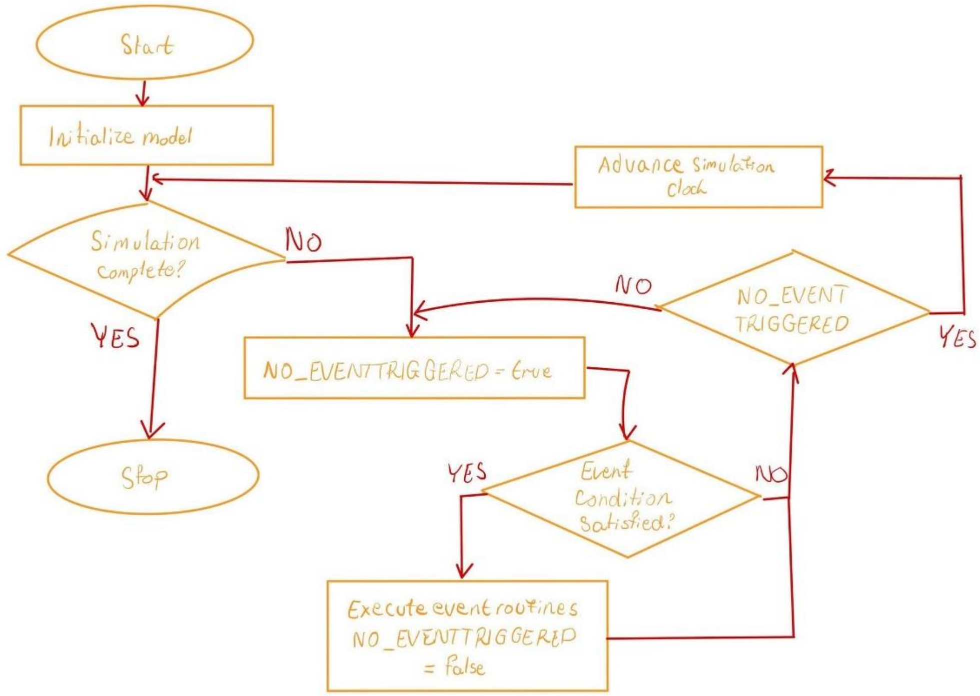

Figure 5 Presents Activity Scanning for an Event

8. Objects and Attributes

In the table below please find the list of objects used in the implementation.

Object Name	Description	Properties
Cargo Batch	This'll be a batch of cargo that will be travelling towards one destination, with the size of back dependent on r variable. R variable is a random variable following normal distribution with mean μr and σr^2.	• The size of the cargo batch generated, the lowest possible size is 0.1 unit and the highest is 10 units, • Cargo batch generation based on the parameters μ_r and σ_r^2, • The ID of the cargo batch object, • Destination of the batch, • Queuing time of the batch, • The time the cargo batch entered the system, so entry time is needed.
Headquarters	This'll be the HQ location that had multiple platforms (Loading and Unloading). The platforms will serve the trucks.	• M platforms (a set of platforms) available at the HQ location (3 loading/unloading), • Que of trucks, • Loading/Unloading time determined by $r*T_l$ and $r*T_u$ of a cargo unit, • Que of cargo batches, they will wait to be taken by the next available truck where the truck destination is selected with strategy S, • Batch generation interval based on C_{HQ} for location.
Regional Depots	This'll be the depot location with multiple platforms to serve the trucks (Loading and Unloading).	• N platforms (a set of platforms) available at the depot location (2 loading/unloading), • Que of trucks, • Loading/Unloading time determined by $r*T_l$ and $r*T_u$ of a cargo unit, • Que of cargo batches, they will wait to be taken by the next available truck where the truck destination is selected with strategy S, • Batch generation interval based on C_D for location.

Transport Company	This'll be the main object carrier. Simply, it will contain all other necessary objects for the simulation to run.	• A will be the headquarters variable (we have 1 HQ), • n will be the regional depots variable (2 so loading/unloading); this will contain a set of depot locations, • k will be the amount of trucks (set amount of specific trucks), • simulation time clock, • simulation agenda.
Trucks	The trucks will be travelling to locations with cargo.	• The truck destination will be determined by the strategy S, here destination depot is selected as the one with the highest number of queued cargo units destined to this point, • Maximum capacity of cargo batch that can be loaded into the truck before travelling, this is the capacity u, • Amount of free space in the truck can be taken into consideration to see how much cargo batch we can load, • Truck ID will be crucial to identify/keep track of trucks, • Service Time, this'll be time of loading/unloading/travel, • Idle Time this'll be total time spent in the system, • A set of cargo batches which are carried by the truck, this is where we store the cargo. We need to know when and how long that load in truck is going to location. • Travel time parameters of the truck from location to location, using parameters μ_t and σ_t^2.

Figure 6 Objects

9. Time and Conditional Events

Event Time	Event Type	Description	Event Procedure
Arrival at the Headquarters	Time	A truck arriving at a headquarter	Enqueue the truck at location HQ/Depot.
Start of Service	Conditional	A truck enters platform and starts unloading of the cargo	If a truck is queued and platform is found to be empty do the following procedure: 1. Dequeue the truck and make the platform busy, 2. Calculate the unloading time and schedule a new event and end the unloading process. Mark the truck status as busy (for loading event, start).
End of unloading	Time	A truck finishes unloading	Mark the truck status as unloaded and change state to ready to load so update the free space and cargo set in the truck.
Start loading cargo batch	Conditional	Start loading of a cargo batch going to a selected destination	If a truck on a said platform and has free space to load next cargo batch (finished unloading/loading) and cargo batches queued do the following: 1. If destination is unknown determine the truck destination according to strategy S, 2. Dequeue next cargo batch going to the same destination and put it in the truck. Update the free space in the truck, 3. Calculate the loading time and schedule a new event so end of loading event and

			proceed to mark the truck as busy. Else if there are no cargo batches to destination queued but no free space to load in the truck, set the truck status as loaded, for this we should periodically check the truck status.
End of loading	Time	Finish loading cargo batch	Mark the truck as idle (ready to load)
Truck departure	Time/Conditional	Truck is loaded and will depart from outgoing location towards ingoing	If the truck status is set as loaded and is idle: 1. Make the platform status set as free, 2. Calculate the travelling time to the destination, next top, and schedule truck arrival event.
Cargo batch arrival	Time	New cargo batch arrives at location	If the truck arrives at a destination: 1. Generate the size and destination of cargo batch, 2. Enqueue cargo batch
Periodic truck status check	Time	Check the status of the truck, if the truck leaves or is found in the platform	1. Determine the probability of the truck leaving based on the amount of load 2. Generate the uniform random number and if it is smaller than probability set the truck status as loaded and if not then schedule a periodic truck status check.

Figure 7 Time Events

Process Name	Description	LiceCycle Phases No.	LiceCycle Phases
Truck	Behavior of truck	Phase 0	Arrival at the HQ
		Phase 1	Start of unloading at HQ, execute Wait(loading_time)
		Phase 2	Finish unloading at HQ and start loading cargo batch – execute WAIT (loading_time)
		Phase 3	Finish loading the cargo batch and if the truck is full go to next phase else Wait_Till(new cargo batch to be loaded)
		Phase 4	Departure from the HQ location – execute Wait(travel_time)
		Phase 5	Arrival at the depot location
		Phase 6	Start unloading at depot, execute Wait(loading_time)
		Phase 7	Finish unloading at depot and start loading cargo batch – execute Wait(loading_time)
		Phase 8	Finish loading the cargo batch and if the truck is fully loaded go to the next phase and if not Wait_Till(new cargo batch to be loaded)
		Phase 9	Departure to the HQ location so Execute_Wait(travel_time)
HQ	Generation of the cargo batches	Phase 0	Generate new cargo batch and enqueue.
Depot		Phase 1	Execute_Wait (new cargo batch arrival) and go to 0

Figure 8 Processes

Usually a model would be sufficient however to step up the organizational aspects of this project I've prepared a timeline overview of how each event within an activity of entities behave over time according to their assigned attributes. I will provide this in basic form to only seek this as a reference guide as I see that the model scheme is much more self-explanatory. I will attempt to recreate some events and use them as examples in the timeline where possible.

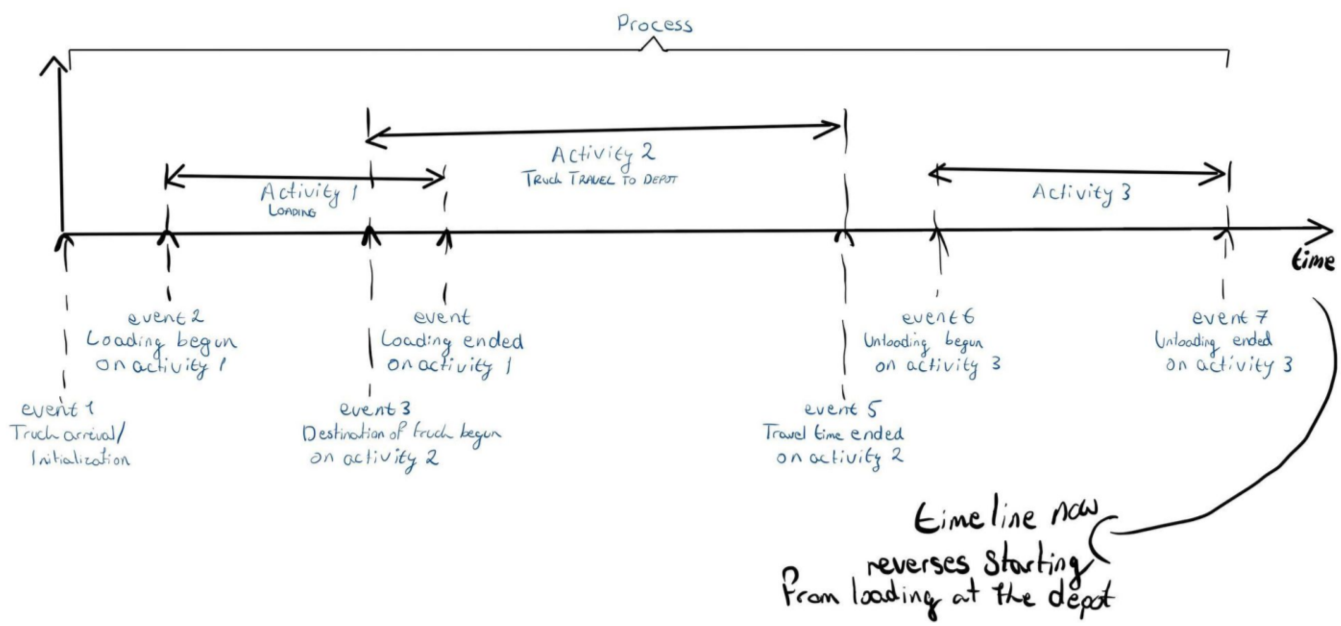

Figure 9 Presents Brief Overview of Timeline of Model Scheme

In fact I only referred to this overview however in the simulation these can be seen.

10. Generators Used

Below please see my uniform generator in my simulation.

```
double Simulation::UniformGenerator(int id)
{
    int h = seeds_[id] / kQ;
    seeds_[id] = kA * (seeds_[id] - kQ * h) - kR * h;
    if (seeds_[id] < 0)seeds_[id] = seeds_[id] + static_cast<int>(kM);
    return seeds_[id] / kM;
}

int Simulation::UniformGeneratorIntercal(int max, int min, int id)
{
    return (int)(UniformGenerator(id) * (max - min) + min);
}
```

Figure 10 Uniform Generator

Below please see my normal distribution generator.

```
double time = 0;
normal_distribution<double> distribution_traveling_time(22.0, 1.8);
default_random_engine generator_traveling_time(seeds_[i]);
```

Figure 11 Normal Distribution

Below please see my exponential distribution generator.

```
default_random_engine generator_cargo_time(seeds_[8]);
UniformGenerator(8);
exponential_distribution<double> distribution_hq_cargo_time(0.3); // lambda = 1/ E[x] -> 1/3.1= 0.3
```

Figure 12 Exponential Distribution

11. Simulation Navigation

The simulation needed to be made in such way that we can somewhat interact with it, in my case I have added various features. Some of course maybe can be even added in the future such as an interactive GUI interface. In the screengrab below please see my example. Firstly, the user is able to specify the duration of our simulation. Then we can specify the duration of the initial time so for example we can see the initial time graph when we start at 0 and then determine at which moment we would like to start in fact. Then the simulation asks the user if we want to load any of our own generation times. Then we have the debugging mode, which is called step by step mode, we press enter to see each step of the simulation, I guess a future development could be inserting the entire simulation information in a separate .txt file. Then we select if we want to graph our initial phase graph. And last but not least we specify the simulation amount. In this case I have selected 1 simulation to show it is working in the screengrabs however really, we would aim to have at least 10 simulations to go through to show that our loop works correctly.

```
Written by Wojciech Ro?ciszewski Wojtanowski 2021.
Specify the duration of the simulation (e.g. 5000) 5000

Specify the duration of the initial time (e.g. 0)0

Do you want to use user specified load generation times? (1 = yes, 0 = no) 0

Do you want to display simulation run (step-by-step) (1 = yes, 0 = no) 0

Do you want to create a graph of the initial phase ? (1 = yes, 0 = no) 0

Enter the number of simulations 1
```

Figure 13 User Input

In the below please see the constructed output terminal that is used to display simulation data. We see the different average values calculated in our simulation. So mostly queuing times which of course are averages, transport time of our cargo and utilizations of our trucks

```
Enter the number of simulations 1

Start simulation
Average queuing time for cargo: 228.623
Average number of queued cargo units in Headquarter: 5.26975
Average number of queued cargo units in Depots:
0 : 39.5519
1 : 11.5368
2 : 10.0778
3 : 41.595
4 : 34.6618
5 : 20.7584
6 : 51.0748
7 : 49.7063
The average transport time for a cargo batch: 260.884
Average utilization of trucks: 0.876189

C:\Users\wojci\Desktop\Politechnika\Simulation Techniques\Main Project (Truck Company Simulator)\Simulation Techniques -
```

Figure 14 Simulation Output

Below please see my step by step (debug mode)

```
Truck with id: 1 has been added to the platform in hq - truck is empty:
Truck with ID: 0 completes loading cargo in headquarter with time: 1.54164 loading: 2.37175
A cargo was generated in depot with id: 5
Simulation Time: 1.88054

Truck with id: 2 has been added to the platform in hq - truck is empty:
A cargo was generated in depot with id: 1
Simulation Time: 2.21863

A cargo was generated in depot with id: 2
Simulation Time: 3.23752

A cargo was generated in depot with id: 5
Simulation Time: 3.30177

A cargo was generated in depot with id: 0
Simulation Time: 3.30287

Truck with id: 0 completes loading cargo  and it's NOT ready to go (probability: 0.197646 ) gen: 0.891035
Simulation Time: 3.34308

A cargo was generated in depot with id: 3
Simulation Time: 3.38166

A cargo was generated in depot with id: 2
Simulation Time: 3.95829

A cargo was generated in depot with id: 6
Simulation Time: 4.02387
```

Figure 15 Step-by-step (debug mode)

12. Simulation Results

In the below please see my results from my simulation, these results are represented in milliseconds. Generally the values are pretty close to each other but sometimes I have some peaks.

	Average Number of Queue Cargo Units at a Localization per Simulation Run									
Simulation No.	**HQ**	**Depot [1]**	**Depot [2]**	**Depot [3]**	**Depot [4]**	**Depot [4]2**	**Depot [6]**	**Depot [7]**	**Depot [8]**	**Average per simulation for Depots**
1	5.35697	404.273	404.083	443.898	385.29	441.022	379.893	561.713	299.994	415.02075
2	5.28064	399.981	554.195	354.434	248.652	660.119	369.99	368.314	451.838	425.940375
3	5.33747	286.453	481.141	356.89	391.192	454.234	460.163	399.907	418.654	406.07925
4	5.30327	446.821	322.829	354.654	487.097	385.921	473.797	285.223	235.215	373.944625
5	5.38126	418.868	431.681	369.434	409.332	319.344	475.384	311.998	485.658	402.712375
6	5.3945	345.372	408.654	253.215	432.553	525.986	410.965	333.768	483.948	399.307625
7	5.38755	609.203	570.381	305.593	379.236	472.977	472.851	386.861	240.447	429.693625
8	5.41288	362.699	473.793	310.83	587.904	548.587	414.09	462.188	405.693	445.723
9	5.34149	346.669	414.1	286.509	482.483	410.727	339.843	221.825	337.587	354.967875
10	5.25128	370.633	388.798	338.192	384.114	283.696	522.381	448.373	366.558	387.843125

Figure 16 Presents averages of localizations

Below please find a similar table as to the one above, the only difference is that I have averaged all depot values for simplicity and ease of graphing this characteristic.

Simulation No.	**Average Number of Queue Cargo Units**	**Average Number of Queue Cargo Units**
1	5.35697	415.0208
2	5.28064	425.9404
3	5.33747	406.0793
4	5.30327	373.9446
5	5.38126	402.7124
6	5.3945	399.3076
7	5.38755	429.6936
8	5.41288	445.723
9	5.34149	354.9679
10	5.25128	387.8431
Confidence Interval	4.8018-4.8374	403.2145-405.032

Figure 17 Presents average of localizations but much smaller table (cargo values are averages of all in a simulation)

In the below please see the final table presenting other averages from my simulation.

Simulation No.	**Average Queuing time for cargo:**	**Average Transport time for cargo batch**	**Average Utilization of Trucks**
1	2985.78	3020.98	0.878078
2	3090.02	3125.03	0.87987
3	2931.96	2966.49	0.881334
4	2708.58	2744.17	0.879699
5	2912.79	2947.77	0.881353
6	2879.31	2914.29	0.880652
7	3051.51	3086.84	0.878371
8	3165.38	3201.24	0.880612
9	2554.73	2589.31	0.882895
10	2805.1	2840.73	0.880452
Confidence Interval	2906.223 - 2910.80944	2941.4034 - 2945.96658	0.643 - 0.8818

Figure 18 Presents continued averaged values of cargo and trucks

Below please see various characteristics of our simulation.

The lowest average found in the graph is at simulation 9 whilst the largest was at simulation 8.

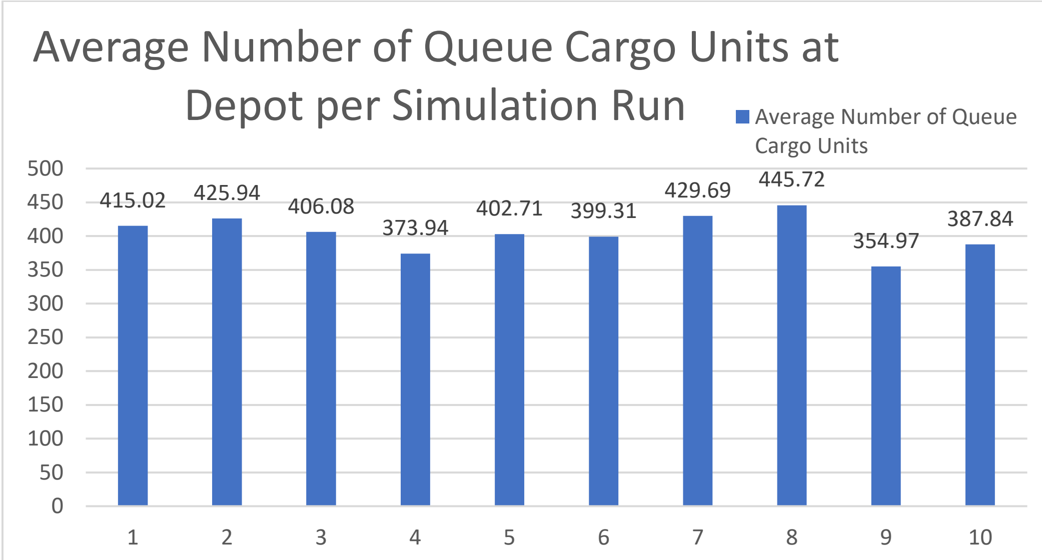

Figure 19 Presents Average Number of Queue Cargo Units at Depot per Simulation Run

The lowest average found in the graph is at simulation 10 whilst the largest was at simulation 8

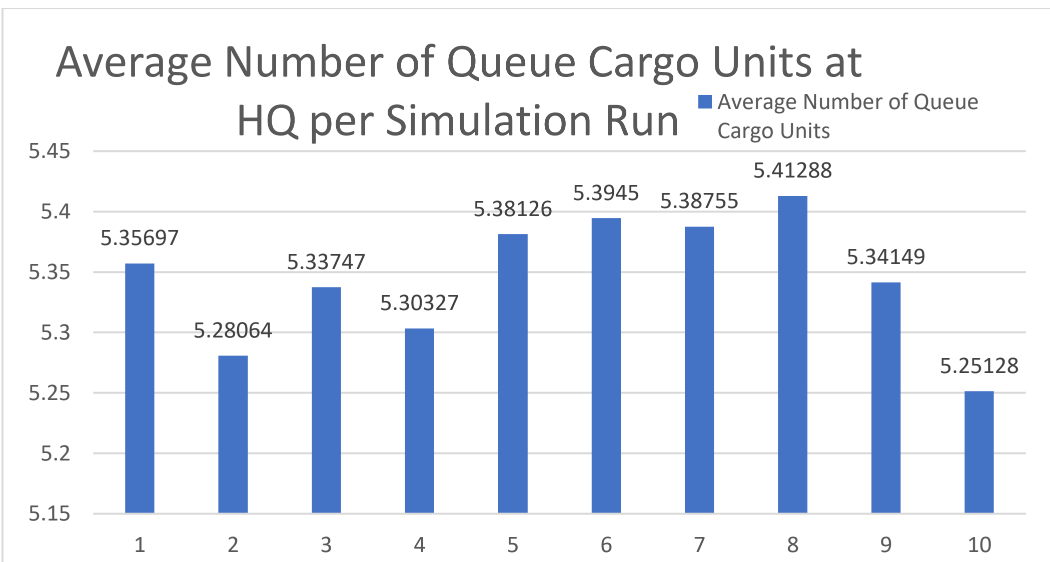

Figure 20 Presents the Average Number of Queue Cargo Units at HQ per Simulation Run

The lowest average found in the graph is at simulation 9 whilst the largest was at simulation 8

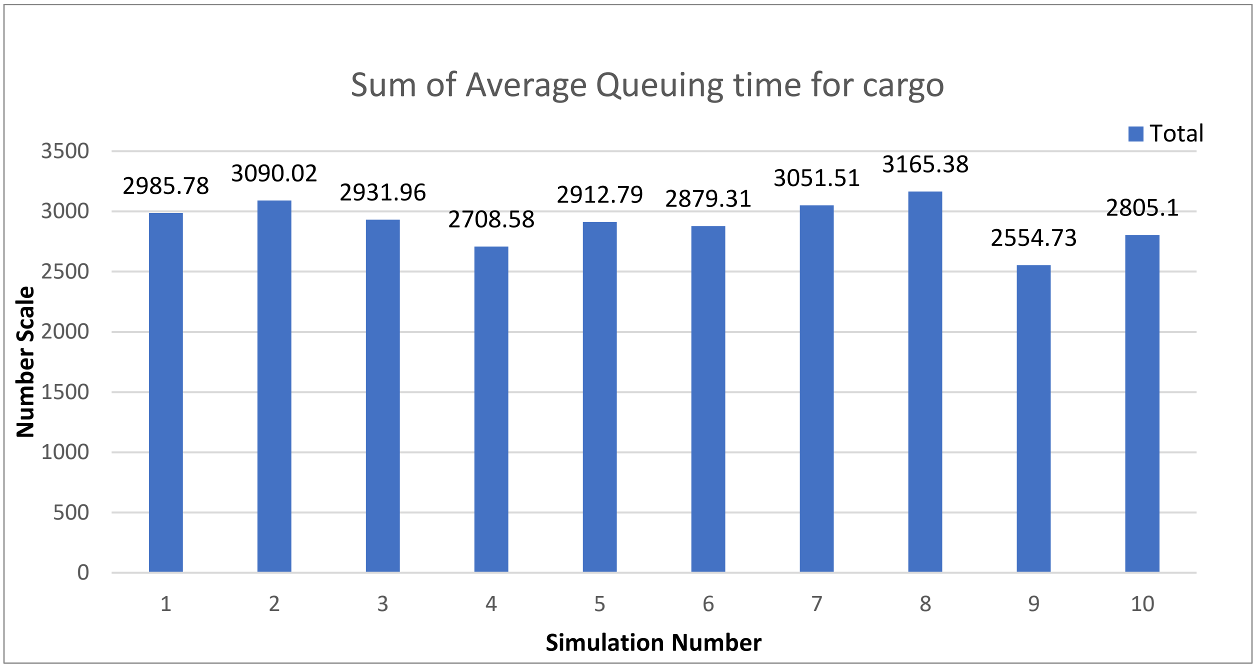

Figure 21 Presents Sum of Average Queuing time for cargo

The lowest average found in the graph is at simulation 8 whilst the largest was at simulation 8

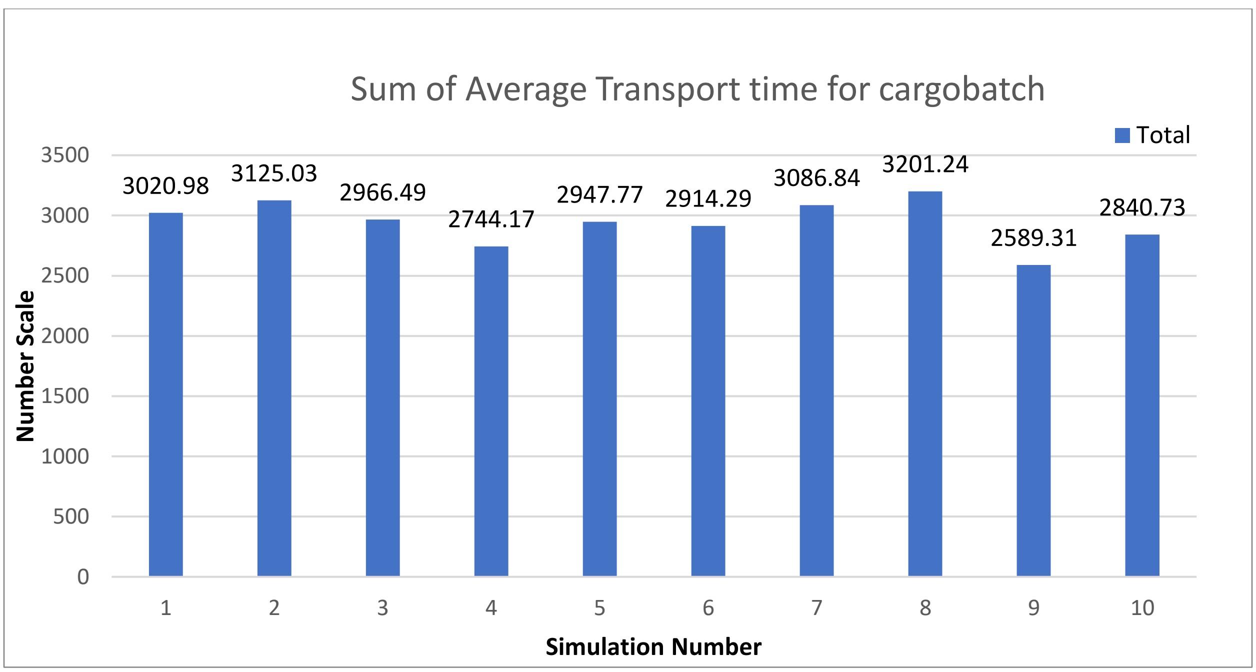

Figure 22 Presents Sum of Average Transport time for cargobatch

The lowest average found in the graph is at simulation 1 whilst the largest was at simulation 9

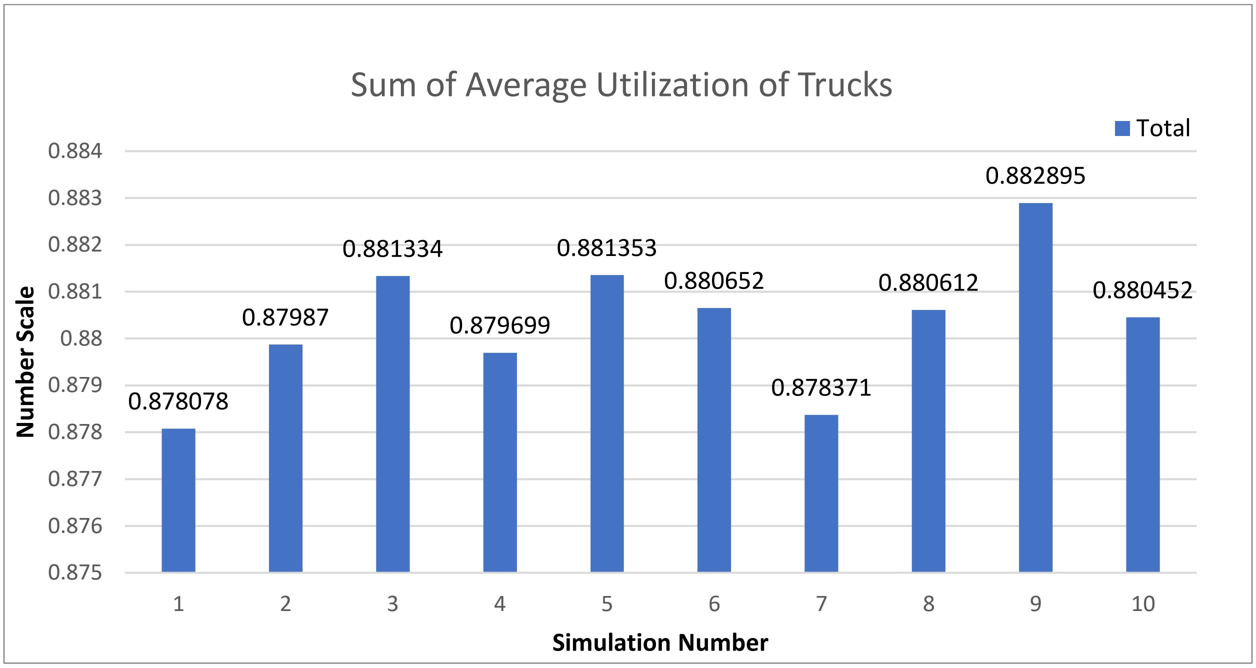

Figure 23 Presents Sum of Average Utilization of Trucks

Below please see the initial phase characteristic. We see that we can start our initial phase at around 20000-time units. (ms)

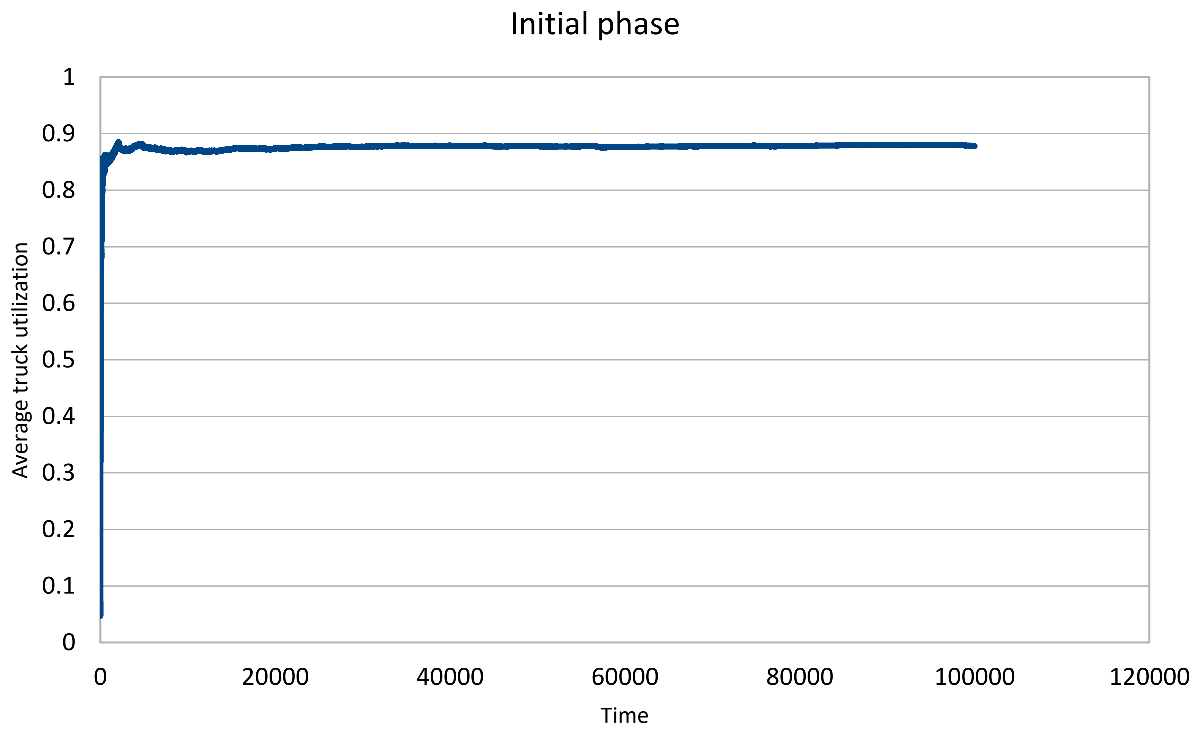

Figure 24 Presents Initial Phase Graph

13. Conclusions

Concluding my simulation project, it can be said that it wasn't easy. From the beginning the diagrams that can be seen in the beginning of this report were very useful during programming stage later, as it helped really to imagine how to implement these objects and in general how the simulation should behave. In the simulation I am satisfied with the results, please see the averaged simulation figures a few pages below where we have very high difference between minimal and maximal average values. Then another thing is that the simulation is quite fast, which I really am happy about that it doesn't waste any computer resources. Downsides are that on my computer sometimes visual basics IDE software tends to crash however I am unsure if it is the simulation fault or computer issue. The simulation obtained some results, we don't really have very large issues or anomalies, so the program is mostly working correctly. I have tried and reset, played around, selecting weird values and the system compiles runs smoothly. Of course, there is plenty of room for upgrades, in my simulation I have really payed attention to making sure that everything is checked before it can be done, so conditionally most methods and activities are checked. Additionally, I have decided to add a feature into my code that if the user wanted to input their own load times for cargo for example. If I was to continue building this simulation, even after handing it in for sure I will attempt to maybe add a graphical user interface, maybe store the statistical data in some database and generate a live dashboard presenting how the simulation behaves -its an improvement idea that I think is worth to explore in the future.