Towson University

Railway Traffic Performance Discrete Event Simulation Model

Neil Labio

COSC 459 Computer Simulation & Modeling
Spring 2022

SUMMARY

Research Requirement:

The purpose of this research was to review and explain the data pertaining a trainyard traffic simulation, regarding cab traffic distribution. Special emphasis was given to time, as the program only allowed time dataset collection for 60 minutes. Regarding track management, the model relates to a simulator-based asset development.

Abstract:

Within the past decade, increased efficiency and competitiveness of the railway transportation network system has sparked the growth of train transport within the United States'. Within this achievement, the modernization of such infrastructure is due to the improvement of system elements regarding their efficiency, such as railway arrivals and departures. Modern computer development can assist in making sound project decisions, nevertheless appropriate and more efficient software tools for analysis must still be developed. In this paper, a simulated mathematical model of two different scenarios regarding railway traffic is provided, created through the modeling simulation tool AnyLogic. The given models were built using a discrete-event approach and a queuing network technique, which allows for the estimation of railroad operating indices and the discovery of bottlenecks in the railway's structure. Based on the simulation's experimental findings, the estimation errors of output parameters were calculated, and the analysis allowed a conclusion to be made on the suitability of the model. Large number of combinations that may be present within a model, which can become even larger in a relatively simple case, can be sampled to get a manageable size for simulations. Considering the properties of synchronous simulation, complex situations with single-track lines can result in gridlocks. Therefore, the number of provided timetables are lowered than what is setup prior to simulation.

Keywords: AnyLogic – Simulation – Discrete Simulation – Macroscopic – Railway organization

Table of Contents

Introduction

Computer simulation is becoming more and more popular even being a part of discipline in todays' field of science. Individuals engage in different scientific problems using a computer's power through simulating artificial experimented environments. Along with this resource, simulated environments are applied within scientific test models to prove or disprove feasibility and accuracy within an event.

Access within these simulated environments is easily accessible within the grasp of technology today, therefore any experiment conducted though the simulated scientific standard could provide results within days, hours, or minutes of the problem conscription. This specialized environment and technicality enables the assumption and simulation of an experiment, without conducting it in the physical world. Within our society's current setting, one of the simulations best studied using modeling are traffic networks and grids. The expediency of simulating a traffic and railway modification within a simulated world increases cost-advantage and efficiency.

Given these simulated events, classifications of simulations may be sorted though:

- Live – Simulation involving real people operating real physical systems
    - Involves Individuals, along with physical equipment
    - Activity/event replication
- Virtual – Where simulation involves real people operating simulated systems, where virtual simulations insert human interaction to provide results.
    - Involves motor skills, decision skills, communication skills
- Constructive - Simulation involving simulated people operating simulated systems. Real individuals provide inputs but are not involved in determining outcomes.
    - Stress test for large samples
    - Generate analysis
    - Outcome prediction

To analyze the simulated railroad traffic system, a model must first be developed mathematically. Such a model should realistically represent a railway traffic flow, based on input constraints such as railway cabs and track exit amounts. Agent-based modeling will be used to identify the constraint of train traffic within the macroscopic level, describing the movement of train cars in terms of density and traffic flow.

Within the given constraints, the task of developing a simulation model of the operation and maintenance process within the given problem using the AnyLogic, is set to provide an opportunity to analyze the indicators of traffic and efficiency of the train yards. The model is designed to assess the Traffic Availability and the Productivity of given tracks, without account for sudden, hidden, and fictitious machine failures. Based on the character of railway junction operation, this paper will assess the discrete event approach that is proposed for mathematical description. With this approach, the attempt to solve and make efficient the railroad is designed. Thus, the problem of a discrete event simulation model construction for the modeled railway junction using AnyLogic software was formulated. The model is supposed to be utilized for the estimation and assessment of a train on the tracks and calculation of train flow within each yard.

Simulation Model

Concept Description (railway yard)

This simulation is a Discrete Simulation model veered towards transit interchange and train cart docking, relating to high volume freight operations. Movement of large rail cars within docks are simulated within this event. Junctions within the tracks are splitters to determine rail car location and destination for storage or transport (storage applies to the rail cars and transport applies to the main locomotive head), branched outwards from the main track, and return assembly. The return assembly of each train locomotive is acted upon a switch between the first and last cars, for the rail cars to dock. Overall, there will be five rail cars each section including the driving locomotive with a gap of five minutes each. Every rail car section of the train is also randomized to provide real word data on rail car range.

The rail cars will first encounter a hump block (where primary engine locomotive is brought to the back of all present rail cars), then are moved to where they are categorized by use and category. A train unit movement consist of individual carloads, and blocks of carloads moving to the same destination, or an entire trainload. The distinguishing characteristic is that some marshaling or gathering of cars is required at origin and/or destination, where car blocks may arrive or depart in way trains, traveling to or from nearby terminals (Illustrated in Figure 1).

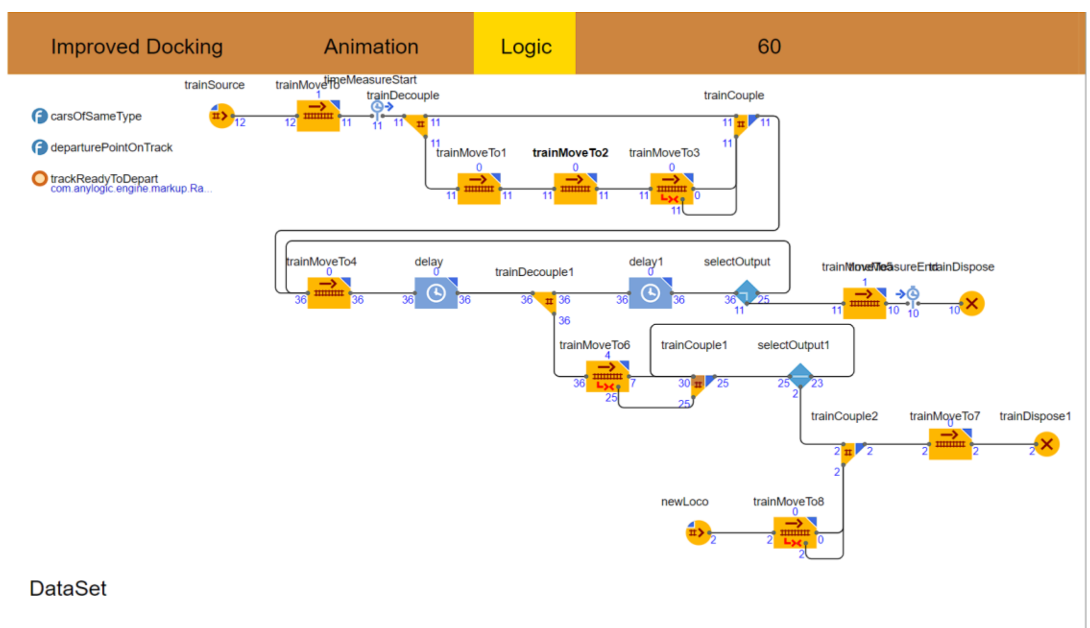

Fig. 1. Improved Docking Figure (Logic)

Discrete Event Simulation

The simulation software AnyLogic was chosen for the modeling and implementation of this project. This railway traffic model uses discrete event techniques, applicable through the logic data and record sets provided in the model. The given logic diagram for this model is structured as a flow chart diagram, where sequential steps need to take place for the program to take path and finish. For clarity, the theoretical description is supplemented by a graphical model, in a graph or state transition diagram form.

Given the two data simulations, the fist independent part of the model is the pre-modification or Initial Docking (ID). The initial model of the train docking accompanies a single track, a hump station (transfer of main cab to rear of sequential cabs) and the separation of all transport rail cars and the main functioning locomotive train driver. The second part of the model incorporates the post-modification or Improved Segregated Docking (ISD). This model of the simulation also begins with a single track with a hump station but is connected to 6 other tracks for rail car sorting and main locomotive departure. This proposed model represents a linear network, as the simulated events are not lost and do not multiply. The model is also open loop, where the train sequences arrive from the external randomly generated medium and leave the area after operation (Figure 2).

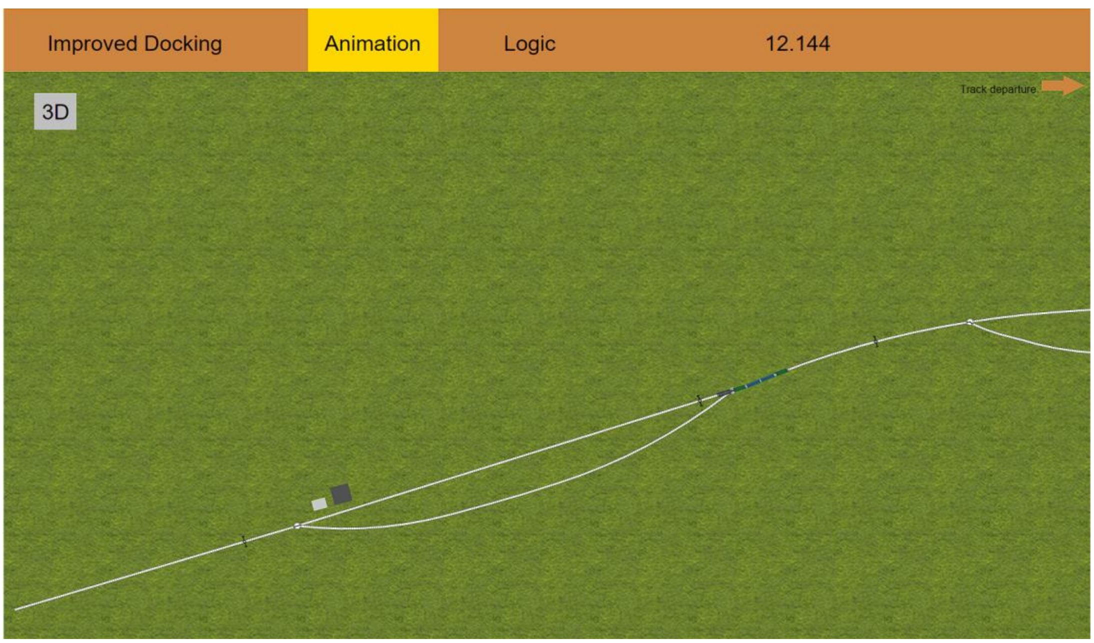

Fig. 2. Docking Figure (Rail-car Departure | Similar on both models)

The categories of railway train will affect the stochastic model output and are categorized randomly. These models are Locomotive, OpenCar, BoxCar, GondolaCar, HooperCar, and TankCar. These attributes were represented as the variable "train" transmitted throughout the model between post and pre modification. Based on the rail cars available across rail movement, there is only one type of rail locomotive which are pick-up trains (trains responsible for hauling other railcars to a given destination). Each junction on the track is a switch, which determines where the current railcar will go, executed by the DeCoupling and MoveTowards function from the software. The first train model will simulate its request to decouple at the Docking, where it will go through the hump yard trail and towards the track it is categorized. The train flows linearly towards a certain point without stop, disregarding train flow and rapid arrival. The requests simulating trains in the model come in the railway switches with the time interval subject to Poisson distribution. As a discrete function, the given train variables can only take specific values in a list. It's also given that the variable cannot take all values in any continuous range, since for the Poisson distribution, the variable can only take values > 0 without any fractions or decimals. The models used for this station were embedded within the software (AnyLogic), providing a unique model for each given railcar for flexibility.

For each Docking diagram, a separate model was assigned (post and pre) considering the difference of data type and terrain view difference and its operations (Initial vs. Improved). Within the data sets, it is seen that different types of trains pass through during various time slots. For the pre-modified model, railcar flow types were the same towards receiving departure, while for post-modified, each unique type of railcar were isolated within one stop lines (Figure 3, Figure 4).

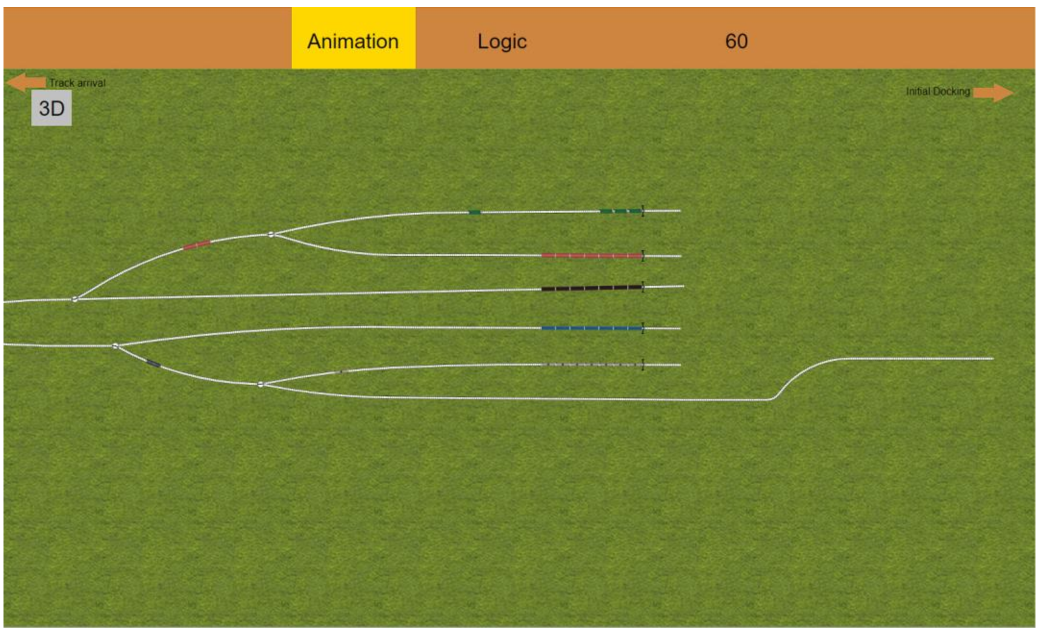

Fig. 3. Improved Docking Figure (Rail-car Group Separation)

The railcars arriving in each segment of the tracks (Figure 3) are the kept until reset maually. Blocks between railways are active objects, allowing separation and unique identificators between each other. There are two train movement directions: forward and backwards. Objects of railway humpblocks are realized with support of the both directions independently. Each activeobject is characterized by 2 different parameters: running time of freight trains, and actual sorting ability.

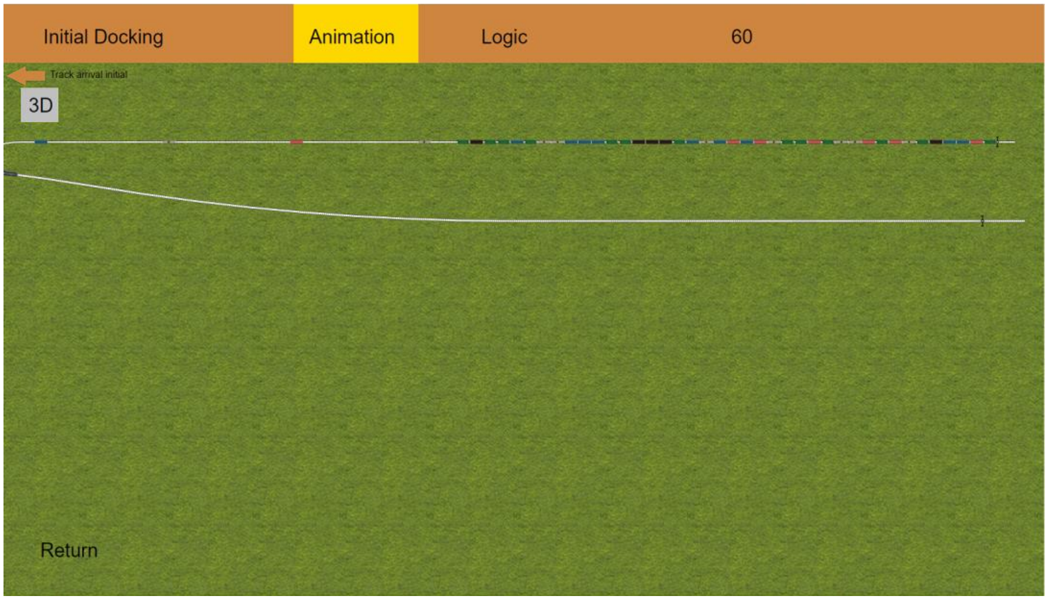

Fig. 4. Initial Docking Figure (Rail-car Group Combination)

Model Experiments

With the implementation of the model, two types of experiments were implemented within the simulation:

1. Analysis of time intervals between the two different rail models (all variables including locomotive size, arrival duration, and rail car amount are the same)
    a. Time Distribution
    b. Time Dataset
2. The simulated modeling of the railway traffic regarding its efficiency and alternatives/modifications.

<u>(Timer was placed at the beginning of each model (train arrival) and at the end (train departure) when the train has unloaded all its railcars)</u>

Analysis of the models were completed to provide a comparison between the difference in Initial Docking and Improved Docking train yard. The Improved Docking and Initial Docking histograms both provide an example of a Discrete distribution. This parametrized distribution shows a rough outline on which the peak time and efficiency the locomotives can move within the models (this is greatly emphasized on the initial docking, as there is only one sorter for all the other railcars). Within these graphs, it is show that

When looking at the Time dataset for both models, a continuous distribution is shown similarly, where the range of each model are shown as a density. The output for the improved docking shows the Probability Density Function and the Cumulative Distribution Functions of both models in Figure 7. With the PDF given, it can be assumed that the Improved Docking is a more efficient way (time wise) of segregating and assembling railcars as it is quicker than a single file line.

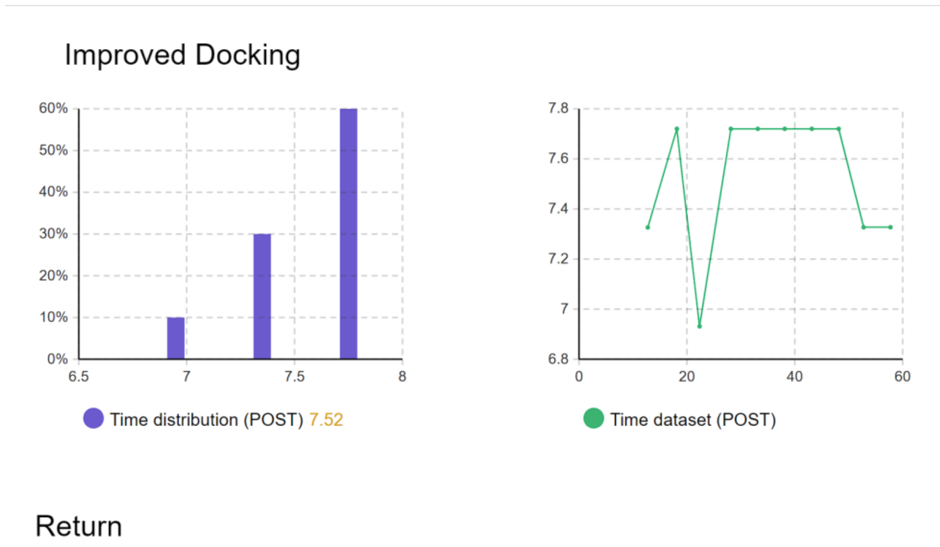

Fig. 5. Improved Docking (dataset and histogram)

Fig. 6. Initial Docking (dataset and histogram)

- Time Distribution avg improved: 7.52
- Time Distribution avg initial: 7.65
- .13 Difference
- 6.65% Relative error (from initial to improved) ((INIT-IMPRO/INIT))

	agent_type	agent	name	start	end	pdf	cdf
1	TimeMeasureEnd	timeMeasureEnd	distribution	7.3	7.4	0	0
2	TimeMeasureEnd	timeMeasureEnd	distribution	7.4	7.5	0	0
3	TimeMeasureEnd	timeMeasureEnd	distribution	7.5	7.6	0	0
4	TimeMeasureEnd	timeMeasureEnd	distribution	7.6	7.7	0	0
5	TimeMeasureEnd	timeMeasureEnd	distribution	7.7	7.8	0	0
6	TimeMeasureEnd	timeMeasureEnd	distribution	7.8	7.9	0	0
7	TimeMeasureEnd	timeMeasureEnd	distribution	7.9	8	0.4	0.4
8	TimeMeasureEnd	timeMeasureEnd	distribution	8	8.1	0	0.4
9	TimeMeasureEnd	timeMeasureEnd	distribution	8.1	8.2	0	0.4
10	TimeMeasureEnd	timeMeasureEnd	distribution	8.2	8.3	0	0.4
11	TimeMeasureEnd	timeMeasureEnd	distribution	8.3	8.4	0.6	1
12	TimeMeasureEnd	timeMeasureEnd	distribution	8.4	8.5	0	0
13	TimeMeasureEnd	timeMeasureEnd	distribution	8.5	8.6	0	0
14	TimeMeasureEnd	timeMeasureEnd	distribution	8.6	8.7	0	0
15	TimeMeasureEnd	timeMeasureEnd	distribution	8.7	8.8	0	0
16	TimeMeasureEnd	timeMeasureEnd	distribution	8.8	8.9	0	0
17	TimeMeasureEnd	timeMeasureEnd	distribution	8.9	9	0	0
18	TimeMeasureEnd	timeMeasureEnd	distribution	9	9.1	0	0
19	TimeMeasureEnd	timeMeasureEnd	distribution	9.1	9.2	0	0
20	TimeMeasureEnd	timeMeasureEnd	distribution	9.2	9.3	0	0
21	TimeMeasureEnd	timeMeasureEnd1	distribution	7.2	7.3	0	0
22	TimeMeasureEnd	timeMeasureEnd1	distribution	7.3	7.4	0	0
23	TimeMeasureEnd	timeMeasureEnd1	distribution	7.4	7.5	0	0
24	TimeMeasureEnd	timeMeasureEnd1	distribution	7.5	7.6	0	0
25	TimeMeasureEnd	timeMeasureEnd1	distribution	7.6	7.7	0	0
26	TimeMeasureEnd	timeMeasureEnd1	distribution	7.7	7.8	0	0
27	TimeMeasureEnd	timeMeasureEnd1	distribution	7.8	7.9	0.2	0.2
28	TimeMeasureEnd	timeMeasureEnd1	distribution	7.9	8	0	0.2
29	TimeMeasureEnd	timeMeasureEnd1	distribution	8	8.1	0	0.2
30	TimeMeasureEnd	timeMeasureEnd1	distribution	8.1	8.2	0	0.2
31	TimeMeasureEnd	timeMeasureEnd1	distribution	8.2	8.3	0.5	0.7
32	TimeMeasureEnd	timeMeasureEnd1	distribution	8.3	8.4	0	0.7
33	TimeMeasureEnd	timeMeasureEnd1	distribution	8.4	8.5	0	0.7
34	TimeMeasureEnd	timeMeasureEnd1	distribution	8.5	8.6	0	0.7
35	TimeMeasureEnd	timeMeasureEnd1	distribution	8.6	8.7	0.3	1
36	TimeMeasureEnd	timeMeasureEnd1	distribution	8.7	8.8	0	0
37	TimeMeasureEnd	timeMeasureEnd1	distribution	8.8	8.9	0	0
38	TimeMeasureEnd	timeMeasureEnd1	distribution	8.9	9	0	0
39	TimeMeasureEnd	timeMeasureEnd1	distribution	9	9.1	0	0
40	TimeMeasureEnd	timeMeasureEnd1	distribution	9.1	9.2	0	0

Fig.7 . PDF and CDF on Time (Model Combination)

	agent_type	agent	name	index	x	y
1	TimeMeasureEnd	timeMeasureEnd	dataset	0	13.136	7.72
2	TimeMeasureEnd	timeMeasureEnd	dataset	1	17.743	7.327
3	TimeMeasureEnd	timeMeasureEnd	dataset	2	22.743	7.327
4	TimeMeasureEnd	timeMeasureEnd	dataset	3	28.136	7.72
5	TimeMeasureEnd	timeMeasureEnd	dataset	4	32.743	7.327
6	TimeMeasureEnd	timeMeasureEnd	dataset	5	38.136	7.72
7	TimeMeasureEnd	timeMeasureEnd	dataset	6	43.136	7.72
8	TimeMeasureEnd	timeMeasureEnd	dataset	7	48.136	7.72
9	TimeMeasureEnd	timeMeasureEnd	dataset	8	53.136	7.72
10	TimeMeasureEnd	timeMeasureEnd	dataset	9	57.741	7.326
11	TimeMeasureEnd	timeMeasureEnd1	dataset	0	13.093	7.688
12	TimeMeasureEnd	timeMeasureEnd1	dataset	1	18.487	8.083
13	TimeMeasureEnd	timeMeasureEnd1	dataset	2	23.093	7.688
14	TimeMeasureEnd	timeMeasureEnd1	dataset	3	28.093	7.688
15	TimeMeasureEnd	timeMeasureEnd1	dataset	4	33.487	8.083
16	TimeMeasureEnd	timeMeasureEnd1	dataset	5	38.487	8.083
17	TimeMeasureEnd	timeMeasureEnd1	dataset	6	42.7	7.295
18	TimeMeasureEnd	timeMeasureEnd1	dataset	7	47.698	7.294
19	TimeMeasureEnd	timeMeasureEnd1	dataset	8	53.094	7.69
20	TimeMeasureEnd	timeMeasureEnd1	dataset	9	58.094	7.69

Fig.8 . Dataset (Model Combination)

Areas For further work

The real-world dynamic and changing environments

- The real world has more than one hump track per trains, therefore this model may not be realistic
- Map data may show pedestrians and commercial trains other than freight trains which transfer goods
- Path finding algorithm to easily identify train tracks and train function
- Accidents and random events
- More than 6 different types of railcars
- Map nodes and other railroads

Result Discussion & Conclusion

When looking at the probability density functions and data set diagrams, both models indicate that the efficiency of Improved docking is a miniscule insight from the initial docking. It is also seen that the relative error estimation ((measured – real)/ real) of both datasets has a very miniscule threshold not exceeding 7% for all given sets.

From the results of the given time intervals between train arrivals and railcar departures, the interval distribution of both models is somewhat similar. This along with a close histogram data provides a very slight difference in time efficiency for modifying the hump tracks.

In this work, a simulation model of a modified railcar departure and arrival was proposed, which was constructed in AnyLogic modeling instrument using discrete events. This project approached and implied Discrete Probability and cumulative distribution techniques. The model allows estimation of such railway operations as it indicates the rehandling of multiple hump tracks and train car partition. Estimation error was calculated not exceeding 7% of the data, indicating the model as adequate.

The purpose of this research was to review and explain the data pertaining a trainyard traffic simulation, regarding cab traffic distribution. To analyze the simulated railroad traffic system, a model must first be developed mathematically. Such a model should realistically represent a railway traffic flow, based on input constraints such as railway cabs and track exit amounts. Agent-based modeling will be used to identify the constraint of train traffic within the macroscopic level, describing the movement of train cars in terms of density and traffic flow.

References

Antognoli et al., 2018Antognoli M., Capodilupo L., Marinacci C., Ricci S., Rizzetto L., Tombesi E.Present and Future Operation of Rail Freight Terminals, in: Transport Systems and Delivery of Cargo on East-West Routes, Springer, Cham (2018), pp. 233-273https://doi.org/10.1007/978-3-319-78295-9_6

Borshchev A 2013 The Big Book of Simulation Modeling: Multimethod Modeling withAnyLogic 6 (Chicago: AnyLogic North America) p 612

Maksimey I V, Sukach E I, Giruts P V and Erofeeva E A 2008 Simulation modeling ofprobabilistic characteristic of railway network operation Mathematical machines and systems 4147–53

Report ITU-R M.2418-0 Description of Railway Radiocommunication Systems between Trainand Trackside (RSTT) (Available at: https://www.itu.int/dms_pub/itu-r/opb/rep/R-REP-M.2418-2017-PDF-E.pdf)

Rizzoli et al., 2002Rizzoli A.E., Fornara N., Gambardella L.M.A simulation tool for combined rail/road transport in intermodal terminalsMath. Comput. Simul., 59 (2002), pp. 57-71https://doi.org/10.1016/S0378-4754(01)00393-7

Zehendner and Feillet, 2014Zehendner E., Feillet D.Benefits of a truck appointment system on the service quality of inland transport modes at a multimodal container terminalEur. J. Oper. Res., 235 (2014), pp. 461-469https://doi.org/10.1016/j.ejor.2013.07.005