Parking Lot Model Simulation with Arena

Report

MIS 64018 – Systems Simulation - Spring 2020

Instructor

Professor Dr. Chaojiang (CJ) Wu, Department of Management and Information Systems
Kent State University, Kent-Ohio.

Team Members

Vijay Bollina, Charity Elijah, Satyasri Pavani Harika Penjerla, Rajendra Pusarapu, Timothy Akintoye

Contents

ABSTRACT

This project describes an animated simulation model that will be used to assess several strategies for improving the Parking Lot C system at Kent State University, Kent Campus. Due to insufficient parking at Lot C it has been a chaos for students rushing for morning classes. We decide on building this simulation in Arena as a project topic.

ARENA SIMULATION SYSTEM DEFINITION

Simulation is one of the most powerful analysis tools available to those responsible for the design and operation of complex processes or systems. In an increasingly competitive world, simulation has become a very powerful tool for the planning, design, and control of systems. No longer regarded as the approach of "last resort" it is today viewed as an indispensable problem-solving methodology for engineers, designers, and managers. We will use Arena Simulation to analyze parking lots at Kent State University.

PURPOSE OF STUDY

Focus of this project is on Kent State University Parking Lot C, the lessons learned, and concepts used in this project can also be applied on other Campus Parking Lots as well. We assume that this study will help respective managers for Kent campus better address the problem of scarce parking space on parking lot C during the morning hours. Moreover, we want to analyze and provide recommendations for improvement in lot C, possibly reduce parking space challenges during the early hours of the day.

MODELLING SETUP

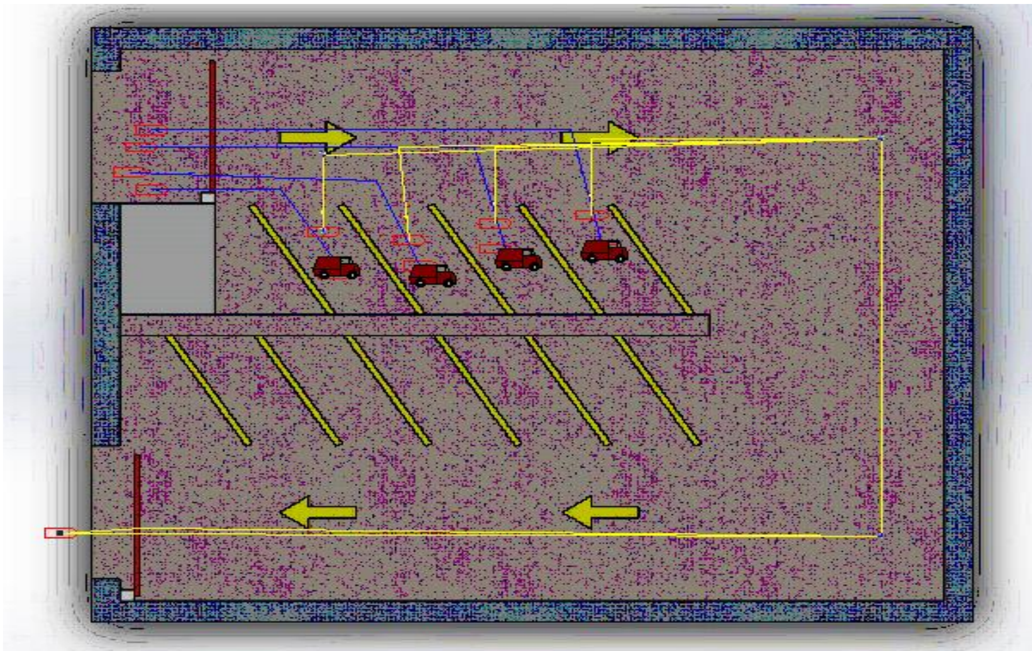


Figure 1 Modelling Setup (Animation)

The above animation depicts the Parking C Lot of Kent State University. The details of this model simulation are given below:

Entities: Cars arrive and depart from a 4-stall parking.

Resources:

- Parking spaces defined by Space 1, Space 2…, etc., where it can accommodate parking.

MODULES

In creating our modules, we utilize varieties of the model implementations tools in arena. In total we created Twenty-one (21) modules of different implementation as required to accomplish our project goal. The most used tool in implementing our modules is the "Basic Process" tool. Out of the twenty-one modules we created, twelve was made with the "Basic Process" tool. Examples of this basic process tools used in this project are; one create, seven assigns, three Decides and one Disposed:

- **Create Module**

This module is intended as the starting point for entities in a simulation model. Entities are created using a schedule or based on a time between arrivals. Entities then leave the module to begin processing through the system. The entity type is specified in this module.

Typical Uses

- The start of a part's production in a manufacturing line

- A document’s arrival (e.g., order, check, application) into a business process
- A customer’s arrival at a service process (e.g., retail store, restaurant, information desk)

➢ **Assign Module**

This module is used for assigning new values to variables, entity attributes, entity types, entity pictures, or other system variables. Multiple assignments can be made with a single Assign module.

Typical Uses

- Accumulate the number of subassemblies added to a part
- Change an entity’s type to represent the customer copy of a multi-page form
- Establish a customer’s priority

➢ **Decide Module**

This module allows for decision-making processes in the system. It includes options to make decisions based on one or more conditions (e.g., if entity type is Gold Card) or based on one or more probabilities (e.g., 75%, true; 25%, false). Conditions can be based on attribute values (e.g., Priority), variable values (e.g., Number Denied), the entity type, or an expression (e.g., NQ (ProcessA.Queue)). There are two exit points out of the Decide module when its specified type is either 2-way Chance or 2-way Condition. There is one exit point for “true” entities and one for “false” entities. When the N-way Chance or Condition type is specified, multiple exit points are shown for each condition or probability and a single “else” exit. The number of entities that exit from each type (true/false) is displayed for 2-way Chance or Condition modules only.

Typical Uses

- Dispatching a faulty part for rework
- Branching accepted vs. rejected checks
- Sending priority customers to a dedicated process

➢ **Disposed Module**

This module is intended as the ending point for entities in a simulation model. Entity statistics may be recorded before the entity is disposed.

Typical Uses

- Parts leaving the modeled facility
- The termination of a business process
- Customers departing the store

Next, we built Six (6) modules with the Advanced Transfer tools. Examples of these modules are; three Routes and three Stations:

➢ **Station Module**

The Station module defines a station (or a set of stations) corresponding to a physical or logical location where processing occurs. If the Station module defines a station set, it is effectively defining multiple processing locations. The station (or each station within the defined set) has a matching Activity Area that is used to report all times and costs accrued by the entities in this station.

- **Route Module**

  The Route module transfers an entity to a specified station or the next station in the station visitation sequence defined for the entity. A delay time to transfer to the next station may be defined. When an entity enters the Route module, its Station attribute (Entity.Station) is set to the destination station. The entity is then sent to the destination station, using the route time specified.

Lastly, we built three modules from the Advanced Process tools, One (1) for each of the modules.

- **Seize Module**
  In Arena Seize means that an entity is seized by a resource (defined in the process module). In simple terms it is like a server seizing a customer in queue when his turn comes.
- **Released Module**
  Release is returning the resource.
- **Delay Module**
  Delay simply indicates that a process delay will be incurred with no resource constraints. Entity just sits here for the specified time; no Resource involved, so multiple entities could be undergoing this Delay simultaneously.

MODEL FLOW

MODEL 1

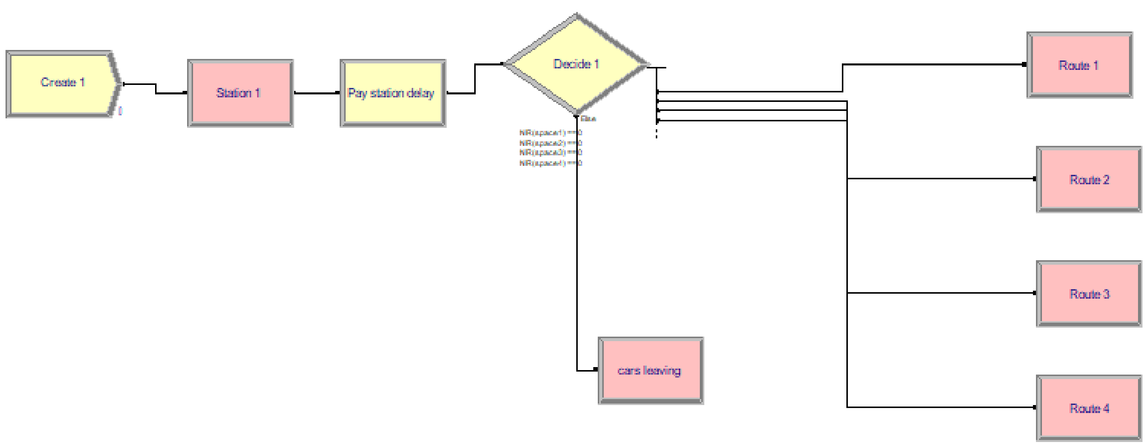


Figure 2 Model-1(Part-A)

Description

- *Create 1:* Creating the entity “car”.
- *Station 1:* Car entering the parking lot.

- *Pay Station Delay:* Entry gate where passenger collects ticket.
  - *Delay Time: 5 minutes*
- *Decide 1:* This module allows for decision-making processes. If Condition is True then Car will be parked in the slot, else entity will leave the parking area.
  - *Conditions: NR(space1) == 0; NR(space2) == 0; NR(space3) == 0; NR(space4) == 0*
- *Route 1,2,3,4:* This is a logical connection to the station. These routes will be connected to the parking slot stations.
  - *Route Time: 5 minutes*
- *Car Leaving:* This route is connected to exit gate station. If the condition is not met in the decision modules, through this route car will exit the gate.

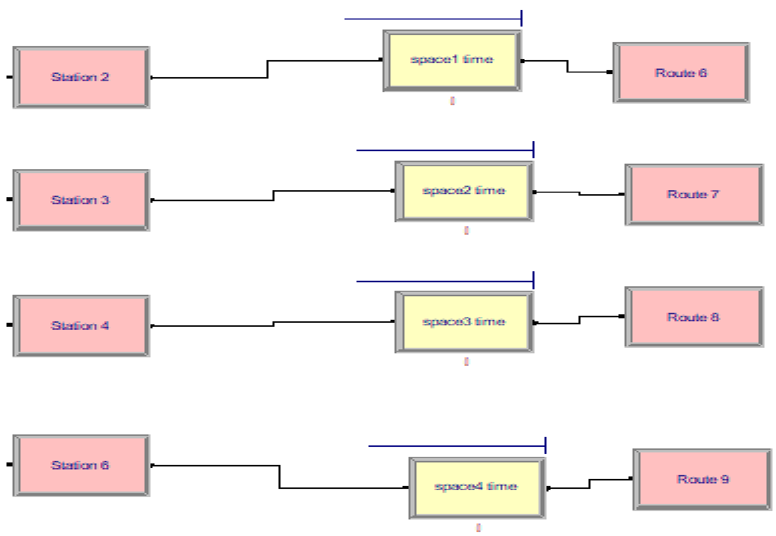


Figure 3 Model-1 (Part-B)

- *Station 2,3,4,6:* These Stations are logical connections to the route modules. Hence entities from respective stations are allocated with resources for stipulated time.
- *Space 1,2,3,4:* If the space is idle then parking space can be utilized else entity has to wait or exit the parking lot.
- *Route 6,7,8,9:* These Routes are connected to the Exit station. The entities are sent to exit station and the disposed.

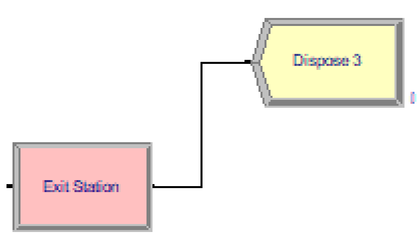


Figure 4 Model-1 (Part-C)

- *Dispose 3:* This is exit point from the parking lot.

MODEL PERFORMANCE

Resource

Resource

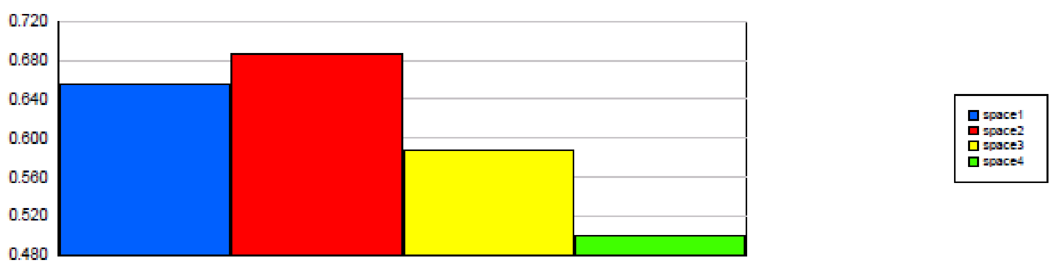
Usage

Instantaneous Utilization	Average	Half Width	Minimum Value	Maximum Value
space1	0.6563	(Insufficient)	0.00	1.0000
space2	0.6875	(Insufficient)	0.00	1.0000
space3	0.5875	(Insufficient)	0.00	1.0000
space4	0.5000	(Insufficient)	0.00	1.0000

Number Busy	Average	Half Width	Minimum Value	Maximum Value
space1	0.6563	(Insufficient)	0.00	1.0000
space2	0.6875	(Insufficient)	0.00	1.0000
space3	0.5875	(Insufficient)	0.00	1.0000
space4	0.5000	(Insufficient)	0.00	1.0000

Number Scheduled	Average	Half Width	Minimum Value	Maximum Value
space1	1.0000	(Insufficient)	1.0000	1.0000
space2	1.0000	(Insufficient)	1.0000	1.0000
space3	1.0000	(Insufficient)	1.0000	1.0000
space4	1.0000	(Insufficient)	1.0000	1.0000

Scheduled Utilization	Value
space1	0.6563
space2	0.6875
space3	0.5875
space4	0.5000



Entity

Entity

Time

VA Time	Average	Half Width	Minimum Value	Maximum Value
Entity 1	0.4167	(Insufficient)	0.00	0.7500

NVA Time	Average	Half Width	Minimum Value	Maximum Value
Entity 1	0.00	(Insufficient)	0.00	0.00

Wait Time	Average	Half Width	Minimum Value	Maximum Value
Entity 1	0.00	(Insufficient)	0.00	0.00

Transfer Time	Average	Half Width	Minimum Value	Maximum Value
Entity 1	0.1874	(Insufficient)	0.1500	0.2500

Other Time	Average	Half Width	Minimum Value	Maximum Value
Entity 1	0.00	(Insufficient)	0.00	0.00

Total Time	Average	Half Width	Minimum Value	Maximum Value
Entity 1	0.6041	(Insufficient)	0.2500	0.9167

MODEL 2

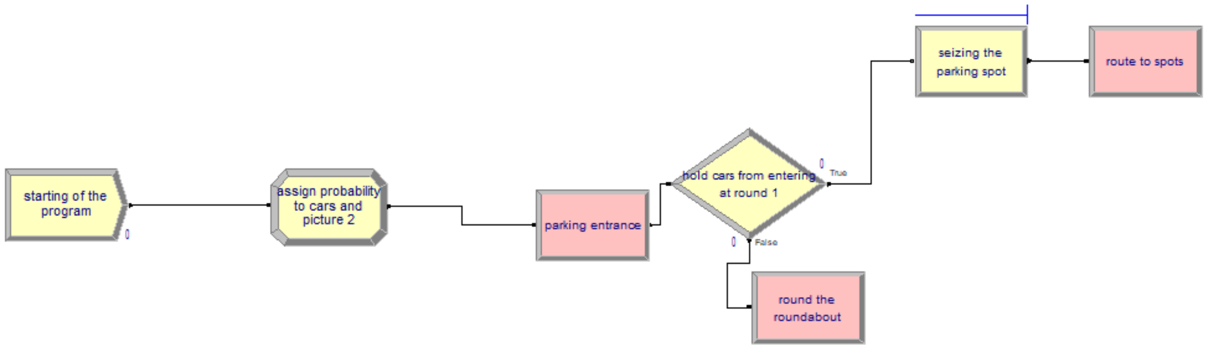


Figure 5 Model-2 (Part-A)

Description

- The entities enter the system and are normally distributed with mean =10 mins and standard deviation=1 min, maximum arrivals = 100.
- The assignment module will assign an attribute to each entity with discrete probability distribution of various colors of cars. The expression is given by cumulative probability,
  - DISC(0.4,1,0.8,2,1,3)
- The Parking entrance station is a logical connection to another route.
- Entity will be processed to Decision module to check if the resource is idle or busy. If all the resources are busy then entity will exit the parking lot else, it will seize the available space. The expression of the condition is given by,
  - SETSUM(parkingspots,5) <> 5
- From the decision module, if the resource is available then seize module will allocate the parking space and through the route it will be connected to the parking spot station.

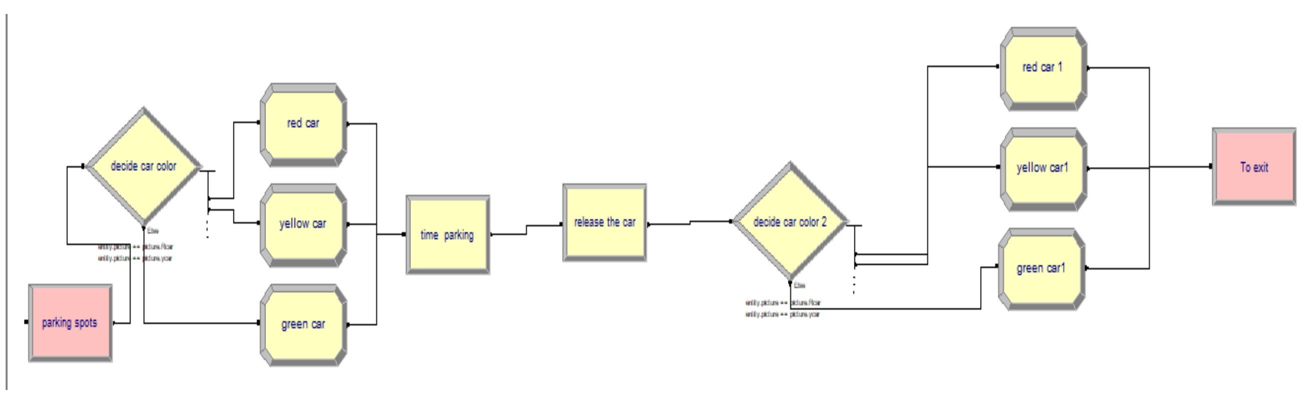


*Figure 6 Model-2 (Part-**B**)*

- Once the car enters the parking spot it will be identified by the attribute and then compares the color in decision module. Parking space will be allocated based on the conditions met.

- Once a car is parked at a specific spot it can only be utilized for 35 minutes. Later car will be released and decided on which respective spot is vacated.

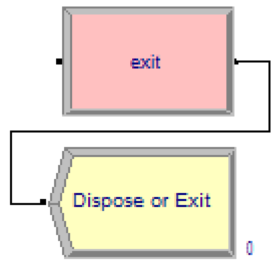


Figure 7 Model-2 (Part-C)

- “To exit” route module will be logically connected to “exit” station and then car leaves the parking lot.

MODEL PERFORMANCE

Resource

Resource

Usage

Instantaneous Utilization	Average	Half Width	Minimum Value	Maximum Value
Spot 1	0.7499	(Insufficient)	0.00	1.0000
Spot 2	0.8104	(Insufficient)	0.00	1.0000
Spot 3	0.7080	(Insufficient)	0.00	1.0000
Spot 4	0.7673	(Insufficient)	0.00	1.0000
Spot 5	0.7500	(Insufficient)	0.00	1.0000

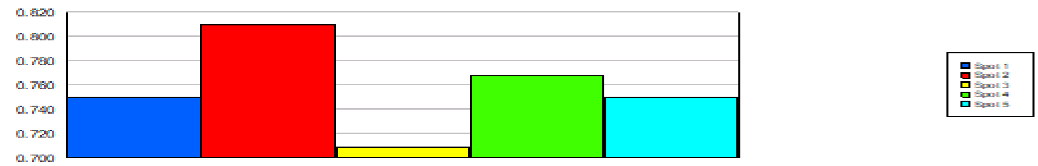
Number Busy	Average	Half Width	Minimum Value	Maximum Value
Spot 1	0.7499	(Insufficient)	0.00	1.0000
Spot 2	0.8104	(Insufficient)	0.00	1.0000
Spot 3	0.7080	(Insufficient)	0.00	1.0000
Spot 4	0.7673	(Insufficient)	0.00	1.0000
Spot 5	0.7500	(Insufficient)	0.00	1.0000

Number Scheduled	Average	Half Width	Minimum Value	Maximum Value
Spot 1	1.0000	(Insufficient)	1.0000	1.0000
Spot 2	1.0000	(Insufficient)	1.0000	1.0000
Spot 3	1.0000	(Insufficient)	1.0000	1.0000
Spot 4	1.0000	(Insufficient)	1.0000	1.0000
Spot 5	1.0000	(Insufficient)	1.0000	1.0000

Usage

Scheduled Utilization

	Value
Spot 1	0.7499
Spot 2	0.8104
Spot 3	0.7080
Spot 4	0.7673
Spot 5	0.7500



Total Number Seized

	Value
Spot 1	9.0000
Spot 2	10.0000
Spot 3	9.0000
Spot 4	10.0000
Spot 5	9.0000

Entity

Entity

Other

Number Out

	Value
Arrival Cars	43.0000
car	0.00

WIP

	Average	Half Width	Minimum Value	Maximum Value
Arrival Cars	3.9648	(Insufficient)	0.00	5.0000
car	0.00	(Insufficient)	0.00	0.00

Queue

Time

Waiting Time

	Average	Half Width	Minimum Value	Maximum Value
seizing the space.Queue	0.00	(Insufficient)	0.00	0.00

Other

Number Waiting

	Average	Half Width	Minimum Value	Maximum Value
seizing the space.Queue	0.00	(Insufficient)	0.00	0.00

CONCLUSION

The two models simulate the operation of car parking system. Two scenarios were considered one with four parking spaces and another one with 5 parking spaces with different color cars and set module. For the first scenario, total 45 cars are out, average utilization is 65% and second scenario it is 43 cars are out, average utilization is 75%. Using this project as prototype we can also analyze using parking meters and expanding the parking spaces near to real world scenarios.

ASSUMPTIONS

1. Model1 Assumptions

The route delays are 5 minutes to each respective space.

We also assumed constant type car space times.

Run setup time is for 8 hours.

2. Model2 Assumptions

Parking time for each car is constant type with 35 minutes.

Distribution for create module is Norm (10,1) distribution with max arrival value is 100.

ANIMATIONS

The *Animate Transfer* toolbar gives you tools to add animation objects to the model

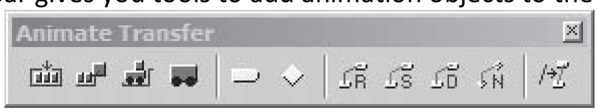


Figure 8 Animate Transfer

These include Storage, Seize, Parking, Transporter, Station, Intersection, Route, Segment, Distance, Network, and Promote Path.
In this project, mostly used route animate for different stations like from entrance to space1, from space1 to exit, etc.

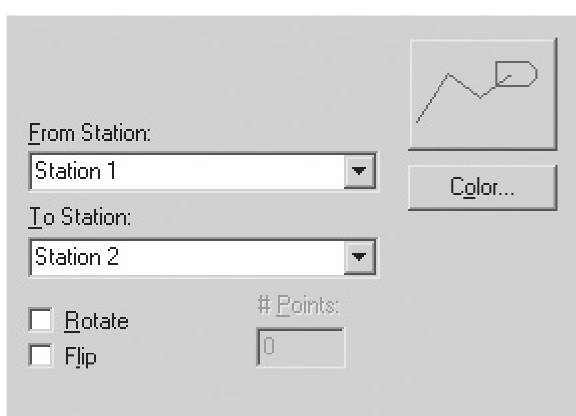


Figure 9 Route Animate

The *Resource Picture* has a collection of general purposes resources pictures like cars, balls, people etc.

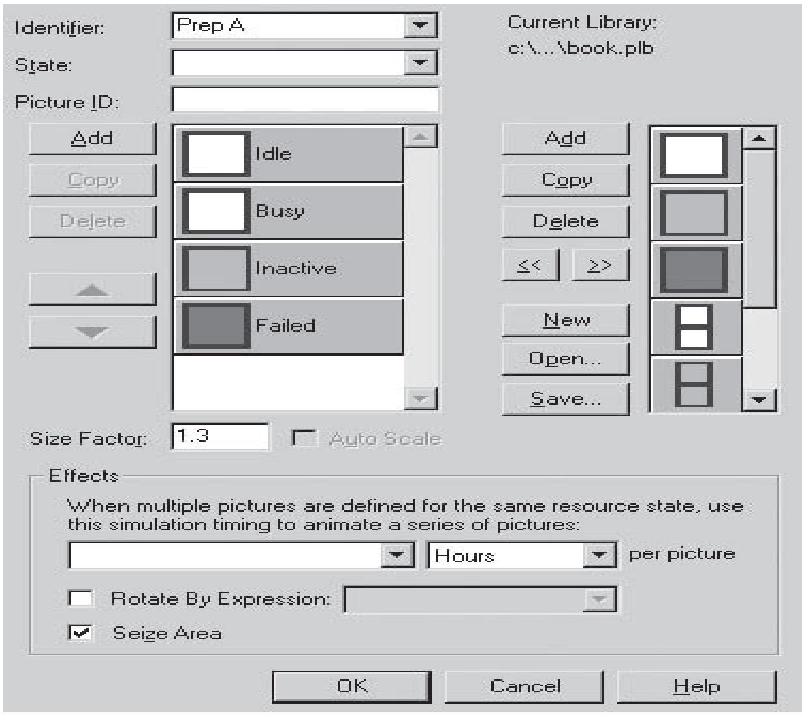


Figure 10 Resource Picture

For both models the base picture is drawn using “Solidworks”

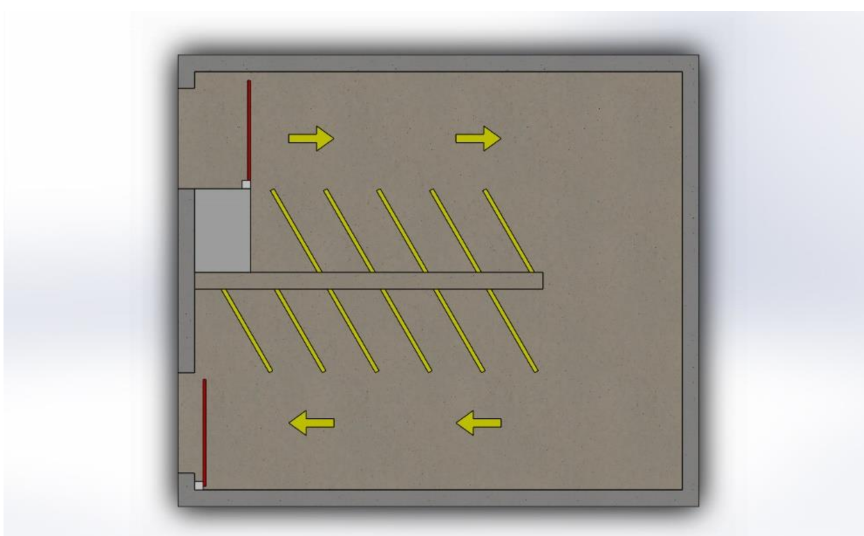


Figure 11 Base Picture

With help of all the above-mentioned animation features in arena simulation build two animated models for both scenarios. Model1 and Model2 animations are same as shown in figure-1 but we used different colored cars for scenario-2.

REFERENCES

Textbook: Simulation with Arena by Kelton, Sadowski, Zupick.

Professor Wu Lecture Notes.

1. Arena Simulation. 2020. Arena Simulation. [ONLINE] Available at: https://www.arenasimulation.com/support. [Accessed 30 April 2020].

2. learn.kent.edu. 2020. No page title. [ONLINE] Available at: https://learn.kent.edu/webapps/blackboard/content/listContent.jsp?course_id=_282215_1&content_id=_10899580_1. [Accessed 30 April 2020].

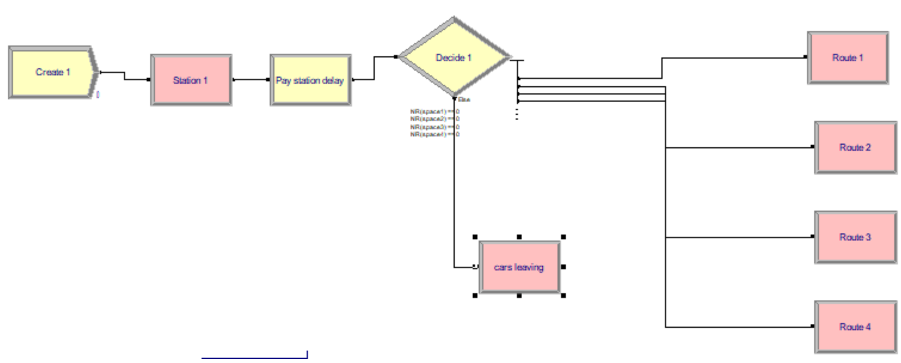
3. Simulation Model of Parking Spaces Through the Example of the Belgorod Agglomeration - ScienceDirect. 2020. Simulation Model of Parking Spaces Through the Example of the Belgorod Agglomeration - ScienceDirect. [ONLINE] Available at: https://www.sciencedirect.com/science/article/pii/S2352146517300194. [Accessed 30 April 2020].

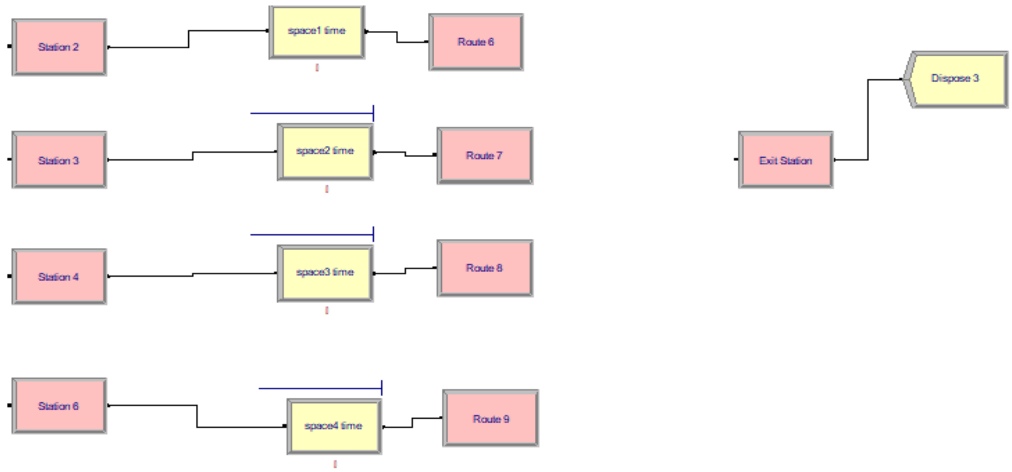
APPENDIX

Arena Models

Model1:

The overall model of parking lot with four parking spaces.





Model2:

The overall model of five parking spaces with three different color cars.