DELHI TECHNOLOGICAL UNIVERSITY

MODELING AND SIMULATION PROJECT

****

Presented by:

Muskan Gupta (2K19/SE/077)

Meedha (2K19/SE/072)

To prof. incharge:

Mr. Rahul

**TITLE:**

VEHICLE SCHEDULING SIMULATION

**INDEX**

1. Acknowledgement
2. Abstract
3. Introduction
4. Conclusion

**ACKNOWLEGDEMENT**

We would like to earnestly acknowledge the sincere efforts and valuable time given by our teacher Mr. Rahul Sir. His valuable guidance and feedback has helped us in completing this project.

We would also like to thank him for providing us with the required material and guidelines and for encouraging us to complete this project.

**ABSTRACT**

This paper discusses simulation of the vehicle schedule with time windows. The simulation model is developed in python using VScode as IDE and using various python libraries like numpy, matplotlib etc. A detailed description of a conceptual model with focus on input and output data is given. The structure of the simulation model and operation of its main blocks are described. Each vehicle is modelled as a separate object used to construct the overall schedule for all vehicles. The simulation model is used as a decision support tool for an analyst, which allows estimating efficiency of vehicle schedules with time windows generated by a standard software or/and modified by a planner.

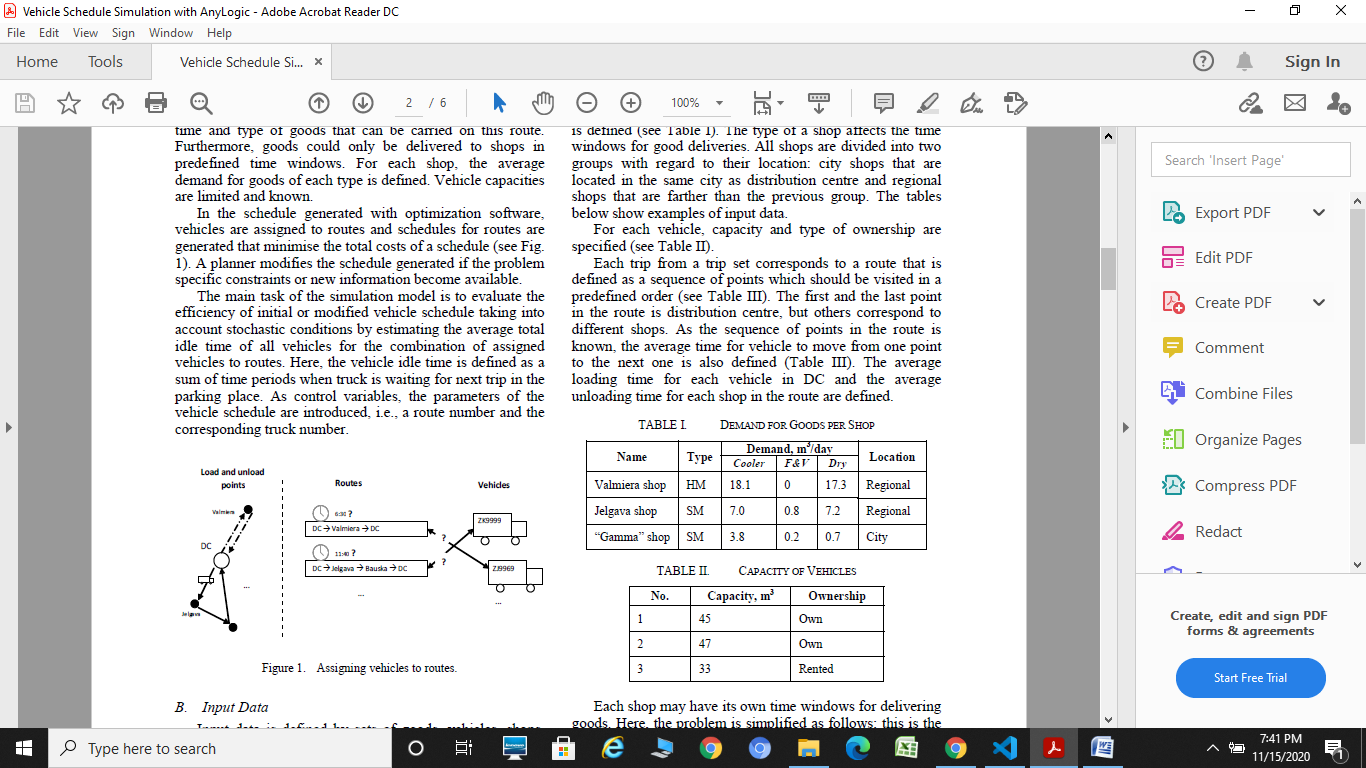
**Overview of the problem**

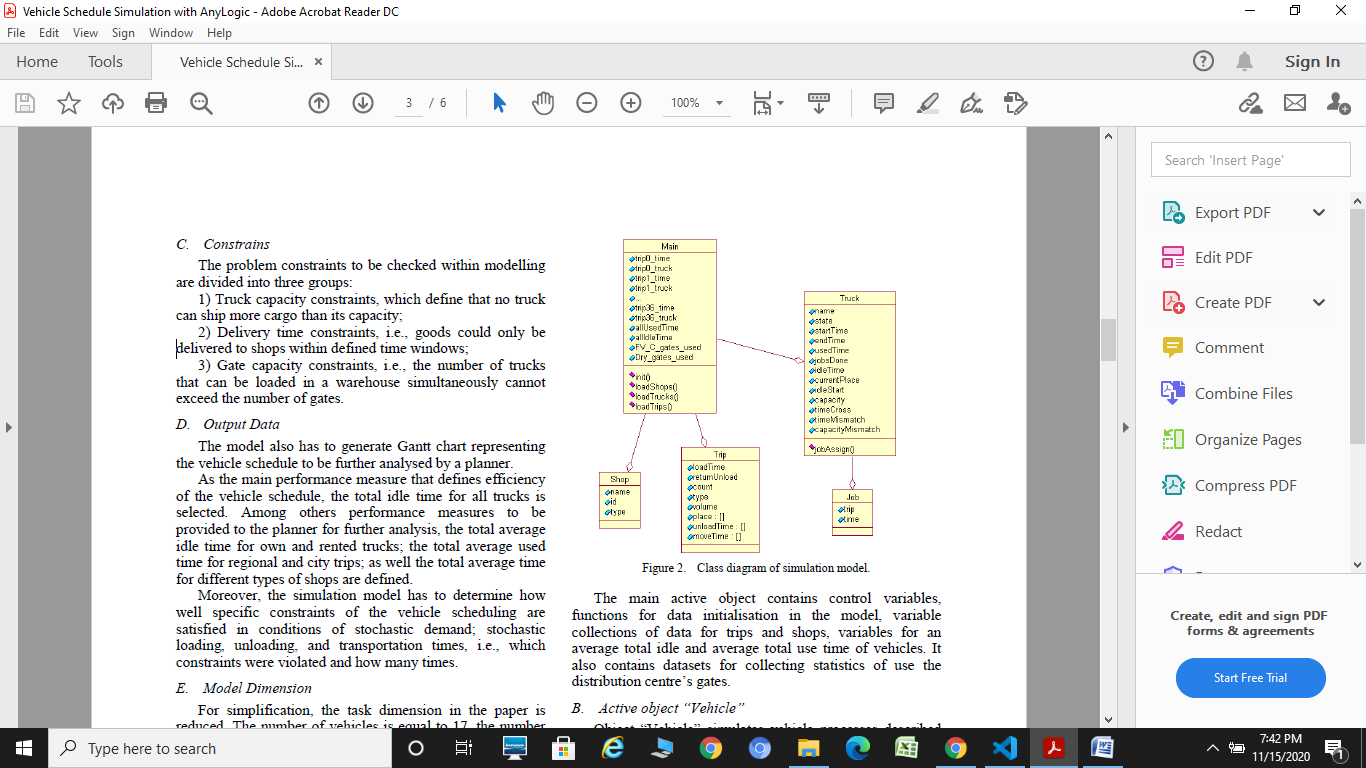
Vehicle scheduling problems (VSP) represent a class of problems aimed at assigning a set of scheduled trips to a set of vehicles, in such a way that each trip is associated with one vehicle, and the cost function for all trips is minimized. This problem is often modified with additional constraints like time windows and different truck capacity. Moreover, in ractice, VSP can also be complicated by stochastic processes existing in the real system, e.g., when the length of a trip is a random variable. In this case evaluation of potential solutions and their comparison can be

done through simulation.

A vehicle schedule defines a schedule of goods deliveries by vehicles (or trucks) from a distribution centre (DC) to a net of shops or supermarkets. The model is aimed to simulate utilization of trucks in order to determine their usage and idle times.

Trucks belong to different groups that have various parameters such as capacity, limited velocity and ownership (own, rented). Distribution routes for vehicles are fixed. For each route, the following parameters are defined: a sequence of shops (route points), average time intervals for vehicle moving between route points, loading and unloading average time and type of goods that can be carried on this route. Furthermore, goods could only be delivered to shops in predefined time windows. For each shop, the average demand for goods of each type is defined. Vehicle capacities are limited and known. A planner modifies the schedule generated if the problem specific constraints or new information become available. The main task of the simulation model is to evaluate the efficiency of initial or modified vehicle schedule taking into account stochastic conditions by estimating the average total idle time of all vehicles for the combination of assigned vehicles to routes. Here, the vehicle idle time is defined as a sum of time periods when truck is waiting for next trip in the parking place. As control variables, the parameters of the vehicle schedule are introduced, i.e., a route number and the corresponding truck number.





**Input Data**

Input Data is defined by a set of trips and constraints. Each trip from a trip set corresponds to a route that is defined as a sequence of points which should be visited in a predefined order. The first and the last point in the route is distribution centre, but others correspond to different shops. As the sequence of points in the route is known, the average time for vehicle to move from one point to the next one is also defined. The average loading time for each vehicle in DC and the average unloading time for each shop in the route are defined.

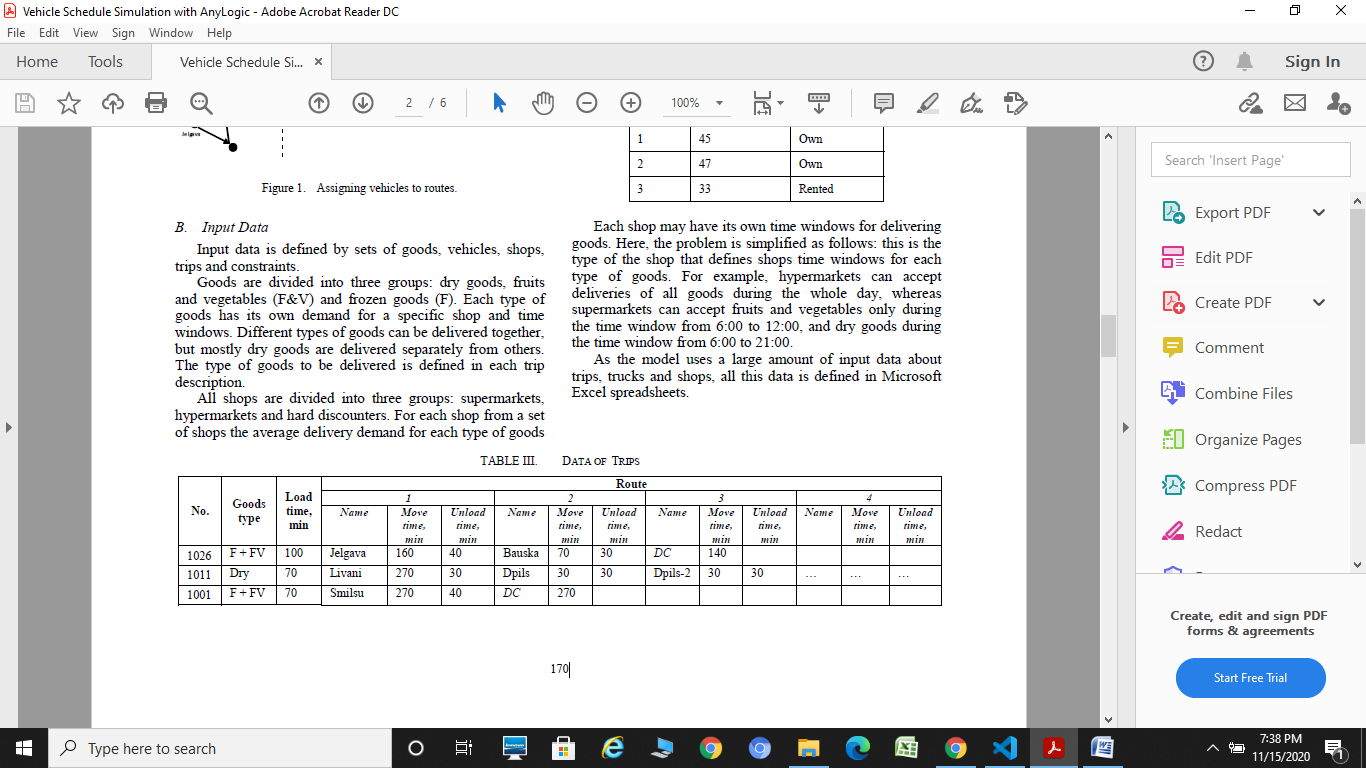
Some constraints can also be included like:-

1) Truck capacity constraints, which define that no truck can ship more cargo than its capacity;

2) Delivery time constraints, i.e., goods could only be delivered to shops within defined time windows;

As the model uses a large amount of input data about trips, trucks and shops, all this data is defined in Microsoft Excel spreadsheets.

A sample data can be described like this



**Output Data**

The model also has to generate Gantt chart representing the vehicle schedule to be further analyzed by a planner.

As the main performance measure that defines efficiency of the vehicle schedule, the total idle time for all trucks is selected. Among others performance measures to be provided to the planner for further analysis, the total average idle time for own and rented trucks; the total average used time for regional and city trips; as well the total average time for different types of shops are defined.

Moreover, the simulation model has to determine how well specific constraints of the vehicle scheduling are satisfied in conditions of stochastic demand; stochastic loading, unloading, and transportation times, i.e., which constraints were violated and how many times.

**Model Dimension**

For simplification, the task dimension in the paper is reduced. The number of vehicles is equal to 6 and the number of shops is equal to 1.

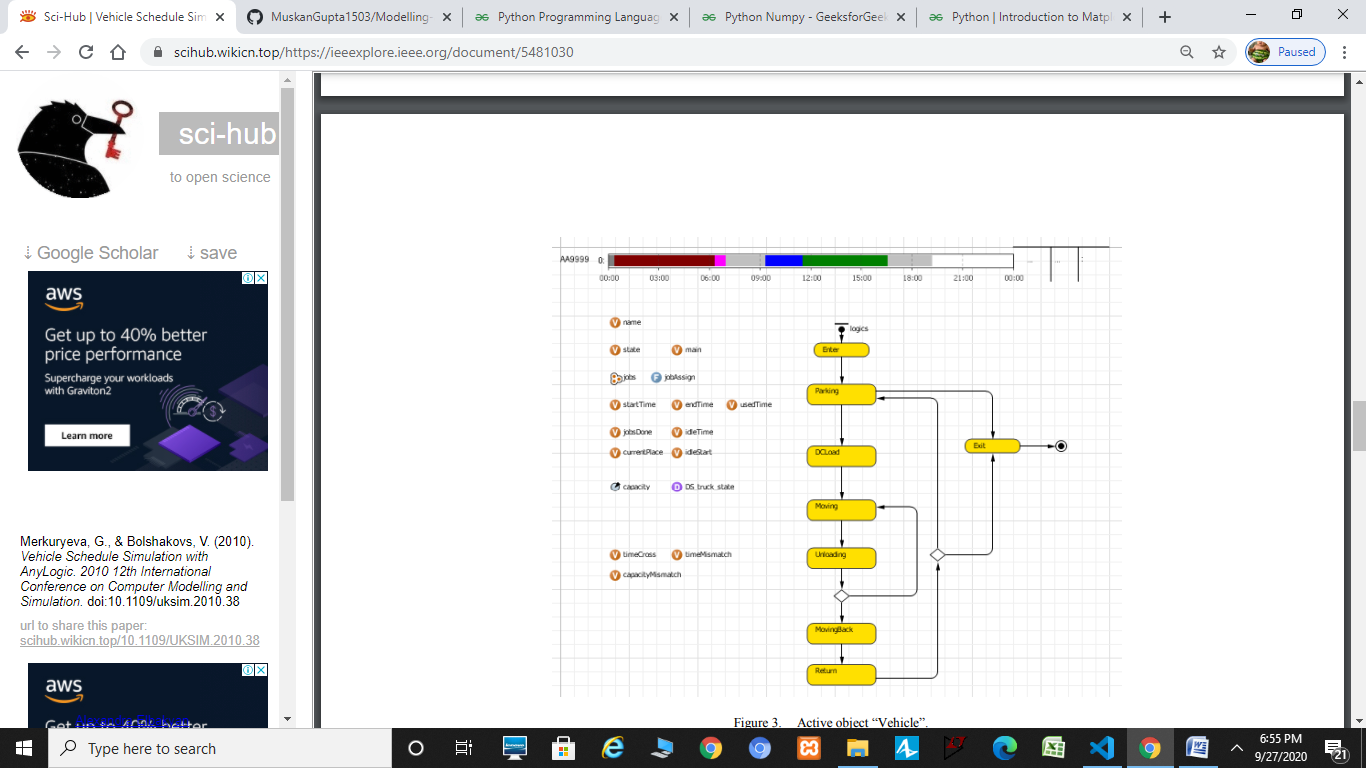
**Different States of Vehicle**

The model is built as a discrete-event simulation model in which each vehicle is modelled as a particular active object. Specific processes related to the DC operations are simulated.

Object “Vehicle” simulates vehicle processes described by a state chart. It defines possible vehicle states and transitions between them.

Object “Vehicle” has variable collection of its job list, variables for accumulation of idle times of vehicles and variables to count the amount of cases in which constraints were not satisfied. The following states are defined in the state chart:

* Enter
* Parking
* Loading
* Moving
* Unloading
* Moving Back
* Return
* Exit



Terminal states, i.e., Enter and Exit, are used for variable initialisation and final calculation, respectively.

Return state is used for some vehicles that have to return empty package (pallets or containers).

Each state has its own entrance and exit actions. In order to store information about the current vehicle state when entering each state of the state chart, variable “State” sets up a value of a new state of a vehicle.

States Moving and Moving Back set the same value Moving to the variable “State”.

Moreover, state Parking does not change an entrance value of variable “State” as two virtual states are proposed for the vehicle in the parking yard: “Idle” and “The truck is out of use”.

The last value is defined on entrance to the states Enter and Exit. In turn, variable “State” is defined with value Idle only after state Return.

Statistics of utilisation of each vehicle over time (value of variable “State”) is collected in dataset and is shown in a timeline chart where different states of vehicle are shown with different colors. Truck capacity constraint mismatch is checked when the trip is assigned. If the total volume of goods in a scheduled trip is greater than vehicle capacity, variable “capacityMismatch” is incremented. Time window constraint is checked when vehicle gets in Unloading state. The current modelling time is compared with time window of the corresponding shop, and if the vehicle arrived at shop out of its time window variable “timeCross” is incremented. Two types of transitions are defined in state chart: timeout and conditional. Timeout transitions are used to hold the vehicle in the corresponding state on the chart. Time period for each timeout transition is taken from data of the corresponding trip. The average values of loading, moving and unloading times are used to define transition timeouts. These timeouts can be modified by using random variables which make schedule more realistic.

Finally, Parking state is defined taking into account starting time of the next trip. If the next trip starts before the vehicle gets in the Parking state, this case is accounted and the variable “timeCross” is incremented. A large value of this variable indicates that vehicles cannot perform the trip without delays. Conditional transition is defined after Unloadling state. If the current trip is complete, a vehicle can move back to the DC or to another shop. Another conditional transition is defined after Return state and checks if there are more trips scheduled for this vehicle.

**Conclusions**

In model screenshot, the Gantt chart for the vehicle schedule is shown. It is combined with the graph of the utilization of DC gates during the daytime. On-line statistics about idle time and completed jobs is provided with timeline of the corresponding vehicle. List of performed trips (respectively, visited shops) is issued for each vehicle.

The visualization of vehicle and distribution centre utilization combined with statistical data can be used by expert to redefine vehicle schedule. For example, it can be easily seen from Screenshots that only few vehicles have idle periods so long (longer than 50 minutes) that they should have been taken into account. Possible transformations of the schedule are also visible, for example, the second trip of the fourth vehicle could be assigned for the third vehicle to prevent long idle period of the fourth vehicle in third screenshot.

This model provides “what if” analysis for further improvement of a vehicle schedule generated with heuristic algorithms. It is also useful for vehicle schedule adjustment for unexpected changes of input data and parameters of vehicle schedule, when heuristics are excessive for regeneration of a new schedule. As all vehicles are simulated separately, simulation model is very flexible in terms of unique parameter definition for each vehicle. Moreover, this model can be applied for simulation-based optimization of vehicle scheduling problem with time windows. In this case the model would provide fast estimation of proposed optimal variants of schedule.

A Statistical Report can also be generated which can provide aggregated information about vehicle utilization and idling, which is not shown in model window.

**Model Building**

To build the model, I leant basics of python and its various Libraries like NumPy, Matplotlib etc. to be able to program the model and implement it.

Secondly, I implemented a sample data on MS Excel and prepared my models and Gantt chart there too.

As the model is to generate Gantt chart, I have prepared the program for creating the Gantt chart in python using Matplotlib library present in python. A Gantt chart is a graphical depiction of a project schedule or task schedule (In OS). It’s is a type of bar chart that shows the start and finish dates of several elements of a project that include resources or deadline.

**Matplotlib** is an amazing visualization library in Python for 2D plots of arrays. Matplotlib is a multi-platform data visualization library built on NumPy arrays and designed to work with the broader SciPy stack. One of the greatest benefits of visualization is that it allows us visual access to huge amounts of data in easily digestible visuals. Matplotlib consists of several plots like line, bar, scatter, histogram etc.

**Numpy**is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays. It is the fundamental package for scientific computing with Python.  
Besides its obvious scientific uses, Numpy can also be used as an efficient multi-dimensional container of generic data.

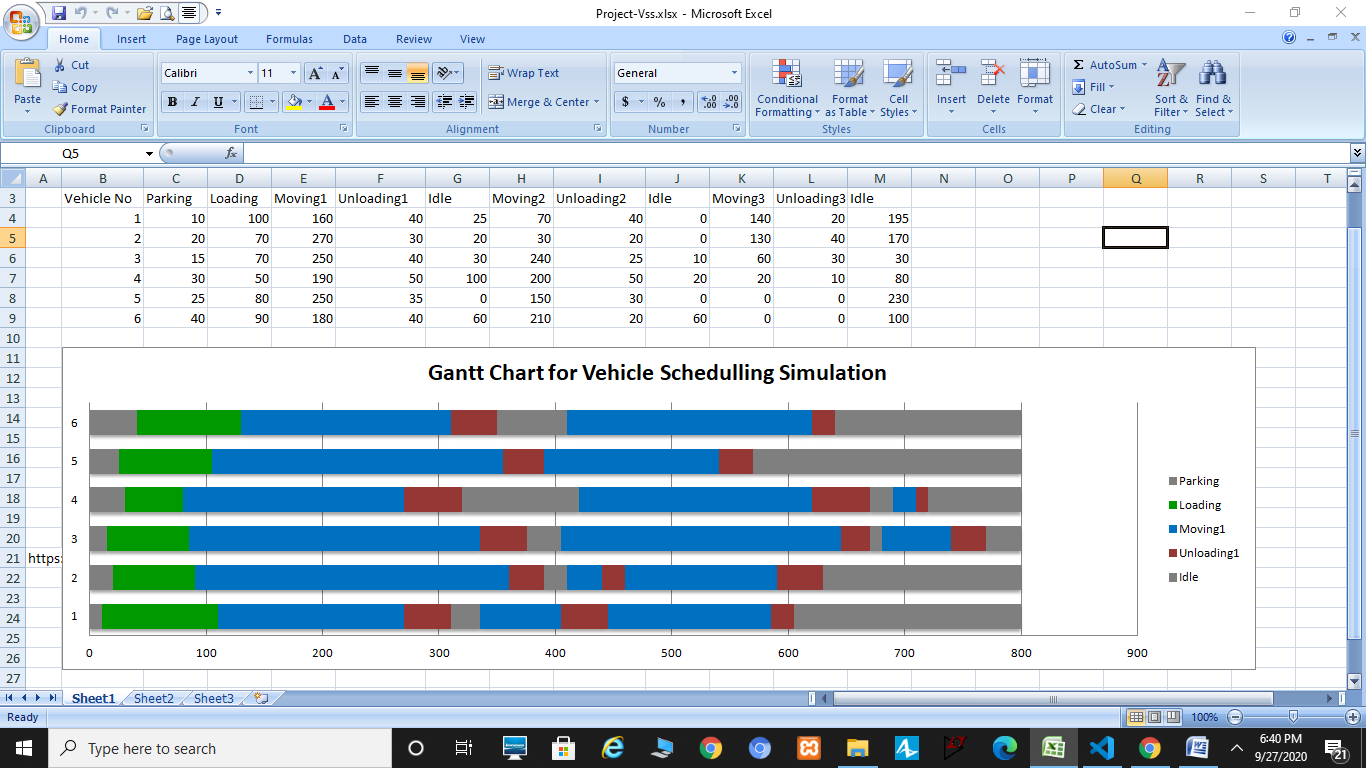
Code for python Program for creating models and Gantt Charts:-

<https://github.com/MuskanGupta1503/Modelling-and-Simulation>

Besides it also contains model implementation in excel sheets as well as python program.

**Screenshots**

Here color codes are

* **Grey- Idle time/ Parking state**
* **Green- Loading time**
* **Maroon/Red- Unloading Time**
* **Blue- Moving Time**
* In Excel, Gantt Charts using excel features. Here the table is present which represents the journey of each of the 6 vehicles having different states like idle, moving, unloading, loading etc. 

Output After Python code Some more screenshots which can be prepared in Anylogic Simulation Software