Simulation Modelling Practice and Theory 49 (2014) 167–179

Contents lists available at ScienceDirect

Simulation Modelling Practice and Theory

journal homepage: www.elsevier.com/locate/simpat

Traffic simulation of two adjacent unsignalized T-junctions during rush hours using Arena software

Mohsen Kamrani, Sayyed Mohsen Hashemi Esmaeil Abadi, Saeed Rahimpour Golroudbary *

Department of Materials, Manufacturing and Industrial Engineering, Faculty of Mechanical Engineering, Universiti Teknologi Malaysia (UTM), Skudai 81310, Johor, Malaysia

ARTICLE INFO

Article history:
Received 31 March 2014
Received in revised form 18 September 2014
Accepted 20 September 2014
Available online 10 October 2014

Keywords:
Traffic simulation
Arena
Junction

ABSTRACT

In this paper, the focus is on simulating the traffic of two adjacent T-junctions during rush hours located at Jalan Universiti in the city of Skudai, Johor, Malaysia. This study was conducted with the objective of simulating the traffic on the network in order to understand and analyze its bottlenecks and propose solutions to improve it. The simulation model was developed with ARENA software, and the initial result shows that there is a substantial queue in one of the routes, arm C. A model with traffic light was proposed to tackle the problem. Results obtained from the improved model revealed that the average waiting time in arm C declined by 67%. Furthermore, the average waiting time of the queues in the entire system decreased by 53%. In addition, in this paper, it was shown how Arena software can be adopted to simulate traffic problems effectively. The method in this research can be applied to investigate various traffic scenarios and their consequences before implementing them in reality.

1. Introduction

1.1. Motivation

Traffic control and optimization have received special attention in recent years. This is due to the increasing number of vehicles and demand for accommodation and handling the traffic volume. Undoubtedly, any attempt to optimize the traffic flow will be rewarded by reduced expenses in terms of pollution, time, etc.

When it comes to expenses, the approach of optimization becomes significant. There have been many costly projects in order to improve the traffic flow, which eventually did not yield expected results. Therefore, the motivation of this research was to solve a local traffic flow problem by identifying its causes and suggesting solutions without incurring considerable operational expenses.

Thus, Arena software was chosen to simulate the situation due to its availability for wide range of researchers. Another reason for this choice is the software versatility so that it can be used in other areas than manufacturing.

1.2. Objectives

The objectives of this paper are as follows:

- To simulate the traffic of the network in order to understand and analyze the problems.

* Corresponding author. Tel.: +60 197128040.
E-mail address: Rahimpour.Saeed@gmail.com (S. Rahimpour Golroudbary).

http://dx.doi.org/10.1016/j.simpat.2014.09.006
1569-190X/

- To estimate the waiting times of the queues that can be attributed to the drivers satisfaction.
- To examine the effectiveness of the current system by obtaining and analyzing the related statistical data obtained from the simulation.
- To observe the effect of changes on the current network in order to improve it.
- To propose solutions to increase the effectiveness of the network.
- To show how different logics and behaviors of traffic at the microscopic level can be simulated in Arena software.

2. Literature review

In recent years, tackling the problems regarding traffic flow has received considerable attention. This is because of the fact that the huge number of vehicles makes various problems such as traffic jams, air pollution, and fuel waste [1]. In attempting to analyze traffic networks and mitigate pertaining problems, also due to the complexity of the traffic systems, simulation approach has become a common method. Another consideration in this discipline is the selection of the appropriate software so that some organizations conduct research for selection of their simulation software. For instance, [2] did a research for an international airport in Spain whether to choose between Arena or Witness as their simulation platform using Analytical Hierarchy Process.

Among different simulation software, Arena has been applied by several researchers in various disciplines. For example, [3] developed a simulation model with Arena to analyze and design the seaport for the management of the Malaysian Kelang Port. [4] used Arena simulation model to ensure whether the barge routings proposed by an optimization program are feasible during river operations. A simulation model of an automated guided vehicle system (AGVs) was established by [5] considering Just in Time philosophy. Having run experimental design, they investigated the applicability of the JIT concept to AGVs in job shop environments. [6] by employing Arena, studied a multi-product job shop problem with transportation queue disciplines in order to analyze the effect of splitting order quantities into equal sub-lots. Arena was applied by [7] to model human being as a reliability system in the health care industry in a macroscopic way.

Moreover, Arena has been widely used to simulate various traffic modes. [8] developed an Arena simulation model of traffic for vessels in Delaware River. They investigated different scenarios considering terminal properties and navigational rules. [9] simulated the freight system in Port of Seville using Arena. Their work included docks and basins of the port associated with logistics activities. Having run different scenarios, they concluded that current infrastructures of the port are sufficient to handle the logistic flows.

A railway network in Newcastle is modeled by [10] using Arena for investigating the current behavior of the system, utilization and analyzing alternatives. The results were used to design the freight network of the city. [11] employed discrete-event based simulation models to evaluate the utilization level of a rail freight route by using Arena software.

When it comes to road traffic network, Arena has been used by researchers to evaluate and improve traffic flows and investigate regarding scenarios. According to [12] traffic control is an important element of the safety of both pedestrians and vehicles. [13] carried out a simulation of Simple Network Management Protocols in Arena for estimation of the traffic flow parameters.

[14] presented a simulation methodology of intersection to help traffic designers who are willing to apply the simulation-before-construction approach. By introducing an intersection example, their research is considered to provide an appropriate guideline for traffic designers to simulate their projects before the construction. [15] established a model to simulate the traffic flow in a construction of an unsignalized T-junction. Their simulation model was based on Markov chain Monte Carlo approach. [16] employed Arena to compare a proposed signal controller with the fuzzy logic one in signalized intersections. The paper verified that the performance of the suggested model is better than the fuzzy logic controller in the case of high-traffic volume. Arena was adopted by [17] for simulation of a traffic signal system, and it was shown how Arena is capable of simulating the traffic systems in improvement of traffic flow in intersections.

Considering above-mentioned projects, simulation models are widely used to analyze the traffic networks in different modes of transportation and various general or specialized simulation packages have been employed. In this regard, previous researchers have confirmed that reliable results can be obtained via utilization of Arena in the mentioned discipline. It is worth mentioning that there have been scarce researches in the road sector. However, this fact does not make using Arena in simulation of the road traffic unjustifiable, because the capability of Arena has been already shown by previous works in the field of transportation. Therefore, Arena software was chosen to fulfill the objectives of this paper.

3. Description of the case study

In this paper, the focus is on simulating the traffic of two adjacent T-junctions during rush hours located at Jalan Universiti in the city of Skudai, Johor, Malaysia (see Fig. 1).

Both junctions are uncontrolled intersections, which means there is no signal light to rule the junctions. Logically, the priority has been given to vehicles in the main road, Jalan Universiti, over the ones who want to join the main road from the branches (hereafter arms). Fig. 2 shows the flow of the traffic and the way each lane has been named in this paper.

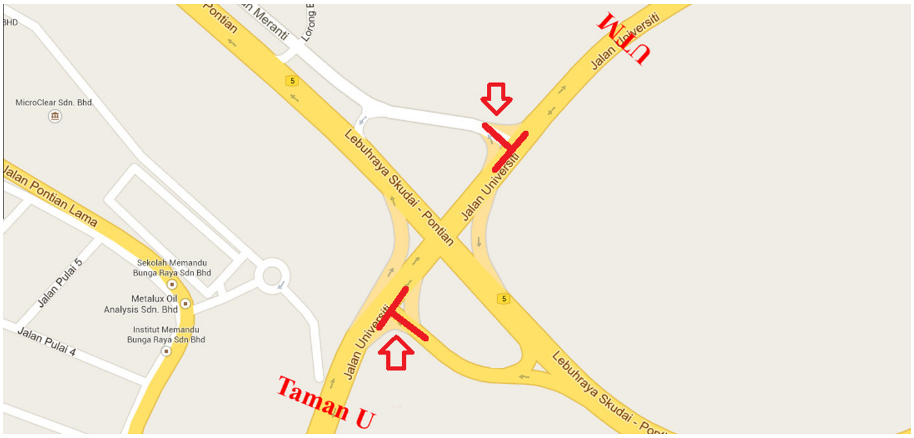


Fig. 1. Overview of the case study from Google.

The problem with mentioned intersections can be classified into two categories: first, it is related to the service level that intersections provide for the drivers, and the second is with regard to the safety of the intersections.

There is usually a queue in arm C because of the priority given to the main stream of the road, i.e. lanes A and B. In addition, based on the observations, the length of the queue varies depending on the traffic volume. This problem exacerbates in special occasions such as celebrations and rush hours. Consequently, drivers in arm C could be dissatisfied with waiting in a relatively long queue. In addition, because of not having a signal light at the intersection, the traffic department has to assign an officer to direct the traffic in rush hours and special occasions.

Moreover, there are always accident hazards in the intersections. Because of the priority given to Lanes A, B, E and D, the drivers in these lanes drive with relatively high speeds. Drivers coming from arms F and C, have to stop even if there is no car coming from the main roads. They need to check and ensure whether it is safe to drive through the junction. The level of danger is dependent on the level of drivers' risk taking. Drivers with risky behaviors consider lower safe distance from coming cars in the main road and they impose more danger to the intersections. Ironically, drivers with lower risky behaviors make the length of the queues (arms F and C) longer.

4. Model development

The mentioned system was simulated in Arena software in order to analyze its significant outputs, i.e. waiting times and queues length. The layout of the model has been shown in Fig. 3. In the following, the model assumptions and elements on which it has been based, will be discussed.

Fig. 2. Traffic flow between the junctions from Google.

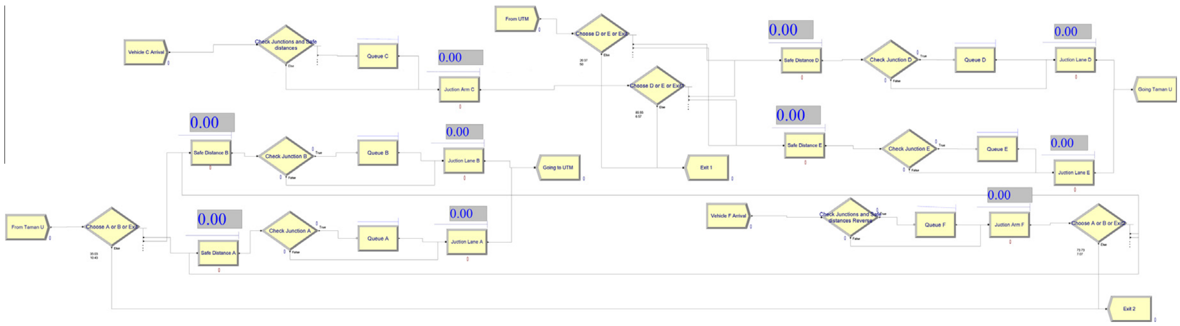


Fig. 3. Simulation model by using Arena.

4.1. Assumptions of the simulation model

The model has been constructed based on the following assumptions:

- There is no jockeying in the system.
- Vehicles do not leave the system after entering the queues.
- No vehicle stops unless the junction is occupied.
- No interruption occurs to the traffic flow because of accidents and breakdowns.

4.2. The proposed model and its elements

There are four entry routes in the system: vehicle arriving from UTM, Taman U, arms C and F. CREATE module was used to simulate the vehicles arrival to the system.

As it was discussed earlier, the drivers in arm C stop at the junction to ensure if it is safe to cross the junction. To do so, they consider a distance so that there is no car coming from lanes A and B. This distance is called Safe Distance in this paper and is simulated by using PROCESS modules. The same thing applies to arm F, where drivers consider a safe distance in lanes D and E to cross the junction. Moreover, when it comes to checking the safe distance by drivers, DECIDE module with related checking codes and rules was applied. Once the drivers passed the junction, they should decide which lane to drive. Thus, another DECIDE module is allocated, which divides the vehicles into different lanes according to the collected data.

As it was already mentioned, lanes A, B, D and E have priority to drive through the junction over arms C and F. Hence, they will drive through the junction except the times that there is a chain of passing vehicles through the junction from arms C or F. To simulate this, HOLD module was applied to scan the mentioned conditions. The discipline of the queues in Hold modules is First In, First Out and their capacity are the infinitive. Finally, DISPOSE modules were assigned when vehicles reach the boundaries of the network and leave it. In the following, more details about the codes and rules in each element of the model is provided.

4.2.1. Create modules

In the model, four different vehicle arrivals were represented by Create modules. The inter arrival times and function of each entity are listed in Table 1.

Table 1
List of the entities.

Name	Entity type	Expression (s)	Function
Vehicle C Arrival	Vehicle C	LOGN(5.76, 5.25)	It creates the cars supposed to travel from arm C to lanes D and E or exit the system
From Taman U	Vehicle From Taman U	LOGN(2.18, 2.07)	It creates the cars supposed to travel from Taman U to lanes A and B or exit the system
From UTM	Vehicle From UTM	LOGN(6.23, 7.36)	It creates the cars supposed to travel from arm C to lanes D and E or exit the system
Vehicle F Arrival	Vehicle F	0.999 + WEIB(22.5, 0.84)	It creates the cars supposed to travel from arm F to lanes A and B or exit the system

Table 2
List of processes.

Name	Capacity-discipline	Action	Delay type	Expression (s)
Safe Distance B	10 – FIFO	Seize Delay Release	Expression	1.5 + 4* BETA(1.67, 2.9)
Safe Distance A	10 – FIFO	Seize Delay Release	Normal	5.05, 0.75
Safe Distance D	10 – FIFO	Seize Delay Release	Normal	5.72, 0.94
Safe Distance E	10 – FIFO	Seize Delay Release	Normal	5.44, 1.05
Junction Arm C	1 – FIFO	Seize Delay Release	Constant	2
Junction Lane B	1 – FIFO	Seize Delay Release	Constant	0.5
Junction Lane A	1 – FIFO	Seize Delay Release	Constant	0.5
Junction Lane D	1 – FIFO	Seize Delay Release	Constant	0.5
Junction Lane E	1 – FIFO	Seize Delay Release	Constant	0.5
Junction Arm F	1 – FIFO	Seize Delay Release	Constant	2.5

4.2.2. Process modules

Process modules are used to model the safe distances and junctions. Table 2 provides details of all Process modules in the model of Fig. 3. The assumption is that there are resources, which provide service for the processes. It is assumed that drivers consider a length of 100 m as the safety distance approximately. Thus, a limit of 10 vehicles (10 vehicles with the distances between them while moving would be 100 m approximately) was put to the each resource of the process modules as their capacity. The capacity of remaining process modules was set to 1 because they have the role of junctions which can accommodate one car at any time.

4.2.3. Decide modules

All Decide modules and their corresponding ruling conditions are listed in Table 3.

4.2.4. Hold modules

In model development section, it was discussed how Hold module scans the condition whether a vehicle is crossing or not. If there is a vehicle, it will stop the drivers from lanes A, B, D and E, otherwise it will allow them to pass. In the model (see Fig. 3), there is a parallel line to Hold modules (queues) that pass vehicles directly to go through junction when there is no vehicle at crossing junction and their corresponding queue lines. Moreover, the junction space does not allow more than one car at the junction itself. Therefore, a component (Hold module) is needed, which should be capable of holding the waiting vehicles, and waiting for a condition to release them. This happens for all vehicles driving to their respective junctions. The Hold modules used in the model and their related conditions are shown in Table 4.

Table 3
List of Decide modules and conditions.

<table>
<tr><th>Name</th><th>Type</th><th>If/Add...</th><th>Expression</th></tr>
<tr><td>Check Junctions and Safe distances</td><td>N-way by condition</td><td>Expression</td><td>(Juction Lane B.WIP == 1 || Juction Lane A.WIP == 1 || Juction Arm C.WIP == 1) || (Juction Lane B.WIP == 0 || Juction Lane A.WIP == 0 && NQ(Queue C.Queue) > 0) || (Safe Distance A.WIP >= 1 || Safe Distance B.WIP >= 1)</td></tr>
<tr><td>Check Junction B</td><td>N-way by condition</td><td>Expression</td><td>(Juction Arm C.WIP == 1 || Juction Lane B.WIP == 1) || ((Juction Arm C.WIP == 0 || Juction Lane B.WIP == 0) && (NQ(Queue B.Queue) > 0))</td></tr>
<tr><td>Check Junction A</td><td>N-way by condition</td><td>Expression</td><td>(Juction Arm C.WIP == 1 || Juction Lane A.WIP == 1) || ((Juction Arm C.WIP == 0 || Juction Lane A.WIP == 0) && (NQ(Queue A.Queue) > 0))</td></tr>
<tr><td>Choose A or B or Exit</td><td>N-way by Chance</td><td>35.03, 10.43</td><td>N/A</td></tr>
<tr><td>Choose D or E or Exit</td><td>N-way by Chance</td><td>20.37, 50</td><td>N/A</td></tr>
<tr><td>Choose D or E or Exit1</td><td>N-way by Chance</td><td>85.93, 6.57</td><td>N/A</td></tr>
<tr><td>Check Junctions and Safe distances Reverse</td><td>N-way by condition</td><td>Expression</td><td>(Juction Lane E.WIP == 1 || Juction Lane D.WIP == 1 || Juction Arm F.WIP == 1) || (Juction Lane E.WIP == 0 || Juction Lane D.WIP == 0 && NQ(Queue F.Queue) > 0) || (Safe Distance D.WIP >=1 || Safe Distance E.WIP >= 1)</td></tr>
<tr><td>Check Junction D</td><td>N-way by condition</td><td>Expression</td><td>(Juction Arm F.WIP == 1 || Juction Lane D.WIP == 1) || ((Juction Arm F.WIP == 0 || Juction Lane D.WIP == 0) && (NQ(Queue D.Queue)) > 0)</td></tr>
<tr><td>Check Junction E</td><td>N-way by condition</td><td>Expression</td><td>(Juction Arm F.WIP == 1 || Juction Lane E.WIP == 1) || ((Juction Arm F.WIP == 0 || Juction Lane E.WIP == 0) && (NQ(Queue E.Queue)) > 0)</td></tr>
<tr><td>Choose A or B or Exit2</td><td>N-way by Chance</td><td>73.73, 7.07</td><td></td></tr>
</table>

4.2.5. Variables

As it was mentioned before, drivers need to check the junctions and safe distances to make sure if there is not a vehicle to pass safely. To count the number of existing vehicles in the model, the WIP variable was defined. The list of WIP variables is listed as follows:

Safe Distance B.WIP	Safe Distance D.WIP
Safe Distance A.WIP	Safe Distance E.WIP
Junction Arm C.WIP	Junction Lane D.WIP
Junction Lane B.WIP	Junction Lane E.WIP
Junction Lane A.WIP	Junction Arm F.WIP

4.2.6. Dispose modules

To simulate the departure of vehicles from the system, Dispose modules were used as follows:
Exit 1, Exit 2, Going UTM, Going Taman U.

5. Data collection and distribution fitting

The next step was collecting required data for the model. All data were collected from 11:00 am to 1:00 pm (based on the observations, this period is the rush hours of the junctions. Thus, by solving the problem in this period, the network problem will be tackled completely) during one week. For distribution fitting, the Input Analyzer tool in Arena was utilized. In Table 5, extended information on data and distribution fitting is provided.

6. Run length and number of replications

Initially, the system is empty, which means there is no vehicle in the model and all resources are idle. This is not in harmony with real condition. Therefore, it is necessary to consider the results after the moment system reaches the steady state (this is called the initial data deletion). To do so, waiting time in arm F was selected as the target parameter to identify the warm-up period. Arm F was selected for this purpose due to its stable and robust behavior in the system. The model was run for 10 h and 5 replications initially, and the results were plotted in Fig. 4. The vertical dashed line shows the moment that the system becomes steady. Therefore, warm-up period was determined 5 h. Since the model should be run for 2 h in steady-state phase, the length of the experiment would be 7 h i.e. warm up plus the simulation period.

To calculate the sufficient number of replications, the below formula [18,19] was used.

$$N_m = \left(\frac{s(m) \times t_{m-1,1-\frac{\alpha}{2}}}{\bar{x}(m)\epsilon} \right)^2 \tag{1}$$

where N_m is the number of replications, $s(m)$ is the data standard deviation, t is the test statistic obtained from t-table, m is the number of initial replications that was assumed to be 5, α is the confidence interval as 90%, $\bar{x}(m)$ is the data mean and ϵ is the allowable percentage error. The allowable error percentage of 10% with $t_{4,0.95}$ equals 2.132. Table 6 shows calculations based on 5 replications. The N_m represents the necessary number of replications as 8.521.

Consequently, the model was run for 8 replications and by putting the results in the formula 1 again, the new number of replications equals 6.96, which verifies that 8 replications were already sufficient. Table 7 shows the result summary.

7. Verification and validation of the model

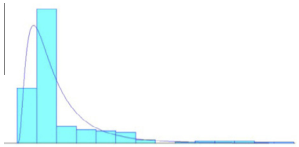
Verification is about constructing the model correctly. It draws a comparison of the conceptual model with the software representation that implements that conception. Having developed the model, its behavior was checked visually, in order to be reasonably similar to real-world condition.

Table 4
List of holds.

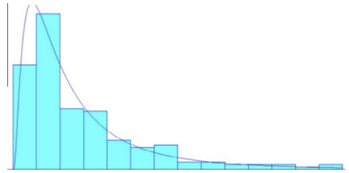
Name	Type	Condition
Queue C	Scan for condition	Juction Arm C.WIP == 0 && Juction Lane B.WIP == 0 && Juction Lane A.WIP == 0 && Safe Distance A.WIP == 0 && Safe Distance B.WIP == 0
Queue B	Scan for condition	Juction Lane B.WIP == 0 && Juction Arm C.WIP == 0
Queue A	Scan for condition	Juction Arm C.WIP == 0 && Juction Lane A.WIP == 0
Queue D	Scan for condition	Juction Arm F.WIP == 0 && Juction Lane D.WIP == 0
Queue E	Scan for condition	Juction Lane E.WIP == 0 && Juction Arm F.WIP == 0
Queue F	Scan for condition	Juction Arm F.WIP == 0 && Juction Lane E.WIP == 0 && Juction Lane D.WIP == 0 && Safe Distance D.WIP == 0 && Safe Distance E.WIP == 0

Table 5
Distribution fitting details of collected data.

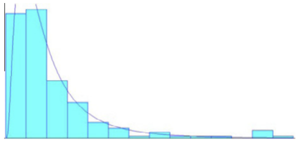
Arm C IAT distribution summary
Distribution: LOGN(5.76, 5.25)
Square error: 0.049316
Number of data points = 207
Sample mean = 6.06
Sample std dev = 6.5
Chi square test
Degrees of freedom = 2
Test statistic = 43.8
Corresponding *p*-value = 0.0973
Kolmogorov–Smirnov test
Test statistic = 0.181
Corresponding *p*-value > 0.15



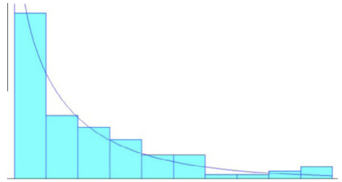
From UTM IAT distribution summary
Distribution: LOGN(6.23, 7.36)
Square error: 0.005221
Number of data points = 202
Sample mean = 6.02
Sample std dev = 5.63
Chi square test
Degrees of freedom = 4
Test statistic = 6.57
Corresponding *p*-value = 0.176
Kolmogorov–Smirnov test
Test statistic = 0.04236
Corresponding *p*-value > 0.15



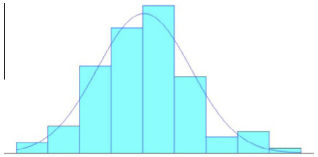
Taman U IAT distribution summary
Distribution: LOGN(2.18, 2.07)
Square error: 0.002302
Number of data points = 200
Sample mean = 2.26
Sample std dev = 2.4
Chi square test
Degrees of freedom = 2
Test statistic = 1.86
Corresponding *p*-value = 0.414
Kolmogorov–Smirnov test
Test statistic = 0.0721
Corresponding *p*-value >0.15



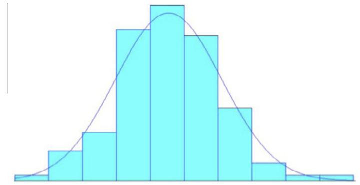
Arm F IAT distribution summary
Distribution: 0.999 + WEIB(22.5, 0.84)
Square error: 0.003802
Number of data points = 100
Sample mean = 25.5
Sample std dev = 26.4
Chi square test
Degrees of freedom = 1
Test statistic = 3.2
Corresponding *p*-value = 0.0783
Kolmogorov–Smirnov test
Test statistic = 0.0753
Corresponding *p*-value > 0.15



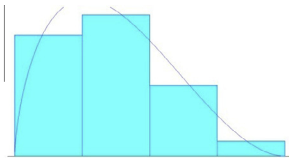
Safe distance A distribution summary
Distribution: NORM(5.05, 0.75)
Square error: 0.003704
Number of data points = 95
Sample mean = 5.05
Sample std dev = 0.754
Chi square
Test degrees of freedom = 1
Test statistic = 0.84
Corresponding *p*-value = 0.389
Kolmogorov–Smirnov test
Test statistic = 0.0912
Corresponding *p*-value > 0.15



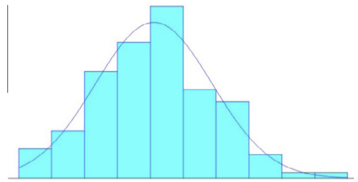
Safe distance D distribution summary
Distribution: NORM(5.72, 0.94)
Square error: 0.002913
Number of data points = 109
Sample mean = 5.72
Sample std dev = 0.945
Chi square test
Degrees of freedom = 2
Test statistic = 1.58
Corresponding *p*-value = 0.465
Kolmogorov–Smirnov test
Test statistic = 0.0568
Corresponding *p*-value > 0.15



Safe distance B distribution summary
Distribution: 1.5 + 4* BETA(1.67, 2.9)
Square error: 0.000546
Number of data points = 69
Sample mean = 2.94
Sample std dev = 0.856
Chi square test
Degrees of freedom = 0
Test statistic = 0.261
Corresponding *p*-value = 0.0651



Safe distance E distribution summary
Distribution: NORM(5.44, 1.05)
Square error: 0.003169
Number of data points = 117
Sample mean = 5.44
Sample std dev = 1.06
Chi square test
Degrees of freedom = 3
Test statistic = 2.41



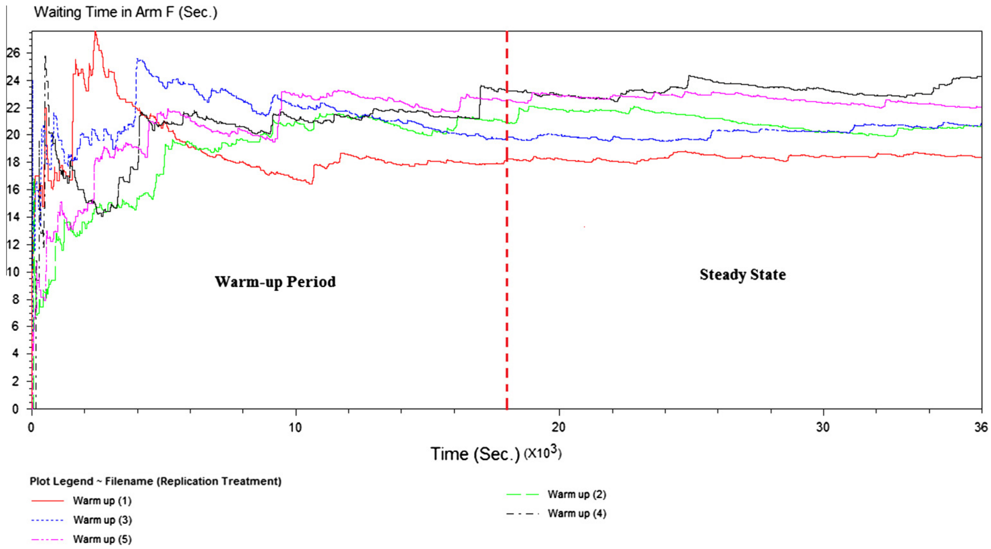


Fig. 4. Specifying warm-up period based on average waiting time of arm F.

Validation is concerned with constructing the correct model. In this phase, usually a steady output of the model is considered to be compared with its real value obtained from data collection. To do so, the average waiting time in arm F was selected. Obtained from 8 replications, the average is 21.51 s. The observed average waiting time for arm F is 23.70 s. Thus, based on the following calculation, the percentage error of the model is 9.24% and considering 90% level of confidence the model is valid.

$$\frac{Average\ Waiting\ Time\ Real\text{-}Average\ Waiting\ Time\ Model}{Average\ Waiting\ time\ Real} = \frac{23.70 - 21.51}{23.70} \times 100 = 9.24\%$$

8. Analysis of the initial model

Having run the model, the average waiting time and the average number of vehicles in the queues for different routes were retrieved and shown in Tables 8 and 9 respectively. Results in Table 8 and corresponding plots in Fig. 5 show that there is a substantial queue in arm C and the average waiting time for this arm is 97.7 s.

Moreover, Table 9 and Fig. 6 show that the average number of vehicles in this queue is 17.2 which is in harmony with real-world condition.

9. Proposed improvement

To tackle the mentioned situation, a traffic light is proposed in the intersection of arm C with lanes A and B. The traffic light would be a sensor-based one which will sense the number of waiting vehicles in arm C. Once it exceeds 10 (this number is considered to be desirable for drivers), the traffic light in arm C changes to green and the traffic light for lanes A and B will change to red simultaneously. To do so, the following changes should be applied to Hold module of arm C.

Name	Type	Value	Limit
Queue C	Wait for Signal	0	10

The model is built by changing some features of the initial model as follows:

- Just lanes A, B and arm C are subject to change in the new model.
- The Decide modules in lanes A, B and arm C are omitted because they will be ruled by the traffic light.

Table 6
Initial replication results from arm F.

Rep.	1	2	3	4	5	Mean	SD	*N*
W.T. (s)	19.55	22.75	19.06	26.32	24.04	22.34	3.06	8.52

Table 7
Replication results from arm F.

Rep.	1	2	3	4	5	6	7	8	Mean	SD	*N*
W.T. (s)	19.55	22.75	19.06	26.32	24.04	18.76	21.12	20.51	21.51	2.66	6.96

Table 8
Average waiting times in the initial model (s).

Rep.	1	2	3	4	5	6	7	8	Average
Q. A	0.27	0.23	0.35	0.29	0.25	0.26	0.33	0.27	0.28
Q. B	0.28	0.16	0.18	0.35	0.23	0.26	0.18	0.16	0.22
Q. C	95.50	98.71	114.05	114.81	43.94	74.65	127.02	112.92	97.70
Q. D	0.24	0.27	0.28	0.26	0.24	0.23	0.26	0.25	0.25
Q. E	0.28	0.35	0.25	0.26	0.22	0.24	0.26	0.28	0.27
Q. F	19.55	22.75	19.06	26.32	24.04	18.76	21.12	20.51	21.51
								Total ave.	20.04

Table 9
Average number of vehicles in the queues of the initial model.

Rep.	1	2	3	4	5	6	7	8	Average
Q. A	0.0037	0.0028	0.0050	0.0047	0.0039	0.0042	0.0046	0.0036	0.00
Q. B	0.0004	0.0002	0.0005	0.0004	0.0002	0.0004	0.0002	0.0002	0.00
Q. C	15.9359	17.8603	20.0008	19.7911	7.4985	11.8816	23.1251	20.0941	17.02
Q. D	0.0037	0.0035	0.0042	0.0032	0.0034	0.0032	0.0031	0.0041	0.00
Q. E	0.0009	0.0009	0.0015	0.0011	0.0008	0.0009	0.0011	0.0011	0.00
Q. F	0.6303	0.7268	0.6483	0.9590	0.7188	0.6774	0.7146	0.7252	0.73
								Total ave.	2.96

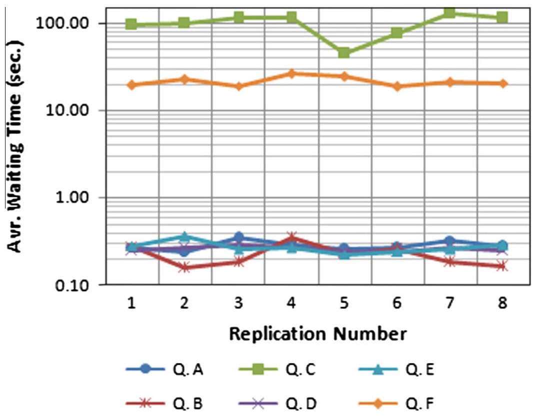


Fig. 5. Average waiting time for different routes in the initial model (logarithmic scale).

- A switch module (see Fig. 7) sends the signal to the whole system to change the traffic light based on the mentioned rationale.

The switch module creates an entity based on its IAT to check the system. In this case 1 s is assigned for IAT, which means the switch checks the model every second. When the entity enters the Decide module, it checks the number of the vehicles in the queue of arm C, and if the value is ⩽10, it proceeds to a assign module named “Light AB green” which assigns the variable “Light AB” the value one (cars coming from lanes A and B are allowed to pass through the junction). Else the entity enters another assign module, which sets the value of variable “Light AB” to 0. Then the Signal module sends the value of the “Light AB” to Hold module “Queue C”. The “Queue C” module (in Fig. 8) has been set to wait for signal 0 and limit value 10, which allows 10 vehicles to pass the junction, if the signal value is 0. Indeed, signal value 0 means the light is red for lane A and B and is green for arm C. In Table 10, the elements of the switch module with their details are provided.

The improved model is depicted in Fig. 8.

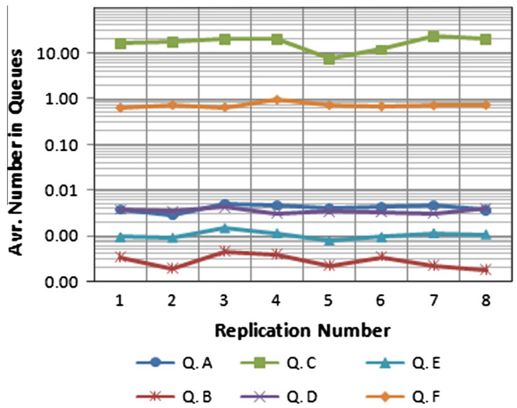


Fig. 6. Average number of vehicles in the queues of the initial model (logarithmic scale).

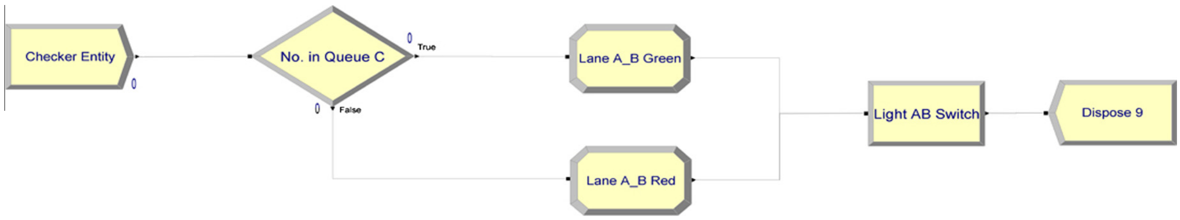


Fig. 7. Switch module to simulate the traffic light.

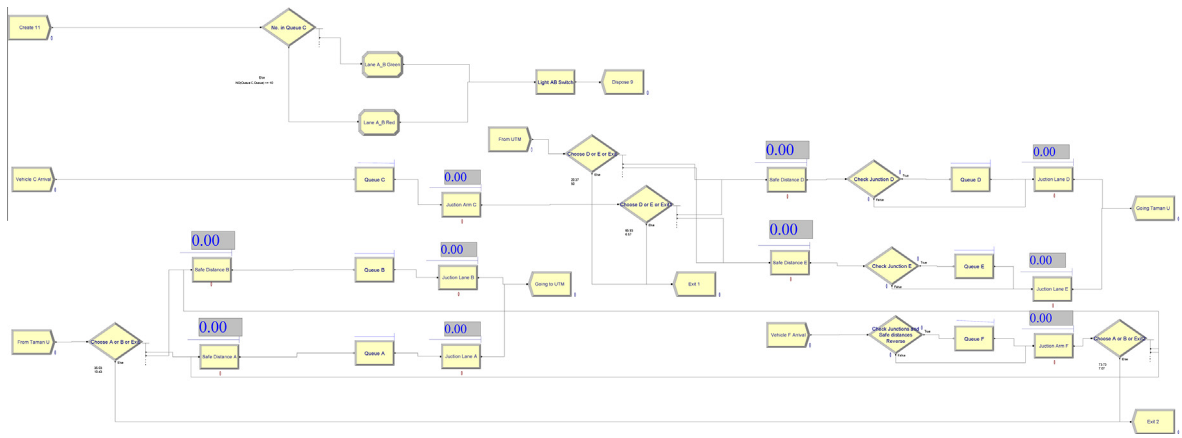


Fig. 8. Improved model by using Arena.

10. Results

The improved model was run based on the previous experimentation procedure. The outcome is summarized in Tables 11, 12 and Figs. 9, 10. Obviously, in arm C, the average waiting time and number of waiting vehicles have dropped dramatically, while there is just a little and reasonable increase in the waiting time of lanes A, B and D.

The average waiting time in initial model and improved model has been compared in Table 13. The results show that total average waiting time of the system has been declined from 20.04 to 9.42 s.

In addition, by comparing the fluctuations in Figs. 5 and 9 it can be concluded that the system has become more consistent in terms of average waiting time of the queues. This shows that in the improved model, the load of traffic from arm C has been shifted to other queues, which has made the network more balanced.

Table 10
Elements of the switch module.

Name	Entity type	Type	Units
Create module			
Checker entity	Entity 1	Constant	1 s
	Type	If	Expression
Decide module			
No. in queue C	2-Way by condition	Expression	NQ(Queue C.Queue) <= 10
	Assignment type	Variable name	New value
Assign module			
Lane A-B Green	Variable	Light AB	1
Lane A-B Green	Variable	Light AB	1
Signal Module			
Name			Signal Value
Light AB Switch			Light AB

Table 11
Results for average waiting time after improvement (s).

Rep.	1	2	3	4	5	6	7	8	Average
Q. A	3.642	3.561	3.937	3.956	3.727	3.608	4.192	3.874	3.81
Q. B	3.169	3.727	3.941	3.432	3.425	3.265	3.450	3.039	3.43
Q. C	33.023	31.884	32.839	32.138	31.812	33.035	31.595	31.708	32.25
Q. D	0.274	0.289	0.289	0.262	0.293	0.282	0.291	0.253	0.28
Q. E	0.012	0.009	0.009	0.008	0.014	0.013	0.013	0.008	0.01
Q. F	16.164	17.863	16.758	17.531	14.479	17.183	18.805	14.988	16.72
								Total ave.	9.42

Table 12
Average number of vehicles in the queues after improvement.

Rep.	1	2	3	4	5	6	7	8	Average
Q. A	0.686	0.664	0.778	0.733	0.737	0.655	0.778	0.733	0.72
Q. B	0.164	0.185	0.212	0.171	0.185	0.186	0.174	0.153	0.18
Q. C	5.639	5.604	5.523	5.677	5.551	5.619	5.773	5.477	5.61
Q. D	0.004	0.005	0.005	0.005	0.0059	0.006	0.005	0.005	0.00
Q. E	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.00
Q. F	0.482	0.491	0.593	0.523	0.461	0.559	0.594	0.458	0.52
								Total ave.	1.17

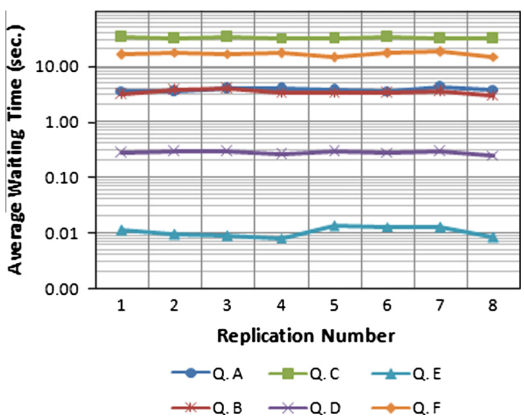


Fig. 9. Average waiting time for different routes in the improved model (logarithmic scale).

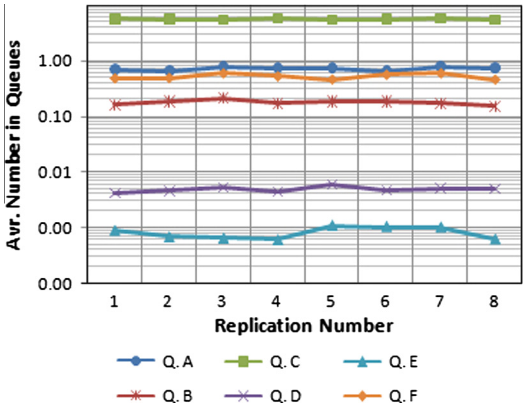


Fig. 10. Average number of the vehicles in the queues of the improved model (logarithmic scale).

Table 13
Summary of average waiting time in both models (s).

Queue	Initial model	Improved model
Q. A	0.28	3.81
Q. B	0.22	3.43
Q. C	97.70	32.25
Q. D	0.25	0.28
Q. E	0.27	0.01
Q. F	21.51	16.72
Total	20.04	9.42

11. Conclusion

Two adjacent T-junctions during rush hours were simulated in Arena to be analyzed and improved. First, the real situation was investigated to realize what modules and logics are needed to be considered in the model. Then, the necessary data for the developed model were collected and analyzed to specify their distributions. In the experiment stage, the run length, warm-up period and the number of replications were specified. Having run the model, it was revealed that arm C was the bottleneck of the system due to its relatively higher average of waiting times and number of vehicles in the queue.

For improvement, it was recommended to use a traffic light in the intersection of lanes A and B with arm C. The required changes were done to embed the traffic light in the model. To investigate the results, the model was run according to previous experiment procedure. The results showed that the average waiting time in arm C declined from 97.7 to 32.25 s, by 67%. Furthermore, the average waiting time of queues in the entire system dropped from 20.04 to 9.42 s, by 53%. Finally, the average number of vehicles in the queues of the whole network decreased from 2.96 to 1.17 by 60%.

References

[1] J.-B. Sheu, A composite traffic flow modeling approach for incident-responsive network traffic assignment, Physica A 367 (2006) 461–478.
[2] J. Otamendi, J.M. Pastor, et al, Selection of the simulation software for the management of the operations at an international airport, Simulat. Model. Pract. Theory 16 (8) (2008) 1103–1112.
[3] R.M. Tahar, K. Hussain, Simulation and analysis for the Kelang container terminal operations, Logist. Inform. Manage. 13 (1) (2000) 14–20.
[4] A. Bush, W.E. Biles, G.W. DePuy, Waterway, shipping, and ports: iterative optimization and simulation of barge traffic on an inland waterway, in: Proceedings of the 35th Conference on Winter Simulation: Driving Innovation, Winter Simulation Conference, 2003, pp. 1751–1756.
[5] S.E. Kesen, Ö.F. Baykoç, Simulation of automated guided vehicle (AGV) systems based on just-in-time (JIT) philosophy in a job-shop environment, Simulat. Model. Pract. Theory 15 (3) (2007) 272–284.
[6] R. Sancar Edis, A. Ornek, Simulation analysis of lot streaming in job shops with transportation queue disciplines, Simulat. Model. Pract. Theory 17 (2) (2009) 442–453.
[7] S.R. Chakravarthy, Reliability, health care, and simulation, Simulat. Model. Pract. Theory 29 (2012) 44–51.
[8] O.A. Almaz, T. Altiok, Simulation modeling of the vessel traffic in delaware river: impact of deepening on port performance, Simulat. Model. Pract. Theory 22 (2012) 146–165.
[9] P. Cortés, J. Muñuzuri, J. Nicolás Ibáñez, J. Guadix, Simulation of freight traffic in the seville inland port, Simulat. Model. Pract. Theory 15 (3) (2007) 256–271.
[10] A. Motraghi, M.V. Marinov, Analysis of urban freight by rail using event based simulation, Simulat. Model. Pract. Theory 25 (2012) 73–89.
[11] C. Woroniuk, M. Marinov, Simulation modelling to analyse the current level of utilisation of sections along a rail route, J. Transport Lit. 7 (2) (2013) 235–252.

[12] Z. Liu, A survey of intelligence methods in urban traffic signal control, IJCSNS Int. J. Comput. Sci. Netw. Secur. 7 (7) (2007) 105–112.
[13] K.V. Nagarajan, P. Vial, G. Awyzio, Simulation of SNMPV3 traffic flow meter MIB using ARENA simulation modelling software, Animation 353 (2002) 031.
[14] A.D. Ambrogio, G. Iazeolla, L. Pasini, A. Pieroni, Simulation model building of traffic intersections, Simulat. Model. Pract. Theory 17 (4) (2009) 625–640.
[15] W.X. Ci, S. Ahmed, F. Zulkifli, A. Ramasamy, Traffic flow simulation at an unsignalized T-junction using Monte Carlo Markov chains, in: IEEE International Conference on Signal and Image Processing Applications (ICSIPA), 2009, IEEE, 2009, pp. 346–351.
[16] H. Pranevičius, T. Kraujalis, Knowledge based traffic signal control model for signalized intersection, Transport 27 (3) (2012) 263–267.
[17] K. Salimifard, M. Ansari, Modeling and simulation of urban traffic signals.
[18] K.I. Ahmed, Modeling Drivers' Acceleration and Lane Changing Behavior, Ph.D. thesis, Massachusetts Institute of Technology, 1999.
[19] T. Toledo, H.N. Koutsopoulos, A. Davol, M.E. Ben-Akiva, W. Burghout, I. Andréasson, T. Johansson, C. Lundin, Calibration and validation of microscopic traffic simulation tools: Stockholm case study, Transport. Res. Rec.: J. Transport. Res. Board 1831 (1) (2003) 65–75.