A Simulation Study to Reduce Staff Overtime at The Department of Motor Vehicles Portsmouth, VA

Randy N. Brooks,
Peter Foytik,
Ryan Frost

Term Project for ECE-505
Old Dominion University,
Discrete Event Simulation

Dr. Rick McKenzie
Old Dominion University

Keywords:
Discrete Event Simulation, Arena, D.M.V.

ABSTRACT: *This paper is the result of a term project for ECE-505 where a Department of Motor Vehicles (DMV) Customer Service Center (CSC) located in Portsmouth, VA is modeled, simulated and analyzed using an academic version of Arena Software. The objective of the project is to reduce staff overtime by limiting the time when a specific service transaction type is permitted. This allows customers already in the system to have their transaction completed, but prevents an influx of customers that will require the staff to work overtime. An optimizer is used in conjunction with the developed model to predict the optimum time to halt servicing longer transactions, thus emptying the queued work in the system to coincide with the staff schedule.*

1 Introduction

The CSC located in Portsmouth, Virginia was studied to understand factors that contribute to timely customer service delivery. Three methods were used to learn and understand DMV operations. First, investigational visits were performed to observe and witness the operations. Secondly, informal and formal personnel interviews were conduced with staff and management. Thirdly, data was collected from an automated information system to provide a base line for transaction durations and customer inter-arrival rates.

The Portsmouth CSC uses the Q-matic processing system to aid in providing timely service to customers. Customers entering the CSC are directed to an entry station where the necessary forms are obtained and a ticket is issued to each customer. Customers go to a sitting area to complete the forms and wait for their number to be displayed on the electronic display, directing them to the appropriate service window. After going to the service window, the transaction is performed by a customer service representative. When the transaction is finished at the service window customers exit the building.

Early in the investigation it became apparent that one primary interest for management and staff alike was to minimize the amount of overtime worked. The staff would enjoy having a dependable work schedule in order to pick-up children and attend to family needs after work on a regular basis. Management would like to improve budget estimation of the overtime expense. With this objective in mind an Arena model of the office was created.

2 Background

DMV services are grouped into seven discrete types that are referred to as ‘ticket types’. Table 1 shows the services that are associated to each ticket type, A through G. Typically the transaction times of these ticket types increase from A to G.

All transactions types are modeled, however, to simplify and limit the number of Arena modules used, only the service window area is modeled. Drive-thru, picture taking and exam stations are implicitly modeled by a roll up of the transaction time. The staff resources defined for these activities are not modeled. The decision not to model these details is based on the main simulation objective of reducing overtime.

A	Pre-Printed Renewals, Handicapped Placards, Transcripts
B	Renewals, Registrations, trip Permits, VIN, Surrender plates
C	Drivers Lic., ID cards, address changes
D	Titles Only
E	Compliance
F	Tests, Misc.
G	Commercial Customers

Table 1

2.1 Service Goals

The service goals established by the head office are used to set maximum waiting and maximum total office visit time. The maximum waiting goals for each customer is twenty minutes and sixty minutes total in-process visit time. The goal for the maximum processing time is seven minutes. The service goals are important since the manager or assistant manager adjusts customer priority in the Q-matic system based on the current customers waiting in queue and the managers own past experience.

3 Model Development

The model has representation for each ticket type queue. An Arena 'Hold' module is used to represent these queues. The first assignment block, assigns a ticket type attribute (A-G) based on a discrete distribution derived from the collected data. The percent processed for each ticket type (in Nov 2006) is shown in Table 2.

	November Ticket Type Averages			
Nov. 2006	**Number Processed**	**%**	**Average Time**	**Max**
A	967	12.04%	5:00	6:12
B	1756	21.87%	6:02	7:58
C	1555	19.36%	8:00	10:08
D	1461	18.19%	10:44	11:22
E	1171	14.58%	8:30	10:14
F	890	11.08%	7:30	9:08
G	231	2.88%	15:24	19:20
Total	**8031**	**1.00**		

Table 2

Customer inter-arrival rate is based on actual data collected during November 2006. The monthly information was delivered in hardcopy. Optical Character Recognition was used to transfer the data into an Excel spreadsheet for analysis.

The weekday arrival rates for each 30-minute period were averaged for the month. Using Arena's non-stationary Poisson distribution feature a daily arrival schedule was developed to reflect the quantity of customers arriving in each thirty-minute period.

Table 3 shows inter-arrival data from November 2, 2006 that was scanned from the original hardcopy. The third column, 'Ticket Taken' is the number of arriving customers in the previous 30 minutes. In the example below, ten customers arrived between 7:30 and 8:00, and twenty-nine between 8:00 and 8:30, and so on.

Time	Cust. Waiting	Ticket Taken	Cust. Served
7:30 AM			
8:00 AM	1.3	10	
8:30 AM	14.6	29	16
9:00 AM	14	12	17
9:30 AM	15.2	22	19
10:00 AM	19.4	20	14
10:30 AM	23.5	25	24
11:00 AM	18.3	21	23
11:30AM	14.2	20	26
12:00 PM	11.8	16	14
12:30 PM	18.7	20	19
1:00 PM	7.7	21	30
1:30 PM	7.7	26	26
2:00 PM	6.5	28	27
2:30 PM	11.1	26	17
3:00 PM	20.8	18	17
3:30 PM	23.7	22	13
4:00 PM	32.8	18	12
4:30 PM	38.1	21	15
5:00 PM	48.3	25	16
5:30 PM	39	1	21
6:00 PM	10.9		29
6:30 PM			5

Table 3

3.1 Conceptual Model

The model was built using the conceptual model as shown in figure 1. The queues represent a customer's service request of a specific ticket type. Staff and systems are represented via the actual staff work schedule and the Q-matic system. Office policy dictates the priority of service granted for different ticket types. These priorities change over time, and thus, proved the most difficult aspect to accurately model. A service matrix described below provides a limited representation of the office policy.

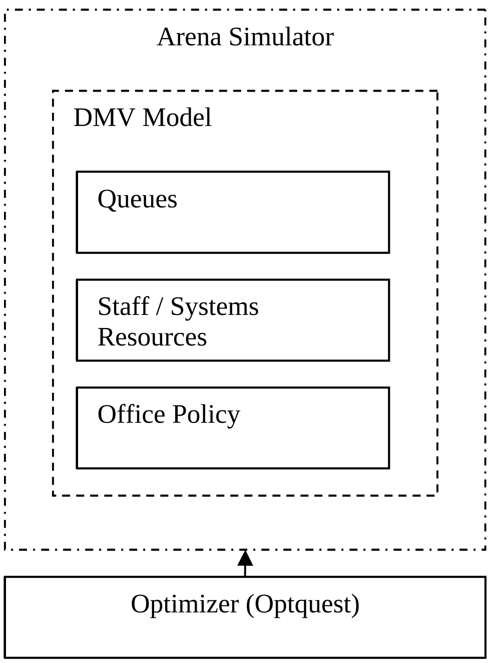

Figure 1

Individual staff schedules were created to reflect each service window and information desk. Office policy was represented using a control entity to identify a window where the work could be performed.

At the heart of the decision logic is the service matrix. The service matrix is a two dimensional array with the rows defining each window. The columns A through G define a Boolean representation of whether that particular service could be performed (1 or 0). In this way, a window can be made to service any ticket type as defined from the column settings.

A set data structure called TypeQueues contains all individual ticket type queues. A variable is used to index the queues. The pseudo code for the work location selection is as follows:

1. Determine if customers are currently waiting in a specific type queue.
2. If no, check next ticket type queue.
3. If yes, determine if the state of the resource at current window is IDLE.
4. If no, check next window.
5. If yes, determine if the window can service current ticket type via the Service Matrix.
6. If no, dispose control entity.
7. If yes, the customer represented by the ticket type is removed from the queue and proceeds to the window for processing.

3.2 Modeling Manager Behavior

The manager's priority adjustment of the Q-matic system during the day proved very difficult to accurately model. Although we found that it could be modeled better, the number of modules required exceeded the limits of the academic version of Arena software.

One DMV service center performance metrics could vary greatly as compared to another due to differing priority management strategy. Portsmouth office manager, Valerie Alexander, stated that inexperienced managers tended to micro-manage the Q-matic priority, thus not giving the system time to load balance. Rarely is it necessary to change ticket servicing priority more than twice an hour.

3.3 Assumptions

The following assumptions are used to describe the system in detail. The conceptual model and assumptions are used together to verify that the model has been built as intended.

1. Customer delivered inter-arrival rates is a true representation. Data is IID.
2. Customer delivered service time is a true representation of the data. Data is IID.
3. Information systems will not normally 'go-down' or stop performing.
4. Wait time information published is valid and can be used for model validation.
5. If a sample of the data supplied by the DMV matches observations then all DMV supplied data will be considered accurate.
6. DMV supplied data from November 2006 is a good representative of April 2007.
7. Balking was difficult to determine since it could happen in the parking lot and was not modeled.
8. Jockeying was not modeled due to the current system design of customers using one entry queue and being assigned a service queue.

9. Actual reneging (queue abandonment) was less than 1% according to DMV data and therefore not modeled.

3.4 Number of Replications

In order to determine the number of replications needed to achieve a confidence-interval half width of 90%, the following sequential-sampling logic used is detailed below.

Arena provides two variables, ORUNAVG and ORUNHALF to report the current point estimator and half width of the aggregated runs of a model variable of interest (average number of customers waiting in our case). The variables are used to calculate the number of replications needed to reach the maximum acceptable error as described in Chapter 12, section 12.5.1 of the course text book [1].

Other variables, such as, NREP (current replication) and MREP (maximum replications) are also used in the model logic to signal termination once the objective is met.

The initial number of replication is set very high (100000). The logic creates a control entity at the beginning of each replication and allows for two replications to create an initial half-width. This is needed to prevent a division by zero error and SIMAN variable display errors.

The point estimate error rate is calculated and compared to the maximum error objective. If the point estimator error is too high; more replications occur, otherwise, the objective has been reached and the runs terminated.

Point Estimate Error = 1 / (X bar / half-width)

Point Estimate Error Percentage =
100 * (1/ (LocORUNAVG / LocORUNHALF))

Where

LocORUNAVG = Last run point estimate computed average across all runs

LocORUNHALF = Last run computed half width across all runs

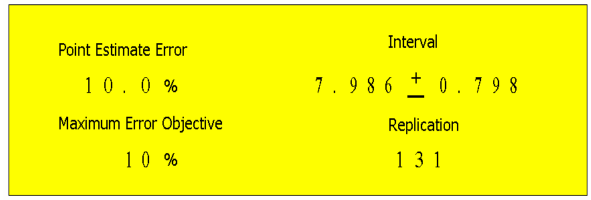

Figure 2

The logic above resulted in determining that thirty-nine replications produced an acceptable error rate of 20% in the average wait time. When the maximum error was reduced from 20% to 10% as shown in figure 2, 131 replications were needed to meet the maximum acceptable error of the mean time in the system. This logic is included in a demonstration model 'autoreplication.doe'. The logic had to be removed from the submitted model due to the academic version restrictions.

3.5 Verification

Input data was modified to produce expected results. Decision logic was tested and corrected when errors were found. By stepping through the simulation slowly, Arena variables were traced and proven to be correct. Assumptions were reviewed and compared to the model to assure the model was built as intended.

3.6 Validation

The DMV's average in-system wait times were compared to the simulation's predicted average wait time. Figure 3 shows the difference between real and simulated average wait times for the month of November 2006, and Figure 4 shows summary statistics for the two data sets.

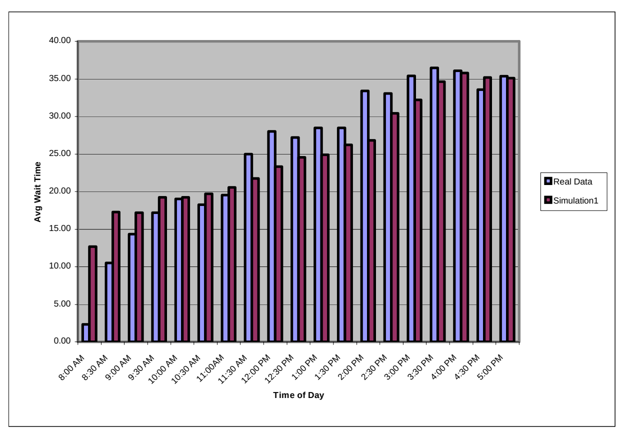

Figure 3

Real Data	
Mean	25.35
Standard Deviation	9.784787214
90% Confidence Interval	3.692341333
90% CI	[21.66 - 29.04]

Simulation1	
Mean	25.09
Standard Deviation	7.077699322
90% Confidence Interval	2.670807365
90% CI	[22.42 - 27.76]

Figure 4

3.7 Population Difference Test

A test was performed to estimate of the difference between two population means with independent small samples:

$$\left(\bar{y}_1 - \bar{y}_2\right) \pm \frac{\tau_\propto}{2}\sqrt{S_P^2\left(\frac{1}{n_1}+\frac{1}{n_2}\right)}$$

Where

$$S_P^2 = \frac{(n_1-1)s_1^2 + (n_2-1)s_2^2}{n_1+n_2-2}$$

$$S_P^2 = \frac{(19-1)9.78^2 + (19-1)7.08^2}{19+19-2} = 72.8874$$

Substituting in to the first equation results in:

$$(25.35 - 25.09) \pm 1.729\sqrt{72.8874\left(\frac{1}{19}+\frac{1}{19}\right)}$$

$$0.26 \pm 4.789$$

Based off the above test, we can be 90% confident that the interval (-4.529, 5.049) encloses the true difference between the mean wait times. Since the interval includes 0, we are unable to conclude that the two means differ.

3.8 Hypothesis Test

A second test was performed to test the difference between two population means of independent small samples:

$$T = \frac{\left(\bar{y}_1 - \bar{y}_2\right) - D_0}{\sqrt{S_2^p\left(\frac{1}{n_1}+\frac{1}{n_2}\right)}}$$

Where

$$S_P^2 = \frac{(n_1-1)s_1^2 + (n_2-1)s_2^2}{n_1+n_2-2}$$

$$S_P^2 = \frac{(19-1)9.78^2 + (19-1)8.35^2}{19+19-2} = 82.685$$

Substituting in to the first equation results in:

$$T = \frac{(25.35 - 25.09) - 0}{\sqrt{72.8874\left(\frac{1}{19}+\frac{1}{19}\right)}}$$

$$\frac{0.26}{2.77} = 0.0939$$

The null hypothesis is:

$$H_0 : \mu_1 = \mu_2$$

Hence, we test for the mean values for each sample to be statistically equal. We determine the critical value by using a Student's T Distribution chart.

$$t_0 = \pm 1.684$$

Therefore the rejection region on a distribution graph would be > 1.684 and < -1.684. Since T (0.0939) does not fall in the rejection region, we cannot reject the null hypothesis, thus we cannot say with a 90% confidence that the two means are different.

4 Simulation Setup

OptQuest is a third-party optimization product that can monitor and manipulate Arena variables [2]. Control variables, response variables, and objectives are required to be defined as part of the setup. As an option you can set constraints to the defined controls and responses. The cut-off time of issuing F ticket types (driver's test) was chosen to be studied since they have a long mean in-system time.

Once the control variables, response variable, and objectives are defined OptQuest then runs the model a set number of times; the default is one hundred. Once it is done running, OptQuest will display the top twenty-five results and whether or not they are feasible solutions. From these results a text file can be created for comparative analysis.

4.1 Experiment

To meet our project objective, an experiment to reduce overtime was designed. One way to decrease the amount of overtime in a DMV system is to stop distributing one of the longer processing ticket types earlier than the normal time to stop distributing tickets. If a ticket type were stopped early enough, the system could finish processing the entities in queue at a reasonable time, and hopefully reduce the amount of overtime the employees have to work.

4.2 Experiment Design

Since ticket type F had the longest process time we chose it to be the ticket type that the system rejected earlier. A QuittingTime variable had to be created which notified the DMV model when to stop distributing and servicing the F tickets. This is also the variable that OptQuest uses to control the experiment. This variable had a constraint that limited its quitting time to be between three and five in the afternoon.

The response of the experiment was a variable that calculated the amount of time the employees worked past 5:30pm, which was their scheduled time to leave, minus a fifteen-minute buffer for close out activities. This variable had constraints in that it had to be no sooner than the DMV closing time.

Thirty-six replications were run per simulation. This was chosen in order to achieve a 20% maximum expected error based on our replication test.

The objective of the experiment was to minimize the response variable by changing the control variable. After running the OptQuest experiment results showed the best QuittingTime that resulted in the least overtime.

5 Results

OptQuest conducted 80 simulations and found the 13th simulation to the best simulation. The best solution was for the control variable QuittingTime to be 913 minutes, with a corresponding objective value, overtime, of 57 minutes. These results can be seen in Figure 5 and 6.

Best Solutions

Select	Simulation	Objective Value	Status	quittingTime
☐	13	57.194444	Feasible	913
☐	8	57.611111	Feasible	914
☐	17	59.027778	Feasible	916
☐	15	59.027778	Feasible	915
☐	16	60.111111	Feasible	912
☐	28	67.666667	Feasible	917

Figure 5

Optimization Completed

Total simulations: 80

Best Simulation 13

Minimize	Objective Value	Status
Best Value	57.194444	Feasible
Current Value		

Control Name	Best Value	Current Value
quittingTime	913	

Constraint Name	Type	Status
New Constraint	Linear	Feasible

Figure 6

6 Conclusions

When running the simulation in Arena, without stopping any ticket type early, the resulting average overtime was 91 minutes. This meant that on average the closing employees were not able to leave until just after 7:00pm. 36 replications were run in order to produce this overtime value.

In comparing these results to the results achieved through OptQuest, we found we could reduce the amount of overtime considerably, by stopping the issuing of F tickets earlier in the day. OptQuest found the best time to stop issuing F tickets was at 3:13pm. By doing this the resulting overtime was reduced 62% to 57 minutes. This overtime would allow the employees to leave the center by 6:30pm.

This reduction of overtime can save the DMV money, as well as increase morale of the employees.

References

References used in this document are cited below:

[1] D. Kelton, R. Sadowski, D. Sturrock: "Simulation with Arena", Fourth Edition, 2007.
[2] OptQuest for Arena User's Manual, Rockwell Automation and OptTek, 2004

Acknowledgment

The authors would like to thank Ms. Linda Ford who graciously provided Q-matic output reports from the Portsmouth, VA office used in this study. Also, Ms. Valerie Alexander, the Portsmouth CSC manager was invaluable for her insight into the day-to-day operations. The team appreciated the hard work and dedication of every employee working at the Portsmouth service center.

Author Biographies

RANDALL N. BROOKS is a Project Manager at Northrop Grumman in Newport News, VA. He holds a BS in Computer Science from Chico State University. His main interest is in solving complex problems and system improvement using modeling and simulation techniques. Other interest include, Java programming, OOD, and software design patterns.

PETER FOYTIK is a Project Scientist at the Virginia Modeling Analysis Center in Suffolk VA. Currently he holds a BS in Computer Science from Christopher Newport University. His interests are in artificial intelligence, simulation design and integration, and software development.

RYAN FROST is a Senior Requirements Analyst at Northrop Grumman in Norfolk, VA. He holds a BS in Integrated Science and Technology from James Madison University. His interests are in business process reengineering and automation through the use of modeling and simulation techniques. Other interests include simulation animation and user interface design.