The Art Gallery Problem

Sarah Clapoff, Lucas Mayall, Tim Rolston

April 11, 2023

CSC 446 Simulation Study
Dr. Kui Wu

Abstract

An ideal art gallery visit would allow customers to view all the paintings at their own pace, without being hindered by crowds. However, the reality is that certain painting styles are more appealing than others, and a customer's tolerance for crowds can influence their decision to view a particular painting or even leave the gallery entirely. In this study, we employ a stochastic, discrete event-driven simulation model to represent the behavior of visitors navigating an art gallery. Our simulation primarily examines how the actions of individual customers affect the choices of others.

Our model incorporates three event types: visitors entering the system and selecting paintings to view based on their style preference, the number of other customers currently viewing the painting, and their tolerance for crowded spaces. If all unvisited paintings are unappealing to a visitor, or they have already viewed all the paintings, they will exit the system. We investigate the impact of changing the number of displayed paintings on the overall gallery quality and the frequency of early departures to determine the optimal number of paintings.

We recognize that factors such as painting quality and crowd tolerance are challenging to measure and quantify. Acquiring concrete data for these parameters was considered unrealistic within the project's scope. Nonetheless, we believe our model could serve as a foundation for future work aiming to develop an accurate simulation of this system, as the code is designed to be easily modified when data becomes available.

Contents

List of Figures

List of Tables

1 Problem Description

The simulation discussed in this paper models the behavior of customers in an art gallery. This abstract gallery contains a number of paintings with each painting having a random quality and style. The customers visiting the gallery want to view all of the paintings in whatever order maximizes the total score of the paintings seen. Each customer has a favorite painting style - Baroque, Impressionist, Modern, or Abstract - and a unique tolerance for viewing paintings with a large crowd. Customers will try and view what they deem to be the best painting they have yet to view but if too many people are already viewing it they may decide to view a less desired painting first, or leave altogether if none of the remaining paintings are desirable. The goal of the simulation is to determine the optimal number of paintings, such that the percentage of unviewed paintings when a customer leaves is minimized.

2 Simulation Model

The simulation uses an event based system to schedule and process the arrival, departure, and movement of each customer. There is no limit on the number of customers allowed inside of the Gallery nor on the number of viewers at a painting. Instead, when faced with viewing a painting that is below some minimum score, the customer can choose to leave the gallery altogether. The code for the simulation relies on two main classes: Customer and Painting with a GallerySim class managing the scheduling and processing of the events. Each Painting has a `Style` Enum, which contains the styles *Baroque*, *Impressionist*, *Modern*, and *Abstract*, `num_viewers` for the current number of viewers at that painting, and `quality` which is a random normalized distribution to determine the quality of the painting. Each Customer has a randomly chosen `favourite_style` and a random normalized `tolerance` level. Note that the `tolerance` attribute is based off of how desirable a painting is to the customer.

3 Simulation Parameters

Simulation parameters of interest are given in Table 1. Mean values were used in the random variate generation of visitor tolerances, painting qualities, visitor inter-arrival times, and visitor viewing times at a painting. The system variable used in this study was `num_paintings`, the number of paintings in the gallery. Performance metrics are given in Table 2.

Variable	**Description**	**Value**
TOLERANCE_MEAN	Visitor crowd tolerance	3
TOLERANCE_STD	Visitor crowd tolerance	2
QUALITY_MEAN	Painting quality	0.5
QUALITY_STD	Painting quality	0.3
INTERARRIVAL_TIME_MEAN	Customer inter-arrival time	1
INTERARRIVAL_TIME_STD	Customer inter-arrival time	1
VIEWING_TIME_MEAN	Customer viewing time	3
VIEWING_TIME_STF	Customer viewing time	0.5
STYLE_CONSTANT	Weight of painting style on score	1
PATIENCE_CONSTANT	Weight of painting crowd on score	1
QUALITY_CONSTANT	Weight of painting quality on score	1
MIN_SCORE	Score below which customers will leave the gallery	130
num_customers	Total number of customers to be simulated	1000
num_paintings	Number of paintings in gallery	*system variable:* (5, 20, 50)

Table 1: Relevant Simulation Parameters.

Performance Metric	**Description**
% Paintings left when cust leaves	Reflects how much of the gallery was viewed
% of customers that leave early	Reflects overall customer satisfaction

Table 2: Relevant Performance Metrics.

4 Methodology

4.1 Simulation Build

The model is implemented in Python, using the framework for an event-driven simulation used throughout this course [1]. Once initialized, the simulation enters a loop and begins processing events until `num_customers` have departed the system. Here an overview of the simulation via its classes is presented; please see the Appendix for the full code.
The program is comprised of the following classes: EventType, Style, Painting, Customer, Event, EventList, CustomerStats, SimStats, and GallerySim. Classes of note for this study are Customer, Painting, and GallerySim.

Classes EventType and Style are enumerators, relating integer constants to the 3 event types: *Arrival, Departure,* and *Move,* and to the 4 painting styles: *Baroque, Impressionist, Modern,* and *Abstract.* Each Visitor has a randomly chosen favourite style, which influences how paintings are scored.

Classes Event and EventList are used to create events, and sort instances of customer events respectively. EventList is used to create and maintain the FutureEventList; it is implemented as a SplayTree data structure [2]. This model consists of three Event types: *Arrival*, *Move*, and *Departure*. A customer may experience multiple *Move* events, but only one *Arrival* and one *Departure*. Furthermore, a customer triggers the creation of their own *Moves* and *Departure*, however an *Arrival* is scheduled by the arrival of the previous customer. This can be seen in Figure 1. It is clear to see that this simulation does not follow a queuing model; instead the customer's future events are dynamically generated based on how all current customer actions influence the system as a whole.

Classes CustomerStats and SimStats were used to calculate, maintain, and report statistics for each Customer, and for the each simulation run. A multitude of statistical values were included, as this code has the potential for future studies of different system variables.

The Painting class creates `num_paintings` unique paintings during run initialization. Each Painting has attributes for style, number of current viewers, ID, and quality. The last of which is a normal random variate. These attributes are used in the statistical report generation and evaluation of the performance metric. Thus, variation of `num_paintings` is directly reflected here, as the distribution of styles, quality, number of viewers, and other statistics are all generated from Paintings.

The Customer class creates an unique customer object for every new arrival into the gallery. Each Customer has attributes for favourite style, tolerance, viewed paintings, current painting, and an instance of CustomerStats. The favourite style is randomly generated, and the tolerance is a normal random variate. This class also includes the functions used to score paintings, which are called when the Customer moves. Similar to Paintings, Customer attributes are used frequently in statistical report generation.

Finally, the GallerySim class creates a single run of the simulation for the given `num_customers`, `num_paintings`, and `seed` for random number generation. This class follows the general outline for a discrete-event simulation: the GallerySim attributes are initialized by creating instances of Paintings, SimStats, FutureEventList, and customer list. The first Arrival is scheduled, and the main event processing loop is entered. Once `num_customers` have departed, the statistical report is generated and the simulation ends. This can be seen in Figure 1. Notably, the only events processed directly from the main loop are Arrivals and Moves. Furthermore, a Move is always processed immediately following an Arrival. In the code, this is implemented by changing the current EventType to the next type before calling that event's processing method (eg. changing evt.EventType from Arrival to Move before calling ProcessMove(evt)). This is due to the nature of the system, the primary event is a Customer moving between paintings. That is, for any given Customer, the next Move is created and scheduled after processing the current one. The choice of next painting is dependant on the current state of the system, rather than the time of the next Move (as is the case for classical queuing models).

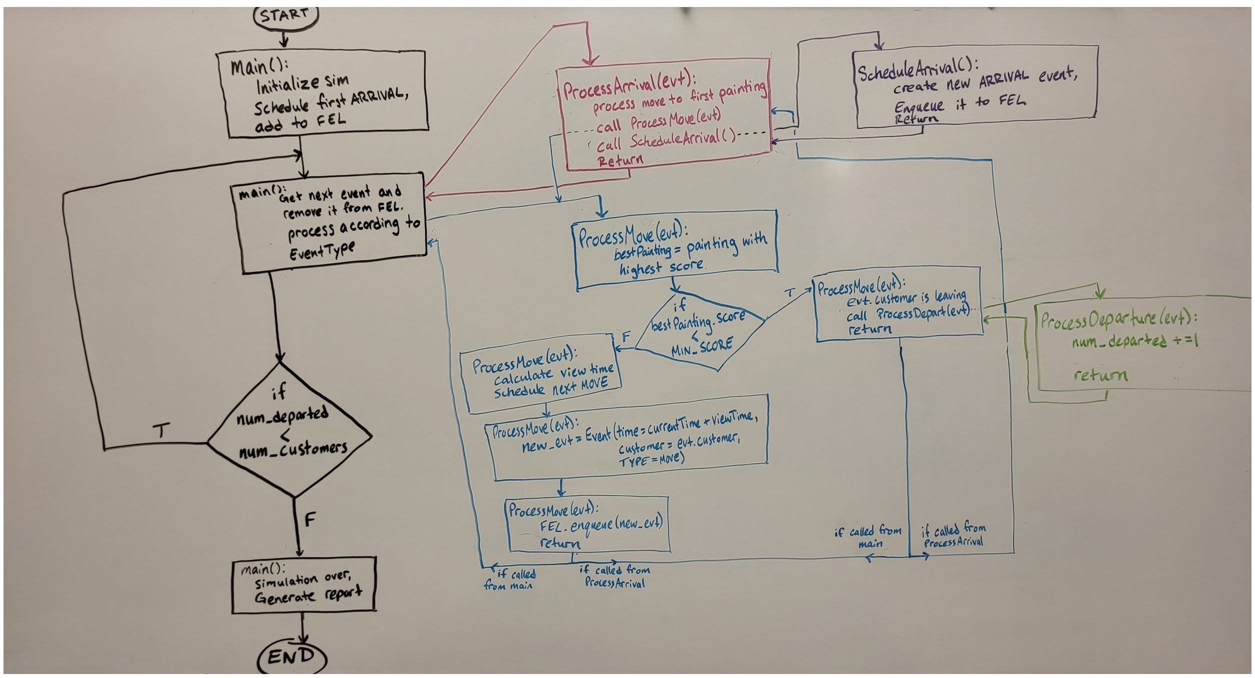

Figure 1: Control flow visualization for event processing in GallerySim class.

4.1.1 Scoring

Figure 2 illustrates the distribution employed for calculating each customer's scoring values. An inverse Sigmoid function was used for this purpose, as it maintains a relatively linear relationship with middle-quality paintings while being highly critical of low-quality ones and highly favorable towards high-quality paintings. Since obtaining such data in the real world is challenging, we opted for this distribution as a suitable representation, assuming that individuals typically enjoy the highest quality paintings much more than the lowest quality ones and find extremely high-quality paintings far more engaging.

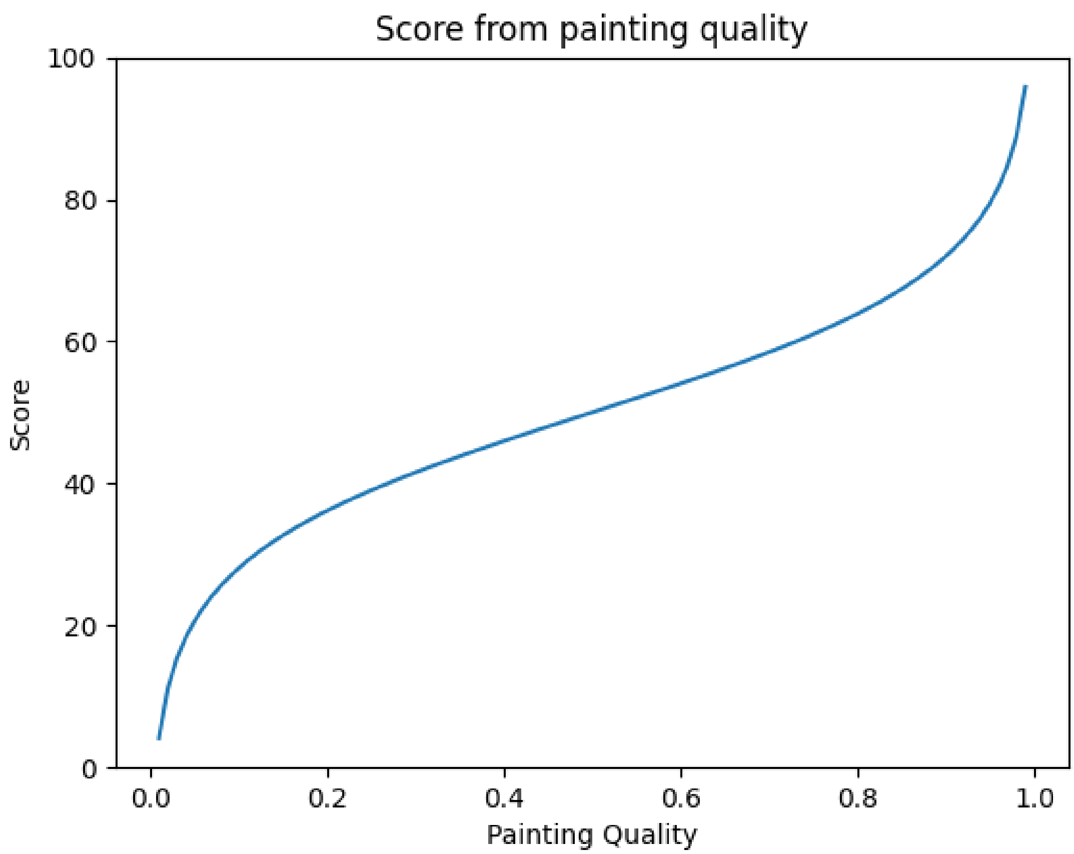

Figure 2: Paintings quality score distribution

Figure 3 displays the distribution of how a painting's score for a customer with a tolerance of 5 is influenced by the number of viewers currently at the painting. This graph indicates that customers generally prefer viewing paintings with 50 or fewer viewers. We selected this distribution as it reflects the notion that customers generally do not enjoy observing a painting with a large crowd, unless they have a high tolerance. It should be noted that this distribution would appear slightly different for customers with varying tolerance levels.

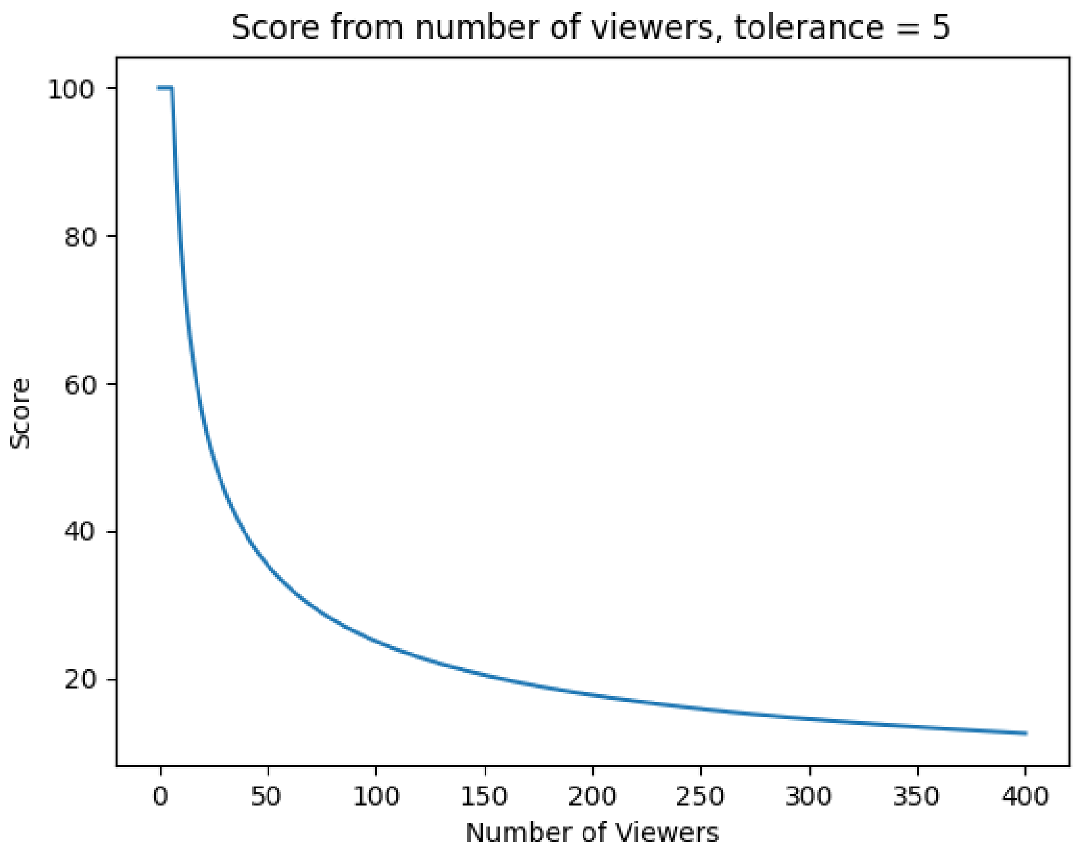

Figure 3: Paintings viewer score distribution

4.2 Simulation Parameter Setup

Our simulation primarily relies on two parameters: `num_paintings` and `num_customers`. Throughout various runs, we tested the simulation with three different values for `num_paintings` (5, 20, and 50), each with 1,000 customers. We chose these values as we believed they best represented the influence of `num_paintings` on the `num_customers` who leave the art gallery early (before viewing all the paintings).

Additionally, we employ several global variables to calculate the distributions for *Tolerance*, *Quality*, and *Arrival Times*. The values for these variables remain constant throughout our simulation and are established based on what we deemed most reasonable for the distribution they represent. Further details on the equations used can be found in the README within the GitHub repository.

4.3 Statistic Collection

In our simulation, we gathered data on two key metrics: the percentage of customers who exit the art gallery prior to seeing all of the paintings and the average percentage of unviewed paintings when they leave. Within the program, these variables are represented as `num_customers_leave_early` and `avg_num_paintings_left`. Although our simulation records numerous other statistics, this paper primarily concentrates on these two performance metrics.

5 Results Analysis

5.1 Data / collected stats

Tables 3, 4, and 5 present the outcomes of 5 simulation runs for each of the 3 distinct scenarios. These scenarios examine the impact of varying the number of paintings on two metrics: the average proportion of unviewed paintings when customers depart prematurely, and the percentage of customers who leave early. Although the goal is to minimize both values, the more crucial metric is the average proportion of paintings left unseen by customers.

Table 3: Scenario 1 Table: 5 Paintings

Run	**Seed**	**% Paintings left when cust leaves**	**% of custs that leave early**
R1	1	77.88%	17.90%
R2	2	77.43%	19.60%
R3	8	66.64%	100.00%
R4	10	74.07%	21.60%
R5	20	69.34%	22.70%

Table 4: Scenario 2 Table: 20 Paintings

Run	Seed	% Paintings left when cust leaves	% of custs that leave early
R1	1	17.32%	100.00%
R2	2	20.92%	100.00%
R3	8	34.98%	100.00%
R4	10	23.47%	100.00%
R5	20	31.50%	100.00%

Table 5: Scenario 3 Table: 50 Paintings

Run	Seed	% Paintings left when cust leaves	% of custs that leave early
R1	1	16.45%	100.00%
R2	2	17.45%	100.00%
R3	8	25.86%	100.00%
R4	10	27.33%	100.00%
R5	20	28.99%	100.00%

5.2 Scenario Analysis

As depicted in Tables 6 and 7, the average results for the scenario are presented, along with their respective 95% confidence intervals. The data from Table 6 indicates that when a gallery has only a small number of paintings, customers tend to leave without viewing a significant portion of the art. However, as demonstrated in Scenario 2, when the number of paintings increases to 20, the proportion of paintings not viewed decreases considerably. This trend continues, albeit at a slower pace, in Scenario 3, where the gallery features 50 paintings. The findings imply that augmenting the quantity of artwork on display leads to customers viewing a larger percentage of the collection before departing. Nevertheless, this effect experiences diminishing returns, suggesting that perpetually increasing the number of paintings would eventually yield little improvement in customer engagement.

Table 6: Final Results

Scenario	Avg - % paintings not seen	Avg - % leave early	Max - Num Paintings
S1	73.07%	36.36%	5
S2	25.64%	100.00%	20
S3	23.22%	100.00%	50

Table 7: Final Results Stats

Scenario	Unseen paintings 95% CI	Leave early 95% CI	Max - Num Paintings
S1	(66.91%, 79.24%)	(-7.87%, 80.59%)	5
S2	(16.47%, 34.79%)	CONSTANT	20
S3	(15.96%, 30.46%)	CONSTANT	50

6 Conclusion

6.1 Results

The results of our simulation show us that the number of paintings can greatly affect both the amount of people who leave early and the number of paintings that are not seen by a customer when they leave. From the simulations we ran, it is clear that generally, when there is a low number of paintings, a majority of customers will leave the art gallery after having viewed all of the paintings. That being said, the customers who leave the art gallery early, generally have not seen a large percent of the remaining paintings. As we increase the number of paintings in the art gallery, we can see that most if not all of the customers leave early before having viewed all of the paintings, but the percentage of paintings not seen is significantly lower than when the gallery has a low number of paintings. We can also see that when the number of paintings is increased even further, the percentage of paintings not seen begins to decrease all while the number of people who leave early stays the same.

6.2 Potential problems and Improvements

The most significant limitation of this simulation is its lack of real-world grounding. Although many distributions and constants used in the simulation are not supported by empirical evidence, we believe that this model could serve as a foundation for future work aiming to collect the required data for estimating these population distributions. Some of the other assumptions, such as allowing instantaneous customer movement from one painting to another and having no limit on the number of simultaneously painting viewers.

In terms of the tested scenarios, the number of customers entering the art gallery was kept constant across all three to simplify the model. This limitation prevents us from determining how the system might perform with fewer or more customers and comparing the performances with the data we collected on how the number of paintings impacts the system.

It was noted in this study that our gallery system did not follow the classic Queuing System model studied in this course. Fitting this system to a known model could also improve the scope of the simulation. Future consideration should be given to a dynamic shortest path model, where the path length corresponds to the painting score.

6.3 Final Thoughts

In summary, our study sheds light on the relationship between the number of paintings displayed in an art gallery and the performance metrics related to customer experience. The clear trend observed demonstrates the potential impact of adjusting the number of paintings on visitor engagement and early departures.

However, it is important to acknowledge the limitations of the simulation, particularly the accuracy of the parameters used for the distributions. These parameters would benefit from further investigation and refinement to increase their reliability and enhance the overall validity of the simulation.

This simulation presents an intriguing and complex problem, as it goes beyond traditional queuing models and delves deeper into customer behavior. By focusing on factors such as crowd tolerance and painting preferences, the study highlights the importance of understanding the nuances of visitor interactions within such environments.

7 References

References

[1] Wu, K., Course Material, CSC446, Spring 2023

[2] Johnson, A., PySplay, `https://github.com/anoopj/pysplay`, 2017

8 Appendix

8.1 Source Code:

For access to the code used in this study, please see `https://github.com/rolstontim/Simulations-research-project`.