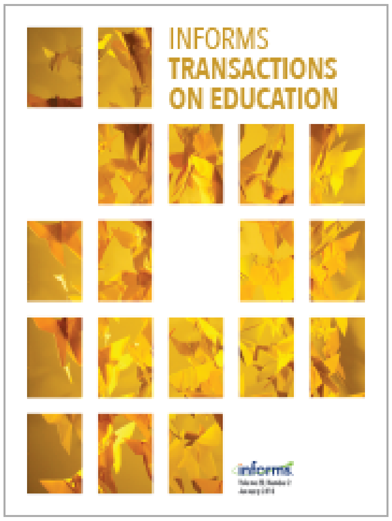

INFORMS Transactions on Education

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Case Article—Chick-fil-A Drive-Through: Managing Congestion with Discrete Event Simulation

Yuting Yuan

To cite this article:
Yuting Yuan (2026) Case Article—Chick-fil-A Drive-Through: Managing Congestion with Discrete Event Simulation. INFORMS Transactions on Education 26(3):200-206. https://doi.org/10.1287/ited.2023.0058ca

INFORMS TRANSACTIONS ON EDUCATION
Vol. 26, No. 3, May 2026, pp. 200–206
ISSN 1532-0545 (online)
https://pubsonline.informs.org/journal/ited

Case Article

Chick-fil-A Drive-Through: Managing Congestion with Discrete Event Simulation

Yuting Yuan[a]

[a] Smeal College of Business, Pennsylvania State University, University Park, Pennsylvania 16802

Contact: yxy5520@psu.edu, https://orcid.org/0000-0003-1720-0394 (YY)

Received: October 12, 2023
Revised: September 4, 2024; May 13, 2025; June 21, 2025
Accepted: June 30, 2025
Published Online in Articles in Advance: September 9, 2025

https://doi.org/10.1287/ited.2023.0058ca

Abstract. This case investigates the drive-through congestion of a fast-food restaurant with a series of learning modules inspired by computational operations research. Motivated by the renovation of a local Chick-fil-A restaurant, this case evaluates the effectiveness of various drive-through designs in reducing congestion-related problems such as long wait times and large car lines. The objective is to introduce students to a typical operational decision-making process that includes (1) modeling the drive-through operations, (2) encoding the service process using discrete event simulation, (3) predicting congestion metrics with simulators, and (4) prescribing optimal design based on simulator-generated predictions. With minimum mathematical analysis and a flexible amount of programming, this case showcases a cohesive decision-making pipeline and has great potential to spur students' interest in operations management through a well-known congestion problem.

Supplemental Material: The Teaching Note and course materials are available at https://www.informs.org/Publications/Subscribe/Access-Restricted-Materials.

Keywords: capacity management • queueing • simulation

1. Introduction

Congestion is ubiquitous in service systems because the demand often exceeds the limited supply of resources. Examples include overcrowded healthcare systems and long waiting lines at restaurants; both are associated with poor quality of service and low patient/customer satisfaction. Hence, a major challenge in managing service systems is mitigating the negative impacts of congestion through careful operational design. This case study introduces students to strategies for managing congestion problems and provides a real-life example to illustrate the practical implementation of these techniques.

Specifically, this case considers the drive-through design of a fast-food restaurant, Chick-fil-A, a well-known restaurant chain specializing in chicken sandwiches and related products. The focus is on drive-through congestion for the following reasons. Although the restaurant offers sit-down and in-store pickup services, a majority of Chick-fil-A revenue (60%, according to Business Insider 2024) is from drive-through customers who drive their cars along prespecified lanes to place and pick up orders. Because of the high demand and limited space, drive-through lanes are often very congested, resulting in slow service and long car lines. The delay in service potentially leads to customer abandonment (i.e., lost sales), and the car line congestion may cause severe traffic problems, including collisions and pedestrian injuries (Gibson 2022). Moreover, the massive car lines were reported to stretch into streets and parking lots regularly, blocking entrances to neighboring businesses. Several business owners sued Chick-fil-A, alleging that the congestion spillover damaged their businesses (Business Insider 2021). Some stores, for example, the Chick-fil-A in Santa Barbara, California (Business Insider 2023), had to rebuild the facility after receiving numerous complaints from residents and businesses. Fortunately, by leveraging modern technologies, such as drones for capturing aerial traffic footage, the restaurant can now continuously monitor traffic flow, identify bottlenecks, and optimize its operations for greater efficiency (The Wall Street Journal 2025). In this context, this case exemplifies a decision-making pipeline for designing drive-through services, a rigorous yet practical approach that can be generalized to similar congestion challenges.

The motivating story is the renovation project of a Chick-fil-A store at the large Marketplace Mall shopping center in Williamsburg, Virginia. The referenced news article (WY Daily 2022) provides the background information regarding this renovation project. Having long been blamed for the drive-through queue blocking the main traffic intersection of the shopping center, the Marketplace Mall Chick-fil-A purchased the adjacent lot and built a new facility with larger capacity. In my experience, students are more actively engaged in classes when examples are relevant to their daily lives. Indeed, students are very familiar with this establishment and show great interest in discussing this renovation and its impact on drive-through congestion.

This case aims to provide opportunities for students to learn about service operations and incorporate predictive and prescriptive analyses in tackling congestion problems. Although basic knowledge of R programming (R Core Team 2022) is required, the teaching plan can be customized based on students' programming levels. For example, I taught this case to data science undergraduate students who learned about the basic usage of R from entry-level statistics courses. The topics are divided into four tasks. The first task models Chick-fil-A and describes its operations with a flowchart that fully describes customers' trajectories and system resources. This example highlights the art of mathematical modeling, that is, constructing a concise description of the problem with the capability of drawing useful insights. Second, students learn to encode service operations with discrete event simulation (DES), which is widely used in practice for simulating service systems. With the help of the simulator, the third task identifies the bottleneck resources that cause inferior drive-through services. Generally, simulation allows system performance to be checked under various system configurations without actually implementing them in reality. In this sense, DES can be considered a *prediction* method that outputs system performance given a parameterized input model. Finally, using a grid search, students *prescribe* optimal drive-through design among several candidate designs inspired by the bottleneck analysis. Specifically, this case discusses two different cost formulations when searching for the best design, reflecting various business objectives in decision-making.

The remainder of this article is organized as follows. Section 2 reviews the relevant literature. Section 3 outlines the teaching plan with potential customization. Section 4 discusses students' learning experiences and feedback, and Section 5 concludes the paper.

2. Literature

Increasing operational efficiency is crucial for service systems, including healthcare (Armony et al. 2015, Chan et al. 2017), call centers (Harrison and Zeevi 2005, Senderovich et al. 2015), border security check (Zhang et al. 2011), etc. To analyze these systems, queueing theory (dated back to Erlang 1909) is one of the most common methodologies, and the fundamental idea is to describe the system evolution with stochastic processes. The rich mathematical theory allows us to *predict* important congestion-related metrics, such as waiting times (Ibrahim and Whitt 2011, Kerner et al. 2021) and queue length (Fendick and Whitt 1989), or *prescribe* operational decisions to mitigate system congestion through scheduling (Long et al. 2020), admission control (Koçağa and Ward 2010), and dynamic staffing (Hu et al. 2025, Yuan 2025). Considering drive-through congestion, this work aims to predict waiting time and car lines and ultimately decide on the best drive-through design. In addition to research articles on drive-through designs (Swart and Donno 1981, van de Kracht and Heragu 2021), to the best of my knowledge, this work is the first educational case that studies drive-through congestion using queueing analysis. A particularly interesting element in our model is the prediction of the car line, which treats the car spaces as constraining resources.

Because of the complexity of the drive-through model, the metrics of interest are derived using discrete-event simulation (DES; Leemis and Park 2006), which is widely used in practice to evaluate performances of large-scale service systems (Kincaid and Alexandrov 2005, Zhang 2018, Bravo et al. 2024). The DES simulator serves as an approach to predict the queue performance given any configuration of the drive-through design. Because predictions can be accurately generated using simulation, the decision-maker searches for the best design with the smallest predicted cost. In this sense, this framework provides opportunities for students with minimum mathematical background to predict/prescribe for a complicated queueing system. Another value of this framework is the integration of mathematical modeling with data analytics. Parameters of the input model can be estimated from field studies. Moreover, the simulator developed from a queueing model generates fictitious data that guides the design decisions for drive-through lanes. Hence, this case demonstrates one real-world example of data analytics in operations management (Mišić and Perakis 2020), an emerging topic with significant practical value.

Finally, this case contributes to developing educational cases for managing congestion-prone service systems. Because of the complications in queueing analysis, cases of queues are relatively sparse and are typically taught with spreadsheet modeling (Grossman 1999, Ingolfsson and Grossman 2002, Leong and Ma 2024). We provide an alternative using DES with ready-to-use simulators. Because data analytics is a trending topic in OR classrooms, instructors can also utilize this case to teach fundamental programming skills.

3. Teaching Plan

3.1. Learning Objective

In addition to understanding a typical operational decision-making pipeline, students should be able to perform the following technical analysis, each corresponding to one of the four tasks outlined in the Teaching Note:

1. Model service operations and describe with a flowchart.
2. Learn to encode the flowchart with discrete event simulation (DES).
3. Predict performance metrics using DES.
4. Prescribe the best drive-through design using DES-based predictions.

I covered the above topics in two 80-minute lectures. The first lecture is dedicated to modeling and programming stochastic systems. The second lecture utilizes DES simulators to predict and optimize drive-through performance. Each task is accompanied by sample exercise questions designed to reinforce students' understanding of key concepts.

3.2. Student Background

This case is suitable for both undergraduate and graduate students with foundational knowledge of R programming as well as basic knowledge of probability and statistics. Students from diverse academic backgrounds, such as operations research, statistics, data science, engineering, and business analytics, can benefit from case discussions focused on business decision-making. I introduced this case in a capstone elective course for senior data science undergraduate students who had previously learned about R in their introductory statistics courses. Despite having limited experience with operations research, these students were able to follow the modeling analysis with no difficulties and could finish coding the processes with some guidance from the instructor when roughly 80% of the code was provided.

Considering the potential variation in the programming background, in addition to the provided R scripts, this case is accompanied by an online companion notebook (link provided in the Teaching Note) that walks through all the code and discussions in R Markdown format. Students with minimal programming experience can take the existing simulation functions and perform all analyses according to the instructions in the notebook. I previously assigned the entire notebook as reading material to undergraduate data science students (mostly sophomores) and asked conceptual questions, all of which could be answered using the content in the notebook.

3.3. Teaching Plan

Because the overall course topics may vary, I recommend assigning selected readings before the first lecture of this case, including an introduction to mathematical modeling (chapter 1 in Bender 2000) and queueing models for service systems (chapter 1.2, chapter 2.1, chapter 2.2, and chapter 3 in Shortle et al. 2018). In my capstone OR course, these materials were covered in earlier lectures leading up to this case. As a result, students had already developed a foundational understanding of modeling service systems, including the challenges of stochasticity in service operations, common probability distributions for arrival and service processes, and the critical roles of system resources in mitigating congestion. In addition, students are expected to read the case in advance to gain a general idea of the business problem and come prepared to participate in class discussions.

3.3.1. Model Service Operations. The lectures begin with a brief overview of the case, highlighting the congestion problem of the Chick-fil-A drive-through. Because the design of this chain restaurant varies by location, I recorded my own experience at the focal Chick-fil-A drive-through featured in this case. This short video, included in the teaching materials, is played in class as a visual introduction to the modeling exercises. After watching the video, students are given several minutes in class to draw a flowchart for the process. Their work will likely vary, creating great opportunities for the instructor to discuss potential issues in their models. The purpose of drawing flow charts is twofold. First, students are guided to recognize the fundamental questions in mathematical modeling: How does the model help address the objective? How to select variables to be included? Is this model useful—that is, can it be estimated from data and used to make predictions? Can it be validated in practice? With hands-on exercises, students learn to strike a balance between precision and conciseness in the model-building process. Second, sketching a flowchart is a preliminary step for the subsequent simulation task. Because commercial DES simulation software (e.g., simEvents of MATLAB, Arena simulation) essentially encodes the flowchart/diagram with predefined building blocks, learning to draw flow charts prepares students to develop simulations in many commonly used DES software.

3.3.2. Build DES Simulators. Given the customer's flowchart, the second task simulates the drive-through implemented with the simmer package (Ucar et al. 2019) in R programming. Similar packages in other languages, such as SimPy (Python) and SimJulia (Julia), follow the same generic framework. The idea is to encode a typical customer's trajectory (i.e., a sequence of activities) and specify the resources and processing time associated with each activity. We provide a simple

Figure 1. A Simulator Example

(a) Set up the customer's trajectory

```
customer = trajectory("Customer's path") %>%
  seize("cashier",amount=1) %>%
  timeout(function() {rexp(1, 0.8)}) %>%
  release("cashier",amount=1) %>%

  seize("cook",amount=1) %>%
  timeout(function() {rexp(1, 1/5)}) %>%
  release("cook",amount=1)
```

(b) Create the simulation environment

```
env = simmer()
env %>%
  add_resource("cashier", 2) %>%
  add_resource("cook", 3) %>%
  add_generator("Customer", customer,
                function() rexp(1, 1/1.8)) %>%
  run(100)

# Retrieve the simulation data
get_mon_arrivals(env, per_resource=TRUE)
get_mon_resources(env)
```

example in Figure 1 to illustrate the key components in a DES simulator. Specifically, Figure 1(a) codes a customer's trajectory, where he or she starts by utilizing a resource named "cashier" to place an order. Because ordering needs only one cashier, the number of resources required for this step is 1 ("amount = 1"). The cashier's processing time is given by the timeout() function. In this example, it is exponentially distributed with a rate of 0.8 orders/minute. After the order is taken, the customer leaves the "cashier" resource and utilizes the "cook" resource, representing the worker who prepares this order. The customer leaves the system after a processing time indicated by the second timeout() function. Figure 1(b) shows the other component of the simulator, that is, the parameters for the environment, including the arrival process (via add_generator() function) and total number of resources (via add_resource() function). In the example, we add two cashiers and three cooks to the system. The customers arrive sequentially with an interarrival time that is exponentially distributed with a rate 1/1.8 customers/minute. Queues occur when the customer attempts to "seize" a particular resource, but none is available because all are busy serving other customers. After the simulator finishes running for 100 minutes (simulator time, not actual run time), one can retrieve the data of arrivals using the get_mon_arrival() function and all the events that have changed the state of the resources using the get_mon_resources() function.

This case includes two DES simulators representing the Chick-fil-A drive-through before and after the renovation (i.e., two different drive-through designs mentioned in the case). Each simulator generates two key performance metrics defined in the case: (1) *customers' flow time* (i.e., the total time a customer spends in the drive-through), and (2) *system carline* (i.e., the sum of carlines at the ordering and pickup stations). Depending on the students' programming experience, the instructor may choose to review the simulator code or live-code some key elements in class. Each simulator includes a parameter block (see e.g., Figure 2), which can be tuned to represent different drive-through designs—varying by staffing levels (the number of cashiers and cooks) and the number of car spaces in the drive-through lanes. With fixed model parameters, students run two types of simulations for one drive-through design (organized into separate blocks in the R script): *a single run* that simulates the drive-through for a sufficiently long time and *a Monte Carlo experiment* that repeats the single-run simulation 1,000 times. The advantage of the Monte Carlo experiments in DES becomes evident as students compare the outputs, where aggregating repeated runs captures the inherent variability in stochastic systems and results in more robust and generalizable insights. A possible customization in the teaching plan is to conduct a field study to estimate model parameters based on a local Chick-fil-A facility. The provided simulators are easily adaptable to incorporate empirical data, for instance, by replacing the interarrival time in the "add_generator()" function with an empirical distribution or a sampling function that randomly selects from a recorded set of interarrival times. At this point, a key takeaway is the versatility of the DES simulators in assessing and comparing the performance of different drive-through designs. After analyzing the focal drive-through before and after the renovation, some curious students may wonder about the underlying causes of the performance differences, naturally leading to the next task that examines the relationship between system performance and bottleneck resources.

Figure 2. Model Parameters

```
#################################
########### Model Parameters ####
#################################
# time unit 1 minute
simTime = 2000

# cashier service rate
mu_cashier = 0.8

# cook service rate
mu_cook = 1/3.5

# number of cashiers
num_cashier = 2

# number of cooks
num_cook = 3

# car spaces in ordering lane
space_ordering = 200

# car spaces in pickup lane
space_pickup = 6

# Poisson arrival rate of customers
arrival_rate = 0.8
```

3.3.3. Identify Bottleneck Resource. Identifying bottlenecks is undoubtedly crucial in optimizing investment allocations. Another usage of the simulators is that by using the "get_mon_resources()" function, students can monitor and visualize evolutions of the state of each resource (utilization level, queue length, etc.) over time. Observing extremely long system carline or prolonged flow time at certain moments naturally prompts the question of whether additional resources are needed and, if so, which resource shall be prioritized. Because of the dynamic and stochastic nature of the drive-through process, the analysis focuses on average performance metrics—specifically, the average flow time across all customers and the average system carline over the entire simulation horizon. Using the previously built simulators, students can conduct a *sensitivity analysis*, a widely used technique for identifying key drivers of system performance. This approach evaluates how the system responds to an incremental increase in a focal resource while holding all other resources constant. A resource is considered a bottleneck for a particular performance metric if increasing its capacity leads to a substantial improvement in that metric. Following this strategy, I typically give students several minutes in class to adjust model parameters in the simulators, record the resulting average performance metrics, and pinpoint the bottleneck resource accordingly.

3.3.4. Business Decision-Making. The final task searches for the best drive-through design, where a design corresponds to a set of decisions, including the staffing level at each station and whether to implement a large-scale renovation project. Because the "best" design depends on the definitions of business objectives, this case explores two alternatives: (1) minimizing operational costs subject to constraints on flow time and system carline and (2) minimizing the sum of operational costs and lost sales, where the latter reflects customer abandonment due to a long system carline upon arrival. Students are asked to propose several designs based on the bottleneck analysis, use simulators to calculate the cost function for each design, and ultimately identify the design with the minimum cost. This simulation-based grid-search methodology is straightforward to implement and does not require students' prior knowledge of optimization or a complex analytical formula mapping system input parameters to costs. All my students can complete this part of the analysis independently without difficulty. This exercise conveys the following key messages. The business objective is fundamental in formulating an accurate optimization problem, underscoring the importance of collaborating with practitioners to derive relevant solutions. Furthermore, even with a well-defined business objective, there can be multiple ways to formulate the optimization problem mathematically. Students may be able to connect the two optimizations explored in this case by viewing the first problem with hard performance constraints as an approximation of the second problem. Because it is generally challenging to calculate abandonment and estimate its economic value, optimizing service operations with target performances can be relatively easier to implement and practically more appealing in real-world applications.

4. Classroom Experience

I have used this case twice since 2023 in a capstone undergraduate course about data analytics in operations research (OR). Each year, the audience consisted of about 15 senior data science students with programming experience but limited exposure to OR. Spanning two 80-minute lectures, this case was introduced in the middle of the semester after lectures on mathematical modeling and tutorial examples of DES (available at Ucar and Smeets 2025). Because of their relatively strong programming background, students in this class were assigned tasks to complete the partially provided simulator script. In another entry-level data science course with mostly sophomores (25 in each semester), the companion R notebook (link in the Teaching Note) was provided as reading materials in one homework assignment with selected exercise questions.

According to students' performance and my observations, the main challenge of this case is constructing the simulators, including devising mathematical models and programming the processes. Once the simulators are defined, the bottleneck identification and decision-making tasks are straightforward tuning of model parameters. Students are generally excited about this topic because they have all visited Chick-fil-A and have been annoyed with the long waiting time in the drive-through. Thanks to their relevant experiences, they can quickly grasp the definitions of flow time and system carline. Students particularly enjoy the coding assignments where approximately 80% of the code is provided and they are responsible for filling in blanks such as distributions of interarrival time and service time. Although much of the code is given, these programming exercises help students understand the code structure, syntax, and the purposes of each block, preparing them to proceed to subsequent tasks. All students in the capstone course performed exceptionally well—they actively participated in class discussions and successfully completed assessments, including programming, conceptual, and quantitative exercise questions. The median score on the case-related assignments is more than 90%.

Student feedback on this case has been generally positive. They have demonstrated great interest in class, and some have even reached out after graduation to share their thoughts when they have come across news articles about Chick-fil-A drive-through. Although in the past I did not collect formal surveys specifically about this case, the course evaluation, collected shortly after we finished the case discussions, could serve as a close proxy for student perceptions. By the time students completed the evaluations, this case was the only case study that built upon materials covered in previous lectures. Below are some quotes from students' evaluations, and I have been revising materials according to their feedback, aiming to improve the flow and clarify certain questions.

- "The lectures were very informative and relevant, and I feel like I learned a lot in this class. The assignments were also very interesting and helpful in improving my understanding."
- "Professor does a great job of connecting the coursework with real-world examples that help students understand the material better."
- "I learned a lot from the class, and I found the topic of operations research to be very interesting."
- "For some of the homework assignments and the midterm, it was hard to understand exactly what was being asked for. I really enjoyed this class, but some more clarity would be great."
- "I could tell that this was the first time that it was taught in the sense that some of the design was a bit bumpy, but overall I felt like I learned a lot and enjoyed the material."

Although none of the students had a prior background in operations research, they were exposed to many fundamental ideas through this single case study. In our class conversation, many of them expressed interest in further studies in OR after seeing the potential of these techniques in tackling such a relevant and well-known congestion problem. Notably, a few students majoring in data science went on to pursue graduate studies in computational operational research after graduation.

5. Conclusion

This case studies Chick-fil-A drive-through congestion through the lens of operations research. I design four modules that introduce several fundamental OR techniques to students at all levels. The main advantages of this case are as follows. First, it is based on a real-life application that strongly motivates students' learning interests. Second, although generally operations research can be very technical, this case presents key ideas using easy-to-implement examples suitable for students with various programming backgrounds. Students can also form a high-level understanding of operations management because these four modules naturally connect as a cohesive and practical decision-making pipeline. Finally, the teaching plan is highly flexible, so the instructors can customize the course materials according to their own design. Although the provided code is in R, the simulators can be written in other languages, such as Python, by following the same framework with simple changes in the syntax. Overall, this Chick-fil-A case could have a great impact in classrooms by spurring students' interest in operations research/management.

Acknowledgments

The author thanks Professor Stefan Creemers, the associate editor, and the two anonymous reviewers for their constructive and insightful suggestions throughout the review process. The author is also grateful to her students at William & Mary for their thoughtful discussions and feedback that helped with the development of this case.

References

Armony M, Israelit S, Mandelbaum A, Marmor YN, Tseytlin Y, Yom-Tov GB (2015) On patient flow in hospitals: A data-based queueing-science perspective. *Stochastic Systems* 5(1):146–194.

Bender EA (2000) *An Introduction to Mathematical Modeling* (Courier Corporation, Mineola, NY).

Bravo F, Rudin C, Shaposhnik Y, Yuan Y (2024) Interpretable prediction rules for congestion risk in intensive care units. *Stochastic Systems* 14(2):111–130.

Business Insider (2021) Frustrated business owners are complaining about Chick-fil-A's massive drive-thru lines. Accessed September 28 2023, https://www.businessinsider.com/chick-fil-a-drive-thru-lines-infuriate-nearby-business-owners-2021-2.

Business Insider (2023) Chick-fil-A is rebuilding a drive-thru in California after residents complained about traffic congestion. Accessed September 9, 2023, https://www.businessinsider.com/chick-fil-a-rebuilds-santa-barbara-drive-thru-traffic-complaints-2023-5.

Business Insider (2024) How Chick-fil-A makes such high sales. Accessed August 31, 2024, https://www.businessinsider.com/how-chick-fil-a-makes-such-high-sales-drive-thru-2024-4.

Chan CW, Farias VF, Escobar GJ (2017) The impact of delays on service times in the intensive care unit. *Management Sci.* 63(7):2049–2072.

Erlang AK (1909) The theory of probabilities and telephone conversations. *Nyt. Tidsskr. Mat. Ser. B* 20:33–39.

Fendick KW, Whitt W (1989) Measurements and approximations to describe the offered traffic and predict the average workload in a single-server queue. *Proc. IEEE* 77(1):171–194.

Gibson K (2022) California city may declare chick-fil-a a "public nuisance". *CBS News* (March 17), https://www.cbsnews.com/news/chick-fil-a-santa-barbara-california-city-public-nuisance-traffic/.

Grossman TA Jr (1999) Teachers' forum: Spreadsheet modeling and simulation improves understanding of queues. *Interfaces* 29(3):88–103.

Harrison JM, Zeevi A (2005) A method for staffing large call centers based on stochastic fluid models. *Manufacturing Service Oper. Management* 7(1):20–36.

Hu Y, Chan CW, Dong J (2025) Prediction-driven surge planning with application to emergency department nurse staffing. *Management Sci.* 71(3):2079–2126.

Ibrahim R, Whitt W (2011) Wait-time predictors for customer service systems with time-varying demand and capacity. *Oper. Res.* 59(5):1106–1118.

Ingolfsson A, Grossman TA Jr (2002) Graphical spreadsheet simulation of queues. *INFORMS Trans. Ed.* 2(2):27–39.

Kerner Y, Roet-Green R, Senderovich A, Shaposhnik Y, Yuan Y (2021) Waiting-time prediction with invisible customers. *Manufacturing Service Oper. Management*, ePub ahead of print July 4, https://doi.org/10.1287/msom.2022.0098.

Kincaid RK, Alexandrov N (2005) Scale-free networks: A discrete event simulation approach. *Internat. Conf. Comput. Sci.* (Springer), 1051–1058.

Koçağa YL, Ward AR (2010) Admission control for a multi-server queue with abandonment. *Queueing Systems* 65(3):275–323.

Leemis LM, Park SK (2006) *Discrete-Event Simulation: A First Course* (Pearson Prentice Hall, Upper Saddle River, NJ).

Leong TY, Ma NL (2024) Spreadsheet simulation of priority queues. *INFORMS Trans. Ed.* 25(1):1–22.

Long Z, Shimkin N, Zhang H, Zhang J (2020) Dynamic scheduling of multiclass many-server queues with abandonment: The generalized $c\mu/h$ rule. *Oper. Res.* 68(4):1218–1230.

Mišić VV, Perakis G (2020) Data analytics in operations management: A review. *Manufacturing Service Oper. Management* 22(1): 158–169.

R Core Team (2022) *R: A Language and Environment for Statistical Computing* (R Foundation for Statistical Computing, Vienna, Austria).

Senderovich A, Weidlich M, Gal A, Mandelbaum A (2015) Queue mining for delay prediction in multi-class service processes. *Inform. Systems* 53:278–295.

Shortle JF, Thompson JM, Gross D, Harris CM (2018) *Fundamentals of Queueing Theory*, vol. 399 (John Wiley & Sons, Hoboken, NJ).

Swart W, Donno L (1981) Simulation modeling improves operations, planning, and productivity of fast food restaurants. *Interfaces* 11(6):35–47.

The Wall Street Journal (2025) Drone and 'Game Film': Inside Chick-fil-A's quest to make fast food faster. Accessed May 8, 2025, https://www.wsj.com/business/hospitality/chick-fil-a-drive-through-service-fast-food-innovation-ed5e2288?st=ApoMpQ\&reflink=desktopwebshare_permalink.

Ucar I, Smeets B (2025) Discrete-Event Simulation (DES) package for R. https://r-simmer.org/.

Ucar I, Smeets B, Azcorra A (2019) simmer: Discrete-event simulation for R. *J. Statist. Software* 90(2):1–30.

van de Kracht T, Heragu SS (2021) Lessons from modeling and running the world's largest drive-through mass vaccination clinic. *INFORMS J. Appl. Anal.* 51(2):91–105.

WY Daily (2022) Monticello marketplace Chick-fil-A expansion to provide more streamlined system for guests, employees. Accessed August 31, 2024, https://wydaily.com/brews-and-bites/2022/03/08/monticello-marketplace-chick-fil-a-expansion-to-provide-more-streamlined-system-for-guests-employees/.

Yuan Y (2025) Managing flexible capacity in service systems with worker shortages. *Manufacturing Service Oper. Management* 27(3): 808–824.

Zhang X (2018) Application of discrete event simulation in health care: A systematic review. *BMC Health Services Res.* 18(1):1–11.

Zhang ZG, Luh HP, Wang CH (2011) Modeling security-check queues. *Management Sci.* 57(11):1979–1995.