

MODELLING THE OPERATION OF A DYNAMIC QUEUE MANAGEMENT SYSTEM IN SUPERMARKETS

Larysa Gumeniuk, Volodymyr Lotysh, Pavlo Humeniuk

Lutsk National Technical University, Lutsk, Ukraine

Abstract. This study investigates key aspects of customer service optimization within retail environments. An algorithm was developed and implemented to construct a discrete-event simulation model of a supermarket checkout system, enabling the analysis of various queue organization configurations. The model facilitates comparative evaluation of their operational efficiency based on predefined performance criteria. Furthermore, a dynamic checkout management algorithm was designed and programmed to adjust the number of active service counters in real time. This mechanism responds adaptively to the current customer arrival rate and queue lengths, with the objective of minimizing customer waiting time and balancing cashier workload. The simulation model enables the assessment of critical performance indicators, including average customer waiting time, queue length, checkout utilization rate, the number of customers who abandon the queue before being served, and the total number of customers processed. The results of the simulation provide a robust analytical basis for informed decision-making regarding the optimal number and types of checkout counters, workforce management strategies, and improvements in customer experience.

Keywords: queueing systems (QS), simulation, identification, data analysis

MODELOWANIE DZIAŁANIA DYNAMICZNEGO SYSTEMU ZARZĄDZANIA KOLEJKAMI W SUPERMARKETACH

Streszczenie. W artykule przeanalizowano kluczowe aspekty optymalizacji obsługi klienta. Opracowano algorytm i implementację programową dyskretno-zdarzeniowego modelu symulacyjnego stanowiska kasowego w supermarkecie dla różnych konfiguracji organizacji kolejek, co pozwala na analizę porównawczą ich efektywności według określonych kryteriów. Opracowano algorytm i implementację programową dynamicznego sterowania liczbą funkcjonujących stanowisk kasowych, które dostosowują się do bieżącej intensywności przepływu klientów i długości kolejek w celu optymalizacji czasu oczekiwania i obciążenia personelu. Model umożliwia oszacowanie kluczowych wskaźników wydajności, takich jak średni czas oczekiwania klienta, długość kolejki, wskaźnik wykorzystania kas, liczba klientów, którzy opuścili kolejkę bez oczekiwania na obsługę oraz całkowita liczba obsłużonych klientów. Wyniki symulacji można wykorzystać do podejmowania świadomych decyzji dotyczących liczby i typów kas, strategii zatrudnienia i obsługi klienta.

Słowa kluczowe: systemy kolejkowe (QMS), modelowanie, identyfikacja, analiza danych

Introduction

In the contemporary landscape of retail, where customers increasingly value their time, the efficient management of queues has become a critical factor in the success of supermarkets. Prolonged waiting times remain among the most prevalent sources of customer dissatisfaction. Excessive delays in service or extended queuing often lead to frustration, potentially deteriorating the overall customer experience within the retail environment. Inefficient queue management not only undermines customer satisfaction but also translates into direct financial losses, as shoppers may abandon their purchases or opt to avoid stores characterized by long waiting lines. Conversely, a well-structured queueing system significantly enhances customer satisfaction, and swift, efficient service at the checkout can directly contribute to increased sales volumes.

Queue management system modelling serves as a vital tool for analyzing and optimizing service processes in supermarkets. This approach encompasses the study of customer flow dynamics, the application of queueing theory and discrete-event simulation, as well as the integration of technology to enhance queue efficiency. The primary objective of such modelling is to minimize customer waiting time, prevent overcrowding, and ensure effective and streamlined service delivery.

An effective queue management system in supermarkets is founded on several core principles. Foremost among these is the organization and control of customer flow during the waiting period for service. The primary objective is to reduce waiting times while ensuring fair and efficient service for each customer. Queue management systems are designed to optimize the customer service process by effectively managing incoming flows. These systems aim to regulate customer traffic, minimize wait times, enhance staff productivity, and improve overall service quality.

An essential aspect is maintaining a balance between customer expectations for minimal waiting times and the operational efficiency of resource utilization within the supermarket, such as cashiers and checkout counters. Effective queue management systems not only organize waiting lines but also provide tools

for systematically managing customer flow and enhancing their experience throughout the entire service process.

In recent years, there has been a growing trend toward the exploration and implementation of technological and organizational innovations aimed at optimizing service processes in the retail sector [1]. The increasing availability of computational resources and the advancement of software tools for modelling complex systems have opened new opportunities for the analysis and improvement of queue management systems. However, many existing approaches do not fully account for the dynamic nature of customer demand and the specific operational characteristics of individual supermarkets, which limits their overall effectiveness.

These aspects highlight the high relevance of developing and investigating queue management system models capable of adapting to changing conditions and optimizing key performance indicators of a supermarket's checkout area. In this context, the use of simulation modelling methods becomes particularly important for evaluating various queue management strategies and identifying the most effective solutions.

1. Analysis of research and publications

In the retail sector, particularly in supermarkets, several types of queue management systems are employed, each with its own advantages and limitations:

- 1) Multi-line queues (traditional queueing systems) – these are based on the "first come, first served" principle [11].
- 2) Single-line queue systems – in such systems, all customers wait in a single queue and proceed to the first available checkout counter [5].
- 3) Virtual queue systems (software-based) – customers register remotely via a mobile application, website, or interactive kiosk and receive an electronic queue number. Information regarding their waiting status is updated in real-time, allowing the customer to view their position in the queue, the estimated waiting time, and the approach of their turn [8].
- 4) Appointment-based queue systems – these systems operate on a time-slot reservation principle, where the customer

selects a specific time window for their visit in advance [4]. This format is widely used in institutions where accuracy and load stability are critical – such as healthcare, administrative, or service sectors – but is increasingly being integrated into the retail environment.

- 5) Hybrid queue systems – these are comprehensive solutions that combine functional elements of several queue types, including multi-line, single-line, virtual, and appointment-based systems [15].
- 6) Self-service queue systems – these systems minimize the involvement of personnel in the service process. Customers independently complete all service stages using interactive kiosks, scanners, or mobile applications [10].
- 7) Priority queue systems – these systems organize customer service not based on the order of arrival, but according to predefined priority rules. For instance, customers who are part of loyalty programs, those with fewer items, or individuals who have paid for premium service access may be served first.
- 8) AI-based queue management systems – these systems rely on the application of artificial intelligence and machine learning algorithms to automatically analyse customer behaviour and operational processes. Their operation involves the use of video analytics to monitor incoming foot traffic, queue lengths, and service speed, as well as integration with POS systems and sensors to obtain accurate and real-time data. The system predicts workload, detects congestion, and generates real-time recommendations for staff allocation or opening additional checkout counters [16].

Queueing theory provides a robust analytical framework for designing and evaluating the performance of queueing systems, including those deployed in supermarkets [13]. It facilitates understanding and forecasting queue behaviour, such as customer waiting times, queue lengths, and the utilization rate of service channels.

The analysis of such systems is typically based on classical queueing theory methods, particularly M/M/1 and M/M/c models, which offer fundamental analytical estimates of waiting time, server utilization, and service efficiency. However, due to their simplified assumptions, these models have limited applicability to real-world scenarios. As a result, contemporary research increasingly focuses on simulation-based modelling and optimization algorithms.

The M/M/1 model represents a fundamental single-server queueing model within the queueing theory framework [17]. It is grounded in a set of key assumptions that define its mathematical structure and applicability.

Firstly, it is assumed that the customer arrival process (Markovian Arrival) follows a Poisson distribution with parameter λ , which characterizes the average rate of incoming flow. This assumption implies that the inter-arrival times between customers are independent random variables that follow an exponential distribution.

Secondly, the service time for each customer (Markovian Service) is also assumed to follow an exponential distribution with parameter μ , representing the average service rate. This implies a memoryless property of the service process – meaning the duration of the current service is independent of any previous services.

A key characteristic of the model is the presence of only one service unit (One Server), which limits the system's throughput capacity. The queue in such a system is assumed to have infinite length (Infinite Queue Capacity), allowing all customers to wait indefinitely for service without loss.

The service discipline adheres strictly to the "First Come, First Serve" (FCFS) principle, excluding any prioritization or special servicing rules. These assumptions enable the derivation of analytical expressions for the system's fundamental performance indicators, although they impose certain limitations on the model's applicability in real-world scenarios.

This model is applied to the analysis of individual checkout lines in supermarkets, enabling the evaluation of key performance indicators such as average waiting time and queue length.

The M/M/c model is an extension of the M/M/1 model for multi-server systems, where "c" denotes the number of parallel service channels operating a single shared queue [6]. The assumptions of this model are similar to those of M/M/1, except for the presence of multiple servers. The M/M/c model is effectively used to simulate scenarios involving multiple checkout counters in a supermarket.

Queueing theory also enables the modelling of multi-server systems with priorities, such as express lanes designated for customers with a limited number of items [9]. These systems incorporate various classes of customers with differentiated service priorities, which can be implemented by assigning distinct arrival and service rates for each customer class.

The introduction of priority-based queueing systems aims to enhance customer satisfaction for specific segments – for instance, customers making small purchases – by reducing their waiting times. However, the effectiveness of such systems depends on the arrival intensity and service rates of different customer types, as well as the chosen classification threshold (e.g., the maximum number of items permitted for express checkout) [13].

Discrete-event simulation (DES) is a powerful modelling technique that represents systems as sequences of events occurring at discrete points in time [3]. Unlike analytical models derived from queueing theory, DES allows for a more realistic simulation of queueing behaviour in supermarkets by accounting for the stochastic nature of arrivals and service times, as well as the individualized experience of each customer.

The *next-event time advance* algorithm is the fundamental mechanism driving discrete-event simulation, controlling the progression of the simulation clock by jumping from one event to the next. This approach enables efficient modelling of complex systems with a large number of interacting elements, where events occur irregularly over time. It allows the model to "focus" only on state-changing events, which is particularly valuable for practical applications such as queue management in supermarkets, logistics hubs, or telecommunications networks [3].

Agent-Based Modelling (ABM) is a modern approach to studying complex systems that relies on simulating the interactions of autonomous agents. Unlike classical methods that analyse the system as a whole, ABM focuses on individual entities, their behaviour, and their mutual influence. This enables the reproduction of real-world processes with high fidelity, accounting for the unique characteristics of each system participant.

Within the ABM framework, network models can be applied to represent the spatial organization of a supermarket and to simulate customer movement between its various zones. These models allow for the exploration of both customer-to-customer interactions, such as congestion in aisles, and customer-to-environment interactions, such as the time spent in specific departments. The complex behaviour of the system emerges from numerous simple agent actions. These phenomena are not pre-defined but arise organically through the interactions. A prominent example is the formation of queues, which are not the result of centralized control, but emerge from the aggregate of individual decisions made by customers and staff members [2].

To optimize the number and configuration of checkout counters in supermarkets, a range of optimization algorithms is employed, with particular attention given to Genetic Algorithms (GA) and Particle Swarm Optimization (PSO).

In the context of optimizing the quantity and placement of checkouts, GA begins by generating an initial population of various checkout configurations. Each configuration is encoded as a set of parameters describing the number of open counters, their locations, and the allocation of resources. A fitness function is then used to evaluate the effectiveness of each configuration.

This function may take into account factors such as the average customer waiting time, cashier workload, and the supermarket's operational costs. A selection process follows, whereby the most fit solutions from the population are chosen to produce a new generation. The crossover operation enables the exchange of information between two selected solutions, combining their best attributes. As this iterative process (generation cycles) continues, the population gradually "evolves" toward an optimal checkout configuration. The algorithm can also incorporate real-world constraints, such as the maximum number of available staff, the physical layout of the retail space, or equipment specifications. This approach allows for effective solution discovery even in complex and high-dimensional problems where traditional optimization methods struggle due to the vast solution space and nonlinearity of the system [7].

Swarm algorithms, or Particle Swarm Optimization (PSO), are optimization techniques inspired by the collective behaviour of natural swarms, such as flocks of birds or schools of fish. In the context of optimizing checkout counters in supermarkets, PSO is employed to identify the optimal number and spatial configuration of counters, where each particle in the swarm represents a potential solution.

During the execution of the algorithm, each particle moves within a multidimensional search space, updating its position based on two key factors: its own best-known solution and the best-known solution found by the entire swarm. This interaction facilitates convergence toward optimal or near-optimal solutions in a rapid and coordinated manner [14].

Similar to Genetic Algorithms, PSO effectively handles complex optimization problems, particularly when the solution space is vast and nonlinear. Due to its ability to quickly adapt to dynamic conditions, PSO is frequently utilized to improve service efficiency and reduce customer waiting times in supermarket environments.

Machine learning techniques are also employed for queue management in supermarkets, offering adaptability and accuracy in decision-making processes [12]. One prominent approach is reinforcement learning (RL), particularly Q-learning. In this case, the system operates as an agent interacting with the dynamic environment of the supermarket, learning to optimally open or close checkout counters. Through repeated iterations, the agent develops a policy by receiving rewards for actions that, for example, reduce the average customer waiting time or enhance overall service quality. This mechanism allows the model to adapt in real time to various operational scenarios and make flexible, context-dependent decisions.

Another promising direction involves predictive modelling using Long Short-Term Memory (LSTM) networks, which are specifically designed for time-series data analysis. These models can capture temporal dependencies in customer flow data, enabling accurate forecasts of checkout load throughout the day or week. Such predictions support proactive scheduling of staff and adjustment of the number of open counters during peak periods, thereby minimizing queue formation and improving personnel efficiency.

Real-time systems enable responsive queue management by continuously analysing incoming data [18]. Kalman filters play a key role in this approach. These algorithms estimate the actual state of a system based on noisy or incomplete observations. In the supermarket context, Kalman filters are used to refine predictions of checkout load by incorporating real-time sensor data, such as footfall counters at entrances or checkout zones. Through an iterative process involving prediction and update steps, the Kalman filter provides a more accurate representation of the system state compared to simpler models.

An important complement to the aforementioned approaches is the mechanism for dynamic opening and closing of checkout counters. Based on predefined thresholds for queue length or customer flow intensity, the system makes real-time decisions about opening additional checkouts to reduce congestion or closing surplus counters when demand subsides. This enables

optimal resource utilization and minimizes customer waiting time without operator intervention.

Queue modelling in supermarkets is significantly more complex than traditional queue modelling due to the multifactorial dynamics of customer behaviour and the specific characteristics of the retail environment.

First and foremost, it is essential to account for the complex behaviour of customers. Shoppers do not merely stand in queues; they interact with the store environment, select queues based on various strategies, may abandon queues, or change decisions in response to factors such as queue length, pricing, or emotional state. Traditional queue models often assume homogeneous customer behaviour and fixed rules for entering queues.

Another important distinction lies in the spatial context. In supermarkets, the physical layout of products, checkout counters, and aisles substantially influences customer routing, time spent in the store, and queue selection decisions. Therefore, network and spatial models are employed to investigate customer interactions within the store's physical space. These aspects are typically overlooked in conventional queuing models.

The model must account for the interaction of multiple agent types – not only customers but also cashiers and automated systems that adapt their operating modes based on current load conditions. Agent-based modelling enables the dynamic opening or closing of checkout counters, adjustment of service rules, and forecasting of the consequences of managerial decisions – capabilities that are typically beyond the scope of traditional modelling approaches.

Another defining feature is non-stationarity. In supermarkets, customer flow varies throughout the day, week, or season. To model such dynamics accurately, advanced stochastic processes are employed, particularly non-homogeneous Poisson processes (NHPP), which capture peak hours, wave-like activity, and the influence of external events. Traditional models often rely on the assumption of constant arrival intensity.

It is also essential to account for different types of checkout counters, such as standard lanes, express checkouts, and self-service stations. Each type has distinct service parameters, transaction times, and varying levels of attractiveness for different customer segments. The distribution of customers among these checkout types significantly affects overall flow, workload balance, and queuing efficiency.

In general, modelling queues in supermarkets requires the integration of stochastic analysis, agent-based approaches, spatial modelling, and consumer behaviour insights to construct an adequate and effective queue management model.

Thus, customer service modelling in supermarkets constitutes a complex task that combines various methodologies, including mathematical computation, computer-based simulation, and intelligent algorithms.

2. Problem statement

The aim of this study is to develop a simulation model for the operation of a dynamic queue management system in supermarkets. This model is intended to evaluate the effectiveness of various queue structures and dynamic checkout management strategies, with the goal of improving customer service quality and optimizing staff performance.

3. Queue modelling software

Queue modelling software can be conditionally classified into three categories: queue management systems, simulation modelling software, and analytical tools.

Queue Management Systems (Commercial Solutions). The primary function of this type of software is to optimize visitor flow, reduce waiting time, and improve the overall quality of service. Among the most popular systems are: WaitWell, Qminder, Wavetec. These platforms are widely used in customer-facing environments such as retail, healthcare, and government

services to enhance operational efficiency and customer satisfaction.

Simulation Modelling Software. Simulation modelling software is used to create accurate digital replicas of queue management systems. This enables detailed analysis of system performance and facilitates forecasting behaviour under various operating conditions. These tools are particularly valuable for testing different queue configurations, customer flow patterns, and service policies without disrupting real operations. Among the most widely used simulation platforms are: Arena, AnyLogic, FlexSim. These tools are commonly employed in operations research, logistics, retail, healthcare, and service industries to support decision-making and optimize system efficiency.

Analytical Tools. Analytical tools are employed for mathematical modelling and analysis of queue management systems. They enable the implementation of theoretical queuing models, calculation of key performance indicators, and investigation of the impact of various parameters on system behaviour. These tools are particularly useful for validating simulation results, conducting sensitivity analyses, and exploring system behaviour under idealized assumptions. The most widely used analytical tools include: Python, MATLAB, SimPy, Mesa, AgentPy, Queue, Salabim. These tools provide researchers and developers with flexible platforms for modelling, experimentation, and performance evaluation of queuing systems in various domains.

Among the existing approaches, Python stands out as the most effective tool due to its flexibility, a broad range of specialized libraries (such as SimPy, Mesa, AgentPy, Queueing-tool, and Salabim), and its ability to seamlessly integrate simulation modelling with analytics and data visualization. This combination ensures high modelling accuracy, scalability, and adaptability, which are critical for the comprehensive analysis and optimization of queue management systems in supermarkets.

Therefore, the use of Python with appropriate libraries represents an optimal solution for scientific research, practical simulation, and the implementation of queue management systems in retail environments. It allows for the integration of theoretical computations with real-world operational tasks and enables efficient control of customer flow.

4. Main results

In the context of increasing market competition and heightened consumer expectations for prompt service, the problem of optimizing the operation of queueing systems has gained particular importance. A typical example of such systems is supermarkets with checkout areas, where it is essential to efficiently organize customer flows and queues. This helps to reduce the average waiting time and ensures a balanced workload for the staff.

The core component of these systems is queueing models, which determine the customer service logic and the distribution of workload among checkout counters. The scientific literature identifies the following key types of queue organization.

Multiple Queue Multiple Server Systems (MQMS) – in these systems, each checkout has its own individual queue, and customers independently choose which line to join. This often results in uneven load distribution and increased waiting times due to inefficient decision-making.

Fixed Queue First-In-First-Out Systems (FIFO) – each checkout counter is assigned a dedicated, ordered queue. This ensures predictability, but can lead to delays if service at one of the checkouts is slower than at others.

Single Queue Multiple Server Systems (SQMS) – customers form a single unified queue and are directed to the first available cashier. This model minimizes waiting time disparities and generally improves fairness in customer service.

To evaluate the advantages and disadvantages of the above queueing strategies, a simulation model was developed that enables a comparative analysis of their efficiency under typical supermarket operating conditions.

4.1. Discrete-event simulation model of a checkout node with different queue structures

The model was implemented using the SimPy library in the Python programming language, enabling an accurate representation of queue dynamics and resource utilization. The primary focus is on comparing different configurations in terms of average customer waiting time, cashier workload, and queue accumulation levels, with the aim of identifying the most efficient service organization structure.

The following types of systems were considered:

- 1) Type 1 – N independent queues for N checkout counters, with customers selecting the least loaded queue;
- 2) Type 2 – N independent queues with random queue selection by the customer;
- 3) Type 3 – a single centralized queue serving N checkout counters.

The simulation model is configured with fixed parameters. The number of checkout counters (NUM_CASHIERS) is set to 7 for all queue system types. Each simulation run lasts 60 minutes (SIM_TIME). Customer arrivals are modelled using an exponential distribution with a variable mean inter-arrival time ranging from 0.5 to 1.1 minutes (ARRIVAL_RATE). The service time is modelled using a uniform distribution within the range of 2 to 5 minutes (SERVICE_TIME). Each queue configuration undergoes 100 independent simulation runs (NUM_RUNS) to ensure statistical significance. Simulation outcomes are evaluated using three key performance metrics:

- the average customer waiting time in the queue,
- the average utilization rate of checkout counters,
- the average queue length, weighted by the time customers spend in the system.

Each individual simulation run consists of five sequential stages.

Stage One – Environment Initialization. A simulation environment (simpy.Environment) is created, along with a corresponding service system object based on the selected queue configuration. A list is initialized to record customer waiting times.

Stage Two – Customer Flow Generation. Customers arrive into the system at intervals generated according to an exponential distribution. Each arriving customer initiates a separate process that simulates their presence in the system.

Stage Three – Customer Handling (handle_customer). Each customer logs their arrival time and, depending on the type of queue system:

- Type 1 (Shortest Queue): selects the checkout with the shortest queue;
- Type 2 (Random Queue): randomly selects one of the available checkouts;
- Type 3 (Single Queue): joins the centralized queue and waits for the first available cashier.

After selecting a resource, the customer waits for access, records the service start time, calculates the waiting time, and undergoes service for a random interval within the defined SERVICE_TIME range. Upon completion, the customer releases the checkout.

Environment Execution. The simulation runs until the predefined simulation time is reached (env.run(until=SIM_TIME)), at which point all active processes are completed.

Metrics Collection. The simulation computes the average customer waiting time, the average utilization of the checkout counters, and the average queue length, adjusted for time-weighted presence in the system. These values are stored for subsequent aggregation and analysis.

The multi-run simulation algorithm includes statistical processing of results for each of the three types of service systems. For each system type, a series of 100 independent simulation runs

is executed. During each run, key performance indicators are recorded: the average customer waiting time, the average cashier utilization rate, and the average queue length. These metrics provide an objective basis for evaluating the system's performance and are stored in global lists (results, utilizations, queue_lengths). Additionally, all individual waiting time values are accumulated into a separate sample (wait_distributions) for further analysis of the empirical distribution.

After completing the entire series of simulation runs, the accumulated data undergo statistical processing. The mean values of each performance metric are calculated along with the standard error of the mean (SEM), allowing for an assessment of the reliability and significance of the obtained

results. To enhance interpretability, a range of visualizations is employed:

- Kernel Density Estimation (KDE) curves of waiting time distributions are generated to reveal characteristic distribution features.
- Bar charts illustrate the average values with error bars for each queueing system type.
- Box plots depict the variability across simulation runs, highlighting the range of fluctuations and potential outliers.

The visualization of the waiting time density distribution is presented in Figure 1.

The visualization of the calculated metrics across different queue types in supermarkets is shown in Figure 2.

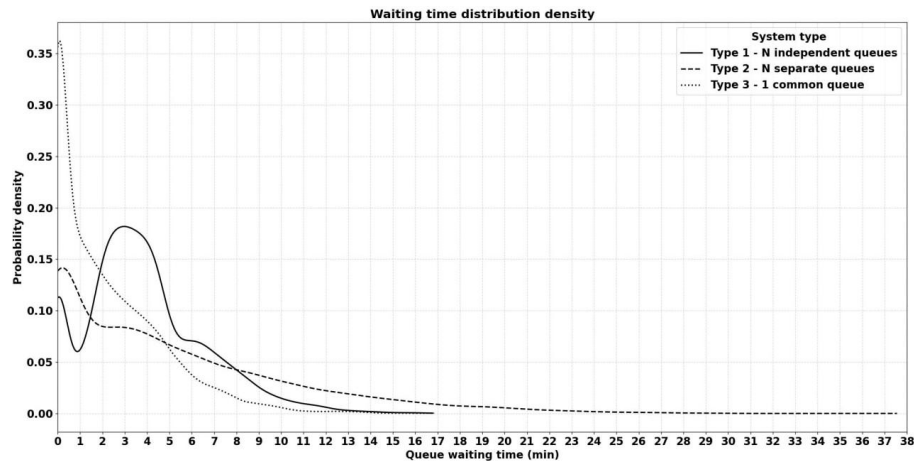


Fig. 1. Waiting time distribution density

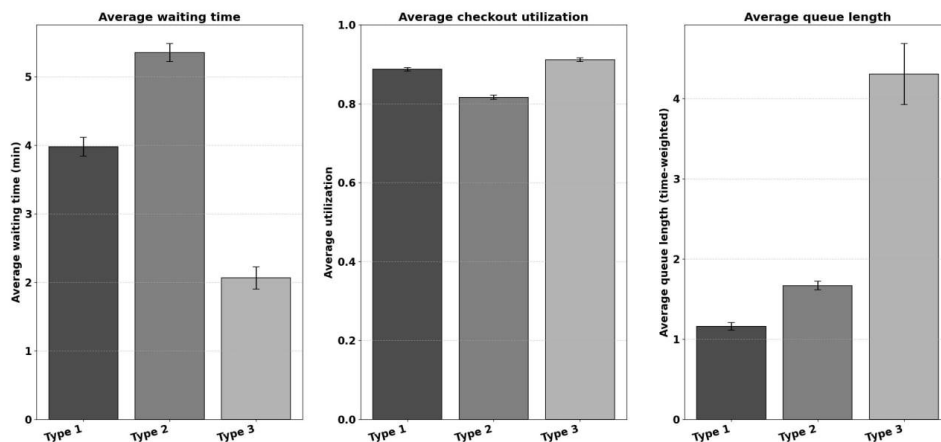


Fig. 2. Average waiting time, checkout utilization, and queue length

Let's analyse the obtained average values.

1. Average Waiting Time.

The most efficient system is Type 3, where the average waiting time is approximately 2 minutes. This result highlights the advantage of a centralized queue, which facilitates faster service due to optimal client distribution among available cash registers.

Type 1 shows a moderate performance (~3.6 minutes). Although customers choose the shortest queue, this strategy does not always ensure minimal waiting time due to variability in service times and load imbalance across cashiers.

The least effective configuration is Type 2 (~5.3 minutes), where customers randomly select queues. This leads to inefficient distribution and unbalanced workloads at cash registers, increasing overall waiting times.

2. Average Cashier Utilization.

Type 3 again demonstrates the best performance (~91%), confirming the high efficiency of the single-queue system. This configuration minimizes idle time and ensures near-full utilization of cashier resources.

In Type 1, the cashier utilization is approximately 88%, indicating partial inefficiency due to the greedy queue selection strategy by customers, which may not always balance the load effectively.

Type 2 shows the lowest utilization level (~82%), as the random distribution of customers leads to uneven workloads and results in idle time at some checkout counters.

3. Average Queue Length.

Although Type 3 has the highest average queue length (~4.4 customers), this is offset by a high service rate. The single queue ensures a steady and balanced flow of customers to all available cashiers.

In Types 1 and 2, the average queue lengths are shorter (~1 and ~1.6 customers, respectively). However, this does not necessarily indicate greater efficiency. The shorter queues in these systems result from fragmentation and suboptimal customer distribution, which leads to an overall increase in waiting time.

The distributions of average performance indicators are presented in Figure 3.

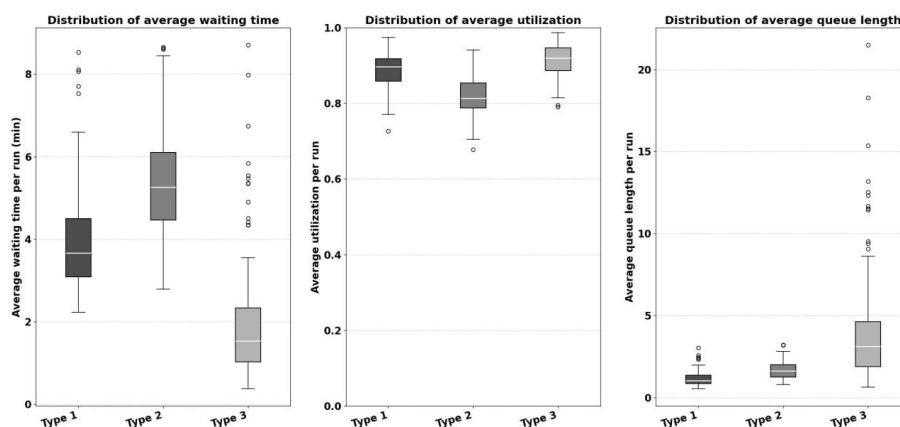


Fig. 3. Distributions of average waiting time, resource utilization, and queue length

Let's analyse the results of the distribution of average values.

1. Distribution of Average Waiting Time.

The study revealed significant differences among the three queue organization types.

Type 3 (Single Unified Queue) demonstrates the lowest median waiting time (~2 minutes) and the smallest variance, indicating a highly efficient and stable system. The absence of prolonged delays further supports the advantages of centralized customer routing, where each arriving client is directed to the first available service point.

In contrast, Type 2 (Independent Queues with Random Assignment) showed the least favourable performance, with a median waiting time of approximately 5 minutes and a broad distribution, reflecting the unpredictability of service times. This outcome is attributed to uneven checkout utilization caused by random customer assignment, which often leads to overloading certain queues while others remain underused.

Type 1 (Independent Queues with Customer Selection) occupies an intermediate position. While its median waiting time is lower than that of Type 2, the greater dispersion suggests higher variability and reduced predictability in service performance.

Conclusion: The simulation results indicate that the single queue system (Type 3) is optimal for minimizing customer waiting time, offering consistent service levels. Conversely, segmented queue systems (Type 2) are least efficient, primarily due to random and unbalanced customer distribution across service points.

2. Distribution of Average Checkout Utilization.

The analysis of cashier utilization confirms the advantages of centralized queue management.

Type 3 (Single Unified Queue) exhibits the highest and most stable level of resource utilization with a median value of approximately 0.95, indicating near-full exploitation of available service capacity.

Type 1 (Independent Queues with Customer Choice) also demonstrates high efficiency, though with slightly lower utilization (median ~0.90). This outcome is attributed to partial imbalance resulting from customers independently selecting queues, which may lead to uneven workload distribution.

Type 2 (Independent Queues with Random Assignment) shows the lowest median utilization (~0.85) and greater variability, suggesting inefficient cashier usage caused by the stochastic and uncoordinated distribution of customer arrivals.

Conclusion: The single queue configuration (Type 3) ensures optimal resource utilization, whereas segmented queue systems (Type 2) lead to idle resources and operational inefficiencies due to poor load balancing.

3. Distribution of Average Queue Length.

Queue length does not necessarily correlate with service speed.

Type 3 (Single Unified Queue) exhibits the longest queues (median ~3 customers, with a maximum exceeding 5),

which is a natural consequence of centralized queuing. However, this is effectively offset by the high service throughput, resulting in the shortest average waiting time among all configurations.

Type 1 (Independent Queues with Customer Selection) is characterized by shorter queues (median < 1), but demonstrates greater variability due to the uneven distribution of customers across service points.

Type 2 (Independent Queues with Random Assignment) shows a similar average queue length (~1 customer); however, due to inefficient service coordination, it results in the highest overall waiting time.

Conclusion: The longer queues observed in Type 3 are not a disadvantage, as they are compensated by rapid processing and balanced resource utilization. Conversely, shorter queues in Types 1 and 2 do not guarantee efficiency, as they often correspond with idle cashier time and poor system performance.

The analysis of average result distributions demonstrates that the single-queue system (Type 3) is the most effective for minimizing customer wait times, while distributed queues (Type 2) are the least efficient. Type 3 also ensures optimal resource utilization, whereas Type 2 leads to idle time and inefficiencies. While Type 3 exhibits longer queues, they are mitigated by faster processing. Conversely, the shorter queues in Types 1 and 2 do not guarantee efficiency due to more frequent cashier idle periods.

4.2. Supermarket model with dynamic queue management

The algorithm of the model represents a service system with a well-defined structure, incorporating client categorization, queue organization, cashier management, and continuous monitoring of system dynamics.

Customers are divided into three groups: VIPs, regular customers, and elderly individuals. Each group is governed by specific rules – such as priority level, service time, and maximum acceptable waiting time.

The number of operating cashiers dynamically adapts to the current load. When customer flow increases, additional checkout counters are activated; when demand decreases, some cashiers are closed. Customers are automatically redistributed to the newly available checkouts to maintain efficiency.

The system includes a monitoring component that continuously tracks queue lengths, collects data on each customer type, and evaluates the performance of different types of checkouts – standard and self-service.

A specialized data structure is used to manage the queues efficiently. It ensures that customers are processed according to predefined rules: priority first, followed by arrival time, and then queue position.

Customer arrivals are randomly generated, taking into account a specified frequency. The customer type is also determined randomly.

The service process accounts for slight variability in transaction time. If a customer waits longer than their patience threshold, they may abandon the queue. Each customer type has a defined maximum waiting time, after which they exit the system if not served.

The algorithm is based on queueing theory, which models the interaction between customers and the service system, encompassing arrival dynamics, waiting behaviour, and service operations.

The algorithm's structure incorporates priority-based scheduling and adaptive resource management principles, enabling the system to respond dynamically to varying customer types and workload conditions. This ensures that service quality and efficiency are maintained under different operational scenarios.

The algorithm is implemented using discrete-event simulation (DES) methods, where changes in the system occur at discrete points in time triggered by specific events.

The formalization of the algorithm is as follows.

Let the set of currently open checkouts at time t be given. At each moment t , the average waiting time of all customers presently standing in queues is computed as:

$$\bar{W}(t) = \frac{1}{|Q(t)|} \sum_{i \in Q(t)} w_i(t), \text{ if } |Q(t)| > 0$$

$$\bar{W}(t) = 0, \text{ if } |Q(t)| = 0 \quad (1)$$

where $Q(t)$ is the set of customers waiting in queues at time t , and $w_i(t)$ is the current waiting time of customer i at moment t .

A new temporary checkout is opened if the average waiting time exceeds the opening threshold and the number of active checkouts has not yet reached the maximum limit:

$$\text{If } \bar{W}(t) > \alpha \text{ and } N(t) < N_{\max}, \text{ then } N(t+1) = N(t) + 1 \quad (2)$$

where $N(t)$ is the number of open checkouts at time t , $N_{\max}$ is the maximum allowable number of checkouts, and α is the opening threshold.

A temporary checkout is closed only when all three conditions are simultaneously satisfied:

$$\text{If } \bar{W}(t) < \beta \text{ and } \text{queue}(k) = 0 \text{ and } \text{idle}(k) \geq \delta, \\ \text{then } N(t+1) = N(t) - 1 \quad (3)$$

where k is the temporary checkout under consideration for closure, $\text{idle}(k)$ is the number of time steps during which checkout k has remained idle and has not served any customer, β is the closure threshold for the temporary checkout, and δ is the minimum required number of idle steps before a checkout becomes eligible for closure (idle_close_steps).

Among all checkouts that satisfy these conditions, the one with the longest idle duration is selected for closure:

$$k^* = \arg \max_k \text{idle}(k) \quad (4)$$

When a temporary checkout is closed, all customers in its queue are allowed to independently choose a new checkout. Each customer selects the checkout with the smallest current queue length:

$$c^* = \arg \min_{c \in C_{\text{open}}} |\text{queue}(c)| \quad (5)$$

where C_{open} is the set of all open checkouts at the current moment.

Each checkout queue is modelled as a priority queue. Every customer i is characterized by a priority triplet:

$$e_i = (p_i, t_i^{\text{arr}}, s_i) \quad (6)$$

where p_i is the priority level associated with the customer type, t_i^{arr} is the arrival time, and s_i is a unique sequential identifier.

Customers are served in ascending order of this triplet, which ensures both inter-type prioritization and fairness within each type (FIFO ordering for equal priorities).

Customer priorities are defined as follows:

$$p_i = \begin{cases} 0, \text{ VIP} \\ 1, \text{ Normal} \\ 2, \text{ Elderly} \end{cases} \quad (7)$$

A customer i abandons the queue if their waiting time exceeds their individual patience threshold pat_i :

$$w_i(t) > \text{pat}_i \quad (8)$$

where the patience value pat_i is generated at the moment of customer creation as:

$$\text{pat}_i = \text{clip}(\text{base_serv_time}_i \cdot \text{patience_fact}_i \cdot 3, 10, 60) \quad (9)$$

The total simulation time spans 480 minutes, corresponding to a full eight-hour workday. Initially, the system operates with three standard checkouts and two self-service kiosks. The system is capable of utilizing up to ten checkouts concurrently. Customer arrival intensity is determined by the parameter λ_{param} , which is calculated based on the average inter-arrival time. The average queue waiting time serves as the trigger for activating or deactivating temporary checkouts: a threshold of 8 minutes prompts the opening of additional checkouts, whereas a 3-minute threshold signals closure. A temporary checkout is also considered for deactivation if it remains idle for more than 15 minutes. Queue length data is collected every 2 minutes, providing detailed insight into system dynamics for further performance analysis and optimization.

Each customer is assigned a type: VIP, NORMAL, or ELDERLY. This attribute determines their service priority, where VIP customers have the highest priority, NORMAL customers receive standard priority, and ELDERLY customers are assigned a lower priority, reflecting their specific service needs. The average service time is also dependent on customer type: 3 minutes for VIPs, 5 minutes for NORMAL customers, and 7 minutes for ELDERLY customers. Each customer is also associated with a patience parameter, which defines the maximum tolerable waiting time before the customer abandons the queue. The arrival time of each subsequent customer is determined stochastically using an exponential distribution with the parameter λ_{param} .

The results of the simulation, focusing on queue dynamics and checkout resource management, are visualized through graphs presented in Figure 4.

The data presented in the upper chart illustrate the variation in the average customer waiting time in queues over the course of 500 simulation steps. The waiting time values fluctuate within a range of 0.00 to 10.00 minutes, with an average of 1.95 minutes recorded over the simulation period. The minimum value was observed at minute 0, while the maximum peak occurred at minute 298.

The threshold for opening a new checkout is set at 8 minutes and was exceeded 4 times during the simulation. The threshold for closing a checkout is set at 3 minutes, with 346 instances falling below this threshold. A total of 480 measurements were recorded. This distribution indicates a generally low system load, punctuated by occasional peak demand periods. The high frequency of values below the closure threshold suggests a potential for optimizing the number of active checkouts.

The lower chart depicts changes in the number of active checkouts throughout the simulation period. The minimum number of open checkouts was 5, while the maximum reached 8. The average number of active checkouts was 6.10. Initially, the system started with 5 checkouts, ending the simulation with 6.

The system remained stable for the majority of the simulation: out of 480 time steps, only 9 instances involved changes in the number of active checkouts (5 openings and 4 closures), corresponding to a stability rate of 98.1%. The transition in the number of active checkouts occurred in discrete steps and exhibited a noticeable delay in response. Periods with the maximum number of active checkouts generally coincided with moments when the waiting time exceeded the 8-minute threshold.

The number of customers in queues fluctuates within a range of 0 to 8 individuals, with a mean value of 3.26 and a median of 3.0. In 7.3% of simulation steps, the number of customers exceeds 6, indicating the presence of isolated peak loads. The customer flow dynamics are characterized by rapid fluctuations over short time intervals, with high variability observed in 20 segments, each spanning 10 simulation steps.

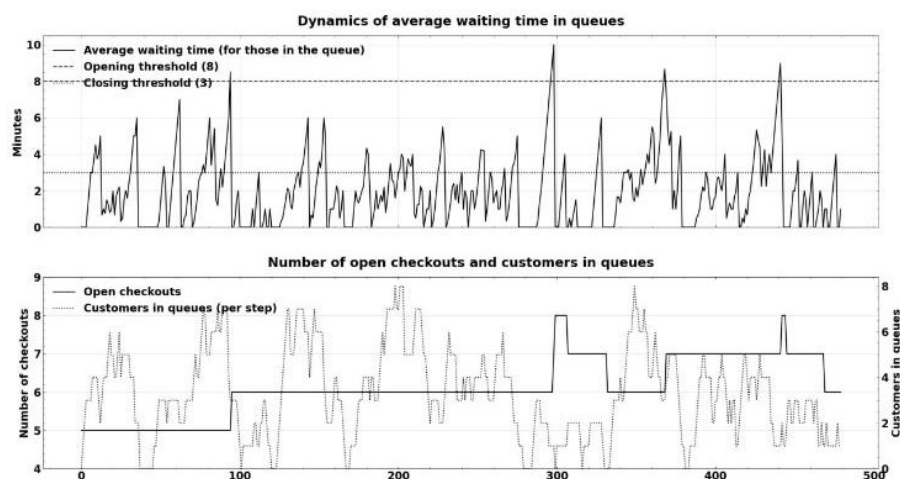


Fig. 4. Queue dynamics and checkout resource management in the supermarket simulation model

These abrupt spikes have a direct impact on average waiting time: queue surges generally coincide with peaks in waiting durations. Analysis of system behaviour reveals a degree of inertia – during periods of increasing queue lengths (e.g., between steps 80 and 100), the number of active checkouts remains unchanged. Adjustments in checkout availability occur with a delay, typically after critical thresholds are reached. Similarly, following queue reductions, the number of active checkouts does not decrease immediately, highlighting sluggish response dynamics.

Overall, the system exhibits basic adaptability to fluctuating loads. However, there exists a risk zone: waiting time peaks exceeding 8 minutes are not mitigated rapidly enough. The system tends to operate near the thresholds for opening and closing checkouts, which poses a risk of underperformance during sudden increases in customer influx.

The results of the simulation, including queue dynamics captured by video monitoring, checkout utilization, and accumulated customer flow, are illustrated in Figure 5.

The first graph illustrates the total queue length as recorded by the video surveillance system at two-minute intervals. The data reveal an uneven pattern, characterized by several distinct peaks. The first peak occurs approximately between the 40th and 80th minute, reaching a maximum value of 8 customers. A second wave of increased queue length is observed between 160 and 240 minutes, again peaking at 8 customers. A third increase is registered after the 280th minute, though it does not exceed the previous maxima. Between these peaks, the queue length drops to levels between 0 and 2 customers.

The average queue length is 3.25 customers, with a median value of 3.0. The minimum recorded length is 0, and the maximum is 8. In 7.5% of cases, the queue length exceeded 6 customers, indicating occasional system overloads. The overall trend, based on the initial and final points of the time series, suggests a slight upward shift: the queue length gradually increases from 0 to around 2 customers by the end of the simulation. This indicates a tendency toward load accumulation in the later stages, despite efficient service performance during other periods.

The second graph displays the utilization levels of conventional checkouts and self-service checkouts. The conventional checkouts exhibit a minimum load of 0.00%, a maximum of 81.82%, and an average utilization of 59.75%. During the first 80 minutes, their utilization increases sharply from 0% to over 80%, followed by a decline to approximately 60%, and then a gradual rise to 75%. Around the 300th minute, a drop in occupancy to 50-55% is observed, after which the utilization stabilizes.

The self-service checkouts show a minimum utilization of 0.00%, a maximum of 79.93%, and a higher average

of 71.48%. These checkouts maintain a more consistent and stable level of usage, without sharp fluctuations, and sustain utilization between 82% and 85% for the majority of the simulation period. This indicates a more sustained demand for self-service options compared to conventional checkouts.

The third graph presents the cumulative customer flow throughout the simulation. The service curve rises linearly with minor fluctuations, reaching approximately 148 customers by the end of the simulation. A parallel line, representing the number of customers who left the system, remains at zero, indicating that no customers abandoned the queue without being served. This result highlights the system's resilience under load.

Cumulative performance indicators reveal that the highest demand occurred during the first quarter of the simulation. During this period, system utilization peaked, and queue lengths increased rapidly. As the system stabilized, the queues began to decline, indicating a successful adaptation to the initial surge in customer arrivals. Subsequent waves of load were less pronounced, suggesting the system maintained control over variations in demand.

Overall, the model demonstrates effective operational performance – the utilization of self-service checkouts remains consistently high, no customer losses were recorded, and the system responds to fluctuations efficiently, as evidenced by the gradual reduction of queue lengths in the second half of the simulation.

The dynamics of the average waiting time at conventional checkouts are illustrated in Figure 6.

The average waiting time at conventional checkouts varies significantly depending on the customer type. Priority customers (VIP) experience the shortest average waiting time – 2.89 minutes, with a maximum wait of no more than 4 minutes. A minimum value of 0 minutes indicates instances of immediate service for this group.

Regular customers wait longer, with an average waiting time of 4.16 minutes, which is approximately 44% longer than that of VIP customers. The maximum wait time for this category reaches 6 minutes, highlighting a more pronounced level of delay.

Elderly customers face the longest average waiting time – 5.28 minutes, with a maximum recorded wait of 7.5 minutes. This suggests that this group consistently experiences the longest delays, potentially due to the specific nature of their service requirements or increased pressure on cashier resources.

The dynamics of average waiting time at self-service checkouts are presented in Figure 7.

For priority customers, the average waiting time at self-service checkouts is 2.97 minutes, with a maximum of 3.19 minutes and a minimum of 0 minutes. This indicates swift and preferential service in self-service settings for this group.

Regular customers experience an average waiting time of 3.24 minutes, which is significantly lower than at conventional

checkouts. Their maximum waiting time reaches 4.33 minutes, while the minimum remains at 0 minutes, reflecting the improved efficiency of self-service for the majority of customers.

In contrast, elderly customers face a substantially longer average waiting time of 7.58 minutes, with a maximum of 10 minutes. Although instances of immediate service (0 minutes) do occur, this group generally experiences

considerable delays, likely due to difficulties interacting with self-service technology or other accessibility-related factors.

These findings suggest that self-service checkouts effectively reduce waiting times for priority and regular customers, while elderly patrons encounter significant challenges, highlighting the need for further optimization and targeted support to improve their service experience.

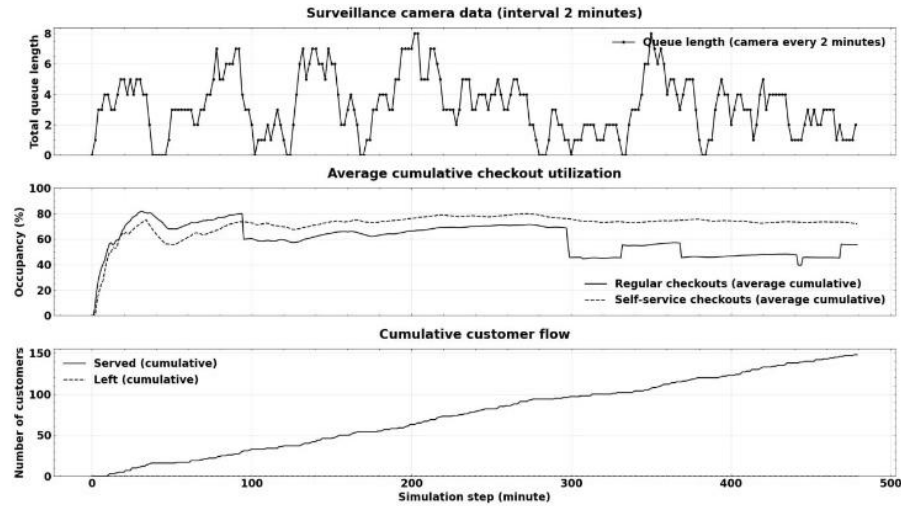


Fig. 5. Queue trends from video surveillance, checkout utilization, and cumulative customer flow during the simulation

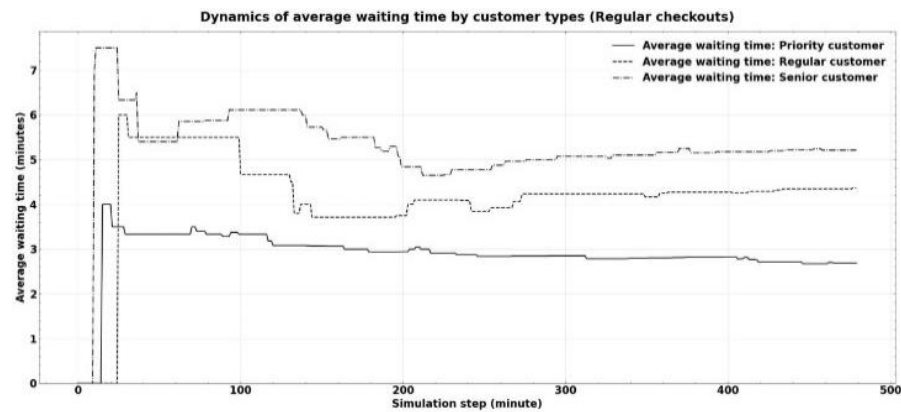


Fig. 6. Average waiting time trends at conventional checkouts

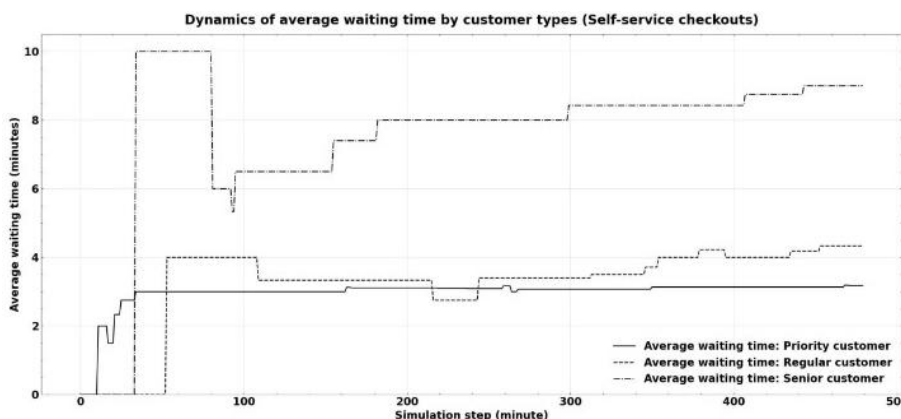


Fig. 7. Average waiting time trends at self-service checkouts

5. Conclusion

The study explores the key aspects of customer service optimization in retail environments. An algorithm was developed and implemented as a discrete-event simulation model of a supermarket checkout system, allowing for comparative analysis of various queue configurations based on defined performance metrics. The results reveal that employing a single shared queue (Type 3) is the most effective strategy for self-

service checkout systems, offering the lowest average waiting time (2.03 minutes) and nearly full resource utilization (over 90%). In contrast, Type 1 (customer-selected checkout) demonstrated a moderate average waiting time (~3.9 minutes), while Type 2 (random selection) proved to be the least efficient.

Overall, the simulation results support the adoption of centralized queuing (Type 3) in environments where operational efficiency, system stability, and automation are critical – particularly in self-service systems. Meanwhile, distributed queuing

models (Types 1 and 2) may be more appropriate for personnel-assisted settings, where flexibility, visual perception of queue lengths, and customer comfort play a more significant role.

A dynamic queue management model was developed and implemented, aimed at optimizing customer waiting times and cashier workload by adapting the number of operational checkout counters to the current intensity of customer flow and queue length. The algorithm is based on discrete-event simulation, where each event (customer arrival, service, cashier status change) is processed individually. The model accounts for different customer types (VIP, regular, and elderly), their service times, patience thresholds, and queue priorities.

The use of priority queues ensures fair allocation of customers across checkouts based on their status. Customers automatically choose the least loaded checkout, while the system continuously analyses the average waiting time to decide when to open or close additional checkouts. Simulated monitoring cameras periodically collect data on queue lengths, enabling the algorithm to dynamically adapt to real-time fluctuations.

Overall, the proposed dynamic queue management algorithm effectively adjusts to variations in customer flow and prevents service loss by automatically regulating the number of active checkouts. It demonstrates operational flexibility across varying load scenarios and ensures consistent service quality with minimal resource expenditure.

Simulation results indicate that the average customer waiting time is 1.95 minutes, with the system programmed to trigger the opening of additional checkouts if the waiting time exceeds the threshold of 8 minutes. The system's response to load fluctuations exhibits a stepwise and inertial behaviour, which occasionally leads to delays in mitigating peak load imbalances. Nevertheless, system stability is maintained in 98.1% of time steps, as measured by the number of active checkout counters.

The average utilization rate of conventional checkouts is 59.75%, characterized by significant fluctuations, whereas self-service checkouts operate more consistently, achieving a higher average utilization of 71.48%. The absence of customer abandonment events throughout the simulation indicates the robustness of the system.

Waiting times vary across customer categories and checkout types. At conventional checkouts, priority customers experience an average waiting time of 2.89 minutes, whereas elderly customers wait approximately 5.28 minutes. Self-service terminals further reduce the waiting time for priority customers to 2.97 minutes and for regular customers to 3.24 minutes. However, elderly customers experience a significant increase in waiting time – up to 7.58 minutes – at self-service checkouts, indicating usability challenges that require further attention and system adjustment.

The developed model enables evaluation of key performance indicators, including: Average customer waiting time, Queue length dynamics, Checkout utilization rate, Customer abandonment rates, Total number of serviced customers.

The developed model is intended to examine the feasibility of applying such approaches to optimize customer service in supermarkets.

The model is designed to simulate a single working day of a supermarket with a duration of 8 hours. Extending the simulation to a longer period without substantial modifications to the underlying logic would lead to statistically unreliable results for several reasons. Such an analysis must account for time-dependent customer arrival rates, staff shift changes, variability in customer patience, and other factors. Therefore, this type of extended modelling can be considered as a subject for future research.

These findings can inform the optimal configuration of checkout zones in supermarkets, support the development of personnel management strategies, and serve as a predictive tool for assessing service quality and reducing operational costs in the retail sector.

References

- [1] Antczak, T., Weron, R., & Zabawa, J. (2020). Data-Driven Simulation Modeling of the Checkout Process in Supermarkets: Insights for Decision Support in Retail Operations. *IEEE Access*, 8, 228841–228852. <https://doi.org/10.1109/ACCESS.2020.3045919>
- [2] Bonabeau, E. (2002). Agent-based modeling: Methods and techniques for simulating human systems. *Proceedings of the National Academy of Sciences*, 99(suppl_3), 7280–7287. <https://doi.org/10.1073/pnas.082080899>
- [3] Borshchev, A. (2013). The big book of simulation modeling: Multimethod modeling with AnyLogic 6. AnyLogic North America.
- [4] Cayirli, T., & Veral, E. (2003). Outpatient Scheduling in Health Care: A Review of Literature. *Production and Operations Management*, 12(4), 519–549. <https://doi.org/10.1111/j.1937-5956.2003.tb00218.x>
- [5] Gross, D., Shortle, J. F., Thompson, J. M., & Harris, C. M. (2008). *Fundamentals of Queueing Theory* (4th edn). Wiley. <https://doi.org/10.1002/9781118625651>
- [6] Hillier, F. S., & Lieberman, G. J. (2014). *Introduction to operations research* (10th ed.). McGraw-Hill Education.
- [7] Holland, J. H. (1992). Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence. The MIT Press. <https://doi.org/10.7551/mitpress/1090.001.0001>
- [8] Kelton, W. D., Sadowski, R. P., & Zupick, N. B. (2014). *Simulation with Arena* (6th Edition). McGraw-Hill Education.
- [9] Laguna, M., & Marklund, J. (2018). *Business Process Modeling, Simulation and Design* (3rd edn). Chapman and Hall/CRC. <https://doi.org/10.1201/9781315162119>
- [10] Meuter, M. L., Ostrom, A. L., Roundtree, R. I., & Bitner, M. J. (2000). Self-Service Technologies: Understanding Customer Satisfaction with Technology-Based Service Encounters. *Journal of Marketing*, 64(3), 50–64. <https://doi.org/10.1509/jmkg.64.3.50.18024>
- [11] Newell, G. F. (1982). *Applications of Queueing Theory*. Springer Netherlands. <https://doi.org/10.1007/978-94-009-5970-5>
- [12] Raeis, M., Tizghadam, A., & Leon-Garcia, A. (2021). Queue-Learning: A Reinforcement Learning Approach for Providing Quality of Service (Version 1). arXiv. <https://doi.org/10.48550/ARXIV.2101.04627>
- [13] Ross, S. M. (2014). *Introduction to probability models* (Eleventh edition). Academic Press is an imprint of Elsevier.
- [14] Schranz, M., Di Caro, G. A., Schmickl, T., Elmenreich, W., Arvin, F., Şekercioğlu, A., & Sende, M. (2021). Swarm Intelligence and cyber-physical systems: Concepts, challenges and future trends. *Swarm and Evolutionary Computation*, 60, 100762. <https://doi.org/10.1016/j.swevo.2020.100762>
- [15] Sztrik, J. (2016). *Basic queueing theory*. GlobeEdit.
- [16] Toopchinezhad, M. P., & Ahmadi, M. (2024). *Machine Learning Approaches for Active Queue Management: A Survey, Taxonomy, and Future Directions* (Version 1). arXiv. <https://doi.org/10.48550/ARXIV.2410.02563>
- [17] Wang, Y. (2023). Research on the Queueing Theory based on M/M/1 Queueing Model. *Highlights in Science, Engineering and Technology*, 61, 80–87. <https://doi.org/10.54097/hset.v61i.10276>
- [18] Welch, G., & Bishop, G. (2006). *An introduction to the Kalman filter* (TR 95-041). University of North Carolina at Chapel Hill.

Ph.D. Larysa Gumeniuk
e-mail: l.gumeniuk@lntu.edu.ua

Ph.D. (technical). Lutsk National Technical University, Department of Automation and Computer-Integrated Technologies.
Research interests: modelling of reliability and safety of the automated control systems.

<https://orcid.org/0000-0002-7678-7060>

Ph.D. Volodymyr Lotysh
e-mail: auvp@lntu.edu.ua

Ph.D. (technical). Lutsk National Technical University.
Research interests: open-source software applied for simulations of problems using distributed platforms.

<https://orcid.org/0000-0003-0899-8015>

Ph.D. Pavlo Humeniuk
e-mail: p.gumeniuk@lntu.edu.ua

Ph.D. (technical). Lutsk National Technical University, Department of Automation and Computer-Integrated Technologies.
Research interests: programming, robotics.

<https://orcid.org/0000-0002-6251-8548>

