

Article

An Asymmetric Polling-Based Optimization Model in a Dynamic Order Picking System

Dan Yang ¹, Sen Liu ²  and Zhe Zhang ^{2,*}¹ College of Innovative Business and Accountancy, Dhurakij Pundit University, Bangkok 10210, Thailand² School of Logistics and Management Engineering, Yunnan University of Finance and Economics, Kunming 650221, China

* Correspondence: zhangzhe0511@njst.edu.cn or zhangzheyufe@163.com

Abstract: The timeliness of order deliveries seriously impacts customers' evaluation of logistics services and, hence, has increasingly received attention. Due to the diverse and large quantities of orders under the background of electronic commerce, how to pick orders efficiently while also adapting these features has become one of the most challenging problems for distribution centers. However, previous studies have not accounted for the differences in the stochastic characteristics of order generation, which may lead to asymmetric optimization problems. With this in mind, a new asymmetric polling-based model that assumes the varying stochastic characteristics to analyze such order picking systems is put forward. In addition, two important indicators of the system, mean queue length (MQL) and mean waiting time (MWT), are derived by using probability-generating functions and the embedded Markov chain. Then, simulation experiments and a comparison of the numerical and theoretical results are conducted, showing the usefulness and practicalities of the proposed model. Finally, the paper discusses the characteristics of the novel order picking system and the influence of the MQL and MWT on it.

Keywords: asymmetric polling theory; order picking system; gated service discipline; efficiency; iterative algorithm



Citation: Yang, D.; Liu, S.; Zhang, Z. An Asymmetric Polling-Based Optimization Model in a Dynamic Order Picking System. *Symmetry* **2022**, *14*, 2283. <https://doi.org/10.3390/sym14112283>

Academic Editor:
Alexander Zaslavski

Received: 28 September 2022

Accepted: 28 October 2022

Published: 31 October 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Currently, as the development of electronic commerce drives the growth of consumer demand, reducing the order-to-delivery time has become the primary challenge for distribution centers under the conditions of smaller order sizes and more varieties of orders [1]. Due to the length of time and financial costs, order picking is considered an important activity of storage and distribution centers. Some studies have shown that the cost of order picking can account for more than 60% of the total warehouse operating costs [2].

Order picking has been an important research topic in logistics studies for the last several decades, and scholars have used a variety of methods in their attempts to improve order picking activities including order batching, order storage assignment strategies, order picking sequencing, and stock-to-picker systems [3–6]. (i) Order batching: The goal of order batching is to minimize the travel distance between the picker and orders to obtain an optimal order batching strategy by picking the orders synchronously in one tour [4,7–10]. For example, Lu et al. (2016) [6] used an interference algorithm to realize dynamic picking path optimization, which is better than the static path algorithm and heuristic dynamic order picking path algorithm. Hong and Kim (2017) [11] proposed a batching model of path selection order based on the S-type path, which has better performance in large-scale order processing environments. To reduce the running distance of freight cars, Wu et al. (2022) [12] proposed an approach that used symmetry/asymmetry traffic context data for multicriteria decision. (ii) Storage assignment strategies: Storage assignment strategies can be divided into random storage, class-based storage, and closest

open location storage [13–15]. Karasawa et al. (1980) [16] designed an automated warehouse system-based on nonlinear mixed integer programming to minimize the storage costs. Quader and Castillo-Villar (2018) [17] proposed a new algorithm based on normative order picking, which can reduce the cycle time and increase the throughput of the system. Torre et al. (2022) [18] weighed storage risk and service level and determined the best storage location of e-commerce products through the multicriteria decision-making method. Boru et al. (2022) [19] used nondominated sorting genetic algorithm to solve inventory routing problems under different demand patterns. (iii) Order picking sequencing: The study of order picking sequencing focuses on optimizing the distribution of the order-picking areas. It has two main aspects: the sequence of order picking and the layout of picking locations [20–22]. Boysen et al. (2017) [23] presented a type of accurate heuristic algorithm to improve the space utilization of the warehouse with mobile racks. Ning et al. (2022) [24] designed a hand-eye collaborative service system based on depth vision for sorting robot. (iv) Stock-to-picker systems: According to the relative motion state between orders and picking workers, these systems can be divided into stock-to-picker and picker-to-stock [25,26]. In addition, the research on exact optimization methods also contributed the field of logistics supply chain. For deep reviews of the methods, consider the works by Amjadian and Abolfazl (2022) [27], Gharaei et al. (2022) [28], Gharaei et al. (2021) [29], and Rezaei et al. (2022) [30].

Motivated by the literature review, this paper focused on a significant order-picking problem: how to improve the picking efficiency and reduce the order picking time at a system level. In response, researchers began to use polling theory. A polling system is usually described as a system that consists of a server and a plurality of queues and includes three processes: queue arrival, serving queue, and switching service object [31]. Polling systems generally have three types of service disciplines: gated, exhaustive, and k-limited. Under gated service disciplines, the server only serves the orders that arrive at the service point before the set time [32]. Under exhaustive service disciplines, the server will not stop serving the order until the current order queue is exhausted. Under k-limited service disciplines, the server serves up to k queues at a service time. Now, polling theory has been widely used in the field of network and communication such as in wireless network random accessing applications [33] and in the antijamming ability of wireless networks to power grid systems [34]. However, we have found few studies based on limited polling theory in the field of order picking. References [35–37] first used polling theory to study the dynamic picking systems of online retailers under e-commerce considering the stochastic variables of the orders and obtained the approximate explicit expressions of the order line waiting time in the systems. Reference [4] showed that compared with traditional batch-picking methods, the proposed polling-based dynamic order-picking system can improve the order-picking efficiency.

However, the feature of the e-commerce businesses that generates an order with variable stochastic characteristics suggests that it is essentially an asymmetric optimizations problem. The diversity of order demands will lead to variations in the stochastic characteristics of the orders, which may lower the efficiency of the order picking and bring potential errors, and traditional picking systems have failed to address this issue. Most of the literature has focused on the traditional way of organizing the picking process through pick batches and releasing orders to the shop floor so that items can be picked and sorted by workers or robots. As a result, they failed to address this gap. The studies in [35–37] attempted to construct a dynamic order picking model based on asymmetric queue assumptions; however, they failed to solve the average queue length of orders. To our knowledge, the existing studies still cannot fully examine the performance of an asymmetric polling order picking system to adapt the multi-stochastic characteristics of orders.

To reveal the difference between symmetric and asymmetric polling-based order picking systems, there is a specific example. Consider that polling systems usually include three processes, queue arrival, serving queue, and switching service object [31], which are determined by three stochastic properties, the queue arrival rate (λ), queue service time

(β), and queue service switch time (γ). In the case of N queues, symmetric polling theory assumes that $\lambda_1 = \lambda_2 = \dots = \lambda_n$, $\beta_1 = \beta_2 = \dots = \beta_n$, and $\gamma_1 = \gamma_2 = \dots = \gamma_n$. For example, due to the different sizes and sales, retailers choose different order intervals to maintain their inventory levels, and the order interval directly affects the order queue arrival rate, λ . Therefore, the assumption that $\lambda_1 = \lambda_2 = \dots = \lambda_n$ is invalid, and in a real situation, the stochastic characteristics of all order queues are variable. At the very least, not every order queue is the same, which is the assumption of asymmetric polling theory [38,39]. The difference between symmetric and asymmetric polling systems is the stochastic properties constraint of order queues. The asymmetric polling system loosens the constraints on the stochastic properties compared to the symmetric polling system, which is closer to the practical production requirements.

Therefore, to resolve the research gap, this paper designed a novel asymmetric polling-based dynamic order picking system that allows orders with diversiform stochastic characteristics to exist in different picking lines. Under gated service discipline, we combined the probability generating function and the embedded Markov chain to gain the explicit solutions of mean queue length (MQL) and mean waiting time (MWT) in the model. Finally, we verified its performance by numerical examples. The rest of the paper is structured as follows. Section 2 introduces the proposed new model. Section 3 presents our numerical example. The conclusions and discussion are presented in Section 4.

2. Model

2.1. Problem Statement

Currently, most Chinese cigarette logistics distribution centers are automating their business processes; therefore, the model in this paper was put forward based on a highly automated picking system. Taking the Qijing tobacco distribution center of the Hongyunchonghe Group in Yunnan Province, China, as an example, the cigarette brands of the distribution center are diverse, and the packaging of each type of cigarette is standardized such that the goods picked are cartons of cigarettes. If cigarette retailers send order requests to the distribution center through a unified ordering system, the distribution center can pick and deliver the order within 24 h of receiving the request.

The order-picking system considered here is similar to that of Gong and De Koster (2008) [35], which they describe as a “dynamic picking system”. To help understand and implement such a complex system, the picking problem can be described as follows. The system consists of a sorter and N order queues and includes three processes: queue arrival, serving queue (picking order), and switching service object. (i) The arrival of an order starts from generation to the picking operation after a customer orders in the system. The arrival order line stream at position i is a Poisson process with expectation arrival rate λ_i . (ii) The performance of the picking service depends on the speed and efficiency of the picking machine. This process is measured by the expectation β_i of the picking time. (iii) Switching the service objects is where the picking machine switches to pick order queue $i + 1$ after serving order queue i . This process is measured by the expectation γ_i of switching times in the picking service. The length of switching time reflects the flexibility of the picking machine and the interval between different orders.

To help understand and implement such a complex system to gain explicit solutions of mean queue length and mean waiting time, the picking problem can be described as shown in Figure 1. The model divides N queues into sub-queue i (containing only queue i) and parent queue set \hat{i} (containing $N - 1$ queues except queue i). Queue $i + 1$ is the first queue of parent queue set \hat{i} . At time t_n , when the sorter machine starts picking the queue in sub-queue. $\xi_i(n)$ is the number of orders that accept the picking service in sub-queue i . Only the orders of parent queue set \hat{i} that arrive before t_{n+1} , except queue $i + 1$, can obtain the picking service at time t_n . $\xi_i(n + 1)$ is the number of orders that accept the picking service in the parent queue set \hat{i} at the time t_{n+1} . Due to the fact that $i + 1 \in \hat{i}$,

$$\sum_{j=1, j \neq i}^N \xi_j(n+1) = \xi_i(n+1).$$

Once $\xi_i(n + 1)$ orders finish the picking services, the next round

of polling starts. In addition, as a reference, we provide an example with the MATLAB program of the asymmetric polling-based dynamic order picking system in Appendix A.

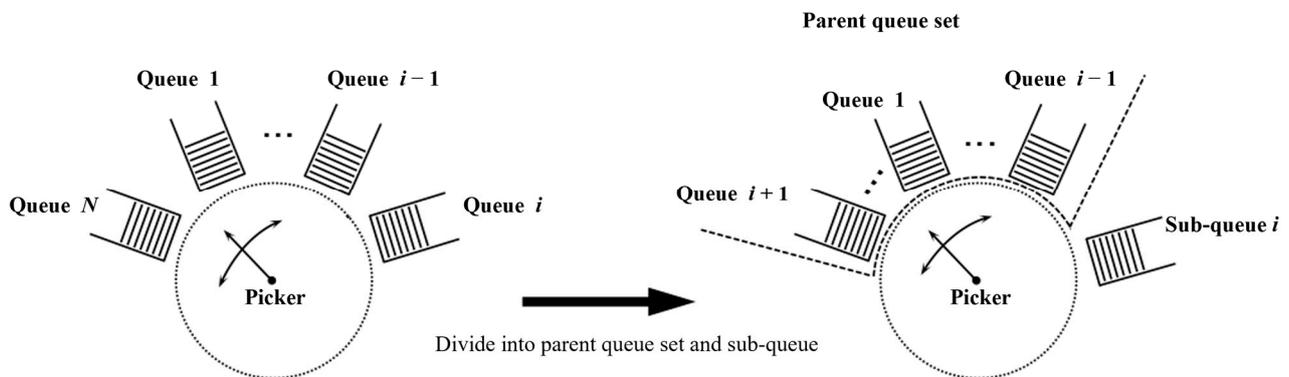


Figure 1. A model of the dynamic order picking system.

In addition, there are several basic conditions for the polling model-based picking system:

(1) The picking machines combine the unit materials into different orders according to the priority and requirements of the orders. The arrival rate, picking time, and service switching time of the orders on the same order queue have certain distributions, and the service switching time determines the distance between the adjacent orders. Each order has only one layer, and we thus only consider its area. The area is determined by the number of standardized smallest unit materials, which can be seen as the order requirements, as shown in Figure 2.

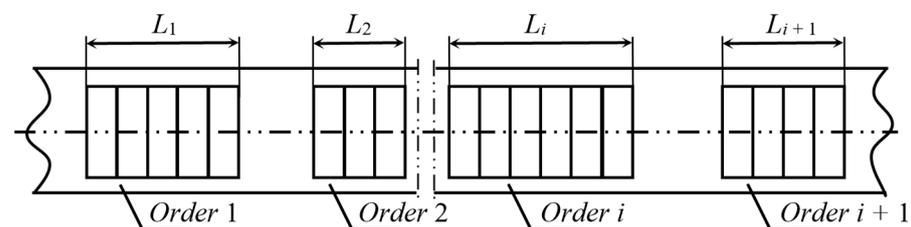


Figure 2. Order queue plan.

(2) The time of the model is divided into N unit time slots, which is a discrete time system model. (a) The arrival process of the orders was that orders form and reach the picking operation point after they are generated by the system. This process is measured by the probability distribution expectation, λ . (b) The picking service process determines the speed and efficiency of the picking machine. This process is measured by the probability distribution expectation, β . (c) The process of switching the service objects involves the picking machine switching to pick order queue $i + 1$ after serving order queue i . This process is measured by the probability distribution expectation, λ . The switching time reflects the flexibility of the picking machine and the interval between different orders.

(3) Every queue buffer (i.e., the area where an order waits to be picked) is sufficiently large, and no order will be lost. It is easy to understand that this assumption guarantees that orders can be generated continuously, i.e., the case of unlimited orders.

(4) The order queue accepts the picking service according to (a) first-come, first-served (FCFS) rules and (b) the gated service discipline, as shown in Figure 3: the discipline requires that the picking machine only sort the orders that enter the queue before starting the service; therefore, the other orders are delayed until the next processing.

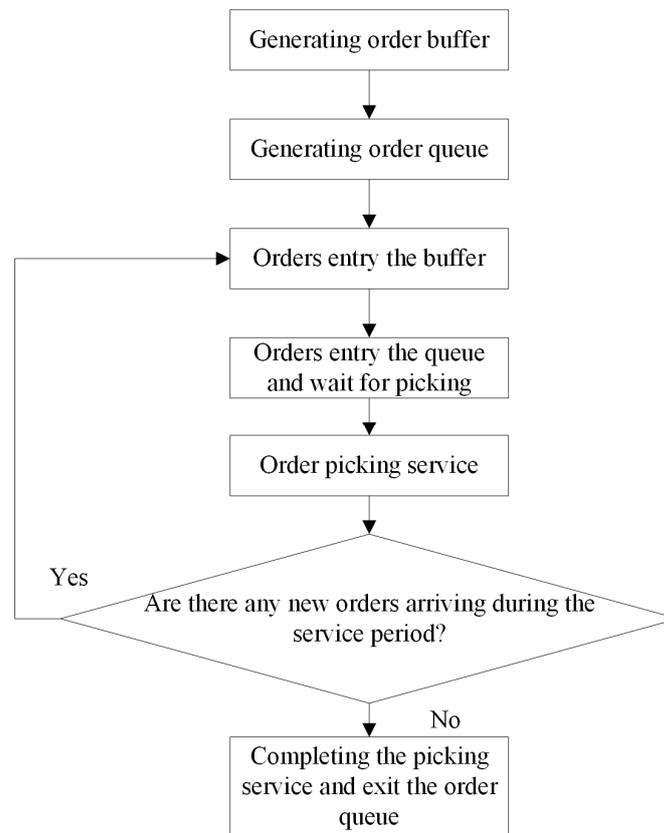


Figure 3. A flow chart of the order queue under a gated service discipline.

2.2. Notation and Definitions

(1) $A_i(z_i)$: The probability distribution function of a queue's arrival rate. $A_i'(z_i)$ is its first derivative, and $A_i''(z_i)$ is its second derivative;

(2) The arrival rates are independent of each other. The expectation λ_i and variance $\sigma_{\lambda_i}^2$ of the probability distribution all satisfy the following conditions: $\lambda_i = A_i'(1)$ and $\sigma_{\lambda_i}^2 = A_i''(1) + \lambda_i - \lambda_i^2$;

(3) $B_i(z_i)$: The probability distribution function of a queue's picking time. $B_i'(z_i)$ is its first derivative, and $B_i''(z_i)$ is its second derivative;

(4) The times of accepting the picking service are independent of each other. The expectation β_i and variance $\sigma_{\beta_i}^2$ of the probability distribution all satisfy the following conditions: $\beta_i = B_i'(1)$ and $\sigma_{\beta_i}^2 = B_i''(1) + \beta_i - \beta_i^2$;

(5) $R_i(z_i)$: The probability distribution function of the adjacent queue's picking service switching time. $R_i'(z_i)$ is its first derivative, and $R_i''(z_i)$ is its second derivative;

(6) The switching times of the picking service are independent of each other. The expectation γ_i and variance $\sigma_{\gamma_i}^2$ of the probability distribution all satisfy the following conditions: $\gamma_i = R_i'(1)$ and $\sigma_{\gamma_i}^2 = R_i''(1) + \gamma_i - \gamma_i^2$. After processing, we obtain

$$\gamma_i = \sum_{\substack{j=1 \\ j \neq i}}^N \gamma_j = \sum_{\substack{j=1 \\ j \neq i}}^N R_j'(1) \text{ and } \sigma_{\gamma_i}^2 = \sum_{\substack{j=1 \\ j \neq i}}^N [R_j''(1) + \gamma_j - \gamma_j^2];$$

(7) $\xi_i(n)$: The number of orders that arrived before moment t_n in the order queue i ; $[\xi_1(n), \xi_2(n), \dots, \xi_i(n), \dots, \xi_N(n)]$ is the state of the whole picking system at moment t_n ; $\pi_i(x_1, x_2, \dots, x_i, \dots, x_N)$ is the probability distribution of the picking system state at moment t_n ; $G_i(z_1, z_2, \dots, z_i, \dots, z_N)$ is the generating function of $\pi_i(x_1, x_2, \dots, x_i, \dots, x_N)$; $g_i(k)$ is the first order characteristic of $G_i(z_1, z_2, \dots, z_i, \dots, z_N)$, and it indicates the number of orders that are waiting to be picked in queue k when queue i starts to obtain the picking service; $g_i(j, k)$ and $g_i''(j, k)$ are its second characteristic quantity;

(8) $u_i(n)$: The time of the picking service switching from sub-queue i to order parent queue set \hat{i} ; $u_i(n)$ represents the time of picking service switching from parent queue set \hat{i} to sub-queue i ; $v_i(n)$ is the time when order queue i has picking service; $\mu_j(u_i)$ is the number of orders that arrive at order queue j within $u_i(n)$, $j \in \hat{i}$; $\mu_j(u_i)$ is the number of orders that arrive at order queue j within $u_i(n)$; $\eta_j(v_i)$ is the number of orders that arrive at order queue j within $v_i(n)$;

2.3. System Generating Function

The model starts to pick the parent queue set \hat{i} at moment t_{n+1} , and the gated control polling system theory requires that the stability of the system must meet the condition $\sum_{i=1}^N \lambda_i \beta_i < 1$ [40]. When the polling system is stable, it has the following relationships:

$$\left. \begin{aligned} \mu_j(u_i) + \eta_j \left(\sum_{\substack{j=1 \\ j \neq i}}^N v_j \right) &= \zeta_j(n), j \neq i \\ \zeta_i(n+1) + \mu_i(u_i) + \eta_i \left(\sum_{\substack{j=1 \\ j \neq i}}^N v_j \right) &= \zeta_i(n) \end{aligned} \right\} \text{at } t_n \text{ time;} \quad (1)$$

$$\left. \begin{aligned} \zeta_i(n+1) + \mu_j(u_i) + \eta_j(v_i) &= \zeta_j(n+1), j \neq i \\ \mu_i(u_i) + \eta_i(v_i) &= \zeta_i(n+1) \end{aligned} \right\} \text{at } t_{n+1} \text{ time.} \quad (2)$$

According to Equations (1) and (2), the state generating function of the parent queue set and sub-queue will be obtained:

$$G_{\hat{i}}(z_1, \dots, z_N) = R_i \left(\prod_{j=1}^N A_j(z_j) \right) G_i \left(z_1, z_2, \dots, B_i \left(\prod_{j=1}^N A_j(z_j) \right), \dots, z_N \right), i = 1, 2, \dots, N; j = 1, 2, \dots, N; \quad (3)$$

$$G_i(z_1, \dots, z_N) = R_i \left(\prod_{j=1}^N A_j(z_j) \right) G_i \left(B_1 \left(\prod_{j=1}^N A_j(z_j) \right), \dots, B_{i-1} \left(\prod_{j=1}^N A_j(z_j) \right), \right. \\ \left. z_i, B_{i+1} \left(\prod_{j=1}^N A_j(z_j) \right), \dots, B_N \left(\prod_{j=1}^N A_j(z_j) \right) \right), i = 1, 2, \dots, N; j = 1, 2, \dots, N. \quad (4)$$

2.4. The Characteristic Parameters of the System Generating Function

2.4.1. Mean Queue Length

When the queue i starts to accept the picking service, the number of orders that wait for picking in queue j is $g_i(j)$. Supposing $i = j$, $g_i(i)$ represents the number of orders that wait for the next picking in queue i buffer; it can also be called the mean queue length of the picking system. $g_i(j)$ is obtained by seeking the partial derivative of $G_i(z_1, z_2, \dots, z_i, \dots, z_N)$ as follows:

$$g_i(j) = \lim_{z_1, z_2, \dots, z_N \rightarrow 1} \frac{\partial G_i(z_1, \dots, z_j, \dots, z_N)}{\partial z_j}, j = 1, 2, \dots, N. \quad (5)$$

Equations (3) and (4) can be substituted into Equation (5) to obtain Equations (6) and (7):

$$\left\{ \begin{array}{l} R'_i(1)A'_i(1) + g_i(i) + A'_i(1) \sum_{\substack{n=1 \\ n \neq i}}^N g_i(n)B'_n(1) = \gamma\lambda_i + g_i(i) + \lambda_i \sum_{\substack{n=1 \\ n \neq i}}^N g_i(n)\beta_n = g_i(i); \\ R'_i(1)A'_i(1) + g_i(i)B'_i(1)A'_i(1) = \gamma\lambda_i + g_i(i)\beta_i\lambda_i = g_i(i); \end{array} \right. \quad (6)$$

$$\left\{ \begin{array}{l} R'_i(1)A'_j(1) + A'_j(1) \sum_{\substack{n=1 \\ n \neq i}}^N g_i(n)B'_n(1) = \gamma\beta\lambda_j + \lambda_j \sum_{\substack{n=1 \\ n \neq i}}^N g_i(n)\beta_n = g_i(j); \\ R'_i(1)A'_j(1) + g_i(i)B'_i(1)A'_j(1) + g_i(j) = \gamma\lambda_j + g_i(i)\beta_i\lambda_j + g_i(j) = g_i(j). \end{array} \right. \quad (7)$$

According to Equations (5)–(7), the mean queue length of the sub-queue and the parent queue set can be obtained as follows (the relevant proofs will be given in the Appendix B):

$$\left\{ \begin{array}{l} g_i(j) = \frac{\lambda_j \sum_{n=1}^N \gamma_n}{1 - \sum_{n=1}^N \beta_n \lambda_n}, j = 1, 2, \dots, N; \hat{i} = 1, \dots, i-1, i+1, \dots, N; \\ g_i(i) = \frac{\lambda_i \sum_{n=1}^N \gamma_n}{1 - \sum_{n=1}^N \beta_n \lambda_n}, i = 1, 2, \dots, N. \end{array} \right. \quad (8)$$

2.4.2. Mean Waiting Time

Another characteristic parameter to measure the performance of the asymmetric polling system is the mean waiting time of the system. It refers to the time from the order entering the buffer of the order queue to finishing the picking service and leaving the picking machine. The mean waiting time of the symmetric polling system considers the waiting time of the order on the arrival time slot. The mean waiting time can be expressed as follows (the relevant proofs are given in the Appendix C):

$$E[W_i]_G = -\frac{A''_i(1)}{2\lambda_i^2} + \frac{(1 + \beta_i\lambda_i)g_i(i, i)}{2\lambda_i g_i(i)}. \quad (9)$$

According to the properties of the probability-generating functions in random process $g_i(j, k)$ and $g_i(j, k)$ are obtained by seeking the second partial derivative of $G_i(z_1, z_2, \dots, z_i, \dots, z_N)$ as follows:

$$\left\{ \begin{array}{l} g_i(j, k) = \lim_{z_1, z_2, \dots, z_N \rightarrow 1} \frac{\partial^2 G_i(z_1, \dots, z_N)}{\partial z_j \partial z_k}, j = 1, 2, \dots, N; k = 1, 2, \dots, N; \\ g_i(j, k) = \lim_{z_1, z_2, \dots, z_N \rightarrow 1} \frac{\partial^2 G_i(z_1, \dots, z_N)}{\partial z_j \partial z_k}, j = 1, 2, \dots, N; k = 1, 2, \dots, N. \end{array} \right. \quad (10)$$

By combining Equations (3), (4) and (10), Equation (11) is obtained, and the relevant proofs are given in the Appendix D.

$$\begin{aligned}
 g_i(i,i) = & \left\{ \left[1 + 2 \sum_{\substack{n=1 \\ n \neq i}}^N \beta_n \lambda_n - \beta_i \lambda_i - \sum_{\substack{n=1 \\ n \neq i}}^N \beta_n \lambda_n - 2\beta_i \lambda_i \left(\sum_{\substack{n=1 \\ n \neq i}}^N \beta_n \lambda_n \right)^2 - 2 \left(\sum_{\substack{n=1 \\ n \neq i}}^N \beta_i \lambda_i \right)^3 \right] \left[R_i''(1) + \gamma_i(1 + 2\beta_i g_i(i)) + g_i(i) (\beta_i + B_i''(1)) \right] \lambda_i^2 \right. \\
 + 2 & \left[1 - \left(\sum_{\substack{n=1 \\ n \neq i}}^N \beta_n \lambda_n \right)^2 - \beta_i \lambda_i \left(\sum_{\substack{n=1 \\ n \neq i}}^N \beta_n \lambda_n \right)^2 \right] \lambda_i \gamma_i g_i(i) + 2\gamma_i \lambda_i^2 \sum_{\substack{j=1 \\ j \neq i}}^N \beta_j g_i(j) \left[1 - \left(\sum_{\substack{n=1 \\ n \neq i}}^N \beta_n \lambda_n \right)^2 + \left(\sum_{\substack{n=1 \\ n \neq i}}^N \beta_n \lambda_n \right) \right] + \lambda_i^2 \left(\beta_i \lambda_i \sum_{\substack{n=1 \\ n \neq i}}^N \beta_n \lambda_n + 1 \right) \\
 \left. \left[\gamma_i + R_i''(1) + \sum_{\substack{n=1 \\ n \neq i}}^N g_i(n) [B_n''(1) + (2\gamma_i + 1)\beta_n] \right] \right\} / & \left\{ \left[1 - 2\beta_i \lambda_i \sum_{\substack{n=1 \\ n \neq i}}^N \beta_n \lambda_n \left(\beta_i \lambda_i + \sum_{\substack{n=1 \\ n \neq i}}^N \beta_n \lambda_n \right) - (\beta_i \lambda_i)^2 - \sum_{\substack{n=1 \\ n \neq i}}^N \beta_n \lambda_n \right] \left(1 - \beta_i \lambda_i \sum_{\substack{n=1 \\ n \neq i}}^N \beta_n \lambda_n \right) \right\}. \tag{11}
 \end{aligned}$$

The mean waiting time of the asymmetric gated polling control system can be obtained by bringing Equations (8) and (11) into Equation (9).

3. Numerical Example and Discussion

In this section, we use several numerical examples to validate the performance of the new model using the MATLAB (R2012b version) software platform. First, we examined the impact of variable stochastic characteristics (i.e., $\lambda_i \beta_i$ and γ_i) on the proposed model, and then, we compared the performance of the new picking system in the cases of different amounts of the picking station. The results of the MQL and MWT illustrate the major results of the paper.

3.1. Numerical Example

To facilitate numerical analysis, we assumed that the picked goods are all the same kind and that the number of materials contained in each order is the same. The distribution center is responsible for the picking operation of the variety order. We suppose the outer package of the unit material width was 100 mm, the order conveyor speed was 60 m/min, the picking speed per of unit material was 2 s, and the order switch time was 1 min. Each group of the parameters was analyzed 5000 times, and the theoretical calculations used normalized parameters. Each order consisted of a combination of unit materials, and the speed of the conveyor was uniform. Furthermore, the system met the stable conditions of $\sum_{i=1}^N \lambda_i \beta_i < 1$ and $i = 1, 2, \dots, N$.

Figures 4 and 5, respectively, show how $\lambda\beta$ impacted the order picking system’s two important performance indexes, mean queue length (MQL) and mean waiting time (MWT), with a different number of picking station (P.S. n , $n = 1, 2, 3, 4$, and 5, meaning there were n picking stations), and we assumed that $\lambda_1:\lambda_2:\lambda_3:\lambda_4:\lambda_5 = 1:2:4:6:8$ and $\gamma_i = 10$ ($i = 1, 2, \dots, 5$). $\lambda\beta = \sum_{i=1}^N \lambda_i \beta_i$ represents the overall load level of the picking system.

The larger $\lambda\beta$ means the higher system load. With the increasing system load $\lambda\beta$, MQL and MWT also grow nonlinearly. When $\lambda\beta$ is small ($\lambda\beta < 0.1$), the difference in the MWT was not obvious and far less than that of the MQL. As the system’s load increased, the divergence of the MWT became apparent. If the other parameters were the same, the size of λ was positively related to the increase in the MQL and MWT.

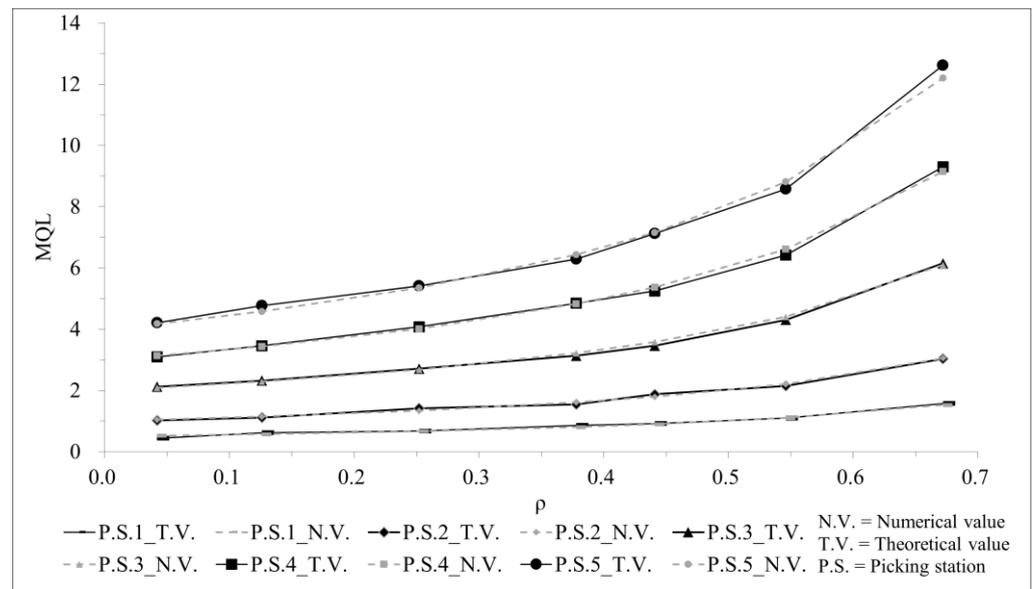


Figure 4. Range of the MQL with $\lambda\beta$'s change.

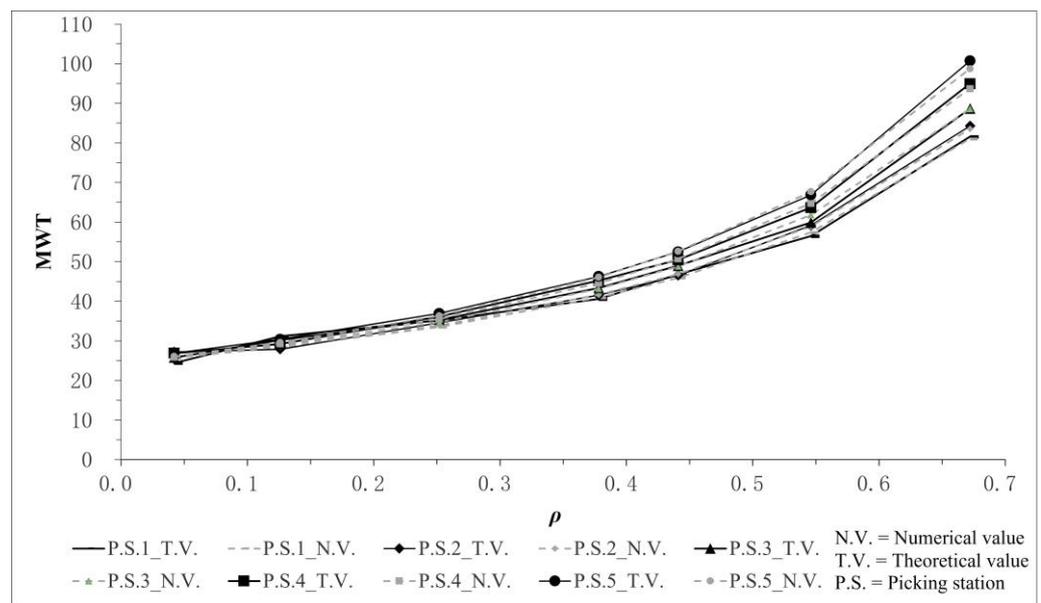


Figure 5. Range of the MWT with $\lambda\beta$'s change.

Figures 6 and 7, respectively, show how γ impacted the order picking system's mean queue length (MQL) and mean waiting time (MWT) under a low system load, and we assumed that $\lambda_1\lambda_2:\lambda_3:\lambda_4:\lambda_5 = 1:2:4:6:8$, $\beta_i = 0.6$ ($i = 1, 2, \dots, 5$), and $\lambda\beta = \sum_{i=1}^N \lambda_i\beta_i = 0.126$. Figures 8 and 9, respectively, show how γ impacted the mean queue length (MQL) and the mean waiting time (MWT) under a high system load, we assume that $\lambda_1:\lambda_2:\lambda_3:\lambda_4:\lambda_5 = 1:2:4:6:8$, $\beta_i = 3.6$ ($i = 1, 2, \dots, 5$) and $\lambda\beta = \sum_{i=1}^N \lambda_i\beta_i = 0.756$. P.S. n ($n = 1, 2, 3, 4$, and 5) means there are n picking stations. Figures 6–9 show the theoretical values of the MQL and MWT at different system load levels. At any system load level, the MQL and MWT will grow with the increasing in γ . When γ is determined, the higher system load expands the MQL and MWT. As shown in Figure 7, the differences in the picking stations' MWT were very small under a low system load.

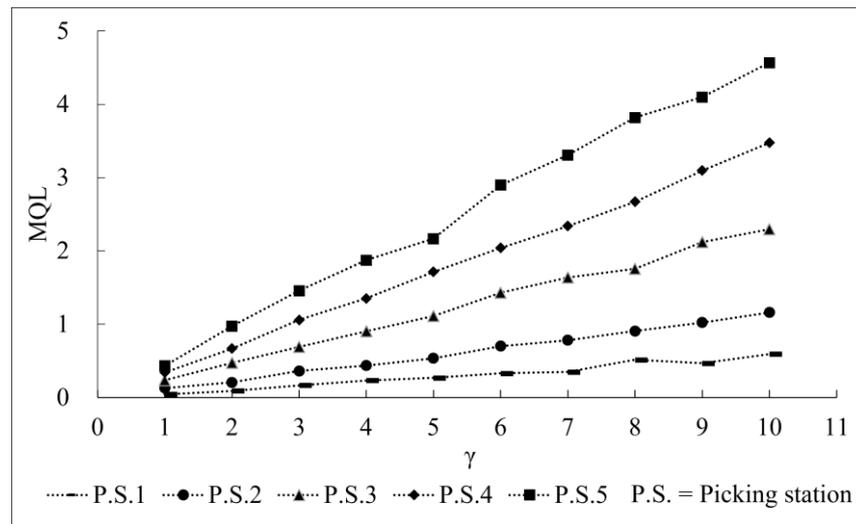


Figure 6. Range of the MQL with γ 's change under a low system load.

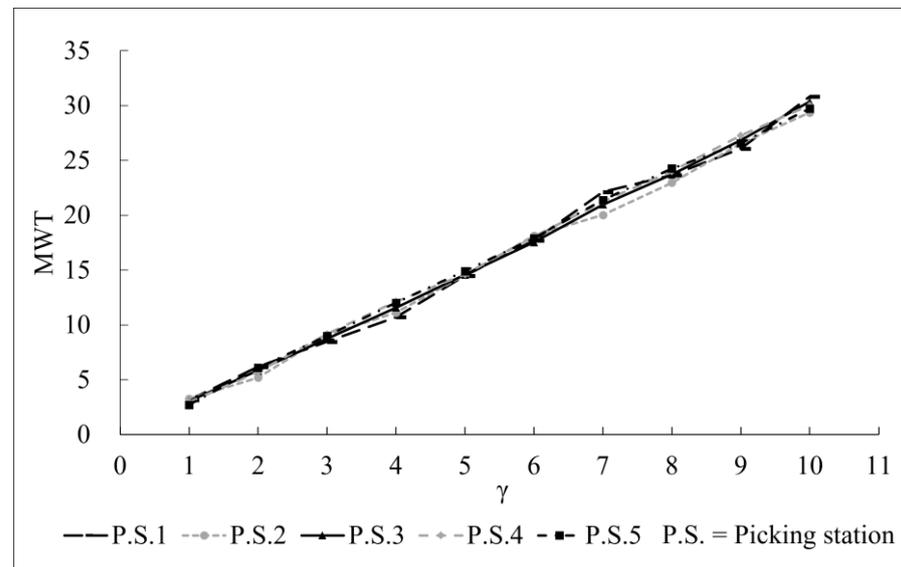


Figure 7. Range of the MWT with γ 's change under a low system load.

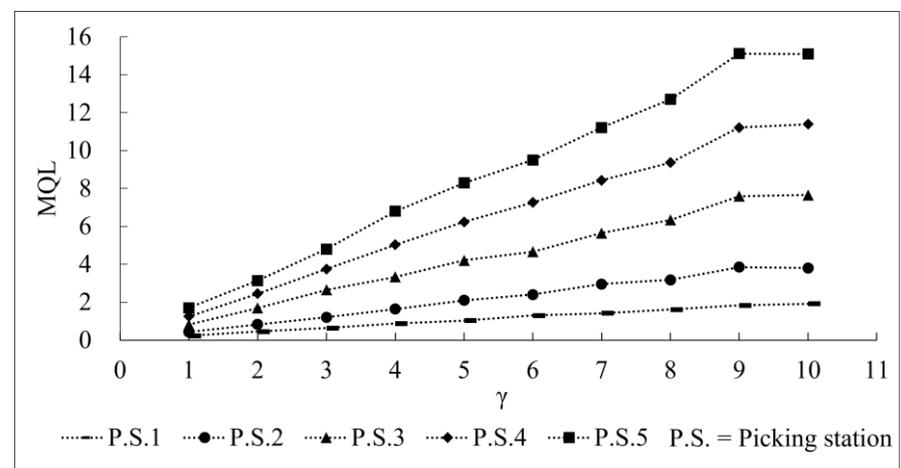


Figure 8. Range of the MQL with γ 's change under a high system load.

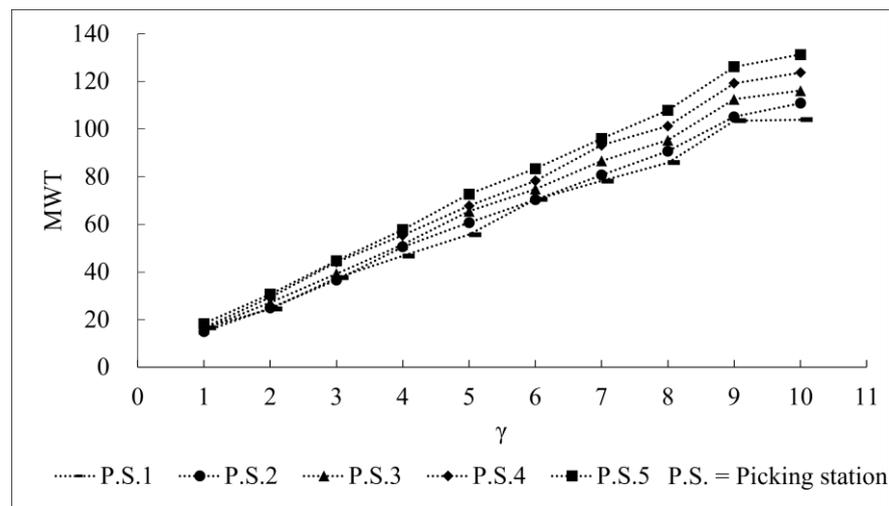


Figure 9. Range of the MWT with γ 's change under a high system load.

Table 1 show that the numerical results of the system’s performance under different amounts of the picking station. The second set of data had more than one picking station in the first set of data, and they had the same stochastic characteristic parameters as the three picking stations. The third set of data had five picking stations, but its system load ($\lambda\beta = 0.078$) was close to the second set of data ($\lambda\beta = 0.0777$).

Table 1. The numerical results of the different picking stations.

λ [β] γ	Picking Station	MQL		MWT	
		N.V.	T.V.	N.V.	T.V.
$\begin{bmatrix} 0.01 & 0.02 & 0.04 \\ 0.6 & 0.6 & 0.6 \\ 10 & 10 & 10 \end{bmatrix}$	1	0.3132	0.3180	15.2819	16.7355
	2	0.6263	0.6320	15.3760	15.8800
	3	1.2526	1.1940	15.5643	16.1084
$\begin{bmatrix} 0.01 & 0.02 & 0.04 & 0.06 \\ 0.6 & 0.6 & 0.6 & 0.6 \\ 10 & 10 & 10 & 10 \end{bmatrix}$	1	0.4338	0.4400	21.3802	21.4732
	2	0.8677	0.8500	21.5107	22.7066
	3	1.7354	1.7940	21.7717	22.0332
	4	2.6030	2.5820	22.0327	22.6792
$\begin{bmatrix} 0.01 & 0.02 & 0.04 & 0.06 & 0.08 \\ 0.37 & 0.37 & 0.37 & 0.37 & 0.37 \\ 10 & 10 & 10 & 10 & 10 \end{bmatrix}$	1	0.5422	0.5720	26.7805	27.1823
	2	1.0845	1.0880	26.8814	26.6361
	3	2.1690	2.0760	27.0830	27.2311
	4	3.2535	3.2200	27.2847	27.1912
	5	4.3380	4.3400	27.4864	27.7259

N.V. = numerical value; T.V. = theoretical value.

3.2. Discussion

(1) The numerical values were very close to the theoretical values; this means that the proposed model of an asymmetric polling-based dynamic order picking system is valid.

(2) With the increase in the $\lambda\beta$ and γ , the MQL and MWT also grew correspondingly. This shows that these stochastic characteristics ($\lambda_i\beta_i$ and γ_i) have an important impact on the performance of the picking system. The higher system load and longer switching times of a picking service can increase the queue length and waiting time of a picking system. In addition, the differences in the picking stations’ MQL were larger than that of the picking stations’ MWT with the increase in the system load, which could be due to the order arrival rate’s greater impact on the length of the order queue.

(3) Under a low system load, the differences in the picking stations' MWT were too small and not obvious. This means that the arrival rate hardly affected the waiting time of the picking system in the case of a low system load. However, with the growth in the system load, the differences in the picking stations' MWT also increased and the effect of the differences in the picking stations' arrival rates on the MWT gradually became apparent. This indicates that the picking system could better identify the stochastic characteristics of the order queues under a high load and is more flexible in actual situations.

(4) An increasing number of picking stations will enlarge the MQL and MWT, even if the system load remains the same. This means that the number of picking stations is also an important factor affecting the system's performance as well as the stochastic characteristic parameters. The diversity of the order queue characteristics complicates the picking system, and a longer queue length and waiting time of the order picking are unavoidable.

4. Conclusions

In this paper, we established a new model for an asymmetric polling-based dynamic order picking system. First, the arrival rate, service time, and service switching time of each order queue were assumed to have different probability distributions, and the picking machine was required to serve different queues according to the gated service discipline for polling. Furthermore, we used the probability generating function and embedded Markov chain to accurately solve the model and to obtain the MQL and MWT of the model.

This paper extended the theory and application of asymmetric polling in automated order picking systems and contributed to existing research. Considering the existing research [35,36,38,39,41,42], which did not take into account the influence of the diversity of the order demand on the probability distribution of the different order queues, we established an asymmetric polling-based order picking system under gated service disciplines, and our model assumed that different queues have different arrival rates, service times, and service switching times. The proposed model can more closely reflect the actual changes in the orders in e-commerce and make the order picking system more efficient and stable. The numerical examples results indicate the usefulness and practicality of the new model. It is conducive for logistics enterprises to complete efficient order assignments and provide timely and satisfactory logistics services.

Finally, our study has some limitations. For example, it has lots of necessary conditions for the new model, such as the conditions of the system stability, the order queue arrival, the picking process, and the picking service switch process. The condition change of random characteristics will directly affect the probability generating function. This may make the problem-solving methods more complex, and they may not even obtain accurate solutions. Therefore, future research should focus on reducing these constraints and making the model more flexible to adapt to the actual situations of distribution centers.

Author Contributions: Conceptualization, D.Y. and Z.Z.; methodology, D.Y. and Z.Z.; formal analysis, D.Y.; data curation, Z.Z.; writing—original draft preparation, D.Y.; writing—review and editing, Z.Z. and S.L.; supervision, S.L.; funding acquisition, S.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation Council of China under Project No. 71862035 and by The 21st Yunnan Young and Middle-Aged Academic and Technical Leaders Reserve Personnel Training Program under Grant No. 2019HB030.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All the relevant data are within the paper.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Example Program

Algorithm A1: An example program of the asymmetric polling-based dynamic order picking system with $n = 5$ picking stations

```

for i = 1:N
seg = [exprnd(1/a(1,1),1,M1);exprnd(1/a(1,2),1,M1);exprnd(1/a(1,3),1,M1);exprnd(1/a(1,4),1,M1);exprnd(1/a(1,5),1,M1)];
    T(i,:) = seg(i,:);
    end
    for i = 1:M
        start_time = t;
        for j = 1:N
            while temp(1,j) < t
                if temp(1,j) + T(j,m(1,j)) <= t
                    temp(1,j) = temp(1,j) + T(j,m(1,j));
                    n(1,j) = n(1,j) + 1;
                    m(1,j) = m(1,j) + 1;
                    each_time1(1,j) = each_time1(1,j) + (t-temp(1,j));
                else
                    break;
                end
            end
            res(j) = n(1,j)-stemp(1,j);%
            if res(j) == 0;
                t = t+r(1,j);
            else
                t = t + res(j)*b(1,j) + r(1,j);
            end
            stemp(1,j) = stemp(1,j) + res(j);
            que_length(j) = que_length(j) + res(j);
            if res(j) > 0
                for q = 1:res(j)
                    each_time2(1,j) = each_time2(1,j) + ((res(j)-q)*b(1,j));
                end
            end
        end
        time1(i) = t-start_time;
    end
    time_sum = sum(sum(time1));
    for k = 1:N
        g1(1,k) = que_length(k)/M;
        w1(1,k) = (each_time1(1,k) + each_time2(1,k))./stemp(1,k);
    end
end

```

Appendix B. Proof of Equation (8)

According to Equation (7), Equation (A1) can be obtained:

$$g_i(j) = \gamma_i \lambda_j + \gamma_i \lambda_j + g_i(i) \beta_i \lambda_j + \lambda_j \sum_{\substack{k=1 \\ k \neq i}}^N g_i(k) \beta_k. \quad (\text{A1})$$

Both sides of Equation (A1) are multiplied by $\sum_{\substack{j=1 \\ j \neq i}}^N \beta_j$:

$$\left(1 - \sum_{\substack{k=1 \\ k \neq i}}^N \beta_k \lambda_k \right) \sum_{\substack{k=1 \\ k \neq i}}^N g_i(k) \beta_k = \sum_{\substack{k=1 \\ k \neq i}}^N \beta_k \lambda_k [\gamma_i + \gamma_i + g_i(i) \beta_i]$$

$$\sum_{\substack{k=1 \\ k \neq i}}^N g_i(k) \beta_k = \frac{\sum_{k=1}^N \beta_k \lambda_k \left[\sum_{k=1}^N \gamma_k + g_i(i) \beta_i \right]}{1 - \sum_{\substack{k=1 \\ k \neq i}}^N \beta_k \lambda_k} \tag{A2}$$

According to Equation (6), Equation (A3) can be obtained:

$$g_i(i) = \gamma_i \lambda_i + \gamma_i \lambda_i + g_i(i) \beta_i \lambda_i + \lambda_i \sum_{\substack{k=1 \\ k \neq i}}^N g_i(k) \beta_k$$

$$= \lambda_i \sum_{k=1}^N \gamma_k + g_i(i) \beta_i \lambda_i + \lambda_i \sum_{\substack{k=1 \\ k \neq i}}^N g_i(k) \beta_k \tag{A3}$$

Then, Equation (A2) can be taken into Equation (A3):

$$g_i(i) = \lambda_i \sum_{k=1}^N \gamma_k + g_i(i) \beta_i \lambda_i + \lambda_i \frac{\sum_{\substack{k=1 \\ k \neq i}}^N \beta_k \lambda_k \left[\sum_{k=1}^N \gamma_k + g_i(i) \beta_i \right]}{1 - \sum_{\substack{k=1 \\ k \neq i}}^N \beta_k \lambda_k}$$

$$\left(1 - \sum_{\substack{k=1 \\ k \neq i}}^N \beta_k \lambda_k \right) g_i(i) (1 - \beta_i \lambda_i) = \lambda_i \left[\sum_{k=1}^N \gamma_k \left(1 - \sum_{\substack{k=1 \\ k \neq i}}^N \beta_k \lambda_k \right) + \sum_{\substack{k=1 \\ k \neq i}}^N \beta_k \lambda_k \left[\sum_{k=1}^N \gamma_k + g_i(i) \beta_i \right] \right]$$

$$g_i(i) \left(1 - \sum_{k=1}^N \beta_k \lambda_k \right) + \beta_i \lambda_i g_i(i) \sum_{\substack{k=1 \\ k \neq i}}^N \beta_k \lambda_k = \lambda_i \sum_{k=1}^N \gamma_k + \beta_i \lambda_i g_i(i) \sum_{\substack{k=1 \\ k \neq i}}^N \beta_k \lambda_k$$

$$g_i(i) = \frac{\lambda_i \sum_{k=1}^N \gamma_k}{1 - \sum_{k=1}^N \beta_k \lambda_k} \tag{A4}$$

Bringing Equation (A4) into Equation (A2) provides the following:

$$\sum_{\substack{k=1 \\ k \neq i}}^N g_i(k)\beta_k = \frac{\sum_{k=1}^N \beta_k \lambda_k \left[\sum_{k=1}^N \gamma_k \left(1 - \sum_{n=1}^N \beta_n \lambda_n \right) + \lambda_i \beta_i \sum_{n=1}^N \gamma_n \right]}{\begin{pmatrix} 1 - \sum_{\substack{k=1 \\ k \neq i}}^N \beta_k \lambda_k \\ \left(1 - \sum_{n=1}^N \beta_n \lambda_n \right) \end{pmatrix}}. \tag{A5}$$

By combining Equations (7) and (A5), Equation (A6) can be obtained:

$$\begin{aligned} g_i(j) &= \gamma_i \lambda_j + \frac{\lambda_j \sum_{\substack{k=1 \\ k \neq i}}^N \beta_k \lambda_k \left[\sum_{k=1}^N \gamma_k \left(1 - \sum_{n=1}^N \beta_n \lambda_n \right) + \lambda_i \beta_i \sum_{n=1}^N \gamma_n \right]}{\begin{pmatrix} 1 - \sum_{\substack{k=1 \\ k \neq i}}^N \beta_k \lambda_k \\ \left(1 - \sum_{n=1}^N \beta_n \lambda_n \right) \end{pmatrix}} \\ &= \gamma_i \lambda_j + \frac{\lambda_j \sum_{\substack{k=1 \\ k \neq i}}^N \beta_k \lambda_k \sum_{k=1}^N \gamma_k \left(1 - \sum_{n=1}^N \beta_n \lambda_n + \lambda_i \beta_i \right)}{\begin{pmatrix} 1 - \sum_{\substack{k=1 \\ k \neq i}}^N \beta_k \lambda_k \\ \left(1 - \sum_{n=1}^N \beta_n \lambda_n \right) \end{pmatrix}} \\ &= \gamma_i \lambda_j + \frac{\lambda_j \sum_{\substack{k=1 \\ k \neq i}}^N \beta_k \lambda_k \sum_{k=1}^N \gamma_k}{1 - \sum_{n=1}^N \beta_n \lambda_n}. \end{aligned} \tag{A6}$$

Taking Equations (A6) and (A4) into Equation (7) gives the following:

$$\begin{aligned} g_i(j) &= \gamma_i \lambda_j + \frac{\beta_i \lambda_j \lambda_i \sum_{k=1}^N \gamma_k}{1 - \sum_{k=1}^N \beta_k \lambda_k} + \gamma_i \lambda_j + \frac{\lambda_j \sum_{\substack{k=1 \\ k \neq i}}^N \beta_k \lambda_k \sum_{k=1}^N \gamma_k}{1 - \sum_{n=1}^N \beta_n \lambda_n} \\ &= \lambda_j \sum_{k=1}^N \gamma_k + \frac{\beta_i \lambda_j \lambda_i \sum_{k=1}^N \gamma_k + \lambda_j \sum_{\substack{k=1 \\ k \neq i}}^N \beta_k \lambda_k \sum_{k=1}^N \gamma_k}{1 - \sum_{k=1}^N \beta_k \lambda_k} \\ &= \frac{\lambda_j \sum_{k=1}^N \gamma_k \begin{pmatrix} 1 - \sum_{k=1}^N \beta_k \lambda_k + \beta_i \lambda_i + \sum_{\substack{k=1 \\ k \neq i}}^N \beta_k \lambda_k \end{pmatrix}}{1 - \sum_{k=1}^N \beta_k \lambda_k} \\ &= \frac{\lambda_j \sum_{k=1}^N \gamma_k}{1 - \sum_{k=1}^N \beta_k \lambda_k}. \end{aligned} \tag{A7}$$

Appendix C. Proof of Equation (9)

According to gated service discipline, the order queue i arrives at the service counter during $\tau_m(\tau_m = t_{m+1} - t_m)$ and accepts the picking service at t_n time. The order k in queue i ends the picking service at $t_m^*(t_m < t_{m+1} < t_n < t_m^*)$. We assume that $\theta_i(n)$ represents the time interval of the picking machine serving the same order queue i and there are $y_i^a(n)$ and $y_i^b(n)$ that satisfy the following conditions:

$$\begin{cases} y_i^a(n) = \xi_i(1)[\theta_i(n) - 1]^+ + \xi_i(2)[\theta_i(n) - 2]^+ + \dots + \xi_i(n)[\theta_i(n) - n]^+, i = 1, 2, \dots, N, \\ [\theta_i(n) - k]^+ = \max[\theta_i(n) - k, 0]; \\ y_i^b(n) = v_{i,1}(n) + [v_{i,1}(n) + v_{i,2}(n)] + \dots + \sum_{k=1}^{\xi_i(n)-1} v_{i,k}(n), i = 1, 2, \dots, N. \end{cases} \tag{A8}$$

From the gated service strategy, the waiting time W_i of the order can be represented by the following formula:

$$W_i = W_i^a + W_i^b. \tag{A9}$$

The expectations of W_i^a and W_i^b can be obtained as follows:

$$\begin{cases} E(W_i^a) = \frac{E[y_i^a(n)]}{E[\xi_i(n)]} = \lim_{T \rightarrow \infty} \frac{\sum_{n=1}^T y_i^a(n)}{\sum_{n=1}^T \xi_i(n)} = -\frac{A_i''(1)}{2\lambda_i^2} + \frac{g_i(i,i)}{2\lambda_i g_i(i)}; \\ E(W_i^b) = \frac{E[y_i^b(n)]}{E[\xi_i(n)]} = \lim_{T \rightarrow \infty} \frac{\sum_{n=1}^T y_i^b(n)}{\sum_{n=1}^T \xi_i(n)} = \frac{\beta_i g_i(i,i)}{2g_i(i)}. \end{cases} \tag{A10}$$

Therefore, the mean waiting time $E[W_i]_G$ can be obtained:

$$E[W_i]_G = E(W_i^a) + E(W_i^b) = -\frac{A_i''(1)}{2\lambda_i^2} + \frac{(1 + \beta_i \lambda_i)g_i(i,i)}{2\lambda_i g_i(i)} \tag{A11}$$

Appendix D. Proof of Equation (11)

Seeking the two-order partial derivative of Equations (10), (A12) and (A13) can be obtained:

$$\begin{cases} g_i(i,i) = \lambda_i^2 R_i''(1) + \gamma_i A_i''(1) + 2\lambda_i \gamma_i g_i(i) + 2\lambda_i \sum_{\substack{n=1 \\ n \neq i}}^N \beta_n g_i(i,n) + \lambda_i^2 \sum_{\substack{n=1 \\ n \neq i}}^N \sum_{\substack{m=1 \\ m \neq i}}^N \beta_m \beta_n g_i(m,n) \\ + g_i(i,i) + \sum_{\substack{n=1 \\ n \neq i}}^N g_i(n) [A_i''(1)\beta_n + 2\lambda_i^2 \gamma_i \beta_n + \lambda_i^2 B_n''(1)]; \\ g_i(i,i) = \lambda_i^2 R_i''(1) + \gamma_i A_i''(1) + g_i(i) [A_i''(1)\beta_i + 2\lambda_i^2 \gamma_i \beta_i + \lambda_i^2 B_i''(1)] + \lambda_i^2 \beta_i^2 g_i(i,i); \\ g_i(i,j) = \lambda_i \lambda_j [R_i''(1) + \gamma_i] + \lambda_i \lambda_j g_i(i) [\beta_i + 2\gamma_i \beta_i + B_i''(1)] + \lambda_i [\gamma_i g_i(j) + \beta_i g_i(i,j) \\ + \lambda_j \beta_i^2 g_i(i,i)]. \end{cases} \tag{A12}$$

$$\left\{ \begin{aligned}
 &g_i(j, j) = \lambda_j^2 R_i''(1) + \gamma_i A_j''(1) + g_i(i) \left[A_j''(1) \beta_i + 2\lambda_j^2 \gamma_i \beta_i + \lambda_j^2 B_i''(1) \right] + g_i(j, j) \\
 &+ \lambda_j [2\gamma_i g_i(j) + 2\beta_i g_i(i, j) + \lambda_j \beta_i^2 g_i(i, i)]; \\
 &g_i(i, j) = \lambda_i \lambda_j \left[R_i''(1) + \gamma_i \right] + \lambda_j \gamma_i g_i(i) + \lambda_j \sum_{\substack{n=1 \\ n \neq i}}^N \beta_n g_i(i, n) + \lambda_i \lambda_j \sum_{\substack{n=1 \\ n \neq i}}^N \sum_{\substack{m=1 \\ m \neq i}}^N \beta_m \beta_n g_i(m, n) \\
 &\sum_{\substack{n=1 \\ n \neq i}}^N g_i(n) [\lambda_i \lambda_j \beta_n + 2\lambda_i \lambda_j \gamma_i \beta_n + \lambda_i \lambda_j B_n''(1)]; \\
 &g_i(j, k) = \lambda_k \lambda_j \left[R_i''(1) + \gamma_i \right] + \sum_{\substack{n=1 \\ n \neq i}}^N g_i(n) [\lambda_k \lambda_j \beta_n + 2\lambda_k \lambda_j \gamma_i \beta_n + \lambda_k \lambda_j B_n''(1)] + \\
 &\lambda_k \lambda_j \sum_{\substack{n=1 \\ n \neq i}}^N \sum_{\substack{m=1 \\ m \neq i}}^N \beta_m \beta_n g_i(m, n).
 \end{aligned} \right. \tag{A13}$$

Accordingly, Equations (A12) and (A14) can be obtained:

$$\begin{aligned}
 g_i(i, i) = & \left\{ \lambda_i^2 \left(\beta_i \lambda_i \sum_{\substack{n=1 \\ n \neq i}}^N \beta_n \lambda_n + 1 \right) \left[\gamma_i + R_i''(1) + \sum_{\substack{n=1 \\ n \neq i}}^N g_i(n) [B_n''(1) + (2\gamma_i + 1) \beta_n] \right] \right. \\
 & + 2\lambda_j \gamma_i g_i(i) + \lambda_i^2 \left(1 + \beta_i \lambda_i \sum_{\substack{n=1 \\ n \neq i}}^N \beta_n \lambda_n \right) \sum_{\substack{n=1 \\ n \neq i}}^N \sum_{\substack{m=1 \\ m \neq i}}^N \beta_m \beta_n g_i(m, n) + 2\lambda_i^2 \gamma_i \sum_{\substack{j=1 \\ j \neq i}}^N \beta_j g_i(j) \\
 & \left. + \lambda_i^2 [R_i''(1) + \gamma_i (1 + 2\beta_i g_i(i)) + g_i(i) (\beta_i + B_i''(1))] \left[1 + (2 - \beta_i \lambda_i) \sum_{\substack{n=1 \\ n \neq i}}^N \beta_n \lambda_n \right] \right\} / \\
 & \left\{ \left(1 - \beta_i \lambda_i \sum_{\substack{n=1 \\ n \neq i}}^N \beta_n \lambda_n \right) (1 - \beta_i^2 \lambda_i^2) - 2\beta_i^2 \lambda_i^2 \sum_{\substack{n=1 \\ n \neq i}}^N \beta_n \lambda_n \right\}
 \end{aligned} \tag{A14}$$

Accordingly, Equations (A13) and (A15) can be obtained:

$$\begin{aligned}
\sum_{\substack{n=1 \\ n \neq i}}^N \sum_{\substack{m=1 \\ m \neq i}}^N \beta_m \beta_n g_i(m, n) &= \left\{ \left(\sum_{\substack{n=1 \\ n \neq i}}^N \beta_n \lambda_n \right)^2 [R_i''(1) + \gamma_i(1 + 2\beta_i g_i(i)) + g_i(i)(\beta_i + B_i''(1))] \right. \\
&\left. \left(1 + \beta_i \lambda_i \sum_{\substack{n=1 \\ n \neq i}}^N \beta_n \lambda_n \right) + 2\gamma_i \sum_{\substack{n=1 \\ n \neq i}}^N \beta_n \lambda_n \sum_{\substack{m=1 \\ m \neq i}}^N \beta_m g_i(m) + \left[1 + \beta_i \lambda_i \left(2 - \sum_{\substack{n=1 \\ n \neq i}}^N \beta_n \lambda_n \right) \right] \right. \\
&\left. \left[\gamma_i + R_i''(1) + \sum_{\substack{n=1 \\ n \neq i}}^N g_i(n) [B_n''(1) + (2\gamma_i + 1)\beta_n] \right] + 2 \left(\sum_{\substack{n=1 \\ n \neq i}}^N \beta_n \lambda_n \right)^2 \beta_i \gamma_i g_i(i) + \right. \\
&\left. \left(\sum_{\substack{n=1 \\ n \neq i}}^N \beta_n \lambda_n \right)^2 \beta_i^2 \left(1 + \beta_i \lambda_i \sum_{\substack{n=1 \\ n \neq i}}^N \beta_n \lambda_n \right) g_i(i, i) \right\} / \\
&\left\{ \left(1 - \beta_i \lambda_i \sum_{\substack{n=1 \\ n \neq i}}^N \beta_n \lambda_n \right) \left(1 - \sum_{\substack{n=1 \\ n \neq i}}^N \beta_n \lambda_n \right)^2 - 2\beta_i \lambda_i \left(\sum_{\substack{n=1 \\ n \neq i}}^N \beta_n \lambda_n \right)^2 \right\}
\end{aligned} \tag{A15}$$

By combining Equations (11), (A14) and (A15) can be obtained.

References

1. Yang, J.Q.; Zhang, X.M.; Fu, H.Y.; Liu, C. Inventory competition in a dual-channel supply chain with delivery lead time consideration. *Appl. Math. Model.* **2017**, *42*, 675–692. [\[CrossRef\]](#)
2. Rubrico, J.I.U.; Higashi, T.; Tamura, H.; Ota, J. Online rescheduling of multiple picking agents for warehouse management. *Robot. Comput.-Integr. Manuf.* **2011**, *27*, 62–71. [\[CrossRef\]](#)
3. Malmborg, C.J.; Al-Tassan, K. An integrated performance model for orderpicking systems with randomized storage. *Appl. Math. Model.* **2000**, *24*, 95–111. [\[CrossRef\]](#)
4. de Koster, R.; Le-Duc, T.; Roodbergen, K.J. Design and control of warehouse order picking: A literature review. *Eur. J. Oper. Res.* **2007**, *182*, 481–501. [\[CrossRef\]](#)
5. Manzini, R.; Accorsi, R.; Gamberi, M.; Penazzi, S. Modeling class-based storage assignment over life cycle picking patterns. *Int. J. Prod. Econ.* **2015**, *170 Pt C*, 790–800. [\[CrossRef\]](#)
6. Lu, W.; McFarlane, D.; Giannikas, V.; Zhang, Q. An algorithm for dynamic order-picking in warehouse operations. *Eur. J. Oper. Res.* **2016**, *248*, 107–122. [\[CrossRef\]](#)
7. Won, J.; Olafsson, S. Joint order batching and order picking in warehouse operations. *Int. J. Prod. Res.* **2005**, *43*, 1427–1442. [\[CrossRef\]](#)
8. Tang, L.; Gong, H. A hybrid two-stage transportation and batch scheduling problem. *Appl. Math. Model.* **2008**, *32*, 2467–2479. [\[CrossRef\]](#)
9. Grubbström, R.W. Dynamic lotsizing with a finite production rate. *Int. J. Prod. Econ.* **2014**, *149*, 68–79. [\[CrossRef\]](#)
10. Lam, C.H.Y.; Choy, K.L.; Ho, G.T.S.; Lee, C.K.M. An order-picking operations system for managing the batching activities in a warehouse. *Int. J. Syst. Sci.* **2014**, *45*, 1283–1295. [\[CrossRef\]](#)
11. Hong, S.; Kim, Y. A route-selecting order batching model with the S-shape routes in a parallel-aisle order picking system. *Eur. J. Oper. Res.* **2017**, *257*, 185–196. [\[CrossRef\]](#)
12. Wu, M.-Y.; Ke, C.-K.; Lai, S.-C. Optimizing the Routing of Urban Logistics by Context-Based Social Network and Multi-Criteria Decision Analysis. *Symmetry* **2022**, *14*, 1811. [\[CrossRef\]](#)
13. Rouwenhorst, B.; Reuter, B.; Stockrahm, V.; van Houtum, G.J.; Mantel, R.J.; Zijm, W.H.M. Warehouse design and control: Framework and literature review. *Eur. J. Oper. Res.* **2000**, *122*, 515–533. [\[CrossRef\]](#)
14. Atmaca, E.; Ozturk, A. Defining order picking policy: A storage assignment model and a simulated annealing solution in AS/RS systems. *Appl. Math. Model.* **2013**, *37*, 5069–5079. [\[CrossRef\]](#)

15. Guerriero, F.; Pisacane, O.; Rende, F. Comparing heuristics for the product allocation problem in multi-level warehouses under compatibility constraints. *Appl. Math. Model.* **2015**, *39*, 7375–7389. [[CrossRef](#)]
16. Karasawa, Y.; Nakayama, H.; Dohi, S. Trade-off analysis for optimal design of automated warehouses. *Int. J. Syst. Sci.* **1980**, *11*, 567–576. [[CrossRef](#)]
17. Quader, S.; Castillo-Villar, K.K. Design of an enhanced multi-aisle order-picking system considering storage assignments and routing heuristics. *Robot. Comput.-Integr. Manuf.* **2018**, *50*, 13–29. [[CrossRef](#)]
18. Torre, N.M.M.; Salomon, V.A.P.; Loche, E.; Gazale, S.A.; Palermo, V.M. Warehouse Location for Product Distribution by E-Commerce in Brazil: Comparing Symmetrical MCDM Applications. *Symmetry* **2022**, *14*, 1987. [[CrossRef](#)]
19. Boru, A.; Dosdoğru, A.T.; Göçken, M.; Erol, R. A Novel Hybrid Artificial Intelligence Based Methodology for the Inventory Routing Problem. *Symmetry* **2019**, *11*, 717. [[CrossRef](#)]
20. Van Nieuwenhuysse, I.; de Koster, R.B.M. Evaluating order throughput time in 2-block warehouses with time window batching. *Int. J. Prod. Econ.* **2009**, *121*, 654–664. [[CrossRef](#)]
21. Theys, C.; Bräysy, O.; Dullaert, W.; Raa, B. Using a TSP heuristic for routing order pickers in warehouses. *Eur. J. Oper. Res.* **2010**, *200*, 755–763. [[CrossRef](#)]
22. Rath, S.; Gutjahr, W.J. A math-heuristic for the warehouse location-routing problem in disaster relief. *Comput. Oper. Res.* **2014**, *42*, 25–39. [[CrossRef](#)]
23. Boysen, N.; Briskorn, D.; Emde, S. Sequencing of picking orders in mobile rack warehouses. *Eur. J. Oper. Res.* **2017**, *259*, 293–307. [[CrossRef](#)]
24. Ning, T.; Wang, C.; Han, Y. Deep Vision Servo Hand-Eye Coordination Planning Study for Sorting Robots. *Symmetry* **2022**, *14*, 152. [[CrossRef](#)]
25. de Koster, R. Performance approximation of pick-to-belt orderpicking systems. *Eur. J. Oper. Res.* **1994**, *72*, 558–573. [[CrossRef](#)]
26. Pan, J.C.-H.; Wu, M.-H. A study of storage assignment problem for an order picking line in a pick-and-pass warehousing system. *Comput. Ind. Eng.* **2009**, *57*, 261–268. [[CrossRef](#)]
27. Amjadian, A.; Gharaei, A. An integrated reliable five-level closed-loop supply chain with multi-stage products under quality control and green policies: Generalised outer approximation with exact penalty. *Int. J. Syst. Sci. Oper. Logist.* **2022**, *9*, 429–449. [[CrossRef](#)]
28. Gharaei, A.; Amjadian, A.; Amjadian, A.; Shavandi, A.; Hashemi, A.; Taher, M.; Mohamadi, N. An integrated lot-sizing policy for the inventory management of constrained multi-level supply chains: Null-space method. *Int. J. Syst. Sci. Oper. Logist.* **2022**, 1–14. [[CrossRef](#)]
29. Gharaei, A.; Amjadian, A.; Shavandi, A. An integrated reliable four-level supply chain with multi-stage products under shortage and stochastic constraints. *Int. J. Syst. Sci. Oper. Logist.* **2021**, 1–22. [[CrossRef](#)]
30. Baradaran Rezaei, H.; Amjadian, A.; Sebt, M.V.; Askari, R.; Gharaei, A. An ensemble method of the machine learning to prognosticate the gastric cancer. *Ann. Oper. Res.* **2022**, 1–42. [[CrossRef](#)]
31. Levy, H.; Sidi, M. Polling systems: Applications, modeling, and optimization. *IEEE Trans. Commun.* **1990**, *38*, 1750–1760. [[CrossRef](#)]
32. Eliazar, I. Gated Polling Systems with Lévy Inflow and Inter-Dependent Switchover Times: A Dynamical-Systems Approach. *Queueing Syst.* **2005**, *49*, 49–72. [[CrossRef](#)]
33. Dorsman, J.-P.L.; Borst, S.C.; Boxma, O.J.; Vlasiou, M. Markovian polling systems with an application to wireless random-access networks. *Perform. Eval.* **2015**, *85–86*, 33–51. [[CrossRef](#)]
34. Abdrabou, A.; Gaouda, A.M. Considerations for packet delivery reliability over polling-based wireless networks in smart grids. *Comput. Electr. Eng.* **2015**, *41*, 368–382. [[CrossRef](#)]
35. Gong, Y.; De Koster, R. A polling-based dynamic order picking system for online retailers. *IIE Trans.* **2008**, *40*, 1070–1082. [[CrossRef](#)]
36. Gong, Y.; Winands, E.M.M.; de Koster, R.e.B.M. A Real-Time Picking and Sorting System in E-Commerce Distribution Centers. 2010. Available online: <http://www.mhi.org/downloads/learning/cicmhe/colloquium/2010/gong.pdf> (accessed on 30 October 2022).
37. Ran, W.; Liu, S.; Zhang, Z. A Polling-Based Dynamic Order-Picking System considering Priority Orders. *Complexity* **2020**, *2020*, 4595316. [[CrossRef](#)]
38. Takine, T.; Takagi, H.; Takahashi, Y.; Hasegawa, T. Analysis of asymmetric single-buffer polling and priority systems without switchover times. *Perform. Eval.* **1990**, *11*, 253–264. [[CrossRef](#)]
39. Levy, H. Delay computation and dynamic behavior of non-symmetric polling systems. *Perform. Eval.* **1989**, *10*, 35–51. [[CrossRef](#)]
40. Konheim, A.G.; Meister, B. Waiting Lines and Times in a System with Polling. *J. ACM* **1974**, *21*, 470–490. [[CrossRef](#)]
41. Gianini, J.; Manfield, D.R. An analysis of symmetric polling systems with two priority classes. *Perform. Eval.* **1988**, *8*, 93–115. [[CrossRef](#)]
42. Takagi, H. On the analysis of a symmetric polling system with single-message buffers. *Perform. Eval.* **1985**, *5*, 149–157. [[CrossRef](#)]