



Article

Online and Offline Model for Housekeeping Order Assignment Based on Dynamic Programming Algorithm

Yongkun Shi ¹, Jiangkuan Li ², Jiaquan Sun ¹, Cunhao Lu ³ , Jian Chen ^{3,*}  and Xiaoguang Sun ^{4,5}

- ¹ College of Electrical, Energy and Power Engineering, Yangzhou University, Huayang West Road 196, Yangzhou 225127, China; 213302218@stu.yzu.edu.cn (Y.S.); 213302219@stu.yzu.edu.cn (J.S.)
 - ² School of Information Engineering (School of Artificial Intelligence), Yangzhou University, Huayang West Road 196, Yangzhou 225127, China; 211803313@stu.yzu.edu.cn
 - ³ School of Mechanical Engineering, Yangzhou University, Huayang West Road 196, Yangzhou 225127, China; lch_ok@yzu.edu.cn
 - ⁴ Guangdong International Cooperation Base of Science and Technology for GBA Smart Logistics, Jinan University, Zhuhai 519070, China; xgsun@jnu.edu.cn
 - ⁵ School of Intelligent Systems Science and Engineering, Jinan University, Zhuhai 519070, China
- * Correspondence: jian.chen@yzu.edu.cn

Abstract: With the booming development of door-to-door housekeeping service, the platform faces the problem of order assignment. Improving the matching mechanism between orders and housekeepers based on a dynamic programming (DP) algorithm can not only achieve flexible order allocation but can also improve the service efficiency and service quality. In this paper, a single objective nonlinear programming model is established, which takes the maximum total weight value as the objective function to study the order allocation problem under offline and online conditions. Under the offline condition, the number of housekeepers is taken as the decision variable. The status of order and housekeeper, order time, and action trajectory are taken as constraints. For online assignment, the order backlog status is treated as the decision variable. The reliability of the model was verified using real data from 20 groups of housekeepers and 50 groups of orders. Finally, the effect of order backlog on online allocation is discussed and the optimal threshold and maximum weight are found. The online order assignment model is compared with the nearest distance assignment model. The results show that the online assignment model with a total weighted score of 1045.14 is better than the nearest distance assignment model with a score of 810.25.

Keywords: dynamic programming; matching mechanism; order assignment; order backlog



Citation: Shi, Y.; Li, J.; Sun, J.; Lu, C.; Chen, J.; Sun, X. Online and Offline Model for Housekeeping Order Assignment Based on Dynamic Programming Algorithm. *Appl. Sci.* **2024**, *14*, 576. <https://doi.org/10.3390/app14020576>

Academic Editor: Chilukuri K. Mohan

Received: 15 November 2023

Revised: 5 January 2024

Accepted: 7 January 2024

Published: 9 January 2024



Copyright: © 2024 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

With the continuous development of the social economy and the improvement in people's living standards, domestic service has gradually become an indispensable part of modern city residents' lives. The related service platform provides house cleaning, babysitting, moving, maintenance, and other services for people's lives. When the customer places a service order on the platform, the platform will assign the order to the service personnel, who will provide the cleaning/takeaway ordering services according to the service time specified by the customer [1–3]. As an important part of the daily life of city residents, this kind of domestic service is of great importance to society for its efficient and high-quality operation.

However, with the increasing demand for domestic service and the continuous growth of service personnel, the issue of order allocation has become increasingly prominent, and order matching is a core issue in the operation of domestic service platforms [4]. Optimizing order matching can significantly improve the efficiency and quality of domestic service, so that the service demand-side can be better satisfied, but can also improve the income and employment satisfaction of domestic service personnel, and then promote the development

of related industries. Therefore, the research and practice of an order matching algorithm has important social significance for promoting the healthy development of urban domestic service industry and improving the quality of life of urban residents.

In the current domestic service platform, there are some pain points and problems in order matching. Firstly, the highly diverse and uncertain demand for domestic services leads to the complexity of order matching. Second, the number and skill level of service personnel are unbalanced, which often leads to a waste of resources or an imbalance between supply and demand in the matching process. In addition, order matching requires high timeliness, and the traditional manual matching method often struggles to meet the need for a rapid response. With fierce platform competition and increasing customer demand, platforms are seeking efficient order matching strategies to reduce costs and improve service quality.

There are many factors that affect order matching, such as butler rating, travel speed, order start time, order location, and so on. How to combine these factors into a comprehensive order assignment strategy is a complex problem. Dynamic programming (DP) is an optimization method for solving multi-stage decision problems. By dividing complex problems into multiple stages and solving them step by step using the optimal substructure property, the global optimal solution is obtained [5]. By defining optimal strategies, this method can be applied to various types of problems, including discrete deterministic, discrete stochastic, infinite stochastic, and continuous deterministic problems [6]. For the order matching problem of the service platform, the most important thing is to match the most suitable orders with the most suitable service personnel. Compared with other algorithms, dynamic programming can efficiently and quickly obtain the most suitable assignment scheme, thus achieving good economic benefits.

This paper explores how the DP algorithm can be applied to the order matching problem of the domestic service platform to achieve the perfect match between the order and the domestic service personnel. The main contributions of this paper are as follows:

(1) Based on DP, with the goal of maximizing the total weighted value, the order assignment models of offline and online domestic service were established.

(2) We studied the effect of order backlog on the total weighted value of online order assignment and obtained the optimal order backlog threshold with the goal of maximizing the total weighted value.

The rest of the paper is organized as follows: Section 2 is a review of the research on order assignment and a summary of the existing problems, challenges, and the work finished in this paper; Section 3 elaborates on the data sources and data processing; Section 4 details the methods used for establishing of the optimization model for order assignment; Section 5 is the results analysis and discussion; finally, Section 6 concludes the full text by summarizing the findings and looking ahead to future research directions.

2. Related Works

Many scholars have conducted research on order assignment. In 2012, based on the cumulative prospect theory, Liu et al. [7] established the subjective utility function of a functional logistics service provider (FLSP) in the assignment of two orders. Considering different FLSP states, a two-stage sequential assignment model for two stages in the logistics service supply chain (LSSC) was established. In 2013, Liu et al. [8] established an LSSC time scheduling model constrained by service order time requirements, with the optimal objectives of total cost, completion time, and satisfaction of functional logistics service providers (FLSPs). The effects of the relationship cost coefficient and delay coefficient on the overall performance of LSSC were discussed. In 2015, Gang et al. [9] used random and fuzzy random variables to model uncertain parameters of construction companies and suppliers. Based on the Kuhn–Tucker condition, segmented genetic algorithm, and fuzzy random simulation, a two-layer model was proposed. In 2016, Hu et al. [10] proposed a mixed integer programming model to select suppliers based on three criteria: quality, delivery performance, and procurement cost. The three objectives have different weights,

which are calculated using the fuzzy analytic hierarchy process (FAHP) method, and orders are allocated based on the weight values.

In 2019, Mao et al. [2] studied the impact of delivery performance on future customer orders for an on-demand delivery service platform, identified the factors affecting delivery performance, and explained how to develop an “order allocation strategy” that can help the platform increase future customer orders. Fan et al. [11] proposed a multi-objective integer nonlinear programming (INLP) formula to solve the problem of multi-product, multi-cycle, and multi-order allocation while considering the product defect rate and carbon emissions. Based on priority encoding, an improved genetic algorithm (IGAUP) was proposed, which can improve the efficiency and effectiveness in solving large instances within a given time limit.

In 2020, Baek et al. [12] proposed a nonlinear integer programming model and developed two different algorithms to improve the usability of the model in real business with large amounts of data. Kim et al. [13] considered an order batch matching problem (OLMP) to minimize the total delay of orders with different expiration dates, which is necessary not only to determine the batch allocation in the production facility, but also to generate a batch release plan within a given time period. Based on the DP model with dominant conditions, the advantage condition of OLMP has been proposed. Sun et al. [14] established an optimization model for the problem using cumulative time with the goal of minimizing the total delivery cost and added a time dimension to the order allocation and path optimization dimensions. The main algorithm is a dynamic allocation algorithm designed from the perspective of scheduling efficiency, and the sub-algorithm is an improved genetic algorithm. Shavaki et al. [15] proposed a rule-based heuristic algorithm that integrates order batch processing decisions, batch selection plans, truck allocation orders, and truck scheduling and routing.

In 2021, Du et al. [1] considered the constraints of customer expected delivery time and vehicle condition and used three soft time windows to combine vehicle routing models with different arrival times, and then used genetic algorithms to solve them. Zou et al. [3] proposed a reinforcement learning framework based on a double deep Q-network (DQN), which gradually tests and learns the order scheduling strategy by communicating with the online to offline (O2O) simulation model developed by the simulation of urban mobility (SUMO). The results show that this framework has good applicability for achieving efficient order scheduling in situations of huge transaction volume and computational complexity of delivery routes. Beiki et al. [16] proposed a method that combines the language entropy weight method (LEWM) and multi-objective programming (MOP) to solve sustainable supplier selection and order assignment problems. Deng et al. [17] established a two-layer multi-follower programming model, where the upper-layer model solves the seat allocation problem for all trains serving multiple origins and destinations (ODs) throughout the reservation layer, while the lower-layer model optimizes the price decision for each train serving each OD at different decision periods, and then proposed an effective solution method based on a divide and conquer strategy. Wu et al. [18] proposed a hybrid intelligent algorithm that uses the water wave optimization (WWO) metaheuristic method to evolve the solution of the main order assignment problem, and uses tabu search to optimize the path of each order selection solution, which can effectively solve the problem of integrated order assignment and delivery path optimization for delivery personnel.

In 2022, Lakshmanpriya et al. [19] introduced a comprehensive model including multiple products and time cycles to determine the demand based on green standards and allocate orders to the best suppliers, in order to increase the total procurement value while reducing the total procurement cost. Jiang et al. [20] proposed a network freight order allocation mechanism under carbon tax constraint and established an optimization model for order allocation under carbon tax constraint, which was solved using a 0–1 integer programming algorithm, and the order allocation arrangement for online freight platforms under carbon tax constraints was more economical and environmentally friendly. Su et al. [21] used the incremental nonlinear control assignment (INCA) method and the

active set quadratic programming algorithm to accurately allocate the virtual control commands, designed a group of incremental normalized objective functions, and combined these functions with the linear weighted sum method to construct an incremental multi-objective function. Using blockchain technology, Sun et al. [4] proposed a new double order allocation method driven by repurchase motivation. Multiple platforms are integrated into a single platform to make allocation decisions for different consumer groups. In the case of high repurchase motivation driven (HRMD) and medium repurchase motivation driven (MRMD) methods, an improved linear time complexity simulated annealing (SA) algorithm was used to solve the NP hard assignment problem. Liu et al. [22] proposed a fuzzy particle swarm optimization (FPSO) algorithm, which uses fuzzy logic reasoning to process domain knowledge, improve the solution quality, and obtain the optimal allocation scheme. By comparing with other algorithms, three different examples were provided to demonstrate the effectiveness of the proposed model and algorithm. Baroto et al. [23] proposed analytic hierarchy process (AHP), simple phase weighting (SAW), and goal planning methods to solve the green supplier selection and order allocation (GSSOA) problem. Five standards and thirteen secondary standards were applicable to plastics manufacturing companies. Yin et al. [24] described the passenger ticket purchase process by using a compound inhomogeneous Poisson point process, and then used the sparse method to simulate the passenger ticket purchase demand in the early stage of ticket sales. With ticket price and distribution as decision variables, they established a stochastic NP mathematical model with the goal of maximizing train revenue, and then designed a particle swarm optimization algorithm to solve the model.

In 2023, Yang et al. [25] proposed a real-time shared parking space allocation model and a timed shared parking space allocation model based on time window constraints. The results show that the real-time allocation mode reflects the advantage of the optimal allocation mode when the supply exceeds the demand for users; On the contrary, the fixed time allocation model has the advantage of optimal system resource utilization efficiency. Wang et al. [26] proposed two resource allocation algorithms for multi-satellite systems based on beam-hopping technology. For offline scenarios, a heuristic algorithm was proposed to allocate time and frequency resources, and a successive convex approximation (SCA) algorithm was proposed to allocate power resources. For online scenarios, a DP algorithm is used to maximize the system throughput. Wang et al. [27] established a mixed integer programming (MIP) model to study the joint optimization problem of dynamic bed allocation and patient admission control. Based on biogeography, a dynamic bed and patient admission optimization (BBO-DBPA) algorithm was proposed to obtain the optimal decision plan. Simoni et al. [28] proposed a sequential batch processing and allocation algorithm that uses graph-based methods to decompose the original problem into more manageable sub-problems and uses clustering, which effectively solves the problem of the same courier being able to process multiple orders on multiple pickup routes.

In 2017, Aboali et al. [29] compared the DP and basic block matching (BBM) methods in stereo matching algorithms from three aspects: parallax map accuracy, noise enhancement, and smoothness. The results show that both methods can be used for many stereo vision applications. In 2018, Sun et al. [30] proposed a DP method based on the reduced state space algorithm (RSS-DP), which greatly reduces the state space of the standard dynamic programming (SDP) model by decomposing the number of surplus by-product gases (SBPGs) into references and subsequent allocations, thereby greatly reducing the computational time. Jiang et al. [31] aimed at the problem of poor adaptability of traditional control strategies to different driving cycles and adopted a controller based on DP to optimize the power distribution of hybrid energy storage systems (HESSs) offline under several typical working conditions. The offline sample data of DP was processed using a combination of correlation analysis and the mean impact value (MIV) analysis method to obtain the characteristic variables of the extreme learning machine (ELM) model. At the same time, ELM is used to learn the offline data of HESS and to obtain the instantaneous power allocation strategy of HESS. In 2021, Taheri et al. [32] proposed an optimal strategy

based on DP, which takes the generalized forces and torques as the control and adopts the decoupling position and attitude dynamics calculation, which effectively controls the spacecraft when the thruster fails, and reconfigures the DP strategy and control allocation. Based on the correct answers submitted by students, Razali et al. [33] developed an online semantic error automatic feedback system for programming using a dynamic template matching model. The results show that 80% of the best matching templates for each student are selected by experts using this technology. In 2023, Fu et al. [34] proposed a multi-frame target detection method based on a dynamic programming ring network (DPRN). This method approximates the target trajectory using piecewise linear functions and uses velocity space partitioning DP (VSP-DP) to accumulate the advantage function of the target on each piecewise linear trajectory segment to avoid the diffusion of the advantage function in different velocity spaces. The results indicate that this method is suitable for radar and infrared search and tracking systems. Hou et al. [35] proposed a dynamic allocation mechanism that divides annual allocation plans into monthly scales and dynamically modifies the allocation plan to adapt to the aforementioned fluctuations. The results show that the coupling of the overall allocation plan and the monthly allocation plan of water resources is more in line with the management requirements. Fu et al. [36] proposed a calculation method for the total transfer capacity (TTC) of the grid including PST and wind power. The optimal planning path of the phase shifting transformer (PST) is derived based on the DP method, and the final results show that this method has good optimization for PST allocation in each planning stage of the power system. Khan et al. [37] proposed a method to solve linear programming (LP) problems with fuzzy left and right sides, and modified LP's dynamic programming technology. The results show that the method can optimize the fuzzy linear programming problem. He et al. [38] used adaptive dynamic programming (ADP) technology to propose an attitude tracking scheme for a super-actuated tailless unmanned aerial vehicle (UAV) with both nonlinear and nonaffine control inputs. This method uses nonlinear dynamic inversion to construct an augmented system and uses a discounted performance function to transform the optimal tracking problem into an optimal adjustment problem.

The order assignment problem is an optimization problem that can be solved by the DP algorithm. In 2019, Duan et al. [39] proposed a new method to optimize the order scheduling system by combining the system assignment method with the driver capture method. In this method, the passenger demand is broadcast to the drivers in the selected scheduling area of the system, and an optimization problem is developed to determine the growth rate of the scheduling area by considering the idle driving distance of the drivers and waiting time of the passengers. The results show that the assignment effect of this method effectively shortens the expected pickup distance of drivers, maintains a short scheduling time, and balances the interests of passengers and drivers. In 2021, Czerniachowska et al. [40] proposed a DP algorithm for the allocating retail shelf space to maximize shelf profits. The results show that this algorithm can solve this problem with less time and computational resources. In 2022, Suo et al. [41] combined fuzzy interval linear programming (FILP) with DP and proposed a fuzzy interval dynamic programming (FIDP) model for regional water resource management under uncertain conditions. The results show that the solution of this model can provide detailed allocation plans and water scarcity rates at different stages. Gong et al. [42] built a mathematical model for optimal water allocation of a single main canal in large irrigation areas, and then solved the model by using the one-dimensional DP method to calculate the minimum water deficit of each branch canal and the corresponding optimal water allocation. The results show that the corresponding water allocation of each branch canal reflects the accuracy and efficiency of a forced model solution. Amuji et al. [43] developed two models, namely a DP model and an optimal allocation strategy model, for optimal allocation of outbound loading container units with dimensions of 20 ft and 40 ft to six major seaports in Nigeria. The results show that the allocation results obtained by the model are highly optimized. Djurdjević et al. [44] combined DP and simulation modeling methods and proposed a method to define the optimal size and product

allocation in the order selection area of the most typical order selection settings. The results show that this method can effectively solve the optimal configuration of products in picking areas of different scales. Wei et al. [45] proposed a large system decomposition aggregation dynamic programming (DADP) method for the optimal allocation of irrigation water for multi-water source projects in water-scarce areas, which transformed the $N + 1$ -dimensional dynamic programming problem into the $N + 1$ dimensional DP problem for the solution. The results indicate that this method has good applicability.

As mentioned above, research on order allocation has made some progress in recent years. Many scholars have proposed various allocation models, such as models based on cumulative prospect theory, integer programming, heuristic algorithms, and genetic algorithms. These studies aim to improve the efficiency of order assignment, reduce the cost, and optimize the customer experience and the utilization of service housekeepers.

However, the problem of order assignment still faces several challenges. On the one hand, as the number of orders increases, order assignment problems become more complex and large-scale, and finding efficient algorithms to solve large-scale order assignment problems remains a challenge; on the other hand, the actual order assignment problem usually involves multiple objectives and constraints, such as cost, delivery time, customer satisfaction, etc. How to balance multiple objectives and satisfy various constraints is a complex problem.

In this paper, we will provide a detailed introduction to the establishment of offline and online delivery models and analyze the optimization results. Finally, through the research in this article, a feasible solution for order allocation is provided, and guidance is given for applications in related fields.

3. Data Processing

The data in this article are real data provided by the “58 City” platform and the “MathorCup” University Mathematical Modeling Competition [46]. The data include information on all orders received in a given area within one day (including order identification, ordering time, earliest service start time, latest service start time, service duration, and distance coordinates of service locations), as well as information on all housekeepers (housekeeper identification, housekeeper service scores, initial location coordinates). The data description is shown in Tables 1 and 2.

Table 1. Description of order data.

Name	Type	Description
id_order	int	Order identification
createTime	int	Order time/s
serviceFirstTime	int	Earliest service start time/s
serviceLastTime	int	Latest service start time/s
serviceUnitTime	int	Service duration/min
x	int	Horizontal coordinate of service location/m
y	int	Vertical coordinate of service location/m

Table 2. Description of housekeeper data.

Name	Type	Description
id_housekeeper	int	Housekeeper identification
serviceScore	double	Housekeeper score
x	int	Horizontal coordinate of housekeeper/m
y	int	Vertical coordinate of housekeeper/m

Example:

```
id_order, createTime, serviceFirstTime, serviceLastTime, serviceUnitTime, x, y
0, 1661422500, 1662786000, 1662789600, 90, -4693, -12716
1, 1661474400, 1662768000, 1662771600, 60, 2176, -17371
2, 1661568300, 1662771600, 1662775200, 60, 3940, -11414
3, 1661575800, 1662789600, 1662793200, 90, 1791, -16026
```

Among them, “1661422500” is a timestamp, which is a sequence of characters or encoded information used to identify when a particular event occurred, usually giving the date and time, sometimes to the nearest fraction of a second.

Example:

```
id_housekeeper, serviceScore, x, y
0, 0.9998, 931, -31576
1, 0.9993, 7529, -7145
2, 0.9992, 5393, 2140
3, 0.999, 5982, -3340
```

The numbers of order and housekeeper are 2304 and 2795, respectively. Let us process the order data first:

(1) Based on the earliest service start time, sort the order data in ascending order and set the earliest service start time as 0, as shown in Table 3, where the interval time is calculated in hours.

Table 3. Processed order data.

id_order	serviceFirstTime/h	serviceLastTime/h	serviceUnitTime/h
0	5	6	1.5
1	0	1	1
2	1	2	1
3	6	7	1.5
4	9	10	1.5
5	5	6	1.5
...
2299	12	13	1
2300	13	14	1
2301	13	14	1
2302	13	14	1
2303	13	13	1

(2) Sort the housekeepers in descending order of their service scores to ensure that housekeepers with higher service scores are given priority when assigning orders.

4. Method

The basic requirements for order assignment are as follows: (1) All orders must be assigned, and each order can be handled by only one housekeeper. (2) The starting time of the order service should be between the earliest time and the latest time and an integral multiple of half an hour. (3) A housekeeper may serve only one order at a time and must arrive at the client’s location prior to the service start time. (4) The housekeeper must start from the starting point position when accepting the order, with a speed set to 15 km/h. (5) Use Euclidean distance [47] to calculate the distance between two points.

There are two modes to consider when sending orders to housekeepers: order matching housekeeper and housekeeper matching order. In order to fully utilize the service capabilities of housekeepers, improve their work efficiency, reduce travel time, and improve service quality, this paper uses order matching housekeeper as the matching mode.

Given the order time, service time, and service location of all orders within a day, as well as the service scores and coordinates of all housekeepers, the weighted value of the i -th housekeeper completing the k -th order is $S(i, k)$, and the dispatch mode is evaluated based on the housekeeper’s total weighted value.

In this paper, offline and online assignment models are discussed, respectively. In the online allocation mode, a fixed allocation frequency is often used, and new orders generated during this period are evenly distributed every 30 min. During allocation, some orders are allowed to temporarily not be dispatched (backlog), but the service start time for backlog orders must be 2 h later than the current time (excluding 2 h). Meanwhile, for customers, the experience of backlog is not very good. Therefore, we add a threshold γ in the constraint to ensure that the total number of backlog orders cannot exceed this threshold.

4.1. Offline Assignment Model

Each order must be matched with a domestic helper and solved by using single objective nonlinear programming.

(1) Objective function

The goal of this model is to maximize the total weighted score Sum , as follows:

$$\text{Max } Sum = \sum_{i=0}^{2303} S(i, k) \quad (1)$$

(2) Constraints

Each order must be assigned one and only one housekeeper, assuming that $order_k$ ($k = 0, 1, \dots, 2303$) is a 0–1 variable representing the order status, as follows:

$$order_k = \begin{cases} 0, & \text{matched} \\ 1, & \text{unmatched} \end{cases} \quad (2)$$

Each order must specify a service start time, which must be within the range of [earliest start time, latest start time] and be an integer multiple of half an hour, as follows:

$$t_k = \{t_{\min}, t_{\min} + 0.5, \dots, t_{\max}\} \quad (3)$$

where t_k is the service start time of the k -th order, t_{\min} is the earliest start time, and t_{\max} is the latest start time; A housekeeper can only serve one order at the same time. Assume that $housekeeper_i$ ($i = 0, 1, \dots, 2794$) is a 0–1 variable representing the housekeeper status, as follows:

$$housekeeper_i = \begin{cases} 0, & \text{free} \\ 1, & \text{working} \end{cases} \quad (4)$$

The coordinates of the order marked with k are represented by (x_k, y_k) , and the coordinates of the housekeeper marked with i are represented by (x_i, y_i) ; thus, the Euclidean distance d can be expressed as follows [47]:

$$d = \sqrt{(x_i - x_k)^2 + (y_i - y_k)^2} \quad (5)$$

The housekeeper's arrival time $gotime_i$ must be earlier than the order service's start time t_i , as follows:

$$\begin{cases} gotime_i \leq t_i \\ gotime_i = historytime + d/v \end{cases} \quad (6)$$

where $historytime$ is Beijing time, and the housekeeper's moving speed is 15 km/h. Assume that the i -th housekeeper score is weighted $S(i, k)$ after completing the k -th order, as follows:

$$S(i, k) = \alpha A_i + \beta B_k + \gamma C_k \quad (7)$$

where A_i is the service score of the i -th housekeeper, B_k is the travel distance of the k -th order, and C_k is the time interval of the k -th order, with a coefficient $\alpha = 0.78$, $\beta = 0.025$, $\gamma = 0.195$.

(3) Final optimization model

The final objective function refers to Equation (1). The decision variable is housekeeper number i . All constraints are as follows:

$$\left\{ \begin{array}{l} order_k = \begin{cases} 0, \text{ matched} \\ 1, \text{ unmatched} \end{cases} \\ housekeeper_i = \begin{cases} 0, \text{ free} \\ 1, \text{ working} \end{cases} \\ t_k = \{t_{\min}, t_{\min} + 0.5, \dots, t_{\max}\} \\ d = \sqrt{(x_i - x_k)^2 + (y_i - y_k)^2} \\ v = 15\text{km/h} \\ gotime_i = historytime + d/v \\ gotime_i \leq t_i \\ S(i, k) = \alpha A_i + \beta B_k + \gamma C_k \end{array} \right. \quad (8)$$

Finally, with the goal of making *Sum* as large as possible, and the number of housekeepers as the decision variable, a single objective nonlinear programming model is established with the status of the order and the housekeeper, the order time limit, and the movement trajectory as constraints.

4.2. Online Assignment Model

Compared to the offline model, the constraints and decision variables in the online model have changed. Define the decision condition *op* for order backlog and determine whether to dispatch orders after every half point. If *op* is set to 1, the order is backlogged, and if *op* is set to 0, the order is allocated as follows:

$$op = \begin{cases} 0, t_{\max} - historytime \leq 2 \\ 1, t_{\max} - historytime > 2 \end{cases} \quad (9)$$

In addition, study the impact of different order backlog thresholds on the model, and adjust the constraint values during the solving process. Use the traversal search method to find the optimal threshold y from 0 to 100%, as follows:

$$\left\{ \begin{array}{l} \sum_{k=0}^{2303} order_k \cdot op = 2304 \times y \\ 0 \leq y \leq 1 \end{array} \right. \quad (10)$$

Other constraints are the same as in the offline mode. The final objective function refers to Equation (1). The decision variable refers to Equation (9). All constraints are listed as follows:

$$\left\{ \begin{array}{l} order_k = \begin{cases} 0, \text{ matched} \\ 1, \text{ unmatched} \end{cases} \\ t_k = \{t_{\min}, t_{\min} + 0.5, \dots, t_{\max}\} \\ d = \sqrt{(x_i - x_k)^2 + (y_i - y_k)^2} \\ v = 15\text{km/h} \\ housekeeper_i = \begin{cases} 0, \text{ free} \\ 1, \text{ working} \end{cases} \\ gotime_i = historytime + d/v \\ \sum_{k=0}^{2303} order_k \cdot op = 2304 \times y \\ op = \begin{cases} 0, t_{\max} - historytime \leq 2 \\ 1, t_{\max} - historytime > 2 \end{cases} \\ gotime_i \leq t_i \\ 0 \leq y \leq 1 \\ S(i, k) = \alpha A_i + \beta B_k + \gamma C_k \end{array} \right. \quad (11)$$

By performing the traversal search method on the model, the threshold value under the maximum weight value and the optimal order assignment can be found. By defining the daily work time utilization rate a and the choiceness rate b , the effectiveness of the algorithm is evaluated. Firstly, it is assumed that the daily working time is 10 h (if the housekeepers work more than 10 h, it is considered as overtime), and the working time of the i -th housekeeper in the daily work is $workingtime_i$. The service score of the i -th housekeeper in the daily work is $serviceScore_i$, the total number of orders taken by housekeepers is $orderquantity_i$, and the number of housekeepers who have received orders is m . The daily working time utilization rate a is the ratio of the total working time of each housekeeper to the total standard working time of each housekeeper, as follows:

$$a = \frac{\sum_{i=1}^m workingtime_i}{m \times 10} \quad (12)$$

The choiceness rate b of the housekeeper is the ratio between the sum of housekeeper service score $serviceScore_i$ multiplied by the housekeeper order number $orderquantity_i$, and the sum of housekeeper order number $orderquantity_i$ multiplied by the maximum service score of the housekeeper m , as follows:

$$b = \frac{\sum_{i=1}^m serviceScore_i \times orderquantity_i}{\sum_{i=1}^m orderquantity_i \times serviceScore_{\max}} \quad (13)$$

4.3. DP Algorithm for Order Assignment

Based on the mathematical model established in the previous section, the design of the algorithm is also divided into two parts: offline ordering and online ordering.

4.3.1. Algorithm for the Offline Model

For the offline order allocation model, the main processes are as follows. Firstly, all offline order data are arranged in chronological order. Then, the order is selected according to the arranged order, the housekeeper with the highest weight is searched through the loop, the location information and service time of the housekeeper are stored, and the order weight value is added to Sum . The order is processed until all offline orders are found and the value of Sum is finally output. The detailed steps are shown in Figure 1.

The specific algorithm steps in Figure 1 are as follows:

- (1) Input the housekeeper data and order data processed by time sorting, and set the weighted value $S(i, k) = \alpha A_i - \beta B_k - \gamma C_k$, $k, i \in N^+$.
- (2) Sort the orders by order and traverse the housekeeper data. Find $\text{Max } S(i, k)$ to receive this order, delete the order, and save $\text{Max } S(i, k)$ to Sum at the same time.
- (3) Repeat Step 2 continuously until all orders are assigned and calculate the update Sum .

4.3.2. Algorithm for the Online Model

For the online order allocation model, the algorithmic flow is as follows. First, the orders are sorted by time. Then, the algorithm collects a batch of orders every 30 min, and the newly generated online orders are collected and processed so that enough order information can be accumulated so that the domestic service staff with higher service quality can be selected more effectively when the distribution is performed.

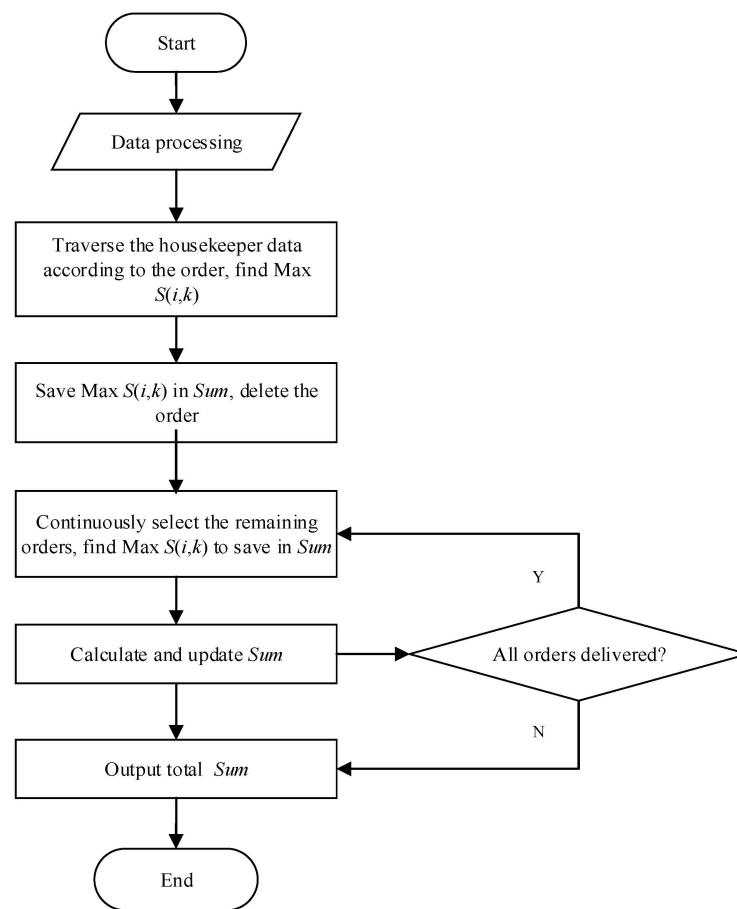


Figure 1. Flowchart of the dynamic programming algorithm used to assign offline orders.

In addition, to control the number of backlogged orders, the model sets a threshold γ that limits the number of backlogged orders to not exceed a certain percentage of the total number of orders. As a result, orders are assigned only in two cases: when the backlog reaches a certain threshold, or when some orders must be assigned immediately due to an imminent need for service.

So, after collecting data every 30 min, the model goes through the housekeeper data for each current order and selects a housekeeper with the highest weighted value to take the order to ensure that the customers can receive the appropriate service. After the housekeeper takes the order, the housekeeper's location information and service time are updated in the dataset for use in future order assignments, and the order is removed from the list to be assigned. As new orders continue to arrive, the algorithm continues to perform the previous collection and backlog steps, distributing work as conditions are met. The entire process continues until all orders have been properly processed.

Finally, the sum of the weighted values of all the housekeepers obtained, i.e., the Sum value, is output as an important indicator of the efficiency of job distribution. The above is an overview of the core logic and operational steps of the online order assignment model. For specific implementation details, please refer to the algorithm flowchart shown in Figure 2.

The specific algorithm steps in Figure 2 are as follows:

(1) Input the housekeeper data and order data processed by time sorting, and set the weighted value $S(i, k) = \alpha A_i - \beta B_k - \gamma C_k$, $k, i \in N^+$.

(2) After 30 min, the clients create a certain number of orders. At this point, check if there are any orders that were created earlier than their start time. If not, execute Step 3; otherwise, continue to determine whether the number of backlog orders is greater than γ of

the total orders. If yes, perform Step 3, or the order will be backlogged, and wait for the next 30 min.

(3) Sort the orders in order and traverse the housekeeper data. Find $\text{Max } S(i, k)$ to receive this order, delete the order, and save $\text{Max } S(i, k)$ to Sum at the same time. Check if all orders are assigned. If not, perform Step 2 and proceed to the next 30 min; otherwise, perform Step 4.

(4) Calculate the updated Sum .

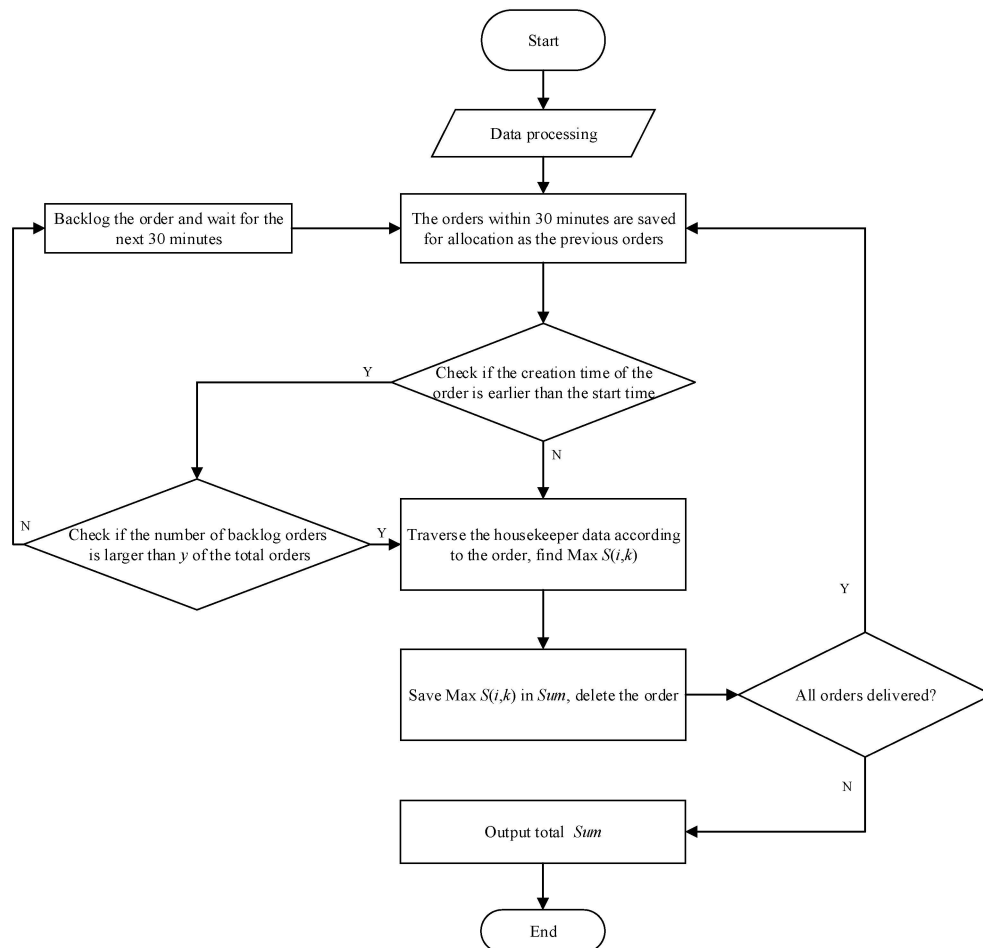


Figure 2. Flowchart of the dynamic programming algorithm used to assign online orders.

5. Results and Discussion

In this part, we show the results of online and offline ordering and use a case to verify the model.

5.1. Results of the Offline Model

Using MATLAB for programming and solving, the order allocation in offline mode is obtained, and some of the results are shown in Table 4:

Table 4. Results of the offline order assignment.

id_order	serviceStartTime	id_housekeeper
0	1662787800	40
1	1662768000	61
...
1189	1662778800	308
1190	1662786000	96
...
2303	1662814800	161

As can be seen, id_order 0 is received by id_housekeeper 40, id_order 1 is received by id_housekeeper 61, id_order 1189 is received by id_housekeeper 308, and id_order 1190 is received by id_housekeeper 96. Finally, the maximum total weight value *Sum* is found to be 1412.93.

5.2. Results of the Online Model

Perform a traversal search on the threshold y . The results are shown in Figure 3:

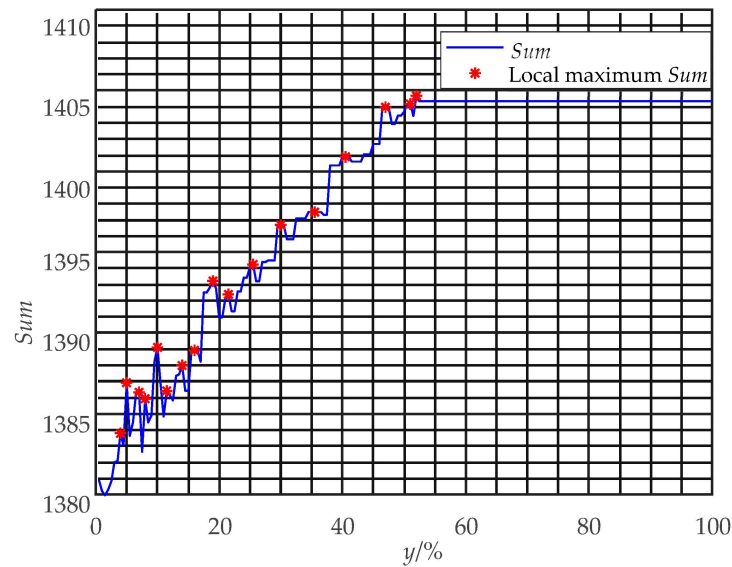


Figure 3. Solution of the optimal order backlog threshold y .

Divide Figure 3 into intervals to represent the corresponding relationship between the local maximum *Sum* and y , as shown in Table 5:

Table 5. Local maximum *Sum* and corresponding threshold y .

Interval	Local Maximum <i>Sum</i>	$y/\%$
[3.5%, 4.5%)	1384.26	4.0%
[4.5%, 5.5%)	1387.38	5.0%
[5.5%, 7.5%)	1386.79	7.0%
[7.5%, 8.5%)	1386.40	8.0%
[8.5%, 11.0%)	1389.56	10.0%
[11.0%, 12.5%)	1386.88	11.5%
[12.5%, 15.0%)	1388.44	14.0%
[15.0%, 17.0%)	1389.38	16.0%
[17.0%, 20.5%)	1393.68	19.0%
[20.5%, 22.5%)	1392.85	21.5%
[22.5%, 26.0%)	1394.70	25.5%
[26.0%, 31.5%)	1397.15	30.0%
[31.5%, 37.5%)	1397.94	35.5%
[37.5%, 42.0%)	1401.36	40.5%
[42.0%, 48.0%)	1404.43	47.0%
[48.0%, 51.5%)	1404.63	51.0%
[51.5%, 53.0%)	1405.14	52.0%

When $y = 52\%$, Sum obtained is the highest and the allocation scheme is the best. When the order quantity and efficiency change, the maximum Sum can be obtained by selecting y corresponding to the local maximum Sum within different values of y . Using DP, the order assignments of the online model are shown in Table 6.

Table 6. Results of online order assignments.

id_order	serviceStartTime	id_housekeeper
0	1662789600	40
1	1662768000	61
...
1189	1662778800	308
1190	1662782400	231
...
2303	1662814800	664

As can be seen, id_order 0 is received by id_housekeeper 40 with the serviceStartTime is 1662789600; id_order 1 is received by id_housekeeper 61 with the serviceStartTime is 1662768000; id_order 1189 is received by id_housekeeper 308 with the serviceStartTime is 1662778800. Finally, the maximum total weight value Sum is found to be 1,405.14. Results of the order backlog are shown in Table 7. Id_order 0 is backlogged, (serviceStartTime -1 , id_housekeeper -1 and retainable 1), and id_order 2032 has been assigned.

Table 7. Results of the order backlog.

currentTime	id_order	serviceStartTime	id_housekeeper	Retainable
1661423400	0	-1	-1	1
1661425200	0	-1	-1	1
...
1662807600	2302	1662818400	106	0
...

According to the calculation results based on this reference data, the number of housekeepers who took orders on the day, m , was 709, and the daily work time utilization rate, a , was 56.19%. According to the housekeepers' data, the number of people who participated in the service score of more than 0.9 on the day reached 516. Accounting for 72.78% of the total number of housekeepers working on the day, the housekeeper choiceness rate b is 95.22%. The higher the value of b , the more orders the housekeeper with higher service scores will receive, which will improve the overall service quality of the platform.

5.3. Verification

Using the offline assignment model, we re-run the data from the top 50 orders and the top 20 housekeepers, and obtain the execution task list of housekeepers, as shown in Table 8:

Table 8. Execution task list for the housekeeper based on the offline model.

id_housekeeper	Execution Order
0	34 1662786000
1	39→5 (1662768000→1662773400)
2	44→10→21 (1662784200→1662796800→1662802200)
3	32→20→38→31 (1662771600→1662778800→1662784200→1662795000)
4	13→46→22→18→43 (1662771600→1662777000→1662782400→1662793200→1662798600)
5	17→45 (1662768000→1662777000)
6	24→48 (1662768000→1662773400)
7	2 (1662771600)
8	8→29→33→23→37 (1662771600→1662778800→1662784200→1662793200→1662798600)
9	47→15→25→36→49 (1662771600→1662778800→1662784200→1662789600→1662798600)
10	30→9→0→3→16→40 (1662768000→1662777000→1662786000→1662793200→1662800400→1662807600)
11	41 (1662800400)
12	14 (1662782400)
13	35 (1662768000)
14	11 (1662768000)
17	1→42 (1662768000→16627734000)
18	12→28→26→27 (1662771600→1662778800→1662787800→1662795000)
19	7→6 (1662768000→1662773400)

It can be seen that id_housekeeper 0 is assigned to id_order 34, id_housekeeper 1 is assigned to id_orders 5 and 39, id_housekeeper 2 is assigned to id_orders 10, 21, and 44, and id_housekeeper 3 is assigned to id_orders 20, 31, 32, and 38. Figure 4 shows the trajectory of the housekeepers. Finally, the total weighted value *Sum* is 24.96.

5.4. Result Discussion

Based on DP, this paper constructs the order assignment models of offline and online dispatch. The results show that the maximum weighted value of the offline model is 1412.93, while the online model is 1404.81. The result of the offline model is slightly larger than that of the online model, because the offline model predetermines the status of all orders and housekeepers throughout the day and searches for the optimal solution without being limited by the order release time. However, the online mode is limited by the order release time, which cannot predetermine the optimal solution for orders that have not been released.

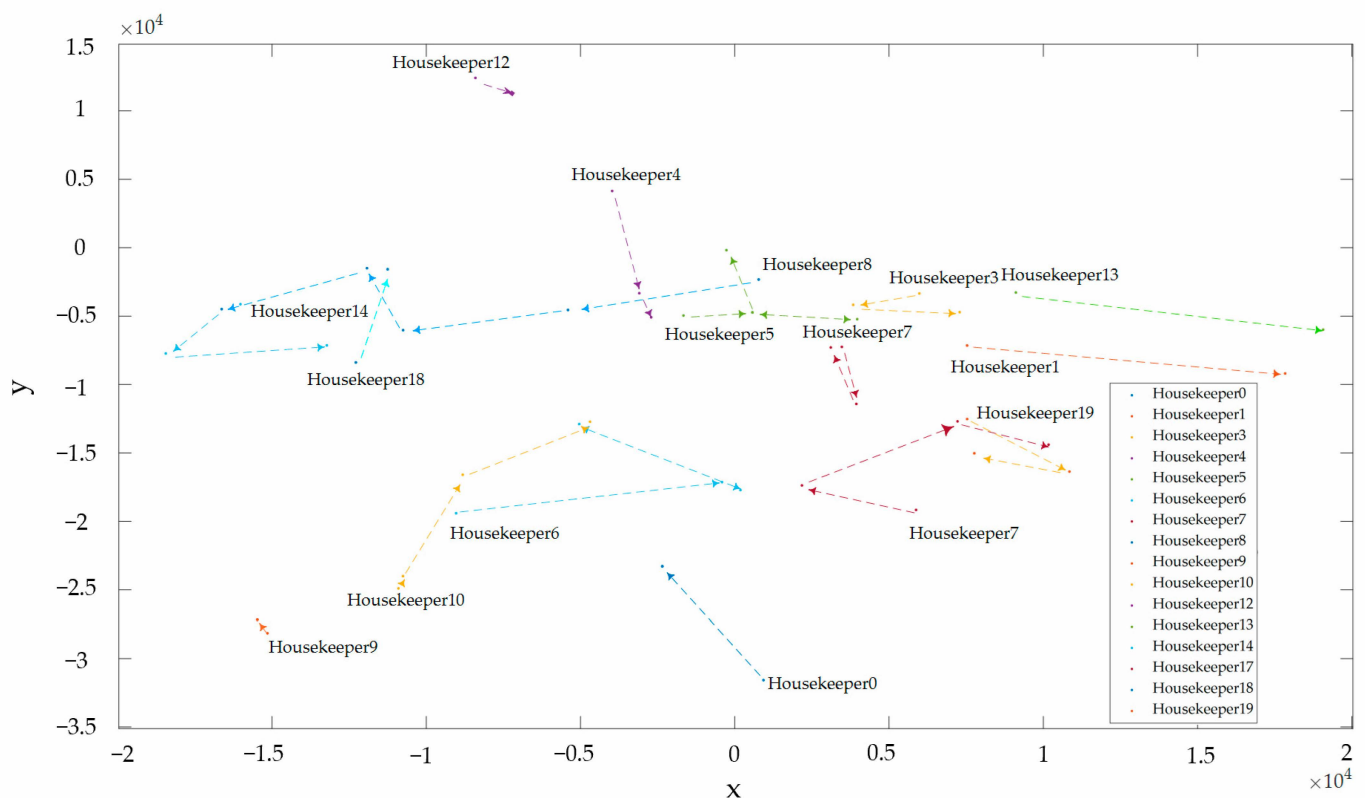


Figure 4. Trajectory of housekeepers.

In addition, the relationship between the backlog threshold and the total weighted value of online orders was investigated. The results show that when the backlog threshold is less than 52%, the total weighted value gradually increases with the increase in the backlog threshold; when the backlog threshold exceeds 52%, the increase in the total weighted value decreases. When the backlog threshold is 52%, the total weighted value reaches 1405.14. During the order allocation, the total weighted value can be optimized by changing the backlog threshold to improve the allocation efficiency. Next, we perform a sensitivity analysis and compare the online assignment model to another model.

5.4.1. Sensitivity Analysis

To verify the influence of relevant variables in the model on the online assignment model, the sensitivity analysis of the online order assignment model is performed below.

The housekeeper's moving speed may affect the time when the housekeeper goes to the service location, and the order update time may affect the order data set. Therefore, in this model, a sensitivity analysis is conducted on the two variables of housekeeper moving speed and order update time. To ensure that when the sensitivity analysis of one of the variables, the other variables remain unchanged, so we change the housekeeper's moving speed from 1 km/h to 60 km/h, observe the change in the total order weight value, and obtain the online order distribution model on the housekeeper's moving speed sensitivity analysis results, as shown in Figure 5.

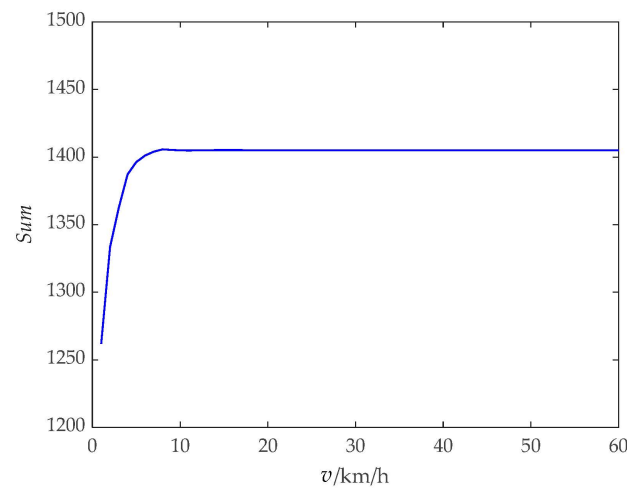


Figure 5. Sensitivity analysis of the online order assignment model for the housekeeper's moving speed.

It can be seen from Figure 5 that when the housekeeper's moving speed v is between 1 km/h and 10 km/h, the online order distribution model is sensitive to the housekeeper's moving speed. When the speed is between 10 km/h and 60 km/h, the total order weighting value Sum is stable. When the moving speed is low, the housekeeper cannot reach the service position in time, and the Sum value at this time increases with the moving speed of the housekeeper, but when the moving speed of the housekeeper increases to 10 km/h, the value of Sum no longer changes.

Then, the order update time is changed from 0.1 h to 2 h to observe the change in the total order weight, and the sensitivity analysis result of the online order assignment model to the order update time is obtained, as shown in Figure 6.

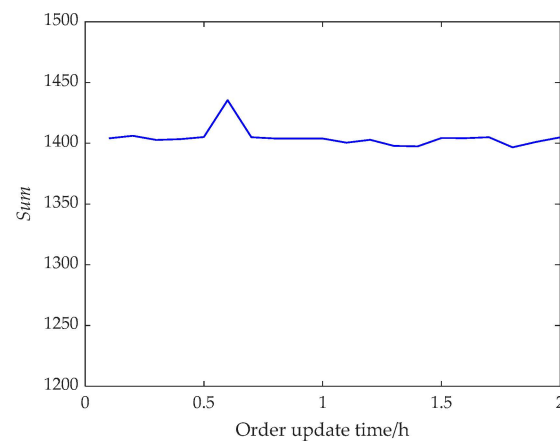


Figure 6. Sensitivity analysis results of online order assignment model to order update time.

As can be seen from the analysis in Figure 6, when the order update time is in the interval [0.1 h, 0.5 h] and [0.6 h, 2 h], the model is insensitive to it and the value of Sum fluctuates slightly. In the interval [0.5 h, 0.6 h], the value of Sum fluctuates strongly, and the model is sensitive to the order update time.

5.4.2. Comparison to the Nearest Distance Method

To validate the online assignment model, the nearest distance method is used as a comparison. The nearest distance method assumes that d_k is the distance between the k -th order and the corresponding housekeeper who receives the order. The process of this model is as follows.

Optimization function:

$$\text{Min } d_k = \sqrt{(x_i - x_k)^2 + (y_i - y_k)^2} \quad (14)$$

Decision variable refers to Equation (9). All constraints are as follows:

$$\left\{ \begin{array}{l} order_k = \begin{cases} 0, \text{ matched} \\ 1, \text{ unmatched} \end{cases} \\ t_k = \{t_{\min}, t_{\min} + 0.5, \dots, t_{\max}\} \\ v = 15\text{km/h} \\ housekeeper_i = \begin{cases} 0, \text{ free} \\ 1, \text{ working} \end{cases} \\ gotime_i = historytime + d/v \\ \sum_{k=0}^{2303} order_k \cdot op = 2304 \times y \\ op = \begin{cases} 0, t_{\max} - historytime \leq 2 \\ 1, t_{\max} - historytime > 2 \end{cases} \\ gotime_i \leq t_i \\ 0 \leq y \leq 1 \\ S(i, k) = \alpha A_i + \beta B_k + \gamma C_k \end{array} \right. \quad (15)$$

The core algorithm of the nearest distance assignment model is to assign orders to the nearest housekeeper to achieve fast service. This method can reduce the distance of the housekeeper's movement, but it ignores the importance of the housekeeper's service quality. The values of *Sum*, *m*, *a*, and *b* of the online assignment model are compared with those of the nearest distance assignment model in Table 9.

Table 9. Comparison with the nearest distance assignment model.

Model	<i>Sum</i>	<i>m</i>	<i>a</i>	<i>b</i>
Online assignment model	1405.14	709	56.19%	95.22%
Nearest distance assignment model	810.25	1200	28.85%	56.76%

Compared with the nearest distance assignment model, the value *Sum* of the online assignment model is larger, the overall service evaluation score is higher, and the platform service quality is better. On the same day, *m* is smaller, and the housekeepers with high service scores take more orders, while the housekeepers with low service scores take fewer or no orders. The values of *a* and *b* are larger, and the daily work time utilization rate and choiceness rate are higher. From the long-term development of the company, the higher the overall service score and the better the service quality, the better the development of the company will be. Therefore, compared with the closest distance assignment model, the online assignment model proposed in this paper is superior.

Overall, both offline and online delivery models have their own advantages and disadvantages, and the choice of which model to use depends on the optimization objectives and acceptable constraints. The offline model can obtain the overall optimal solution, but it requires a comprehensive understanding of the entire day's orders and housekeeper situation and may not handle unexpected situations properly. The online model takes into account time constraints and the clients' experience, which is more in line with the needs of the real situation and has better adaptability to unpredictable problems.

6. Conclusions

For order assignment, this paper establishes offline and online order assignment models based on dynamic programming, both of which can achieve optimal order assignment. For the online order assignment model, an order backlog mechanism is added to prevent

the algorithm from falling into the local optimization dilemma. The online assignment model has strong applicability and can be applied to other resource assignment problems.

In order to demonstrate the stability of the model, a sensitivity analysis is conducted. By analyzing the effects of the two variables, household travel speed and order update time, it is concluded that the household's moving speed is sensitive from 1 km/h to 10 km/h and insensitive from 10 km/h to 60 km/h. The order update time is insensitive in the range of [0.1 h, 0.5 h] and [0.6 h, 2 h], but sensitive in the range of [0.5 h, 0.6 h], thus proving the superiority and stability of the model proposed in this paper. Furthermore, the online order assignment model is compared with the nearest distance assignment model. The results show that the total weighted score of the nearest distance assignment model is 810.25, while the proposed online assignment model is 1405.14. The higher the total weighted score, the better the service quality of the company. Therefore, the online assignment model is superior to the nearest distance assignment model.

However, the order assignment models do not address practical issues such as temporary order cancellations and temporary leave requests by housekeepers, which require further research.

Author Contributions: Conceptualization, J.C.; methodology, J.C. and C.L.; software, Y.S. and J.L.; validation, Y.S. and J.S.; formal analysis, J.C. and J.S.; investigation, C.L. and J.L.; resources, J.C.; data curation, J.C. and Y.S.; writing—original draft preparation, J.C., J.S. and J.L.; writing—review and editing, J.C., Y.S. and C.L.; visualization, Y.S.; supervision, C.L., X.S. and J.C.; project administration, C.L. and J.C.; funding acquisition, X.S. and J.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (grant number: 52005435), the Natural Science Foundation of Jiangsu Province (grant number: BK20190873), the Undergraduate Education Reform Project of Yangzhou University (Special Funding for Mathematical Contest in Modeling) (grant number: xkjs2023049), and the Shuangchuang Program of Jiangsu Province of China (grant number: (2019)30756).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors on request.

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

References

1. Du, Y. Online Ordering Platform City Distribution Based on Genetic Algorithm. *arXiv* **2021**, arXiv:2106.11578.
2. Mao, W.; Ming, L.; Rong, Y.; Tang, C.S.; Zheng, H. *Faster Deliveries and Smarter Order Assignments for an On-Demand Meal Delivery Platform*; Social Science Electronic Publishing: Rochester, NY, USA, 2019.
3. Zou, G.; Tang, J.; Yilmaz, L.; Kong, X. Online food ordering delivery strategies based on deep reinforcement learning. *Appl. Intell.* **2021**, *52*, 6853–6865. [[CrossRef](#)]
4. Sun, Q.; Dong, M.; Tan, A. An order allocation methodology based on customer repurchase motivation drivers using blockchain technology. *Electron. Commer. Res. Appl.* **2022**, *56*, 101218. [[CrossRef](#)]
5. Bradley, S.P.; Hax, A.C.; Magnanti, T.L. *Applied Mathematical Programming*; Addison-Wesley Publishing Company: Boston, MA, USA, 1977.
6. Bellman, R. The theory of dynamic programming. *Bull. Am. Math. Soc.* **1954**, *60*, 503–515. [[CrossRef](#)]
7. Liu, W.; Yang, Y.; Li, X.; Xu, H.; Xie, D. A Time Scheduling Model of Logistics Service Supply Chain with Mass Customized Logistics Service. *Discret. Dyn. Nat. Soc.* **2012**, *2012*, 482978. [[CrossRef](#)]
8. Liu, W.; Liu, C.; Ge, M. An order allocation model for the two-echelon logistics service supply chain based on cumulative prospect theory. *J. Purch. Supply Manag.* **2013**, *19*, 39–48. [[CrossRef](#)]
9. Gang, J.; Tu, Y.; Xu, J. Multidemand Multisource Order Quantity Allocation with Multiple Transportation Alternatives. *Math. Probl. Eng.* **2015**, *2015*, 363056. [[CrossRef](#)]
10. Hu, H.; Xiong, H.; You, Y.; Yan, W. A Mixed Integer Programming Model for Supplier Selection and Order Allocation Problem with Fuzzy Multiobjective. *Sci. Program.* **2016**, *2016*, 9346781. [[CrossRef](#)]
11. Fan, Z.; Li, S.; Gao, Z. Multiobjective Sustainable Order Allocation Problem Optimization with Improved Genetic Algorithm Using Priority Encoding. *Math. Probl. Eng.* **2019**, *2019*, 8218709. [[CrossRef](#)]

12. Baek, S.H.; Kim, J.S. Efficient Algorithms for a Large-Scale Supplier Selection and Order Allocation Problem Considering Carbon Emissions and Quantity Discounts. *Mathematics* **2020**, *8*, 1659. [\[CrossRef\]](#)
13. Kim, J.-G.; Bang, J.-Y.; Jun, H.-B.; Shin, J.-H. Dominance Conditions for Optimal Order-Lot Matching in the Make-To-Order Production System. *Processes* **2020**, *8*, 255. [\[CrossRef\]](#)
14. Sun, G.; Tian, Z.; Liu, R.; Jing, Y.; Ma, Y. Research on Coordination and Optimization of Order Allocation and Delivery Route Planning in Take-Out System. *Math. Probl. Eng.* **2020**, *2020*, 7248492. [\[CrossRef\]](#)
15. Shavaki, F.H.N.; Jolai, F. A rule-based heuristic algorithm for joint order batching and delivery planning of online retailers with multiple order pickers. *Appl. Intell.* **2021**, *51*, 3917–3935. [\[CrossRef\]](#)
16. Beiki, H.; Seyedhosseini, S.M.; Ponkratov, V.V.; Zekiy, A.O.; Ivanov, S.A. Addressing a sustainable supplier selection and order allocation problem by an integrated approach: A case of automobile manufacturing. *J. Ind. Prod. Eng.* **2021**, *38*, 239–253. [\[CrossRef\]](#)
17. Deng, L.; Xu, J.; Zeng, N.; Hu, X. Optimization Problem of Pricing and Seat Allocation Based on Bilevel Multifollower Programming in High-Speed Railway. *J. Adv. Transp.* **2021**, *2021*, 5316574. [\[CrossRef\]](#)
18. Wu, J.-Y.; Zhang, M.-X.; Wu, X.; Zheng, Y.-J. A Water Wave Optimization Algorithm for Order Selection and Delivery Path Optimization for Takeaway Deliverymen. In Proceedings of the 2021 11th International Conference on Information Science and Technology (ICIST), Chengdu, China, 21–23 May 2021; pp. 620–628.
19. Lakshmanpriya, C.; Kumaravel, A.; Saravanan, M.; Kumar, P.M. Selecting the Optimal Green Supplier and Order Allocation under Linear Discount. *Math. Probl. Eng.* **2022**, *2022*, 2453703. [\[CrossRef\]](#)
20. Jiang, C.; Xu, J.; Li, S.; Zhang, X.; Wu, Y. The Order Allocation Problem and the Algorithm of Network Freight Platform under the Constraint of Carbon Tax Policy. *Int. J. Environ. Res. Public Health* **2022**, *19*, 10993. [\[CrossRef\]](#)
21. Su, M.; Hu, J.; Wang, Y.; He, Z.; Cong, J.; Han, L. A Multiobjective Incremental Control Allocation Strategy for Tailless Aircraft. *Int. J. Aerosp. Eng.* **2022**, *2022*, 6515234. [\[CrossRef\]](#)
22. Liu, Q.; Lu, Z.; Liu, Z.; Lin, P.; Wang, X. Ballast Water Dynamic Allocation Optimization for Revolving Floating Cranes Based on a Hybrid Algorithm of Fuzzy-Particle Swarm Optimization with Domain Knowledge. *J. Mar. Sci. Eng.* **2022**, *10*, 1454. [\[CrossRef\]](#)
23. Baroto, T.; Utama, D.M.; Ibrahim, M.F. Green supplier selection and order allocation using AHP-SAW and goal programming. *AIP Conf. Proc.* **2022**, *2453*, 020044.
24. Yin, X.; Liu, D.; Rong, W.; Li, Z. Joint Optimization of Ticket Pricing and Allocation on High-Speed Railway Based on Dynamic Passenger Demand during Pre-Sale Period: A Case Study of Beijing–Shanghai HSR. *Appl. Sci.* **2022**, *12*, 10026. [\[CrossRef\]](#)
25. Yang, Y.; Yang, B.; Yuan, Z.; Meng, R.; Wang, Y. Modelling and comparing two modes of sharing parking spots at residential area: Real-time and fixed-time allocation. *IET Intell. Transp. Syst.* **2023**. [\[CrossRef\]](#)
26. Wang, Y.; Zeng, M.; Fei, Z. Efficient Resource Allocation for Beam-Hopping-Based Multi-Satellite Communication Systems. *Electronics* **2023**, *12*, 2441. [\[CrossRef\]](#)
27. Wang, C.; Yang, F.; Li, Q.-L. Optimal Decision of Dynamic Bed Allocation and Patient Admission with Buffer Wards during an Epidemic. *Mathematics* **2023**, *11*, 687. [\[CrossRef\]](#)
28. Simoni, M.D.; Winkenbach, M. Crowdsourced on-demand food delivery: An order batching and assignment algorithm. *Transp. Res. Part C Emerg. Technol.* **2023**, *149*, 104055. [\[CrossRef\]](#)
29. Aboali, M.; Abd Manap, N.; Yusof, Z.M. Performance Analysis between Basic Block Matching and Dynamic Programming of Stereo Matching Algorithm. *J. Telecommun. Electron. Comput. Eng.* **2017**, *9*, 7–16.
30. Sun, W.; Wang, Y.; Zhang, F.; Zhao, Y. Dynamic allocation of surplus by-product gas in a steel plant by dynamic programming with a reduced state space algorithm. *Eng. Optim.* **2018**, *50*, 1578–1592. [\[CrossRef\]](#)
31. Jiang, X.; Hu, J.; Jia, M.; Zheng, Y. Parameter Matching and Instantaneous Power Allocation for the Hybrid Energy Storage System of Pure Electric Vehicles. *Energies* **2018**, *11*, 1933. [\[CrossRef\]](#)
32. Taheri, A.; Assadian, N. Dynamic-programming-based failure-tolerant control for satellite with thrusters in 6-DOF motion. *Adv. Space Res.* **2020**, *65*, 2857–2877. [\[CrossRef\]](#)
33. Razali, M.K.; Suhailan, S.; Mohamed, M.A.; Sufian, M.D.M. Online Programming Semantic Error Feedback using Dynamic Template Matching. *Int. J. Adv. Comput. Sci. Appl.* **2021**, *12*, 325–329.
34. Fu, J.; Wei, H. Dynamic programming network for point target detection. *EURASIP J. Adv. Signal Process.* **2023**, *2023*, 74. [\[CrossRef\]](#)
35. Hou, J.; Wang, N.; Luo, J.; Zuo, G.; Yang, L.; Xie, J. A dynamic allocation mechanism for formulating the allocation schemes of water resources. *Water Supply* **2023**, *23*, 996–1009. [\[CrossRef\]](#)
36. Fu, K.; Du, Z.; Li, F.; Li, Z.; Xia, C. Optimal allocation of phase shifting transformer with uncertain wind power based on dynamic programming. *Front. Energy Res.* **2023**, *10*, 1003315. [\[CrossRef\]](#)
37. Khan, I.U.; Aftab, M. Adaptive fuzzy dynamic programming (AFDP) technique for linear programming problems lps with fuzzy constraints. *Soft Comput.* **2023**, *27*, 13931–13949. [\[CrossRef\]](#)
38. He, Z.; Hu, J.; Wang, Y.; Cong, J.; Bian, Y.; Han, L. Attitude-Tracking Control for Over-Actuated Tailless UAVs at Cruise Using Adaptive Dynamic Programming. *Drones* **2023**, *7*, 294. [\[CrossRef\]](#)
39. Duan, Y.; Wang, N.; Wu, J. Optimizing Order Dispatch for Ride-Sharing Systems. In Proceedings of the 2019 28th International Conference on Computer Communication and Networks (ICCCN), Valencia, Spain, 29 July–1 August 2019.

40. Czerniachowska, K.; Lutosławski, K. Dynamic programming approach for solving the retail shelf-space allocation problem. *Procedia Comput. Sci.* **2021**, *192*, 4320–4329. [[CrossRef](#)]
41. Suo, M.; Xia, F.; Fan, Y. A Fuzzy-Interval Dynamic Optimization Model for Regional Water Resources Allocation under Uncertainty. *Sustainability* **2022**, *14*, 1096. [[CrossRef](#)]
42. Gong, Y.; Zou, W.; Yuan, X.; Yang, X.; Chen, Y. Study on Water Quantity Allocation Optimization for Single Main Canal in Large-Scale Irrigation Area Based on DP Method. *Water* **2022**, *14*, 3917. [[CrossRef](#)]
43. Amuji, H.O.; Onwuegbuchunam, D.E.; Aponjolosun, M.O.; Okeke, K.O.; Mbachu, J.C.; Ojutalayo, J.F. The dynamic programming model for optimal allocation of laden shipping containers to Nigerian seaports. *J. Sustain. Dev. Transp. Logist.* **2022**, *7*, 69–79. [[CrossRef](#)]
44. Djurdjević, D.; Bjelić, N.; Popović, D.; Andrejić, M. A Combined Dynamic Programming and Simulation Approach to the Sizing of the Low-Level Order-Picking Area. *Mathematics* **2022**, *10*, 3733. [[CrossRef](#)]
45. Wei, C.; Cheng, J. Optimal allocation of irrigation water in a single-reservoir and a single-pumping-station system under deficit irrigation conditions. *Water Supply* **2022**, *22*, 8418–8433. [[CrossRef](#)]
46. MathorCup. “MathorCup” University Mathematical Modeling Competition. 2022. Available online: <https://www.saikr.com/c/nd/11336> (accessed on 16 August 2023).
47. Shirkhorshidi, A.S.; Aghabozorgi, S.; Wah, T.Y. A Comparison Study on Similarity and Dissimilarity Measures in Clustering Continuous Data. *PLoS ONE* **2015**, *10*, e0144059. [[CrossRef](#)] [[PubMed](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.