

Article

# Neural Network Control of Perishable Inventory with Fixed Shelf Life Products and Fuzzy Order Refinement under Time-Varying Uncertain Demand

Ewelina Chołodowicz \*  and Przemysław Orłowski \* 

Faculty of Electrical Engineering, West Pomeranian University of Technology, 70-310 Szczecin, Poland

\* Correspondence: ewelina.cholodowicz@zut.edu.pl (E.C.); przemyslaw.orlowski@zut.edu.pl (P.O.)

**Abstract:** Many control algorithms have been applied to manage the flow of products in supply chains. However, in the era of thriving globalization, even a small disruption can be fatal for some companies. On the other hand, the rising environmental impact of a rapid industry is imposing limitations on energy usage and waste generation. Therefore, taking into account the mentioned perspectives, there is a need to explore the research directions that concern product perishability together with different demand patterns and their uncertain character. This study aims to propose a robust control approach that combines neural networks and optimal controller tuning with the use of both different demand patterns and fuzzy logic. Firstly, the demand forecast is generated, following which the parameters of the neural controller are optimized, taking into account the different demand patterns and uncertainty. As part of the verification of the designated controller, the sensitivity to parameter changes has been determined using the OAT method. It turns out that the proposed approach can provide significant waste reductions compared to the well-known POUT method while maintaining low stocks, a high fill rate, and providing lower sensitivity for parameter changes in most considered cases. The effectiveness of this approach is verified by using a dataset from a worldwide retailer. The simulation results show that the proposed approach can effectively improve the control of uncertain perishable inventories.

**Keywords:** perishable inventory management; sustainable control system; artificial intelligence; evolutionary computation; uncertain demand; robust optimization; energy sustainability



**Citation:** Chołodowicz, E.; Orłowski, P. Neural Network Control of Perishable Inventory with Fixed Shelf Life Products and Fuzzy Order Refinement under Time-Varying Uncertain Demand. *Energies* **2024**, *17*, 849. <https://doi.org/10.3390/en17040849>

Academic Editor: Ramiro Barbosa

Received: 27 December 2023

Revised: 6 February 2024

Accepted: 8 February 2024

Published: 11 February 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

Perishable inventories, such as food and pharmaceuticals, require careful management to ensure their quality and safety. One of the key considerations for the efficient management of such inventories is the energy consumption required to preserve and extend the shelf life of perishing goods. In recent years, there has been an increasing interest in reducing the energy consumption of supply chains, both to decrease costs and reduce environmental impacts (i.e., pollution reduction and promote carbon neutrality). There is a lot of emphasis on the promotion of the so-called “green” supply chain [1]. Green supply chains include green manufacturing, green products, green resources, green movement, and green logistics, i.e., green supply chain management (GSCM) [2]. In the era of Industry 4.0, not only is improving operational efficiency highly necessary but also compliance with the regulatory and market pressures for pollution, waste control (Sarkar et al., 2022 [3]), and energy consumption.

An effective inventory management system is crucial for ensuring the optimal functioning of the entire supply chain. A well-controlled inventory system can lead to improved efficiency and productivity throughout the entire supply chain. One of the essential aspects that must be considered in the management of perishable inventory systems is the role of energy. It is imperative that energy is utilized to maintain the appropriate temperature, humidity, and other essential environmental conditions that are necessary to preserve the

quality and safety of perishable goods. For example, refrigeration systems are commonly used to cool and preserve perishable foods, pharmaceuticals, blood, and other perishable products. These systems consume a significant amount of energy, and their efficient operation is essential for minimizing costs and reducing the environmental impact of inventory management. In addition, energy costs can represent a significant portion of the operating expenses for perishable inventory systems, and optimizing energy use can result in significant cost savings. Therefore, the efficient control of energy use is essential for the effective management of perishable inventories.

There exists a strong correlation between energy consumption and other inventory costs. Specifically, the holding cost, which is the cost of maintaining an inventory over a period of time, can impact the energy utilized in perishable inventory systems in a couple of ways. First, holding costs can motivate inventory managers to reduce their energy consumption in order to lower their overall costs. For example, if the holding cost of a perishable inventory is high, managers may be motivated to reduce energy consumption by using more energy-efficient refrigeration systems or optimizing the scheduling of energy-intensive operations. Second, the holding cost can also influence the tradeoff between energy use and product quality. In some cases, it may be more cost-effective to use more energy to maintain optimal environmental conditions for perishable goods, even if it means higher energy consumption. However, in other cases, it may be more cost-effective to allow some degradation of product quality to reduce energy utilization and lower the holding costs. The optimal balance between energy use and product quality depends on a diverse set of aspects, including the specific characteristics of the perishable products, the holding cost, and the energy cost.

## 2. Literature Review

One of the initial attempts to achieve optimal inventory performance was the development of the Economic Order Quantity (EOQ) model, which involves placing a constant order when the inventory level falls below a predetermined level. Going further, the next policy was invented where the order quantity was adjusted based on a difference between the current inventory level and the desired target level, which is called the Order-Up-To (OUT) policy. This policy has been widely used in inventory systems [4] for decades and continues to be relevant in many industries thanks to its relatively straightforward implementation compared to the more complex models. It does not require extensive data or sophisticated algorithms, making it accessible to a wide range of businesses. However, with the rise of digital technologies and the Internet of Things (IoT), there has been a significant increase in the amount of data available for businesses. This plethora of data can be harnessed by advanced algorithms to generate more precise predictions regarding demand, lead times, and other critical factors. Traditional inventory management approaches, including OUT policies, are often based on static parameters. Advanced algorithms are capable of dynamically adapting to changing conditions, considering factors such as seasonality, market trends, and unforeseen events in real time.

Many modern techniques have been applied to control the flow of products in uncertain perishable inventories. One well-known method is the Mixed Integer Programming (MIP) model. The MIP model was used in [5], where it was developed with a view to ordering and holding costs. As the optimization problem evolves, an increasing number of criteria are being considered, including carbon emissions and the freshness of perishable products, in addition to economic costs. Therefore, an increased number of criteria forced the usage of optimization methods to include the possible impacts on inventory profitability. Moreover, in current times, demand uncertainty is inevitable, so on top of that, robust methods for optimization are being more frequently used. The two-stage robust optimization (RO) model for a perishable inventory problem was proposed by Hooshangi-Tabrizi et al., 2022 [6], where an exact robust algorithm based on the column-and-row generation method was introduced. Another approach applied to inventory control is based on Benders' decomposition [7], in which the central limit theorem is used to represent uncertainty sets.

Another method that has been extensively used, as far as inventory optimization problem is concerned, is stochastic programming (SP). In RO, the decision maker does not have any distributional knowledge about the uncertainty in demand—minimizing worst-case cost, whereas in SP, it is assumed that the algorithm has full distributional knowledge about the uncertainty [8]. Also, approaches for inventory control that combine these two popular methods, RO and SP, were designed and verified in a real case study in the platelet supply chain [9].

In the era of Industry 4.0, there is rapid development in digital technologies, where enterprises invest in technologies such as automation and robotics, artificial intelligence (AI), blockchain, cloud computing, and big data analytics [1]. AI methods can contribute to the development of dynamic inventory control policies, which can adapt to changing conditions, such as fluctuations in demand patterns or variations in perishable goods' shelf life [10]. It is important to note that the methods that are using neural networks can be computationally complex. However, there are ongoing efforts to develop more efficient architectures, algorithms, and hardware solutions to address the computational challenges. Moreover, it is important to note that quantization, model pruning, parallelization, and transfer learning, where a pre-trained model is fine-tuned for a specific task, can reduce the need for extensive retraining, thus improving computational efficiency. On the other hand, more and more new approaches are being invented using neural networks to provide better convergence; for example, Input Convex Neural Networks (ICNNs) are used in Model Predictive Control (MPC) algorithms [11] to solve the problem of multiple shallow local minima. Another approach in which a neural network is incorporated is presented in [12], where fuzzy reasoning techniques and neural network structures are combined with model-based predictive control (MPC). Employing this approach contributed to simplifying the MPC algorithm.

It is worth noting that enhancing the practical efficiency of neural networks is a pursuit worth undertaking, as they are capable of identifying intricate patterns in perishable goods' consumption and degradation, thereby aiding in the determination of optimal inventory levels. The current applications of artificial neural networks to inventory management include inventory classification [13], demand forecasting [14], inventory optimization [15–17], and dynamic pricing [18]. According to the authors of [19], an artificial neural network (ANN) and genetic algorithm (GA) were employed to make predictions. A neural network was developed for forecasting purposes, and subsequently, the GA was implemented to determine the ANN's weights. Prior research has indicated that the integration of an imperialist competitive algorithm with artificial neural networks (ICA-ANN) has been proven to be more effective in predicting the demand for veterinary drugs than a genetic algorithm-based approach (GA-ANN) [14]. Artificial neural networks have also been applied to solve inventory control problems, which also include inflation [20]. The bi-objective optimization of multi-period and multi-product inventories was investigated in [20], resulting in the rapid convergence of the ANN. However, this study did not utilize authentic industry data, which could have further supported the stated advantages. On the other hand, in [21], a real-world application scenario of a German bakery chain that primarily sells highly perishable goods is presented, and in [22], a new replenishment strategy that introduces an upgraded particle swarm optimization (PSO)–backpropagation together with the (BP) neural network prediction model is investigated based on the automotive spare parts' industry. Furthermore, AI methods are very useful for minimizing the losses in perishable pharmaceutical products in a healthcare supply chain. More and more so-called 'intelligent inventory management' systems have been created, which help to increase overall cost-effectiveness [23]. The next subfield of artificial intelligence, which has also been widely implemented to address inventory control problems, is reinforcement learning. Numerous prior studies have explored the utilization of reinforcement learning (RL) in addressing inventory management challenges, exemplified by the noteworthy contributions [24–27]. For example, in reference [24], a deep reinforcement learning framework was introduced for inventory replenishment problems. However, it is worth mentioning that

this study did not explicitly account for the impact of product age, indicating an absence of perishability considerations.

In this paper, we employ a robust approach for perishable products facing demand uncertainty to minimize the overall costs. Moreover, the abovementioned studies either consider non-perishable products or do not explicitly incorporate the product's aging into the model. In other works, it is proposed that a scenario-based RO, where an optimal policy that minimizes the worst-case cost, including the product's aging, is presented [8]. Nevertheless, demand categorization-based RO was not properly addressed in the current state of the art, which is a significant research gap. The next void in the literature is about using zero lead time. To exemplify, in [9], products' aging is included only during storage, and zero lead time is assumed, which significantly simplifies the considered inventory control problem. Moreover, the popularly used EOQ models [28] have limitations in their application to complex demand patterns and nonlinearity handling compared to artificial intelligence methods, which provide the possibility of performance improvement thanks to their learning ability.

Unlike these studies, we explicitly incorporate different demand patterns and aging of products during transportation and storage, non-zero lead time, and uncertain demand. To the best of our knowledge, only a limited number of articles deal with perishable products, non-zero lead times, and uncertain demand, including demand categorization. For this reason, we propose a robust inventory system to control the perishable inventory system, which is exposed to highly different demand patterns, uncertain fluctuations, and the product's aging process.

The contribution of this study is three-fold:

- First, in contrast to existing studies, the proposed strategy distinguishes itself by permitting the incorporation of various demand patterns into the optimization process, including intermittent, lumpy, smooth, and erratic demands.
- Second, a new solution methodology for perishable inventory control under uncertain demand is proposed. Artificial neural networks and fuzzy logic are used to control the inventory systems. The ANN is used for order signal calculation, and fuzzy logic is used for the refinement of the generated orders, taking into account the forecasted uncertainty. Together with that, robust optimization is performed with the usage of the Genetic Algorithm (GA) and Wald criterion.
- Third, the proposed method holds a significant advantage in that the controller behavior is adapted to forecasted uncertainty without requiring a significant amount of computation due to the two-stage offline optimization process. This research has the potential to contribute to the development of software solutions for warehouse management, with the aim of enhancing the algorithms that control the replenishment cycles and automate these processes.

### 3. Problem Definition and Assumptions

The considered problem concerns controlling the flow of products in the perishable inventory system exposed to demand uncertainty. The difficulty is in calculating the order quantity for perishable products that are having varied demand characteristics. The determination of order quantity needs to be carried out with a view to two conflicting goals: high fill rates and low occupancy of inventory space. The problem also considers the determination of the timing of the orders. In this paper, we present a proposal that outlines the procedure for optimizing the inventory system in the presence of uncertain demand. The control approach has been tested using the nonlinear, discrete-time perishable inventory model proposed in [29] and carried out within the MATLAB environment. The model of the inventory is represented in the form of state equations, in which the state variable  $x_i(k) \in \mathbb{R}_{\geq 0}$  represents: (a) the on-hand stock per age  $x_{s+1}(k), x_{s+2}(k), \dots, x_l(k)$ ; and (b) the work-in-progress deliveries  $x_1(k), x_2(k), \dots, x_s(k)$ , where  $s$  denotes the deterministic lead-time. The available stock is a sum of the products stored in the inventory with different ages,  $i$ , which can be expressed as follows:

$$y(k) = \sum_{i=s}^l x_i(k) \quad (1)$$

where  $l$  is the expiration period of the product. This model assumes that the products have a fixed shelf-life. To formulate the model and the investigated problem, we relied on the following assumptions:

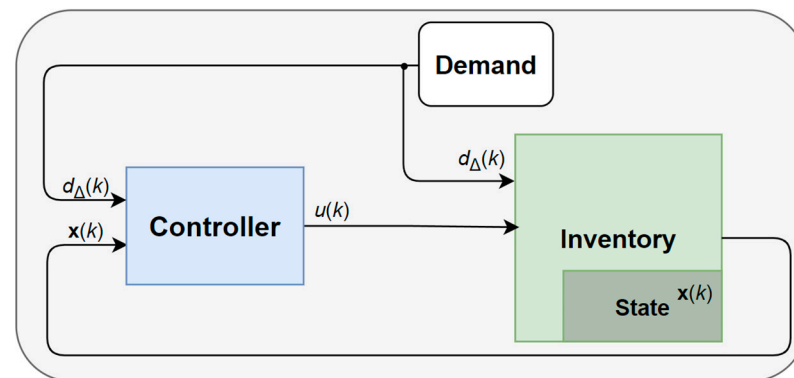
- The review interval remains constant and is set at one day.
- The products are sold in accordance with the FIFO principle.
- The inventory is holding a single type of product.
- The demand  $d(k)$  is uncertain.
- Shortages are permitted but are not backlogged. Surplus demand is lost.
- There is one replenishment point in each period.
- The demand is a nonstationary function and is categorized using Croston's method-Syntetos and Boylan method (described in Section 5.1).
- Deterioration commences immediately upon the items' arrival in the inventory.
- The shelf life  $l$  is fixed and predetermined. After  $l$  days, items from the same batch are expired and subsequently transformed into unsellable waste. Lost items are not replaced.
- In an initial state, the inventory is devoid of any items, with the inventory conditions  $x_0$  set to zero.

#### 4. Control Design

This section is divided into smaller parts that describe the design of the fuzzy switching control of perishable inventory systems under demand uncertainty and for various demand patterns.

##### 4.1. Classical Approach

The control task in terms of inventory control is defined as follows: the controller has to generate orders in the cases of quantity and timing  $u(k)$  based on the current inventory state  $x(k)$  and current demand signal  $d_{\Delta}(k)$ . The goal is to fulfill customer orders and keep inventory stock low. The classical closed-loop inventory control system is depicted in Figure 1.



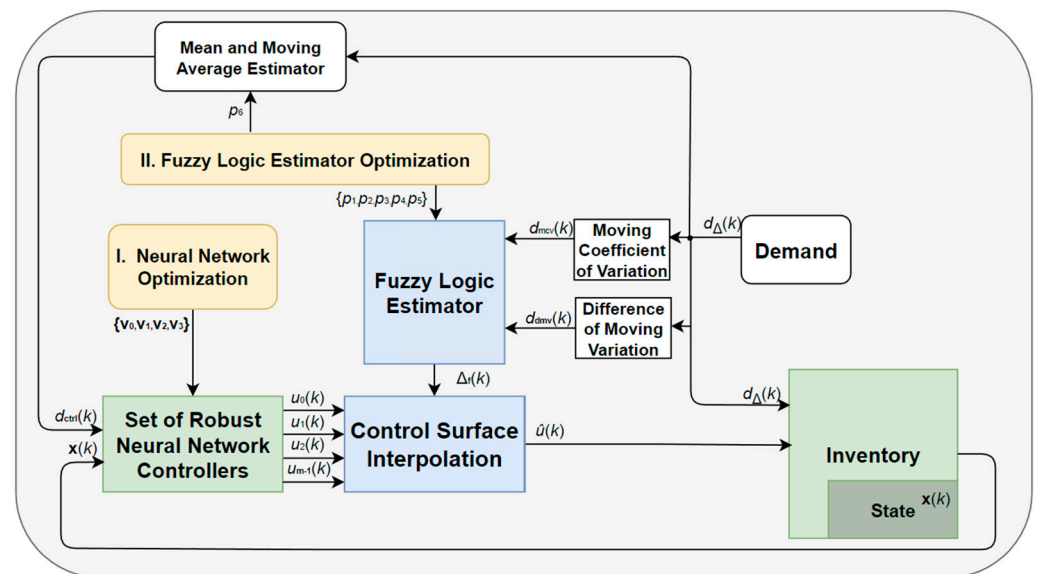
**Figure 1.** The diagram of the classical control approach devoted to inventory systems.

The challenges associated with controlling such a system stem from the dynamics and uncertain nature of demands, which are subjected to frequent changes. Moreover, these fluctuations in demand are amplified throughout the supply chain, making the dynamics of the system difficult to manage. In addition to this, the controller must generate control signals that are smooth and reasonable to follow in order to minimize the so-called bullwhip effect. Traditional approaches are not adequate for addressing this problem as they suffer from several disadvantages, such as their limited adaptability to uncertainty,

difficulty in handling non-linearities and incomplete information, and reduced robustness to disturbances.

#### 4.2. Proposed Approach

Considering the limitations of the classical closed-loop control approach, we have proposed a method that can better handle uncertainty and make use of the enormous amount of data that such systems are storing. In addition, due to the fast-changing and precise nature of these systems, control systems that can learn complex relationships from data and generalize from incomplete information are more suitable. This is particularly important in scenarios where information may be partial or imprecise. The architecture of the proposed control system is depicted in Figure 2.



**Figure 2.** The diagram of the Fuzzy Robust Neural Network (FRNN) control approach with fuzzy demand uncertainty forecasting designated for the perishable inventory systems with time-varying uncertain demand.

The Fuzzy Robust Neural Network control system (FRNN) consists of two main components: the Fuzzy Logic Estimator and a set of Robust Neural Network controllers with the weights given in vectors  $\{v_0, v_1, v_2, \dots, v_{m-1}\}$ . The generated optimal set of weights is used in the second stage of optimization, which is intended to find the optimal values of the parameters of the Fuzzy Logic estimator  $\{p_1, p_2, p_3, p_4, p_5\}$  and demand estimator  $p_6$ , taking into account the inventory performance criteria (described in Section 4). The core of the automated inventory control system is a Robust Neural Network controller, which is responsible for generating order signals for four distinct levels of demand uncertainty. Specifically, the orders generated by the controller for demands with no uncertainty are denoted as  $u_0$ , while those for demands with the highest uncertainty are denoted as  $m - 1$ , where  $m$  is the number of considered uncertainty levels. Hence, the vector of orders can be defined as follows:

$$\mathbf{u}(k) = [u_0(k), u_1(k), \dots, u_{m-1}(k)] \tag{2}$$

In the subsequent round, a linear interpolation is carried out utilizing the provided weight vector  $\mathbf{v}$  and uncertainty forecast  $\Delta_f$ , resulting in the generation of the ordering signal  $\hat{u}$ . The inventory level is initialized with an initial state, representing the initial stock, including the expiration date and customer needs  $\mathbf{d}_\Delta$ . The demand signal is divided into two parts: current customer needs, represented by signal  $\mathbf{d}_\Delta$ , and estimated demand, which is marked as  $\mathbf{d}_{ctrl}$ . Following the sale of goods and the arrival of new orders, the inventory achieves a new state,  $\mathbf{x}$ , that represents the current inventory level with information about the age of the products.

4.3. Moving Coefficient of Variation, Difference of Moving Coefficient of Variation, and Mean and Moving Average Estimator

To ascertain the anticipated demand for future products, operations were conducted on the present uncertain demand signal. Two signals were computed: the Moving Coefficient of Variation (MCV) and the Difference of Moving Coefficient of Variation (DMV). These two signals serve as inputs for the Fuzzy Logic Estimator. The Moving Coefficient of Variation is expressed as follows:

$$d_{mcv}(k, \mathbf{d}_\Delta, w_s) = \frac{\text{Moving standard deviation of the demand}(k, \mathbf{d}_\Delta, w_s)}{\bar{d}_\Delta} \tag{3}$$

where

$$\bar{d}_\Delta - \text{mean of the demand}(\mathbf{d}_\Delta)$$

whereas the subtraction of two Moving Coefficients of Variation, each characterized by a different window size  $w_s$ , represents the Difference of Moving Coefficients of Variation:

$$d_{dmv}(k, \mathbf{d}_\Delta, w_s) = d_{mcv}(k, \mathbf{d}_\Delta, w_s) - d_{mcv}(k, \mathbf{d}_\Delta, 2w_s) \tag{4}$$

On the other hand, the input demand for the Neural Controllers is made based on the weighted sum of the mean current demand  $\bar{d}_\Delta$  and the Moving Average (MA) of current demand, as expressed in the following equation:

$$d_{ctrl}(k) = (1 - p_6)\bar{d}_\Delta + p_6(\text{MA}(\mathbf{d}_\Delta(w_s))) \tag{5}$$

4.4. Fuzzy Logic Estimator

In this section, details about the Fuzzy Logic Estimator are presented. The Fuzzy Logic Estimator system’s aim is to generate the demand uncertainty forecast  $\Delta f$  based on the moving coefficient of demand variation  $d_{mcv}$  and the difference of moving demand variation  $d_{dmv}$ . In Figure 3, a flow chart of the Fuzzy Inference System (FIS) is depicted.

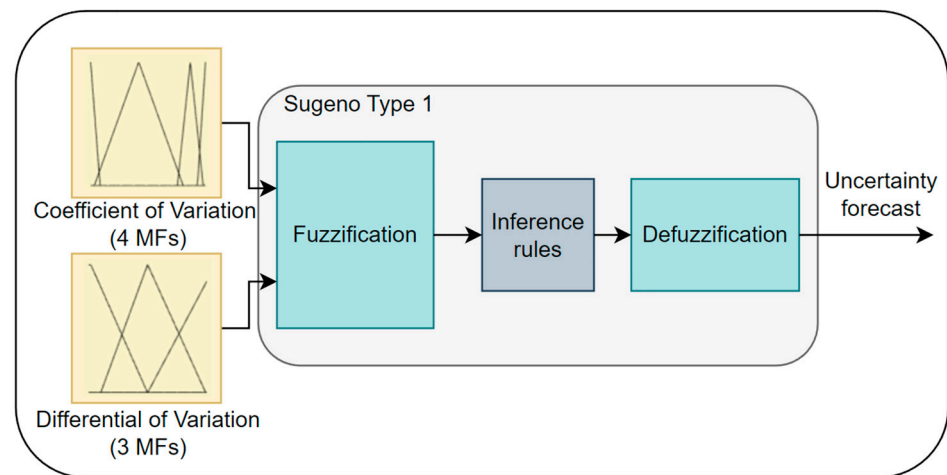


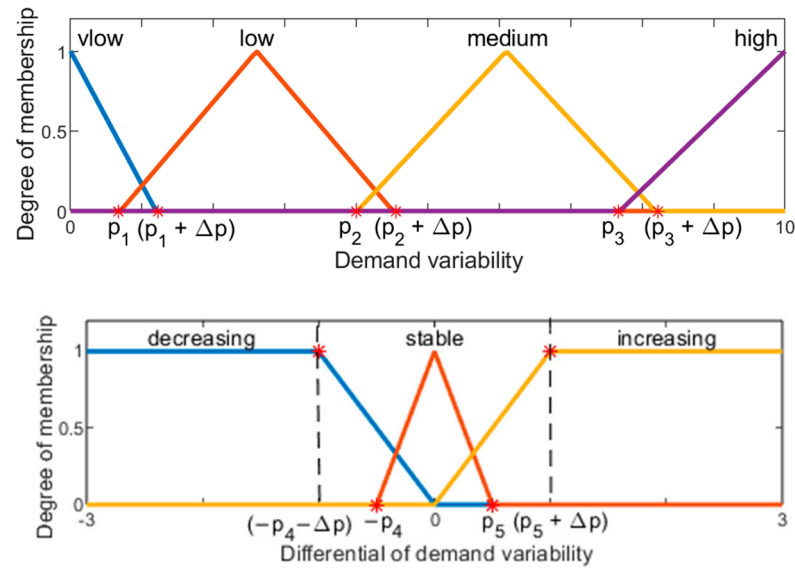
Figure 3. The flow chart of the FIS model incorporates two inputs: the coefficient of demand variation (CV) and the differential of this coefficient.

The values of the moving coefficient of variation  $d_{mcv}$  and its difference  $d_{dmv}$  are used as the inputs to the fuzzy inference system, which depends on the membership functions and the fuzzy rules presented in Table 1. The FIS observes the pattern of the coefficient of variation and its differential and correspondingly updates the uncertainty forecast so that the generated order provides optimal inventory performance (optimization details are presented in Section 4).

**Table 1.** Rule matrix for the fuzzy logic forecast generator.

| $d_{dmv}$  | $d_{mcv}$ |      |        |      |
|------------|-----------|------|--------|------|
|            | Vlow      | Low  | Medium | High |
| Decreasing | NN0       | NN0  | RNN1   | RNN2 |
| Stable     | NN0       | RNN1 | RNN2   | RNN3 |
| Increasing | RNN1      | RNN2 | RNN3   | RNN3 |

The membership function shapes and locations of optimization variables ( $p_1, p_2, p_3, p_4$ , and  $p_5$ ) are shown in Figure 4.



**Figure 4.** The optimization variables on membership functions.

The first input’s membership functions are all isosceles triangles, while the second input’s membership functions consist of an isosceles triangle and two linear functions, a z-shaped saturation function, and an s-shaped saturation function. A constant value  $\Delta p$  is introduced to provide the offset between each membership function, and it is empirically adjusted.

4.5. Neural Network Optimization

The two-stage optimization approach aims to efficiently manage inventory levels by addressing both the robustness of neural network control and efficient adaptability to demand uncertainties in separate stages. The first stage of optimization involves determining the optimal weights  $\mathbf{v}$  for the worst-case scenario of demand uncertainty, which is represented by the estimates  $\delta \in \{0, 0.1, 0.2, 0.3\}$ . The values of  $\delta$  are evenly distributed within the range of 0 to 0.3, with 0.3 serving as the assumed upper bound of the acceptable range, corresponding to a relative perturbation of  $\pm 30\%$ . The initial step of the first stage is to produce random initial conditions for the inventory state within the range (0, 2). The learning set comprises 180 various inventory states, each representing a different level of initial stock for products with varying shelf lives. The state vector  $\mathbf{x}(k) \in \mathbb{R}_{\geq 0}$ , which serves as the input for the neural network controller, consists of the number of products on every shelf, with each shelf representing the age of the product. The neural network corresponds to the control signal  $u(k) \in \mathbb{R}_{\geq 0}$ , which is the order quantity calculated to fulfill the demand  $d_{\Delta}(k) \in \mathbb{R}_{\geq 0}$ . The goal of the optimization problem is to determine the neural network’s weights with the aim of ensuring that the inventory system can meet customer needs and minimize holding costs simultaneously. Specifically, the following criteria have been used:

$$J_h(\mathbf{v}, \Delta) = \sum_{k=s+1}^N (d_{\Delta}(k) - h(k, \mathbf{v}, \Delta)) \quad (6)$$

$$J_y(\mathbf{v}, \Delta) = \sum_{k=s+1}^N m(k, \mathbf{v}, \Delta) \quad (7)$$

where Equation (6) represents the cost of lost sales because of the shortage or spoilage of products, while Equation (7) denotes the cost of holding stock surplus over current demand. The latter can be calculated using the following formula:

$$m(k, \mathbf{v}, \Delta) = \begin{cases} y(k, \mathbf{v}, \Delta) - d_{\Delta}(k) & \text{for } y(k, \mathbf{v}, \Delta) \geq d_{\Delta}(k) \wedge \hat{y}(k, \Delta) \leq d_{\Delta}(k) \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

The aforementioned relationship, Equation (8), highlights the presence of inequalities that nullify the penalty associated with the stock's initial conditions,  $x_0$ , where  $\hat{y}(k)$  represents the system's free response. The formalized criteria may be expressed as the weighted cost function:

$$J(\mathbf{v}, \Delta) = c \cdot J_h(\mathbf{v}, \Delta) + J_y(\mathbf{v}, \Delta) \quad (9)$$

The weighting factor  $c$ , which is a scaling factor that is empirically tuned to ensure an equitable balance between the aforementioned objective functions (6) and (7), is assigned a value of 3.

#### 4.6. Fuzzy Logic Estimator Optimization

The second stage of optimization seeks to provide the ability to adapt to uncertain demand by utilizing a fuzzy logic approach. In this stage, the obtained weights of the neural network in the first optimization stage are utilized as inputs. For the purpose of the second stage of optimization, a model has been developed, which comprises an artificial neural network proposed in [30], the perishable inventory model developed in [29], and the demand uncertainty estimation system based on fuzzy logic developed in this work. Formalizing the optimization problem may be expressed as follows:

$$\begin{aligned} \min_{\mathbf{p}} \quad & \max_{\Delta} J(\mathbf{p}, \Delta) \\ \text{s.t.} \quad & -\delta \leq \Delta \leq \delta \\ & p_1 \leq p_2 \\ & p_2 \leq p_3 \\ & p_4 \leq p_5 \\ & 0 \leq p_1, p_2, p_3 \leq 10 \\ & 0 \leq p_4, p_5 \leq 3 \\ & 0 \leq p_6 \leq 1 \end{aligned} \quad (10)$$

where  $p_1, p_2$ , and  $p_3$  are the parameters of membership function for the first input, and  $p_4$  and  $p_5$  are the parameters of membership function for the second input. The remaining parameter,  $p_6$ , is a parameter that determines the proportions of the mean and moving average of the demand signal that are inputted into the RNN. The detailed steps of the second stage of the optimization process are shown in Figure 5.

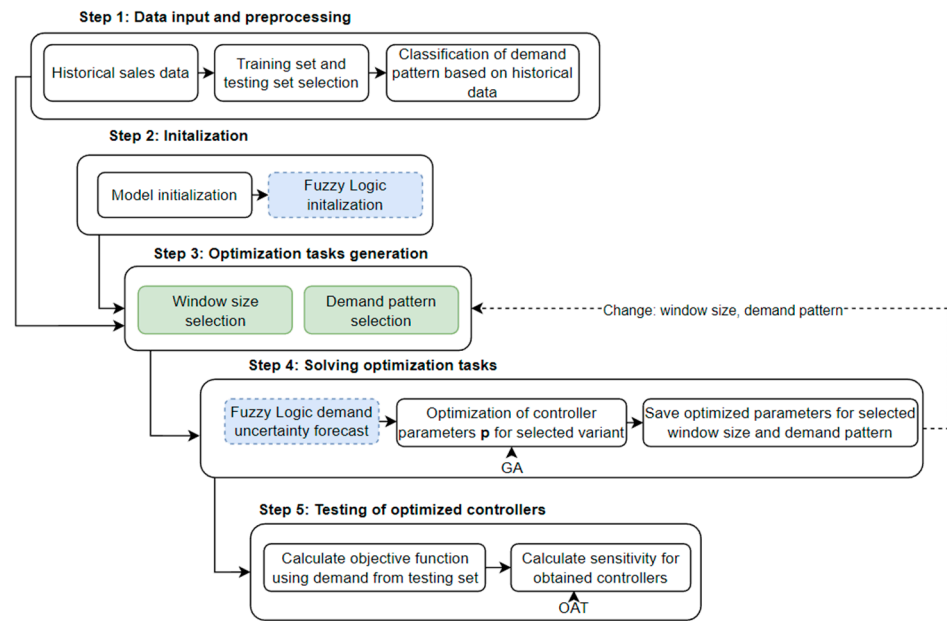


Figure 5. Fuzzy Logic estimator optimization and verification process diagram.

In the second step, the perishable inventory models, Robust Neural Network controllers, and Fuzzy Logic are initialized. Subsequently, the optimization process is executed separately for each considered demand pattern and for various window sizes of the mean, moving average, and moving standard deviation of the demand signal. The optimization of the FRNN controller was then carried out for a specific variant, which involved a chosen window size and demand pattern. In order to classify the demand patterns, the classification method called Croston's method-Synetetos and Boylan method is used, which was proposed in [31]. This method considers the calculation of two indicators: the average demand interval (ADI) and the coefficient of variation (CV). These indicator values were calculated utilizing the following formulas [32]:

$$ADI = \frac{\text{Total number of demand periods}}{\text{The number of periods with non-zero demand}} \quad (11)$$

The ADI indicator, which stands for the average demand interval, refers to the period of time that elapses between two consecutive demands.  $CV^2$  can be expressed in the following manner:

$$CV^2 = \left( \frac{\text{Standard deviation of the demand}}{\text{Mean of the demand}} \right)^2 \quad (12)$$

This method enables us to identify various types of demand, including smooth demand, which presents a consistent quantity and timing ( $ADI < 1.32$  and  $CV^2 < 0.49$ ). Erratic demands are defined by unpredictable fluctuations in the size of demand in place of changes in the demand period ( $ADI < 1.32$  and  $CV^2 > 0.49$ ), while intermittent demands can be described by a lower level of variability in the demand quantity than the demand period ( $ADI > 1.32$  and  $CV^2 < 0.49$ ), lumpy demand ( $ADI > 1.32$  and  $CV^2 > 0.49$ ), and intensely unpredictable and sporadic fluctuations in demand, which include periods of zero demand [33].

#### 4.7. Verification

The performance of the optimized FRNN controller was verified using the demand signals from a testing set. The demand signals were applied to the inventory system, and for each applied testing demand signal, the criterion (9) was calculated. Additionally,

sensitivity analysis was conducted using the One-at-a-Time (OAT) method to assess the impact of changes in the controller parameters on the inventory control quality.

## 5. Simulation Research

This section details the simulation research, the primary objective of which is to demonstrate the effectiveness of the proposed controller in enhancing the performance of perishable inventory systems in the face of uncertain demand. The focus is on evaluating the impact of the controller on stock levels, shortage rates, and wastage due to perishability across various demand patterns.

The dataset comprises time series data related to the sales of diverse items offered by Walmart stores, arranged in a hierarchical structure, with the following levels: item level, department, product category, and geographical region. The dataset consists of 3075 products that have been sorted into 3 product categories, namely hobby, food, and household, and 7 product departments, where the aforementioned categories are further disaggregated. This dataset was released with the intention of participating in the M5 Competition hosted on Kaggle [34]. The items are distributed throughout ten retail locations across the states of California, Texas, and Wisconsin. For the purposes of this research, the item category under consideration is food. In the food category, there are 1437 products whose time series have been classified into four demand patterns dependent on two indicators, denoted as Equations (11) and (12), namely the average demand interval (ADI) and the coefficient of variation (CV). The aforementioned research incorporates the following time series for each demand group: 23,042 for intermittent, 5971 for lumpy, 981 for smooth, and 496 for erratic. From these data, a training set and testing set were created in a ratio of 4:1.

The results of the simulation were assessed by evaluating criterion (9) in different simulation scenarios, which include the different window size values used for optimization and testing from the set {7, 14, 30} and different demand patterns.

### 5.1. Reference Controllers: SRNN and POUT

To ensure a fair comparison between the proposed FRNN's performance and existing approaches, two reference controllers were selected: the Switching Robust Neural Network (SRNN) controller described in [35] and the Proportional Order-Up-To Level (POUT) controller described in [36].

The Switching Robust Neural Network controller (SRNN) is a robust neural network (RNN) controller that utilizes adaptive weight switching based on the estimated demand uncertainty and predefined threshold parameters. Accordingly, the SRNN involves the utilization of four RNN controllers that can be toggled depending on the proportions of the MSTD and MAVG within a specified window size and category of demand signal. The switching operation can take place daily. On the other hand, the POUT controller is a successful modification of a classical OUT policy that seeks to mitigate the effects of the bullwhip effect in traditional inventory systems by smoothing out order variations. Additionally, the POUT controller is equipped with the ability to fine-tune its performance through the use of positive constants, which can be adjusted to optimize inventory management.

The comparison with the reference approaches, the Switching Robust Neural Network (SRNN) and Proportional Order-Up-To Level (POUT), was conducted. The cost function value, fill rate, sum of stock level, number of perished products, and time response analysis for the selected demand scenarios were considered for this comparison. Furthermore, a sensitivity analysis was carried out using the OAT method. For the purpose of fair comparison, two reference systems were optimized using the same criteria, and a two-stage optimization process was carried out for the SRNN, as it also contains a Robust Neural Network as part of the controller. During the second stage of optimization, switching thresholds were calculated for the SRNN  $t_1$ - $t_4$ , and for POUT, two parameters were optimized: the maximal inventory level  $o_1$  and a positive constant  $o_2$  that determines the share of the difference between the maximal stock level and the current stock level, as well as the work-in-progress orders in the order signal generation.

## 5.2. Parameters

The parameters utilized in the simulation research are presented in Table 2.

**Table 2.** Parameter values that are used for simulation research.

| Area              | Parameter                          | Value                     |
|-------------------|------------------------------------|---------------------------|
| General           | $N$ (simulation interval)          | 365 days                  |
|                   | Replenishment period               | 1 day                     |
|                   | $l$ (product shelf-life)           | 9 days                    |
|                   | Perishability type                 | fixed                     |
|                   | Issuing policy                     | FIFO                      |
|                   | Lead time                          | 3 days                    |
|                   | $m$ (number of uncertainty levels) | 4                         |
| Neural Network    | Type of moving average             | Triangular                |
|                   | Window size                        | Parametrized              |
|                   | Number of neurons                  | 3                         |
|                   | Type                               | Feedforward               |
| FRNN              | Type of fuzzy logic                | Sugeno Type 1             |
|                   | Type of moving average window      | Triangular                |
|                   | Range of input 1 (CV)              | [0, 10]                   |
|                   | Range of input 2 (DCV)             | [-3, 3]                   |
|                   | Type of MF for input 1             | trimf                     |
|                   | Types of MF for input 2            | linzmf, trimf, and linsmf |
|                   | Output range                       | [0, 3]                    |
|                   | Defuzzification method             | wtaver                    |
|                   | AndMethod                          | prod                      |
|                   | OrMethod                           | probor                    |
| ImplicationMethod | prod                               |                           |
| AggregationMethod | sum                                |                           |

## 5.3. Optimization Results

The results of the optimization process are presented utilizing the cost function values, optimal parameter values, and the performance of the inventory system demonstrated through time response charts. Additionally, the sensitivity analysis was conducted to show the robustness and performance of the control system in cases of parameter change.

### 5.3.1. Cost Function Values

Table 3 presents the obtained results from the optimization process for the SRNN, FRNN, and POUT controllers. The results include the cost function values according to Formula (5) obtained in the optimization and in the testing process (in brackets). The computation of the cost function value is undertaken for selected window sizes  $w_s$ , with the aim of assessing the performance of the controllers optimized with a specific window size  $w_{opt}$  in the context of different window sizes  $w_{test}$ . In the present investigation, the controllers optimized on selected  $w_{opt}$  are tested in the scenarios wherein the window size is altered to  $w_{test}$ . In Table 3, the cost function values are provided for different window sizes,  $w_{opt}$  and  $w_{test}$ , and four selected demand patterns, where  $w_{opt}$  is the window size used during the optimization process (listed in rows) and  $w_{test}$  is the window size used for testing purposes (listed in columns).

**Table 3.** The cost function values of the FRNN, SRNN, and POUT control systems were obtained for two different demand sets: the optimization set and the testing set (in brackets), for different window sizes.

| Demand Pattern | $w_{opt}$ \ $w_{test}$ | FRNN $\times 10^4$ |        |        | SRNN $\times 10^4$ |        |        | POUT            |
|----------------|------------------------|--------------------|--------|--------|--------------------|--------|--------|-----------------|
|                |                        | 7                  | 14     | 30     | 7                  | 14     | 30     | -               |
| Intermittent   | 7                      | 20.81              | 20.14  | 20.79  | 20.84              | 20.31  | 20.86  | 29.92<br>(6.76) |
|                |                        | (4.79)             | (4.54) | (4.76) | (4.75)             | (4.56) | (4.76) |                 |
|                | 14                     | 20.83              | 20.10  | 20.71  | 21.00              | 20.22  | 20.90  |                 |
|                |                        | (4.77)             | (4.52) | (4.72) | (4.82)             | (4.58) | (4.81) |                 |
|                | 30                     | 20.81              | 20.12  | 20.64  | 20.96              | 20.26  | 20.79  |                 |
|                |                        | (4.77)             | (4.52) | (4.73) | (4.79)             | (4.58) | (4.77) |                 |
| Lumpy          | 7                      | 30.43              | 28.97  | 29.25  | 30.14              | 29.00  | 29.40  | 32.72<br>(7.27) |
|                |                        | (6.75)             | (6.36) | (6.42) | (6.72)             | (6.39) | (6.45) |                 |
|                | 14                     | 31.31              | 28.49  | 28.80  | 31.70              | 28.66  | 28.81  |                 |
|                |                        | (7.15)             | (6.43) | (6.48) | (7.28)             | (6.50) | (6.52) |                 |
|                | 30                     | 31.60              | 28.62  | 28.54  | 31.63              | 28.78  | 28.67  |                 |
|                |                        | (7.25)             | (6.47) | (6.44) | (7.27)             | (6.51) | (6.47) |                 |
| Smooth         | 7                      | 14.51              | 14.44  | 14.60  | 14.53              | 14.53  | 14.74  | 17.09<br>(4.19) |
|                |                        | (3.32)             | (3.29) | (3.42) | (3.32)             | (3.31) | (3.45) |                 |
|                | 14                     | 14.80              | 14.34  | 14.54  | 14.62              | 14.46  | 14.70  |                 |
|                |                        | (3.47)             | (3.33) | (3.48) | (3.40)             | (3.33) | (3.49) |                 |
|                | 30                     | 14.66              | 14.39  | 14.42  | 14.71              | 14.55  | 14.57  |                 |
|                |                        | (3.39)             | (3.31) | (3.45) | (3.40)             | (3.34) | (3.49) |                 |
| Erratic        | 7                      | 20.75              | 20.21  | 20.87  | 20.78              | 20.50  | 21.13  | 25.57<br>(5.98) |
|                |                        | (4.23)             | (4.15) | (4.40) | (4.21)             | (4.19) | (4.43) |                 |
|                | 14                     | 20.85              | 20.11  | 20.81  | 21.05              | 20.33  | 21.00  |                 |
|                |                        | (4.24)             | (4.12) | (4.40) | (4.28)             | (4.17) | (4.43) |                 |
|                | 30                     | 20.90              | 20.13  | 20.73  | 21.27              | 20.41  | 20.99  |                 |
|                |                        | (4.23)             | (4.10) | (4.36) | (4.34)             | (4.18) | (4.44) |                 |

Based on the cost function values obtained for the optimization set, the FRNN controller appears to be delivering the best performance in the majority of scenarios considered. Specifically, in the training phase, the FRNN controller provided the lowest cost function value for 94% of the scenarios compared to the SRNN and POUT controllers. In the testing phase, the advantage of the FRNN controller is visible in 83% of the calculated cases, which consist of three different window sizes and four different demand patterns. The SRNN controller ranked second, followed by the POUT controller. The advantage of the FRNN controller over the POUT controller is significant, ranging from 7% for lumpy demand patterns with a window size of 7 to 33% for intermittent demand with a window size of 14. The mean advantage of the FRNN controller over the POUT controller for intermittent demand is 31%; for lumpy demand, it is 10%; for smooth demand, it is 18%; and for erratic demand, it is 24%. The advantage of the FRNN over the SRNN is most pronounced in cases of lumpy demand patterns, where the maximal advantage is around 2%. The total performance improvement calculated as the mean for all demand patterns is approximately equal to 1% for the two-week window size used for optimization and testing ( $w_{opt} = 14$  and  $w_{test} = 14$ ) and 0.63% for the window sizes  $w_{opt} = 14$  and  $w_{test} = 7$ . The lowest advan-

tage for smooth demand is observed, indicating that the FRNN and SRNN controllers are functioning similarly for smooth demand.

### 5.3.2. FRNN Parameters

Based on the presented membership function design in Section 4 (Figure 4), the optimized membership functions are shown in Figure 6.

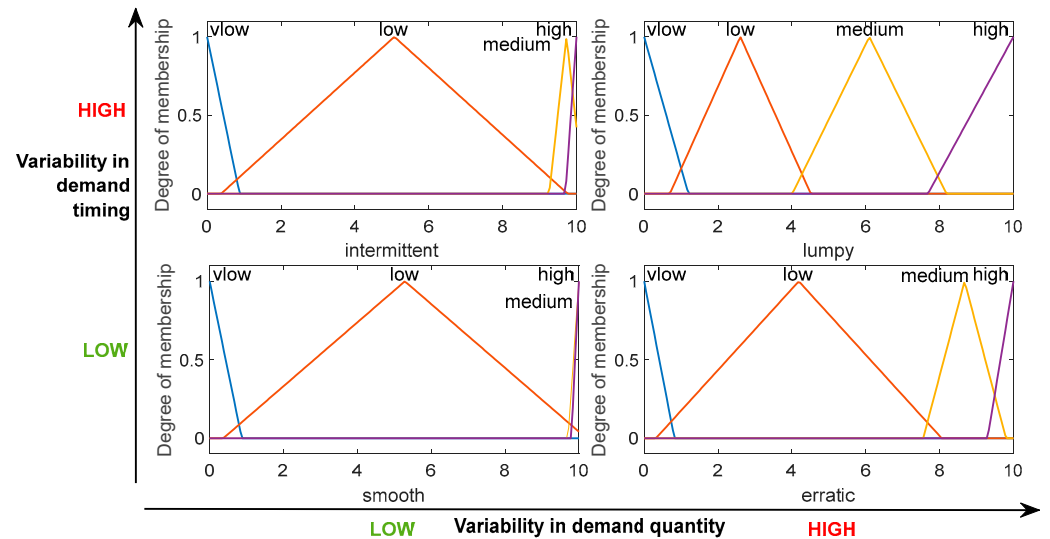


Figure 6. Optimization results of FRNN parameters of membership functions.

In cases of smooth demand, the MF ‘low’ is strongly dominant and covers almost the whole input range. Conversely, in the presence of intermittent demand, while the MF ‘low’ and MF ‘high’ shares remain very similar, like for smooth demand, the MF ‘medium’ experiences a slight increase. However, in cases of erratic demand, the MF ‘medium’ assumes a larger share than for smooth and intermittent demands. Notably, in the case of lumpy demand, the shares of all MFs become more balanced.

Going further, the relationship between the first input, the coefficient of variation, and the output, uncertainty forecast  $\Delta_f$ , is presented in Figure 7.

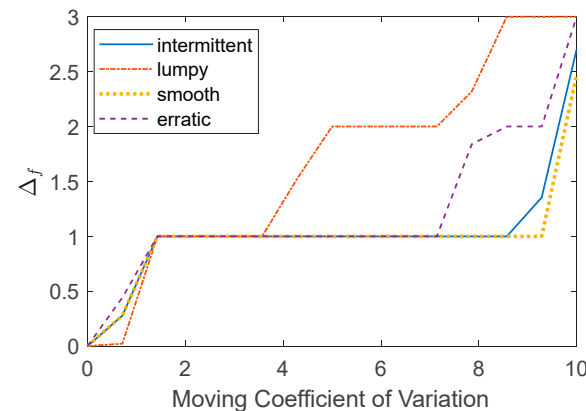


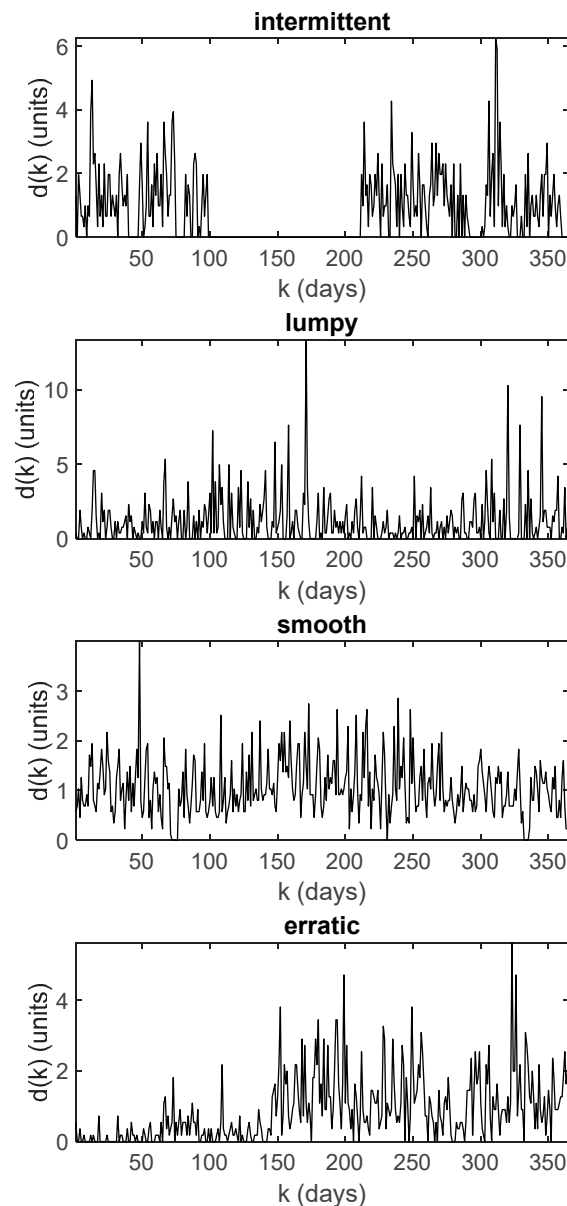
Figure 7. Control plot showing the relationship between the Moving Coefficient of Variation and the generated output from the Fuzzy Logic system, which is uncertainty estimation.

It is observed that the discrepancy in the forecast uncertainty commenced to surface when the coefficient of demand variation exceeded a value of 3.6. For a range of 5–7.1, the forecast uncertainty for lumpy demand is found to be double that of other demand patterns. Subsequently, the uncertainty forecast increased for erratic demand at a value of 7.1 on the coefficient of variation scale and for intermittent demand at 8.6. Conversely, the forecast uncertainty for smooth demand remained constant at a value of 1 within the range

of 1–9.5, indicating that this demand pattern is characterized by a consistent and stable pattern with minimal fluctuations in demand levels. In contrast, the obtained forecast uncertainty for lumpy demand changed three times within the same range of coefficient of variation, underscoring its irregularity and unpredictability.

#### 5.4. Time Responses

For the purpose of supporting the analysis of the perishable inventory performance provided by the considered controllers, individual demand signals that represent certain categories are selected from the testing set and given in Figure 8.

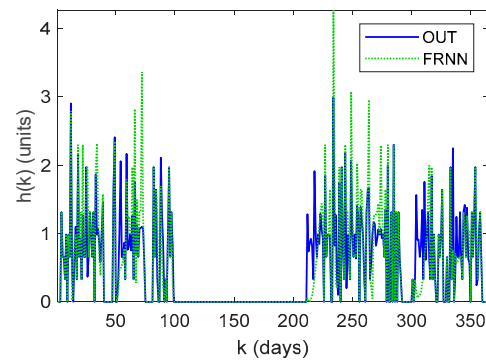


**Figure 8.** The demand signals that were selected for the time response analysis.

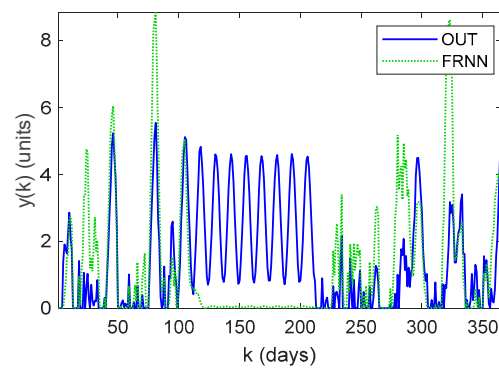
To start with, an intermittent demand pattern is applied to the perishable inventory system. In Figure 9, there is a chart that shows the daily sales of certain products in the inventory, whereas Figure 10 shows the daily stock level.

It is evident that the FRNN is able to sell more; it has about a 2% higher fill rate while keeping 20% less stock and having 56% less losses due to product perishability in comparison to POUT.

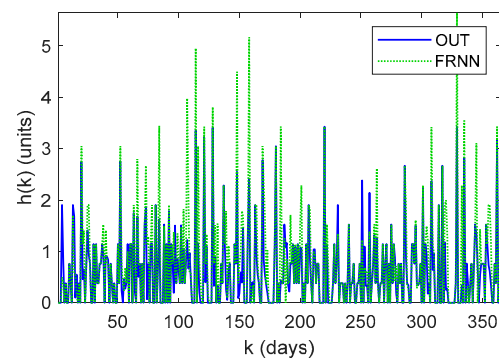
Next, the system response for lumpy demand is provided in Figures 11 and 12 with the sold product signal and stock level, respectively.



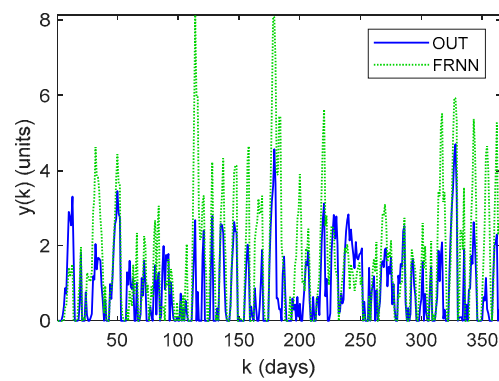
**Figure 9.** Sold products for the FRNN and POUT in cases of intermittent demand.



**Figure 10.** Stock level for the FRNN and POUT in cases of intermittent demand.



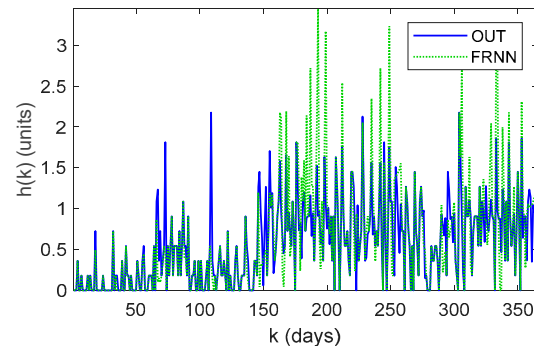
**Figure 11.** Sold products for the FRNN and POUT in cases of lumpy demand.



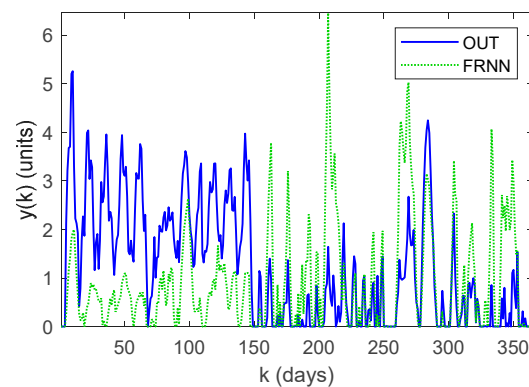
**Figure 12.** Stock level for the FRNN and POUT in cases of lumpy demand.

In cases of lumpy demand, the advantage of the FRNN in the case of fulfilling customers' needs is significant and equal to 14% compared to the results obtained by POUT. Although the FRNN provided this advantage by incurring higher inventory holding costs, it ultimately proved to be more profitable than providing a lower fill rate.

Moving on to the erratic demand pattern, Figures 13 and 14 present the sold products and stock levels for the POUT and FRNN controllers, respectively.



**Figure 13.** Sold products for the FRNN and POUT in cases of erratic demand.



**Figure 14.** Stock level for the FRNN and POUT in cases of erratic demand.

It is important to note that the advantage of the FRNN in cases of product perishability is significantly visible for the erratic demand pattern because, in this case, the FRNN is able to provide about 74% less product waste while keeping about a 5% higher fill rate and 24% less stock. In turn, for a smooth demand pattern, the results are shown in Figures 15 and 16.

Despite the 20% higher stock level, the FRNN controller managed to reduce product perishability by 42% compared to the POUT method, which indicates that it is more effective in adapting to uncertainties. Additionally, the fill rate of the FRNN is 1.5% higher than for POUT. Additionally, the generated orders are shown in Figure 17.

In terms of order signal generation, POUT is characterized by the generation of the highest peak values among the considered controllers, which can cause higher stock levels and eventually higher stock perishability. Although the SRNN and FRNN generate similar orders, the ordering signal for the SRNN is less suited to the demand shape, and it can be observed that the signal for the SRNN is smoother than that for the FRNN. Delving further, based on the data for the smooth demand pattern, the SRNN allows us to reduce product perishability by 29.19% compared to POUT, whereas the FRNN is able to reduce it by up to 42.29%.

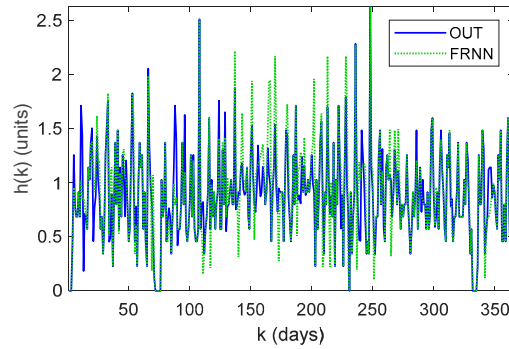


Figure 15. Sold products for the FRNN and POUT in cases of smooth demand.

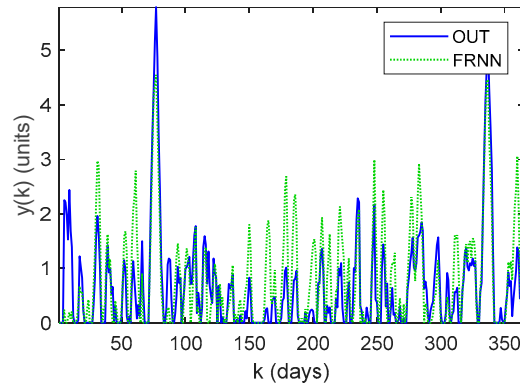


Figure 16. Stock level for the FRNN and POUT in cases of smooth demand.

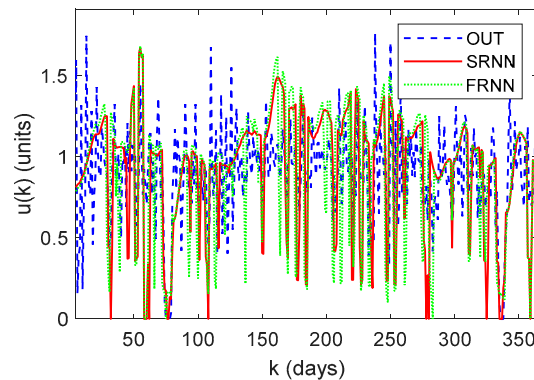


Figure 17. Orders for the FRNN, SRNN, and POUT in cases of smooth demand.

5.5. Sensitivity Analysis

In the subsequent phase of the simulation research, the sensitivity is evaluated through the One-at-the-Time (OAT) method. Supporting data for the sensitivity analysis are presented in Tables 4–6. In Table 7, the maximum absolute sum of OAT sensitivity for the considered controllers SRNN, FRNN, and POUT is presented, including separate calculations per demand pattern.

It is generally observed that the lowest sensitivity value is exhibited by POUT in all instances. Considering the window sizes, the FRNN is about 16% less sensitive to a one-week window ( $w_{opt} = 7$ ) and about 8% to a two-week window size ( $w_{opt} = 14$ ). For the one-month window size ( $w_{opt} = 30$ ), the FRNN approximately displays a 6% higher sensitivity level than the SRNN.

**Table 4.** The maximum absolute sum of OAT sensitivity for parameter perturbation is  $\pm 0.1\%$  per demand pattern and per window size for POUT.

| Parameter  | Intermittent | Lumpy  | Smooth | Erratic |
|------------|--------------|--------|--------|---------|
| $\sigma_1$ | 0.0008       | 0.0005 | 0.0031 | 0.0012  |
| $\sigma_2$ | 0.0003       | 0.0002 | 0.0006 | 0.0003  |

**Table 5.** The maximum absolute sum of OAT sensitivity for parameter perturbation is  $\pm 0.1\%$  per demand pattern and per window size for the SRNN.

| Parameter | Window Size | Intermittent | Lumpy  | Smooth | Erratic |
|-----------|-------------|--------------|--------|--------|---------|
| $t_1$     | 7           | 0.1728       | 0      | 0.0863 | 0.1427  |
|           | 14          | 0            | 0      | 0.2172 | 0.0120  |
|           | 30          | 0            | 0.0084 | 0.3801 | 0.0485  |
| $t_2$     | 7           | 0.0352       | 0.0420 | 0.0775 | 0.0177  |
|           | 14          | 0.1996       | 0.0988 | 0.1090 | 0.2626  |
|           | 30          | 0.1941       | 0.1247 | 0.0645 | 0.2121  |
| $t_3$     | 7           | 0.0503       | 0.0073 | 0.0124 | 0.0550  |
|           | 14          | 0.0959       | 0.0968 | 0.0288 | 0.1002  |
|           | 30          | 0.0862       | 0.1388 | 0.1033 | 0.0845  |
| $t_4$     | 7           | 0.1145       | 0.1244 | 0.1084 | 0.1413  |
|           | 14          | 0.3567       | 0.3044 | 0.2364 | 0.3633  |
|           | 30          | 0.3480       | 0.5553 | 0.1841 | 0.4541  |

**Table 6.** The maximum absolute sum of OAT sensitivity for parameter perturbation is  $\pm 0.1\%$  per demand pattern and per window size for the FRNN.

| Parameter | Window Size | Intermittent | Lumpy  | Smooth | Erratic |
|-----------|-------------|--------------|--------|--------|---------|
| $P_1$     | 7           | 0.0238       | 0.0419 | 0.0661 | 0.0125  |
|           | 14          | 0.0002       | 0.0143 | 0.0001 | 0.0156  |
|           | 30          | 0.0349       | 0.0109 | 0.1469 | 0.0088  |
| $P_2$     | 7           | 0.0165       | 0.0161 | 0.0233 | 0.0208  |
|           | 14          | 0.1219       | 0.1408 | 0.0575 | 0.2015  |
|           | 30          | 0.3527       | 0.2620 | 0.0711 | 0.2141  |
| $P_3$     | 7           | 0.0121       | 0.0009 | 0.0359 | 0.0016  |
|           | 14          | 0.0173       | 0.0578 | 0.0173 | 0.0762  |
|           | 30          | 0.2305       | 0.2517 | 0.0177 | 0.1258  |
| $P_4$     | 7           | 0.0222       | 0.0549 | 0      | 0.0219  |
|           | 14          | 0.0173       | 0.0218 | 0.0751 | 0.0621  |
|           | 30          | 0.0365       | 0.0470 | 0      | 0.0777  |
| $P_5$     | 7           | 0.0185       | 0.0123 | 0.0053 | 0.0087  |
|           | 14          | 0.0355       | 0.0892 | 0.0700 | 0.1099  |
|           | 30          | 0.2785       | 0.1289 | 0.0764 | 0.1889  |
| $P_6$     | 7           | 0.0967       | 0.1312 | 0.1495 | 0.0796  |
|           | 14          | 0.2689       | 0.4399 | 0.0916 | 0.2715  |
|           | 30          | 0.6477       | 0.3948 | 0.2485 | 0.3196  |

**Table 7.** The maximum absolute sum of OAT sensitivity for parameter perturbation  $\pm 0.1\%$  per controller per demand pattern.

| Window Size | Demand Pattern | FRNN   | SRNN   | POUT                    |
|-------------|----------------|--------|--------|-------------------------|
| 7           | Intermittent   | 0.0967 | 0.1728 | $3.0274 \times 10^{-4}$ |
|             | Lumpy          | 0.1312 | 0.1244 | $2.1668 \times 10^{-4}$ |
|             | Smooth         | 0.1495 | 0.1084 | $6.0458 \times 10^{-4}$ |
|             | Erratic        | 0.0796 | 0.1413 | $3.3116 \times 10^{-4}$ |
| 14          | Intermittent   | 0.2689 | 0.3567 | $3.0274 \times 10^{-4}$ |
|             | Lumpy          | 0.4399 | 0.3044 | $2.1668 \times 10^{-4}$ |
|             | Smooth         | 0.0916 | 0.2364 | $6.0458 \times 10^{-4}$ |
|             | Erratic        | 0.2715 | 0.3633 | $3.3116 \times 10^{-4}$ |
| 30          | Intermittent   | 0.6477 | 0.3480 | $3.0274 \times 10^{-4}$ |
|             | Lumpy          | 0.3948 | 0.5553 | $2.1668 \times 10^{-4}$ |
|             | Smooth         | 0.2485 | 0.1841 | $6.0458 \times 10^{-4}$ |
|             | Erratic        | 0.3196 | 0.4541 | $3.3116 \times 10^{-4}$ |

## 6. Conclusions

In this paper, we addressed the problem of uncertain, perishable inventory control optimization. Regarding the solution, we proposed an approach for the automatic control of uncertain, perishable inventory systems, supporting the resilience of the whole supply chain. The proposed solution combines neural networks and optimal controller tuning with the fuzzy logic system that forecasts the demand uncertainty. This approach considers different demand patterns: intermittent, lumpy, smooth and erratic. The focus was placed on enhancing the effectiveness of the inventory system by providing a demand uncertainty forecast, which is provided to refine the generated order signal due to the dynamically changing demand patterns. With respect to the automatic order generation process, it can be stated that the proposed FRNN controller outperformed the reference methods. Consequently, the FRNN is more responsive to the demand uncertainties in the inventory system than the POUT and SRNN.

The proposed approach is compared with existing approaches, including the extensively popular classical method POUT and SRNN, which possess similar core functionality—Artificial Neural Networks. The results indicate that the proposed approach significantly outperforms the classical POUT method in terms of its fill rate; the highest advantage is achieved in the scenario with lumpy demand, where the fill rate is increased by 14% for the FRNN than for POUT. Additionally, the FRNN can be utilized to reduce product perishability, as demonstrated by the 74% reduction in waste for an erratic demand pattern scenario. Furthermore, the stock level was lower for the FRNN than for POUT in the erratic demand pattern scenario, with a reduction of approximately 24%. Conversely, when comparing the FRNN to the SRNN, the advantage is visible for specific demand patterns. To be more precise, the computational results indicate that the FRNN provides more robust solutions, especially for lumpy demands, compared to the SRNN. However, for the other considered demand patterns, the difference in performance is slight, which means that the FRNN method can improve the inventory control for products that have a lumpy demand pattern. During the sensitivity analysis, it was established that despite the FRNN not attaining the lowest sensitivity, the sensitivity level can be considered low (less than one). Furthermore, despite the extended functionality of the FRNN, it is shown that the sensitivities of the SRNN and FRNN are very similar. To conclude, the proposed FRNN approach has two main advantages: (1) the controller's behavior can be easily changed based on forecasted uncertainty without the need for extensive calculations thanks to the offline optimization process and switching capability; and (2) it provides low sensitivity in cases of parameter change. It provides a new perspective to solve the perishable inventory problem. It is found that this study can contribute to the development of software solutions for warehouse management to improve the algorithms that control replenishment cycles

and automate them. The aforementioned study has limitations, as it primarily focuses on a specific category of perishable goods, limiting the generalizability of the proposed neural network and fuzzy logic controller. Additionally, the study's outcomes are contingent on the quality and availability of the data, raising concerns about potential inaccuracies in representing the dynamics of perishable inventory systems. Moreover, this research does not thoroughly explore the computational complexity associated with implementing the proposed controller, overlooking the potential challenges related to real-time applications and resource constraints.

Possible future research directions include the development of advanced forecasting techniques specifically tailored for perishable goods and the investigation of applying a dynamic pricing strategy to optimize revenue and inventory levels in cases of different characteristics of demand patterns.

**Author Contributions:** Conceptualization, E.C. and P.O.; methodology, P.O.; software, E.C. and P.O.; validation, E.C. and P.O.; formal analysis, E.C. and P.O.; investigation, E.C. and P.O.; data curation, E.C. and P.O.; writing—original draft preparation, E.C.; writing—review and editing, P.O. and E.C.; visualization, E.C.; supervision, P.O. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Data Availability Statement:** No new data were created or analyzed in this study. Data sharing is not applicable to this article.

**Conflicts of Interest:** The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

## References

- Feng, Y.; Lai, K.H.; Zhu, Q. Green Supply Chain Innovation: Emergence, Adoption, and Challenges. *Int. J. Prod. Econ.* **2022**, *248*, 108497. [CrossRef]
- Enhancing Green Supply Chain Efficiency Through Linear Diophantine Fuzzy Soft-Max Aggregation Operators. Available online: [https://www.acadlore.com/article/JII/2023\\_1\\_1/jii010102](https://www.acadlore.com/article/JII/2023_1_1/jii010102) (accessed on 29 January 2024).
- Sarkar, B.; Ganguly, B.; Pareek, S.; Cárdenas-Barrón, L.E. A Three-Echelon Green Supply Chain Management for Biodegradable Products with Three Transportation Modes. *Comput. Ind. Eng.* **2022**, *174*, 108727. [CrossRef]
- Axsäter, S. *Inventory Control*; Springer: Berlin, Germany, 2015; Volume 225. [CrossRef]
- Paldrak, M.; Erol, E.; İnan, A.; Firat, D.; Miran, A.E.; Çetinkaya, E.; Polat, I.N.; Staiou, E.; Kasap, B.; Aydın, P. Demand Forecasting and Inventory Control System for Industrial Valves. *Lect. Notes Mech. Eng. Towards Ind. 5.0* **2023**, 780–796. [CrossRef]
- Hooshangi-Tabrizi, P.; Hashemi Doulabi, H.; Contreras, I.; Bhuiyan, N. Two-Stage Robust Optimization for Perishable Inventory Management with Order Modification. *Expert. Syst. Appl.* **2022**, *193*, 116346. [CrossRef]
- Thorsen, A.; Yao, T. Robust Inventory Control under Demand and Lead Time Uncertainty. *Ann. Oper. Res.* **2017**, *257*, 207–236. [CrossRef]
- Pathy, S.R.; Rahimian, H. A Resilient Inventory Management of Pharmaceutical Supply Chains under Demand Disruption. *Comput. Ind. Eng.* **2023**, *180*, 109243. [CrossRef]
- Ensafian, H.; Yaghoubi, S. Robust Optimization Model for Integrated Procurement, Production and Distribution in Platelet Supply Chain. *Transp. Res. E Logist. Transp. Rev.* **2017**, *103*, 32–55. [CrossRef]
- Sel, Ç. The Use of Parametric Programming and Simulation-Optimisation Approaches for Stochastic Inventory Control in the Food Markets under Fuzzy Deterioration Rate. *Comput. Ind. Eng.* **2022**, *168*, 108141. [CrossRef]
- Ławryńczuk, M. Input Convex Neural Networks in Nonlinear Predictive Control: A Multi-Model Approach. *Neurocomputing* **2022**, *513*, 273–293. [CrossRef]
- Tatjewski, P.; Ławryńczuk, M. Soft Computing in Model-Based Predictive Control. *Int. J. Appl. Math. Comput. Sci.* **2006**, *16*, 7–26.
- Assawawongmethee, W.; Laosiritaworn, W. Application of Neural Networks in Perishable Inventories Management. *Appl. Mech. Mater.* **2015**, 752–753, 1424–1429. [CrossRef]
- Bardeji, S.F.; Saghih, A.M.F.; Pooya, A.; Mirghaderi, S.H. Perishable Inventory Management Using GA-ANN and ICA-ANN. *Int. J. Procure. Manag.* **2020**, *13*, 347–382. [CrossRef]
- De Moor, B.J.; Gijsbrechts, J.; Boute, R.N. Reward Shaping to Improve the Performance of Deep Reinforcement Learning in Perishable Inventory Management. *Eur. J. Oper. Res.* **2022**, *301*, 535–545. [CrossRef]
- Chołodowicz, E.; Orłowski, P. Control of Perishable Inventory System with Uncertain Perishability Process Using Neural Networks and Robust Multicriteria Optimization. *Bull. Pol. Acad. Sci. Tech. Sci.* **2022**, *70*, 141182. [CrossRef]

17. Chłodowicz, E.; Orłowski, P. Impact of Control System Structure and Performance of Inventory Goods Flow System with Long-Variable Delay. *Elektron. Elektrotechnika* **2018**, *24*, 11–16. [[CrossRef](#)]
18. Kastius, A.; Schlosser, R. Dynamic Pricing under Competition Using Reinforcement Learning. *J. Revenue Pricing Manag.* **2022**, *21*, 50–63. [[CrossRef](#)]
19. Li, J.; Yang, A.; Dai, W.; Pan, H. On Multiplex Combination Forecasting Model Based on GA and ANN. In Proceedings of the 26th Chinese Control Conference, CCC 2007, Zhangjiajie, China, 26–31 July 2007; pp. 251–255. [[CrossRef](#)]
20. Sarkar, A.; Guchhait, R.; Sarkar, B. Application of the Artificial Neural Network with Multithreading Within an Inventory Model Under Uncertainty and Inflation. *Int. J. Fuzzy Syst.* **2022**, *24*, 2318–2332. [[CrossRef](#)]
21. Huber, J.; Stuckenschmidt, H. Intraday Shelf Replenishment Decision Support for Perishable Goods. *Int. J. Prod. Econ.* **2021**, *231*, 107828. [[CrossRef](#)]
22. Shao, X.; Chang, D.; Li, M. Optimization of Lateral Transfer Inventory of Auto Spare Parts Based on Neural Network Forecasting. *J. Intell. Syst. Control.* **2022**, *1*, 2–17. [[CrossRef](#)]
23. Ahmadi, E.; Mosadegh, H.; Maihami, R.; Ghalehkhondabi, I.; Sun, M.; Süer, G.A. Intelligent Inventory Management Approaches for Perishable Pharmaceutical Products in a Healthcare Supply Chain. *Comput. Oper. Res.* **2022**, *147*, 105968. [[CrossRef](#)]
24. Gokhale, A.; Trasikar, C.; Shah, A.; Hegde, A.; Naik, S.R. A Reinforcement Learning Approach to Inventory Management. *Adv. Intell. Syst. Comput.* **2021**, *1133*, 281–297. [[CrossRef](#)]
25. Mohamadi, N.; Niaki, S.T.A.; Taher, M.; Shavandi, A. An Application of Deep Reinforcement Learning and Vendor-Managed Inventory in Perishable Supply Chain Management. *Eng. Appl. Artif. Intell.* **2024**, *127*, 107403. [[CrossRef](#)]
26. Mo, D.Y.; Tsang, Y.P.; Wang, Y.; Xu, W. Online Reinforcement Learning-Based Inventory Control for Intelligent E-Fulfilment Dealing with Nonstationary Demand. *Enterp. Inf. Syst.* **2023**, 2284427. [[CrossRef](#)]
27. Boute, R.N.; Gijbrecchts, J.; van Jaarsveld, W.; Vanvuchelen, N. Deep Reinforcement Learning for Inventory Control: A Roadmap. *Eur. J. Oper. Res.* **2022**, *298*, 401–412. [[CrossRef](#)]
28. Patriarca, R.; Di Gravio, G.; Costantino, F.; Tronci, M. EOQ Inventory Model for Perishable Products under Uncertainty. *Prod. Eng.* **2020**, *14*, 601–612. [[CrossRef](#)]
29. Chłodowicz, E.; Orłowski, P. Development of New Hybrid Discrete-Time Perishable Inventory Model Based on Weibull Distribution with Time-Varying Demand Using System Dynamics Approach. *Comput. Ind. Eng.* **2021**, *154*, 107151. [[CrossRef](#)]
30. Chłodowicz, E.; Orłowski, P. Robust Control of Perishable Inventory with Uncertain Lead Time Using Neural Networks and Genetic Algorithm. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*; 13350 LNCS; Springer International Publishing: Cham, Switzerland, 2022; pp. 46–59. [[CrossRef](#)]
31. Syntetos, A.A.; Boylan, J.E.; Croston, J.D. On the Categorization of Demand Patterns. *J. Oper. Res. Soc.* **2005**, *56*, 495–503. [[CrossRef](#)]
32. Rožanec, J.M.; Fortuna, B.; Mladenčić, D. Reframing Demand Forecasting: A Two-Fold Approach for Lumpy and Intermittent Demand. *Sustainability* **2022**, *14*, 9295. [[CrossRef](#)]
33. Ghobbar, A.A.; Friend, C.H. Sources of Intermittent Demand for Aircraft Spare Parts within Airline Operations. *J. Air Transp. Manag.* **2002**, *8*, 221–231. [[CrossRef](#)]
34. Makridakis, S.; Spiliotis, E.; Assimakopoulos, V. M5 Accuracy Competition: Results, Findings, and Conclusions. *Int. J. Forecast.* **2022**, *38*, 1346–1364. [[CrossRef](#)]
35. Chłodowicz, E.; Orłowski, P. Switching Robust Neural Network Control of Perishable Inventory with Fixed Shelf Life Products under Time-Varying Uncertain Demand. *J. Comput. Sci.* **2023**, *70*, 102035. [[CrossRef](#)]
36. Ignaciuk, P.; Bartoszewicz, A. Dead-Time Compensation in Continuous-Review Perishable Inventory Systems with a Remote Supply Source. *Arch. Control Sci.* **2011**, *21*, 61–84. [[CrossRef](#)]

**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.