International Journal of Information Systems and Supply Chain Management
Volume 17 • Issue 1

Research on Optimization Strategies for Closed-Loop Supply Chain Management Based on Deep Learning Technology

Chunjuan Gao, Nanjing Xiaozhuang University, China*

ABSTRACT

This study explores the integration of deep learning (DL) technology and the guided simulated annealing algorithm (GSAA) to optimize closed-loop supply chains (CLSC) for sustainable development. By applying DL for predictive analysis and GSAA for optimization, the research aims to enhance CLSC operational efficiency and environmental sustainability. The methodology combines a review of the CLSC framework with practical applications of DL and GSAA, aiming to reduce waste, maximize resource utilization, and minimize environmental impact. An experimental comparison of this approach against traditional optimization strategies demonstrates the proposed method's superior effectiveness and efficiency. The findings reveal that the DL-GSAA optimization significantly improves CLSC sustainability and efficiency, with GSAA showing promising convergence properties. This study underscores the importance of advanced technological solutions in achieving sustainable supply chain management, offering practical insights for businesses and supply chain managers.

KEYWORDS

Closed-Loop Supply Chain, Deep Learning Technology, Gated Recurrent Unit, Genetic Algorithm, Simulated Annealing Algorithm

INTRODUCTION

The conventional supply chain, while foundational to global commerce, increasingly falls short in addressing critical challenges such as resource scarcity and environmental degradation. This shortfall impedes sustainable development, necessitating innovative approaches to reconcile economic growth with ecological stewardship. The closed-loop supply chain (CLSC) has emerged as a promising solution, offering pathways to mitigate environmental impact while fostering economic development and promoting harmony between human activities and natural ecosystems. In the realm of enterprise development, the rapid pace of scientific advancement exacerbates the challenge, leading to frequent product updates that contribute to resource waste and environmental harm. This dynamic places enterprises at the crossroads of economic and environmental objectives, driving a shift towards CLSC practices to achieve a balance. Notable contributions in this field include the work of Bressanelli et

DOI: 10.4018/IJISSCM.341802
*Corresponding Author

al. (2019), who explored the economic aspects of CLSC management and proposed a quantitative model for recycling and reusing waste products, providing insights into reducing pollution from product production. Similarly, Kazancoglu et al. (2021) detailed the structure and evolution of CLSC, highlighting its advantages over traditional models and laying out the guiding principles for its design. Recognizing the potential of CLSC to address these pressing issues, this study delves into an exhaustive examination of CLSC, beginning with an analysis of its core principles and unique characteristics. Building on this foundation, it introduces an innovative application of deep learning algorithms to refine CLSC management. Specifically, it develops a predictive model, employing the Multi-Head Attention Gated Recurrent Unit (MAGRU) algorithm, a novel approach in this context. Furthermore, it harnesses the power of the genetic simulated annealing algorithm for simulation purposes, achieving an optimal solution with remarkable convergence properties. This methodology not only underscores the viability of CLSC in mitigating environmental impacts but also showcases the potential for advanced computational techniques to revolutionize enterprise development in China, offering new perspectives for future research and implementation.

LITERATURE REVIEW

As the global trade competition becomes further intensified, Supply Chain Management (SCM) technology has become critical to maintaining competitive advantages for enterprises. Two Deep Reinforcement Learning (DRL) based methods are proposed to solve multi-period capacitated supply chain optimization problems under demand uncertainty (Peng et al., 2019). Intuition-based approaches are replaced by supply chain computerized solutions such as inventory management, warehousing, allocation, and replenishment. Hachaïchi et al. (2020) aim at building a reinforcement learning agent capable of placing optimal orders for the sake of constructing a replenishment plan for next period. Current supply chain efficiency management methods cannot effectively control the risk caused by inefficient supply chain management. In order to study improvement in supply chain efficiency management supported by machine learning and neural network technology, Han and Zhang (2020) built a supply chain risk management model based on learning and neural networks. However, the cost control ability of the model is poor. For this reason, Guan and Yu (2021) designed a supply chain resource distribution allocation model based on deep learning. Huang and Tan (2021) introduced the strategy research of supply chain management order based on a reinforcement learning algorithm. The supply chain order management process involves conducting questionnaire surveys and seminars to understand the current process of supply chain order management and the problems derived from the analysis of data based on the deep learning algorithm.

Supply chain management and communication are a key research direction in the IoT environment, while inventory management (IM) has increasingly become a core part of the whole life cycle management process of the supply chain. The IM process is firstly formulated as a mathematical model, in which the objective is to minimize logistic cost and jointly maximize profit. On this basis, a deep inventory management (DIM) method is proposed to address this model by using the long short-term memory (LSTM) theory of deep learning (DL) (Deng & Liu, 2021). Kegenbekov and Jackson (2021) demonstrate how a deep reinforcement learning agent based on the proximal policy optimization algorithm can synchronize inbound and outbound flows and support business continuity operating in the stochastic and nonstationary environment if end-to-end visibility is provided. The deep reinforcement learning agent is built upon the Proximal Policy Optimization algorithm, which does not require hardcoded action space and exhaustive hyperparameter tuning. The manufacturer and retailer are two supply chain players, where the retailer is unreliable and may not send accurate demand information to the manufacturer. Sardar et al. (2021) propose a Machine Learning (ML) approach for on-demand forecasting under smart supply chain management. Guo and Zou (2022) introduce deep learning neural network to cross-border logistics and supply chain based on the analysis of the existing cross-border logistics model and supply chain model and the status quo of e-commerce development. They show that introducing deep learning neural networks into CBEC logistics and supply chain can improve the efficiency of logistics and supply chain. Long et al. (2023) use artificial intelligence systems to make intelligent decisions for supply chain mode selection in healthcare. This paper presents the intelligent choice optimization method of supply chain mode in healthcare based on deep reinforcement learning algorithm.

MATERIALS AND METHODS

DL Technology

Deep Learning (DL) falls under the extensive domain of Machine Learning (ML) and stands as one of the most widely embraced methodologies. It involves feature learning due to its structured nature and is also referred to as featureless supervised learning (Garcia-Buendia et al., 2021). DL operates on a neural network, exhibiting greater complexity compared to traditional ML. Its processes are more intricate than those of traditional algorithms, and the operational principles diverge from those of conventional algorithms. Nonetheless, the learning outcomes yielded by the DL model significantly surpass those of traditional algorithms, with notably because of faster execution times (Reimann et al., 2019). Initially limited to image processing, DL has undergone continuous advancement and finds application in various domains such as face recognition, smart home technology, environmental surveys, and beyond. Its growing integration into human activities and society has led to profound impacts (Su et al., 2019). Figure 1 illustrates the parallels between DL and traditional machine learning. The advantages and disadvantages of DL are exhibited in Table 1.

Convolutional Neural Network (CNN)

Convolutional Neural Networks (CNNs) represent one of the most extensively employed neural network architectures, excluding the DL network, which builds upon the foundations of traditional

Figure 1. Comparative analysis of deep learning (DL) and traditional machine learning (ML) techniques

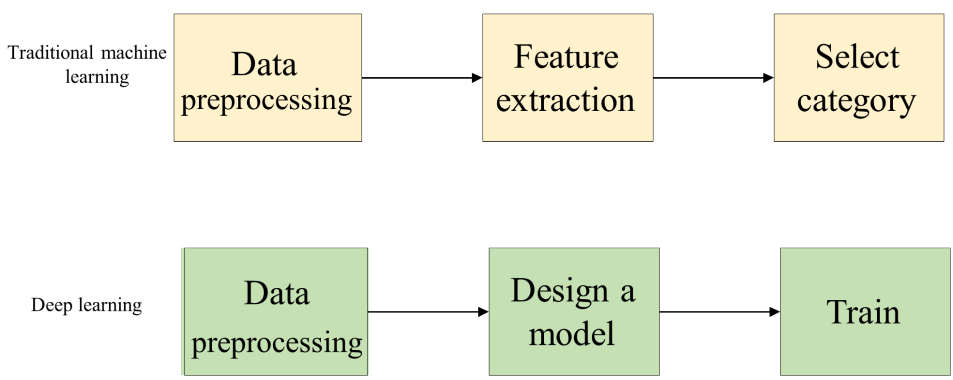


Table 1. Advantages and Disadvantages of DL

Advantages	Disadvantages
Strong learning ability	Large amount of computation and poor portability
Wide coverage and good adaptability	High hardware requirements
Data-driven, high ceiling	Complex model design
Good portability	Prone to bias

networks (Abdi et al., 2021). The CNN structure follows a systematic and attainable design, comprising an input layer, convolutional layer, pooling layer, and output layer. The algorithm operates by feeding the dataset into the CNN, performing data calculations, and producing results through multiple steps, including pooling. Its application domain primarily revolves around image processing, where it has made significant contributions to environmental monitoring and medical fields. Furthermore, CNN can be synergistically integrated with diverse algorithms to process data and achieve desired outcomes, boasting exceptional recognition rates, accuracy, and robustness (Giri & Masanta, 2020) (see Figure 2).

Recurrent Neural Network (RNN)

In contrast to CNN, which processes image data as input, RNN takes sequential data as input, establishing connections between network nodes based on the sequence (Su & Sun, 2018). RNN is iterative, possessing comprehensive parameters and the ability to retain historical datasets accurately, rendering it superior to alternative methods for sequence processing (Zare Mehrjerdi & Lotfi, 2019). Beyond its prowess in image recognition, RNN consistently demonstrates efficacy in handling textual and video information. The RNN model is depicted in Figure 3.

Auto-Encoder (AE)

The principle of the AE neural network is to use backpropagation to make the input value equal to the output value, that is, $y(i) = x(i)$, which belongs to the same unsupervised learning algorithm as

Figure 2. Architecture of the convolutional neural network (CNN) model

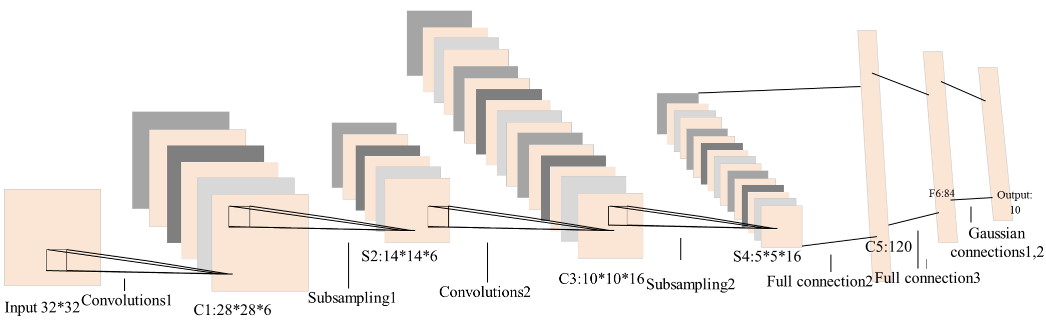
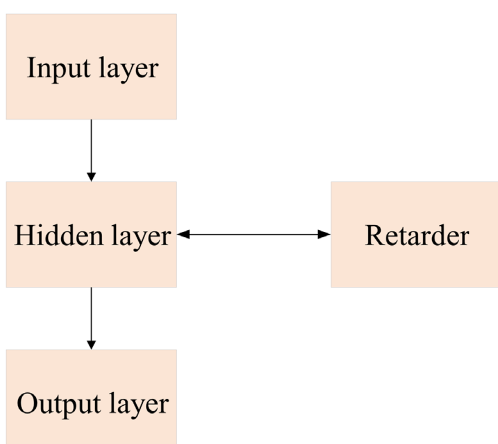


Figure 3. Structure of the RNN model



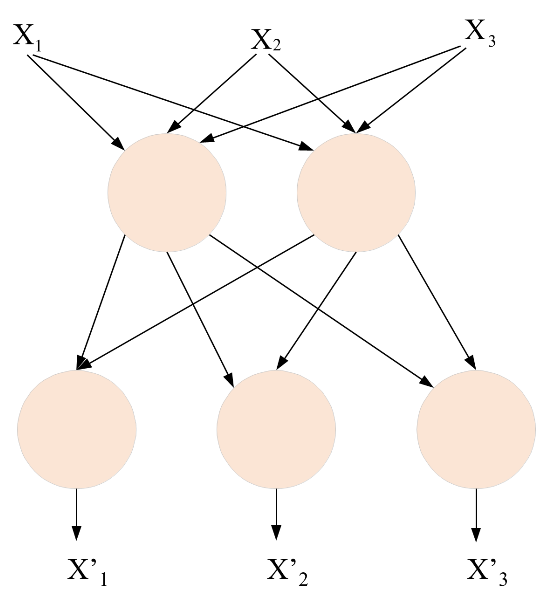
DL. The composition of the AE neural network has two parts, namely the encoding part and the decoding part (Prakash et al., 2020). AE is usually used in data processing, which can compress data for subsequent calculation (Goli et al., 2020). It has the characteristics of fast calculation speed and high robustness. An example of an AE neural network is shown in Figure 4.

When applying deep learning to supply chain management, the research leverages the advantages of RNN in time series prediction, leading to the introduction of the Gated Recurrent Unit (GRU) algorithm in this research. Given an input sequence $X = \left\{x_1, x_2, \ldots, x_t\right\}$, the encoder is used to learn the mapping from *x* to h_t, where *t* denotes the time step, and the calculation is as shown in Equation 1:

$$h_t = f\left(h_{t-1}, x^t\right) \tag{1}$$

Equation 1 represents the calculation used to update the hidden layer state h_t of the encoder in the research on applying deep learning to supply chain management. The hidden layer state, represented by h_t, captures the encoded information at time step *t*. It is updated based on the previous hidden layer state h_{t-1} and the current input data x^t. The mapping function $f\left(h_{t-1}, x^t\right)$ determines how the hidden layer state is updated using the previous state and the current input. In this research, the GRU algorithm is chosen as the mapping function. GRU is a type of RNN that is specifically designed to capture temporal dependencies in sequential data. The use of GRU as the mapping function allows the model to capture and learn the temporal dependencies present in the input sequence. Temporal dependencies refer to the relationship between data points at different time steps and are crucial for accurate time series predictions in supply chain management. By utilizing the GRU algorithm as the mapping function in Equation 1, this research aims to leverage the advantages of deep learning and RNNs in capturing the temporal patterns and predicting future outcomes in the context of supply chain management. The GRU's ability to model long-term dependencies and handle vanishing or

Figure 4. Architecture of the autoencoder (AE) neural network model



exploding gradients makes it suitable for analyzing time series data and making predictions based on historical trends.

Each GRU unit is governed by reset and update gates, and their updating algorithms are expressed in Equations 2 and 3:

$$R\_t = \sigma\left(X_t W_{xr} + h_{t-1} W_{hr} + b_r\right) \tag{2}$$

$$Z_t = \sigma\left(X_t W_{xz} + h_{t-1} W_{hz} + b_z\right) \tag{3}$$

In Equations 2 and 3, *h* denotes the number of hidden units. $x_t \in R^{n \times d}$ represents a batch of inputs at a given time step *t* with *n* samples and *d* input dimensions. h_{t-1} represents the hidden state from the previous step, R_t represents the reset gate, and Z_t represents the update gate. W_{xr}, W_{hr}, W_{xz}, and W_{hz} are weight parameters that the model needs to learn, while b_r and b_z are bias parameters. Finally, the sigmoid function is applied to constrain the values to the range [0,1]. By incorporating the reset and update gates, the GRU units are equipped to update the hidden state adaptively based on the current input and the previous hidden state, allowing the model to capture long-term dependencies and effectively handle sequential data in the context of supply chain management. The hidden state at time step *t* is given by Equations 4 and 5:

$$R_t = \sigma\left(X_t W_{xr} + h_{t-1} W_{hr} + b_r\right) \tag{4}$$

$$Z_t = \sigma\left(X_t W_{xz} + h_{t-1} W_{hz} + b_z\right) \tag{5}$$

Dealing with lengthy sequences poses challenges for traditional RNN or GRU models due to issues such as vanishing or exploding gradients, impeding the capture of long-range dependencies. To address this, the Multi-Head Attention (MA) mechanism allows the model to focus on information from various positions, mitigating these problems and facilitating superior capture of long-distance dependencies in sequences. Consequently, the MAGRU algorithm is introduced, incorporating the MA mechanism. By concurrently attending to information from diverse positions in sequence data, MA aids in capturing global context and correlations. It thereby enhances the performance of the GRU model, particularly in managing long-range dependencies and crucial sequence information.

In the MAGRU algorithm, given a sequence of length *q*, MAGRU utilizes the hidden state and the current state of the GRU unit to establish an attention mechanism, as shown in Equation 6:

$$s_t^q = g_l^T \tan h\left(W_l\left(h_{t-1}; s_{t-1} + U_l x_{i,q} + b_l\right)\right) \tag{6}$$

In Equation 6, g_l , W_l, U_l represent parameters that need to be learned, and b_l denotes bias values. The calculation of attention weights is given by Equation 7:

$$\omega_j^t = \frac{exp\left(s_{t,j}\right)}{\sum_{q=1}^{T} exp\left(s_{t,q}\right)} \tag{7}$$

Equation 7 defines the calculation of importance weights ω_j^t for the *j*-th input sequence at time step *t* using an attention mechanism. ω_j^t represents the importance weight assigned to the *j*-th input sequence at time step *t*. The importance weights are calculated using the softmax function, which transforms the results obtained from the attention mechanism into probability values. By assigning weights to different input sequences, the model can focus on relevant information and emphasize certain parts of the input data during the encoding process. The softmax function normalizes the scores $s_{t,j}$ obtained from the attention mechanism by exponentiating them and dividing by the sum of exponentiated scores across all input sequences at time step *t*. This normalization process ensures that the importance weights sum up to 1, making them interpretable as probabilities.

Each batch of data is assigned a batch of samples *p* based on their weights, and these batches of samples form the input sequence to the GRU. The sequence can be encoded based on importance through these weights, and the final output after the encoder is given by Equation 8:

$$x_t^{'} = \left(\omega_t^1 x_t^1, \omega_t^2 x_t^2, \ldots, \omega_t^n x_t^n\right)^T \tag{8}$$

After calculating the importance weights, the input sequences are combined based on these weights to form a weighted input sequence $x_t^{'}$ that captures the essential information. Each input sample is multiplied by its corresponding importance weight and aggregated to create the encoded representation of the input data at time step *t*. This weighted encoding approach allows the model to focus on relevant information while considering the importance of each input sequence in the context of the entire batch.

To mitigate information loss and improve predictive performance, a decoder is employed after the encoding process. The decoder takes the encoded data as input and generates predictions for the target variables, leveraging the encoded information to make accurate forecasts or decisions in supply chain management scenarios. By incorporating attention mechanisms and weighted encoding, the model can effectively leverage relevant information from input sequences, address issues related to information loss and sequence length limitations, and enhance predictive capabilities in supply chain management tasks.

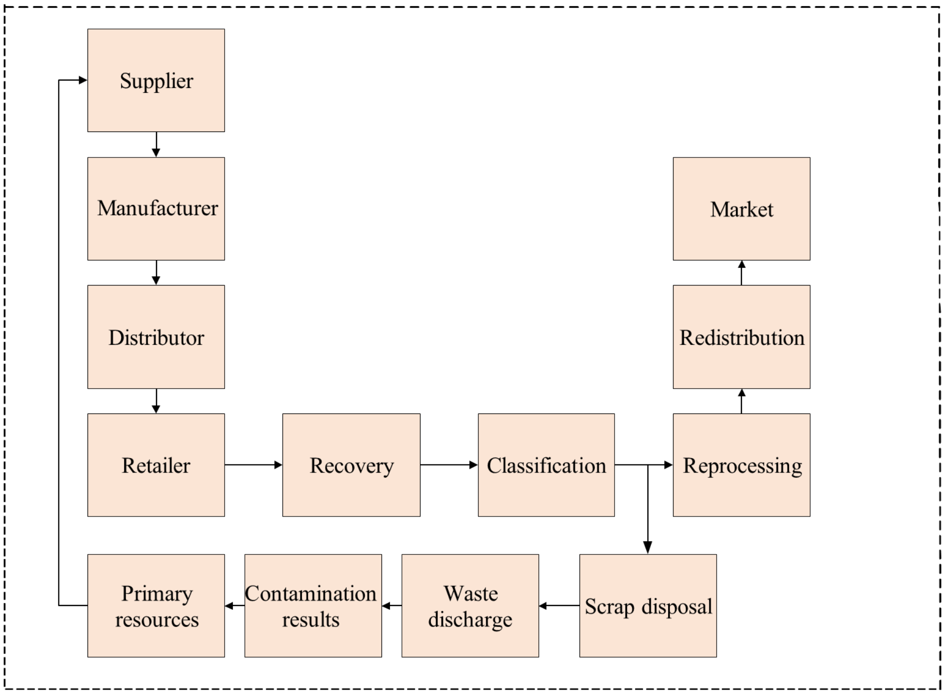
Analysis and Construction of a Closed Loop Supply Chain Management Prediction Model Based on MAGRU

CLSC comprises two parts: the forward supply chain and the reverse supply chain. The former focuses on reducing production costs, improving efficiency, and minimizing waste while meeting production requirements. It involves the process of delivering products to consumers without causing resource waste. In contrast, the reverse supply chain involves recovering products from consumers, reprocessing them, and disposing of them in an environmentally friendly and cost-effective way. Both chains have the common purpose of protecting the environment and achieving economic and environmental benefits, which matches the characteristics of a traditional supply chain. However, CLSC has its own features that require advanced planning, consideration of recycling methods, environmental impact, reprocessing processes, and cost factors. More significant responsibility is involved in protecting the environment while ensuring economic benefits, as the recycling of waste resources has a profound influence on human living conditions.

The structure of CLSC consists of several components; the forward and reverse supply chains are the most crucial. According to Zailani et al. (2020), the value of the reverse supply chain is more significant, and it requires more effort to ensure the secondary utilization of resources, reduce environmental pollution, and achieve environmentally friendly production. Figure 5 represents the structure of CLSC.

CLSC has five different models: “Retailer sells and recycles, Retailer sells but third party recycles, Retailer sells but manufacturer recycles, Manufacturer sells directly and recycles, Manufacturer sells to 3rd party recycling,” as shown in Figure 6.

Figure 5. Structure of a closed-loop supply chain (CLSC)



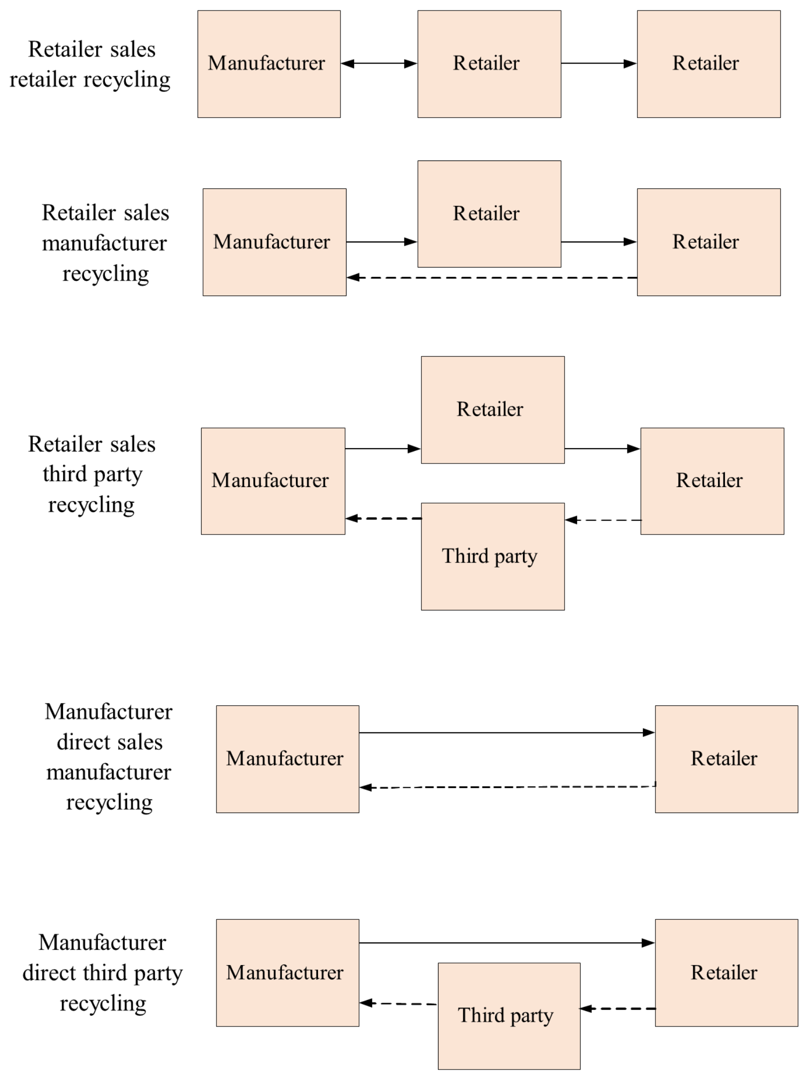
The construction of the CLSC management prediction model based on MAGRU involves several primary steps. These steps include preparing time series data, which encompasses crucial information necessary for effective supply chain management such as production quantities, inventory levels, and demand changes. To capture long-distance dependencies and global context information in sequence data, an innovative encoder is established using the MAGRU algorithm, incorporating a MA into the GRU unit (Wu & Zhou, 2019). The output of this encoder is then passed to the decoder, which resolves the issues of information loss and excessively long sequences resulting from encoding information into a fixed-length vector. Consequently, the predictive performance of the model is enhanced. During the model training phase, the constructed MAGRU model is adjusted using historical data to meet the specific requirements of supply chain management. This trained model accurately predicts future supply chain data, facilitating the formulation of effective management strategies, optimization of resource allocation, and improvement of supply chain efficiency and sustainability. The key innovation lies in the integration of the MAGRU algorithm with the decoder, which overcomes the limitations of traditional models in handling long sequences and complex dependencies. Thus, this provides a novel predictive model for CLSC management. The algorithmic flow of applying the MAGRU algorithm to CLSC management prediction is illustrated in Algorithm 1.

Coordination of CLSC Contract

The efficient operation of a supply chain relies on predetermined principles that ensure the interests of both suppliers and sellers, promote transparency, and aim for a win-win situation (Tang et al., 2020). To achieve this, supply chain contracts play a crucial role in coordinating the efforts and benefits of each party. There are four fundamental types of contracts in a supply chain:

1. The wholesale price contract involves the retailer determining the wholesale quantity based on customer demand, negotiating the wholesale price with the customer, and informing the supplier

Figure 6. Five models of closed-loop supply chains (CLS)



of the price and quantity. Suppliers produce and sell to retailers, and the retailer is responsible for handling any unsalable products (Modak & Kelle, 2021).

2. In the repurchase contract, if the retailer has unsalable products, the supplier agrees to repurchase them at a negotiated price, providing a safeguard for the retailer. This contract also allows retailers to increase the quantity when purchasing products, achieving mutual benefit (Mirzaei et al., 2023).
3. A revenue sharing contract enables the supplier and retailer to share profits and revenues by reducing the product's price. This contract ensures a proportional sharing of revenue between retailers and suppliers, fostering a mutually beneficial relationship.
4. The quantity flexible contract involves the retailer predicting the number of products that can be sold and placing orders accordingly. If the predetermined amount is insufficient to meet customer demand, the supplier increases production to fulfil the retailer's needs (Rajabzadeh Gatari et al., 2021).

Algorithm 1. MAGRU algorithm applied to CLSC management prediction algorithm

1	Start
2	Input: Supply chain timing data such as production quantity, inventory level, and demand changes
3	Output: Prediction results of CLSC management
4	# Build the MAGRU model
5	def build_magru_model(input_shape, output_shape, num_heads=4, gru_units=64):
6	# Encoder part
7	inputs = tf.keras.Input(shape=input_shape, name='input_sequence')
8	gru_output, gru_state = GRU(gru_units, return_sequences=True, return_state=True)(inputs)
9	attention_output = MultiHeadAttention(num_heads=num_heads)([gru_output, gru_output])
10	# Decoder part
11	decoder_input = tf.keras.Input(shape=(output_shape[0], output_shape[1]), name='decoder_input')
12	decoder_gru_output = GRU(gru_units, return_sequences=True)(decoder_input)
13	decoder_attention_output = MultiHeadAttention(num_heads=num_heads)([decoder_gru_output, decoder_gru_output])
14	# Combine encoder and decoder outputs
15	combined_output = tf.concat([attention_output, decoder_attention_output], axis=-1)
16	# Output layer
17	outputs = Dense(output_shape[1], activation='linear')(combined_output)
18	model = tf.keras.Model(inputs=[inputs, decoder_input], outputs=outputs, name='magru_model')
19	return model
20	# Compile and train the model (using an appropriate optimizer, loss function, and training data)
21	End

Effective CLSC management balances economic interests and environmental protection, leading to sustainable social development. In the current challenging economic environment, achieving economic development extends beyond competition between individual enterprises and has transitioned into comparative competition between supply chains. Implementing effective supply chain management relies on well-designed supply chain contracts that maximize benefits for all stakeholders (Mishra et al., 2018).

However, implementing CLSC is challenging due to complex structures and incomplete information. Each member of the supply chain has their own selfish desires, making the practice process difficult and the management of CLSC complicated. To ensure operational efficiency and to meet overall interests, it is crucial to establish appropriate supply chain management measures, such as supply chain contracts (Zhao & Sun, 2020). These contracts protect the rights and interests of each member while also setting boundaries on their behavior. By strictly adhering to the contract and regulating their actions, members maintain order and guarantee enterprise interests on a small scale while preserving ecological balance on a larger scale, resulting in numerous benefits.

Supply Chain Emergencies and Emergency Management

In the 21st century, emergencies such as natural disasters, accidents, social security incidents, and public health crises have had a severe impact on China, affecting the stability of society and economic development (Hajiaghaei-Keshteli et al., 2018). These emergencies have led to various consequences, impacting human safety and the functioning of supply chain systems (Ashtab & Tosarkani, 2023).

Supply interruptions, operational disruptions, and sudden changes in demand represent the risks associated with these emergencies (Ghasemi & Abolghasemian, 2023). Emergency strategies within supply chain enterprises are designed to address different levels of operational and supply disruptions (Liu et al., 2023). When facing the risk of sudden demand changes, vertical coordination among enterprises is an effective emergency strategy, while risks faced by upstream and downstream enterprises in the supply chain can be mitigated through emergency contracts.

The significant impact of emergencies on enterprises and supply chain management has been evident in events such as SARS, the Sanlu milk powder incident, and the Qinghai Yushu earthquake. These emergencies have led to supply chain imbalances, including disruptions to logistics and capital flow, interrupted information sharing, production-sales mismatches, drastic shifts in demand, raw material shortages, increased production costs, fluctuating social demand, and damage to goods and services. Addressing emergencies in the supply chain system has become a crucial area of interest due to its practical significance and guiding role in safeguarding human lives, property, and consumer rights.

While the probability of sudden emergencies is low, their occurrence can lead to catastrophic consequences, disrupting the flow of resources within the supply chain and affecting trade between nodes (Pan & Miao, 2023). The disruption caused by emergencies can lead to significant losses for individual enterprises and the entire supply chain system (Kim et al., 2023). As the frequency of emergencies continues to rise, improving the ability of supply chain members to manage emergencies has become an essential and valuable research focus. Effective emergency management aims to control emergencies by integrating the resources of all parties, improving the ability to foresee, respond to, and recover from emergencies, and minimizing losses. Coordination mechanisms within the supply chain are adjusted to address market size changes resulting from emergencies, ensuring supply chain system coordination and minimizing disruptions.

Applying coordinated contracts in the context of emergency situations within the CLSC greatly enhances the ability of each node enterprise to manage emergencies and adapt to the evolving production environment. This approach also fosters better coordination of member benefits at all levels within the CLSC and ensures overall benefits for the entire supply chain during emergencies.

Genetic Algorithm

The Genetic Algorithm (GA) is a computational model designed and proposed based on the evolutionary laws found in nature. It simulates both the natural selection process described by Darwin's theory of biological evolution and the genetic mechanisms involved in biological evolution. By simulating the natural evolutionary process, the model aims to find the optimal solution. Through mathematical transformations, the algorithm used in biological evolution can be simulated in computer operations involving crossover and mutation similar to chromosomal genes. When addressing complex combinatorial optimization problems, this algorithm demonstrates faster computation and achieves superior optimization results compared to conventional algorithms. As a result, GA has extensive applications in solving various problems in the fields of combinatorial optimization, ML, signal processing, adaptive control, and artificial life.

In GA, the probability of selection is closely related to fitness. The higher the fitness, the greater the probability of being selected. The specific expression for the selection operator is given in Equation 9:

$$p_{i=\frac{f_i}{\sum_{i=1}^{n} f_i}} \tag{9}$$

Equation 9 describes the selection operator in a GA, where the probability (p_i) of selecting an individual, *i*, is proportional to an individual's fitness (f_i). Higher fitness values result in greater probabilities of being selected for reproduction or further processing. Fitness is a measure of how well an individual performs in the given problem domain. It indicates the quality or suitability of the individual's solution or phenotype. The probability of selection is calculated by dividing the fitness of individual *i* by the sum of fitness values across all individuals in the population $\sum_{i=1}^{n} f_i$. This normalization process ensures that the selection probabilities sum up to 1, as they represent probabilities.

The selection operator in GA aims to mimic natural selection, whereby individuals with higher fitness have a higher chance of being selected for reproduction and passing their genetic material to subsequent generations. By assigning selection probabilities based on fitness, GA can bias the selection process towards better-performing individuals, promoting the convergence towards optimal solutions over successive generations.

The fitness function *Fit* can be expressed as shown in Equation 10:

$$Fit = \frac{p_1 + p_2}{p_{all}} \tag{10}$$

Equation 10 describes the fitness function for a population of species in a genetic algorithm, where the fitness (*Fit*) is defined as a ratio of probabilities. *Fit* represents the fitness value associated with a population of a species in the GA. The fitness function evaluates the quality of solutions or individuals in the population and guides the evolutionary process towards better-performing individuals. p_1 and p_2 represent the probability values associated with two different species in the population, typically the parent and offspring in a generational GA. p_{all} refers to the sum of probability values across all species in the population. The fitness value is calculated by taking the sum of two probability values and dividing by the total probability across all species in the population. This ratio-based approach ensures that the fitness value ranges from 0 to 1, with higher values indicating better-performing solutions. The fitness function in GA plays a crucial role in evaluating the quality of solutions and guiding the search towards optimal solutions.

In Equation 10, p_1 and p_2 refer to the initial fitness value and the fitness value of the new generation of species, respectively. p_{all} represents the total number of records in the sample dataset. Further calculations include the self-adaptive mutation rate p_m and the crossover rate p_c, as illustrated in Equations 11 and 12:

$$p_{m=pm_1} + \left(pm_2 - pm_1\right) * \frac{popsize - n}{popsize} \tag{11}$$

$$p_c = pc_1 - \left(pc_2 - pc_1\right) * \frac{popsize - n}{popsize} \tag{12}$$

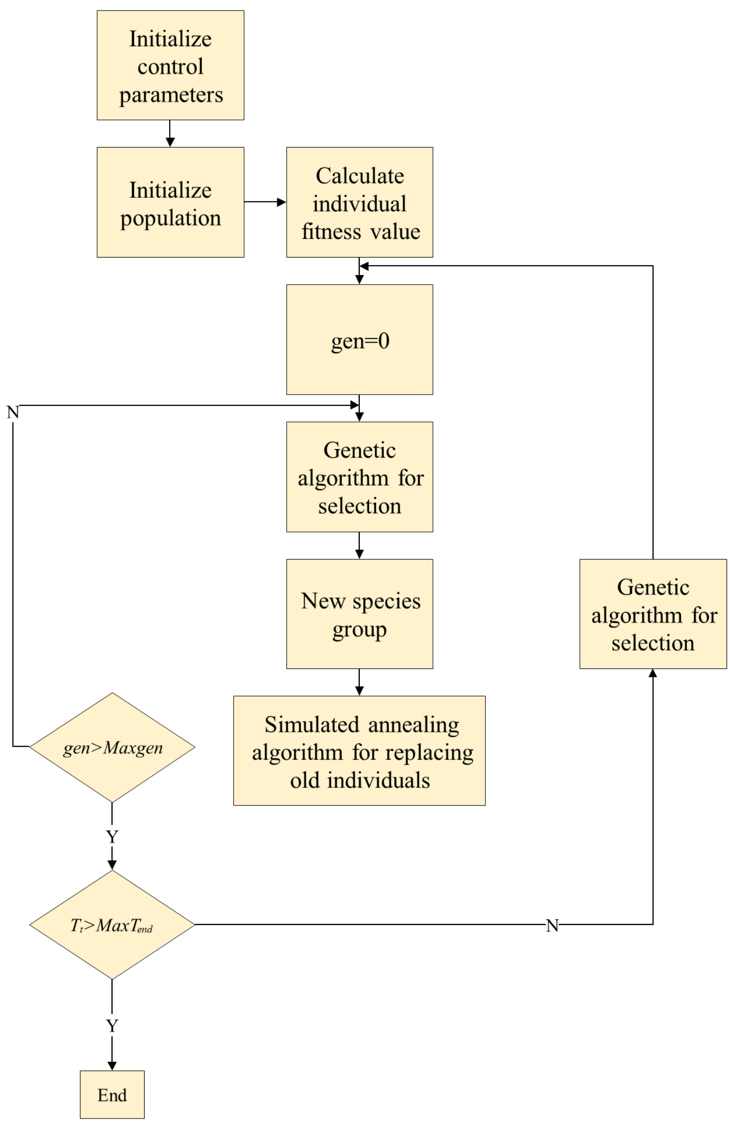
In Equations 11 and 12, pm_1 and pm_2 represent the maximum and minimum self-adaptive mutation rates, while pc_1 and pc_2 represent the maximum and minimum crossover rates. The parameter *n* denotes the number of individuals. The variations in *n* and *pc* accompany the algorithm's iteration process. The changes in *pm* and *pc* are determined based on the variations in *n*.

Simulated Annealing Algorithm (SAA)

The basic principle of solid annealing serves as the foundation for SAA, a fundamental probability algorithm. This principle involves heating the solid's temperature and then gradually cooling it down. As the temperature rises during heating, the particles within the solid become disordered, leading to an increase in internal energy. Conversely, during the slow cooling process, the particles reorganize, reaching an equilibrium state at each temperature. Eventually, at normal temperature, the particles reach the ground state, resulting in the lowest internal energy.

The combination of GA and SAA enhances the accuracy of the algorithm while addressing the premature convergence issue of GA and the low robustness of SAA. Additionally, a fitness function is designed to facilitate the algorithm in obtaining the optimal solution. The specific algorithmic flow is depicted in Figure 7.

Figure 7. Flow of GA and SAA



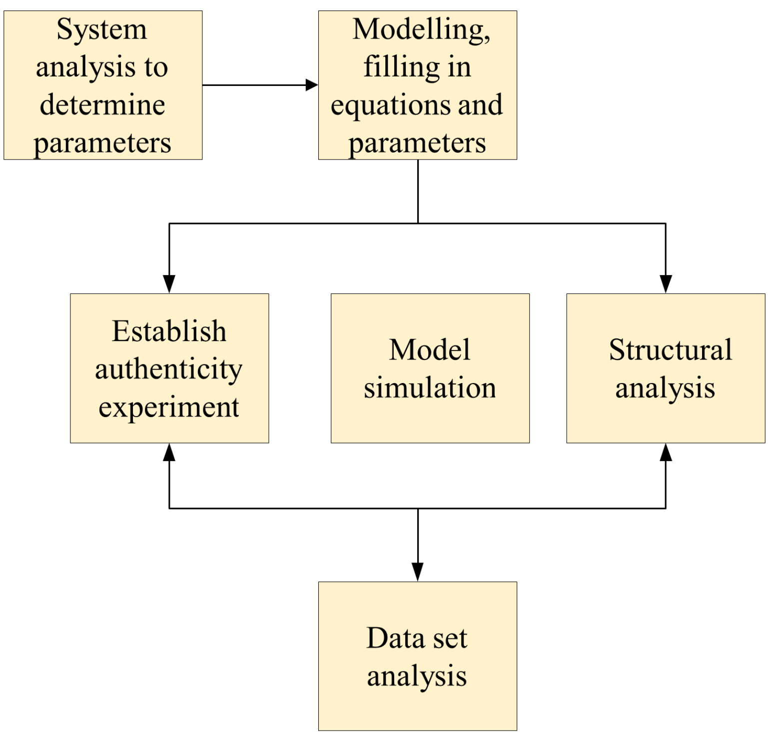
The specific algorithm flow is as follows:

1. Algorithm parameters are initialized: population size, number of iterations, crossover and mutation probability, and temperature.
2. The fitness value corresponding to each group is calculated to determine the decision variable of each group.
3. The initial value of the loop count variable is 0, that is, $gen = 0$.
4. SAA is used to select new individuals and to perform genetic manipulation and selection on new individuals.
5. It should be determined whether *gen* is less than the maximum number of iterations. If it is less than, then $gen = gen + 1$, one should go to step 4; otherwise, one should go to step 6.
6. It should be judged whether the temperature is lower than the final temperature. If it is lower, the algorithm ends, if it is not lower, one should start the cooling operation and turn to step 3.

VENSIM Simulation Software

The software used for the simulation is VENSIM PLE, which is a dynamics-based installation package that enables modeling operations, features visualization, optimizes and conceives models that have been built, and records data. It can establish a dynamic model, mark the parameters of each variable, analyze the relationship, and finally establish an equation and record it in the model. In the process of use, the parameters are continuously adjusted to make the model more stable and accurate. The general process of using software to deal with problems is demonstrated in Figure 8.

Figure 8. Utilizing software for problem-solving processes



RESULTS AND DISCUSSION

Parameter Settings of the CLSC Model

The parameters of the model are the basic parameters related to the recycling and remanufacturing of a product by a product recycling remanufacturer (see Table 2).

C_{in}: Inspection and Classification Cost of Recycled Products (Unit: \$5)

This parameter represents the cost associated with inspecting and classifying recycled products. It accounts for expenses incurred during the quality assessment and categorization of recycled items before further processing or remanufacturing. The value assigned to C_{in} is \$5, denoting the cost per unit of recycled product.

C_{re}: Cost of Direct Processing and Remanufacturing (Unit: \$5)

C_{re} indicates the cost involved in directly processing and remanufacturing recycled products. It encompasses expenses related to transforming recycled materials into finished goods or components. The assigned value for C_{re} is \$5, reflecting the cost per unit of direct processing and remanufacturing.

P_{rma}: Profit from Processing Into Recycled Raw Materials (Unit: \$6)

P_{rma} signifies the profit obtained from processing recycled products into recycled raw materials. It represents the financial gain achieved by converting recycled items into reusable materials that can be sold or utilized in subsequent manufacturing processes. The value assigned to P_{rma} is \$6, denoting the profit per unit of processed recycled raw material.

C_{rma}: Cost of Recycling and Processing Used as Raw Materials (Unit: \$4)

C_{rma} denotes the cost associated with recycling and processing used materials to obtain raw materials. It encompasses expenses incurred during the collection, sorting, and transformation of used items into raw materials suitable for manufacturing purposes. The assigned value for C_{rma} is \$4, representing the cost per unit of recycling and processing used as raw materials.

Fitness Simulation of GA

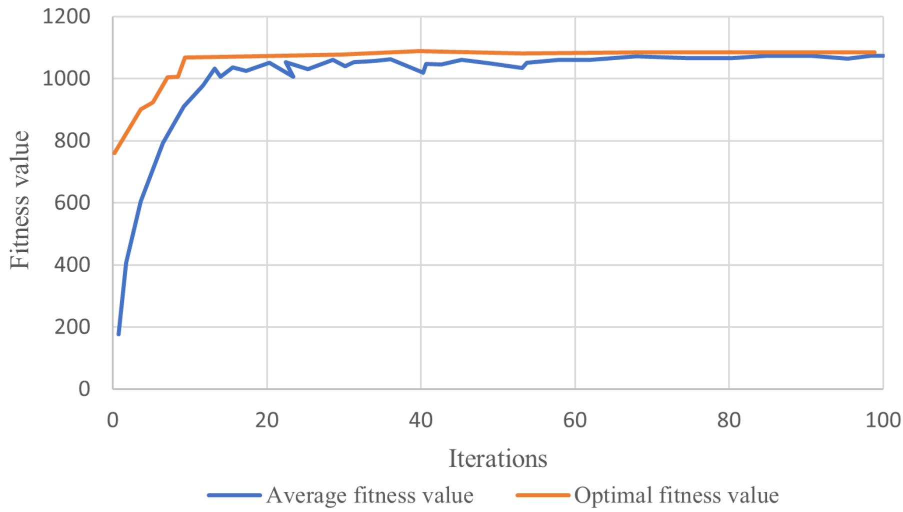
According to the specified parameters and GA, the fitness curve of the shown function can be obtained by programming.

Figure 9 visually represents the variation in fitness values throughout the iterative process of the GA as the number of iterations increases. This graphical depiction offers valuable insights into the algorithm's performance and efficiency. The plot compares the optimal fitness value with the average fitness value, emphasizing the convergence that occurs around the 15th iteration. This convergence indicates that the GA algorithm has identified a relatively excellent solution within a short timeframe. The sharp decrease in fitness values signifies the algorithm's ability to improve its performance quickly and approach the optimal solution. Moreover, the graph showcases that the algorithm can achieve satisfactory results with only a small number of iterations, highlighting its efficiency in finding solutions. The rapid convergence observed after 15 iterations illustrates the algorithm's effectiveness in optimizing its search process and reaching a desirable outcome promptly. The trend displayed by the optimal fitness value further confirms the GA algorithm's superiority

Table 2. Related basic parameters

C_{in}	5	C_{re}	5
P_{rma}	6	C_{rma}	4

Figure 9. Fitness curve of GA solution



in conducting a global search. This trend demonstrates the algorithm's capability to explore a wide range of potential solutions and converge towards the most optimal one efficiently.

Fitness Simulation of SAA

According to the specified parameters and SAA, the fitness curve of the expressed function can be obtained by programming.

Figure 10 visually represents the variation in fitness values throughout the iteration process of the SAA as the number of iterations increases. This graphical illustration provides valuable insights into the algorithm's performance. The plot compares the average fitness value with the optimal fitness value, highlighting the algorithm's effectiveness. Notably, when the number of iterations

Figure 10. Fitness curve of SAA solution

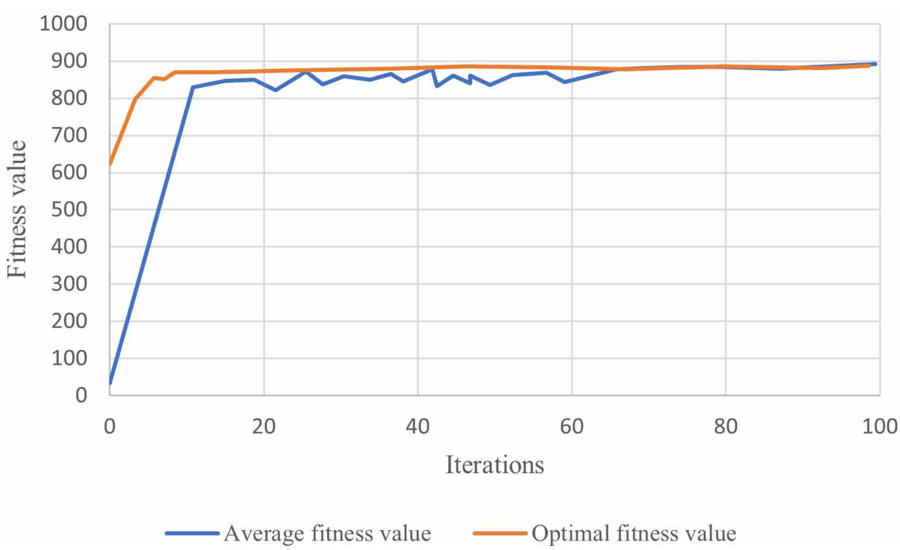
reaches 160, the optimal fitness value is attained, indicating that the algorithm has discovered the best solution within the current population. This finding is significant in evaluating the algorithm's ability to identify the most optimal solution. Additionally, an analysis of the difference between the average fitness value and the optimal fitness value may reveal a discernible trend. This observation becomes crucial in assessing the algorithm's global search capability and population diversity. By examining this trend, we can determine if the algorithm consistently improves its performance over time. Furthermore, the favorable convergence observed with a higher number of iterations suggests that the SAA algorithm can find excellent solutions within a relatively extended period. This characteristic is particularly valuable for problems that require comprehensive exploration of potential solutions. The trend displayed by the optimal fitness value further strengthens the algorithm's superiority in global search. This finding has practical implications, especially for problems that demand a globally optimal solution. In conclusion, Figure 10 provides a visual representation of the fitness curve of the SAA algorithm. The convergence observed at around 160 iterations indicates the algorithm's capability to discover the best solution within the population. The trend displayed by the optimal fitness value highlights the algorithm's effectiveness in conducting a global search, making it applicable to problems requiring a globally optimal solution.

Fitness Simulation of GSAA

According to the specified parameters and GSAA, the fitness curve of the displayed function can be obtained by programming.

Figure 11 visually represents the convergence of the fitness function as the number of iterations progresses in the GSAA. The graphical illustration offers key insights into the algorithm's convergence behavior and effectiveness. The plot indicates that the fitness function reaches a state of convergence around the 15th iteration. Despite the longer running time per iteration compared to other algorithms, the relatively large number of iterations contributes to a strong convergence effect. This convergence demonstrates the efficacy of the designed algorithm in consistently improving its solution quality over time. Additionally, Figure 11 highlights that the GSAA is capable of obtaining the optimal solution and the corresponding values of each decision variable. This capability sets it apart from other algorithms, showcasing its ability to

Figure 11. Fitness curve of GSAA solution



find the best possible solution for the given problem. While acknowledging that GSAA may have longer running times compared to alternative algorithms, it is essential to recognize that the primary objective is to achieve the optimal solution. The algorithm's capacity to fulfil this goal underscores its effectiveness and relevance in solving complex optimization problems. By incorporating visualizations of the convergence process, Figure 11 effectively communicates the algorithm's convergence behavior and its ability to deliver optimal solutions. The convergence analysis presented in the plot reinforces the algorithm's effectiveness and emphasizes its success in achieving the desired outcomes through a systematic and iterative approach.

CONCLUSION

The continuous advancement of science and societal progress has brought to light critical issues such as resource scarcity and environmental pollution, underscoring the urgent need to address these challenges. Achieving sustainable development has thus emerged as a paramount goal, presenting significant hurdles in the process. Given that the evolution of society is deeply intertwined with the dynamics of the supply chain, it is evident that traditional supply chain models fall short of addressing the requisites for sustainable development. Our research underscores the effectiveness of adopting a CLSC approach to safeguard the environment while promoting economic progress and ensuring a sustainable coexistence between humanity and nature. Through an in-depth exploration of CLSC's concept and defining features, this study introduces a management optimization strategy leveraging DL technology; the GSAA facilitates simulation and the derivation of optimal solutions. The algorithm's favorable convergence properties not only enhance the efficiency of CLSC management but also pave the way for groundbreaking advancements in the operational strategies of Chinese enterprises.

From a practical standpoint, the implications of our research for businesses and supply chain management are multifaceted. Firstly, the adoption of a CLSC model represents a strategic pivot towards sustainability, enabling companies to minimize waste, reduce environmental impact, and comply with increasing regulatory demands for sustainable practices. This shift not only contributes to environmental conservation but also offers a competitive edge in the marketplace through improved brand image and customer loyalty.

Secondly, the integration of DL technology and the application of the GSAA in optimizing CLSC operations underscore the potential for technological innovation to revolutionize supply chain management. By harnessing these advanced computational tools, businesses can achieve greater efficiency, agility, and resilience in their supply chain operations. This optimization strategy facilitates more accurate forecasting, inventory management, and resource allocation, thereby enhancing overall operational effectiveness and reducing costs.

However, the challenge of long running times associated with the proposed method highlights the need for continued innovation and improvement. Future research should focus on refining these computational techniques to ensure that they are more accessible and practical for widespread implementation across various industries. By addressing these limitations, the pathway to integrating advanced technologies into everyday business operations becomes clearer, further unlocking the potential for sustainable development within the realm of supply chain management.

In conclusion, this study not only contributes to the theoretical understanding of CLSC and its importance in achieving sustainability but also offers practical insights for businesses aiming to navigate the complexities of modern supply chains. As we move forward, it is imperative for enterprises to embrace these innovative approaches, fostering a culture of sustainability and technological adaptation that will define the future of global supply chain management.

CONFLICTS OF INTEREST

We wish to confirm that there are no known conflicts of interest associated with this publication and there has been no significant financial support for this work that could have influenced its outcome.

FUNDING STATEMENT

No funding was received for this work.

PROCESS DATES

March, 2024
Received: January 25, 2024, Revision: March 7, 2024, Accepted: February 7, 2024

CORRESPONDING AUTHOR

Correspondence should be addressed to Chunjuan Gao; ttkxgcj1919@126.com

REFERENCES

Abdi, A., Abdi, A., Fathollahi-Fard, A. M., & Hajiaghaei-Keshteli, M. (2021). A set of calibrated metaheuristics to address a closed-loop supply chain network design problem under uncertainty. *International Journal of Systems Science: Operations & Logistics*, *8*(1), 23–40.

Ashtab, S., & Tosarkani, B. M. (2023). Scenario-based multi-objective optimisation model based on supervised machine learning to configure a plastic closed-loop supply chain network. *International Journal of Business Performance and Supply Chain Modelling*, *14*(1), 106–128. doi:10.1504/IJBPSCM.2023.130469

Bressanelli, G., Perona, M., & Saccani, N. (2019). Challenges in supply chain redesign for the circular economy: A literature review and a multiple case study. *International Journal of Production Research*, *57*(23), 7395–7422. doi:10.1080/00207543.2018.1542176

Deng, C., & Liu, Y. (2021). A deep learning-based inventory management and demand prediction optimization method for anomaly detection. *Wireless Communications and Mobile Computing*, *2021*, 1–14. doi:10.1155/2021/9969357

Garcia-Buendia, N., Moyano-Fuentes, J., Maqueira-Marín, J. M., & Cobo, M. J. (2021). 22 Years of lean supply chain management: A science mapping-based bibliometric analysis. *International Journal of Production Research*, *59*(6), 1901–1921. doi:10.1080/00207543.2020.1794076

Ghasemi, P., & Abolghasemian, M. (2023). A Stackelberg game for closed-loop supply chains under uncertainty with genetic algorithm and gray wolf optimization. *Supply Chain Analytics*, *4*, 100040. doi:10.1016/j.sca.2023.100040

Giri, B. C., & Masanta, M. (2020). Developing a closed-loop supply chain model with price and quality dependent demand and learning in production in a stochastic environment. *International Journal of Systems Science: Operations & Logistics*, *7*(2), 147–163.

Goli, A., Zare, H. K., Tavakkoli-Moghaddam, R., & Sadegheih, A. (2020). Multiobjective fuzzy mathematical model for a financially constrained closed-loop supply chain with labor employment. *Computational Intelligence*, *36*(1), 4–34. doi:10.1111/coin.12228

Guan, Y., & Yu, L. (2021). Design of supply chain resource distribution allocation model based on deep learning. In *Proceedings of Multimedia Technology and Enhanced Learning: Third EAI International Conference, ICMTEL* (pp. 321-332). Springer International Publishing. doi:10.1007/978-3-030-82562-1_30

Guo, H., & Zou, T. (2022). Cross-border e-commerce platform logistics and supply chain network optimization based on deep learning. *Mobile Information Systems*, *2022*, 1–10. Advance online publication. doi:10.1155/2022/2203322

Hachaïchi, Y., Chemingui, Y., & Affes, M. (2020). A policy gradient based reinforcement learning method for supply chain management. In *Proceedings of Fourth International Conference on Advanced Systems and Emergent Technologies (IC_ASET)* (pp. 135-140). IEEE. doi:10.1109/IC_ASET49463.2020.9318258

Hajiaghaei-Keshteli, M., Abdallah, K. S., & Fathollahi-Fard, A. M. (2018). A collaborative stochastic closed-loop supply chain network design for tire industry. *International Journal of Engineering*, *31*(10), 1715–1722.

Han, C., & Zhang, Q. (2021). Optimization of supply chain efficiency management based on machine learning and neural network. *Neural Computing & Applications*, *33*(5), 1419–1433. doi:10.1007/s00521-020-05023-1

Huang, H., & Tan, X. (2021). Application of reinforcement learning algorithm in delivery order system under supply chain environment. *Mobile Information Systems*, *2021*, 1–11. doi:10.1155/2021/5880795

Kazancoglu, Y., Ekinci, E., Mangla, S. K., Sezer, M. D., & Kayikci, Y. (2021). Performance evaluation of reverse logistics in food supply chains in a circular economy using system dynamics. *Business Strategy and the Environment*, *30*(1), 71–91. doi:10.1002/bse.2610

Kegenbekov, Z., & Jackson, I. (2021). Adaptive supply chain: Demand–supply synchronization using deep reinforcement learning. *Algorithms*, *14*(8), 240. doi:10.3390/a14080240

Kim, Y. G., Yang, G. H., & Do Chung, B. (2023). Estimated model-based robust optimization of closed-loop supply chain under uncertain carbon tax rates and demand. *Computers & Industrial Engineering*, *182*, 109368. doi:10.1016/j.cie.2023.109368

Liu, Z., Hu, L. M., & Yeh, W. C. (2023). Risk-averse two-stage stochastic programming-based closed-loop supply chain network design under uncertain demand. *Applied Soft Computing*, *147*, 110743. doi:10.1016/j.asoc.2023.110743

Long, P., Lu, L., Chen, Q., Chen, Y., Li, C., & Luo, X. (2023). Intelligent selection of healthcare supply chain mode–an applied research based on artificial intelligence. *Frontiers in Public Health*, *11*, 1310016. doi:10.3389/fpubh.2023.1310016 PMID:38164449

Mirzaei, M. G., Goodarzian, F., Mokhtari, K., Yazdani, M., & Shokri, A. (2023). Designing a dual-channel closed loop supply chain network using advertising rate and price-dependent demand: Case study in tea industry. *Expert Systems with Applications*, *233*, 120936. doi:10.1016/j.eswa.2023.120936

Mishra, J. L., Hopkinson, P. G., & Tidridge, G. (2018). Value creation from circular economy-led closed loop supply chains: A case study of fast-moving consumer goods. *Production Planning and Control*, *29*(6), 509–521. doi:10.1080/09537287.2018.1449245

Modak, N. M., & Kelle, P. (2021). Using social work donation as a tool of corporate social responsibility in a closed-loop supply chain considering carbon emissions tax and demand uncertainty. *The Journal of the Operational Research Society*, *72*(1), 61–77. doi:10.1080/01605682.2019.1654413

Pan, W., & Miao, L. (2023). Dynamics and risk assessment of a remanufacturing closed-loop supply chain system using the internet of things and neural network approach. *The Journal of Supercomputing*, *79*(4), 3878–3901. doi:10.1007/s11227-022-04727-6

Peng, Z., Zhang, Y., Feng, Y., Zhang, T., Wu, Z., & Su, H. (2019). Deep reinforcement learning approach for capacitated supply chain optimization under demand uncertainty. In *Proceedings of Chinese Automation Congress (CAC)* (pp. 3512-3517). IEEE. doi:10.1109/CAC48633.2019.8997498

Prakash, S., Kumar, S., Soni, G., Jain, V., & Rathore, A. P. S. (2020). Closed-loop supply chain network design and modelling under risks and demand uncertainty: An integrated robust optimization approach. *Annals of Operations Research*, *290*(1-2), 837–864. doi:10.1007/s10479-018-2902-3

Rajabzadeh Gatari, A., Amini, M., Azar, A., & Kolyaei, M. (2021). Design of integrated mathematical model for closed-loop supply chain. *Management Research in Iran*, *20*(1), 1–32.

Reimann, M., Xiong, Y., & Zhou, Y. (2019). Managing a closed-loop supply chain with process innovation for remanufacturing. *European Journal of Operational Research*, *276*(2), 510–518. doi:10.1016/j.ejor.2019.01.028

Sardar, S. K., Sarkar, B., & Kim, B. (2021). Integrating machine learning, radio frequency identification, and consignment policy for reducing unreliability in smart supply chain management. *Processes (Basel, Switzerland)*, *9*(2), 247. doi:10.3390/pr9020247

Su, J., Li, C., Zeng, Q., Yang, J., & Zhang, J. (2019). A green closed-loop supply chain coordination mechanism based on third-party recycling. *Sustainability (Basel)*, *11*(19), 5335. doi:10.3390/su11195335

Su, Y., & Sun, W. (2018). Analyzing a closed-loop supply chain considering environmental pollution using the NSGA-II. *IEEE Transactions on Fuzzy Systems*, *27*(5), 1066–1074. doi:10.1109/TFUZZ.2018.2870693

Tang, J., Li, B. Y., Li, K. W., Liu, Z., & Huang, J. (2020). Pricing and warranty decisions in a two-period closed-loop supply chain. *International Journal of Production Research*, *58*(6), 1688–1704.

Wu, X., & Zhou, Y. (2019). Buyer-specific versus uniform pricing in a closed-loop supply chain with third-party remanufacturing. *European Journal of Operational Research*, *273*(2), 548–560. doi:10.1016/j.ejor.2018.08.028

Zailani, S., Iranmanesh, M., Foroughi, B., Kim, K., & Hyun, S. S. (2020). Effects of supply chain practices, integration and closed-loop supply chain activities on cost-containment of biodiesel. *Review of Managerial Science*, *14*(6), 1299–1319. doi:10.1007/s11846-019-00332-9

Zare Mehrjerdi, Y., & Lotfi, R. (2019). Development of a mathematical model for sustainable closed-loop supply chain with efficiency and resilience systematic framework. *International Journal of Supply and Operations Management*, *6*(4), 360–388.

Zhao, J., & Sun, N. (2020). Government subsidies-based profits distribution pattern analysis in closed-loop supply chain using game theory. *Neural Computing & Applications*, *32*(6), 1715–1724. doi:10.1007/s00521-019-04245-2