scientific reports

OPEN

Energy optimization and plant comfort management in smart greenhouses using the artificial bee colony algorithm

Muhammad Jawad[1], Fazli Wahid[1,2,3], Sikandar Ali[2], Yingliang Ma[3], Ahmed Alkhyyat[4,5,6], Jawad Khan[7]✉ & Youngmoon Lee[8]✉

Agriculture is an essential component of human sustenance in this world. These days, with a growing population, we must significantly increase agricultural productivity to meet demand. Agriculture moved toward technologies as a result of the demand for higher yields with less resources. Increasing awareness of the significance and influence of agricultural practices in global climate change has made the use of energy-efficient innovations a vital aspect of the agriculture sector. The use of greenhouses to provide controlled environments that encourage effective plant growth is one of the current associated approaches. If not properly maintained, the energy used to run the greenhouses' chillers, heaters, humidifiers, carbon dioxide (CO_2) generators, and carbon emissions becomes expensive. The goal of this research is to create a sustainable greenhouse model while achieving the best plant growth requirements with minimal use of energy. In order to achieve the lowest possible amount of energy consumption, the optimization model considered temperature, humidity, CO_2 levels, and sunlight as essential parameters in the environment. The Artificial Bee Colony (ABC) optimization technique was utilized for setting the environmental parameters for plant growth, considered for the suggested system. The system's inputs were plant-preferred factors, and plant comfort was achieved by applying ABC to boost the parameters' efficiency. A fuzzy controller was utilized to regulate different devices, including humidifiers, heaters, chillers, and CO_2 generators, by entering the introduced values. The overall efficacy of the fuzzy controllers that switch On/Off the actuators was obtained by minimizing the error between the best estimates of environmental factors and the ABC optimized values. Additionally, the suggested method was contrasted with other effective algorithms, such as Genetic Algorithm (GA), Firefly Algorithm (FA), and Ant Colony Optimization (ACO). Based on the results of the comparison analysis between the ABC algorithm and current practices, present procedures do not minimize the fluctuations in the inaccuracy between the target and actual environmental parameters, which is a necessary step towards increasing energy efficiency. The suggested method used 162.19 kWh for temperature control, 84.65405 kWh for Humidity, 131.2013 kWh for Sunlight, and 603.55208 kWh for CO_2 management, indicating the maximum energy efficiency. ACO needed 172.2621 kWh, 88.269 kWh, 175.7127 kWh, and 713.2125 kWh, in contrast to FA 169.7983 kWh, 86.04496 kWh, 155.8442 kWh, and 743.7986 kWh. Temperature, Humidity, Sunlight, and CO_2 were measured by GA at 164.1609 kWh, 86.19566 kWh, 174.6429 kWh, and 734.9514 kWh, respectively. In terms of Plant comfort, the suggested approach also outperformed 0.986770848 ACO (0.944043), FA (0.949832), and GA (0.946076). It is important to note that the research being done has the potential to minimize operating costs and maximize the amount of energy needed for plant growth, thereby creating a model for sustainable greenhouse agriculture.

Keywords Plant's preferred environment, Greenhouse environment, Energy consumption, Optimization, Smart greenhouse, Fuzzy logic

[1]Department of Information Technology, The University of Haripur, Haripur 22620, Pakistan. [2]College of Science and Engineering, School of Computing, University of Derby, Derby DE22 3AW, UK. [3]School of Computing Sciences, University of East Anglia, Norwich NR4 7TJ, UK. [4]College of Technical Engineering, The Islamic University, Najaf 54001, Iraq. [5]College of Technical Engineering, Islamic University of Diwaniya, Al Diwaniyah 58001, Iraq. [6]College of Technical Engineering, The Islamic University of Babylon, Babylon 512002, Iraq. [7]School of Computing, Gachon

University, Seongnam 13120, Republic of Korea. [8]Department of Robotics, Hanyang University, Ansan 15588, Republic of Korea. ✉email: jkhanbk1@gachon.ac.kr; youngmoonlee@hanyang.ac.kr

Agriculture plays a key factor in ensuring food security, stability, and strengthening the economy of countries. It has to fulfill a growing number of environmental and quality regulations. Integrating new technologies in the agro-alimentary sector may help meet these challenges[1]. Worldwide, access to sustainable food and energy sources is a major challenge[2]. As a method of intensive and sustainable food production, greenhouse farming has the potential to feed the globe[3]. Promoting the extensive use of greenhouses in farming could be one way to overcome the obstacles in the way of the transition to sustainable and precision farming practices. A sustainable smart approach could help minimize the effects of climate change, conserve energy and water resources, improve quality of life, mitigate environmental problems, and produce local socioeconomic and environmental benefits[4]. A greenhouse's primary goal is to create the ideal atmosphere for growing crops[5]. The roof and walls of a greenhouse are often constructed from transparent materials like glass or plastic. Plants are cultivated in a controlled environment in greenhouses, where moisture levels, soil nutrients, light levels, temperature, etc. are all regulated. Thus, by creating optimal environmental conditions, greenhouse technology enables humans to grow plants at any time[6].

The greenhouse sector has recently gained a lot of attention and has grown rapidly all around the world. Fresh vegetable production may be done all year long in greenhouses, which have a production rate that is about 50% higher than open-air farming. However, greenhouse production costs are primarily driven by labor and energy costs, which together account for more than 50% of total costs. So even a small performance boost can result in substantial cost savings. Numerous production systems have been developed for the production of food that is entirely natural for humans and leaves the fewest traces of microorganisms and diseases[7]. To maintain the environment while handling the resources in a greenhouse, skilled farmers are required. Because professional cultivators are exceedingly hard to locate and command high pay, hiring them takes time and money. Therefore, an IoT-based system must be created that can facilitate the first and enable continuous greenhouse environment monitoring. By keeping track of several essential indicators, such as temperature, humidity, CO_2 level, etc., the system may obtain results from producing crops and cultivating a productive greenhouse environment for plant growth[8]. Utilizing resources as efficiently as possible lowers costs, therefore agriculture machinery should only be utilized when truly necessary. The system can assist in gathering data related to farms for a long-term assessment and accurate decision-making to maximize profit and use resources efficiently. Furthermore, it improves the efficiency of the farmer's labor by keeping him informed through prompt messages, reminders, and alarms[9]. The approach significantly lowers the expense of hiring professional farmers by empowering the average farmer to properly run and maintain the farm with little to no training[10].

In the paper, we explain the optimization approach through ABC regarding the means to automate the key greenhouse environmental operations automatically. The proposed approach is set with the aim of providing optimal interior climate conditions to maximize plant productivity and resource efficiency. Therefore, our optimization framework is pursued in detail through mathematical modeling while tested with rigorous experimentation. The proposed method dynamically balances the key greenhouse parameters, especially temperature, CO_2 concentration, sunlight, and humidity while minimizing energy usage for efficient and sustainable greenhouse management. The main goals of this work are to minimize energy costs with comfortable conditions for plant growth, to identify and integrate the parameters necessary for control of greenhouse environmental conditions, and to design an intuitive yet efficient optimization algorithm that achieves efficacy in energy efficiency and comfort for the plants. The contributions go in three aspects: the development of a complete energy-efficient optimization model, validation of the case by comparisons with other metaheuristics, such as ACO, FA, and GA, and laying down a basis for potential applications in real-world scenarios regarding sustainable greenhouse operations.

To the best of our knowledge, no optimization algorithms have been used to optimize energy use in smart greenhouse systems, despite a thorough investigation of these algorithms in various fields. The algorithms ABC, ACO, FA, and GA are introduced to such a situation in this study. Among the algorithms tested above, ABC outperformed ACO, FA, and GA in terms of efficiency and had the greatest results in terms of plant comfort and energy savings. In addition to having a high plant comfort index, the ABC used the least amount of energy of any significant greenhouse parameter when temperature, humidity, sunshine, and CO2 were managed optimally. Therefore, this work has set a standard for the future by highlighting the ABC algorithm's potential as an innovative and effective technique for maximizing energy utilization in smart greenhouses.

The remaining papers are structured as follows. A thorough overview of the literature review and associated work is provided in Sect. 2, the suggested work is presented in Sect. 3, the implementation and experimental findings are shown in Sect. 4, and a thorough discussion is provided in Sect. 4. Section 5 concludes the paper. The abbreviations and descriptions used in this study are listed in Table 1

Related work

Different authors have applied IoT in various domains considering parameters according to their specific problems and factors that influence the System's operation. In this section, several remarkable applications of IoT in different contexts are shown. To forecast a smart building's interior temperature, the authors in[11] suggested integrating IoT and machine learning algorithms. Also, IoT architecture is employed to deliver standardized, interoperable, portable, and safe solutions for forecasting energy usage in smart buildings. The prediction of interior temperature, where variables handle values in several quantities, is the primary focus of this article. It uses meteorological factors such as sun radiation, ambient temperature, and wind speed that have developed over time. Using a time series solution with the help of various well-known machine learning techniques, such as Support Vector Machine (SVM), Random Forest, and Neural Networks, allows us to predict the temperature

Abbreviation	Description
TNP	Total Number of Parameters
UpBdi	Upper Bound of Parameters i
LowBdi	Lower Bound of Parameters i
Rng	Range
CS	Size of Colony
Max C	Maximum Cycles
CI	Comfort Index
Tu	Plant Preferred Temperature
Cu	Plant Preferred Carbon Dioxide
Hu	Plant Preferred Humidity
Su	Plant Preferred Sunlight
err1	Error Difference between Environmental Temperature and Plant-Preferred Temperature
err2	Error Difference between Environmental Carbon Dioxide and Plant Preferred Carbon Dioxide
err3	Error Difference between Environmental Sunlight and Plant Preferred Sunlight
err4	Error Difference between Environmental Humidity and Preferred Humidity
Prr1	Preference Parameter for Temperature
Prr2	Preference Parameter for Carbon Dioxide
Prr3	Preference Parameter for Sunlight
Prr4	Preference Parameter for Humidity
TRp	Total Required Power
RP1	Power Required for Temperature
RP2	Power Required for Sunlight
RP3	Power Required for Carbon Dioxide
RP4	Power Required for Humidity
ABC	Artificial Bee Colony
ACO	Ant Colony Optimization
FA	Firefly Algorithm
GA	Genetic Algorithm

Table 1. Abbreviation and their description.

of the conditioned space. In another study, the authors in[12] addressed interior environment management using IoT-based smart building technology which seeks to supply the following key features: checking concerning the environmental conditions in the area, identification of the room's occupants, a cloud-based system where virtual mega entities use machine learning techniques to do data analysis jobs, and digital entities gather the information produced by the sensors. The Raspberry was equipped with a light neural network, which allowed it to determine how many people were in the room based on measurements taken by the sensors and photos taken by the camera. Based on environmental criteria, a bagging tree is used for population estimation and classification. Gaussian process regression provides a Pearson correlation coefficient of 0.939 since it forecasts the total volatile organic compounds (TVOCs) with the number of occupants based on environmental conditions. The article's primary goal is to forecast energy use and enhance energy management in commercial smart buildings, aiming for improved efficiency. For prediction, k-NN (k-nearest neighbor) is employed. The technique will forecast the peak load based on the amount of electricity used, expressed in kilowatts. Current and voltage comprise the highest demand feature used in the k-NN algorithm[13]. The control of smart building ventilation subsystems is the goal of the paper's discussion and demonstration of a black-box modeling application realized using data mining techniques. Two steps form the foundation of the data processing and learning framework: Raw data streams are compressed using the symbolic aggregate approximation approach, and the resulting section is inserted into the support vector machine algorithm[14].

A Personalized comfort technique is presented in[15] to forecast occupants' thermal comfort in terms of temperature. Four stages make up the suggested method: gathering data, preparing it, extracting features, choosing features, balancing the data, and predicting thermal sensing using machine learning and deep learning in uncomfortably warm, impartial, and cold climate zones using a combination of logistic regression (LG), random forest (RF), Hoeffding Decision Tree (HDT), and deep artificial neural network (DANN). The authors in[16] suggest the use of intelligent irrigation systems to forecast a crop's water needs. This system includes sensors for temperature, humidity, and moisture that are placed in a field of crops and transmit data via a microprocessor to create an Internet of Things (IoT) device with the cloud. Data in real-time collected from the surroundings is used to anticipate outcomes effectively results are predicted by the decision tree algorithm and sent to the farmer. To identify pests, the authors in[17], combine environmental sensors, artificial intelligence (AI), the Internet of Things (IoT), and picture recognition technology. Using the YOLOv3 (You Only Look Once) series, a neural network approach was utilized to locate insects using image identification and to predict their appearance using Long Short-Term Memory (LSTM) analysis of environmental data from weather stations. The article gives farmers the information they need to precisely apply pesticides at the right time and place, reducing the number of agricultural workers needed for effective pest management and attaining the objective of smart agriculture. Utilizing predictive computational intelligence approaches, wheat crop nitrogen status is assessed. Analysis of crop photographs that were taken in the field under various lighting conditions will be used to determine the evaluation. The wheat crop is first exposed to the HSI color normalization procedure, which is followed by crop precision status categorization utilizing genetic algorithms (GA) and artificial neural networks (ANN) based prediction. By categorizing the crop yield age into groups, an ANN-based optimized technique can effectively distinguish wheat crops from other undesirable plants and weeds[18]. The emerging Internet of Things (IoT) environment generates large volumes of unstructured multimedia data. To remove important data derived through the continuously created data, deep learning (DL) techniques are being applied. Because of their low processing power, DL approaches are difficult to integrate with Internet of Things devices. Even so, this problem is addressed with cloud computing, it also contains some drawbacks like service delays and congestion on the network. This paper suggests a way for distributing a part of the DL layers for the fog nodes in an intelligent agriculture system on fog computing. With consideration for each fog node's available processing power and bandwidth, the best layers of the DL model are selected for each fog node using the suggested deep learning entrusted to fog nodes (DLEFN) approach. A DL framework with modularity and swift speed named Caffe (convolutional architecture for fast feature embedding) is initially employed for the project[19]. The authors in[20] anticipate the recommendation of an IoT-based intelligent irrigation architecture that uses machine learning to estimate soil moisture, leading to a clearer outcome. This research suggests a special EDGE-Fog-IoT-Cloud platform for an IoT-based sustainable agricultural design. This framework aims to provide an example of how AI methods can help in making informed irrigation decisions for agriculture that improve water efficiency. Because the Internet of Things (IoT) devices in a smart city network are linked to sensors that are connected to large cloud servers, they are susceptible to threats and malicious assaults. It is crucial to devise methods to thwart these assaults and keep IoT devices from malfunctioning. In this paper, the authors evaluate an attack and anomaly detection method using machine learning algorithms (LR, SVM, DT, RF, ANN, and KNN) to reduce IoT cybersecurity risks in a smart city. Unlike previous works that focused on single classifiers, the research also explores ensemble methods, such as bagging, boosting, and stacking, to increase the detection system's performance. Moreover, consider including multiclass classification, cross-validation, and feature selection for the topic at hand[21].

IoT-based smart brain bleeding detection system is debated and uses machine learning methods. It is extremely difficult to detect brain bleeding automatically, which typically results in death or permanent abnormalities. To reduce the likelihood of injuries or lifelong disabilities and to provide high-quality medication right to the patient's door, this research has created an intelligent Internet of Things application that can accurately identify brain hemorrhage. This Internet of Things application uses a feedforward neural network and support vector machine to identify the various kinds of brain hemorrhages[22]. A Machine Learning Approach to Modelling an Intelligent Ecosystem (IEML) is created in[23] to evaluate ecosystem performance intelligently and effectively using machine learning techniques. This study describes a smart city that uses ANN techniques to predict the ecosystem instantly. Using the cloud and the Internet of Things, the structure is created to make it easier to store, index, and visualize the data that a smart city's input parameters create. To create a predictive and intelligent ecosystem model, using a suggested ANN-based method, the Scaled Conjugate Gradient (SCG), Levenberg–Marquardt (LM), and Bayesian Regularization (BR) algorithms are employed. To diagnose COVID-19 patients in smart hospitals, an approach that utilizes machine learning (ML) and the Internet of Things (IoT) is presented in[24]. In this sense, it was highlighted to show how useful IoT technology and machine learning models work in smart healthcare settings. It is possible to improve the diagnostic (classification) accuracy based on laboratory data by utilizing basic machine learning models. Three machine learning (ML) models—Naive Bayes (NB), Random Forest (RF), and Support Vector Machine (SVM)—were developed and evaluated using lab datasets. Diagnostic decisions can be supported by the suggested method, which is based on ML and IoT. Additionally, the results may decrease the workload for medical professionals, address the issue of patient congestion, and lower the COVID-19 pandemic fatality rate. An anomaly detection system, or "ADS," is suggested in[25] for smart hospital infrastructures. It has two modules: IDC for identifying network anomalies and threats (the SVM method is used to classify the received data), and EDC for identifying e-health-related events (SVM is used to find the events of interest, just like IDC). IDC and EDC interact well in a single, integrated system, making administration simple and reducing the cost of system management. The events that EDC detects are carefully analyzed and are no longer trusted after IDC indicates an intrusion. For hospital infrastructures to avoid providing inaccurate signals or making poor decisions about patients' healthcare, such reliability is necessary. Crop production in greenhouses is now a well-established method, and automation in many areas of this system is now possible due to advancements in technology. Thus, installing an automated smart greenhouse powered by the Internet of Things (IoT) and an Adaptive Neuro-Fuzzy Inference System (ANFIS) might be the best option to increase food output on the land. Real-time sensors are used to gather four types of weather data: temperature, humidity, direct sunlight, and soil moisture. The fuzzy control system then receives these gathered data as input variables. The data are subsequently modified by the optimal values for the climate parameters are then determined by ANFIS and the fuzzy control system. When the optimal winter temperature is 24 °C and the relative humidity is 76.00%, crops grown during the winter are considered in the final simulation. The technique has a 93.62% detection accuracy at the perception layer, with precision at 0.83, recall at 0.78, and FI score at 0.81[26]. The learning-to-prediction approach was proposed by the authors as a way to improve the predictive algorithms' accuracy in dynamic circumstances. To evaluate the sustainability of the proposed approach, an artificial neural network (ANN) learning module is built to increase the accuracy of the Kalman filter method. In experiments, the Kalman filter technique is used to accurately predict the parameters of the indoor climate (temperature, CO_2, and humidity) from noisy sensor measurements in an inside setting that resembles a greenhouse[27]. The implementation of deep learning techniques on an IOT-based greenhouse tried to give it the capacity to identify and categorize various types of diseases seen inside the room, track the growth and development of fruit, and perform the general management required to keep the system fully operational without the involvement of humans. The entire learning and prediction task is executed utilizing MASK-RCNN and Convolutional Neural Network (CNN). Weighted class and average accuracy are calculated to be 0.93 and 0.91, respectively[28]. To better manage the greenhouse's energy needs and resources, the authors in[29] focuses on forecasting the electrical amount of energy used by the air conditioner and the electrical output of photovoltaic modules. These predictions may also assist in reducing some of the costs associated with measurement and monitoring equipment. Different supervised machine learning algorithms, including Boosting, Support.

Vector Machine (SVM), Gaussian Process Regression (GPR), and Artificial Neural Networks (ANN), are used to make the prediction. Three statistical measures were used to evaluate and contrast the performance of the models: the normalized mean absolute error (nMAE), the normalized root mean square error (nRMSE), and the coefficient of correlation (R). To simulate predictions regarding soil temperature and water content, the authors of[30] propose combining physical models, and machine learning techniques with dynamic topology, and meteorological forecast data in a greenhouse. This study employed HYDRUS-1D, the random forest model, the ICON (inferring connections of networks) model, and Newton's law of cooling to reproduce and confirm the measured soil temperature. The ICON, random forest, and HYDRUS-1D models were used to simulate and confirm the measured volumetric water content. Additionally, the simulation results of several models were compared using RMSE (root mean square error). The authors in[31] evaluate the possibility of improving food security for mountain communities as climate change and lack of resources have been exacerbated through an innovative tool called a Smart Greenhouse (SGH). The SGH has equipped an 8 m^2 area with a solar water heating system, integrated weather stations, and sensors within the greenhouse that monitor light, humidity, and temperature. This enhanced microclimate facilitated experiments during the otherwise barren winter months. Equipped with sensors and communication technologies, the SGH automates heating, ventilation, and air conditioning regulation. Located in Genekha, Thimphu, it utilizes renewable energy for winter heating, eliminating reliance on external sources in summer. Performance assessments were conducted using TRNSYS and OpenStudio software, focusing on temperature, humidity control, soil pH, nutrient content, and crop growth timelines. Three cooling packages for smart greenhouses were proposed in[32], as well as the development of an energy model with the integration of plant transpiration using Energy Plus. 1,296 simulation datasets were generated based on installation area, scale, crops, and covering materials. These datasets have been used to develop a performance evaluation tool using temperature, humidity control, and energy cost indices so that efficient cooling solutions can be selected for sustainable crop production. An actor-critic algorithm with a shared attention mechanism is used in[33] to present a multi-agent deep reinforcement learning (MADRL) control framework for energy management in networked greenhouses. To handle the variations brought on by the production of renewable energy and fluctuating electricity costs, a network of interconnected greenhouses with renewable energy is built to communicate with the power grid when needed. A network of five greenhouses with different capacities is used to assess the performance of this multi-agent strategy to prove its feasibility and scalability. Comparing the suggested MADRL-based control strategy to well-known algorithms, it shows efficiency in preserving the interior climate in every greenhouse while guaranteeing a 28% decrease in net load demand. To improve energy efficiency and thermal comfort in buildings under four distinct Iranian city climates, the authors of[34] considered integrating PCMs with thermal insulation. Except for Tehran, they used the Response Surface Method (RSM) to optimize the thermostat settings for heating (20 °C) and cooling (25–28 °C); the choice of materials (BioPCMDSCM27Q21 and polyurethane insulation); and the thicknesses of PCMs (~5 cm) and insulation (between 6.9 and 9.8 cm). In addition to 25% to 60% improvements in thermal comfort, energy savings of 43% to 99% for heating and 38% to 52% for cooling were noted. Optimizing thermal comfort (Tc) while minimizing heating load (HL) and cooling load (CL) was the goal of a residential building energy-saving approach in[35]. EnergyPlus and Jeplus software were used to simulate the building model for cities with varying climates, taking into account multi-objective optimization using the NSGA-II method. To improve the energy efficiency of the insulating strategy, the findings show an average maximum improvement of 17–39% in CL, a maximum improvement of 38–62% in Tc, and a significant value of 61–100% improvement in HL. As the authors in[36] optimized the components that contribute to indoor air quality, or IAQ, they adequately addressed this study topic in terms of the following: clean air transfer, insulation levels, air speed, activity levels, and air conditioner temperature settings. As a result, they were successful in their attempt to reduce the concentration of CCO2 and PCO2 while maintaining elevated levels of Tc. They employed JEPLUS software and the NSGA-II method, which is supported by EnergyPlus, to do a sensitivity analysis and multi-objective optimization. The findings show improvements of 17–30% in CCO2, 15–37% in PCO2, and 52–80% in Tc. They also show that additional pollutants such CH4, N2O, NOx, and SO2 have been reduced. The authors created a model for predicting and optimizing CO2 emissions in apartment buildings in six US cities with varying climates in[37]. They predicted the monthly and annual CO2 levels from 2020 to 2025 using an artificial neural network of the GMDH type, and discovered that they would rise by 1–3 percent per month and 1.25–1.8% annually. Problems resolved Thermostat set points, garment insulation levels, and clean air transfer by air conditioning systems are some of the five design elements that were taken into consideration in order to handle this problem. Reducing CO2 emissions and yearly electricity expenses while improving thermal comfort are among the goals to be minimized or improved. Using JEPLUS + EA software with the NSGA-II algorithm for optimization and EnergyPlus for energy performance analysis, this study was able to find ideal configurations that offer flexible solutions for improved indoor air quality and energy efficiency. The focus of[38] was on optimizing smart shading systems to lower energy usage and improve thermal and visual comfort in buildings. Applying JEPLUS and EnergyPlus software to Tehran, a total of 21 design variables were examined. According to the NSGA-II algorithm, external blinds outperformed inside ones, and narrower slat angles increased visual comfort but at the expense of higher illumination energy consumption. Visual and thermal comfort increased by 70–100% and 10–40%, respectively, while the annual energy usage was decreased by 40–50%.

Methodology

In the suggested architecture as shown in Fig. 1, the plant's preferred temperature, humidity, CO_2, and sunlight are entered into the ABC optimizer along with Greenhouse environmental temperature, humidity, Co2, and sunlight. Based on plant preferences, the ABC optimizer maximizes environmental factors to enhance plant comfort. The input for the fuzzy controllers (temperature fuzzy controller, CO_2 fuzzy controller, humidity fuzzy controller, and sunlight fuzzy controller) is the difference between the optimized values and the environmental parameters. The power needed to regulate the operation of the actuators (chiller, heater, humidifier, dehumidifier, CO_2 generator, and Solar heat lamp) is the result of the fuzzy controllers. After receiving the required power as input, the coordinator checks to see if power is available from the power sources and then provides power to each actuator based on the status data that the fuzzy controllers have provided. In addition to the ABC-optimized values, the fuzzy controllers further get inputs from the environment's temperature, CO_2, humidity, and sunlight. The fuzzy controllers' output values are determined by the variations between the measured values of CO_2, sunlight, humidity, and temperature in the environment and by the values of these four parameters that were optimized by ABC. An ABC optimization's primary goal is to reduce these error differences. The significant error differences that result from not using the ABC optimization procedure ultimately lead to greater output levels and increased consumption of energy. As optimization is implemented, the error differences decrease, leading to optimal energy utilization.

Artificial bee colony optimization

Artificial Bee Colony (ABC) is a nature-inspired optimization technique modeled after the foraging behavior of bees. There have been a lot of algorithms created over the past few decades that simulate the behaviors of natural systems. Among these are evolutionary algorithms (EA)[39,40], harmony search (HS)[41], ant colony optimization (ACO)[42], and particle swarm optimization (PSO)[43]. They are referred to as versatile algorithms since they can be used to solve several issues. In 2005, Karaboga first presented Artificial Bee Colony as a technical report to solve numerical optimization problems. The operation of the ABC algorithm, similar to other optimization methods, is illustrated in Fig. 2. According to[44] the following benefits listed by several authors, artificial bee colonies are preferable to alternative optimization methods.

i. According to[45], and[46], ABC is a simple, reliable, and scalable system.
ii. ABC only makes use of a limited set of control Parameters[47] in contrast to alternative optimization methods.
iii. It is easy to include ABC in hybrid optimization methods[46].
iv. ABC is capable of handling stochastic objective functions[48].

The primary steps of artificial bee colony optimization which are adapted to overcome the challenges of optimization are provided in the following section[48]. The pseudocode for ABC is as follows.

Step 1: Set the initial values for the parameters special to the ABC algorithm and the demands of the task for the Artificial Bee Colony.

Step 2: Create a population of initial food sources.

Repeat Steps 3 through 6 until the goal or termination condition is satisfied.

Step 3: Worker bees investigate the available food sources within the colony.

Step 4: Onlooker bees choose food supplies based on how good they are.

Step 5: Scout bees explore new food sources outside the existing community.

Step 6: Retain knowledge of the best food source discovered so far.

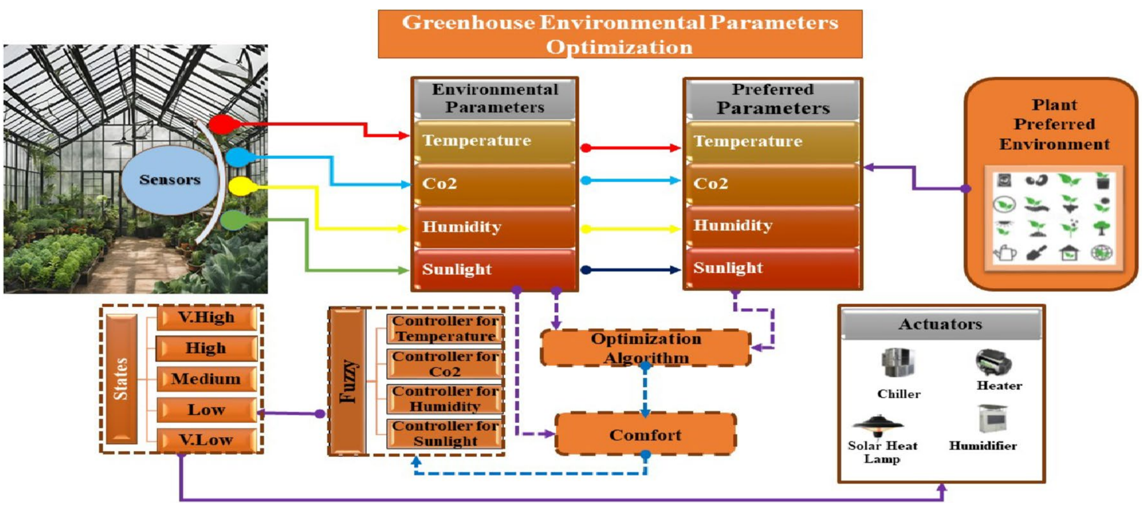

Fig. 1. Proposed Methodology.

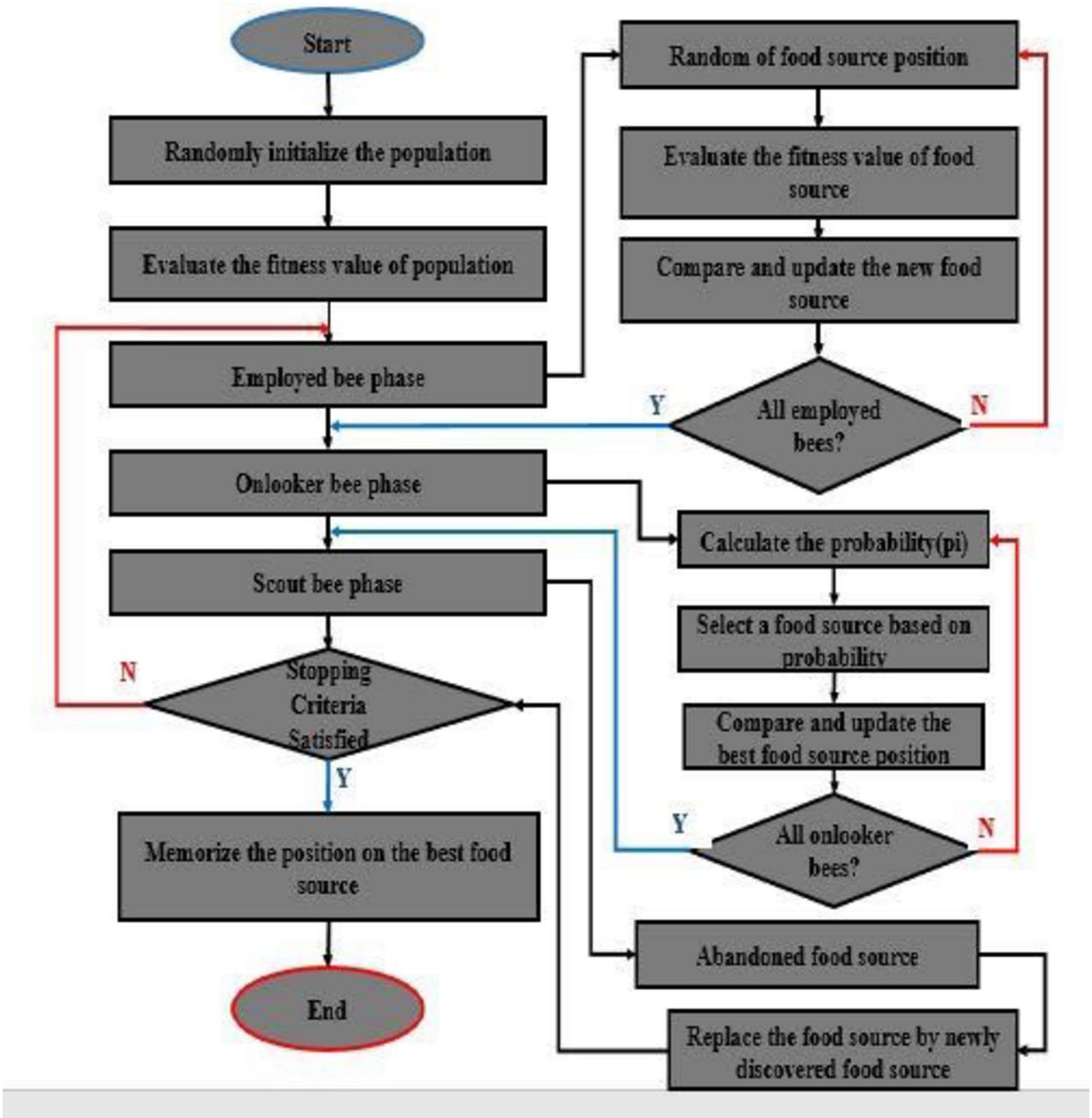

Fig. 2. Proposed ABC Algorithm.

Carry on with the iteration until the objective has been reached or the termination condition has been met.
Step1: Initializing Parameters

i. **Important Parameters:** There will be a maximum number of parameters. The amount of parameters that require optimization is indicated by the total number of parameters (TNP). Temperature (Temp), Humidity (Humd), Carbon dioxide (CO_2), and Sunlight (S) are the four variables we are optimizing.
ii. **Upper Bound (UpBdi):** The upper bound of the parameters i is represented by UpBdi, Where $i = 1,2,3,\ldots,TNP$, with TNP representing the total number of parameters requiring optimization. The maximum values for Humidity (Humd) is 80%, Temperature (Temp) is 24°C, Sunlight intensity (S) is 400 μmdi/m2/sec and Carbon dioxide (CO_2) is 1000 ppm.
iii. **Lower Bound (LowBdi):** The lower bound of the parameters i is represented by LowBdi, Where $i = 1,2,3,\ldots,TNP$, with TNP representing the total number of parameters requiring optimization. The minimum values for Humidity (Humd) is 40%, Temperature (Temp) is 18 °C, Sunlight intensity (S) is 300 μmdi/m2/sec and Carbon dioxide (CO_2) is 400 ppm.
iv. **Range (Rng):** "Range" refers to the difference of the parameters' upper and lower boundaries as given by UpBdi—LowBdi. There are four different ranges: 40, 600, 100, and 6 for humidity, carbon dioxide, sunlight, and temperature respectively.

v. **Size of Colony (CS):** It serves as an indicator of the population's overall amount of food sources or solutions. This statistic represents the total number of bees employed or onlooker. In order to identify the ideal colony size for the algorithm to work at its best, it has been tested for various numbers of colony sizes.
vi. **Foods (NoF):** Food is a representation of the entire population of the food supply. To achieve the optimum outcomes from optimization, the population as a whole has been varied.
vii. **Maximum Cycles (MaxC):** In an algorithm run, they stand for the majority of generations. Different cycles of testing have been conducted on the algorithm.
viii. **Limit (LN):** The limit denotes the highest number of generations during which employed bees will keep using a food supply before moving on if it remains unchanged. Various values for the limit have been established to achieve the best performance.
ix. **Objective Function:** It is necessary to make improvements to this process. The comfort index value, as stated in (4), is the goal when developing the algorithm.
x. **Objective Value:** The value of each food source's related objective function is represented by objective value.

Step 2: Create an Initial Food Source Population

We require the TNP, UpBd, LowBd, Rng, and CS parameters mentioned above to initialize the food supply. A matrix of dimension CS * TNP called "food source" has an entry for each possible food source. By employing (1), each vector is created[47]. Consider

$$Z_j\,(i) = LowBdi + (UpBdi - LowBdi) \times ŕ \tag{1}$$

∀j ε (1, 2, 3, ….., CS), ∀ i ε (1,2,….., TNP), where ŕ ~ (0,1) generates a uniform random number within the range of 0 and 1. Initializing the food source is done by (2)[47] taking the previously mentioned variables into account. only one

$$Food = Random(NoF, TNP) * Rng + \ln \tag{2}$$

Step 3: Employed Bees Investigate the Food Sources Among the Population

At this point, a new solution is generated by the neighbor using[47], where (3) is used to assign working bees to a food source.

$$Z\prime(i) = Z_j(i) + ŕ(Z_j(i) - Z_k(i)), \quad ŕ \sim (0, 1). \tag{3}$$

where ŕ ~ (0,1) and κ≠j for every κ ϵ (1, 2,…….CS), The pseudocode is given in Pseudocode 1. The total probability of all employed bees is represented by s_prob in Pseudocode 1.

```
for (j = 1 to CS
{ for (i = 1 to TNP)
  {
   Z′(i) = Zj(i) + ŕ (Zj(i) − Zκ(i)) , ŕ ~ (0,1).
  ∀ κ ϵ (1, 2,,…….CS), κ ≠ j and ŕ ~ (0,1)
  }
               end for
 Calculate of f(Zi)
If (f(Z′) ≥ f(Zi))
 {
              Zi = Z′
               f(Zi) = f(Z′)
 }
              end if
 }
end for
```

Pseudocode 1. Employed Bee Phase[47].

Step 4: Food sources are selected by Onlooker bees based on their quality.

Both bees in employed and that onlooker have an equal number of food sources. The selection probability of each food source generated by the employed bees is initially determined by the onlooker bee. Next, the optimal food source is selected using the Roulette selection procedure. Pseudocode 2 depicts the entire observer bee phase in its entirety.

```
for (i=1 to CS)
{
ŕ ~ (0, 1) s
j
s_prob = = 0;
While (s_prob <=ŕ)
{
 s_ prob = s_ prob + pj
j = j + 1;
}
end
while for (k=1 to TNP)
{
 Z′(j) = Zj(κ) + ŕ (Zj (κ) − Zj(n))
 n ϵ (1,2,3, … … . , CN)
 }
 end for
 Calculation of (Zj)
 if
{
(f(Z′) >= f(Zj))
Zi = Zj′
f(Zi) = f(Zj′)
}
end if
}
end for
```

Pseudocode 2. Onlooker Bee Phase[47].

Step 5: Scout bees search for alternative food sources outside of the current population.

After conducting a random search, the scout bees use **(2)** to replenish abandoned food sources. An abandoned food source is a food source that cannot be upgraded after a specified number of cycles. In Pseudocode 3, the scout bee algorithm is described.

```
fo(i = 1 to CS)
{
if(S(i) == Limi == Limit)
 {
Zj is generated using (1)
}
end if
}
 end for
```

Pseudocode 3. Scout Bee Phase[47].

Step 6: Memorize the Best Food Source Discovered So Far

This stage has the highest objective value since it requires learning the locations of the food sources by memory.

Stopping State. The maximum number of cycles is reached by repeatedly doing Steps 3 through 6.

Justification for choosing standard ABC

We chose the classical ABC algorithm because it is already well proven to be in an adequate balance between simplicity, adaptability, and efficiency of multi-objective optimization problems. Although the last variants of the ABC, including GABC (Global ABC) and IABC (Improved ABC), might better be considered for potentially higher performance, they most likely also increase computational complexity and demand a great tuning of parameters, which in itself may complicate an implementation within resource-limited environments such as smart greenhouses. Although the alternative metaheuristic algorithms such as FA, ACO, and GA proved their strength in many applications, they may not match ABC's balance of convergence speed accuracy and computational complexity within this context.

This was made clear by the comparison of the proposed ABC algorithm with FA, ACO, and GA in the key performance metrics such as energy efficiency, convergence speed, and maintaining optimal conditions for plant growth. Table 2 makes this comparison crystal clear, stating the benefits of ABC algorithms in maintaining good energy use while holding computational simplicity.

Plants comfort

Comfort value is a metric used in optimization algorithms to measure how comfortable the Plant is with the algorithm's results. It is a measure of Plant's satisfaction with the solution presented by the algorithm and is used to determine whether the algorithm is providing satisfactory results or not. Comfort value is usually calculated by analyzing the Plant Preferred Environment, such as preferences, and comparing it to the algorithm's output. If the output matches the Plant Preferred Environment, then the comfort value is high. If the output does not match the Plant Preferred Environment, then the comfort value is low. To calculate the Comfort Index, we used the following formula[49],

$$\mathrm{CI} = \mathrm{prr1}\left[1-\left(\frac{err1}{Tu}\right)^2\right] + prr2\left[1-\left(\frac{err2}{Cu}\right)^2\right] + prr3\left[1-\left(\frac{err3}{Su}\right)^2\right] + prr4\left[1-\left(\frac{err4}{Hu}\right)^2\right] \quad (4)$$

CI denotes the Plant comfort level index. The parameters set by the Plant Preferred Environment, namely err1, err2, err3, and err4, correspond to the four respective parameters, and the Environmental parameters are given by err1, err2, err3, and err4. CI can have a maximum value of 1. *Cu* is the Plant Preferred concentration of carbon dioxide, Su is the Plant Preferred amount of sunlight, Hu is the Plant Preferred humidity, and *Tu* is the Plant Preferred temperature.

Fuzzy controller

A fuzzy controller is a kind of computational algorithm that determines the optimal result for a given input by applying fuzzy logic, which is a sort of many-valued logic. Applications for fuzzy controllers are numerous, ranging from industrial control systems to home appliances. Fuzzy controllers are particularly useful when an exact solution is difficult to obtain or not possible due to the presence of multiple variables. In such cases, fuzzy logic provides a more accurate and reliable solution than traditional algorithmic methods.

In our designed structure, we used four Mamdani fuzzy controllers to control the significant environmental parameters in the smart greenhouse system: temperature, sunlight, humidity, and carbon concentration. Each controller is made up of four major components: a fuzzifier, rule base, inference engine, and defuzzifier. The fuzzifier converts the crisp values of the inputs into membership values with the use of triangular membership functions defined for each variable input, in which nine membership functions are established for every input variable. The set of rules defining the output membership values is contained in the rule base. Using an inference engine, this rule base determines the defuzzifier to translate fuzzy outputs into crisp values that determine the required power for cooling/heating (temperature controller), lighting (sunlight controller), humidification/dehumidification (humidity controller), and carbon dioxide concentration regulation (CO_2 controller). The Mamdani model was adopted because its interpretability suits multi-variable systems and, therefore, complex greenhouse conditions can be controlled in an effective manner with simplicity in rule formulation.

Algorithm	Energy Efficiency (kWh)	Convergence Speed (Iterations)	Accuracy (% within Target Range)	Computational Complexity
Standard ABC	981.59	200	98.6%	Low
FA	1155.48606	200	97.92%	Moderate
ACO	1149.45637	200	97.97%	Moderate
GA	1159.95095	200	97.9693%	Moderate

Table 2. Comparison of Standard ABC with FA, ACO, and GA.

Fuzzy controller for temperature

In the proposed design, the temperature fuzzy controller receives as input the error between the optimized temperature values by the optimizer and the temperature in the surroundings. The temperature fuzzy controller's output is the amount of power required for the heating or cooling system. The necessary power for the actuator status is the temperature fuzzy controller's output, and an artificial bee colony is used to measure the error variations between the optimized parameters and the actual environmental parameters, thereby modifying the status of the cooling and/or heating actuators. The Figure A shows a temperature fuzzy controller:

If (e 1 = = EC) then RP1 = R1EC.
If (e 1 = = VC) then RP1 = R1VC.
If (e 1 = = CD) then RP1 = R1CD.
If (e 1 = = CL) then RP1 = R1CL.
If (e 1 = = MD) then RP1 = R1MD.
If (e 1 = = WM) then RP1 = R1WM.
If (e 1 = = HT) then RP1 = R1HT.
If (e 1 = = EH) then RP1 = R1EH.

A. Temperature Fuzzy Controller.

In these rules, the error difference between the actual and ABC-optimized temperatures is represented by e1, which serves as the temperature fuzzy controller's input. Based on this error differential, the temperature fuzzy controller supplies the energy to the cooling/heating actuators as its output, with RP1 (required power 1) indicating the energy needed. Following EC are VC, CD, CL, MD, WM, HT, and EH. EC stands for the minimum error between the environmental temperature and the ABC-optimized temperature. Thus, the error difference grows as we move from EC to EH and vice versa. Therefore, for error difference EC, the amount of required power (RP1) is minimal (RP1 = R1EC), while for error difference EH, the required power (RP1) is maximal (RP1 = R1EH). Therefore, the lowest error between the surrounding temperature and the ABC-optimized temperature is denoted by the letter EC, whereas the maximum differences between the two are denoted by the letter EH. As a result, R1EC indicates the smallest amount of power required to control the cooling/heating system, and R1EH for the maximum power required.

Fuzzy controller for sunlight

The sunlight fuzzy controller receives as its input the difference in errors between the sunlight optimized by the ABC optimizer and the surrounding sunlight. The necessary power for the actuator status is the sunlight fuzzy controller's output, and the lighting actuators' status is adjusted based on the variations in errors between the optimized parameters for the artificial bee colony and the real environmental conditions. The following rules for the Sunlight fuzzy controller are shown in Figure B: These rules define e2 as the error difference between the actual and ABC-optimized sunlight, which serves as an input to the sunlight fuzzy controller. Based on this error difference, the sunlight fuzzy controller provides the energy output, labeled as RP2 (needed power 2), for the lighting system. The sequence after MiSL includes LSL, MSL, BSL, ISL, SSL, VSSL, ESL, and MXSL. MiSL signifies the minimum difference between the environmental sunlight and the ABC-optimized sunlight. Thus, the error difference grows as we move from MiSL to MxSL and vice versa. Therefore, for error difference MiSL, the amount of required power (RP2) is minimal (RP2 = R2 MiSL), while for error difference MxSL, the required power (RP2) is maximal (RP2 = R2 MxSL). MiSL thus denotes the minimum error between the sunlight from the environment and that optimized by ABC, whereas MxSL indicates the maximum error between the environmental and ABC-optimized sunlight. As a result, R2 MiSL indicates the smallest amount of power required to control the lighting system, and R2 MxSL for the maximum power required.

If (e 2 = = MSL) then RP2 = R2MSL.
If (e 2 = = LSL) then RP2 = R2LSL.
If (e 2 = = MS) then RP2 = R2MS.
If (e 2 = = BSL) then RP2 = R2BSL.
If (e 2 = = ISL) then RP2 = R2ISL.
If (e 2 = = SSL) then RP2 = R2SSL.
If (e 2 = = VSSL) then RP2 = R2VSSL.
If (e 2 = = ESL) then RP2 = R2ESL.
If (e 2 = = MXL) then RP2 = R2MXL.

B. Sunlight Fuzzy Controller.

Fuzzy controller for carbon dioxide

The carbon dioxide fuzzy controller receives as input the error difference between the environmental and optimized carbon dioxide from the ABC optimizer. The output of the carbon dioxide concentration supplies the energy required for the carbon dioxide generators. The needed power for the actuator status is the carbon dioxide fuzzy controller's output, and the real environmental parameters and the optimized parameters for the artificial bee colony fluctuate in error, affecting the actuator status of the carbon dioxide generator. Figure C describe the fuzzy rules for the fuzzy controller for carbon dioxide:

If (e 3 = = VL) then RP3 = R2VL.
If (e 3= = L) then RP3 = R2L.
If (e 3 = = ML) then RP3 = R3ML.
If (e 3 = = M) then RP3 = R3M.
If (e 3 = = MH) then RP3 = R3MH.
If (e 3= = H) then RP3= R3H.
If (e 3 = = VH) then RP3 = R3VH.
If (e 3= = EH) then RP3= R3EH.
If (e 3= = HZ) then RP3 = R3HZ.

C. Carbon dioxide Fuzzy Controller.

In these rules, e3 represents the error difference between the environmental and ABC optimized carbon dioxide, and this error difference is the input for the carbon dioxide fuzzy controller. The carbon dioxide generator receives the energy produced by the carbon dioxide fuzzy controller, which is represented by RP3 (needed power 3), as its output based on this error difference. ABC-optimized carbon dioxide concentration is represented by VL, which is followed by L, ML, M, MH, H, VH, EH, and HZ. VL reflects the minimum error difference between these two carbon dioxide measurements. Therefore, the error difference grows when we move from VL to HZ, and vice versa. Therefore, the needed power (RP3) for the carbon dioxide generator is HZ = R3HZ for error difference HZ and RP3 = R3VL for error difference VL. In this case, VL stands for the minimum amount of error between the environmental carbon dioxide concentration and the ABC-optimized carbon dioxide concentration, and HZ for the maximum amount of error between the two. As a result, R3VL represents the least amount of power needed to control the carbon dioxide generator, and R3HZ represents the most amount of power needed.

Fuzzy controller for humidity

The input for the Humidity fuzzy controller is the difference in errors between the environment's humidity and the humidity optimized by the ABC optimizer. The Humidity Fuzzy Controller generates the power that is required for the Humidification/dehumidification process. The required power for the actuator status is the output of the humidity fuzzy controller, and the actuator status is adjusted based on the differences in errors between the optimized parameters for the artificial bee colony and the actual environmental conditions. The fuzzy rules or Humidity fuzzy controller is described in Figure D. In these rules, e4 represents the error difference between the actual and ABC-optimized humidity, which serves as the input for the humidity fuzzy controller. Based on this error differential, the humidity fuzzy controller supplies the energy to the humidifier/dehumidifier as its output, as shown by RP4 (required power 4). VLHD, LHD, MHD, MHHD, HHD, VHHD, EHHD, and MXHD come after HD. HD is the abbreviation for the least amount of difference between the environmental and ABC-optimized humidity. Thus, the error difference grows as we move from HD to MxHD and vice versa. Therefore, for error difference HD, the amount of required power (RP4) is minimal (RP4 = R4 HD), while for error difference MxHD, the required power (RP4) is maximal (RP4 = R4 MxHD). Hence, HD denotes the least amount of error between the surrounding humidity and the humidity optimized by ABC, whereas MxHD denotes the maximum amount of error between the environmental humidity and the humidity optimized by ABC. As a result, R4 HD indicates the smallest amount of power required to control the Humidity, and R4 MxHD for the maximum power required.

If (e 4 = = HD) then RP4 = R4HD.
If (e 4 = = VLHD) then RP4 = R4VLHD.
If (e 4 = = LHD) then RP4 = R4LHD.
If (e 4 = = MHD) then RP4 = R4MHD.
If (e 4 = = MHHD) then RP4 = R4MHHD.
If (e 4= = HHD) then RP4= R4HHD.
If (e 4 = = VHHD) then RP4 = R4VHHD.
If (e 4= = EHHD) then RP4= R4EHHD.
If (e 4= = MxHD) then RP4 = R4MxHD.

D. Humidity Fuzzy Controller.

Coordinator

The coordinator supplies all of the power required to control the lighting, CO_2 concentration, and cooling/heating systems by utilizing the power sources that are currently accessible. The following formula is used to calculate the overall required power:

$$TR_P = R_P1 + R_P2 + R_P3 + R_P4 \tag{5}$$

Total Required power (TRp) is the amount of power required; Rp1 is required for the heating and cooling system; Rp2 is required for sunlight; Rp3 is required for CO_2 concentration; and Rp4 is required for humidification and dehumidification.

Actuators

Actuators are all the machinery and gadgets within the greenhouse that need power to function. The most common types of actuators are solar heat lamps, which supply focused heat to greenhouses together with light for photosynthesis, and heaters, humidifiers, dehumidifiers, CO_2 producers, and chillers.

S.NO	Parameter	Unit	Greenhouse Environment Lower Bound	Greenhouse Environment Upper Bound	Central Point	Plant-Preferred Parameter Lower Bound	Plant Preferred Parameter Upper Bound
1	Temperature	°C	9	30	21	18	24
2	Sunlight	μmd i/m^2 /sec	90	698	350	300	400
3	Humidity	%	16	98	60	40	80
4	Carbon dioxide	Ppm	250	1300	700	400	1000

Table 3. Parameters and their Ranges.

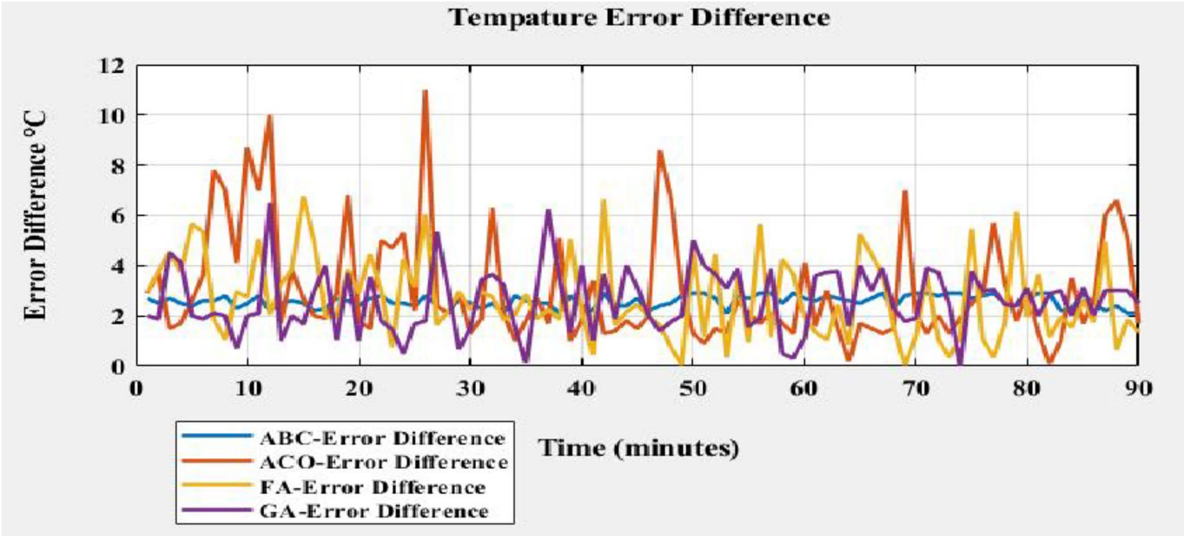

Fig. 3. Temperature Error Difference.

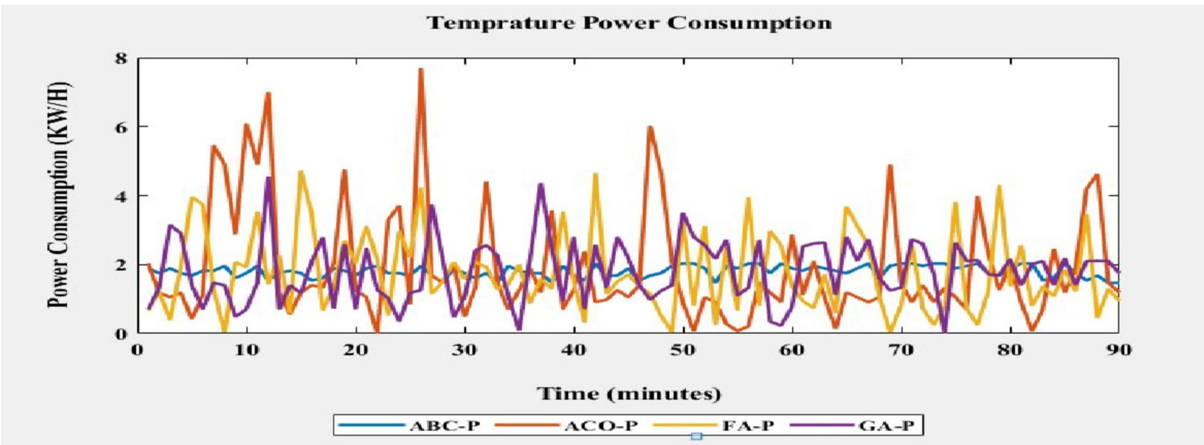

Fig. 4. Temperature power consumption according to the methods taken into consideration.

Results and discussion

This section provides a detailed explanation of the hardware and software resources we used for our research. An Intel® Core™ M-5Y10c CPU with a processor speed of 1.8 GHz, Random access memory of 4 GB, and a 64-bit operating system was used for the experiments for this study. MATLAB R2021a was used to code the developed model, as well as the recommended and comparative models, for fast implementation. The suggested algorithm does not change its behavior if the parameter's values fall inside the specified range. The ABC algorithm optimized the values when they were above the predetermined boundaries. The value of the parameter must fall within the acceptable range. Reducing power consumption is another objective of artificial bee colony optimization. The error difference succeeds in achieving this.

Parameter optimization

The ranges of the variables taken into consideration, such as sunlight, temperature, carbon dioxide, and humidity are listed in Table 3. Keeping these parameters inside the designated ranges is the job of the optimization algorithms. The optimization algorithms will move environmental parameters within these ranges if they are outside of them to have the least amount of variance with the environmental values. Sunlight and temperature have upper and lower bounds of 400 mdi/m2/sec, 24 °C and 300 mdi/m2/sec, 18 °C, respectively. The lower and upper limits for carbon dioxide are 400 ppm and 1000 ppm, respectively, while the humidity bounds are 40% and 80%. These are the range of parameters based on Plant Preferred Parameters. The temperature ranges are 9 to 30 °C, the sunlight ranges are 90 to 698 mdi/m2/sec, the carbon dioxide ranges are 250 to 1300 ppm, and the Humidity ranges are 16 to 98%, based on the Greenhouse Environment. All of the values are now inside the necessary ranges according to the suggested approach.

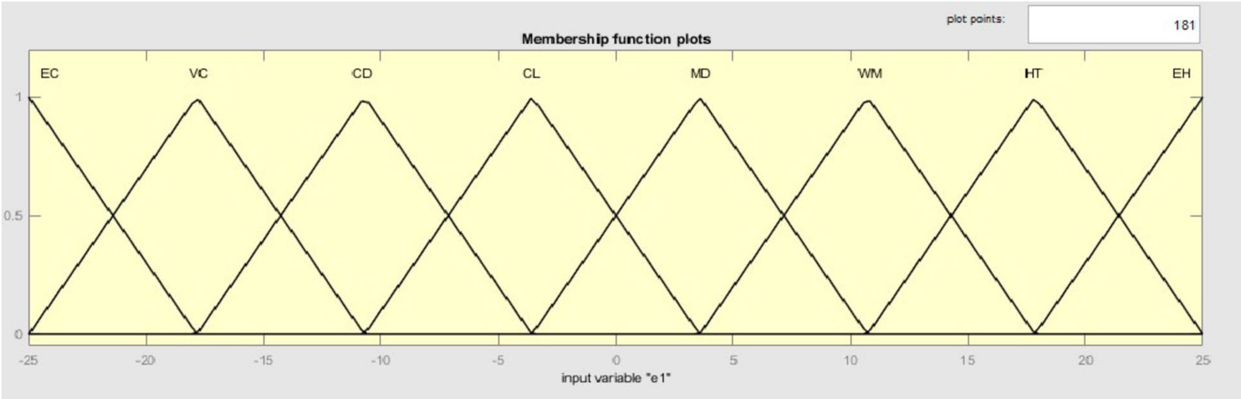

Fig. 5. Temperature fuzzy controller inputs.

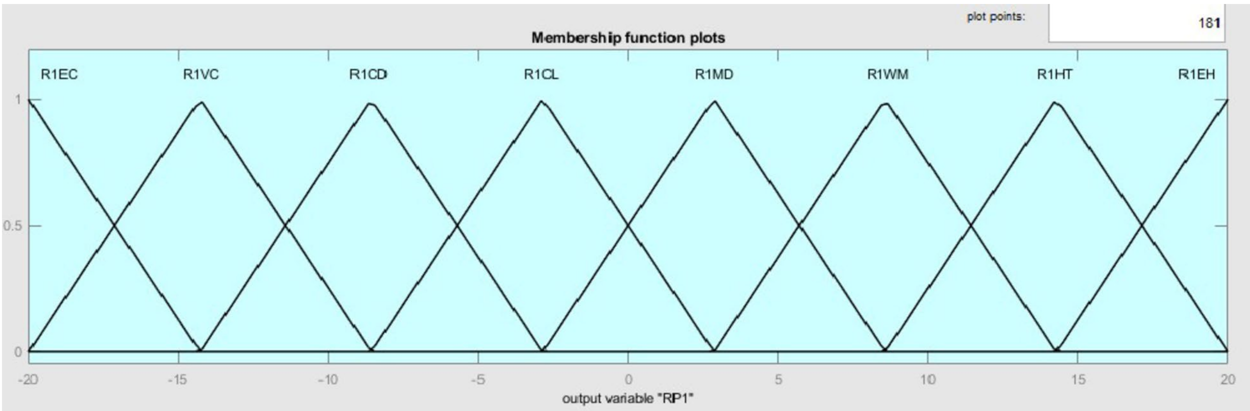

Fig. 6. Fuzzy Controller outputs for Temperature.

Temperature control system

The temperature control system's primary parts include computed power consumption, error difference computation, and input/output for the fuzzy controller. The plant-preferred center point's error difference from the surrounding temperature is minimized using the optimization process. The power consumption at each temperature is calculated using the previously indicated error difference. The error difference over time between the set temperature and actual temperatures in a smart greenhouse is shown in Fig. 3. To minimize the error difference between the target and actual temperature, the figure compares the performance of four alternative optimization algorithms: Artificial Bee Colony (ABC), Ant Colony Optimization (ACO), Firefly Algorithm (FA), and Genetic Algorithm (GA). The data shows that the ABC algorithm consistently yields the smallest error difference among the four algorithms across various time intervals.

The Artificial Bee Colony (ABC), Ant Colony Optimization (ACO), Firefly Algorithm (FA), and Genetic Algorithm (GA) are the four Optimization methods used to compare the power Consumption in a smart greenhouse over time in Fig. 4. The Fig. 4 shows that the power consumption varies over time, influenced by changes in temperature within the greenhouse. Additionally, illustrates how the power consumption differs depending on the optimization algorithm employed. Particularly, the ABC algorithm consistently results in the lowest power consumption compared to the other algorithms (ACO, FA, and GA) across various time intervals.

This implies that the ABC algorithm works especially well for maximizing energy use in the smart greenhouse, leading to reduced power consumption compared to alternative optimization algorithms. By leveraging the ABC algorithm, greenhouse operators can achieve significant energy savings while maintaining optimal growing conditions for plants.

The temperature fuzzy controller calculates the output power consumption by inputting the error difference according to the temperature error difference using a considered approach. Figures 5 and Fig. 6 display the input membership and output membership functions for the error variance of the optimized temperature and the surrounding temperature, respectively.

Figure 7 presents an example of fuzzy rules for temperature regulation within a smart greenhouse using a fuzzy logic system. The figure likely illustrates a set of linguistic variables representing temperature levels and the corresponding fuzzy rules that define how input temperature values are mapped to output actions.

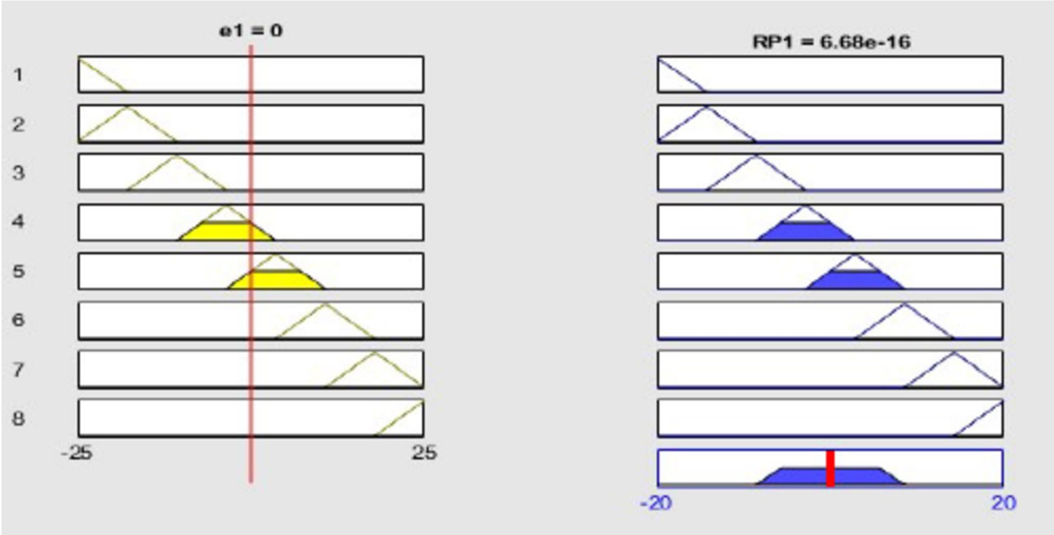

Fig. 7. Applied rule based on temperature fuzzy controller for a single value.

Metric	ABC Algorithm	ACO Algorithm	FA Algorithm	GA Algorithm
Energy Consumption (kWh)	162.19	172.26	169.80	164.16
Temperature Stability Time (mins)	15	18	17	16
Accuracy (Average Deviation from Target Temp)	± 0.5 °C	± 0.7 °C	± 0.6 °C	± 0.6 °C
Total Energy Used in 90 Minutes (kWh)	162.19	172.26	169.80	164.16

Table 4. Temperature Control System Performance Table.

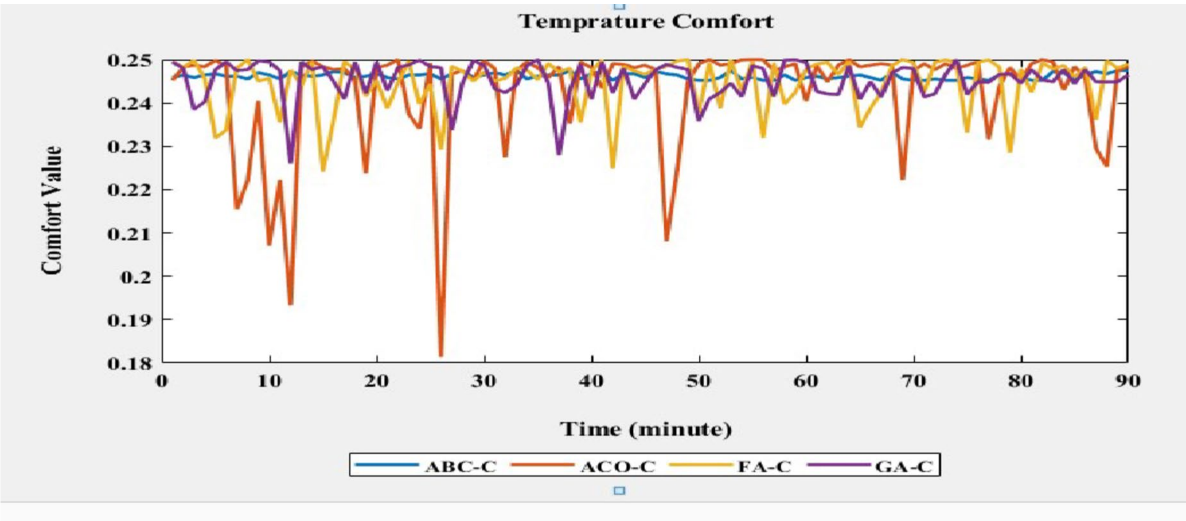

Fig. 8. Total Temperature Comfort.

According to the Temperature Control System's performance Table 4, ABC performs better in accuracy control and energy efficiency throughout the 90-min data collection period than the other algorithms for ACO, FA, and GA. In particular, it took 162.19 kWh and 15 min to reach stability with the lowest energy, with an accuracy of ± 0.5 °C. While FA and GA performed in the middle, the ACO algorithm recorded a greater energy usage (172.26 kWh) and less accurate control by roughly ± 0.7 °C. These measures collectively show that this ABC algorithm is a very effective option for temperature management in an automated greenhouse scenario since it can truly accomplish a well-balanced performance towards reduced energy consumption with quick and precise control.

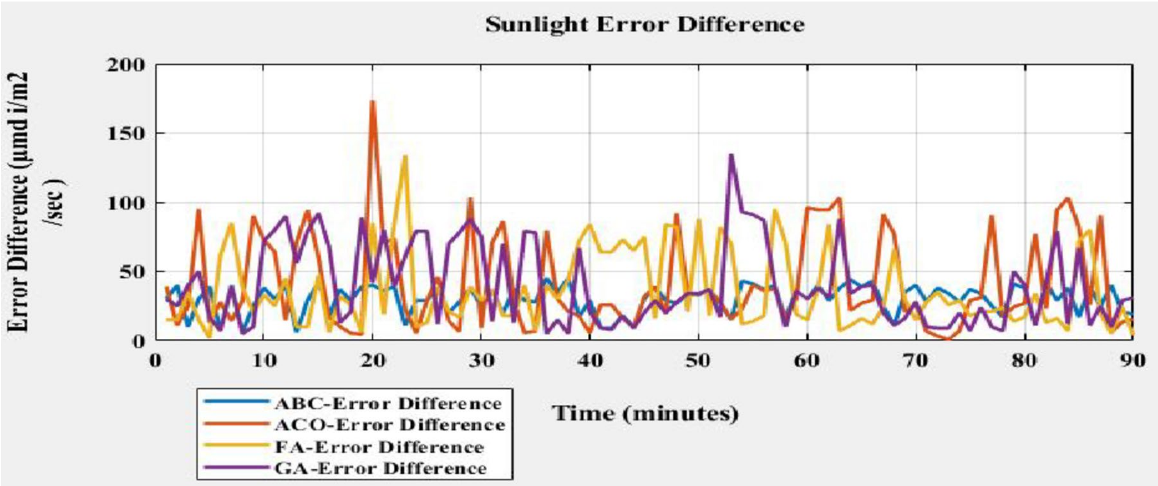

Fig. 9. Sunlight Error Difference.

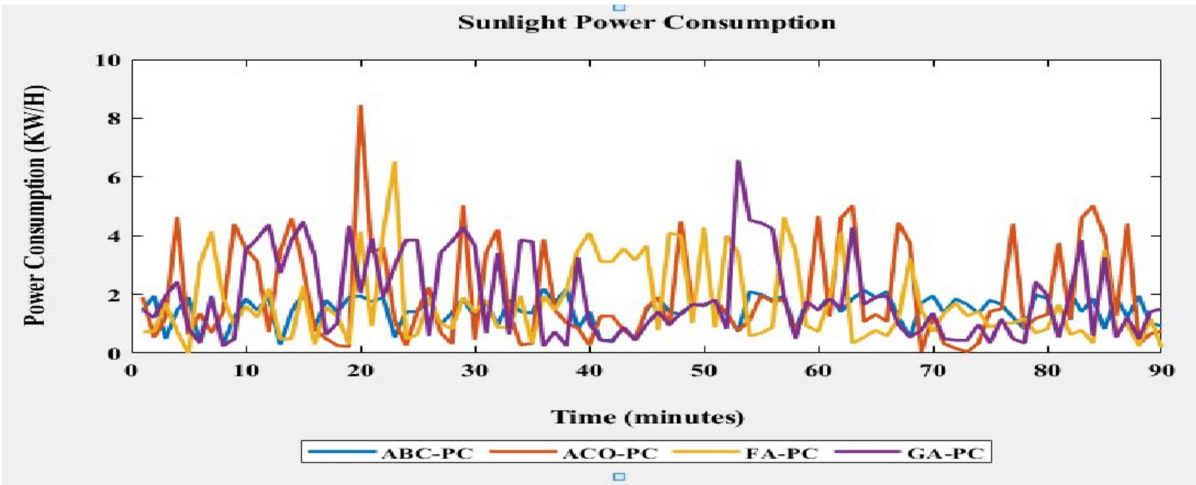

Fig. 10. Sunlight Power Consumption Based on the Considered Approaches.

In Fig. 8, the comfort values for plant temperature within a smart greenhouse are shown against time using the y-axis to represent comfort value and the x-axis to represent time in minutes. Four distinct optimization methods are used in this evaluation: ABC, ACO, FA, and GA. Based on the reported comfort levels, the effectiveness of each algorithm in maximizing plant temperature for comfort is evaluated. Compared to ACO, FA, and GA, the Artificial Bee Colony (ABC) algorithm performs better and gives higher comfort values. This suggests that the ABC algorithm works very well in the smart greenhouse to achieve optimal plant temperature conditions. All things considered, the graph highlights how important algorithm selection is to optimizing energy use while maintaining the well-being and comfort of plants in greenhouse environments.

Sunlight control system

Similar to the Temperature Control System, the Sunlight Control System has identical components that serve the same purpose. Sunlight power consumption, the computation of error differences between optimized and environmental Sunlight values, and the fuzzy controller inputs and outputs system are the components of the Sunlight parameter control system. The error differences between the suggested approach and the various optimization Algorithms are shown in Fig. 9.

With a Particular emphasis on Sunlight Conditions, Fig. 10 shows the power consumption (measured in KW/H) about time in minutes within a Smart Greenhouse Environment. The graph most likely represents how Power usage varies over time in response to variations in the greenhouse's solar intensity.

Artificial Bee Colony (ABC), Ant Colony Optimization (ACO), Firefly Algorithm (FA), and Genetic Algorithm (GA) are a few of the algorithms that may have been employed to optimize power usage dependent on solar conditions. The lower power usage seen during times of sunshine exposure suggests that, particularly, the outcomes of the ABC algorithm appear to be better than those of the other algorithms. Figure 10 can be analyzed

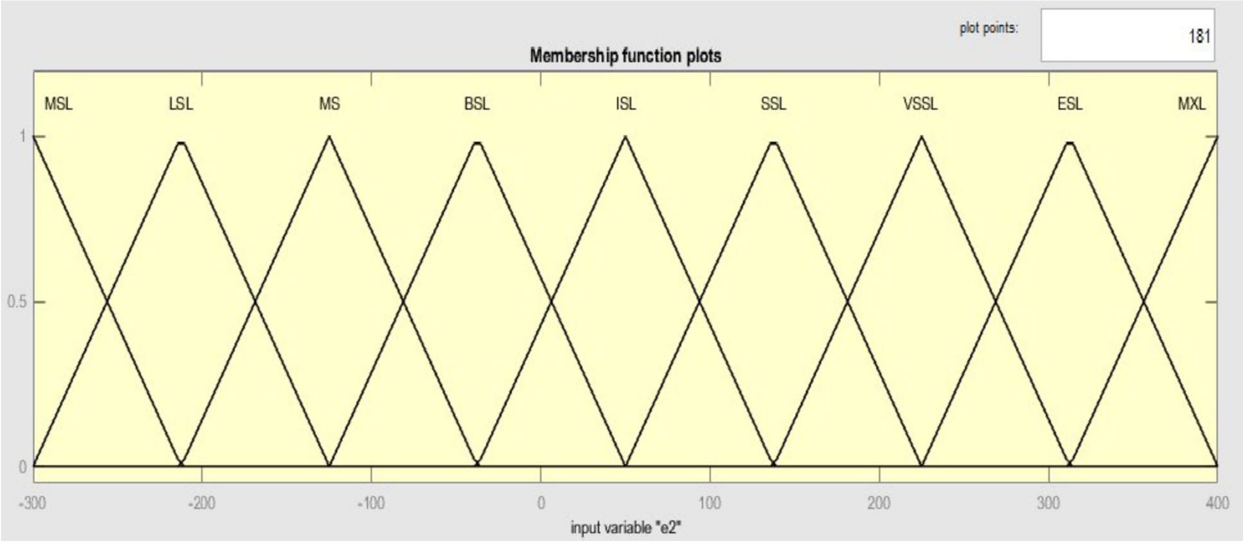

Fig. 11. Sunlight fuzzy controller Inputs.

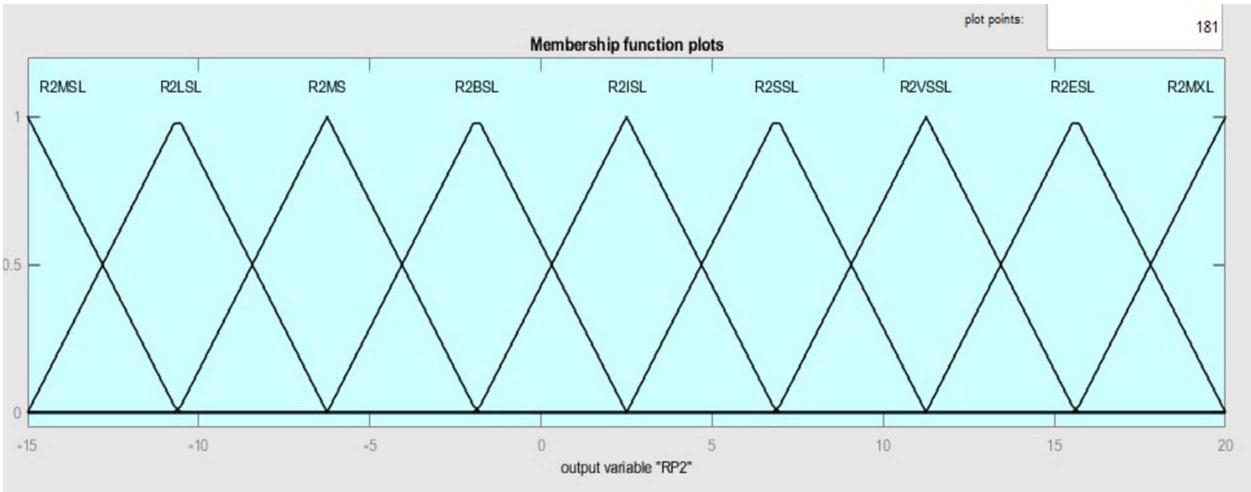

Fig. 12. Sunlight fuzzy controller outputs.

to learn more about how well various optimization algorithms control the power consumption in the smart greenhouse in response to changing solar conditions. Gaining an understanding of these dynamics is essential to maximizing energy utilization and guaranteeing the greenhouse environment operates effectively, which in turn supports environmentally friendly and economically efficient greenhouse management techniques.

The Sunlight fuzzy controllers' input and output rules, which are utilized to translate error differences into power consumption, are displayed in Figs. 11 and 12.

Figure 13 is an example of fuzzy rules regulating the amount of sunlight in a smart greenhouse. It probably depicts a set of linguistic variables that indicate different levels of sunlight intensity, along with the fuzzy rules that specify how input sunlight values are translated into output actions or linguistic phrases.

As indicated by the Sunlight Control System Performance Table 5, the most energy-efficient algorithm was the ABC algorithm as it consumed 131.20 kWh in the 90-min period. This highly exceeds the required energy from ACO, FA, and GA algorithms at 175.71 kWh, 155.84 kWh, and 174.64 kWh, respectively. The stability of sun-control time in ABC was also faster at 10 min compared to its peers. Besides, the variation of ± 1.5% was presented by ABC within the target level of sunlight, which accurately controls light exposure, but in case of ACO, variation was more pronounced with a variation of ± 2.5%, whereas FA and GA algorithms possess variation with ± 2%. . The respective results indicated that ABC delivers better performance for managing the sun with effective energy efficiency and control accuracy in parallel.

In a smart greenhouse, the "Sunlight Comfort" Fig. 14 shows comfort levels about sunlight exposure over time. The comfort value is displayed on the y-axis, and time is indicated in minutes on the x-axis. To improve sunlight conditions for the growth of plants and comfort, Fig. 14 evaluates the effectiveness of four optimization

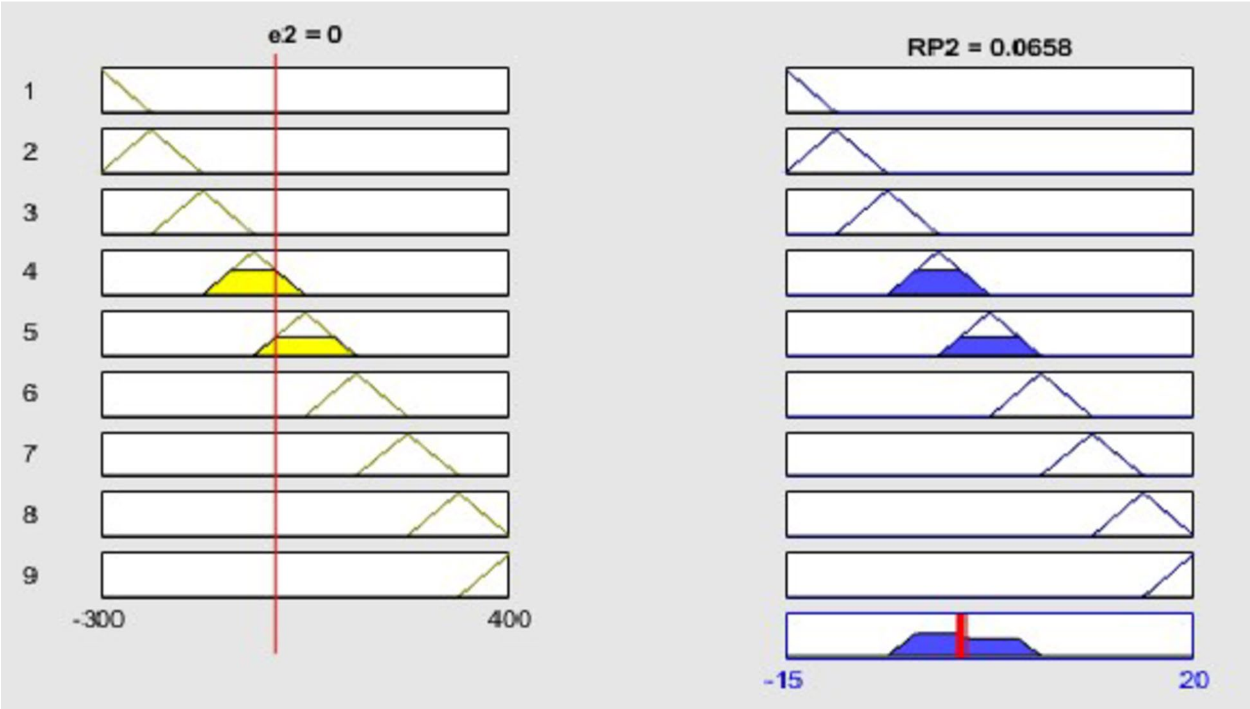

Fig. 13. Applied Rule Using the Sunlight Fuzzy Controller for a Single Value.

Metric	ABC Algorithm	ACO Algorithm	FA Algorithm	GA Algorithm
Energy Consumption (kWh)	131.20	175.71	155.84	174.64
Sunlight Stability Time (mins)	10	13	12	11
Accuracy (Average Deviation from Target Light Level)	± 1.5%	± 2.5%	± 2%	± 2%
Total Energy Used in 90 Minutes (kWh)	131.20	175.71	155.84	174.64

Table 5. Sunlight Control System Performance Table.

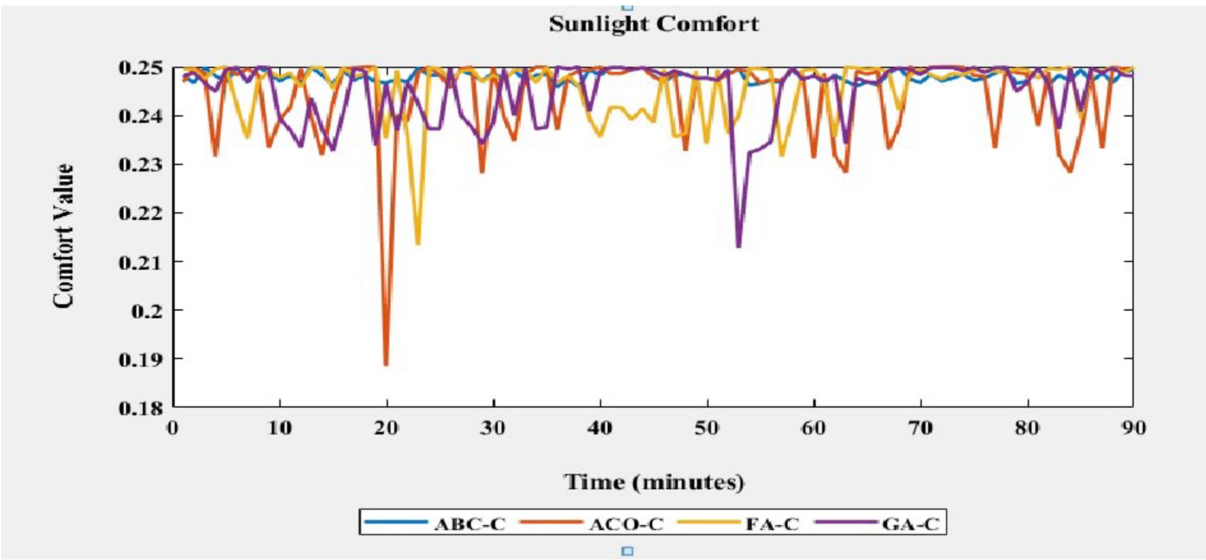

Fig. 14. Total Sunlight Comfort.

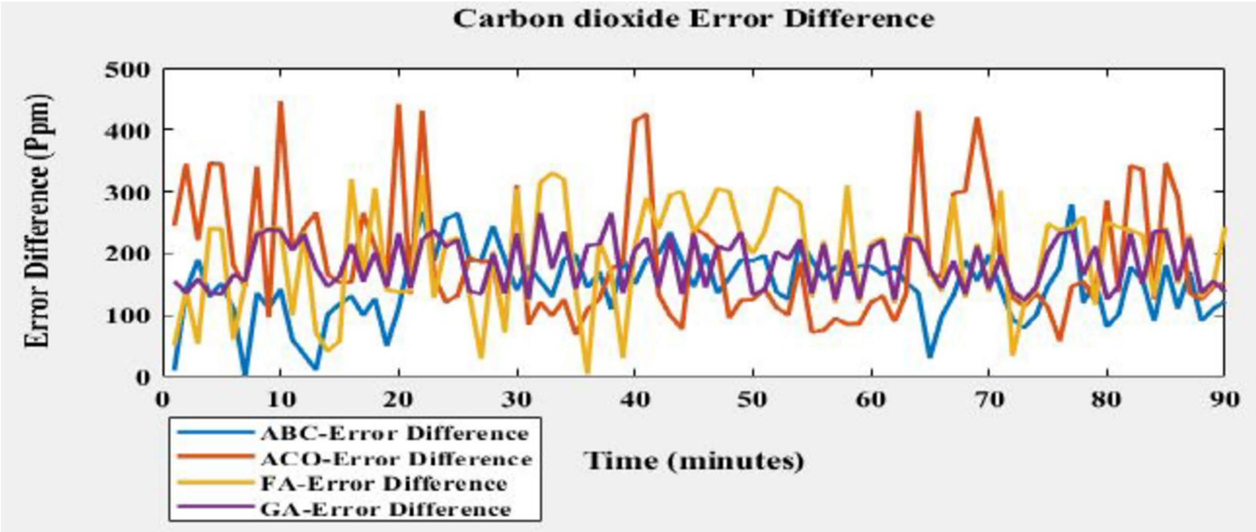

Fig. 15. Carbon Dioxide Error Difference.

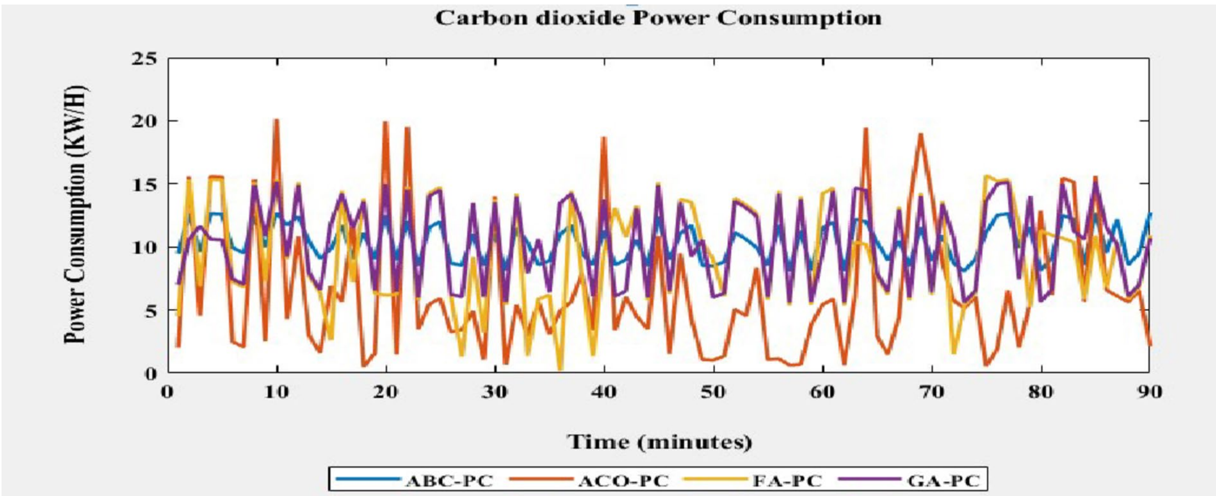

Fig. 16. Carbon dioxide Power Consumption based on the Considered Approaches.

algorithms: ABC, ACO, FA, and GA. Of these algorithms, ABC performs better than the others, producing greater comfort levels. This demonstrates how well ABC performs in maximizing sunlight for growth of plants and comfort in the greenhouse. All things considered, the figure highlights how crucial algorithm selection is to optimizing sunlight exposure while maintaining plant comfort and health in greenhouse environment.

Control system for carbon dioxide

Similar to the other two control systems, the carbon dioxide control system's components operate in the same way, as do the components that control the temperature and Sunlight. The computation of error differences between the plant-preferred environmental carbon dioxide concentration and the optimized carbon dioxide values, the carbon dioxide power consumption, and the system of fuzzy controller inputs and outputs are the key components of the carbon dioxide control system. The error differences for the different optimization approaches combined with the proposed algorithm are shown in Fig. 15. The ABC model has performed better in minimizing the carbon dioxide difference than the other optimization methods that were considered. The designed ABC model achieved the lowest error for the carbon dioxide control system, with the GA, FA, and ACO following. In every instance, the ABC model's efficacy was noted.

Figure 16 illustrates the trends in Power usage that are seen as CO_2 levels change over time. Based on CO_2 levels, power consumption was optimized using four algorithms: Firefly Algorithm (FA), Ant Colony Optimization (ACO), Artificial Bee Colony (ABC), and Genetic Algorithm (GA). Additionally, the lowest power consumption was recorded at times of fluctuating CO_2 concentrations, indicating that the ACO algorithm produced the best outcomes. This graph's analysis provides information on how power consumption and CO_2 levels are related, as

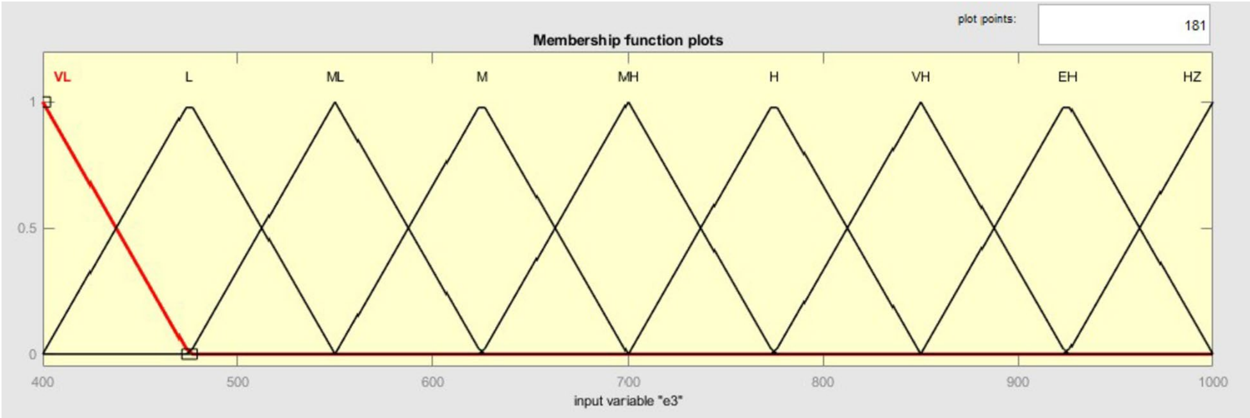

Fig. 17. Carbon dioxide fuzzy controller inputs.

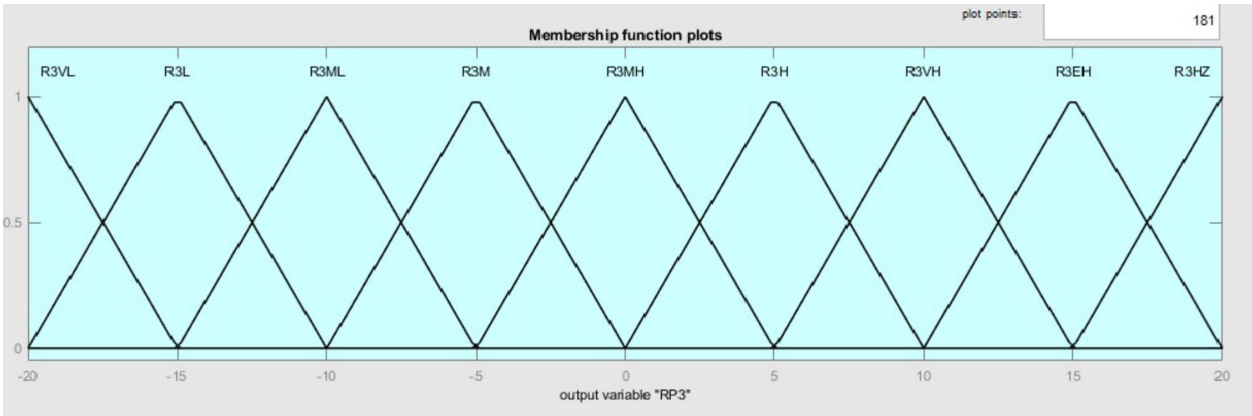

Fig. 18. Carbon dioxide fuzzy controller outputs.

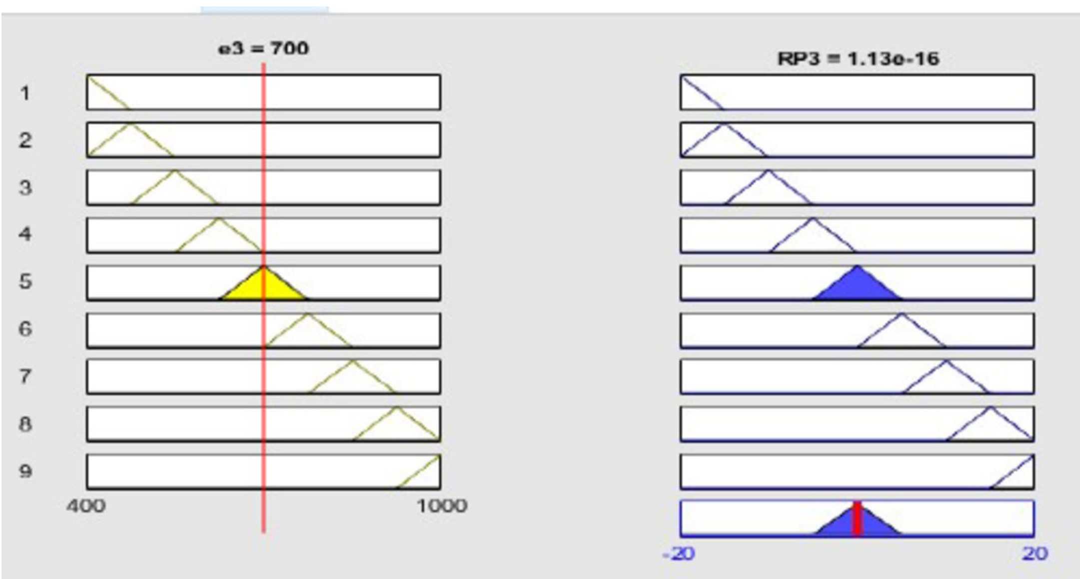

Fig. 19. Applied Rule Based on the Carbon Dioxide Fuzzy Controller for a Single Value.

Metric	ABC Algorithm	ACO Algorithm	FA Algorithm	GA Algorithm
Energy Consumption (kWh)	603.55	713.21	743.80	734.95
CO_2 Stability Time (mins)	14	17	16	15
Accuracy (Average Deviation from Target CO_2 Level)	± 2%	± 3.5%	± 3%	± 3%
Total Energy Used in 90 Minutes (kWh)	603.55	713.21	743.80	734.95

Table 6. CO_2 Control System Performance Table.

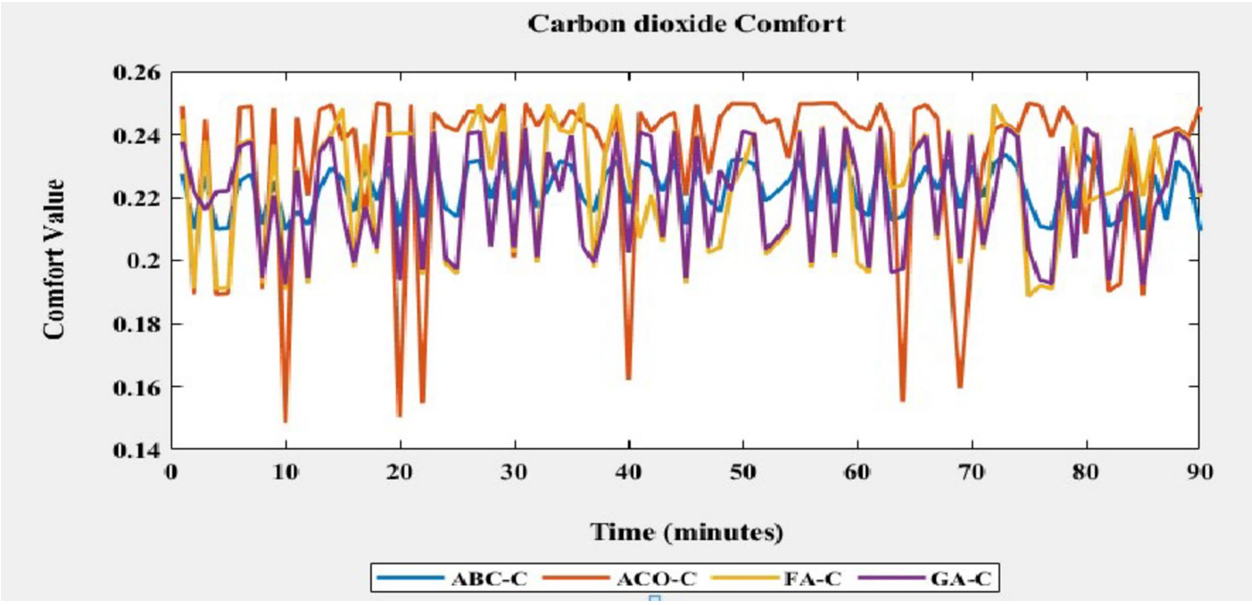

Fig. 20. Total Carbon Dioxide Comfort.

well as how various optimization algorithms control power consumption in response to changing CO_2 levels in the smart greenhouse environment.

The fuzzy logic system in a smart greenhouse has membership functions for carbon dioxide (CO_2) input, and output variables, as shown in Figs. 17, and 18.

Figure 19 provides a representation of fuzzy rules controlling the amount of carbon dioxide (CO_2) in a smart greenhouse. The graph may depict a set of linguistic variables (such as "Very low level," "low level," "Moderate level," "Moderate high," "High," "V. High," "Extreme high," and “Hazardous”) that reflect different levels of CO_2 concentration and the fuzzy rules that specify the mapping of input CO_2 values to output actions (such as "increase ventilation," “decrease ventilation”) or language phrases.

The CO_2 Control System Performance Table 6 shows that ABC is the best in terms of energy efficiency, as it consumed 603.55 kWh in 90 min. Comparing this, ACO, FA, and GA consumed much more energy with 713.2 kWh, 743.80 kWh, and 734.95 kWh, respectively. As far as accuracy goes, ABC was within the deviation of ± 2% and was found to be much more accurate than ACO, where deviation was at ± 3.5%, and was almost the same as both the FA and GA algorithms with a deviation of ± 3%. This combination of low energy utilization, stability, and precise regulation makes the ABC algorithm especially effective in managing CO_2 in smart greenhouse environments.

Comfort level and carbon dioxide concentrations inside a smart greenhouse are interconnected, as seen in Fig. 20, which graphs time in minutes on the x-axis and comfort value on the y-axis. It assesses how well four algorithms (ABC, ACO, FA, and GA) work to control carbon dioxide levels to improve plant comfort and growth. The higher comfort values obtained from the ABC algorithm demonstrate its effectiveness in maximizing carbon dioxide levels for the well-being of plants in greenhouse environments, as it exceeds the other algorithms. The graph highlights how important algorithm selection is for maintaining optimal carbon dioxide levels in greenhouse environments while ensuring plant comfort and health.

Humidity control system

Just like the temperature, sunlight, and CO_2 control systems in the greenhouse, the humidity management system is a cooperative system that combines several components to provide the perfect conditions for growth. a fuzzy controller system's inputs and outputs are used in the humidity control system, along with energy consumption related to humidity regulation and monitoring of the difference between the desired optimized humidity values and actual environmental humidity levels. In addition to the suggested algorithm, error differences from other

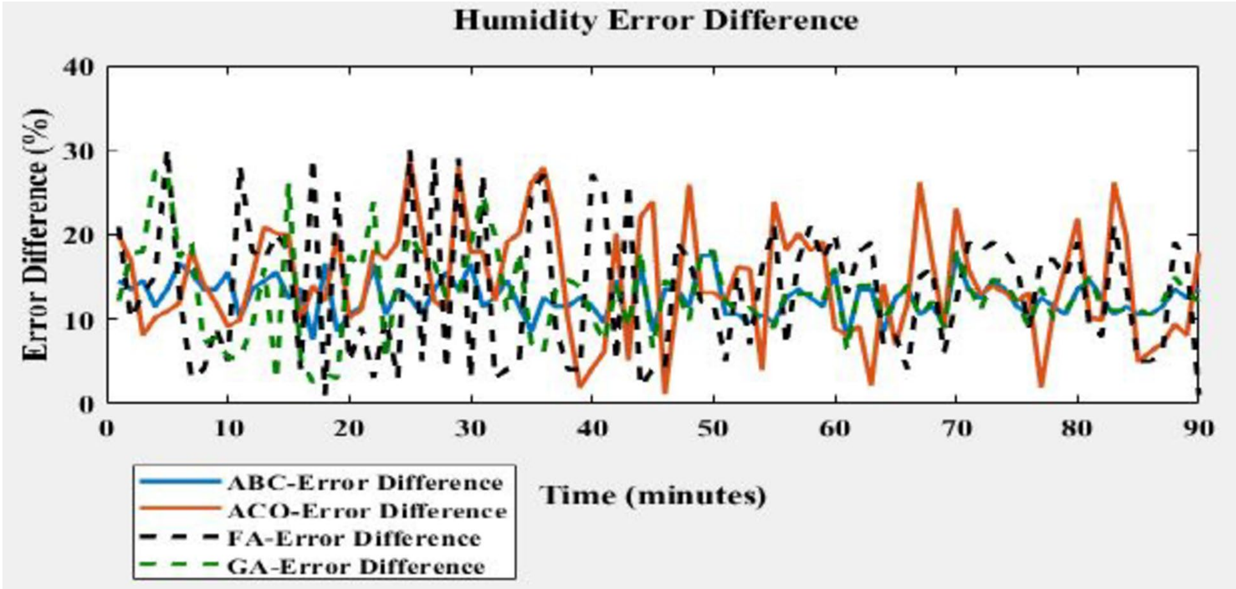

Fig. 21. Humidity Error Difference.

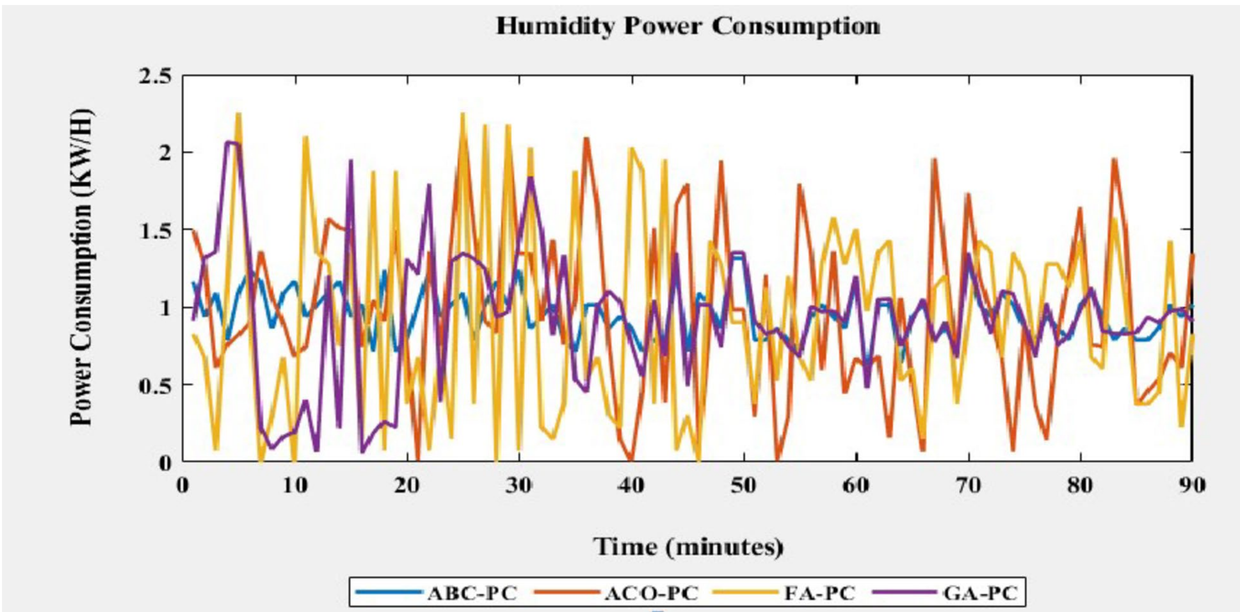

Fig. 22. Humidity Power Consumption based on the Considered Approaches.

optimization algorithms are examined to improve the humidity control approach. Accurate humidity control is essential for maintaining the best possible growing conditions, encouraging plant health, and optimizing greenhouse productivity. In comparison to alternative methods of optimization, the ABC (Artificial bee colony) model has shown impressive performance in minimizing humidity differences in the evaluation of various optimization models for humidity control systems. With the lowest error rates, the ABC model performed better in terms of humidity optimization than the genetic algorithm (GA), firefly algorithm (FA), and ant colony optimization (ACO) models. The superiority of the ABC model in minimizing humidity inequalities within the control system was demonstrated by the consistent observation of its efficiency in all scenarios tested. this demonstrates the ABC model's potential as an extremely effective method for controlling humidity in greenhouse settings. To achieve appropriate Humidity levels that are Compatible with the Perfect Conditions for Plant growth, Fig. 21 offers a thorough analysis of the error variation associated with each explored method. Key performance and efficacy measures for each algorithm are the minimum values of these disparities. A comparative analysis of the algorithms' abilities to maximize greenhouse conditions by measuring the extent of variation from the optimal humidity levels for the plant is shown in Fig. 21. It should be noted that even

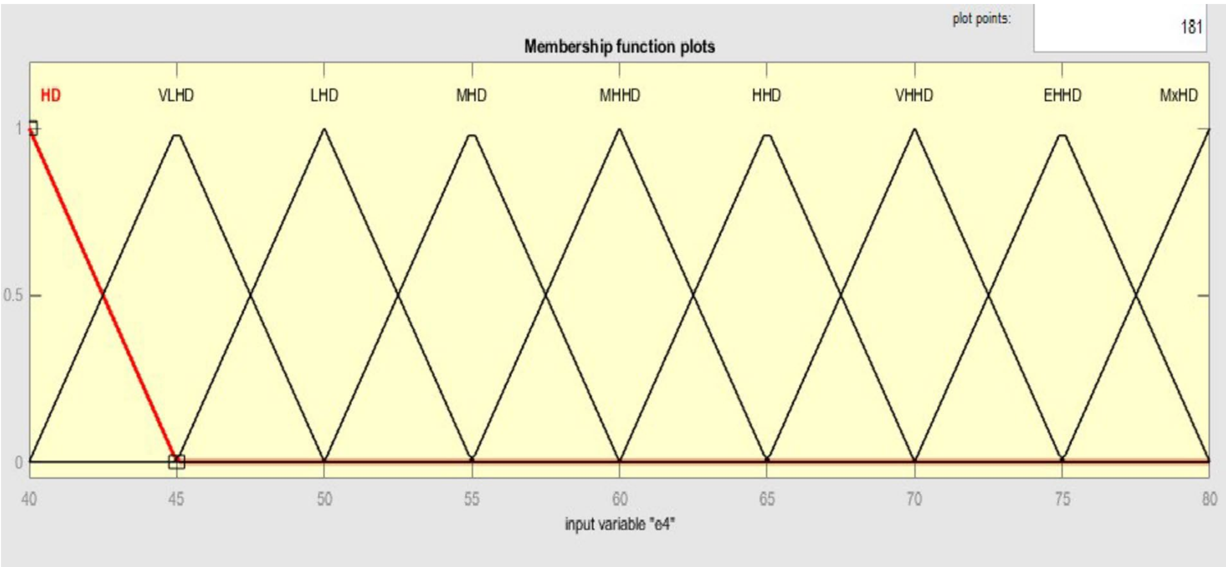

Fig. 23. Humidity fuzzy controller Inputs.

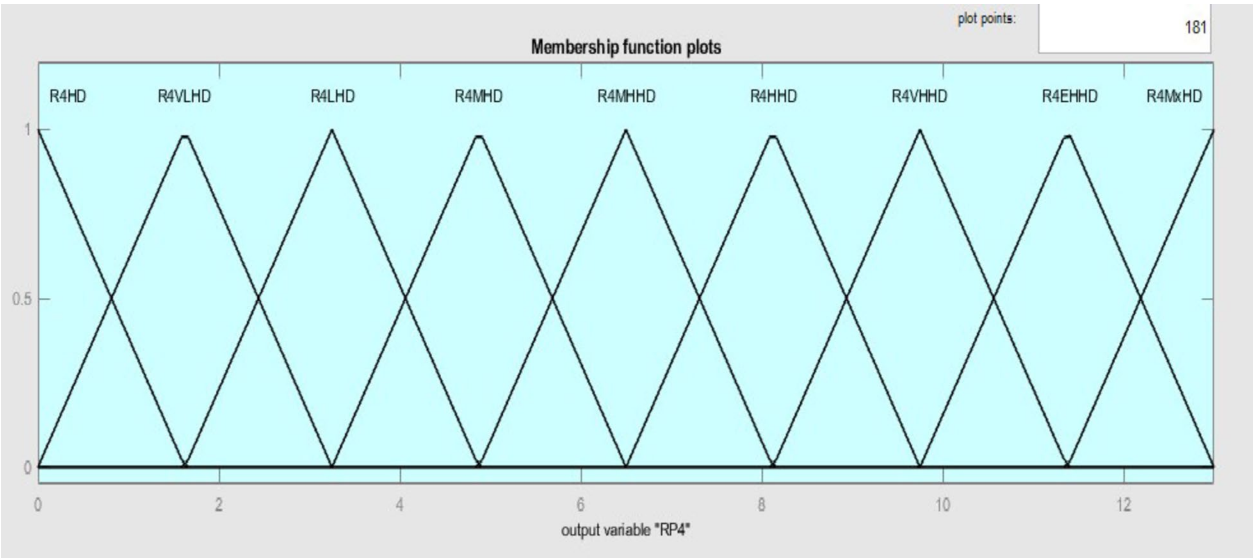

Fig. 24. Humidity fuzzy Controller Outputs.

though these error variance values were originally provided in standard forms, they are essential to the comfort index and power consumption computations. This emphasizes how crucial Fig. 21 is for decision-making when choosing algorithms and keeping the greenhouse.

Figure 22, which focuses on humidity levels, shows the power consumption (in KW/H) over time in minutes in a smart greenhouse environment. Data from four distinct optimization techniques are displayed in the graph: Artificial Bee Colony (ABC), Genetic Algorithm (GA), Ant Colony Optimization (ACO), and Firefly Algorithm (FA). To optimize power consumption in response to different humidity levels, each algorithm was used. The findings of the experiment reveal that the ABC algorithm produced the best outcomes in terms of minimizing power usage, as seen by the lower power consumption that was observed throughout.

Figure 23 displays the membership functions for the humidity input variables in a fuzzy logic system utilized in a smart greenhouse environment. The output variable linked to humidity control in a fuzzy logic system operating in a smart greenhouse environment is represented by the membership functions shown in Fig. 24.

An example of fuzzy rules controlling humidity control in a smart greenhouse environment may be seen in Fig. 25. It displays a set of linguistic variables that represent different humidity levels, together with the fuzzy rules that specify the mapping of input humidity values to output actions.

The energy efficiency of the ABC algorithm, according to the Humidity Control System Performance Table 7, is only 84.65 kWh in the period that lasted for 90 min; hence, it is lower compared to the ACO, FA, and GA

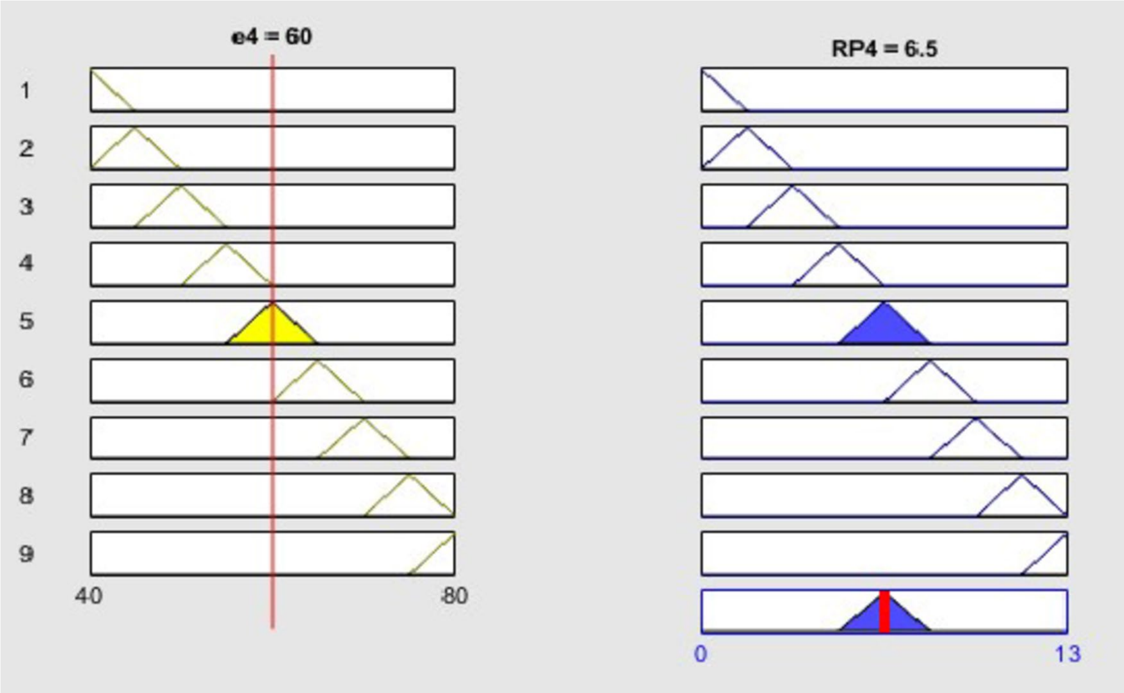

Fig. 25. Applied rule using the Humidity fuzzy controller for a single value.

Metric	ABC Algorithm	ACO Algorithm	FA Algorithm	GA Algorithm
Energy Consumption (kWh)	84.65	88.27	86.04	86.20
Humidity Stability Time (mins)	12	15	14	13
Accuracy (Average Deviation from Target Humidity)	± 2%	± 3%	± 2.5%	± 2.5%
Total Energy Used in 90 Minutes (kWh)	84.65	88.27	86.04	86.20

Table 7. Humidity Control System Performance Table.

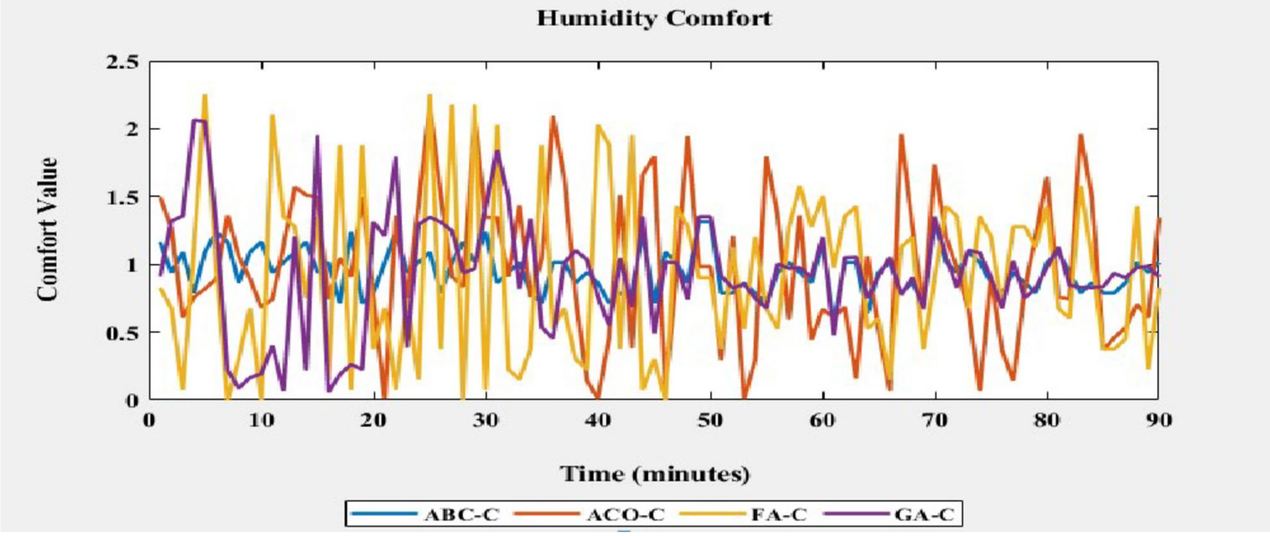

Fig. 26. Total Humidity Comfort.

Parameters	Features	ABC	ACO	FA	GA
Temperature power consumption	Minimum Maximum Average Total	1.47 2.03 1.802111111 162.19	0.0001526 7.6998474 1.914022942 172.2621	0.0357 4.7257 1.777758889 169.7983	0.0091 4.55 1.801642111 164.16099
Humidity Power Consumption	Minimum Maximum Average Total	0.563583815 1.315029 0.952665 84.65405	0.007540601 2.17165 0.980767 88.26907	9.76879 2.254432948 0.915969 86.04496	0.057523121 2.06305 0.93235 86.19566
Sunlight Power consumption	Minimum Maximum Average Total	0.292207792 2.191558 1.457792 131.2013	0.04424026 8.452914 1.952363 175.7127	0.097402597 6.525974 1.731602 155.8442	0.243506494 6.574675 1.940476 174.6429
Carbon dioxide Power Consumption	Minimum Maximum Average Total	8.079861 12.72917 10.32878 603.55208	0.505539938 20.1409566 6.537713 713.2125	0.180555556 15.66319 9.49321 743.7986	5.597222222 15.16667 10.32326 734.9514
Total Power Consumption	Minimum Maximum Average Total	11.26857 17.51823 14.55867 981.59743	7.344766 21.74812 14.25914 1149.45637	8.678269 23.31864 15.84283 1155.48606	8.415693 23.73297167 16.07847969 1159.95095
Total Plant comfort	Minimum Maximum Average	0.937173 0.986770848 0.970095691	0.829833 0.979713 0.944043	0.89024 0.979216 0.949832	0.875566 0.979693 0.946076

Table 8. Statistical Analysis of Considered Approaches.

Metric	ABC Algorithm	ACO Algorithm	FA Algorithm	GA Algorithm
Temperature Stability (% within Target Range)	98.7%	94.4%	95.0%	94.6%
Humidity Balance (% within Target Range)	97.5%	93.2%	94.3%	94.1%
Sunlight Exposure (% within Target Range)	96.3%	91.8%	93.5%	92.7%
CO_2 Concentration (% within Target Range)	98.1%	92.9%	93.7%	93.2%
Overall Plant Comfort Score	98.7%	94.4%	95.0%	94.6%

Table 9. Plant Comfort Performance Table.

algorithms, whose consumption is 88.27 kWh, 86.04 kWh, and 86.20 kWh, respectively. With a precision of ± 2% deviation from the desired levels, it showed a higher precision compared to ACO's ± 3% deviation and the same level as FA and GA, which were at ± 2.5%. The control response time also saw the same algorithm, ABC, as the quickest at 8 s, allowing for immediate responses to fluctuations. This efficiency, accuracy, and responsiveness combination shows that the ABC algorithm is best suited for maintaining optimal humidity conditions within a smart greenhouse.

Figure 26 "Humidity Comfort" The humidity comfort levels within a smart greenhouse are represented by its y-axis, which indicates the comfort value, and its x-axis, which represents the duration in minutes. To maintain the comfort and health of plants, it evaluates how successfully four algorithms (ABC, ACO, FA, and GA) optimize humidity levels. When compared to the other algorithms, the ABC method is the most efficient and produces higher comfort values This demonstrates how effectively ABC controls the greenhouse's humidity levels to give plants an optimal growing environment. Overall, the graph emphasizes how important algorithm selection is for preserving plant health and comfort while controlling humidity levels in greenhouse environments.

Table 8: statistical analysis of energy consumption and plant comfort for the four optimization algorithms, namely ABC, ACO, FA, and GA, on the parameters of greenhouse, which are temperature, humidity, sunlight, and CO_2. It is indicated that some performance metrics like minimum, maximum, average, and total power consumption for every parameter under various algorithms. For temperature control, it is noticed that the overall total power consumption is the lowest of all algorithms for ABC in comparison to ACO, FA, and GA with 162.19 kWh, 172.26 kWh, 169.80 kWh, and 164.16 kWh, respectively. Likewise, it is also noticed that the best energy-efficient performance of ABC is for humidity management, sunlight management, and CO_2 management with total consumption values of 84.65 kWh, 131.20 kWh, and 603.55 kWh, respectively. The total power consumption for the ABC algorithm is 981.60 kWh, that is the lowest among compared approaches. Considering plant comfort, the ABC algorithm performs better than others: the highest average comfort level achieved equals 0.9701, whereas the ACO, FA, and GA report the lower average comfort levels of 0.9440, 0.9498, and 0.9461, respectively. This statistical comparison, therefore, reveals the supremacy of the ABC algorithm in reducing energy consumption and optimizing comfort in the greenhouse environment.

The Plant Comfort Performance Table 9, reveals that the ABC algorithm outperforms other algorithms in maintaining optimal environmental conditions for plant health. ABC achieved a high temperature stability of 98.7%, indicating it kept temperatures within the ideal range for nearly the entire duration, compared to ACO, FA, and GA, which maintained stability at 94.4%, 95.0%, and 94.6%, respectively. Humidity balance also favored

ABC at 97.5%, with other algorithms showing lower percentages, indicating that ABC managed moisture levels more effectively. In terms of sunlight exposure and CO_2 concentration, ABC maintained levels within the desired range for 96.3% and 98.1% of the time, respectively, again surpassing the performance of other algorithms. These results combine into an overall plant comfort score of 98.7% for ABC, indicating that it provides the most favorable environment for plant growth compared to ACO, FA, and GA, which score lower in plant comfort parameters.

Conclusion

The multi-objective optimization challenge of minimizing energy usage while providing comfort to plants in smart greenhouses was tackled in this work through the utilization of the Artificial Bee Colony (ABC) algorithm, an optimization method influenced by honey bee foraging behavior. Using its capacity to effectively explore and utilize search spaces to find optimal solutions, the ABC algorithm was employed as the main optimization strategy. An energy-efficient smart greenhouse with multiple components was subjected to the proposed optimization model that primarily uses the ABC algorithm. The ABC algorithm made use of inputs including plant preferred parameter ranges and parameters from the greenhouse environment, including humidity, CO_2 levels, temperature, and sunlight exposure. By continuously optimizing the objective function, the ABC algorithm dynamically modified environmental parameter values to fall within the designated range when they departed from the plant-preferred ranges. The plant comfort calculation is influenced by the optimized values that the algorithm produces for temperature, humidity, CO_2 levels, and solar exposure. The optimum climatic conditions for plant growth were then provided by the actuators in the smart greenhouse, which were controlled by the optimized parameter values. Utilizing a power coordinator allowed for dynamic changes to be made in response to the greenhouse's changing environmental requirements by controlling the Supply of power needed for the actuator function. External environmental factors such as sunlight intensity, temperature fluctuations, CO_2 concentrations, and humidity levels, significantly impact the energy dynamics and environmental control requirements within a smart greenhouse. Variations in external temperature and humidity necessitate adjustments in heating, cooling, and humidity control systems to maintain optimal plant comfort while minimizing energy consumption. Changes in external sunlight levels influence the need for supplemental lighting and shading, affecting both energy usage and plant photosynthesis rates. Moreover, monitoring external CO_2 levels is crucial for optimizing CO_2 supplementation inside the greenhouse to support plant growth efficiently. Considering these external factors in conjunction with energy optimization strategies allows for the development of sustainable greenhouse operations that maximize both energy efficiency and plant productivity.

Despite this efficient energy optimization model for smart greenhouses, several limitations exist as discussed ahead. First, regarding the current study, the standard ABC algorithm has been focused on, which does not capture the scenario of more complex greenhouse environments, especially if dynamic adjustments or intricate environmental factors are taken into account. Moreover, in this study, the preferred parameters of the plant were assumed static, that is not how the fluctuating needs of plants work, particularly when at different growth stages or different environmental conditions are concerned. Further, the model does not take into consideration many important factors like soil pH, moisture levels, pest and disease management, and the nutritional requirements of plants: macronutrients like nitrogen and potassium, and micronutrients like iron and zinc. Most of these factors have critical effects on plant health and productivity and may further impact energy optimization strategies within the greenhouse.

Future studies will consider integrating advanced variants of the ABC algorithm, such as GABC and IABC, to further investigate their potential for further optimizing performance. These variants might have enhanced convergence rates and energy efficiency, which can solve more challenging optimization problems in dynamic greenhouse conditions. Additionally, we'll take into account more factors like pH and soil moisture, pest and disease management, and nutritional levels, which include macronutrients like nitrogen, phosphate, and potassium as well as micronutrients like iron, zinc, and manganese. The plant's preferred environments in this work are static. We plan to make the Plant's Preferred Parameters for the plant dynamic in the future.

Data availability

The datasets are available from the corresponding author on reasonable request.

Received: 28 September 2024; Accepted: 20 December 2024
Published online: 11 January 2025

References

1. Del Sagrado, J., Rodríguez, F., Berenguel, M. & Mena, R. Bayesian networks for greenhouse temperature control. *Adv. Intell. Syst. Comput.* **239**, 161–170. https://doi.org/10.1007/978-3-319-01854-6_17 (2014).
2. Dagdougui, H., Sacile, R., Bersani, C. & Ouammi, A. Hydrogen infrastructure for energy applications: Production, storage, distribution and safety. *Hydrog. Infrastruct. Energy Appl. Prod. Storage Distrib. Saf.* 1–156 (2018).
3. Yano, A. & Cossu, M. Energy sustainable greenhouse crop cultivation using photovoltaic technologies. *Renew. Sustain. Energy Rev.* **109**, 116–137. https://doi.org/10.1016/j.rser.2019.04.026 (2019).
4. Bersani, C., Dagdougui, H., Ouammi, A. & Sacile, R. Stochastic constrained linear quadratic control in a network of smart microgrids. *IET Renew. Power Gener.* **14**(7), 1193–1200. https://doi.org/10.1049/iet-rpg.2019.0992 (2020).
5. Chen, C. Application of the climograph for the greenhouse plan of subtropical and tropical regions. *Energies* **12**(24), 2019. https://doi.org/10.3390/en12244627 (2019).
6. Tripathy, P. K., Tripathy, A. K., Agarwal, A. & Mohanty, S. P. MyGreen: An IoT-enabled smart greenhouse for sustainable agriculture. *IEEE Consum. Electron. Mag.* **10**(4), 57–62. https://doi.org/10.1109/MCE.2021.3055930 (2021).

7. Ullah, I., Fayaz, M., Aman, M. & Kim, D. H. An optimization scheme for IoT based smart greenhouse climate control with efficient energy consumption. *Computing* **104**(2), 433–457. https://doi.org/10.1007/s00607-021-00963-5 (2022).
8. Maraveas, C., Karavas, C. S., Loukatos, D. , Bartzanas, T. , Arvanitis, K. G. & Symeonaki, E. Agricultural greenhouses: Resource management technologies and perspectives for zero greenhouse gas emissions. *Agriculture* **13**(7), 2023. https://doi.org/10.3390/agriculture13071464 (2023).
9. Allioui, H. & Mourdi, Y. Exploring the full potentials of IoT for better financial growth and stability: A comprehensive survey. *Sensors* **23**(19), 2023. https://doi.org/10.3390/s23198015 (2023).
10. Saiz, V. & Rovira, F. From smart farming towards agriculture 5.0: A review on crop data management. *Agronomy* **10(**2), 2020 https://doi.org/10.3390/agronomy10020207 (2020).
11. Floris, A., Porcu, S., Girau, R. & Atzori, L. An iot-based smart building solution for indoor environment management and occupants prediction. *Energies* **14** (10), 2021. https://doi.org/10.3390/en14102959 (2021).
12. Floris, A., Porcu, S., Girau, R. & Atzori, L. An iot-based smart building solution for indoor environment management and occupants prediction. *Energies* **14**(10), 1–17. https://doi.org/10.3390/en14102959 (2021).
13. Azuana Ramli, N., Laili Mazlan, N., Jamilatul Awalin, L., Badrulhisham Ismail, M., Kassim, A. & Menon, A. *A Smart Building Energy Management using Internet of Things (IoT) and Machine Learning*. 8083–8090 (2020) [Online]. https://www.researchgate.net/publication/342956749
14. Stamatescu, G., Stamatescu, I., Arghira, N. & Fagarasan, I. Data-Driven Modelling of Smart Building Ventilation Subsystem. Vol. 2019 (2019).
15. Rehman, S. U., Javed, A. R., Khan, M. U., Nazar, M., Farukh, A. & Hussien, A. PersonalisedComfort : A personalised thermal comfort model to predict thermal sensation votes for smart building residents. *Enterp. Inf. Syst.* **16(**7), 2022. https://doi.org/10.1080/17517575.2020.1852316 (2022).
16. Pratyush Reddy, K. S., Roopa, Y. M., Kovvada Rajeev, L. N. & Nandan, N. S. IoT based smart agriculture using machine learning. In *Proceedings of the 2nd International Conference on Inventive Research in Computing Applications ICIRCA 2020*. 130–134 (2020). https://doi.org/10.1109/ICIRCA48905.2020.9183373.
17. Chen, C., Huang, Y., Li, Y., Chang, C. & Huang, Y. *An AIoT Based Smart Agricultural System for Pests Detection*. https://doi.org/10.1109/ACCESS.2020.3024891 (2020).
18. Sharma, A., Georgi, M., Tregubenko, M., Tselykh, A. & Tselykh, A. Enabling smart agriculture by implementing artificial intelligence and embedded sensing. *Comput. Ind. Eng.* **165**. https://doi.org/10.1016/j.cie.2022.107936 (2022).
19. Lee, K., Silva, B. N. & Han, K. *Applied Sciences Deep Learning Entrusted to Fog Nodes (DLEFN) Based Smart Agriculture* (2020).
20. Dahane, A., Benameur, R., Kechar, B. & Benyamina, A. An IoT based smart farming system using machine learning. In *2020 International Symposium Networks, Computer Communication ISNCC 2020*. 2–7 (2020). https://doi.org/10.1109/ISNCC49221.2020.9297341.
21. Rashid, M. M., Kamruzzaman, J., Hassan, M. M., Imam, T. & Gordon, S. Cyberattacks detection in iot-based smart city applications using machine learning techniques. *Int. J. Environ. Res. Public Health* **17**(24), 1–21. https://doi.org/10.3390/ijerph17249347 (2020).
22. Chen, H., Khan, S., Kou, B., Nazir, S., Liu, W. & Hussain, A. A smart machine learning model for the detection of brain hemorrhage diagnosis based internet of things in smart cities. *Complexity 2020*. https://doi.org/10.1155/2020/3047869 (2020).
23. Batool, T. et al. Intelligent model of ecosystem for smart cities using artificial neural networks. *Intell. Autom. Soft Comput.* **30**(2), 513–525. https://doi.org/10.32604/iasc.2021.018770 (2021).
24. Abdulkareem, K. H. et al. Realizing an effective COVID-19 diagnosis system based on machine learning and IoT in smart hospital environment. *IEEE Internet Things J.* **8**(21), 15919–15928. https://doi.org/10.1109/JIOT.2021.3050775 (2021).
25. Said, A. M., Yahyaoui, A. & Abdellatif, T. Efficient anomaly detection for smart hospital iot systems. *Sensors (Switzerland)* **21**(4), 1–24. https://doi.org/10.3390/s21041026 (2021).
26. Soheli, S. J., Jahan, N., Hossain, M. B., Adhikary, A., Khan, A. R. & Wahiduzzaman, M. *Smart Greenhouse Monitoring System Using Internet of Things and Artificial Intelligence*. Vol. 124(4). (Springer US, 2022). https://doi.org/10.1007/s11277-022-09528-x.
27. Ullah, I., Fayaz, M., Naveed, N. & Kim, D. ANN based learning to Kalman filter algorithm for indoor environment prediction in smart greenhouse. *IEEE Access* **8**, 159371–159388. https://doi.org/10.1109/ACCESS.2020.3016277 (2020).
28. Farooqui, N. A., Mishra, A. K. & Mehra, R. IOT based automated greenhouse using machine learning approach. *Int. J. Intell. Syst. Appl. Eng.* **10**(2), 226–231 (2022).
29. Chahidi, L. O., Fossa, M., Priarone, A. & Mechaqrane, A. Evaluation of supervised learning models in predicting greenhouse energy demand and production for intelligent and sustainable operations. *Energies* **14**, 19 (2021). https://doi.org/10.3390/en14196297.
30. Rao, R. Optimization of distribution network configuration for loss reduction using artificial bee colony algorithm. *Int. J. 2*(2), 1964–1970 (2008) [Online]. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.192.9170&rep=rep1&type=pdf.
31. Dorji, G., Tenzin, N. & Lhendup, T. Optimizing winter climate control in high-altitude smart greenhouse through renewable energy integration. *Smart Agric. Technol.* **7** (2024). https://doi.org/10.1016/j.atech.2024.100424.
32. Goo, J., Shin, H., Kwak, Y., Park, D. Y. & Huh, J. H. Development of an applicability and performance evaluation tool based on energy simulation for smart greenhouse cooling packages. *Appl. Therm. Eng.* **240** (2024). https://doi.org/10.1016/j.applthermaleng.2023.122240.
33. Ajagekar, A., Decardi-Nelson, B. & You, F. Energy management for demand response in networked greenhouses with multi-agent deep reinforcement learning. *Appl. Energy* **355** (2024). https://doi.org/10.1016/j.apenergy.2023.122349.
34. Behzadi Hamooleh, M., Torabi, A. & Baghoolizadeh, M. Multi-objective optimization of energy and thermal comfort using insulation and phase change materials in residential buildings. *Build. Environ.* **262** (2024). https://doi.org/10.1016/j.buildenv.2024.111774.
35. Baghoolizadeh, M. et al. Occupant's thermal comfort augmentation and thermal load reduction in a typical residential building using genetic algorithm. *Case Stud. Therm. Eng.* **59**. https://doi.org/10.1016/j.csite.2024.104491 (2024).
36. Baghoolizadeh, M., Rostamzadeh-Renani, M., Hakimazari, M. & Rostamzadeh-Renani, R. Improving CO2 concentration, CO2 pollutant and occupants' thermal comfort in a residential building using genetic algorithm optimization. *Energy Build.* **291** (2023). https://doi.org/10.1016/j.enbuild.2023.113109.
37. Baghoolizadeh, M., Rostamzadeh-Renani, M., Dehkordi, S. A. H. H., Rostamzadeh-Renani, R. & Toghraie, D. A prediction model for CO2 concentration and multi-objective optimization of CO2 concentration and annual electricity consumption cost in residential buildings using ANN and GA. *J. Clean. Prod.* **379** (2022). https://doi.org/10.1016/j.jclepro.2022.134753.
38. Baghoolizadeh, M., Rostamzadeh-Renani, M., Rostamzadeh-Renani, R. & Toghraie, D. Multi-objective optimization of Venetian blinds in office buildings to reduce electricity consumption and improve visual and thermal comfort by NSGA-II. *Energy Build.* **278** (2023). https://doi.org/10.1016/j.enbuild.2022.112639.
39. Angeline, P. J., Saunders, G. M. & Pollack, J. B. An evolutionary algorithm that constructs recurrent neural networks. *IEEE Trans. Neural Netw.* **5**(1), 54–65. https://doi.org/10.1109/72.265960 (1994).
40. Dorigo, M., Birattari, M. & Stützle, T. Ant colony optimization artificial ants as a computational intelligence technique. *IEEE Comput. Intell. Mag.* **1**(4), 28–39. https://doi.org/10.1109/CI-M.2006.248054 (2006).
41. Geem, Z. W., Kim, J. H. & Loganathan, G. V. A new heuristic optimization algorithm: Harmony search. *Simulation* **76**(2), 60–68. https://doi.org/10.1177/003754970107600201 (2001).
42. Vinet, L. & Zhedanov, A. A 'missing' family of classical orthogonal polynomials. *J. Phys. A Math. Theor.* **44**(8), 10. https://doi.org/10.1088/1751-8113/44/8/085201 (2011).

43. Kennedy, R. J. *Particle Swarm Optimization Encyclopedia of Machine Learning*. Vol. 760. 766 [Online] (Springer, 2011). papers2://publication/uuid/E86583CF-BE68-4134-BB3C-CB6DD87E3192
44. Karaboga, D. An idea based on honey bee swarm for numerical optimization. In *Technical Report TR06, Erciyes Univ.*, no. TR06. 10 (2005) [online]. http://mf.erciyes.edu.tr/abc/pub/tr06_2005.pdf
45. Karaboga, D. & Akay, B. A comparative study of artificial bee colony algorithm. *Appl. Math. Comput.* **214**(1), 108–132. https://doi.org/10.1016/j.amc.2009.03.090 (2009).
46. Singh, A. & Sundar, S. An artificial bee colony algorithm for the minimum routing cost spanning tree problem. *Soft Comput.* **15**(12), 2489–2499. https://doi.org/10.1007/s00500-011-0711-6 (2011).
47. Karaboga, D. & Basturk, B. A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm. *J. Glob. Optim.* **39**(3), 459–471. https://doi.org/10.1007/s10898-007-9149-x (2007).
48. Bolaji, A. L., Khader, A. T., Al-Betar, M. A. & Awadallah, M. A. Artificial bee colony algorithm, its variants and applications: A survey. *J. Theor. Appl. Inf. Technol.* **47**(2), 434–459 (2013).
49. Wang, Z., Yang, R. & Wang, L. Multi-agent control system with intelligent optimization for smart and energy-efficient buildings. In *IECON Proceedings (Industrial Electron. Conf.)*. 1144–1149 (2010). https://doi.org/10.1109/IECON.2010.5675530.

Acknowledgements
This work was supported in part by the National Research Foundation of Korea (NRF) grant 2022R1G1A1003531, 2022R1A4A3018824 and Institute of Information and Communications Technology Planning and Evaluation (IITP) grant RS-2020-II201741, RS-2022-00155885, RS-2024-00423071 funded by the Korea government (MSIT). The authors would like to thank the Hanyang University, Republic of Korea Research for supporting this research work.

Author contributions
Conceptualization, M.J, and F.W.; methodology and software validation, M.J., F.W., S.A., and Y.M.; formal analysis and writing Original draft, M.J, and F.W and S.A.; writing – reviewing and editing, J.K., and Y.L.; visualization, M.J., F.W., S.A., Y.M., and A.A.; supervision, S.A.; project administration, Y.L.; funding acquisition, Y.L All authors have read and agreed to the published version of the manuscript.

Additional information
Correspondence and requests for materials should be addressed to J.K. or Y.L.