Optimal Environment Control Mechanism Based on OCF Connectivity for Efficient Energy Consumption in Greenhouse

Atif Rizwan, Anam Nawaz Khan, Rashid Ahmad, and Do Hyeun Kim

Abstract—**Greenhouses are a productive system that allows us to respond to the growing global demand for fresh and healthy food throughout the year, but the greenhouse environment is not easily controlled because its climate parameters are interrelated. However, the numbers of the actuator are operated parallelly to maintain the greenhouse environment; as a result, the energy consumption of greenhouses is high. In this study, we presented the optimization module by considering the outdoor environment with the aim of minimum energy consumption. Metaheuristic-based differential evaluation (DE) is used to optimize the climate parameters by considering indoor and outdoor environmental constraints. Furthermore, the long short-term memory (LSTM)-based inference model is offloaded on the Internet of Things (IoT) device to predict the next environmental situation. The objective function selects the optimal parameters within user preferences with minimum energy consumption based on the inferred parameter value. The open-source software framework IoTivity, implementing open connectivity foundation (OCF) technical standards, is used for the real-time connection between IoT devices and the IoT platform. Greenhouse owners can set the preferences based on the requirements of plants in the greenhouse by using a smart and remotely accessible Android-based interface. A fuzzy logic-based control module operates on an IoT device that maps the optimized parameters with the actuator and operates accordingly. The proposed model is analyzed, and the performance is evaluated in terms of energy consumption for each climate parameter and actuator in the greenhouse. The results show that the proposed mechanism saves 36% of energy.**

Index Terms—**Energy optimization, greenhouse, inference mechanism, open connectivity foundation (OCF) IoTivity, optimization problems.**

Manuscript received 6 January 2022; revised 13 October 2022; accepted 10 November 2022. Date of publication 15 November 2022; date of current version 7 March 2023. This work was supported in part by the Energy Cloud Research and Development Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT under Grant 2019M3F2A1073387; in part by the Institute of Information & Communications Technology Planning & Evaluation (IITP) Grant funded by the Korea Government (MSIT) under Grant 2021-0-00188 (Open Source Development and Standardization for AI Enabled IoT Platforms and Interworking); and in part by the Institute for Information & Communications Technology Promotion (IITP) under Grant 2022-0-00980 (Cooperative Intelligence Framework of Scene Perception for Autonomous IoT Device). *(Corresponding author: Do Hyeun Kim.)*

Atif Rizwan and Anam Nawaz Khan are with the Department of Computer Engineering, Jeju National University, Jeju 63243, Republic of Korea.

Rashid Ahmad is with the Department of Computer Science, COMSATS University Islamabad (Attock Campus), Attock 43600, Pakistan, and also with the Bigdata Research Center, Jeju National University, Jeju 63243, Republic of Korea.

Do Hyeun Kim is with the Department of Computer Engineering and the Advanced Technology Research Institute, Jeju National University, Jeju 63243, Republic of Korea (e-mail: kimdh@jejunu.ac.kr).

Digital Object Identifier 10.1109/JIOT.2022.3222086

I. Introduction

ENERGY saving is crucial for all sectors, especially the greenhouse, due to their unequivocal role in total energy consumption. Due to long-term temperature alterations, arable land reduction, and exploding population size, sustainable food production with optimal energy use has evolved from technological fringe to today's reality [1]. Since its inception greenhouse farming industry has played a key role in sustainable food production by providing a year-round production facility for fresh vegetables with around 50% alongside strengthening the global economy, ensuring food security and stability for replenishing the food needs across the globe [2]. Greenhouse tries to provide optimal conditions for cultivating crops to facilitate the growth cycle in a seasonal period of unsuitable plant growth, promoting sustainable food and energy provision. Planting crops requires special environmental conditions, such as temperature, humidity CO_2, etc., which cannot be achieved through traditional manual control methods relying upon a nonscientific basis. Due to rapid urbanization, global climate changes, and water and energy resource shortages, traditional farming methods are unable to meet the growing global food needs in a sustainable manner. Recently, the farming industry has gradually developed into a larger-scale planting sector. Thus, cultivating crops and plants in a greenhouse can be an effective solution for meeting environmental regulations along with optimizing energy use. However, greenhouse farming suffers from the problems of efficient operations, and resource management [3]. Some of the challenges involve extensive monitoring and optimal control of the greenhouse environment with minimum energy consumption and maximum crop yield and quality.

According to the findings of the study, [4], the appropriate planning for a conventional greenhouse can consume up to 80% of energy. Moreover, the facade material involved in the construction of greenhouses provides insufficient thermal resistance, which becomes a notable cause of heating and cooling energy demands. Alongside the physical structure and materials, the cultivation types also greatly influence energy consumption. Furthermore, the climatic region of the greenhouse greatly influences its energy consumption. Considering the rising energy prices, depletion of nonrenewable resources, carbon emissions, global warming, and climate change, energy saving has become a de facto necessity. Energy is a scarce resource; thus, strategies for optimizing energy consumption

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

in the greenhouse are being considered worldwide. Recent studies revealed that energy produced by renewable resources could fulfil 14% of total energy demand [5]. In this regard, optimizing the energy use in smart greenhouses can be treated as a potential energy-saving opportunity for effectively minimizing energy wastage and mitigating carbon emissions [6]. A reduction of the rapidly growing energy need for greenhouses is imperative for achieving sustainable cultivation [7].

To resolve the aforementioned challenges, the integration of agricultural Internet of Things (IoT) in the greenhouse can be a potential solution. In the era of technological advancements, the greenhouse has emerged as an active actor in the agricultural industry. It is a major contributor to precision agriculture based on advanced environmental control techniques, wireless sensor networks, management, communication and metering infrastructure. Smart greenhouses are equipped with heating, ventilation and air-conditioning components, supply air fans, CO_2 generators, physical sensors, actuators and a standard communication infrastructure constituting the agricultural IoT. The agricultural IoT-based greenhouse provides monitoring and ease of access to environmental resources for subsequent data acquisition through physical sensors [8]. IoT-empowered greenhouse facilitates farm owners in efficient decision-making and resource utilization with little human interventions. In terms of software implementation of optimal control in the smart greenhouse, technologies, such as Arduino and Raspberry Pi have received tremendous attention. However, constrained devices may suffer from challenges, such as privacy, storage, security and network communication abilities. To overcome such issues, open connectivity foundation (OCF) enables the resource-constrained devices deployed in the smart greenhouse to communicate securely and reliably with enhanced accessibility and seamless connectivity. The open-source software framework IoTivity implementing OCF provides a customized protocol for the constrained hardware, ensuring lower round trip time and smaller packet size employing Constrained Application Protocols (CoAPs) [9]. Hence, there is a dire need to develop an agricultural IoT system that can optimally control the greenhouse environment through efficient utilization of resources. As optimal resource utilization results in maximum crop yield along with cost reduction through optimal energy use. Furthermore, the collected environmental data based on OCF-IoTivity helps farmers in making accurate and timely decisions.

The study presents the optimization of energy consumption by using outdoor climate parameters. The application programming interface (API) from the Korean Meteorological Administration (KMA) is used to get the outdoor parameters, and the sensors are placed inside the greenhouse for indoor environment situation. The climate parameters for the time $(t + 1)$ are predicted using the TFLite model based on long short-term memory (LSTM) artificial recurrent neural network (RNN) architecture. The model is first trained on the IoT platform and offloaded to the IoT device. Then, the values of temperature, humidity, and CO_2 are sent to the IoT platform to get the optimal combination of these parameters within user preferences based on the outdoor environment and energy price. The real-time price of energy is obtained from a smart

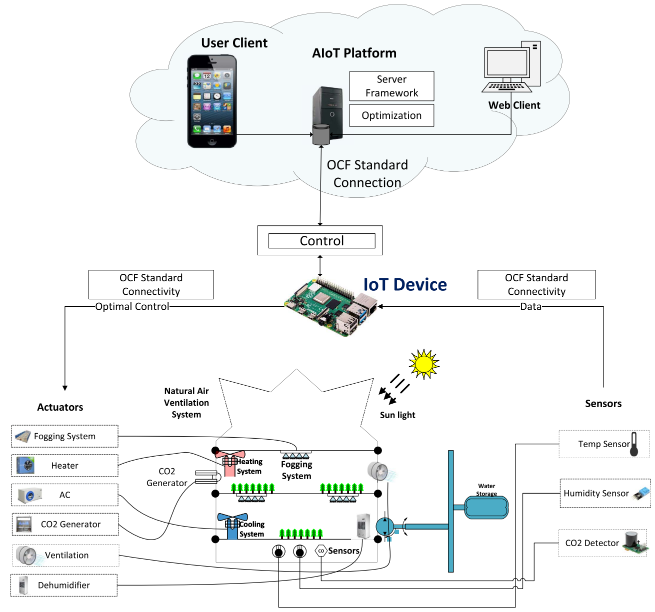


Fig. 1. System configuration for optimal control in greenhouse IoT networks.

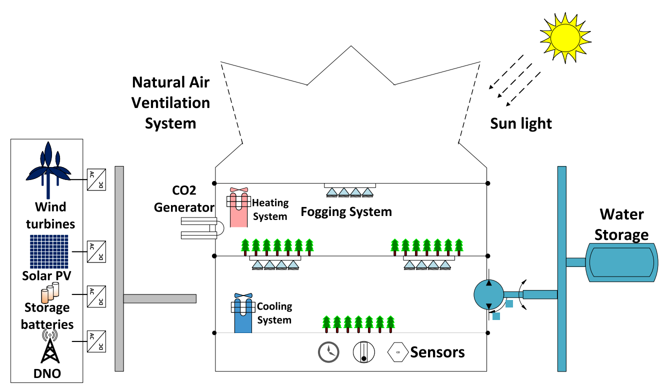


Fig. 2. Typical model of the greenhouse.

meter. The smart meter directly gets the cost of energy for each house from the grid station. Finally, based on optimal parameters selected by the optimization function, the fuzzy logic-based control module operates the actuators to set the indoor environment accordingly, as shown in Fig. 1.

The remainder of this article is formatted as follows. Section II presents the brief literature on IoT-based greenhouses and energy optimization techniques. Section III describes the optimal configuration mechanism with predictive optimization. Section IV shows the results of the prediction of climate parameters of LSTM and deep neural network (DNN); moreover, the section presents a brief comparison of energy consumption with and without optimization. Finally, Section V conclude the study with some future directions.

II. Related Work

To control the climate parameters of greenhouse, multiple actuators have been used. Fig. 2 shows the basic structure of the greenhouse system. The environmental control of a greenhouse is a multivariable problem; there is a need for optimal coordination between heating, cooling, ventilation, fogging and CO_2 supply. Automated control systems (ACSs) are presented in greenhouses to control greenhouse climate remotely [10]. Some ACSs are developed to control the climate variable, such as temperature, humidity, and CO_2 concentration. Data processing systems and sensors are connected through communication protocols, including ModBus and RS-232 [11], [12]. These systems are used to control the greenhouse environment, but there is a lack of optimization techniques to minimize energy consumption and prefer the user-desired parameters. Energy management systems (EMSs) are deployed in residential and commercial sectors to minimize energy use [13], [14].

Tuning different actuators, including AI lights (Aqua Illumination LEDs), setting up the thermostat and activating security systems are different tasks executed in home automation systems (HASs). The tasks are executed to fulfil the requirements of the household owner. These systems are controlled by smartphones or Web applications to set the climate parameters automatically based on seasons and outdoor temperature. ZigBee, X10, Z-Wave, and Wi-Fi are protocols of HAS with the capability of a central controller that are commonly used for two-way communication to turn on and off actuators [15].

The core objective is to maintain the indoor climate (i.e., inside humidity, temperature and CO_2) within a predefined range. The indoor parameters can be affected by outdoor weather conditions and other operation time of the actuators. The indoor and outdoor climate parameters differ at a time (t) and time ($t+1$). Actuators take time to change the environment of the greenhouse from the current parameter set to the desired range. In general, there is a lack of prediction of the next environmental parameters to tune the actuators as per the next climate situation. The thermal energy model by considering outdoor temperature, which affects the indoor temperature, is given in [16]

$$\varphi_{io}(h) = \frac{(T_{\text{in}} - T_{\text{out}}(h))}{R_{io}} \tag{1}$$

where T_{in} and T_{out} are the indoor temperature at hour (h) and R_{io} is thermal resistance. The thermal comfort level is given by the objective function

$$\text{Max} : \text{TCL} = \sum_{h=1}^{T} \text{CL}_{\text{th}}(h) \tag{2}$$

where the thermal comfort level is given by

$$\text{CL}_{\text{th}}(h) = \begin{cases} \beta_c.\ \exp(T_{\text{in}}(h) - T_{\text{set}} + \Delta T_{ther}) \\ \quad T_{\text{in}}(h) - T_{\text{set}} < \Delta T_{\text{ther}} \\ \quad |T_{\text{in}}(h) - T_{\text{set}}|\Delta T_{\text{ther}} \\ \beta_h.\ \exp(T_{\text{set}}(h) - \Delta T_{\text{ther}} + T_{\text{in}})) \\ \quad T_{\text{in}}(h) - T_{\text{set}} < \Delta T_{\text{ther}} \end{cases} \tag{3}$$

where T_{set} is the user-defined preference set and T_{ther} is the threshold of the temperature difference, and β_c and β_h are the leading coefficients used to adjust the penalty values. This study considered user preferences, but the outdoor parameters also affect indoor comfort levels. Moreover, the operational time of other actuators also affects the indoor temperature; for instance, the fogging system can increase the humidity but also change the indoor temperature.

Smart greenhouses are also controlled remotely by wireless sensors networks (WSNs). The centralized control with the aim of optimal energy consumption for greenhouse in a microgrid is presented [17]. The objective function was based on indoor climate parameters, storage system and water reservoir level of the greenhouse. Along with WSNs, some other control strategies, including fuzzy control [18], robust controller [19], and feedback controller [20] were presented. The Bayesian network was used for good approximation based on environmental conditions and set points of control signals [21]. The main focus was to control the air temperature within the greenhouse. At the same time, the inside air temperature is highly dependent on the outdoor temperature when air ventilation and forced ventilation are enabled.

IoT devices are recently developed for sensing and actuating services in different industries, including energy management, healthcare, and manufacturing. The devices are managed and connected by various protocols, including OCF, oneM2M (machine-to-machine) and open mobile alliance (OMA) [22], [23], [24]. The OMA is LightWeight M2M (LWM2M) and an international standard for mobile communication, which provides device management [25]. The management commands are executed on OMA nodes, and OMA DA manages the devices by using request and response transactions. The hybrid version of OMA with the European Telecommunications Standards Institute (ETSI) supports fault management and performance monitoring along with DM [26], [27]. Service-oriented architecture (SOA) IoT systems use service management; for that purpose, the Arrowhead framework was proposed, which supports interconnected operations, and efficient development [28]. The services include information infrastructure service, system management, and information assurance services. These services provide various functions, including service registration and discovery, authentication and security, and orchestration in local cloud [29], [30]. The sensors and actuators are connected to IoT devices and pass the data through these different standards. Resource registration and accessibility are also done by using these communication standards.

OCF is another versatile standard (used in the proposed study) which supports various software platforms, including Tizen, Android, Linux, Windows, and iOS [31]. OCF can also work with any transport protocol, such as Bluetooth, Ethernet, Wi-Fi, bluetooth for low energy (BLE), and ZigBee. The default protocol of OCF is CoAP, which provides communication services in OCF-based networks using REST APIs. The energy consumption for the greenhouse is also minimized by using the Kalman filter-based ANN prediction model [32]. The authors used the Kalman filter and ANN to get the optimal value of K for prediction. The results are evaluated in terms of root means square error (RMSE), and the results show that the proposed model has better results than others.

Overall, there is a lack of consideration of outdoor parameters and the indirect effect of actuators with other parameters; for instance, ventilation fans decrease CO_2 level but can also change inside humidity and temperature.

TABLE I
HARDWARE DEVICES, OPERATING SYSTEM, AND LIBRARIES FOR THE IMPLEMENTATION OF SELF-OPTIMAL CONTROL

Entity	Platform	Framework and Libraries
IoT Device	Raspberry pi 4 Model B, Ubuntu server 20.04.2 LTS,	TensorFlow-Lite 1.10.0, IoTivity 2.2.2, Flask 1.1.2, Python3
OCF Platform	PC, Windows 10 64 bit	Putty (For remote terminal), WinSCP(for FTP), IoTivity 2.2.2, MySQL, Eclipse IDE 2021-03, PyCharm for remote interpreter, Xampp Server (Database services)
Android Phone	Android 6.0 Marshmallow	Volley 1.1.0 (HTTP Request)

The energy consumption is highly dependent on outdoor parameters; by considering outdoor parameters, air ventilation can be preferred, which comparatively consumes low energy. Moreover, the actuators take some time to change the climate parameters within the desired range. To achieve this goal, actuators can be tuned on the value of climate parameters at the time $(t+1)$ by predicting the next temperature, humidity and CO_2.

III. OPTIMAL CONFIGURATION MECHANISM WITH PREDICTIVE OPTIMIZATION

The IoT network consists of multiple devices which provide services by transferring the data from the environment to the intelligent platform. IoT devices can also be an AI-based platform to process and decide something based on the situation. These devices use different sensors to recognize the environmental parameters. IoT devices are not fast in terms of processing power, so in most cases, these devices send data to the central platform, or the trained models are deployed on the devices for inference. In the proposed model, a trained model is deployed on the IoT device, which intelligently infers climate parameters and sends the data to the OCF platform for optimization.

The experiments are conducted on a greenhouse with a size of 170 × 30 × 4 m. Raspberry pi 4 is used as an IoT device, and Ubuntu server 20.0.4 LTS is installed on the IoT device. Windows PC with 24 GB of RAM is used as an OCF platform. OCF IoTivity 2.2.2 is used as a communication standard between the OCF Platform and IoT Device. Furthermore, an Android device with Marshmallow OS is used to provide an interface to the owner of the greenhouse. Putty is installed on the OCF platform to access the device terminal using SSH, while WinSCP is used to transfer files using file transfer protocol (FTP). The greenhouse is divided into three equal compartments, where each zone is equipped with standard sensors and actuators. The continuous roof ventilation (area = 0.5 m^2, equipped with anti-thrips netting), a fogging system with maximum capacity of 300g/m^2/h were available. The rated power of the actuators in (kW) placed inside each zone is given as: (heating: 150, chiller: 170, CO_2 generator: 200, dehumidifier: 50, and forced ventilation: 15). The hardware devices, operating system, and libraries for implementation of self optimal control are listed in Table I.

In the first step of processing, data is collected from sensors and prediction models are trained on the collected data to infer climate parameters. The trained model predicts the next (at time $t+1$) value of temperature, humidity, and CO_2. Next, the collected data is sent to the platform using a request-response between the client and server. Finally, the OCF platform returns the optimized climate parameters to the IoT device using the OCF standard. The optimization of parameters is done by achieving minimum energy consumption within user preferences. The conceptual model of the proposed mechanism is shown in Fig. 3.

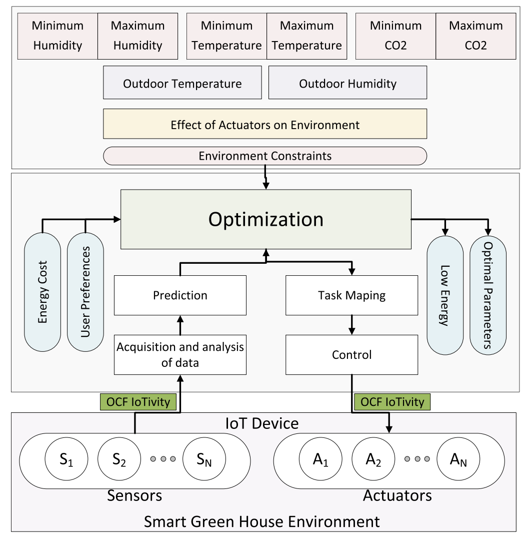


Fig. 3. Conceptual model of the greenhouse.

A. Prediction Mechanism for Environmental Sensors

The inference module based on LSTM is deployed on IoT Device to get the next temperature, humidity, and CO_2 value. LSTM and DNN models are first trained on server and the inference performance is analyzed. By considering certain parameters (Accuracy and loss) the best model (LSTM) is offloaded to IoT devices. The general disadvantage of LSTM is that it is computationally intensive but the proposed model analyzes the greenhouse environment on hourly basis and update the setting of actuators based on values computed by optimization module. Due to the delay-tolerant environment of greenhouse, the proposed model bears the lag (LSTM inference time, i.e., few seconds). Therefore, the accuracy of the model is prioritized over the inference time because the plants in the greenhouse are sensitive to the environment. LSTM is converted to a tf-lite model and deployed to the IoT device. The data used in this study is time series, where

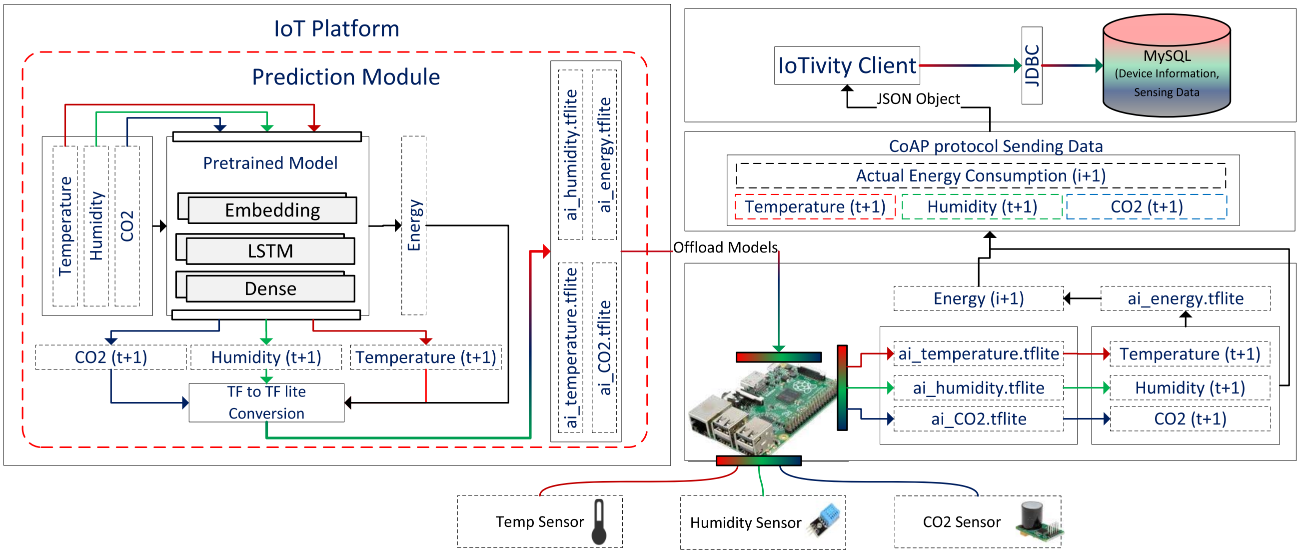


Fig. 4. LSTM model for the prediction of environmental parameters.

the previous values highly impact the next values. In addition, the LSTM model has a short memory, which helps the model to predict the next environmental parameters more accurately. Moreover, when the data contain long-, medium-, or short-term trends, the LSTM model adds more parameters to be learned, which makes it a powerful forecasting model [33]. Four models for temperature, humidity, CO_2 and for energy prediction are trained and converted to tf-lite. These tf-lite models are deployed on raspberry-pi (IoT device). Three sensors for considered climate parameters are connected to the device to get the current value of temperature, humidity, and CO_2 as shown in Fig. 4. The ML models predict parameters for time $(t+1)$ by getting the value at time (t). The energy consumption is also predicted by using future climate parameters. Lastly, IoT device send the data to the IoTivity client using the OCF IoTivity standard. Client stores the data in the database using JDBC for the optimization module.

The inference models send the data to the OCF platform when the request from the client is received. The client initiates the process by requesting climate parameters. IoTivity server on the IoT device is always ready to respond with climate parameters when the request is received. At the same time, the flask server is running on an IoT device to provide prediction services using the trained tf-lite model. The detailed flow of data will be explained in Section IV-B. The client first discovers the server by initiating a request using CoAP. The request is received by services running on IoT device. The request contains the number of climate parameters and resource information. The type of request can be a GET or a POST. For real-time climate parameters, the IoT device senses data from sensors and sends it to the flask library running on the same device. The trained inference models predict the next climate parameters and the energy consumption on these predicted parameters. All four parameters are returned to the server in the form of the JSON object. Finally, the server returns climate parameters to the client and the client store data in the database using JDBC. The IoTivity module receives the data sent from the flask server on IoT Device. The data is in JSON format, which is easy to unpack on the client side.

Indoor climate parameters are sensed from sensors, and the outdoor parameters are obtained from KMA weather API. The API returns outdoor temperature and humidity values. Outdoor parameters are stored in the central database for the optimization module. The indoor and outdoor climate parameters (i.e., temperature, humidity, and CO_2) are used in the optimization function to minimize energy consumption. The energy returned from the tf lite model is used for the comparison with energy after optimization. Indoor and outdoor parameters are also visualized to the owner of greenhouse on their smartphone. Greenhouse owner can set the preferred climate parameters from smartphone and can also can check current environmental indoor situation of greenhouse. Smartphone initiates HTTP request using the volley library to connect with the central database. The preferred parameters are stored in the database and later used as upper and lower bounds for the optimization module. The flow of requests and overall architecture of connections and control is shown in Fig. 5.

Nowadays, most electricity companies are still using flat rate pricing (FPR) models for electricity pricing mechanisms. In FPR fixed price is charged for each time because of traditional electromechanical meters. Some companies are replacing FPR with real time pricing (RTP). In RTP the price of electricity changes for different time as per the requirement of end user. During high energy demand, the price of energy is increased while the price is decreased in other cases. In the proposed model smart meter is used to get the real-time energy price, and the parameters are selected based on the energy price at that specific time (t).

B. Mathematical Objective Function

Fig. 6 shows the flow of optimization of climate parameters. Initially, required parameters, including indoor parameters at time t and $(t + 1)$, outdoor parameters (temperature and

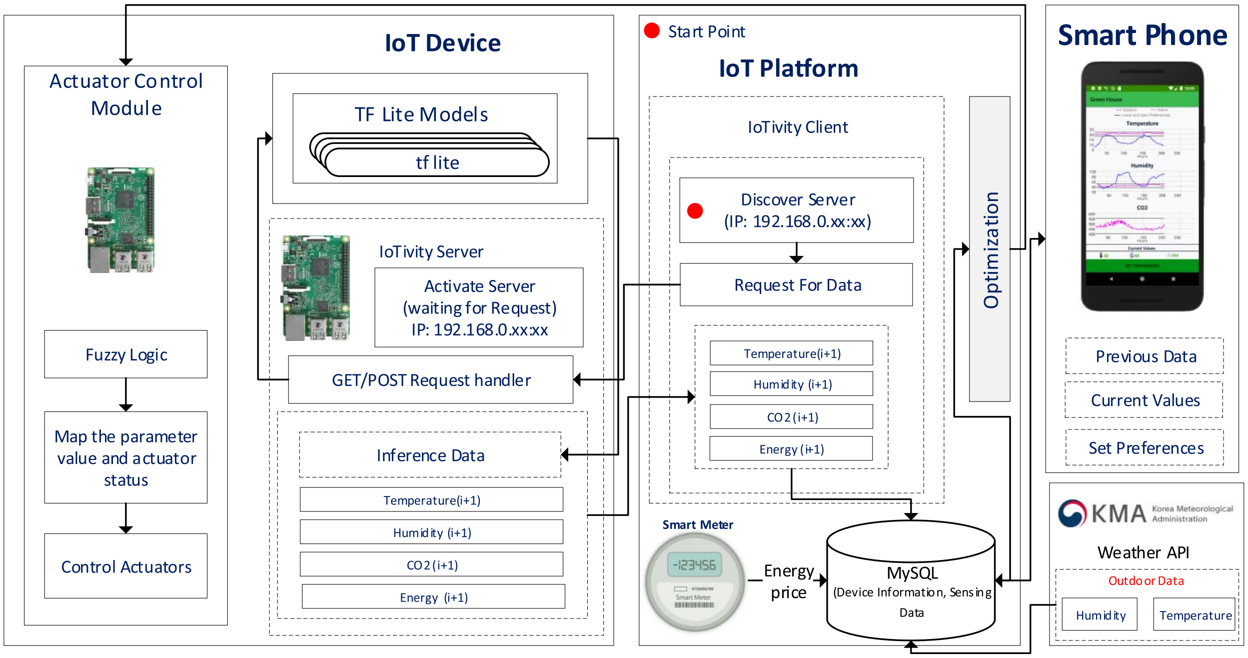


Fig. 5. IoTivity control and flow of requests between the IoT device and platform, and the smart phone and platform.

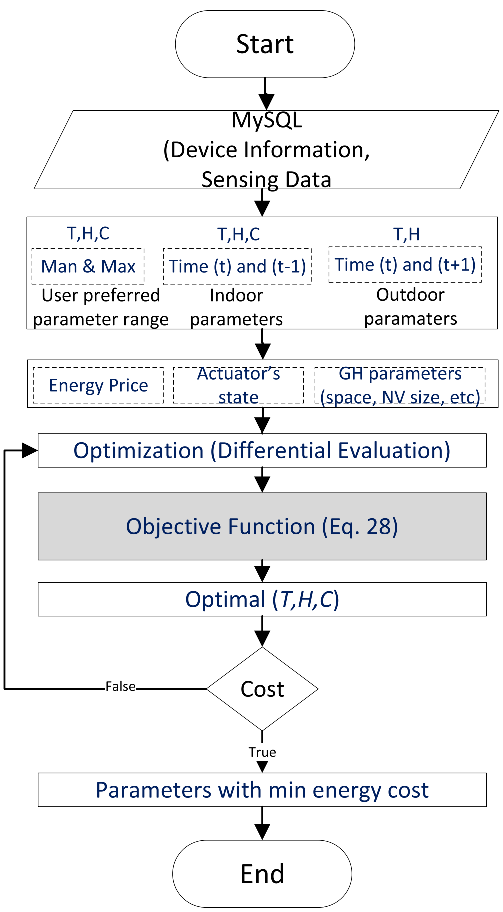


Fig. 6. Flow of optimization of the climate parameters using indoor, outdoor, user preferences, and current energy price.

humidity) at $(t+1)$, user preferences range, and greenhouse attributes, including the state of actuates, size of zones, and natural ventilation area is extracted from the central database. The parameters are used in the cost function (28) given to the differential evaluation (DE) algorithm to minimize energy consumption. Based on the cost function, DE minimizes energy consumption and selects optimal parameters. The parameters are selected on the basis of energy cost. The real-time energy cost on a specific time is given by the smart meter. The energy price is stored in the database and retrieved along with other parameters while starting the optimization process. The energy consumption also depends on the outdoor parameters, indoor relative temperature, humidity, and CO_2 and the state of actuators. When the optimal global value is achieved by the algorithms, then these climate parameters are passed to the control module.

Indoor temperature can also be affected by the outdoor temperature and the state of the indoor actuators

$$R_T(t) = R_T(t-1) + \frac{\tau}{V_z}\Big[\gamma_z^{htp}(t) - \gamma_z^{chp}(t) - \gamma_z^{wl}(t) - \gamma_z^{nv}(t) - \gamma_z^{fv}(t) + \gamma_z^{cg}(t) + \gamma_z^{dh}(t)\Big]. \quad (4)$$

The equation shows that the indoor temperature at time t can be affected by the temperature at time $(t-1)$ and the volume of zone z with the following parameters: 1) the heat transferred from heating and chiller pipes; 2) the head loss through Natural and forced ventilation and walls; and 3) the heat produced by CO_2 generator and dehumidifier.

The temperature for each zone is depended on the multiple parameters as shown in the following equations [34].

Parameters for Heating:

$$R_{T_{htp},z}(t) = R_{T_{htp},z}(t-1) + \frac{\tau}{V_z}\big[3.6 \times P_{ht}^{\max} S_{ht,z}(t) - R_{\text{pipe}} A_{htp,z} S_{htp,z}(t) \times \big(T_{htp}(t) - T_z(t)\big)\big] \quad \forall_t \in Tm. \quad (5)$$

Parameters for Cooling:

$$R_{T_{chp},z}(t) = R_{T_{chp},z}(t-1) + \frac{\tau}{V_z}\Big[3.6 \times P_{ch}^{\max} S_{ch,z}(t) - R_{\text{pipe}} A_{chp,z} S_{chp,z}(t) \times \Big(R_{T_{chp},z}(t) - T_z(t)\Big)\Big] \quad \forall_t \in Tm. \tag{6}$$

The above equations shows that the inside temperature at time t can be given as: 1) the temperature of chiller (heater) pipes at time $(t-1)$; 2) the cooling (heating) absorbed from the operation of chilling (heating) system; and 3) cooling (heating) passed from the cooling (heating) pipes to the greenhouse.

The inside temperature must be within the given range of user preferences

$$D_T = [P_{T\min}, P_{T\max}]. \tag{7}$$

So, the relative temperature at time t should be within the user preferences

$$R_T(t) \in D_T. \tag{8}$$

Relative humidity of greenhouse is given by [34]

$$R_H = \frac{p_{\text{par}}}{p_{\text{wat}}} \tag{9}$$

whereas

$$p_{\text{wat}} = \theta_1\left(-\theta_2 + \theta_3 e^{\theta_4 \mu}\right) \tag{10}$$

and

$$p_{\text{par}} = \frac{w P_{\text{atm}}}{\theta_5}. \tag{11}$$

All the scenario makes the problem as mixed-integer linear programming (MILP) problem. The humidity inside the greenhouse is based on the mass balance principle

$$R_H(t) = R_H(t-1) + \frac{\tau}{\rho(a) V_z}\Big[A_z + Q_z \rho_a A_z S_{(v,z)}(t)(H_{\text{out}} - R_H(t)) + S_{fg,z} H_{fg}^{\max} A_z - S_{dh,z} H_{fg}^{\max} A_z\Big] \quad \forall_t \in Tm. \tag{12}$$

The equation shows that the indoor humidity can be affected by multiple parameters, including the area of smart greenhouse, airflow rate of ventilation fans, outdoor humidity, and the state of both actuators mapped with humidity in a particular zone.

The desired humidity has a minimum and maximum range and the greenhouse user can set the range by considering the play type

$$D_H = \left[P_{H_{\min}}, P_{H_{\max}}\right]. \tag{13}$$

The relative humidity should be within the range of desired humidity

$$R_H(t) \in D_H. \tag{14}$$

Almost all types of plants need a certain level of CO_2 and the sunlight for photosynthesis. The plants consume CO_2 in day time so the CO_2 concentration will be low. So, there is a need to keep high level of photosynthesis by controlling CO_2 optimally. The level of CO_2 depends on the type of plants in the greenhouse. The user-preferred range for CO_2 can be given as

$$D_{CO_2} = \left[P_{CO_2\min}, P_{CO_2\max}\right]. \tag{15}$$

The desired CO_2 is between the minimum and maximum levels of CO_2. The concentration of CO_2 within the

TABLE II
SYMBOLS, ACRONYMS, AND THEIR DESCRIPTION

Symbol	Description
A	Area of Greenhouse
Z	Set of zones of greenhouse (3)
T	Set of time interval
J	Objective Function of Model
ch	Chiller (For Cooling Purpose)
ch_p	Chiller pipes
ht	Heater (For heating purpose)
ht_p	Heating pipe
a	Air
cg	CO_2 generator
dh	Dehumidifier
hd	Humidifier
fv	Forced Ventilation
fg	Fogging System
gh	Greenhouse
nv	Natural Ventilation
out	Outdoor
in	Indoor
P	Plants
GC	greenhouse Canopy
p_{wat}	Saturated water vapor pressure (Pa)
p_{par}	Partial vapor pressure (Pa)
P_{atm}	Atmospheric air pressure (Pa)
P_i	Power rate of actuator i
τ	Length of time interval $(1hour)$
ρ_a	Density of air $(kg/m3)$; 1.27
Q_z	Volumetric airflow rate of ventilation fans $m_a^3/(hm_{gh}^2)$ 17.4
A_z	Area of the zone z
A_v	Area of ventilation pipe in zone z (m^2)
$S_{(i,z)}$	State of device i in zone z
R_H	Relative Humidity
t	Index for a time interval (h)
wl	Greenhouse wall
$C_{(gen,z)}^{max}$	Maximum CO_2 generated by CO_2 generator in zone z
H	Humidity (ppm)
CO_2	CO_2 in greenhouse (ppm)
Tem	Temperature $°C$
V_z	Volume of zone z 6800 (m^3) each zone
γ_i	The effect of actuator i on temperature kJ/h
R_{pipe}	Heat transfer coefficient from pipes $(kJ/(hKm^2))$
φ	The quantity of wind in percentage enter in greenhouse (no dim) 0.0085
θ_6	Conversion factor from hour to seconds (3600)
ϑ_z	The effect of fans on CO_2 Concentration in zone z
$C_{(pht,z)}$	The coefficient used for photosynthesis of plants in zone z g/J 40.04e-3
$I(t)$	Solar irradiation at time t (W/m^2)

greenhouse can be modeled as

$$\begin{aligned} R_{\mathrm{CO}_{2z}}(t) = R_{\mathrm{CO}_{2z}}(t-1) \\ + \frac{\tau}{\rho_a V_z}\Big[\mathrm{CO}_{2cg,z}^{\max} S_{\mathrm{CO}_2,z}(t)\ A_{gh,z} \\ + \theta_6 \varphi A_{nv,z} S_{nv_z}(t)\ \big(\mathrm{CO}_{2\mathrm{out}}(t) - R_{CO_2z}(t)\big) \\ + \vartheta_z S_{fv,z}(t)\big(\mathrm{CO}_{2\mathrm{out}}(t) - R_{CO_2z}(t)\big) \\ - \theta_6 C_{(pht,z)} I(t) A_{gh,z}\Big]. \end{aligned} \tag{16}$$

The relative CO_2 within the greenhouse at time t is dependent on the CO_2 at time $(t-1)$, CO_2 generated by cg, nv, fv and $C_{(pht,z)}$ and state of all the actuators.

The owner sets the desired temperature D_T, desired humidity D_H, and desired D_{CO_2} by considering the plant type in the greenhouse.

The optimal climate parameters should be in the desired range set by the greenhouse owner

$$O_T \in D_T, O_H \in D_H \text{ and } O_{\mathrm{CO}_2} \in D_{\mathrm{CO}_2}.$$

The optimal state of the greenhouse will be $[O_T, O_H, O_{\mathrm{CO}_2}]$ and the current state of the greenhouse at time (t) will be $[T_c, H_c, \mathrm{CO}_{2_c}]$. The greenhouse will be in an optimal state when the current parameters are within the desired range

$$\Delta T = \begin{cases} |T_c - P_{T_{\min}}|, & \text{if } T_c < P_{T_{\min}} \\ |T_c - P_{T_{\max}}|, & \text{if } T_c > P_{T_{\min}} \end{cases} \tag{17}$$

$$\Delta H = \begin{cases} |H_c - P_{H_{\min}}|, & \text{if } H_c < P_{H_{\min}} \\ |H_c - P_{H_{\max}}|, & \text{if } H_c > P_{H_{\min}} \end{cases} \tag{18}$$

$$\Delta \mathrm{CO}_2 = \begin{cases} |\mathrm{CO}_{2_c} - P_{\mathrm{CO}_{2_{\min}}}|, & \text{if } \mathrm{CO}_{2_c} < P_{\mathrm{CO}_{2_{\min}}} \\ |\mathrm{CO}_{2_c} - P_{\mathrm{CO}_{2_{\max}}}|, & \text{if } \mathrm{CO}_{2_c} > P_{\mathrm{CO}_{2_{\min}}}. \end{cases} \tag{19}$$

The temperature ΔT, ΔH, and $\Delta \mathrm{CO}_{2_c}$ show the difference between the current and desired upper and lower ranges of temperature, humidity, and CO_2, respectively.

The higher difference shows higher energy consumption because the difference can be treated as a Euclidean distance between the current and desired parameter range.

If the unit energy consumption to increase temperature, humidity, and CO_2 by one degree is ET_o, EH_o, and $ECO2_o$ (by excluding the external factors), respectively, then the minimum energy required to set the parameters within the user preferences will be

$$E_T = ET_o \times \Delta T \tag{20}$$

$$E_H = EH_o \times \Delta H \tag{21}$$

$$E_{\mathrm{CO}_2} = \mathrm{ECO}_{2_o} \times \Delta \mathrm{CO}_2. \tag{22}$$

The indoor environmental parameters can be affected by outdoor climate parameters. By involving outdoor parameters and assigning weights to each parameter, the above equations will become

$$\omega_T = \begin{cases} \frac{1}{|T_c - T_{\mathrm{out}}|}, & \text{if } T_c < T_{\mathrm{out}} < P_{T_{\min}} \\ 1 - \frac{1}{|T_c - T_{\mathrm{out}}|}, & \text{if } T_{\mathrm{out}} < T_c < P_{T_{\min}} \\ \frac{1}{|T_c - T_{\mathrm{out}}|}, & \text{if } T_c > T_{\mathrm{out}} > P_{T_{\min}} \\ 1 - \frac{1}{|T_c - T_{\mathrm{out}}|}, & \text{if } T_{\mathrm{out}} > T_c > P_{T_{\min}} \\ 1, & T_{\mathrm{out}} = T_c > P_{T_{\max}} \text{ or } T_c = T_{\mathrm{out}} < P_{T_{\min}}. \end{cases} \tag{23}$$

The estimated energy consumption for Temperature, Humidity, and CO_2 can be represented by the following equation:

$$E_T = (ET_o \times \Delta T) \times \omega_T \tag{24}$$

$$E_H = (EH_o \times \Delta T) \times \omega_H \tag{25}$$

$$E_{\mathrm{CO}_2} = \big(\mathrm{ECO}_{2_o} \times \Delta T\big) \tag{26}$$

and the total energy consumption will be

$$\text{Total Energy}(E_{\text{total}}) = E_T + E_H + E_{\mathrm{CO}_2}. \tag{27}$$

The objective is to minimize the energy consumed to set the climate parameters within the desired range by considering outdoor temperature and humidity. Therefore, the objective function is convex and has one optimum global point.

A common practice in optimization problems is to include the penalty function, which allows the optimization algorithm to move to the global optimum. DE is used as an optimization function which accepts the cost function to minimize it. DE is a real-valued multidimensional, population-based function that designs a search space by maintaining a population candidate solution [35]. The constraints limit the solution within a specific bound.

By combining (24)–(26), the objective function becomes

$$J(x) = \min\left(\sum_{X \in [T,H,\mathrm{CO}_2]} \sum_{i=1}^{n} \sum_{z=1}^{m} \big(E_{X(n,z)} \omega_X\big)\right) \tag{28}$$

subjected to

$$P_{T\min} < T < P_{T\max}$$
$$P_{H\min} < H < P_{H\max}$$
$$P_{\mathrm{CO}_{2_{\min}}} < T < P_{\mathrm{CO}_{2_{\max}}}.$$

We need to minimize the energy consumed for each actuator in each zone.

Operational constraints are also included to ensure that two different types of actuators will not operate simultaneously. The control of forced ventilation air ventilation depends on outdoor parameter values. Suppose outdoor temperature or humidity helps to get the desired optimal values, then air ventilation and forced ventilation will be operated. Also, the dehumidification and fogging systems should not be operated simultaneously, and heating and cooling will not be injected into the greenhouse environment at the same time.

Algorithm 1 shows the self-optimal control mechanism for the greenhouse. Initially, the user-preferred parameter range is extracted from the database. The owner of the greenhouse sets the range by using an android-based greenhouse application. Next, the parameters of DE are initialized, and the population size is set. The self-optimal mechanism is activated each hour to optimize the parameters. The current energy price is obtained from the smart meter, and current indoor humidity, temperature and CO_2 are taken from the IoT device. By initializing the required parameters, the DE algorithm selects the optimal combination of parameters and returns optimal temperature, humidity, and CO_2.

Algorithm 1: DE-Based Optimal Parameter Selection

```
Data: DE Control Parameters: Population size N,
      Mutation factor F and Crossover Rate CR
Result: Optimal climate parameters (O_T, O_H, O_CO2);
initialization;
D_T ← [P_Tmin, P_Tmax] // Preferred Range for
    Temperature
D_H ← [P_Hmin, P_Hmax] // Preferred Range for
    Humidity
D_CO2 ← [P_CO2min, P_CO2max] // Preferred Range
    for CO2
Population of N individuals; // Uniformly Distributed
X_G = X_{1,G}, X_{2,G}, ..., X_{N,G} where
    X_{i,G} = [x_{i,G}(1), x_{i,G}(2), ..., x_{i,G}(N)] represents the
    climate parameters (O_T, O_H, O_CO2) of classifiers.
    // generation iteration
G ← 0; // Self Optimality for each hour
foreach hour do
    // Get current energy price
    E_o ← SamrtMeter(); T_c, H_c, CO2_c ← OCFRequest()
    T_out, H_out ← KMA_API()
    while stopping criteria not satisfied do
        for i = 0; i < N; i++ do
            Select distinct indexers r1, r2, r3 // should be
                different from i
            V_{i,G} = X_{r1,G} + F × (X_{r2,G} − X_{r3,G})
                // Compute mutant vector
            j_rand ← random() ;
             // Generate Random value
            for j = 0; j < D; j++ do
                T_{i,G} ← { v_{i,G}(j), if (i,j)[0,1] < CR
                          { x_{i,G}(j),     otherwise
            X_{i,G+1} ← Fitness of T_{i,G} and T_{X,G} using eq. (28)
        G = G+1 // Increase termination
            parameter
```

C. Actuator Control Based on Fuzzy Logic

The optimized values are passed from the OCF server to the IoT device to control actuators. The actuators are first scheduled and selected based on the combination of optimal climate parameters. Fig. 7 shows the fuzzy logic-based control structure to control the actuators. A chiller and heater are used to decrease and increase the temperature, respectively. Similarly, a dehumidifier, fogging system, CO_2 generator, and forced ventilation are used to control the humidity and CO_2 of the greenhouse. The heater will be operational if $Tc < Td$, and the air ventilation system will be turned on if the outdoor temperature will help to increase the temperature. The air and forced ventilation systems will be turned on if the temperature and humidity both agree to inject outdoor climate into the greenhouse; in other words, if outdoor temperature and humidity help to achieve desired values, then both actuators will be operational. Similarly, the chiller will be turned on if $Tc > Td$ and $Tc >> Td$, and air ventilation and forced ventilation will be operated to inject outdoor temperature within the greenhouse based on the outdoor environmental situation. In case of humidity, the fogging system is used to increase, and a dehumidifier is used to decrease indoor humidity. Outdoor humidity is also considered to use fewer power actuators, i.e., air ventilation and forced ventilation. For CO_2 outdoor environment is not considered because the carbon level in natural air is 0.04%. Therefore, the outdoor CO_2 will always help to decrease the indoor CO_2 level. The forced ventilation system is used to decrease the level of CO_2, and a CO_2 generator is used to inject carbon within the greenhouse. Plants also inject carbon within the environment, so usually, the CO_2 level within the greenhouse is high.

IV. Implementation Results

A. Implementation of Environment Evaluation Measures

To select the best model from LSTM and DNN for prediction of next climate parameters based on the current values, RMSE, mean absolute error (MAE), and R2 are selected as evaluation metrics. In addition, for the optimization module, energy consumption is compared with and without optimization for each parameter (i.e., temperature, humidity, and CO_2).

The RMSE is a square of the difference between actual and predicted values. In contrast, MAE is the absolute value of the difference between actual and predicted values. R2 coefficient shows the accuracy of prediction instead of an error. The higher value of R2 shows a better performance of the model. All three measures are used to evaluate the performance of models, and one best model is deployed on IoT devices to infer the next parameters

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{n}(y_i - x_i)^2}{n}} \tag{29}$$

$$\mathrm{MAE} = \frac{\sum_{i=1}^{n}|y_i - x_i|}{n} \tag{30}$$

$$R2 = 1 - \frac{\text{Unexplained Variation}}{\text{Total Variation}}. \tag{31}$$

B. OCF Standard Connectivity

The connection between IoT devices and the IoT platform is established using the OCF standard. The research project named Campus as a Mashups Platform for IoT Experiments (CAMPIE) proposed a framework to deploy an IoT in a large area. The CAMPIE service discovery (CSD) based on OCF IoTivity is a versatile service discovery architecture. Fig. 8 shows the demonstration of connectivity between IoT device and platform using the OCF standard. The client initiates a request by discovering the server within the same network and request for climate parameters. When the server receives a request from a client, it requests the simulator running on the same device (IoT device) for climate parameters. The figure shows the CoAP request from coap://192.168.0.2:58284 (IP address and port) sent by the IoTivity client and received by the server. The server sends an HTTP request to the simulator, and the simulator gets the values from trained models. The simulator returns the JSON object to the server, and the server sends the data back to the client. Finally, the client sends the data to the database for the optimization module.

C. Predictive Optimization

The data set is collected from the sensors, including the temperature sensor, humidity sensor, and CO_2 sensor. The

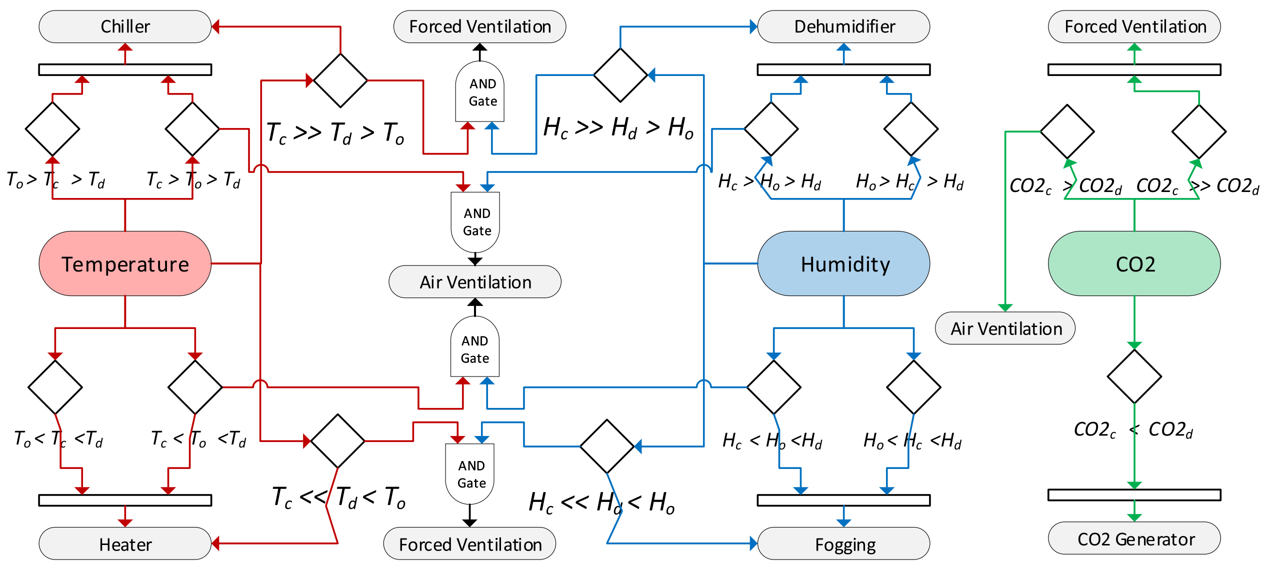


Fig. 7. Fuzzy control module to control greenhouse actuators. The edges shows the condition when the actuator is operational. The AND gate is used to verify that both (temperature and humidity) are agree to turn on the actuator.

TABLE III
PREDICTION RESULTS OF LSTM AND DNN IN TERMS OF RMSE, MAE, AND R2 SCORE

LSTM						DNN					
Training			Testing			Training			Testing		
RMSE	MAE	R2	RMSE	MAE	R2	RMSE	MAE	R2	RMSE	MAE	R2
Temperature											
0.11	0.09	0.94	0.21	0.12	0.83	0.14	0.10	0.92	0.18	0.15	0.82
Humidity											
0.37	0.27	0.99	0.5	0.36	0.98	1.97	1.56	0.89	1.76	1.47	0.77
CO_2											
21.09	14.01	0.96	24.68	15.99	0.9	27.44	18.77	0.63	33.66	23.07	0.64
Energy Prediction											
6.49	4.73	0.942	6.51	4.81	0.95	8.74	6.19	0.89	9.21	6.82	0.90

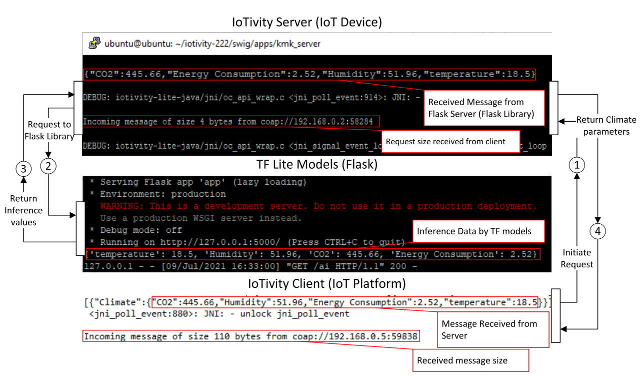
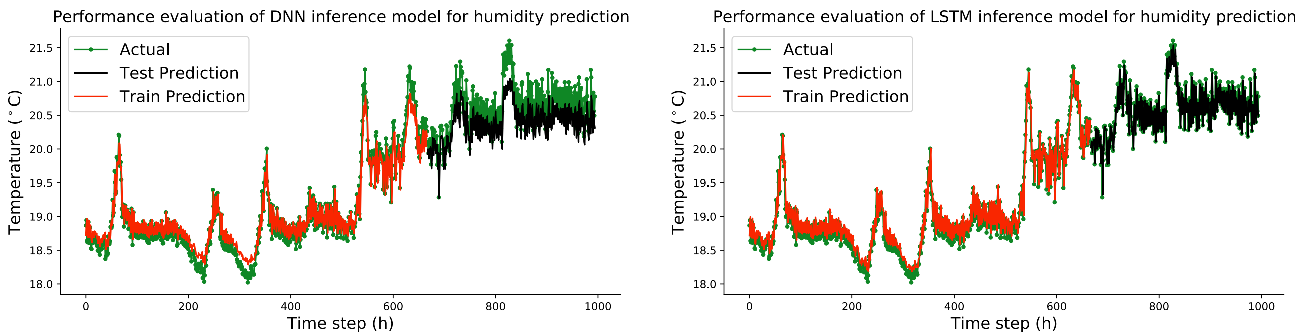


Fig. 8. Implementation and flow of data between IoT device, IoT platform, and prediction module.

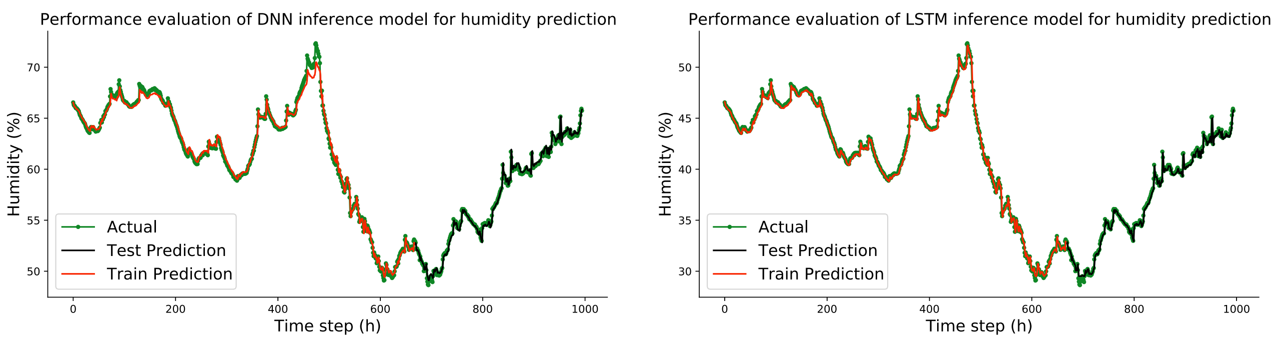
data is used to train the DNN and LSTM model to predict the next value of climate parameters and energy consumption. The energy consumption is the dependent variable and changes based on the value of temperature, humidity, and CO_2. Table III shows the results of prediction in terms of RMSE, MAE, and R2 coefficient for LSTM and DNN. The purpose of comparing the predicted value and actual measured value at time $t+1$ is to select the best model for the IoT device that gives the best climate parameters based on current parameter values. It is clear from the table that the results of LSTM are reasonable compared to DNN. The testing and training results of LSTM and DNN are reported. The R2 coefficient shows the accurate prediction of models, while RMSE and MAE show the error rate. The error rate of LSTM for both the training and testing phase is less, while the R2 score is high compared to the DNN model for prediction and inference.

The migration of ML techniques on edge provides a faster process of inference. To migrate the models to IoT devices, LSTM is converted to a tflite model and deployed on IoT devices for future predictions. Three models for inference of temperature, humidity, and CO_2 and one for energy prediction on these three inferred climate parameters are deployed on edge. For humidity, the model outperforms and shows 99% R2 score in training and 98% in testing. Hold-out techniques are used to train the model. The sensor's history data of six months is used to train the model, while the energy consumption on these attributes is used to forecast energy consumption. The source codes of Energy optimization



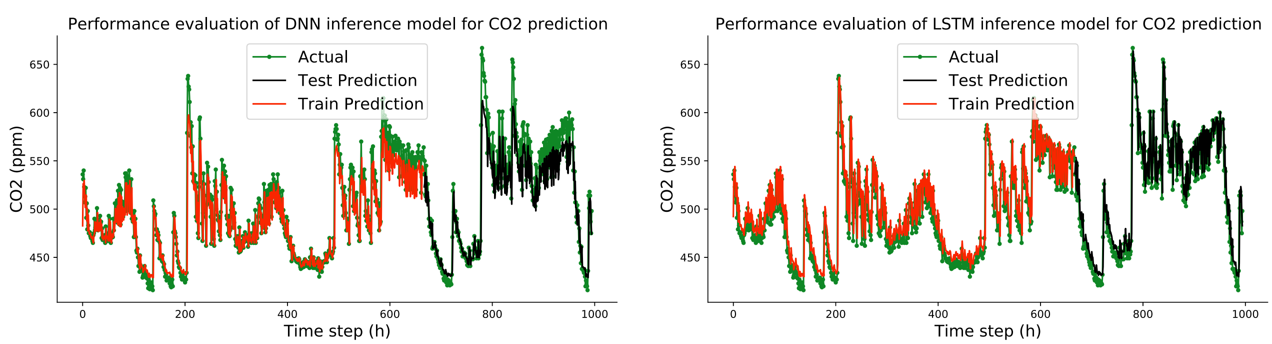
(a) (b)

Fig. 9. Inference results of temperature by using (a) DNN and (b) LSTM.



(a) (b)

Fig. 10. Inference results of humidity by using (a) DNN and (b) LSTM.



(a) (b)

Fig. 11. Inference results of CO2 by using (a) DNN and (b) LSTM.

and OCF connectivity (for client and server) are online available.[1]

Fig. 9(a) and (b) shows the inference results of temperature for each hour. The results show that the prediction of LSTM is better in comparison to DNN. Similarly for Humidity [Fig. 10(a) and (b)] and for CO_2 [Fig. 11(a) and (b)] LSTM performs well. For temperature, the value is taken from 18 to 21, which is the normal temperature for the broccoli in the greenhouse, and humidity is taken from 50 to 70, which is the normal humidity for broccoli. The user preferences are also considered, and the value of these climate parameters is adjusted by getting greenhouse experts' opinions. CO_2 is measured in terms of parts-per-million (ppm), and the carbon dioxide level is selected from 400 to 650 ppm for indoor greenhouse parameters. CO_2 is also injected by plants into the environment of greenhouse. The level of CO_2 in the greenhouse is normally high in the evening and while it is low in the daytime.

After predicting all climate parameters, the energy consumption is computed by using a trained model. The results of the trained LSTM and DNN model are shown in Fig. 12(a) and (b). The illustration shows that the performance of LSTM is good compared to DNN. Next, the LSTM model is deployed on edge to forecast the energy consumption for inferred parameters. The consumption of energy is predicted in KW/H. The predicted energy is treated as energy consumption without optimization. After predicting climate parameters and energy

[1] https://github.com/atifrizwan91/Greenhouse/

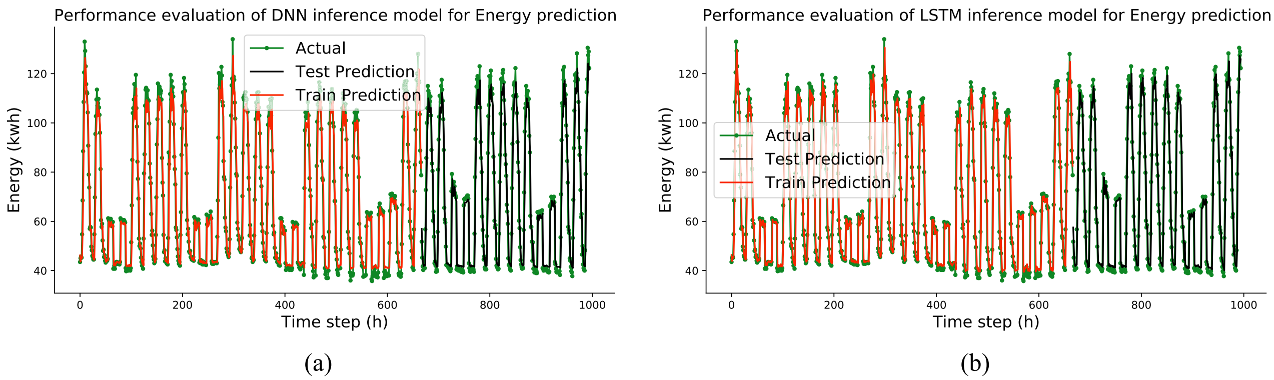


Fig. 12. Prediction results of energy by using (a) DNN and (b) LSTM.

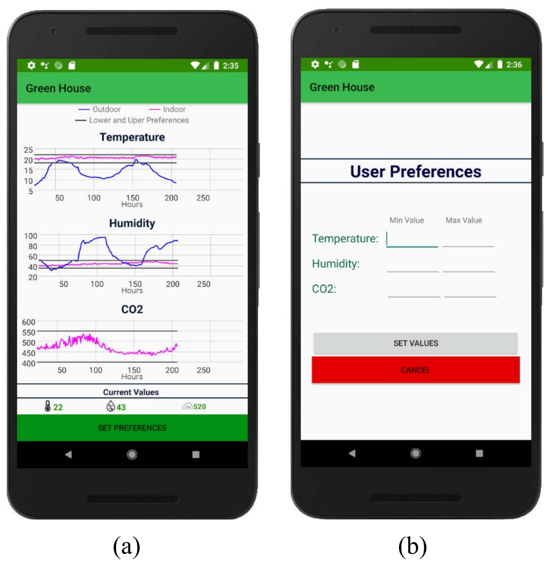


Fig. 13. Smartphone interface to control and view the current climate situation of greenhouse. (a) Greenhouse monitoring. (b) User preferences.

consumption, the values are passed to the OCF platform to optimize energy consumption.

D. Comparison of Optimization Results

The optimization function tries to select the optimal parameters between the user's preferred range. The optimization function aims to select minimum energy consumption within the specified constraints. The user preferences for temperature, humidity, and CO_2 are by using smartphone application as shown in Fig. 13. The user gives upper and lowers bounds of parameters, and the optimal value is selected between the bounds based on minimum energy consumption. Greenhouse owner can also view the comparison of indoor and outdoor temperature and humidity. Indoor parameters will always be within the user-preferred range. The current climate parameter values are also shown on the main screen of the application.

The LSTM model is used to predict energy consumption based on the predicted climate parameters. After that, the optimization function selects the optimal parameters by considering outdoor parameters. Finally, the energy consumption on the optimal parameters is compared with the predicted energy. Fig. 14 shows the comparison of energy consumption

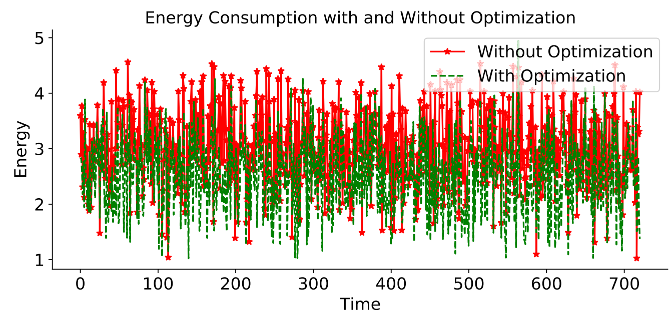


Fig. 14. Comparison of energy consumption with and without optimization.

with and without optimization. The results are based on energy consumption for each hour for one month. It is clear from the figure that energy consumption is minimized by selecting the optimal parameters with DE by minimizing the cost function. Furthermore, the energy cost is also decreased, which is directly gotten from the smart meter.

The objective function minimizes energy consumption by considering outdoor temperature and humidity and considers the state of other actuators in the greenhouse. Broccoli is grown in the greenhouse, and the user preferences are set based on the type of plants. The energy consumption in the greenhouse is optimized by using: 1) indoor parameters; 2) outdoor parameters; and 3) user-preferred range. The level of actuators is set by the fuzzy logic control module. The greenhouse owner can set the range of temperature, humidity, and CO2 based on the crop grown in the greenhouse. The crop in the greenhouse affects the indoor parameters, but by allowing the greenhouse owner to set its preferred range, we ignored the effect of the crop. The study used broccoli as a test case, but the proposed model can be implemented with other crops in a greenhouse. Fig. 15 shows the comparison of optimal temperature with indoor and outdoor temperature. For each instance, a different level of the current and outdoor temperature is selected to apply the optimization algorithm to get the optimal climate parameter based on the price of energy obtained from the smart meter. The outdoor temperature does not always help to get the optimal temperature; sometimes, the outdoor temperature resists getting the optimal indoor temperature. The current temperature is also important because the actuators need to increase/decrease the temperature from the current to an optimal value obtained from the optimization function. The

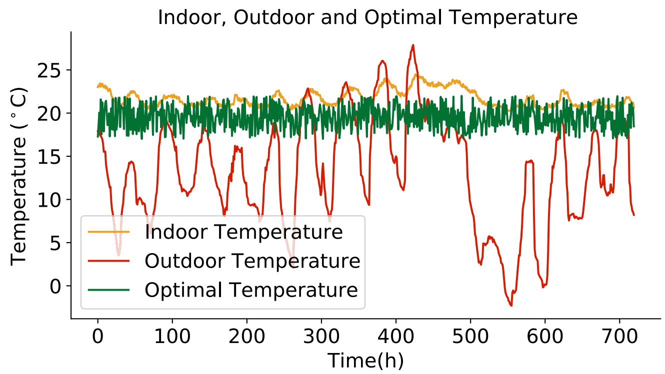


Fig. 15. Comparison of indoor, outdoor, and optimal temperature.

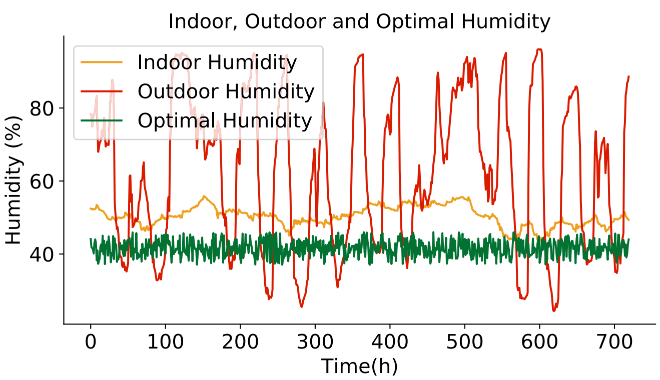


Fig. 16. Comparison of indoor, outdoor, and optimal humidity.

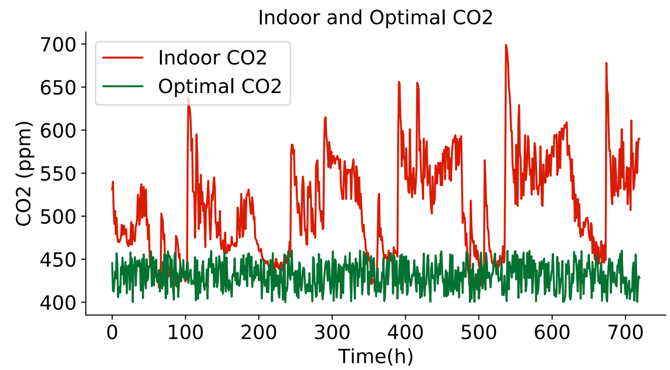


Fig. 17. Comparison of indoor and optimal CO2.

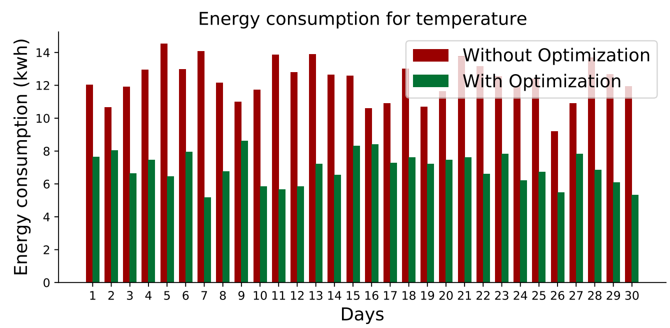


Fig. 18. Comparison of energy consumed to set the optimal temperature.

distance between the current and the closely bound of user preference is directly proportional to the energy consumption. Similarly, for humidity, to gain the optimal humidity inside the greenhouse, the current and outdoor humidity is considered as shown in Fig. 16. The fogging system is used to increase the indoor humidity, and the dehumidifier is used to decrease the level of humidity. KMA API returns the level of outdoor humidity in %age, and the humidity sensor senses the indoor humidity in the same unit. The data shows high fluctuations in outdoor parameters because of uncertainty in the outdoor environment. Different levels of indoor and outdoor humidity are used to select the optimal humidity, same as temperature. The optimal humidity is always between the user preferences with minimum energy consumption. For the experiments, the user preferences for temperature are selected from 17 °C to 22 °C, and for humidity, the bounds are 55%–70%.

CO_2 concentration is important for all types of plants. Different plants inject CO_2 into the environment at different times. The level of CO_2 can be increased in the evening, so the level should be optimized. By selecting the optimal CO_2 within user preferences, energy consumption is minimized. The outdoor CO_2 is not considered for the optimal values because the level of carbon in natural air is too low. For broccoli, the level of CO_2 is selected from 400 to 450 ppm. The comparison of indoor and optimal CO_2 is shown in Fig. 17.

The optimal temperature is achieved by considering the outdoor and current indoor temperature values. A heater and chiller are used to get the desired temperature and achieve optimal environmental temperature. The energy consumption for the heater and chiller is given in Fig. 18. The energy-saving for each day is not constant because of the change in outdoor temperature. When the ventilation fan or air ventilation is operated to achieve the optimal temperature, the energy consumption is low. The optimization function selects the best temperature, and then the fuzzy control selects the best actuator based on the value of indoor and outdoor climate parameters. The outdoor temperature on Jeju island normally lies between 15 and 25 in summer, so the energy consumption is low because the optimal temperature for broccoli can be achieved by operating forced and natural ventilation. The humidity level is usually high because of the high rain rate in the experimental area. To achieve the desired humidity level within the greenhouse, a fogging system and dehumidifier is used. Due to high outdoor humidity, the dehumidifier consumes more energy as compared to fogging system. Fig. 19 shows the energy consumption for the humidification and dehumidification process for one month (30 days). The figure shows that the energy consumption with optimization is comparatively low. When outdoor humidity allows for open-air ventilation, the energy consumption is low. Broccoli needs a low level of humidity; because of that, the dehumidification process consumes more energy.

Multiple plants inject a different level of CO_2 into the environment. Forced ventilation and CO_2 generator are used to increase or decrease the CO_2 level within the greenhouse. The forced ventilation is also used for temperature and humidity to inject fresh outdoor air into the greenhouse. Fig. 20 shows the comparison of energy consumption for CO_2 with and without optimization. It is clear that the energy consumption for optimal CO_2 is less as compared to without optimization.

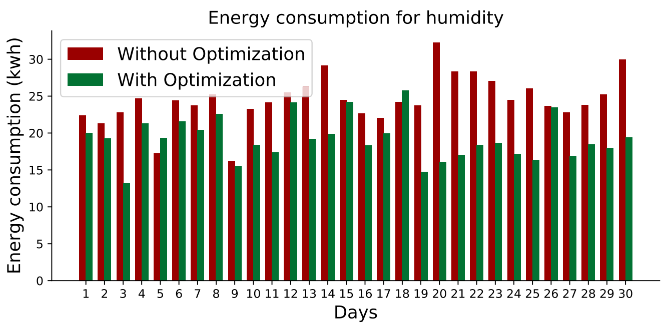


Fig. 19. Comparison of energy consumed to set the optimal humidity.

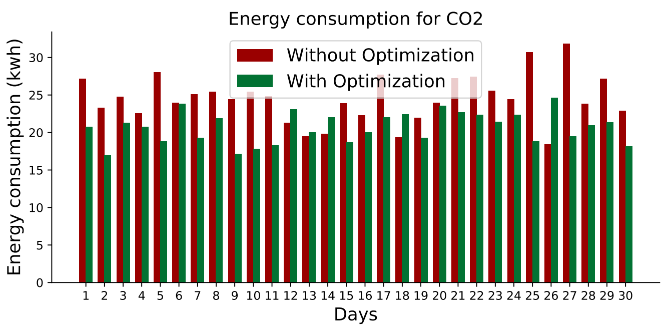


Fig. 20. Comparison of energy consumed to set the optimal CO2.

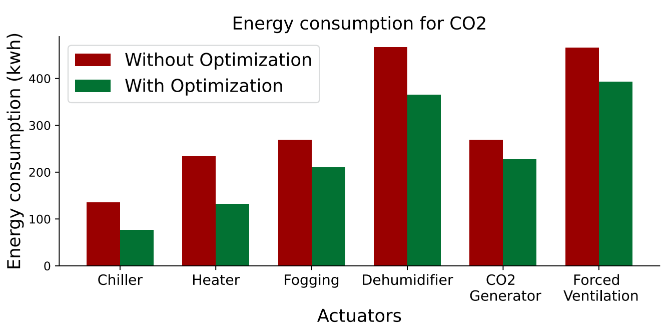


Fig. 21. Comparison of energy consumption with and without optimization for each actuator.

The energy consumption for each actuator is given in Fig. 21. Six different actuators are used to control the environment of the smart greenhouse, including a chiller, heater, fogging system (humidifier), dehumidifier, CO_2 generator, and forced ventilation. Each actuator consumes different levels of energy on different levels. For example, dehumidifiers and forced ventilation take more energy compared to other actuators. The fuzzy control selects the best actuator according to the indoor climate environment. It is clear from the figure that when the actuators are controlled without optimization, the energy consumption is too high, while in another case, the consumption of energy is low.

V. Conclusion

In this study, a cost function is used to minimize energy consumption by considering outdoor parameters has been presented. The AIOT platform contributes an essential role in optimization in the self-optimization-based greenhouse. For optimization, user preferences are considered to select the best parameters within the specified range. The owner sets the range of climate parameters based on the plant type in the greenhouse. The LSTM-based inference model is offloaded on the smart edge device to predict the next climate parameters using current values. The energy consumption on current environmental parameters is also predicted; later, the energy is compared with the energy computed after optimization. The predicted climate parameters and energy consumption are passed to the IoT platform using the CoAP protocol. CoAP protocol is used by OCF IoTivity, which is used to connect the IoT Device and platform. The platform receives the current energy price from the smart meter and user preference from the database. The greenhouse owner can set the preferences (min and max values) from the android-based interface. By considering all these parameters, the platform uses DE to get the optimal parameters and send them to the IoT device by using the same protocol. The fuzzy logic-based mapping is applied, and the desired actuator is operated. The climate parameters of the greenhouse are interrelated, so the indirect effect of each actuator is considered along with outdoor parameters, which helps to decrease energy consumption. The results are reported in terms of energy consumption for each parameter and each actuator.

Authors' Contributions

Atif Rizwan contributed to study the conception and design of proposed work. Material preparation and data collection is performed by Atif Rizwan, Anam Nawaz Khan, and Rashid Ahmad. The experiments are conducted by Atif Rizwan and Anam Nawaz Khan. The first draft is written by Atif Rizwan and reviewed and edited by Rashid Ahmad and Do Hyeun Kim. All authors commented on previous versions of manuscript and approved the final version.

Code Availability

The code is available at: https://github.com/atifrizwan91/Greenhouse/.

Conflict of Interest

The authors declare no competing interests.

References

[1] J. Širokỳ, F. Oldewurtel, J. Cigler, and S. Prívara, "Experimental analysis of model predictive control for an energy efficient building heating system," *Appl. Energy*, vol. 88, no. 9, pp. 3079–3087, 2011.
[2] I. Ullah, M. Fayaz, M. Aman, and D. Kim, "An optimization scheme for IoT based smart greenhouse climate control with efficient energy consumption," *Computing*, vol. 104, pp. 433–457, Feb. 2022.
[3] R. Rayhana, G. Xiao, and Z. Liu, "Internet of Things empowered smart greenhouse farming," *IEEE J. Radio Freq. Identif.*, vol. 4, no. 3, pp. 195–211, Sep. 2020.
[4] E. Cuce, D. Harjunowibowo, and P. M. Cuce, "Renewable and sustainable energy saving strategies for greenhouse systems: A comprehensive review," *Renew. Sustain. Energy Rev.*, vol. 64, pp. 34–59, Oct. 2016.
[5] E. Cuce, P. M. Cuce, C. J. Wood, and S. B. Riffat, "Toward aerogel based thermal superinsulation in buildings: A comprehensive review," *Renew. Sustain. Energy Rev.*, vol. 34, pp. 273–299, Jun. 2014.
[6] E. Cuce, P. M. Cuce, and S. Riffat, "Novel glazing technologies to mitigate energy consumption in low-carbon buildings: A comparative experimental investigation," *Int. J. Energy Res.*, vol. 40, no. 4, pp. 537–549, 2016.

[7] L. H. Aramyan, A. G. O. Lansink, and J. A. Verstegen, "Factors underlying the investment decision in energy-saving systems in dutch horticulture," *Agr. Syst.*, vol. 94, no. 2, pp. 520–527, 2007.
[8] X. Wang and H. Yu, "Research on control system of intelligent greenhouse of IoT based on ZigBee," *J. Phys. Conf. Series*, vol. 1345, no. 4, 2019, Art. no. 42036.
[9] W. Jin and D. Kim, "Resource management based on OCF for device self-registration and status detection in IoT networks," *Electronics*, vol. 8, no. 3, p. 311, 2019.
[10] W. Puspitasari and R. H. Y. Perdana, "Real-time monitoring and automated control of greenhouse using wireless sensor network: Design and implementation," in *Proc. Int. Seminar Res. Inf. Technol. Intell. Syst. (ISRITI)*, 2018, pp. 362–366.
[11] G. Soto-Zarazua et al., "Trends in automated systems development for greenhouse horticulture," *Int. J. Agr. Res.*, vol. 6, no. 1, pp. 1–9, 2011.
[12] R. Morais, M. A. Fernandes, S. G. Matos, C. Serôdio, P. Ferreira, and M. Reis, "A ZigBee multi-powered wireless acquisition device for remote sensing applications in precision viticulture," *Comput. Electron. Agr.*, vol. 62, no. 2, pp. 94–106, 2008.
[13] J. Han, H. Lee, and K.-R. Park, "Remote-controllable and energy-saving room architecture based on ZigBee communication," *IEEE Trans. Consum. Electron.*, vol. 55, no. 1, pp. 264–268, Feb. 2009.
[14] D. Dietrich, D. Bruckner, G. Zucker, and P. Palensky, "Communication and computation in buildings: A short introduction and overview," *IEEE Trans. Ind. Electron.*, vol. 57, no. 11, pp. 3577–3584, Nov. 2010.
[15] G. Patricio and L. Gomes, "Smart house monitoring and actuating system development using automatic code generation," in *Proc. 7th IEEE Int. Conf. Ind. Inform.*, 2009, pp. 256–261.
[16] A. Anvari-Moghaddam, H. Monsef, and A. Rahimi-Kian, "Optimal smart home energy management considering energy saving and a comfortable lifestyle," *IEEE Trans. Smart Grid*, vol. 6, no. 1, pp. 324–332, Jan. 2015.
[17] A. Ouammi, Y. Achour, H. Dagdougui, and D. Zejli, "Optimal operation scheduling for a smart greenhouse integrated microgrid," *Energy Sustain. Develop.*, vol. 58, pp. 129–137, Oct. 2020.
[18] R. Caponetto, L. Fortuna, G. Nunnari, L. Occhipinti, and M. G. Xibilia, "Soft computing for greenhouse climate control," *IEEE Trans. Fuzzy Syst.*, vol. 8, no. 6, pp. 753–760, Dec. 2000.
[19] N. Bennis, J. Duplaix, G. Enéa, M. Haloua, and H. Youlal, "Greenhouse climate modelling and robust control," *Comput. Electron. Agr.*, vol. 61, no. 2, pp. 96–107, 2008.
[20] G. Pasgianos, K. Arvanitis, P. Polycarpou, and N. Sigrimis, "A nonlinear feedback technique for greenhouse environmental control," *Comput. Electron. Agr.*, vol. 40, nos. 1–3, pp. 153–177, 2003.
[21] J. del Sagrado, J. A. Sanchez, F. Rodríguez, and M. Berenguel, "Bayesian networks for greenhouse temperature control," *J. Appl. Logic*, vol. 17, pp. 25–35, Sep. 2016.
[22] "OneM2M sets standards for the Internet of Things & M2M." OneM2M. Accessed: Jul. 8, 2021. [Online]. Available: https://www.onem2m.org/
[23] "Open connectivity foundation (OCF)." Accessed: Jul. 7, 2021. [Online]. Available: https://openconnectivity.org/
[24] "OMA SpecWorks." OMA. Accessed: Jul. 11, 2021. [Online]. Available: https://omaspecworks.org/
[25] C. A. L. Putera and F. J. Lin, "Incorporating OMA lightweight M2M protocol in IoT/M2M standard architecture," in *Proc. IEEE 2nd World Forum Internet Things (WF-IoT)*, 2015, pp. 559–564.
[26] Y. Daradkeh, D. Namiot, and M. Sneps-Sneppe, "M2M standards: Possible extensions for open API from ETSI," *Eur. J. Sci. Res.*, vol. 72, no. 4, pp. 628–637, 2012.
[27] S. K. Datta and C. Bonnet, "A lightweight framework for efficient M2M device management in oneM2M architecture," in *Proc. Int. Conf. Recent Adv. Internet Things (RIoT)*, 2015, pp. 1–6.
[28] P. Varga et al., "Making system of systems interoperable—The core components of the arrowhead framework," *J. Netw. Comput. Appl.*, vol. 81, pp. 85–95, Mar. 2017.
[29] P. Varga and C. Hegedus, "Service interaction through gateways for inter-cloud collaboration within the arrowhead framework," in *Proc. 5th IEEE WirelessVitae*, 2015, pp. 1–6.
[30] M. Albano, P. M. Barbosa, J. Silva, R. Duarte, L. L. Ferreira, and J. Delsing, "Quality of service on the arrowhead framework," in *Proc. IEEE 13th Int. Workshop Factory Commun. Syst. (WFCS)*, 2017, pp. 1–8.
[31] S. Park, "OCF: A new open IoT consortium," in *Proc. 31st Int. Conf. Adv. Inf. Netw. Appl. Workshops (WAINA)*, 2017, pp. 356–359.
[32] I. Ullah, M. Fayaz, N. Naveed, and D. Kim, "ANN based learning to Kalman filter algorithm for indoor environment prediction in smart greenhouse," *IEEE Access*, vol. 8, pp. 159371–159388, 2020.
[33] W. Kong, Z. Y. Dong, Y. Jia, D. J. Hill, Y. Xu, and Y. Zhang, "Short-term residential load forecasting based on LSTM recurrent neural network," *IEEE Trans. Smart Grid*, vol. 10, no. 1, pp. 841–851, Jan. 2019.
[34] G. Van Straten, G. van Willigenburg, E. van Henten, and R. van Ooteghem, *Optimal Control of Greenhouse Cultivation*. Boca Raton, FL, USA: CRC Press, 2010.
[35] M. Georgioudakis and V. Plevris, "A comparative study of differential evolution variants in constrained structural optimization," *Front. Built Environ.*, vol. 6, p. 102, Jul. 2020.

Atif Rizwan received the Bachelor of Science degree from the University of the Punjab, Lahore, Pakistan, in 2015, and the M.C.S. and M.S. degrees in computer science from COMSATS University Islamabad (Attock Campus), Attock, Pakistan, in 2018 and 2020, respectively.

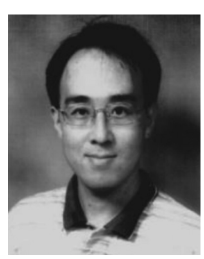
He is currently working as a Ph.D. Researcher with the Department of Computer Engineering, Jeju National University, Jeju, Republic of Korea. He has good industry experience in mobile and Web application development and testing. He was awarded a fully funded scholarship for the entire duration of his Ph.D. studies. His research interests include applied machine learning, data and Web mining, analysis, optimization of core algorithms, federated learning, and IoT-based applications.

Anam Nawaz Khan received the B.S. and M.S. degrees in computer science from COMSATS University Islamabad, Islamabad, Pakistan, in 2016 and 2019, respectively.

She joined various reputable academic institutions and later decided to expand her knowledge by continuing her academic career. In 2020, she moved to the Republic of Korea and started working as a Ph.D. Research Fellow with the Department of Computer Engineering, Jeju National University, Jeju, South Korea. She is currently working on integrating the AI-enabled IoT platforms and edge computing to provide autonomous intelligent services for smart environments. Her research interests include development and optimization of artificial intelligence and machine learning-based applications in smart spaces.

Rashid Ahmad received the B.S. degree in computer science from the University of Malakand, Totakan, Pakistan, in 2007, the M.S. degree in computer science from the National University of Computer and Emerging Sciences (FAST-NUCES), Islamabad, Pakistan, in 2009, and the Ph.D. degree in computer engineering from Jeju National University, Jeju, South Korea, in 2016.

He is currently working as a Brainpool Research Fellow with the Bigdata Research Center, Jeju National University. He is on research leave from COMSATS University Islamabad (Attock Campus), Attock, Pakistan, where he is working as an Assistant Professor. His research work spans around the applied machine learning and the application of prediction and optimization algorithms for building IoT-based edge intelligence solutions. His research interests include machine learning, distributed machine learning, continual learning, and optimization of deep learning models.



Do Hyeun Kim received the B.S. degree in electronics engineering and the M.S. and Ph.D. degrees in information telecommunication from Kyungpook National University, Daegu, South Korea, in 1988, 1990, and 2000, respectively.

He was with the Agency for Defense Development, Daejeon, South Korea, from 1990 to 1995. He has been with Jeju National University, Jeju, South Korea, since 2004, where he is currently a Professor with the Department of Computer Engineering. He was a Visiting Researcher with the Queensland University of Technology, Brisbane, QLD, Australia, from 2008 to 2009. His research interests include sensor networks, M2M/IOT, energy optimization and prediction, intelligent service, and mobile computing.