

 Open access • Journal Article • DOI:10.1109/JIOT.2016.2627403

Design and Implementation of a Cloud Enabled Random Neural Network-Based Decentralized Smart Controller With Intelligent Sensor Nodes for HVAC

— [Source link](#) 

Abbas Javed, Hadi Larijani, Ali Ahmadinia, Rohinton Emmanuel ...+2 more authors

Institutions: Glasgow Caledonian University, University of the West of Scotland

Published on: 01 Apr 2017 - IEEE Internet of Things Journal (IEEE)

Topics: Key distribution in wireless sensor networks, Sensor node, Wireless sensor network, Intelligent sensor and HVAC

Related papers:

- [Smart Random Neural Network Controller for HVAC Using Cloud Computing Technology](#)
- [Experimental testing of a random neural network smart controller using a single zone test chamber](#)
- [Random neural networks with negative and positive signals and product form solution](#)
- [A Web-Based Office Climate Control System Using Wireless Sensors](#)
- [Advanced control systems engineering for energy and comfort management in a building environment—A review](#)

Share this paper:    

View more about this paper here: <https://typeset.io/papers/design-and-implementation-of-a-cloud-enabled-random-neural-3tdjxqnpk>

Design and implementation of cloud enabled random neural network-based decentralized smart controller with intelligent sensor nodes for HVAC

Javed, Abbas; Larijani, Hadi; Ahmadiania, Ali; Emmanuel, Rohinton; Mannion, Mike; Gibson, Des

Published in:
IEEE Internet of Things Journal

DOI:
[10.1109/JIOT.2016.2627403](https://doi.org/10.1109/JIOT.2016.2627403)

Publication date:
2017

Document Version
Author accepted manuscript

[Link to publication in ResearchOnline](#)

Citation for published version (Harvard):

Javed, A, Larijani, H, Ahmadiania, A, Emmanuel, R, Mannion, M & Gibson, D 2017, 'Design and implementation of cloud enabled random neural network-based decentralized smart controller with intelligent sensor nodes for HVAC', *IEEE Internet of Things Journal*, vol. 4, no. 2, pp. 393 - 403. <https://doi.org/10.1109/JIOT.2016.2627403>

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy

If you believe that this document breaches copyright please view our takedown policy at <https://edshare.gcu.ac.uk/id/eprint/5179> for details of how to contact us.

Design and Implementation of a Cloud Enabled Random Neural Network based Decentralized Smart Controller with intelligent sensor nodes for HVAC

Abbas Javed, Hadi Larijani, Ali Ahmadiania, Rohinton Emmanuel, Mike Mannion, and Des Gibson,

Abstract—Building Energy Management Systems (BEMS) monitor and control the Heating Ventilation and Air Conditioning (HVAC) of buildings in addition to many other building systems and utilities. Wireless Sensor Networks (WSN) have become the integral part of BEMS at the initial implementation phase or latter when retro fitting is required to upgrade older buildings. WSN enabled BEMS however has several challenges which are managing data, controllers, actuators, intelligence, and power usage of wireless components (which might be battery powered). The wireless sensor nodes have limited processing power and memory for embedding intelligence in the sensor nodes. In this work, we present a random neural network (RNN) based smart controller on a Internet of Things (IoT) platform integrated with cloud processing for training the RNN which has been implemented and tested in an environment chamber. The IoT platform is modular and not limited to but has several sensors for measuring temperature, humidity, inlet air coming from the HVAC duct and PIR. The smart RNN controller has three main components: base station, sensor nodes, and the cloud with embedded intelligence on each component for different tasks. This IoT platform is integrated with cloud processing for training the RNN. The RNN based occupancy estimator is embedded in sensor node which estimates the number of occupants inside the room and sends this information to the base station. The base station is embedded with RNN models to control the HVAC on the basis of setpoints for heating and cooling. The HVAC of the environment chamber consumes 27.12% less energy with smart controller as compared to simple rule based controllers. The occupancy estimation time is reduced by our proposed hybrid algorithm for occupancy estimation that combines RNN based occupancy estimator with door sensor node (equipped with PIR and magnetic reed switch). The results show that accuracy of hybrid RNN occupancy estimator is 88%.

I. INTRODUCTION

According to CISCO [1], 50 billion devices will be connected to internet by 2020. Different types of devices can be connected to internet from small devices (RFIDs, Sensors) to large devices like TVs, Cameras etc and mobile devices like vehicles. In [2], the authors highlighted three characteristics

Manuscript received December 31, 2016; revised May 29, 2016 and July 06, 2016; accepted October 24, 2016. This work was supported in part by Innovate U.K. and in part by CENSIS U.K. Paper no.-IoT-0828-2015.R2. (Corresponding author: Hadi Larijani.)

Abbas Javed, Hadi Larijani, Ali Ahmadiania, Rohinton Emmanuel, Mike Mannion are with the School of Engineering and Built Environment, Glasgow Caledonian University, UK,
e-mail: (H.Larijani@gcu.ac.uk).

Des Gibson is with Institute of Thin Films, Sensors and Imaging, Scottish Universities Physics Alliance, University of the West of Scotland

Copyright ©2016 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org

of internet of things i.e. 1) Comprehensive sensing capabilities with the help of sensors, RFID, cameras, GPS and all kinds of devices to collect the information from the objects, 2) a reliable communication network that can collect information in a timely manner, 3) Intelligent processing by using cloud processing to process the large data and implement intelligent control technique. In Internet of Things, sensors energy consumption, the ability to communicate with heterogeneous systems, scalability, network connectivity, naming and identification are the challenging problems [3].

The generic architecture of WSN enabled BEMS is described in [4]. The WSN is an essential part of the IoT and it requires a gateway to connect to the internet. Often the nodes in the WSN have limited processing power and memory, hence limited intelligence can be embedded. This shortcoming can be covered by interfacing the WSN with the cloud processing. In [5], a cloud computing based solution is proposed for BEMS with ZigBee sensor networks. Other applications of cloud computing are for demand-side management [6], [7], [8] and load shifting [9]. A cloud computing based BEMS can also be used for implementing the complex control algorithms for HVAC control.

The HVAC control is an integral part of BEMS. There are two common technical approaches to HVAC control 1) Physical Model based techniques (such as Model Predictive Control), 2) black box techniques (such as Random Neural Networks (RNN) Artificial Neural Networks (ANN), and Support Vector Machines (SVM)). The physical model based techniques are used for HVAC control in [10], [11], [12], [13], [14], which require detailed physical model of the building. Therefore, it is difficult to implement on low cost WSN. The black box techniques are used for different types of HVAC control in [15], [16], [17], [18], [19], [20]. In [21], we compared a MPC and an ANN with an RNN and it was shown that the ANN was reliable only for the patterns included in the training dataset whereas the RNN exhibited accuracy for the patterns not included in the training dataset.

In [22], we proposed a centralised architecture for a smart controller in which intelligence is embedded in the base station. The smart HVAC controller was implemented on the base station by integrating three RNN based models i.e. RNN Occupancy Estimator model for estimating the number of occupants, RNN PMV based Setpoint Estimator model for estimating the predicted mean vote (PMV) based setpoints for heating and cooling the building and RNN HVAC control model for controlling the heating, cooling and ventilation of

the building through HVAC. The base station of the smart controller was developed on an Arduino Mega-2560 board (8KB RAM) and its battery consumption was high. In this work, we proposed a decentralised architecture for the smart controller in which intelligence is embedded in base station and indoor environment monitoring sensor node. The occupancy estimation algorithm of our previous work is improved by our proposed hybrid RNN occupancy estimation. The hybrid RNN occupancy estimation algorithm is implemented on an environment monitoring sensor node whereas the RNN based PMV setpoint estimator, and the RNN HVAC controller are implemented on the base station. Due to this decentralized architecture, it is now possible to implement the base station and sensor nodes with a low power ATmel 328P microcontroller based Moteino [23] board (2KB RAM). The environment sensor node estimates the number of occupants and sends this information to the base station which controls the HVAC for maintaining a comfortable indoor environment in the room. The base station is integrated with a gateway to upload the data on a web portal (<http://sensors.traceallglobal.com/>) and to download the trained RNN models for base station and environment monitoring sensor nodes. The main contributions of this work are:

- Implementation of novel IoT based decentralized architecture of RNN based smart controller for BEMS.
- Development and implementation of hybrid algorithm for estimating the number of occupants in the room using RNN model which integrates the RNN based occupancy estimator (using CO₂ sensor) and magnetic reed switches/PIR sensor based occupancy estimation. This was implemented on a low power embedded sensor node.
- Integration of cloud computing with low power atmel ATmega328P micro controller for data storage and to train the RNN models, in a RNN base station.

A. Related Work

The BEMS aims to reduce the energy consumption in a building by controlling the HVAC with minimum user interaction. The primary objective of a BEMS is to save energy by switching off the HVAC when it is not required and to maintain comfortable environment for the occupants. This requires BEMS to estimate the occupancy so that it can turn off the HVAC/Lights when the building is unoccupied and ensure a comfortable indoor environment when it is occupied.

1) *Learning from Occupants Preferences*: NEST [24] is a commercial self programming thermostat that learns occupant preferences for an indoor environment. A NEST turns off the heating/cooling when a home is unoccupied. It learns from occupant preferences and preheats/ precools a building by anticipating the demand for HVAC. The intelligent Dormitory (iSpace) [25] is a student study bedroom that is fitted with sensors. It learns the behaviour of the occupants under different environmental conditions e.g. closing the blinds, dimming the lights etc and then makes decisions according to the habits learnt. In [26], the authors presented a rule based BEMS that reduces energy consumption and maintains a comfortable indoor environment. The rule sets were created to ensure

comfortable temperature and humidity, luminance, air quality and track movement of occupants in order to control HVAC for reducing energy consumption.

2) *Occupancy Estimation*: Many authors have used either PIR sensors or a combination of PIR and door sensors to detect occupancy patterns. In [27], a self programming thermostat is developed which creates a setback schedule for the HVAC by using occupancy information. The system used a PIR sensor for sensing the occupancy. In [28], occupancy and sleep patterns were used to turn off the HVAC in order to save energy. The occupancy and sleep patterns were sensed by using a door sensor and a PIR sensor. Similarly in [29], the PIR and door sensor are used to detect whether the room is occupied or not. The system can conserve 10% to 15% energy by using the developed algorithm. In [30], a markov chain occupancy model is used for occupancy prediction with real time monitoring of occupancy achieved using a camera sensor network.

CO₂ sensors can also be used for occupancy detection, counting, and location of occupants. In [31], the presence and location of the occupants are detected. The number of occupants is also estimated in [32],[33]. In [34], the authors exploit the correlation between the CO₂ concentration and the occupancy levels.

A PIR sensor gives a binary output and is usually used to detect the occupancy and location of the occupant. The systems with PIR and door sensors can estimate the wrong number of occupants inside the building when more than one person is entering or leaving the building at the same time. CO₂ concentration based methods are usually slow to estimate the occupancy and can often take 20 minutes to correctly estimate the number of occupants, and the performance of the system can be effected by disturbances (opening of windows/doors, change in CO₂ concentration of inlet air supplied by HVAC).

3) *Cloud Computing enabled WSN*: In [35], [36], a sensor cloud architecture is presented in which a WSN and cloud processing are combined. In [37], the concept of virtualizing the physical sensor on the cloud is demonstrated to show that user can access it without knowing the location of the physical sensor. In [38], sensor-cloud integrations are used to run EnergyPlus simulations for optimization of different energy related objectives. Similarly in this work, sensor cloud integration is used to implement the smart controller. The training of RNN model and data storage/representation is performed on cloud platform. The trained RNN models are implemented on sensor nodes in order to reduce the computational load on the base station. The WSN is interfaced with cloud through gateway connected with the base station.

The rest of this paper is organized as follows. A brief introduction to RNNs and how training algorithms are used to train RNN models is provided in Section II. The architecture of a smart controller is described in Section III followed by an implementation of the smart controller in Section IV. The experimental results of the smart controller are given in Section V followed by conclusions in Section VI.

II. RANDOM NEURAL NETWORKS

Gelenbe [39][40] proposed a Random Neural Network (RNN) in which signals are either +1 or -1. An RNN can give a more detailed system state description because the potential of a neuron is represented by an integer rather than a binary value [41]. An RNN is easy to implement on hardware as its neurons can be represented by simple counters [42][43]. Abdelbaki in [44], compared the performance of an RNN with an ANN for unseen patterns not covered in the training data and found that an RNN accurately measured the output while an ANN failed to predict accurate output. Similarly in [45], the training time for an RNN was greater than an ANN but the RNN outperformed the ANN during the run-time phase in total calculation time. The RNN also had a strong generalization capability for the patterns not covered in the training phase. The ANNs were sensitive to the number of hidden neurons and if over-trained, the ANN memorizes the input patterns but gives poor generalization for new inputs. The applications of RNN are reported for optimization [46], [47], [48], function approximation [49], [50], and image processing [51], [52], [53]. To the best of our knowledge, there is no such application of RNN for implementing the smart controllers for buildings. The details about RNN architecture and exchange of signals between the neurons are presented in [22].

A. Hybrid Particle Swarm Optimization with Sequential Quadratic Programming

Many researchers have used a Gradient Descent (GD) algorithm for learning the weights of an RNN model. The GD algorithm is relatively easy to implement but zigzag behaviour may cause it to be stuck near a local minimum for the problems with multiple local minima. The evolutionary algorithms are used for solving optimization problem. These techniques are better than gradient base techniques as they do not get stuck in local minima. The Particle Swarm Optimization (PSO) algorithm performs well in finding the global minimum but it might be slow to converge to the global minimum while in the presence of multiple local minima, Sequential Quadratic Programming (SQP) optimization method [54] usually converges to local minima. The problem of slow convergence of PSO and local minima problem of SQP optimization is addressed by the hybridization of PSO and SQP optimization algorithm [55].

In this paper, we used a hybrid PSO-SQP algorithm for RNN training first proposed in [22]. First, RNN is trained with PSO algorithm to find the global minima, and then based on feasible start point from Adaptive Inertia Weight-Particle Swarm Optimization Algorithm(AIW-PSO) [56], SQP [54] optimization algorithm converges to global minima. The flow chart of the training algorithm is shown in Figure 1. The position vector for PSO is formulated as $X_{sd} = [w_{ih}^{+L1} w_{ho}^{+L2} w_{ih}^{-L1} w_{ho}^{-L2}]$. w_{ih}^{+L1} is positive interconnection weights between node i of layer 0 and node h of layer 1. Similarly, w_{ih}^{-L1} is negative interconnection weights between node i of layer 0 and node h of layer 1. The details of hybrid PSO-SQP algorithm are given in [22].

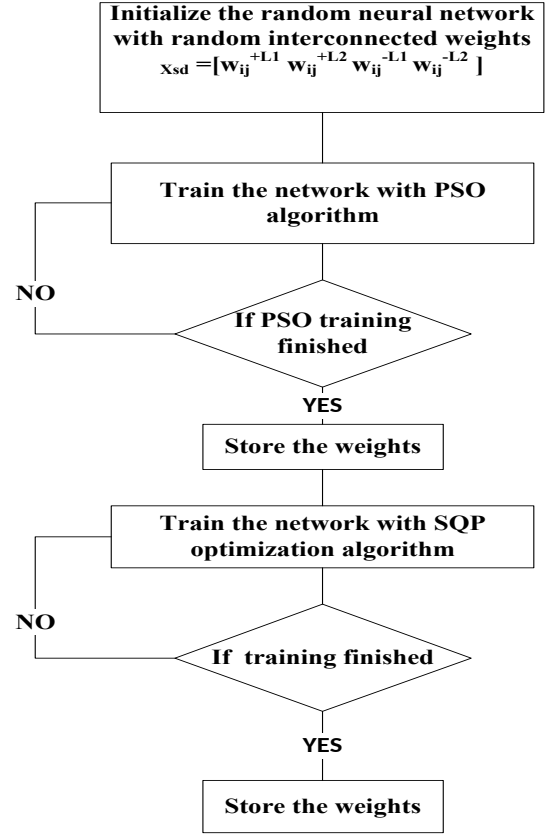


Fig. 1. Flow chart of Hybrid PSO-SQP

III. ARCHITECTURE OF DECENTRALIZED SMART CONTROLLER

The architecture of smart controller is shown in Figure 2. We assume that each zone/room has one indoor environment sensor node that communicates with a HVAC duct sensor node for monitoring the air supply and a door sensor node for estimating the number of occupants in each zone. The indoor environment sensor node estimates the number of occupants and sends this information to base station alongwith indoor air parameters (i.e., temperature, humidity, CO₂, light intensity). If the room is occupied, base station controls the HVAC to maintain PMV based setpoints (estimated by RNN PMV setpoint estimator) or maintains user defined setpoints for heating and cooling. The data from sensor nodes is uploaded on a webportal through gateways connected to the base station.

A. Indoor Environment Sensor Node

The environment monitoring sensor node is developed on Moteino R4 board [23]. The board has program memory of 32 KB, 2KB of RAM, and 1KB for EEPROM. The indoor environment sensor node is responsible for monitoring the indoor environment of the room. It monitors the temperature, humidity, dewpoint temperature, CO₂ concentration of the indoor air and light intensity of a particular room and sends this information to the base station. The DHT22 sensor is used for monitoring temperature and humidity, CO₂ ambient

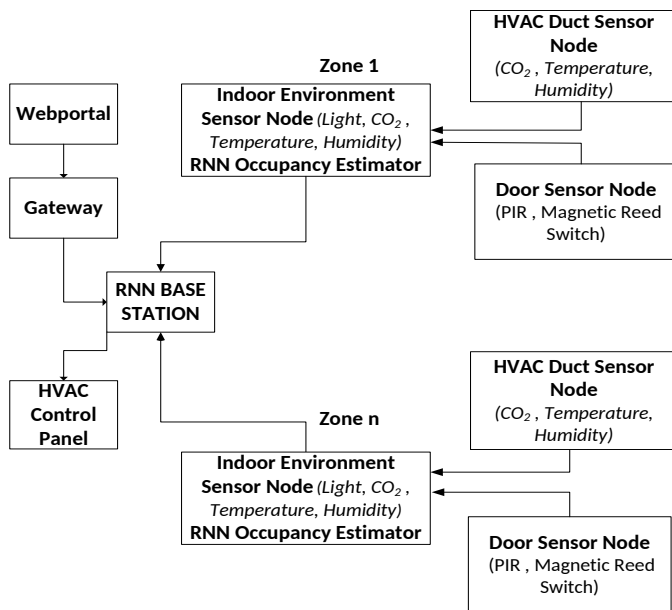


Fig. 2. Control Structure of Smart Controller



Fig. 3. Environment Monitoring Sensor Node. Algorithm for Occupancy Estimation using Door Sensor Node, RNN Occupancy Estimator and Hybrid RNN Occupancy Estimator are embedded in Environment Monitoring Sensor Node

sensor [57] is used to measure the CO₂ concentrations while photoresistor is used to measure the light intensity. The indoor environment monitoring sensor node has three algorithms for occupancy estimation i.e., RNN occupancy estimator model, occupancy estimation with door sensor node and a hybrid RNN Occupancy estimator. The occupancy estimation algorithm is implemented on the environment sensor node in order to reduce the computational load from the base station. The environment sensor node estimates the number of occupants and sends this information to the base station which will control the HVAC accordingly. The environment monitoring sensor node is shown in Fig. 3.

B. HVAC Duct Sensor Node

The environment of the room is controlled by the inlet air supplied from the HVAC. The HVAC duct sensor node calculates the temperature, humidity and CO₂ concentration of the inlet air and sends this information to indoor environment sensor node. The HVAC duct sensor node is also developed on Moteino R4 board and shown in Fig. 4.



Fig. 4. HVAC duct Sensor Node



Fig. 5. Door Sensor Node, PIR sensor Node for detecting the movement and magnetic reed switch to detect the door opening/closing

C. Door Sensor Node

This sensor node is responsible for monitoring the occupant movement and opening/closing of the door. The PIR sensor node detects the movement of the occupants and magnetic reed switch detects the door opening/closing. The magnetic reed switch is attached to the door and PIR sensor is placed inside the room to detect movements. The door sensor node is shown in Fig. 5. The door sensor node is also developed on Moteino R4 board.

D. Base Station

The base station are developed with Moteino R4 boards [23]. The base station has three responsibilities:

- 1) to implement the control algorithm for the HVAC
- 2) to upload the data on the webportal by sending the sensor node's information to the gateway
- 3) to download the trained RNN from cloud platform and install it on the environment sensor node and the base station.

E. Gateway

The Global System for Mobile Communications (GSM) module is integrated in the gateway for communication between WSN and cloud platform. For this study, the gateway

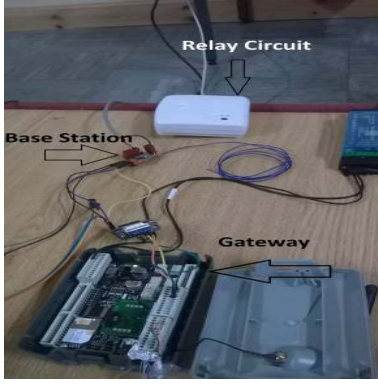


Fig. 6. Gateway connected with base station and controlling the HVAC through relay circuit. The RNN HVAC control model and RNN PMV based Setpoint Estimator model are embedded in base station.

stores the data from all sensor nodes in memory and transmits this information to Web portal after every 15 minutes. The control panel of HVAC controls the heater and chiller with 0-24 DC Voltage (VDC) relays. The base station operates at 3.3 V so the relay circuit is connected with the base station. The base station is also connected with the GSM gateway as shown in Fig. 6.

F. Webportal

The control algorithm is implemented on the base station and the environment sensor node therefore the data is uploaded on Web Portal after every 15 minutes. For each sensor node, the webportal displays Node ID, upload time in milli seconds (the time sensor node is powered), light intensity, CO₂ concentrations, temperature, humidity, dewpoint temperature, data receiving time, motion sensor, heating setpoint, cooling setpoint, heating output for HVAC, cooling output for HVAC, ventilation output for HVAC, and number of occupants in the room.

IV. IMPLEMENTATION OF SMART CONTROLLER

We have implemented the smart controller to test its performance in a single zone environment chamber.

A. Description of Testbed

The smart controller was tested in single zone environment chamber located at Glasgow Caledonian University. The environment chamber has dedicated HVAC that can humidify/dehumidify it, and can heat or cool it. The HVAC can vary the temperature of the environment chamber between 5 °C and 40°C. The size of the environment chamber is $3.65 \times 2.43 \times 2.43m$. The indoor view of the environment chamber is shown in Fig. 7.

B. Implementation of RNN Occupancy Estimator in Environment Sensor Node

The RNN Occupancy Estimator is trained on a cloud platform and the trained RNN model is implemented on an indoor environment sensor node. The inputs for the model are : HVAC

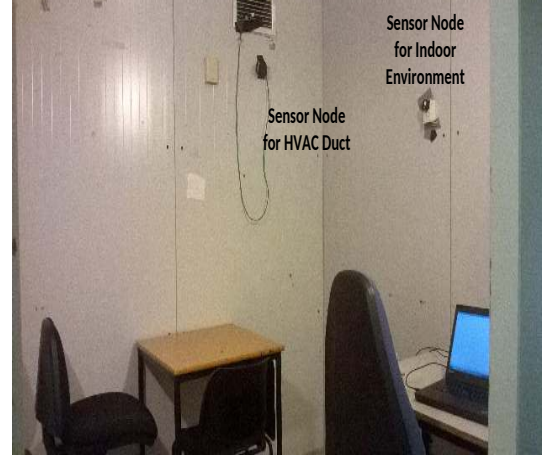


Fig. 7. Indoor View of Environment Chamber

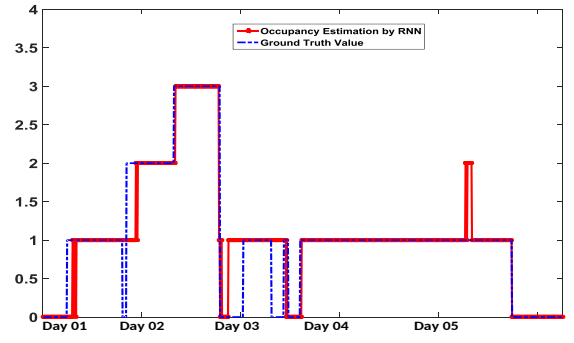


Fig. 8. Training results of RNN Occupancy Estimator, MSE = 1.40×10^{-2} . Y-axis represents number of occupants while x-axis represents number of Days.

inlet air temperature, HVAC inlet air CO₂ concentrations, inlet air temperature of the environment chamber, CO₂ concentration inside the environment chamber. The output of the model is the number of occupants inside the environment chamber. The training data set for the RNN model is downloaded from the webportal. The ground truth value is recorded manually for training purpose of RNN. In [22], the occupancy estimation algorithm used inlet air actuation signal as fifth input. In this work, the inlet air actuation signal is removed from the input of the RNN model due to which accuracy of occupancy estimation is more challenging. The RNN occupancy estimator model has four inputs in the input layer, five neurons in the hidden layer and one neuron in the output layer. The RNN model was trained with a dataset collected over a period of 5 days. The Hybrid PSO-SQP training algorithm was used for training the RNN and the mean squared error for the training data set is 1.40×10^{-2} . The training results of the RNN model are shown in Fig 8. The ground truth values for occupancy and the number of occupants estimated by the RNN model are shown in Fig. 8.

C. Occupancy Counting with Door Sensor Node

The door sensor node has a PIR sensor for detecting the movement of the occupant and magnetic reed switch to

detect the door opening/closing. The door sensor transmits the data to environment monitoring sensor node whenever an event occurs for PIR sensor/door sensor. The environment monitoring sensor node is programmed to detect two states

1) occupant enters the room : if the door sensor node detects the door opening first and the PIR sensor detects the movement afterwards, the environment monitoring sensor counts one person entering in the room. The false detection is avoided by putting the condition that the time difference between sensing the door opening and the PIR sensor should be less than 5 seconds.

2) occupant leaves the room : if the PIR sensor detects the movement first and the door sensor node detects the door opening afterwards, the environment monitoring sensor counts one person leaving the room. The false detection is avoided by putting in the condition that the time difference between sensing the door opening and the PIR sensor detecting movement should be less than 5 seconds.

D. Hybrid RNN Occupancy Estimation Algorithm

The hybrid RNN occupancy estimation algorithm is proposed by integrating the output of RNN occupancy estimator model and occupancy counting using a door sensor node. Some drawbacks of the occupancy estimation using a door sensor node are :

- 1) If more than one person comes inside the room through the door at the same time, it detects only one person.
- 2) If, at the same time, one person leaves and one person enters, this method will not work.

The RNN occupancy estimation algorithm takes time to estimate the number of occupants as CO₂ concentrations take some time to accumulate in the room in presence of ventilation. The shortcomings of both techniques are reduced by proposing the RNN based hybrid algorithm for occupancy estimation in which occupancy estimation with door sensor node is combined with RNN occupancy estimator. The RNN based hybrid occupancy algorithm is trained with the dataset collected from testing of both techniques in environment chamber. The hybrid RNN occupancy estimator has three inputs i.e. 1) occupancy estimation with RNN based occupancy estimator 2) the CO₂ concentration of the environment chamber 3) the occupancy estimation from PIR and magnetic reed switch based occupancy estimation algorithm. The hybrid RNN occupancy algorithm has four neurons in the hidden layer and one neuron in the output layer. The output of the hybrid RNN is the number of occupants in the environment chamber. The flow chart of the algorithm is shown in Fig. 9.

E. Implementation of RNN HVAC and RNN PMV estimator on base station

Predicted Mean Vote (PMV) is a commonly used indoor thermal comfort index in buildings [58]. Fanger [58], developed a thermal sensation scale of 7 values to determine thermal comfort. PMV is developed as a function of six variables: air temperature, mean radiant temperature, air velocity, air humidity, clothing resistance, and activity level. The base station is responsible for controlling the HVAC on the basis of

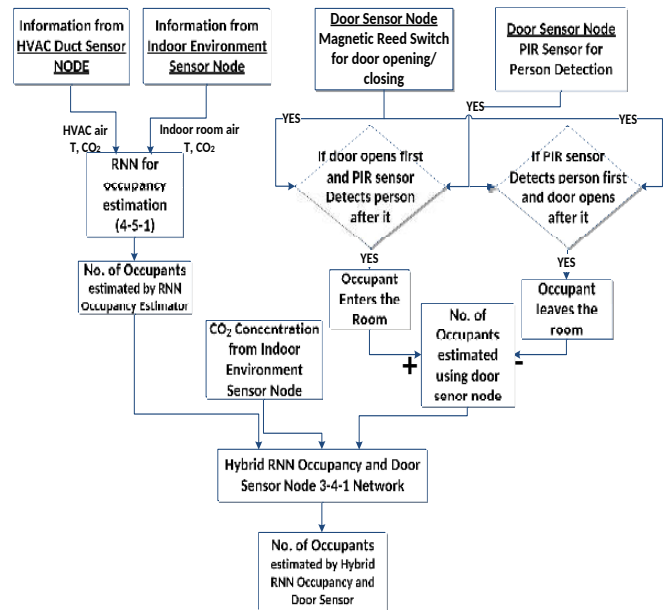


Fig. 9. Hybrid RNN Occupancy Estimation Algorithm

the information sent from the environment monitoring sensor node. If the environment monitoring sensor node detects the occupancy in a particular room, the base station will control the environment of the HVAC accordingly. The base station controls the HVAC according to two modes i.e.

- 1) thermal mode: if user selects the thermal mode, the user will enter the required PMV value and RNN PMV based setpoint estimator will determine the setpoints for heating and cooling.
- 2) user will enter its own setpoints for heating and cooling if the setpoints estimated by PMV based setpoint estimator are not satisfactory.

1) *RNN PMV based setpoint estimator*: Fanger developed seven point index of comfort/discomfort which is dependent on six variables as shown in Figure10. In this work, training data set is generated by using Fanger equation for PMV. To reduce the human interference, we assumed the typical office environment. Therefore, clothing insulation of 0.8, metabolic rate of 1.1, and air velocity of 0.15 m/s are assumed to be constant. After generating training dataset, RNN is trained with PMV and humidity as an input and temperature as an output. A 2-4-1 RNN is trained with hybrid PSO-SQP training algorithm. In this work, PMV of -0.1,-0.3,-0.5 is tested for heating setpoint and PMV of 0.3, 0.5 and 0.7 is tested for cooling set point. The RNN PMV based setpoint estimator is implemented on the base station. The estimated setpoints from the RNN model will be used by RNN HVAC control model for controlling the HVAC. In this work, when PMV value of -0.5 is selected the estimated setpoints for temperature are varied between 22.34°C and 22.47 °C.

2) *RNN HVAC Controller*: The RNN HVAC controller controls the HVAC on the basis of the setpoints estimated by the RNN PMV based setpoint estimator or user defined setpoints for heating and cooling. The inputs for the RNN model are 1) heating setpoint 2) cooling setpoint 3) heating error i.e. the difference between heating setpoint and current

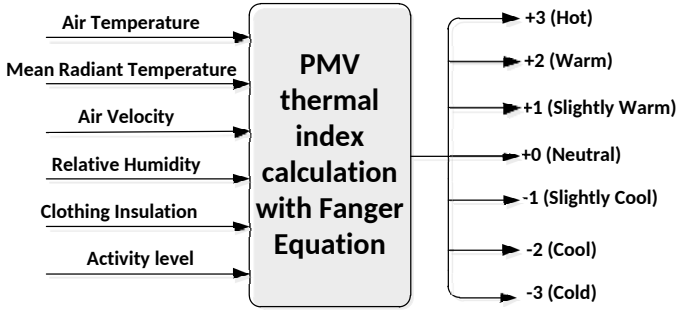


Fig. 10. PMV and thermal sensation

room temperature 4) cooling error i.e. the difference between cooling setpoint and air temperature of the zone and 5) CO₂ concentrations. The outputs of the RNN model are 1) Heating output for turning on the HVAC heating 2) cooling output for turning on the HVAC cooling and 3) ventilation for the zone. The RNN HVAC controller model is trained by the dataset collected from the environment chamber.

V. RESULTS

To investigate the impact of occupancy estimation by a hybrid RNN occupancy estimation algorithm, we evaluated occupancy estimation with RNN occupancy estimation algorithm, occupancy estimation with door sensor node and occupancy estimation with hybrid RNN occupancy estimation algorithm. In Subsection V-D, the energy consumption of smart controller is compared with simple thermostat.

A. Experiments Results of Occupancy Estimation With RNN

The RNN model for occupancy estimation is tested in an environment chamber for estimating the number of occupants. The ground truth values for occupancy are recorded manually for generating the training dataset. Multiple tests have been carried with different conditions to check the robustness of the method. The performance of the RNN algorithm is evaluated for occupancy estimation of upto 3 persons.

1) *Test 1:* During the test, the initial air temperature of the chamber was 25.6 °C and CO₂ concentrations were 610 PPM. The occupancy estimation with RNN occupancy estimator and ground truth values during the test are shown in Fig. 11. The estimation time for occupancy estimation with RNN is shown in Table I. The first column shows the change in number of occupants, the second column shows the minimum and maximum estimation time for correct occupancy estimation during the test and the third column shows the average time for estimating the occupancy. The accuracy of the method is 72.49% and when we examined the results when there were 2 occupants inside the environment chamber and RNN estimates 1 occupant, the accuracy of the method is 79.59%.

2) *Test 2:* We carried out another test where the indoor environment conditions were changed as compared to Test 1. The initial condition for the air temperature was 20.9 °C and CO₂ concentration was 587 PPM. The accuracy of the RNN estimation is 81.34% but if we compared an error of +/- 1 person when occupancy is greater than 1 i.e. 2 or 3,

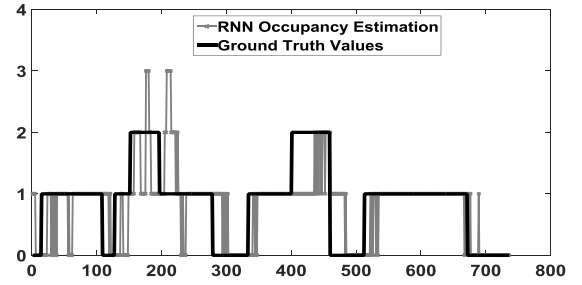


Fig. 11. Occupancy Estimation by RNN Occupancy Estimator for Test 1, y-axis represents the number of occupants while x-axis represent the time in minutes.

TABLE I
OCCUPANCY ESTIMATION TIME FOR TEST 1

Occupancy	Estimation Time (min - max)	Average Time
0 to 1 Occupant	6 - 9 minutes	7.5 minutes
1 to 2 Occupants	4 - 21 minutes	12.4 minutes
1 to 0 Occupant	4 - 16 minutes	10 minutes
2 to 1 Occupant	7 - 22 minutes	14.5 minutes
2 to 0 Occupant	25 minutes	25 minutes

the accuracy of the system is 87.45%. The results are shown in Fig. 12 and estimation time for occupancy estimation is provided in Table II.

B. Experiment results of Occupancy Estimation with Door Sensor Node

The occupancy estimation with door sensor node is tested in an office of PhD students at Glasgow Caledonian University. The office is occupied with 10 PhD students and during test the maximum number of occupants goes upto 8 occupants. The experiment was conducted during 4 days. The average

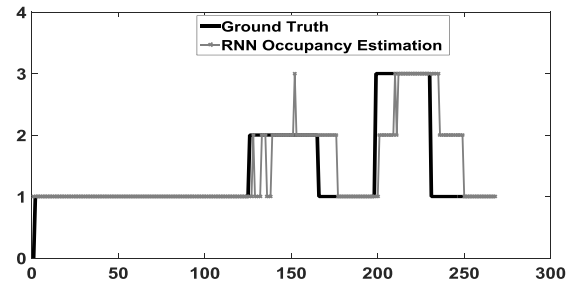


Fig. 12. Occupancy Estimation by RNN Occupancy Estimator for Test 2, y-axis represents the number of occupants while x-axis represent the time in minutes.

TABLE II
OCCUPANCY ESTIMATION TIME FOR TEST 2

Occupancy	Estimation Time (min - max)	Average Time
0 to 1 Occupant	0-1 minute	1 minute
1 to 2 Occupants	5-8 minutes	6.5 minutes
1 to 3 Occupant	10 minutes	10 minutes
2 to 1 Occupant	11-15 minutes	13 minutes
3 to 1 Occupant	16 minutes	16 minutes

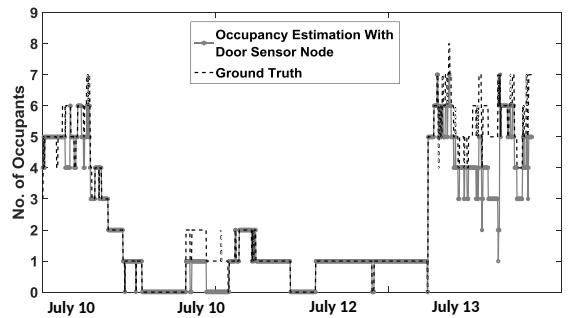


Fig. 13. Occupancy Estimation with Door Sensor Node, y-axis represents number of occupants while x-axis represents number of Days.

accuracy of occupancy estimation with PIR and magnetic reed switches was 71% and maximum error of 6 person was produced due to which occupancy estimation with door sensor node is not suitable. The results are shown in Fig. 13.

C. Experiment results of RNN Hybrid Occupancy Estimation

The RNN hybrid occupancy estimation algorithm is tested in GCU environment chamber. The occupancy estimation with RNN occupancy estimator is slow whereas the occupancy estimation with PIR and magnetic reed switch is not accurate. The hybrid algorithm reduced the drawbacks of both techniques. The experiment result shows that the occupancy estimation time is reduced as compared to RNN occupancy estimator and accuracy of occupancy estimation is also increased. The ground truth values, RNN occupancy estimator, occupancy estimation with hybrid RNN algorithm and occupancy estimation with door sensor are shown in Fig. 14. The occupancy estimation with RNN occupancy estimator is slow and due to frequent changes in occupancy, the accuracy of RNN occupancy estimator is 63% whereas accuracy of door sensor node is 72.9%. The hybrid algorithm takes advantage of both techniques and as a result the accuracy is 87.3%. As shown in Fig. 14, the estimation time for occupancy estimation is the same as door sensor node. During the test, when two persons entered in the environment chamber at the same time, the door sensor node detects only one person. This error has been corrected by hybrid occupancy algorithm after 13 minutes. We carried out another test for checking the performance of hybrid RNN occupancy estimation. The comparison of RNN occupancy estimator, and hybrid occupancy estimator is shown in Fig. 15. During this test, the occupancy estimation with door sensor node is 100% accurate whereas the accuracy of hybrid occupancy algorithm is 88%.

D. Energy Consumption of the HVAC

The energy consumption of the RNN controller is compared with simple rule based controller. The RNN smart controller maintains the setpoints for heating, cooling and ventilation if the room is occupied. The setpoint for heating was 23 °C, and the setpoint for cooling was 26 °C. During the test, the temperature of environment chamber is brought to 17.3 °C for energy consumption comparison. The air temperature of the

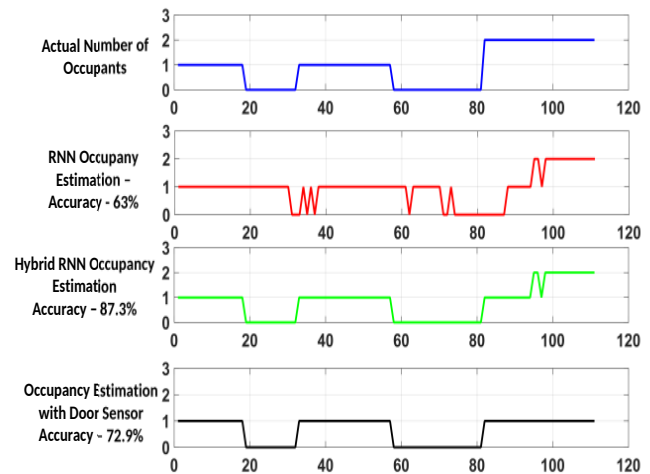


Fig. 14. Comparison of Occupancy Estimation with Hybrid Occupancy Algorithm, RNN Occupancy Estimator, Door Sensor Node and Ground Truth Values for Occupancy for Test 1, y-axis represents number of occupants, x-axis shows time in minutes

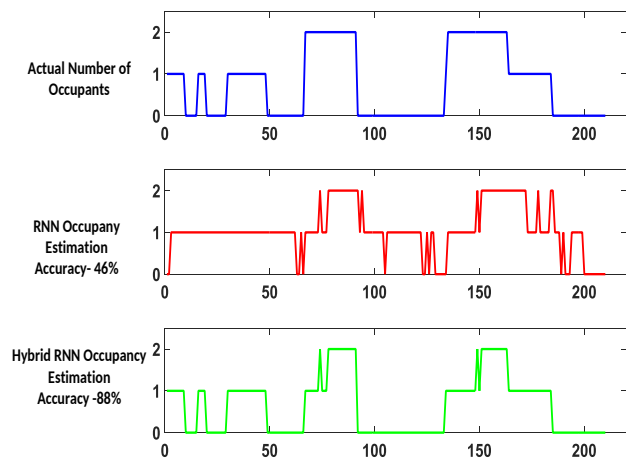


Fig. 15. Comparison of Occupancy Estimation with Hybrid Occupancy Algorithm, RNN Occupancy Estimator, Door Sensor Node and Ground Truth Values for Occupancy for Test 2, y-axis represents number of occupants, x-axis shows time in minutes

environment chamber during the test with RNN occupancy estimation is shown in Fig. 16 and occupancy estimation is shown in Fig. 17 whereas the energy consumption comparison is shown in Fig. 18. The energy consumption with occupancy estimation with RNN is 38.7 KWh and without occupancy estimation is 49.2 KWh. The total energy saving with RNN controller is 27.12 %.

E. Data uploading on Webportal

The information from sensor nodes is uploaded on the webportal. The Node ID of the environment monitoring node is 2 and Node ID of the HVAC duct sensor node is 1. The updated results are shown in Fig. 19 and the control

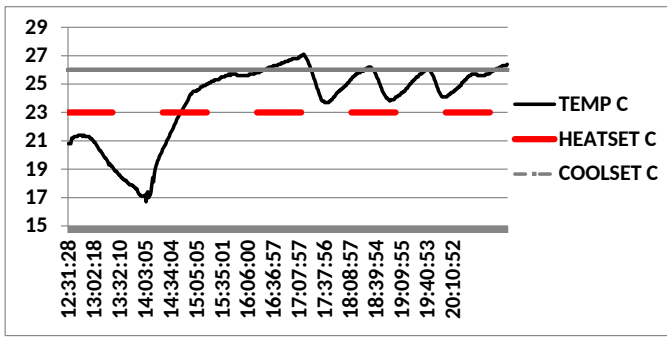


Fig. 16. Air Temperature of the Environment Chamber with RNN controller

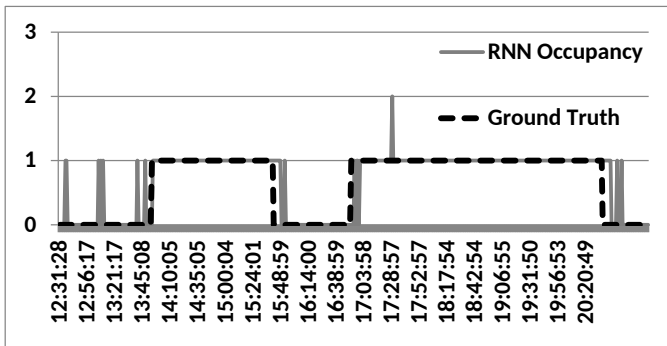


Fig. 17. Occupancy Estimation during the test for comparison of Energy Consumption

parameters for HVAC alongwith the occupancy estimated by the algorithm is shown in Fig. 20.

VI. CONCLUSION

In this work, we implemented the smart controller by integrating the internet of things with cloud computing. The building indoor environment parameters, HVAC inlet air parameters and control parameters for the HVAC are uploaded on the web portal. The WSN has been implemented with low power Moteino Board which has limited memory and processing power. The decentralized architecture of smart controller is proposed in which control algorithms are partially implemented on base station and environment monitoring sensor node placed in each zone. The sensor node for monitoring

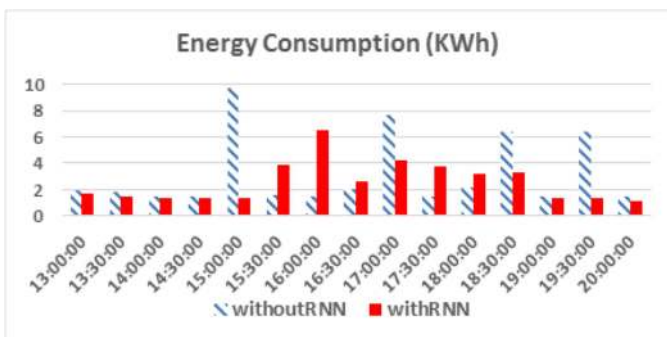


Fig. 18. Energy Consumption Comparison with RNN Smart Controller and without RNN Smart Controller



Fig. 19. View of Webportal- Information from Sensor Nodes are displayed on <http://sensors.traceallglobal.com/sensordata.asp>



Fig. 20. View of Webportal- Control Parameters of HVAC from Smart Controller and Occupancy Estimation are displayed by clicking on More Button of Main Page.

the indoor environment of the room is made intelligent by embedding the hybrid RNN based occupancy estimator. The RNN models for smart controller are trained with hybrid PSO-SQP training algorithm on cloud platform and trained RNN models are embedded in the base station and environment monitoring sensor node.

The occupancy estimation with PIR and magnetic reed switches, and RNN based occupancy estimator are tested in the environment chamber. The experiment results showed that occupancy estimation algorithm based on PIR and magnet switch may not be accurate due to its inability to count more than one person entering/leaving the room at the same time. On the other hand, the RNN based occupancy estimator can accurately estimate the number of occupants but estimation time is slow as CO₂ concentrations take time to accumulate in the room. These shortcomings of both techniques are addressed by proposing the hybrid RNN based occupancy estimation algorithm which takes output of both techniques to estimate the occupancy estimation. The results show that accuracy of occupancy estimation of hybrid algorithm is 88% and estimation time for single person is the same as PIR/magnetic reed switch based occupancy algorithm. The energy consumption of smart controller is compared with a simple thermostats and results show the energy consumption of smart controller is 27.12% less than the simple thermostat. For future work, control level of HVAC will be optimized according to the number of occupants estimated by the hybrid RNN occupancy estimation algorithm.

REFERENCES

[1] D. Evans, "The internet of things: How the next evolution of the internet is changing everything," *CISCO white paper*, vol. 1, p. 14, 2011.

- [2] M. Yun and B. Yuxin, "Research on the architecture and key technology of internet of things (iot) applied on smart grid," in *2010 International Conference on Advances in Energy Engineering (ICAEE)*. IEEE, 2010, pp. 69–72.
- [3] M. Zorzi, A. Gluhak, S. Lange, and A. Bassi, "From today's intranet of things to a future internet of things: a wireless-and mobility-related view," *Wireless Communications, IEEE*, vol. 17, no. 6, pp. 44–51, 2010.
- [4] A. H. Kazmi, M. J. O'grady, D. T. Delaney, A. G. Ruzzelli, and G. M. O'hare, "A review of wireless-sensor-network-enabled building energy management systems," *ACM Transactions on Sensor Networks (TOSN)*, vol. 10, no. 4, p. 66, 2014.
- [5] I. Hong, J. Byun, and S. Park, "Cloud computing-based building energy management system with zigbee sensor network," in *2012 Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS)*. IEEE, 2012, pp. 547–551.
- [6] H. Kim, Y.-J. Kim, K. Yang, and M. Thottan, "Cloud-based demand response for smart grid: Architecture and distributed algorithms," in *2011 IEEE International Conference on Smart Grid Communications (SmartGridComm)*. IEEE, 2011, pp. 398–403.
- [7] X. Li and J.-C. Lo, "Pricing and peak aware scheduling algorithm for cloud computing," in *2012 IEEE PES Innovative Smart Grid Technologies (ISGT)*. IEEE, 2012, pp. 1–7.
- [8] L. Ji, W. Lifang, and Y. Li, "Cloud service based intelligent power monitoring and early-warning system," in *2012 IEEE Innovative Smart Grid Technologies-Asia (ISGT Asia)*. IEEE, 2012, pp. 1–4.
- [9] C.-T. Yang, W.-S. Chen, K.-L. Huang, J.-C. Liu, W.-H. Hsu, and C.-H. Hsu, "Implementation of smart power management and service system on cloud computing," in *2012 9th International Conference on Ubiquitous Intelligence & Computing and 9th International Conference on Autonomic & Trusted Computing (UIC/ATC)*. IEEE, 2012, pp. 924–929.
- [10] J. Široký, F. Oldewurtel, J. Cigler, and S. Prívvara, "Experimental analysis of model predictive control for an energy efficient building heating system," *Applied Energy*, vol. 88, no. 9, pp. 3079–3087, 2011.
- [11] F. Oldewurtel, A. Parisio, C. N. Jones, M. Morari, D. Gyalistras, M. Gwerder, V. Stauch, B. Lehmann, and K. Wirth, "Energy efficient building climate control using stochastic model predictive control and weather predictions," in *2010 American Control Conference (ACC)*. IEEE, 2010, pp. 5100–5105.
- [12] F. Oldewurtel, A. Parisio, C. N. Jones, D. Gyalistras, M. Gwerder, V. Stauch, B. Lehmann, and M. Morari, "Use of model predictive control and weather forecasts for energy efficient building climate control," *Energy and Buildings*, vol. 45, pp. 15–27, 2012.
- [13] Y. Ma, F. Borrelli, B. Hancey, B. Coffey, S. Bengea, and P. Haves, "Model predictive control for the operation of building cooling systems," *IEEE Transactions on Control Systems Technology*, vol. 20, no. 3, pp. 796–803, 2012.
- [14] S. Prívvara, J. Široký, L. Ferkl, and J. Cigler, "Model predictive control of a building heating system: The first experience," *Energy and Buildings*, vol. 43, no. 2, pp. 564–572, 2011.
- [15] A. Ruano, E. Crispim, E. Conceição, and M. Lucio, "Prediction of building's temperature using neural networks models," *Energy and Buildings*, vol. 38, no. 6, pp. 682–694, 2006.
- [16] J. Liang and R. Du, "Thermal comfort control based on neural network for hvac application," in *CCA 2005. Proceedings of 2005 IEEE Conference on Control Applications*. IEEE, 2005, pp. 819–824.
- [17] M. Gouda, S. Danaher, and C. Underwood, "Application of an artificial neural network for modelling the thermal dynamics of a building's space and its heating system," *Mathematical and Computer Modelling of Dynamical Systems*, vol. 8, no. 3, pp. 333–344, 2002.
- [18] J.-Y. Lee, M.-S. Yeo, and K.-W. Kim, "Predictive control of the radiant floor heating system in apartment buildings," *Journal of Asian Architecture and Building Engineering*, vol. 1, no. 1, pp. 105–112, 2002.
- [19] J. W. Moon and J.-J. Kim, "Ann-based thermal control models for residential buildings," *Building and Environment*, vol. 45, no. 7, pp. 1612–1625, 2010.
- [20] I.-H. Yang, M.-S. Yeo, and K.-W. Kim, "Application of artificial neural network to predict the optimal start time for heating system in building," *Energy Conversion and Management*, vol. 44, no. 17, pp. 2791–2809, 2003.
- [21] A. Javed, H. Larjani, A. Ahmadinia, and R. Emmanuel, "Comparison of the robustness of rnn, mpc and ann controller for residential heating system," in *2014 IEEE Fourth International Conference on Big Data and Cloud Computing (BdCloud)*. IEEE, 2014, pp. 604–611.
- [22] A. Javed, H. Larjani, A. Ahmadinia, R. Emmanuel, D. Gibson, and C. Clark, "Experimental testing of a random neural network smart controller using a single zone test chamber," *Networks, IET*, vol. 4, no. 6, pp. 350–358, 2015.
- [23] "Moteino R4," <https://lowpowerlab.com/moteino/>, 2015, [Online; accessed 21-May-2015].
- [24] "NEST thermostat," <https://nest.com/thermostat/meet-nest-thermostat/>, 2015, [Online; accessed 16-August-2015].
- [25] V. Callaghan, M. Colley, H. Hagra, J. Chin, F. Doctor, and G. Clarke, "Programming ispacesa tale of two paradigms," in *Intelligent Spaces*. Springer, 2006, pp. 389–421.
- [26] H. Doukas, K. D. Patlitzianas, K. Iatropoulos, and J. Psarras, "Intelligent building energy management system using rule sets," *Building and Environment*, vol. 42, no. 10, pp. 3562–3569, 2007.
- [27] G. Gao and K. Whitehouse, "The self-programming thermostat: optimizing setback schedules based on home occupancy patterns," in *Proceedings of the First ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings*. ACM, 2009, pp. 67–72.
- [28] J. Lu, T. Sookoor, V. Srinivasan, G. Gao, B. Holben, J. Stankovic, E. Field, and K. Whitehouse, "The smart thermostat: using occupancy sensors to save energy in homes," in *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*. ACM, 2010, pp. 211–224.
- [29] Y. Agarwal, B. Balaji, R. Gupta, J. Lyles, M. Wei, and T. Weng, "Occupancy-driven energy management for smart building automation," in *Proceedings of the 2nd ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Building*. ACM, 2010, pp. 1–6.
- [30] V. L. Erickson and A. E. Cerpa, "Occupancy based demand response hvac control strategy," in *Proceedings of the 2nd ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Building*. ACM, 2010, pp. 7–12.
- [31] N. Nassif, "A robust co 2-based demand-controlled ventilation control strategy for multi-zone hvac systems," *Energy and buildings*, vol. 45, pp. 72–81, 2012.
- [32] S. A. Mumma, "Transient occupancy ventilation by monitoring co₂," *ASHRAE IAQ Applications*, pp. 21–23, 2004.
- [33] G. Ansanay-Alex, "Estimating occupancy using indoor carbon dioxide concentrations only in an office building: a method and qualitative assessment," in *11th REHVA World Congress "Energy efficient, smart and healthy buildings: Clima 2013*, 2013.
- [34] A. Ebadat, G. Bottegal, D. Varagnolo, B. Wahlberg, and K. H. Johansson, "Estimation of building occupancy levels through environmental signals deconvolution," in *Proceedings of the 5th ACM Workshop on Embedded Systems For Energy-Efficient Buildings*. ACM, 2013, pp. 1–8.
- [35] W. Kurschl and W. Beer, "Combining cloud computing and wireless sensor networks," in *Proceedings of the 11th International Conference on Information Integration and Web-based Applications & Services*. ACM, 2009, pp. 512–518.
- [36] S. K. Dash, J. P. Sahoo, S. Mohapatra, and S. P. Pati, "Sensor-cloud: assimilation of wireless sensor network and the cloud," in *Advances in Computer Science and Information Technology. Networks and Communications*. Springer, 2012, pp. 455–464.
- [37] M. Yuriyama and T. Kushida, "Sensor-cloud infrastructure-physical sensor management with virtualized sensors on cloud computing," in *Network-Based Information Systems (NBIS), 2010 13th International Conference on*. IEEE, 2010, pp. 1–8.
- [38] I. Petri, O. Rana, Y. Rezgui, H. Li, T. Beach, M. Zou, J. Diaz-Montes, and M. Parashar, "Cloud supported building data analytics," in *2014 14th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid)*. IEEE, 2014, pp. 641–650.
- [39] E. Gelenbe, "Random neural networks with negative and positive signals and product form solution," *Neural computation*, vol. 1, no. 4, pp. 502–510, 1989.
- [40] —, "Stability of the random neural network model," *Neural computation*, vol. 2, no. 2, pp. 239–247, 1990.
- [41] S. Timotheou, "The random neural network: a survey," *The computer journal*, vol. 53, no. 3, pp. 251–267, 2010.
- [42] C. Çerkez, I. Aybay, and U. Halici, "A digital neuron realization for the random neural network model," in *International Conference on Neural Networks, 1997.*, vol. 2. IEEE, 1997, pp. 1000–1004.
- [43] H. Abdelbaki, E. Gelenbe, and S. E. El-Khamy, "Analog hardware implementation of the random neural network model," in *IJCNN 2000 Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks, 2000.*, vol. 4. IEEE, 2000, pp. 197–201.
- [44] H. Abdelbaki, "Random neural network simulator (rnnsim) v. 2," *Free simulator available at ftp://ftp.mathworks.com/pub/contrib/v5/nnet/rnnsimv2*, 1999.

- [45] S. Mohamed and G. Rubino, "A study of real-time packet video quality using random neural networks," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, no. 12, pp. 1071–1083, 2002.
- [46] J. Aguilar and E. Gelenbe, "Task assignment and transaction clustering heuristics for distributed systems," *Information Sciences*, vol. 97, no. 1, pp. 199–219, 1997.
- [47] A. Jose, "Definition of an energy function for the random neural to solve optimization problems," *Neural Networks*, vol. 11, no. 4, pp. 731–737, 1998.
- [48] E. Gelenbe and S. Timotheou, "Random neural networks with synchronized interactions," *Neural Computation*, vol. 20, no. 9, pp. 2308–2324, 2008.
- [49] E. Gelenbe, Z.-H. Mao, and Y.-D. Li, "Function approximation with spiked random networks," *IEEE Transactions on Neural Networks*, vol. 10, no. 1, pp. 3–9, 1999.
- [50] —, "Function approximation by random neural networks with a bounded number of layers," *Journal of Differential Equations and Dynamical Systems*, vol. 12, no. 1-2, pp. 143–170, 2004.
- [51] E. Gelenbe, K. F. Hussain, and H. Abdelbaki, "Random neural network texture model," in *Electronic Imaging. International Society for Optics and Photonics*, 2000, pp. 104–111.
- [52] A. Teke and V. Atalay, "Texture classification and retrieval using the random neural network model," *Computational Management Science*, vol. 3, no. 3, pp. 193–205, 2006.
- [53] R. Lu and Y. Shen, "Image segmentation based on random neural network model and gabor filters," in *Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005. 27th Annual International Conference of the. IEEE*, 2006, pp. 6464–6467.
- [54] W. Hock and K. Schittkowski, "A comparative performance evaluation of 27 nonlinear programming codes," *Computing*, vol. 30, no. 4, pp. 335–358, 1983.
- [55] T. A. A. Victoire and A. E. Jeyakumar, "Hybrid pso–sqp for economic dispatch with valve-point effect," *Electric Power Systems Research*, vol. 71, no. 1, pp. 51–59, 2004.
- [56] M. Georgiopoulos, C. Li, and T. Kocak, "Learning in the feed-forward random neural network: A critical review," *Performance Evaluation*, vol. 68, no. 4, pp. 361–384, 2011.
- [57] D. Gibson and C. MacGregor, "A novel solid state non-dispersive infrared co2 gas sensor compatible with wireless and portable deployment," *Sensors*, vol. 13, no. 6, pp. 7079–7103, 2013.
- [58] P. Fanger, "Thermal comfort: analysis and applications in environmental engineering," 1972.