Journal of Engineering Science and Technology
Vol. 17, No. 1 (2022) 0798 - 0819

BIG DATA DRIVEN HEMS TOWARDS DG INTEGRATION USING ML ALGORITHM BASED LMA

M. PRATAPA RAJU*, A. JAYA LAXMI

Electrical and Electronics Engineering Dept., Jawaharlal Nehru Technological University Hyderabad, Hyderabad-500085, India.
*Corresponding Author: pathu.sush@gmail.com

Abstract

In Smart Grid (SG) environment, digitalization of micro grid to effectively balance the energy sources, loads and accommodating renewable Distributed Generation (DG) integration are few growing concerns. Active management of distributed network has provided numerous solutions to achieve DG integration. But, the advent of Information and Communication Technologies (ICT), Artificial Intelligence (AI) and data analytics has totally changed the paradigm of renewable DG integration. On the other hand, equipping the existing Home Energy Management System (HEMS) with the said ICT and AI based prediction abilities has equally grown. Towards that, HEMS with latest Demand Response (DR) and Demand Side Management (DSM) algorithms enhancing the penetration of renewable DGs into micro grid is also seen in the literature. To move a step further, this paper proposes Big Data driven HEMS to accommodate the said DG integration by implementing DSM based Load Management Algorithm (LMA) and Load Priority Assignment Algorithms (LPAA). LMA and LPAA used in this paper are designed using load priority and peak clipping Demand Side Management (DSM) techniques. Hourly Load Priority Table (LPT) and Hourly Threshold Power (P_{Th}) required to run LMA are obtained from ML based predictions. Temperature (T) based LPAA is used to obtain hourly LPT, where T is predicted using ML based prediction model. And to obtain P_{Th}, ML based total Load Demand (P_T) predictions are used along with LPT resulted from LPAA. Four ML algorithms namely Linear Regression (LR), Ensemble Bagged Tree (EBT), SVR (Support Vector Regression), and GPR (Gaussian Process Regression) are trained with load and weather Big Data in order to make T and P_T predictions. SIMULINK model of the Data driven HEMS is developed in MATLAB-SIMULINK environment using MATLAB 2018b. This model can predict the P_T, P_{Th}, LPT and CO_2 emissions for any given date, time and weather parameters. Thus, size of the DGs for any given date can be predicted. A hardware model of HEMS is fabricated to demonstrate proposed AI based DG integration. Simulation and experimental results obtained are presented to showcase the ability of Big Data driven HEMS in achieving DG integration. Inferences drawn in terms of reduction in CO_2 emissions resulted using proposed Big Data driven HEMS are also presented.

Keywords: Big data, Demand side management algorithms, DG integration, Home energy management system, Machine learning algorithms.

1. Introduction

The escalating energy demand due to ever raising population globally has direct effect on energy generation and management. The dependency on fossil fuels to meet the energy needs has contributed to increase in CO_2 emissions. This poses a significant threat of global warming, as revealed in the "World Energy Outlook 2019" by the International Energy Agency. Looking into energy statistics, monthly details of energy consumption recorded by Energy Information Administration (EIA) in US, reveals that, a major portion of the energy is consumed by residential dwellings [1].

Residential load centres are one of the potential reasons triggering a significant mismatch between supply and demand. To address this, Micro grids with DGs are considered as a promising measure, as presented by Puttgen et al. [2] and Ackermann and Knyazkin [3]. These DGs and Micro grid environment are effectively functional since the failure of the conventional grid during 2003 in the US.

Peera et al. [4], unleashed a new methodology of renewable DG integration with emphasis on solar, wind and load power predictions. This article also presents the use of prediction models to estimate the DG requirements and DG integration processes. Boehm et al. [5] presented forecasting models focusing on flexibilities in energy supply and demand. This helps to manage the DG availability and power consumption in the smart grid environment. Prediction models presented can track the changes accurately with new measurements and enable proper load scheduling when necessary. This model uses the combination of SVM and ensemble ML algorithms to predict load demand.

Barbato et al. [6] and Reinhardt et al. [7] predicted the load demand through meter readings from domestic load centres is discussed by. This accommodates the detection of the flexible energy usage by the connected autonomous users on the demand side. The work presented by Ulbricht et al. [8] is extended by Kaulakien, et al. [9] by suggesting methods to extract the flexibilities from electricity time series. Medrano et al. [10] have explained about the integration of DG systems into generic type of commercial buildings of southern California. This presents the study of main economic, energy - efficient and environmental impacts of the DG integration. Detailed load profiles for four commercial building types during the peak electric and peak gas consumption were analysed.

Dyson et al. [11] have presented solar energy integration using smart meter data through Demand Response (DR). This work is carried out with hourly load and weather data of northern California. In this process of DG integration, linear regression and unsupervised classification ML algorithms are developed. These models use entire home consumption data and outdoor temperature data to estimate the active hours of Air conditioners. Obviously, dependency of the cooling load on the temperature is studied.

Al-Ali et al. [12] have established two-way communication network in order to integrate the renewable sources with the conventional grid supply. In this work, H gate way at the house and U server at the service provider are installed. Forecasting of 50 peak hours are informed to the customer based on which the loads at the load centre are shifted from conventional to renewable sources of energy.

Researchers have also presented Artificial Intelligence based renewable DG integration through the arrangements made at domestic load centre. To mention few of them, DSM using Artificial Intelligence in smart grid environment reported

by Macedo et al. [13] who presented load classification using ANN based DSM choice for system optimization and dynamic pricing.

Jaramillo and Weidlich [14] focused on peak load reduction for load centres such as hospitals, hotels, educational institutions, and commercial buildings by integrating Renewable Energy Sources (RES) and ESS. This work also emphasizes on optimal load scheduling to manage the energy available at higher efficiency. An intelligent optimization algorithm is published by Elkazaz et al. [15] for the optimal operation of distributed energy resources (DERs) online (both Fuel Cell and PV) for domestic load applications.

Peak reduction and cost saving for smart homes by embedding Genetic Algorithm (GA), Cuckoo search algorithm and Binary Particle Swarm Optimization algorithms into HEMS is reported by Javaid et al. [16], which resulted in enhanced power system reliability. Residential energy management by integrating PV systems, wind turbine system, battery storage and electrical vehicle using Mixed Integer Linear Programming (MILP) is reported by Melhem et al. [17] which witnessed an increase in application efficiency with reduced environmental hazards. AI based Load Management Algorithm (LMA) using load priority and peak clipping DSM techniques is presented by Raju and Laxmi [18].

In smart home environment, installation of HEMS at a domestic load is inevitable, and it can be effectively configured to execute the AI based DG integration. In 1990, Dick et al. [19] have first developed an Energy Management Unit (EMU) with its complete architecture to manage the domestic load consumption.

In recent past, HEMS has taken the place of EMU in smart grid environment by incorporating the latest Information and Communication Technology (ICT) with it. A lot of progress in HEMS hardware designs with simulations is witnessed in the work presented by Farhangi [20] and Vojdani [21], to prove that the HEMS is a pathway to the future smart grid environment. As presented by Wang et al. [22] DSM Techniques are also implemented in HEMS to effectively manage the loads using intelligent billing and trading system. It is witnessed by Das et al. [23] that HEMS can also incorporate the future prediction algorithms to plan for the load priority and load scheduling. In 2013, Hu and Li [24] presented a hardware system named Smart Home Management System (SHMES) focusing on dynamic price response. They have presented ML algorithms based intelligent system in order to suggest economical electricity usage. Pipattanasomporn et al. [25] have presented an intelligent algorithm for HEMS to manage the loads as per the priority assignment. This paper also presents a GUI showing the comfort level and priority indicated by the customer. This GUI updates every single minute, which is simulated using C-language. Pipattanasomporn et al. have considered four loads with presumed priority.

In 2017, Mahapatra et al. [26] have proposed an energy efficient decision-making system, named as HEMSaaS by applying neural fitting Q-learning method on the load data collected from a Canadian residential building. This work uses an on-off wireless sensor which operates with 90-250V, 10A (maximum). This wireless sensor is connected in series with power lines and allows the system to turn the devices on and off. The Main Command and Control Unit (MCCU) consists of Raspberry-Pi3 deploying node-red platform, which is programmed using python code. Logenthiran et al. [27] have presented a HEMS with day-ahead DSM strategy based on shifting technique by using evolutionary algorithm, but the user comfort and RES integration are not mentioned.

Angelis et al. [28] published about an Energy Management System (EMS) for smart home with RES, ESS using Mixed Integer Linear Programming (MILP), however, formation of Peak during peak time of the day at the load end is not reported. Di Santo et al. [29] published an active DSM for smart home with PV and ESS using Artificial Neural Networks (ANN) for HEMS, but the range of appliances to be controlled and their specifications are ignored.

Yuce et al. [30] have reported a HEMS scheduling algorithm for the appliances in smart home using ANN, GA, ANN-GA, but customers choice, comfort and electricity charges incurred for the suggested schedules are ignored. Ahmed et al. [31] have presented an explicit home energy management scheduling for HEMS using Lightning Search Algorithm (LSA) and ANN. However, it has a limitation that, the efficiency of the system falls if more devices are considered.

Kazmi et al. [32] have published energy consumption predictions, control, and appliances management Polak-Ribiére Gradient Back Propagation Networks (PRGBNNs) without considering end users' comfort. Ali et al. [33] have presented Distributed DR program for islanded Micro Grid (MG) using Diffusion strategy, consensus algorithm disregarding user's comfort. Atef and Eltawil [34] have presented electricity price forecasting-based HEMS using SVR and Deep Learning (DL) but explicit prices are not considered for different consumer sections. Lu et al. [35] have presented hour-ahead DR algorithms for HEMS using Reinforced Learning (RL) and ANN, but RES integration and peak clipping is not considered. A low-cost HEMS architecture implementing renewable DG integration is reported by Raju and Laxmi [36], but the system proposed is limited to only ANN and NARX predictions.

In summary, the work presented in [4-12] discusses the feasibility of the DG integration, load and supply balancing but AI based methodology for DG integration is not presented. Research presented in [13-17] demonstrated optimal load scheduling and AI based integration of renewable energy sources, but ML based peak clipping and load priority predictions using weather, load and time stamp variables are not presented in any of the articles.

However, in [18] and [36], a LMA is proposed which uses ANN based predictions to integrate DGs available at domestic load centre. But in [18], load priority is assumed and also hardware implementation of AI based DG integration is not presented, whereas in [36], only ANN and NARX based LPAA and LMA are implemented to investigate the AI based DG integration. The work presented in [20-35] report HEMS Technology using prediction algorithms such as ML and ARIMA prediction models and suggests optimal load scheduling. Mostly, they address the issue of dynamic and optimal pricing, PAR reduction. Similarly, recent updates in HEMSs presented in [22-30] focus on prediction algorithms, DSM, DR based optimal load scheduling towards RES integration. However, Big Data based peak clipping and load priority predictions for HEMS to integrate renewable and conventional DGs is not emphasized.

This paper aims to simulate and fabricate a cost-effective HEMS for a domestic load centre to execute AI based DG integration by implementing LR, EBT, SVR and GPR ML algorithms. In addition, ML predictions based future learning of hourly load priority, peak clipping power, renewable and non-renewable DG requirements is also presented in this article. The HEMS presented in this work is a self-learning system which not only learns the above-mentioned aspects but also integrates DGs available in smart home environment.

Rest of the paper is organized as follows: Section 2 presents different stages of the AI based DG integration using ML based predictions. Section 3 discusses the deployment with simulation stages of HEMS development. Section 4 presents the simulation and experimental results exhibiting the ability of AI based Integration. Section 5 describes the conclusion, limitations and future scope of the research work presented in this article.

2.Methodology for AI based DG Integration: Approach

A specific objective function is proposed to study the feasibility of the AI based DG integration. It is a maximization function, which represents the difference between total DG supply available at the load centre and hourly load peak clipping power. If load peak clipping power is effectively estimated then the function gets maximized. The complete objective function is presented in Eq. (1) below.

$$\max\ Diff = [Supply\ (k) - P_{nh}\ [i][j]\ ] \tag{1}$$

$$P_{nh}\ [i][j]\ = P_{Th} = f\left[f\left(P_{Tf}\right), f(P_{LR}), f(P_{LS})\right] \tag{2}$$

$$P_{nh}[i][j] = P_{Th} = P_{Tf}[i] - \left[\int_0^{LS} P_{LS}\, dLS - \int_0^{LR} P_{LR}\ dLR\right]\ [j] \tag{3}$$

where, 'i' is the factor of accuracy of total power consumption, 'j' is the factor of accuracy of load priority prediction (estimation), 'P_{LS}', 'P_{LR}' represent least and lower priority loads respectively, 'P_{Tf}' represents forecasted/ predicted total load power/ load demand for the hour and $P_{nh}[i][j]$ $or\ 'P_{Th}'$ is the threshold power (peak clipping power) for the hour.

This work presents a peak clipping power for the hour which is the peak power after clipping out the lower/ least priority load powers. This power is termed as 'Threshold Power (P_{Th})' throughout this article. 'P_{Th}' is formulated in terms of 'P_{Tf},' 'P_{LS},' and 'P_{LR}' and its complete expression is represented in equation (2) and (3). These 'P_{Th},' 'P_{LS}' and 'P_{LR}' plays a vital role in implementing LMA towards the proposed DG integration. Apparently, estimation of 'P_{Th},' 'P_{LR}' and 'P_{LS}' relays on the accuracy of the AI based prediction algorithms. This work demonstrates Big Data trained ML algorithms for required predictions and investigates the feasibility of the proposed method of DG integration.

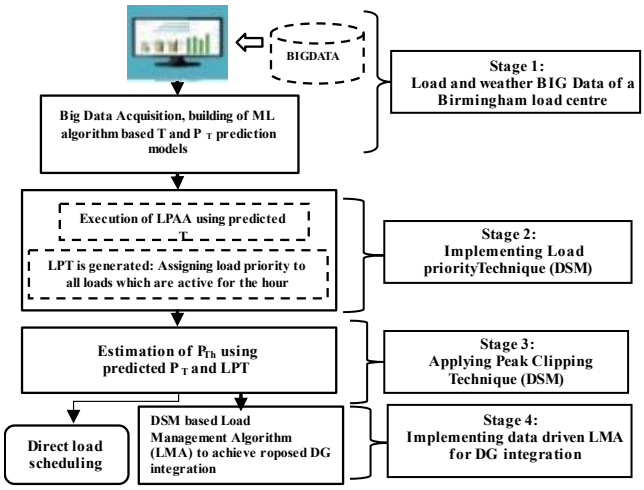


Fig. 1. HEMS implementation stages.

A specific methodology is followed to achieve the proposed AI based DG integration using low-cost HEMS prototype. Detailed description of all four stages is presented in Fig.1 and the same is discussed in the subsequent sections.

2.1. Stage-1: Big Data collection and building ML algorithms

This section includes Big Data acquisition and building of ML based T and P_T prediction models. Big Data of load and weather data of a domestic load centre located at Birmingham municipality, city of Alabama, USA, is used for prediction purposes [37]. Big Data considered for domestic load centre is arranged hour wise with equal time stamps. The Big Data comprises of 8760 samples of individual load profiles such as 'heating', 'cooling', 'water heater', exterior and interior 'lighting' loads which are presented in Fig. 2. Also, 8760 hourly samples of weather parameters of the year 2013 are collected from [38], and the parameters such as 'temperature', 'dew point temperature', 'humidity' and 'pressure' are presented in Fig. 3. These load and weather data sets are used to train ML algorithms. In addition to load and weather data sets, date and time stamps are also recorded and used for training ML algorithms to make predictions for any given date.

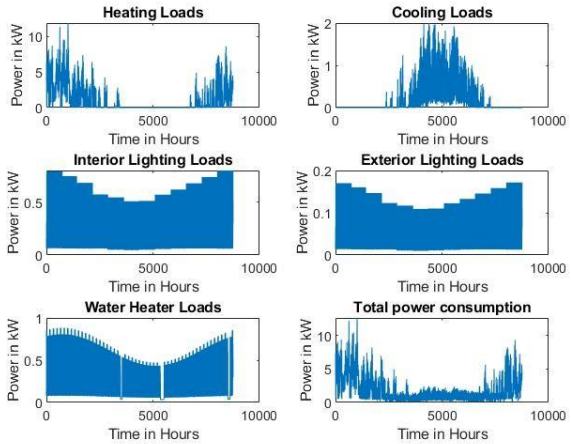


Fig. 2. Big data - load consumption details: domestic load centre.

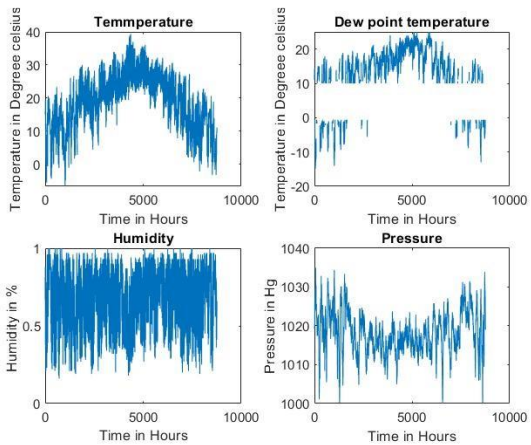


Fig. 3. Big data - weather parameters: domestic load.

Using the collected data sets, two prediction models such as Temperature (T) and P_T (total load demand) are built with ML algorithms. And these models will be discussed in the subsequent section.

2.1.1. ML algorithm based Temperature (T) prediction model

T prediction model plays a key role in assigning the priority to the loads present at the load centre. Based on T predictions obtained, LPAA is executed in order to generate hourly LPT. Four different ML algorithms are developed and trained to predict T. 8760 samples of four variables (date and time stamp) such as, 'day of the week', 'date of the month', 'month of the year' and 'time (hour in the day)' are considered as predictors to train ML algorithms in order to predict T (response). By this, prediction of T can be obtained for any given date and time stamps. Out of four MLs trained, best fit ML algorithm with lowest RMSE and highest R^2 is finally used to build the SIMULINK based prediction model which can be simulated whenever needed. Furthermore, P_T prediction is also carried in stage 1, and it is presented in subsequent section.

2.1.2. ML algorithm based Total Power (P_T) prediction model

P_T prediction model is vital in estimating P_{Th} which, in turn, is crucial in successfully executing LMA. 8760 samples of date and time stamp records such as 'hour of day', 'day of the week', 'date of the month', and 'month of the year' and 8760 hours samples of historical weather data such as T, 'humidity', 'dew point', 'pressure' for the said domestic load centre are used as predictors to train ML algorithms for predicting the total load consumption P_T. By this, prediction of P_T can be made for any given date and time stamps. As mentioned earlier, four different ML algorithms are trained to predict P_T, out of which the best fit algorithm is considered for developing a SIMULINK based P_T prediction model. The simulation and the results concerning P_T prediction are presented in the implementation and results sections.

2.2. Stage 2: Executing LPAA for assigning priority to the loads

This is the second stage in which predicted T values from stage 1 are used to run LPAA and generate LPT. LPAA fundamentally functions based on a DSM technique namely load priority technique. DSM Techniques are discovered in early 1970's [39] by Gellings, CW. In the last five decades, numerous DSM techniques are reported, out of all of them load priority technique is widely used in load scheduling schemes. As per [40], by Xin, L. et al., LPAA is a real time load priority algorithm proposed for the household loads which assigns the priority to all the loads using T predictions and availability of renewable sources. Similarly, we have presented an extended LPAA in [36], to assign priority for heating, cooling, lighting load and water heater loads separately. The work presented in this paper enhances the accuracy of the LPAA by predicting the T needed using ML algorithms. The unique feature of the LPAA presented in this paper is that it generates LPT for any given date and hour, because the ML algorithms set to predict T are trained with date and time stamp variables.

2.3. Stage 3: Applying peak clipping DSM Technique to estimate P_{Th}

Following the execution of LPAA, it is essential to estimate P_{Th} using LPT obtained in stage 2 so as to implement LMA. Detailed expression of P_{Th} is presented in

equation (3), where P_T is predicted by ML algorithms (from stage 1) and least priority (P_{LS}), lower priority (P_{LR}) loads are identified by perusing LPT (from stage 2).

P_{Th} is directly forecasted using feed forward NN model in [18], with fixed load priority assignment. The work presented in this paper estimates P_{Th} using the best fit ML algorithms-based T and P_T predictions. Thus, the LPAA results in accurate LPT with all the priorities of the loads, and this plays a key role in estimating P_{Th} and subsequently in LMA execution. Experimental values of T, LPT, P_T and P_{Th} are presented for a specific date and time in results section.

2.4. Stage 4: LMA for Data driven HEMS towards DG integration

This implementation assumes that domestic load centre considered is installed with two DGs, out of which DG1 is conventional and DG2 is renewable. LMA initiates its working with LPT and P_{Th} obtained from Stage 2 and Stage 3, respectively. LMA presented in [18] is now driven by ML algorithms-based prediction results to achieve better performance and the same is described in Fig. 4. Steps of LMA are presented in the subsequent paragraphs.

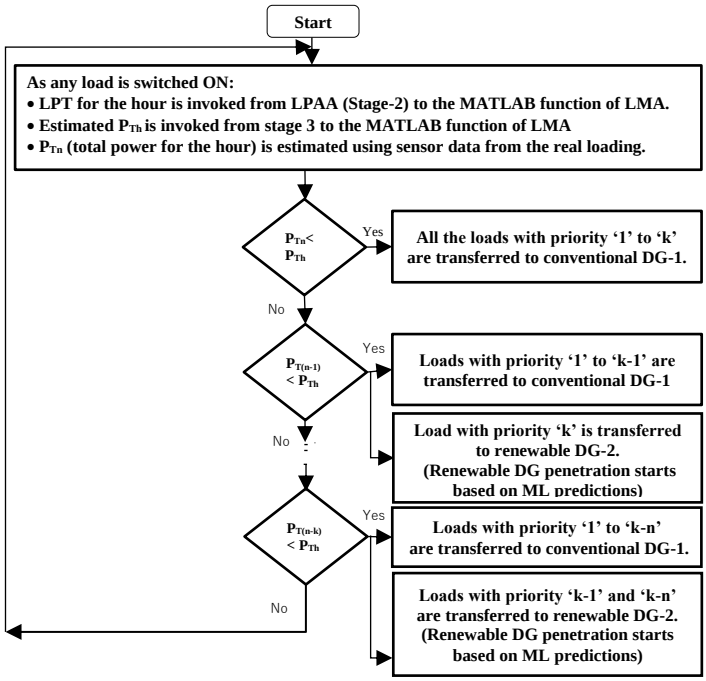


Fig. 4. LMA flowchart.

Step-1: Whenever user turns ON any load at the load centre, 'P_{Tn}' (Total power of all 'n' loads at the domestic load centre - read from sensory system) is compared with estimated P_{Th} which is based on ML algorithm predictions. If P_{Tn} is less than P_{Th}, then the loads with priority from '1' to 'k' (loads of all priorities) are transferred to conventional DG1, else control moves to the next step.

Step-2: '$P_{T(n-1)}$' (total power excluding load of kth priority) is compared with 'P_{Th}'. If '$P_{T(n-1)}$' is less than 'P_{Th}', then loads of priorities '1' to '(k-1)' are transferred to

conventional DG1 and the load of 'k' priority is transferred to renewable DG2, else control moves to the next step.

Step-3: '$P_{T(n-2)}$' (total power excluding loads of 'k' and '(k-1)' priority) is compared with 'P_{Th}'. If '$P_{T(n-2)}$' is less than 'P_{Th}', loads of priorities '1' to '(k-2)' are transferred to conventional DG1 and loads of priorities 'k' and '(k-1)' are transferred to renewable DG2, else control moves to next step.

Therefore, LMA algorithm proceeds to exclude loads of priorities 'k,' '(k-1)', '(k-2)', '(k-3)', '(k-4)' in each step and transfers the least, lower and medium priority loads to renewable DG2. And eventually the load of priority '1' is transferred to conventional DG1. Thus, LMA accommodates DG integration by increasing the penetration of renewable DG in distribution network, which eventually reduces burden on conventional DG and causes significant reduction in carbon emissions. The equation of the carbon emissions per kW is presented in Eq. 5 and the expression of total load demand is presented in Eq. 4.

$$P_{Tn} = \sum_{i=1,k=1}^{n,m} P_{ik} \tag{4}$$

$$\mathrm{CO}_2\ Emissions = 0.0428 * P_{Tn} \tag{5}$$

where, 'n' = number of appliances/loads at the load centre (connected to HEMS), and 'k' = order of priorities assigned for the appliances/ loads at the load centre.

3. Simulation of HEMS to achieve DG integration: Deployment

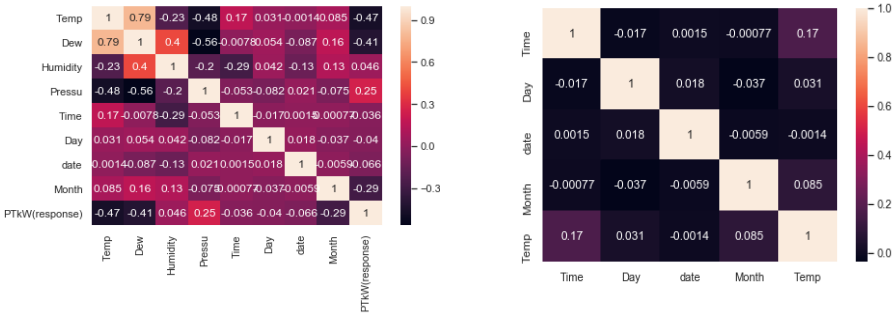
Simulation of the Big Data driven HEMS to achieve DG integration is carried out in MATLAB-SIMULINK environment. As per the methodology, HEMS implementation is done in four stages, and they are detailed in the subsequent sections.

3.1. Stage 1: Evaluation of big data collected using heat maps and training ML algorithms to predict 'T' and 'PT'

Before training ML algorithms, load, weather, time stamp Big Data acquired is first investigated using heat maps to analyse the relation between various predictors and the response variable as well as for understanding the correlation between all predictor variables. From the heat map represented in Fig. 5(a), it can be inferred that the response variable P_T has positive correlation with pressure whereas predictors like temperature, dew point temperature and humidity have positive correlation between them. Further, it can be observed that P_T has negative correlation with temperature, dew point temperature and month whereas pressure, temperature and dew point temperature have a negative correlation between them. The heat map shown in Fig. 5(b) describes the correlation between the predictor and response variables used in predicting *T*. According to heat map, the predictor variables hour of the day, day of the week, date of the month and month of the year have negative correlation with response variable temperature (*T*). Despite the negative correlation between the variables, the said predictors are still used to train advanced ML algorithms so as to predict *T* at any given date and time.

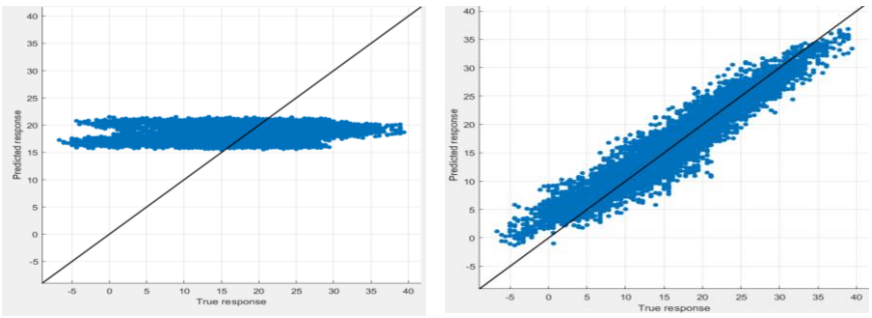
Firstly, four ML algorithms such as LR, EBT, Fine Gaussian SVR (FG-SVR) and Squared Exponential GPR (SE-GPR) are trained to predict *T* using Regression Learner Tool box of MATLAB version 2018 b. The said ML algorithms are trained with 4 different predictor variables such as 'day of the week', 'hour of the day', 'date of the month' and 'month of the year' and one response variable *T*. Amongst four ML algorithms trained, GPR ML algorithm with kernel function Squared

Exponential is observed with lowest Root Mean Square Error (RMSE), Mean Square Error, Mean Average Error and highest Squared - R (R^2) values. The training results concerning ML algorithms are presented in Table 1. The regression plots obtained for all the ML algorithms in predicting *T* are presented in Figs. 6(a)-(d), which show that SE-GPR is much superior with about 99% of the samples aligning to the regression curve during training. Therefore, a SIMULINK prediction model to predict *T* is developed using SE-GPR ML algorithm for the simulation purposes. This prediction model is simulated for any given input time and date variables and used to predict *T* which will be useful in implementing LPAA in the HEMS SIMULINK model.

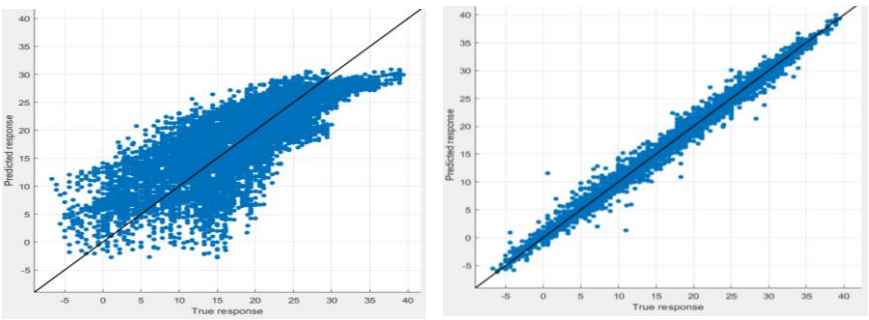


(a) Heat map for P_T prediction. **(b) Heat map for 'T' prediction.**

Fig. 5. Heat maps used for P_T and *T* prediction.



(a) Regression plot of LR. **(b) Regression plot of EBT.**



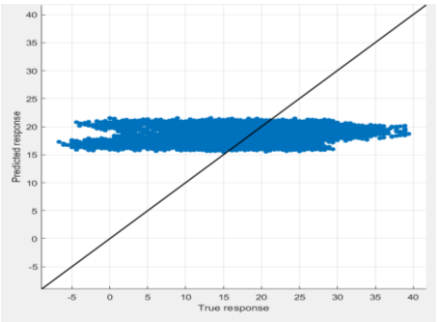
(c) Regression plot of FG-SVM. **(d) Regression plot of SE-GPR.**

Fig. 6. Regression plots of ML algorithms for *T* prediction.

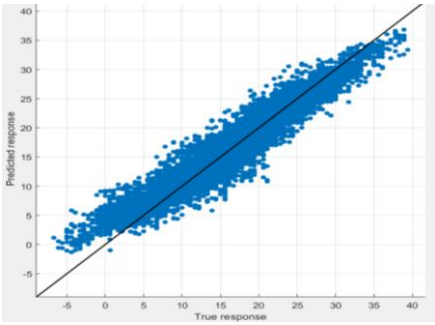
Table 1. Performance parameters of ML algorithms in predicting *T*.

Name of the ML Algorithm	Performance Parameters			
	RMSE	**MSE**	**MAE**	R^2
LR	8.255	68.147	6.767	0.03
EBT	2.356	5.553	1.804	0.92
FG-SVR	1.411	1.993	0.99	0.97
SE-GPR	0.870	0.756	0.601	0.99

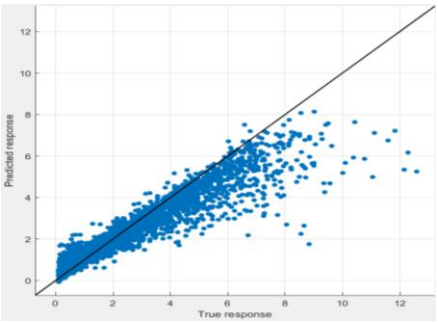
After predicting *T*, the prediction of P_T is also carried out and for these four different ML algorithms such as LR, EBT, FG-SVR and Rotational Quadratic GPR (RQ-GPR) are trained. As already mentioned in section 2.1.2, eight predictor variables are used to train the said ML algorithms. Performance parameters such as RMSE, MSE, MAE and R^2 are presented in Table 2. Out of all the ML algorithms GPR ML algorithm with kernel function Rotational Quadratic (RQ) is observed to have lowest RMSE, MSE, MAE and highest R^2. And the regression plots presented in Figs. 7(a)-(d) also reveal that RQ-GPR ML algorithm exhibits about 98% of the samples aligning the regression curve, which is the finest regression amongst all. Therefore, P_T prediction model for predicting hourly P_T is developed using RQ-GPR ML algorithm for the simulation purposes and it is indicated in P_{Th} SIMULINK model shown in Fig. 9.



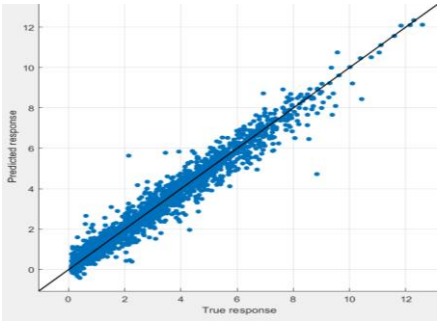
(a) Regression plot of LR.



(b) Regression plot of EBT.



(c) Regression plot of FG-SVM.



(d) Regression plot of RQ-GPR.

Fig. 7. Regression plots of ML algorithms for P_T prediction.

Table 2. Performance parameters of ML algorithms in predicting P_T

Name of the ML Algorithm	Performance Parameters			
	RMSE	**MSE**	**MAE**	R^2
LR	1.4363	2.063	1.0463	0.3
EBT	0.8401	0.7058	0.5227	0.79
FG-SVR	0.5930	0.3174	0.2973	0.88
RQ- GPR	0.27046	0.0731	0.1516	0.98

This prediction model is simulated for any given set of weather, date and time stamp input variables to predict P_T which is essential in estimating P_{Th}.

3.2. Stage-2: SIMULINK model to execute LPAA

LPAA is an algorithm implemented to assign the hourly priorities for the loads at the domestic load centre. Theoretical background of LPAA is already given in section 2.2. A SIMULINK model is built to run LPAA using multiple MATLAB functions and SE-GPR ML based *T* prediction model, and this model is presented in Fig. 8. As it is indicated in Fig. 8, LPAA model takes two inputs and gives one output. The inputs are *T* (which is obtained from SE-GPR ML based prediction model), the 'time vector' (generated using MATH code), and the output is LPT with time vector. LPAA model is developed with a MATLAB function which processes the inputs according to the steps of the algorithm [36] and produces an output LPT. LPT resulted from LPAA SIMULINK model is used to estimate P_{Th} and implement LMA which are carried out in stages 3 and 4, respectively. Predicted *T* and the output LPT of LPAA model are indicated in Fig. 8.

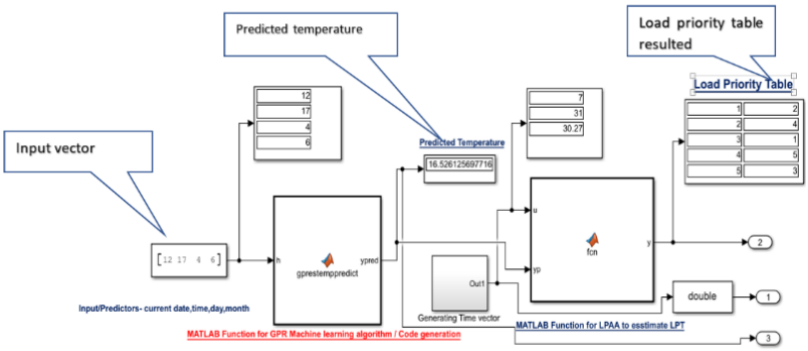


Fig. 8. LPAA Model built to generate hourly LPT.

3.3. Stage 3: Estimation of P_{Th} using P_T SIMULINK model

As per Eq. (3), prediction of P_T plays an important role in estimating P_{Th}. P_T prediction SIMULINK model using RQ- GPR ML algorithm developed in stage 1 is simulated for the given set of weather, date and time stamp input variables. To be specific, *T*, 'dew point temperature,' 'humidity,' 'pressure,' 'hour of the day,' 'day of the week,' 'date of the month,' 'month of the year' of the current date are supplied to the P_T prediction model in order to predict hourly P_T. The predicted P_T is forwarded to P_{Th} model which is developed with seven inputs and three outputs. The seven inputs given to the P_{Th} MATLAB function are, predicted P_T, LPT vector

generated from LPAA (stage 2), and five individual loads' real time load power consumption data sensed by sensors installed on the HEMS hardware setup. The MATLAB code embedded in P_{Th} MATLAB function processes all the seven inputs and estimates P_{Th} as per the equation (3). The details of the P_{Th} model with P_T predicted are presented in Fig. 9.

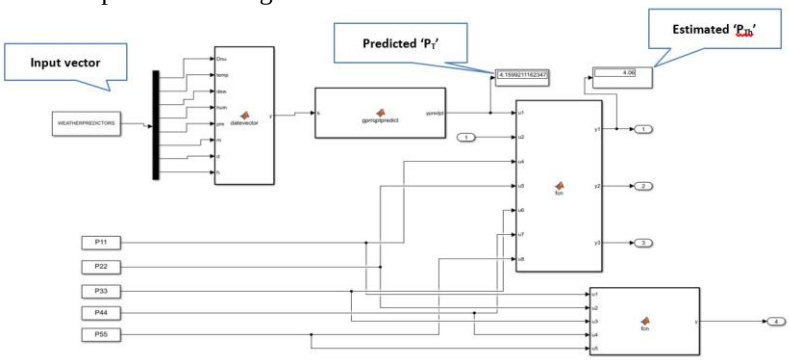


Fig. 9. 'PTh' estimation model.

3.4. Stage 4: LMA execution - LMA MATLAB function

Estimated LPT and P_{Th} resulted from stage 2 and 3 respectively are forwarded to a MATLAB function where LMA is executed to accommodate the AI based DG integration. This MATLAB function is developed with eight inputs (u1 - u8) and fifteen outputs (y1 - y15) in which a MATLAB code to execute LMA is embedded. The eight inputs are, LPT vector (u6) resulted from LPAA (stage 2), the time vector (u7) and five individual loads (u1-u5) real time power consumption data sensed by sensory units installed on the hardware and read by ARDUINO read ports. The MATLAB code embedded in MATLAB function runs the LMA with respect to the time of the day and gives out fifteen outputs. The outputs are, five output signals (y1 - y5) to single channel relays, four output signals (y6 - y9) to double channel relays, five (y10 - y14) are to workspace to save the real time individual load power consumption data in the workspace for further use, and last 15th output (y15) is the total power consumption to be stored in workspace. ANALOG READ and DIGITAL WRITE ports present in the ARDUINO supporting package are used to establish communication between MATLAB-SIMULINK and ARDUINO Mega 2560 and the same are presented in Fig. 10.

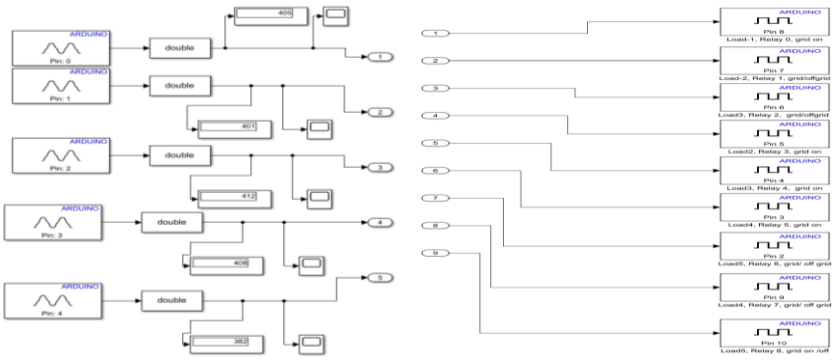


Fig. 10. Arduino read and write ports.

3.5. HEMS SIMULINK model to achieve DG integration

SIMULINK model of Big Data driven HEMS is built by interconnecting subsystems (SIMULINK models and MATLAB functions) such as P_T prediction model, T prediction model, LPAA model, P_{Th} estimation model and LMA model. The SIMULINK models and the MATLAB functions used in building SIMULINK model of HEMS are already described in earlier sections. Big Data HEMS SIMULINK model built is simulated in three modes. Firstly, in normal mode to check for any simulation errors and once it is cleared from the possible errors, it is then run in external model. In external mode, the model runs on both the Desktop and on the Hardware setup. In this mode, simulation and experimental results for the given date and time can be recorded. To be specific, predicted P_T, predicted T, generated LPT and estimated P_{Th} for the given date and time can be recorded. And also, effectiveness of LMA execution in Big Data driven HEMS transferring least priority loads to renewable DG and highest priority loads to conventional DG can be observed. Detailed description of HEMS SIMULINK model with associated MATLAB functions and blocks is presented in Fig. 11.

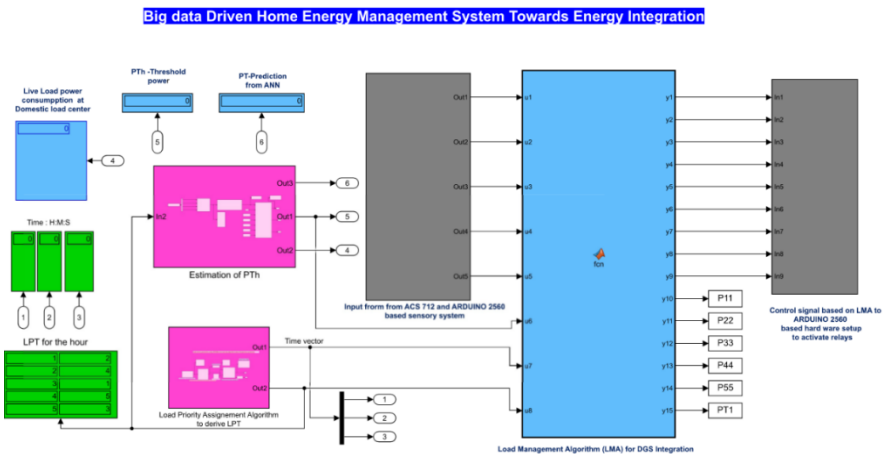


Fig. 11. Big Data driven HEMS SIMULINK model.

4. Experimental Setup and Results

Hardware of the proposed Big Data driven HEMS is designed and fabricated to perform ML based DG integration. Detailed circuit diagram of Big Data driven HEMS showing an ARDUINO 2560 micro controller unit sensing the currents flowing through five different loads and operating relays is presented in Fig. 12.

DG1 is considered to be conventional whereas DG2 is renewable DG. The HEMS designed is connected with two RCDs with supply lines, and five MCBs with all the loads to safeguard entire system from over currents. Both RCDs and MCBs used are rated for 20A. A relay board with nine relays is designed to connect the loads and supply and switch on / off the loads as per the output command from ARDUINO. To be specific, five single channel relays are used to turn the loads on/off and remaining four double channel relays are used to connect DGs to the loads. Relay board is also equipped with a relay driver to make them compatible with ARDUINO. Hardware setup of Big Data driven HEMS fabricated with all the components and connections between them is presented in Fig. 13. SIMULINK

model developed and presented in previous section is set for code generation and this code is set to run on the target hardware ARDUINO 2560.

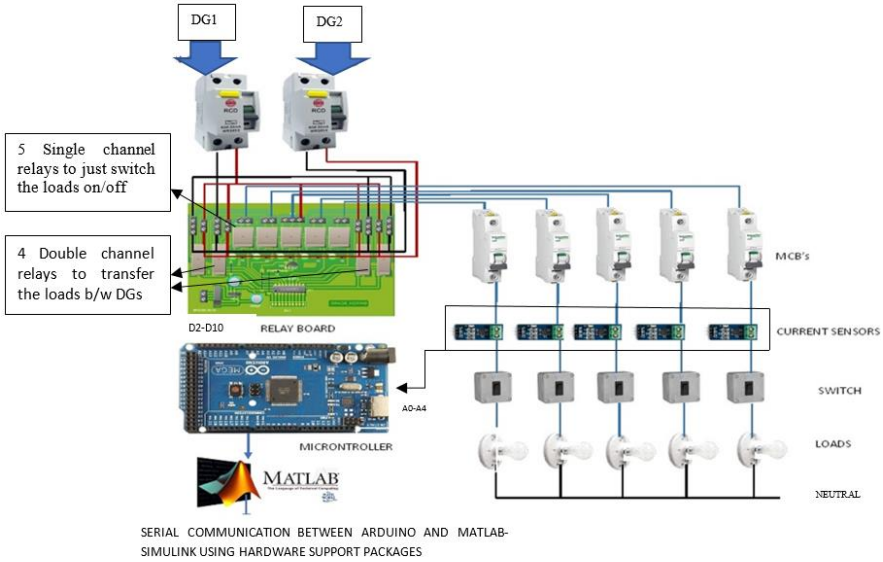


Fig. 12. Circuit diagram of the big data driven HEMS.

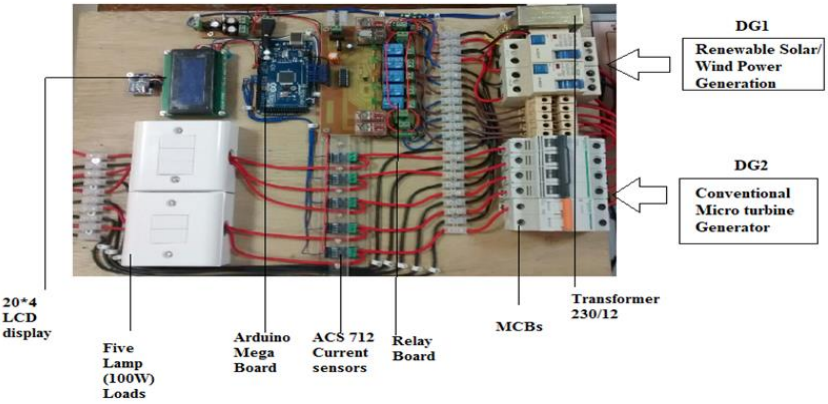


Fig. 13. Hardware Setup of the data driven HEMS.

The experimental results of DG integration observed on 17th December, at 6 am, 2019 are considered for the discussion in this section. To begin with, input predictors such as month, date, day and time for the current time stamp are fed to the SE-GPR ML based T prediction model to predict T. Simultaneously, input predictors temperature, dew point, humidity, pressure, month, date, day and hour for the current date are fed to RQ-GPR ML based P_T prediction model to predict P_T. With the given inputs, the HEMS SIMULINK model is run in external mode and predicted T and P_T from T and P_T prediction models, respectively. Subsequently, LPT is generated by the LPAA model. And P_{Th} is estimated from its MATLAB function driven by P_T prediction model. Eventually, HEMS Model executes LMA using the LPT and P_{Th} estimated, which further triggers the relays

to transfer the loads between the DGs. Detailed inputs/ predictor variables for T and P_T prediction are presented in tables 3 and 4. As the HEMS SIMULINK model runs on the target hardware, T, P_T are predicted, as a result LPT and P_{Th} are generated, and these values are presented in table 5.

Table 3. Input variables for the SE-GPR ML based SIMULINK prediction model.

Calendar date	Day	Date	Month	Time in hours
17th December	Wednesday (4th day of the week)	17	12	6

Table 4. Input variables for the RQ-GPR ML based SIMULINK prediction model.

Day	Temperature in ^{O}C	Dew point in ^{O}C	Humidity in %	Pressure in Inches	Month	Date	Time in hours
4	14.4	12.8	0.9	1009.1	12	17	6

Table 5. Outputs of HEMS with SE-GPR and RQ-GPR ML algorithms: LPT, P_T, P_{Th} for a specific date.

T predicted in ^{O}C For the hour	Hourly LPT generated	Hourly P_T predicted in kW	Hourly P_{Th} estimated in kW	Hourly reduction in CO_2 emissions in kg
16.5	[1 2; 2 4;3 1; 4 5;5 3]	4.159	4.059	0.0428

In addition, CO_2 reductions achieved due to peak clipping is also presented in Table 5. Furthermore, LMA executes, and DG integration is manifested, the status of the load priorities, relays and the loads for the specific hour are presented in Table 6.

From Fig. 14, it can be witnessed that, load 1 with priority 2 is turned off but connected to DG-1 (conventional), load 2 with priority 4 is on and connected to DG-1 (conventional), load 3 with priority 1 (highest priority) is on and connected to DG-1 (conventional), load 5 with priority 3 is on and connected to DG-1 (conventional) and lastly, load 4 assigned with least priority 5 (renewable) is turned on and connected to DG-2.

Thus, it is proved that the DGs are integrated to feed the load demand of the domestic load centre with loads of least priority connected to renewable DG and others connected to conventional DG. This concludes the experimental demonstration of the proposed AI based DG integration using the data driven HEMS with the support of ML algorithms. The proposed DG integration increases the penetration of DGs in domestic load network and reduces CO_2 emissions. The hourly CO_2 emissions are calculated using Eq. (5) and presented in Table 5.

In addition to the execution of the load transfer, Big Data trained HEMS can predict the LPT, P_T, P_{Th} and CO_2 for any given future/past date. To validate this, date, time and weather parameters concerning a date back in fall 2013 are taken as inputs to the SIMULINK model of the HEMS, and T, P_T, LPT, P_{Th} and reduction in CO_2 are predicted prior. Thus, the load priority (LPT) and essential power (P_{Th}) for every hour are learned with the aid of supervised learning of the ML algorithms.

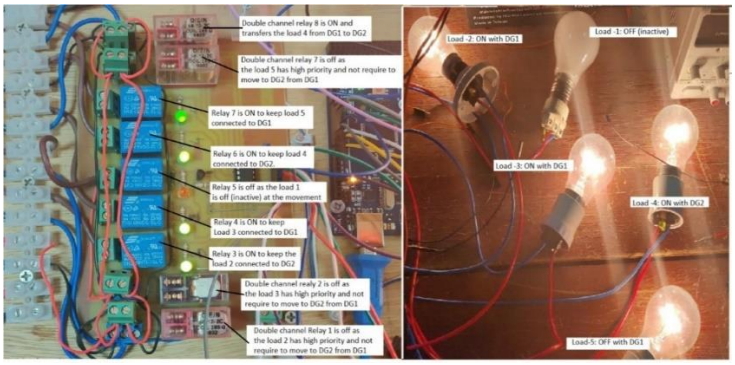


Fig. 14. Loads transfer between the DGs.

Table 6. Execution of the proposed integration of DGs.

Load description	Priority	DG integration
Load-1	2	Off (Inactive)
Load-2	4	ON and connected to DG-1
Load-3	1	ON and connected to DG-1
Load-4	5	ON and connected to DG-2
Load-5	3	ON and connected to DG-1

The details of the predictions for different hours of date 14-09-2013 are presented in Table 7. This is a novel aspect presented in this paper which opens up new initiatives in planning and management of DG integration.

Table 7. Prediction of load priority, P_T, P_{Th} and CO_2 emissions of date 14-09-2013.

	Input values to predict power consumption for a domestic load centre.									Predicted terms of DG integration for the given day		
Fall 9/14/2013	**Day**	**Temperature in ^oC**	**Dew point in ^oC**	**Humidity in %**	**Pressure in Inches**	**Month**	**Date**	**Time in Hrs**	**(Actual power from Big Data) P_T**	**Predicted power (P_T)**	**P_{Th} = P_T-P_{LS}**	**CO_2 Reduction in kg**
6:00 AM	1	17.8	14.4	0.8	1022.9	9	14	6	0.59	0.453	0.09	0.0155
11:00 AM	1	24.4	16.7	0.62	1024	9	14	11	0.75	0.714	0.25	0.0198
3:00 PM	1	28.3	15.6	0.46	1021.8	9	14	15	1.05	1.159	0.55	0.0260
6:00 PM	1	27.2	15.6	0.46	1021.8	9	14	18	1.39	1.165	0.89	0.0117
11:00 PM	1	22.8	16.7	0.68	1022.4	9	14	23	0.87	0.661	0.37	0.0124

5. Conclusions

The AI based DG integration is implemented and investigated using Big Data driven HEMS with ML algorithms. Load priority assignment using SE-GPR ML algorithm-based T prediction and peak clipping power P_{Th} estimation using RQ-GPR ML algorithm are utilized to integrate DGs available at the load centre.

It is investigated that, SE-GPR and RQ-GPR ML algorithms trained to predict T and P_T, outperform all other ML algorithms with lowest RMSE of 0.870 and 0.2704, respectively. The HEMS SIMULINK model driven by these ML algorithms successfully demonstrated AI based DG integration.

Experimental results such as predicted T, LPT, predicted P_T, estimated P_{Th}, load transfer observed on a specific date and hour by running HEMS SIMULINK model on target hardware are presented to witness the ability of the proposed HEMS. Based on the LPT and P_{Th} observed, least priority loads are transferred to renewable DG2 while the other priority loads are transferred to conventional DG1.

Eventually, CO_2 emissions for that specific hour are observed to have a decay of 24% after the implementation of AI based DG integration, and it is expected to be dynamic every hour based on the ML predictions. In addition, the ability of HEMS to predict load priority, peak clipping power P_{Th}, total power consumption P_T, DG requirement in kW for any given date and time stamp is witnessed in Table 7. However, proposed Big Data driven HEMS does not take the comfort and choice of the user in consideration while describing the priority of the loads for any date and hour, because the HEMS solely learns from the historical data.

6. Future Scope

Incorporating human presence sensor in the existing Big Data driven HEMS to capture and accumulate data sets concerning human presence in the domestic load centre enhances the accuracy at which the load priority is assigned by LPAA. To add to it, sensing renewable DG availability and acquiring DG power availability data sets allows the ML predictions to run LPAA and assign load priority accurately in order to accommodate AI based DG integration with a higher accuracy.

Nomenclatures

CO_2	Carbon dioxide, kg
i	Number of the load appliance
k	Order of load priority
n	Total number of load appliances at the domestic load centre
P_{ik}	i^{th} Load appliance of kth priority, kW
P_{LR}	Lower Priority Load power, kW
P_{LS}	Least Priority Load power, kW
P_T	Total power consumption, kW
P_{Tf}	Forecasted total power, kw
P_{Th}	Threshold power, kW
P_{Tn}	Real time power consumption, kW
P_{Tn}	Real time total power, kW
R^2	Squared R
T	Temperature, ^{O}C

Abbreviations

AI	Artificial Intelligence
ANN	Artificial Neural Networks
DG	Distributed Generation
DSM	Demand Side Management
EBT	Ensemble Bagged Algorithm
EIA	Energy Information Authority
FG-SVR	Fine Gaussian Space Vector Regression
GA	Genetic Algorithm

HEMS	Home Energy Management System
LMA	Load Management Algorithm
LPAA	Load Priority Assignment Algorithm
LPT	Load Priority Table
LR	Linear Regression
LSA	Lightning Search Algorithm
MAE	Mean Average Error
MCCU	Main Command Control Unit
MILP	Mixed Integer Linear Programming
ML	Machine Learning
MSE	Mean Square Error
PRGBNNs	Polak-Ribiére Gradient Back Propagation Networks
RL	Reinforced Learning
RMSE	Root Mean Square Error
RQ-GPR	Rotational Quadratic Gaussian Process Regression
SE-GPR	Squared Exponential Gaussian Process Regression

References

1. U.S Energy Information Administration, Electricpowerannual 2010, Table 9.4. Retrived January 5, 2020, from https://www.eia.gov/electricity/annual.
2. Puttgen, H.B.; Macgregor, P.R.; and Lambert, F.C. (2003). Distributed generation: Semantic hype or the dawn of a new era?. *IEEE Power Energy Magazine*, 3(1), 22-29.
3. T. Ackermann, V.; and Knyazkin, B. (2002). Interaction between distributed generation and the distribution network: Operation aspects. *Proceedings of IEEE/PES Transmission and Distribution Conference and Exhibition: Asia Pacific*, Yokohama, Japan, 1357-1363.
4. Perera, K.S.; Zeyar, A.; and Woon, L.W. (2014). Machine learning techniques for supporting renewable energy generation and integration: A survey. *Proceedings of* 2nd *International Conference on Data Analytics for Renewable Energy Integration*, Cham (ZG), Switzerland, 81-96.
5. Boehm, M.; Dennecker, L.; Doms, A.; Dovgan, E.; Filipic, B.; Phischer, U.; Lehner, W.; Pederson, T.B.; Pitarch, Y.; Siksnys, L.; and Tusar, T. (2012). Data management in the MIRABEL smart grid system. *EDBT-ICDT '*12*: Proceedings of the* 2012 *Joint EDBT/ICDT Workshops* , Germany, 95-102.
6. Barbato, A.; Capone, A.; Rodolfi, M.; and Tagliaferri, D. (2011). Forecasting the usage of household appliances through power meter sensors for demand management in the smart grid. *Proceedings of the* 2011 *IEEE International Conference on Smart Grid Communications*, Belgium, 404-409.
7. Reinhardt, A.; Christin, D.; and Kanhere, S.S. (2013). Predicting the power consumption of electric appliances through time series pattern matching. *BuildSys'*13*: Proceedings of the* 5*th ACM Workshop on Embedded Systems for Energy-Efficient Buildings*, New York, US, 1-2.
8. Ulbricht, R.; Fischer, U.; Lehner, W.; and Donker, H. (2013). First steps towards a systematical optimized strategy for solar energy supply forecasting. *Proceedings of the* 2013 *ECML/PKDD International Workshop on Data Analytics for Renewable Energy Integration* (*DARE*), Macedonia, 14-25.

9. Kaulakienė, D.; Šikšnys, L.; and Pitarch, Y. (2013). Towards the automated extraction of flexibilities from electricity time series. *EDBT '*13*: Proceedings of the Joint EDBT/ICDT* 2013 *Workshops*, Germany, 267-272.
10. Medrano, M.; Brouwer, J.; Mc Donell, V.; Mauzey, J.; and Samuelsen, S. (2008). Integration of distributed generation systems into generic types of commercial buildings in California. *Energy and Buildings*, 40(4), 537-548.
11. Dyson, M.E.H.; Borgeson, S.D.; Tabone, M.D.; and Callaway, D.S. (2014). Using smart meter data to estimate demand response potential, with application to solar energy integration. *Energy Policy*, 73, 607-619.
12. Al-Ali, A.R.; El-Hag, A.; Bahadiri, M.; Harbaji, M.; and El Haj, Y.A. (2011). Smart home renewable energy management system. *Energy Procedia*, 12, 120-126.
13. Macedo, M.N.Q.; Galo, J.J.M.; de Almeida, L.A.L.; and Lima, A.C. de C. (2015). Demand side management using artificial neural networks in a smart grid environment. *Renewable and Sustainable Energy Rev*iews, 41, 128-133.
14. Jaramillo, L.B.; and Weidlich, A. (2016). Optimal micro grid scheduling with peak load reduction involving an electrolyser and flexible loads. *Applied Energy*, 169, 857-865.
15. Elkazaz, M.H.; Hoballah, A.A.; and Azmy, A.M. (2016). Operation optimization of distributed generation using artificial intelligence techniques. *Ain Shams Engineering Journal*, 7(2), 855-866.
16. Javaid, N.; Ullah, I.; Akbar, M.; Iqbal, Z.; Khan, F.A.; Alrajeh, N.; and Alabed, M.S. (2017). An Intelligent load management system with renewable energy integration for smart homes. *IEEE Access*, 5, 13587-13600.
17. Melhem, F.Y.; Moubayed, N.; and Grunder, O. (2014). Residential energy management in smart grid considering renewable energy sources and vehicle-to-grid integration. *Proceedings of the* 2016 *IEEE Electrical Power and Energy Conference* (*EPEC*), Ottawa, ON, Canada, 1-6.
18. Raju, M.P.; and Laxmi, A.J. (2017). A novel load management algorithm for emu by implementing demand side management techniques using ANN. *Proceeding of* 2017 *International Conference on Electrical and Computing Technologies and Applications* (*ICECTA*), Ras Al Khaimah, UAE, 1-6.
19. Dick, A.J.; Allera, S.V.; and Horsburgh, A.C. (1990). EMU - the energy management unit. *Proceeding of sixth International Conference on Metering Apparatus and Tariffs for Electricity Supply*, Manchester, 177-182.
20. Farhangi, H. (2010). The path of the smart grid. *IEEE Power Energy Magazine*, 8(1), 18-28.
21. Vojdani, A. (2008). Smart integration. *IEEE Power Energy Magazine*, 6(6), 71-79.
22. Wang, P.; Huang, J.Y.; Ding, Y.; Loh, P.; and Goel, L. (2010). Demand side load management of smart grids using intelligent trading/metering/billing system. *Proceedings of* 2011 *IEEE Trondheim PowerTech*, Trondheim, Norway, 1-6.
23. Das, S.K.; Cook, D.J.; Bhattacharya, A.; Heierman, E.O.; and Lin, T.-Y. (2002). The role of prediction algorithms in the My Home smart home architecture. *IEEE Wireless Communications*, 9(6), 77-84.
24. Hu, Q.; and Li, F. (2013). Hardware design of smart home energy management system with dynamic price response. *IEEE Transactions on Smart Grid*, 4, 1878-1887.

25. Pipattanasomporn, M.; Kuzlu, M.; and Rahman, S. (2012). An algorithm for intelligent HEM and demand response analysis. *IEEE Transactions on Smart Grid*, 3(4), 2166-2173.
26. Mahapatra, C.; Moharana, A.K.; and Leung, V.C.M. (2017). Energy management in smart cities based on internet of things: peak demand reduction and energy savings. *Sensors*, 17(12): 2812.
27. Logenthiran, T.; Srinivasan, D.; and Shun, T.Z. (2012). Demand side management in smart grid using heuristic optimization. *IEEE Transactions on Smart Grid*, 3(3), 1244-1252.
28. Angelis, F.; Boaro, M.; Fuselli, D.; Squartini, S.; Piazza, F.; and Wei, Q. (2013). Optimal home energy management under dynamic electrical and thermal constraints. *IEEE Transactions Industrial Informatics*, 9(3), 1518-1527.
29. Di Santo, K.G.; Santo, S.G.; Monaro, R.M.; and Saidel, M.A. (2018). Active demand side management for households in smart grids using optimization and artificial intelligence. *Measurement*, 115, 152-161.
30. Yuce, B.; Rezgui, Y.; and Mourshed, M. (2016). ANN-GA smart appliance scheduling for optimized energy management in the domestic sector. *Energy and Buildings*, 111, 311-325.
31. Ahmed, M.S.; Mohamed, A.; Homod, R.; and Shareef, H. (2016). Hybrid LSA-ANN based home energy management scheduling controller for residential demand response strategy. *Energies*, 9(9): 716.
32. Kazmi, S.; Javaid, N.; Mughal, M.J.; Akbar, M.; Ahmed, S.; and Alrajeh, N. (2019). Towards optimization of metaheuristic algorithms for IOT enabled smart homes targeting balanced demand and supply of energy. *IEEE Access*, 7, 24267-24281.
33. Ali, H.; Hussain, A.; Bui, V.-H.; Jeon, J.; and Kim, H.-M. (2019). Welfare maximization-based distributed demand response for islanded multi-microgrid networks using diffusion strategy. *Energies*, 12(19): 3701.
34. Atef, S.; and Eltawil, A. (2019). A comparative study using deep learning and support vector regression for electricity price forecasting in smart grids. *Proceedings of the* 2019 *IEEE International Conference on Industrial Engineering and Applications* (*ICIEA*), Tokyo, Japan, 603-607.
35. Lu, R.; Hong, S.H.; and Yu, M. (2019). Demand response for home energy management using reinforcement learning and artificial neural network. *IEEE Transactions on Smart Grid*, 10(6), 6629-6639.
36. Raju, M.P.; and Laxmi, A.J. (2019). Home energy management system for a domestic load centre using artificial neural networks towards energy integration. *International Journal of Recent Technology and Engineering*, 8(4), 12548-557.
37. NREL's Building America house protocols. (2018). Retrieved January 7, 2018, from https://openei.org/doe-opendata/dataset/.
38. Historical Weather Data (2018). Birmingham Municipality, City of Alabama, US. Retrieved April 5, 2018, from https://www.wunderground.com.
39. Gellings, C.W. (1981). Power/energy: demand-side load management: the rising cost of peak-demand power means that utilities must encourage customers to manage power usage. *IEEE Spectrum*, 18(12), 49-52.
40. Liu, X.; Ivanescu, L.; Kang, R.; and Maier, M. (2012). Real-time household load priority scheduling algorithm based on prediction of renewable source availability. *IEEE Transactions on Consumer Electronics*, 58(2), 318-326.

Appendix A

MATLAB code for SE-GPR ML algorithm based 'T' prediction

```
inputTable = array2table(trainingData, 'VariableNames', {'column_1', 'column_2',
'column_3', 'column_4', 'column_5'});
 predictorNames = {'Date', 'time', 'day', 'month'};
predictors = inputTable(:, predictorNames);
response = inputTable.column_2;
isCategoricalPredictor = [false, false, false, false];
% Train a regression model
% This code specifies all the model options and trains the model.
regressionGP = fitrgp(predictors,response, 'BasisFunction', constant','KernelFunction',
'squaredexponential', 'Standardize', true);
% Create the result struct with predict function
predictorExtractionFcn = @(x) array2table(x, 'VariableNames', predictorNames);
gpPredictFcn = @(x) predict(regressionGP, x);
trainedModel.predictFcn = @(x) gpPredictFcn(predictorExtractionFcn(x));
% Add additional fields to the result struct
trainedModel.RegressionGP = regressionGP;
% Extract predictors and response
inputTable = array2table(trainingData, 'VariableNames', {'Date', 'time', 'day', 'month'});
predictorNames = {'Date', 'time', 'day', 'month'};
predictors = inputTable(:, predictorNames);
response = inputTable.column_2;
isCategoricalPredictor = [false, false, false, false];
% Compute validation RMSE
validationRMSE = sqrt(kfoldLoss(partitionedModel, 'LossFun', 'mse'));
```

MATLAB code for RQ-GPR ML algorithm based 'P_T' prediction

```
inputTable = Yearweatherdata1;
predictorNames = {'Day', 'Temp', 'Dew', 'Humidity', 'Pressu', 'Month', 'date', 'Time'};
predictors = inputTable(:, predictorNames);
response = inputTable.PTkW;
isCategoricalPredictor = [false, false, false, false, false, false, false, false];
% Train a regression model
regressionGP = fitrgp( predictors, response, 'BasisFunction', 'constant',  'KernelFunction',
'rationalquadratic', 'Standardize', true);
 % Create the result struct with predict function
predictorExtractionFcn = @(t) t(:, predictorNames);
gpPredictFcn = @(x) predict(regressionGP, x);
trainedModel.predictFcn = @(x) gpPredictFcn(predictorExtractionFcn(x));
 % Add additional fields to the result struct
trainedModel.RequiredVariables = {'Day', 'Temp', 'Dew', 'Humidity', 'Pressu', 'Month',
'date', 'Time'};
trainedModel.RegressionGP = regressionGP;
% Extract predictors and response
inputTable = trainingData;
predictorNames = {'Day', 'Temp', 'Dew', 'Humidity', 'Pressu', 'Month', 'date', 'Time'};
predictors = inputTable(:, predictorNames);
response = inputTable.PTkW;
isCategoricalPredictor = [false, false, false, false, false, false, false, false];
validationPredictions = kfoldPredict(partitionedModel);
 % Compute validation RMSE
validationRMSE = sqrt(kfoldLoss(partitionedModel, 'LossFun', 'mse'));
```