Predictive-based Models for Efficient Energy Management in Smart Buildings

Osama Mohammed[1], Marwa Ibrahim[2], Abdalrahman Fatikhan Ataalla[3,*]

[1]Middle Technical University, Baghdad, Iraq
[2]Islamic University of Lebanon, Lebanon
[3]Department of Computer Engineering Techniques, College of Engineering, University of Al Maarif, Al Anbar, 31001, Iraq
Emails: osama_mohammed@mtu.edu.iq; marwaibrahimbrahim@hotmail.com; engrahumi@uoa.edu.iq

Abstract

The integration of sensing technologies with residential buildings raises the concept of a smart home, which has facilitated the life of the habitant nowadays. This technology helps us to track and understand the behavior of the client in the house to give him maximum comfort. A neighborhood area is an interconnected set of houses that exist in the same geographical region and share the same energy resources. The most important component in the process of decision-making is the energy usage in the smart building. The energy optimization problem in the smart building created a challenge for enterprises and the government for a long time. A lot of research were made to solve this energy optimization problem. One of these problems is the organization of energy usage within a neighborhood area network. The main challenges are to maintain the user comfort in each house and to not exceed the total energy offered to the network. For this, we proposed a technique that predicts, based on historical data of each house, its future behavior and created for each one a weekly schedule with hourly annotated field with: high, normal, or low, where each one represents the amount of energy user is able to use at this time. At the end, an incentive-based program is created to give the client an incentive on his bill if he used the daily high energy consumption in the annotated high in his schedule. To create the schedules, we extracted some features from the data, then we used the genetic algorithm to create schedules, then we did an improvement to the technique using dynamic programming that stores the features of a house with created schedule, later when we meet a similar house we can directly give a schedule that fits the need.

Keywords: Smart home; energy management; incentive-based program; dynamic programming; genetic algorithm

1. Introduction

The evolution of new technologies has introduced new concepts and information technologies in various domains of life. New sensors have enabled the collection of data about the target environment, allowing for more control over household aspects such as ventilation, heating, lighting, air quality, and lighting. Researchers use this collected data to analyze and make decisions based on the models, aiming to reduce energy use and maintain user comfort.

The smart grid is a collection of technologies, concepts, topologies, and components that allow for efficient exchange of data, services, and transactions [1]. The topology of the smart grid consists of many components and layers, with sensors collecting data from the house and the environment, actuators being adjusted, and communication applied through a network layer with different technologies (bluetooth, zigbee, wifi). Advanced control layers are responsible for computing and decision-making.

However, the domain faces challenges such as equipment topology, design, maintenance, security, risk prevention, etc. One of the most important challenges is energy management, maintaining user comfort. Deploying various sensors to achieve the comfort index with minimal energy consumption is a challenge [2]. Home Energy

DOI: https://doi.org/10.54216/JCIM.150202
Received: August 22, 2024 Revised: September 29, 2024 Accepted: October 28, 2024

Management Systems (HEMS) assist end-users in managing energy prices by solving scheduling problems considering energy resources [3]. Two main approaches to maintaining energy management efficiency in neighborhood area networks are Demand Side Management (DMS) and Demand Response (DR) [4]. DSM programs include activities or programs organized by utilities that provide energy and control consumer use time [5]. In neighborhood areas, where many houses share the same energy resource, energy providers face problems when the total usage exceeds a limit set by the providers. To reduce energy usage without affecting user comfort, the integration of technology demonstrated good results in smart homes. Researchers can analyze energy usage data, understand user behavior, create schedules, or adjust actuators in the house.

A technique using genetic algorithms was developed to generate weekly schedules for each house using historical data. The schedule contains the amount of energy allowed at each hour in each day of the week, marked as low, medium, or high, indicating approximate energy consumption. An incentive-based program was applied to encourage the client to use the most consumption within high consumption hours. One of the problems that face the energy provider is not to exceed a certain limit 17 during a certain time. One of the solutions for this problem is the scheduling, when each house is given a schedule of the way of energy consumption. A lot of techniques and machine learning algorithm were applied in this field [6][7][8].

In our contribution, we developed a technique based on historical data collected from houses, and extracted features from this data, then using these features we applied the genetic algorithm to create the best set of schedules. To improve the performance of the technique, we added a database where we store the vector of features with the schedule associated for it, so in the next time we meet a similar house, we can directly give the schedule.

- Develop Predictive Models for Energy Consumption: To create predictive models that analyze historical energy usage data from individual homes, enabling the forecast of future energy consumption patterns and behaviors within a smart neighborhood.
- Optimize Energy Usage Schedules: To design and implement a genetic algorithm-based scheduling system that generates weekly energy consumption schedules for each house, ensuring that energy usage aligns with both user comfort and the total energy limits of the neighborhood network.
- Implement Incentive Programs for Energy Conservation: To establish an incentive-based program that encourages residents to adhere to their scheduled energy consumption levels by providing financial rewards or discounts on energy bills for utilizing energy during designated low-consumption periods, ultimately promoting energy efficiency across the neighborhood.

The reminder of this paper is structured as follow: After Introduction, section 2 provides acronyms and definitions. Section 3, related works. In section 4, we describe the method of Our System. In section 5, Simulation of this system. Finally, Conclusion and Future work.

2. Related Work

It numerous studies have been conducted to reduce energy consumption in smart homes, as energy is a crucial area of study. This paper discusses related works in smart home energy management, how researchers broaden their search to neighbourhood area networks, and the techniques used in these studies.

2.1. Energy Management

The smart home has been optimized by reducing energy consumption across various building parts, using various data criteria to assess occupant comfort. In their paper,[9] the study developed a function using temperature, illumination, and CO2 indoor information to calculate comfort levels. Using genetic algorithm and particle swarm optimization algorithms, they achieved 27.32% and 31.42% reductions in energy consumption and 10% and 4.19% improvement in comfort index with PSO and GA. For instance, [2] the study used humidity and temperature measurements to determine comfort index and energy consumption estimates, resulting in an 8.34% reduction in energy consumption.

The study utilized fuzzy logic and the Kalman filter to control the environment, predict energy consumption, and calculate comfort based on temperature, lighting, and air quality [10]. The system was designed with a coordination agent for optimization and a load agent for client comfort during low energy periods [11]. The management system consists of two levels: a lower level that controls air and water flow rates, and an upper level that uses external temperature, solar gains, building shell, and internal loads.

In this study, [12] a smart home energy management system (SHEMS) utilizing a parallel-processing-implemented, GPU-accelerated neurocomputing-based time-series load modelling and forecasting mechanism is proposed for smart home automation. Energy decomposition is used to facilitate the time-series load modelling and forecasting mechanism, which tracks appliance-level electrical energy consumption to be quantitatively modelled from circuit-level consumption, with no intrusive deployment of networked plug-level power meters for

DOI: https://doi.org/10.54216/JCIM.150202
Received: August 22, 2024 Revised: September 29, 2024 Accepted: October 28, 2024

individual electrical home appliances This study lacks a comparative analysis of the proposed neurocomputing-based time-series load modelling and forecasting mechanism with existing methods or technologies. This comparative analysis could provide a better understanding of the effectiveness and efficiency of the proposed system. Although, Utilization of Deep Learning Models: The study shows an innovative and effective method for load modelling and forecasting by using deep learning models that are accelerated by a GPU for parallel processing. In this paper [13], provides three original and novel smart home energy management algorithms that depend on the most common residential tariff specifically in developing countries. Three different management concepts have been studied for a typical Egyptian house. The proposed energy management scheme of PV-powered home reduces the electrical power bill significantly in a wide range from 61% to only 19% of the default case bill according to the applied management technique. The weakness of this paper is that insufficient attention to the technological, legal, and cultural barriers that might prevent practical use in real-world situations.

2.2. Neighbourhood Area Network

In this section explores managing energy usage in homes and discusses various approaches, including scheduling, which is a crucial method for organizing energy usage in smart grids, as suggested by researchers. Researchers [14] used evolutionary algorithms to residential communities for scheduling and compared the outcomes with and without emphasizing customer comfort. the study proposed an adaptive scheduling strategy using ADHDP neural networks to manage power in homes with shared energy sources, which are continuously updated [15].

3. Method of Our System

The proposed method aims to generate weekly schedules for each home in a neighborhood network, ensuring that the energy consumed doesn't exceed the available resources and that the schedules fit each house's needs. The process is illustrated in Fig. 1, with each phase detailed later. The algorithm is expected to provide the best schedules for each home. The neighboring grid N1 consists of three homes H1, H2, and H3, all sharing the same energy source E1. The server will collect and store the energy's hourly usage data.

Figure 1. Steps of the technique of scheduling

4. Simulation

The data collection [16] utilized for the simulation includes measurements from several dwellings in various locations and throughout various time periods. We chose 10 residences and split them into two groups for the simulation; the first set has 4 houses with readings between 2015-01-30 and 2018-01-29. They will be used as our initial collection of data. The second one has readings for 6 residences that range from 2015- 02-21 to 2018-02-20. We next examined the combination of the 10 houses as one data set (taking into account the shared dates) to see the impact of the increase in the number of houses on the algorithm. We applied the model to each of these sets individually. The information is compiled from several csv files, each of which represents a dwelling. The file provides information on the energy consumption every hour. Three columns make up the csv file: date, hour (0–23), and energy consumed. Figure 2 displays the amount of energy used during each of the 720 hours in a month, represented in KWh.

DOI: https://doi.org/10.54216/JCIM.150202
Received: August 22, 2024 Revised: September 29, 2024 Accepted: October 28, 2024

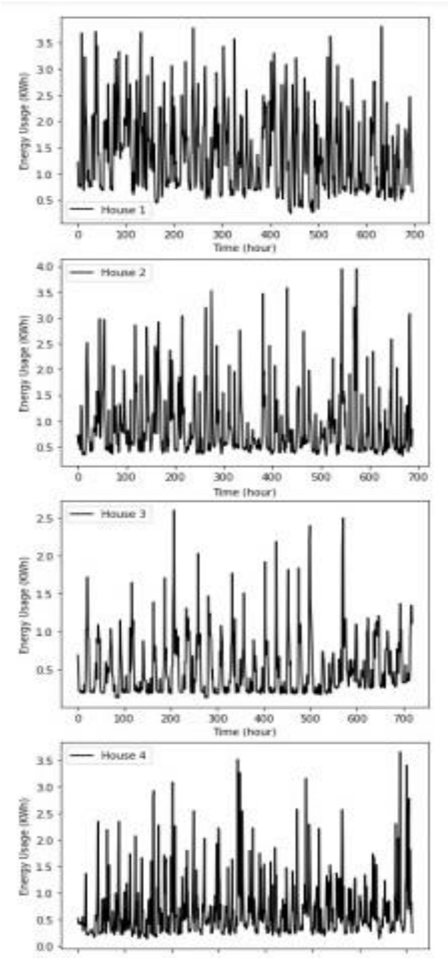

Figure 2. Energy consumption in each of the four houses in the data set 1.

4.2. Pre-processing

Since some of the data were missing, we used information from other fields to fill in the gaps. The value of the missed data will be calculated as the mean of other energy use figures at the same hour from various days, taking 10 days before and 10 days following into account because there is a resemblance between the data at the same hour of the day. For instance, if hour 15 of the date 18 March 2015 has a missing value. In the unlikely event that one of the required numbers for the computation is overlooked, it is removed and the denominator is lowered by one.

4.3. Coding

Python and the Jupiter notebook were chosen because of its comprehensive library for machine learning and ease of working with csv data, as well as its flexibility and simplicity in creating and testing the code.

4.4 Windows and Features Extraction

We introduced the parameter window size, a dynamic parameter that indicates the number of days or hours that should be viewed as one window to be utilized to acquire the characteristics, in order to achieve correct results. Table 1 displays the window sizes for both hours and days.

Table 1: Different size of windows adapted for the simulation

Parameters	Windows Size		
Days	3	4	5
Hours	7	(5,2)	10

DOI: https://doi.org/10.54216/JCIM.150202
Received: August 22, 2024 Revised: September 29, 2024 Accepted: October 28, 2024

The day's parameter window size indicates the number of days considered for features. For example, if Wday=0(Monday) = 3, we consider the first three Mondays of the next three weeks. For Whour=0 = (7), we use hour 0 for the next 7 days. The hour parameter window size is a vector of n values, each indicating the size of the next window.

4.5 Schedules Data Base

The created schedules should be stored at the conclusion of the scheduling phase so that we may use the optimization approach later on and retrieve them with minimal complexity. A scheduling database is built for this, and in order to get the record with the slowest possible processing time, the data are stored in a no-sql database using the mongoDB technology. Due to its accessibility and the data, we chose the "hash" data structure. The hash map's structure, as illustrated in Table 2, is a key-value structure. The value is a matrix with the days of the week as columns and the hours as lines, and the values are the labeled energy consumption. The key is specified as the similarity vector. The vectors are kept in a sorted sequence and a binary search is used to speed up the search process.

Table 2: Structure of the hash map

Key	Value
Similarity	Schedule Matrix
Vector	

4.6 Scenarios

Three situations were used with the simulation:

• Data sets 1 and 2 are not optimized in Scenario 1 and Scenario 2, respectively.

• Data set 2 with optimization is the third scenario.

Because we didn't yet have any saved data in the first case, we performed the scheduling algorithm without doing the optimization step on the features. However, after the procedure was complete, we stored the schedules. In order to obtain a result and compare it to the outcome of the third scenario, we performed the scheduling algorithm in the second scenario without first applying the optimization step to the characteristics. Since we are dealing with the same data in this scenario and would have identical houses, we did not keep any data to ensure that we would not have an impact on the outcome of the third scenario. To compare the outcome with the outcome of the second scenario, we performed the scheduling method with the feature optimization step in the third scenario.

4.6.1. Features Selection

We chose to use 12 features for the simulation, which is less than half of the total amount of characteristics we have. Final characteristics include total, mean, min, max, root mean square, standard deviation, variance, and list of low usage hours, List of high usage hours, List of out hours, Number of low usage hours per day, Number of high usage hours per day.

4.6.2. Scenario 1

Table 3 displays the various outcomes when modifying the window sizes. It lists nine potential adaptations for the sizes, along with the accuracy and necessary processing time for each

Table 3: Results of scenario 1

	3		4		5	
	Accuracy	Time	Accuracy	Time	Accuracy	Time
(7)	87.9%	48 minutes	86.5%	42 minutes	86.3%	39 minutes
(2,5)	92%	55 minutes	90.6%	53 minutes	89.9%	49 minutes

DOI: https://doi.org/10.54216/JCIM.150202
Received: August 22, 2024 Revised: September 29, 2024 Accepted: October 28, 2024

(10)	89.5%	47 minutes	89.4%	40 minutes	89.3%	38 minutes

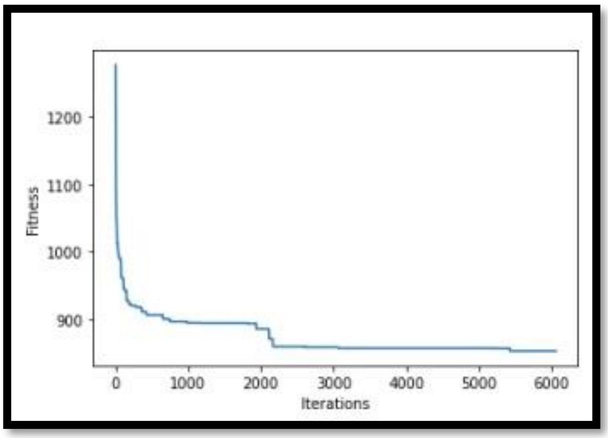

Figure 3. Graph showing the evolution of the algorithm in the first scenario

With a 92% accuracy rate, the results indicated that the optimal choice for the day window size was 3, and for the hour window size was 2.5. The findings demonstrated that the method will finish the computation faster when the window size is increased. It took the algorithm about 3310 seconds (55.1 minutes) to reach this result in the first scenario, where we used 6500 iterations. At the first iteration, the fitness function was 1277; at iteration 600, it was decreased to attend 906; at iteration 5000, it continued to attend 850; and at iteration 1000, iteration 1000 remained unchanged. Fig 3 displays the evolution of the algorithm in the first scenario.

According to the Fig 4, the fitness of the algorithm began off the same as in Fig 5, but it quickly dropped from roughly 1200 to 220 over the first 1000 iterations. The algorithm then steadily decreased over the next 4500 iterations, reaching 204 attendees as indicated in the Fig 4.

As a consequence of the scoring function's modification, we received faster results, and the algorithm spread more quickly. We can see that the algorithm reached the low point twice as quickly as the first one.

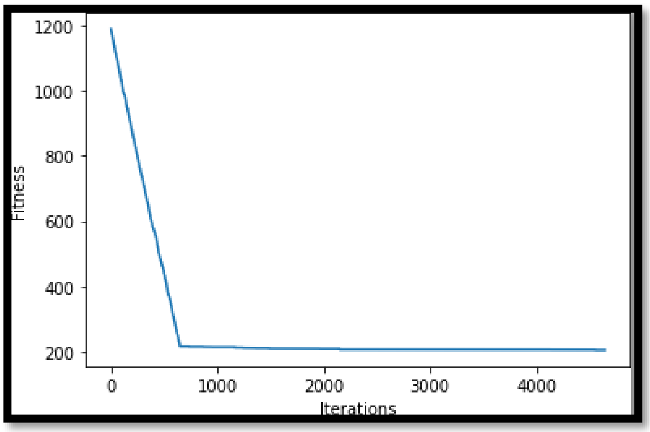

Figure 4. Grade method add point evolution of the algorithm

DOI: https://doi.org/10.54216/JCIM.150202
Received: August 22, 2024 Revised: September 29, 2024 Accepted: October 28, 2024

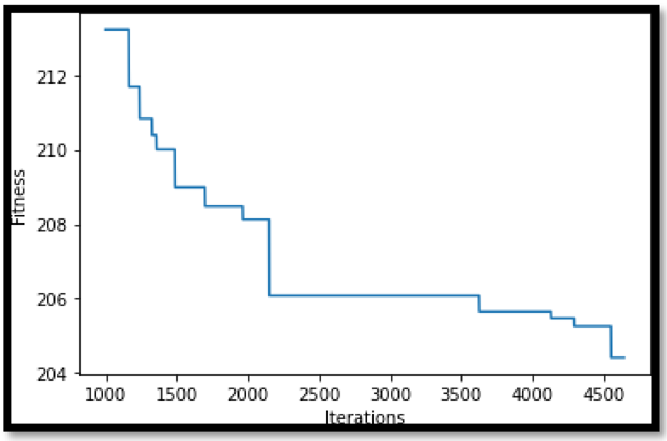

Figure 5. Grade method add point evolution after 1000 iterations

4.6.2.2. Saving Schedules

The results were maintained in a database where the key was a vector of 12 characteristics that was about 0.5 and the schedule was stored for each row in order to be able to apply the optimization procedure.

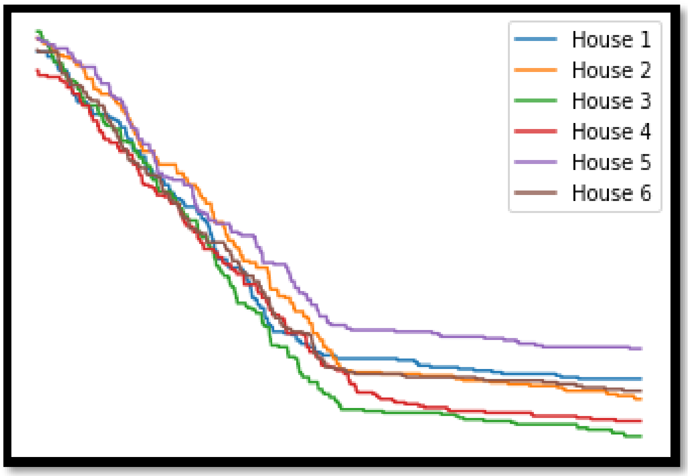

Figure 6. Evolution of the algorithm in the second scenario value

4.6.3. Scenario 2

Instead of applying the similarity approach with the scenario 1 data that was already recorded, we used the methodology in this scenario on data set 2, which has 6 dwellings.

We changed the window widths to 3 for days and 2.5 for hours because the prior case shown that these sizes were quite accurate.

The results revealed that the values declined from the initial fitness and started to stay steady between iterations 700 and 800. The decline of the fitness value for each dwelling is illustrated in Fig 6. The algorithm ran for 32 minutes with 1500 iterations.

Table.4 displays the varying accuracy for each dwelling after 1500 iterations. After accounting for the requirement that simultaneous energy consumption should not exceed a particular threshold, the algorithm's overall accuracy in scenario 2 was 82%. The entire procedure took 32 minutes to finish.

Table 4: Accuracy of each house after the application of the algorithm on data set 2

	Hours1	Hours2	Hours3	Hours4	Hours5	Hours6
Accuracy	812%	83%	90%	88.5%	78.4%	82.4%

DOI: https://doi.org/10.54216/JCIM.150202
Received: August 22, 2024 Revised: September 29, 2024 Accepted: October 28, 2024

Table 5: Comparing results of house 4 between scenario 2 and scenario 3

	Starting Fitness	Low Fitness	Accuracy
Scenario2	292	83	88.5%
Scenario3	292	89	87.6%

4.6.4. Scenario 3

In this case, we applied the technique to the second data set, which contains six houses, but after applying an optimization step and computing all necessary data features, we concluded that there was a similarity between House 1 from the first data set and House 4 from the second data set at a similarity rate of 2.5.

The timetable is specifically given for that home because of this, while the other houses' scheduling was carried out as usual. Fig .6 depicts the fitness score decline over time for each of the homes, and it reveals that Houses 1, 2, 3, 5 and 6 are somewhat similar to those in Scenario 2 in that they all began with high values and then began to evolve slowly after 700 to 800 iterations.

However, Table 5 compares the outcomes for house 4 in scenarios 2 and 3. The house 4 started with a high fitness value (roughly 292) at iteration 0 in both scenarios, but then quickly dropped to a low fitness value of 89 in scenario 3, which is still acceptable when compared to the fitness of the house 4 in scenario 2. On home 4, the algorithm's accuracy dropped by 0.9%. The algorithm took 23.5 minutes instead of the scenario 2's duration, a 26% reduction.

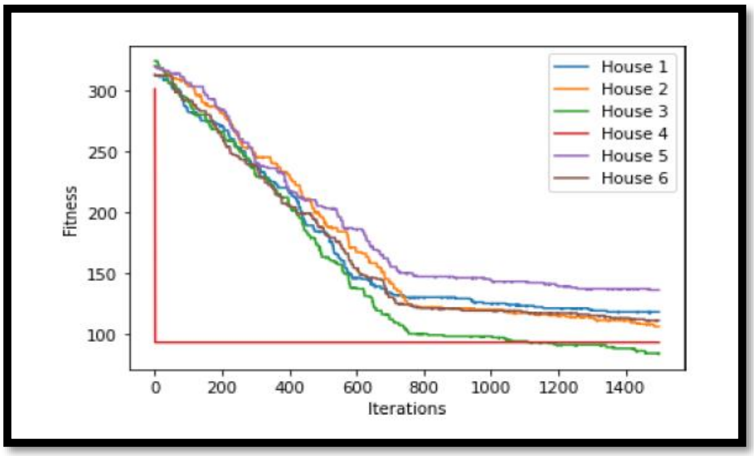

Figure 6. Evolution of the algorithm in the second scenario

5. Results and Discussion

We will have a weekly timetable for each residence at the conclusion of the scheduling, as seen in Table 6. The weekdays are shown as columns in the schedule, and each row corresponds to each hour of the day, from 0 at midnight to 23. Low, Normal, and High values are filled in the values, which indicate the range of energy that the client should maintain**.**

The optimization of the method shown a reduction in the amount of time needed to finish the scheduling. In our situation, we only had one prior data set to work with, but if we had 100 prior data sets, the likelihood of finding comparable houses would rise, and we could be able to immediately generate schedules for every property.

Table 6: Weekly schedule for a house

h	Mon	Tue	Wed	Thu	Fri	Sat	Sun
0	Normal	High	High	Low	Low	Low	High
1	Normal	Normal	Low	High	Normal	Low	High
2	Normal	Normal	Low	High	Normal	Low	Normal
3	Normal	Normal	Low	Normal	Normal	Low	Low
4	High	Low	Normal	Normal	High	Normal	Low
5	High	Normal	Normal	Normal	High	Normal	Normal
6	Low	Normal	Normal	Normal	High	Low	Normal

DOI: https://doi.org/10.54216/JCIM.150202
Received: August 22, 2024 Revised: September 29, 2024 Accepted: October 28, 2024

7	Low	Low	High	Low	Normal	Low	Normal
8	Low	Low	High	Low	Normal	Normal	Normal
9	Low	Low	Normal	Low	Normal	Normal	Normal
10	Normal	Normal	Normal	Low	Normal	Normal	Normal
11	Normal	Low	Normal	Low	Low	Normal	Low
12	High	Normal	Low	High	Normal	Low	Normal
13	High	High	Low	Low	Normal	Normal	Low
14	Low	High	Low	High	Normal	Normal	Low
15	Normal	Normal	Normal	Low	Low	Normal	Low
16	Low	Low	Low	Low	Normal	High	High
17	Normal	Low	Normal	High	Normal	High	Normal
18	Low	Normal	Normal	Low	Low	Normal	High
19	Normal	Low	Low	Low	Low	Normal	High
20	Normal	Low	Low	Low	Low	High	Normal
21	Normal	Normal	Normal	High	Normal	High	Normal
22	Normal	Normal	Low	High	Normal	High	Normal
23	Normal	High	High	Normal	Normal	High	Low

6. Conclusion

In this research, we suggested a method for neighborhood area network energy suppliers, which allows them to schedule homes to ensure that the overall energy consumption will not exceed a specific quantity of energy delivered at the same time. The incentive-based program will persuade the client to use the majority of his energy during the hours marked as "High," in which case he will receive a bonus on his bill. We presented the simulation's findings in this part. We first provided the key features that were chosen after using the feature selection method, followed by a display of the outcomes for each scenario. The algorithm produces n 7x24 schedules, each of which reflects the ideal timetable for n dwellings. We employed the evolutionary algorithm for scheduling, which took 55 minutes to complete, using characteristics that were taken from historical data for each house. In the best circumstances, the program had a 92% accuracy rate. We implemented an optimization technique where we would keep the house's features and its timetable in a database. The timetable for that residence will be assigned when we have calculated the features and looked for a comparable vector of features in the database. In our simulation, this strategy demonstrated a 26% improvement in time with no reduction in accuracy. By evaluating more characteristics and other distance calculation techniques, we hope to enhance the algorithm's performance in terms of its fitness function in the future. Additionally, we'll test Particle Swarm Optimization, another optimization method.

Funding: “This research received no external funding”

Conflicts of Interest: “The authors declare no conflict of interest.”

References

[1] H. Farhangi, “The path of the smart grid,” IEEE Power Energy Mag., vol. 8, no. 1, pp. 18–28, 2010, doi: 10.1109/MPE.2009.934876.

[2] KETSEAS, D. (2024). Stochastic Response of an Airfoil and Its Effects on Lco’s Behavior Under Stall Flutter Regime. International Journal of Mathematics, Statistics, and Computer Science, 2, 168–172. https://doi.org/10.59543/ijmscs.v2i.8663

[3] H. Merdanoğlu, E. Yakıcı, O. T. Doğan, S. Duran, and M. Karatas, “Finding optimal schedules in a home energy management system,” Electr. Power Syst. Res., vol. 182, no. January, p. 106229, 2020, doi: 10.1016/j.epsr.2020.106229.

[4] B. Celik, R. Roche, S. Suryanarayanan, D. Bouquain, and A. Miraoui, “Electric energy management in residential areas through coordination of multiple smart homes,” Renew. Sustain. Energy Rev., vol. 80, no. May, pp. 260–275, 2017, doi: 10.1016/j.rser.2017.05.118.

[5] M. N. Q. Macedo, J. J. M. Galo, L. A. L. De Almeida, and A. C. De, “Demand side management using artificial neural networks in a smart grid environment,” Renew. Sustain. Energy Rev., vol. 41, pp. 128–133, 2015, doi: 10.1016/j.rser.2014.08.035.

[6] Fatikhan, A. Abed, K. Hikmat, A. Yousif, M. Salim, A. Hatem, Q. Bader, S. "A Hybrid GA-GWO Method for Cyber Attack Detection Using RF Model," Journal of Cybersecurity and Information Management, vol. , no. , pp. 225-232, 2025. DOI: https://doi.org/10.54216/JCIM.150117

DOI: https://doi.org/10.54216/JCIM.150202
Received: August 22, 2024 Revised: September 29, 2024 Accepted: October 28, 2024

[7] N., S. Yousif, M. Bader, S. Mohammed, O. Hikmat, A. "A Public Key Infrastructure Based on Blockchain for IoT-Based Healthcare Systems," Journal of Cybersecurity and Information Management, vol. , no. , pp. 233-243, 2025. DOI: https://doi.org/10.54216/JCIM.150118.

[8] Salim, A. N., S. Abul, D. Awad, M. "Credit Card Fraud Detection Model Based on Correlation Feature Selection," Journal of Cybersecurity and Information Management, vol. , no. , pp. 334-342, 2024. DOI: https://doi.org/10.54216/JCIM.140224

[9] I. Ullah and D. Kim, "An improved optimization function for maximizing user comfort with minimum energy consumption in smart homes," Energies, vol. 10, no. 11, 2017, doi: 10.3390/en10111818.

[10] S. Ali and D. H. Kim, "Effective and comfortable power control model using Kalman filter for building energy management," Wirel. Pers. Commun., vol. 73, no. 4, pp. 1439–1453, 2013, doi: 10.1007/s11277-013-1259-9.

[11] Z. Wang, L. Wang, A. I. Dounis, and R. Yang, "Multi-agent control system with information fusion based comfort model for smart buildings," Appl. Energy, vol. 99, pp. 247–254, 2012, doi: 10.1016/j.apenergy.2012.05.020.

[12] Y. H. Lin, H. S. Tang, T. Y. Shen, and C. H. Hsia, "A Smart Home Energy Management System Utilizing Neurocomputing-Based Time-Series Load Modeling and Forecasting Facilitated by Energy Decomposition for Smart Home Automation," IEEE Access, vol. 10, pp. 116747–116765, 2022, doi: 10.1109/ACCESS.2022.3219068.

[13] R. Elazab, O. Saif, A. M. A. A. Metwally, and M. Daowd, "New smart home energy management systems based on inclining block-rate pricing scheme," no. June, pp. 503–511, 2022.

[14] T. Roy, A. Das, and Z. Ni, "Optimization in load scheduling of a residential community using dynamic pricing," 2017 IEEE Power Energy Soc. Innov. Smart Grid Technol. Conf. ISGT 2017, no. c, 2017, doi: 10.1109/ISGT.2017.8086087.

[15] D. Liu, Y. Xu, Q. Wei, and X. Liu, "Residential energy scheduling for variable weather solar energy based on adaptive dynamic programming," IEEE/CAA J. Autom. Sin., vol. 5, no. 1, pp. 36–46, 2018, doi: 10.1109/JAS.2017.7510739.

[16] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," Comput. Electr. Eng., vol. 40, no. 1, pp. 16–28, 2014, doi: 10.1016/j.compeleceng.2013.11.024.

DOI: https://doi.org/10.54216/JCIM.150202
Received: August 22, 2024 Revised: September 29, 2024 Accepted: October 28, 2024