Optimizing Energy Efficiency in IoT-Based Smart Home Systems

Gamze AKÇAY
Necmettin Erbakan University

Behice Gülsüm ÇELİK
Necmettin Erbakan University

Alperen EROĞLU
Necmettin Erbakan University

To Cite This Chapter

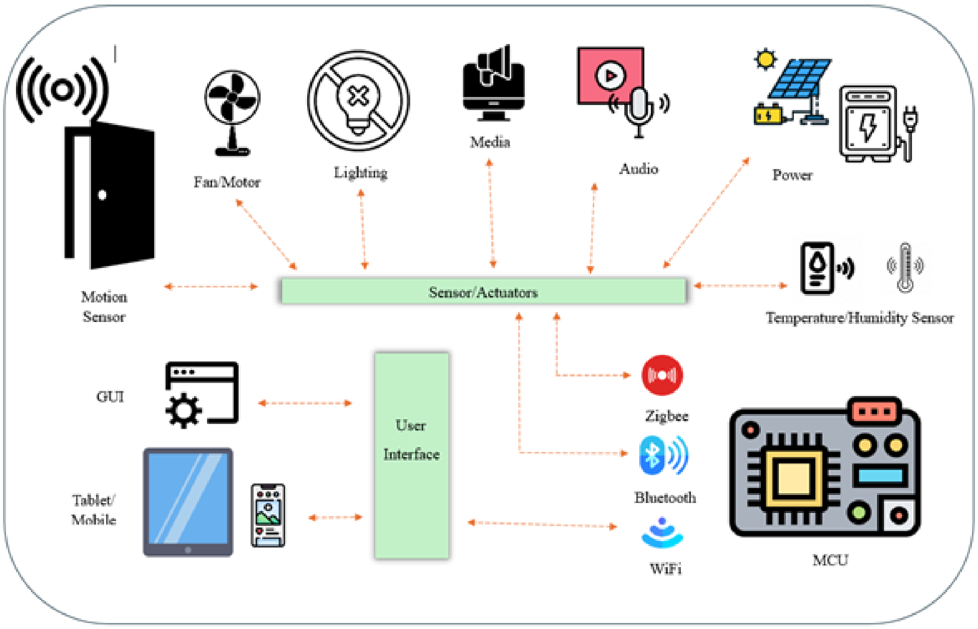
Akçay, G., Çelik, B. G., & Eroğlu, A. (2024). Optimizing Energy Efficiency in IoT-Based Smart Home Systems. In S. Koçer & Ö. Dündar (Eds.), *Intelligent Systems and Optimization in Engineering* (pp. 53-66). ISRES Publishing.

Introduction

The advancement of technology, the equipping of communication infrastructures with new-generation systems, and the need for a completely digital and autonomous world of future communications such as 5G and beyond (5BG) and 6G have required people to enter a different evolution with the adaptation of these technologies not only in their business lives but also in many areas such as health, agriculture, and defense (Eroglu, 2022). New communication and sensing innovations also allow connections to be ubiquitous, allowing communication among devices from anywhere at any time. This connectivity form is called the Internet of Things (Kaya et al., 2019). One of these important areas is that smart homes must be built with new-generation requirements. The smart home automation paradigm has increased significantly in recent years due to the growing demands of people and their desire for comfortable living. However, with the rising use of smart home automation, significant energy consumption poses a problem (Geraldo Filho, et al., 2019).

A smart home can be seen as a living space where devices, systems, and services received by users are automated, optimized, and integrated using sensors, digital systems, and IoT platforms. With smart home technology, users can remotely manage devices in the home and live in a way that provides energy efficiency and increased comfort. As depicted in Figure 1, a smart home uses some off-the-shelf IoT sensors and actuators such as motion sensors, thermal sensors, lighting, power, audio systems, air conditioning, and a fan or motor. A smart home also includes various communication technologies such as Wi-Fi, Zigbee, and Bluetooth. A smart home has a user interface can be controlled via remote access with a web interface or mobile app. All these basic requirements consume energy.

Figure 1
A Basic Smart Home Consisting of IoT- instruments.



In IoT-based smart home applications, many situations such as making life easier, controlling health status, etc. are realized using the data collected. While various artificial intelligence algorithms are analyzed by statistical and mathematical modeling and turned into valuable insight, on the other hand, the production, transmission, storage, and processing of this data, i.e. the use and existence of software, hardware, and data, increase energy consumption (Eroglu and Unlu Eroglu, 2023). At this point, using energy efficiently by applying the optimization method to find the best-desired result is inevitable. Hence, in this chapter, we address this problem.

Goal and Motivation

The Internet of Things (IoT) paradigm has gained great importance. It is a global network of physical objects embedded with sensors, software, and other technologies for data connection and sharing with other devices/systems over the Internet that can communicate with each other via IoT communication protocols. The Internet of Things shows its presence in many areas. It has made progress, especially in smart home automation. In addition to important innovations such as remote electronic device control and system security in home automation, it has also provided comfort and convenience. In addition to its positive effects, it has also brought problems that need to be solved. One of these main problems is that it has seriously increased energy consumption. The problem of reducing and optimizing energy consumption has been the subject of many research studies. The study focuses on the pivotal issue of energy efficiency in IoT-driven smart home applications.

Research Questions

In this study, we want to understand and find answers to the research questions listed below.

- How are IoT devices and the networks created by these appliances used in smart home applications?
- What are the devices commonly used in smart home applications?
- What are the consumption amounts of the instruments?
- Can we turn off the devices, put them in sleep mode, or limit their communication to provide less energy, taking into account the user's comfort?

- What are the data sets used in the literature?
- Can optimization solutions be applied in smart home applications using these data sets?
- Can these solutions be generalized?

Problem Definition

In recent years, there has been a significant increase in energy consumption associated with smart home systems. Effective management of energy consumption is crucial for both reducing environmental impacts and lowering energy costs for users. This study focuses on preserving user comfort and flexibility while enhancing energy efficiency within smart homes. In this context, the primary aim is to optimize energy consumption levels in smart home environments.

For this research, we applied three different optimization algorithms to our dataset: Grey Wolf Optimization (GWO), Genetic Algorithm (GA), and Particle Swarm Optimization (PSO). Each algorithm approaches energy utilization in the smart home context from diverse perspectives, offering a range of potential improvements. We assume the desired values of temperature and humidity are provided by the home user so that we consider user comfort.

Methodology

In our study, the most up-to-date studies in the literature on energy-efficient smart home applications are investigated and presented with a comparative analysis. In this study, we first find the data set obtained from an IoT-based smart home application. Then, the most commonly applied optimization algorithms are applied to this data set by considering user provided optimal temperature and humidity values.

Contributions

This study focuses on IoT-based smart home applications with real-life implementations and simulations. Studies implemented using embedded systems and shared data sets are both categorically examined and implementing the three most commonly used optimization algorithms is carried out using a data set. Since very few studies generally examine efficient IoT-based home applications using energy optimizations, this gap in the literature is filled in this study.

In this context, we approach the smart home energy optimization problem with a comparative analysis and comprehensive evaluation. We also compare the results of three different optimization algorithms (PSO, GA, and GWA) on the Appliances Energy Prediction dataset published on Kaggle, which to our knowledge has yet to be run before.

The rest of the chapter is like the following.

The subsequent section presents the state of the art for energy-efficient applications in IoT-based smart home systems. The “Energy Optimization in IoT-based Smart Home Systems” section defines the common datasets in the literature and three of the most common preferred optimization algorithms. In the “Experimental Results and Discussion” section, we present the outcomes of The algorithms PSO, GA, and GWA, and a detailed discussion regarding the energy optimization problem in IoT-based smart homes. The section “Future Directions” discusses the most possible open research questions for IoT-based smart home energy management systems. Ultimately, we conclude the chapter by focusing on the most prominent research observations and future studies.

Related Work

This section comprehensively analyzes smart home energy management research. In the literature, studies on smart homes can be divided into different categories. These vary according to how the relevant scenario is realized and at which stages the optimization solution is used. The studies in the first category can be seen as those using a real testbed or experimenting and verifying with simulation. A real testbed can be directly obtained on platforms using embedded systems such as Arduino or Raspberry Pie (Francis et al., 2023) and various sensors such as temperature and humidity, while other studies are IoT-supported smart home applications, where there are many more devices and their designs can be considered as studies where the energy efficiency of smart buildings (Goudarzi et al., 2021) and studies can be evaluated in this context. The studies in the second category are encountered with studies such as minimizing energy consumption of optimization algorithms, user comfort, and bills. In contrast, the other category is used in solutions on parameter optimization and selection of algorithms that will be used in predictive analysis, scheduling systems, and decision support systems (Priyadarshini et al., 2022).

Obtaining energy consumption for energy management using accurate data taken at the correct intervals is emphasized in many studies (Leitao et al., 2020). Home energy management systems (HEMs) are important systems that can guide the user with solutions like timing by using the correct consumption data. A typical HEMs consists of a user interface which can be a mobile application or a remote terminal to make communication with a device and some measurements and state information about appliances; a central management unit to monitor and control energy consumption; measuring and sensing devices to get some physical phenomena such as light, humidity, and temperature; other electronic units including air conditioners, and other smart domestic appliances (Leitao et al., 2020). In a smart home management system smart meters are used to record the energy consumption of each device as well as the consumption resulting from people's activities. In a HEMS, sensors continuously collect data. This data is generated by monitoring the activities in the home. Usually, the energy consumption signals of the devices are recorded, but techniques such as Non-Intrusive Load Monitoring (NILM) can be used to determine the consumption of individual devices. This data is transmitted to a central management unit and processed. Weather information and billing forecasts can also be collected and used in the optimization steps. A HEMS monitors the characteristics and preferences of the home users, takes into account user behavior and profiles, and optimizes the operating schedules of the devices according to physical constraints. The communication protocols between the central platform and the smart devices ensure the implementation of the most appropriate planning. In particular, the communication protocols are also being developed to ensure energy efficiency.

Environmentally friendly applications and preventing waste of resources are made possible with smart homes, smart buildings, and green applications. In particular, there is a need for efficient energy management for the optimal use of electrical energy-requiring devices and systems such as lighting. At this point, IoT-based designs and applications that facilitate accurate and timely data monitoring enable energy management and efficiency to be carried out dynamically. Internet of Things (IoT) home automation systems provide reliable and flexible communication between home devices and the user via the Internet. The increasing use of smart home devices, the widespread use of the Internet, the development of smartphone technology, and the rise of mobile communication standards offer comfort, security, and energy efficiency for users.

Big data (Al-Ali et al., 2017, Machorro-Cano et al., 2020), machine learning, and deep learning algorithms are important for smart home energy management systems. With the help of IoT devices, which are used especially in smart home management systems, it has become an easy application to track device behaviors, create certain patterns, and calculate energy consumption with various measurement techniques. Various analysis studies, predictive analytics studies, and optimization studies can be done on the data to be created here. In particular, the realization of models for making predictions with machine learning and deep learning methods has become an important solution that has taken its place in many studies (Devi et al., 2023).

Most of the studies utilize optimization techniques to make energy-aware smart home management. There are various kinds of optimization algorithms such as PSO, GA, GWA, Butterfly Optimization Algorithm (BOA), artificial neural network (ANN), decision support model (DSM), heuristic system identification (HIS), linear reinforcement learning controller (LRLC), Markov decision problems (MDP), model-based predictive control strategy (MBP), multinomial logistic regression (MLR), ant colony optimization algorithm (ACO), bat algorithm (BAT), and artificial bee colony (ABC), and Simplex Optimization Algorithm (SOA) to find an optimal solution for HEMs. In the literature, most of these algorithms are applied to consider single - or multiple-user comfort, reduction of energy consumption, home user behavior, and learning how to manage (Shah et al., 2019). The multi-objective optimization method has paramount importance for energy optimization. Multi-objective studies take into account important parameters such as temperature, humidity, climate data, the accurate information on energy consumption of different devices and services, as well as user satisfaction parameters, which can include the effect of user behaviors, especially in energy consumption problems (Wang et al., 2021).

All in all, when the studies on energy management in smart homes are examined, it is revealed that there are goals such as reducing power consumption and increasing the comfort index of users with optimization techniques. Especially in smart home environments where annual, weekly, hourly, and even instantaneous data can be collected, it is seen that energy consumption optimization, indoor environment parameter estimation, and energy-aware improvement techniques are important areas of study to achieve the relevant goals of energy management. In this study, we focus on the energy consumption optimization problem and try to show how different optimization techniques, which have not been applied to the relevant data set to the best of our knowledge, can be applied.

Energy Optimization in IoT-based Smart Home Systems

This section explains the dataset from Kaggle obtained from IoT-based smart home systems, and how the three most frequently encountered optimization algorithms in the literature are applied to this data set will be discussed.

Data Set

This dataset, obtained from Kaggle (Appliances Energy Prediction, 2017), is designed to analyze and optimize energy use in smart homes, consisting of 29 columns with 19,735 data entries each. The data headers include the measurement date (date), energy consumption of household appliances (appliances), lighting consumption (lights), and temperature measurements (T1-T9) obtained from various rooms (including the kitchen, living room, and bedroom) indicative of indoor temperature levels, along with relative humidity measurements (RH_1-RH_9) from these rooms, illustrating indoor humidity conditions. Furthermore, the dataset contains external conditions such as outdoor temperature and wind speed that can influence indoor temperatures. Data is collected using two sensors for temperature and humidity measurements, with four actuators (heater, cooler, humidifier, and dehumidifier) to regulate conditions. This dataset is highly suitable for optimizing energy use based on temperature and humidity, managing device energy consumption, and controlling actuators in response to environmental conditions in a smart home context.

Comparison and Implementation of Optimization Algorithms

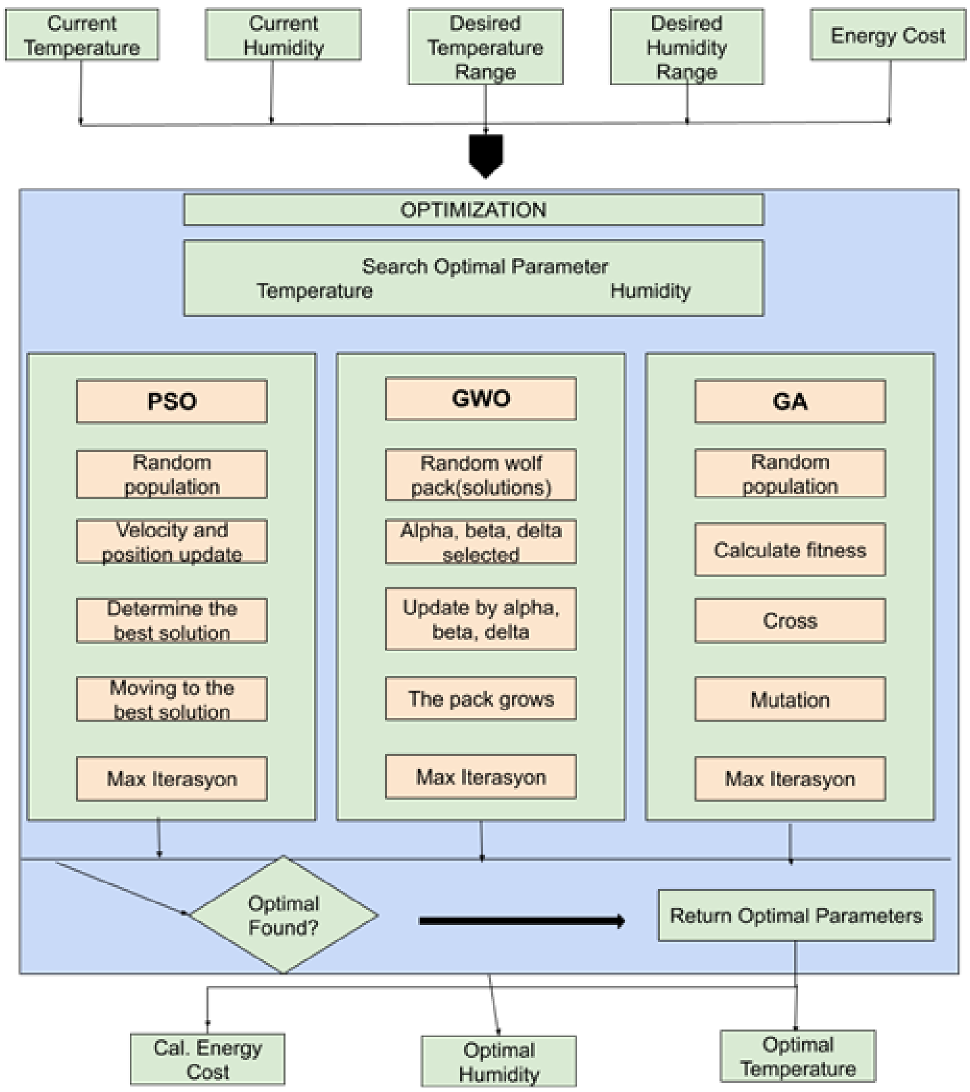
This section explains how our optimization mechanism works, which data set we use, what is our objective function, and which algorithms we implement.

Optimization Approach

Our methodology is demonstrated in Figure 2. We use IoT-based measurements such as temperature, humidity, and the amount of energy cost of each device. We use a multi-objective optimization technique, our methodology uses two important parameters which are temperature and humidity. The objective function is inspired from (Malik et al., 2020). After that, we apply three different well-known optimization techniques to see the differences between each technique and how to apply them to the same problem.

Figure 2

Our Methodology to Apply Optimization Technique by Using IoT-based Measurements.



Objective Function

This section explains the partially modified objective function design we used. Table 1 presents the parameters and their descriptions. In our study, where we have taken the data from the user and thus the user's comfort is seen as optimal values, the aim is to minimize the energy consumption of the function and to achieve this by using temperature and humidity values.

Table 1
The Notations for Parameters

Notation	Description of Parameters
E_{total}	Total energy cost
T_{init}	Initial temperature value
H_{init}	Initial humidity value
T_{max}	Upper temperature target value.
T_{min}	Lower temperature target value
H_{max}	Upper humidity target value
H_{min}	Lower humidity target value.
Tc	Temperature Control
Hc	Humidity control
$Temp\_Error$	Difference between the desired and current temperature.
$Humidity\_Error$	Difference between the desired and current humidity
C_{cost}	Cost of differences
E_T	Energy cost for temperature.
E_H	Energy cost for humidity

$$Temp\_Error = |Tmax - Tc| \tag{1}$$

$$Humidity\_Error = |Hmax - Hc| \tag{2}$$

$$C\text{cost} = (Temp\_Error * mean(E\text{T})) + (Humidity\_Error * mean(E\text{H})) \tag{3}$$

$$E\_balance = [ET + EH] - C\text{cost} \tag{4}$$

$$E\_total = (T\text{init} * mean(E\text{T})) + (H\text{init} * mean(E\text{H})) \tag{5}$$

(1) calculates the absolute difference between the desired maximum temperature and the current temperature. It is used to determine the necessary adjustments for temperature control. (2) calculates the absolute difference between the desired maximum humidity and the current humidity. This difference is used to determine the necessary adjustments for humidity control. (3) calculates the energy costs associated with temperature and humidity errors. Each error is multiplied by the corresponding average energy cost to determine the total cost. (4) computes the energy balance by taking the total current energy costs and subtracting the calculated cost from it. This shows how effective the current energy usage is. (5) calculates the energy costs associated with the initial temperature and humidity values. Each initial value is multiplied by the corresponding

average energy cost to find the total initial energy cost.

The Particle Swarm Optimization Algorithm

Particle Swarm Optimization constitutes an optimization algorithm motivated by the communal conduct of fish schools and avian flocks, predicated on the principle that individuals within a swarm enhance their positions by mutually updating one another (Bahmanyar et al., 2022; Önder, 2011). Algorithm 1 explains the impalmantatiın details of the PSO approach. In the initial phase, a swarm (population) is established, with each individual referred to as a "particle," and a search domain is delineated wherein each particle is allocated a position and velocity. Subsequently, each particle formulates a solution predicated on its fitness value. During each iteration, the particle's optimal value (pbest) and the swarm's optimal value (gbest) are ascertained, and particles revise their positions and velocities before proceeding to the subsequent iteration. This procedure persists until a predetermined cessation criterion, such as a maximum number of iterations (maxIter), is attained, ultimately discerning the optimal solution.

Algorithm 1

The Implementation of Particle Swarm Optimization

- **Start:**
- **Define parameters:**
- *tmax, tmin, hmax, hmin, t0, h0, ect, eht, lb, ub*
- **Define PSO parameters:**
- *maxiter* (maximum number of iterations)
- *pbest* (best particle positions)
- *gbest* (global best position),
- **Define objective function (objective_function):**
- Get *tc, hc* values
- Calculate *t_err: |tmax - tc |*
- Calculate *h_err: |hmax - hc |*
- Calculate *c_cost: (t_err * ecost_t) + (h_err * ecost_h)*
- Calculate energy costs *eb, ec0, et*
- **Run PSO algorithm:**
- Initially determine random particle positions
- Assign *pbest* and *gbest* values
- For iteration in range(*maxiter*):
- For each particle:
- Evaluate objective function and update *pbest*
- If pbest value is better than *gbest*, update *gbest*
- Update particle speed and position according to *pbest* and *gbest*
- Obtain *xopt, fopt* values
- **Print results:**
- Optimized temperature and humidity (*xopt*)
- Minimum energy cost (*fopt*)

The Genetic Algorithm

It is a metaheuristic optimization algorithm inspired by biological evolution processes (Mahmood et al.,2023; Önder, 2011). It is generally used to find the best solutions in large and complex solution spaces. In Algorithm 2, we explain the implementation of the genetic algorithm. The working principle is as follows: an initial population is created from random individuals. A ranking is made among the generated solutions using a fitness function. The most suitable individuals selected from this ranking are designated as parent individuals, and they produce new solutions (offspring) by altering their genetic material. This ensures the diversity of the population and allows for the discovery of new solutions. Small random changes are made. This is used to find potential good solutions that might be overlooked during crossover. After crossover and mutation, new individuals are created, and the process continues with the new population. It uses genetic operators (selection, crossover, and mutation) to produce better solutions from generation to generation.

Algorithm 2

The Implementation of Genetic Algorithm

- **Start:**
- **Define parameters:**
- *tmax, tmin, hmax, hmin, t0, h0, ect, eht*
- *lb, ub, n_generations, cxpb, mutpb, n_runs*
- **Define objective function (objective_function):**
- Get *tc, hc* values
- Calculate *t_err,h_err*
- Calculate *ccost*
- Return *et*
- **Define genetic algorithm structures:**
- Fitness function, individual structure, crossover(*cxpb*), mutation(*mutpb*), selection operations
- **Perform each run (up to n_runs):**
- Create a population (*n=250* individuals) in each run.
- hof (Hall of Fame): Create a structure to store the best individuals.
- stats: Create a structure to calculate statistics.
- **Implement genetic algorithm:**
- algorithms.eaSimple:
- Evolve population (up to *n_generations*).
- Apply crossover, mutation, and selection operations.
- Update the *best individual* and *fitness values* at the end of each generation.
- Get the *best individual* and its *cost* in the hof structure and add it to the list.
- Record the *best individual* and *cost* of each run
- **Find the best result:**
- Find the index of the minimum cost in *all_best_fitnesses* (*min_fitness_index*)
- Set the values of *best_individual* and *best_fitness*
- **Print the results:**
- Print the best individual and cost for each run
- Print the best individual and cost at the end of the *n_runs run*

The Grey Wolf Optimization Algorithm

Grey Wolf Optimization is a meta-heuristic optimization algorithm conceived by deriving inspiration from the hierarchical configuration and predation tactics of grey wolves (Erdoğan, 2023; Makhadmeh et al., 2021). Within this hierarchical configuration, the pack is subdivided into four principal categories: the alpha (α) wolves who govern the pack, the beta (β) wolves occupying the second rank, the delta (δ) wolves positioned third, and ultimately the omega (ω) individuals at the lowest tier. The implementation of GWA is depicted in Algorithm 3. The operational principle of the GWO algorithm is delineated as follows: The initial populace is generated with randomly produced values within specified confines. Subsequently, the fitness value of each individual in the populace is computed, and the locations of the three preeminent individuals are designated as alpha, beta, and delta. Through these steps, both the initial populace and the leading individuals are ascertained, and the algorithm's principal loop commences. Thereafter, the positions of all individuals within the populace are revised utilizing the pertinent objective function equations. Based on the amended positions, the fitness values are recalibrated, and the top three individuals adjust the positions of the alpha, beta, and delta individuals. The principal loop persists until the cessation criterion is satisfied. Upon the conclusion of the loop, the position and fitness value of the alpha individual, who possesses the most favorable fitness value, are recognized as the optimal solution, and the algorithm is finalized (Karakoyun ve Özkış; Mirjalili et al., 2014; Karakoyun, 2021).

Algorithm 3
The Implementation of Grey Wolf Optimization

- **Start:**
- **Define the parameters:**
- *tmax, tmin, hmax, hmin, t0, h0, ect, eht*
- *lb, ub, epoch, pop_size*
- **Define the Grey Wolf Optimization (GWO) problem:**
- *fit_func*: Objective function.
- *lb*: Lower bounds.
- *ub*: Upper bounds.
- *minmax*: "min"
- *verbose*: Detailed output
- **Define objective function (objective_function):**
- Get *tc, hc* values
- Calculate *t_err: |tmax - tc|*
- Calculate *h_errh: |hmax - hc|*
- Calculate *ccost:(t_err * ecost_t) + (h_err* ecost_h)*
- Calculate *ec, eb, et* energy costsa
- **Create a GWO model and find the solution:**
- *model = BaseGWO(problem, epoch=epoch, pop_size=pop_size)*
- *best_position, best_fitness = model.solve()*
- **Find the best result:**
- *best_position*: Best position (*tc, hc*) found as a result of optimization.
- *best_fitness*: Determine the value representing the lowest energy cost.
- **Print results:**
- Print best (*tc, hc*) values found as a result of optimization.
- Print *optimum energy cost*.

Experimental Results and Discussion

We implement three different optimization methods PSO, GWO, and GA are used to reduce energy consumption in a smart home environment. We apply these algorithms to analyze the data set obtained from Kaggle. These methods are tested and analyzed with various objectives focused on temperature and humidity control.

The PSO method is tested to reduce energy consumption by optimizing based on the average temperature and humidity data, providing stable and reliable results in minimizing energy usage. The results show that PSO delivers consistent outcomes with average calculations, focusing on reducing overall energy costs. The GWO method aims to reduce energy consumption by minimizing deviations from target temperature and humidity, effectively achieving comfort conditions. This approach helps us to maintain a low humidity level, though it resulted in higher energy expenditures compared to PSO. The Genetic Algorithm is analyzed through 50 different trials, each using a population of 250 individuals. Crossover and mutation operations are applied over 150 generations, selecting the individuals with the lowest energy consumption. GA effectively controls temperature and humidity with moderate energy usage, and balancing crossover and mutation rates played an important role in reducing energy consumption. According to the light of the results, each optimization method offers unique advantages in reducing energy consumption.

Figure 3

Temperature Humidity and Energy Costs are Obtained as A Result of Applying Our 3 Algorithms to 3 Different Rooms in The Dataset

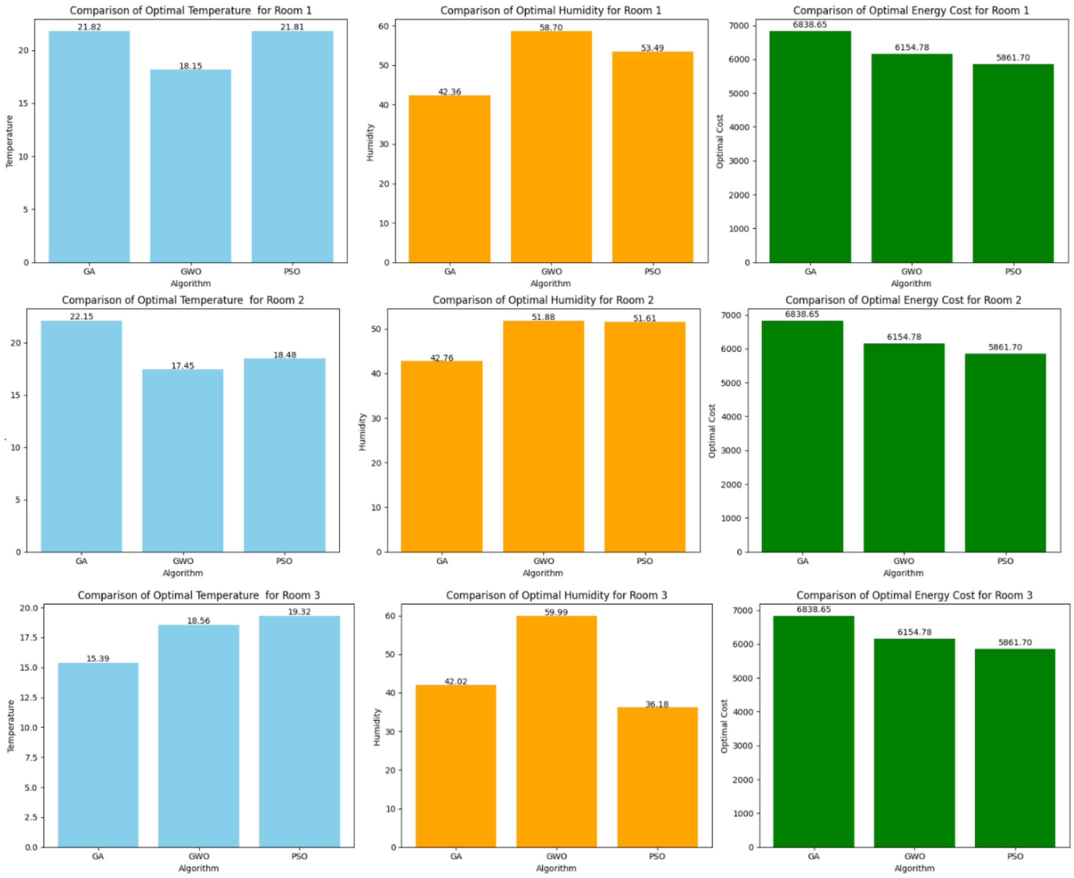


Figure 3 demonstrates that applying our three algorithms to three different rooms in the dataset yields values for temperature, humidity, and energy costs. When we examine the results of the three algorithms, we see that each one produces different outcomes due to its unique optimization strategy. GA provides a balanced solution by keeping humidity within an acceptable range and maintaining costs at a reasonable level. GA's ability to explore a wide solution space allowed it to balance both humidity and cost-effectively and also performed well in temperature regulation. GWO has shown effectiveness in maintaining a low humidity level; however, this has caused an increase in energy costs. GWO's focus on localized solutions may have contributed to the rise in energy costs while optimizing humidity. In contrast, PSO focuses on reducing overall energy costs and achieving the lowest cost. However, this focus led to a humidity level slightly above the target. PSO's rapid convergence has been effective in lowering costs, but it came with a compromise in humidity control. Based on our approach and experimental results, GWO is useful for humidity management, PSO for cost reduction, and GA for a balanced solution. Since the primary goal of our study is to optimize energy costs, PSO offers the best results for this purpose while also maintaining an acceptable temperature and humidity level.

Future Directions

There are some key research questions to enhance our methodology: How does the efficacy of PSO, GWO, and GA algorithms for energy optimization in intelligent residences fluctuate under varying climatic and environmental circumstances? In what manner can the generalizability of these algorithms be augmented through more extensive and diversified datasets? When contrasting the efficacy of PSO, GWO, and GA algorithms with alternative evolutionary algorithms such as Differential Evolution, Simplex Optimization, or Ant Colony Optimization Algorithms for energy optimization, what distinctions become apparent? What are the ramifications of alternative evolutionary methodologies on solution quality and optimization velocity? What influence do multi-objective optimization functions, which reconcile user comfort with energy efficiency, exert on energy optimization in intelligent residences? How can these multi-objective optimization functions be modified to accommodate diverse user behaviors and comfort thresholds? How can PSO, GWO, and GA algorithms be engineered to adapt in real-time to fluctuating environmental conditions in intelligent residences? How can adaptive optimization algorithms reconcile real-time energy cost mitigation and user comfort? In which situations can deep learning-based predictive models furnish more efficacious solutions for energy optimization in intelligent residences? How does the efficacy of AI-enhanced predictive models in forecasting energy demand affect the performance of optimization algorithms?

Conclusion

This chapter aims to increase energy efficiency in IoT-based smart home applications, fills an important gap in the literature, and demonstrates the applicability of energy optimization-based solutions. The use of algorithms such as Gray Wolf Optimization, Genetic Algorithm, and Particle Swarm Optimization on IoT-based data sets provides significant contributions in terms of minimizing energy consumption and preserving user comfort. The findings obtained reveal that energy-efficient management of IoT devices offers both environmental and economic benefits. In future studies, more in-depth analyses can be made in the field of energy consumption management in smart homes by using different data sets and optimization algorithms, and energy management solutions can be addressed from a broader perspective.

Acknowledgments

This study is supported by the TUBITAK 2209-A University Students Research Projects Support Program with project number 1919B012338526.

References

Al-Ali, A. R., Zualkernan, I. A., Rashid, M., Gupta, R., & AliKarar, M. (2017). A smart home energy management system using IoT and big data analytics approach. *IEEE Transactions on Consumer Electronics*, *63*(4), 426-434.

Appliances energy prediction. (2017, September 16). Kaggle. https://www.kaggle.com/datasets/loveall/appliances-energy-prediction/data

Bahmanyar, D., Razmjooy, N., & Mirjalili, S. (2022). Multi-objective scheduling of IoT-enabled smart homes for energy management based on Arithmetic Optimization Algorithm: A Node-RED and NodeMCU module-based technique. *Knowledge-Based Systems*, *247*, 108762.

Devi, M., Muralidharan, S., Elakiya, R., & Monica, M. (2023). Design and Implementation of a Smart Home Energy Management System Using IoT and Machine Learning. In *E3S Web of Conferences* (Vol. 387, p. 04005). EDP Sciences.

Erdoğan, F. (2023). *Application Of Binary Grey Wolf Optimization Algorithm To Binary Optimization Problems* (Master's thesis, Necmettin Erbakan University (Turkey)).

Eroglu, A. & Unlu Eroglu, H. (2023). Topological Data Analysis for Intelligent Systems and Applications. In S. Kocer. & O. Dundar (Eds.), Artificial Intelligence Applications in Intelligent Systems (pp. 27–60). ISRES Publishing.

Eroglu, A. (2022). 5G and Beyond Networks for Digital Transformation: Opportunities and Productivity, M.H. Calp. & R. Butuner (Eds .), Current Studies in Digital Transformation and Productivity (pp. 98–122). ISRES Publishing.

Francis, A., Madhusudhanan, S., Jose, A. C., & Malekian, R. (2023). An Intelligent IoT-based Home Automation for Optimization of Electricity Use. *Przeglad Elektrotechniczny*, *99*(9), 123-127.

Geraldo Filho, P. R., Villas, L. A., Gonçalves, V. P., Pessin, G., Loureiro, A. A., & Ueyama, J. (2019). Energy-efficient smart home systems: Infrastructure and decision-making process. *Internet of Things*, *5*, 153-167.

Goudarzi, S., Anisi, M. H., Soleymani, S. A., Ayob, M., & Zeadally, S. (2021). An IoT-based prediction technique for efficient energy consumption in buildings. *IEEE Transactions on Green Communications and Networking*, *5*(4), 2076-2088.

Karakoyun, M. (2021). *Developing a hybrid method based on Shuffled Frog Leaping and Gray Wolf Optimization algorithms to solve multi-objective optimization problems* (Doctoral dissertation, Dissertation Doctoral Thesis, Konya Technical University).

Karakoyun, M., & Özkış, A. (2021). Development of binary moth-flame optimization algorithms using transfer functions and their performance comparison. *Necmettin Erbakan University Journal of Science and Engineering*, *3*(2), 1-10.

Kaya, M. C., Eroglu, A., Karamanlioglu, A., Onur, E., Tekinerdogan, B., & Dogru, A. H. (2019). Runtime adaptability of ambient intelligence systems based on component-oriented approach. *Guide to Ambient Intelligence in the IoT Environment: Principles, Technologies and Applications*, 69-92.

Leitao, J., Gil, P., Ribeiro, B., & Cardoso, A. (2020). A survey on home energy management. *IEEE Access*, *8*, 5699-5722.

Machorro-Cano, I., Alor-Hernández, G., Paredes-Valverde, M. A., Rodríguez-Mazahua, L., Sánchez-Cervantes, J. L., & Olmedo-Aguirre, J. O. (2020). HEMS-IoT: A big data and machine learning-based smart home system for energy saving. *Energies*, *13*(5), 1097.

Mahmood, Z., Cheng, B., Butt, N. A., Rehman, G. U., Zubair, M., Badshah, A., & Aslam, M. (2023). Efficient scheduling of home energy management controller (hemc) using heuristic optimization techniques. *Sustainability*, *15*(2), 1378.

Makhadmeh, S. N., Khader, A. T., Al-Betar, M. A., Naim, S., Abasi, A. K., & Alyasseri, Z. A. A. (2021). A novel hybrid grey wolf optimizer with min-conflict algorithm for power scheduling problem in a smart home. *Swarm and Evolutionary Computation*, *60*, 100793.

Malik, S., Lee, K., & Kim, D. (2020). Optimal control based on scheduling for comfortable smart home environment. *IEEE Access*, *8*, 218245-218256.

Mirjalili, S., Mirjalili, S. M., & Lewis, A. (2014). Grey wolf optimizer. *Advances in engineering software*, *69*, 46-61.

Önder, E. (2011). Araç rotalama problemlerinin parçacık sürü ve genetik algoritma ile optimizasyonu. *PhD, İstanbul Üniversitesi, İstanbul, Türkiye*.

Priyadarshini, I., Sahu, S., Kumar, R., & Taniar, D. (2022). A machine-learning ensemble model for predicting energy consumption in smart homes. *Internet of Things*, *20*, 100636.

Shah, A. S., Nasir, H., Fayaz, M., Lajis, A., & Shah, A. (2019). A review on energy consumption optimization techniques in IoT-based smart building environments. *Information*, *10*(3), 108.

Wang, X., Mao, X., & Khodaei, H. (2021). A multi-objective home energy management system based on the Internet of Things and optimization algorithms. *Journal of Building Engineering*, *33*, 101603.

About the Authors

Gamze AKÇAY is an undergraduate student at the Department of Computer Engineering at Necmettin Erbakan University.
E-mail: 20010011039@ogr.erbakan.edu.tr, **ORCID:** 0009-0003-5314-1544

Behice Gülsüm ÇELİK, received her B.Sc. degree from the Department of Electrical and Electronics Engineering at KTO Karatay University, Konya, Turkey, in 2021 and completed a double major in Computer Engineering the same year. She is currently a research assistant and pursuing her M.Sc. degree in Computer Engineering at Necmettin Erbakan University, Konya, Turkey. Her research interests are in deep learning and optimization.
E-mail: behicegulsum.celik@erbakan.edu.tr, **ORCID:** 0000-0003-4008-1258

Alperen EROĞLU, received his B.Sc. degree from the Department of Electronics and Computer Education as the valedictorian of the department from Fırat University, Elazığ, Turkey in 2009, the M.Sc. and Ph.D. degrees in computer engineering from METU, Ankara, Turkey in 2015 and 2020, respectively. During his M.Sc. and Ph.D. studies, he researched and studied robotics, artificial intelligence, software engineering, system architecture and modeling, embedded systems, wireless networks, and computer networks. He accomplished his M.Sc. and Ph.D. studies in the field of wireless networks. He has a post-doctorate degree in Computer Engineering at METU. He is currently an assistant professor at the Department of Computer Engineering at Necmettin Erbakan University. His research interests are in 5G mobile networks, the Internet of Things, and wireless and computer networks. He is a member of the Wireless Systems, Networks, and Cybersecurity Laboratory (WINSLab). He is a member of IEEE.
E-mail: aeroglu@erbakan.edu.tr, **ORCID:** 0000-0002-1780-7025

Similarity Index
The similarity index obtained from the plagiarism software for this book chapter is 7%.