energies

Article

Hardware Implementation of a Home Energy Management System Using Remodeled Sperm Swarm Optimization (RMSSO) Algorithm

Senthil Prabu Ramalingam and Prabhakar Karthikeyan Shanmugam *

School of Electrical Engineering, Vellore Institute of Technology, Vellore 632014, Tamil Nadu, India; rsenthilprabu77@gmail.com
* Correspondence: sprabhakarkarthikeya@vit.ac.in; Tel.: +91-9894610689

Abstract: A remodeled sperm swarm optimization (RMSSO) algorithm for a home energy management (HEM) system is proposed, and its real-time efficacy was evaluated using a hardware experimental model. This home environment comprised sixteen residential loads, a smart meter and a Raspberry Pi controller to optimize the energy consumption cost (ECC) in response to the Indian day-ahead pricing (DAP) scheme. A wired/wireless communication network was considered to communicate with the smart meter and controller. To address this optimization problem, the sperm swarm optimization (SSO) algorithm's constriction coefficient was remodeled to improve its global searching capability and proposed as RMSSO. For the first time, salp swarm optimization (SSA), SSO, and RMSSO algorithms were employed to schedule home appliances in the Indian scenario. To validate the proposed technique's outcome, the results were compared to those of the conventional SSO and SSA algorithms. This problem was solved using the Python/GUROBI optimizer tool. As a consequence, consumers can use this control strategy in real-time to reduce energy consumption costs.

Keywords: home energy management system; day-ahead pricing; constriction factor; remodeled sperm swarm optimization; salp swarm optimization; sperm swarm optimization; user satisfaction

check for updates

Citation: Ramalingam, S.P.; Shanmugam, P.K. Hardware Implementation of a Home Energy Management System Using Remodeled Sperm Swarm Optimization (RMSSO) Algorithm. *Energies* **2022**, *15*, 5008. https://doi.org/10.3390/en15145008

Academic Editor: Radu-Emil Precup

Received: 19 May 2022
Accepted: 27 June 2022
Published: 8 July 2022

1. Introduction

The climatic condition and the advancement of technology are the motivating factors behind energy management in residential buildings. In developing countries like India, electricity consumption is increasing in tandem with the country's economic development. It is expected to rise by 140% by 2021–2022, with Tamil Nadu, Telangana, and Karnataka accounting for nearly 80% of the increase [1]. As a result, making optimal use of power by consumers is a key step in reducing energy demand growth. An energy management control unit (EMU) is used in residential buildings to make sure that energy is used efficiently. This is done by properly monitoring, regulating, and optimizing energy usage.

Nowadays, the conventional electrical grid has been transformed into a smart grid (SG). The European Union Commission Task Force for Smart Grids has provided the smart grid definition as follows: "SG is an electricity network that can cost efficiently integrate the behavior and actions of all users connected to it—generators, consumers and those that do both—in order to ensure economically efficient, sustainable power system with low losses and high levels of quality, security, of supply and safety." Thus, the SG reduces energy waste and consumption costs and increases reliability, efficiency, and transparency of the energy supply. In the development of SG, demand-side management (DSM) is considered as an important feature in providing economic benefits to the consumer through controlling, monitoring, protecting, and optimizing the home appliance operation. DSM is also aimed at benefiting the utility control center by reducing the stress during peak hours. The utility control center implements the demand response (DR) program by bringing consumers into the picture in the process of energy management program. The DR can be divided into three categories: price-based or rate-based DR programs, incentive- or event-based DR programs, and demand-reduction-bid-based DR programs. This proposed work follows price-based DR programs that include real-time pricing (RTP), day-ahead pricing (DAP), time-of-use (TOU) pricing, and critical peak pricing (CPP) programs. These pricing schemes have played a vital role in attaining the monetary benefit for the smart home consumers [2–5].

Typically, a smart home is a part of smart grid (SG) and is defined as "A smart home is a residence incorporating a communications network between electric household appliances and services" [6]. A smart home consists of a real-time monitoring system that communicates with each device to optimize energy use. Optimization of energy consumption cost can be accomplished via stochastic optimization approaches with accurate probabilistic parameter estimation. Thus, real-time home energy optimization is the ideal solution, even when energy demand and consumption costs fluctuate.

Most of the research was carried out on optimization techniques to address the energy management problems for residential users: linear programming (LP), integer linear programming (ILP), mixed-integer programming (MIP), non-convex programming, mixed-integer linear programming (MILP), and non-deterministic polynomial-time hardness (NP-hardness) techniques. However, the computational time for these optimization algorithms is prohibitively long. On the other hand, evolutionary algorithms provide a fast and near-optimal solution to these problems [7–9].

In smart homes, the utility control center manually performs load shifting or sheds a particular load for a certain period of time through the existing electricity system to minimize peak formation during peak hours [10,11]. As a result, only the consumers benefit from such actions, not the utility control center. Furthermore, moving the load from peak to off-peak periods lowers peak demand and energy consumption costs, but it still outrages the user's satisfaction level. It should be mentioned here that there is always a trade-off between the user satisfaction level and energy consumption cost, and achieving both concurrently is the most difficult task [12]. Thus, to achieve them together, some of the major constraints like daily energy consumption (kW), peak-average ratio (PAR), energy price signals, and user satisfaction have to be considered. As a consequence of these challenges, effective energy management algorithms that can handle all sorts of loads and adapt to the uncertainties of energy prices are required [13].

In this regard, the authors of [14,15] have developed scheduling algorithms based on consumption cost reduction and consumer preference to manage residential appliances, which achieve the desired trade-off between economic benefits and consumer preference. Similarly, machine learning techniques, linear and dynamic programming, particle swarm optimization (PSO), fuzzy methods, and game theory are among the optimization techniques used in home energy management systems to schedule and control home appliances to provide economic benefits to consumers [16–21]. However, consumers are still not able to attain both user satisfaction and cost savings together, which are the drawbacks of the existing DR programs for DSM.

Recent literature suggests that home appliances can be categorized based on their operational behavior and energy consumption pattern as non-schedulable, schedulable, and controllable appliances to maximize the consumer satisfaction level and to attain the flexibility of scheduling [22–25]. The authors of [26] have presented the definition of energy management as a set of strategies and functions that can optimize energy use. These sets of strategies effectively balance the demand and supply. Energy management is the process of monitoring, controlling, and optimizing the energy usage in residential buildings. It efficiently optimizes energy consumption costs and minimizes the peak-average ratio.

The authors of [27,28] have suggested that the Harris Hawks Optimization (HHO) algorithm and the Water Cycle Algorithm (WCA) effectively minimize the overall power losses and maximize the load balance at the distribution network level. The authors of the Harris Hawks Optimization algorithm and the Water Cycle Algorithm have compared them with particle search optimization (PSO), the harmony search algorithm (HSA), the fireworks algorithm (FWA), the Cuckoo search algorithm (CSA), and the uniform-voltage-distribution-based constructive algorithm (UVDA). The authors claim that their algorithms are the best at improving the efficiency and sustainability of the distribution grid.

In [29], a hybrid optimization algorithm predicts the PV power generation by combining a convolutional neural network (CNN) and the salp swarm algorithm (SSA). This forecast is based on the weather (rainy, heavy cloudy, moderately cloudy, lightly cloud, and sunny). The CNN is applied to predict the next day's weather type, and the SSA technique is used to optimize each model. Thus, to enhance the SSA technique's exploring and exploiting capabilities, a simulated annealing mechanism is employed, which is based on symmetric perturbation for automated compliance checking in residential microgrids [30]. Therefore, residential microgrids and smart homes require an effective energy management unit that is capable enough to forecast and solve the microgrids' problems in advance and provide the ideal solution to balance the demand and supply. In [31], the authors have proposed a rainfall algorithm with TOU pricing to schedule the home appliances' operation through which the energy demand issues in residential buildings are predicted and optimized. As a part of the smart home/smart grid, electric vehicles (EV) can be used to balance the demand and supply. Mohammad et al. [32] have proposed an energy management unit for residential buildings with local PV power generation to maximize the user comfort, including the availability of EVs, PAR reduction, and minimize the energy consumption costs.

The authors of [33] present a distributionally robust optimization algorithm to optimally schedule the energy storage system that is integrated with a PV source. This problem has been presented as a two-stage programming model. The first stage reduces the energy consumption costs, and the second stage includes a real-time dispatch with a forecasted PV power output. With system uncertainties such as DC voltage fluctuation, disturbance from the utility grid system, and variation of the circuit parameters, traditional linear control methods cannot ensure the quality issues of the grid-connected inverter system. The authors of [34] propose a robust model predictive control (RMPC) technique that effectively schedules the battery energy storage system to minimize the total economic cost of multicarrier microgrids.

Thus, for effective energy management, this paper proposes a novel optimization algorithm with the Indian electricity pricing scheme to schedule consumers' demands. For the first time, the Indian DAP scheme was implemented along with the SSA, SSO and proposed RMSSO algorithms to reduce energy consumption cost and PAR. Timing and energy constraints were defined. Additionally, a variable was defined to ensure the user satisfaction level. The system was supported only by grid supply. The best sperm position was determined with the help of a remodeled inertia weight/constriction function. This paper employed qualitative and quantitative metrics to check and validate the correctness and accuracy of the proposed optimization algorithm. The proposed system used an effective communication technology to schedule energy demand in the most economical way.

Highlights and Organization of the Paper

The following features make this approach more distinct from existing DR algorithms.

(i). A remodeled sperm swarm optimization (RMSSO) algorithm was proposed for the HEM system.
(ii). The optimization process was carried out with varied computational parameters to demonstrate that the optimization algorithms could handle five distinct Indian climatic conditions.
(iii). A day-ahead pricing (DAP-(₹//kWh)) scheme was used as a part of the DR program.
(iv). This paper provides a unique comparison of SSO, modified SSO (MSSO), and the proposed RMSSO algorithms.
(v). Reduction in energy consumption costs, peak-average ratio (PAR), and increase in the level of user comfort were the objectives of this paper.

The remaining part of the paper is structured as follows: The proposed system architecture is explained in Section 2, and the mathematical formulation is described in Section 3. Section 4 discusses the proposed RMSSO algorithm. The simulation results, evaluation, and description of the experimental setup are presented in Section 5. Finally, this paper is concluded in Section 6.

2. System Architecture

The proposed system aimed to reduce energy consumption costs by collecting and evaluating all electricity-related data to provide an optimal solution. This system gives a simulation/hardware-based solution for controlling and monitoring the energy in a lab environment. All five scenarios (climates) were developed with a controller device for smooth integration and operation of home appliances. The proposed system comprised 16 appliances with different power ratings, each controlled by an individual relay switch actuated by the controller for every time slot. These appliances were divided into two groups (schedulable and non-schedulable) to simplify operation and improve consumer satisfaction. The setup includes incandescent lamps, a mixer, and a kettle. Table 1 shows the type of loads which are connected across each phase.

Table 1. Load details.

Phase	Load Type	Wattage	Total Number
R	Incandescent	200 W & 100 W	2 each (total 4 nos)
Y	Incandescent	60 W & 40 W	5 nos and 4 nos, respectively
B	CFL	9 W	3 nos

The HEMs laboratory setup consisted of a controller (Raspberry Pi 3B+), a smart meter (Schneider Conzerv EM6400NG-model-NHA2768503-0104/2018), RS485 communication modules that employ the MODBUS protocol and loads. The algorithm was developed on the Raspberry Pi, and its input/output options were enabled to connect to the internet through Wi-Fi communication to manage and monitor the appliances. The smart meter (gateway) was connected to the controller via an RS485 module. The IP modem (Four-Faith) and the control unit were Wi-Fi-enabled for remote access. A dynamic domain name system (DDNS) was used when the server IP was dynamic. The modem communicated with the smart meter using the same settings. A Python script read data from smart meter data registers using the pymodbus package [35] and stored it in a local database which was available in the Raspberry Pi.

The timestamp, active power, reactive power, apparent power, frequency, power factor, current, and voltage were recorded. The database got updated every second. Every minute, the controller calculated and recorded the average of all collected fields. If the smart meter failed to read or send the readings to the controller for more than 60 s, the controller sent an alert notification message. Several privacy and security standards are described in [36], and the proposed prototype system uses HTTPS for secure data transmission between local databases and cloud storage. The controller uploaded all collected data as a CSV file to the cloud storage (Thingspeak).

3. Mathematical Formulation

The scheduling process was modeled as a MILP problem that was addressed using problem formulation, decision variables, cost functions, and constraints, and described in the upcoming subsections.

3.1. Problem Formulation

In the scheduling process, sixteen appliances with different power ratings are considered, which are categorized into two groups: schedulable appliances (SA), and non-schedulable appliances (NSA). Table 1 lists the power rating of appliances used in scheduling. The appliance's operating duration is split into equal intervals of time (1 h each slot) of the day (K) as shown in Equations (1) and (2).

$$K = k_1, k_2, k_3, \ldots, k_n \tag{1}$$

where K denotes the cumulative number of time intervals in a day (24 h), k_n represents the nth time interval, and n is the number of time intervals per day = $(1, 2, 3, \ldots, N)$ and is as follows,

$$n = \frac{24\ hour\ of\ the\ day}{no.\ of\ intervals} = \frac{24\ hour\ of\ the\ interval}{24\ intervals} = 1\ hour\ per\ interval \tag{2}$$

The group of appliances that are considered for scheduling is denoted as G, and this is a combination of both schedulable and non-schedulable appliances, as shown in Equation (3).

$$G = g_1, g_2, g_3, \ldots, g_n \tag{3}$$

where $g_1, g_2, g_3, \ldots, g_n$ specifies the individual appliance.

3.2. Binary Decision Variable

The binary variable $\left(bin_{g_n,j}^{k_n}\right)$ is formulated to determine whether the considered set of schedulable and non-schedulable appliances is in an ON or OFF condition, as shown in Equation (4).

$$bin_{g_n,j}^{k_n} \in 0,\ 1 \tag{4}$$

If $bin_{g_n,j}^{k_n} = 1$ for appliance g_n, the jth set has been scheduled in the time interval k_n. An additional binary decision variable $\left(bin_{Cg_n,j}^{k_n}\right)$ is defined to determine whether a specific jth set of appliances has finished their operation by the time k_n, and the same is given in Equation (5).

$$bin_{Cg_n,j}^{k_n} = 1 \tag{5}$$

As discussed in earlier sections, the aim is to reduce the total electricity usage costs. The total energy consumption costs are computed using Equation (6) with a 24 h electricity tariff: day-ahead price (DAP-(₹/kWh)) [37].

$$\sum_{k=1}^{24} \pi_{k_n}^{price} \left(\sum_{g=1}^{g_n} \sum_{j=1}^{n_{g_n}} P_{Total}^{\mathrm{ON}} \right) \tag{6}$$

where $\pi_{k_n}^{price}$ represents the electricity tariff for the respective time slots (k_n), and the total energy consumed by all appliances (P_{Total}^{ON}) on the particular day is determined using Equation (7).

$$P_{Total}^{ON} = P^{NSA^{ON}} + P^{SA^{ON}} \tag{7}$$

Equations (8) and (9) describe the amount of energy consumed by schedulable and non-schedulable appliances scheduled in the appropriate time interval (k_n).

$$P^{NSA^{ON}} = \sum_{k=1}^{24} \left(\sum_{g=1}^{G_{NSA}} P_{g_n k_n}^{NSA} \right) = \left[P_{g_1 k_1}^{NSA} + P_{g_2 k_2}^{NSA} + \ldots + P_{g_n k_{24}}^{NSA} \right] \tag{8}$$

$$P^{SA^{ON}} = \sum_{k=1}^{24} \left(\sum_{g=1}^{G_{SA}} P_{g_n k_n}^{SA} \right) = \left[P_{g_1 k_1}^{SA} + P_{g_2 k_2}^{SA} + \ldots + P_{g_n k_{24}}^{SA} \right] \tag{9}$$

where G_{SA} *and* G_{NSA} are the sets of SA and NSA appliances, $P^{NSA^{ON}}$ and $P^{SA^{ON}}$ are the power consumed by non-schedulable and schedulable appliances, respectively. P^{ON}_{Total} is the total power consumption of all appliances.

3.3. Constraints

Two types of constraints are described in this section: timing and energy constraints.

3.3.1. Timing Constraints

1. Non-schedulable appliance: As specified in Equation (10), non-schedulable appliances must remain *ON* throughout the day (24 h), regardless of whether it is a peak time or not.

$$\sum_{k=1}^{24} K = \left(\left|G^{NSA^{ON}}\right|\right) \tag{10}$$

2. User satisfaction level: The significant constraint is that all appliances must be run for a specified number of times $(T_{app_x k_n})$, as defined in Equation (11).

$$\sum_{k=1}^{24} T_{app_x k_n} = t_{appN} \tag{11}$$

where $T_{app_x k_n}$ refers to the total number of times that an appliance has to be operated. The appliance type (SA and NSA) is indicated as $appN$. An appliance is operated for the desired number of times per day. The frequency of appliance operation (provided by the consumer) is referred to as app_x, which guarantees that all appliances are operated the required number of times, as shown in Equation (12).

$$\sum_{k=1}^{n} t_{app_x,k} + t_{app_{x'} k_n+1} + t_{app_{x'} k_n+2} + \ldots + t_{app_{x'} k_n+(N-1)} = T_{app_{x'} k_n} \tag{12}$$

In addition, constant for the availability of appliances, three schedulable appliances, such as ($A02$, $A05$, $A08$), are considered unavailable during the time slots k_1, k_2, k_3, respectively, to reduce peak demand issues, as given in Equation (13).

$$x(appN, k_n) \in \{0\} \tag{13}$$

3. Time constraint: The appliance operating time constraint specifies the scheduled time interval of the jth set of appliances as given in Equation (14).

$$bin^{k_n}_{g_n, j} \le K^{k_n}_{P_{g_n}} \qquad \forall g_n, j,\ k_n \tag{14}$$

where $K^{k_n}_{P_{g_n}}$ is the scheduled time interval of the jth set of appliances.

3.3.2. Energy Constraints

1. Energy consumption threshold limit (E_{max}): For any time interval of the day, the total power consumed by both schedulable and non-schedulable appliances (P^{ON}_{Total}) must be less than or equal to the threshold limit, E_{max} = 1.2 kW, as stated in Equation (15).

$$P^{ON}_{Total} \le\ E_{max} \tag{15}$$

After the scheduling process, if the condition $P^{ON}_{Total} < E_{max}$ still exists, consumers can turn ON additional appliances (part of schedulable loads which are not involved in the scheduling for the specific time interval). This condition takes care of the consumers' satisfaction level. Then, the total power consumption is as follows:

$$P_{Total}^{ON'} = P^{NSA^{ON}} + P^{SA^{ON}} + P^{additional^{ON}} \tag{16}$$

where $P^{additional^{ON}}$ is the additional appliances' power consumption. If $P_{Total}^{ON} > E_{max}$ or $P_{Total}^{ON'} > E_{max}$, the additional appliance that the user desires to operate is not feasible to run. In continuation of this, *NSA* minimal demand (E_{min}) is stated in Equation (17).

$$E_{min} \leq P_{Total}^{ON} \leq E_{max} \tag{17}$$

where E_{min} and E_{max} represents the lower and upper power limitations to the *j*th set of appliances, respectively.

2. Total energy consumption: This constraint (Equation (18)) is imposed to guarantee that, through scheduling, the required overall energy demand ($E_{g_n,j}$= 10 kW) of the day has been met.

$$\sum_{k=1}^{24} P_{Total}^{ON} = E_{g_n,j} \qquad \forall g_n, j, k_n \tag{18}$$

3. Peak-average ratio (PAR): The peak-average ratio is an important factor to consider while scheduling and must be mitigated. It is computed using Equation (20).

$$P_{peak} = max\left(\sum_{t=1}^{24}\left(P^{ON}(t)\right)\right) \text{and} P_{avg} = \frac{P_{Total}^{ON}}{D} \tag{19}$$

$$PAR = \frac{P_{peak}}{P_{avg}} \tag{20}$$

where P_{peak} and P_{avg} are the maximum power and average power of the day, respectively.

3.3.3. Objective Function

After meeting the above-stated constraints, the optimal energy consumption cost can be attained and consequently, the objective function (*OF*) of this paper is as defined in Equation (21).

$$OF = min\left(\sum_{k=1}^{24} \pi_{k_n}^{price}\left(\sum_{g=1}^{g_n}\sum_{j=1}^{n_{g_n}} P_{Total}^{ON}\right)\right) \tag{21}$$

where P_{Total}^{ON} is the total power consumption of all scheduled appliances in the specific time interval. $\pi_{k_n}^{price}$ indicates the cost of energy consumption. As a consequence, Equation (21) is the optimization model of MILP. It can be efficiently solved using metaheuristic methods such as SSO, SSA, and the proposed RMSSO algorithms.

4. Optimization Techniques

A novel remodeled sperm swarm optimization (RMSSO) algorithm was proposed for home energy optimization. For the first time, an Indian home environment was taken as the case study to solve this optimization problem using the salp swarm optimization (SSA) and sperm swarm optimization (SSO) algorithms. All three algorithms were developed using the Python/GUROBI tool without integrating renewable energy sources.

4.1. Salp Swarm Optimization Algorithm (SSA)

The salp swarm optimization algorithm (SSA) mimics the salps group (family of Salpidae) foraging mechanism in the oceans. Salps are intelligent and have translucent barrel-shaped bodies, similar to those of jellyfish. A salp chain is a network (swarm) formed by a group of salps in the deep sea [38].

The formation of swarms (salp chain) evolves into a hierarchy with a leader and followers. The first salp in the chain is the leader, while the rest of the salps are followers. The leader salp communicates either directly or indirectly with the follower salps using search directions. The positions of all salps are defined as $X_j^d = \left[x_1^1, y_1^1, \ldots, x_m^n, y_m^n\right]$, where $d = 1, 2, 3, ..n$; n is the number of salps; and $j = 1, 2, 3, ..m$; m is the number of variables. The SSA assumes that the food source (F) is available in the search area as a target. The position of leader salp is defined as in Equation (22), with three components: direction, personal best, and the team's best.

$$X_j^1 = \begin{cases} F_j + \lambda_1\left((u_j - l_j)\lambda_2 + l_j\right), & \lambda_3 \geq 0 \\ F_j - \lambda_1\left((u_j - l_j)\lambda_2 + l_j\right), & \lambda_3 < 0 \end{cases} \tag{22}$$

where X_j^1 denotes the leader salp's position at the jth dimension in the search space, F_j is the position of food at the jth dimension in the search space, $\lambda_1, \lambda_2, \lambda_3$ are the coefficient factors, u_j and l_j indicate the upper and lower bounds of the jth dimension. The significant controlling coefficient parameter of this algorithm is λ_1. During the optimization process, this parameter decreases as the iteration count increases, allowing the SSA technique to explore more in the early phases and then exploit it extensively in the search space. The coefficient factors λ_2 and λ_3 are consistently generated random numbers, where λ_2 is in the range of $[0,\ 1]$, which is responsible for widening the search space. λ_3 indicates whether the next positions of the current leader salp and follower salps are within the boundary or not. If $\lambda_3 < 0.5$, the salps are moving out of the boundary on a negative scale, while if $\lambda_3 \geq 0.5$, salps are travelling towards the direction of food on a positive scale. The boundary for λ_3 is fixed as 0.5 to ensure equal weightage is given to the salps travelling in both forward and reverse directions. Thus, λ_2 and λ_3 help decide the next position of the salp in the jth dimension of the search space. Additionally, the coefficient factors λ_1, λ_2, and λ_3 are used to reposition the solution that goes outside the search space. The position of salp followers is updated using Equation (23).

$$x_j^d = \frac{1}{2}\left(X_j^d + X_j^{d-1}\right) \tag{23}$$

where x_j^d represents the position of the dth follower in the jth dimension in the search space. Equation (24) shows how to bring the salp back into the search space.

$$X_j^d = \begin{cases} l_j, & if X_j^d \leq l_j \\ u_j, & if X_j^d \geq u_j \\ X_j^d, & otherwise \end{cases} \tag{24}$$

The SSA optimization is started by initializing the salps in a random position. Consequently, the fitness of every single salp is determined based on the distance between the food source (F) and the salp. For each dimension, with the help of coefficient factors $(\lambda_1, \lambda_2$, and $\lambda_3)$, the positions of both leader and follower salps are updated frequently. X_j is considered as the optimum load scheduling for the cost-saving of a day (24 h). The salp chain exploits the search space to get the most appropriate global optimum solution and avoids the local optimum.

4.2. Sperm Swarm Optimization Algorithm

This section discusses the sperm swarm optimization (SSO) algorithm. The SSO mimics the sperm navigation and mobility to inseminate the ovum. The swarm swims from the cervix, a low-temperature region, to the fallopian tubes, a high-temperature region, where the egg waits to be fertilized by the swarm. Sperm migration has to meet constraints, like pH value and temperature inside the reproductive system. Finally, one sperm penetrates and fertilizes the egg, which is referred to as the winner [39]. The symmetrical side stroke of the sperm swarm is shown in Figure 1.

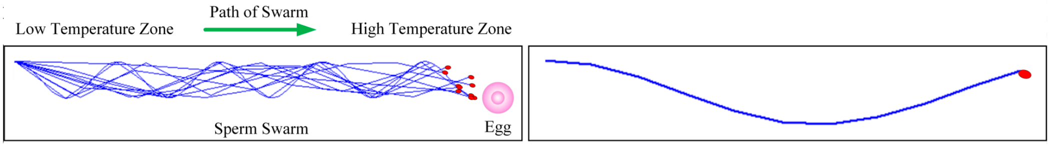


Figure 1. Symmetric side strokes of sperm swarm and the egg.

The sperm swarm optimization algorithm reaches an optimal solution by updating the individual's current position. This is achieved by determining the initial velocity of sperm ($Initial_{Velocity}$), the distance between the current position and $sb_{solution}$ (the sperm's best solution achieved so far), and $sgb_{solutioni}$ (the sperms' global best position). The mathematical model of SSO can be represented as,

$$V_i\ (t) = Initial_{Velocity} + Current_{Bestposition} + Global_{Bestposition} \tag{25}$$

where $V_i(t)$ is the velocity of sperm i ($i = 1, 2, 3, \ldots, N$; N is the maximum sperm counts) at iteration t. The initial velocity, personal best position, and global best position of the sperms are determined using Equations (26)–(29).

$$Initial_{Velocity} = D \cdot V \cdot log_{10}(pH_{Rand1}) \tag{26}$$

$$Current_{Bestposition} = log_{10}(pH_{Rand2}) \cdot log_{10}(Temp_{Rand1}) \cdot (sb_{solution} - current_i) \tag{27}$$

$$Global_{Bestposition} = log_{10}(pH_{Rand3}) \cdot log_{10}(Temp_{Rand2}) \cdot (sgb_{solutioni} - current_i) \tag{28}$$

$$current_i = current_i + V_i \tag{29}$$

where D is the velocity (inertia weight) damping parameter, a random number between 1 and 0 that is used to control and regulate the sperm velocity. pH_{Rand1}, pH_{Rand2}, and pH_{Rand3} are the random numbers of the pH value that varies between 7 and 14. The $Temp_{Rand1}$ and $Temp_{Rand2}$ are the random temperatures that range between 35.1 and 38.5 °C.

SSO uses an inertia weight that converges to an optimal value over the course of iteration which is represented as D in Equation (26). There are two linear methods of determining the inertia (constriction) weight which are given in Equations (30) and (31).

$$D_{i+1} = D_{max} - i \times \frac{D_{max} - D_{min}}{i_{max}} \tag{30}$$

$$D_{i+1} = \Delta_D\ D_i \tag{31}$$

where i_{max} is the maximum iteration, i is the current iteration, D_{max} *and* D_{min} are the upper and lower limits of the velocity damping parameter. ΔD denotes the random value that varies between 1 and 0. The second linear (dynamic) method outperforms the first method in terms of convergence rate.

4.3. Remodeled Sperm Swarm Optimization Algorithm (RMSSO)

Constriction Coefficient of the Sperm Swarm Optimization Algorithm

As discussed in Section 4.2, the authors of [39] have presented the sperm swarm optimization using the linear Equation (31) that results in sperm convergence over the course of iterations. That is, the magnitude of sperm decreases linearly (adaptively) as it focuses more on the local and previous best points [40]. For low-dimensional optimization problems, this algorithm converges to the optimal point over a period of time. However, when dealing with high-dimensional and complicated energy optimization problems, this SSO is not capable of giving a promising solution [41]. Therefore, to improve the global search capability of the SSO algorithm, the constriction coefficient of SSO was restructured and proposed as a remodeled SSO algorithm in this paper.

For the past 350 years, researchers believed that sperm travels to the egg by swimming, as discussed in Section 4.2. Gadêlha et al. [42] has found that the sperm actually swims by spinning in a helical shape, as shown in Figure 2. Researchers have developed various optimization algorithms using different spiral trajectories such as Archimedes spiral, Cycloid spiral, Epitrochoid spiral, Hypotrochoid spiral, Logarithmic spiral, Rose spiral, Inverse spiral, and Overshoot spirals [43]. Despite all these trajectories being fast in computation, due to inadequate exploration of the search space, this technique eventually converges to the local optimum value [44,45]. It is worth mentioning here that the most commonly used spiral trajectory is the logarithmic spiral. The logarithmic spiral is also referred to as an equiangular or growth spiral, owing to the fact that the spiral distance increases with the number of iterations. In metaheuristic techniques, the unique processes of engendering a logarithmic spiral have been realized as an effective search behavior [46–48].

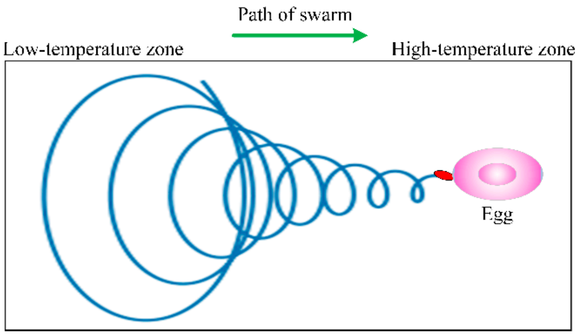


Figure 2. The sperm swarm helical spin in the fallopian tubes.

In the proposed remodeled sperm swarm optimization algorithm, the constriction mechanism allows the algorithm to find the best solution by continuously avoiding the trapping of local solutions. The mathematical model of sperm's helical shape movement is given in Equation (32). This equation represents the nature of helical movement of the sperm swarm. The radius (D-inertia weight) of the sperm movement is decreased as the iteration count increases.

$$D_{l+1} = 2e^{-\left(\frac{4.m}{M}\right)^2} \quad (32)$$

where m and M are the current and maximum number of iterations, respectively. Thus, the inertia parameter is adaptively decreased from 1 to 0. Figure 3 illustrates the RMSSO algorithm's search process from diversification to intensification, and the flow chart of the proposed RMSSO algorithm is illustrated in Figure 4.

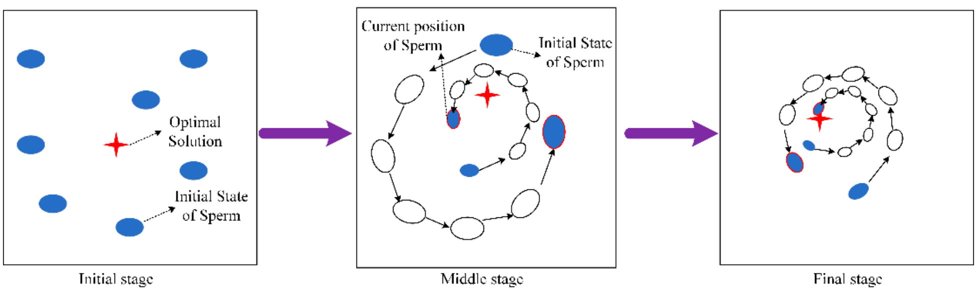


Figure 3. Illustration of the RMSSO algorithm's sperm swarm and the winner process.

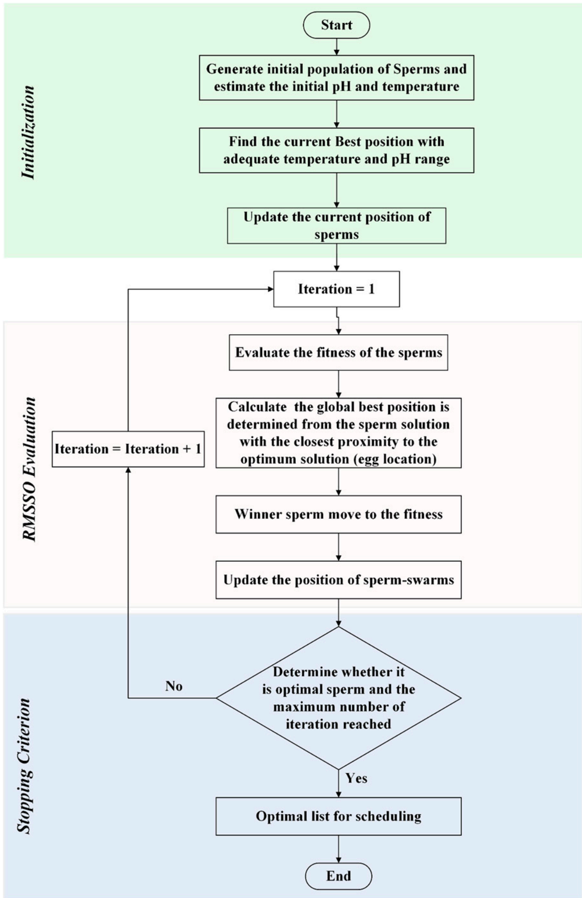


Figure 4. Flowchart of the proposed RMSSO algorithm.

4.4. Comparison Optimization Algorithms Inspired by the Sperm Swarm

A unique comparison of optimization algorithms inspired by the sperm swarm is made and illustrated in Table 2.

Table 2. Comparison of optimization algorithms inspired by the sperm swarm.

Comparison Standards	Algorithms		
	Sperm Swarm Optimization (SSO)	**Modified Sperm Swarm Optimization (MSSO)**	**Remodeled Sperm Swarm Optimization (RMSSO)**
	[39], 2018	**[41], 2021**	**Proposed**
Metaphor type	Nature-inspired approach, mimics the motility of sperm swarms during the fertilization process.		
Flow of sperm and control parameter	Swim stroke and velocity damping coefficient function.	Swim stroke and chaotic velocity damping coefficient function.	Helical stroke and the inertia weight decreased exponentially.
Type of approach	It continuously updates the swarm's position and velocity.		
Fitness value	Use the optimal value of the winner sperm as a reference value to adjust the velocities of the remaining sperms in the swarm.		
Impact of sperm swarms on the solution	Linear	Nonlinear	Nonlinear
Results	Local Optimum	Global Optimum	Global Optimum

5. Results and Discussion

An efficient swarm-based optimization technique balances both local and global search [49]. A suitable balance between these two processes can approximate an optimized home energy management system. If algorithms pay more attention to local search, a solution will quickly converge to an optimum point and get trapped at a local optimum.

A larger search coefficient (constriction coefficient) on the global search helps to avoid the local optimum solutions, but it takes more computation time to achieve the global optimum solution. The SSA technique uses the first salp (leader) in the swarm chain to balance the local and global search; the global search coefficient (constriction coefficient) decreases with an increase in the number of iterations. Hence, the SSA faces difficulty in achieving a proper balance between local search and global search [30]. In the SSO algorithm, each sperm optimizes its position by considering its location, velocity, the local best solution, and global best solution. However, this algorithm cannot converge at a global minimum and is trapped in local optima and faces premature convergence in complex problems [41]. Thus, in terms of exploration and exploitation, both SSA and SSO algorithms have enough exploration ability, but their exploitation ability is comparatively low.

Therefore, in this paper, a remodeled SSO (RMSSO) algorithm is proposed with an effective modification (Equation (32)) that improves the diversity of solutions by each sperm and keeps the proper balance between local search and global search during optimization. Thus, the searching ability of RMSSO was better than that of SSO and SSA for all five climates. This is the reason why the proposed RMSSO outperformed SSA and SSO consistently in all conditions.

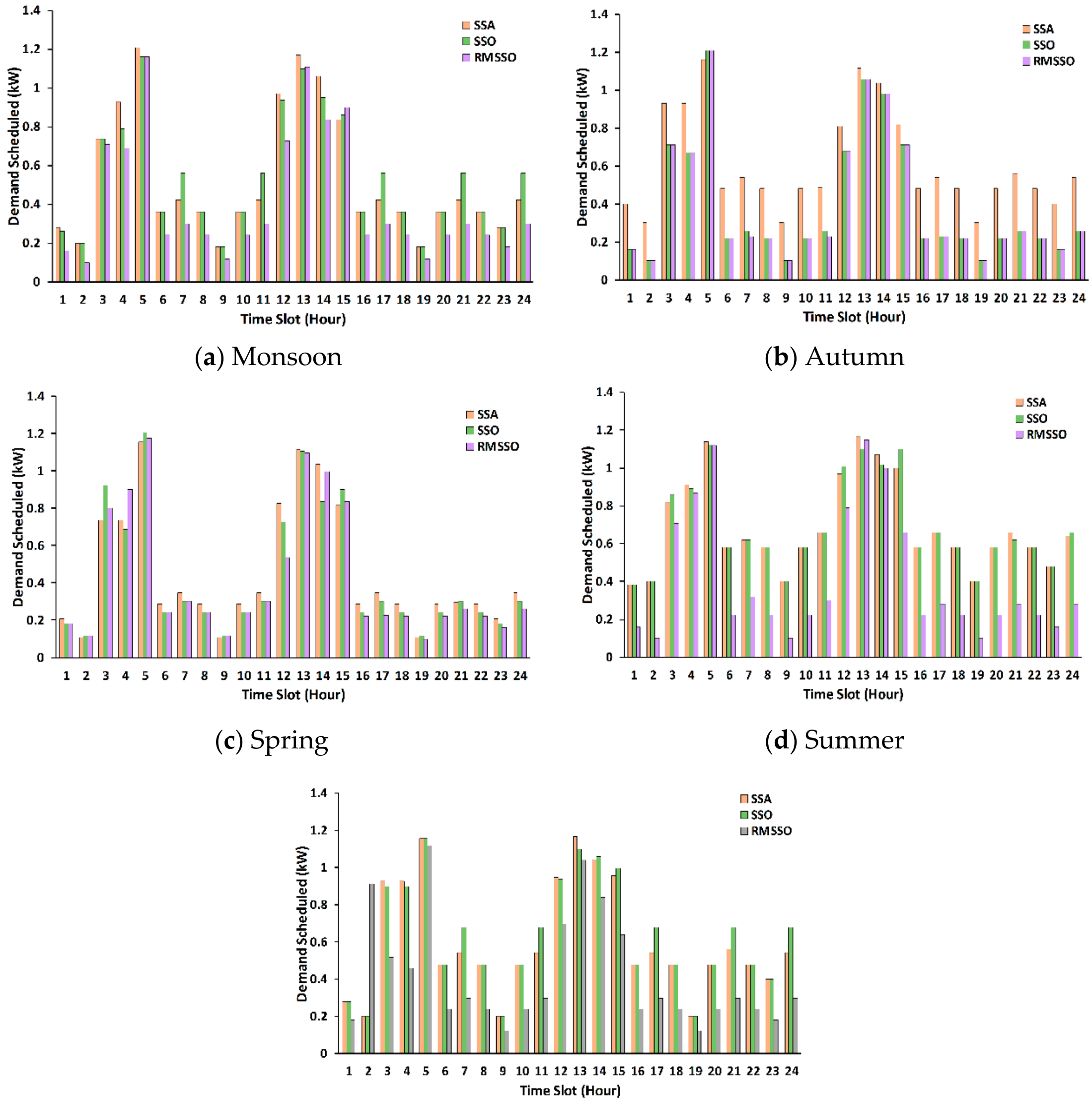
5.1. Simulation Results

The simulation results achieved using SSA, SSO, and the proposed RMSSO algorithms under the day-ahead pricing scheme are compared in this section. A simulation was carried out for sixteen different appliances with a total demand of 10 kW under five climatic conditions and constraints using the Python/GUROBI tool. The specifications of the system used were as follows: processor Intel (R) Core (TM) i3-7020 U CPU @ 2.30 GHz; 12.0 GB RAM; 64-bit operating system type; x64-based processor. All algorithms scheduled the load demand as shown in Figure 5a–e, which are discussed in the upcoming sections. Table 3 illustrates the simulation parameters.

Table 3. Simulation parameters of SSA, SSO, and RMSSO techniques.

Parameters	Values
$pH1$, $pH2$, $pH3$	7 to 14
$Temp1$, $Temp2$	35.1 to 38.5 °C
Iterations	50
$\mathrm{E}_{\mathrm{max}}$	1.2 kW
$E_{g_n,j}$	10 kW
D_{Max}	0.9
D_{Min}	0.2
dim	24
λ_2 and λ_3	0 to 1

DAP hourly price for five climatic conditions was taken from the Indian Energy Exchange, and the same is represented in Table 4 [37]. Recent research [50–54] has proved that both TOU and DAP pricing schemes reduce the energy consumption cost and the peak-average ratio. Hence, this paper considered the DAP scheme as the preferred option to use. The highlighted price in Table 4 represents the peak price of the day.



(**a**) Monsoon (**b**) Autumn

(**c**) Spring (**d**) Summer

(**e**) Winter

Figure 5. Demand comparison of SSA, SSO, and RMSSO techniques.

Table 4. Day-ahead price—Indian scenario.

Hours	DAP Tariff in ₹ (INR)				
	Monsoon	Autumn	Spring	Summer	Winter
	27 August 2020	25 September 2020	11 March 2021	1 May 2021	25 December 2021
0–1	2.51985	2.18985	3.71946	4.90168	2.02185
1–2	2.25144	2.42961	3.36554	4.00521	1.92394
2–3	2.13724	2.32931	3.28878	3.44357	1.80898
3–4	2.01204	2.32506	3.23159	3.16281	1.78339
4–5	2.0372	2.38732	3.29752	3.40298	1.9668
5–6	2.1493	2.43325	3.91974	3.44873	2.15122
6–7	2.55065	2.50596	4.87516	3.24231	2.81923
7–8	2.52455	2.43629	5.30393	2.78748	4.27534

Table 4. *Cont.*

Hours	DAP Tariff in ₹ (INR)				
	Monsoon	Autumn	Spring	Summer	Winter
	27 August 2020	25 September 2020	11 March 2021	1 May 2021	25 December 2021
8–9	2.42361	2.42647	5.22559	2.7634	5.42572
9–10	2.5095	2.54149	6.2124	3.00443	5.70012
10–11	2.49729	2.42869	4.74233	2.74779	5.19892
11–12	2.32273	2.47529	3.91527	2.74951	4.2254
12–13	2.30237	2.45759	3.55424	2.7469	3.53248
13–14	2.31985	2.40138	3.16004	2.79346	3.01789
14–15	2.43494	2.44749	3.33993	2.89731	2.8544
15–16	2.46284	2.54099	3.80886	3.03256	3.04502
16–17	2.48485	2.73988	5.26423	2.99972	3.93748
17–18	2.19728	2.77925	4.68741	2.86562	4.05037
18–19	2.54438	3.52344	5.2987	2.9462	4.80026
19–20	3.6701	4.30019	5.28742	3.52369	3.64377
20–21	3.80249	3.79221	4.69549	3.37001	2.9043
21–22	3.79318	3.46787	4.11488	3.6675	2.62675
22–23	2.81023	3.14441	3.8067	3.7993	2.10085
23–24	2.34433	2.87783	3.60389	3.84698	2.16325

In the monsoon, spring, and summer seasons, one non-schedulable appliance is taken into account, and in the autumn and winter seasons, two non-schedulable appliances were considered for 24 h operation. Table 5 represents the number of times that an appliance is to be operated to ensure that a specific appliance task is completed, and the same is achieved 100% by satisfying the constraints provided in Equations (12)–(19). The highlighted numbers in Table 5 represent the number of times the appliance (non-schedulable) must be operated.

Table 5. Number of times that an appliance has to be operated.

Appliances	Number of Times That an Appliance Is to Be Operated				
	Monsoon	Autumn	Spring	Summer	Winter
A01	21	3	5	24	12
A02	5	6	5	23	20
A03	5	3	5	20	24
A04	5	3	5	5	24
A05	5	3	5	7	20
A06	5	7	5	21	7
A07	24	21	19	5	7
A08	23	6	23	5	3
A09	5	4	21	5	3
A10	18	24	24	5	3
A11	9	5	8	5	3
A12	5	19	5	5	3
A13	5	9	5	5	3
A14	5	3	5	5	7
A15	5	10	5	5	7
A16	5	24	5	5	4

5.1.1. Demand Comparison

Figure 5a–e shows the comparison of load scheduling using the SSO, SSA, and proposed RMSSO techniques for individual time slots. The total load request by the consumer during the day was limited to the threshold of 10 kW to avoid peak demand issues. In Figure 5a, (monsoon) the t_{20}, t_{21}, and t_{22} time slots with high electricity prices shows that both SSO and SSA algorithms had scheduled the load of 0.4 to 0.6 kW, whereas RMSSO scheduled 0.2 to 0.3 kW, which is less than the load scheduled by SSO and SSA. Similarly, during the t_{23} slot, SSO and SSA scheduled 0.3 kW of demand. However, RMSSO scheduled the appliances that had a total load of 0.2 kW. Note that the proposed RMSSO distributed the appliance across all time intervals, resulting in a reduced energy consumption cost without exceeding the threshold limit of 1.2 kW.

Likewise, in the remaining climatic conditions (autumn, spring, summer, and winter), the proposed RMSSO effectively scheduled load in all time slots in comparison to other techniques, as illustrated in Figure 5b–e. Consequently, this comparison reveals that the proposed RMSSO algorithm scheduled the appliance with a reduced cost of energy consumption and regulated peak demand, which is a benefit for both the utility and the consumer.

5.1.2. Total Load Comparison

Figure 6 presents the total load comparison results of all three algorithms. The proposed RMSSO scheduled its load at 10 kW in all five climatic conditions. Considering the autumn season, both SSO and RMSSO scheduled the loads with 9.43 and 10 kW respectively, whereas SSA scheduled the load with 14.5 kW. In the summer season, both SSA and SSO scheduled the load of 16.43 kW each, which was greater than the total load requirement (10 kW) for the day. This excess load scheduling causes concerns such as power loss, increased peak demand, and the high cost of energy consumption. The RMSSO algorithm covered the total demand of 10 kW, which indicated that 100% of the consumer's maximum demand was satisfied.

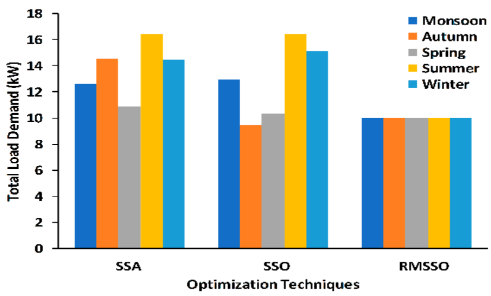
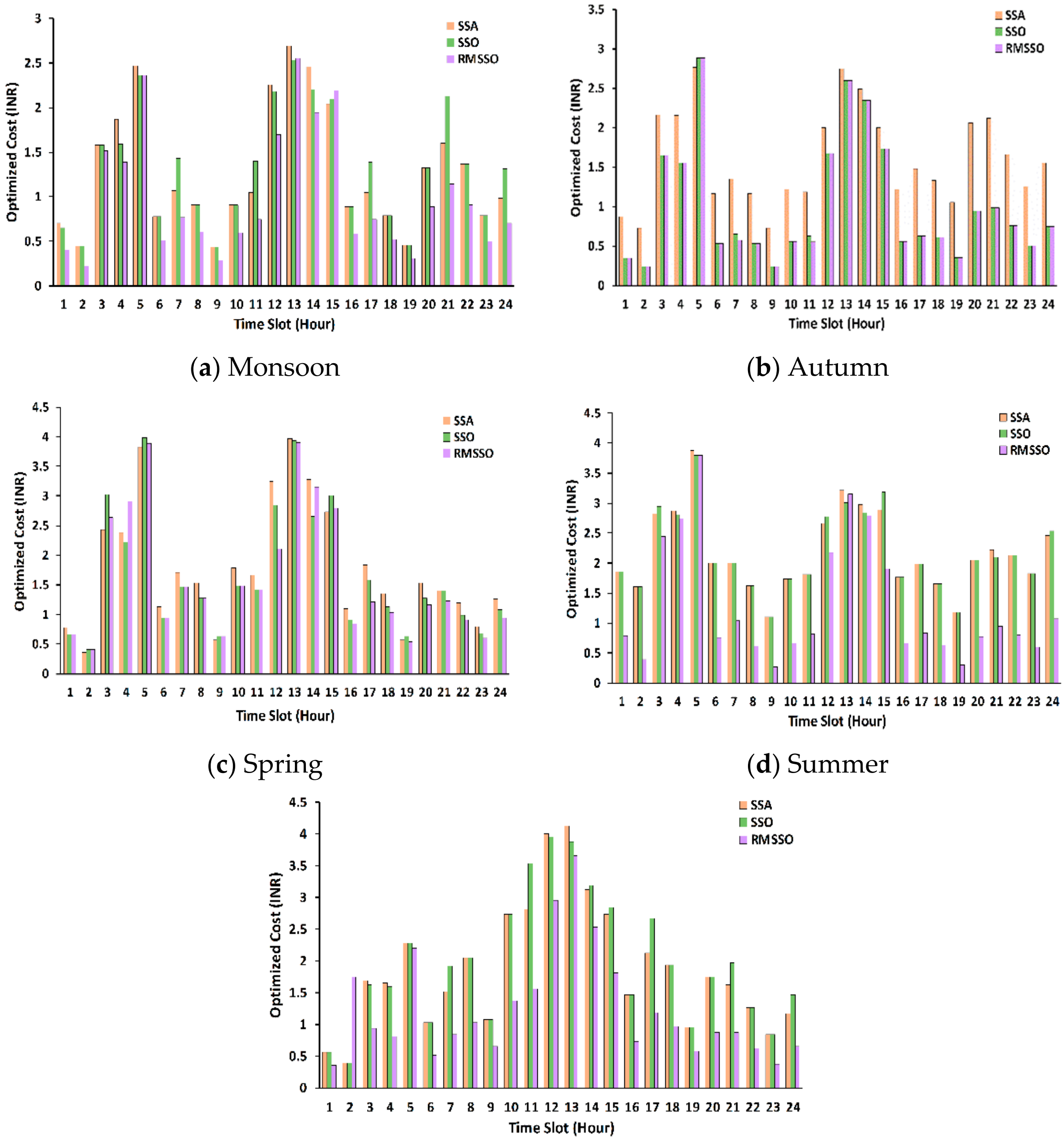


Figure 6. Comparison of total scheduled loads.

5.1.3. Cost Comparison

Figure 7a–e shows the cost comparison of RMSSO with those of SSA and SSO techniques for each time slot. Considering Figure 7a with day-ahead pricing (monsoon), the RMSSO algorithm achieved energy consumption costs lower than those of SSO and SSA between 0.90₹ and 1.15₹ during the high-cost time (t_{20}, t_{21}, and t_{22}) slots. At the same time, the SSA and SSO techniques scheduled the load between 1₹ and 2.25₹. In comparison, the RMSSO energy consumption costs were lesser than those of the SSA and SSO algorithms in all the time slots. Figure 7b shows the individual time slot cost comparison in the autumn season. Notably, in the peak period (t_{19} to t_{23}), the SSA showed maximum energy consumption cost, and SSO and RMSSO showed around 0.5₹ to 1₹. RMSSO scheduled with the energy consumption cost of 2.5₹ in the t_5 time slot, which was higher than the load scheduled by the SSA technique, indicating that RMSSO scheduled higher demand than the other techniques did in these slots. Nonetheless, even in the other climates (Figure 7c–e), the proposed RMSSO algorithm outperformed the other two algorithms in saving the energy consumption cost.



(**a**) Monsoon (**b**) Autumn

(**c**) Spring (**d**) Summer

(**e**) Winter

Figure 7. Individual time slot cost comparison of SSA, SSO, and RMSSO techniques.

5.1.4. Total Cost Comparison

The total cost comparison of the proposed RMSSO with those of SSA and SSO is given in Figure 8. Considering the monsoon season, the RMSSO energy consumption cost was 24.11₹ which was less than those of the SSA and SSO techniques, i.e., 30.88₹ and 31.91₹, respectively. Similarly, during the autumn season, SSO and RMSSO attained the lowest costs of 24.30₹ and 24.15₹, respectively, compared to that of the SSA technique. Accordingly, the proposed RMSSO stood ahead of SSO and SSA techniques. Table 6 provides a detailed comparison of the total energy consumption cost and load scheduled by each technique.

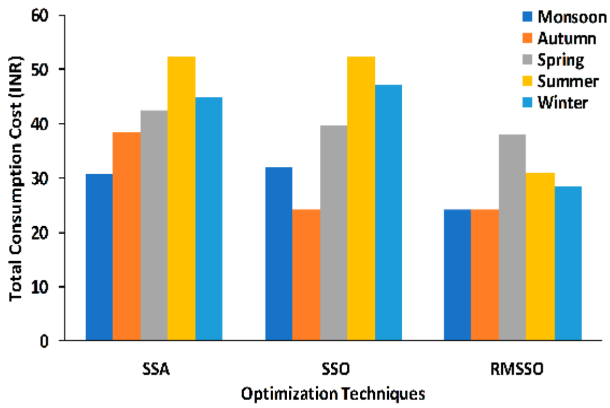


Figure 8. Total cost comparison.

Table 6. Cost for energy consumption and load scheduled.

	Seasons	Techniques		
		RMSSO	**SSO**	**SSA**
Total Cost (₹)	Monsoon	24.11	31.91	30.88
	Autumn	24.15	24.3	38.52
	Spring	38.11	39.66	42.43
	Summer	31.01	52.34	52.33
	Winter	29.87	47.03	44.94
Load Scheduled (kW)	Monsoon	10	12.95	12.65
	Autumn	10	9.43	14.52
	Spring	10	10.29	10.89
	Summer	10	16.43	16.43
	Winter	10	15.08	14.49

5.1.5. Task Completion Analysis

From Table 7, it is found that in the monsoon season, the proposed RMSSO algorithm completed 100% of the task (10 kW), while the SSO and SSA techniques completed 129.5% and 126.5%, respectively. In the same scenario, the percentage cost difference of SSO and SSA from RMSSO was 24.44% and 21.92%, respectively. The proposed RMSSO completely satisfied the demand (100%) for the autumn season, and only 94.3% of the demand was satisfied by the SSO algorithm. This shows that SSO could not meet the total demand requirement. In the same scenario, the SSA technique finished 145.2% of the task. The percentage of cost difference between SSO and SSA from RMSSO for the same autumn season was 0.617% and 37.30%, respectively. Further, while considering the spring season, RMSSO satisfied 100% of the demand required. However, the SSO and SSA techniques were completed at 102.9% and 106.9%, respectively. Additionally, the percentage of cost difference between SSO and SSA from RMSSO in the same season was 3.908% and 10.188%, respectively.

During the summer season, RMSSO met 100% of the required demand, whereas the SSO and SSA approaches met 164.4% and 164.3% of the requirement, respectively. The percentage cost difference of SSO and SSA from that of RMSSO in the summer season was 40.75% and 40.74% respectively. Similarly, RMSSO completed the required demand of 100% in the winter season, while the SSO and SSA techniques accomplished demand of about 150.8% and 144.9% respectively. While considering the percentage cost difference of SSO and SSA techniques from the proposed RMSSO algorithm, both SSO and SSA techniques scheduled their demand with high cost with a difference of 36.48% and 33.53%

respectively. It was because the SSO and SSA techniques were trapped in local optima solutions, whereas RMSSO attained the global optimal solution. Similarly, the comparison of remaining climatic conditions also proved that 100% of the tasks were completed at the lowest response time with maximum consumer satisfaction by the proposed RMSSO algorithm. Meanwhile, both SSO and SSA techniques scheduled the loads at higher energy consumption costs. Load scheduling by SSO and SSA techniques led to power loss, increased peak demand, and a high cost of energy consumption.

Table 7. Percentage cost difference and task completion comparison.

	Seasons	Techniques		
		RMSSO	**SSO**	**SSA**
Percentage of cost difference from RMSSO (%)	Monsoon	-	24.44	21.92
	Autumn	-	0.617	37.30
	Spring	-	3.908	10.18
	Summer	-	40.75	40.74
	Winter	-	36.48	33.53
Percentage of Task Completion (%)	Monsoon	100	129.5	126.5
	Autumn	100	94.3	145.2
	Spring	100	102.9	108.9
	Summer	100	164.4	164.3
	Winter	100	150.8	144.9
Program Run Time (seconds)		0.42	0.57	0.59

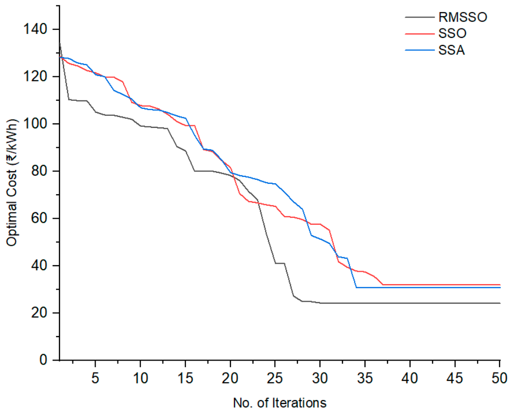
5.1.6. Robustness

An effective optimization algorithm should converge to the same global solution over iterations. The convergence curve of the proposed RMSSO algorithm was compared with those of the SSO and SSA techniques (Figure 9a–e). These convergence curves were plotted against the number of iterations (50 counts). From the figure, it is proven that the RMSSO algorithm attained energy consumption costs lower than those with the other algorithms in all five climate conditions, which was because of the effective modification of Equation (32). Further, the curves prove that RMSSO was capable of exhaustively exploring and exploiting the search space in order to determine the best optimal cost with the lowest response time.

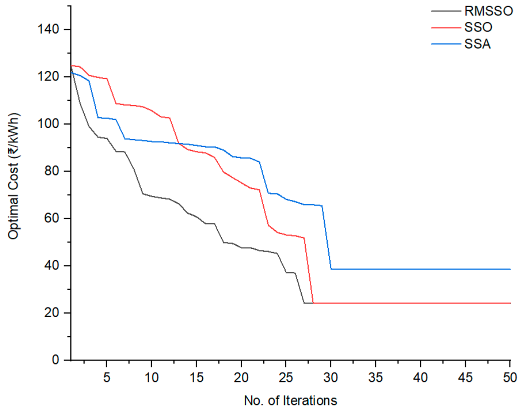
This paper adopted and used existing methodologies to check and validate the correctness and accuracy of the proposed optimization algorithm. Firstly, the quantitative results were used to measure how much better RMSSO was compared to SSA and SSO algorithms. Exploration occurs before exploitation, which supports RMSSO to improve the accuracy towards global optimum. To provide a fair comparison, the controlling parameters for all the algorithms, like number of search agents and maximum iteration, were kept same (i.e., 50). The dimension considered was also common to all algorithms (i.e., 24) along with the boundary limit of (E_{min}, E_{max}), as given in Equation (17). For other controlling parameters, the respective mathematical model of algorithms was used, and its best performance was obtained.

Table 8 illustrates the best, average, and worst outcomes obtained from the proposed RMSSO, SSO, and SSA techniques (which were tuned for 20 trail runs) in the scheduling process. These computational results were obtained from the common home environment, constraints, number of search agents, and number of iterations. This quantitative analysis was done to benchmark the performance of the proposed RMSSO algorithm, which can solve challenging problems even with a large number of variables. Furthermore, from this table, the performance difference between the RMSSO, SSO, and SSA techniques has become more pronounced. While RMSSO attained optimal cost in all five climates, the

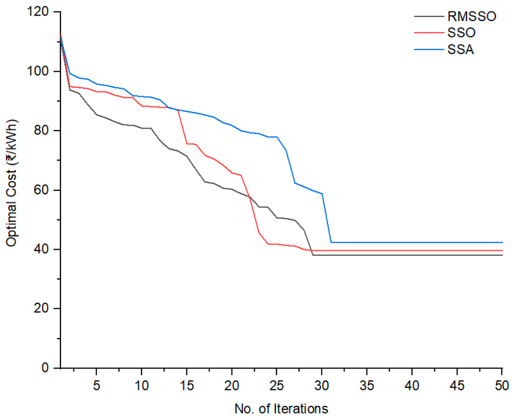
SSO and SSA techniques showed poor performance by attaining high energy consumption costs in all climates. Thus, the results have proved that the proposed RMSSO algorithm outperformed the SSA and SSO algorithms.



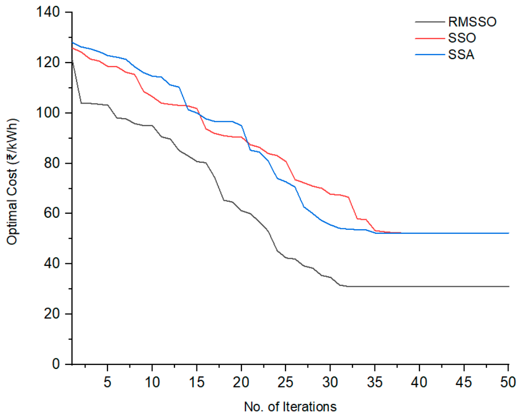
(**a**) Monsoon



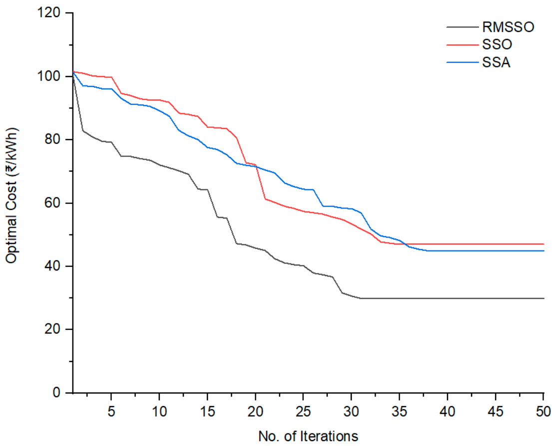
(**b**)Autumn



(**c**) Spring



(**d**) Summer



(**e**) Winter

Figure 9. Convergence characteristics' comparison of RMSSO, SSO, and SSA techniques.

Table 8. Best, average, and worst optimal cost comparison.

Algorithm	Climates	Best (INR)	Average (INR)	Worst (INR)
RMSSO	Monsoon	24.1101	24.1120	24.2144
	Autumn	24.1516	24.1631	24.3376
	Spring	38.1102	38.1127	38.2670
	Summer	31.0100	31.0126	31.1692
	Winter	29.8708	29.8847	29.8894
SSA	Monsoon	30.8829	30.8926	30.9196
	Autumn	38.5231	38.5291	38.5329
	Spring	42.4306	42.4660	42.5368
	Summer	52.3312	52.3333	52.4259
	Winter	44.9415	44.9422	44.9514
SSO	Monsoon	31.9102	31.9124	31.9262
	Autumn	24.3009	24.3119	24.3991
	Spring	39.6604	39.6761	39.6776
	Summer	52.3410	52.3540	52.3544
	Winter	47.0321	47.1512	47.2178

5.1.7. Computational Complexity Analysis

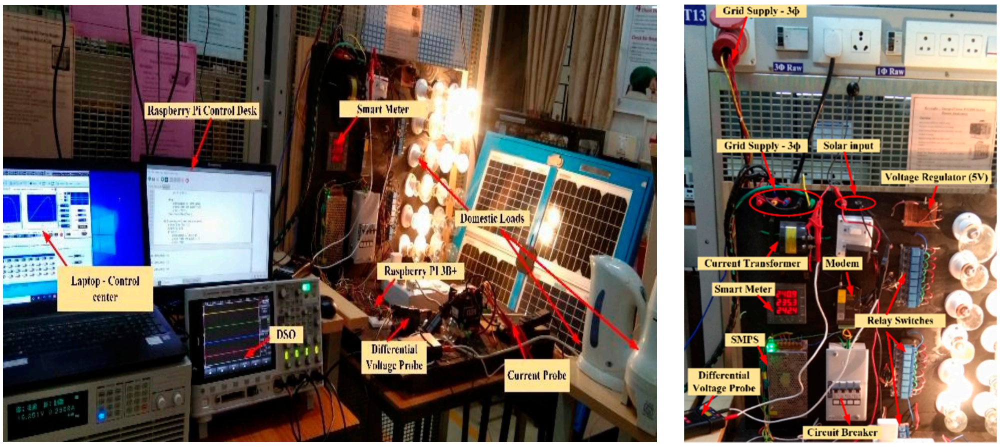
In comparison with that of the SSO algorithm, the time complexity (quantitative analysis) of the proposed RMSSO algorithm mainly depends on two aspects: (1) random initialization and (2) sperm velocity and location/position updates. These two aspects qualitatively describe the algorithm's time complexity and are expressed as O ($N \times D$) by the *Big O notation*, where N is the population size and D is the search space dimension. The authors of this paper did not modify the algorithm's initialization process and the loop body of the algorithm. Therefore, the time complexity was compared in terms of sperm velocity and position updates. Table 9 shows that the proposed RMSSO algorithm had less computational complexity/cost than the traditional sperm swarm optimization (SSO) algorithm and SSA technique did. It was because the proposed RMSSO did not follow the inertia weight updates and the calculation procedure of the SSO algorithm. Despite this, the proposed RMSSO algorithm's time complexity remained O ($N \times D$), since the algorithm's loop body was not altered.

Table 9. Actual computational time.

Dimension (D)	Population Size (N)	Computational Time (Seconds)		
		RMSSO	SSO	SSA
24	10	0.19	0.28	0.32
	50	0.42	0.57	0.59
	100	0.58	0.63	0.71

5.2. Hardware Implementation

A complete experimental model of a home energy management system was designed based on the specification given in Tables 1 and 3, presented in Figure 10a. The power circuit of HEMs is shown in Figure 10b.



(**a**) (**b**)

Figure 10. Hardware setup: (**a**) complete experimental setup, (**b**) power circuit of HEMs.

Comparison of Peak-Average Ratio (PAR)

In this section, a comparison of the peak-average ratio (PAR) is discussed. From Figure 11, it is observed that a remarkable difference in PAR value was achieved by RMSSO in all five climatic conditions compared to that of SSO and SSA techniques.

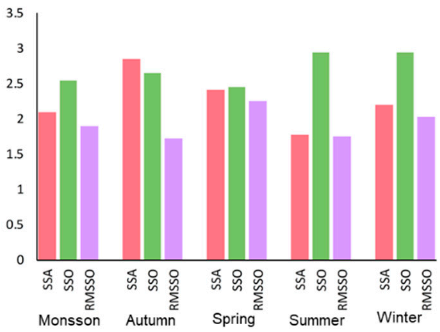


Figure 11. Comparison of peak-average ratio.

6. Conclusions

This paper proposed a novel remodeled sperm swarm optimization algorithm for optimal energy usage in an Indian home environment along with multiple constraints to reduce energy consumption cost. For the first time, the SSA, SSO, and RMSSO algorithms were used to solve the energy optimization problem in the Indian scenario. The algorithms were simulated using the Python/GUROBI tool and verified with experimental performance. The SSO, SSA, and proposed RMSSO algorithms were implemented in a common home environment under the DAP (₹/kWh) scheme. The comparison of percentage energy consumption cost difference by SSO and SSA from RMSSO is listed in Table 7 (without RES) for all five climatic conditions. It proved that the proposed RMSSO scheduled the load optimally with 100% task completion (user satisfaction) in less computation time (0.42 s) than the SSO and SSA algorithms did without violating any constraints. The comparative analysis between SSA and SSO showed that both differed by negligible values in their performance and response time. All the comparisons and validation of results were made without incorporating renewable energy sources. Thus, RMSSO could therefore be recommended as an efficient optimization algorithm to be used to control and manage home appliances.

Furthermore, the authors believe that there is a necessity to develop a certain rule-based algorithm that is capable enough to alleviate the intermittent characteristics of the RES with a battery and implement it together in the scheduling process. It is also important to add more loads so that these algorithms gain significance, which can also be treated as the future scope of this work.

Author Contributions: S.P.R. and P.K.S. have conceived the idea and converted it to an article. The authors confirm their contribution to the paper as follows: P.K.S. encouraged to investigate and supervised the findings of this work, S.P.R. developed the theory and performed the computations, verified the analytical methods, and drafted the article. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by, VELLORE INSTITUTE OF TECHNOLOGY, VELLORE, TAMIL NADU, INDIA. Grant Number: SEED-2634.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets used and analyzed during the current study are available from the corresponding author on reasonable request. All the data analyzed during this study is included in this article itself.

Acknowledgments: This research was carried out at Vellore Institute of Technology, Vellore, Tamil Nadu, India. The authors would like to sincerely thank the Management of Vellore Institute of Technology for providing the VIT SEED GRANT to carry out this research work.

Conflicts of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as potential conflicts of interests.

References

1. Derick, M.; Rani, C.; Rajesh, M.; Busawon, K.; Binns, R. Estimation of solar photovoltaic parameters using pattern search algorithm. In *International Conference on Emerging Trends in Electrical, Electronic and Communications Engineering*; Springer: Cham, Switzerland, 2016; pp. 184–191.
2. Paul, S.; Padhy, N.P. Real Time Energy Management for Smart Homes. *IEEE Syst. J.* **2020**, *15*, 4177–4188. [CrossRef]
3. Melhem, F.Y.; Grunder, O.; Hammoudan, Z.; Moubayed, N. Energy management in electrical smart grid environment using robust optimization algorithm. *IEEE Trans. Ind. Appl.* **2018**, *54*, 2714–2726. [CrossRef]
4. Yan, Y.; Qian, Y.; Sharif, H.; Tipper, D. A survey on smart grid communication infrastructures: Motivations, requirements and challenges. *IEEE Commun. Surv. Tutor.* **2012**, *15*, 5–20. [CrossRef]
5. Amer, M.; Naaman, A.; M'Sirdi, N.K.; El-Zonkoly, A.M. A hardware algorithm for PAR reduction in smart home. In Proceedings of the 2014 International Conference on Applied and Theoretical Electricity (ICATE), Craiova, Romania, 23–25 October 2014; pp. 1–6.
6. Klaimi, J.; Rahim-Amoud, R.; Merghem-Boulahia, L.; Jrad, A. Energy management algorithms in smart grids: State of the art and emerging trends. *Int. J. Artif. Intell. Appl. (IJAIA)* **2016**, *7*, 25–45. [CrossRef]
7. Huang, Y.; Wang, L.; Guo, W.; Kang, Q.; Wu, Q. Chance Constrained Optimization in a Home Energy Management System. *IEEE Trans. Smart Grid* **2018**, *9*, 252–260. [CrossRef]
8. Molderink, A.; Bakker, V.; Bosman, M.G.; Hurink, J.L.; Smit, G.J. Domestic energy management methodology for optimizing efficiency in smart grids. In Proceedings of the 2009 IEEE Bucharest Power Tech, Bucharest, Romania, 28 June–2 July 2009; pp. 1–7.
9. Behrangrad, M.; Sugihara, H.; Funaki, T. Analyzing the system effects of optimal demand response utilization for reserve procurement and peak clipping. In Proceedings of the IEEE PES General Meeting, Minneapolis, MN, USA, 25–29 July 2010; pp. 1–7.
10. Zhong, W.; Xie, K.; Liu, Y.; Yang, C.; Xie, S.; Zhang, Y. Online Control and Near-Optimal Algorithm for Distributed Energy Storage Sharing in Smart Grid. *IEEE Trans. Smart Grid* **2019**, *11*, 2552–2562. [CrossRef]
11. Pipattanasomporn, M.; Kuzlu, M.; Rahman, S. An Algorithm for Intelligent Home Energy Management and Demand Response Analysis. *IEEE Trans. Smart Grid* **2012**, *3*, 2166–2173. [CrossRef]
12. Kuzlu, M.; Pipattanasomporn, M.; Rahman, S. Hardware Demonstration of a Home Energy Management System for Demand Response Applications. *IEEE Trans. Smart Grid* **2012**, *3*, 1704–1711. [CrossRef]
13. Kazmi, S.; Javaid, N.; Mughal, M.J.; Akbar, M.; Ahmed, S.H.; Alrajeh, N. Towards Optimization of Metaheuristic Algorithms for IoT Enabled Smart Homes Targeting Balanced Demand and Supply of Energy. *IEEE Access* **2017**, *7*, 24267–24281. [CrossRef]
14. Anvari-Moghaddam, A.; Monsef, H.; Rahimi-Kian, A. Cost-effective and comfort-aware residential energy management under different pricing schemes and weather conditions. *Energy Build.* **2015**, *86*, 782–793.
15. Jiang, X.; Wu, L. A residential load scheduling based on cost efficiency and consumer's preference for demand response in smart grid. *Electr. Power Syst. Res.* **2020**, *186*, 106410. [CrossRef]

16. Wu, T.-Y.; Jiang, Y.-Z.; Su, Y.-Z.; Yeh, W.-C. Using Simplified Swarm Optimization on Multiloop Fuzzy PID Controller Tuning Design for Flow and Temperature Control System. *Appl. Sci.* **2020**, *10*, 8472. [CrossRef]
17. Pedrasa MA, A.; Spooner, T.D.; MacGill, I.F. Coordinated scheduling of residential distributed energy resources to optimize smart home energy services. *IEEE Trans. Smart Grid* **2010**, *1*, 134–143. [CrossRef]
18. Mohsenian-Rad, A.-H.; Leon-Garcia, A. Optimal Residential Load Control with Price Prediction in Real-Time Electricity Pricing Environments. *IEEE Trans. Smart Grid* **2010**, *1*, 120–133. [CrossRef]
19. Mohsenian-Rad, A.H.; Wong, V.W.; Jatskevich, J.; Schober, R. Optimal and autonomous incentive-based energy consumption scheduling algorithm for smart grid. In Proceedings of the 2010 Innovative Smart Grid Technologies (ISGT), Gaithersburg, MD, USA, 19–21 January 2010; pp. 1–6.
20. Paterakis, N.G.; Erdinç, O.; Pappi, I.N.; Bakirtzis, A.G.; Catalão, J.P. Coordinated operation of a neighborhood of smart households comprising electric vehicles, energy storage and distributed generation. *IEEE Trans. Smart Grid* **2016**, *7*, 2736–2747. [CrossRef]
21. Li, W.; Logenthiran, T.; Phan, V.T.; Woo, W.L. Implemented IoT-based self-learning home management system (SHMS) for Singapore. *IEEE Internet Things J.* **2018**, *5*, 2212–2219. [CrossRef]
22. Naz, M.; Iqbal, Z.; Javaid, N.; Khan, Z.A.; Abdul, W.; Almogren, A.; Alamri, A. Efficient Power Scheduling in Smart Homes Using Hybrid Grey Wolf Differential Evolution Optimization Technique with Real Time and Critical Peak Pricing Schemes. *Energies* **2018**, *11*, 384. [CrossRef]
23. Iqbal, Z.; Javaid, N.; Iqbal, S.; Aslam, S.; Khan, Z.A.; Abdul, W.; Almogren, A.; Alamri, A. A Domestic Microgrid with Optimized Home Energy Management System. *Energies* **2018**, *11*, 1002. [CrossRef]
24. Carli, R.; Cavone, G.; Pippia, T.; De Schutter, B.; Dotoli, M. Robust Optimal Control for Demand Side Management of Multi-Carrier Microgrids. *IEEE Trans. Autom. Sci. Eng.* **2022**, 1–14. [CrossRef]
25. Barbato, A.; Capone, A. Optimization Models and Methods for Demand-Side Management of Residential Users: A Survey. *Energies* **2014**, *7*, 5787–5824. [CrossRef]
26. Lashkari, B.; Chen, Y.; Musilek, P. Energy Management for Smart Homes—State of the Art. *Appl. Sci.* **2019**, *9*, 3459. [CrossRef]
27. Helmi, A.M.; Carli, R.; Dotoli, M.; Ramadan, H.S. Efficient and Sustainable Reconfiguration of Distribution Networks via Metaheuristic Optimization. *IEEE Trans. Autom. Sci. Eng.* **2022**, *19*, 82–98. [CrossRef]
28. Muhammad, M.A.; Mokhlis, H.; Naidu, K.; Amin, A.; Franco, J.F.; Othman, M. Distribution Network Planning Enhancement via Network Reconfiguration and DG Integration Using Dataset Approach and Water Cycle Algorithm. *J. Mod. Power Syst. Clean Energy* **2020**, *8*, 86–93. [CrossRef]
29. Aprillia, H.; Yang, H.-T.; Huang, C.-M. Short-Term Photovoltaic Power Forecasting Using a Convolutional Neural Network–Salp Swarm Algorithm. *Energies* **2020**, *13*, 1879. [CrossRef]
30. Duan, Q.; Wang, L.; Kang, H.; Shen, Y.; Sun, X.; Chen, Q. Improved Salp Swarm Algorithm with Simulated Annealing for Solving Engineering Optimization Problems. *Symmetry* **2021**, *13*, 1092. [CrossRef]
31. Alhasnawi, B.; Jasim, B.; Siano, P.; Guerrero, J. A Novel Real-Time Electricity Scheduling for Home Energy Management System Using the Internet of Energy. *Energies* **2021**, *14*, 3191. [CrossRef]
32. Mohammad, A.; Zuhaib, M.; Ashraf, I.; Alsultan, M.; Ahmad, S.; Sarwar, A.; Abdollahian, M. Integration of Electric Vehicles and Energy Storage System in Home Energy Management System with Home to Grid Capability. *Energies* **2021**, *14*, 8557. [CrossRef]
33. Zhao, P.; Gu, C.; Huo, D.; Shen, Y.; Hernando-Gil, I. Two-Stage Distributionally Robust Optimization for Energy Hub Systems. *IEEE Trans. Ind. Inform.* **2019**, *16*, 3460–3469. [CrossRef]
34. Xiao, D.; Chen, H.; Wei, C.; Bai, X. Statistical Measure for Risk-Seeking Stochastic Wind Power Offering Strategies in Electricity Markets. *J. Mod. Power Syst. Clean Energy* **2021**, 1–6. [CrossRef]
35. Sirohi, S.; Rishiraj Adhikary, N.B. Getting Data from Multiple em6400ng Smart Electric Meters 2013. Available online: https://sustainability-lab.github.io/2019/05/24/getting-data-from-multiple-smart-meters.html (accessed on 18 February 2022).
36. Quintal, F.; Garigalli, D.; Vasconcelos, D.; Cavaleiro, J.; Santos, W.; Pereira, L. Energy Monitoring in the Wild: Platform Development and Lessons Learned from a Real-World Demonstrator. *Energies* **2021**, *14*, 5786. [CrossRef]
37. Indian-Energy-Exchange. Iex, Indian Energy Exchange. 2021. Available online: https://www.iexindia.com/MarketData/AreaPrice.aspx?Area=S2 (accessed on 18 February 2022).
38. Mirjalili, S.; Gandomi, A.H.; Mirjalili, S.Z.; Saremi, S.; Faris, H.; Mirjalili, S.M. Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems. *Adv. Eng. Softw.* **2017**, *114*, 163–191. [CrossRef]
39. Shehadeh, H.A.; Ahmedy, I.; Idris MY, I. Empirical study of sperm swarm optimization algorithm. In Proceedings of the SAI Intelligent Systems Conference, London, UK, 6–7 September 2018; Springer: Cham, Switzerland, 2018; pp. 1082–1104.
40. Pranava, G.; Prasad, P. Constriction coefficient particle swarm optimization for economic load dispatch with valve point loading effects. In Proceedings of the 2013 International Conference on Power, Energy and Control (ICPEC), Dindigul, India, 6–8 February 2013; pp. 350–354.
41. Eslami, M.; Babaei, B.; Shareef, H.; Khajehzadeh, M.; Arandian, B. Optimum Design of Damping Controllers using modified Sperm Swarm Optimization. *IEEE Access* **2021**, *9*, 145592–145604. [CrossRef]
42. Gadêlha, H.; Hernández-Herrera, P.; Montoya, F.; Darszon, A.; Corkidi, G. Human sperm uses asymmetric and anisotropic flagellar controls to regulate swimming symmetry and cell steering. *Sci. Adv.* **2020**, *6*, eaba5168. Available online: https://spectrum.ieee.org/spermatowhoaa-human-sperm-dont-swim-like-we-thought (accessed on 18 February 2022). [CrossRef] [PubMed]

43. Cao, Y.; Rad, H.N.; Jamali, D.H.; Hashemian, N.; Ghasemi, A. A novel multi-objective spiral optimization algorithm for an innovative solar/biomass-based multi-generation energy system: 3E analyses, and optimization algorithms comparison. *Energy Convers. Manag.* **2020**, *219*, 112961. [CrossRef]
44. Omar, M.B.; Bingi, K.; Prusty, B.R.; Ibrahim, R. Recent, Advances and Applications of Spiral Dynamics Optimization Algorithm: A Review. *Fractal Fract.* **2022**, *6*, 27. [CrossRef]
45. Limlawan, V.; Pongchairerks, P. A PSO with ability to avoid being trapped in a local optimum A PSO with ability to avoid being trapped in a local optimum. In Proceedings of the 7th International Joint Conference on Computer Science and Software, Bangkok, Thailand, 12–14 May 2010.
46. Tamura, K.; Yasuda, K. The spiral optimization algorithm: Convergence conditions and settings. *IEEE Trans. Syst. Man Cybern. Syst.* **2017**, *50*, 360–375. [CrossRef]
47. Yüzgeç, U.; Tufan, I. Adaptive spiral optimization algorithm for benchmark problems. *Bilecik Şeyh Edebali Üniversitesi Fen Bilimleri Derg.* **2016**, *3*, 8–15.
48. Zahariadis, T.; Kalivas, D. A spiral search algorithm for fast estimation of block motion vectors. In Proceedings of the 1996 8th European Signal Processing Conference (EUSIPCO 1996), Trieste, Italy, 10–13 September 1996; pp. 1–4.
49. Yang, B.; Zhong, L.; Zhang, X.; Shu, H.; Yu, T.; Li, H.; Jiang, L.; Sun, L. Novel bio-inspired memetic salp swarm algorithm and application to MPPT for PV systems considering partial shading condition. *J. Clean. Prod.* **2019**, *215*, 1203–1222. [CrossRef]
50. Prabu, R.S.; Karthikeyan, S.P. Scheduling and energy management of smart homes using customer choice based algorithm. In Proceedings of the 2018 International Conference and Utility Exhibition on Green Energy for Sustainable Development (ICUE), Phuket, Thailand, 24–26 October 2018; pp. 1–6.
51. Ramalingam, S.P.; Shanmugam, P.K. Scheduling Smart Home Appliances using ACO Algorithm with Different Electricity Tariff Schemes. In Proceedings of the 2021 Innovations in Power and Advanced Computing Technologies (i-PACT), Kuala Lumpur, Malaysia, 27–29 November 2021; pp. 1–7. [CrossRef]
52. Ramalingam, S.P.; Shanmugam, P.K. A Home Energy Management System with Peak Demand Reduction Using Ant Colony Optimization and Time of Use Pricing Scheme. In *Soft Computing for Problem Solving. Advances in Intelligent Systems and Computing*; Tiwari, A., Ahuja, K., Yadav, A., Bansal, J.C., Deep, K., Nagar, A.K., Eds.; Springer: Singapore, 2021; Volume 1393. [CrossRef]
53. Ramalingam, S.P.; Shanmugam, P.K. A Comprehensive Review on Wired and Wireless Communication Technologies and Challenges in Smart Residential Buildings. *Recent Adv. Comput. Sci. Commun.* **2021**, *14*, e040921190451. [CrossRef]
54. Singh, J.; Poddar, S.; Ramalingam, S.P.; Shanmugam, P.K.; Kalam, A. Investigation on Dynamic Economic Dispatch Problem of Micro Grid Using Cuckoo Search Algorithm- Grid Connected and Islanded Mode. In Proceedings of the TENCON 2019–2019 IEEE Region 10 Conference (TENCON), Kochi, India, 17–20 October 2019; pp. 1886–1891. [CrossRef]