Journal of Computers Vol. 34 No. 3, June 2023, pp. 195-206
doi: 10.53106/199115992023063403014

Household Electricity Scheduling Strategy Solution Based on SA-α-QLearning

Yun Wu, Dan-Nan Zhang, Jie-Ming Yang, Zhen-Hong Liu*, Xing-Yu Pan, Yi-Fan Huang, Wei Zheng

School of Computer Science, Northeast Electric Power University, Jilin 132012, China
{838558160, 965435538, 670172713, 33249648, 1208500189, 385521224, 2998096789}@qq.com

Received 29 June 2022; Revised 20 October 2022; Accepted 11 December 2022

Abstract. Traditional household power dispatching methods are difficult to deal with the complexity of dispatching environment and the randomness of power consumption behavior, and the QLearning algorithm is prone to fall into local optimal solutions and slow convergence when solving problems, this paper proposes a new method based on SA-α-QLearning's home electricity scheduling strategy solution method. Firstly, a multi-intelligent Markov decision process model is established based on household electrical equipment; then the learning rate of a single value in the QLearning algorithm is replaced by a linear iterative learning rate; finally, a simulated annealing (SA) is used to optimize the QLearning algorithm to solve the model, by taking the Q value change difference as the new solution acceptance probability of Metropoils criterion and the dynamic adjustment temperature reduction coefficient, it alleviates the problem that the QLearing algorithm is easy to fall into the local optimal solution and the convergence speed is slow. Through a large number of comparative experiments, it is proved that the proposed method has a significant improvement in the solution of household electricity dispatching strategy.

Keywords: home energy scheduling, markov processes, simulated annealing, QLearning

1 Introduction

With the promotion and use of distributed energy technology and the growing demand for household electricity, more and more households reduce household electricity expenditure by installing household photovoltaic equipment. However, due to the lack of a reasonable household electricity scheduling strategy, there are still problems such as high electricity bills, low utilization of photovoltaic power generation, and unstable power supply during peak hours.

Home Energy Management System (HEMS) connects household electrical equipment and smart grid through smart meters, and provides a platform for unified control and management of equipment. HEMS informs users of electricity price and other information in time during operation, and recommends reasonable electricity consumption time and electricity supply method to users, guides users to adjust electricity consumption behavior, and saves electricity consumption costs for users. There are a variety of scheduling models and scheduling strategy solutions adopted by the home energy management system:

(1) Dispatching by establishing a fixed mathematical model of household electricity consumption: Literature [1] proposes multiple optimization objectives based on load peak and electricity cost minimization, and uses hybrid coding genetic algorithm to solve the problem to realize household electricity scheduling. Literature [2] aims at reducing the peak-to-average ratio (PAR), reducing energy costs and ensuring grid stability, and proposes a model solution method combining genetic algorithm and neural network to achieve a balance between user costs and grid stability. However, these fixed mathematical models of household electricity use are difficult to deal with the complexity of the scheduling environment and the randomness of electricity use behavior.

(2) The reinforcement learning method has the advantages of simple application process and short solution time, and has achieved better results in household energy optimal scheduling methods [3-4]: Literature [5] proposed a home energy management (HEM) model based on reinforcement learning. The real-time electricity price predicted by the extreme learning machine algorithm was input into the HEM model, and uses the QLearning algorithm to solve the home electricity consumption strategy of one hour in advance; Literature [6] proposed a home energy management method based on QLearning algorithm, using a single agent to control home electric equipment, and constructed a reward function based on fuzzy logic inference evaluation, by responding to elec-

* Corresponding Author

tricity price signals and consumers preference, shifting load demand from peak hours when electricity prices are high to off-peak demand when electricity prices are low, thereby minimizing energy efficiency and electricity bills. The QLearning algorithm can quickly and effectively learn the optimal strategy for both finite state and action of Markov problems, but it is easy to fall into the problem of local optimal solution and slow convergence speed [7-8]. Aiming at the problems existing in the above-mentioned household electricity scheduling model, in order to improve the effectiveness of household electricity scheduling strategy and reduce household electricity expenditure, this paper proposes a solution method for household electricity scheduling strategy based on SA-α-QLearning.

The main contributions are as follows:

Firstly, a Markov decision-making process model based on multi-agent is proposed, which simulates household electrical equipment into several individual agents, and establishes a reward function combining power cost and comfort, better simulation of user behavior randomness and improved user experience [9-10]. On this basis, a scheduling strategy solution method for SA-α-QLearning is proposed to optimize the QLearning algorithm by adjusting the learning rate α and introducing simulated annealing strategy, it alleviates the problem that the QLearing algorithm is easy to fall into the local optimal solution and the convergence speed is slow.

This article is organized as follows. The second part introduces the related work of the algorithm of scheduling strategy. The third part introduces Markov decision process model of multi-agent in detail. The fourth part introduces the improved QLearning algorithm in detail. The experimental results and discussion are presented in Part V. The sixth part summarizes the content of the paper and looks forward to future work.

2 Related Algorithms

2.1 Reinforcement Learning Composition Structure

Reinforcement learning is mainly composed of agents and environments, and its communication channels include rewards, states and actions [11-12]. The framework of reinforcement learning is shown in Fig. 1, S_t is the state of the environment at time t, A_t is the action performed by the agent at time t in the environment, A_t makes the state of the environment change to S_{t+1}, and in the new state the environment produces a new feedback R_{t+1}, the agent performs a new action A_{t+1} according to S_{t+1} and R_{t+1}, and so on until the end of the iteration.

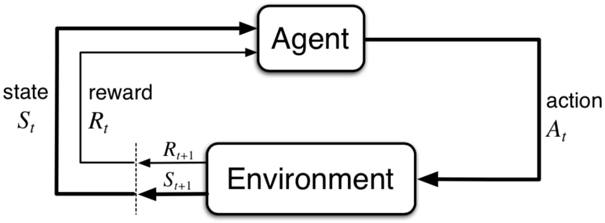


Fig. 1. Framework of reinforcement learning

2.2 MDP

Markov Decision Process (MDP) is a mathematically idealized form of reinforcement learning problems. The MDP process can be represented as a (S, A, R^{π}, P^{π}) quadruple, where: S represents the state set composed of all states of the agent; ε represents the action set taken by the agent; R^{π} is the reward function, π represents the strategy adopted; $P^{\pi} = p(s' \mid s, a)$ represents the state transition probability, that is, the probability of transitioning from state S and action a to state $s^{'}$ through policy π_i .

Define the reward-reward function G_t, which represents the sum of rewards received after k time steps in the future from time t. According to the reward-reward function, the agent continuously tries to select actions to maximize the sum of future rewards for actions under a certain discount rate. The reward function G_t is as follows:

$$G_t = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + ... = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \quad . \tag{1}$$

In the formula: r_t is the reward at time t; k is any time step; γ is the discount rate ranging from 0 to 1.

2.3 Simulated Annealing Algorithm

In 1983, S. Kirkpatrick et al proposed the simulated annealing algorithm. Compared with other intelligent optimization algorithms, simulated annealing algorithm has better robustness and search ability, and can efficiently deal with complex problems and obtain high-quality solutions [13]. The key to the simulated annealing algorithm is the Metropolis criterion and the temperature decay function of the annealing process.

The mathematical formula of the Metropolis criterion is shown in (2).

$$p = \begin{cases} 1, E(n) > E(n+1) \\ e^{-\Delta E/T_i}, E(n) < E(n+1) \end{cases} . \tag{2}$$

Among them: p is the acceptance probability, T_i is the temperature at that time, and ΔE is the return difference between the current solution and the adjacent solution.

It can be seen from formula (3) that if $E(n+1)$ is less than $E(n)$, the transition is accepted (probability is 1), and if $E(n+1)$ is greater than $E(n)$, it means that the system deviates further from the global optimal value, so the simulated annealing algorithm is helpful to jump out of the local optimal solution.

3 Multi-Agent Markov Decision Process Model

This paper proposes a household electricity model based on Multi-Agent Markov Decision Process (MAMDP), which simulates all kinds of household electrical equipment as separate agents. Firstly, the power supply mode of each agent is abstracted as a state set, and the actions of the agent when it is in a certain power supply mode are abstracted as an action set; then a reward function considering the cost of electricity consumption and comfort is established. The state set, action set and reward function of the household electricity consumption model based on the multi-agent Markov decision process are shown in Table 1.

Table 1. Household electricity consumption model of multi-agent Markov decision process

Multi-agent	State collection	Action collection	Reward function
Rigid equipment	$S_1 = \{\lambda^G_{(t)}, P^{PV}_{(t)}, P^{BT}_{(t)}\}$	$A^{NS}_t = \{a^{NS}(t) \dots a^{NS}_i(t)\}$	r^{NS}_t
Power-adjustable equipment	$S_2 = \{\lambda^G_{(t)}, P^{PV}_{(t)}, P^{BT}_{(t)}\}$	$A^{PS}_t = \{a^{PS}_{1,t}(n) \dots a^{PS}_{j,t}(n)\}$	r^{PS}_t
Time-adjustable device	$S_3 = \{\lambda^G_{(t)}, P^{PV}_{(t)}, P^{BT}_{(t)}\}$	$A^{TS}_t = \{u^{TS}_1(t) \dots u^{TS}_m(t)\}$	r^{TS}_t
Electric vehicles	$S_4 = \{\lambda^G_{(t)}, P^{PV}_{(t)}, P^{BT}_{(t)}\}$	$A^{EV}_t = \{a^{EV}_t(l) \dots a^{EV}_t(l)\}$	r^{EV}_t
Energy storage equipment	$S_5 = \{\lambda^G_{(t)}, P^{PV}_{(t)}, X^{NS}_{(t)}, X^{PS}_{(t)}, X^{TS}_{(t)}, X^{EV}_{(t)}\}$	$A^{BT}_t = \{v^{BT}_{t,out}, v^{BT}_{t,in}\}$	r^{BT}_t

In the state set: $\lambda_{(t)}^{G}$ represents the electricity price at time t ; $P_{(t)}^{PV}$ represents the predicted photovoltaic power generation output at time t ; $P_{(t)}^{BT}$ represents the output of the energy storage battery at time t ; $X_{(t)}^{NS}, X_{(t)}^{PS}, X_{(t)}^{TS}, X_{(t)}^{EV}$ respectively represent rigid equipment, power-adjustable equipment, time-adjustable devices, electric vehicles agents at time t [14].

In the action set: $a_{i}^{NS}(t)$ is the running state of the rigid equipment at time t ; $a_{j,t}^{PS}(n)$ is the running level N of the power-adjustable device j at time t ; $u_{m}^{TS}(t)$ is the running state of the time-adjustable device m at time t ; $a_{t}^{EV}(l)$ is the operating level l of electric vehicle at time t , among which l = 0, 3, 6 ; $v_{t,in}^{BT}$ and $v_{t,out}^{BT}$ respectively represent the charging state and discharging state action of the t battery, and their values are 0 and 1. When the value is 0, it means idle, and when the value is 1, it means the working state.

In this paper, the inverse of the sum of electricity expenditure and dissatisfaction cost is used as the reward function, so that when the reward return is maximized, the sum of electricity expenditure and discomfort cost is the smallest. The specific rewards of each agent are as follows [5]:

(1) Rigid equipment:

$$r_{t}^{NS} = \sum_{i\in\Omega^{NS}} -\lambda_{t}^{G}[P_{i,t}^{NS} - P_{i,t}^{PV} - P_{i,t}^{BT}]^{+} \quad . \tag{3}$$

In the formula: $P_{i,t}^{NS}$ represents the operating power of the rigid device i at time t ; $P_{i,t}^{PV}$ represents the photovoltaic power allocated and used by the rigid device i at the time t ; $P_{i,t}^{BT}$ represents the energy storage power allocated and used by the rigid device i at the time t ; Ω^{NS} is the set of rigid devices. $[\cdot]^{+}$ represents the projection above the non-negative positive, i.e. $[x]^{+} = \max(x,0)$. Since rigid equipment is invariant, the reward of rigid equipment is only related to electricity costs.

(2) Power-adjustable equipment:

$$r_{t}^{PS} = \sum_{j\in\Omega^{PS}} -(\lambda_{t}^{G}[P_{j,t}^{PS}(n) - P_{j,t}^{PV} - P_{j,t}^{BT}]^{+} + U_{j,t}^{PS}) \quad . \tag{4}$$

In the formula: $P_{j,t}^{PS}(n)$ is the power of the power-adjustable equipment j at the operating level n at time t ; $P_{j,t}^{PV}$ is the photovoltaic power allocated and used by the power-adjustable equipment j at the time t ; $P_{j,t}^{BT}$ represents the energy storage power allocated and used by the power-adjustable equipment j at the time t ; $U_{j,t}^{PS}$ is the dissatisfaction function of the power adjustable equipment j at time t ; Ω^{PS} is the set of power adjustable equipment.

(3) Time-adjustable equipment:

$$r_{t}^{TS} = \sum_{m\in\Omega^{TS}} -(\lambda_{t}^{G}[P_{m,t}^{TS} - P_{m,t}^{PV} - P_{m,t}^{B}]^{+} + U_{m,t}^{TS}) \quad . \tag{5}$$

In the formula: $P_{m,t}^{TS}$ is the operating power of the time-adjustable equipment m at time t , $P_{m,t}^{PV}$ is the photovoltaic power allocated and used by the time-adjustable equipment m at time t ; $P_{m,t}^{B}$ is the energy storage power allocated and used by the time-adjustable equipment m at time t ; $U_{m,t}^{TS}$ is the dissatisfaction function of the time-adjustable power device m at time t ; Ω^{TS} is the set of time-adjustable devices.

(4) Electric vehicles:

$$r_{t}^{EV} = -(\lambda_{t}^{G}[P_{t}^{EV} - P_{t}^{PV} - P_{t}^{B}]^{+} + U_{t}^{EV}) \quad . \tag{6}$$

In the formula: P_t^{EV} is the charging power of the electric vehicle at time t, P_t^{PV} is the photovoltaic power allocated to the electric vehicle at time t, P_t^B is the energy storage power allocated to the electric vehicle at time t; U_t^{EV} represents the anxiety cost when the electric vehicle is not full, namely in the case of insufficient power, there may be anxiety about not being able to reach the destination, where $t \in [t_E^{start}, t_E^{end}]$ is the period when the user needs to use the electric vehicle.

(5) Energy storage equipment:

$$r_t^{BT} = -\lambda_t^G [v_{t,in}^{BT} P_t^{BT}]^+ . \tag{7}$$

In the formula: r_t^{BT} is the reward function of the energy storage device, only considering the cost of the energy storage device purchasing electricity from the grid.

The overall reward function R_t of the household electrical equipment model in this paper is the cumulative reward function of each agent on the day:

$$R_t = \sum_{t \in T} \left\{ r_t^{NS} + r_t^{PS} + r_t^{TS} + r_t^{EV} + r_t^{BT} \right\} . \tag{8}$$

4 Improved QLearning Algorithm

In the process of solving the household electricity model with QLearing algorithm, the algorithm is easy to fall into the local optimal solution or the convergence time is long. The main reason is that in the ε -greedy strategy used by the original algorithm, the exploration probability ε is generally set to a single constant value. Setting a smaller ε is likely to lead to premature maturity, while the larger one can ensure the full exploration of the algorithm in the early stage, but it makes the algorithm vibrate in the later stage and difficult to converge quickly. In response to the above problems, this paper proposes a scheduling strategy solution method based on the improved QLearning algorithm. First, the learning rate α of the linear iteration is used to replace the single value learning rate in the QLearning algorithm; then the simulated annealing strategy is used to replace the ε -greedy strategy in the original algorithm. The difference of Q value change is used as the acceptance probability of the new solution of the Metropoils criterion, so that the agent chooses a non-optimal strategy with a certain probability in the current state exploration, so as to jump out of the local optimal solution; finally, the temperature decay function is used to control the cooling rate of simulated annealing strategy, in the temperature decay function, by comparing the difference between the change of the Q value and the fixed temperature reduction coefficient, the largest value is selected as the actual temperature reduction coefficient to update the temperature, which ensures the full exploration of the algorithm and improves the algorithm convergence speed. The specific improvement formula is as follows:

(1) Learning rate reduced by linear iteration

The QLearning algorithm finds action a_t according to the greedy strategy (ε -greedy), and uses the state action value function $Q(s_t, a_t)$ to iteratively update the strategy. The action value function in the QLearning algorithm is shown in formula (9) [15]:

$$Q(s_{t+1}, a_{t+1}) \leftarrow Q(s_t, a_t) + \alpha[R(s_t, a_t) + \gamma \max_{\alpha'} Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)] . \tag{9}$$

Learning rate α, as an important parameter in the algorithm, determines the active degree of the agent's response to the received reward. The higher the α value, the greater the fluctuation of the Q value in each learning stage. In the optimization process, this paper uses the linear iterative reduction method to update the α value, and the action value function is as follows [16]:

$$Q(s_{t+1}, a_{t+1}) \leftarrow Q(s_t, a_t) + \alpha(t)[R(s_t, a_t) + \gamma \max_{\alpha'} Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)]. \tag{10}$$

Where the learning rate $\alpha(t)$ is:

$$\alpha(t) = 1 - 0.9(\frac{eval}{\max eval}) \ . \tag{11}$$

In the formula: $eval$ and $\max eval$ are the number of iterations at time t and the maximum number of function iterations, respectively. After using the linearly decreasing learning rate, the value of *α* in the early stage is high, the agent responds positively to the reward received, and the Q value fluctuates greatly; the value of *α* in the later stage is low, and the agent does not respond positively to the reward received, the Q value fluctuates less, and the algorithm converges quickly.

(2) A new temperature decay function in the simulated annealing strategy [7]:

$$T_t = \max(f(Q(s_t, a_t) - Q(s_{t-1}, a_{t-1}), d) \cdot T_{t-1} \ . \tag{12}$$

In the formula: T_t is the current state temperature, T_{t-1} is the previous state temperature; *d* is the temperature reduction coefficient, which is generally a fixed value between 0.8 and 0.99; $f\,()$ is the $Sigmoid$ normalization function, $f(Q(s_t, a_t) - Q(s_{t-1}, a_{t-1}) = \frac{1}{1 + e^{-(Q(s_t, a_t) - Q(s_{t-1}, a_{t-1}))}}$; " . " represents the multiplication operation.

For the temperature of each iteration of the temperature decay function, by selecting the larger value of the change of the Q function and the temperature reduction coefficient as the actual temperature reduction coefficient, the algorithm can slow down the temperature reduction ratio when the Q value function changes greatly in the early stage. In this way, the exploration of non-optimal solution actions is increased; when the late algorithm is close to convergence, the change of the Q value function gradually decreases, and the decrease of the temperature is accelerated at this time, so that the algorithm can converge stably.

(3) Metropolis new solution acceptance criterion in simulated annealing strategy [17]

When constructing the new solution determination formula, the changes of the Q function in each step are integrated in each iteration process. The new solution determination formula is as follows:

$$\pi(s_t, a_t) = \begin{cases} \arg\max Q(s_t, a_p), p < \tau \\ \quad \text{else} \qquad , p > \tau \end{cases} . \tag{13}$$

In the formula: $\arg\max Q(s_t, a_p)$ is the state-action pair that obtains the expected cumulative reward in the current state; *τ* is the random number of (0, 1) generated; $p = e^{(\frac{Q(s_t, a_r) - Q(s_t, a_p)}{T_t})}$ is the probability of accepting the new solution, and a_r is the randomly selected action in state s_t.

In the Metropolis new solution acceptance criterion, the accepted action is selected by comparing the new solution acceptance probability *p* and the random number τ. If $p < \tau$, the state action that obtains the maximum return is selected as the action a_p of the (s_t, a_p) as the accepting action; if $p > \tau$, select another action in the action space.

Specific steps are as follows:

(1) Initialization related parameters: initial temperature T_0, temperature reduction coefficient d, initial state s_0, learning rate *α*, discount rate γ;

(2) Record the status of time *t* as s_t, and start looping to find action a_t;

(3) When in the current state s_t, select action a_p according to strategy π, and randomly select action a_r from the action space;

(4) Calculate $Q(s_t, a_p), Q(s_t, a_r)$ according to formula (10), if $Q(s_t, a_r) > Q(s_t, a_p)$, select the action a_r as the current action; otherwise, according to the Metropolis new solution acceptance criterion, namely formula (13), calculate the acceptance probability *p* of the new solution, and generate the acceptance action. And record the accept action as the current action a_t;

(5) Execute the selected action a_t, reach the new state s_{t+1}, and get the immediate reward r of environmental feedback;

(6) Calculate the updated value of $Q(s, a)$ according to formula (10);

(7) Judging whether the termination condition is met, usually the termination condition is set as the temperature reaching the minimum value or whether the current state is the final state. If the termination condition is not satisfied, select the temperature reduction coefficient according to the temperature decay function formula (12), update the temperature and reset the number of iterations; then go to step (2) to enter the next training; if it is satisfied, output the optimal strategy π^*.

5 Case Analysis

The household power data used in this experiment is from the Pecan Street website, which contains the photovoltaic household power data on the streets of an area in Austin. The electricity price data is from the time of use electricity price provided by Austin Energy, a local power grid operator, for users in Austin. The PV historical power generation data is predicted. The hardware configuration of the experimental platform in this paper is as follows: Processor-Intel i7-6700HQ, GPU graphics card-Tesla P100, memory 8GB; software configuration is as follows: development software-Pycharm, development language-Python.

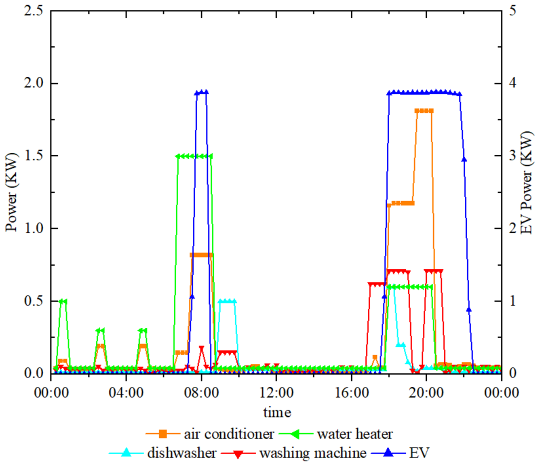
Since the electrical equipment contained in each household in the data set is not exactly the same, in the experiment of this paper, families with photovoltaic, rigid equipment (refrigerator, alarm, electric light), power adjustable equipment (air conditioner, water heater), time adjustable equipment (washing machines, dishwashers), electric vehicles are selected for research. The addition of energy storage equipment can store surplus photovoltaic power, and can also be used for high power prices by storing the power at low power prices. Therefore, this article adds a battery with a capacity of 17kwh to the family, and sets the charging efficiency η_{in}^{B} and discharging efficiency η_{out}^{B} of the battery to 0.9. The relevant parameters of household electrical equipment are shown in Table 2.

Table 2. Parameters of household electrical equipment

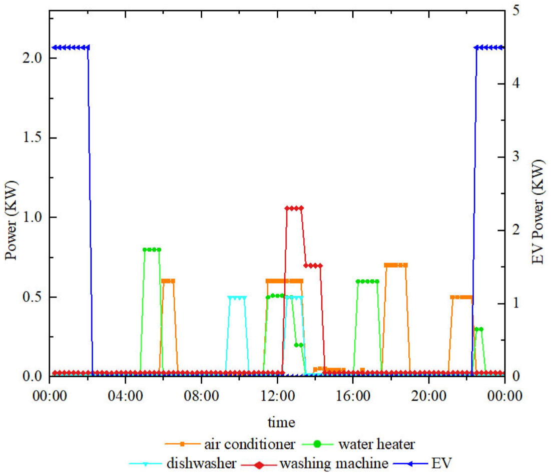
Electrical equipment	Power/KW	Dissatisfaction coefficient	Schedule time
Siren	0.1	-	24h
Electric light	0.3	-	18:00—22:00
Refrigerator	0.5	-	24h
Air conditioner	0.5-1.4	0.04	24h
Water heater	0.3-1.5	0.05	24h
Washing machine	0.7	0.05	7:00—22:00
Dishwasher	0.5	0.05	8:00—22:00
Electric vehicles	0-6	0.04	22:00-7:00
Energy storage equipment	-1.2-1.2	-	24h

5.1 Analysis of Optimal Scheduling Results of Household Electricity Consumption Model

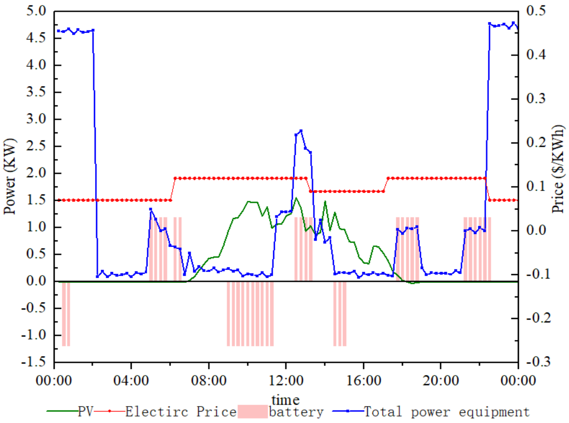
In this paper, after the predicted photovoltaic power generation is obtained, the established MAMDP household power consumption model is solved using the QLearing algorithm, and the electricity consumption strategy is obtained. And use α-QLearning and SA-α-QLearning algorithm in this paper to compare the electricity consumption curve and electricity expenditure of household electrical equipment before and after optimization, as shown in Fig. 2. and Table 3. Set the related parameters in the QLearning algorithm to: $\gamma = 0.9$, $\alpha = 0.1$, $\varepsilon = 0.1$; the related parameters in α-QLearning: $\gamma = 0.9$, $\varepsilon = 0.1$; the related parameters in SA-α-QLearning: $\gamma = 0.9$, $T_0 = 300$, $d = 0.7$.



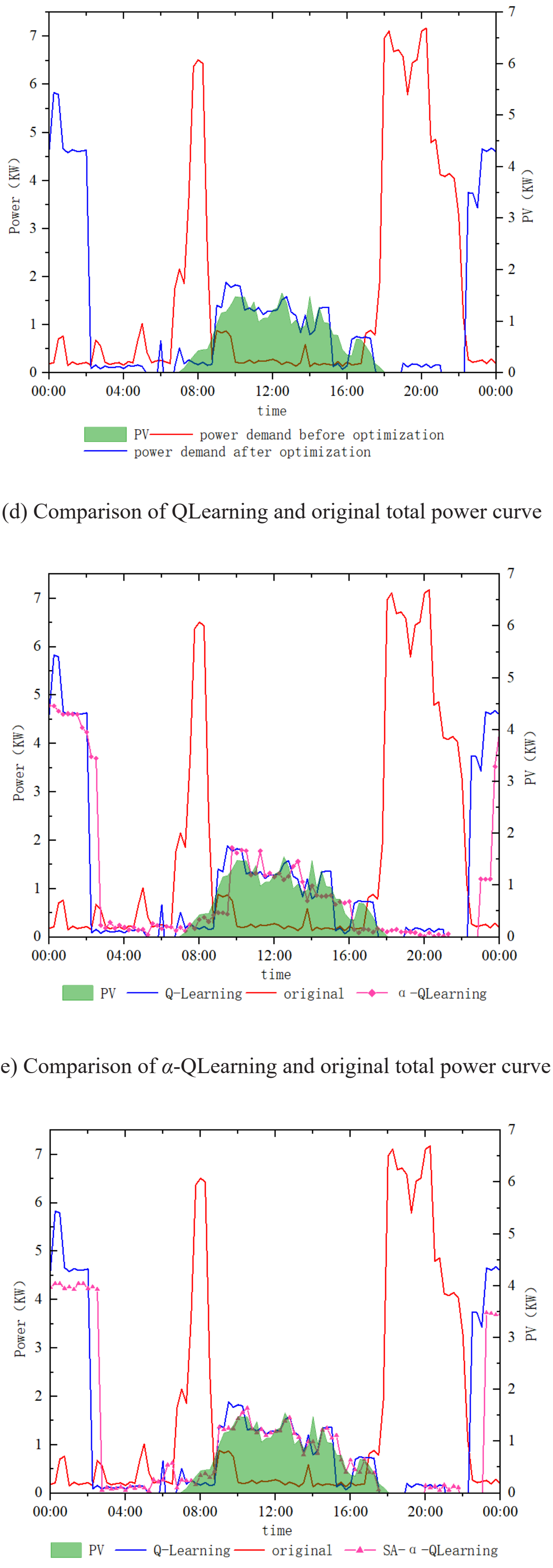
(a) The operation of household electrical equipment before optimization



(b) The operation of household electrical equipment after the optimization of the method in this paper



(c) The working conditions of the energy storage equipment after the optimization of the method in this paper



(d) Comparison of QLearning and original total power curve

(e) Comparison of *α*-QLearning and original total power curve

(f) SA-*α*-QLearning and original total power curve comparison

Fig. 2. Comparison before and after optimization of household electrical equipment

Table 3. Household electricity expenditure before and after optimal scheduling

Equipment type	Electrical equipment	Electricity expenses ($)			
		Original	QLearning	α-QLearning	SA-α-QLearning
Rigid equipment	Rigid equipment	1.3	1.3	1.3	1.3
Power adjustable device	Air conditioner	1.768	1.216	1.201	1.195
	Water heater	1.476	1.101	1.097	1.078
Time adjustable device	Washing machine	0.956	0.812	0.809	0.803
	Dishwasher	0.468	0.395	0.392	0.389
Electric vehicles	Electric vehicles	3.612	1.964	1.963	1.836
Energy storage equipment	Energy storage equipment	0	0.272	0.220	0.321
Total electricity bill		9.574	7.06	6.982	6.922

From Fig. 2(a), it can be seen that the unoptimized equipment operates according to the user's own electricity consumption habits. The operation of such equipment is mainly concentrated at 6:00-9:00 in the morning and 18:00-21:00 in the afternoon. At this time, the electricity price is the peak period and the photovoltaic power is almost zero. During this period, various electrical equipment and electric vehicle charging are intensively used, resulting in high electricity cost for users. In Fig. 2(b), after optimized operation of the method in this paper, the power of the power-adjustable equipment decreases and time disperses, and is transferred to the stage of normal electricity price. The operating time of the time-adjustable equipment is mostly transferred to the period of high photovoltaic power generation or the period of normal electricity price. It can be seen in Fig. 2(c) that when the power consumption is high, the energy storage battery will discharge at this time. During the period of 9:00-11:00, the photovoltaic power generation power gradually increases, but the user consumes less electricity at this time, so the surplus electricity is input into the storage battery. The electric vehicle is charged during the valley electricity price period in the evening.

From Fig. 2(d), the comparison chart of the total electricity demand curve solved by the QLearning algorithm is carried out. For the convenience of comparison, the total electricity demand curve is the sum of the power required by each electrical device and the power of the energy storage device, as shown in Table 3. It can be seen that the electricity cost of the QLearning algorithm is lower than that before optimization; in Fig. 2(e), the total electricity demand curve obtained after the solution of the α-QLearning algorithm is obtained. It is lower than the QLearning algorithm; in Fig. 2(f), the total electricity demand curve after the solution of the SA-α-QLearning algorithm proposed in this paper is compared. From Table 3, it can be seen that this method is lower than other algorithms in terms of electricity expenditure, and it is better than other algorithms in photovoltaic utilization, indicating that the improved method proposed in this paper can effectively help the QLearning algorithm to jump out of the local optimal solution and obtain a better strategy.

5.2 Algorithm Effect Comparison

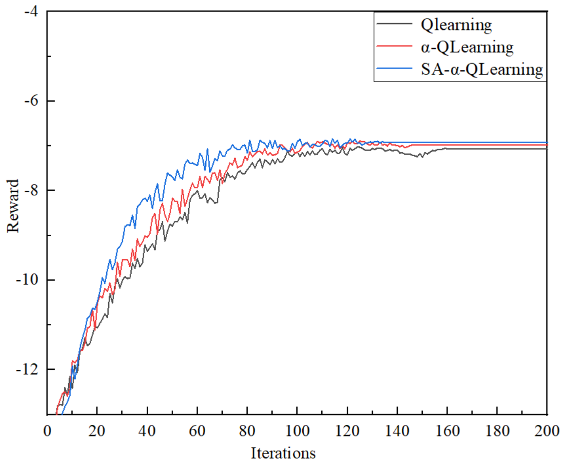


Fig. 3. Comparison of QLearning iterations before and after the improvement

In order to further verify the performance of the algorithm proposed in this paper, the rewards of the QLearning algorithm, the *α*-QLearning and the SA-*α*-QLearning in this paper are compared.

It can be seen from the reward of each algorithm iteration in Fig. 3. that the unimproved QLearning algorithm tends to converge around 1500 iterations, and the reward after convergence is -7.07; the *α*-QLearning is in convergence around 1450 iterations, the reward after convergence is -6.98, and the change in reward value during the exploration process is smaller than that of the traditional QLearning algorithm. The QLearning algorithm after linear iteration of the learning rate improves the convergence speed of the algorithm; the SA-*α*-QLearning converges around 1300 iterations, and the reward after convergence is -6.92, and the improved algorithm is more cautious when selecting state actions, and the fluctuations are small during convergence, which can converge stably.

5.3 Training Time Contrast

In order to compare the performance of the algorithm before and after the improvement, this paper compares the number of iterations in the first search for the optimal solution and the time spent in iterating 2000 times.

Table 4. Comparison of QLearning convergence time before and after the improvement

Serial number	QLearning		*α*-QLearning		SA-*α*-QLearning	
	Number of iterations	Discovery time (s)	Number of iterations	Discovery time (s)	Number of iterations	Discovery time (s)
1	155	4.25	150	4.03	125	3.34
2	153	4.60	147	4.52	126	3.76
3	162	4.67	149	4.55	138	4.09
4	155	4.57	142	4.31	122	3.35
5	168	5.29	157	4.69	141	4.15

It can be concluded from Table 4 that the number of iterations required by the QLearing algorithm to find the optimal solution is the largest, mainly because the initial search space of the QLearing algorithm is too large at the initial stage, and the algorithm converges slowly to the optimal solution during the exploration process, resulting in a long algorithm time; at the same time, it can also be obtained that the time of *α*-QLearing algorithm to find the optimal solution for the first time in the exploration process is shortened, which shows that the convergence speed of the algorithm is improved after the algorithm learning is improved; improved SA-*α*-QLearing can find the optimal solution earlier, and the iteration time is shorter, because the adaptive change exploration strategy ensures the diversity of the early solution and the convergence stability of the later algorithm.

6 Conclusion

In view of the fact that traditional household electricity scheduling methods are usually based on a fixed mathematical model, it is difficult to deal with the complexity of the scheduling environment and the randomness of electricity consumption behavior. Therefore, this paper proposes a multi-agent-based Markov decision process model. The electrical equipment is simulated as multiple separate agents, the proposed SA-*α*-QLearning algorithm is used to solve the household electricity consumption model, and the scheduling strategy is obtained, which alleviates the problem that the QLearning algorithm is easy to fall into the local optimal solution and the convergence speed is slow. The performance of scheduling policy solution has been significantly improved.

With the increase of household electrical equipment, the exploration scope of the QLearing algorithm gradually increases. Later, the algorithm solving ability can be further improved by optimizing the exploration matrix or using deep reinforcement learning and other methods. In addition, with the popularity of household PV, it is possible to connect PV to the grid. In the future, we can study and recommend dispatching strategies combined with new energy to bring some benefits to household users and improve the safe and stable operation of the grid.

7 Acknowledgement

This work was supported by the Science and Technology Research Project of Jilin Province Education Department (JJKH20210100KJ).

References

[1] Q. Lu, H. Yu, Y.-J. Leng, J.-C. Hou, P.-J. Xie, Research on Model and Algorithm of Smart Electricity Consumption Task Scheduling Optimization in Household, Proceedings of the CSEE 38(13)(2018) 3826-3836.

[2] N. Shaheen, N. Javaid, N. Nisa, A.-M. Zeb, Z.-A. Khan, U. Qasim, Appliance Scheduling for Energy Management with User Preferences, in: Proc. 2016 International Conference on Innovative Mobile & Internet Services in Ubiquitous Computing, 2016.

[3] W. Tang, S. Wu, R. Li, X. Jin, D. Yang, Z.-Y. Liu, W.-P. Hong, Optimization of Regional Cooling pipe Network Based on Genetic Algorithm, Journal Of Northeast Electric Power University 40(6)(2020) 86-91.

[4] Z.-M. Wang, L. Sun, L.-G. Sun, R.-T. Yuan, Optimization Scheduling of CCHP Micro-Energy Network Based on Phase Change Energy Storage Thermal Resistance Model, Journal Of Northeast Electric Power University 42(1)(2022) 96-103.

[5] X. Xu, Y.-W. Jia, Y. Xu, Z. Xu, S.-J. Chai, C.-S. Lai, A Multi-agent Reinforcement Learning-based Data-driven Method for Home Energy Management, IEEE Transactions on Smart Grid 11(4)(2020) 3201-3211.

[6] F. Alfaverh, M. Denai, Y.-C. Sun, Demand Response Strategy Based on Reinforcement Learning and Fuzzy Reasoning for Home Energy Management, IEEE Access 8(2020) 39310-39321.

[7] X.-S. Ma, J. Zhu, J. Tan, H. Tang, J.-T. Zhou, Physarum polycephalum algorithm based improved Q-learning for shortest path solution, Journal of Electronic Measurement and Instrumentation 33(5)(2019) 148-157.

[8] H. Zhang, X. Shen, H.-Y. Mu, A.-D. Liu, H. Wang, Research on Online Optimal Dispatching of Residential Energy Consumption Based on Multi-agent Asynchronous Deep Reinforcement Learning, Proceedings of the CSEE 40(1)(2020) 117-127.

[9] L.-M. Yin, L. Wang, G. Lei, J.-Y. Jiang, Z.-D. Yang, M.-W. Ni, Optimal Dispatch of Microgrid Considering Demand Response and Comprehensive Battery Loss, Journal Of Northeast Electric Power University 40(2)(2020) 37-48.

[10] Y.-F. Huang, W. He, S.-Y. Liu, Research on Energy Optimization of Home Microgrid for Intelligent Power Consumption, Journal Of Northeast Electric Power University 40(4)(2020) 29-34.

[11] X.-Y. Zhang, S.-J. Han, Q.-C. Tao, Y.-M. Yu, Path Planning Algorithm Based on Improved Q-Learning, Modern Computer 28(2)(2022) 67-72.

[12] Y.-J. Wang, M.-L. Chen, X.-F. Mou, Y.-Y. Li, Z. Zhang, Operation of Microgrid Considering Energy Storage and Response of Power and Thermal Loads, Journal Of Northeast Electric Power University 41(2)(2021) 108-118.

[13] A.-A. Hafez, A.-Y. Abdelaziz, M.-A. Hendy, A.-F.-M. Ali, Optimal sizing of off-line microgrid via hybrid multi-objective simulated annealing particle swarm optimizer, Computers & Electrical Engineering 94(2021) 107294.

[14] W.-C. Dai, A.-D. Liu, X. Shen, H.-J. Ma, H. Zhang, Online Optimization of Charging and Discharging Behavior of Household Electric Vehicle Cluster Based on MADDPG Algorithm, Journal Of Northeast Electric Power University 41(5)(2021) 80-89.

[15] Y.-L. Yuan, Research on Deep Reinforcement Learning Algorithm and Applications, [dissertation] Guangzhou: South China University of Technology, 2019.

[16] T.-N. Huynh, D.-T.-T. Do, J. Lee, Q-Learning-based parameter control in differential evolution for structural optimization, Applied Soft Computing 107(2021) 107464.

[17] X.-L. Wang, W.-N. Hao, G. Chen, X.-H. Yu, The Sarsa Reinforcement Learning Method Based on Simulated Annealing Strategy, Computer Simulation 36(4)(2019) 219-222+228.