An Algorithm for Unit Commitment Based on Hopfield Neural Network

Weixin Gao, Nan Tang, Xiangyang Mu
Shaanxi Key Laboratory of Oil-Drilling Rigs Controlling Technique, Xi'an Shiyou University; gwxd@tom.com

Abstract

This paper presents an algorithm, which is based on a Hopfield neural network, for determining unit commitment. By constructing an appropriate energy function, a single layer Hopfield neural network can solve the problem of assigning output power of generators at any given time. Based on this single layer Hopfield neural network, a multi-layer Hopfield neural network is presented. The multi-layer Hopfield neural network can solve the problem of power system unit commitment. The energy functions of single layer and multi-layer Hopfield neural network and the corresponding algorithm are given in the paper. The restricted conditions of the balance between power supply and demand, maximum and minimum outputs of power plants are considered in the energy function. So is the speed of propulsion and decreasing power of generators. An example shows that the result obtained by Hopfield neural network is somewhat similar to that obtained by genetic algorithm, but the calculation time is much shorter.

Keywords: unit commitment, Hopfield neural network, optimization, power system, planning

1. Introduction

Unit commitment is an important task of short term power system economic dispatch. Researchers have studied some algorithms for solving the problem. In general, the existing algorithms can be grouped into two classes-the traditional optimal technology and the artificial intelligent technology. Dynamic programming[1,2] and Lagrangian relaxation method[3~5] are two representative algorithms in traditional optimization technology. The algorithms based on random search optimization method attract most researchers' attention in artificial optimization technologies. Reference [6] and [7] optimize the unit commitment problem by using evolutionary optimization method and social evolutionary programming respectively. Reference [8-10] used genetic algorithms to optimize unit commitment problem. Recently, particle swarm optimization method is attracting more and more researchers' attention. Reference [11-13] used this algorithm for optimizing unit commitment problem. Through optimization power unit's on/off status, reference [14] optimizes unit commitment problem by an improved discrete particle swarm optimization algorithm (DPSO). The improved DPSO overcome the DPSO's defect of falling into a local optimal result through a new initializing machine.

However, from the existed literature it is rear to see unit commitment algorithms based on calculating a group of differential functions. In this paper we put forward an algorithm, which is based on a multi-layer Hopfield neural network, to optimize the unit commitment problem.

2. Mathematical Model

The mathematical model of the unit commitment can be written as:

$$\min C = \sum_{i=1}^{N}\sum_{t=1}^{H}\left\{F_{it}\left(P_{it}\right)U_{it} + S_{Ti}U_{it}\left(1-U_{i,t-1}\right)\right\} \tag{1}$$

s.t.

$$\sum_{i=1}^{N} U_{it}P_{it} = C_t + P_{Dt}$$

$$\sum_{i=1}^{N} U_{it}P_{i\max} \geq \left(1+k\right)P_{Dt}$$

$$P_{i\min} \leq P_{it} \leq P_{i\max}$$

$$P_{it} - P_{i,t-1} \leq R_{URi}$$

$$P_{i,t-1} - P_{i,t} \leq R_{DRi}$$

$$\sum_{t=ku}^{kd-1} U_{it} \geq M_{Ui}$$

$$\sum_{t=kd}^{ku-1} U_{it} \geq M_{Di}$$

$$U_{i,ku} = 1, U_{i,ku-1} = 1, U_{i,kd} = 1, U_{i,kd-1} = 1$$

978-0-7695-3304-9/08 $25.00 © 2008 IEEE
DOI 10.1109/ICNC.2008.148

Where $F_{it}\left(P_{it}\right)$ is the fuel cost function of i-th unit at the t-th hour which can be expressed as $F_{it}\left(P_{it}\right)=a_i P_{it}^2+b_i P_{it}+C_i$. In the expression, a_i, b_i and C_i are all constants; P_{it} is the power output of i-th unit at the t-th hour; S_{Ti} is the start-up cost of the i-th unit. U_{it} is the on/off status of the i-th unit at the t-th hour, and $U_{it} \in \{0,1\}$; N is the number of thermal generating units; H is the number of hours in the period studied.

In constraints, C_t is the power loss at t-th hour; P_{Dt} is the system load demand at t-th hour. k is spare coefficient. $P_{i\min}$ and $P_{i\max}$ are the i-th unit's minimum and maximum power outputs. R_{UPi} and R_{DRi} are the ramp-up and ramp-down rate limits of the i-th unit. M_{Ui} and M_{Di} are the minimum up-time and down-time of the i-th unit. ku is the hour at which the i-th unit is started up and kd is the hour at which the i-th unit is shut down.

In the above formulation, the decision variable U_{it} of either 0 or 1, and variables P_{it} of continuous values constitute a mixed integer programming problem. We propose a multi-layer Hopfield neural network to solve the problem.

3. The structure of Hopfield neural network

Before calculating equation (1), it is necessary to establish a neural network model for expressing the result of optimization. If we only want to optimize each unit's power output at hour t and don't take the second part of equation (1) into consideration, we can establish a matrix as shown in table 1 for any given time t.

Table 1. The planning result table

	W_1	W_2	…	W_M
F_1	0	1	…	0
F_2			…	
…			…	
F_N	1	0	…	0

In table1, the number in line refers to a unit and the number in column refers to the unit's minimum power output adjustment value, and there is $W_1=W_2\ldots\ldots=W_M=W$. An element in the table with value 1 indicates that the corresponding column output value will be assigned to the corresponding row unit. For example, if element (i,j) is 1, this indicates that i-th unit will increase its output by W_j. The relation between W_j and P_{Dt} can be written as:

$$\sum_{j=1}^{M} W_j = P_{Dt} \tag{2}$$

From equation (2) we can see that there is only one element with a value of 1 in each column, the others with a value of 0. Otherwise, the output would not be equal to the system load demand.

The value of M can be written as:

$$M = Int\left(\frac{P_{Dt}}{INCR_{\min}}\right) \tag{3}$$

Where $Int(\ )$ is the integral function and $INCR_{\min}$ is the minimum value for adjusting a unit.

Let v_{ij} denote element (i,j) of table1. On the basis of the matrix shown in table 1, we can establish the Hopfield neural network energy function for unit commitment at t-th hour as follows:

$$\begin{aligned} E= & \frac{E1}{2}\sum_{j=1}^{M}\left(\sum_{i=1}^{N} v_{ij}-1\right)^2+\frac{E2}{2}\sum_{i=1}^{N}\frac{1}{1+e^{O}} \\ & \frac{E3}{2}\sum_{i=1}^{N}\left(a_i\left(w\sum_{j=1}^{M} v_{ij}\right)^2+b_i\left(w\sum_{j=1}^{M} v_{ij}\right)+c_i\right) \end{aligned} \tag{4}$$

Where $O=\left.-\left(\sum_{j=1}^{M} v_{ij}W_j-P_{i\min}\right)\left(\sum_{j=1}^{M} v_{ij}W_j-P_{i\max}\right)\middle/ u_v\right.$

In equation (4), E1 ~ E3 and u_v are coefficients. The first part of equation (4) makes each column to have only one element with a value of 1. The second part makes each unit's power output to be within the given region. The third part corresponds to the first part of equation (1).

Equation (4) doesn't take the second part of equation (1) into consideration and can not decide each unit's power output for all hours. So in order to solve equation (1), it's necessary to establish a multi-layer Hopfield neural network.

The multi-layer Hopfield neural network has H layers of neural network which correspond to t_1-hour ~ t_H-hour. The structure of multi-layer Hopfield neural network is shown in Fig.1, and the output of each layer is similarly to Table1. The number of columns of each layer neural network can be decided according to equation (3).

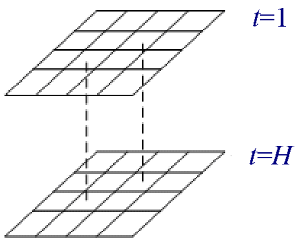

Figure 1. The sketch map of multilayer neural network

So, u_{it} in equation (1) can be written as:

$$u_{it}=\frac{1}{1+e^{i_t}} \tag{5}$$

Where $i_t=\dfrac{-\left(\sum_{j=1}^{M(t)} v_{tij}-0.5\right)}{u_v}$

Let v_{tij} denote element (t,i,j), the energy function of the multi-layer Hopfield neural network can be written as:

$$E=\frac{E1}{2}\sum_{t=1}^{H}\sum_{j=1}^{M(t)}\left(\sum_{i=1}^{N} v_{tij}-1\right)^2+\frac{E2}{2}\sum_{t=1}^{H}\sum_{i=1}^{N}\frac{1}{1+e^{Ot}}+$$
$$\frac{E3}{2}\sum_{t=1}^{H}\sum_{i=1}^{N}\left(a_i\left(w\sum_{j=1}^{M(t)} v_{ij}\right)^2+b_i\left(w\sum_{j=1}^{M(t)} v_{ij}\right)+c_i\right)+$$
$$\frac{E4}{2}\sum_{t=1}^{H}\sum_{i=1}^{N}\frac{1}{1+e^{O_{UPti}}}+\frac{E5}{2}\sum_{t=1}^{H}\sum_{i=1}^{N}\frac{1}{1+e^{O_{DRti}}}+ \tag{6}$$
$$\frac{E6}{2}\sum_{t=1}^{H}\sum_{i=1}^{N} S_{Ti}U_{it}\left(1-U_{i,t-1}\right)+$$
$$\frac{E7}{2}\sum_{t=1}^{H}\sum_{i=1}^{N}\left(1-U_{i,t-1}\right)\left(w\sum_{j=1}^{M(t)} v_{ij}-P_{i\min}\right)^2+$$
$$\frac{E8}{2}\sum_{t=1}^{H}\sum_{i=1}^{N}\left(1-U_{i,t+1}\right)\left(w\sum_{j=1}^{M(t)} v_{ij}-P_{i\min}\right)^2+$$

Where $Ot=-\left(\sum_{j=1}^{M(t)} v_{tij}W_j-P_{i\min}\right)\left(\sum_{j=1}^{M(t)} v_{tij}W_j-P_{i\max}\right)\Big/u_v$

$$O_{UPti}=-\frac{\left(\sum_{j=1}^{M(t)} v_{tij}w_j-\sum_{j=1}^{M(t-1)} v_{t-1,ij}w_j\right)-R_{UPi}}{u_v}$$

$$O_{DRti}=-\frac{\left(\sum_{j=1}^{M(t-1)} v_{t-1,ij}w_j-\sum_{j=1}^{M(t)} v_{tij}w_j\right)-R_{DRi}}{u_v}$$

In equation (6), E1～E8 are coefficients; H is the number of hours in the period studied. $M(t)$, which varies with t, is the number of columns of the neural network, and can be calculated by using equation (3). u_v is a coefficient. The meanings of other parameters are similar to those of equation (1).

In equation (6), item E1 makes the output of each layer neural network fit the form of table1 and also makes the power output of all units equal to the power load demand at each hour. The item E2 forces the output of each unit to be within the unit's minimum and maximum power output. The item E3 minimizes the fuel cost and corresponds to part 1 of the equation (1). Item E4 makes each unit's ramp-up rate to be less than R_{UPi}. Item E5 forces each unit's ramp-down rate to be less than R_{DRi}. Item E6 minimizes the start-up cost. Item E7 minimizes the power output of each unit when the unit is going to be started-up. Item E8 minimizes the power output of each unit when the unit is going to be shut-down.

We use the following equation[15] to calculate (6)

$$\frac{du_{tij}}{dt}=-\frac{\partial E}{\partial v_{tij}} \tag{7}$$

Where u_{tij} is the input of neural node (t,i,j) and v_{tij} is the output of neural node (t,i,j).

After establishing the energy function (6), we can deduce the dynamic function as follow:

$$\left\{\begin{aligned}
&\frac{du_{tij}}{dt}=-E1\left(\sum_{i=1}^{N} v_{tij}-1\right)\\
&-\frac{E2}{2u_v}\frac{e^{Ot}w_j\left(2\sum_{j=1}^{M(t)} v_{tij}W_j-P_{i\min}-P_{i\max}\right)}{\left(1+e^{Ot\min}\right)^2}\\
&-\frac{E3}{2}\left(a_i\left(w\sum_{j=1}^{M(t)} v_{ij}\right)+b_iw\right)\\
&-\frac{E4}{2u_v}\frac{w_je^{O_{UPti}}}{\left(1+e^{O_{UPti}}\right)^2}+\frac{E5}{2u_v}\frac{w_je^{O_{DRti}}}{\left(1+e^{O_{DRti}}\right)^2}\\
&-\frac{E6}{2u_v}S_{Ti}\left(1-U_{i,t-1}\right)\frac{e^{i_t}}{\left(1+e^{i_t}\right)^2}
\end{aligned}\right. \tag{8}$$

$$
\begin{cases}
\quad -w \cdot E7\left(1-U_{i,t-1}\right)\left(w\sum_{j=1}^{M(t)} v_{ij} - P_{i\min}\right) \\
\quad -w \cdot E8\left(1-U_{i,t+1}\right)\left(w\sum_{j=1}^{M(t)} v_{ij} - P_{i\min}\right) \\
V_{tij} = 0.5 + 0.5 \cdot \tanh\left(\dfrac{u_{tij}}{u_v}\right)
\end{cases}
$$

It is easy to get the result by solving equation (8) through the euler method.

4. Calculation algorithm

The constraints of M_{Ui} and M_{Di}, which are not taken into consideration in the energy function, will be considered in the algorithm shown in Fig.2.

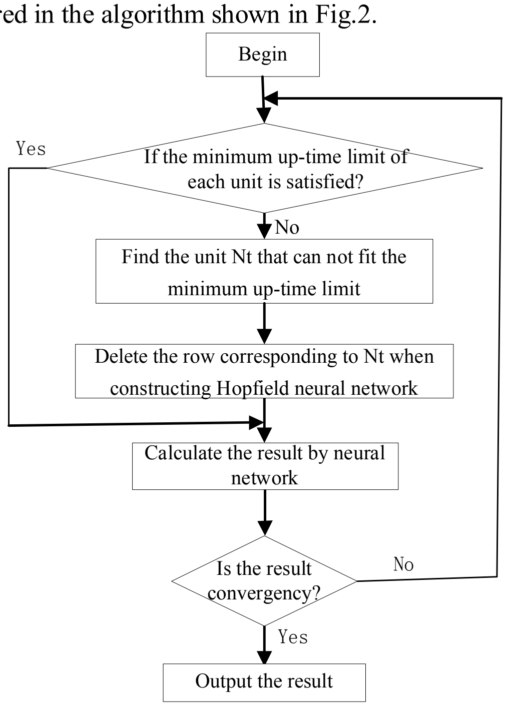

Figure 2. Optimization algorithm

From equation (8) we can see that E4 has little influence on $\dfrac{du_{vij}}{dt}$. When O_{UPti}>0, the square term in denominator makes E4 to be 0. When O_{UPti}<0, the denominator tends to 1, the e function in numerator makes E4 to tend to 0. So, the coefficient E4 should be very large to ensure the fourth part of equation (8) to work. In real calculation, the following codes can ensure the fourth part will work.

if vw>RUP[i] then s4=-E4 else s4=0;

E4 is the coefficient E4 of equation (6), and $vw = \sum_{j=1}^{M(t)} v_{tij} w_j - \sum_{j=1}^{M(t-1)} v_{t-1,ij} w_j$. Real calculations show that such program can improve calculation efficiency and save much calculation time. The same method is adopted to realize the E2, E5 and E6 terms.

5. Example

In order to validate the energy function and algorithm, we took a system which was shown in reference [16] with 10 units as an example. In the calculation, the values of coefficients are E1=2800000, E2=400000, E3=1000, E4=70000, E5=70000, E6=40000, E7=200000 and E8=200000. The calculation result is shown in table2, the fuel cost is 81332.3, the start-up cost is 298.4, and the total cost is 81630.7. Reference [16] calculated the same example, the total cost was 81245.461. There is little difference in the total cost by using Hopfield neural network and another algorithm, but calculating a group of differential functions is somewhat simpler.

Table 2. Calculation result

	1-unit	2-unit	3-unit	4-unit	5-unit
1-hour	0	0	0	104	106
2-hour	0	0	0	90	102
3-hour	0	0	0	101	90
4-hour	0	0	0	66	73
5-hour	0	0	0	79	68
6-hour	0	0	0	71	65
7-hour	0	0	0	65	69
8-hour	0	0	0	69	62
9-hour	0	0	0	40	59
10-hour	0	0	0	36	59
11-hour	0	0	0	34	57
12-hour	0	0	0	32	54
13-hour	0	0	0	32	50
14-hour	15	0	0	28	52
15-hour	15	20	30	26	51
16-hour	15	20	30	25	50
17-hour	15	20	30	25	50
18-hour	15	20	30	25	50
19-hour	19	22	34	30	51
20-hour	20	24	32	31	56
21-hour	18	24	35	34	60
22-hour	22	22	34	35	62
23-hour	21	21	38	34	66
24-hour	23	30	35	39	69
	6-unit	7-unit	8-unit	9-unit	10-unit
1-hour	280	261	385	519	345
2-hour	223	283	445	520	317
3-hour	139	262	397	520	431
4-hour	115	247	329	520	550
5-hour	123	227	391	402	550
6-hour	109	216	339	520	550
7-hour	104	207	305	520	550
8-hour	96	205	274	520	474
9-hour	82	137	209	433	550
10-hour	77	129	180	379	550

11-hour	78	135	158	363	495
12-hour	83	125	136	348	422
13-hour	79	123	138	357	421
14-hour	75	120	140	317	413
15-hour	76	120	125	296	381
16-hour	75	120	125	309	391
17-hour	75	120	137	345	443
18-hour	75	120	152	359	534
19-hour	81	137	197	439	550
20-hour	81	151	266	489	550
21-hour	97	184	298	520	550
22-hour	101	221	333	520	550
23-hour	124	228	348	520	550
24-hour	112	240	372	520	550

6. Conclusion

(1) The unit commitment problem can be changed to a transport problem which can not be subdivided with many constraints.

(2) The energy function of multi-layer Hopfield neural network can take most mathematical constraints into consideration, so we can obtain the result of unit commitment problem by calculating a group of differential functions.

(3) Some items in differential functions can be simplified, and the simplification process will not affect the result precision.

7. References

[1] W.L. Snyder, H.D. Powell Jr, "Dynamic Programming Approach to Unit Commitment", IEEE Trans on Power Systems, Vol.2, No.2, 1987, pp. 339~350.

[2] Wang Cheng-min, Guo Zhi-hong, Yu Er-keng, "An Improved Dynamic Programming Method for Determining Unit Commitment", Power System Technology, Vol.25, No.5, 2001, pp. 20-24.

[3] S.J. Wang, S.M. Shahidehpour, D.S. Kirschen, "Short-Term Generation Scheduling With Transmission and Environmental Constraints Using an Augmented Lagrangian Relaxation", IEEE Trans on Power Systems, Vol.10, No.3, 1995, pp. 1294~1301.

[4] Zhai Qiaozhu, Guan Xiaohong, Lai Fei, "A New Method for Scheduling Hydro-thermal Systems With Identical Units", Proceedings of the CSEE, Vol.22, No.3, 2002, pp. 38-42.

[5] Yang Peng-peng, Han Xue-shan, "Unit Decommitment Based on Improved Lagrangian Multiplier Modification Method", Power System Technology, Vol.30, No.9, 2006, pp. 40-45.

[6] Chen Haoyong, Zhang Kaoshe, Wang Xifan, "Evolutionary Optimization Method of Power System Unit Commitment Problem", Proceedings of the CSEE, Vol.19, No.12, 1999, pp. 9-13.

[7] Wang Zhe, Yu Yi-xin, Zhang Hong-peng, "Social Evolutionary Programming Based Unit Commitment", Proceedings of the CSEE, Vol.24, No.4, 2004, pp. 12-17.

[8] Cai Chaohao, Cai Yuanyu, "Optimization of Unit Commitment by Genetic Algorithm", Power System Technology, vol.21, No.1, 1997, pp. 44-47.

[9] I.G. Damousis, A.G. Bakirtzis, P.S. Dokopoulos, et al., "A Solution to The Unit-Commitment Problem Using Integer-Coded Genetic Algorithm", IEEE Trans on Power Systems, Vol.19, No.2, 2004, pp. 1165-1172.

[10] Yang Jun-jie, Zhou Jian-zhong, YU Jing, et al., "A Hybrid Intelligent Genetic Algorithm For Large-Scale Unit Commitment", Power System Technology, Vol.28, No.19, 2004, pp. 47-50.

[11] Hu Jiasheng, Guo Chuangxin, Cao Yijia, "A Hybrid Particle Swarm Optimization Method for Unit Commitment Problem", Proceedings of the CSEE, Vol.24, No.4, 2004, pp. 24-28.

[12] Zhao Bo, Cao Yijia, "An Improved Particle Swarm Optimization Algorithm for Power System Unit Commitment", Power System Technology, Vol.28, No.21, 2004, pp. 6-10.

[13] Hou Yunhe, Lu Lijuan, Xiong Xinyin, et al., "Enhanced Particle Swarm Optimization Algorithm and Its Application On Economic Dispatch of Power Systems", Proceedings of the CSEE, Vol.24, No.7, 2004, pp. 95-100.

[14] Lou Suhua, Yu Xinmei, Xiong Xinyin, et al., "Unit Commitment Using Improved Discrete Particle Swarm Optimization Algorithm", Proceedings of the CSEE, Vol.25, No.8, 2005, pp. 30-35.

[15] S.V.B Aiyer, M Niranjan, F Fallside, "A Theoretical Investigation Into The Performance of The Hopfield Model", IEEE trans on Neural Networks, Vol.1, No.2, 1990, pp. 204~215.

[16] Han Xueshan, Liu Zuo, "Optimal Unit Commitment Considering Unit's Ramp-Rate Limits", Power System Technology, Vol.18, No.6, 1994, pp. 11-16.