Reinforcement Learning Solution for Unit Commitment Problem Considering Minimum Start Up and Shut Down Times

Jasmin.E.A[1], , Imthias Ahamed T.P.[2], Jagaty Raj V.P.[3]
[1]Dr.E.A.Jasmin, Dept. of Electrical& Electronics, Govt Engg.College,Thrissur, Kerala, eajasmin@gmail.com
[2]Dr.T.P.Imthias Ahamed, Dept. of Electrical& Electronics, T.K.M.College of engineering, Kerala, imthiasa@gmail.com,

***Abstract*— Unit Commitment Problem (UCP) in power system refers to the problem of determining the on/ off status of generating units that minimize the operating cost during a given time horizon. Since various system and generation constraints are to be satisfied while finding the optimum schedule, UCP turns to be a constrained optimization problem in power system scheduling. Numerical solutions developed are limited for small systems and heuristic methodologies find difficulty in handling stochastic cost functions associated with practical systems. This paper models Unit Commitment as a multi stage decision making task and an efficient Reinforcement Learning solution is formulated considering minimum up time /down time constraints. The correctness and efficiency of the developed solutions are verified for standard test systems.**

***Index Terms*— Unit Commitment, reinforcement learning, Q learning.**

I. INTRODUCTION

This paper proposes an efficient Reinforcement Learning (RL) based solution using state aggregation strategy to one of the optimization problems in the power generation sector: Unit Commitment problem (UCP) [1]. Reinforcement Learning based solutions have been proposed to several control and optimization tasks like playing Backgammon [2], robotics and control [3 -5], medical imaging[6] etc.

In the field of power system also a few applications of RL has been proposed [7 - 10]. UCP is one constrained optimization problem in power system scheduling. It involves scheduling the ON / OFF status of a set of units to meet the forecasted load demand over a time horizon under different operational constraints so that the total generation cost is minimized. Since an improved Unit commitment schedule may bring forth savings of large amount for an electric utility, Unit Commitment is an important optimization task in the daily operation planning of power system today.

Priority list methods [11], Dynamic Programming[12], Lagrange Relaxation [13] etc. have been explored by various researchers. Priority List method is simple but the solution obtained is not optimum always. Dynamic Programming provides optimum solution to large scaleproblems [14]. Several soft computing strategies including Genetic Algorithm [15], Simulated Annealing [16] are also being proposed. But these methods are also limited in computational efficiency when a large number of units are to be considered.

We had recently proposed an RL solution to this optimization problem without considering the minimum up time / down time constraints [17]. Since the minimum up time / down time constitutes one of the important boundaries in practical power system operation, to get an efficient scheduling solution, it is recommended to incorporate these unit constraints also.

Having formulated as a Multistage Decision making Problem, implementable solution is proposed using Reinforcement Learning approach. Considering minimum up time and down time constraints, which needs the state of the system (status of the units) to be represented as integer instead of binary representation in the basic solution strategy [17].

The organization of the rest of the paper is as follows. Mathematical formulation of Unit Commitment Problem (UCP) is given in section 2. To make the paper self explanatory a brief description on Reinforcement Learning is given in section 3. In section 4, UCP is formulated as a Multi stage decision making task. The minimum up time / down time constraints are incorporated and an efficient solution is proposed through state aggregation strategy. Performance of the developed algorithms is evaluated in section 5. Concluding remarks are given in section 6.

II. REINFORCEMENT LEARNING

Reinforcement Learning is a learning strategy which discovers a best policy, mapping of situations to actions [18, 19]. By continued interaction with the environment, learning agent discovers the best action suitable for each situation. Learning agent gets the state of the system and chooses a suitable action from the available action set. On performing this action a_k in state x_k, the agent receives a reward from the environment and the system proceeds to the next state x_{k+1}. Reward is a numerical measure of the goodness of the action and depends on the state transition.

DOI: 01.IJRTET.03.03.171

That is, reward $r_k = g(x_k, a_k, x_{k+1})$. Reinforcement learning agent keeps track of the rewards received at different system states which are used in action selection when the same situation arises in future. For the same, Q learning is a widely used method in Reinforcement Learning. Here Q values associated with each state – action pair, *Q(x, a)* is updated based on the reward value on performing an action *a* at state *x*. These Q values of the different actions can then be compared for selecting an action when the same state *x* is encountered in future.

In Q learning algorithm we will first initialize all Q values with some initial value, $Q^0(x_k, a_k)$. At each iteration *n*, on reaching x_k an action a_k is taken based on the current estimate of $Q^*(x_k, a_k)$ ie, $Q^n(x_k, a_k)$. Once action is taken at state x_k, it makes transition to x_{k+1} and the reward $g(x_k, a_k, x_{k+1})$ can be found from simulation. We update the Q value using the equation,

$$Q^{n+1}(x_k, a_k) = Q^n(x_k, a_k) + \alpha \left[g(x_k, a_k, x_{k+1}) + \gamma \min_{a' \in A_{k+1}} Q^n(x_{k+1}, a') - Q^n(x_k, a_k) \right]$$

$0 < \alpha < 1$ is a constant and is called step size of learning. As the learning proceeds, $Q^{n+1}(x_k, a_k)$ will be converging to the optimum value $Q^*(x_k, a_k)$ so that best action can be selected by just finding the greedy action (action having minimum Q value (a_g) or in other words best action found from previous experience) . That is, $a_g = argmin_{a' \in A} Q(x_k, a')$

During the learning phase, the agent should explore the action space at the same time exploit the previous acquired knowledge on good actions found so far. To provide sufficient exploration and exploitation in the action selection, different methods are employed in Reinforcement Learning. One method is ε - greedy method in which exploration rate is decided by the parameter ε. The action which has already found as good in previous attempts is termed as greedy action.

In ε - greedy strategy, on reaching at any state the greedy action is chosen with a probability of (1- ε) while one among the remaining actions from the action space in random is performed with a probability of ε.

III. Unit Commitment Problem as a Multi Stage Decision Making Task

In the case of Unit Commitment Problem, the state of the system at any time slot (hour) *k* can represent the status of each of the *N* units. That is, the state x_k can be represented as a tuple (k, p_k) where p_k is a string of integers, $[p_k^0, p_k^1, \ldots\ldots\ldots\ldots p_k^{N-1}]$, p_k^i is an integer representing the status of the unit. Including the minimum up time and minimum down time constraints force to include the number of time units each unit has been ON /OFF in the state representation**.** Then the variable p_k^i can take positive or negative value ranging from $-D_i$ to $+U_i$, D_i being the minimum down time and U_i the minimum up time.

The part of the state space at time slot or stage *k* can be denoted by

$\chi_k = \{(k, [p_k^0, p_k^1, \ldots\ldots p_k^{N-1}])$, and the state space can be defined as ,

$$\chi = \chi_0 \cup \chi_1 \cup \ldots\ldots\ldots \chi_{T-1}$$

Next is to identify the actions or decisions at each stage of the multi stage problem. In case of UCP, the action or decision on each unit is either to commit or not, the particular unit during that particular hour or time slot. Therefore action set at each stage *k* can be defined as $\mathcal{A}_k = \{[a_k^0, a_k^1, \ldots\ldots\ldots\ldots, a_k^{N-1}], a_k^i = 0 \text{ or } 1\}$. When certain generating units are committed during particular hour *k*, ie, $a_k^i = 1$ for certain values of *i*, then the load demand or power to be generated by these committed units is to be decided. This is done through an Economic Dispatch solution.

The next part to be defined in this MDP is the transition function. Transition function defines the transition from the current state to the next state on applying an action. That is, from the current state x_k , taking an action a_k, it reaches the next state x_{k+1}. Since the action is to make the units ON /OFF, the next state x_{k+1} is decided by the present state x_k and action a_k. Transition function $f(x_k, a_k)$ depends on the state representation.

Last part to be defined is the reinforcement function. It should reflect the objectives of the Unit commitment Problem. Unit Commitment Problem can have multiple objectives like minimization of cost, minimizing emissions from the thermal plants etc. Here, the minimization of total cost of production is taken as the objective of the problem. The total reward for the *T* stages should be the total cost of production. Therefore, the reinforcement function at k^{th} stage is defined as the cost of production of the required amount of generation during the k^{th} period.

That is,

$$g(x_k, a_k, x_{k+1}) = \sum_{i=0}^{N-1} [C_i(P_{i\,k}) u_{i\,k} + ST_i\,(u_{i\,k})(1 - u_{i\,k-1})], \quad (1)$$

Here, P_{ik} is the power generation by i^{th} unit during k^{th} time slot and $u_{i\,k}$ is the status of i^{th} unit during k^{th} time slot and ST_i the start up cost of i^{th}unit.

In short, Unit Commitment Problem is now formulated as a Multi Stage decision making problem, which passes through *T* stages. At each stage *k*, from one of the states $x_k = (k, p_k)$ an action or allocation a_k is chosen depending on some exploration strategy. Then a state transition occurs to x_{k+1} based on the transition function. Each state transition results in a reward corresponding to power allocation to the committed units. Then the problem reduces to finding this optimum action a_k at each state x_k and corresponding to each time slot *k*.

DOI: 01.IJRTET.03.03.171

In the next sections a Reinforcement Learning solutions is proposed using state aggregation strategy

IV. Solution of Unit Commitment Problem Using State Aggregation strategy

While looking into the Unit Commitment Problem with minimum up time and minimum down time constraints, the state space become very huge. The huge state space is difficult to handle in a straight forward manner proposed before [17] especially when the minimum up time / minimum down time increases or the number of generating units increases. Storing of Q value corresponding to each state – action pair becomes computationally expensive. Some method is to be thought of to reduce the number of Q values to be handled. In the perspective of Unit Commitment problem one can group the different states having the same characteristics so that the goodness of the different groups is stored instead of goodness of the different states corresponding to an action. The grouping of states can be done based on the number of hours a unit has been UP or DOWN.

A machine which has been already UP for duration equal to or greater than the minimum up time can be considered as to occupy a state 'can be shut down'.

A unit which is already UP but not have covered minimum up time can be considered as to represent a state 'cannot be shut down'.

An already offline unit which has been DOWN for number of hours equal to or more than its minimum down time can be represented as a state 'can be committed'.

A unit which has been DOWN but has not covered the minimum down time so that cannot be committed in the next immediate slot of time can be represented as a state 'cannot be committed'.

From the above explained categorization of states, in order to store the Q values, the state x_k is mapped into set of aggregate states. Each aggregate state $ag\_x_k = (k, [\, ag\_p_k^0, ag\_p_k^1, \ldots \ldots ag\_p_k^{N-1}\,]), ag\_p_k^i \in \{0,1,2,3\}$.

$ag\_p_k^i$ can be found corresponding to any $x_k = (k, [\, p_k^0, p_k^1, \ldots \ldots p_k^{N-1}\,])$ as:

p_k^i positive and $p_k^i \geq U_i$, $ag\_p_k^i = 0$;
p_k^i positive and $p_k^i < U_i$, $ag\_p_k^i = 1$;
p_k^i negative and $p_k^i \leq D_i$, $ag\_p_k^i = 2$;
p_k^i negative and $p_k^i > -D_i$, $ag\_p_k^i = 3$.

Thus, at any slot of time, each of the generating unit will be in any of the above mentioned four representative states. If these four conditions are denoted as decimal integers (0, 1, 2, 3), regardless of the UP time and DOWN time of a generating unit, the state is represented by one of this integer value. By aggregating the numerous states visited in the previous algorithm to a limited number of states, number of Q values to be stored and updated in the look up table is greatly reduced.

Now, for an N generating unit problem, there will be 4^{N-1} possible states, regardless of minimum up time and down time of the different units. This reduction in the number of states drastically reduces the size of look up table for storing the Q values. Now an algorithm is formulated making use of state aggregation technique for handling the up/ down constraints of the units.

The number of states, nstates is initialized to 4^{N-1} and the number of actions naction to 2^{N-1} for an N generating unit system. At any stage *k* of MDP, the state of the system is represented as a string of integers as in the previous algorithm, integer value representing the number of hours the unit has been up or down.

From any state x_k an action is selected using one of the exploration strategies. On selecting an action a_k, the ON / OFF status of the units will change as, $x_{k+1} = f(x_k, a_k)$

The reward function for the state transition is found using the cost evaluations of the different generating units. At each state *k*, estimated Q value corresponding to the state – action pair (ag_x_k, a_k) is updated using the equation,

$$Q^{n+1}(ag\_x_k, a_k) = Q^n(ag\_x_k, a_k) + \alpha\,[\, g(x_k, a_k, x_{k+1}) + \gamma \min_{a' \in \mathcal{A}_{k+1}} Q^n(ag\_x_{k+1}, a') - Q^n(ag\_x_k, a_k)] \quad (2)$$

During the last hour, omitting the term to account future pay –off Q value is updated using the equation,

$$Q^{n+1}(ag\_x_k, a_k) = Q^n(ag\_x_k, a_k) + \alpha\,[\, g(x_k, a_k, x_{k+1}) - Q^n(ag\_x_k, a_k)] \quad (3)$$

After a number of iterations, learning converges and the optimum schedule or allocation for each state xk can be easily retrieved after finding the corresponding aggregate state as,

$$a_k^* = argmin_{a_k \in \mathcal{A}_k} \{\, Q(ag\_x_k, a_k)\,\}, k = 0, \ldots . T-1,$$

The entire algorithm using state aggregation method is given below:

TABLE I
ALGORITHM FOR UNIT COMMITMENT THROUGH STATE AGGREGATION

Read Unit Data
Read the Load forecast for next T hours.
Initialize nstates (number of states) and nactions (number of actions)
Initialize $Q_0\,[ag\_x_k, a_k] = 0$, $\forall\, ag\_x_k$, $\forall\, a_k$
Find the set of permissible actions corresponding to each hour k
Initialize the learning parameters
For n=1 to max _ episode
Begin
Read the initial status of the units x0
For k=0 to T-1
Do
Find aggregate state ag_ xk corresponding to xk
Find the feasible set of actions $\mathcal{A}k$ *corresponding to state xk* considering *up and down times.*
Choose an action using ε*- greedy strategy from the feasible set of actions*

DOI: 01.IJRTET.03.03.171

Find the next state x_{k+1}
Find the corresponding aggregate state ag_x_{k+1} *of* x_{k+1}
Calculate the reward $g(x_k, a_k, x_{k+1})$
If (k != T-1) Update Q value using equation (2)
Else Update Q value using equation (3)
End do
Update the value of ε.
End
Save Q values.

The optimal schedule $[a_0^*, a_1^*, \ldots . a_{T-1}^*]$ is by choosing the greedy action at each stage.

V. Simulation Results

For efficient learning, we have to choose suitable values for the learning parameters α and γ. The parameter α decides the extent to which a training sample modifies the Q value and its value affects the speed of convergence and accuracy of result. By trial and error we choose a value of 0.1. The discount parameter γ is chosen based on the nature of the problem and in this case we choose a value of 1.

In order to validate the algorithm, four generating unit system with different minimum up time and down time are considered. The unit characteristics and load profile for a four generating unit system are given Table II and III. The obtained commitment schedule is given in Table V.

For comparing with the recently developed stochastic strategies a ten generating unit system with different minimum up time and down time limits are taken for case study. The generating unit details are given in Table VI. The schedule obtained and the computation time is compared with two hybrid methodologies: Simulated Annealing with Local Search (SA LS) and Lagrange Relaxation with Genetic Algorithm (LRGA)

TABLE II
UNIT CHARACTERISTICS

Unit	Pmin (MW)	Pmax (MW)	Inc. cost	No Load Cost	Startup Cost	Min.Up time	Min.Down time
1	75	300	17.46	684.74	1100	4	2
2	60	250	18	585.62	400	5	3
3	25	80	20.88	213	350	5	4
4	20	60	23.8	252	0.02	1	1

TABLE III
LOAD PROFILE

Hour	0	1	2	3
Load	450	530	600	540
Hour	4	5	6	7
	400	280	290	500

TABLE IV
COMMITMENT STATUS OF UNITS

Hour	Status	Hour	Status
1	0110	5	0110
2	0110	6	0110
3	0111	7	0110
4	0110	8	0110

TABLE V
UNIT CHARACTERISTICS OF TEN UNIT SYSTEM

Unit	P min (MW)	P max (MW)	a	b	c	Start up cost (Rs.)	Min.Up hrs.)	(hrs.)	Initial status
	15					450			
1	0		100			0			
2	15	455	0	16.19	0.00048	400	8	8	8
3	0	455	970	17.26	0.00031	0	8	8	8
4	20	130	700	16.6	0.00211	550	5	5	-5
5	20	130	700	16.6	0.002	360	5	5	-5
6	25	160	450	19.7	0.00031	300	6	6	-6
7	20	85	370	22.26	0.0072	340	3	3	-3
8	25	85	480	27.74	0.00079	520	3	3	-3
9	10	55	660	25.92	0.00413	60	1	1	-1
1	10	55	665	27.37	0.00222	60	1	1	-1
0	10	55	670	27.79	0.00173	60	1	1	-1

TABLE VI
COMPARISON WITH OTHER RECENT METHODS

Algorithm	Cost(Rs.)	Execution Time(sec.)
LRGA	564800	518
SA LS	535258	393
RL_UCP4	545280	268

VI. Conclusion

In this paper Reinforcement learning is suggested as a good solution strategy for solving one of the major optimization problems in the power system. Several Numerical and stochastic solutions have been proposed for solution of this constrained optimization problem. Reinforcement Learning provides with a good and faster solution strategy which provide optimum scheduling with lesser computation time. The developed solution is also capable of handling the stochastic nature of cost functions and uncertainty associated with the availability of

DOI: 01.IJRTET.03.03.171

generating units. Thus it provides with a more suitable solution strategy for practical generator scheduling.

In this paper, only thermal generating units are considered As a next step the algorithm can be made to take actual data from a practical system incorporating other generating sources. Also the solution strategy provides with the scope of solving the power system problems in an efficient and faster mode.

REFERENCES

[1] A.J.Wood, B.F.Wollenberg., "Power Generation and Control" John Wiley Sons 2002.
[2] G.J. Tesauro, TD Gammon. "Temporal difference Learning and TD gammon", Communications of ACM, 38 (3) (1995): 58 – 67.
[3] Robert H.Crites, Andrew G.Barto, "Elevator control using multiple Reinforcement Learning Agents" Kulwer Academeic Pulishers, Boston , (1997)
[4] Hirashi Handa, Akira Ninimy, "Adaptive state construction for Reinforcement Learning and its application to Robot Navigation problem", IEEE Transaction on Industrial Electronics, 7(2) 1436: 1441, 2001.
[5] Andrew Y.N., A.Coats, M.Diel, V.Ganpapthi, "Autonomous helicopter flight via Reinforcement Learning" Symposium on Experimental robotics, (2004).
[6] R.S.Sutton, A.G.Barto. "Reinforcement Learning : An introduction", MIT Press, Cambridge, MA, 1998
[7] Farhad Sahba, Hamid R.Tizhoosh, "Application of Reinforcement Learning for segmentation of transrectal ultra sound images", BMC Medical Imaging, 2008
[8] T.P.Imthias Ahamed, P.S.nagendra Rao, P.S.sastry "A Reinforcement Learning approach to automatic generation control" Electric Power System research 63 (2002): 9-26
[9] Damien Earnst, Mevludin Glavic, "Power system stability control: A Reinforcement Learning framework.", IEEE Transactions on Power Systems, 19 (1) (2004).
[10] D. Ernst, M.Glavic "Approximate value iteration in the Reinforcement Learning context: Application to Electric Power system control" International journal of Emerging Electric Power Systems, 3(1) (2005).
[11] E.A.Jasmin,T. P. Imthias Ahamed,V.P.Jagathyraj "A Reinforcement Learning algorithm to Economic Dispatch considering transmission losses", Proceedings of TENCON 2008.
[12] F.N.Lee, "Short term Unit Commitment – a new method", *IEEE Transactions on Power Systems* 99 (2) (1988): 691 – 698.
[13] Walter L.Snyder, H.david Powell. "Dynamic Programming approach to Unit Commitment", IEEE Transactions on Power Systems, 2 (2) (1987): 339 – 349.
[14] John A. Muckstadt, Sherri A.Koenig. "An Application of Lagrange Relaxation to scheduling in power generation systems", Operations research, 25 (3) (1977): 387 – 403.
[15] G.S.Lauer, N.R. Sandell, D.P. Bertsekas, T.S.Posbergh, Solution of Large scale optimal Unit Commitment problems, IEEE Transactions on Power Apparatus and Systems,101 (1) (1982): 79 - 86.
[16] Charles W.Richter, Gerald B.Sheble, "A profit based unit commitment GA for the competitive environment", IEEE Transactions on Power systems 15, 2 (2000): 715 – 722.
[17] A F.Zhaung and F.D.Galiana, "Simulated Annealing Approach to Unit Commitment solution", *IEEE Transactions on Power Systems* 5(1) (1990) : 311 – 317.
[18] E.A.Jasmin,T. P. Imthias Ahamed,V.P.Jagathyraj, " Reinforcement Learning solution to Unit Commitment problem through pursuit method" *Proceedings of International Conference ACEE 2009,*.
[19] D.P.Bertsekas, J.N. Tsitsikilis. "Neuro Dynamic Programming" Athena Scientific, Belmount MA., 1996.

DOI: 01.IJRTET.03.03.171