Proceedings of the Fourth IASTED International Conference
POWER AND ENERGY SYSTEMS (AsiaPES 2008)
April 2-4, 2008 Langkawi, Malaysia
ISBN CD: 978-0-88986-732-1

UNIT COMMITMENT SOLUTION USING AGGLOMERATIVE AND DIVISIVE CLUSTER ALGORITHM - AN EFFECTIVE NEW METHODOLOGY

N. Malla Reddy[1] Dr. N.V.Ramana[2] Dr. K.Ramesh Reddy[1]
nomula_nomula@yahoo.co.in nvrjntu@yahoo.co.in kollirameshreddy@yahoo.com
[1]Faculty, Electrical Department, G.Narayanamma Institute of Technology and Science, Hyderabad, A.P, INDIA.
[2]Professor, Electrical Department, JNTU College of Engineering, Jagityala, A.P, INDIA

ABSTRACT
A new methodology for the most complex unit commitment problem using agglomerative and divisive hierarchical clustering is presented. Euclidean costs, which is a measure of difference in fuel cost and start-up costs of any two units are first calculated. Depending upon the value of Euclidean costs, similar type of units are placed in a cluster. This cost is also useful for preparing priority lists for the units in a cluster and forming the different clusters. Proposed methodology has two individual algorithms. While the load is increasing, agglomerative cluster algorithm is proposed. Divisive cluster algorithm is used when the load is decreasing. The search is carried for an optimal solution from a minimal number of clusters and cluster data points. The performance of the method is evaluated on a standard ten thermal unit power system for a period of 24 hours. Numerical results are presented and compared with the existing popular methods in most judicious way.

KEY WORDS
Power System operation and control, Unit Commitment Solution, Cluster Algorithms, Objective Function and Constraints.

1. Introduction

A Power Generating plant consists of number of generating units with different generating capacities, fuel cost per MWH generated, minimum up/down times, start-up/shut-down costs etc. The Unit Commitment (UC) problem in power system involves determining the start-up and shut-down schedules of thermal generating units to meet forecasted load over a future short term for a period of one to seven days. UC is defined mathematically as a nonlinear, mixed integer complex combinational optimization problem. The objective of UC solution is to minimize total production costs while observing large number of operating constraints. The exact solution of UC can be obtained by complete enumeration, which cannot be applied at real time environment due to excessive computational effort that shall lead to high computational time [1]. To date, researchers have proposed various methodologies to mitigate UC, can be mainly categorized into: *Mathematical Programming* and *Intelligent* methods.

The most talked deterministic mathematical programming methods include: Priority List [1-2] Dynamic Programming [1,3], Lagrange Relaxation [1,4], Branch-and-Bound [5], Integer and Mixed- Integer methods [6] and annealing method[7]. Mathematical methods are impractical in terms of computational effort, time and memory requirements when considering many units or a longer study period. It may be observed these methods have following general *limitations*: i) They are not guaranteed to converge to global optimum of the general non convex problems like UC. ii) Inconsistency in the results due to approximations made while linearising some of the nonlinear objective functions and constraints. iii) Consideration of certain constraints makes difficulty in obtaining the solution. iv) The process may converge slowly due to the requirement for the satisfaction of large number of constraints.

Therefore, research interest has been focused on heuristic search methods based on Artificial Intelligent techniques. Intelligent methods have come to be popular tool for solving many optimization problems. Recent contributions in the area of intelligent methods for UC problem include application of Expert Systems [8], Neural Networks [9], Fuzzy Logic [10], and Tabu Search [11], Basic Genetic Algorithm approach [12] and Extended Priority List (EPL) method based on GA [13]. These methods seem to be promising and are still evolving. It can be observed that, the optimization problems with large number of constraints are quite difficult by solving neural networks and fuzzy logic approaches. Though the approaches have yielded attractive results, the linguistic descriptions for generating static crisp output under large number of time dependent constraints in UC may make the approaches highly complex and may even confusing. The GA approach similar to Tabu search, iteratively evaluates the best solution each time the neighborhood is updated. The method has good capability only for suboptimal search. In general, GA is a typical heuristic method. Literature in GA for UC is vast.

The following limitations may be observed in the popular GA approach: i) The solution deteriorates with the increase of chromosome length. Hence, to limit its size, limitations are imposed in consideration of number of control variables. ii) GA method tends to fail with the more difficult problems and need good problem knowledge to be tuned. iii) Careless representation in any of the schemes that are used in the formation of chromosomes shall nullify the effectiveness of mutation and crossover operators. iv) The use is restricted for small problems such as those handling less variables, constraints etc. GA is a stochastic approach where the problem is not guaranteed to be the optimum. v) Execution time is high. vi) The quality of solution is found to be deteriorating with the increase in length of chromosome i.e. the UC problem size. vii) If the size of power system is growing, the GA approach can produce more infeasible strings which may lead to wastage of computational time, memory etc.

Inspired by the results of GA method and to overcome the general difficulties in GA approach, a novel method with the application of cluster algorithms [14,15] is proposed in this paper. The method uses Agglomerative and Divisive Cluster Algorithms. The proposed methodology can be unfolded in to three stages. In the first stage, four clusters are formed namely base load, intermittent load, semi-peak load and peak load clusters. All the generating units of the plant are segregated into corresponding clusters based on Euclidean costs. This cost is a measure of difference in fuel cost and start-up costs of any two units and is useful in forming clusters in addition to preparing the priority list for cluster units. In the second stage, UC solution is obtained by developing Agglomerative Cluster (AC) algorithm for increasing load pattern. Finally in the third stage a Divisive Cluster (DC) algorithm is developed for decreasing load pattern.

The remaining paper is organized as follows: Section 2 deals with problem formulation; General purpose agglomerative cluster and divisive cluster algorithms are discussed in the Section 3. The new methodology ACDC algorithm is presented in Section 4. Simulation results and discussions can be seen in Section 5 and finally conclusions are drawn in the Section 6.

2. Unit Commitment Problem Formulation

Subject to the minimization of the cost-objective function in the unit commitment problem, certain units are stated to be as 'ON' and remaining as 'OFF'.

2.1 Notations

N :Number of generating units in the plant; T:Scheduling period in hours (h); i:Index of Unit (i = 1,2,….., N); t:Index of time (t = 1,2,….., T); $I_i(t)$:i^{th} unit status at t^{th} hour (= 1, if the Unit is ON; =0, if the unit is OFF) ; $P_i(t)$: Generation of i^{th} unit at t^{th} hour; P_i^{max}, P_i^{min} : Maximum / Minimum output power (MW) of i^{th} unit; $D(t)$:Demanded power at t^{th} hour; $R(t)$:System reserve at t^{th} hour; T_i^{on}:Minimum up time of i^{th} unit; T_i^{off}: Minimum down time of i^{th} unit; $X_i^{on}(t)$:Duration during which i^{th} unit is continuously ON;$X_i^{off}(t)$:Duration during which i^{th} unit is continuously OFF; $SC_i(t)$: Start-Up cost of i^{th} unit; $FC_i(t)$:Fuel cost of i^{th} unit; TC:Total Cost of generation; $HC(i)$: Hot start cost of i^{th} unit; $CC(i)$: Cold start cost of i^{th} unit; $CS(i)$: Cold start hour of i^{th} unit; a_i, b_i, c_i : Fuel cost coefficients.

2.2 Objective Functions

The objective function of UC problem is the minimization of the TC which has the components of FC and SC and is given by:

$$\text{Min } (TC) = \sum_{t=1}^{T}\sum_{i=1}^{N}(FCi(t)+SCi(t)) \quad (1)$$

where Fuel cost of i^{th} unit:

$$FC_i(t)=a_i+b_i\,P_i(t)+c_i\,P_i(t)^2 \quad (2)$$

and Start-up cost

$$SCi(t) = HC(i) : \text{ if } T_i^{off} \le X_i^{off}(t) \le H_i^{off}(t) \quad \text{or}$$

$$= CC(i) : \text{ if } X_i^{off}(t) \ge H_i^{off}(t) \quad (3)$$

where $H_i^{off}(t) = T_i^{off} + CS(i)$ (4)

2.3 System Constraints

The constraints, which must be considered during the optimization process of UC problem (1), are given below.

2.3.1 Load Demand

All the committed units must generate total power equal to load demand as:

$$D(t)=\sum_{i=1}^{N}Pi(t) \quad (5)$$

2.3.2 Spinning Reserve

To maintain system reliability for sudden variation of loads, system should have adequate amount of spinning reserve capacity. In this paper more than 10% of the load demand is taken and which satisfies:

$$\sum_{i=1}^{N}Ii(t)\cdot P_i^{\max} \ge D(t)+R(t) \quad (6)$$

2.3.3 Generated Power Limits

The power output of each unit should satisfy:

$$P_i^{min} \le P_i(t) \le P_i^{max} \quad (7)$$

2.3.4 Minimum Up/Down Time

Once the unit is committed there is a minimum time before it is de-committed and viz.

$$T_i^{on} \le X_i^{on}(t) \text{ or } T_i^{off} \le X_i^{off}(t) \quad (8)$$

3. Cluster Algorithms

The purpose of Cluster Algorithms (CA) can be stated as, to divide a given group of objects into a number groups or clusters in order that the objects in a particular cluster would be similar among the objects of the other ones. In the first stage of CA, an attempt is made to place an N object in M clusters according to some optimization criterion additive to clusters. Once the optimization criterion is selected, CA searches the space of all classifications and finds the one that satisfies the optimization function. For detailed cluster analysis that includes basic concepts, algorithms and types, the interested reader can refer in [15].The proposed methodology for UC problem considers two clustering techniques: Agglomerative Clustering Technique and Divisive Clustering Technique.

In the first type of cluster technique, initially individual data points are treated as clusters. Based on some criteria (say minimum Euclidean distance values of various data points) successively two closest clusters are merged until there is only one cluster remains. The basic algorithm is given in [15] and is given below for convenience.

3.1 Basic Agglomerative Clustering Algorithm

Step-1: Compute Euclidean Distance (Proximity) Matrix; Step-2: *Repeat;* Step-3: Merge two closest clusters based on least distance value; Step-4: Update the proximity Matrix to reflect the proximity between the new cluster and the original clusters; Step-5: *Until* Only one cluster remains.

In the Divisive type clustering technique, successively each cluster is separated from the others until a singleton cluster of individual point(s) remain. A suitable methodology is required to take the decision on which cluster must be removed from the others. The basic algorithm is given below.

3.2 Basic Divisive Clustering Algorithm

Step-1: Compute Euclidean Distance (Proximity) Matrix; Step-2: *Repeat;* Step-3: Separate a cluster from other clusters based on maximum distance value; Step-4: Update the proximity Matrix to reflect the proximity between the clusters those remaining; Step-5: *Until* All the clusters are removed.

4. Proposed Methodology for Unit Commitment Problem

The proposed methodology can be unfolded in to three stages.

Stage-1:This stage carries some pre-requisite calculations (such as determining Euclidean costs etc.) for the purpose, to prepare a priority list for all the units in the plant and to formulate clusters. Each cluster contains similar type of units, judged based on Euclidean cost values. Section 4.1 which follows, shall demonstrate stage-1 in detail.

Stage-2: The pattern of load variation on the plant is a cycle of increasing and then decreasing takes place. Two separate algorithms are designed for load increasing pattern and for decreasing pattern. In this stage, an algorithm based on agglomerative clustering technique is developed for increasing load pattern. Section 4.2 presents the algorithm.

Stage-3: This stage presents an algorithm for UC solution for the decreasing load condition. The algorithm is designed based on divisive clustering technique. Section 4.3 presents the algorithm proposed.

Flow Chart in Fig.1 explains these stages. The following discussion is important in developing strategy:

Discussion-1: Characteristics of the Units: Base load (BL) and Intermittent load (IL) units operate for long period in the day and they generate more number of units (MWH). Therefore, ideally speaking they should have minimum fuel cost, maximum generating capabilities but, can have high start-up costs and start-up times for the reason they are switched 'on' for the most of time. In addition, System reliability aspect is decided by the performance of these units. Semi-Peak Load (SPL) and Peak load (PL) units in contrast should have low start-up costs and start-up times as these units are rapidly switched 'on' and 'off' frequently. These units can have less generating capabilities and can have relatively high costs as they take up small loads above high base load and intermittent loads.

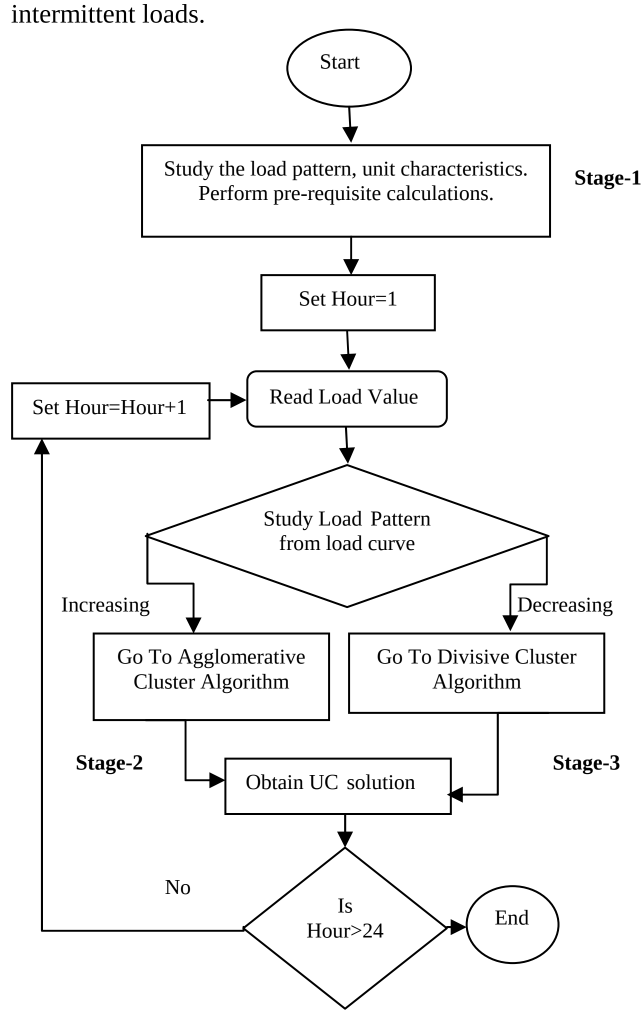

Figure 1. Flow Chart explain stages in ACDC Methodology (Main Program)

Discussion-2: Priority List of the various units with in the plant: The variation of load on the plant is, for some hours it increases; then decreases and cycle repeats. Fig.2 provides these details. Usually UC solution begins first committing units for increasing load and then to de-commit the units for decreasing load. Committing and de-committing the units is subject to the minimization of objective function and satisfaction of the constraints. The proposed methodology provides two separate algorithms for this purpose. Firstly, each generator unit is taken as an individual cluster. These individual clusters are descending priority listed. The priority of the units for the two algorithms is:

For Agglomerative clustering Algorithm (ACA): Priority list of units in descending order based on their decreasing generation capacity, decreasing start-up cost/time and increasing cost of generation.

For Divisive clustering Algorithm (DCA) Priority list of units in descending order based on their increasing generation capacity, increasing start-up cost/time and decreasing cost of generation.

System for Simulation Study: The simulation study includes test run on a ten unit standard system for a 24-hour demand schedule.

The data is presented in the Tables 1 and 2 in Section 4.1.

4.1 Stage-1: Pre-Requisite Calculations

In this stage, the following steps are to perform for preparing a priority list of units.

Step-1: Draw the forecasted chronological load curve from the forecasted load pattern of a season. The load pattern is given in Table-1 and the Load curve is drawn and is shown in Fig.2.

Step-2: Draw Load Duration Curve from the Load Curve. Load Duration curve describes the duration of occurrence of each predicted load. In this curve, load elements of load curve are arranged in descending order with respect to their duration of occurrence as shown in Fig.3.

Step-3: Divide the Load Duration Curve into four areas to represent Base Load (BL), Intermittent Load (IL), Semi-Peak Load (SPL) and Peak- Load (PL). The threshold values (max.) of these loads are taken as:

BL: Load: up to 1000 MWs. Dration; 0-24h; *IL*: Load : between 1000-1200 MWs, Duration: 0-18 h

SPL: Load; 1200-1400 MWs, Duration: 0-6h; PL: Load: 1400-1500 MWs, Duration:0-3h

Step-4: Set the maximum limits for the four loads as:

BL-Max: 1000 MW; IL-Max: 1200 MW; SPL-Max: 1400 Mw and PL-Max: 1500 MW.

Step-5: Read the Generator (Unit) Characteristics. Table-2 below provides various details of all the Units.

Step-6: Compute Average Fuel cost at maximum power generation and average start-up costs. The values are computed and are given in Table-3 below.

Step-7: Compute Euclidean cost of all the units using average fuel and start-up costs. These costs are given in Table-4.

Step-8: Following the discussions 1 and 2 two separate priority lists are prepared as shown in Table-5.

Step-9: Segregate 10 units in to four clusters, using the Euclidean Costs. Table-6 furnishes cluster information.

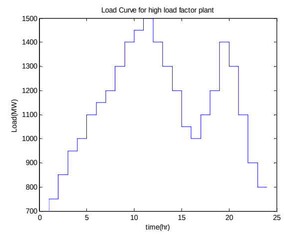

Figure 2. Daily Load Curve of the plant

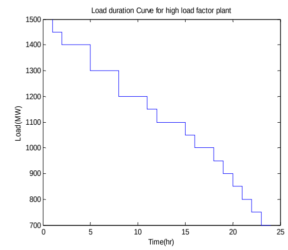

Figure 3. Daily Load Duration Curve of the plant

4.2. Stage-2: Design of Agglomerative Clustering Algorithm for UC Problem

Section-2 provides the general purpose Agglomerative Cluster (AC) Algorithm. This section presents the AC algorithm applied for UC problem. The priority list of various units and strategy has been presented earlier. The method is described in flow chart Fig.4. The algorithm has following steps:

Table 1
Daily Load pattern on the plant

Hour	Load (MW)	Hour	Load (MW)	Hour	Load (MW)	Hour	Load (MW)	Hour	Load (MW)	Hour	Load (MW)
1	700	5	1000	9	1300	13	1400	17	1000	21	1300
2	750	6	1100	10	1400	14	1300	18	1100	22	1100
3	850	7	1150	11	1450	15	1200	19	1200	23	900
4	950	8	1200	12	1500	16	1050	20	1400	24	800

Table 2
Unit Characteristics and Cost Coefficients

Unit No. (*i*)	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
P_i^{max} (MW)	455	455	130	130	162	80	85	55	55	55
P_i^{min}(MW)	150	150	20	20	25	20	25	10	10	10
a_i	1000	970	700	680	450	370	480	660	665	670
b_i,	16.19	17.26	16.6	16.5	19.7	22.26	27.74	25.92	27.27	27.79
c_i	0.00048	0.00031	0.002	0.00211	0.00398	0.00712	0.00079	0.00413	0.00222	0.00173
T_i^{on}	8	8	5	5	6	3	3	1	1	1
T_i^{off}	8	8	5	5	6	3	3	1	1	1
HC(i) ($)	4500	5000	550	560	900	170	260	30	30	30
CC(i) ($)	9000	10000	1100	1120	1800	340	520	60	60	60
CS(i)	5	5	4	4	4	2	2	0	0	0

Table 3

Average Fuel and Start-up Costs of Units

Unit No. (*i*)	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
A	18.6062	19.53292	22.24462	22.00507	23.12254	27.4546	33.45421	38.14715	39.48301	40.06697
B	9.89011	10.98901	4.230769	4.307692	5.555556	2.125	3.058824	0.545455	0.545455	0.545455

A = Avg. Fuel Cost ($/MW): B = Avg. Start up cost ($/MW)

Euclidean Cost of i[th] Units = $\sqrt{(A_i - A_{low})^2 + (B_i - B_{low})^2}$

A_i = Avg fuel cost of i[th] unit, B_i = Startup cost of i[th] unit,
A_{low} = Avg fuel cost of the unit with lowest fuel cost
B_{low} = Startup cost of the unit having lowest startup cost

Table 4

Euclidean Cost of 10 Units

Ini.State	8	8	-5	-5	-6	-3	-3	-1	-1	-1
	Unit-1	Unit-2	Unit-3	Unit-4	Unit-5	Unit-6	Unit-7	Unit-8	Unit-9	Unit-10
Unit-1	0	1.437495	6.728019	6.535726	6.259847	11.77247	16.34411	21.66036	22.87277	23.40699

Table 5

Priority order of various units with respect to Euclidean Cost value of Unit-1

Priority order	1	2	3	4	5	6	7	8	9	10
For ACA	1	2	5	4	3	6	7	8	9	10
For DCA	10	9	8	7	6	3	4	5	2	1

Table 6

Segregation of 10 units into Clusters and their priority

Cluster Type	Base Load Cluster	Intermittent Load Cluster	Semi-Peak Load Cluster	Peak-Load Cluster
Priority Units in the Cluster	1,2	5,4,3	6,7	8,9,10

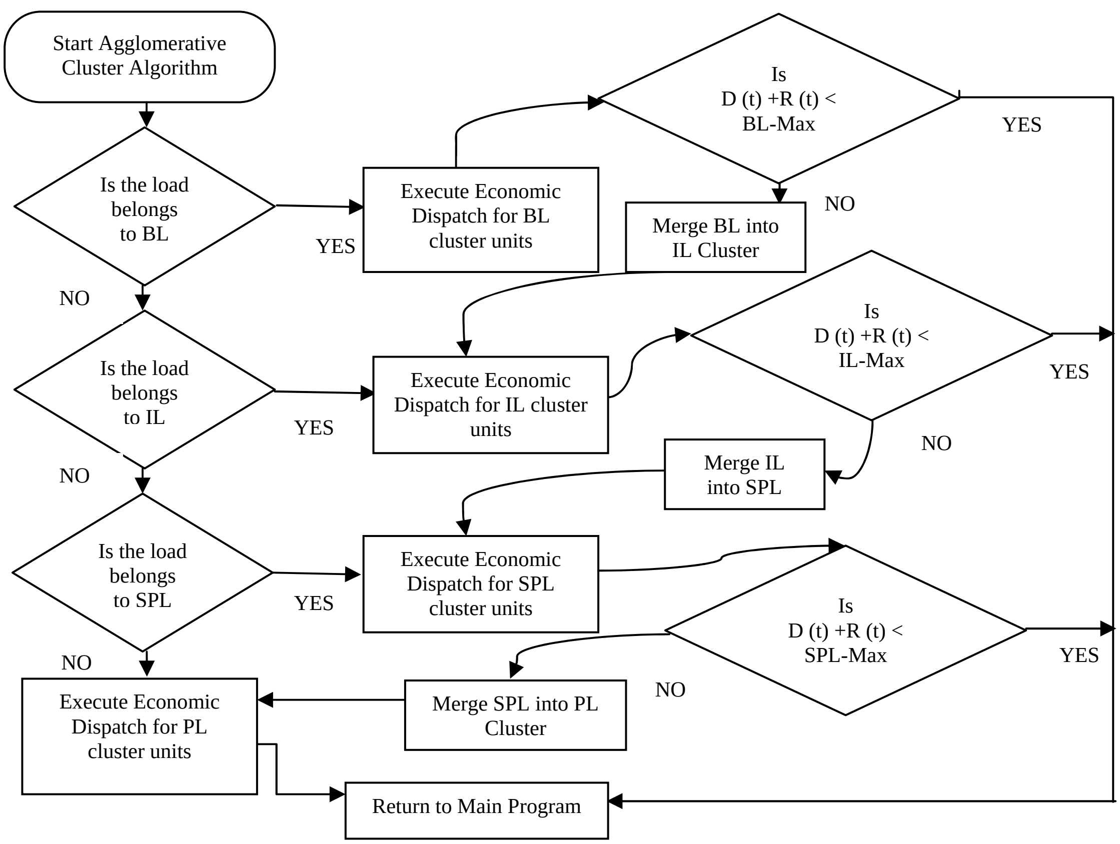

Figure 4. AC algorithm for UC problem

4.2.1 AC Algorithm

Step-1: Read the load value $D(t)$. Spinning Reserve requirement $R(t)$. Threshold values of four clusters.

Step-2: From the load duration curve, identify the load as any: BL, IL, SPL or PL. Step-3: Commit the units in corresponding cluster by executing subroutine for Economic Dispatch (ED). Step-4: Check the constraint: $D(t) + R(t) <$ Cluster Threshold value. If condition is satisfied, go to main program. Else, go to next step. Step-5: Merge next priority list cluster to previous cluster. Step-6: Go to Step-4; Step-7: Return

The subroutine for ED is standard Lambda-Iteration Method. The ED has following steps.

4.2.2 ED by Lambda-Iteration Method

Step-1: Set λ value. Step-2: Calculate P_i for $i = 1,2 \ldots n$. Where n is the number of units in the cluster. P_i is calculated subject to the minimization of objective function (1) under the constraints (5)-(8). Step-3: Calculate error € value (difference between demanded load and sum of generations). Step-4: Check € with tolerance value. If yes Go to main program to print UC results. Else set new value of λ. Go To Step-2.

4.3 Stage-3: Design of Divisive Clustering (DC) Algorithm for UC Problem

This DC algorithm is proposed for UC when the load is decreasing after it stopped from increasing. The DCA starts at the point where some units in various clusters are already under 'on' condition. Now the requirement is to put some units under 'off' condition, so as to meet the present $D(t)$. The priority list is prepared based on the start up time/costs. The strategy is, to put off the unit with minimum start-up time first; next high start-up time unit second and so on. This strategy is undertaken keeping system reliability in to consideration. Even when the load is suddenly is increased after is decreased, there will be no problem to put back the unit back in to service as the start-up time of the unit is low. However, in this paper, the DCA strategy is not implemented for *must-run* units such as BL and IL units.

4.3.1 DC Algorithm

Step-1: Read the system load. Step-2: De-Commit the next unit with minimum start-up time according to priority list. Step-3: Commit the units in corresponding cluster by executing subroutine for Economic Dispatch (ED). Step-4: Check the constraint: $D(t)+R(t) \leq$ sum of all generations. If condition is satisfied, go to main program. Else, go to step-2. Step-5: Return

5. Simulation Results and Discussion

The performance of the proposed ACDC algorithm is tested on a 10 unit thermal plant. The data required to carryout simulation study is presented in Section 4.1. The simulation is performed on a 1.4 GHZ Pentium-4 personal computer. Coding is through MATLAB. Simulation results are presented in Table-7.

Table 7
UC Solution for 10- Unit system by proposed ACDC

S.No	Load(MW)	COMMITMENT SCHEDULE-UNIT NUMBERS										Operational Cost ($)	Start up cost ($)
		1	2	3	4	5	6	7	8	9	10		
1	700	455	245	0	0	0	0	0	0	0	0	13591	0
2	750	455	295	0	0	0	0	0	0	0	0	14454	0
3	850	455	370	0	0	25	0	0	0	0	0	16693	900
4	950	455	455	0	0	40	0	0	0	0	0	18459	560
5	1000	455	390	0	130	25	0	0	0	0	0	19899	550
6	1100	455	455	0	130	60	0	0	0	0	0	21722	0
7	1150	455	410	130	130	25	0	0	0	0	0	23137	0
8	1200	455	455	130	130	30	0	0	0	0	0	24013	0
9	1300	455	455	130	130	110	0	20	0	0	0	26668	520
10	1400	455	455	130	130	162	48	20	0	0	0	29202	340
11	1450	455	455	130	130	162	80	20	18	0	0	31073	60
12	1500	455	455	130	130	162	80	20	40	10	10	32767	120
13	1400	455	455	130	130	162	38	20	10	0	0	29775	0
14	1300	455	455	58	130	162	20	20	0	0	0	26976	0
15	1200	433	455	20	130	162	0	0	0	0	0	23984	0
16	1050	455	303	0	130	162	0	0	0	0	0	20894	0
17	1000	455	390	0	130	25	0	0	0	0	0	19899	0
18	1100	455	455	0	130	60	0	0	0	0	0	21722	0
19	1200	455	455	130	130	30	0	0	0	0	0	24013	0
20	1400	455	455	130	130	162	38	20	10	0	0	29775	920
21	1300	455	455	58	130	162	20	20	0	0	0	26976	0
22	1100	455	353	0	130	162	0	0	0	0	0	21757	0
23	900	455	283	0	0	162	0	0	0	0	0	17688	0
24	800	455	345	0	0	0	0	0	0	0	0	15317	0
Total Cost												**550454**	3970

A Sample of UC result in a form of Dendrogram is shown in Fig.5. The UC results of ACDC methodology is compared with the popular superior existing methods. The comparative results are presented in the Tables-8 and 9.

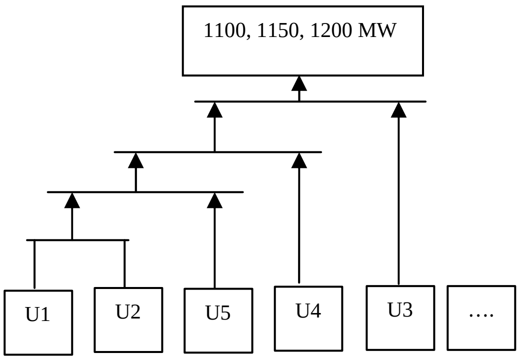

Fig.5 Dendrogram. Result of AC Algorithm applied to UC. Commitment of certain units for the loads mentioned.

5.1 Discussion

Through the simulation results obtained, the proposed methodology can be stated to has the following features:

1. The method is simple and demonstrates the use of cluster techniques applied to power systems.
2. Since search for optimal UC solution is carried with in small number of units, the CPU time is quite less.
3. Logic behind Priority List formation is straight fold and can be more technically justified.
4. Compared in other methods, total cost of generation in the proposed method is obtained least as the method does not generate any kind of infeasible solutions as in the case of genetic approach.
5. The proposed method has coding advantages and do not involve with complex computational work.
6. The method can be easily extendable for large size power system UC problem

Table 8
Performance comparison of ACDC Methodology

Method used to solve UC problem	Total Cost of Operation($)
GA[13]	603423
EPL[12]	563977
Proposed ACDC Method	**550454**

6. Conclusion

A novel method based on clustering technique is proposed to mitigate Unit Commitment problem. The proposed method is more realistic and less heuristic. Following load pattern, two individual algorithms based on Agglomerative and Divisive cluster algorithms are proposed for increasing and decreasing load patterns. Euclidean Costs of units segregate the units in to clusters. Two separate priorities lists one for increasing and another for decreasing load conditions are prepared based on Euclidean costs. A 10-thermal unit system is considered for simulation study. The strategy employed proved to be quite effective and satisfactory as evident through simulation results and comparison with the existing latest superior techniques.

Table 9
Performance comparison of ACDC Methodology (Execution time)

Method used to solve UC problem	Computational Time (Seconds)
GA[13]	73.86
EPL[12]	0.72
Proposed ACDC Method	**0.625**

References

[1] Allen J. Wood and Bruce F. Woolenberg, *Power Generation Operation and Control* (Wiley India Edition, 2006.)
[2] Happ. H. H., R. C .Johnson, W. J .Wright, Large Scale hydro-thermal unit commitment-method and results, *IEEE Transactions on Power Apparatus and Systems, Vol. PAS-90*, 1971, 1373-1383.
[3] W. L. Snyder, H. D. Powell, Jr., and J. C. Rayburn, Dynamic Programming Approach to Unit Commitment, *IEEE Transactions on Power Systems*, *Vol. PWRS-2*, No. 2, May 1987, 339-350.
[4] F. Zhuang, F. D. Galiana, Towards a more rigorous and practical unit commitment by Lagrangian relaxation, *IEEE Tr. On PS*, *Vol. PWRS-3*, May 1988, 763-770.
[5] A. J. Cohen, M. Yoshimpura, A branch-and-bound algorithm for Unit commitment, *IEEE Tr.on PAS, Vol.PAS-102*, Feb.1983,441-451,.
[6] J. A. Muckstadt and R. C. Wilson, An application of mixed-integer programming duality to scheduling thermal generating units, *IEEE Transactions on Power Apparatus and Systems,* 1968, 1968 – 1978.
[7] F. Zhuang and F. D. Galiana, Unit Commitment by simulated annealing, *IEEE Tr.on PS*, *Vol.5,* No.1, 1990, 311-317.
[8] G. B. Sheble and T. T. Maifeld, Unit commitment by genetic algorithm and expert systems, *ESPR, Vol.30*, No.2, 1994, 115-121.
[9] H. Sasaki, M. Watanabe, R. Yokoyama, A Solution method for unit commitment by artificial neural networks, *IEEE Tr.on PS*, *Vol.7*, No.3, Aug 1992, 974-981.
[10] S. Sanscifard, N. R. Prasad and H. A. Smolleck, A Fuzzy Logic Approach to Unit Commitment, *IEEE Tr.on PS*, *Vol.12*, No.2, May 1997.
[11] A. H. Mantawy, Youssef L. A. Majid, S. Z. Selim, A New genetic based tabu search algorithm for unit commitment problem, *EPRS,*49(1999), 71-78.
[12] T. Senjyu, K. Shimabukuro, A Fast technique for unit commitment problem by extended priority list, *IEEE Tr.on PS, Vol.18,* No.2, May 2003.
[13] K. S. Swarup and S. Yamashiro, Unit Commitment solution methodology using genetic algorithm, *IEE Tr.on PS*, *Vol.17*, No.11, Feb,2002, pp87-9.
[14] Thomas Bailey and John Cowles, Cluster definition by the optimization of simple measures, *IEEE Trans. on Pattern Analysis and Machine Intelligence,* Sep 1983.
[15] Pang Ning Tan, *Introduction to data mining* (Chapters 8,9, Michigan state university, Adison Wiley, 2006)