GENERATION CAPACITY BIDDING UNDER FUZZY ENVIRONMENT

A DISSERTATION

Submitted in partial fulfilment of the
requirements for the award of the degree
of
MASTER OF ENGINEERING
in
ELECTRICAL ENGINEERING
(With Specialization in Power System Engineering)

By

SARAB DEO RAM

DEPARTMENT OF ELECTRICAL ENGINEERING
UNIVERSITY OF ROORKEE
ROORKEE–247 667 (INDIA)

MARCH, 2000

CANDIDATE'S DECLARATION

I hereby declare that the work presented in the thesis entitled, "**GENERATION CAPACITY BIDDING UNDER FUZZY ENVIRONMENT**", submitted in partial fulfillment of the requirement for the award of the degree of master of Engineering, in Electrical engineering, with specialization in Power System Engineering, in the Department of Electrical Engineering, University of Roorkee, Roorkee. It is an authentic reward of my own work, carried out with effect from July 1999 to March 2000, under the guidance of Dr. N. P. Padhy, Lecturer, Department of Electrical Engineering, University of Roorkee, Roorkee.

Author has not submitted the Matter embodied in this thesis for the award of any other degree.

(Sarab Deo Ram)

Dated: 15|03|2000

CERTIFICATE

This is to certify that the above statement made by the candidate is correct to the best of my knowledge and belief.

Date: 15/03/2000

(Dr. N.P.Padhy)

Department of Electrical Engineering,

University of Roorkee

AKNOWLEDGEMENT

I am greatly indebted to my guide Dr. N. P. Padhy, Lecturer, Department of Electrical Engineering, University of Roorkee, Roorkee, for his kind support and guidance during my work. His co-operation and in-depth knowledge has made my work possible.

I am also thankful to Sri Bharat Gupta, Who in the capacity of being our Lab O.C has given me the complete freedom for doing work in the lab at any time.

I am thankful to Dr. B. Das, Lecturer, Department of Electrical Engineering, University of Roorkee, Roorkee, who helped to work in the computer lab in odd hours.

Last but not least, I am also grateful to Power System Engg. Faculty members for their time to time guidance and my classmates, whose presence have created a workable atmosphere in the lab.

(Sarab Deo Ram)

Dated: 15/03/2000

ABSTRACT

Due to regulatory change in power demand at each hour, it becomes an important and highly complex to aid capacity of private or foreign producers. This thesis report has proposed a software tool to support the planning body under utility in selection of bids. This thesis discusses the use of mathematical model to aid the decision-makers to select generation capacity bids. A three stage mathematical scheme has been developed and in the first stage, Analytic Hierarchy Process (AHP) that converts linguistic qualitative informations into quantitative informations and scores the bids under fuzzy environment. Such type of informations to the planning body depends upon the desirability of the utility. Second stage, unit commitment using dynamic programming has been proposed to find the status of the generating units that are eligible for bidding. Last stage, Hurwitz criteria for decision making combines both the solutions obtained from first and second stages to evaluate accurately the composite priority ordering among the specific projects. This composite priority ordering of the bids makes the planning body efficient, reliable, easy and economical in selection of the bids in complex electric marketplace. In the proposed model, multiple objective frameworks to capture both quantitative as well as qualitative informations (through expert's judgements) have been employed in the selection of bids for better results.

CONTENTS

CHAPTER-1

INTRODUCTION

1.1 General

Electrical energy is one of the most versatile forms of energy and it controls the economic growth of the nation. With time, demand of electrical energy has been increasing with industrial growth inspaite of adequate generation of electrical energy. There is a wide gap between supply and demand of energy due to high demand against less generation. Hence, generation, supply, demand and management of energy in adequate form are very important to assure the economic development strategy of any country.

For example, in India, in the first five year plan in 1951 the total generation of electrical power was 1712 MW and at the end of 1996 the total generation of power was 83,288 MW. The increase in power generation becomes 49 times in 45 years is an imprecisely improvement by all standards.

Despite of this imprecisely improvement in generation of electrical energy, the utility failed to meet the increase in demand. Then the utilities and Govt. parallel decided to offer and encourage private and foreign investors to invest, build and operate the generation power projects independently. Therefore, private and multinational companies becoming very important ensues for developing countries to invest, build and operate the generation power projects and meet high demands. The utility purchase powers from the private producers are compelled to follow Govt. rules and they have to undergo competitive marketplace.

In this point of view the United Kingdom (UK), Norway and Chile have recently re-regulated to encourage private investors and to allow more competitive marketplace. The Federal Energy Regulatory Commission (FERC), in its recent notice, has announced to expend competition in the US electric marketplace [3]. California has recently announced plans to adopt a structure of electric market similar to that used in the UK.

A competitive electric marketplace envisages competition in both investment and supply. Competition in supply, on the other hand, is concerned with how owners of existing plant prices their commodities so as to be commercially successful in the contemporary supply -demand marketplace [4]. The present situation in the UK privatized market, where sellers are requested to declare their production costs (or marginal production cost). But this is not the final decision, because the costs of production can never be made uniform for all sellers, can never be wholly scrutinized for reasons of commercial confidentially and therefore cannot be the basis for setting up a "level playing field". The final decision of market price is fixed by NGC (non utility generation company) and then it is adopted by all the power producers.

Energy treading in a competitive electricity market can be modeled as a two-level optimization. At the top, level a Centralized Economic Dispatch (CED) uses a priority list method to solve the *fundamental problem of reliable market clearing with price* discovery. The lower level consists of a set of decentralized bidding (DB) subproblems. The DB model uses a self -unit scheduling simulator based on parametric dynamic programming to produce hourly bid curves for the dispatch co-ordinate. Unit operating constraint and costs such as the minimum -up and minimum down times unit start -up and shut-down costs affect the over all costs of the units.

Electric resources planning, whether in the traditional sense or through a bidding process, should result in the minimum cost of providing electric power. In competitive environment, it is said that firms achieving enhanced efficiency and lower costs will reap greater returns and flourish against competitor [4]. Hence in a competitive scenario, price formation will remain an internal decision of the sellers, and the mechanism used by each seller to arrive at the final offer price will depend on a variety of internal commercial, strategic, accounting and cost related objectives. All objectives of new resources contributing to its cost must be taken into account in a least cost planning process. The traditional planning approach considers several attribute mainly quantitative attributes which effecting the generation and transmission and distribution systems. But in actual practice quantitative and qualitative attributes both effect its cost, efficiency, transparency, regularity etc.

1.2 An Introduction to Bidding

The operation of the system by a system operator (SO) based on offers and bids to supply electric power is a common element of the electric market. The system operator receives bids for demands as well as supply [6]. The bids indicate the price at which a purchaser is willing to buy a certain amount of energy. But the offers indicate the prices and quantities at which a seller will supply. Suppliers will submit a price, called offers, to supply the commodities (thermal power, hydel power, nuclear power and etc.) in each hour. The bids for energy will be in a finite number of time blocks normally twenty-four hours.

In competitive electric market, selection of bid is more tedious task. Selection of bids depends on various quantitative and qualitative factors. In the proposed bidding evaluation, Analytical Hierarchy Process (AHP) and Hurwitz criteria for decision making have been used due to their capability in handling both qualitative and quantitative information.

1.3 Literature Review

The basic function of an electric power system is to provide adequate supply of electric energy to its customers as economically as possible within a reasonable level of quality and continuity. In this regard, the planning bodies in the utility encourage the private bodies and permit them to generate power based on priority ordering and hence for bidding. The method of priority ordering should be on line, efficient, economical, reliable, flexible, transparent competitive bidding.

At present various technical and institutional limitations do not allow an accurate evaluation of the benefits of a new resource to be performed in a bidding system. Lack of sufficient data and /or analytical tools constitute the major technical limitation [1]. Institutional limitations stem mainly from the need to prevent gaming in the bidding process. In order to prevent gaming and to ensure maximum economic benefit, the process should be transparent. A transparent process would provide maximum information and minimum specifications on a pre-bid basis. The information would enable the bidders to self- score the merits of their projects; hence, they will be able to optimize their bids to enhance customer benefits and the profitability of their projects. A bidder will score points by providing attributes and deemed desirable by the utility. Bidders with the highest points scored will win the bid. The main drawback of these bid scoring system is that the award procedure for points is based on direct analysis of the costs and benefits attributable to the resource and this bid evaluation procedure uses information associated with the transmission impact of a new resource due to its location which is shown in Fig. 1. The cost considered in the bidding process includes: cost due to losses, cost of connecting thermal loading and problems, cost due to protection, and stability requirements. In the proposed model costs of generation, on & off time, start cost, shut down costs are also included. Therefore this procedure of evaluating priority order and bidding is expected to be accurate.

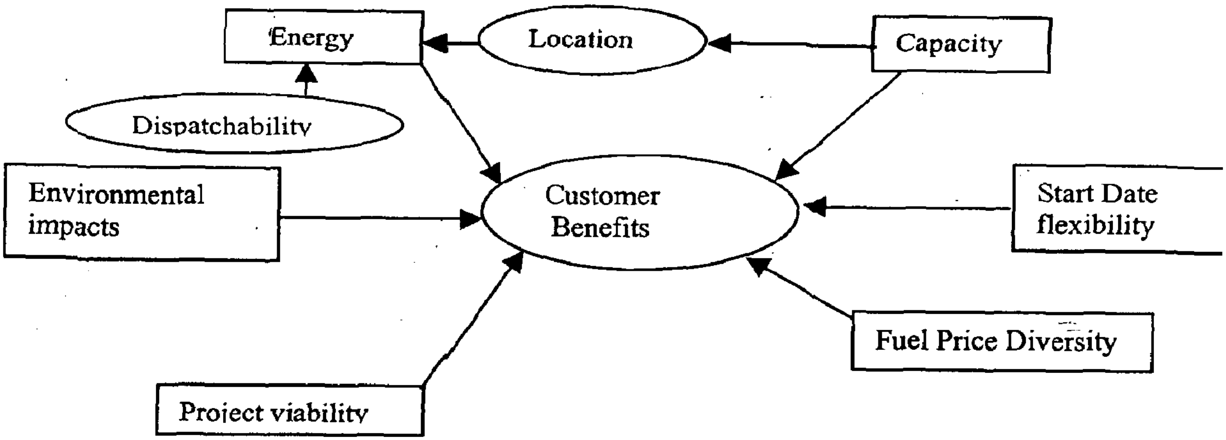

Fig. 1. Framework for bid evaluation

Optimal economical scheduling of resources is a major task in electric utility system operation [2]. Considering the load prediction and the specific properties of a utility, the optimal operation schedule has to be determined. Optimal operation of the generation units concerning the primary energy allocation is also equally important. A number of constraints have to be considered during the optimization algorithm to compare different plants and to select the best one for the actual scenario. Optimization problem[2] has been formulated by using Mixed Integer Dynamic Programming method. The complete optimization problem is divided into tree steps: such as Long term optimization, Short term optimization and Instantaneous optimization.

A.K.David [4] has considered different types of commodities (factors) which affects the efficiency of bidding, reliability, cost of the power supply, scoring of the bids in the competitive bidding in electricity supply. A competitive marketplace envisages *competition in both investment and supply*. The *former is concerned with the prospect* of several entrepreneurs competing to obtain regulatory franchise to build, operate and make commercial gains from investment in future plants. Competition in supply, on the other hand, it is concerned with how owners of existing plants price their commodities so as to be commercially successfully in the contemporary supply—demand market. It has also focused on the issue of competition in supply, and not on the investment analysis problems. In this point of view it has discussed in detail about the uncertainties regarding price bids made by competitors, and uncertainties about the expected grid power demand at any time. The main demerits of the above model are due to exclusion of qualitative attributes for selection of the generating power plants in the competitive market places. As we know the qualitative criteria also affect the cost, efficiency, reliability, and other factors, therefore these criteria also should be considered.

Edward P. Kahn[5] described the competitive bidding for private power contractors, which is an important element in utility planning. In this bidding, optimization method has been used for incorporating non-price factors in the evaluation process. This paper has emphasized on the 1998 Virginia Power (VP) and simulated the selection process. In this study the bid evaluation problem associated with dispatchability by conducting a simulation of the Virginia projects process. This paper does not represent the procedures actually used by the utility but it does relay on public data associated with their competition and the conditions under which VP operates. According to this paper the dispatchebility of bulk power makes it very difficult to use simple linear self-scoring system. Finally he has concluded that there is no way to assign a rank ordering of value that accounts for the attractive effects. He has evaluated number of non-price factors to evaluate the bids. He has used linear self—scoring evaluation system to attach non-price factors with price score to evaluate overall ranking of the bids for practical systems. But unfortunately the model failed to incorporate qualitative criteria in the evaluation of bids that affects the bid evaluation.

Narayan S.Rau [6] has described various types of competitive markets for contract and spot market studies, which may further help for bidding in dynamic market.

CHAPTER-2

BIDDING AND ITS EVALUATION

2.1 Introduction

Competitive bidding for private power contracts is becoming an important element in utility planning.. Private power production is becoming an increasingly important source of electricity generation. Utilities purchasing power from private producers have been turning to competitive bidding as the best mechanism for managing this process. To implement competitive bidding successfully, evaluation methods must be developed to account for the various factors affecting the desirability of particular projects. This paper creates an environment in which utility purchasing energy from large number of power producers. In this regard private bidders offer their projects, to be evaluated by the utility. In the traditional process of bids evaluation, price factors and non-price factors both have been considered, based on linear self- scoring evaluation systems. In the linear self scoring bid evaluation systems, all these bidding process are base on point systems. A bidder will score points by providing attributes and features deemed desirable by the utility. Bidder with the highest points scored will win the bid [7].

Main objective of the bid evaluation is to make transparent competitive bidding, ensuring supply reliable, faster efficiency, and economical to select the bids. Evaluations of bids both for investment analysis problems and for operational problems are relatively new area. The planning body in the utility is confronted with large volume of information, which need to be analyzed to select the winning bids. The planning body is often left with the traditional planning tools which do not quite meet the requirements because they do not aid the planner perform the following in an integrated environment.

- rank the bids based on their marginal contribution
- analyze the impact of special contracts that may exist between the bidder and the utility relating to fuel purchase, power purchase, and wheeling .
- Analyze effect of uncertainties associated with various factors like fuel cost, project completion time, load etc.
- Analyze the trade-off among multiple conflicting objectives including qualitative aspects of generators and transmission like environmental impact, project viability, reputation of the company etc.
- Simultaneous consideration of demand side options by the utility.

The bid evaluation process emulates the actual system operation closely to produce realistic estimate of the priority ordering of the bids. In bidding process of self - scoring system where the bidders get to know the attributes (both qualitative and quantitative) deemed desirable by the utility on a pre-bid basis. The main drawback of the self-scoring system is that the award procedure for points is based on experience and engineering judgement of the utility planners rather than a direct analysis of the cost and benefits attributes to the resources. Actually self-scoring system is a project -by project analysis using traditional tools.

Hence it has been proposed to develop a multiobjective mathematical model which is expected to perform better than existing approach's due to the following reasons:

- the individual projects cannot be evaluated in isolation ignoring the mutual interaction with other bidders and utility owned projects.
- Advance numerical techniques from social science makes it possible to formalize the process of expressing desirability or undesirability of projects by experts quantitatively .

- fuzzy set theory for linguistic data become more easy with respect to classical method which easily combined with other variable for desirable purposes.
- it efficiently combine with unit commitment and can be combined with optimal power flow and gives final priority ordering depending upon the all attributes (qualitative and quantitative).

This thesis attempted to develop a multi-criteria mathematical programming framework to significantly enhance the capability and accuracy of the earlier models for bidding. More specifically the model consisting of the following:

- Implementation of **Analytic Hierarchy Process** (AHP) to assign scores to the bids.
- **Dynamic programming** for unit commitment, which indirectly affects the bidding process.
- **Hurwitz criteria** for priority ordering of the bids.

CHAPTER - 3

FUZZY DECISION MAKING ALGORITHMS

3.1 Introduction to Fuzzy Logic

Fuzzy logic systems address the imprecision of the input and output variable, directly by defining them by fuzzy numbers or fuzzy sets, that can be expressed in linguistic terms such as low, high etc. Further they allow for greater flexibility in formulating system descriptions at the approximate level. In fuzzy logic, complex process behavior can be described in general terms without precisely defining the complex (usually nonlinear) phenomena involved.

Definitions:

(A) Fuzzy Set:

The set of objects that may function as a criterion is a fuzzy set. After identifying the variables associated with decision making, the fuzzy sets defining these variables are selected and normalized between 0 and 1. The vales for particular variables between 0 and 1 are called fuzzy set. This can be written as

Where, μ_A is function of, every element between 0 and 1;

$$\mu_A(x): X \to \{0,1\}$$

X_A= called membership function.

(B) Membership function:

Membership function of any variable depends on its characteristic function, which is expended by generating the valuation set from the pair of numbers $\{0,1\}$ to all numbers found in $[0,1]$.

(C) Maximum value (MAX):

Maximum operation is defined through max (V) is analogous to product (.) in algebra. EXP: V(0.01,0.04,0.02,0.09) =0.09;is written as

$$\mu = VA;$$

$$= V(\mu_1, \mu_2, \mu_3, \ldots, \mu_m);$$

$$= V^m{}_{k=1}(\mu_k); \quad \cdots\cdots\cdots \quad 3.1$$

Where,μ is an element of A-that is $\mu \in A$.

(D) Minimum value (min):

Minimum operation is similar to the Max. Operation is defined through min ($\wedge$) is analogous to sum (+) in algebra.

EXP: $\wedge(0.01,0.04,0.02,0.09) = 0.01$;

Is written as: $\mu = \wedge A$;

$$= \wedge(\mu_1, \mu_2, \mu_3, \ldots, \mu_m);$$

$$= \wedge^m{}_{k=1}(\mu_k); \quad \cdots\cdots\cdots \quad 3.2$$

Where: μ is an element of A- that is $\mu \in A$.

They both have the same properties of associatively and distributivity and thus in equation that involve min and max we may employ them in the same manner multiplication (.) and addition (+).

(E) Normalization:

Let A be a fuzzy set is denoted by $\mu_A(x)$;

Where, x is the element of this set.

$$Then\,normalization\,of\ fuzzy\,\mu_A(x)$$

$$\mu_{NORA} = \frac{x_i}{VA} \; ; \quad \cdots\cdots\cdots \qquad 3.3$$

Where, VA means one of the values, which is maximum among the membership elements.

(F) Complement of a fuzzy set:

The complement of a fuzzy set A is a new fuzzy set, A, with membership function

$\mu_A(x) = 1 - \mu_A(x)$ ………….3.4

Fuzzy set complementation is equivalent to negation (NOT) in fuzzy logic.

3.2 The Analytic Hierarchy process (AHP)

AHP is a technique developed by Thomas L. Saaty [7] to compute weightage of the particular criteria for particular bid, and then the scoring vector relative to the importance of the other criterion. The need for AHP to evaluate bids in investment planning model is due to the significance of the qualitative aspects which might be as important as the quantitative features like cost, SO2 emission coefficient, total operating times, capacity charges and unit status etc. In reality, one does come across situations when a project proposal is turned down because the company has a bad track record, or a transmission project is shelved because there is strong opposition from environment groups, it also provides a unique way of formalizing the judgement of experts of different background and assimilate such information for final decision making.

The input to the model is qualitative by expert judgements on pair-wise comparisons of relative importance of the criterions. In the proposed method AHP has been used for assigning **scoring vector** to different criterions. AHP technique is popular due to its simplicity, flexibility, intuitive appeal and its ability to convert from qualitative information to quantitative information of the criteria in the same decision framework. The AHP used in this study gives transparent solution to BIDs evaluation. The utility may specify a scoring system for the qualitative parameters for environmental impact, company goodwill, project starting date flexibility etc. The step by step algorithm of AHP is described briefly as follows.

Step 1. Identify the important qualitative criteria to be used in the evaluation of bids.

Step 2. Prepare a questionnaire according to the criteria and request the decision-makers to compare the criteria pair-wise.

Step 3. Transform the verbal values into the numerical intensity values using Saaty's intensity Table[7].

Step 4. Prepare a pair-wise comparison matrix A=[Aij] with a row and column for each criterion.

Step 5. Calculate the Eigen-vector or weight vector of matrix A for different criteria corresponding to the largest eigen-value (Lmax). Normalize this Eigen-vector to get the weight vector for the criteria '**i**'.

Step 6. Calculate the consistency ratio (CR) of matrix A using the following equation.

CR=[((Lmax-n)/(n-1)]/RI, where RI is the random index.

If CR <0.1;then the judgements are consistent and can be accepted. Otherwise repeat from step 2 to step 6;

Step 7. Carry out steps 2-6 for the judgements provided by all the decision makers, and take an arithmetic average of weight vectors to get the weight of the criteria.

Step 8. For each criteria define three grades of bids (Good-G, Normal-N, Bad-B). By using step 1 to step 7 compute the weight of these grades V_g.

Step 9. Final weight is given by;

$$W_{ij} = C_i * V_g.$$

Where, C_i = weight vector for criteria

Step 10. Classify the competing bids into any of the three grades for all the criteria. The final AHP score of a bid is then computed as the sum of the corresponding Wig values.

3.3 Hurwitz Criteria

When there exists a trade-off among several planning objectives, it is often impossible to arrive in a certain solution that contains all the optimal values simultaneously and the best option is a selection as a compromise among the objectives. Hurwitz criteria for decision-making [9] has been used to obtain the priority ordering among all the projects. It assigns the normalized quantitative values for different criteria such as AHP score, generation cost, unit status and emission co-efficient. The step by step algorithm for Hurwitz criteria is as follows:

Step 1. Identify all the comparative quantitative informations.

Step 2. Normalize all the quantitative informations within 0and 1.

Step 3. Find the maximum and minimum value of all criteria for each option.

Step 4. Find the weight vector of the ith option using the following equation:

$$H_{vi} = \alpha H_i + (1-\alpha) L_i ;$$

Where,

α=Constant (co-efficient) varies between 0.1 to 0.9;

H_{vi} =weightage of the i^{th} bid;

H_i = higher value of all criteria corresponding i^{th} bids;

L_i = lower value of all criteria corresponding i^{th} bids;

We have taken 0.5 in this thesis because at this value result more converse;

Step 5. Based on the weight vectors from step 4, priority ordering of different options can be determined.

CHAPTER-4

PROPOSED MODEL

4.1 Introduction

Main objective of this thesis is to develop a software so that, it makes the electric marketplace transparent, flexible, reliable, and economical depending up on the various type of the variables. In this regard, the proposed modeling framework comprises three modules, third module i.e. Hurwitz criteria is basically utilizes the results obtained from the rest two independent models such as Analytic Hierarchy Process (AHP) and Unit Commitment module to compute the priority ordering among the bids.

4.2 Planning Module

Planning module (PM) is developed as a multi- period dynamic model, which combines Analytic Hierarchy Process (AHP), and Hurwitz criteria for decision-making. Load representation is made in the form of multiple-block load duration curve for individual demand nodes. The basic decision is to select power projects in priority ordering to minimize a composite objective function comprising both qualitative and quantitative objectives. The decision variables include:

$X_{k,t}$ $\quad selection of\ generation\ project\ k\ in\ period\ t$

$P_{i,q,t}$ $\quad MW\ generation by unit\ i\ in\ time block\ q$
$in\ period\ t$

S_t $= load\ power\ in\ period\ t$

Where, X is investment variable of binary type.

P and S are operation variables of continuous type.

The objective function comprises the following attributes:

- Flexibility of project starting date (FLEX)
- Fuel diversity (FUEL)
- Company goodwill (REPU)
- Generation related environmental impact which are not directly measurable and qualitative in nature (ENGEN)
- Transmission related environmental impact which are not directly measurable and qualitative in nature (ENTR)

Here all the attributes are qualitative in nature. The procedure of quantifying these attributes for each bid using Analytic Hierarchy Process is described in section 3.2.

The main constraints include:

- Meeting demand at each bus,
- Budget constraints,
- Minimum utilization of generation capacity.

4.3 Unit Commitment Module

Generation allocation is dealing with the problem of supplying the loads as they occur on the system at minimum production cost, while satisfying a multiplicity of constraints and set points. This problem is too complex to be solved in one piece even by computer control technology. It has to be separated into subfield, after solving subfields, they are interfaced.

Unit commitment is the planning under uncertainty. This problem is to determine an optimal schedule of what generation units must be started or shut-off to meet the anticipated demand in an economical and reliable manner. The scheduling procedure should be fast and flexible enough to handle the unexpected changes in the loads and in the unavailability of system components. It is used to evaluate the schedule of short notices overhaul work.

4.3.1 Unit Commitment Using Dynamic Programming

The main objective of the unit commitment problem in power system is to find the commitment schedule and operating level of all generating units at each time interval in each day in order to minimize the production cost. The unit commitment is to commit enough units and leave them on line and turning off when they are not needed .It is necessary to secure reliability by which is meant continuos electric supply within its rang of power generation.

4.3.2 Factors Affecting Dynamic Programming

Unit commitment is a pre-dispatch mathematical programming problem. It is a type of scheduling operation that fits between the economic dispatch and maintenance and production scheduling in the management of power generation resources. There are several factors affecting this [10]. They are as follow:

Unit constraints

- **Individual unit constraints:** It represents the incremental or total operating cost of generating unit as a function of megawatt power level. In modeling these curves, the incorporating of minimum and maximum limits is very important work as shown in Fig. 4.1.

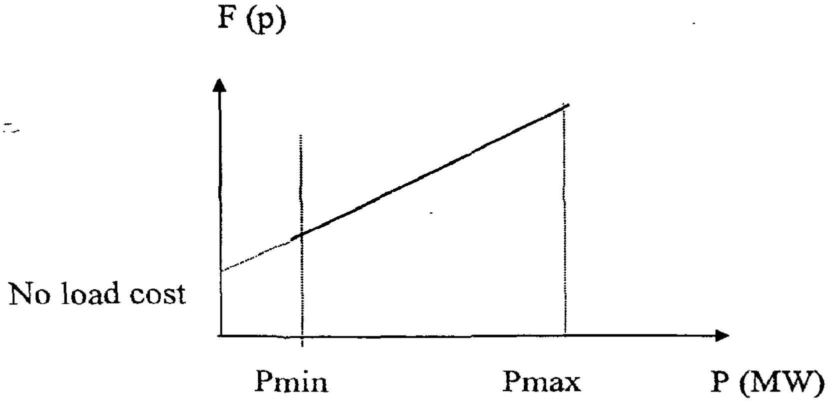

Fig. 4.1 Linear Cost Characteristics of the Generating Units

- **Thermal unit constraints**

A thermal unit can undergo only gradual temperature changes and this some hours to bring the unit on line. Due to these restrictions various constraints arises such as.

- **Maximum and minimum output limits on generators**: For stable operation, each operating unit should operate within it's limits, thus imposing the inequality constraint on the individual upper and lower power output limits of each generator Gi i.e.

$$P_{Gi}^{min} < P_{Gi} < P_{Gi}^{max}$$

- **Minimum runs time**: Once the unit is running it should not turn off immediately. It takes some time to remove the unit from line. It is a minimum time before it can be deconected.

  $0 <= T_{iu} <=$number of hours unit Gi has been on-line.

- **Minimum down time**: Once the unit is decomitted, it should not be turned on immediate. It takes some time to bring the unit on line. It is a minimum time before it can be recommitted.

  $0 <= T_{id} <=$number of hours unit Gi has been off-line.

(v) Spinning reserve and peak capacity

Spinning reserves, the term used to describe the total amount of generation available from all the units synchronized on the system, minus the present load and losses being supplied. In simple, if one unit is lost, or during scheduling period when unit changes their peak capacity, there must be ample reserve on the other units to make for the losses in a specified period by adding it. These are necessary in the operation of a power system if load interruption is to be minimum.

Not only the reserve must be sufficient to make up load for a generation unit outage but also the reserve must be among the fast responding units and slow responding units. This allows AGC (Automatic Generation Control) system to restore frequency and interchange power quickly in the event of a generating unit outage. Reserve must be spread around the power system to avoid transmission system limitations.

4.3.3 Methods for Unit Commitment

The unit commitment based on least cost is conflicting with the dispatch based on maximum continuous operating time. The priority order of the various generating plants will be based on the operating cost and maximum operating time. Unit commitment is a nonlinear optimization problem and hence can be solved by various methods as Lagrangian relaxation method, Priority -lists scheme programming, & dynamic programming.

4.3.3.1 Priority -List Scheme Programming Method (PLSP)

This is one of the earliest and simple approaches to determine unit commitment. It is still widely used in the industry. PLSP commits units according to a priority order based on unit average full load cost (AFLC). Commitment proceeds one hour at a time.

Commitment for the previous hour is used when the load remains unchanged. More units are added when the load increases and units may be shut down when the load decreases. When increasing the on-line capacity of the system, an available unit with a lower AFLC will start as long as this unit can be shut down when the system load falls below the load for present hour. Similarly when reducing the on -line system capacity, an on-line unit with higher AFLC is shut down first, provided it can be restarted when the system load picks up at some future hour. In this way, the operation of more economical units is maximized while the operation of less efficient unit is minimized.

The problem associated whit the PLSP methods is that if there are n units, PLSP encounters only n states and leave behind (2^n -n) states and hence is not an efficient method.

4.3.3.2 Lagrangian Relaxation Method Programming (LRP)

Lagrangian relaxation method involves decomposition of the problem into a sequence of master problem and easy sub-problems, whose solution converges to an optimal solution to the original problem.

The solution method thus consists of determining a set of Lagrangian multipliers that generate a solution that meets all capacity and reserve constraints and is very close to optimal solution. Given this commitment, the demand constraint can be satisfied via an economic dispatch calculation that takes into accounts reserve contributions of the individual units. The main problem associated with LR method is to get a good starting value for lagrangian multipliers to speeds up the iteration process.

4.3.3.3 Dynamic Programming Method

It is a multi-period scheduling method. It is based on the principle of optimality that states a optimal policy if any, at any stage. Dynamic programming has many advantages over the enumeration scheme, the chief advantage being a reduction in the dimensionality of the problem.

In the dynamic programming approach , the following assumption have been made:

1. A state consists of array of units with specified units operating and the rest off-line.
2. The start-up cost of a unit is independent of the time it has been off-line (i.e., it is a fixed amount).
3. There are no costs for shutting down a unit.
4. There is a strict priority order, and in each interval a specified minimum amount of capacity must be operating.

One could set up a dynamic -programming algorithm to run backward in time starting from the final hour too be studied, back to the initial hour. Conversely, one could set up the algorithm to run forward in time from the initial hour to the final hour. The forward approach has distinct advantages in solving generation unit commitment. For example, if the start-up cost of a unit is a function of the time it has been off-line, then a forward dynamic -program approach is more suitable since the previous history of the unit can be computed at each stage. There are other practical reasons for going forward. The initial conditions are easily specified and the computations can go forward in time as long as required. Forward dynamic programming algorithm has been used and the flow-chart is shown in Fig. 4.2. The search path of the dynamic programming is shown in Fig. 4.3.

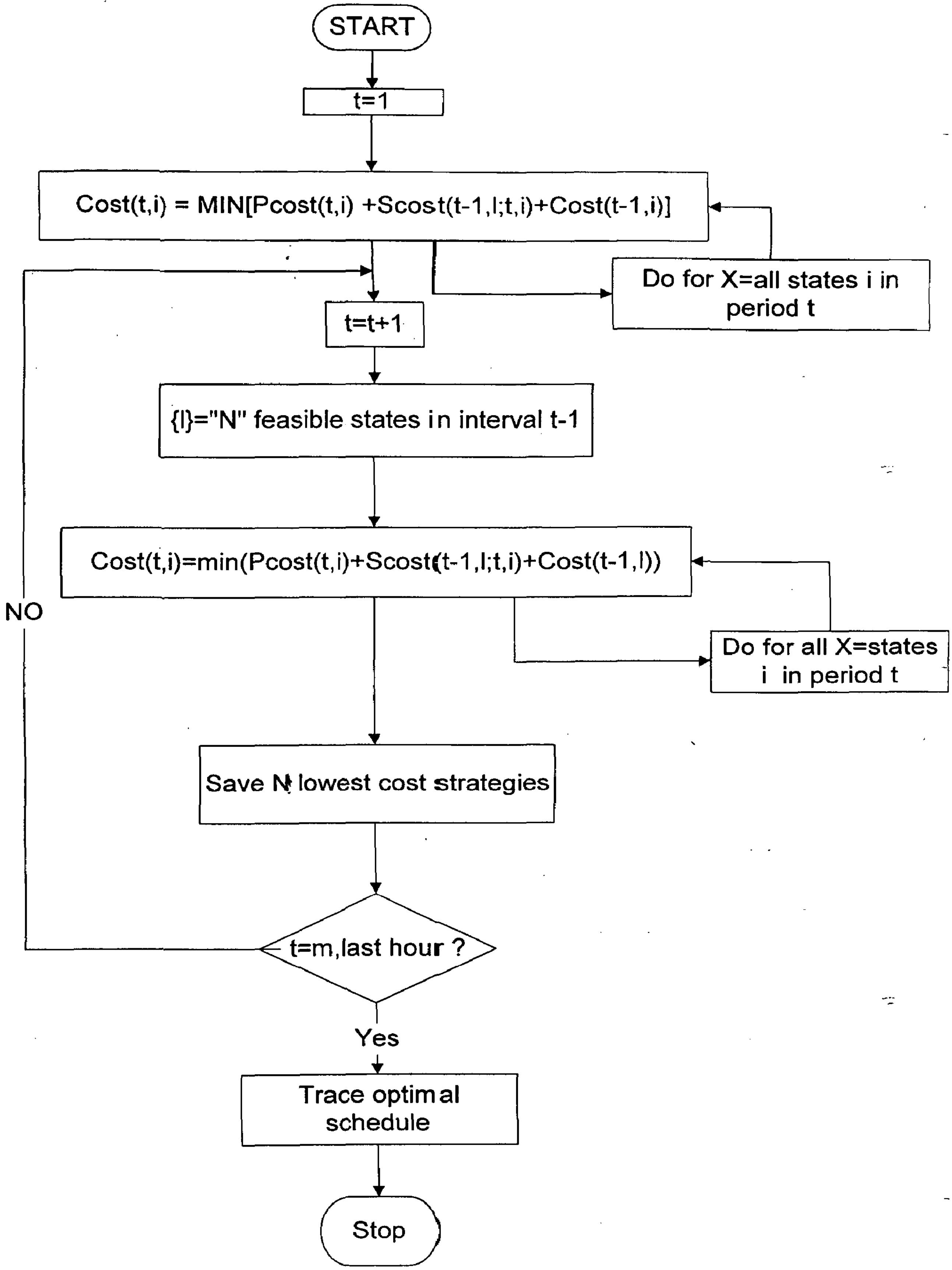

Fig. 4.2 Flow Chart of Dynamic Programming

The recursive algorithm to compute the minimum cost in hour 't', with combination 'i' is

$$Cost(t, i) = \min [P_{cost}(t, i) + S_{cost}(t-1, l; t, i) + Cost(t-1, i)]$$

Where, $Cost(t, i)$ = least total cost to arrive at state (t, i)

$P_{cost}(t, i)$ = production cost for state (t, i)

$S_{cost}(t-1, l; t, i)$ = transition cost from state (t-1, l) to state (t, i)

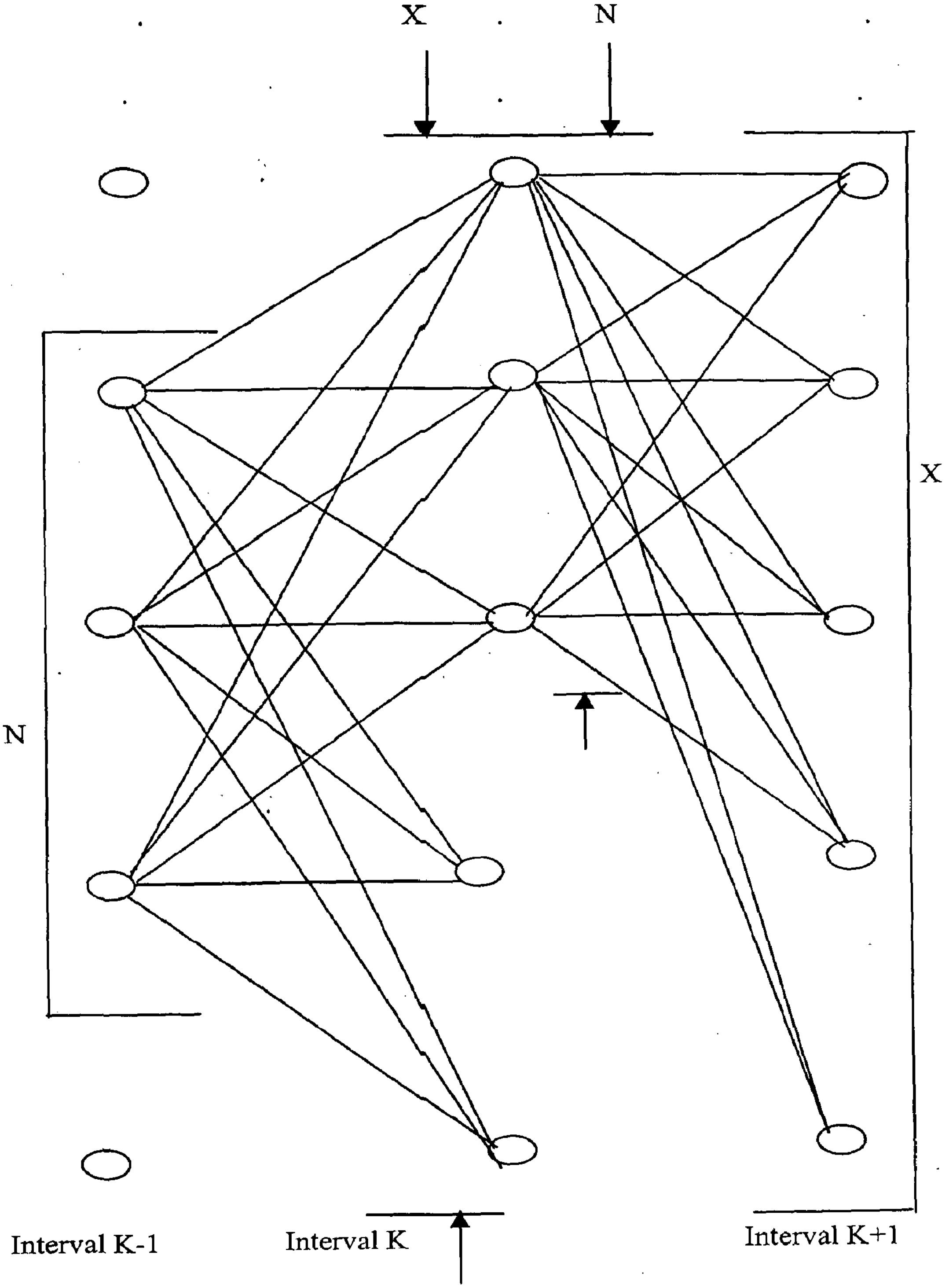

Fig 4.3 Search path of Dynamic Programming

Where,

X= number of states to search each period

N= number of strategies, or paths, to save at step

These variables allow control of the computation effort. For complete enumeration; the maximum number of the value of X or N is 2^N-1.

With a simple priority-list ordering, the upper bound on X is N, the number of units. Reducing the number N means that we are discarding the highest cost schedules at each time interval and saving only the lowest N paths or strategies. There is no assurance that the theoretical schedule will be found using a reduced number of strategies and search range only experimentation with a particular program will indicate the potential error associated with limiting values of X and N below their upper bounds.

Steps used for Unit Commitment using Dynamic Programming are as follows:

Step 1. Read all generator parameters

Step 2. Calculate total number of combinations that can meet the load demand at each hour.

Step 3. Evaluate maximum net generation capacity for every combination.

Step 4. Arrange maximum net capacities obtained in step. 3 in ascending order.

Step 5. If maximum net capacity is greater than load demand, calculate redistribution of load for each working unit based on their priority.

Step 6. Checks weather all the constraints have been satisfied or not.

Step 7. If yes find the unit commitment schedule with minimum cost.

4.4 Proposed Algorithm

Step1. Identify the important qualitative criteria to be used in the evaluation of bids. Following are the criteria used in this thesis are:

(a) Flexibility of project starting date (FLEX)

(b) Fuel diversity (FUEL)

(c) Reputation of the private bidders(REPU)

(d) Generation related environmental impact which are qualitative in nature(ENGEN)

(e) Transmission related environmental impact which are qualitative in nature(ENTR)

Step 2. Prepare a questionnaire and request the decision-makers (experts in the utility) to compare the criteria pair-wise with the help of the Saaty's intensity scale.

Questionnaire matrix: -

Table 1: Questionnaire matrix

	A	B	C	D	E	F	G	H	I
FLEX									
FUEL									
REPU									
ENGE-N									
ENTR									

A: Equal

B: In between equal and moderate importance

C: Moderate importance

D: In between Moderate importance and essential strong

E: Essential or strong

F: In between Strong and very strong

G: Very strong

H: In between very strong and extreme

I: Extreme

The Saaty's intensity table is given below: -

Table 2: Saaty's Intensity Table

Verbal scale	Numerical scale
Equal	1
Moderate importance	3
Essential or strong	5
Very strong	7
Extreme	9
Intermediate values	2, 4, 6, 8

Step 3. Transform the verbal values into the numerical intensity for different opinion obtained from the utility experts as shown in Table 3. For example comparison of FLEX vs. other criteria has been reported in the Table 3. The matrix for other criteria's can also be obtained in the similar pattern.

Step 4. Prepare a pair-wise comparison matrix $A=[A_{ij}]$, with a row and column for each criterion using Saaty's table and it has been presented in the Table 4. Similarly the results for other variables are given in Appendix A.

Step 5. Calculate the eigen vectors corresponding to the largest eigen-value (L_{max}) and normalize this eigen-vector to get the weight vector for the criteria '**i.**'

Table 3: Expert Opinion for FLEX Vs Other Criteria

	A	B	C	D	E	F	G	H	I
FLEX	*								
FUEL		*							
REPU			*						
ENGE-N					*				
ENTR								*	

Note: * is prvided by the expert.

Table 4: Pair-Wise Comparison Matrix

	FLEX	FUEL	REPU	ENGEN	ENTR
FLEX	1	2	3	5	8
FUEL	1 /2	1	2	3	4
REPU	1 /3	1 /2	1	5	2
ENGEN	1 /5	1 /3	1 /5	1	3
ENTR	1 /8	1 /4	1 /2	1 /3	1

Step 6. Calculate the consistency ratio (CR) of matrix an as

$CR=[((L_{max}-n)/(n-1)]/RI$,

Where, RI is the random index is assumed 0.9

n = order of the matrix or No. of bids to be score

If CR <0.1;then the judgements are consistent and can be accepted. Otherwise repeat from step 2 to step 6;

Step 7. Carry out steps 2-6 for the judgements provided by all the decision makers, and take an arithmetic average of weight vectors to get the weight of the criteria i.

Step 8. For each criteria define three grades of bids (Good-G, Normal-N, Bad-B). By using step from 1 to step 7 to compute the weight of these grades V_g.

Step 9. Final weight $W_{i,j} = C_i * V_g$.

Where, C_i = weight vector for criteria

Step 10. Classify the competing bids into any of the three grades for all the criteria. The utility authorities may do this. The final AHP score of a bid is then computed as the sum of the corresponding Wig values.

For bid No.1 the AHP will calculated using the following formula.

$$W_{bid} = W_{flex,good} + W_{fuel,bad} + W_{repu,bad} + W_{engen,normal} + W_{entr,good}$$

$$= 1.001640$$

Similarly the AHP score of the other bids can be calculated and the results have been presented in the Table. 5

Table 5: AHP Score of the Bids

Bids No.	FLEX	FUEL	REPU	ENGEN	ENTR	AHP
1	1	3	3	2	1	1.001640
2	3	1	2	3	3	0.714079
3	1	3	1	2	3	1.166358
4	1	1	2	3	3	1.320928
5	2	3	3	1	2	0.571200
6	2	2	3	3	1	0.601541
7	1	1	3	3	2	1.285869
8	1	3	3	3	2	0.924581
9	1	2	2	1	3	1.186355
10	1	3	3	3	2	0.924581

Step 11. Obtain the unit commitment schedule of units interested to participate in bidding process based on the data given in Appendix B.

Step 12. Based on the results obtained from step 11. And determine the importance of the bidders and finally assign the weightage to the bidders.

Step 13. Assign the generation cost associated to each and every bidder.

Step 14. Identify all the quantitative variables. Such as AHP scores, unit status, generation cost of the units and emission (SO_2) co-efficient. The quantitative values used in the proposed model is shown in Table 6.

Table 6: Quantitative Values

No. Bids	AHP Score	Total time(hour)	Generation cost (RS)	SO_2 co-efficient
1	1.001640	8.0000	14380.0	9.6
2	0.714079	8.0000	15433.0	9.5
3	1.166358	3.0000	7280.0	6.8
4	1.320926	8.0000	9815.0	8.2
5	0.571200	2.0000	5160.0	5.8
6	0.601546	6.0000	7830.0	9.4
7	1.285860	3.0000	3188.0	7.1
8	0.924581	6.0000	95816.25	7.8
9	1.186355	1.0000	14797.150391	7.2
10	0.924581	6.0000	16150.0	4.8

Step 15. Select Maximum value corresponding to AHP score, total operating time, generating cost, and SO2 co-efficient using equation 3.2.

Step 16. Normalize all the variables by using the equation (3.3) of the results of Step 15 and the normalized values of all variables are given in the Table 7.

Table 7: Normalized values of Step 15.

No. Bids	AHP Score	Total time(hour)	Generation cost (RS)	SO_2 coefficient
1	0.758286	1.0000	0.150079	1.0000
2	0.540590	1.0000	0.161069	0.989583
3	0.882985	0.3750	0.075979	0.708333
4	1.00000	1.0000	0.102436	0.854167
5	0.432424	0.2500	0.053853	0.604167
6	0.455397	0.7500	0.081719	0.979167
7	0.973453	0.3750	0.033272	0.739583
8	0.699949	0.7500	1.00000	0.81250
9	0.898124	0.1250	0.154433	0.7500
10	0.699949	0.7500	0.168552	0.5000

Step 17. Our main aim is to maximize total time & AHP score and minimize generating cost & SO2 emission co-efficient. For this purpose complements of SO_2 emission co-efficient and generating costs using equation (3.4) have been calculated and presented in the Table 8.

Step 18. Assign approximate values to all the elements of the all variables using the following equation and the modified results of the step 17 is as follows and shown in Table 9.

d[i][j]= (n/10-(i-1)/10);

Where, i= 1,2,3,10;

J= types of variables AHP, Total time, Generating cost, and SO2 emission co-efficient respectively.

Table 8:Normalize values with complement of generation cost and SO_2 co-efficient

No. Bids Table	AHP Score	Total time(hour)	Generation cost (RS)	SO_2 coefficient
1	0.758286	1.0000	0.849921	0.0000
2	0.540590	1.0000	0.838931	0.010417
3	0.882985	0.3750	0.924021	0.291667
4	1.00000	1.0000	0.897564	0.145833
5	0.432424	0.2500	0.946147	0.395833
6	0.455397	0.7500	0.918281	0.020833
7	0.973453	0.3750	0.966728	0.260417
8	0.699949	0.7500	0.00000	0.18750
9	0.898124	0.1250	0.845567	0.2500
10	0.699949	0.7500	0.831448	0.5000

Table 9:Approximte Assigned Values

No. Bids	AHP Score	Total time(hour)	Generation cost (RS)	SO_2 co-efficient
1	0.6000	1.0000	0.5000	0.1000
2	0.3000	0.9000	0.3000	0.2000
3	0.7000	0.4000	0.8000	0.8000
4	1.0000	0.8000	0.6000	0.4000
5	0.1000	0.2000	0.9000	0.9000
6	0.2000	0.7000	0.7000	0.3000
7	0.9000	0.3000	1.0000	0.7000
8	0.5000	0.6000	0.1000	0.5000
9	0.8000	0.1000	0.4000	0.6000
10	0.4000	0.5000	0.2000	1.0000

Step 19. By using equation (3.1) find the maximum value of all criteria for each bus,

Table10: Maximum Value of the variables for given bus

No. Bids	Maximum value of all criteria for each bid
1	1.0000
2	0.9000
3	0.8000
4	1.0000
5	0.9000
6	0.7000
7	1.0000
8	0.6000
9	0.8000
10	1.0000

Step 20. By using equation (3.2) find the minimum value of all criteria for each bus, which is given below:

Table 11 Minimum Value of all Variables for a given bus

No. Bids	Minimum value of all criteria for each bid
1	0.1000
2	0.2000
3	0.4000
4	0.4000
5	0.1000
6	0.2000
7	0.3000
8	0.1000
9	0.1000
10	0.2000

Step 21. By using step 5 and step 6 find the weightage of the ith bus using the equations:

$H_{vi} = \mu H_i + (1-\alpha)L_i$

α=Constant (co-efficient) varies between) 0.1 to 0.9;

H_i = higher value of the all criteria corresponding i^{th} bid ;

L_i = lower value of the all criteria corresponding i^{th} bid ;

H_{vi} =weightage of the i^{th} bid;

We have taken 0.5 in this thesis because at this value result more converse;

The calculated weightage of the bids is presented as follows.

Table 3: Weightage of the Bids

No.Bids	Weightage of the bids
1	0.5500
2	0.5500
3	0.6000
4	0.7000
5	0.5000
6	0.4500
7	0.6500
8	0.3500
9	0.4500
10	0.6000

Step 22. Arrange bid values in the decreasing order. And finally the bids with higher weightage have been selected. The bids with priority order has been presented as:

Bid	**Weightage**
4	0.700000
7	0.650000
3	0.600000
10	0.600000
1	0.550000
2	0.550000
5	0.500000
9	0.450000
8	0.450000

FINAL SELECTION OF BIDS IN DECENDING ORDER IS

4 7 3 10 1 2 5 9 6 8

CHAPTER-6

RESULTS AND CONCLUSION

The planning body in an utility is confronted with a large volume of information for generation capacity bidding, due to which existing classical techniques for bidding may face difficulties to handle the imprecisely defined datas. In this thesis work, three-stage mathematical Model under fuzzy environment has been developed. It makes easy to handle the imprecisely defined data and aid the planning body in the utility for evaluation of bidding score and priority order of bids and to finally select the bid based on their desirability. The input to the model obtained from the experts in evaluating the score of the bids of an electric resource plan is assumed to be consistent. This methodology impact on new resources, with transparent and economical bidding system in complex electric marketplace. This work marks significant departure from the earlier efforts made by other authors. This model accounts for both qualitative and quantitative features of bids. Analytic Hierarchy Process (AHP) that converts qualitative expects into quantitative value and unit commitment using Dynamic Programming (DP) evaluate bid status Hurwitz Criteria for decision-making all together on evaluate bid value efficiently . This technique gives better priority ordering among the bids, which are eligible for competition. This model has obtained solutions to realistic systems. The results obtained in this model can be further improved by incorporating with optimal power flow models. The in passed model help will help both buyers and sellers in the electric market place understand the terms of competition more clearly. It would also be helpful for those wishing to develop bidding strategies for other types of markets.

REFFERENCES

[1] Dariush Shirmohammadi and Chifong L.Thomas "Valuation of the transmission Impact in a resource bidding process" IEEE Transactions on power systems, Vol.6, No.1, February 1991.pp 316-323.

[2] Handke.J, Handschin.E, Linke.K.and Sandeers.H.H "Coordination of long- and Short-term generation planning in thermal power systems" IEEE Transaction on power systems, Vol.10, No.2, May 1995, PP 803-809.

[3] Charles W.Richter, Gerald B.sheble, " Genetic Algorithm Evaluation of Utility Bidding Strategies for the Competitive Marketplace. IEEE Transaction on Power Systems, Vol. 13.No. 1, February 1998, PP 256-261.

[4] A.K. David, " Competitive Bidding in Electricity supply", IEE Proceedings-c, Vol. 140, No. 5, September 1993.

[5] Edward P.Kahn, " Optimization Method for evaluating Competitive Bids from Privet Producers", IEEE Transaction on Power Systems, Vol. 6, No. 2, May 1991, PP 421-426.

[6] Narayan S.Rau, " Optimal Dispatch of a System based on Offer s and Bids-A Mixed Integer LP Formulation", IEEE Transaction on Power Systems, Vol. 14, No. 1, February 1999, PP 274-279.

[7] Chttopadhyyay .D. And Ramanathan.R. " A New Approach To Evaluate Generation Capacity Bids" IEEE Transaction on Power Systems, Vol.13, No.4, November 1998, PP 1232-1237

[8] Yang. H.T.and Chel. S.T. " Incorporating A multi-Criteria Decision Procedure into the Combined Dynamic Programming/ Production Simulation Algorithm for Generation Planning" IEEE Transactions on Power systems, Vol. 4, No.1, February 1989, PP 165-173.

[9] Chao-an Li et.al, “Revenue Adequate Biding Strategies in Competitive Electricity Markets”, IEEE Transaction on Power Systems, Vol. 14, No. 2,May 1999, PP 429-497.

[10] Lotfi A Zadeh, Fuzzy Set Theory.

[11] Wood and Woolen B., Power System Generation Operation and Control.

[12] LEFTERI H. TSOUKALAS AND ROBERT E UHRIG “ FUZZY and NUERAL Approaches in Engineering.

APPENDIX-A

Input to The AHP Pair-Wise Comparison Matrices

Table A1: For bid 1 pair-wise comparison matrix

	FLEX	FUEL	REPU	ENGEN	ENTR
FLEX	1	2	3	5	8
FUEL	0.5	1	2	3	4
REPU	0.333	0.5	1	5	2
ENGEN	0.2	0.333	0.5	1	3
ENTR	0.125	0.25	0.333	0.333	1

Table A11: For criteria FLEX pair-wise comparison matrix

	Good	Normal	Bad
Good	1	3	5
Normal	0.333	1	2
Bad	0.20	0.5	1

Table A12: For criteria FUEL pair-wise comparison matrix

	Good	Normal	Bad
Good	1	4	5
Normal	0.25	1	4
Bad	0.20	0.25	1

Table A13: For criteria REPU pair-wise comparison matrix

	Good	Normal	Bad
Good	1	3	4
Normal	0.333	1	2
Bad	0.25	0.50	1

Table A14: For criteria ENGEN pair-wise comparison matrix

	Good	Normal	Bad
Good	1	5	6
Normal	0.20	1	2
Bad	0.176	0.50	1

TableA15: For criteria ENTR pair-wise comparison matrix

	Good	Normal	Bad
Good	1	3	6
Normal	0.333	1	2
Bad	0.167	0.50	1

Table A2: For bid 2 pair-wise comparison matrix

	FLEX	FUEL	REPU	ENGEN	ENTR
FLEX	1	3	4	6	2
FUEL	0.333	1	2	8	3
REPU	0.25	0.50	1	2	6
ENGEN	0.167	0.125	0.50	1	9
ENTR	0.5	0.333	0.167	0.111	1

Table A21: For criteria FLEX pair-wise comparison matrix

	Good	Normal	Bad
Good	1	3	6
Normal	0.333	1	2
Bad	0.167	0.5	1

Table A22: For criteria FUEL pair-wise comparison matrix

	Good	Normal	Bad
Good	1	5	8
Normal	0.20	1	6
Bad	0.125	0.167	1

Table A23: For criteria REPU pair-wise comparison matrix

	Good	Normal	Bad
Good	1	5	2
Normal	0.2	1	4
Bad	0.5	0.25	1

Table A24: For criteria ENGEN pair-wise comparison matrix

	Good	Normal	Bad
Good	1	6	7
Normal	0.167	1	5
Bad	0.142	0.20	1

Table A25: For criteria ENTR pair-wise comparison matrix

	Good	Normal	Bad
Good	1	2	3
Normal	0.50	1	5
Bad	0.333	0.20	1

TableA3: for bid 3 pair-wise comparison matrix

	FLEX	FUEL	REPU	ENGEN	ENTR
FLEX	1	2	3	4	5
FUEL	0.50	1	2	3	4
REPU	0.333	0.50	1	4	6
ENGEN	0.25	0.333	0.25	1	3
ENTR	0.20	0.25	0.167	0.333	1

Table A31: For criteria FLEX pair-wise comparison matrix

	Good	Normal	Bad
Good	1	4	6
Normal	0.25	1	2
Bad	0.167	0.50	1

Table A32: For criteria FUEL pair-wise comparison matrix

	Good	Normal	Bad
Good	1	2	4
Normal	0.50	1	2
Bad	0.25	0.50	1

Table A33: For criteria REPU pair-wise comparison matrix

	Good	Normal	Bad
Good	1	4	5
Normal	0.25	1	2
Bad	0.20	0.50	1

Table A34: For criteria ENGEN pair-wise comparison matrix

	Good	Normal	Bad
Good	1	3	5
Normal	0.333	1	2
Bad	0.20	0.50	1

Table 35: For criteria ENTR pair-wise comparison matrix

	Good	Normal	Bad
Good	1	4	7
Normal	0.25	1	3
Bad	0.142	0.333	1

Table A4: For bid 4 pair-wise comparison matrix

	FLEX	FUEL	REPU	ENGEN	ENTR
FLEX	1	6	7	9	2
FUEL	0.167	1	2	3	4
REPU	0.142	0.50	1	3	5
ENGEN	0.111	0.333	0.333	1	6
ENTR	0.50	0.25	0.20	0.167	1

Table A41: For criteria FLEX pair-wise comparison matrix

	Good	Normal	Bad
Good	1	3	7
Normal	0.333	1	6
Bad	0.142	0.167	1

Table A42: For criteria FUEL pair-wise comparison matrix

	Good	Normal	Bad
Good	1	3	5
Normal	0.333	1	2
Bad	0.20	0.50	1

Table A43: For criteria REPU pair-wise comparison matrix

	Good	Normal	Bad
Good	1	7	5
Normal	0.142	1	6
Bad	0.20	0.167	1

Table A44: For criteria ENGEN pair-wise comparison matrix

	Good	Normal	Bad
Good	1	4	3
Normal	0.25	1	2
Bad	0.333	0.50	1

Table A45: For criteria ENTR pair-wise comparison matrix

	Good	Normal	Bad
Good	1	9	5
Normal	0.111	1	3
Bad	0.20	0.333	1

Table A5: For bid 5 pair-wise comparison matrix

	FLEX	FUEL	REPU	ENGEN	ENTR
FLEX	1	3	5	6	7
FUEL	0.333	1	2	7	6
REPU	0.20	0.50	1	9	8
ENGEN	0.167	0.142	0.111	1	6
ENTR	0.142	0.167	0.125	0.167	1

Table A51: For criteria FLEX pair-wise comparison matrix

	Good	Normal	Bad
Good	1	7	2
Normal	0.142	1	6
Bad	0.50	0.167	1

Table A52: For criteria FUEL pair-wise comparison matrix

	Good	Normal	Bad
Good	1	6	7
Normal	0.167	1	7
Bad	0.142	0.142	1

Table A53: For criteria REPU pair-wise comparison matrix

	Good	Normal	Bad
Good	1	5	6
Normal	0.20	1	7
Bad	0.167	0.142	1

Table A54: For criteria ENGEN pair-wise comparison matrix

	Good	Normal	Bad
Good	1	7	6
Normal	0.142	1	3
Bad	0.167	0.333	1

Table A55: For criteria ENTR pair-wise comparison matrix

	Good	Normal	Bad
Good	1	7	2
Normal	0.142	1	3
Bad	0.50	0.333	1

Table A6: For bid 6 pair-wise comparison matrix

	FLEX	FUEL	REPU	ENGEN	ENTR
FLEX	1	7	6	2	4
FUEL	0.142	1	4	2	7
REPU	0.167	0.25	1	7	6
ENGEN	0.50	0.50	0.142	1	4
ENTR	0.25	0.142	0.167	0.25	1

Table A61: For criteria FLEX pair-wise comparison matrix

	Good	Normal	Bad
Good	1	4	8
Normal	0.25	1	2
Bad	0.125	0.50	1

Table A62: For criteria FUEL pair-wise comparison matrix

	Good	Normal	Bad
Good	1	7	4
Normal	0.142	1	6
Bad	0.25	0.67	1

Table A63: For criteria REPU pair-wise comparison matrix

	Good	Normal	Bad
Good	1	4	8
Normal	0.25	1	9
Bad	0.125	0.111	1

Table A64: For criteria ENGEN pair-wise comparison matrix

	Good	Normal	Bad
Good	1	3	2
Normal	0.333	1	4
Bad	0.50	0.25	1

Table A65: For criteria ENTR pair-wise comparison matrix

	Good	Normal	Bad
Good	1	6	9
Normal	0.167	1	4
Bad	0.111	0.25	1

TableA7: for bid 7 pair-wise comparison matrix

	FLEX	FUEL	REPU	ENGEN	ENTR
FLEX	1	9	8	2	5
FUEL	0.111	1	2	6	3
REPU	0.125	0.50	1	3	2
ENGEN	0.50	0.167	0.333	1	4
ENTR	0.20	0.333	0.50	0.125	1

Table A71: For criteria FLEX pair-wise comparison matrix

	Good	Normal	Bad
Good	1	5	8
Normal	0.20	1	5
Bad	0.125	0.20	1

Table A72: For criteria FUEL pair-wise comparison matrix

	Good	Normal	Bad
Good	1	4	2
Normal	0.125	1	3
Bad	0.50	0.333	1

Table A73: For criteria REPU pair-wise comparison matrix

	Good	Normal	Bad
Good	1	9	2
Normal	0.111	1	3
Bad	0.50	0.333	1

Table A74: For criteria ENGEN pair-wise comparison matrix

	Good	Normal	Bad
Good	1	8	2
Normal	0.125	1	4
Bad	0.50	0.25	1

Table A75: For criteria ENTR pair-wise comparison matrix

	Good	Normal	Bad
Good	1	5	2
Normal	0.20	1	4
Bad	0.50	0.25	1

Table A8: For bid 8 pair-wise comparison matrix

	FLEX	FUEL	REPU	ENGEN	ENTR
FLEX	1	6	3	2	9
FUEL	0.167	1	2	5	7
REPU	0.333	0.50	1	2	8
ENGEN	0.50	0.20	0.50	1	4
ENTR	0.111	0.142	0.125	0.25	1

Table A81: For criteria FLEX pair-wise comparison matrix

	Good	Normal	Bad
Good	1	6	2
Normal	0.167	1	3
Bad	0.50	0.333	1

Table A82: For criteria FUEL pair-wise comparison matrix

	Good	Normal	Bad
Good	1	4	2
Normal	0.25	1	9
Bad	0.50	0.111	1

Table A83: For criteria RUPE pair-wise comparison matrix

	Good	Normal	Bad
Good	1	7	3
Normal	0.142	1	2
Bad	0.333	0.50	1

Table A84: For criteria ENGEN pair-wise comparison matrix

	Good	Normal	Bad
Good	1	3	8
Normal	0.333	1	2
Bad	0.125	0.50	1

TableA85: For criteria ENTR pair-wise comparison matrix

	Good	Normal	Bad
Good	1	9	8
Normal	0.111	1	2
Bad	0.125	0.50	1

Table A9: For bid 9 pair-wise comparison matrix

	FLEX	FUEL	REPU	ENGEN	ENTR
FLEX	1	7	3	4	2
FUEL	0.142	1	5	7	3
REPU	0.333	0.20	1	2	7
ENGEN	0.25	0.142	0.50	1	2
ENTR	0.50	0.333	0.142	0.50	1

Table A91: For criteria FLEX pair-wise comparison matrix

	Good	Normal	Bad
Good	1	4	2
Normal	0.25	1	3
Bad	0.50	0.333	1

Table A92: For criteria FUEL pair-wise comparison matrix

	Good	Normal	Bad
Good	1	2	3
Normal	0.50	1	5
Bad	0.333	0.20	1

Table A93: For criteria REPU pair-wise comparison matrix

	Good	Normal	Bad
Good	1	9	2
Normal	0.111	1	3
Bad	0.50	0.333	1

Table A94: For criteria ENGEN pair-wise comparison matrix

	Good	Normal	Bad
Good	1	6	3
Normal	0.167	1	2
Bad	0.333	0.50	1

Table A95: For criteria ENTR pair-wise comparison matrix

	Good	Normal	Bad
Good	1	2	3
Normal	0.50	1	2
Bad	0.333	0.50	1

Table A10: For bid 10 pair-wise comparison matrix

	FLEX	FUEL	REPU	ENGEN	ENTR
FLEX	1	4	2	3	5
FUEL	0.25	1	3	4	6
REPU	0.50	0.333	1	2	9
ENGEN	0.333	0.25	0.50	1	3
ENTR	0.20	0.167	0.111	0.333	1

Table A101: For criteria FLEX pair-wise comparison matrix

	Good	Normal	Bad
Good	1	6	9
Normal	0.167	1	2
Bad	0.111	0.50	1

Table A102: For criteria FUEL pair-wise comparison matrix

	Good	Normal	Bad
Good	1	8	2
Normal	0.125	1	4
Bad	0.50	0.25	1

Table A103: For criteria REPU pair-wise comparison matrix

	Good	Normal	Bad
Good	1	2	9
Normal	0.50	1	4
Bad	0.111	0.25	1

Table A104: For criteria ENGEN pair-wise comparison matrix

	Good	Normal	Bad
Good	1	7	2
Normal	0.142	1	3
Bad	0.50	0.333	1

Table A105: For criteria ENTR pair-wise comparison matrix

	Good	Normal	Bad
Good	1	5	2
Normal	0.20	1	4
Bad	0.50	0.25	1

APPENDIX-B

Unit Characteristics, Load patterns, and Initial Status of the generators

Table 1: ABC Co-efficients

No. Bids	A	B	C
1	30.0	2.25	0.005
2	30.0	1.75	0.0045
3	30.0	2.1	0.004
4	30.0	2.25	0.005
5	30.0	1.75	0.009
6	30.0	2.1	0.004
7	30.0	2.0	0.00375
8	30.0	1.75	0.00175
9	30.0	1.0	0.0625
10	30.0	3.25	0.00834

Table 2: Unit Characteristics and Initial status

No. Bids	Max(MW) Power	Min(MW) Power	Up time (hour)	Down-time(hour)	Hot cost (RS)	Cold cost (RS)
1	520	250	5	3	267	0
2	320	120	4	2	187	0
3	160	50	3	2	113	0
4	200	80	3	2	176	0
5	280	80	4	2	180	0
6	150	50	3	2	113	0
7	120	30	3	2	94	0
8	110	30	3	2	114	0
9	80	20	0	0	101	0
10	60	20	0	0	85	0

Table 3: Hourly load Patterns

Time duration	Load (MW)
1	1000
2	950
3	900
4	1200
5	1300
6	1230
7	1150
8	1300

APPENDIX-C

```
      /* PROGRAM FOR Analytic Hierarchy Process (AHP) */

#include<stdio.h>
#include<math.h>
#define toler 0.0001
#define RI 0.9
#define N 10

void main()
{
FILE*fp,*fq,*f;
float a[10] [10],b[10],c[10] [10],temp[10],d[10] [10];
float maxc,p[10] [10],m,dd[10] [10],k[10] [10],CR;
float pp[10] [10],mt,mtt,bb[10],ddd,d2,z,ddn[10] [10],dda[10] [10];
float w[10] [10],W[10],sum;
int i,ii,j,n,t,g,bus_no;
clrscr();
fp=fopen("uor.dat","w");
for(i=1;i<=72;i++)
fprintf(fp,"-");
fprintf(fp,"\nBid#at bus#\tFLEX\tFUEL\tREPU\tENGEN\tENTR\tAHP SCORE\n");
for(i=1;i<=72;i++)
fprintf(fp,"-");
fprintf(fp,"\n");
f=fopen("sd.dat","w");
for(ii=1;ii<=N;ii++)
{
printf("ENTER BUS NO.\n");
scanf("%d",&bus_no);

fprintf(fp,"%d\t(bus#%d)\t",ii,bus_no);
fq=fopen("mt1.dat","r");
for(g=0;g<=5;g++)
{
top:
fscanf(fq,"%d",&n);
for(i=1;i<=n;i++)
for(j=1;j<=n;j++)
fscanf(fq,"%f",&a[i] [j]);

for(j=1;j<=n;j++)
b[j]=1;
temp[j]=0.0;

/*start*/
abc: for(i=1;i<=n;i++)
          {
       c[g] [i]=0;
       for(j=1;j<=n;j++)
       {
       c[g] [i]+=a[i] [j]*b[j];
       }
}
for(i=1;i<=n;i++)
{
```

```
      k[g][i]=c[g][i];
      d[g][i]=c[g][i];
 }
maxc=c[g][1];
for(i=1;i<=n;i++)
{
      if(fabs(maxc)<fabs(c[g][i]))
      maxc=c[g][i];
}
for(i=1;i<=n;i++)
{
      p[g][i]=c[g][i]/maxc;
      dd[g][i]=p[g][i];
}
if(fabs(k[g][2])>fabs(temp[2]))
{
m=(fabs(k[g][2])-fabs(temp[2]));
}
 else
 {
 m=(fabs(temp[2])-fabs(k[g][2]));
 }
 if(m>toler)
 {
 for(i=1;i<=n;i++)
 {
 b[i]=dd[g][i];
 }
 temp[2]=k[g][2];
 goto abc;
 }

/*consistency check*/

 CR=((maxc-n)/(n-1))/RI;
 if(CR>0.1)
 {
printf("repeat for same decison maker\n");
  goto top;
  }
/*normalization of eigen vector*/

  else
  {
 z=0;
for(i=1;i<=n;i++)
{
z+=dd[g][i]*dd[g][i];
}
z=sqrt(z);
for(i=1;i<=n;i++)
ddn[g][i]=dd[g][i]/z;  /*  normelize value */
/*finish*/
}

fprintf(f,"\n");
fprintf(f,"CR=%f/n",CR);
printf("give new matrix of 3x3  \n");
for(i=1;i<=n;i++)
fprintf(f,"Normalize value=%f\n",ddn[g][i]);
}
```

```
fprintf(f,"\n");
for(i=1;i<=5;i++)
{
for(j=1;j<=3;j++)
{
w[i][j]=ddn[0][i]*ddn[i][j];
}
}
for(i=1;i<=5;i++)
{
for(j=1;j<=3;j++)
{
fprintf(f,"w[%d][%d]=%f\t",i,j,w[i][j]);
}
fprintf(f,"\n");
}
fprintf(f,"\n");
/*selection start*/
for(i=1;i<=5;i++)
{
printf("input good/normal/bad (1/2/3) for cretira %d\n",i);
scanf("%d",&j);
fprintf(fp,"%d\t",j);

W[i]=w[i][j];
}
sum=0;
for(i=1;i<=5;i++)
sum+=W[i];
fprintf(f,"AHP Score == %f\n",sum);

fprintf(fp,"%f\t",sum);
fprintf(fp,"\n");
}
fprintf(f,"\n");

fprintf(fp,"1=Good,2=Normal,3=Bad\n");
getch();
}
```

```
/*PROGRAMM FOR UNIT COMMITMENT USING DYNAMIC PROGRAMMING*/

#include<math.h>
#include<stdio.h>
#define tp 350
#define thh 1050
#define th 10
int n,t,mm;
float pput[th];
void sdram(float putt[th],float min[th],float p[th],float pl[th],int
count);
arang (float pcost[th],int up_time[th],int *time);
void main()
{
FILE*st,*stt,*ff;
int j,y,ta,x,q,i,n2,l,tt[thh][th],count,ttmp[th],ttemp[th],ttt[thh][th];
int vvh[th],vvc[th],cont[th],dcont[th],vk[th],ttem[th],time,lkk[th][th];
int ls[th],lk[th][th],co,npp,lss[th][th],hk[th],up_time[th],ttime[th];
float
p[th],pl[th],s[thh],aa[th][3],cost[th],put[th],putt[th],cctt[th][thh];
float
min[th],pputt[th],temp,pp[thh][th],kk,kkk,bb[th],cc[th],tpp,we,ppcost[th
];
float
ddh[th],ddc[th],hh[th],coc[th],cstt[th][thh],aahh,aacc,ahc,pcost[th];
float hkk[th];
  st=fopen("uc.dat","r");
  fscanf(st,"%d",&n);
  for(j=1;j<=n;j++){
  fscanf(st,"%f\n",&p[j]);}
  fclose(st);
  ff=fopen("final.dat","w");
  clrscr();
for(i=0;i<=thh;i++)
{
s[i]=0.0;
}
x=pow(2,n)-1;
for(i=0;i<=x;i++)
{
y=i;
  for(j=1;j<=n;j++)
    {
      ta=y/2;
      tt[i][j]=(y-2*ta);
       y=ta;
       s[i]+=p[j]*tt[i][j];
    }
}
stt=fopen("ucc.dat","w");

for(i=0;i<=x;i++)
{
for(j=1;j<=n;j++)
{
 fprintf(stt,"%d",tt[i][j]);
 }
fprintf(stt,"\t\t\t s[%d]\t\t=%f\n",i,s[i]);
```

```
}
  for(i=0;i<=x;i++)
  {
    for(q=i+1;q<=x;q++)
      {
      if(s[i]>s[q])
       {
           temp=s[q];for(j=1;j<=n;j++) ttemp[j]=tt[q][j];
           s[q]=s[i];for(j=1;j<=n;j++) tt[q][j]=tt[i][j];
           s[i]=temp;for(j=1;j<=n;j++) tt[i][j]=ttemp[j];
       }
      }
}

for(i=0;i<=x;i++)
{
  for(j=1;j<=n;j++)fprintf(stt,"%d",tt[i][j]);
    fprintf(stt,"\t\ts[%d]\t\t=%f\n", i,s[i]);
}
  st=fopen("abc.dat","r");
fscanf(st,"%d",&n);
for(i=1;i<=n;i++){
for(j=1;j<=3;j++){
fscanf(st,"%f\n",&aa[i][j]);}}
fclose(st);
for(j=1;j<=n;j++)
{
put[j]=cost[j]=pcost[j]=0.0;
}
for(j=1;j<=n;j++)
{
cost[j]=(aa[j][1]+aa[j][2]*p[j]+aa[j][3]*p[j]*p[j]);
pcost[j]=cost[j];
put[j]=cost[j]/p[j];
}
for(j=1;j<=n;j++){
putt[j]=pputt[j]=pput[j]=0.0;
}
st=fopen("pload.dat","r");
fscanf(st,"%d",&n2);
for(j=1;j<=n2;j++){
fscanf(st,"%f\n",&pl[j]);  }
  fclose(st);

st=fopen("in.dat","r");
fscanf(st,"%d",&n);
for(j=1;j<=n;j++){
fscanf(st,"%f\n",&min[j]);}
fclose(st);
x=pow(2,n)-1;
       npp=0;
for(t=1;t<=n2;t++){
for(i=0;i<=x;i++){
cstt[t][i]=pp[t][i]=0.0;
}}
for(t=1;t<=n2;t++){
for(j=1;j<=n;j++)lss[t][j]=0;
}
for(i=0;i<=x;i++){
for(j=1;j<=n;j++){
ttt[i][j]=up_time[j]=0;
```

```
})
for(j=1;j<=n;j++){cont[j]=dcont[j]=ls[j]=hk[j]=0;}

st=fopen("up.dat","r");
fscanf(st,"%d",&n);
for(j=1;j<=n;j++){
fscanf(st,"%f",&bb[j]);}
fclose(st);
st=fopen("down.dat","r");
fscanf(st,"%d",&n);
for(j=1;j<=n;j++){
fscanf(st,"%f",&cc[j]);}
fclose(st);
time=0;
for(t=1;t<=n2;t++)
{
printf("\na\n");
for(i=0;i<=x;i++)
{
            if(s[i]>=pl[t])
            {
                 count=0;
                   for(j=1;j<=n;j++ )
                   {
                   putt[j]=put[j]*tt[i][j];
                       if(putt[j]>0 .1)
                        {
                        count=count+1;
                        }
                  }

sdram(putt,min,p,pl,count);
                  for(j=1;j<=n;j++)
                  {
                  pputt[j]=pput[j]*tt[i][j];

      cost[j]=(aa[j][1]+aa[j][2]*pputt[j]+aa[j][3]*pputt[j]*pputt[j]);
                  cstt[t][i]=cstt[t][i]+cost[j];
                  ttt[i][j]=tt[i][j];
                  }
                  pp[t][i]=cstt[t][i];
                  /*for(j=1;j<=n;j++)fprintf(stt,"%d",ttt[i][j]);
                  fprintf(stt,"\t%d %d %f\n",t,i,pp[t][i]);*/
            }
      }
      for(i=0;i<=x;i++)
      {
            if(s[i]>=pl[t])
            {
                  for(q=1+i;q<=x;q++)
                  {
                        if(cstt[t][i]<cstt[t][q])
                        {
                        kk=cstt[t][i];for(j=1;j<=n;j++)ttmp[j]=tt[i][j];

     cstt[t][i]=cstt[t][q];for(j=1;j<=n;j++)tt[i][j]=tt[q][j];
                        cstt[t][q]=kk;for(j=1;j<=n;j++)tt[q][j]=ttmp[j];
                        }
                        else
                        {
                        kk=cstt[t][q];for(j=1;j<=n;j++)ttmp[j]=tt[q][j];
```

```
                 }
             }
        }
    }

for(j=1;j<n;j++)printf("%d",ttmp[j]);
printf("\tkk=%f\n",kk);

st=fopen("hot.dat","r");
fscanf(st,"%d",&n);
for(j=1;j<=n;j++){
fscanf(st,"%f\n",&hh[j]);}
fclose(st);

st=fopen("cold.dat","r");
fscanf(st,"%d",&n);
for(j=1;j<=n;j++){
fscanf(st,"%f\n",&coc[j]);}
fclose(st);
for(j=1;j<=n;j++)
{
vk[j]=0;
}
if(t>1)
{
       for(i=0;i<=x;i++)
       {
         if(s[i]>=pl[t])
         {
            for(j=1;j<=n;j++)
            {
                 if(ttt[i][j]==0)
                  {
                    vvh[j]=0 ;vvc[j]=1;
                  }
                 else
                  {
                   vvh[j]=1;vvc[j]=0;
                  }

           }
         for(j=1;j<=n;j++){ ddh[j]=ddc[j]=0.0;  }
          aahh=aacc=ahc=0.0;

         for(j=1;j<=n;j++)
         {
              ddh[j]=vvh[j]*hh[j];
              aahh=aahh+ddh[j];
              ddc[j]=vvc[j]*coc[j];
              aacc=aacc+ddc[j];
          }
       if(npp==i)
       {
         cstt[t][i]=cstt[t][i];
       }
       else
       {
              ahc=(aahh+aacc);
              cstt[t][i]=pp[t][i]+ahc+tp;
       }
              pp[t][i]=cstt[t][i];
```

```
               for(j=1;j<=n;j++)tt[i][j]=ttt[i][j];
               for(j=1;j<=n;j++)fprintf(stt,"%d",tt[i][j]);
               fprintf(stt,"\tcstt[%d][%d]=%f\n",t,i,pp[t][i]);

    }
}

for(i=0;i<=x;i++)
{
    if(s[i]>=pl[t])
    {
          for(q=1+i;q<=x;q++)
          {
               if(cstt[t][i]<cstt[t][q])
               {
               kkk=cstt[t][i];for(j=1;j<=n;j++)ttem[j]=ttt[i][j];

     cstt[t][i]=cstt[t][q];for(j=1;j<=n;j++)ttt[i][j]=ttt[q][j];
               cstt[t][q]=kkk;for(j=1;j<=n;j++)ttt[q][j]=ttem[j];
               }
               else
               {
               kkk=cstt[t][q];for(j=1;j<=n;j++)ttem[j]=ttt[q][j];
               }
          }
    }
}
for(j=1;j<=n;j++)fprintf(stt,"%d",ttem[j]);
fprintf(stt,"\t%d kkk=%f\n",t,kkk);
}
     if(t==1)
     {
     for(j=1;j<=n;j++)vk[j]=ttmp[j];
     }
          else
          {
          for(j=1;j<=n;j++)vk[j]=lss[t][j];
          }

if(t>1){
for(j=1;j<=n;j++)fprintf(stt,"%d",lss[t][j]);
fprintf(stt,"\n");
}
     for(j=1;j<=n;j++)
     {
          if(vk[j]==1)
          {
          cont[j]=cont[j]+1;dcont[j]=0;
          }
               else
               {
               cont[j]=0;dcont[j]=dcont[j]+1;
               }
fprintf(stt,"cont[%d]=%d     dcont[%d]=%d\n",j,cont[j],j,dcont[j]);
   }
fprintf(stt,"\n");

     if(t==1)
     {
     for(j=1;j<=n;j++)hk[j]=vk[j];
     tpp=kk;
```

```
       }
 else
 {
     for(j=1;j<=n;j++)
     {
         if((cont[j]<=bb[j])&&(lk[t-1][j]==1))
          {
           ls[j]=1;
          }
                   else
                   {
                         if((dcont[j]<=cc[j])&&(lk[t-1][j]==0))
                         {
                         ls[j]=0;
                         }
                               else
                               {
                               ls[j]=vk[j];
                               }
                   }
}}
for(j=1;j<=n;j++)fprintf(stt,"%d",ls[j]);
if(t>1){
for(i=0;i<=x;i++)
{
      if(s[i]>=pl[t])
      {
         co=0;
            for(j=1;j<=n;j++)
            {
                  if(ls[j]==tt[i][j])
                  {
                       co=co+1;
                             if(co==4)
                             {
                             tpp=pp[t][i];
                             npp=i;
                             for(j=1;j<=n;j++)hk[j]=tt[i][j];
                             i=x+1;
                             j=n+1;
                             }
                         }
                  }
             }
       }
}
for(j=1;j<=n;j++)lss[t][j]=hk[j];
for(j=1;j<=n;j++)lk[t][j]=hk[j];
for(j=1;j<=n;j++)lkk[t][j]=hk[j];
fprintf(stt,"\n Selected Combination and Their Cost \n");
for(j=1;j<=n;j++)fprintf(stt,"%d",hk[j]);
fprintf(stt,"\t T=%d tpp=%f\n\n",t,tpp);
hkk[t]=tpp;

      for(i=0;i<=x;i++)
      {
            if(s[i]>=pl[t])
            {
                  for(j=1;j<=n;j++)
                  {
                        if(vk[j]==tt[i][j])
```

```
                         {
                         npp=i;
                         i=x+1;
                         j=n+1;
                         }
                   }
             }
       }
for(j=1;j<=n;j++)
{
  if(hk[j]>=1)
  {
  up_time[j]+=1;
  }
}}
fprintf(ff,"\n TOTAL UP-TIME OF GENERATING PLANTS\n");
for(j=1;j<=n;j++)
{
i=j+2;
ttime[i]=up_time[j];
fprintf(ff,"Total Working Time for Unit %d\t =%d\n",i,ttime[i]);
}
fprintf(ff,"\nCOST OF Particular GENERATING PLANTS\n");
for(j=1;j<=n;j++)
{
i=j+2;
ppcost[i]=pcost[j];
fprintf(ff,"pcost[%d]=%f\n",i,ppcost[i]);
}
fprintf(ff,"\nPRIORITY ORDER DEPENDING UP-ON UP-TIME AND COST OF
GENERATING PLANTS\n");
for(j=1;j<=n;j++)
{
arang(pcost,up_time,&time);
i=time+2;
fprintf(ff,"time=%d\t   %d\n",i,mm);
up_time[time]=-1;
}
fprintf(ff,"\nTime    combination        cost of combination \n");
for(t=1;t<=n2;t++)
{
j=t+2;
fprintf(ff,"%d\t",j);
for(j=1;j<=n;j++)fprintf(ff,"%d", lkk[t][j]);
j=t+2;
fprintf(ff,"\thkk[%d] = %f\n",j,hkk[t]);
fprintf(ff,"\n");
}
printf("\nsdram\n");
getch();
}


arang(float pcost[th],int up_time[th],int *time)
{
int j;
mm=up_time[1];
*time=1;
for(j=2;j<=n;j++)
{
  if(up_time[j]==mm)
```

```
  {
      if(pcost[j]<pcost[*time])
      {
      *time=j;
      }
  }
else
{
if(up_time[j]>mm)
{
*time=j;
mm=up_time[j];
}}
}
return ;
}

void sdram(float putt[th],float min[th],float p[th],float pl[th],int
count)
{
int j,tat,kp[th],ii;
float sum,z;
sum=0.0;tat=0;ii=0;z=0;
if(count==1)
{
ram(putt,&tat);
pput[tat]=p[tat];
}
else
{
 do
 {
     ii=ii+1;
    ram(putt,&tat);
            pput[tat]=p[tat];
            sum=sum+pput[tat];
            kp[ii]=tat;
            count=count-1;
            putt[tat]=-1;
   }
while((count>=2)&&(sum<pl[t]));
do
  {
     ram(putt,&tat);
            pput[tat]=min[tat];
            pput[kp[ii]]=pput[kp[ii]]-min[tat];
            count=count-1;
            putt[tat]=-1;
                  while(pput[kp[ii]]<min[kp[ii]])
                  {
                      z=fabs(pput[kp[ii]]-min[kp[ii]]);
                      pput[kp[ii]]=min[kp[ii]];
                      ii=ii-1;
                      pput[kp[ii]]=pput[kp[ii]]-z;

                      }
      }
       while(count>0.1);
      }
 }
ram(float putt[th],int *tat)
```

```
{
int j;
float m;
        m=0.1;
        for(j=1;j<=n;j++)
        {
                if(putt[j]>m)
                {
                *tat=j;
                m=putt[j];
                }
        }
        return ;
```

/PROGRAM FOR DECISION MAKING (HURWITZ CRITERIA) */

```
 #define N2 4
 #define N1 10
 #include<stdio.h>
 #include<conio.h>
 #include<math.h>
 #include<dos.h>
 void fun(float k[20] [10],int i,int a[20]);
float   maximum(float,float,int *,int);
void  sdram(float hv[20],float mhv[20], float dd[20] [10],int pp[20]);
void main()
{
FILE*fp;
FILE*fb;
FILE*fq;
float
d[20] [10],tt[20] [10],item[20],dd[20] [10],t[10],k[20] [10],term,h[20];
float  kk[20] [10],lt[20],hv[20],m,mhv[20],bus[20],maxd[10];
int i,j,N,p,ct,l,a[20],x[20],pp[20],n,sp,ctt;
fp=fopen("ss.dat","r");
fq=fopen("thesis.dat","w");
fscanf(fp,"%d",&N);
clrscr();
for(i=1;i<=N1;i++)
{
for(j=1;j<=4;j++)
{
fscanf(fp,"%f\t",&d[i] [j]);
}}
fprintf(fq,"\n DATAT FIND FOR FINAL PRIORITY ORDERING ARE AS:\n");
fprintf(fq,"bid No.  Total operating time  operating cost  SO2 co-
efficient \n");
for(i=1;i<=N1;i++)
{
for(j=1;j<=4;j++)
{
fprintf(fq,"%f\t",d[i] [j]);
}
fprintf(fq,"\n");
}
for(j=1;j<=4;j++)
{
for(i=1;i<=N1;i++)
{
dd[i] [j]=d[i] [j];
kk[i] [j]=k[i] [j];
}}
for(j=1;j<=4;j++)
{
for(p=1;p<=N1;p++)
{
for(i=p+1;i<=N1;i++)
{
if(d[p] [j]>d[i] [j])
{
t[j]=d[p] [j];
d[i] [j]=t[j];
```

```
}
else
{
t[j]=d[i][j];
}}}}
fprintf(fq,"\n MAXIMUM VALU OF ALL CRITERIA \n");
for(j=1;j<=4;j++)
{
fprintf(fq,"maxd[%d]= %f\n",j,t[j]);
}
fprintf(fq,"\n");

for(j=1;j<=4;j++)
{
for(i=1;i<=N1;i++)
{
k[i][j]=dd[i][j]/t[j];
}}
fprintf(fq,"\n Variables after Normalization \n");
for(i=1;i<=N1;i++)
{
for(j=1;j<=4;j++)
{
fprintf(fq,"%f\t",k[i][j]);
}
fprintf(fq,"\n");
}
fprintf(fq,"\n");

for(j=3;j<=4;j++)
{
for(i=1;i<=N1;i++)
{
k[i][j]=1-k[i][j];
}}
fprintf(fq,"\nCOMPLIMENT OF THE VARIABLES WHICH HAVE TO BE MINIMIZE
\n\n");
for(i=1;i<=N1;i++)
{
for(j=1;j<=4;j++)
{
fprintf(fq,"%f\t",k[i][j]);
}
fprintf(fq,"\n");
}
fprintf(fq,"\n");

     /*START NOW*/

for(j=1;j<=4;j++)
 {
 fun(k,j,a);
 for(i=1;i<=N1;i++)
 {
 k[a[i]][j]=1-(float) (i-1)/10;
 }}
 fprintf(fq," \n Approximation of the variables \n\n");
for(i=1;i<=N1;i++)
{
for(j=1;j<=4;j++)
{
```

```
fprintf(fq,"%f\t",k[i][j]);
}
fprintf(fq,"\n");
}
fprintf(fq,"\n");

 for(i=1;i<=N1;i++)
 {
 for(j=1;j<=4;j++)
 {
 tt[i][j]=k[i][j];
 }}
 for(i=1;i<=N1;i++)
 {
 for(j=1;j<=4;j++)
 {
 fprintf(fq,"%f\t",tt[i][j]);
 }
 fprintf(fq,"\n");
 }
 fprintf(fq,"\n");

 for(i=1;i<=N1;i++)
 {
 for(l=1;l<=4;l++)
 {
 for(j=l+1;j<=4;j++)
 {
 if(k[i][l]>k[i][j])
 {
  h[i]=k[i][l];
  k[i][j]=h[i];
  }
  else
  {
  h[i]=k[i][j];
  }}}}
   fprintf(fq,"Higher value from all  criteria for particular Bid \n\n");
   for(i=1;i<=N1;i++)
   {
   j=i+2;
 fprintf(fq,"h[%d]=%f\t",j,h[i]);
  }
 fprintf(fq,"\n");
 fprintf(fq,"\n");
 for(i=1;i<=N1;i++)
  {
  for(l=1;l<=4;l++)
  {
  for(j=l+1;j<=4;j++)
  {
  if(tt[i][l]<tt[i][j])
  {
  lt[i]=tt[i][l];
  tt[i][j]=lt[i];
 }
 else
 {
 lt[i]=tt[i][j];
 tt[i][l]=lt[i];
 }}}}
```

```
fprintf(fq,"\nLower value from all criteria for particural Bid \n\n");
 for(i=1;i<=N1;i++)
    {
    j=i+2;
 fprintf(fq,"lt[%d]=%f\t",j,lt[i]);
      }
fprintf(fq,"\n");
fprintf(fq,"\nFinal result of BIDS selection:-\n\n");
for(i=1;i<=N1;i++)
    {
    hv[i]=0.5*h[i]+0.5*lt[i];
    }
for(i=1;i<=N1;i++)
    {
    j=i+2;
 fprintf(fq,"hv[%d]=%f\n",j,hv[i]);
 }
fprintf(fq,"\n");
fprintf(fq,"\n");
     /*  ARRANGE  TO  THE  BUSES  */

     fprintf(fq,"\nselection of BIDs in decrising order:-\n\n");
sdram(hv,mhv,dd,pp);
 for(i=1;i<=N1;i++)
 {
 ctt=pp[i]+2;
 fprintf(fq,"%d\t%f\n",ctt,mhv[i]);
 fprintf(fq,"\n");
  }
fprintf(fq,"\n FINAL SELECTION OF BIDS INCLUDING TWO HIGHER
PRIORITY\n\n");
for(j=1;j<=10;j++)
{
if(j<=2)
{ct=j;}
else
{
i=j-2;
ct=pp[i]+2;
}
fprintf(fq,"%d\t",ct);
}

   getch();
   }

/*     FUNCTION   */

void fun(float k[20][10],int i,int a[20])
{
int s,t,j,p;
float m;
for(s=1;s<=N1;s++)
{
m=k[1][i];
p=1;
for(t=2;t<=N1;t++)
{
m = maximum(m,k[t][i],&p,t);
}
    a[s]=p;
```

```
    k[p][i]=-1;
    }}
float maximum(float m,float h,int *p,int t)
      {
  if(m<h)
       {
       *p=t;
       m=h;
       }
       return m ;
}

/*  U.O R  */

int nmax(float hv[20],int *p,int *np)
{
  int i,s;
  float m;
  m=hv[1];
  s=1;
  *p=1;
  for(i=2;i<=N1;i++)
  {
    if(hv[i]>m)
    {
     m=hv[i];
     *p=i;
     }}
return s;
}
 /*  ELECTRICAL  ENGG */

void  sdram(float hv[100],float mhv[20], float dd[20][10],int pp[20])
  {
  int i,s,p,np;
  for(i=1;i<=N1;i++)
  {
   s=nmax(hv,&p,&np);
   if(s!=0)
   {
   pp[i]=p;
   mhv[i]=hv[p];
   hv[p]=-1;
   }
 else
 {
   if(dd[p][4]>dd[np][4])
   {
   pp[i]=p;
   mhv[i]=hv[p];
   hv[p]=-1;
   }
 else
    {
      pp[i]=np;
      mhv[i]=hv[np];
      hv[np]=-100;
      }}}}

  /*  THE  END   */
```