OPTIMAL SCHEDULING OF HYDRO-THERMAL POWER GENERATION SYSTEMS

Pedro Nuno Ferreira Pinto de Oliveira
Department of Mathematics
University of Strathclyde

PhD thesis

Submitted 1992

Declaration of Author's Rights.

To the loving memory of my father, Amadeu Pinto de Oliveira (10/10/90)

ITHACA

When you start on your journey to Ithaca,
then pray that the road is long,
full of adventure, full of knowledge.
Do not fear the Lestrygonians
and the Cyclopes and the angry Poseidon.
You will never meet such as these on your path,
if your thoughts remain lofty, if a fine
emotion touches your body and your spirit.
You will never meet the Lestrygonians,
the Cyclopes and the fierce Poseidon,
if you do not carry them within your soul,
if your soul does not raise them up before you.

Then pray that the road is long.
That the summer mornings are many,
that you will enter ports seen for the first time
with such pleasure, with such joy!
Stop at Phoenician markets,
and purchase fine merchandise,
mother-of-pearl and corals, amber and ebony,
and pleasurable perfumes of all kinds,
buy as many pleasurable perfumes as you can;
visit hosts of Egyptian cities,
to learn and learn from those who have knowledge.

Always keep Ithaca fixed in your mind.
To arrive there is your ultimate goal.
But do not hurry the voyage at all.
It is better to let it last for long years;
and even to anchor at the isle when you are old,
rich with all that you have gained on the way,
not expecting that Ithaca will offer you riches.

Ithaca has given you the beautiful voyage.
Without her you would never have taken the road.
But she has nothing more to give you.

And if you find it poor, Ithaca has not defrauded you.
With all the great wisdom you have gained, with so much experience,
you must surely have understood by then what Ithacas mean.

C. Cavafy

Rae Dawen (1971). The Complete Poems of Cavafy. The Hoggarth Press

ACKNOWLEDGEMENTS

I would like to thank my supervisors Prof. S. McKee and Dr. C. W. Coles for their time, assistance and enthusiasm through my years at the University of Strathclyde. I am also indebted to the late Prof. Hj. Wacker with whom I spent sometime in Linz. Also from Linz I would like to thank A. Gangl and H. Guttenberger for their true academic spirit and for the many discussions we had regarding the work of this thesis. Thanks are due to Mr. D. Isles from Scottish Hydro-Electric who supplied the initial data for this thesis. I am also indebted to Dr. D. Nicol, Mr. A. Brennan and Mr. M. Harris from Scottish Power for their time, assistance and useful and penetrating discussions. I would like to thank Dr. J. Blair-Fish for all the help and assistance with the Edinburgh Concurrent Supercomputer. Thanks are also due to Prof. L. Thomas and Dr. K. McKinnon for their constructive criticisms and suggestions. I would like to acknowledge and thank the constant support from Prof. E. Fernandes. Thanks are also due to L. Barroca who read this thesis. I would like to acknowledge my sponsors throughout these years: Universidade do Minho, The British Council and Junta Nacional de Investigação Científica e Tecnológica.

I also would like to acknowledge the "contributions" of my friends through those simple but so important unspoken words that helped me keep going through some difficult moments. In particular, I would like to thank Vangelis Magnissalis who has been a true and good friend during these Glasgow years. To Luis, Dulcinêa, Mário, São, Paulo, Ninucha, Pedro e Joana, I would like to thank for all the support, sacrifices and enthusiasm, but above all the fact of being "my" family. Finally, I want to thank Vera, my mother for all her love, courage and faith.

ABSTRACT

This thesis is concerned with the optimal scheduling of hydro-thermal power generation systems. This problem, usually referred to as the unit commitment and economic dispatch problem, manifests itself as a large scale mixed integer programming problem. In the first instance a linear model is built and solved using branch-and-bound. This approach is, however, very expensive in terms of computational time. Using Lagrangian relaxation the original primal problem may be written in a dual formulation: the problem then admits decomposition into more tractable subproblems. Furthermore, the primal solution can be approximated closely from the dual solution using the duality gap as a termination criterion. A heuristic is used to construct nearly optimal solutions to the primal problem based on the information provided by the dual problem. The decomposition is such as to allow an implementation on a transputer array with significant reductions in the computational time. An investigation into the application of genetic algorithms to power scheduling shows that this approach is feasible although expensive in terms of computational time. Lagrangian relaxation is next used to solve a nonlinear model incorporating the purchasing and selling of electricity. The information provided by the Lagrange multipliers which can be interpreted as shadow prices, is used to determine the best strategy for the purchasing and selling of energy. Nonconvex programming problems such as this may exhibit a duality gap, that is a difference between the optimal solution of the primal and dual problems. An investigation of this problem for power scheduling linked the existence of this gap to the operating constraints of the system.

CONTENTS

LIST OF FIGURES

CHAPTER 6

CHAPTER 7

CHAPTER 8

CHAPTER 9

CHAPTER 1

POWER SYSTEMS SCHEDULING

1.1. Introduction

Electric power generation is a key factor in the economy of every nation. Indeed, decisions concerning energy production can affect the future prosperity of a country. Since the 70's there has been a growing concern about the efficient use of energy as a result of the increasing costs of resources, fuel, labour, capital expenditure, etc: see, for example, Wood and Wollenberg (1984). For the generating companies the aim is to produce electricity at minimum cost; this has been achieved not only by more efficient conversions (e.g. heat into electricity), but also by better management of the scheduling of the units to be operated. A power system can comprise hundreds of different power stations (e.g. thermal, nuclear and hydro) with different running costs, and thousands of transmission lines; this system operates under continuous fluctuations of consumer demand. There have been many management strategies. Indeed, the increase in the complexity of these strategies can be seen as a measure of the important role that power scheduling plays in reducing the overall cost.

The two main decision-making processes [Wood and Wollenberg (1984), Cohen and Sherkat (1987), Tong and Shahidehpour (1989)] in power scheduling are:

- **Unit commitment** - that is which of the generating units are committed (on or off) in every time interval of the scheduling horizon; this decision must take into account system capacity requirements and the economic implications of starting up or shutting down various units.

- Economic dispatch - that is the allocation of the demand of power (system load) to the generating units; such an allocation is determined according to the characteristics of the constituent units.

The objective of a least cost solution requires the simultaneous consideration of these two decisions. Of course, this is one large problem, manifesting itself mathematically as a mixed integer programming problem. However, historically it was regarded as two separate problems, largely because early engineers, prior to Garver (1963), did not know how to solve the combined problem. Throughout this thesis this problem will be referred to as the unit commitment and economic dispatch problems or more correctly as the unit commitment/economic dispatch problem. Although attempts will be made to use the former when referring to early works and the latter when discussing more modern papers it will be necessary on occasions to use these ideas interchangeably.

1.2. General remarks on power systems operation

One of the major difficulties in operating a power system arises from the number of random processes involved. The demand varies considerably over a 24 hour period, but there are also differences over one week, or one year (see Figure 1.1).

Additionally, other stochastic inputs include: the inflows to the hydro units reservoirs, precipitation and more generally weather conditions.

The usual short term horizon ranges from one day to one week. The schedule produced is based on a forecasted demand. However, some reserve must be carried in order to meet discrepancies between real and forecasted demand and the possibility of unit outages or any other difficulties that might arise. In operating the system a short time reserve, referred to as spinning reserve, is

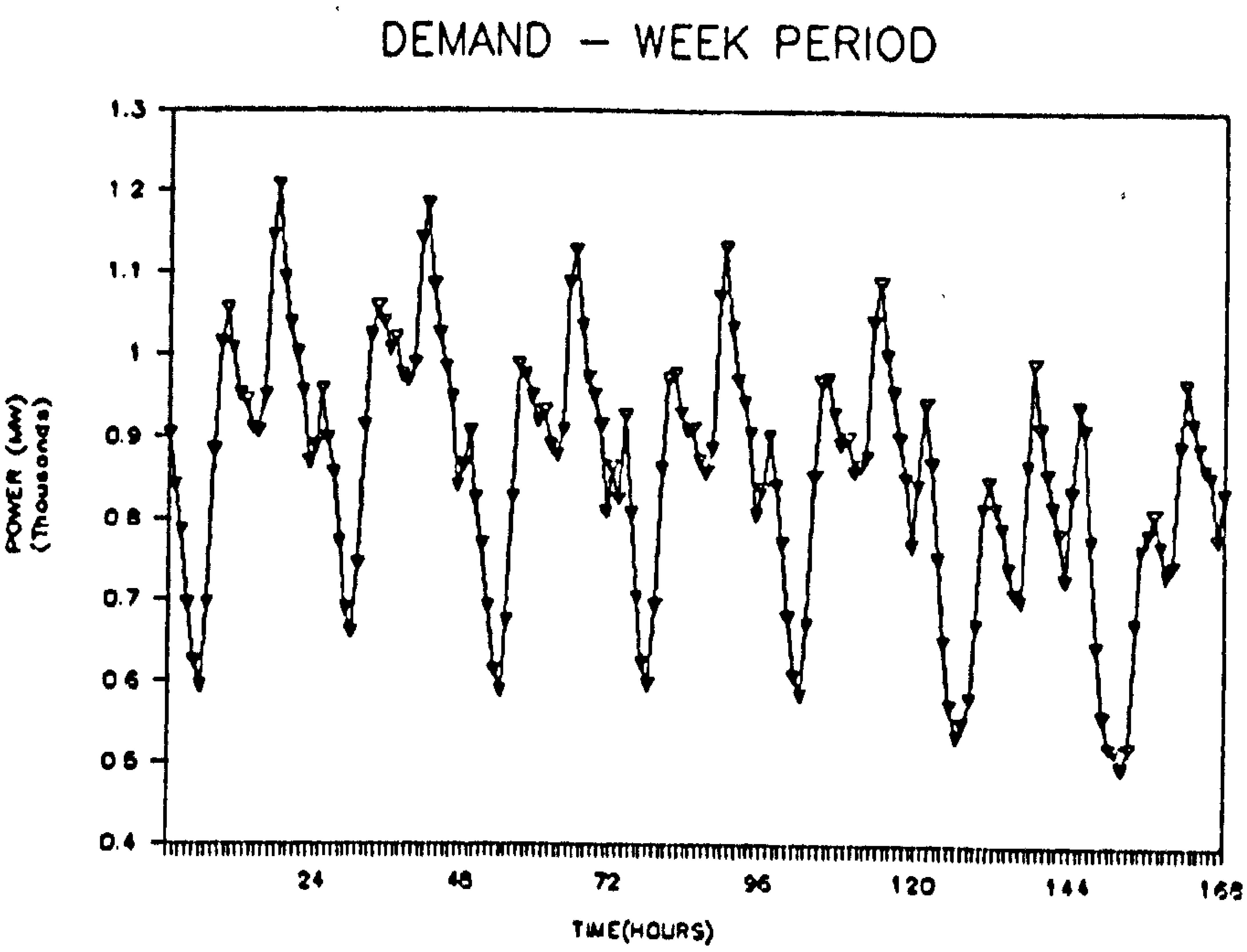


Fig. 1.1. Demand vs time

usually modelled as a deterministic constraint and includes the units already on-line. This spinning reserve must be available to come on-line in a matter of minutes, and in systems with a large hydro component is usually provided by hydro energy. It is sometimes defined as a percentage of the load, or it may be chosen to be just enough to cover the loss of the largest unit in operation. Many systems operate a supplemental reserve requirement specifying the amount of additional reserve that the system must be able to provide in a longer time interval, typically from 1/2 to 1 hour.

The operation of a power system must also take into consideration the maintenance programme of the various units, and this can be a very important constraint on the operation of the system. Some utilities, or power pools (interconnected groups of generating boards), have limits on the power flow between different regions of their system. These limits are usually referred to as transmission or multi-area constraints. In Scotland, the main transmission constraint comes from the interconnector to the south of the border. Associated with transmission there are losses which are related to the distance from the load center. Usually, these loss terms are included in the economic dispatch models through penalty terms.

Recent privatization of the generating boards introduced another important input to the management of power systems, i.e. the possibility of purchases from and sales to the pool system. Many utilities also establish contracts with industries or other utilities, guaranteeing a certain level of power supply at a pre-contracted price.

The size and the complexity of the constraints affecting the operation of the system can make the unit commitment/economic dispatch problem an extremely difficult one to solve. The scheduling of power systems requires fast solution methods, since errors between the forecast and actual data may lead to several re-runs of the scheduling program in one day.

1.3. The generating units

In order to achieve an optimal schedule, the operating characteristics of the different generating units must be taken into account. In short, the generating units may be classified as follows:

- **Nuclear** - based on nuclear technology and usually run continuously at maximum output.

- **Thermal** - operating with oil, coal, or gas, with unlimited fuel supply, and classified as follows: must-run units that are required to be on-line due to operating characteristics and/or economic considerations; cycling units which can be turned on or off, subject to the minimum up and down time constraints; peakers, usually gas turbines, which can be started almost immediately and with no restrictions concerning minimum up and down times.

- **Fuel constrained** - essentially thermal units, where there are, due to contracts, legislation, or scarcity of resources, limited fuel supply in certain time periods.

- Hydro - where potential energy is converted into electricity: they can either be of running type (rivers which do not dry up), pondage (water is stored in a reservoir for reasonably long periods of time), or pump-storage (water can be pumped up to fill a reservoir); it should be noted that the modelling of these units requires a forecast of the influxes.

In addition the scheduling problem is further constrained by other operating conditions imposed on each generating unit, such as:

generating limits - each unit is designed to operate within a feasible region defined by its minimum and maximum capacity;

minimum up time - once a thermal unit is committed, it must be operated for a specified minimum period;

minimum down time - once a thermal unit is decommitted, it cannot be started up until a minimum time interval has elapsed;

ramping limits - the rate of change in power output in any time period is constrained by the characteristics of the particular generating unit.

The thermal units have start-up costs, since fuel is required to heat up the boiler to the appropriate temperature before generation can take place (these costs depend on the number of hours the unit has been down); fixed costs, which include labour and maintenance costs; variable costs, which are the running costs (the main component being essentially fuel consumption); and finally shut-down costs, which result from the fact that decommitment is a gradual process. The running costs are usually modelled by a linear, or piecewise linear function. Some models have also used a polynomial curve (usually second order). The start-up costs are modelled as a fixed cost incurred when the unit is started up, or by a linear or nonlinear function of the number of hours the unit has been shut-down. The shut-down costs are considered fixed and sometimes are, for simplicity, included in the start-up costs. In the case of hydro units the associated costs are very low when compared with thermal units, and are sometimes neglected; indeed, one of the policies is to produce the maximum hydro energy possible in order to minimize the thermal cost. It should be noted that in certain plants with several thermal units there might also be plant crew constraints affecting the number of units that can start-up in a given time period.

Finally, power systems differ so much, that it could be said there are no two alike; they differ not only in the type of generating units available, but also in the different components of hydro, nuclear, or thermal energy generation.

1.4. Overview of solution methods

Earlier attempts to solve the scheduling of power generating units involved dividing the problem up into the unit commitment problem and the economic dispatch problem. The first methods for solving the unit commitment problem were based on heuristic approaches such as "brute force" enumeration or the less inefficient priority listing.

Priority listing involves the ordering of the possible commitment combinations of the available units, based either on the maximum power output just sufficient to meet the load, or by minimizing the marginal cost. For a system with 10 units a priority ordering would reduce the search space in every time period to 11 states which is much more manageable than the possible 2^{10} states required to be examined by enumeration. Having selected those units which should be switched on, the economic dispatch, in the case of thermal units at least, was determined using optimization techniques such as the lambda iteration method, the gradient method or linear programming [Wood and Wollenberg

(1984)]. As a result of the commitment heuristic used, the solutions were often far away from the optimum, not only because of limited search range, but also because factors like start-up and shut-down costs were not considered. Moreover this heuristic method of solving the unit commitment problem was not appropriate for hydro units since it was both difficult to cost the water and to include the constraints affecting the reservoir head.

During the 60's, several methods were developed to solve the unit commitment and economic dispatch problems at one and the same time. Garver (1963) presented a mixed integer model in which binary variables were associated with the commitment states and continuous variables with the economic dispatch. Despite the difficulties arising from excessive computational times and large memory requirements this model opened the way to different and more accurate formulations.

During the same period, a dynamic programming (DP) approach [Bellman and Dreyfus (1962)] was developed by Lowery (1966), possibly the first reported application of DP to power scheduling. In his approach Lowery considered a state as being the power output and the stages as the number of units in the system, the maximum number of stages being equal to the number of units available. There were some shortcomings in this particular application regarding the time dependent start-up costs and minimum up and down times, since each time period was regarded as independent from the previous and the succeeding ones. However, this work showed the applicability of DP in solving the unit commitment/economic dispatch problem, and acted as a catalyst for more sophisticated applications.

Pang and Chen (1976) presented an innovative application of DP to power scheduling by assigning a time period to each stage, and in each stage, the state space consists of all possible combinations of the committed units (Figure 1.2).

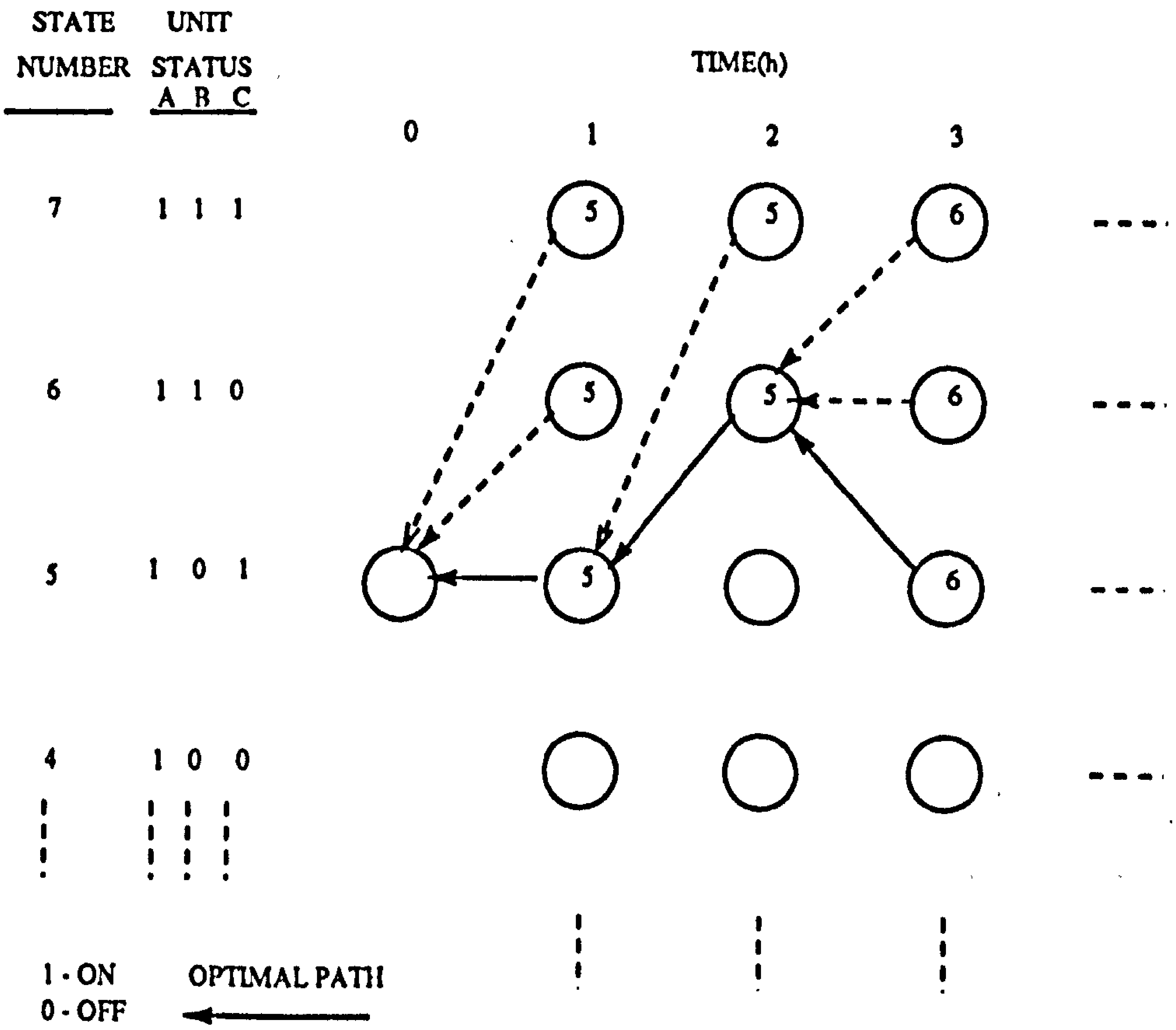


Fig. 1.2. Dynamic programming in power scheduling

This forward approach can incorporate time dependent start-up costs and minimum up and down time restrictions. Notice that as the number of units grows, the number of combinations to be examined rapidly exhausts the computer resources available (for 10 units the number of combinations that may potentially need to be examined in every time period is 2^{10}). Consequently, most of the proposed solutions have opted for methods which try to reduce the search space according to some criteria. Indeed, Pang and Chen (1976) used a priority list of the available units (based on marginal costs, type of units and forced outages) in order to limit the search of the units to be committed.

Pang, Sheble and Albuyeh (1981) compared four unit commitment methods, one based on priority listing and the other three based on DP. Acknowledging that the application of full DP, i.e. considering all the possible combinations, is impracticable due to computer resources, they proposed three DP approaches with limited search range based on heuristics. Dynamic Programming - Sequential Combinations (DP-SC) generates a subset of all the possible combinations by using a priority list sequence to decide the order in which units should be switched on. Dynamic Programming - Truncated Combinations (DP-TC) generates a subset of the possible combinations by selecting a fixed number of the schedulable units to be searched. In Dynamic Programming - Sequential Truncated Combinations (DP-STC) the two previous methods are combined, by first generating some sequential combinations and then a subset of possible units are used to produce the truncated combinations. The results obtained consistently show that the solutions with the best economic dispatches were the ones generated by (DP-TC), which was also the most expensive in terms of computational time. However, as many combinations are neglected in all these three methods, the optimal path may not be found.

Snyder, Powell and Rayburn (1987) presented a DP approach where considerable effort is used to reduce the number of combinations required to be evaluated. This approach for controlling problem size involves the subdivision of the generating units into 'classes', which have similar characteristics with respect to costs, run/down times and capacity. The units within a class are ordered in a priority list. For each class a 'threshold' is defined as the highest priority units which should be committed for a given demand level. Also a 'window' is defined as the next highest priority units below the threshold level which may or may not be committed. As a result the definition of threshold and window are linked to demand levels and, for a given system, this definition has to be tuned. The economic dispatch is solved by a set of linearized economic dispatch routines using piecewise linear cost functions. However, the execution times grow exponentially with the number of units evaluated.

Solution techniques for power scheduling involving specific implementations of DP abound in the literature, e.g. Cohen and Wan (1985), Sherkat *et al.* (1985), Yang and Chen (1989) and Harhammer and Infanger (1990).

Commercial packages for DP do exist which are very efficient in terms of computational time, e.g. Wescougar. However, these are based on limited search approaches which lead to a loss of optimality. Furthermore, there is no measure of how far the solution may be from the optimum.

In systems which involve hydro and thermal units the scheduling problem is usually referred to as hydro-thermal coordination: this is clearly more complex than a pure thermal system. Additional factors come into play such as the coupling of hydro units both electrically and hydraulically. Furthermore, the short-term hydro scheduling has to be consistent with the long-term schedule. This may involve, apart from the consideration of seasonal cycles, some form of evaluating the hydro energy. However, the most frequent approach has been simply to minimize the thermal production cost; the hydro units which are the least expensive to operate are committed first, with the thermal units used to cover the remaining load.

Engles *et al.* (1976) and Larson and Casti (1982) suggested an implementation of DP, decomposing the hydro-thermal coordination problem over different time periods: yearly, weekly and daily optimization. These programs form an hierarchical optimization procedure, in which the optimal long term costs of a higher ordered program are transferred as an input to the next lower time duration program.

For seasonal production planning Sjelvgren *et al.* (1983) used a network flow model of the hydro component, in which the state variables and the control variables are network flows in the arcs; each node represents a unit at a given time period. In McKinnon and Buchanan (1988) stochastic dynamic programming is used to schedule the hydro generation over the long-term period of a year. The results from the long-term schedule (the expected future value of the water as a function of the reservoir levels) are the inputs to the short-term schedule, the objective of which is, to minimize the generation cost over the schedule period.

Some formulations consider hydro generation only and are based on a non-linear objective function incorporating head losses [Bauer et al. (1987), Gangl (1989) and Gutenberger (1989)].

Alternative solution techniques for solving the unit commitment/economic dispatch problem have been based on branch-and-bound [Beale (1988), Fletcher (1987)]. This approach involves solving a sequence of simpler problems derived from the original problem. The search is conducted on a tree of problems, the branch-and-bound tree. Muckstadt and Koenig (1977) developed a pioneering approach to the unit commitment/economic dispatch problem. Based on a mixed integer linear model they implemented a branch-and-bound approach, using a Lagrangian reformulation which permits the problem to be decomposed into single generator problems. The dual problem is created by the inclusion of the demand and reserve constraints into the objective function via two Lagrange multipliers. This function is additive separable and this fact became the cornerstone of many subsequent approaches. The single generation problems are solved very efficiently by DP since the number of states is limited by the minimum up and down times. Each node in the tree is characterized by a set of fixed binary variables denoting the commitment states of the different units. The Lagrangian relaxation of the problem at each node is solved to obtain a lower bound on the optimal solution for that problem. If the Lagrangian solution satisfies the reserve requirement, the economic dispatch solution can easily be found by considering the marginal costs of the committed units, thus providing an upper bound. The subgradient method is used to determine the multipliers in the maximization of the dual function, providing a lower bound to the original primal problem. If the difference of the incumbent solution (upper bound) and its lower bound is below a certain value, the node is fathomed. The next variable to branch to is chosen so as to minimize the violation of the reserve constraint.

Lauer *et al.* (1982) formulate the scheduling problem as a mixed integer nonlinear model, and follow the branch-and-bound approach of Muckstad and Koenig (1977). Since the dual function is not differentiable, a sequence of smooth functions are generated which provide increasingly more accurate approximations to the non-differentiable dual function. The resulting smooth functions are then maximized by standard optimization techniques. The solutions of the approximate problems converge to the solution of the exact problem. An upper bound is generated by modification of the dual solution to feasibility, i.e. schedules that satisfy unit, demand, reserve and branch-and-bound constraints.

Cohen and Yoshimura (1983) formulate the problem as a nonlinear mixed integer programming problem. In order to reduce the search of feasible states, a restriction is imposed that each cycling unit can only be started or stopped once a day. Defining start and stop intervals for each unit, with the top node the entire period of 24 hours. The successors are disjoint partitions of these intervals till, at the bottom, each interval consists of a single hour. An economic dispatch algorithm provides the lower bounds to each node. Each day is considered separately, and so a heuristic has to be used to couple schedules of several days.

One of the main obstacles in power systems scheduling is the size of the problem. Some decomposition approaches have successfully been implemented. Benders' decomposition approach [Benders (1962), Geoffrion (1972), Lasdon (1970)] decomposes the problem into a master problem coordinating a subproblem. In power scheduling the master problem involves only the discrete variables of the unit commitment problem, and the subproblem involves only the continuous generation variables of the economic dispatch problem. Bender's 'cut' is generated from the dual values of the subproblem, constraining the allowed commitments in the master problem. These dual values are associated with the coupling constraints between the continuous and integer variables. Muckstad and Wilson (1968), Turgeon (1978), van den Bosch and Honderd (1985) present implementations for the thermal case only. Habibollahzadeh and Bubenko (1986) tackle the hydro-thermal coordination problem using a linear model. Baptistella and Geromel (1980) also solve a hydro-thermal problem but use a mixed integer nonlinear model. Benders' approach produces an upper and lower bound estimate of the optimal value, and at each iteration a feasible solution to the original problem.

The most frequent approach to decomposition in power scheduling is the use of Lagrangian relaxation. The unit commitment/economic dispatch problem possesses special features which are particularly suitable for decomposition:

- the cost function is a sum of terms involving the operational cost of each generating unit,

-the coupling constraints, the demand and reserve constraints, are also a sum of terms related to the power outputs of all units in each time period.

The introduction of these constraints in the objective function via two sets of Lagrange multipliers creates a dual problem that is additive separable. Therefore, the dual problem is the maximization of a dual function with respect to the multipliers. The separability of the dual function reduces the dual problem to a series of subproblems, one for each unit subject to their local operating constraints, all of them controlled by a master problem via the Lagrange multipliers. For the solution of the dual problem the following three points need to be addressed:

- the method of solution of the subproblems,
- the generation of the Lagrange multipliers,
- the termination criteria.

The solution methods for the subproblems include DP, the gradient method, linear programming, the choice of method depending on the particular characteristics of the subproblem under examination. In each iteration a new set of multipliers has to be computed, and the most widely used approach is the subgradient method. Other approaches utilise second derivative information [Bertsekas *et al.* (1983) and Aoki *et al.* (1987)]. For nonconvex programming, duality theory shows that a duality gap can exist, i.e. the difference between the cost of the optimal primal and dual solutions. Bertsekas and Sandell (1982) showed that the duality gap will tend to zero as the problem size increases. A considerable number of applications based on Lagrangian relaxation have been reported: for thermal systems Bertsekas *et al.* (1983), Merlin and Sandrin (1983), Zhuang and Galiana (1988), Virmani, Imhof and Mukherjee (1989); for hydro-thermal coordination Sandell *et al.* (1982), Shaw and Bersekas (1985), Aoki *et al.* (1987, 1989), Tong and Shahidehpour (1990); Bard (1988), Ruzic and Rajakovic (1991) and Cohen (1991) include ramping constraints in their models. The main advantages of Lagrangian relaxation are that it makes possible the solution of very large problems and allows the bracketing of the optimal primal value between the primal and dual solutions obtained in the iteration process.

There are also reported some semi-rigorous approaches, e.g. Khodaverdian, Brameller and Dunnett (1986), which use heuristics combined with rigorous methods. The main strength of these approaches lies in the consideration of all real world operational constraints, generating suboptimal but feasible schedules. Mokhtari, Singh and Wollenberg (1988) noted that some constraints affecting the unit commitment problem are difficult to formulate mathematically, and when formulated, frequently lead to substantial increases in computational time. Furthermore, the experience of the control operator is critical to the generation of a good schedule. From this background, Mokhtari, Singh and Wollenberg (1988) developed a rule-based expert system for the unit commitment problem, where some of the more complex constraints were not included in the unit commitment solver (based on DP) but were dealt with externally by adjusting the input data. Zhuang and Galiana (1990) use simulated annealing [Kirkpatrick, Gelatt and Vecchi (1983)] to solve the thermal unit commitment/economic dispatch problem. A random unit commitment heuristic generates an initial feasible commitment, on which the Metropolis optimization algorithm [Metropolis *et al.* (1953)] proceeds by generating feasible trial commitment solutions; these are points in the search space which will either be rejected or accepted with probability (Boltzmann distribution) depending on the annealing temperature. The algorithm generates near optimal solutions, coping with plant crew constraints, but it is quite slow.

This overview of solution methods displays both a steady increase in the complexity of the models and the size of the problems which can be tackled: nowadays this can involve several hundred generating units.

CHAPTER 2

THE MATHEMATICAL MODEL

The first model developed was based on the problem presented by Scottish Hydro-Electric (see Appendix 1). The nuclear component of the must-run type need not be considered within the optimal schedule, since it produces a constant output. Therefore, the system contained three main components: thermal units, hydro units and pumped-storage units. As the main operating characteristics of these three components are very different the problem was partitioned accordingly. The following notation is used:

<u>Notation</u>

Subscripts

i A thermal unit. There are I thermal units, i.e. $i = 1, 2, \ldots, I$.

k A hydro unit. There are K hydro units, i.e. $k = 1, 2, \ldots, K$.

l A pump-storage unit. There are L pump-storage units, i.e. $l = 1, 2, \ldots, L$.

Superscripts

t A time interval. There are T time intervals, i.e. $t = 1, 2, \ldots, T$.

Sets

$\mathcal{X}_i$	The power output of the i^{th} thermal unit belongs to this set, $[\underline{x}_i, \bar{x}_i]$, i.e. the operating lower and upper limits of the i^{th} unit.
$\mathcal{V}_k$	The volume of the k^{th} hydro reservoir belongs to this set $[\underline{v}_k, \bar{v}_k]$, i.e. the lower and upper volume (operating) limits of the k^{th} reservoir.
$\mathcal{Y}_k$	The discharge from the k^{th} hydro unit belongs to this set $[\underline{y}_k, \bar{y}_k]$, i.e. the lower and upper discharge limits of the k^{th} unit.
$\mathcal{R}_l$	The volume of the l^{th} pump-storage reservoir belongs to this set $[\underline{r}_l, \bar{r}_l]$, i.e. the lower and upper volume (operating) limits of the l^{th} pump-storage reservoir, where $\underline{r}_l$ is equal to $\underline{r}_{1l}$ for $t = 1, \ldots, T-1$, and $\underline{r}_{2l}$ for $t = T$.
$\mathcal{Q}_l$	The discharge from the l^{th} pump-storage reservoir belongs to this set $[\underline{q}_l, \bar{q}_l]$, i.e. the lower and upper discharge limits of the l^{th} unit.
$\mathcal{P}_l$	The pumping to the l^{th} pump-storage belongs to this set $[\underline{p}_l, \bar{p}_l]$, i.e. the lower and upper pumping limits of the l^{th} unit.

Constants

F_i	The fixed cost for the i^{th} thermal unit (£).
V_i	The cost per megawatt for the i^{th} thermal unit (£/MW).
U_i	The start-up cost for the i^{th} thermal unit (£).
D_i	The shut-down cost for the i^{th} thermal unit (£).
Ξ_i	The minimum down time for the i^{th} thermal unit (h).
Ψ_i	The minimum up time for the i^{th} thermal unit (h).
H_k^0	The value of the equivalent quantity of water used for generating one unit of power in the k^{th} hydro unit (£/MW).
S_k	The value of spillage in terms of the equivalent quantity of water used for generating one unit of power in the k^{th} hydro unit (£/MW).
G_l^0	The value of the equivalent quantity of water in the l^{th} pump-storage unit used for generating one unit of power (£/MW).
P_l^0	The value of the water pumped to the l^{th} pump-storage reservoir, in terms of its equivalence for generating one unit of power (£/MW).
Θ_l	The inverse of the thermodynamic efficiency of the pumping process in the l^{th} pump-storage unit.
d^t	The demand which the system has to meet in every time period (MW).
R	The constant reserve in every time period (MW).

Variables

x_i^t	The power produced by the i^{th} thermal unit (MW).
α_i^t	The commitment (integer) variable of the i^{th} thermal unit.
β_i^t	The start up (integer) variable of the i^{th} thermal unit.
γ_i^t	The shut down (integer) variable of the i^{th} thermal unit.
v_k^t	The volume of the k^{th} reservoir (MWh).
y_k^t	The discharge from the k^{th} hydro unit (MW).
f_k^t	The inflow to the k^{th} hydro unit (MW).
s_k^t	The spillage from the k^{th} hydro unit (MW).
r_l^t	The volume of the l^{th} pump-storage reservoir (MWh).
q_l^t	The discharge from the l^{th} pump-storage unit (MW).
p_l^t	The pumped water to the l^{th} pump-storage unit (MW).
g_l^t	The inflow to the l^{th} pump-storage unit (MW).
μ_l^t	The commitment (integer) variable associated with generation of the l^{th} pump-storage unit.
ν_l^t	The commitment (integer) variable associated with pumping of the l^{th}pump-storage unit.

2.1. The thermal system

Suppose there are I thermal units, and let the power (MW) produced by unit i during period t be x_i^t. The capacity constraint on the unit i results in

$$x_i^t \in \mathcal{X}_i \cup \{0\}, \tag{2.1.1}$$

where

$$\mathcal{X}_i \equiv [\underline{x}_i, \bar{x}_i], \tag{2.1.2}$$

$$\text{for } i = 1,2,\ldots,I, \text{ and for } t = 1,2,\ldots,T.$$

Each unit has an associated integer variable α_i^t which denotes whether the unit is committed (on) or not (off)

$$\alpha_i^t = \begin{cases} 1, & \text{if unit } i \text{ is on during period } t, \\ 0, & \text{otherwise;} \end{cases} \tag{2.1.3}$$

so

$$x_i^t = \begin{cases} 0, & \text{if } \alpha_i^t = 0, \\ x_i \in \mathcal{X}_i, & \text{if } \alpha_i^t = 1. \end{cases} \tag{2.1.4}$$

Note that this implies [Glover (1975)],

$$\underline{x}_i \alpha_i^t \le x_i^t \le \bar{x}_i \alpha_i^t \tag{2.1.5}$$

$$\text{for } i = 1,2,\ldots,I, \text{ and for } t = 1,2,\ldots,T.$$

The costs are:

(a) Running costs per unit

$$F_i \alpha_i^t + V_i x_i^t, \tag{2.1.6}$$

$$\text{for } i = 1,2,\ldots,I, \text{ and for } t = 1,2,\ldots,T,$$

where F_i is the fixed cost, and V_i is a linear approximation to the fuel cost over the range $[\underline{x}_i, \bar{x}_i]$.

(b) Start-up cost per unit

$$U_i \beta_i^t, \tag{2.1.7}$$

for $i = 1, 2, \ldots, I,$ and for $t = 1, 2, \ldots, T,$

where U_i is the start-up cost for unit i, and β_i^t is an integer variable defined as,

$$\beta_i^t = \begin{cases} 1, & \text{if unit } i \text{ is started in period } t, \\ 0, & \text{otherwise.} \end{cases} \tag{2.1.8}$$

Hence

$$\beta_i^t \geq \alpha_i^t - \alpha_i^{t-1}, \tag{2.1.9}$$

for $i = 1, 2, \ldots, I,$ and for $t = 1, 2, \ldots, T.$

(c) Shut-down cost per unit

$$D_i \gamma_i^t, \tag{2.1.10}$$

for $i = 1, 2, \ldots, I,$ and for $t = 1, 2, \ldots, T,$

where D_i is the shut-down cost for unit i, and γ_i^t is an integer variable defined as,

$$\gamma_i^t = \begin{cases} 1, & \text{if unit } i \text{ is shut-down in period } t, \\ 0, & \text{otherwise.} \end{cases} \tag{2.1.11}$$

So

$$\gamma_i^t \geq \alpha_i^{t-1} - \alpha_i^t, \tag{2.1.12}$$

for $i = 1, 2, \ldots, I,$ and for $t = 1, 2, \ldots, T.$

(d) If the minimum down time is Ξ_i for unit i, then necessarily the following restriction holds,

$$\gamma_i^t + \sum_{j=t+1}^{t+\Xi_i-1} \beta_i^j \leq 1, \tag{2.1.13}$$

for $i = 1, 2, \ldots, I,$ and for $t = 1, 2, \ldots, T - \Xi_i,$

where Ξ_i is an integer greater than 1.

(e) If the minimum up time is Ψ_i for unit i, then necessarily the following restriction holds,

$$\beta_i^t + \sum_{j=t+1}^{t+\Psi_i-1} \gamma_i^j \le 1, \tag{2.1.14}$$

for $i = 1, 2, \ldots, I,$ and for $t = 1, 2, \ldots, T - \Psi_i,$

where Ψ_i is an integer greater than 1.

2.2. The hydro system

Suppose there are K conventional hydro units of the "must-run" type, i.e. there must always exist a minimum flow in order to prevent the river from running dry.

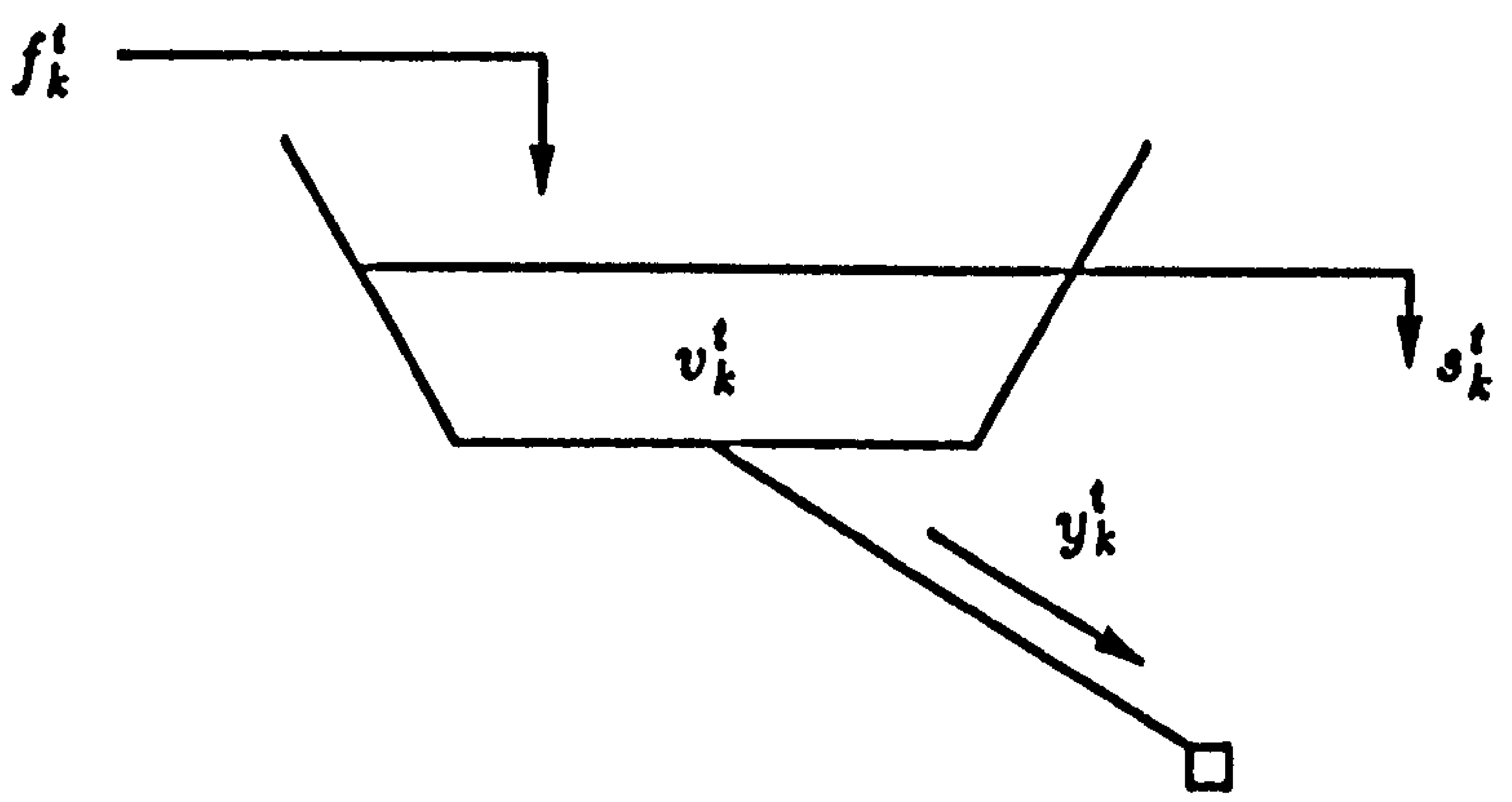


Fig. 2.1. Schematic diagram of a hydro unit

The continuity equation gives the following relation,

$$v_k^t = v_k^{t-1} + f_k^t - y_k^t - s_k^t, \tag{2.2.1}$$

for $k = 1, 2, \ldots, K,$ and for $t = 1, 2, \ldots, T,$

where

v_k^t is the volume of the k^{th}reservoir at the end of period t,

y_k^t is the discharge of the k^{th} hydro plant during period t,

f_k^t is the influx to the k^{th} hydro unit reservoir during period t, and $f_k^t \geq 0$,

s_k^t is the spillage from k^{th}reservoir during period t, and $s_k^t \geq 0$.

The operation of the hydro units must be within the reservoir limits and the operating limits of the turbines; thus

$$\mathcal{V}_k \equiv [\underline{v}_k, \bar{v}_k], \tag{2.2.2}$$

$$v_k^t \in \mathcal{V}_k, \tag{2.2.3}$$

$$\mathcal{Y}_k \equiv [\underline{y}_k, \bar{y}_k], \tag{2.2.4}$$

$$y_k^t \in \mathcal{Y}_k, \tag{2.2.5}$$

for $k = 1, 2, \ldots, K$, and for $t = 1, 2, \ldots, T$.

The operating costs of the hydro units are taken to be linear,

$$H_k^0 y_k^t + S_k s_k^t, \tag{2.2.6}$$

for $k = 1, 2, \ldots, K$, and for $t = 1, 2, \ldots, T$,

where H_k^0 and S_k are, respectively, the value of the discharge and spillage.

2.3. The pump-storage system

Suppose there are L pump-storage plants.

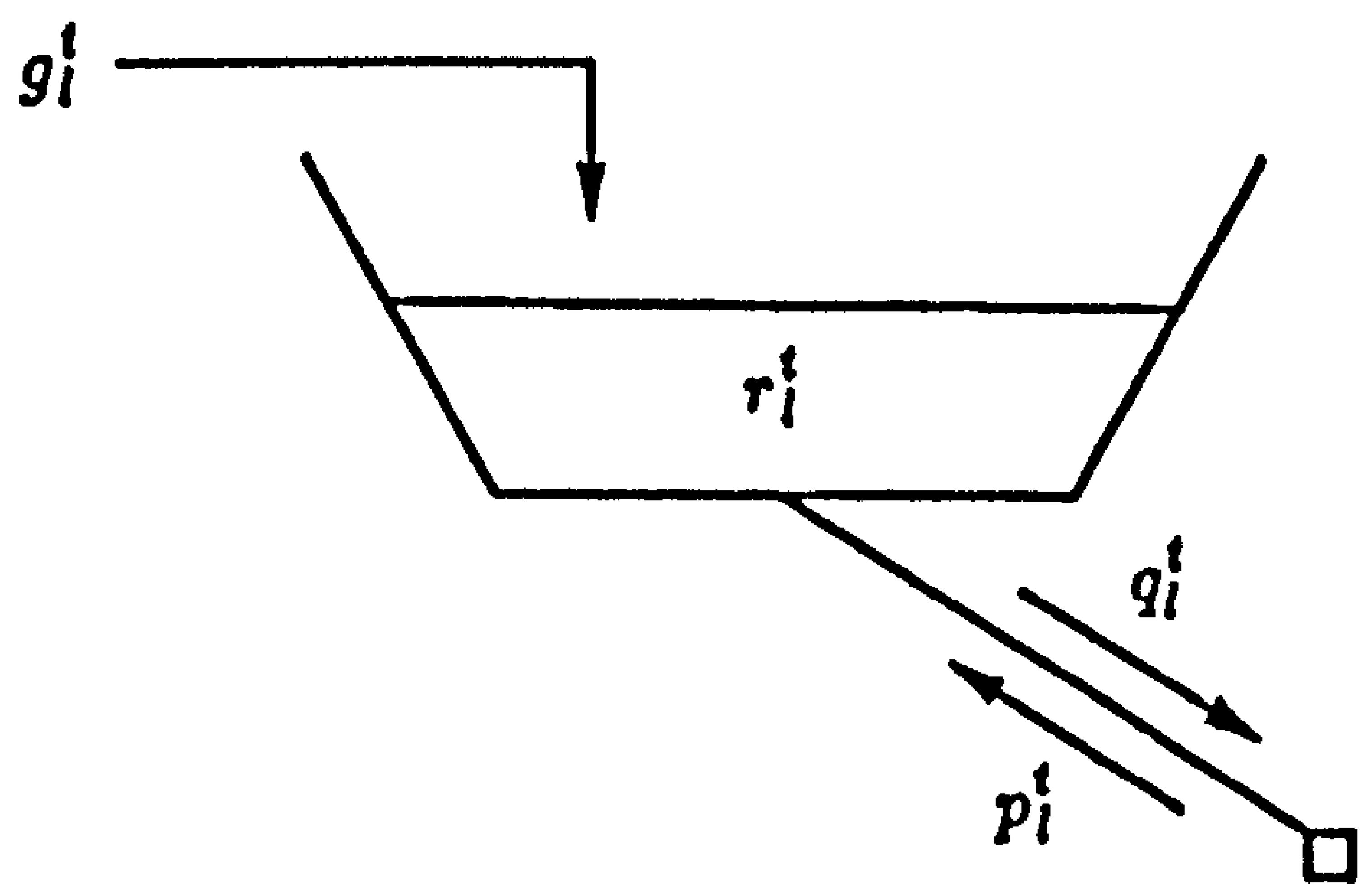


Fig. 2.2. Schematic diagram of a pump-storage unit

From continuity, we get

$$r_l^t = r_l^{t-1} + g_l^t - q_l^t + p_l^t, \tag{2.3.1}$$

$$\text{for } l = 1, 2, \ldots, L, \text{ and for } t = 1, 2 \ldots, T,$$

where

r_l^t is the l^{th} reservoir storage at the end of period t,

q_l^t is the discharge from the l^{th} unit during period t,

p_l^t is the pumped water to the l^{th} reservoir during period t,

g_l^t is the influx to the l^{th} reservoir during period t, and $g_l^t \geq 0$.

The operation of the pumped-storage units must be within the operating limits of the reservoir, turbines and pumps. We introduce the following sets:

$$\mathcal{R}_l \equiv [\underline{r}_l, \bar{r}_l], \tag{2.3.2}$$

$$\mathcal{Q}_l \equiv [\underline{q}_l, \bar{q}_l], \tag{2.3.3}$$

$$\mathcal{P}_l \equiv [\underline{p}_l, \bar{p}_l]. \tag{2.3.4}$$

So that

$$r_l^t \in \mathcal{R}, \tag{2.3.5}$$

$$q_l^t \in \mathcal{Q}_l \cup \{0\}, \tag{2.3.6}$$

$$p_l^t \in \mathcal{P}_l \cup \{0\}, \tag{2.3.7}$$

$$\text{for } l = 1,2,\ldots,L, \text{ and for } t = 1,2,\ldots,T.$$

Due to the design of the plant there cannot be generation and pumping at one and the same time. This can be expressed as

$$q_l^t p_l^t = 0. \tag{2.3.8}$$

Intuitively, it may be thought that (2.3.8) is unnecessary since one might argue that any economic optimization would select one or the other. However, for particular values of the demand and reserve it is easy to construct examples where pumping and generation can occur simultaneously [see Appendix 2]. In order to overcome this nonlinearity, two integer variables are introduced, one associated with generating and the other with pumping.

$$\mu_l^t = \begin{cases} 1, & \text{if unit } l \text{ is generating in period } t, \\ 0, & \text{otherwise.} \end{cases} \tag{2.3.9}$$

$$\nu_l^t = \begin{cases} 1, & \text{if unit } l \text{ is pumping in period } t, \\ 0, & \text{otherwise.} \end{cases} \tag{2.3.10}$$

Thus

$$q_l^t = \begin{cases} 0, & \text{if } \mu_l^t = 0, \\ q_l \in \mathcal{Q}_l, & \text{if } \mu_l^t = 1. \end{cases} \tag{2.3.11}$$

$$p_l^t = \begin{cases} 0, & \text{if } \nu_l^t = 0, \\ p_l \in \mathcal{P}_l, & \text{if } \nu_l^t = 1. \end{cases} \tag{2.3.12}$$

$$\text{for } l = 1,2,\ldots,L, \text{ and for } t = 1,2,\ldots,T.$$

The constraint that there cannot be generation and pumping at the same time may therefore be expressed as,

$$0 \leq \mu_l^t + \nu_l^t \leq 1, \tag{2.3.13}$$

for $l = 1, 2, \ldots, L$, and for $t = 1, 2, \ldots, T$.

The cost of operating the l^{th} pump-storage unit is

$$G_l^0 q_l^t - P_l^0 p_l^t, \tag{2.3.14}$$

for $l = 1, 2, \ldots, L$, and for $t = 1, 2, \ldots, T$,

where G_l^0 is the value of the equivalent quantity of water used for generating each unit of power while P_l^0 is the value of the water pumped to the reservoir.

2.4. The demand and reserve constraints

Clearly, the power output has to satisfy a stochastic demand. Also, if one unit breaks down it is not possible to start an uncommitted thermal unit immediately and so a certain amount of reserve, known as spinning reserve, should be available. There are several reserve policies that have been adopted: a constant reserve in every time period [Muckstadt and Wilson (1968)]; a variable reserve [Muckstadt and Koenig (1977), Merlin and Sandrin (1983)]; a reserve large enough to cover the loss of the largest thermal unit [Turgeon (1978), Baptistella and Geromel (1980)]; or a reserve assessed on the basis of a risk analysis [Muckstadt and Wilson (1968), Turgeon (1978)]. In this study we shall consider satisfying the demand d^t, while allowing just sufficient excess capacity to cover the constant reserve R imposed upon the Scottish Hydro-Electric plc, by the pool system operating in the United Kingdom.

Therefore,

$$\sum_{i=1}^{I} x_i^t + \sum_{k=1}^{K} y_k^t + \sum_{l=1}^{L} (q_l^t - \Theta_l p_l^t) \geq d^t, \tag{2.4.1}$$

$$\sum_{i=1}^{I} \bar{x}_i \alpha_i^t + \sum_{k=1}^{K} \bar{y}_k + \sum_{l=1}^{L} (\bar{q}_l \mu_l^t - \Theta_l p_l^t) \geq d^t + R, \tag{2.4.2}$$

$$\text{for } t = 1, \ldots, T$$

where Θ_l is the inverse of the thermodynamic efficiency of the pumping process.

2.5. The mixed integer linear model

The mixed integer linear programming problem takes the form

$$\min_{\alpha_i^t, x_i^t, \delta_k^t, y_k^t, \mu_l^t, q_l^t, \nu_l^t, p_l^t} \sum_{t=1}^{T} \Big\{ \sum_{i=1}^{I} (U_i \beta_i^t + F_i \alpha_i^t + V_i x_i^t + D_i \gamma_i^t) + \sum_{k=1}^{K} (H_k y_k^t + S_k s_k^t) + \sum_{l=1}^{L} (G_l q_l^t - P_l p_l^t) \Big\}, \tag{2.5.1}$$

subject to

$$x_i^t = \begin{cases} 0, & \text{if } \alpha_i^t = 0, \\ x_i \in \mathcal{X}_i, & \text{if } \alpha_i^t = 1, \end{cases}$$

$$\mathcal{X}_i \equiv [\underline{x}_i, \bar{x}_i],$$

$$\alpha_i^t = \begin{cases} 1, & \text{if unit } i \text{ is on during period } t, \\ 0, & \text{otherwise,} \end{cases}$$

$$\beta_i^t \geq \alpha_i^t - \alpha_i^{t-1},$$

$$\gamma_i^t \geq \alpha_i^{t-1} - \alpha_i^t,$$

$$\gamma_i^t + \sum_{k=t+1}^{t+\Xi_i-1} \beta_i^k \leq 1,$$

for $t = 1, 2, \ldots, T - \Xi_i$, where Ξ_i is an integer greater than 1,

$$\beta_i^t + \sum_{j=t+1}^{t+\Psi_i-1} \gamma_i^j \leq 1,$$

and for $t = 1, 2, \ldots, T - \Psi_i$, where Ψ_i is an integer greater than 1,

$$y_k^t \in \mathcal{Y}_k,$$

$$\mathcal{Y}_k \equiv [\underline{y}_k, \bar{y}_k],$$

$$v_k^t = v_k^{t-1} + f_k^t - y_k^t - s_k^t,$$

$$v_k^t \in \mathcal{V}_k,$$

$$\mathcal{V}_k \equiv [\underline{v}_k, \bar{v}_k],$$

$$f_k^t, s_k^t \geq 0,$$

$$q_l^t = \begin{cases} 0, & \text{if } \mu_l^t = 0, \\ q_l \in \mathcal{Q}_l, & \text{if } \mu_l^t = 1, \end{cases}$$

$$\mathcal{Q}_l \equiv [\underline{q}_l, \bar{q}_l],$$

$$p_l^t = \begin{cases} 0, & \text{if } \nu_l^t = 0, \\ p_l \in \mathcal{P}_l, & \text{if } \nu_l^t = 1, \end{cases}$$

$$\mathcal{P}_l \equiv [\underline{p}_l, \bar{p}_l],$$

$$\mu_l^t = \begin{cases} 1, & \text{if unit } l \text{ is generating in period } t, \\ 0, & \text{otherwise,} \end{cases}$$

$$\nu_l^t = \begin{cases} 1, & \text{if unit } l \text{ is pumping in period } t, \\ 0, & \text{otherwise,} \end{cases}$$

$$r_l^t = r_l^{t-1} + g_l^t - q_l^t + p_l^t,$$

$$\mathcal{R}_l \equiv [\underline{r}_l, \bar{r}_l],$$

$$g_l^t \geq 0,$$

$$0 \leq \mu_l^t + \nu_l^t \leq 1,$$

$$\sum_{i=1}^{I} x_i^t + \sum_{k=1}^{K} y_k^t + \sum_{l=1}^{L} (q_l^t - \Theta_l p_l^t) \geq d^t,$$

$$\sum_{i=1}^{I} \bar{x}_i \alpha_i^t + \sum_{k=1}^{K} \bar{y}_k + \sum_{l=1}^{L} (\bar{q}_l \mu_l^t - \Theta_l p_l^t) \geq d^t + R,$$

where d^t, f_k^t, g_l^t, v_k^0, r_l^0, α_i^0, β_i^0, Ξ_i, Ψ_i, Θ_l and R are known.

CHAPTER 3

BRANCH-AND-BOUND

In the introduction several techniques were discussed for solving a mixed integer linear programming problem (MILP). In this chapter, the branch-and-bound method [Beale (1988), Fletcher (1987)] is considered. This approach, examines a sequence of simpler linear programming problems derived from the original problem. These constitute the nodes of a tree.

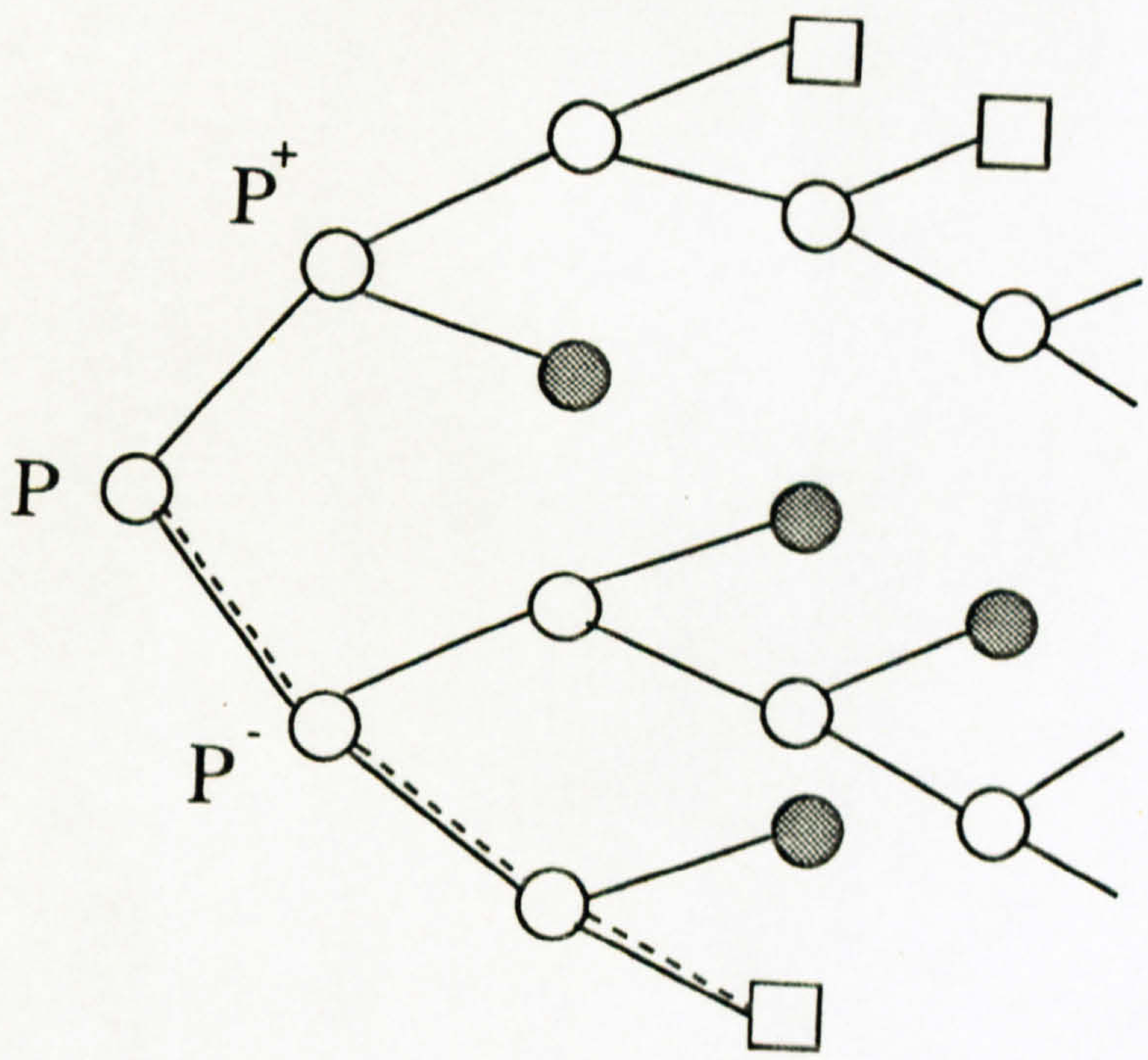


Fig. 3.1. Enumeration tree

The top node in the tree is a relaxation of the original problem (i.e. all integer constraints are replaced by interval constraints) which is relatively easy to solve. The successors of the top node are a set of problems (also relaxations of the original problem) each having a disjoint solution space; the union of the solution spaces of these successors being the solution space of the top node

[Cohen and Sherkat (1987)].

3.1. The branch-and-bound method

To illustrate this technique consider a pure integer programming problem. The aim is to find the solution $\mathbf{x}^*$ of the problem

$$P_I : \min f(\mathbf{x})$$
$$\text{subject to } \mathbf{x} \in \mathcal{E},\ x_i \text{ integer } \forall i \in \mathcal{I}, \tag{3.1.1}$$

where $\mathbf{x}$ is a vector $(x_1, x_2, \ldots, x_n)$, $\mathcal{I}$ is the set of non-negative integer variables and $\mathcal{E}$ is the (closed) feasible region of the continuous problem

$$P : \min f(\mathbf{x})$$
$$\text{subject to } \mathbf{x} \in \mathcal{E}. \tag{3.1.2}$$

Let the minimizer $\mathbf{x}'$ of P exist: if it is feasible in P_I then it solves P_I. If not, then there exists an $i \in \mathcal{I}$ for which x_i' is not integer. In this case two problems can be defined by branching on variable x_i' in problem P, giving

$$P^- : \min f(\mathbf{x})$$
$$\text{subject to } \mathbf{x} \in \mathcal{E},\ x_i \leq [x_i'], \tag{3.1.3}$$

sometimes defined as the **down problem**, and

$$P^+ : \min f(\mathbf{x})$$
$$\text{subject to } \mathbf{x} \in \mathcal{E},\ x_i \geq [x_i'] + 1, \tag{3.1.4}$$

sometimes referred to as the **up problem**, where $[x]$ denotes the largest integer not greater than x. With each of the continuous problems defined above, it is possible to define P_I^- and P_I^+, respectively, with the integrality constraints $x_i, i \in \mathcal{I}$.

It is important to note that $\mathbf{x}^*$ is feasible in either P^- or P^+ but not both, in which case it solves P_I^- or P_I^+. Also any feasible point in P_I^- or P_I^+ is feasible in P_I.

It is usually possible to repeat the branching process by branching on P^- and P^+, and again on the resulting problems, so as to generate a tree structure.

Each node corresponds to a continuous optimization problem, the root is problem P, and the nodes are connected by branches defined above. As can be seen from Figure 3.1, there are two cases in which no branching is possible at any given node:

- when the solution is integer feasible (a square in the tree in Figure 3.1),

- and when the problem has no feasible point (a dark circle in the tree in Figure 3.1).

Otherwise each node is a parent problem (a circle in a tree in Figure 3.1) and gives rise to two branched problems.

If the feasible region is bounded then the tree is finite and each path through the tree terminates in either a square or a circle. Assuming that the solution $\mathbf{x}^*$ of P_I exists, then it is feasible among just one path through the tree. The solution of every problem (square in the tree in Figure 3.1) is feasible in P_I and so the required solution vector $\mathbf{x}^*$ is that solution of a problem (square in the tree in Figure 3.1) whose objective function takes the least value. Often in the tree there are nodes (circle in the tree in Figure 3.1) whose solution violates more than one integrality constraint, so in fact the tree is not uniquely defined until the branching variable is defined. The number of nodes grow exponentially with the number of variables.

The branch-and-bound method attempts to find the solution of P_I by making only a partial search of the tree. This is achieved by calculating upper and lower bounds on the objective function in order to accelerate the fathoming process and thereby curtail the enumeration. If, for instance, at any point in the tree, one solution is obtained for which the objective function value f_j associated with that particular node j is greater than a known feasible solution to the original problem, i.e. $f_j \geq f_i$, (where f_i is a solution at node i), then all possible solutions corresponding to descendants of node j cannot reduce f_j and therefore are non optimal. Hence, these branches need not be examined.

Trees can grow very large, and so to produce an effective partial search of the tree requires defining

a) the problems in the tree,

b) which to solve next,

c) which variable to branch.

The branch-and-bound tree is completely described by the definition of the parent node problem and the method of obtaining the successors of any node. The choice of problem to solve next determines the next active node. Basically there are two strategies: depth-first search, or last in, first out, where a path is followed deep in the tree till a feasible solution is found. Then the algorithm works back rejecting problems or creating subtrees and/or updating the best integer solution. In breadth-first search, whenever two new branches are considered, each one is assigned an estimate of the decrease in the value of the objective function; the next problem to be solved is the one with the lowest assigned value.

Choosing which variable to branch is decided through the use of penalties; these are estimates, e_i^+, e_i^-, of the increase in the objective function which result from consideration of the newly added constraints, the up problem $x_i \geq [x_i] + 1$ and the down problem $x_i \leq [x_i]$, respectively. The rule is to choose the variable which minimizes the increase in the value of the objective function and place the one corresponding to the other branch in the stack of unsolved problems. By following this approach, the worst case is stored with the anticipation that it will be rejected at a later stage.

3.2. Sciconic/VM package

The Sciconic/VM package was used to solve the problem by the branch-and-bound method. The package demands, in the first stage, the formulation of the mathematical programming problem in a specific language - MGG. This is a program generator which produces a FORTRAN matrix generator from a formulation written in a mathematical programming language [Beale (1984), Van Roy and Wolsey (1987)] and from the data of the problem. Also a report writer program is generated to produce the output reports enabling an interpretation of the solution obtained.

In the second stage, the Sciconic/VM package is run, and an optimal solution is eventually obtained. This package [Sciconic/VM (1981)]can be used for linear, integer and certain non-linear problems. With regard to the branch-and-bound method, the package allows the user to define the strategy of searching the tree. It is possible to specify the kind of search, whether depth-first or breadth-first, the priorities associated with vectors, and the criteria by which the direction of branching is decided.

3.3. Results

The model has been tested under several different operating conditions in order to evaluate its performance. From the beginning the aim was to reduce to a minimum the use of "rules of thumb" (e.g. minimum up time, pumping during weekends, etc.) and to determine from the mathematics which conclusions should be inferred, so that these "rules of thumb" may be discarded or, possibly, validated.

The results obtained are summarized in Tables 3-1 and 3-2. Table 3-1 presents the data for two to seven days of forward planning; the integer solution in £; the number of solutions (N.S.) found in the branch-and-bound tree; the corresponding number of branches; the continuous solution in £ (Primal, which in this case is a lower bound to the problem); the percentage difference between the integer and continuous solution and the CPU time used. Table 3-2 presents the data for seven days of forward planning for different operating conditions under the restriction that the reservoir at the beginning and end of the planning period must be at a specified level (i.e. 3500 MW of equivalent water):

(1) Refilling each day up to a constant intermediate level (i.e. 3500 MW of equivalent water);

(2) Pumping during the weekend only;

(3) Pumping whenever required at any time during the week;

(4) Continuous pumping at 200 MW with generation restricted to the limits 80-300 MW.

Table 3-1
Numerical Results - 2 to 7 days

Days	Integer†	N.S.	Branches	Primal†	%Diff.	CPU
2	$3.386E5$	1	137	$3.386E5$	0.0	0 : 09 : 48
3	$4.938E5$	1	205	$4.938E5$	0.0	0 : 07 : 32
4	$6.486E5$	1	224	$6.486E5$	0.0	0 : 17 : 18
5	$8.002E5$	3	728	$7.992E5$	0.1	4 : 28 : 00
6	$9.282E5$	1	726	$9.255E5$	0.3	2 : 50 : 00
7	$1.059E6$	1	783	$1.050E6$	0.8	4 : 40 : 00

†(Solution in £)

Table 3-2
Numerical Results - 7 day period

	Integer†	N.S.	Branches	Primal†	%Diff.	CPU
(1)	1.059E6	1	1417	1.050E6	0.8	4 : 40 : 00
(2)	1.059E6	1	1223	1.050E6	0.8	4 : 40 : 00
(3)	1.059E6	1	783	1.050E6	0.8	4 : 40 : 00
(4)	1.063E6	1	946	1.050E6	1.2	4 : 40 : 00

†(Solution in £)

Figures 3.2, 3.3, 3.4, 3.5 present a summary in graphical form of the unit commitment (U. C.) and economic dispatch (E. D.) for the different operating conditions that have been tested. The top graph displays the U. C. for the several generating units, and the bottom one the E. D. with the demand on the system.

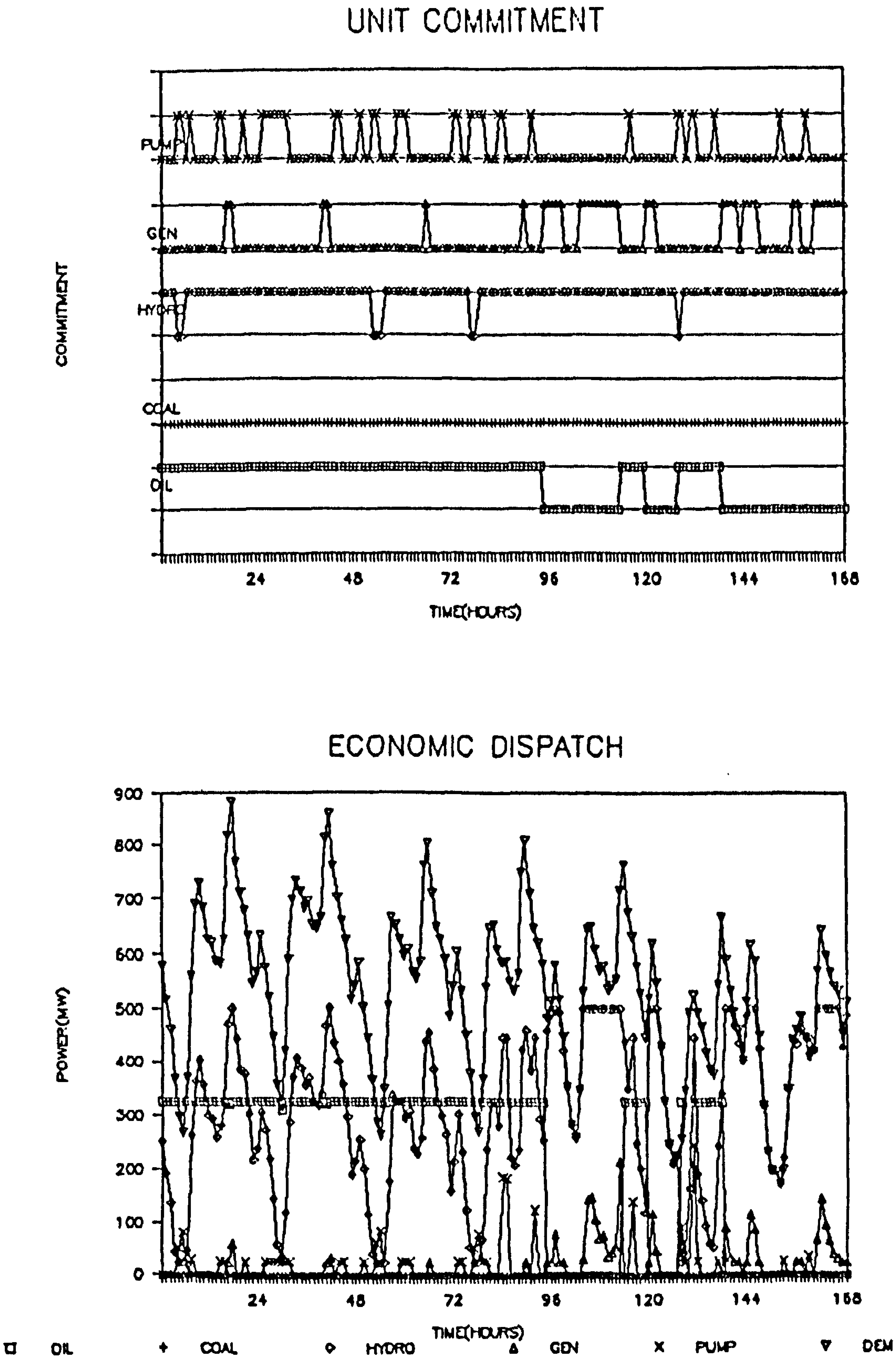
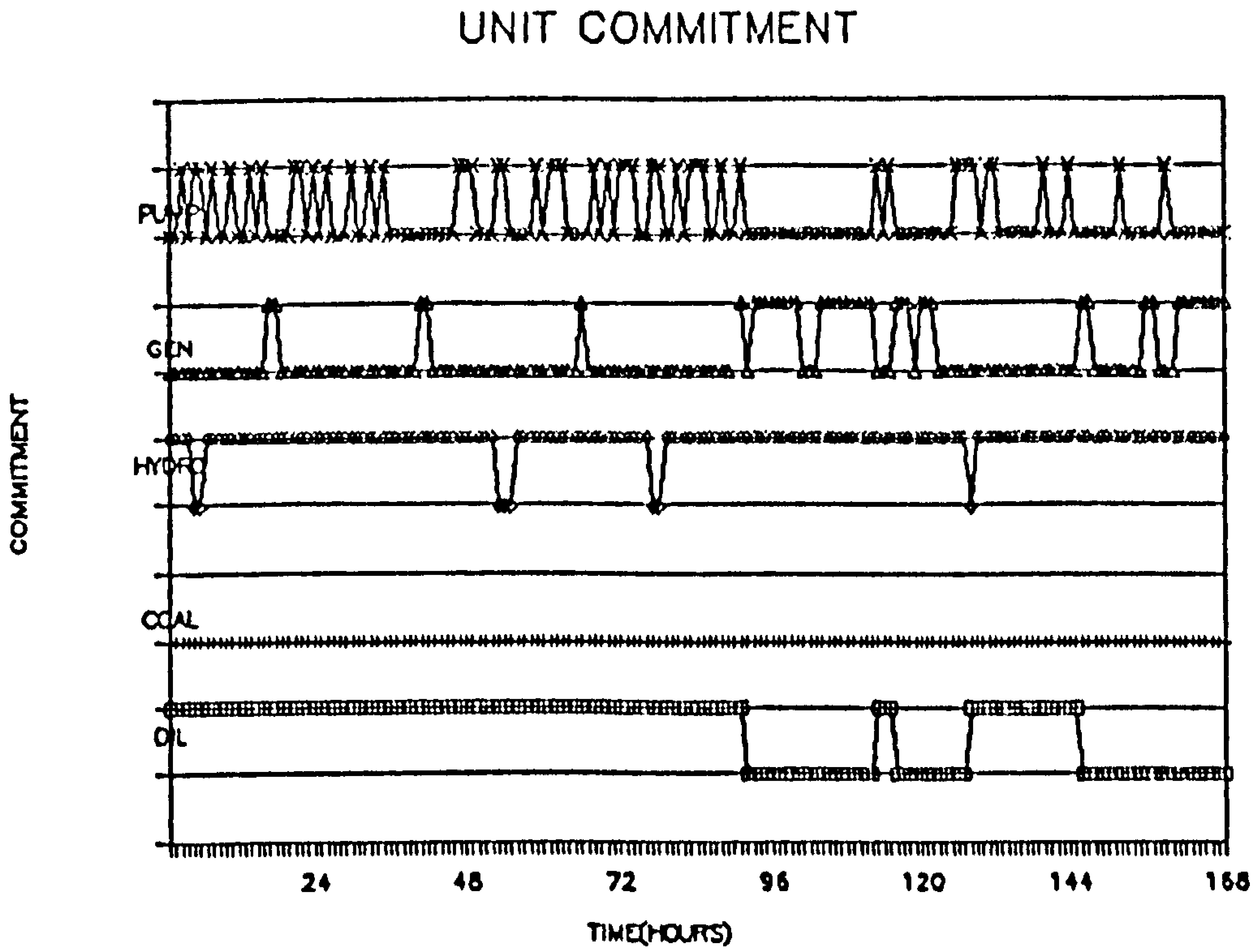


Fig. 3.2. Normal weekly operation



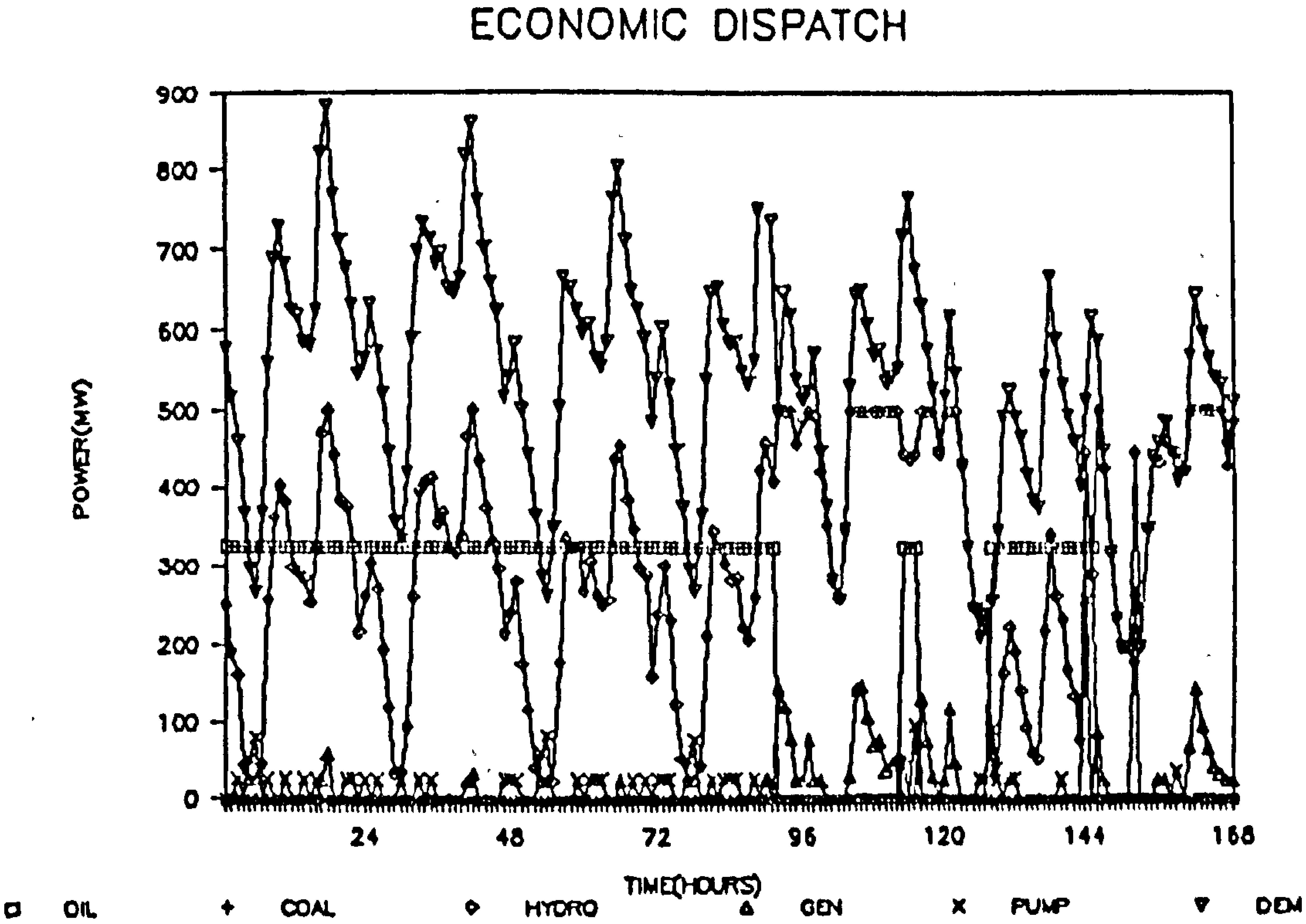


Fig. 3.3. Everyday refilling

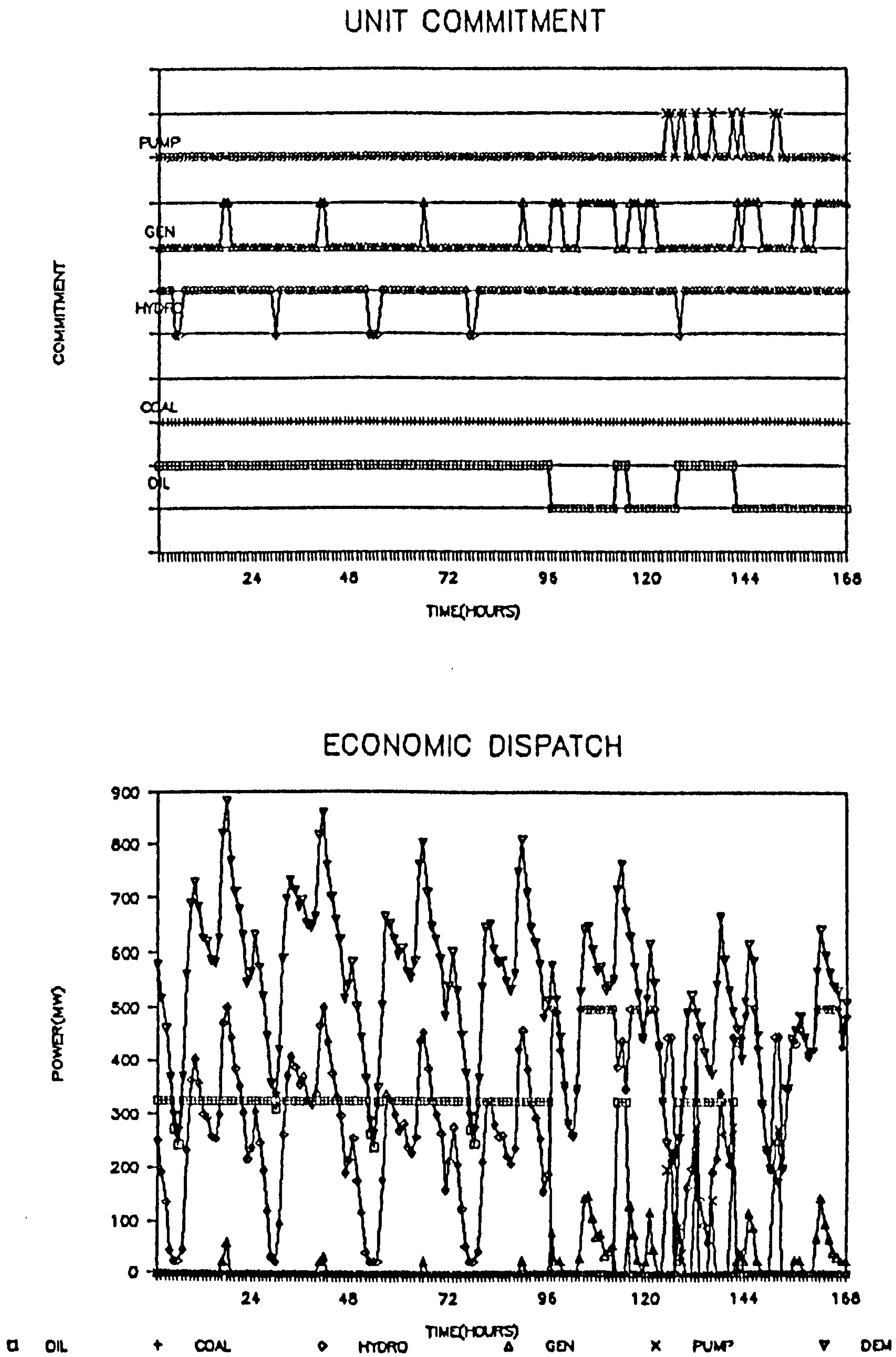


Fig. 3.4. Refilling during weekend

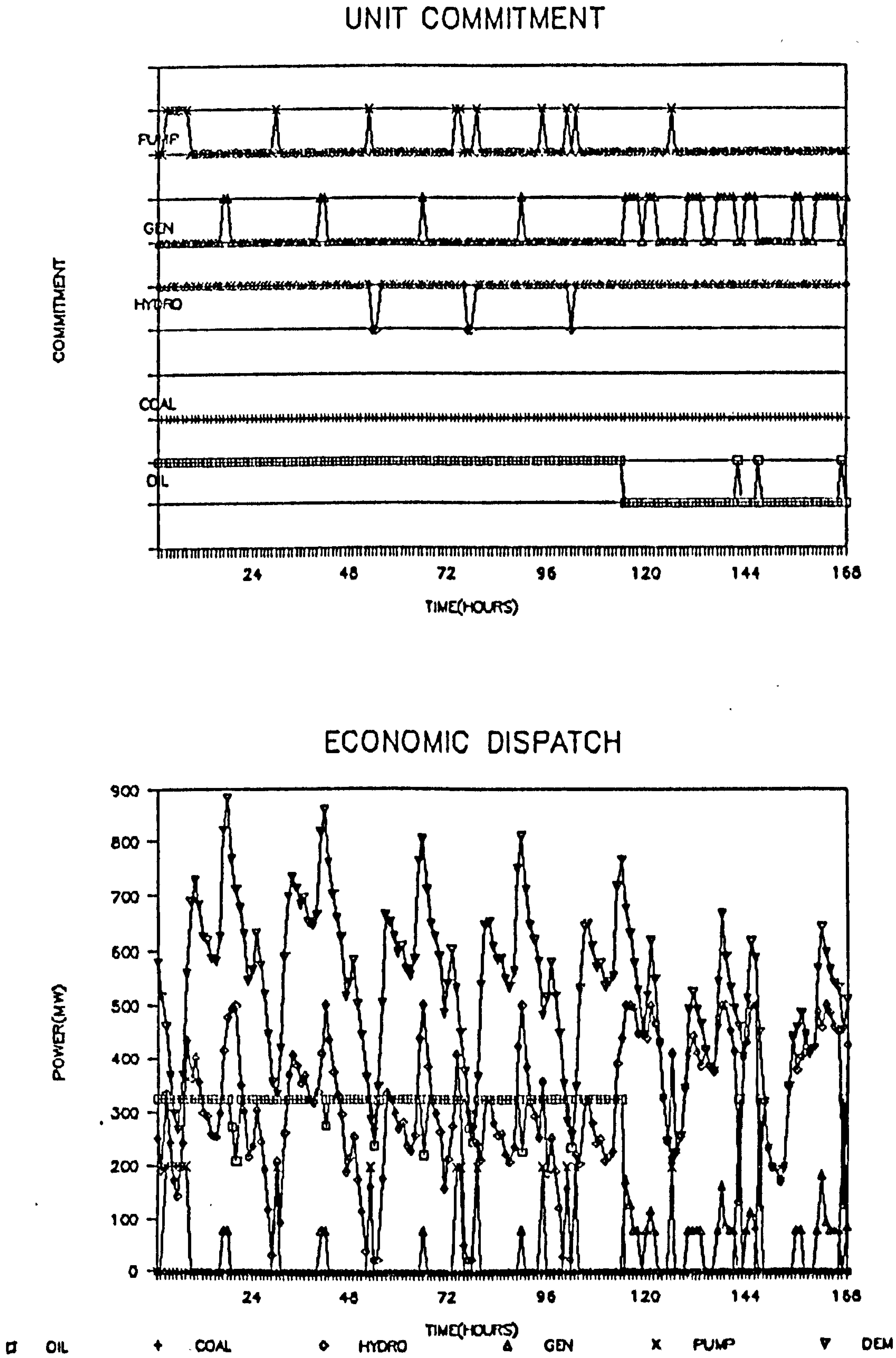


Fig. 3.5. Weekly operation with generation 80-300 MW and pumping 200 MW

3.4. Summary

Extensive runs have been made under various demands, different climatic conditions and a variety of operating conditions from which we are able to draw a number of conclusions.

Several runs have been made for higher levels of the demand and for higher oil prices as well as for drier conditions, when it was observed that the coal unit needs to be switched on. The level of the pump-storage unit reservoir may vary between the lower and upper acceptable limits, but at the beginning of the next planning period it has to be greater than or equal to a given level. Figures 3.2, 3.3 and 3.4 present different policies for the operation of the pump-storage unit. From the resulting costs obtained, which do not differ significantly, the most economic operation occurs when we impose pumping at any time during the week which allows the level to vary between the operating limits of the reservoir. It was found that there was a markedly different cost between the three operating policies for higher levels of the demand. Figure 3.5 presents the weekly operation with the pump-storage unit generating between the limits 80-300 MW and constant pumping at 200 MW, which is close to the actual policy of the Scottish Hydro-Electric plc. As a result, the solution space is greatly reduced, and so it is much more difficult to find a solution close to the optimum. Thus, the operating cost is significantly increased.

The results that have been derived show that it is not possible to carry out a full search of the branch-and-bound trees in reasonable time. However, for a seven day schedule, a solution within 2% of the optimum can be obtained in less than 1 to 2 hours of CPU time on a VAX 11/785. Nevertheless, the underlying matrix (Figure 3.6) points to a suitable structure for the implementation of Benders' decomposition technique [Benders (1962)] and Lagrangian relaxation [Muckstadt and Koenig (1977)] which should reduce greatly the computation times [Habibollahzadeh and Bubenko (1986), Merlin and Sandrin (1983)].

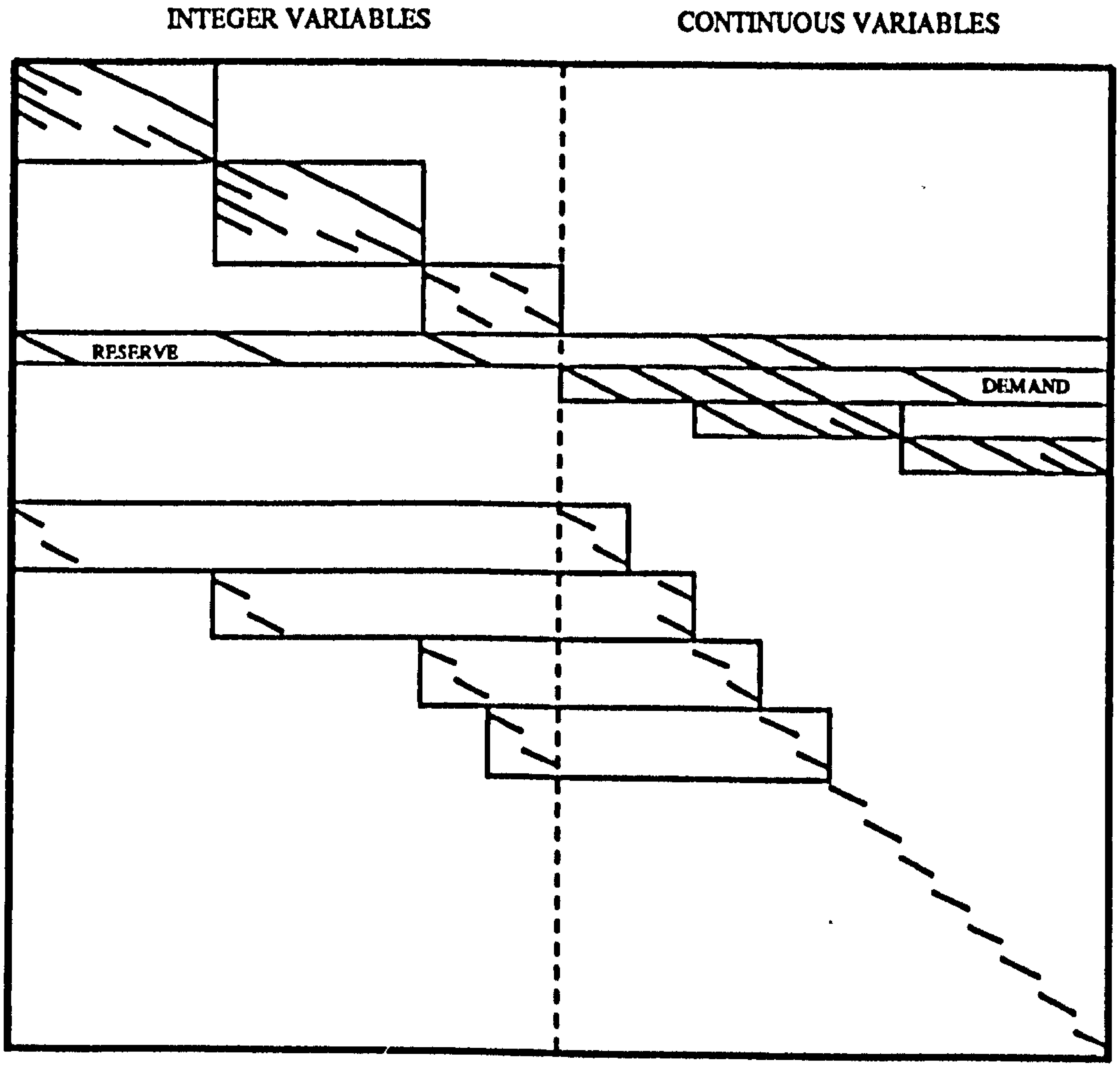


Fig. 3.6. Matrix structure

CHAPTER 4

DUALITY IN MATHEMATICAL PROGRAMMING

4.1. Introduction

The solution of large linear and nonlinear mathematical programming problems by decomposition relies on duality theory. The fundamental results of optimality and duality are presented, based on the concept of a perturbation function [Geoffrion (1971)]. For proofs of theorems and results reference can be made to Lasdon (1968, 1970) and Geoffrion (1971, 1974).

4.2. The primal and dual problems

Consider the following primal problem (P)

$$\min_{x \in X} f(x)$$
$$\text{s.t. } g(x) \leq 0$$

where $g(x) = (g_1(x), g_2(x), \ldots, g_m(x))'$ and f and g_i are real valued functions defined on $X \subseteq R^n$. X is a nonempty convex set on which all functions are convex.

The dual (D) of the primal problem (P) with respect to the g-constraints is defined by

$$\max_{\lambda \geq 0} \left[\inf_{x \in X} \{ f(x) + \lambda' g(x) \} \right]$$

where λ is an m-vector of dual variables. The Lagrangian function is defined as $f + \lambda' g$. The maximand of (D) is a concave function of λ alone for it is the pointwise infimum of a collection of functions linear in λ.

4.3. Definitions and fundamental theorems

Optimality

A pair $(\mathbf{x}, \lambda)$ is said to satisfy the optimality conditions for (P) if

i) $\mathbf{x}$ minimizes $f + \lambda' g$ over X,

ii) $\lambda' g(\mathbf{x}) = 0$,

iii) $\lambda \geq 0$,

iv) $g(\mathbf{x}) \leq 0$.

A vector λ is said to be an optimal multiplier vector for (P) if $(\mathbf{x}, \lambda)$ satisfies the optimality conditions for some $\mathbf{x}$.

Constrained Saddle Point

A pair $(\mathbf{x}^*, \lambda^*)$ is a constrained saddle point of the Lagrangian function $f + \lambda' g$ satisfying the optimality conditions if, and only if, $\lambda^* \geq 0$, $\mathbf{x}^* \in X$ and

$$f(\mathbf{x}^*) + \lambda' g(\mathbf{x}^*) \leq f(\mathbf{x}^*) + \lambda^{*\prime} g(\mathbf{x}^*) \leq f(\mathbf{x}) + \lambda^{*\prime} g(\mathbf{x}).$$

Since $\mathbf{x}^*$ solves the primal problem, one way to find multipliers which cause a solution of the Lagrangian function to solve the primal problem is to locate a saddle point for the Lagrangian function. Another way of stating the optimality conditions is: $(\mathbf{x}^*, \lambda^*)$ satisfies the optimality conditions if and only if $\mathbf{x}^*$ is optimal in (P) and λ^* is optimal in (D), and the optimal values of (P) and (D) are equal.

Perturbation function

The perturbation function $\Phi(\mathbf{y})$ associated with (P) is defined on R^m as

$$\Phi(\mathbf{y}) = \inf_{\mathbf{x} \in X} \{f(\mathbf{x}) \text{ subject to } \mathbf{g}(\mathbf{x}) \leq \mathbf{y}\}$$

where $\mathbf{y}$ is called a perturbation vector.

It is possible to show that $\Phi(\mathbf{y})$ is a convex function [Geoffrion (1971)]. The perturbation function defines a family of problems. Clearly, $\Phi(\mathbf{0})$ is the optimal value of (P), but the study of this function at points other than the origin may be of interest in terms of sensitivity analysis.

Subgradient

Let $\bar{\mathbf{y}}$ be a point at which Φ is finite. An m-vector $\bar{\gamma}$ is said to be a subgradient of Φ at $\bar{\mathbf{y}}$ if

$$\Phi(\mathbf{y}) \geq \Phi(\bar{\mathbf{y}}) + \bar{\gamma}'(\mathbf{y} - \bar{\mathbf{y}}) \text{ for all } \mathbf{y}.$$

The inequality sign is reversed in the case of concave functions. A subgradient can be seen as a generalization of the gradient at nondifferentiable points of nonsmooth functions.

Stability

The problem (P) is said to be stable if $\Phi(\mathbf{0})$ is finite and there exists a scalar $M > 0$ such that

$$\frac{\Phi(\mathbf{0}) - \Phi(\mathbf{y})}{\|\mathbf{y}\|} \leq M \text{ for all } \mathbf{y} \neq \mathbf{0}.$$

The stability property requires that the perturbation function Φ does not decrease infinitely steeply in any direction.

Weak Duality Theorem

If $\overline{\mathbf{x}}$ is feasible in (P) and $\overline{\lambda}$ is feasible in (D), then the objective function of (P) evaluated at $\overline{\mathbf{x}}$ is not less than the objective function of (D) evaluated at $\overline{\lambda}$.

$$\inf_{\mathbf{x}\in X}\{f(\mathbf{x})+\overline{\lambda}'\mathbf{g}(\mathbf{x})\} \leq f(\overline{\mathbf{x}})+\overline{\lambda}'\mathbf{g}(\overline{\mathbf{x}}) \leq f(\overline{\mathbf{x}}).$$

Any feasible solution of (D) provides a lower bound on the value of (P), $v(P)$; and any feasible solution of (P) provides an upper bound on the optimal value of (D), $v(D)$; thus, for given feasible primal and dual points, the associated values bracket the primal optimum, and this can be used as a termination criteria in an iterative algorithm. The equality can only be achieved [Lasdon (1968)] if and only if there exists a saddle point for

$$f(\mathbf{x})+\lambda'\mathbf{g}(\mathbf{x}).$$

This is also a result from the strong duality theorem [Geoffrion (1974)] and precludes the existence of a duality gap, i.e. a difference between the optimal values of the primal and dual problems.

Strong Duality Theorem

If (P) is stable, then

(a) (D) has an optimal solution,

(b) the optimal values of (P) and (D) are equal,

(c) λ^* is an optimal solution of (D) if and only if $-\lambda^*$ is a subgradient of Φ at $\mathbf{y}=\mathbf{0}$,

(d) every optimal solution λ^* of (D) characterizes the set of all optimal solutions (if any) of (P) as the minimizers of $f+\lambda'\mathbf{g}$ over X which also satisfy the feasibility condition $\mathbf{g}(\mathbf{x}) \leq \mathbf{0}$ and the complementary slackness condition $\lambda^{*\prime}\mathbf{g}(\mathbf{x})=0$.

It should be pointed out that this theorem defines the criteria for equality of $v(P)$ and $v(D)$, the optimal values of (P) and (D), in terms of the existence of a subgradient at the origin.

4.4. Related problems

In 1963, Everett studied the relation between the perturbed and unperturbed problems. Let a unique finite optimal solution for any value of $\lambda \geq 0$ be denoted by $x(\lambda)$.

Theorem [Everett (1963)]

If $x(\lambda)$ solves the Lagrangian problem

$$\min\{f(x) + \lambda' g(x)\}$$
$$\text{s.t. } x \in X$$

with $\lambda \geq 0$, then $x(\lambda)$ solves the modified primal problem

$$\min f(x)$$
$$\text{s.t. } g_i(x) \leq y_i \quad i = 1, 2, \ldots, m$$
$$x \in X$$

where

$$y_i = g_i(x(\lambda)) \quad \text{if } \lambda_i > 0$$

$$y_i \geq g_i(x(\lambda)) \quad \text{if } \lambda_i = 0.$$

Let the constraints $g_i \leq 0$ be regarded as expensive resource limitors and λ_i be viewed as prices for these resources. Then the theorem says that any vector which minimizes $f + \lambda' g$ solves a primal problem which uses no more of the valuable resources then the vector itself. In other words, it specifies how much the right hand side may be perturbed without affecting the optimal solution. Thus the multipliers λ convert a constrained to an unconstrained problem (except for the restrictions $\mathbf{x} \in X$).

Theorem [Everett (1963)]
Let λ^1, λ^2 be nonnegative m-vectors with

$$\lambda_k^2 > \lambda_k^1 \qquad \lambda_j^2 = \lambda_j^1, \quad j \neq k.$$

If $\mathbf{x}(\lambda^i)$ solves the Lagrangian problem with $\lambda = \lambda^i$, then

$$g_k(\mathbf{x}(\lambda^2)) \leq g_k(\mathbf{x}(\lambda^1)).$$

Since λ_k has an interpretation as the price of the resource k, if λ_k is increased while all other prices are held fixed, the amount of the k^{th} resource used will decrease.

Theorem [Everett (1963)]
Let $\mathbf{x}^*$ come within $\epsilon > 0$ of solving the Lagrangian problem;

$$f(\mathbf{x}^*) + \lambda' \mathbf{g}(\mathbf{x}^*) \leq f(\mathbf{x}) + \lambda' \mathbf{g}(\mathbf{x}) + \epsilon \quad \text{for all } \mathbf{x} \in X$$

then $\mathbf{x}^*$ comes within ϵ of solving the modified primal problem.

A "good" solution to the Lagrangian problem is thus a "good" solution to the modified primal problem.

4.5. Geometric interpretation of the dual problem

Lasdon (1968, 1970) and Geoffrion (1971) show that the minimization of the Lagrangian problem is equivalent to finding a supporting hyperplane. Some insight is gained from the geometric interpretation, and for simplicity, consider the case with only one inequality, e.g. Bazaraa and Shetty (1979), i.e. $g(\mathbf{x}) \leq 0$. In the (z_1, z_2) plane, consider the set

$$G = \{(z_1, z_2) : z_1 = g(\mathbf{x}), z_2 = f(\mathbf{x}) \text{ for some } \mathbf{x} \in X\}$$

in Figure 4.1. G is the image of X under the map $(g(\mathbf{x}), f(\mathbf{x}))$.

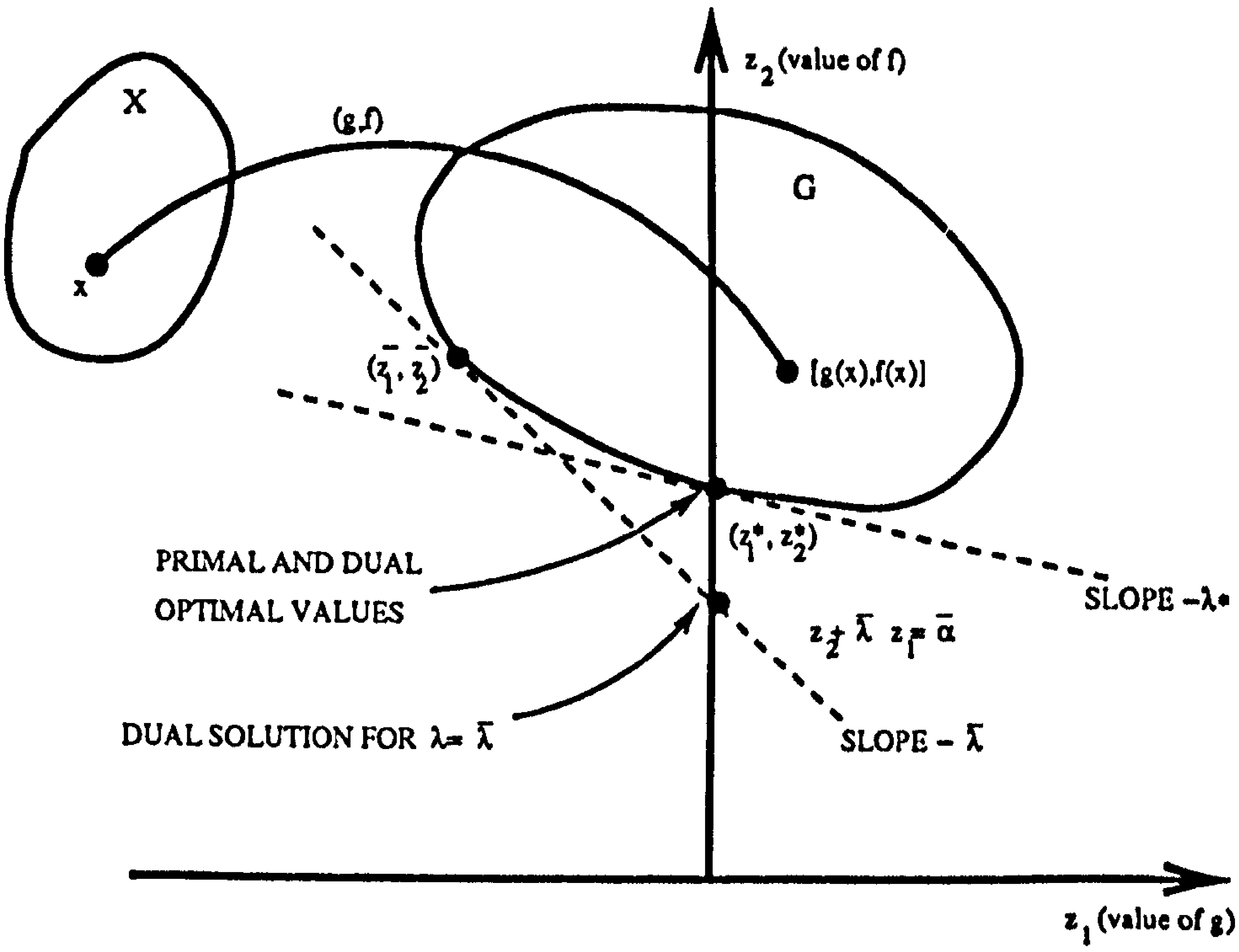


Fig. 4.1. Supporting hyperplanes for the set G.

The value of the dual function is the intercept of the supporting plane with the z_2 axis, and the slope of the plane is $-\lambda$. The value of the primal is the one that minimizes z_2 subject to $z_1 \leq 0$. The optimal solution is then the point (z_1^*, z_2^*).

For a given $\overline{\lambda} \geq 0$, to evaluate the dual function $v(D)$, the minimization of $f + \overline{\lambda} g$ over X is required. Setting $z_1 = g(\mathrm{x})$ and $z_2 = f(\mathrm{x})$ for all $\mathrm{x} \in X$, this is equivalent to minimizing $z_2 + \overline{\lambda} z_1$ over points in G, subject to $(z_1, z_2) \in G$. However, $z_2 + \overline{\lambda} z_1 = \overline{\alpha}$ is an equation of a straight line with slope $-\overline{\lambda}$ and intercept $\overline{\alpha}$ on the z_2 axis. From Figure 4.1 it can be seen that minimizing $z_2 + \overline{\lambda} z_1$ over G corresponds to finding a line with maximum intercept which supports G, i.e. the line tangent to $(\overline{z}_1, \overline{z}_2)$. Therefore, the dual problem is that of finding the support plane with nonpositive slopes having a maximal z_2 intercept, while the primal problem is that of finding a point in G with minimal z_2 intercept, the value of $\Phi(0)$.

If G is convex (Figure 4.1) there exists a supporting hyperplane at all boundary points, and in particular at $z_1 = 0$, i.e. $\Phi(0)$. However, if G is not convex, (the primal is not a convex program), there might not exist a supporting hyperplane at the origin (Figure 4.2). In this case, the values of the primal and dual objective functions are not equal and a duality gap is said to exist.

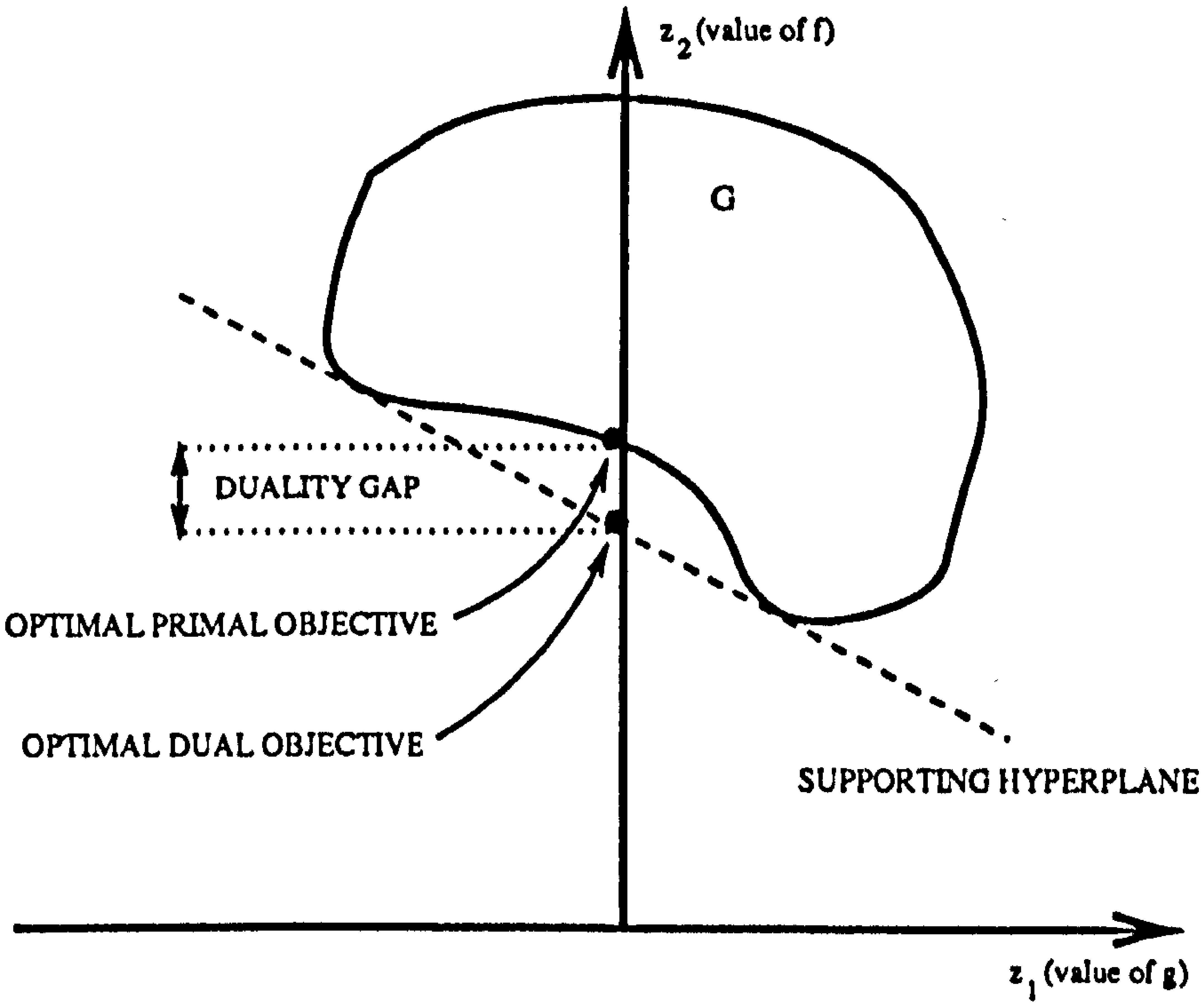


Fig. 4.2. Duality gap.

CHAPTER 5

LAGRANGIAN RELAXATION AND THE UNIT COMMITMENT AND ECONOMIC DISPATCH PROBLEM

5.1. Introduction

The highly combinatorial nature of the Unit Commitment/Economic Dispatch problem has led to a search for more and more efficient methods.

Large Scale Programming is not determined simply by the size of the problem, 'but rather size in conjunction with structure' [Geoffrion (1970a)]. Different structures, for example, multidivisional, combinatorial, dynamic and/or stochastic appear in many problems. The exploitation of these various special structures is one of the main objectives of Large Scale Programming.

An essentially equivalent reformulation of some problems can sometimes make them more tractable. This is done so that the problem can be solved by an existing optimization algorithm taking advantage of the special structure of the original problem. Considered from a hierarchical point of view, the original problem is partitioned into a number of subproblems, sometimes referred to as infimal subproblems, with a master or supremal subproblem at the top level. This partition is dependent upon some parameter specified by the master problem which coordinates the infimal subproblems. The fundamental assumption underlying this multilevel approach [Dirickx and Jennergren (1979)] is that the solution to the original problem can be obtained, or closely approximated by the solutions of the subproblems. Certain problems have a separable structure [e.g. Geoffrion (1970a, 1970b), Luenberger (1989)]

$$\min_{\mathbf{x}} \sum_{i=1}^{q} f_i(\mathbf{x}_i)$$

$$\text{s.t.} \sum_{i=1}^{q} h_i(x_i) = 0$$

$$\sum_{i=1}^{q} g_i(x_i) \leq 0$$

where the components of the n-vector $\mathbf{x}$ are partitioned into q disjoint sets $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_q)$ not necessarily with the same number of components. As Luenberger notes, 'separable problems are ideally suited to dual methods, because the required unconstrained minimization problem decomposes into small subproblems'. Associating the dual variables λ with the equality constraints and $\mu \geq 0$ with the inequality constraints, the dual function becomes

$$\Phi(\lambda, \mu) = \min_{\mathbf{x}} \sum_{i=1}^{q} \{ f_i(\mathbf{x}_i) + \lambda' h_i(\mathbf{x}_i) + \mu' g_i(\mathbf{x}_i) \}$$

which decomposes into q separable problems

$$\min_{\mathbf{x}_i} \{ f_i(\mathbf{x}_i) + \lambda' h_i(\mathbf{x}_i) + \mu' g_i(\mathbf{x}_i) \}$$

which in principle can be solved more efficiently than the original problem.

The Unit Commitment and Economic Dispatch problems have special features which can be exploited in order to construct separable problems. These features [Tong and Shahidiehpour (1989)] can be stated as follows:

1) the commitment variables are the only ones to be restricted to integer values, and from the moment these are fixed, the problem becomes a continuous optimization problem,

2) the constraints can be classified as global or local; the global constraints involve both the demand and reserve coupling all the generating units, while the local constraints state the different operating characteristics of each unit.

The exploitation of the first feature is based on Benders' decomposition [e.g. Benders (1962), Geoffrion (1972)] which allows the mixed integer problem to be partitioned into two subproblems: an integer programming problem and a continuous programming problem. The idea is that, by fixing the integer variables, the problem is reduced to solving a continuous programming problem parameterized by the vector of integer variables.

In Lagrangian relaxation the second feature is used to create a separable problem by relaxing the coupling constraints so that each subproblem involves only one individual unit subject to its own local (operating) restrictions. The local subproblems are parameterized by the Lagrange multipliers and the master problem maximizes a dual function, producing new estimates of the Lagrange multipliers and assuring that the two global constraints are met.

5.2. Lagrangian relaxation in mixed integer linear programs

A minimizing problem (Q) is said to be a relaxation of a minimizing problem (P) if

$$\mathcal{F}(Q) \supseteq \mathcal{F}(P)$$

($\mathcal{F}(.)$ is the set of feasible solutions) and the objective function of (Q), $v(Q)$, is less than or equal to that of (P), $v(P)$, on $\mathcal{F}(P)$. That is,

$$v(Q) \leq v(P)$$

Consider the minimizing problem (P)

$$\begin{aligned} &\min_{\mathbf{x}} \mathbf{cx} \\ &\text{s.t. } A\mathbf{x} \geq \mathbf{b} \\ &\qquad \mathbf{x} \geq 0 \\ &\qquad \mathbf{x}_j \text{ integer}, j \in \mathcal{I} \end{aligned}$$

where x is $n \times 1$, c is $1 \times n$, b is $(m+k) \times 1$, A has conformable dimensions and $\mathcal{I}$ denotes the index set of the variables required to be integer. The minimizing problem (P) is equivalent to

$$\begin{aligned} &\min_{\mathbf{x}} \mathbf{cx} \\ \text{s.t. } &A_1\mathbf{x} \geq \mathbf{b}_1 \\ &A_2\mathbf{x} \geq \mathbf{b}_2 \\ &\mathbf{x} \geq \mathbf{0} \\ &\mathbf{x}_j \text{ integer}, \ j \in \mathcal{I} \end{aligned}$$

with

$$A = \left[A_1 : A_2\right]', \mathbf{b} = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix}$$

where $A_1\mathbf{x} \geq \mathbf{b}_1$ is the set of 'complicating' constraints, $\mathbf{b}_1$ is $m \times 1$ and $\mathbf{b}_2$ is $k \times 1$.

To construct the Lagrangian relaxation formulation, the set of 'complicating' constraints is included in the objective function premultiplied by a nonnegative $m \times 1$ vector λ:

$$\begin{aligned} &\min_{\mathbf{x}} \left\{\mathbf{cx} + \lambda'(\mathbf{b}_1 - A_1\mathbf{x})\right\} \\ \text{s.t. } &A_1\mathbf{x} \geq \mathbf{b}_1 \\ &A_2\mathbf{x} \geq \mathbf{b}_2 \\ &\mathbf{x} \geq \mathbf{0}, \\ &\mathbf{x}_j \text{ integer}, \ j \in \mathcal{I}, \\ &\lambda \geq 0. \end{aligned}$$

Each component of $A_1\mathbf{x} \geq \mathbf{b}_1$ is an inequality; if some of these constraints were prescribed as equalities, then the corresponding components of λ would not be required to be nonnegative. The inclusion of the nonpositive term $\mathbf{b}_1 - A_1\mathbf{x}$

creates a lower bound to the original problem (P). Clearly the optimal value of this problem, for λ fixed at a nonnegative value, is a lower bound on $v(P)$ because only a nonpositive term has been added.

The Lagrangian problem is created by removing the 'difficult/complicating' constraints $A_1\mathbf{x} \geq \mathbf{b}_1$. Then the Lagrangian relaxation (PR_λ) takes the form

$$\min_{\mathbf{x}}\{\mathbf{cx} + \lambda'(\mathbf{b}_1 - A_1\mathbf{x})\}$$

$$\text{s.t. } A_2\mathbf{x} \geq \mathbf{b}_2$$

$$\mathbf{x} \geq 0$$

$$\mathbf{x}_j \text{ integer, } j \in \mathcal{I}$$

$$\lambda \geq 0.$$

Since removing the constraint cannot increase the optimal value, $v(PR_\lambda)$ is also a lower bound on $v(P)$. In principle, for the Lagrangian relaxation to be justified, it must be much simpler to solve than the original problem (P). The potential usefulness of any relaxation of (P), and of (PR_λ) in particular, is largely determined by how near its optimal value is to that of (P). This therefore provides a criterion by which to measure the 'quality' of a particular choice of λ. The ideal choice would be to take λ as an optimal solution to the concave problem (D)

$$\max_{\lambda \geq 0} v(PR_\lambda)$$

which coincides with the Lagrangian dual of (P) with respect to $A_1\mathbf{x} \geq \mathbf{b}_1$. Geoffrion (1974) stresses the fact that the interest in Lagrangian relaxation comes not only from the lower bounds it can provide, but most importantly from the real possibility that it can yield an optimal or near optimal solution to (P). Indeed, Lagrangian relaxation for which the integrality requirement is dropped, produces a tighter lower bound than the linear programming relaxation [e.g.

Geoffrion (1974), Fisher (1981)]. Defining the linear programming relaxation $(\overline{P})$ as

$$\begin{aligned} &\min_{\mathbf{x}} \mathbf{cx} \\ &\text{s.t. } A_1\mathbf{x} \geq \mathbf{b}_1 \\ &\quad A_2\mathbf{x} \geq \mathbf{b}_2 \\ &\quad \mathbf{x} \geq 0 \end{aligned}$$

then

$$v(D) \geq v(\overline{P}).$$

This follows from

$$\begin{aligned} v(D) = &\max_{\lambda}\left\{\min_{\mathbf{x}}\{\mathbf{cx} + \lambda'(\mathbf{b}_1 - A_1\mathbf{x})\}\right\} \\ &\text{s.t. } A_2\mathbf{x} \geq \mathbf{b}_2 \\ &\quad \mathbf{x} \geq 0 \\ &\quad \mathbf{x}_j \text{ integer}, j \in \mathcal{I} \\ &\quad \lambda \geq 0 \end{aligned}$$

$$\begin{aligned} \geq &\max_{\lambda}\left\{\min_{\mathbf{x}}\{\mathbf{cx} + \lambda'(\mathbf{b}_1 - A_1\mathbf{x})\}\right\} \\ &\text{s.t. } A_2\mathbf{x} \geq \mathbf{b}_2 \\ &\quad \mathbf{x} \geq 0 \\ &\quad \lambda \geq 0. \end{aligned}$$

The equality holds when the integrality property [Geoffrion (1970a)] is satisfied, in which case Lagrangian relaxation can do no better than linear programming relaxation. A problem is said to possess the integrality property when the optimal value of (PR_λ) is not altered by dropping the integrality conditions on its variables, i.e. $v(PR_\lambda) = v(\overline{PR}_\lambda)$ for all $\lambda \geq 0$.

Introducing a nonnegative vector μ and using linear programming duality results in

$$\max_{\lambda}\left\{\min_{x}\{cx + \lambda'(b_1 - A_1x)\}\right\} = \max_{\lambda}\max_{\mu}\{\mu' b_2 + \lambda' b_1\}$$

$$\begin{array}{ll} \text{s.t. } A_2x \geq b_2 & \qquad \text{s.t. } \mu' A_2 \leq c - \lambda' A_1 \\ x \geq 0 & \qquad \lambda \geq 0 \\ \lambda \geq 0 & \qquad \mu \geq 0 \end{array}$$

$$= \max_{\lambda,\mu}\{\mu' b_2 + \lambda' b_1\}$$

$$\text{s.t. } \mu' A_2 + \lambda' A_1 \leq c$$

$$\lambda \geq 0$$

$$\mu \geq 0$$

$$= \min_{x} cx$$

$$\text{s.t. } A_1x \geq b_1$$

$$A_2x \geq b_2$$

$$x \geq 0$$

$$= v(\overline{P})$$

using linear programming duality once again.

The implementation of a Lagrangian relaxation to a particular problem must answer certain questions [Fisher (1981, 1985)] and these questions condition the implementation itself. For instance, which restrictions should be relaxed from all those constraining the problem? Clearly the answer to this question is problem specific and the answer must be such as to create a much simpler problem to solve.

The maximization of $v(D)$ with respect to λ implies the successive computation of the multipliers λ and this begs the question as to which method should be chosen to compute those multipliers. There are several methods available like the subgradient method, versions of the simplex method and multiplier adjustment methods. It should be noted that the concave function $v(PR_\lambda)$ is not differentiable. Fisher (1981, 1985) notes the success of the subgradient method on a large number of different applications, pointing out not only its ease of programming but also its robustness. Held, Wolfe and Crowder (1974) describe the method, assessing its computational performance and presenting theoretical convergence properties. Recently, Aoki *et al.* (1987, 1989) proposed a variable metric method for updating the multipliers in the context of power scheduling optimization.

Given the solution to problem (D), the question arises as to how to construct a feasible solution to the original problem (P). Usually the solution obtained from the optimization of (D) will be nearly feasible for (P), and it is possible to construct some kind of heuristic to obtain feasibility. Again the answer to this question is problem specific, and Fisher (1981) reports on several heuristics for different kinds of problems.

In conclusion, a solution to problem (D) provides a lower bound on problem (P), $v(D) \leq v(P)$. Equality holds only when the conditions of the Strong Duality Theorem are satisfied. The difference $v(P) - v(D)$ is referred to as the duality gap, and this may be used to define an interval in which the optimal solution lies. For power scheduling this duality gap has been shown to be strictly positive, and the relative duality gap $[v(P) - v(D)]/v(P)$ decreases as the problem size increases with the number of generating units [Bertsekas *et al.* (1983)].

5.3. Lagrangian relaxation in power scheduling

In applying Lagrangian relaxation the first question to be answered is which constraints should be relaxed. In power scheduling the most suitable constraints are the global constraints, i.e., the demand and reserve constraints. So, the objective function to be minimized is

$$\begin{aligned} \min_{\alpha_i^t, x_i^t, y_k^t, s_k^t \mu_l^t, q_l^t, \nu_l^t, p_l^t} \sum_{t=1}^{T} \Big[\sum_{i=1}^{I} (U_i \beta_i^t + F_i \alpha_i^t + V_i x_i^t + D_i \gamma_i^t) \\ + \sum_{k=1}^{K} (H_k y_k^t + S_k s_k^t) \\ + \sum_{l=1}^{L} (G_l q_l^t - P_l p_l^t) \Big] \end{aligned} \tag{5.3.1}$$

subject to all the local constraints (Chapter 2, Section 2.1., 2.2. and 2.3.) and to the following global constraints

$$\begin{aligned} \sum_{i=1}^{I} x_i^t + \sum_{k=1}^{K} y_k^t + \sum_{l=1}^{L} (q_l^t - \Theta_l p_l^t) \geq d^t \\ \sum_{i=1}^{I} \bar{x}_i \alpha_i^t + \sum_{k=1}^{K} \bar{y}_k + \sum_{l=1}^{L} (\bar{q}_l \mu_l^t - \Theta_l p_l^t) \geq d^t + R \end{aligned} \tag{5.3.2}$$

$$\text{for } t = 1, \ldots, T$$

where Θ_l is the inverse of the thermodynamic efficiency of the pumping process.

The inclusion of these two constraints gives the following Lagrangian problem

$$\Phi(\lambda_1,\lambda_2) = \min_{\alpha_i^t, x_i^t, y_k^t, s_k^t \mu_l^t, q_l^t, \nu_l^t, p_l^t} \Bigg\{ \sum_{t=1}^{T} \Big[\sum_{i=1}^{I} (U_i\beta_i^t + F_i\alpha_i^t + V_i x_i^t + D_i\gamma_i^t) + \sum_{k=1}^{K} (H_k y_k^t + S_k s_k^t) + \sum_{l=1}^{L} (G_l q_l^t - P_l p_l^t) \Big] + \sum_{t=1}^{T} \Big[\lambda_1^t (d^t - \sum_{i=1}^{I} x_i^t - \sum_{k=1}^{K} y_k^t - \sum_{l=1}^{L} (q_l^t - \Theta_l p_l^t)) \Big] + \sum_{t=1}^{T} \Big[\lambda_2^t (d^t + R - \sum_{i=1}^{I} \bar{x}_i \alpha_i^t - \sum_{k=1}^{K} \bar{y}_k - \sum_{l=1}^{L} (\bar{q}_l \mu_l^t - \Theta_l p_l^t)) \Big] \Bigg\}. \tag{5.3.3}$$

In a more condensed form which emphasizes the problem decomposition, the Lagrangian formulation can be written as

$$\Phi(\lambda_1,\lambda_2) = \sum_{i=1}^{I} \Phi_i^I(\lambda_1^t,\lambda_2^t) + \sum_{k=1}^{K} \Phi_k^{II}(\lambda_1^t,\lambda_2^t) + \sum_{l=1}^{L} \Phi_l^{III}(\lambda_1^t,\lambda_2^t) + \sum_{t=1}^{T} \Big[\lambda_1^t d^t + \lambda_2^t (d^t + R) \Big] \tag{5.3.4}$$

where

$$\Phi_i^I(\lambda_1,\lambda_2) = \min_{\alpha_i^t, x_i^t} \sum_{t=1}^{T} (U_i\beta_i^t + F_i\alpha_i^t + V_i x_i^t + D_i\gamma_i^t - \lambda_1^t x_i^t - \lambda_2^t \bar{x}_i \alpha_i^t) \tag{5.3.5}$$

for $i = 1, \ldots, I$

represents the thermal units subproblems,

$$\Phi_k^{II}(\lambda_1,\lambda_2) = \min_{y_k^t, s_k^t} \sum_{t=1}^{T} (H_k y_k^t + S_k s_k^t - \lambda_1^t y_k^t - \lambda_2^t \bar{y}_k) \tag{5.3.6}$$

for $k = 1, \ldots, K$

the conventional hydro units subproblems and

$$\Phi_l^{III}(\lambda_1,\lambda_2) = \min_{\mu_l^t, q_l^t, \nu_l^t, p_l^t} \sum_{t=1}^{T} \Big[G_l q_l^t - P_l p_l^t - \lambda_1^t (q_l^t - \Theta_l p_l^t) - \lambda_2^t (\bar{q}_l \mu_l^t - \Theta_l p_l^t) \Big] \qquad (5.3.7)$$

$$\text{for } l = 1, \ldots, L$$

the pumped-storage units subproblems. Each one of these subproblems is locally constrained by the operating characteristics of the individual units. The final term

$$\sum_{t=1}^{T} \Big[\lambda_1^t d^t + \lambda_2^t (d^t + R) \Big] \qquad (5.3.8)$$

represents the overall demand and reserve.

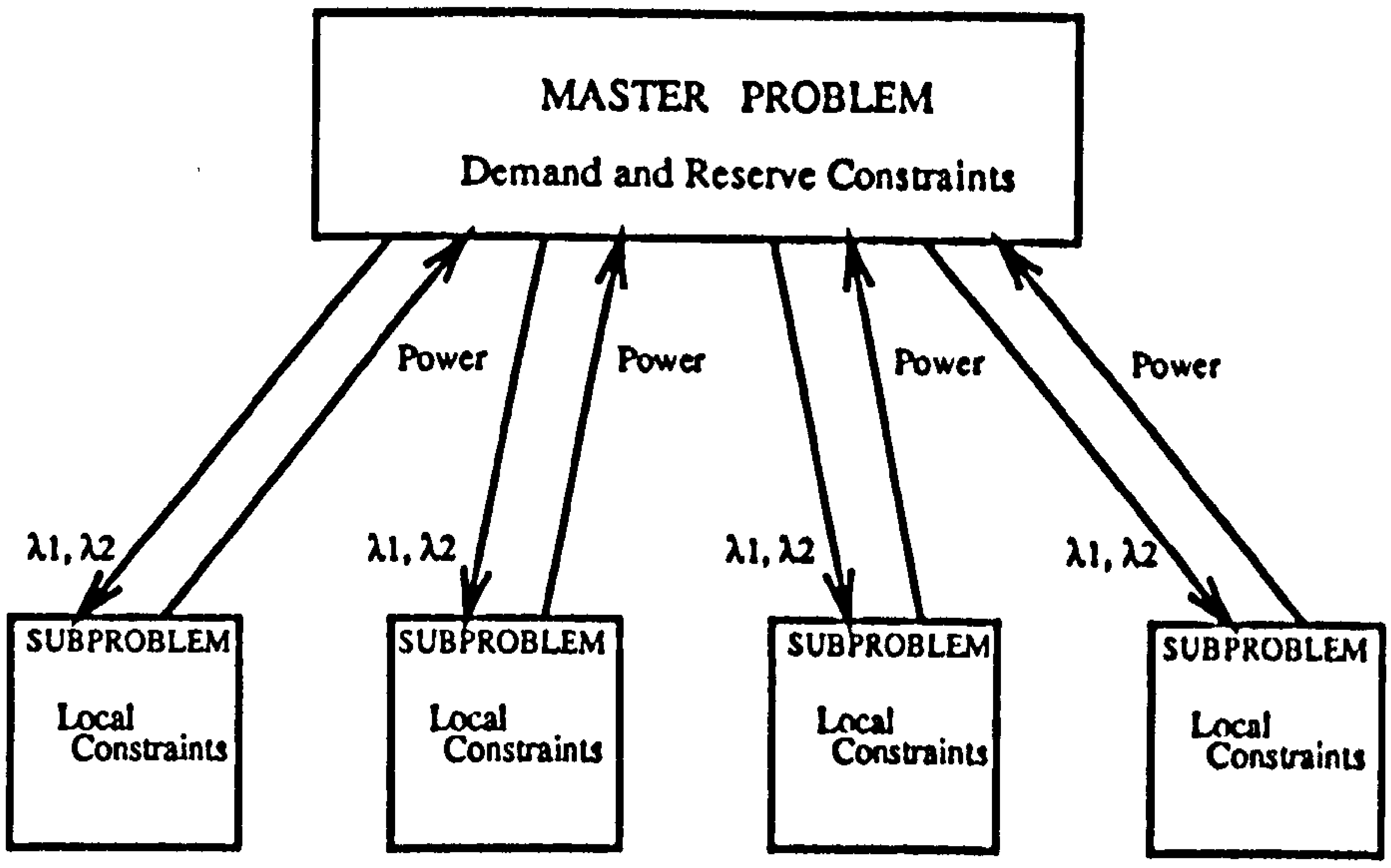


Fig. 5.1. Lagrangian relaxation

Figure 5.1 shows the information exchange between the master and the local subproblems. Clearly, given the values of the Lagrange multipliers, the subproblems can be solved independently of each other. Consequently an algorithmic parallelisation can be implemented such that all subproblems are solved simultaneously; so the subproblems may be regarded as one stage in a two-stage sequential program consisting of this stage and the master problem. This will be efficient provided that the master program is not too time consuming compared with the subproblems and there are not great imbalances between the computational times of the subproblems.

So, the dual problem can be stated as

$$\max \Phi(\lambda_1, \lambda_2) \tag{5.3.9}$$

subject to

$$\lambda_1 \geq 0, \ \lambda_2 \geq 0.$$

5.4. Computation of the Lagrange multipliers

The subgradient technique [Held, Wolfe and Crowder (1974)] is used to compute the multipliers. The dual function $v(PR_\lambda)$ is concave but not differentiable at points where the Lagrangian problem has multiple optimal values. The subgradient method can be regarded as an application of the gradient method in which gradients are replaced by subgradients. Given an initial value λ^0, the sequence $\{\lambda^m\}$ is generated by the rule

$$\lambda^{m+1} = \lambda^m + s^m(\mathbf{b}_1 - A_1\mathbf{x}^m)$$

where $\mathbf{x}^m$ is an optimal solution to (PR_{λ^m}) and s^m is a positive scalar step size (m is the iteration number), with

$$v(PR_{\lambda^m}) \to v(D) \quad \text{if} \quad s^m \to 0 \quad \text{and} \quad \sum_{j=1}^{m} s^j \to \infty \quad \text{as} \quad m \to \infty.$$

In the particular problem under consideration, and only for the demand constraint, the application of the subgradient technique will produce the following new estimate for the Lagrange multiplier λ_1,

$$\lambda_1^{t,m+1} = \max\left\{0, \lambda_1^{t,m} + s^m\left[d^t - \sum_{i=1}^{I} x_i^t - \sum_{k=1}^{K} y_k^t - \sum_{l=1}^{L}(q_l^t - \Theta_l p_l^t)\right]\right\} \quad (5.4.1)$$

$$\text{for } t = 1, \ldots, T$$

where

$$s^m = \frac{h_m\Big(v(P') - \Phi(\lambda_1, \lambda_2)\Big)}{W} \quad (5.4.2)$$

with

$$W = \sum_{t=1}^{T}\left\{\left[d^t - \sum_{i=1}^{I} x_i^t - \sum_{k=1}^{K} y_k^t - \sum_{l=1}^{L}(q_l^t - \Theta_l p_l^t)\right]^2 + \left[d^t + R - \sum_{i=1}^{I} \bar{x}_i \alpha_i^t - \sum_{k=1}^{K} \bar{y}_k - \sum_{l=1}^{L}(\bar{q}_l \mu_l^t - \Theta_l p_l^t)\right]^2\right\} \quad (5.4.3)$$

and $h_m \in (0,2]$ and $v(P')$ is an upper bound on $v(D)$. Similarly, the same kind of estimate can be produced for λ_2. Held, Wolfe and Crowder (1974) provide a description of the subgradient method as well as a justification for the formulae used above.

In order to reduce the oscillatory character of the method, another approach [Merlin and Sandrin (1983)] was implemented for the sequence s^m, with $s^m = a_1/(1 + a_2 m)$, where a_1, a_2 are two fixed parameters defining the sequence of steplengths. Some information regarding the marginal costs is used to update the multipliers from a feasible solution.

$$\lambda_1^{t,m+1} = \rho\lambda_1^{t,m} + (1-\rho)\overline{M}^{t,m} \quad (5.4.4)$$

where $\rho \in (0,1)$ is a predetermined scalar and $\overline{M}^{t,m}$ is the weighted average marginal cost over all committed units in time period t.

5.5. Dynamic programming

The solution of the subproblems is accomplished by Dynamic Programming. By considering each individual unit as a subproblem within the system, the number of states that require to be considered is limited. Furthermore, dynamic programming can handle the discrete nature of these subproblems.

Dynamic programming (DP) is an optimization technique developed in the late fifties by R. Bellman who enunciated the Principle of Optimality:

"An optimal policy has the property that whatever the initial state and decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision"[Bellman and Dreyfus (1962)].

Dynamic programming is a very effective method for solving optimization problems involving a sequence of interrelated decisions, often also referred to as multistage decision processes. The DP approach consists of decomposing the interrelated decisions into a sequence of decisions. Because a low cost decision may trigger future high costs, a decision is taken at each stage which optimizes the current stage and the optimum that may arise from future stages. In other words, the DP approach gives the optimal rule for choosing, at each stage, the optimal sequence of decisions.

Though variational techniques are used for deterministic problems, DP has wider applicability since it can handle difficult constraint sets such as integer or discrete sets. Also DP leads to a globally optimal solution whereas this cannot in general be guaranteed using variational techniques.

Computationally, DP is very efficient when compared, for instance, with 'brute force' enumeration. The efficiency is a direct result of the Principle of Optimality because, having chosen some initial decision, there is no need to examine all policies involving that particular choice, but rather only those policies which are optimal for the remaining states [Bellman and Dreyfus (1962)]. However, as additional variables are incorporated into the definition of a state, the optimization problem grows exponentially in the number of state variables. This disadvantage has long since been coined by Bellman as the 'curse of dimensionality'.

Definitions

Dynamic Programming involves the use of some technical terms which are defined below:

States - A state is a description of a particular configuration of the system. Thus, a system can be seen as the set of all possible states over which it can evolve. Therefore, the state space is a nonempty set and an element of this set $s \in S$ describes the particular condition of the system. States are represented by variables which can be single or vector, continuous or discrete.

Stages - A stage corresponds to the transition from one state to the next. So, an integer N exists such that the state space can be partitioned into $N+1$ sets, $S_1, S_2, \ldots, S_{N+1}$, which enumerates the stages.

Decisions - A decision causes a transition from one state to the next. A state must contain all the information that is relevant to characterize the decision set $D(s)$ associated with this state s, i.e., all the possible choices that can be made from state s. A particular decision $d \in D(s)$ involves a cost and causes a transition from state s to state $t(s, d)$.

Transitions - The system under study evolves from one state to another state in the next stage, and so a transition $t(s, d)$ occurs from a state s in S

to another state in S as a result of a decision d. The set of states $T(s, d_n(s))$ which result from decision $d_n(s)$ is the set to which the system can evolve to as a result of decision $d_n(s)$. The set function, sometimes referred to as the transformation or transition function, determines the evolution of the process from state to state; at the last stage the set is empty.

Policies - A policy p is an ordered set of decisions, one for each state. The set of all possible policies P is defined as the cartesian product of the decision sets $P = \times_{s \in S} D(s)$.

Returns - The system being optimized generates a return at every stage for a given policy. The return function $r(s, d)$ is associated with policy p and state s. Then $r_p(s)$ is the return which would be obtained if the process were initiated in state s and if policy p were followed. The total return is no more than the accumulation of all returns generated by the particular policy. The DP approach aims to find the optimal return for any state and most importantly the value of the state variable for the original problem.

Functional equations - The functional equation specifies the optimality criterion in the sequential decision process [Hastings (1973)] and can be understood as the value assigned to a state by taking the optimum over all relevant actions of the generated returns from the transition to the next state.

For a multistage problem with N stages $f_n(s_n)$ is defined as the optimal value of the objective function when there are n states and the state variable is s_n. Once s_n and d_n are selected the vector of the remaining $n-1$ state variables is given by $t(s_n, d_n)$. An example of a functional equation is

$$f_n(s_n) = \max_{d_n \in D(s_n)} \{r_n(s_n, d_n) + f_{n-1}(t(s_n, d_n))\}$$

$$f_1(s_1) = 0.$$

Bellman's Principle of Optimality

An optimal policy $(d_1, d_2, \ldots, d_N)$ has the property that whatever the initial state s_0 and the initial decision d_1 are, the remaining decisions $(d_2, d_3, \ldots, d_N)$ must constitute an optimal policy for the $N-1$ stage process starting in the state s_1, which results from the first decision d_1.

The application of the principle of optimality [e.g. Hastings (1973), Cooper and Cooper (1981)] requires two conditions to be met:

1) Separability of the objective function

The objective function must be separable, otherwise the value of a state cannot be calculated by a recursive algorithm for a given policy. This condition implies that for all k, the effect of the final k stages on the objective function of an n stage problem depends only on state s_{n-k} and upon the final k decisions $d_{n-k+1}, d_{n-k+2}, \ldots, d_N$.

2) State separation condition

It might appear that policies depending upon the knowledge of the entire history of the system would be superior to those using simply the current state s_n. However, once decision d_{k+1} is made, the resulting state s_{k+1} depends only on s_k and d_{k+1} and does not depend upon the previous states $s_0, s_1, s_2, \ldots s_{k-1}$ [Bertsekas (1987)]. This very important condition means that the only relevant information available regarding past states is contained in s_k.

5.6. Thermal unit

A thermal unit can either be generating power, in which case its commitment variable takes the value, $\alpha_i^t = 1$, or can be switched off in which case $\alpha_i^t = 0$. Thus the state of a particular unit is the number of hours τ_i^t the unit has been on or off. Figure 5.2 diagrammatically portrays the different states for a thermal unit with $\bar{\tau}_i = 2$ hours of minimum up time and $\underline{\tau}_i = 3$ hours of minimum down time. Expressed recursively this is

$$\tau_i^{t+1} = \begin{cases} \tau_i^t + 1, & \text{if } \tau_i^t \geq 1, \quad \alpha_i^{t+1} = 1 \\ 1, & \text{if } \tau_i^t \leq -1, \quad \alpha_i^{t+1} = 1 \\ \tau_i^t - 1, & \text{if } \tau_i^t \leq -1, \quad \alpha_i^{t+1} = 0 \\ -1, & \text{if } \tau_i^t \geq 1, \quad \alpha_i^{t+1} = 0. \end{cases} \tag{5.6.1}$$

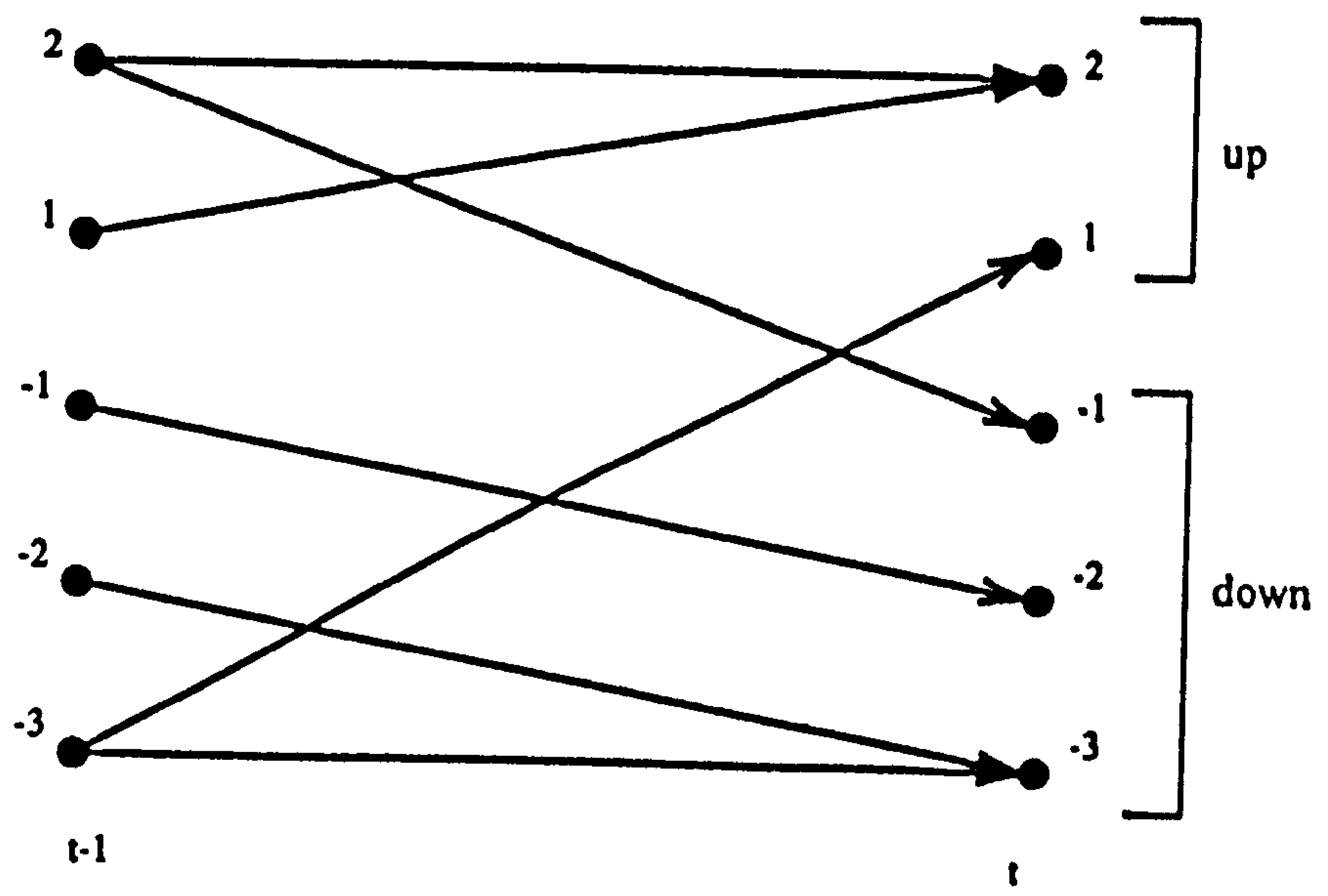


Fig. 5.2. State transition matrix

The different transition costs are a function of the state in the previous time step. These transition costs can be classified in three groups:

- normal operational costs, state variable greater than or equal to $\bar{\tau}_i$,
- start up costs plus operational costs, state variable changing from $\underline{\tau}_i$ to 1,
- shut down costs, state variable changing from $\bar{\tau}_i$ to -1.

All other transitions have zero costs.

The functional equation then takes the following form,

$$f_i^I(\tau_i^t, x_i^t) = \min_{\alpha_i^{t+1}, x_i^{t+1}} \Big\{ U_i\beta_i^{t+1} + F_i\alpha_i^{t+1} + V_i x_i^{t+1} + D_i\gamma_i^{t+1} - \lambda_1^{t+1}x_i^{t+1} - \lambda_2^{t+1}\bar{x}_i\alpha_i^{t+1} + f_i^I(\tau_i^{t+1}, x_i^{t+1}) \Big\} \tag{5.6.2}$$

$$\text{for } i = 1, \ldots, I \text{ and for } t = 0, \ldots, T-1,$$

with boundary conditions

$$\tau_i^0 = \bar{\tau}_i^0, \qquad x_i^0 = \bar{x}_i^0, \qquad f_i^I(\tau_i^T, x_i^T) = 0$$

and subject to the local constraints given in Chapter 2, Section 2.1.

5.7. Hydro unit

The solution of the hydro subproblem by DP involves continuous variables and consequently for its solution by DP a discretization of the search space is necessary. Figure 5.3 shows an example of such a discretization, the fineness of which conditions the computation time. Figure 5.3 also shows the limits on the search space and consequently the accuracy of the solution. The minimum and maximum level of the reservoir are imposed by the operating characteristics of the unit. The other restrictions on the search space result from extreme modes of operation. The broken line from V_{in} to V_{max} is the result of considering the natural inflows and minimum discharge; the line from V_{in} to V_{min} is constructed considering the natural inflows and maximum discharges. It should be pointed out that large natural inflows may result in spillage at the end of the planning period.

The use of DP for this subproblem, due to the large search space, can be expensive in terms of computational time. For the particular case studied, and for a discretization of the order of the minimum discharge, the computation times are very large when compared to the other subproblems. The reduction of those times can be accomplished by using a coarse grid, which then results in less accurate solutions.

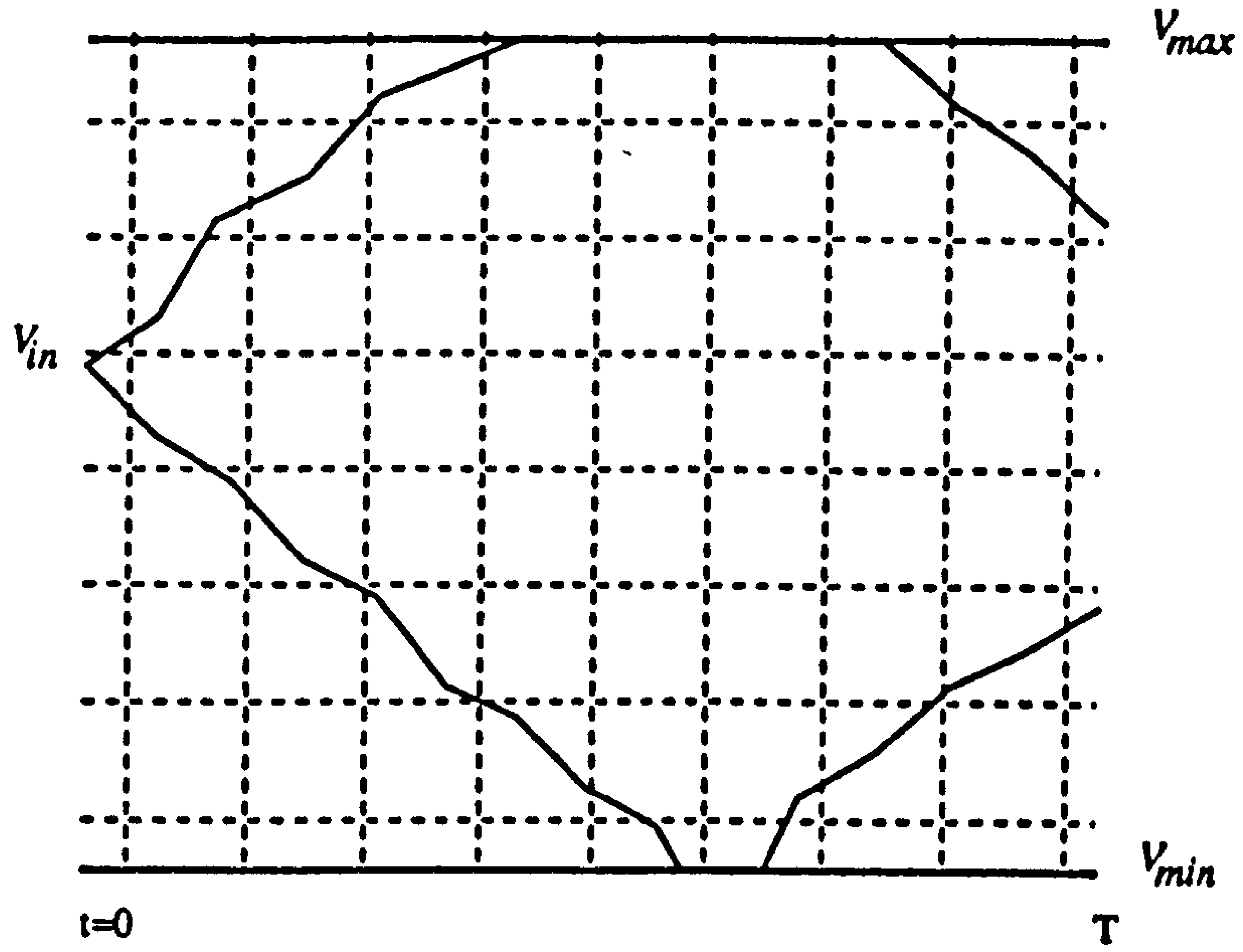


Fig. 5.3. Hydro unit search space

The functional equation is of the form

$$f_k^{II}(v_k^t) = \min_{y_k^{t+1}, s_k^{t+1}} \left\{ H_k y_k^{t+1} + S_k s_k^{t+1} - \lambda_1^{t+1} y_k^{t+1} - \lambda_2^{t+1} \bar{y}_k + f_k^{II}(v_k^{t+1}) \right\} \quad (5.7.1)$$

for $k = 1, \ldots, K$ and for $t = 0, \ldots, T-1$.

This can be expressed in terms of v_k^t, noting that,

$$y_k^t = v_k^{t-1} - v_k^t + f_k^t, \qquad \text{if } \underline{y}_k \leq y_k^t \leq \bar{y}_k \quad (5.7.2)$$

and

$$s_k^t = v_k^{t-1} - v_k^t + f_k^t - \bar{y}_k, \qquad \text{if } y_k^t = \bar{y}_k. \quad (5.7.3)$$

Then, for $\underline{y}_k \leq y_k^t \leq \bar{y}_k$

$$f_k^{II}(v_k^t) = \min_{v_k^t} \left\{ (H_k - \lambda_1^{t+1})(v_k^t - v_k^{t+1} + f_k^{t+1}) - \lambda_2^{t+1} \bar{y}_k + f_k^{II}(v_k^{t+1}) \right\} \quad (5.7.4)$$

and for $s_k^t > 0$

$$f_k^{II}(v_k^t) = \min_{v_k^t}\Big\{(H_k - \lambda_1^{t+1})\bar{y}_k + S_k(v_k^t - v_k^{t+1} + f_k^{t+1} - \bar{y}_k) - \lambda_2^{t+1}\bar{y}_k + f_k^{II}(v_k^{t+1})\Big\} \tag{5.7.5}$$

with boundary conditions

$$v_k^0 = \bar{v}_k^0, \qquad f_k^{II}(v_k^T) = 0$$

and subject to the local constraints given in Chapter 2, Section 2.2.

5.8. Pump-Storage unit

The model of the pump-storage unit includes two integer variables, one associated with pumping, the other with generation. These two variables account for the nonlinearity since there cannot be generation and pumping at the same time. Also in this model, a given daily inflow is prescribed and the operation of the unit must avoid spillage.

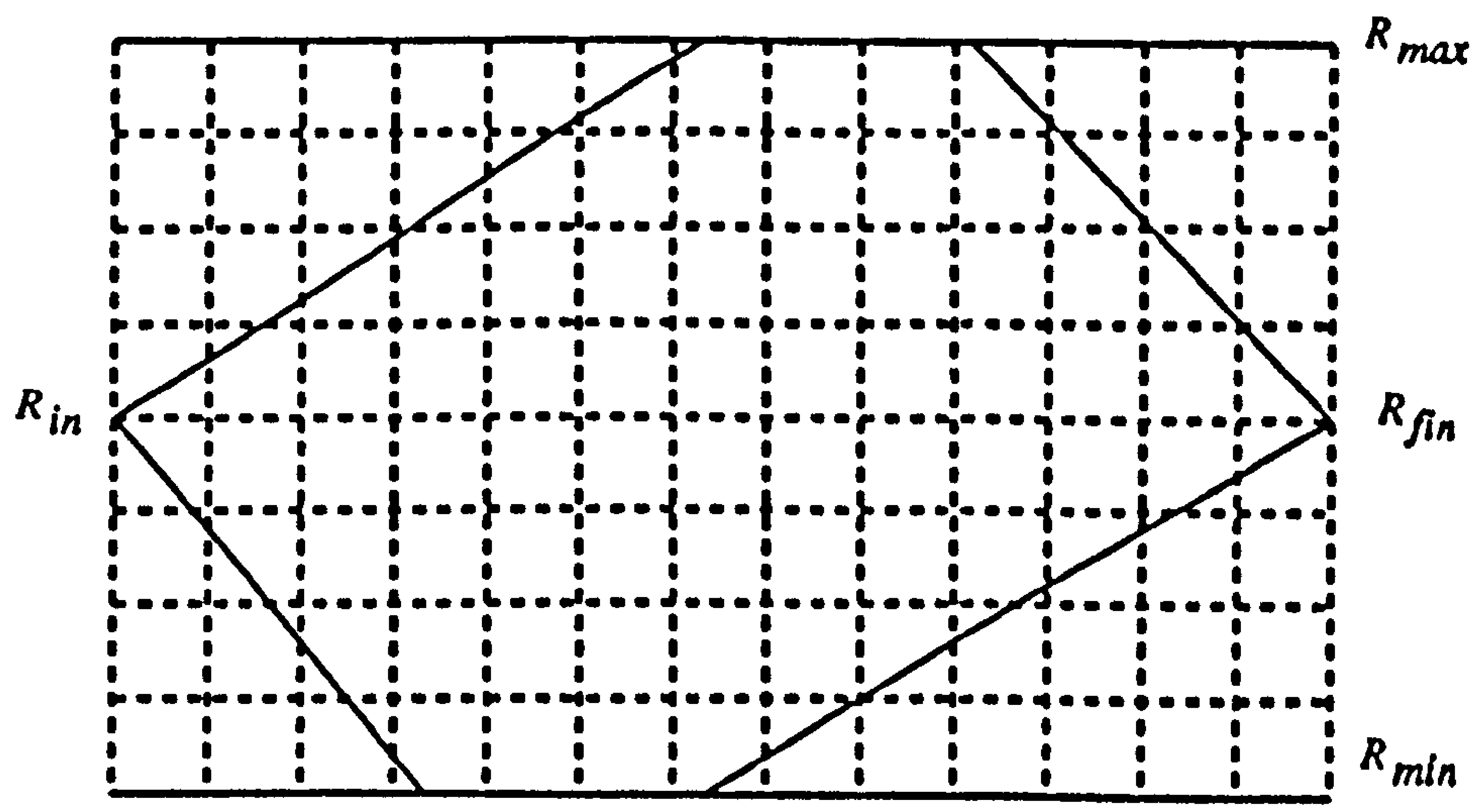


Fig. 5.4. Pump-storage unit search space

For the implementation of the DP algorithm, a discretization of the search space is considered. Figure 5.4 depicts this discretization; clearly the computational effort is closely related to the fineness of the grid. Also in Figure 5.4, limits are imposed on the search space as a result of extreme modes of operation; the line from the initial reservoir level to the maximum level represents an operation where pumping would be at maximum power from the very beginning until the maximum level was reached. Conversely, the line from the origin to the minimum level represents a situation where generation would be used at maximum power until the minimum reservoir level was attained. Similar considerations can be made to determine the other limits.

Bearing in mind that throughout all the planning period the reservoir level has to be within the permissible limits, the functional equation is constructed with the level of the reservoir as the state variable. By continuity arguments the flows, either during generation or pumping, may be expressed as a function of the reservoir level. The number of stages will be the number of time periods of the planning horizon.

The functional equation describing this unit takes the following form

$$f_l^{III}(r_l^t) = \min_{\mu_l^{t+1}, q_l^{t+1}, \nu_l^{t+1}, p_l^{t+1}} \Big\{ G_l q_l^{t+1} - P_l p_l^{t+1} - \lambda_1^{t+1}(q_l^{t+1} - \Theta_l p_l^{t+1}) - \lambda_2^{t+1}(\bar{q}_l \mu_l^{t+1} - \Theta_l p_l^{t+1}) + f_l^{III}(r_l^{t+1}) \Big\} \tag{5.8.1}$$

for $l = 1, \ldots, L$ and for $t = 0, \ldots, T-1$

which can be expressed in terms of r_l^t, since

$$q_l^t = r_l^{t-1} - r_l^t + g_l^t \tag{5.8.2}$$

$$p_l^t = r_l^t - r_l^{t-1} - g_l^t. \tag{5.8.3}$$

Then, if generation takes place, $\nu_l^{t+1} = 0, \mu_l^{t+1} = 1$

$$f_l^{III}(r_l^t) = \min_{r_l^t}\Big\{(G_l - \lambda_1^{t+1})(r_l^t - r_l^{t+1} + g_l^{t+1}) - \lambda_2^{t+1}\bar{q}_l + f_l^{III}(r_l^{t+1})\Big\} \quad (5.8.4)$$

or in the case of pumping, $\nu_l^{t+1} = 1, \mu_l^{t+1} = 0$

$$f_l^{III}(r_l^t) = \min_{r_l^t}\Big\{[P_l + (\lambda_1^{t+1} + \lambda_2^{t+1})\Theta_l](r_l^{t+1} - r_l^t - g_l^{t+1}) + f_l^{III}(r_l^{t+1})\Big\} \quad (5.8.5)$$

or when the unit is off, $\nu_l^t = \mu_l^t = 0$

$$f_l^{III}(r_l^t) = f_l^{III}(r_l^{t+1}) \quad (5.8.6)$$

with initial and final conditions,

$$r_l^0 = \bar{r}_l^0, \qquad r_l^T = \bar{r}_l^T, \qquad f_l^{III}(r_l^T) = 0$$

subject to the local constraints given in Chapter 2, Section 2.3. The inclusion of a natural inflow, for reasonably small quantities (less than 10% of maximum discharge) does not affect significantly the limits of the search space. The computational time required seems not to be excessive for a mesh spacing of the order of the minimum permissable level of generation.

5.9. A heuristic for the feasible solution

Given the solution to the dual problem, the question arises as to how to construct a feasible solution to the original primal problem. Usually the solution obtained from the optimization of the dual will be nearly feasible for the primal, and it is possible to construct some kind of heuristic to obtain feasibility. Again the answer to this question is problem specific: Fisher (1981) reports on several heuristics for different kinds of problems.

The heuristic used to generate a feasible solution takes into account the information provided by the dual solution. At each time step, the dual solution will either be in the regime of overproduction or underproduction, that is, the power output of all the committed units will either be above or below the demand, while satisfying the reserve constraint. In each case, the marginal costs (the gradient of the cost function) of the different committed units are taken into consideration, and the power output is adjusted accordingly. So, for instance, in the case of overproduction, the power output of the unit or units with greater marginal costs will be reduced until the demand restriction is satisfied as an equality. Conversely, the power output of the unit or units with smaller marginal costs will be increased for the case of underproduction. It is perhaps worth pointing out that the Lagrange multipliers may be regarded as shadow prices, that is the cost of the additional power needed to satisfy the demand and reserve constraints. The model also considers the possibility of spillage. In this case the commitment of the dual solution may be changed since the optimal policy is to reduce spillage to a minimum.

5.10. Results

The results obtained are listed in Table 5-1: the primal solution, the dual solution, the percentage difference between the dual and primal solutions, the number of iterations, and the CPU time used (min:sec).

Table 5-1
Numerical Results - 1 to 7 days

Days	Primal †	Dual †	% Diff.	Iterations	CPU
1	$1.656E5$	$1.649E5$	0.4	9	00 : 07
2	$3.392E5$	$3.385E5$	0.2	11	00 : 19
3	$4.940E5$	$4.928E5$	0.2	16	00 : 49
4	$6.488E5$	$6.481E5$	0.1	23	01 : 41
5	$8.009E5$	$7.976E5$	0.4	20	02 : 01
6	$9.281E5$	$9.237E5$	0.5	42	06 : 00
7	$1.056E6$	$1.049E6$	0.7	150	20 : 55

†(Solution in £)

5.11. Summary

The implementation of Lagrangian relaxation using dynamic programming to optimize the individual units has considerably reduced the computational times over those taken by the branch-and-bound implementation of Oliveira, McKee and Coles (1991). From CPU times in excess of 4 hours on a VAX 11/785, this implementation has reduced the computational time to less than 1/2 hour in the worst case of scheduling for a whole week. Furthermore, the Lagrange multipliers can be perceived as shadow prices in the sense that they represent the costs required to satisfy the demand and reserve constraints. This feature, in the context of the recent privatization of the generating boards in the UK, makes this implementation a valuable tool as it effectively costs the energy, whether from thermal, hydro or pump-storage. Finally, the special structure which results from this implementation, with a master problem coordinating several subproblems, is suitable for a parallel implementation which as is shown in the next chapter leads to further reduction in computational time.

CHAPTER 6

PARALLEL LAGRANGIAN IMPLEMENTATION

6.1. Introduction

The advent of high performance computing has had an enormous impact on scientific computing and numerical analysis mostly because problems that, a few years ago, were thought to be unsolvable in a reasonable time are today within reach.

Power system scheduling involves decisions concerning which units should be run and what their level of output should be. Together these manifest themselves as a large scale mixed integer programming problem as has been shown in the previous chapters. Power systems can vary from a small number of units to several hundred; and the planning period can vary from short term (one to seven days) to long term (several weeks up to a year). This highly combinatorial problem is further complicated by two main stochastic inputs: the demand which the system must satisfy and, in the case of hydro systems, the inflows to the reservoirs. The full extent of this problem is such that it is essential to employ the most efficient algorithm and the most appropriate computer architecture configuration. To this end, a dual formulation has been obtained through Lagrangian relaxation of the original primal problem. This admitted decomposition into more tractable subproblems which has allowed the implementation of the algorithm on the Edinburgh Concurrent Supercomputer (ECS).

6.2. Parallel Lagrangian relaxation

The system considered here is a small part of the total set of generating units of Scottish Hydro-Electric plc, consisting of two thermal units, one conventional hydro unit and a pump-storage unit. This subsystem may be regarded as a paradigm for the real problem: the approach described here can in principle be used to solve the full system. Mathematically, the objective is to minimize the following mixed integer linear programming problem

$$F = \min_{\alpha_i^t, x_i^t, y_k^t, s_k^t, \mu_l^t, q_l^t, \nu_l^t, p_l^t} \sum_{t=1}^{T} \left\{ \sum_{i=1}^{I} f_i^I(\alpha_i^t, x_i^t) + \sum_{k=1}^{K} f_k^{II}(y_k^t, s_k^t) + \sum_{l=1}^{L} f_l^{III}(\mu_l^t, q_l^t, \nu_l^t, p_l^t) \right\}, \tag{6.2.1}$$

where f_i^I refers to the costs of starting, running and closing down the i^{th} thermal unit, f_k^{II} to the running costs of the k^{th} hydro unit and f_l^{III} to the running costs of the l^{th} pump-storage unit [Oliveira, McKee and Coles (1991)], subject to the demand and reserve constraints, which are global in the sense that they couple all the committed generating units

$$\sum_{i=1}^{I} x_i^t + \sum_{k=1}^{K} y_k^t + \sum_{l=1}^{L} (q_l^t - \Theta_l p_l^t) \geq d^t, \tag{6.2.2}$$

$$\sum_{i=1}^{I} \bar{x}_i \alpha_i^t + \sum_{k=1}^{K} \bar{y}_k + \sum_{l=1}^{L} (\bar{q}_l \mu_l^t - \Theta_l p_l^t) \geq d^t + R, \tag{6.2.3}$$

and the local constraints $h_i^I, h_k^{II}, h_l^{III}$ representing the operating restrictions of each unit

$$h_i^I(\alpha_i^t, x_i^t) \leq 0 \text{ for } i = 1, 2, \ldots, I, \tag{6.2.4}$$

$$h_k^{II}(y_k^t, s_k^t) \leq 0 \text{ for } k = 1, 2, \ldots, K, \tag{6.2.5}$$

$$h_l^{III}(\mu_l^t, q_l^t, \nu_l^t, p_l^t) \leq 0 \text{ for } l = 1, 2, \ldots, L, \tag{6.2.6}$$

$$\text{for } t = 1, 2, \ldots, T.$$

Lagrangian relaxation is used to create a separable problem by relaxing the coupling constraints so that each subproblem involves only one individual unit subject to its local (operating) restrictions. The problem is then decomposed into a number of local subproblems which are parameterized by the Lagrange multipliers, and a master problem which coordinates the subproblems producing new estimates of the multipliers while ensuring that the two global constraints are met. The inclusion of the two global constraints through the multipliers λ_1, λ_2 gives the following Lagrangian problem

$$\begin{aligned}\Phi(\lambda_1,\lambda_2) = &\\ \min_{\alpha_i^t, x_i^t, y_k^t, s_k^t \mu_l^t, q_l^t, \nu_l^t, p_l^t} \Bigg\{ &\sum_{t=1}^{T}\Big[\sum_{i=1}^{I} f_i^I(\alpha_i^t, x_i^t) \\ &\quad + \sum_{k=1}^{K} f_k^{II}(y_k^t, s_k^t) + \sum_{l=1}^{L} f_l^{III}(\mu_l^t, q_l^t, \nu_l^t, p_l^t)\Big] \\ &+ \sum_{t=1}^{T}\Big[\lambda_1^t\big(d^t - \sum_{i=1}^{I} x_i^t - \sum_{k=1}^{K} y_k^t - \sum_{l=1}^{L}(q_l^t - \Theta_l p_l^t)\big)\Big] \\ &+ \sum_{t=1}^{T}\Big[\lambda_2^t\big(d^t + R - \sum_{i=1}^{I} \bar{x}_i \alpha_i^t - \sum_{k=1}^{K} \bar{y}_k \\ &\quad - \sum_{l=1}^{L}(\bar{q}_l \mu_l^t - \Theta_l p_l^t)\big)\Big]\Bigg\}. \end{aligned} \tag{6.2.7}$$

In a more condensed form which emphasizes the problem decomposition, the Lagrangian formulation can be written as

$$\begin{aligned}\Phi(\lambda_1,\lambda_2) = \sum_{i=1}^{I}\Phi_i^I(\lambda_1^t,\lambda_2^t) + \sum_{k=1}^{K}\Phi_k^{II}(\lambda_1^t,\lambda_2^t) + \sum_{l=1}^{L}\Phi_l^{III}(\lambda_1^t,\lambda_2^t) \\ + \sum_{t=1}^{T}\Big[\lambda_1^t d^t + \lambda_2^t(d^t + R)\Big]\end{aligned} \tag{6.2.8}$$

where $\Phi_i^I(\lambda_1,\lambda_2)$ represents the thermal units subproblems, $\Phi_k^{II}(\lambda_1,\lambda_2)$ the conventional hydro units subproblems and $\Phi_l^{III}(\lambda_1,\lambda_2)$ the pumped-storage units subproblems. Each one of these subproblems is locally constrained by the operating characteristics of the individual units $h_i^I, h_k^{II}, h_l^{III}$. Thus, the dual problem can be stated as

$$\Phi^* = \max \Phi(\lambda_1, \lambda_2) \tag{6.2.10}$$

subject to

$$\lambda_1 \geq 0, \; \lambda_2 \geq 0.$$

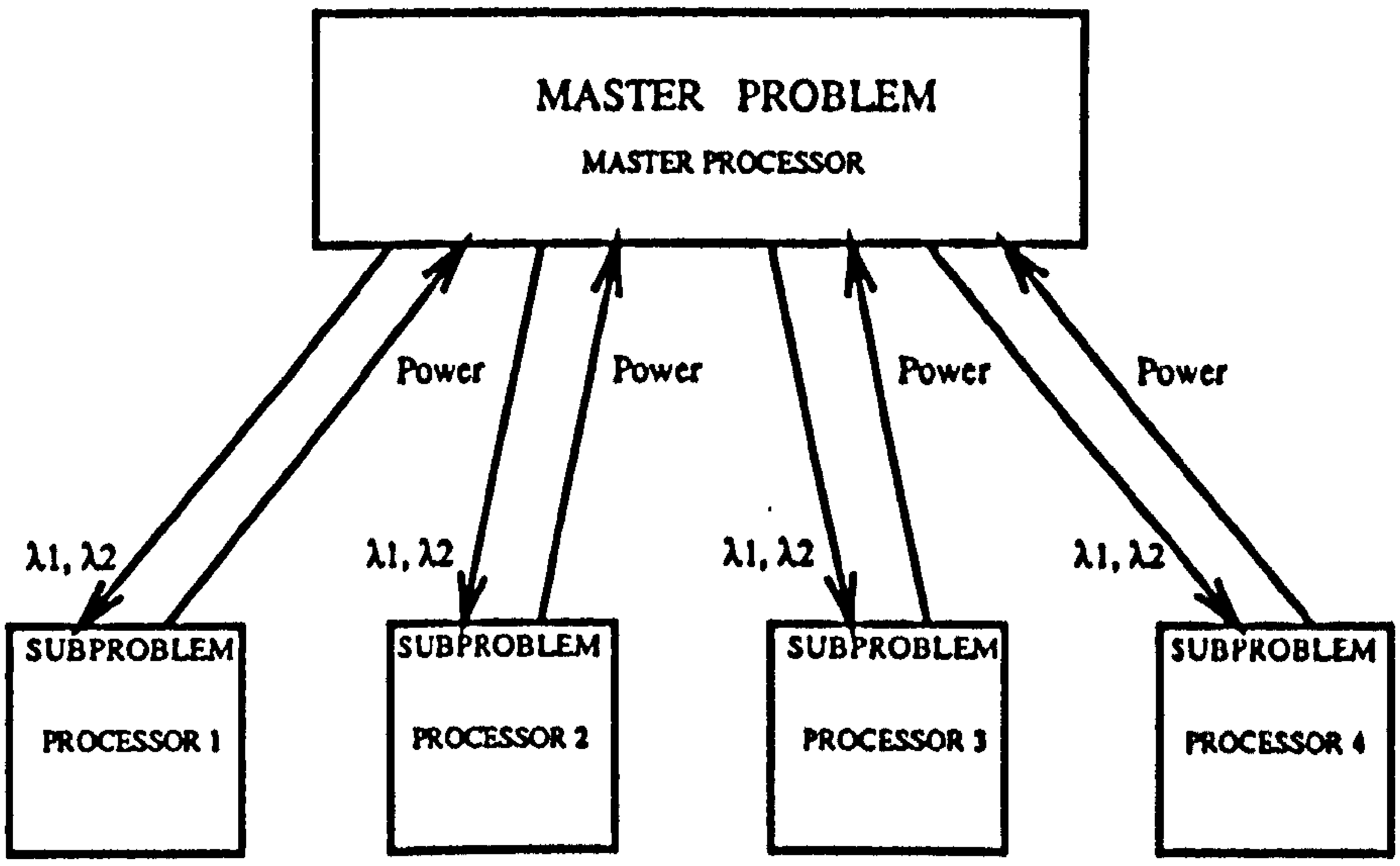


Fig. 6.1. Parallel Lagrangian relaxation

Figure 6.1 shows the information exchange between the master problem and the local subproblems. Clearly, given the values of the Lagrange multipliers, the subproblems can be solved independently of each other. Consequently an algorithmic parallelisation can be implemented such that all subproblems are solved simultaneously. This will be efficient provided that the master problem is not too time consuming compared with the subproblems and there are not great imbalances between the computational times of the individual subproblems.

6.3. Approaches to parallelisation

The main idea behind parallelism is that there are problems where it is possible to perform several independent tasks concurrently and, by executing them on different processors running in parallel, so attain significant reductions in overall computational time compared with sequential execution.

The Edinburgh Concurrent Supercomputer is a Meiko Computing Surface built as a single processor in a MIMD concurrent system. This is a very large transputer array, currently with 300 T800 transputers and 100 T414 transputers. In order to provide multiuser service the surface is organized in domains with a variable number of transputers, the biggest of which has 131 transputers. A transputer can support nine concurrent activities: the processor itself and transferring data through four links in both directions simultaneously. The system runs under Meikos, a UNIX-like operating system. For further information on ECS see Wexler and Prior (1989), Wallace (1991) and Thornton, Blair-Fish and Wilson (1991).

The conversion of sequential algorithms to parallel algorithms is loosely defined as parallelisation. There are several approaches [Wexler and Prior (1989), Thornton, Blair-Fish and Wilson (1991)]: event parallelism, geometric parallelism and algorithmic parallelism. Event parallelism can be used when the global problem consists of many independent but similar problems; these problems do not share any data, and can take different solution times; a master processor controls the worker processors, sending new work as soon as an individual processor has completed its task. Geometric parallelism is particularly suited for cases where there is inter-processor communication, between a problem and its neighbours, as in the case of image processing. Algorithmic parallelism describes the situation where it is possible to divide the algorithm into different component parts which are each allocated to a separate processor;

the algorithm is now distributed and efficiency is obtained when the different component parts are closely balanced in terms of computational time.

In the present study an algorithmic parallelism approach was implemented. Each processor was allocated a subproblem corresponding to an individual generating unit and the master problem allocated to a master processor which controlled the iterative process by exchanging information between the different processors (see Figure 6.1). No information exchange takes place between the subproblems.

Message passing, that is information exchange, was accomplished using CS-tools, a communication 'harness' which is Meiko's environment for multiprocessor programming. This is a toolset for program development for multiprocessor computer systems, and it supports the programming of single and multiprocessor applications using familiar development environments and standard languages [Blair-Fish *et al.* (1990), Meiko (1988)]

6.4. Implementation and results

The conversion from serial to parallel involved the division of the original code into five programs, one for each generating unit plus a controlling master program. Each of these five programs was allocated to an individual processor. Changes in the original code mainly concerned provision for message passing and termination.

Routines were used to communicate between processors using the FORTRAN version of CS-tools and the CS-tools utility 'mrun'. The interface allowed the user to specify the number of processors and prescribe the data to be sent and received in bytes. Termination was accomplished by a flag which signals to the subprograms that the master problem has stopped.

The results obtained for planning periods ranging from one day up to seven days are listed in Tables 6-1 and 6-2. These give the number of iterations, the CPU time (min:sec) for a serial and parallel execution and the speed-up (SU). The last is defined as the ratio of the serial cpu time to the parallel cpu time on 4+1 transputers. Also a measure of the attainable speed-up (ASU) is listed. This is defined as the ratio of the total computation time of the subproblems to the most time consuming subproblem for a serial execution. Table 6-1 displays the results obtained using dynamic programming to solve all the subproblems.

Table 6-1
Time Comparisons - 1 to 7 days

Days	Iterations	Serial	Parallel	SU	ASU
1	9	00 : 07	00 : 21	0.33	1.81
2	11	00 : 19	00 : 31	0.61	1.79
3	16	00 : 49	00 : 47	1.04	1.89
4	23	01 : 41	01 : 24	1.20	1.75
5	20	02 : 01	01 : 38	1.23	1.72
6	42	06 : 00	03 : 41	1.36	1.72
7	150	20 : 55	14 : 19	1.46	1.72

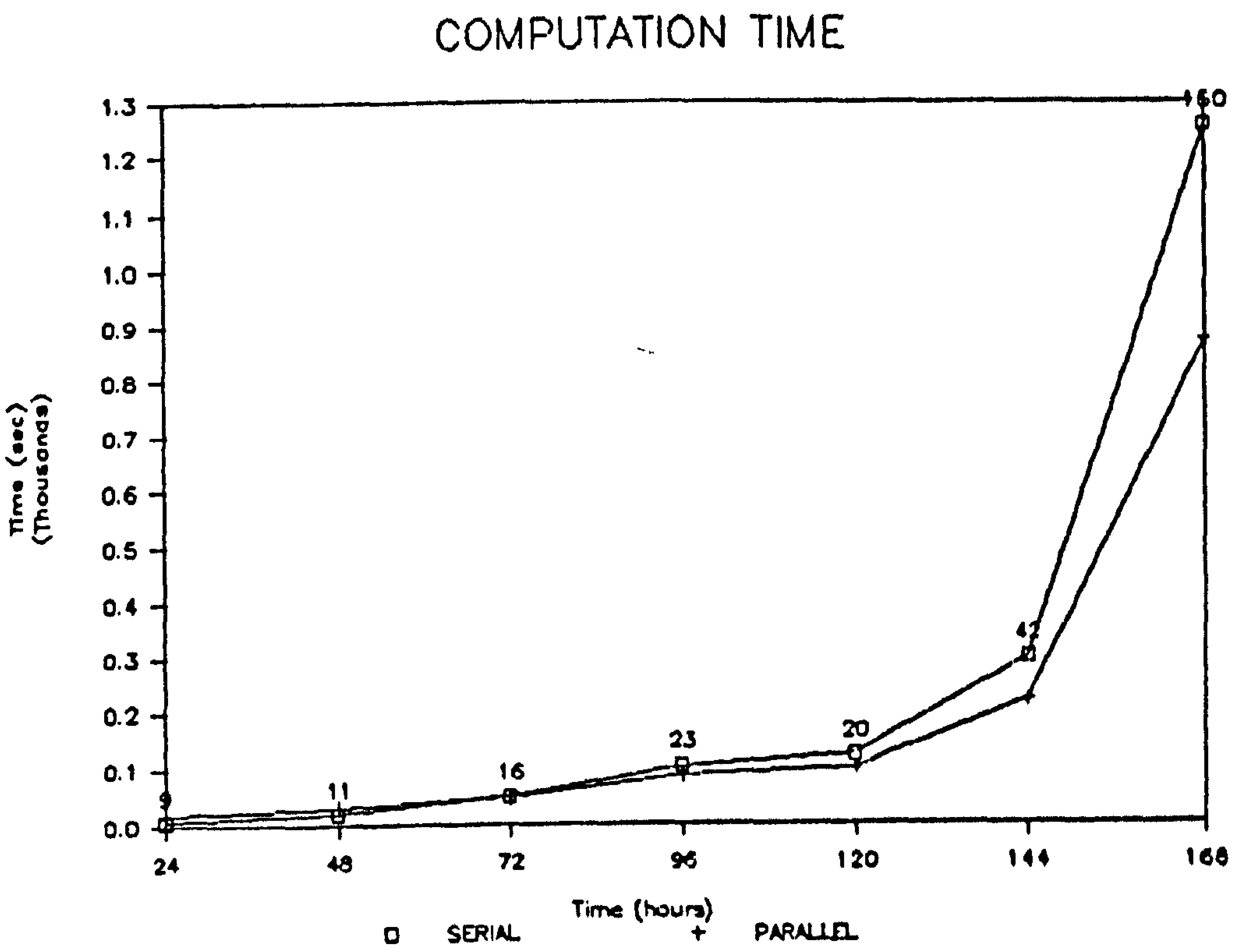
The hydro subproblem is by far and large the most time consuming. In order to achieve a greater speed-up a heuristic was implemented for this subproblem. The hydro problem considers a constant value for the water energy in the reservoir. Spillage, considered as a waste of energy, is an uneconomical operation. Therefore, a useful heuristic is to require that spillage be always kept to its minimum. The dual solution will select a minimum or maximum discharge depending on the value of the Lagrange multipliers. Thus, if the restrictions on the level of the reservoir are ignored, three outcomes are possible: the end level is above the maximum, below the minimum, or between these limits that define the feasible region. In the latter case, and for the operating conditions tested, these limits were never exceeded. If the end level is above the maximum limit, then the discharges on the previous time steps have to be increased till the excess volume is distributed. A time period is selected for which the difference between the hydro energy cost and the value of the Lagrange multipliers is minimum. The corresponding discharge is increased to the maximum. This procedure is then repeated till all the excess volume has been distributed. Conversely, similar considerations can be made for the case where the end level is below the minimum level. Table 6-2 presents the results using the heuristic to solve the hydro unit subproblem. Figures 6.2 and 6.3 display the computational times and the iteration times for both implementations (a linear regression was superimposed on the data concerning the iteration times).

Table 6-2
Time Comparisons - 1 to 7 days

Days	Iterations	Serial	Parallel	SU	ASU
1	9	00 : 05	00 : 19	0.26	1.05
2	21	00 : 19	00 : 34	0.56	1.04
3	10	00 : 17	00 : 34	0.50	1.03
4	31	01 : 05	01 : 19	0.82	1.04
5	23	01 : 04	01 : 20	0.80	1.04
6	40	02 : 09	02 : 24	0.90	1.04
7	42	02 : 40	02 : 55	0.91	1.05

6.5. Summary

In the present system, with four generating units, each one allocated to one transputer, the theoretical maximum speed-up is 4. The actual speed-up falls well short of this value and this is due to an unbalanced workload. However, Table 6-1 shows that the difference between SU and ASU is small for large planning periods. For small planning periods the speed-up can fall below unity due to the excessive communication burden.



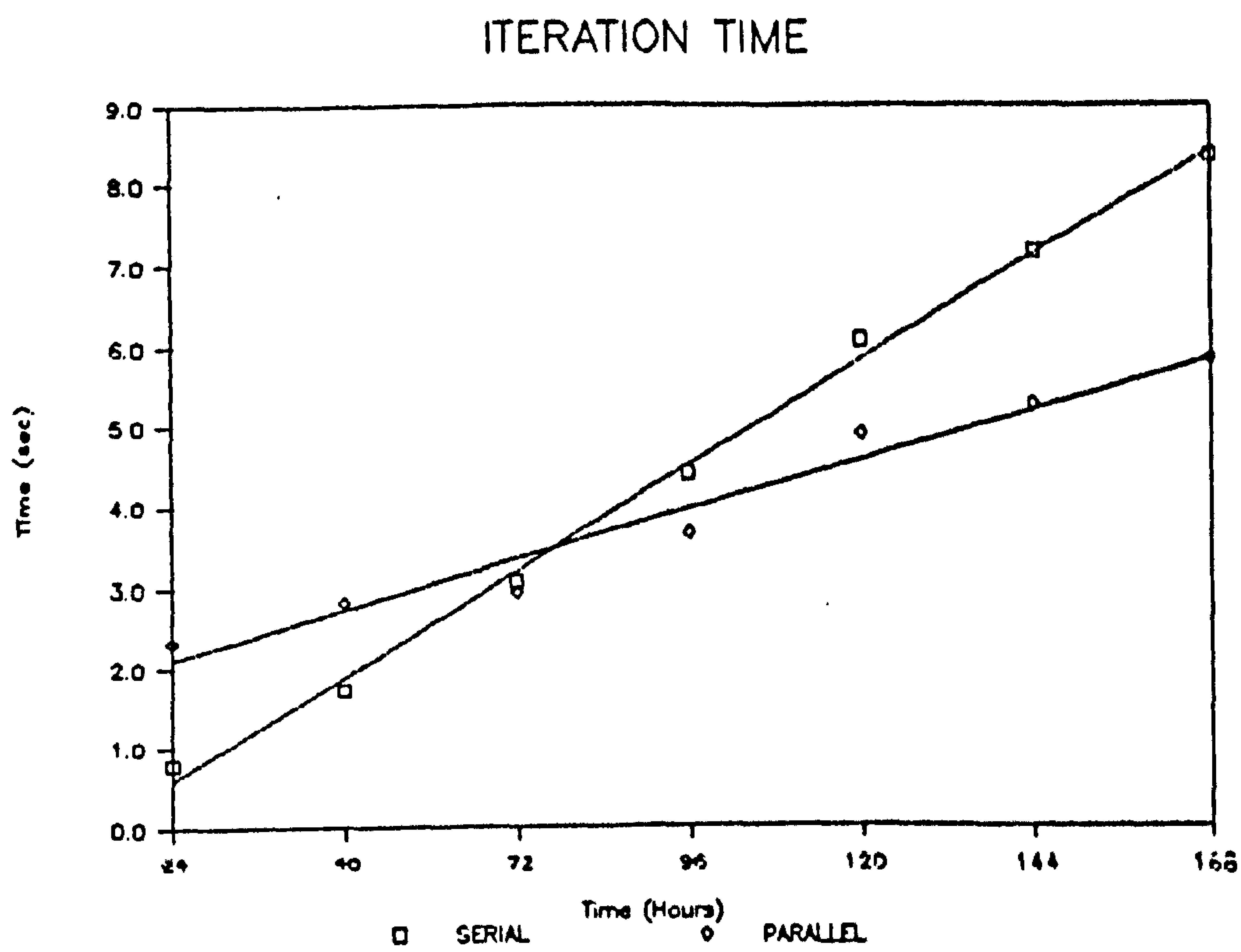


Fig. 6.2. Serial vs parallel - Table 6-1

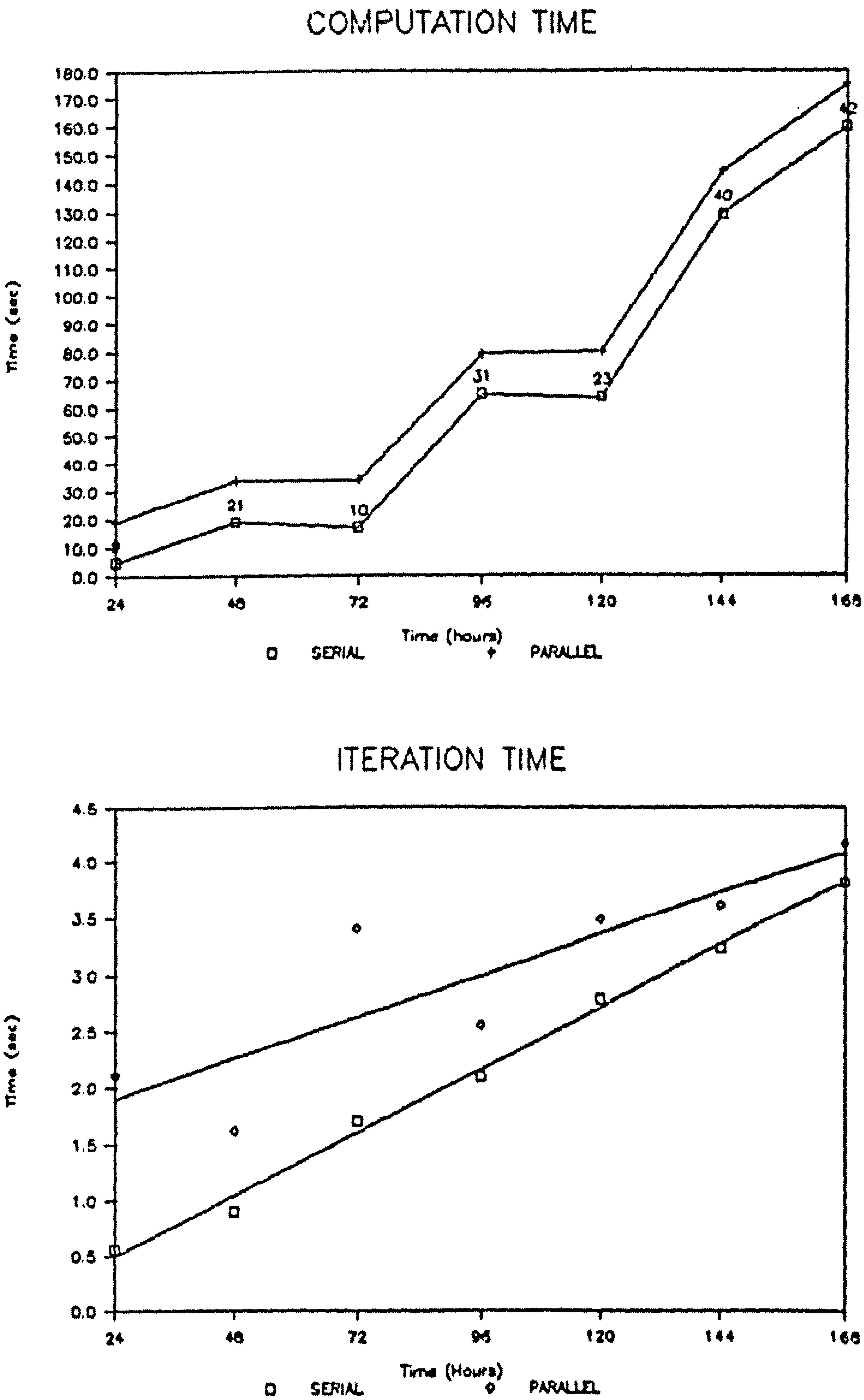


Fig. 6.3. Serial vs parallel - Table 6-2

The fact that the hydro unit was the most time consuming subproblem has lead to the development of an heuristic to solve this subproblem. It should be pointed out that the heuristic used can only be applied to the hydro subproblem because of the linear nature of the restrictions involved. The pump-storage subproblem is, however, nonlinear and so the same heuristic cannot be applied. As a result the hydro and thermal subproblems can therefore be mutually balanced and it is the pump-storage subproblem which now takes the most computing time.

Some tests were also made on a shared memory Sequent system in order to identify and exploit loop level parallelism. However, the nature of the algorithm discourages this approach since there are recurrence relations where a value computed in some previous iteration is needed to compute the value of the current iteration [Edinburgh Portable Compilers (1991)]. This was particularly crucial in the subroutines involving the hydro and pump-storage units as the actual level of the reservoir depends on the previous iteration.

The maximum speed-up is dictated either by the hydro unit or the pump-storage unit. In certain cases when there are more thermal units than hydro units it is possible to construct a well balanced system by assigning several thermal units to one processor. Certainly this is a more efficient use of the transputer array but it does not lead to greater speed-up. Nevertheless, this approach has established that a transputer array can be used to solve very large power systems with a corresponding decrease in the computational times.

CHAPTER 7

GENETIC ALGORITHMS

7.1. Introduction

Genetic algorithms (GAs) are search and optimization algorithms; they were first developed by John Holland (1975). In his work Holland drew attention to the process of natural selection and genetic evolution as an analogy for search and optimization. Different species have been able to survive despite dramatically changing environments; the knowledge to cope with these new conditions has been embodied in their genetic make-up. The analogy exists because complex structures can be represented by a simple code of bit strings, which mimic the genes in a chromosome. In nature, chromosomes are a means of "storage" of information [Dawkins (1989)] containing the instructions governing the make up of the organisms; this information is modified and exchanged through a series of simple processes (reproduction, crossover, mutation and inversion), whose main functions are to pass on the fundamental bits of this information to future generations. Dawkins (1989) argues that the "fundamental unit of selection, and therefore of self-interest, is not the species, nor the group, nor even, strictly the individual", but the gene as the unit of heredity. So the gene can be seen as the fundamental unit of information that lasts for enough generations to serve as a unit of natural selection. In this sense, evolution is the process by which some units of information, that is certain genes, become more numerous while others become less numerous [Dawkins (1989)].

Genetic algorithms may be considered part of the field of Artificial Intelligence (AI). There are many applications mostly on classifier systems [Smith

(1984)]. The main feature that makes GAs close to AI is that information is passed through generations. Moreover, in domain independent algorithms like GAs, the learning process can be viewed as a search [Smith (1984)] whose main function is to exploit the knowledge embodied in the good structures so far created and the exploration of new regions in the solution space through the combining operators. All this process is performed in parallel, based on a pool of strings; this prevents the algorithm from being trapped in a local minimum and, perhaps most importantly, by allowing for movements in the search space which are nonoptimal, to reach regions which could not be reached by a conventional descent algorithm.

7.2. Genetic algorithm implementation

In order to apply a genetic algorithm to a particular problem, there are two main requirements: a (possibly) binary string representation of the solution space and an objective function which evaluates the fitness of the different point solutions. Most of the works on GAs use a binary representation [Davis (1989)], but other codes of higher cardinality have been used [Goldberg (1990)]. The fitness function can be seen as the bridge connecting the genetic algorithm to the real world problem under study. Care must be taken in order to ensure that strings with higher fitness values do correspond to good performances of the real world process. Unlike most common algorithms, GAs do not start from one individual point in the search space, but from a population of strings, usually referred to as the gene pool.

The search of the solution space in a simple genetic algorithm is performed by means of the following operators:

1. Reproduction,
2. Crossover,

3. Mutation,

4. Inversion.

In the reproduction process, strings are copied into the next generation mating pool with a probability associated with their fitness value. By assigning to the next generation a higher proportion of the highly fit strings, reproduction mimics the survival of the fittest in the natural world.

The search of the solution space is done by creating new chromosomes from old ones. The most important search process is crossover. Firstly two parents are randomly selected from the mating pool. Secondly, a point along their common length is randomly selected, and the characters of the two parent strings are swapped, thus creating two new children. Figure 7.1 presents an example of crossover in two chromosomes A^1, B^1 of length 8 with a crossover point between the third and fourth position.

Parents

$A^1 = a_1 \quad a_2 \quad a_3 \quad a_4 \quad a_5 \quad a_6 \quad a_7 \quad a_8$

$B^1 = b_1 \quad b_2 \quad b_3 \quad b_4 \quad b_5 \quad b_6 \quad b_7 \quad b_8$

Children

$A^2 = a_1 \quad a_2 \quad a_3 \quad b_4 \quad b_5 \quad b_6 \quad b_7 \quad b_8$

$B^2 = b_1 \quad b_2 \quad b_3 \quad a_4 \quad a_5 \quad a_6 \quad a_7 \quad a_8$

Fig. 7.1. The crossover operator

As can be seen, crossover does not involve any change in the actual values of the chromosome. In fact only an exchange of bits of a string takes place. This exchange of information together with reproduction is the most powerful process by which GAs perform the search of the solution space. This search is not just a simple random search because through reproduction the most promising regions of the solution space are explored.

The mutation operator randomly selects a position in the chromosome and changes the corresponding gene value, thereby destroying information. The need for mutation comes from the fact that as the less fit members of successive generations are discarded, some aspects of the genetic material could be lost forever. By performing occasional random changes in the chromosomes, GAs ensure that new parts of the search space are reached, which reproduction and crossover alone could not fully guarantee. In doing so, mutation ensures that no important features are prematurely lost, thus maintaining the mating pool diversity.

The inversion operator allows for information exchange within the chromosome. Two points are randomly selected and the elements between these two points are inverted. Figure 7.2 shows the result of the inversion operator when applied to a string of length 8, between two points corresponding respectively to the second and third, and the sixth and seven positions.

$$C^1 = c_1 \quad c_2 \quad c_3 \quad c_4 \quad c_5 \quad c_6 \quad c_7 \quad c_8$$
$$C^2 = c_1 \quad c_2 \quad c_6 \quad c_5 \quad c_4 \quad c_3 \quad c_7 \quad c_8$$

Fig. 7.2. The inversion operator

Inversion provides a means of shifting information within the chromosome and in this way reinforcing and altering the linkage between different features of the chromosome. The frequency of mutation and inversion is ususally chosen to be considerably less than the frequency of crossover. In this sense, these two operators play a secondary role in the genetic algorithm search.

Having discussed the mechanics of GAs, the question arises as to what is their strength? To answer this the concept of a schema or similarity template has to be introduced. As an example consider the schema consisting of a binary string of length 8

$$H = 1 \quad * \quad * \quad 1 \quad 0 \quad * \quad * \quad *.$$

This represents all the strings with 1 in the first position and 1 and 0 in the fourth and fifth positions respectively (* stands for a "don't care symbol"). GAs operate on these schemata in a very efficient way. For strings of length l and a binary alphabet $\{0,1\}$, there are 2^l defined strings but 3^l schemata. Also for each string there are 2^l schemata, and in a population of size n, at most $n2^l$ schemata. The different schemata can be distinguished according to their order $o(H)$ (the number of fixed positions in a schema) and to their size $\delta(H)$ (the distance between the first and last specific string position). Figure 7.3 presents some examples of schemata with their respective length and order.

Chromosome string								δ	o
0	1	1	0	1	0	0	1	7	8
*	*	*	0	1	*	*	*	1	2
0	1	1	*	*	*	*	*	2	3
0	1	*	*	*	*	*	1	7	3
*	*	*	*	*	*	*	*	0	0
0	*	*	*	*	*	*	*	0	1

Fig. 7.3. Examples of schemata

Schemata can also be understood as hyperplanes on a hypercube of dimension l [Goldberg (1989)].

In the gene pool the fittest chromosomes share some common features, the genes which make them successful chromosomes. Schemata define these shared features, and can be regarded as the building blocks of the different chromosomes. In fact, the average time span of a chromosome can be a generation, but the time span of a successful gene can be measured in many generations [Dawkins (1989)]. The fundamental role of crossover is to shuffle the different building blocks. As a result, short schemata will be more likely to survive as opposed to long schemata, which will be more prone to being interrupted. The combined action of reproduction and crossover defines the corner-stone of the theory of schemata: short, low-order, above average schemata receive exponentially increasing trials in subsequent generations.

7.3. Genetic algorithms applied to power scheduling

Power scheduling is a highly constrained problem. There are local constraints affecting the individual power units and global constraints coupling all the units. The main restrictions affecting the thermal units are minimum up and down times (once a unit is started up or closed down, it has to remain in that state for a minimum period); start up costs are dependent on the time a unit has been off and its minimum and maximum outputs. The operation of the conventional hydro units has to consider minimum and maximum reservoir levels, the natural inflows, minimum and maximum discharges, the possibility of spillage, plus the effect of the reservoir's head on the power output. The pump-storage units have the same restrictions as any conventional hydro unit, but since pumping to refill the reservoir is allowed, restrictions concerning the pumping level and the fact that there cannot be generation and pumping at the same time have to be considered. It is clear that in general this results in a mixed integer nonlinear programming problem.

The power scheduling problem is further restricted by the consideration of two global constraints concerning the demand and reserve requirements which couple all the units within the system. The total power output of all the committed units has to satisfy a given demand, and the operation of the system has to be carried out in such a way that a reserve will always be available to cope with sudden increases in the demand or the case of a unit failure. The desired aim is to minimize the operational cost of the system while satisfying the constraints [Oliveira, McKee and Coles (1991)]. This can be stated as the minimizing problem (P)

$$\begin{aligned} &\min_{\mathbf{x}} f(\mathbf{x}) \\ &\text{s.t. } \mathbf{g}(\mathbf{x}) \geq \mathbf{0} \\ &\qquad \mathbf{x} \geq \mathbf{0} \\ &\qquad \mathbf{x}_j \text{ integer}, \; j \in \mathcal{I} \end{aligned} \tag{7.3.1}$$

where $\mathbf{x}$ is an $n \times 1$ vector, and $\mathcal{I}$ denotes the index set of the variables required to be integer.

The implementation of a fitness function has to take account of the constraints on the solution. There are two basic approaches to constraints in GA's. Either penalty terms involving the constraints are included within the objective function, or a decoding mechanism is incorporated to avoid the creation of individuals which violate the constraints. Both these approaches have an associated cost in terms of efficiency and computation time. In this work, a decoder is incorporated into the algorithm to guarantee that no illegal strings are generated from the different genetic operators. The two global constraints are considered by transforming the original constrained problem (P) into an unconstrained minimization problem of the form,

$$\begin{aligned} &\min_{\mathbf{x}} \{ f(\mathbf{x}) + r_1 \Phi_1[g_1(\mathbf{x})] + r_2 \Phi_2[g_2(\mathbf{x})] \} \\ &\text{s.t. } \mathbf{x} \geq \mathbf{0} \\ &\qquad \mathbf{x}_j \text{ integer}, \; j \in \mathcal{I} \end{aligned} \tag{7.3.2}$$

with all local constraints incorporated into the genetic operators, and where Φ_1 and Φ_2 are penalty functions of the squares of the constraint violations with r_1, r_2 penalty parameters for the two global constraints $g_1(\mathbf{x})$ and $g_2(\mathbf{x})$ respectively.

7.4. Genetic operators

The implementation of the four genetic operators requires special consideration of the different characteristics imposed by the various generating units. The hydro unit presents no special problems in the string representation as its values are fairly independent of each other, except for the case where spillage occurs. The pump-storage unit and the thermal unit representations are context sensitive in the sense that a particular value may depend on the values chosen for other positions. In the pump-storage case this results from the imposition of minimum and maximum reservoir levels as well as a final value for the reservoir level. The thermal unit values are context insensitive except for minimum up and down times. A decoder, specially designed for each particular type of unit has been implemented thus ensuring that the genetic operators produce feasible strings.

Figure 7.4 presents the structure of the program developed for this particular application. Each string, chromosome, for a period of 24 hours had a length of $(n \times 24)$, i.e. each position denoting the state of each unit for the planning period and n the number of units. A real representation of the variables was implemented, implying a discretization of the continuous variables. The fitness function considered the contribution of each unit.

Due to the different nature of the power units, genetic operators were specifically designed for each unit. The mutation operator did not present special difficulties, save for the fact that in the thermal unit provision had to be made

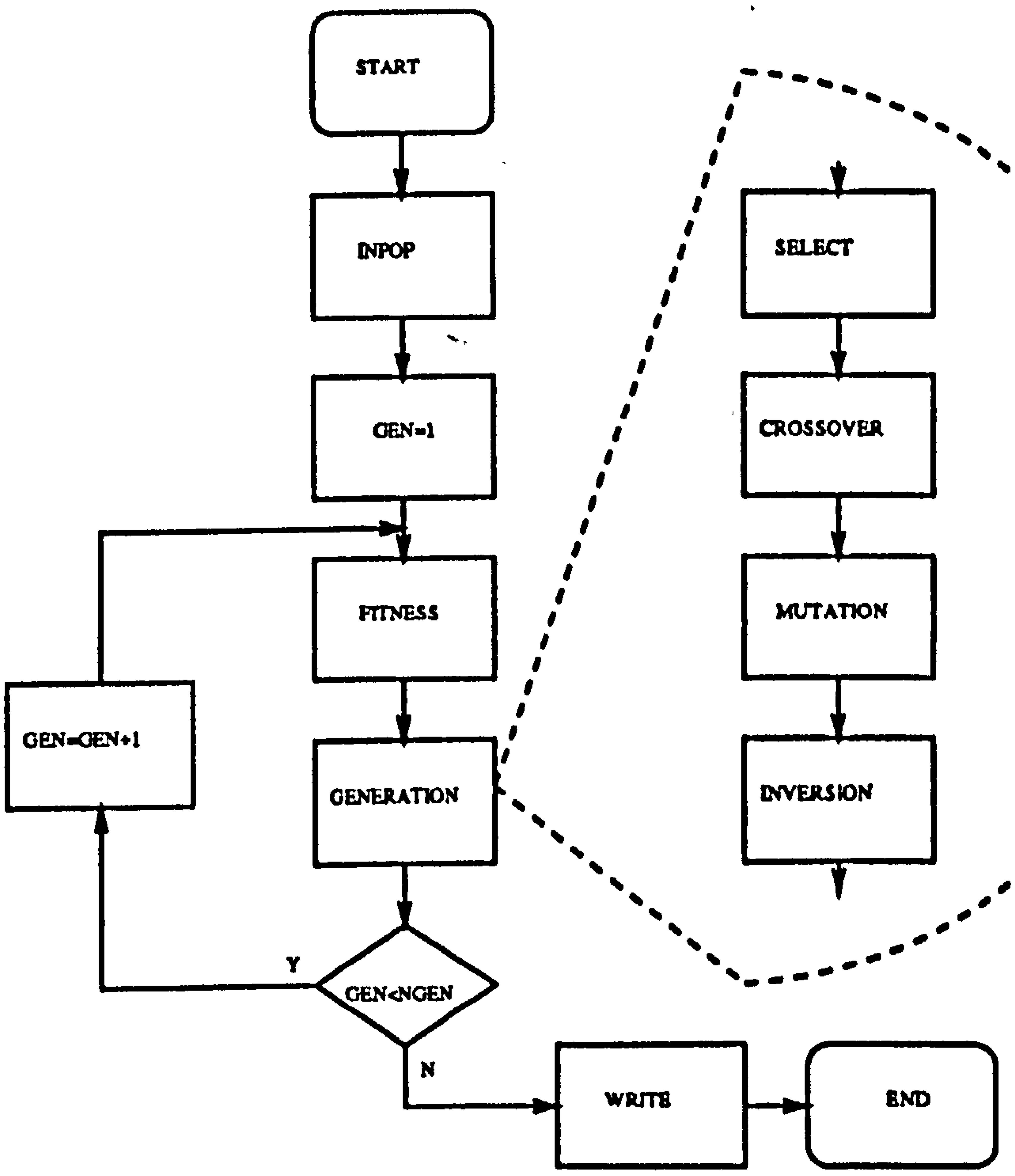


Fig. 7.4. GA Flowchart

to ensure that the minimum up and down times were respected. In the case of the pump-storage unit this operator interchanged two randomly selected points, thus always generating feasible solutions in terms of the initial and final volumes of the reservoir. An inversion operator was designed for the thermal unit in order to cope with the restrictions affecting this unit. Similarly, the crossover operator had to take into account the particular restrictions concerning each unit. In the case of the pump-storage unit, an order based crossover operator was constructed [Goldberg (1989)].

7.5. Results

Several runs were executed with different settings of the parameters. These runs allowed the tuning of the parameters, i.e. the initial population number, the probabilities of crossover, mutation and inversion, the number of generations and finally the penalty terms. Figure 7.5 shows the evolution of a run with the following parameter settings: initial population 1000, number of generations 500, probability of crossover 0.65, probability of mutation 0.01, probability of inversion 0.001. The penalty terms were increased by a factor of 10 every 50 generations. The graph presents the absolute and mean deviation from the optimal solution which has been obtained from solving the problem using a branch-and-bound approach [Oliveira, McKee and Coles (1991)]. The solution is within 2% of the optimum.

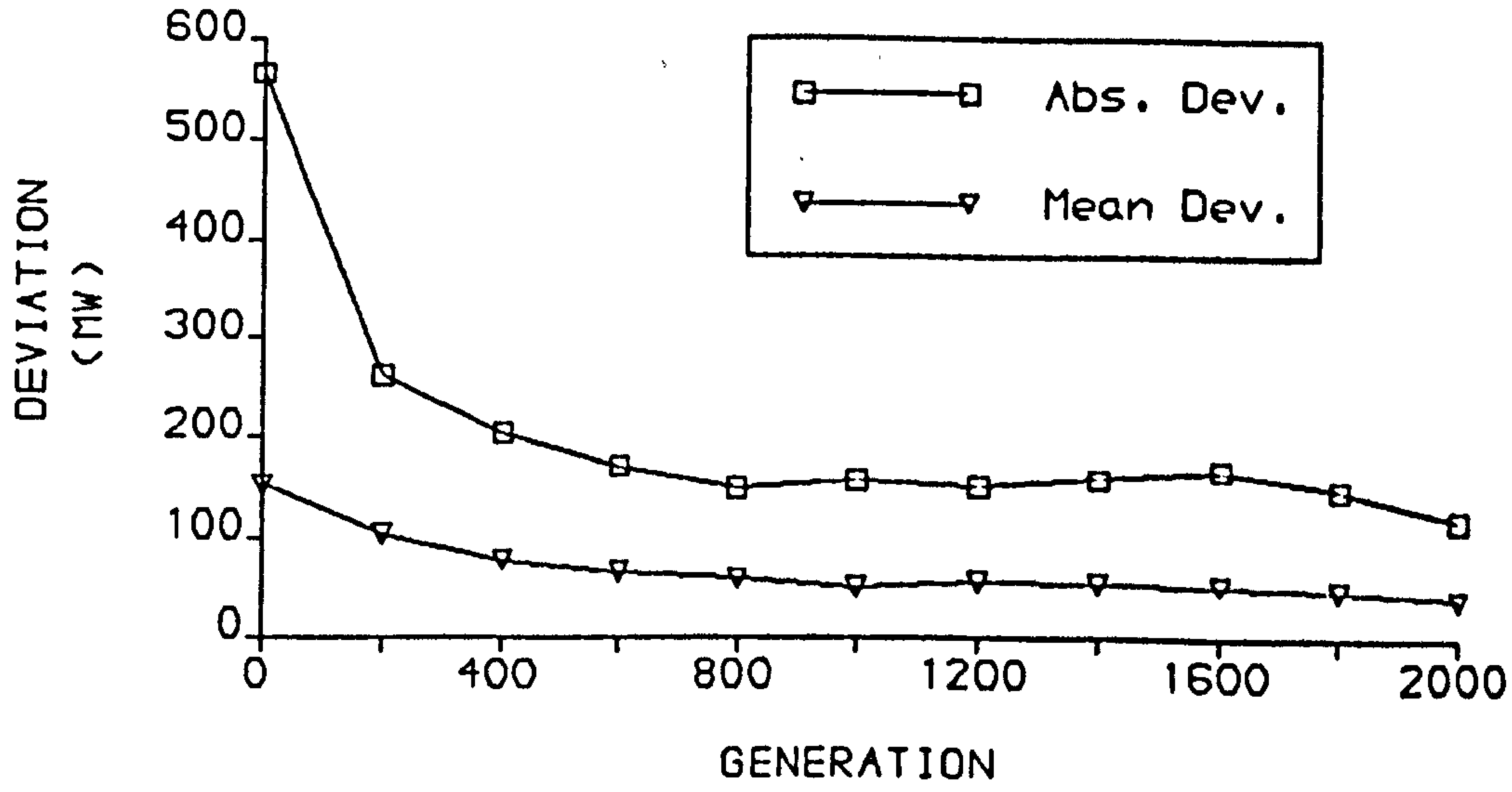


Fig. 7.5. Results for a 24 hour period

7.6. Summary

The application of genetic algorithms to a relatively simple power scheduling problem has provided some insight into their feasibility. The results show that it is possible to use a genetic search successfully on a constrained problem such as power scheduling. The strength of this algorithm lies in the fact that, unlike other algorithms, GAs search from a population of strings; the fitness function only requires an evaluation of the objective function and no other auxiliary information such as derivatives. The use of reproduction rules gives this approach a distinct advantage over straightforward random search. The main disadvantage of GAs is the computational time which, at the present state of development, is considerably more expensive than other methods. Though GAs show a very efficient behaviour in "circling" the optimal region, they are quite slow at hillclimbing.

It is envisaged that a GA dealing exclusively with the integer commitment variables coupled with an efficient linear programming solver for the continuous problem could improve on the results obtained; the consideration of simple rules to decide on the rejection of strings could further improve the algorithm since it would imply a reduced number of calls to the linear solver. One crucial point is that there is a static domain (hydro systems do not change much in 50 years) which must be solved many times. Thus, a knowledge base could be progressively built up and this could be used to deselect poor off-spring; in other words, there is the possibility for the development of genetically engineered algorithms. Furthermore, there is great scope for parallelisation.

Further work however is required to establish this approach as a efficient method for optimizing large linear and nonlinear scheduling systems.

CHAPTER 8

PURCHASING AND SELLING ELECTRICITY IN THE PRIVATE MARKET

8.1. Introduction

The Electricity Supply Industry in the U.K. has recently been restructured resulting in the introduction of competition to both energy generation and supply. Twelve private electricity companies have been set up in competition and this is likely to have a significant impact on the price of electricity to the consumer, on power station building programmes, on the environment and indeed on industry in general, through customised contracts.

Profit management rather than cost minimization is now in vogue. Furthermore, operational planning is subject to the market, or pool system as it is called. Under this system each company has the opportunity to submit bids of the prices they are prepared to sell electricity at for the next 24 hours. Those bids are currently for each half-hourly interval. Thus the pressing problem exercising the minds of the 12 Electricity Companies on a daily basis is at what price should they buy or sell their electricity.

8.2. The system

The system considered here is an example set created by Scottish Power, which emulates their own real system on a smaller scale. In this system there are three thermal stations, with two units each, including two peakers; a pondage hydro unit and a pump-storage unit. The natural inflow to the hydro unit is given as a constant value over each 24 hour period with the possibility of spillage. For convenience the water level in the reservoir and the water flow to and from the reservoir are given in units of energy and power respectively;

this is reasonable provided there are no great variations in the reservoir heads. The cost functions for the hydro and pump-storage units are dependent on the reservoir levels. Lower limits are prescribed for both generation and pumping. The demand is assumed to be known and constant over each 1 hour period, that is, it is assumed it has been obtained by time-series or other forecasting techniques.

Purchases from and sales to the pool system are restricted by the capacity of the interconnector to the south of the border.

8.3. The thermal system

The thermal system has been modelled in a similar manner to that in Chapter 2. New formulations of some of the restrictions are presented here for the sake of generality.

(a) Start-up cost per unit

$$U_i(1-\alpha_i^{t-1})\alpha_i^t, \tag{8.3.1}$$

for $i = 1,2,\ldots,I$ and for $t = 1,2,\ldots,T$,

where U_i is the start-up cost for unit i.

(b) Shut-down cost per unit

$$D_i(1-\alpha_i^t)\alpha_i^{t-1}, \tag{8.3.2}$$

for $i = 1,2,\ldots,I$ and for $t = 1,2,\ldots,T$,

where D_i is the shut-down cost for unit i.

(c) Minimum up time

$$(\alpha_i^t - \alpha_i^{t-1})(\sigma_i^{t-1} - \overline{\tau}_i) \leq 0, \tag{8.3.3}$$

$$\sigma_i^t = (\sigma_i^{t-1} + 1)\alpha_i^t, \tag{8.3.4}$$

$$\text{for } i = 1, 2, \ldots, I \text{ and for } t = 1, 2, \ldots, T,$$

where $\overline{\tau}_i$ is the minimum up time and σ_i^t is the number of time periods the unit has been on-line.

(d) Minimum down time

$$(\alpha_i^t - \alpha_i^{t-1})(\varrho_i^{t-1} - \underline{\tau}_i) \geq 0, \tag{8.3.5}$$

$$\varrho_i^t = (\varrho_i^{t-1} + 1)(1 - \alpha_i^t), \tag{8.3.6}$$

$$\text{for } i = 1, 2, \ldots, I \text{ and for } t = 1, 2, \ldots, T,$$

where $\underline{\tau}_i$ is the minimum down time and ϱ_i^t is the number of time periods the unit has been off-line.

8.4. The hydro system

The main difference from the model of Chapter 2 arises from the inclusion of costs associated with the level of the reservoir. Figure 8.1 presents an example of this relation: when the level of the reservoir is quite high, there is the possibility of running into spillage. Therefore, the cost of this hydro energy is decreased; on the other hand, if the level is too low, the cost of energy is increased since the risk of drying up the reservoir is more likely.

Thus, the running costs are

$$H_k(v_k^t)y_k^t + S_k s_k^t, \tag{8.4.1}$$

for $k = 1, 2, \ldots, K$ and for $t = 1, 2, \ldots, T$,

where $H_k(v_k^t)$ and S_k are respectively the values of the discharge and spillage per MW.

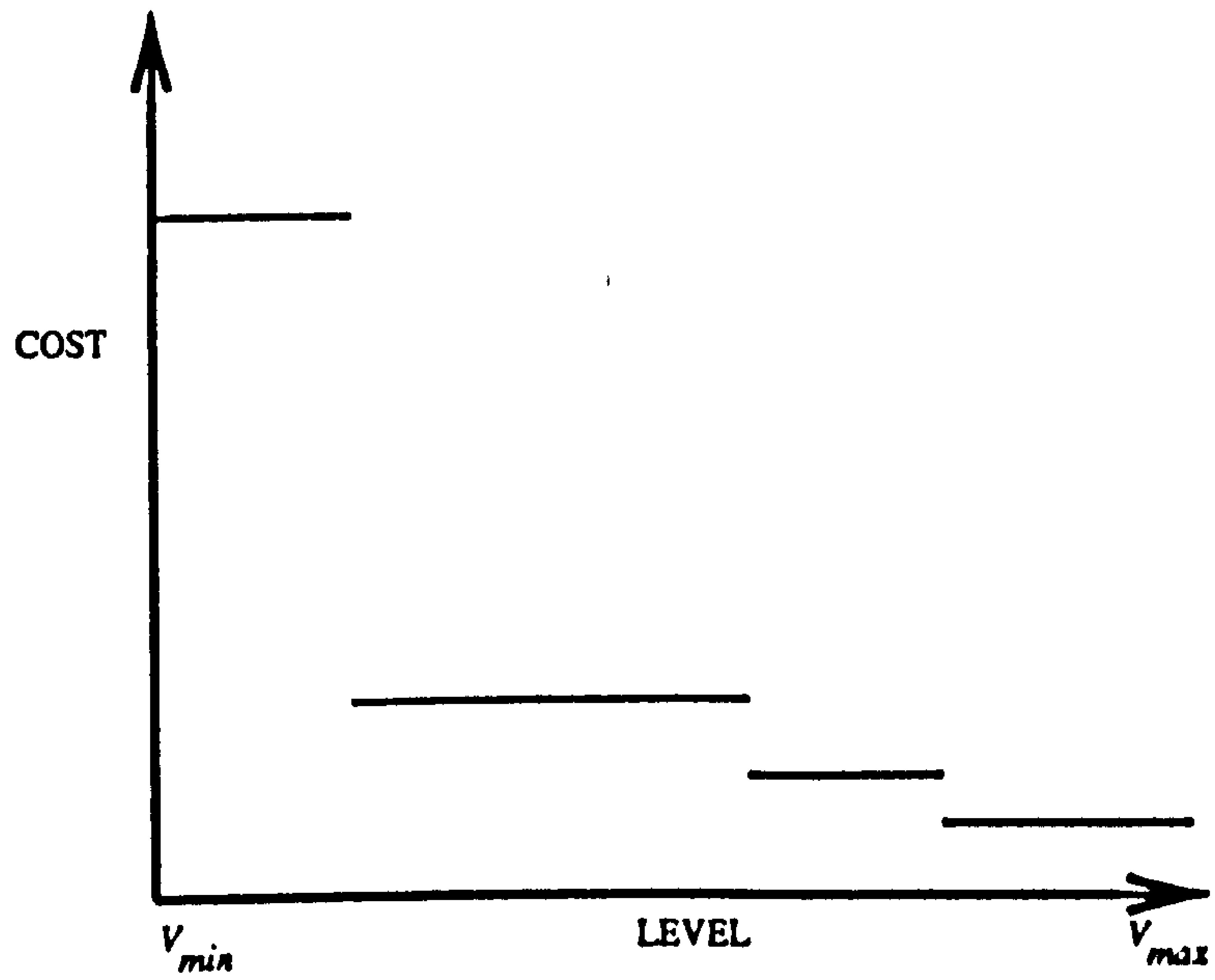


Fig. 8.1. Cost vs level for hydro unit

8.5. The pump-storage system

The pump-storage unit can operate one or more turbines/pumps at the same time. In this particular system there are J turbines/pumps. When generating, the power output is a continuous function between the lower and upper limits; the power output

$$q_l^t \in [\underline{q}_l, j\bar{q}_l] \quad \text{if} \quad \mu_l^t = 1 \tag{8.5.1}$$

for $l = 1, 2, \ldots, L$, and for $t = 1, 2, \ldots, T$.

where $j = 1,2,\ldots,J$ is the number of available turbines for the scheduling period. In pumping mode, the water flow is only allowed to vary in discrete steps $\underline{p}_l = \overline{p}_l$, depending on the number of pumps in operation,

$$p_l^t \in \{\underline{p}_l, \ldots, j\underline{p}_l\} \quad \text{if} \quad \nu_l^t = 1 \tag{8.5.2}$$

$$\text{for } l = 1,2,\ldots,L, \text{ and for } t = 1,2,\ldots,T.$$

The operating costs of the pump-storage unit are also a function of the level of the reservoir. Similar considerations to the ones made for the hydro unit can be made for the cost associated with the level of the reservoir. Figure 8.2 depicts this relation.

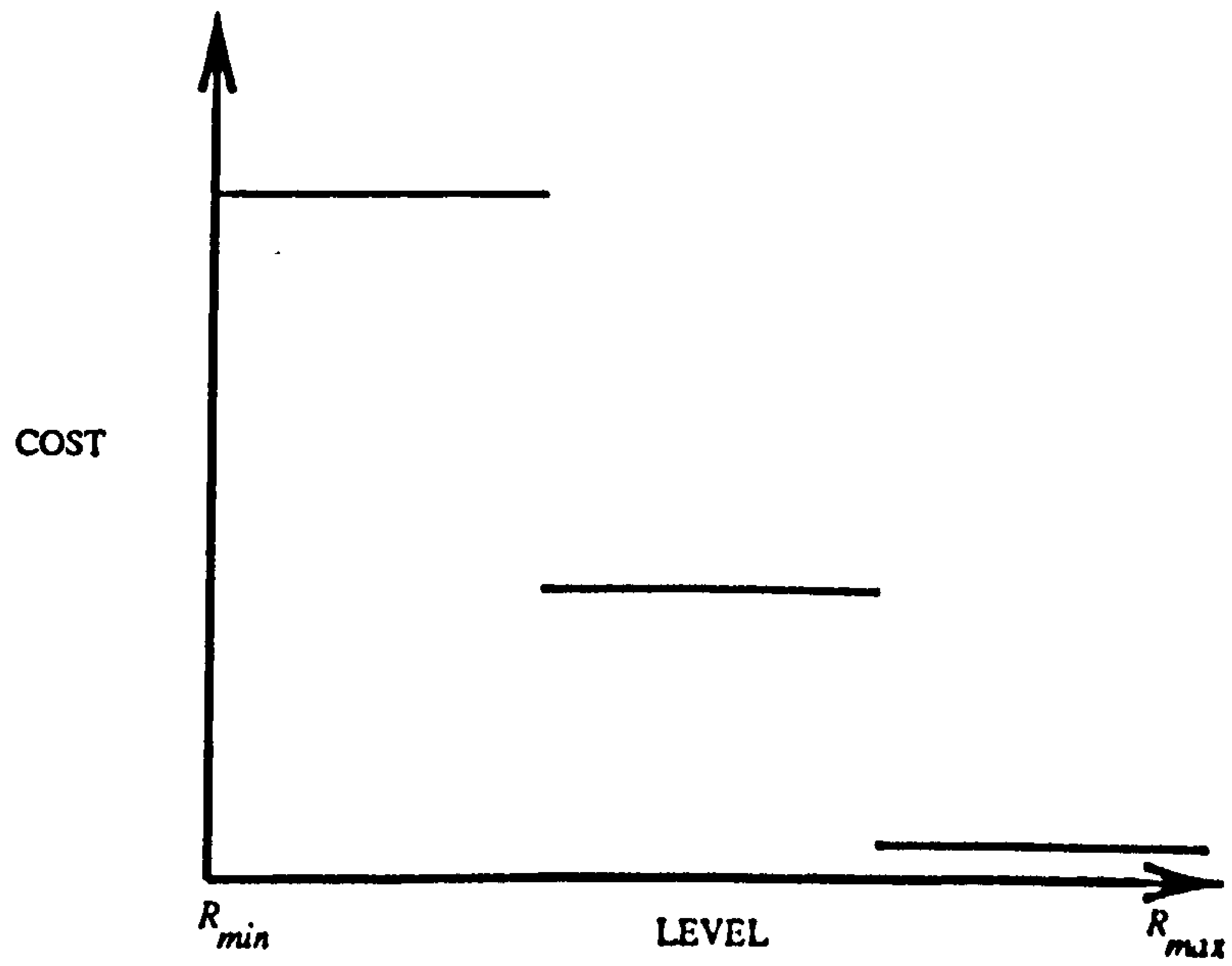


Fig. 8.2. Cost vs level for pump-storage unit

The operating costs are

$$G_l(r_l^t)q_l^t - P_l(r_l^t)p_l^t, \tag{8.5.3}$$

$$\text{for } l = 1,2,\ldots,L \text{ and for } t = 1,2,\ldots,T,$$

where $G_l(r_l^t)$ is the value of the equivalent quantity of water used for generating each unit of power while $P_l(r_l^t)$ is the value of the water pumped per MW to the storage reservoir.

8.6. Purchase and sale

In every time period t there is the possibility of either purchasing from, or selling power to, the market or pool system. The pool system does not permit the simultaneous purchase and sale of electricity.

Let z_p^t be the power purchased for period t. The amount purchased has to be within the permissible limits $[0, \bar{z}_p]$, where $\bar{z}_p$ is the maximum amount that is allowed to be purchased in any one time period. The cost associated with purchasing is

$$m_p^t z_p^t \tag{8.6.1}$$

$$\text{for } t = 1, 2, \ldots, T,$$

where m_p^t is the purchase price during period t dictated by the pool system.

Let z_s^t denote the amount of power sold during period t. The selling limits are $[0, \bar{z}_s]$, where $\bar{z}_s$ is the maximum amount that is allowed to be sold in time period t. Since selling implies a revenue it is convenient to treat it as a negative cost, viz

$$-m_s^t z_s^t \tag{8.6.2}$$

$$\text{for } t = 1, 2, \ldots, T,$$

where m_s^t is the selling price during period t.

It could be argued that m_p^t and m_s^t should be one and the same, the pool price. However, to sell energy during a low demand period does not have the same risk as for a period of high demand. This difference can thus be perceived as a security, and may therefore not be constant in every time period.

8.7. The demand and reserve constraints

Clearly, in reality the power output has to satisfy a stochastic demand. Also, if one unit breaks down it is not possible to start an uncommitted thermal unit immediately and so a certain amount of reserve, known as spinning reserve, should be available. Therefore,

$$\begin{aligned} &\sum_{i=1}^{I} x_i^t + \sum_{k=1}^{K} y_k^t + \sum_{l=1}^{L}(q_l^t - \Theta_l p_l^t) + z_p^t - z_s^t \geq d^t \\ &\sum_{i=1}^{I} \overline{x}_i \alpha_i^t + \sum_{k=1}^{K} \overline{y}_k + \sum_{l=1}^{L}(\overline{q}_l \mu_l^t - \Theta_l p_l^t) + z_p^t - z_s^t \geq d^t + R \end{aligned} \tag{8.7.1}$$

$$\text{for } t = 1, \ldots, T$$

where Θ_l is the inverse of the thermodynamic efficiency of the pumping process, d^t is the demand and R the required reserve.

8.8. The mixed integer model

In summary the scheduling task can be modelled as the mixed integer non-linear programming problem (P)

$$\begin{aligned} \min_{\alpha_i^t, x_i^t, y_k^t, s_k^t, v_k^t, \mu_l^t, q_l^t, \nu_l^t, p_l^t, r_l^t, z_p^t, z_s^t} \sum_{t=1}^{T} \Big[&\sum_{i=1}^{I} (U_i(1-\alpha_i^{t-1})\alpha_i^t + F_i \alpha_i^t \\ &\quad + V_i x_i^t + D_i(1-\alpha_i^t)\alpha_i^{t-1}) \\ &+ \sum_{k=1}^{K} (H_k(v_k^t) y_k^t + S_k s_k^t) \\ &+ \sum_{l=1}^{L} (G_l(r_l^t) q_l^t - P_l(r_l^t) p_l^t) \\ &+ m_p^t z_p^t - m_s^t z_s^t \Big] \end{aligned} \tag{8.8.1}$$

subject to the demand and reserve constraints and the constraints affecting the individual units.

8.9. Lagrangian relaxation in power systems

In power scheduling the most suitable restrictions to be relaxed are the global ones, that is, the demand and reserve constraints. The inclusion of these two constraints gives the following Lagrangian problem

$$
\begin{aligned}
&\Phi(\lambda_1, \lambda_2) = \\
&\min_{\alpha_i^t, x_i^t, y_k^t, s_k^t, v_k^t, \mu_l^t, q_l^t, \nu_l^t, p_l^t, r_l^t, z_p^t, z_s^t} \Bigg\{ \sum_{t=1}^{T} \Bigg[\sum_{i=1}^{I} (U_i(1 - \alpha_i^{t-1})\alpha_i^t + F_i \alpha_i^t \\
&\qquad + V_i x_i^t + D_i(1 - \alpha_i^t)\alpha_i^{t-1}) \\
&\qquad + \sum_{k=1}^{K} (H_k(v_k^t) y_k^t + S_k s_k^t) \\
&\qquad + \sum_{l=1}^{L} (G_l(r_l^t) q_l^t - P_l(r_l^t) p_l^t \\
&\qquad + m_p^t z_p^t - m_s^t z_s^t) \Bigg] \\
&\quad + \sum_{t=1}^{T} \Bigg[\lambda_1^t \Big(d^t - \sum_{i=1}^{I} x_i^t - \sum_{k=1}^{K} y_k^t \\
&\qquad - \sum_{l=1}^{L} (q_l^t - \Theta_l p_l^t) - z_p^t + z_s^t \Big) \Bigg] \\
&\quad + \sum_{t=1}^{T} \Bigg[\lambda_2^t \Big(d^t + R - \sum_{i=1}^{I} \overline{x}_i \alpha_i^t - \sum_{k=1}^{K} \overline{y}_k \\
&\qquad - \sum_{l=1}^{L} (\overline{q}_l \mu_l^t - \Theta_l p_l^t) - z_p^t + z_s^t \Big) \Bigg] \Bigg\}.
\end{aligned}
\tag{8.9.1}
$$

In a more condensed form which emphazises the problem decomposition, the Lagrangian formulation can be written as

$$\begin{aligned}\Phi(\lambda_1,\lambda_2) = \sum_{i=1}^{I}\Phi_i^I(\lambda_1^t,\lambda_2^t) + \sum_{k=1}^{K}\Phi_k^{II}(\lambda_1^t,\lambda_2^t) + \sum_{l=1}^{L}\Phi_l^{III}(\lambda_1^t,\lambda_2^t)\\ + \Phi_p^{IV}(\lambda_1^t,\lambda_2^t) + \Phi_s^V(\lambda_1^t,\lambda_2^t) + \sum_{t=1}^{T}\left[\lambda_1^t d^t + \lambda_2^t(d^t + R)\right]\end{aligned} \quad (8.9.2)$$

where $\Phi_i^I(\lambda_1,\lambda_2)$ represents the thermal units subproblems; $\Phi_k^{II}(\lambda_1,\lambda_2)$ the conventional hydro units subproblems; $\Phi_l^{III}(\lambda_1,\lambda_2)$ the pumped-storage units subproblems; $\Phi_p^{IV}(\lambda_1,\lambda_2)$ the purchase subproblem, and $\Phi_s^V(\lambda_1,\lambda_2)$ the selling subproblem. Each one of these subproblems is locally constrained by the operating characteristics of the individual units. Clearly, given the values of the Lagrange multipliers, the subproblems can be solved independently of each other. Thus, the dual problem (D) can be stated concisely as

$$\Phi^* = \max \Phi(\lambda_1,\lambda_2) \quad (8.9.3)$$

subject to

$$\lambda_1 \geq 0,\ \lambda_2 \geq 0.$$

8.10. The subproblems and the master problem

The subproblems are solved using dynamic programming and the subgradient method is used to generate new multipliers (see Chapter 5). The heuristic developed for the problem of Scottish Hydro-Electric (see Chapter 5, Section 5.9) had to be changed in order to incorporate purchases and sales. The logic behind the heuristic remains the same in what regards the adjustment of the dual solution to construct a primal feasible solution. However, since the global cost function exhibits discontinuities at points where another unit is brought on-line, at every time period a test was made to verify if the sale revenue was greater than the cost to supply the market.

8.11. Results

The results obtained are listed in Table 8-1: the total cost (£), the percentage difference between the dual and primal solutions, the thermal, pump-storage, hydro, purchase and sale costs (£) for the different problems tested: (1) is the solution for the given demand and reserve, using only thermal units; in (2) a pump-storage unit is added; (3) depicts the previous set up plus a hydro unit; in (4) the possibility of purchasing and selling is considered for the system comprising all the thermal and hydro units.

Table 8-1
Numerical Results

Set	Cost	% Dif	Ther	Pump	Hyd	Purc	Sale
(1)	$1.82E5$	1.3	$1.82E5$	-	-	-	-
(2)	$1.81E5$	1.6	$1.78E5$	$2.30E3$	-	-	-
(3)	$1.80E5$	1.8	$1.71E5$	0.0	$8.70E3$	-	-
(4)	$1.58E5$	0.8	$2.33E5$	0.0	$8.06E3$	0.0	$-8.24E4$

8.12. Summary

The special structure which results from this implementation, with a master problem coordinating several subproblems, is suitable for a parallel implementation; however, if the problems are not well balanced in terms of computational time (the hydro and pump-storage subproblems are the most time consuming), the maximum speed-up that can be obtained is conditioned by the slowest subproblem. As it can be seen from the results presented in Table 8-1, the possibility of purchasing and selling significantly improves the economic operation of the system.

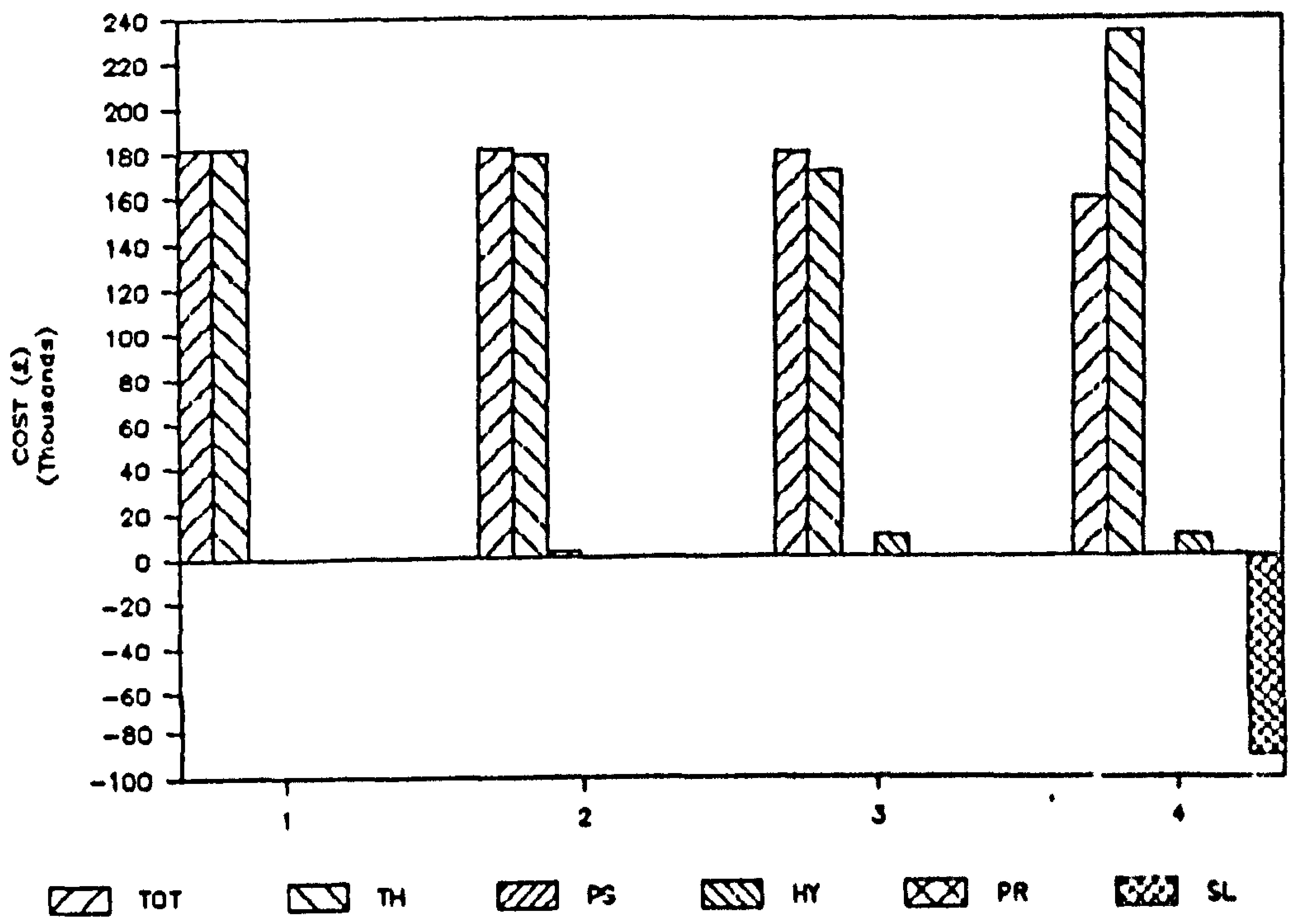


Fig. 8.3. Total cost

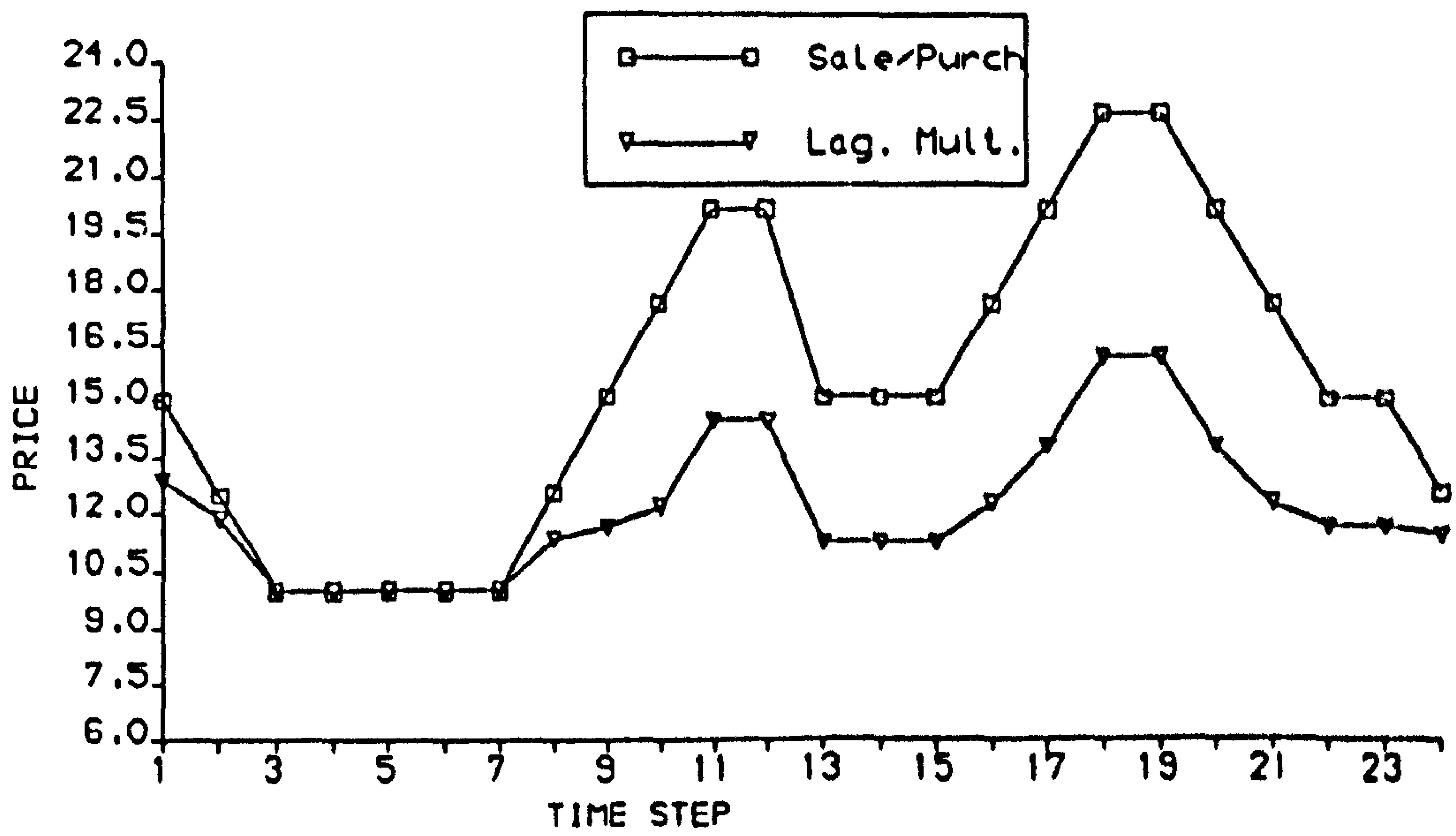


Fig. 8.4. Lagrange multipliers vs pool prices

The Lagrange multipliers can be perceived as shadow costs in the sense that they represent the costs required to satisfy the demand and reserve constraints. This feature, given the recent privatization of the generating boards in the UK, makes this implementation a valuable tool as it effectively associates a cost to the energy, whether from thermal, hydro or pump-storage units.

CHAPTER 9

THE DUALITY GAP IN POWER SCHEDULING

9.1. Introduction

The occurrence of a duality gap was referred to in Chapter 4, and this was related to the convexity of the primal problem. In power scheduling problems, the likely existence of a duality gap is used to terminate the iterative optimization process when the relative duality gap (Chapter 5, Section 5.2) is below a given bound. Bertsekas and Sandell (1982), Lauer *et al.* (1982) and Bertsekas *et al.* (1983) showed that, for thermal systems, as the number of thermal units i goes to infinity, the relative duality gap decreases to zero,

$$\lim_{i\to\infty} \frac{v(P_i) - v(D_i)}{v(P_i)} = 0$$

where $v(P_i)$ and $v(D_i)$ are, respectively, the optimal values of the primal and dual problems. However, this measure is only true in the limit and might not be verified for small systems. A survey of other contributions to the power scheduling problem [Merlin and Sandrin (1983), Shaw and Bertsekas (1985), Zhuang and Galiana (1988), Bard (1988) Aoki *et al.* (1989)] indicates that these authors have relied on this result as a termination criterion, and on the experimental observation of very small gaps, typically less than 1%.

The results from the Scottish Power model (Chapter 8) indicate a relative duality gap which is always above 1%. This fact seems to contradict previous results. However, some results based on the work of Geoffrion (1974) are presented for the case of thermal systems which explain the larger than expected gaps obtained for the Scottish Power model.

Consider the following minimization problem

$$\min_{\mathbf{x}} \mathbf{cx}$$
$$\text{s.t. } A\mathbf{x} \geq \mathbf{b}$$
$$\mathbf{x} \geq \mathbf{0}$$
$$\mathbf{x}_j \text{ integer}, j \in \mathcal{I}$$

where $\mathbf{x}$ is $n \times 1$, $\mathbf{b}$ is $(m+k) \times 1$, $\mathbf{c}$ is $1 \times n$, A has conformable dimensions and $\mathcal{I}$ denotes the index set of the variables required to be integer. This problem is equivalent to the following one, where the constraints have been partitioned in some convenient form,

(P)

$$\min_{\mathbf{x}} \mathbf{cx}$$
$$\text{s.t. } A_1\mathbf{x} \geq \mathbf{b}_1$$
$$A_2\mathbf{x} \geq \mathbf{b}_2$$
$$\mathbf{x} \geq \mathbf{0}$$
$$\mathbf{x}_j \text{ integer}, j \in \mathcal{I}$$

with

$$A = \left[A_1 : A_2\right]', \mathbf{b} = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix}.$$

Here $A_1\mathbf{x} \geq \mathbf{b}_1$ is identified as the set of m 'complicating' constraints, with $\mathbf{b}_1$ $m \times 1$ and $\mathbf{b}_2$ $k \times 1$.

The Lagrangian relaxation of (P) with respect to $A_1\mathbf{x} \geq \mathbf{b}_1$ and a conformable nonnegative vector λ is

(PR_λ)

$$\begin{aligned}&\min_{\mathbf{x}}\{\mathbf{c}\mathbf{x}+\lambda'(\mathbf{b}_1-A_1\mathbf{x})\}\\ &\text{s.t. } A_2\mathbf{x}\geq\mathbf{b}_2\\ &\quad\mathbf{x}\geq 0\\ &\quad\mathbf{x}_j \text{ integer}, j\in\mathcal{I}\\ &\quad\lambda\geq 0.\end{aligned}$$

The optimal choice for λ is given by the optimal solution to the concave problem,

$$(D) \qquad \max_{\lambda\geq 0}(PR_\lambda)$$

which coincides with the Lagrangian dual of (P) with respect to $A_1\mathbf{x}\geq\mathbf{b}_1$. The following relaxation of (P)

(P^*)

$$\begin{aligned}&\min_{\mathbf{x}}\mathbf{c}\mathbf{x}\\ &\text{s.t. } A_1\mathbf{x}\geq\mathbf{b}_1\\ &\quad\mathbf{x}\in Co\{\mathbf{x}\geq 0: A_2\mathbf{x}\geq\mathbf{b}_2,\\ &\qquad\mathbf{x}_j \text{ integer}, j\in\mathcal{I}\}\end{aligned}$$

where Co denotes the convex hull of a set, is related to the problem (D) since they are linear programming duals. An optimal multiplier vector corresponding to $A_1\mathbf{x}\geq\mathbf{b}_1$ will be denoted by λ^* when (P^*) has a finite optimal value. Let $(\overline{P})$ be the linear programming relaxation of (P), where the integer variables are treated as continuous.

Theorem [Geoffrion (1974)]

a) The following inclusions and inequalities hold

$$\mathcal{F}(\overline{P}) \supseteq \mathcal{F}(P^*) \supseteq \mathcal{F}(P) \qquad \mathcal{F}(PR_\lambda) \supseteq \mathcal{F}(P)$$

$$v(\overline{P}) \leq v(P^*) \leq v(P) \qquad v(PR_\lambda) \leq v(P).$$

b) If, for a given λ, a vector $\mathbf{x}$ satisfies the three conditions

i) $\mathbf{x}$ is optimal in (PR_λ),

ii) $A_1\mathbf{x} \geq \mathbf{b}_1$,

ii) $\lambda'(\mathbf{b}_1 - A_1\mathbf{x}) = 0$,

then $\mathbf{x}$ is an optimal solution of (P). If $\mathbf{x}$ satisfies i) and ii) but not iii), then $\mathbf{x}$ is an ϵ-optimal solution of (P) with $\epsilon = \lambda'(A_1\mathbf{x} - \mathbf{b}_1)$.

c) If (P^*) is feasible, then

$$v(D) \equiv \max_{\lambda \geq 0} v(PR_\lambda) = v(PR_{\lambda^*}) = v(P^*).$$

The last statement of the theorem can be proved by noticing that,

$$\begin{aligned} v(PR_\lambda) = \Big[& \min_{\mathbf{x}} \{\mathbf{cx} + \lambda'(\mathbf{b}_1 - A_1\mathbf{x})\} \\ & \text{s.t. } \mathbf{x} \in Co\{\mathbf{x} \geq 0 : A_2\mathbf{x} \geq \mathbf{b}_2, \\ & \qquad \mathbf{x}_j \text{ integer}, j \in \mathcal{I}\} \Big] \end{aligned}$$

because the minimum value of a linear function over any compact set is not changed if the set is replaced by its convex hull.

Integrality Property

The optimal value of (PR_λ) is not altered by dropping the integrality conditions on its variables,

$$v(PR_\lambda) = v(\overline{PR}_\lambda) \text{ for all } \lambda \geq 0.$$

In other words, if every corner of a convex polyhedron in an n-space has all integral coordinates, then this polyhedron has the integrality property.

Theorem [Geoffrion (1974)]

Let $(\overline{P})$ be feasible and (PR_λ) have the Integrality Property. Then (P^*) is feasible and

$$v(\overline{P}) = v(PR_\lambda) = v(D) = v(PR_{\lambda^*}) = v(P^*).$$

Proof

$$\begin{aligned}
v(\overline{P}) &= \max_{\lambda \geq 0} v(\overline{PR}_\lambda) && \text{(by duality)} \\
&= \max_{\lambda \geq 0} v(PR_\lambda) && \text{(by the Integrality Property)} \\
&= \max_{\lambda \geq 0} \Big[\min_{\mathbf{x}} \{\mathbf{cx} + \lambda'(\mathbf{b}_1 - A_1\mathbf{x})\} \\
&\qquad \text{s.t. } \mathbf{x} \in Co\{\mathbf{x} \geq 0 : A_2\mathbf{x} \geq \mathbf{b}_2, \\
&\qquad\qquad \mathbf{x}_j \text{ integer}, j \in \mathcal{I}\}\Big] \\
&= v(P^*) && \text{(by duality).}
\end{aligned}$$

9.2. The matrix structure in power scheduling

Consider the following problems, for the thermal case,

(P^1)

$$\min_{\alpha_i^t, x_i^t} \left\{ \sum_{t=1}^{T} \left[\sum_{i=1}^{I} (F_i \alpha_i^t + V_i x_i^t) \right] \right\}$$

$$\text{s.t.} \sum_{i=1}^{I} x_i^t \geq d^t, \qquad t = 1, \ldots, T$$

$$\sum_{i=1}^{I} \overline{x}_i \alpha_i^t \geq d^t + R, \qquad t = 1, \ldots, T$$

$$\underline{x}_i \alpha_i^t \leq x_i^t \leq \overline{x}_i \alpha_i^t \qquad i = 1, \ldots, I, \quad t = 1, \ldots, T$$

$$0 \leq \alpha_i^t \leq 1, \text{ integer}, \qquad i = 1, \ldots, I, \quad t = 1, \ldots, T$$

and

$(PR^1_{\lambda_1, \lambda_2})$

$$\min_{\alpha_i^t, x_i^t} \left\{ \sum_{t=1}^{T} \left[\sum_{i=1}^{I} (F_i \alpha_i^t + V_i x_i^t) \right] + \sum_{t=1}^{T} \left[\lambda_1^t (d^t - \sum_{i=1}^{I} x_i^t) \right] + \sum_{t=1}^{T} \left[\lambda_2^t (d^t + R - \sum_{i=1}^{I} \overline{x}_i \alpha_i^t) \right] \right\}$$

$$\text{s.t. } \underline{x}_i \alpha_i^t \leq x_i^t \leq \overline{x}_i \alpha_i^t \qquad i = 1, \ldots, I, \quad t = 1, \ldots, T$$

$$0 \leq \alpha_i^t \leq 1, \text{ integer}, \qquad i = 1, \ldots, I, \quad t = 1, \ldots, T$$

Hoffman and Kruskal(1956) showed that for the polyhedron,

$$Q(\mathbf{b},\mathbf{c}) = \{\mathbf{x} | A\mathbf{x} \geq \mathbf{b}, \mathbf{x} \geq \mathbf{c}\}$$

to have the integrality property, where A, $\mathbf{b}$, and $\mathbf{c}$ are integral and A is fixed, the matrix A has to be unimodular [A matrix A is said to have the unimodular property if every minor determinant of A equals 0, +1, or −1]. This is a very strong condition which is not satisfied in the problem $(PR^1_{\lambda_1,\lambda_2})$, since a necessary condition for unimodularity is that the entries can only be 0, +1, or −1, while

$$A_{PR^1_\lambda} = \begin{pmatrix} \overline{x}_1 & -1 & 0 & 0 & \dots & 0 \\ -\underline{x}_1 & 1 & 0 & 0 & \dots & 0 \\ -1 & 0 & 0 & 0 & \dots & 0 \\ 1 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & \overline{x}_2 & -1 & \dots & 0 \\ 0 & 0 & -\underline{x}_2 & 1 & \dots & 0 \\ 0 & 0 & -1 & 0 & \dots & 0 \\ 0 & 0 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \dots & 0 \end{pmatrix}$$

However, a reformulation of the problem by scaling x_i^t produces a unimodular matrix. Setting

$$x_i^t = \underline{x}_i \alpha_i^t + (\overline{x}_i - \underline{x}_i) y_i^t$$

then

(P^2)

$$\min_{\alpha_i^t, y_i^t} \left\{ \sum_{t=1}^{T} \left[\sum_{i=1}^{I} [(F_i + V_i \underline{x}_i)\alpha_i^t + V_i(\overline{x}_i - \underline{x}_i) y_i^t] \right] \right\}$$

$$\text{s.t. } \sum_{i=1}^{I} \left[\underline{x}_i \alpha_i^t + (\overline{x}_i - \underline{x}_i) y_i^t \right] \geq d^t, \qquad t = 1, \dots, T$$

$$\sum_{i=1}^{I} \overline{x}_i \alpha_i^t \geq d^t + R, \qquad t = 1, \dots, T$$

$$0 \le y_i^t \le \alpha_i^t \qquad i = 1,\ldots,I, \quad t = 1,\ldots,T$$

$$0 \le \alpha_i^t \le 1, \text{ integer,} \qquad i = 1,\ldots,I, \quad t = 1,\ldots,T$$

and

$(PR^2_{\lambda_1,\lambda_2})$

$$\min_{\alpha_i^t, y_i^t} \left\{ \sum_{t=1}^{T} \left[\sum_{i=1}^{I} [(F_i + V_i \underline{x}_i)\alpha_i^t + V_i(\overline{x}_i - \underline{x}_i)y_i^t] \right] + \sum_{t=1}^{T} \left[\lambda_1^t \left(d^t - \sum_{i=1}^{I} (\underline{x}_i \alpha_i^t + (\overline{x}_i - \underline{x}_i)y_i^t) \right) \right] + \sum_{t=1}^{T} \left[\lambda_2^t (d^t + R - \sum_{i=1}^{I} \overline{x}_i \alpha_i^t) \right] \right\}$$

$$\text{s.t.} \quad 0 \le y_i^t \le \alpha_i^t \qquad i = 1,\ldots,I, \quad t = 1,\ldots,T$$

$$0 \le \alpha_i^t \le 1, \text{ integer,} \quad i = 1,\ldots,I, \quad t = 1,\ldots,T$$

and the underlying matrix has now the form

$$A_{PR^2_\lambda} = \begin{pmatrix} 1 & -1 & 0 & 0 & \ldots & 0 \\ -1 & 1 & 0 & 0 & \ldots & 0 \\ -1 & 0 & 0 & 0 & \ldots & 0 \\ 1 & 0 & 0 & 0 & \ldots & 0 \\ 0 & 0 & 1 & -1 & \ldots & 0 \\ 0 & 0 & -1 & 1 & \ldots & 0 \\ 0 & 0 & -1 & 0 & \ldots & 0 \\ 0 & 0 & 1 & 0 & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \ldots & 0 \end{pmatrix}$$

However, for the polyhedron

$$P(\mathbf{b}) = \{\mathbf{x} | A\mathbf{x} \ge \mathbf{b}\}$$

to have the integrality property, Hoffman and Kruskal (1956) show that a weaker condition is required. For any set S of rows of A, let

$$
gcd(S) = \begin{cases} 0, & \text{if each minor determinant in } S\text{, which has as many rows as } S\text{, equals 0;} \\ gcd, & \text{greatest common divisor of all those minor determinants in } S \text{ which have as many rows as } S. \end{cases}
$$

If r is the rank of A, then for every set S of r linearly independent rows of A, $gcd(S) = 1$, and the polyhedron has the integrality property. Thus, it can be seen that the matrix associated with $(PR^1_{\lambda_1,\lambda_2})$ has the integrality property, and

$$
v(\overline{P}^1) = v(D^1) = v(P^{1*}).
$$

This last condition associated with the polyhedron can be depicted for a single thermal unit as,

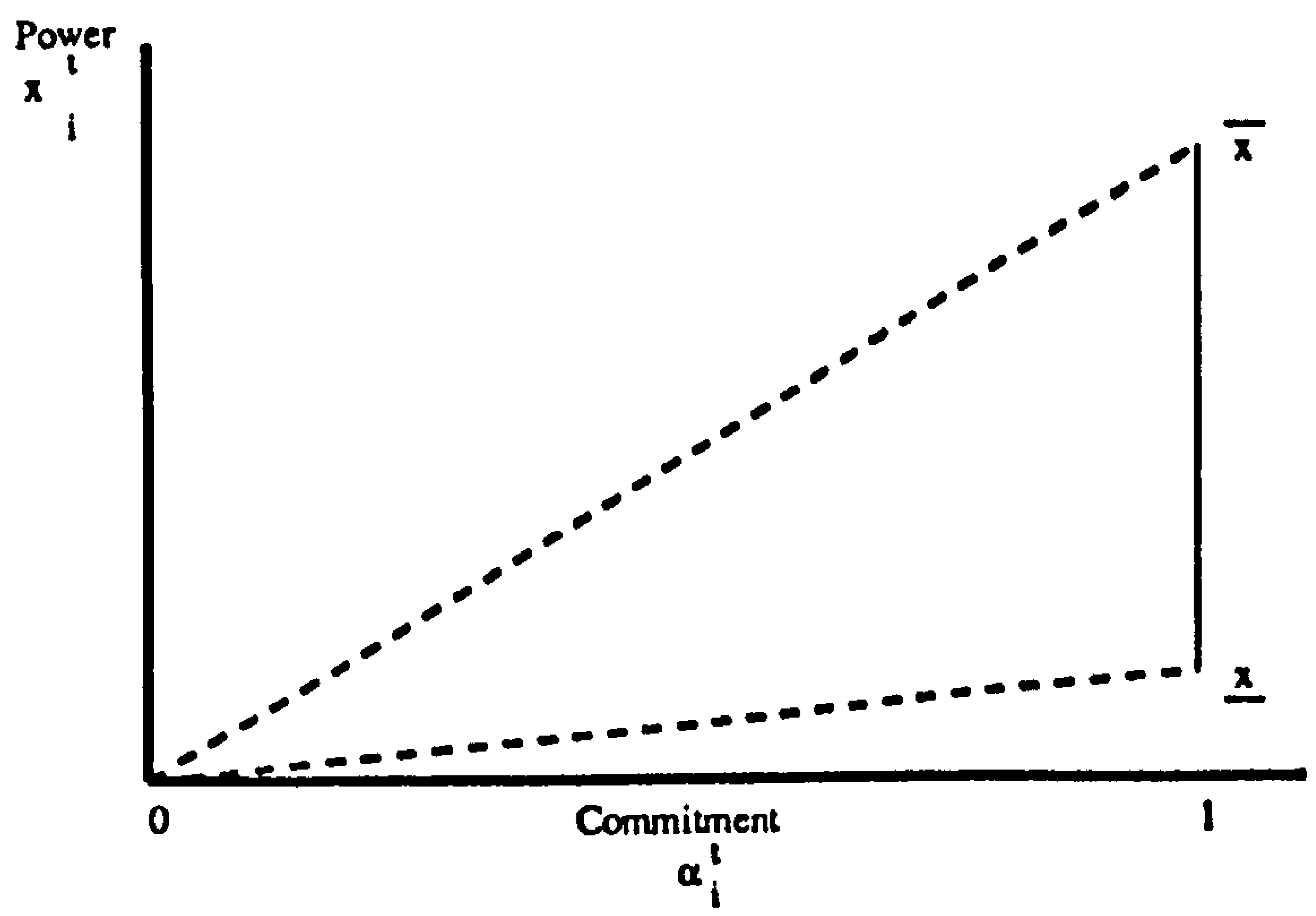


Fig. 9.1. Thermal unit solution space

9.3. The duality gap in power scheduling

The perturbation function associated with (P) is defined by

$$\Phi(\mathbf{y}) \equiv \Big[\inf_{\mathbf{x}} \mathbf{c}\mathbf{x} \quad \text{s.t. } A_1\mathbf{x} \geq \mathbf{b}_1 - \mathbf{y}, \quad A_2\mathbf{x} \geq \mathbf{b}_2, \quad \mathbf{x} \geq 0, \quad \mathbf{x}_j \text{ integer}, \ j \in \mathcal{I} \Big]$$

Theorem [Geoffrion (1974)]

The perturbation function Φ^* associated with (P^*) is precisely the lower convex envelope of the perturbation function Φ associated with (P) (Φ^* is the convex hull of the epigraph of Φ).

$$Epi\big[\Phi^*\big] = Co\Big\{Epi\big[\Phi\big]\Big\}.$$

For the power scheduling problem,

$$\Phi(\mathbf{y}) = \Bigg[\inf_{\alpha_i^t, x_i^t} \Bigg\{ \sum_{t=1}^{T} \Bigg[\sum_{i=1}^{I} (F_i\alpha_i^t + V_i x_i^t) \Bigg] \Bigg\}$$

$$\text{s.t. } \sum_{i=1}^{I} x_i^t \geq d^t - y_1, \qquad t = 1, \ldots, T$$

$$\sum_{i=1}^{I} \overline{x}_i \alpha_i^t \geq d^t + R - y_2, \qquad t = 1, \ldots, T$$

$$\underline{x}_i \alpha_i^t \leq x_i^t \leq \overline{x}_i \alpha_i^t \qquad i = 1, \ldots, I, \quad t = 1, \ldots, T$$

$$0 \leq \alpha_i^t \leq 1, \text{ integer}, \qquad i = 1, \ldots, I, \quad t = 1, \ldots, T \Bigg]$$

and

$$\Phi^*(y) = \left[\inf_{\alpha_i^t, x_i^t} \left\{ \sum_{t=1}^{T} \left[\sum_{i=1}^{I} (F_i \alpha_i^t + V_i x_i^t) \right] \right\} \right.$$

$$\text{s.t.} \quad \sum_{i=1}^{I} x_i^t \geq d^t - y_1, \qquad t = 1, \ldots, T$$

$$\sum_{i=1}^{I} \overline{x}_i \alpha_i^t \geq d^t + R - y_2, \qquad t = 1, \ldots, T$$

$$\{\alpha_i^t, x_i^t\} \in Co \left\{ \underline{x}_i \alpha_i^t \leq x_i^t \leq \overline{x}_i \alpha_i^t \right.$$

$$i = 1, \ldots, I, \quad t = 1, \ldots, T$$

$$0 \leq \alpha_i^t \leq 1, \text{ integer},$$

$$\left. \left. i = 1, \ldots, I, \quad t = 1, \ldots, T \right\} \right].$$

The duality gap $v(P) - v(D)$ is precisely equal to the difference between the perturbation function of (P) and its lower convex envelope, both evaluated at the origin.

9.4. Summary

Intuitively, it is possible to relate the duality gap to the reserve constraint, since this constraint is seldom strictly satisfied. Figure 9.2 presents the perturbation function for a single thermal unit (unit A1 from the Scottish Power model, Appendix 4) and its lower convex envelope. The demand is fixed at 300 MW and only the reserve constraint is perturbed, in the range $y_2 \in [-200, 200]$. Clearly, it can be seen that for values of the reserve requirement less than 200 MW, a duality gap will always exist. When the reserve is such as to be equal to the difference $\overline{x}_i - d^t$, i.e. the maximum output minus the actual demand, no duality gap is present and a global subgradient [Geoffrion (1974)] exists. Figure 9.3 presents similar results for a system comprising three thermal units (units A1, A2 and B1 from the Scottish Power model, Appendix 4), and it shows the presence of the duality gaps associated with the plateaux on the perturbation function, signalling the commitment of an extra unit if the reserve requirement is to be satisfied.

These two examples show an important characteristic of the duality gap in power scheduling: the gap is data dependent. This dependence is directly linked to the level at which the reserve is operated. It also shows that the biggest gaps occur when there is no reserve requirement, or when an extra unit is required to be started up.

Units with no associated load costs, such as peakers, hydro units and pump-storage units do not individualy exhibit any duality gap with respect to the perturbation on the reserve constraint; however, duality gaps may be present for perturbations on the demand constraint, e.g. for loads below the minimum power output. For mixed systems it is possible to see a reduction in the duality gap, and in fact this was observed with the introduction of extra units and selling. The last case is clear since it corresponds to the sale of the extra available power.

Although the literature suggests that the duality gap is less than 1% for large systems, the arguments of this chapter have demonstrated that for small systems, illustrated by the Scottish Power model, a duality gap greater than 1% can be observed.

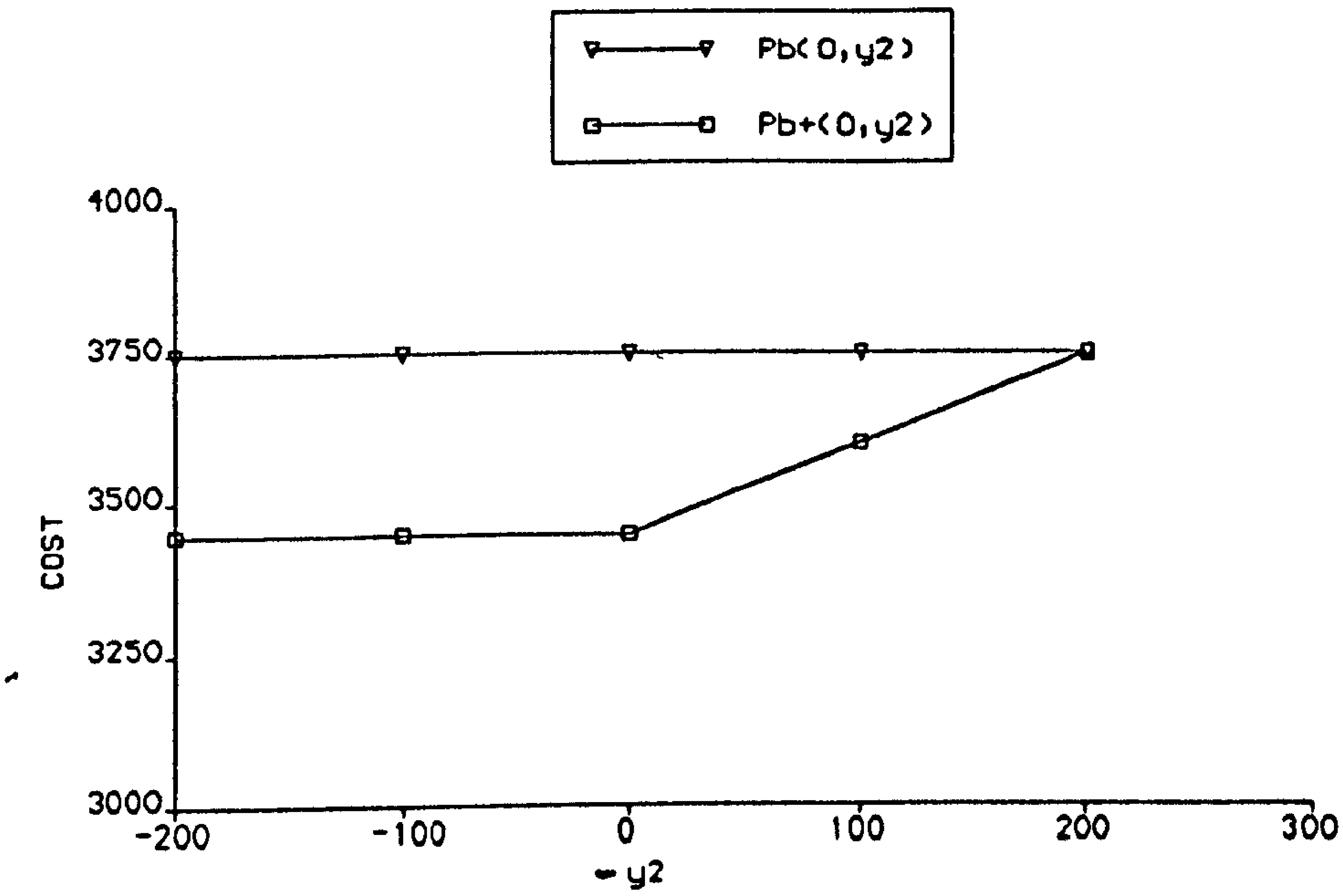


Fig. 9.2. Perturbation function Pb and lower convex envelope Pb+ for unit A1

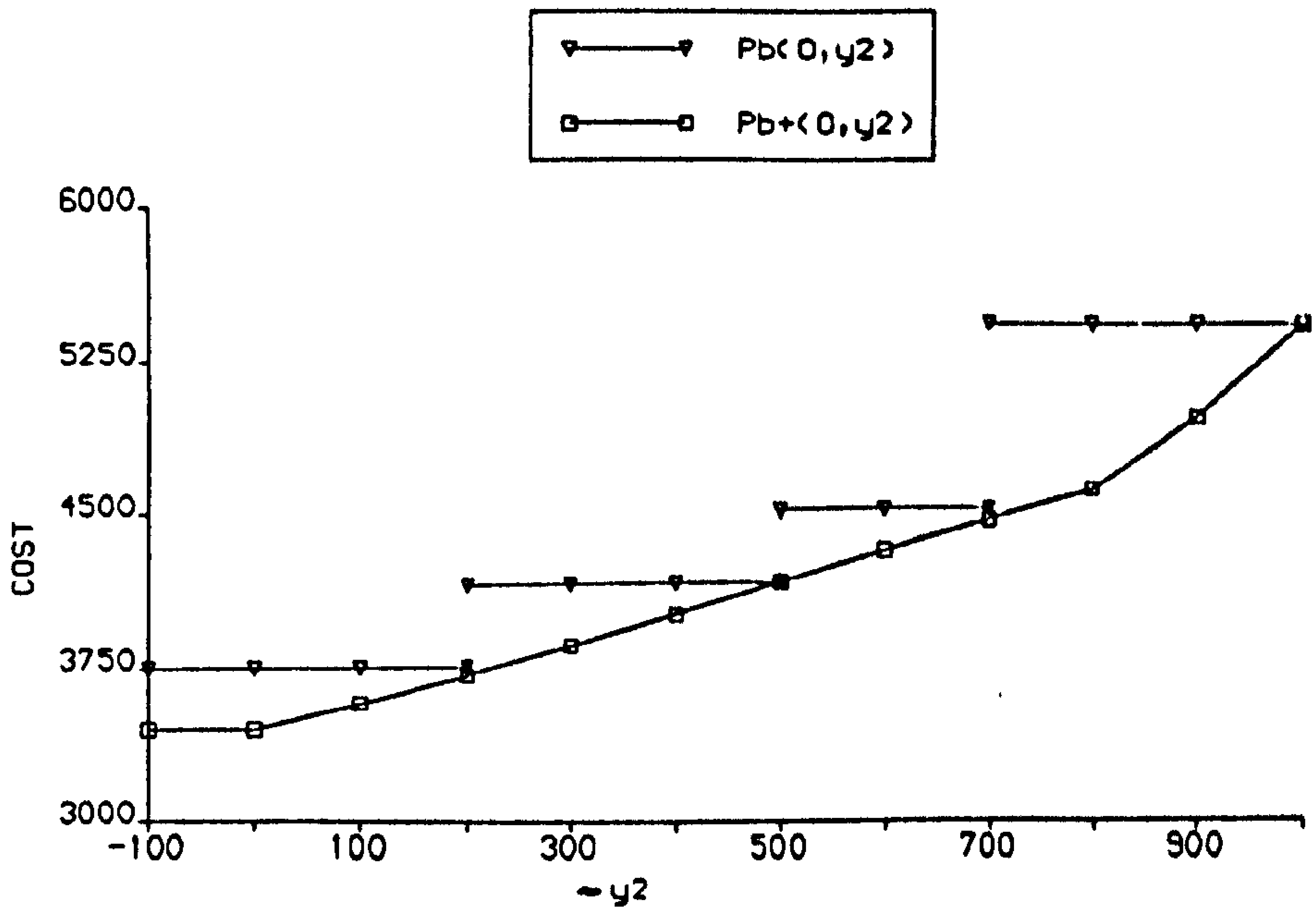


Fig. 9.3. Perturbation function Pb and lower convex envelope Pb+ for units A1, A2 and B1

CHAPTER 10

CONCLUSIONS AND FUTURE WORK

This thesis has been concerned with two power systems which are closely related to the real systems of the companies Scottish Hydro-Electric plc and Scottish Power plc, respectively. Both systems included thermal, hydro and pump-storage units. In addition, one of the models included the possibility of purchasing from, and selling electricity to the pool system. Few systems that consist of a combination of all types of units have been considered in the literature, and such systems have rarely been solved by Lagrangian relaxation techniques.

The unit commitment/economic dispatch problem has been investigated through three different approaches: branch-and-bound, Lagrangian relaxation and genetic algorithms.

Branch-and-bound is the most rigorous approach in that it does not depend on any heuristic. However, as the problem size grows, the computational time increased exponentially, thus preventing the use of this approach on a real time basis.

In Lagrangian relaxation a trade-off between execution time and the use of a heuristic is made. The dual problem was not solved exactly and a primal feasible solution was constructed by means of a heuristic. With this compromise Lagrangian relaxation proved to be of practical value since it produced better solutions in much shorter computational time despite the fact that multiple solutions are known to exist near the optimum due to non-convexity. A rapid near approximation in the dual is attractive because the dual is itself a good approximation to the primal problem. The weakness of this method was that both the parameters of the subgradient method and the heuristic had to be tuned for different systems. Nevertheless, once tuned Lagrangian relaxation was by far and away the fastest algorithm. Furthermore, the decomposition permitted by this approach also offered the possibility for a parallel implementation resulting in significant reductions in the computational time. This parallel implementation would permit very large problems indeed to be solved in real time.

Genetic algorithms are reasonably effective methods for highly combinatorial problems. In power scheduling problems genetic algorithms proved to be efficient for getting from an infeasible region to a feasible region, but were comparatively slow in their search for the optimum.

The inclusion of the cost of hydro energy revealed a close link between fixing the reservoir level at the end of the planning period and the pricing of that energy. This, incidently is also an effective means of pricing water supplies for other uses such as irrigation, domestic and/or industrial. The various simulations revealed that the pump-storage unit cost is a relative one, that is it is dependent on the cost of the energy used to pump up the water to the reservoir, and therefore on the system as a whole.

Purchases from and sales to the pool system were also modelled. The results showed that the market system could have an enormous impact on the management of the generating boards. The Lagrange multipliers can be perceived as shadow prices and used to define the best strategy towards the market. A generating company often has to make a decision as to whether to sell electricity or simply to store it using their hydro units. This decision necessarily requires that hydro energy be priced.

A large structured code using some features of FORTRAN 90 was developed for the Lagrangian relaxation approach. This code was tested under very different conditions. The heuristic accounted for a large part of the code, though the number of calls to this section were relatively few.

An investigation of the nature of the duality gap was carried out. The results showed that the duality gap could be greater than 1% for small systems. Further, the duality gap can be shown to be dependent on the operating reserve constraint. Thus, generally the gap proved to be problem dependent.

Future work should tackle the inclusion of different constraints like ramping, transmission, plant crew constraints and the head effect of the reservoirs. The inclusion of ramping constraints and the head effects is likely to increase significantly the computational time, and thus efficient modelling of these constraints should be an important line of research. A more heuristic independent approach to the primal solution from the dual needs to be pursued. The method of setting up new multipliers deserves further study since problems of convergence would appear to be dependent on the particular method employed. The study of the duality gap in the nonlinear case is still essentially an open problem.

APPENDIX 1

SCOTTISH-HYDRO ELECTRIC DATA

The system under consideration is a small part of the total system of the North of Scotland Hydro-Electric Board, now called Scottish Hydro-Electric plc. This system consists of five generating units. Some information concerning demand and inflows to the hydro schemes is also provided.

This information is summarized in the following tables.

Table A1-1
Nuclear and Thermal Units

	load		cost			min/down
Unit	min	max	up	run	down	time
Nuclear	510	510	-	†	-	-
Oil	70	325	1000	$450 + 10x_i^t$	100	3
Coal	90	290	3000	$600 + 15x_i^t$	300	4

(load - MW, cost - £, time - h, x_i^t - power output)
†(must run continuously)

The inflows to the hydro units are assumed constant over a 24 hour period. This information has been given in energy units [Cohen and Wan (1985)], assuming that there are no great variations in reservoir heads. The inflow to the pump-storage unit is very small, and in the present study it has been assumed to be 1% of the hydro unit inflow; also lower limits are prescribed for both generation and pumping.

Table A1-2
Daily Run Off (GWh)

MON	TUE	WED	THU	FRI	SAT	SUN
12.4	9.6	19.2	10.4	14.4	10.1	7.8

Table A1-3
Hydro Units

	load		level of reservoir			run
Unit	min	max	min	max	start	cost
Hydro	25	500	180	280	240	12
Pump/storage	-	-	0	6†	3.5	12
- generation	25	300	-	-	-	-
- pumping ‡	18	225	-	-	-	-

(load - MW, level - GWh, cost - £)
†(At the end of planning period level of the reservoir $\geq$ 3.5 10^3MWh)
‡(The turn-round efficiency of 75%)

The demand has been assumed constant over an 1 hour period.

Table A1-4
Demand (MW)

h	MON	TUE	WED	THU	FRI	SAT	SUN
1	1088	1141	1092	1112	1089	1128	1125
2	1027	1084	1011	1043	1028	1057	1097
3	971	1031	953	960	957	940	960
4	879	956	877	888	865	837	830
5	809	869	798	808	794	759	745
6	779	846	775	782	771	721	708
7	880	930	859	879	858	737	705
8	1069	1098	1013	1049	1040	768	683
9	1199	1207	1173	1157	1154	857	707
10	1238	1242	1161	1162	1158	1001	859
11	1193	1223	1135	1117	1117	1034	951
12	1136	1192	1106	1095	1080	1002	969
13	1129	1205	1118	1097	1087	977	995
14	1096	1161	1076	1059	1046	928	955
15	1092	1155	1065	1045	1054	895	920
16	1135	1174	1094	1074	1064	888	931
17	1331	1326	1273	1258	1226	1054	1078
18	1393	1368	1313	1318	1275	1176	1153
19	1278	1270	1220	1220	1186	1099	1106
20	1221	1210	1158	1156	1141	1043	1076
21	1187	1169	1136	1130	1186	1004	1051
22	1140	1133	1100	1091	1038	970	1042
23	1055	1027	996	993	955	914	964
24	1075	1052	1050	1024	1030	1022	1021

APPENDIX 2

THE NONLINEARITY IN THE PUMP-STORAGE UNIT

Usually pumping and generation do not occur at the same time when constraint (2.3.8) is not imposed. However, it is possible to create situations in which pumping and generation do occur simultaneously: for instance, in systems where the final level of the reservoir has to be the same as that at the beginning of the planning period. As an example a situation based on the data provided by Scottish Hydro-Electric was created, for a reserve of 50 MW, where the constraint (2.3.8) was included in the model (Table A2-1) (all units in MW). The constraint has then been removed and the same computation performed (Table A2-2). It can be seen that the constraint can be active. Table A2-3 presents the levels (in MWh) of the reservoir for the two situations created. As a final remark, pumping and generation occurring at the same time has also been observed by Wacker in a very different model (private communication).

Table A2-1

Time	Oil	Coal	Hydro	Gen1	Total	Pump1	Dem
1	325.0	0.0	450.0	25.0	800.0	0.0	800.0
2	325.0	0.0	450.0	25.0	800.0	0.0	800.0
3	325.0	0.0	450.0	25.0	800.0	0.0	800.0
4	325.0	0.0	450.0	25.0	800.0	0.0	800.0
5	325.0	0.0	450.0	25.0	800.0	0.0	800.0
6	325.0	0.0	450.0	25.0	800.0	0.0	800.0
7	325.0	0.0	450.0	25.0	800.0	0.0	800.0
8	325.0	0.0	450.0	25.0	800.0	0.0	800.0
9	325.0	126.6	500.0	0.0	951.6	151.6	800.0
10	325.0	90.0	500.0	0.0	915.0	115.0	800.0

(Total cost £99649.0)

Table A2-2

Time	Oil	Coal	Hydro	Gen2	Total	Pump2	Dem
1	325.0	0.0	475.0	25.0	825.0	25.0	800.0
2	325.0	0.0	475.0	25.0	825.0	25.0	800.0
3	325.0	0.0	475.0	25.0	825.0	25.0	800.0
4	325.0	0.0	475.0	25.0	825.0	25.0	800.0
5	325.0	0.0	475.0	25.0	825.0	25.0	800.0
6	325.0	0.0	475.0	25.0	825.0	25.0	800.0
7	325.0	0.0	483.3	25.0	833.3	33.3	800.0
8	325.0	0.0	500.0	25.0	850.0	50.0	800.0
9	325.0	0.0	500.0	25.0	850.0	50.0	800.0
10	325.0	0.0	500.0	25.0	850.0	50.0	800.0

(Total cost £94999.0)

Table A2-3

Pump-storage unit

Time	Gen1	Pump1	Level1	Gen2	Pump2	Level2
1	25.0	0.0	3475.0	25.0	18.8	3493.8
2	25.0	0.0	3450.0	25.0	18.8	3487.5
3	25.0	0.0	3425.0	25.0	18.8	3481.3
4	25.0	0.0	3400.0	25.0	18.8	3475.0
5	25.0	0.0	3375.0	25.0	18.8	3468.8
6	25.0	0.0	3350.0	25.0	18.8	3462.5
7	25.0	0.0	3325.0	25.0	25.0	3462.5
8	25.0	0.0	3300.0	25.0	37.5	3475.0
9	0.0	113.7	3413.7	25.0	37.5	3487.5
10	0.0	86.3	3500.0	25.0	37.5	3500.0

(Initial volume 3500.0 MWh, pump efficiency 3/4)

APPENDIX 3

SCOTTISH POWER DATA

The system considered here is an example set created by Scottish Power, which emulates their own real system. There are three thermal stations, with two units each, including two peakers; a pondage hydro unit and a pump-storage unit. Some information concerning demand and inflows to the hydro schemes is also provided.

This information is summarized in the following tables.

Table A3-1
Thermal Units

	load		cost		min/up	min/down
Unit	min	max	up	run	time	time
A1	150	500	10000	$750 + 10x_i^t$	4	4
A2	150	500	10000	$750 + 10x_i^t$	4	4
B1	50	300	3000	$300 + 12x_i^t$	2	2
B2	50	300	3000	$300 + 12x_i^t$	2	2
C1	10	100	0	$20x_i^t$	1	1
C2	10	100	0	$20x_i^t$	1	1

(load - MW, cost - £, time - h, x_i^t - power output)

The inflows to the hydro unit is assumed constant over a 24 hour period. There are no inflows to the pump-storage unit. The operating characteristics are the following ones:

Hydro unit

Max Load (MW)	100
Run-off (MWh)	500
Start Level (MWh)	2500

Pump-storage unit

Number of sets	2
Pumping per set (MW)	100
Max Generation per set (MW)	100
Min Generation per set (MW)	10
Efficiency	0.75
Start Level (MWh)	5000

Table A3-2
Hydro Unit Price Levels

Level of reservoir		
From	To	Price
1000	1500	21.0
1500	2500	12.0
2500	3000	10.5
3000	4000	9.5

(Level - MWh, Price - £/MWh)

Table A3-3
Pump-storage Unit Price Levels

Level of reservoir		
From	To	Price
4000	5000	15.3
5000	6000	12.3
6000	7000	10.3

(Level - MWh, Price - £/MWh)

The demand has been assumed constant over an 1 hour period. Three sets of demand data were given (D1, D2, D3). The sale price is the pool price (purchase price) reduced from a certain amount (£0.5). The maximum amount of purchase and sale in any time period is, respectively, 200 MW and 300 MW.

Table A3-4
Demand (MW) and Pool Price (£/MW)

h	D1	D2	D3	Pool Price
1	500	500	600	15.0
2	400	400	500	12.5
3	300	300	400	10.0
4	300	300	400	10.0
5	300	300	400	10.0
6	300	300	400	10.0
7	300	300	400	10.0
8	400	400	500	12.5
9	500	500	600	15.0
10	600	600	700	17.5
11	750	700	800	20.0
12	800	700	800	20.0
13	800	600	600	15.0
14	800	600	600	15.0
15	800	600	600	15.0
16	900	700	700	17.5
17	900	800	800	20.0
18	1000	900	900	22.5
19	1000	900	900	22.5
20	900	800	800	20.0
21	800	700	700	17.5
22	750	600	600	15.0
23	700	600	600	15.0
24	500	500	500	12.5

REFERENCES

Aoki, K., Satoh, T. and Itoh, M. (1987). Unit commitment in a large-scale power system including fuel constrained and pumped-storage hydro. *IEEE Trans. Power Syst.*, Vol. PWRS-2, 4, 1077-1084.

Aoki, K., Itoh, M., Satoh, T., Nara, K. and Kanezashi, M. (1989). Optimal long-term unit commitment in large scale systems including fuel constrained thermal and pumped-storage hydro. *IEEE Trans. Power Syst.*, Vol. PWRS-4, 3, 1065-1073.

Baptistella, L. and Geromel, J. (1980). Decomposition approach to problem of unit commitment schedule for hydrothermal systems. *IEE Proc.*, Vol. 127, 6, 250-258.

Bard, J. (1988). Short-term scheduling of thermal-electric generators using lagrangian relaxation. *Oper. Res.*, Vol. 36, 5, 756-766.

Bauer, W., Gfrerer, H., Lindner, E., Schwarz, A. and Wacker, Hj. (1987). Optimization of the Gosau hydro energy power plant system. *Math. Eng. Industry*, Vol. 1, 3, 169-190.

Bazaraa, M. and Shetty, C. (1979). Nonlinear Programming - Theory and Algorithms. New York: John Wiley.

Beale, E. M. (1984). Mathematical programming. In R. Eglese and G. Rand (eds.) *Developments in Operational Research*, Oxford: Pergamon, 1-10.

Beale, E. M. (1988). Introduction to Optimization. Chichester: John Wiley.

Bellman, R. and Dreyfus, S. (1962). Applied Dynamic Programming. Princeton: Princeton University Press.

Benders, J. (1962). Partitioning procedures for solving mixed-variables programming problems. *Nume. Math.*, Vol. 4, 238-252.

Bertsekas, D. (1987). Dynamic Programming: Deterministic and Stochastic Models. Englewood Cliffs, N. J.: Prentice-Hall.

Bertsekas, D., Lauer, G., Sandell, N. and Posbergh, T. (1983). Optimal short-term scheduling of large-scale power systems. *IEEE Trans. Aut. Control*, Vol. AC-28, 1, 1-11.

Bertsekas, D. and Sandell, N. (1982). Estimates of the duality gap for large-scale separable nonconvex optimization problems. *Proc. IEEE Conf. Decision and Control*, 782-785.

Blair-Fish, J., Brown, S., MacDonald, N., Smith N. and Stroud, N. (1990). Using CS tools on the Edinburgh Concurrent Supercomputer, Edinburgh Parallel Concurrent Centre UG91-02, Edinburgh: Edinburgh University.

Cohen, A. (199). Modelling unit ramp limitations in unit commitment. *Proc.* 10^{th} *Power Systems Computation Conf.*, London: Butterworths, 1107-1114.

Cohen, A. and Sherkat, V. (1987). Optimization-based methods for operations scheduling. *IEEE Proc.*, Vol. 75, 12, 1574-1591.

Cohen, A. and Yoshimura, M. (1985). A branch-and-bound algorithm for unit commitment. *IEEE Trans. Power App. Syst.*, Vol. PAS-102, 2, 444-450.

Cohen, A. and Wan, S. (1985). An algorithm for scheduling a large pumped storage plant. *IEEE Trans. Power App. Syst.*, Vol. PAS-104, 8, 2099-2104.

Cooper, L. and Cooper, M. (1981). Introduction to Dynamic Programming. Oxford: Pergammon Press.

Davis, L. (1987). Genetic Algorithms and Simulated Annealing. London: Pitman.

Dawkins, R. (1989). The Selfish Gene. (2^{nd} ed.) Oxford: Oxford University Press.

Dirickx, Y. and Jennergren, L. (1979). System Analysis by Multilevel Methods. Chichester: John Wiley.

Edinburgh Portable Compilers (1991). Parallel Fortran 77 - User's reference manual - version 2.4. Edinburgh: Edinburgh Portable Compilers Ltd.

Efthymoglou, P. (1987). Optimal use and the value of water resources in electricity generation. *Management Sci.*, Vol. 33, 12, 1622-1634.

Engles, L., Larson, R., Peschow, J. and Stanton, N. (1976). Dynamic programming applied to hydro and thermal generation scheduling. *IEEE Tutorial Course Test - Application of optimization methods in power system engineering*, 52-68.

Fletcher, R. (1987). Practical Methods of Optimization. (2^{nd} ed.) Chichester: John Wiley.

Fisher, M. (1981). The lagrangian relaxation method for solving integer programming problems. *Management Sci.*, Vol. 27, 1, 1-18.

Fisher, M. (1985). An applications oriented guide to lagrangian relaxation. *Interfaces*, Vol. 15, 2, 10-21.

Gangl, A. (1989). Short term optimization of an extension of a storage power plant system. *Diplomarbeit.* Linz, Austria: Johannes Kepler Universitat.

Garey, M. and Johnson, D. (1979). Computers and Intractability: a guide to the theory of NP-Completeness. New York: Freeman.

Garver, L. (1963). Power generation scheduling by integer programming - development of theory. *AIEE Trans.*, Vol. 81, 3, 1212-1218.

Geoffrion, A. (1970a). Elements of large scale mathematical programming - Part I: concepts. *Management Sci.*, Vol. 16, 11, 652-675.

Geoffrion, A. (1970b). Elements of large scale mathematical programming - Part II: synthesis of algorithms and bibliography. *Management Sci.*, Vol. 16, 11, 676-691.

Geoffrion, A. (1971). Duality in nonlinear programming: a simplified applications oriented development. *SIAM Review*, Vol. 13, 1, 1-37.

Geoffrion, A. (1972). Generalized Benders decomposition. *J. Optimiz. Theory and Appl.*, Vol. 10, 4, 237-260.

Geoffrion, A. (1974). Lagrangian relaxation for integer programming. *Math. Programming Stud.*, Vol. 2, 82-114.

Glover, F. (1975). Improved linear integer programming formulations of nonlinear integer problems. *Management Sci.*, Vol. 22, 4, 455-460.

Goldberg, D. (1989). Genetic Algorithms in Search, Optimization and Machine Learning. Reading, Mass.: Addison Wesley.

Grefenstette J. J. (1988). Incorporating problem specific knowledge into genetic algorithms. In L. Davis (ed.) *Genetic Algorithms and Simulated Annealing*, London: Pitman, 42-60.

Gutenberger, J. (1989). Yearly optimization and comparison of two extensions of a storage power plant system. *Diplomarbeit.* Linz, Austria: Johannes Kepler Universitat.

Habibollahzadeh, H. and Bubenko, J. (1986). Application of decomposition techniques to short-term operation planning of hydrothermal power system. *IEEE Trans. Power Syst.*, Vol. PWRS-1, 1, 41-47.

Harhammer, P. G. and Infanger, G. M. (1990). Energy operation planning. *Proc.* 10^{th} *Power Systems Computation Conf.*, London: Butterworths, 631-637.

Hastings, N. (1973). Dynamic Programming with Management Applications. London: Butterworth.

Held, M., Wolfe, P. and Crowder, H. (1974). Validation of subgradient optimization. *Math. Program.*, Vol. 6, 62-88.

Hoffman, A. and Kruskal, J. (1956). Integral boundary points of convex polyhedra. In H. Kuhn and A. Tucker (Eds.) *Linear Inequalities and Related Systems*, Princeton: Princeton University Press, 233-246.

Holland, J. H. (1975). Adaptation in Natural and Artificial Systems. Ann Arbor: University of Michigan Press.

Kirkpatrick, S., Gelatt, C. and Vecchi, M. (1983). Optimization by simulated annealing. *Science*, Vol. 220, 4598, 671-680.

Khodervarian, E., Brameller, A. and Dunnet, R. M. (1986). Semi-rigorous thermal unit commitment for large scale electrical power systems. *IEE Proc.*, Vol. 133, 4, 157-164.

Larson, R. and Casti, J. (1982). Principles of Dynamic Programming-Part II Advanced Theory and Applications. New York: Marcell Dekker, 403-421.

Lasdon, L. (1968). Duality and decomposition in mathematical programming. *IEEE Trans. Syst. Science Cyb.*, Vol. SSC-4, 2, 86-100.

Lasdon, L. (1970). Optimization Theory for Large Systems. London: Macmillan.

Lauer, G., Bersekas, D., Sandell, N. and Posbergh, T. (1982). Solution of large-scale optimal unit commitment. *IEEE Trans. Power App. Syst.*, Vol. PAS-101, 1, 79-86.

Lowery, P. (1966). Generating unit commitment by dynamic programming. *IEEE Trans. Power App. Syst.*, Vol. PAS-85, 5, 422-426.

Luenberger, D. (1989). Linear and Nonlinear Programming. (2^{nd} ed.) Reading, Mass.:Addison-Wesley.

Meiko (1988). FORTRAN on the computing surface. Almondsbury, Bristol: Meiko Ltd.

Merlin, A. and Sandrin, P. (1983). A new method for unit commitment at Electricite de France. *IEEE Trans. Power App. Syst.*, Vol. PAS-102, 5, 1218-1225.

Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A. and Teller, E. (1953). Equation of state calculations by fast computing machines. *J. Chem. Phys.*, Vol. 21, 6, 1087-1092.

McKinnon, K. and Buchanan, J. (1988). The short term scheduling of a hydro-thermal electricity generating system. In A. Osiadacz (ed.) *Simulation and Optimisation of Large Systems*, Oxford: Clarendon Press, 229-244.

Mokhtari, S., Singh, J. and Wollenberg, B. (1988). A unit commitment expert system. *IEEE Trans. Power Syst.*, Vol. PWRS-3, 1, 272-277.

Muckstadt, J. and Wilson, R. (1968). An application of mixed-integer programming duality to scheduling thermal generating systems. *IEEE Trans. Power App. Syst.*, Vol. PAS-87, 12, 1968-1978.

Muckstadt, J. and Koenig, S. (1977). An application of lagrangian relaxation to scheduling in power-generation systems. *Oper. Res.*, Vol. 25, 3, 387-403.

Oliveira, P., McKee, S. and Coles, C. (1991). The optimal scheduling of hydro electric power generation. In M. Hellio (ed.) *Proc.* 5^{th} *ECMI Conf.*, Dordrecht: Kluwer Academic, 109-114.

Oliveira, P., McKee, S. and Coles, C. (1992). Optimal scheduling of a hydro thermal power generation system. *Europ. J. Oper. Res.*, (to appear).

Oliveira, P., McKee, S. and Coles, C. (1992). Lagrangian relaxation and its application to the unit commitment/economic dispatch problem. *IMA J. Math. Appl. Bus. Industry*, (submitted for publication).

Oliveira, P., Blair-Fish, J., McKee, S. and Coles, C. (1992). Parallel Lagrangian relaxation in power scheduling. *Software Pract. Exper.*, (submitted for publication).

Oliveira, P., McKee, S., Coles, C. and Blair-Fish, J. (1992). Optimal scheduling of hydro thermal power systems. In F. Hodnett (ed.) *Proc. of the* 6^{th} *ECMI Conf.*, (to appear).

Pang, C. and Chen, H. (1976). Optimal short-term thermal unit commitment.

IEEE Trans. Power App. Syst., Vol. PAS-95, 4, 1336-1346.

Pang, C., Sheble, F. and Albuyeh, F. (1981). Evaluation of dynamic programming based methods and multiple area representation for thermal unit commitments. *IEEE Trans. Power App. Syst.*, Vol. PAS-100, 3, 1212-1218.

Ruzic, S. and Rajakovic, N. (1991). A new approach for solving extended unit commitment problem. *IEEE Trans. Power Syst.*, Vol. PWRS-6, 1, 269-277.

Sandell, N., Bertsekas, D., Shaw, J., Gully, S. and Gendron, R. (1982). *Proc. IEEE Conf. Large Scale Systems*, 141-147.

Scicon (1981). SCICONIC/VM User Guide. Milton Keynes: Scicon Computer Services Ltd.

Shaw, J. and Bersekas, D. (1985). Optimal scheduling of large hydrothermal power systems. *IEEE Trans. Power App. Syst.*, Vol. PAS-104, 2, 286-293.

Sherkat, V., Campo, R., Moslehi, K. and Lo, E. (1985). Stochastic long-term hydrothermal optimization for a multireservoir system. *IEEE Trans. Power App. Syst.*, Vol. PAS-104, 8, 2040-2049 .

Sjelvgren, D., Anderssson, S., Andersson, T., Nyberg, U. and Dillon, T. S. (1983). Optimal operations planning in a large hydro-thermal power system. *IEEE Trans. Power App. Syst.*, Vol. PAS-102, 11, 3644-3651.

Smith, S. F. (1984). Adaptive learning systems. In R. Forsyth (ed.) *Expert Systems - Principles and Case Studies*, London: Chapman and Hall, 168-189.

Snyder, W., Powell, D. and Rayburn, J. (1987). Dynamic programming approach to unit commitment. *IEEE Trans. Power Syst.*, Vol. PWRS-2, 2, 339-349.

Thornton, P., Blair-Fish, J. and Wilson, W. (1991). Crop simulation modelling using a transputer-based parallel computer. *Agricul. Syst.*, Vol. 35, 321-337.

Tong, S. and Shahidehpour, S. (1989). An overview of power generation scheduling in the optimal operation of a large scale power system. *Proc. Power Industry Computer Applications.*

Tong, S. and Shahidehpour, S. (1990). An innovative approach to generation scheduling in large-scale hydro-thermal power systems with fuel constrained units. *IEEE Trans. Power Syst.*, Vol. PWRS-5, 2, 665-673.

Turgeon, A. (1978). Optimal scheduling of thermal generating units. *IEEE Trans. Automat. Contr.* Vol. AC-23, 6, 1000-1005.

Van den Bosch, P. and Honderd, G. (1985). A solution of the unit commitment problem via decomposition and dynamic programming. *IEEE Trans. Power App. Syst.*, Vol. PAS-104, 7, 1684-1690.

Van Roy, T. and Wolsey, L. (1987). Solving mixed integer programming problems using automatic reformulation. *Oper. Res.*, Vol. 35, 1, 45-57.

Virmani, S., Imhof, K. and Mukherjee, S. (1989). Implementation of a Lagrangian relaxation based unit commitment problem. *IEEE Trans. Power Syst.*, Vol. PWRS-4, 4, 1373-1379.

Yang, J. and Chen, N. (1989). Short term hydrothermal coordination using multi-pass dynamic programming. *IEEE Trans. Power Syst.*, Vol. PWRS-4, 3, 1050-1056.

Zhuang, F. and Galiana, F. (1988). Towards a more rigorous and pratical unit commitment by Lagrangian relaxation. *IEEE Trans. Power Syst.*, Vol. PWRS-3, 2, 763-773.

Zhuang, F. and Galiana, F. (1990). Unit commitment by simulated annealing. *IEEE Trans. Power Syst.*, Vol. PWRS-5, 1, 311-318.

Wallace, D. (1991). Supercomputing with transputers. *Comput. System. Eng.*, Vol. 1, 1, 131-141.

Wells, D. (1976). Power system scheduling using integer programming. In L. Dixon (ed.) *Optimisation in Action*, Oxford: Academic Press, 467-477.

Wexler, J. and Prior, D. (1989). Solving problems with transputers: background and experience. *Microprocess. and Microsyst.*, Vol. 13, 67-78.

Wood, A. and Wollenberg, B. (1984). Power generation and control. New York: John Wiley.