T-4499

FORMULATION AND SOLUTION STRATEGIES FOR SHORT-TERM SCHEDULING OF POWER SYSTEM RESOURCES

by

Terrance J. Fulp

A thesis submitted to the Faculty and the Board of Trustees of the Colorado School of Mines in partial fulfillment of the requirements for the degree of Doctor of Philosophy (Mathematical and Computer Sciences).

Golden, Colorado

Date 11/11/96

Signed: Terrance J. Fulp

Terrance J. Fulp

Approved: Robert G. Underwood

Dr. Robert G. Underwood
Thesis Advisor

Golden, Colorado

Date 11/11/96

Graeme Fairweather

Dr.

Profe Dr. Graeme Fairweather
Department of Mathematical and
Computer Sciences

ABSTRACT

The short-term resource scheduling (STRS) of power system resources focuses on determining an operational schedule for each generating resource in an electric utility's system. The goal is to determine a schedule that will meet the forecasted electrical demand (load), provide for system security, and meet a variety of other constraints (both policy- and physically-based), all at minimum cost.

The types of resources considered include thermal plants (both steam and gas turbines) and hydroelectric (hydro) plants and the problem is decomposed into thermal and hydro subproblems. The hydro subproblem is solved in a separate application to ensure that water objectives and constraints are met. An adaptive, linear systems model is developed that accurately predicts the river flow downstream of controlled facilities and is used to set cumulative energy constraints over variable time horizons. These surrogate constraints are provided to the STRS model to allow optimal scheduling of the limited hydro energy.

Solutions to the STRS model are further constrained by the capacity and ramping constraints of each resource. Costs of generation are assumed linear; however, constant fixed costs and start-up costs are considered. The STRS model is formulated as a mixed-integer linear program and even with some simplifications, the resulting model cannot be solved by typical branch and bound methodologies. A new Lagrangian relaxation is developed that, under certain circumstances, provides tighter lower bounds than those previously achieved. A Lagrangian decomposition methodology is then developed that takes advantage of both relaxations and is guaranteed to generate lower bounds which dominate either relaxation. Feasible solutions are obtained with a simple rounding heuristic applied to the decomposition solution.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

I thank the U.S. Bureau of Reclamation, Lower Colorado Regional Office, and the U.S. Western Area Power Administration, Phoenix Area Office, for providing financial support for this research, and the U.S. Bureau of Reclamation, Yuma Area Office, for providing the data used in the water modeling study. The opinions expressed in this study are solely mine and do not necessarily represent those of the U.S. Bureau of Reclamation. Any recognition of a product used in the study is not an endorsement of the product by the U.S. Bureau of Reclamation.

I also thank Mr. Tim Ulrich of the U.S. Bureau of Reclamation, Lower Colorado Dams Facilities Office, for hiring me and introducing me to the field of hydropower. I extend a special gratitude to Mr. Tim Magee for taking an interest in this research and providing countless hours of discussion and moral support.

DEDICATION

To my Mom and Dad, who taught me at a very early age that the pursuit of knowledge is one of the greatest gifts of life and that perserverance is a virture.

Chapter 1

INTRODUCTION

The short-term resource scheduling (STRS) of power system resources focuses on determining an operational schedule for each generating resource in an electric utility's system. The goal is to determine a schedule that will meet the forecasted electrical demand (load), provide for system security, and meet a variety of other constraints (both policy- and physically-based), all at minimum cost. The problem is part of a hierarchy of decisions ranging from long-term planning over decades to real-time generation and transmission control every few seconds. Typically, the short-term schedules are produced at one-hour time steps over a horizon of a few days to a week.

Large utilities and power pools have a variety of generation resources available which can be used to meet the load. Each resource may have a suite of constraints that limit its operation and some constraints may couple other resources and time steps. Many of the resources (namely the fossil fuel and nuclear generators) have large costs associated with their start-up and once started must generate at some minimum level for some minimum time. Therefore, there is a substantial trade-off between having too many units "on-line" versus having to pay to start them when needed. Generation from hydro facilities is constrained by the physical constraints imposed by the watershed (requiring simulation of the water flow throughout the watershed), as well as policy constraints on the operation of the dams and reservoirs (known as "operating rules"). Transmission limits (either contractual or physical) may also constrain which generators may be used and at what levels. The problem is further complicated by the stochastic uncertainty of many of the variables, including the electrical and water demand, the water inflows, and the availability of the particular generators. These uncertainties affect the security of the system and impose further constraints to ensure that sufficient generating reserve is available at all times to prevent catastrophic failure of the system.

The STRS problem is of substantial importance from both economic and social points-of-view. Since fuel for the thermal resources represents the major cost of generation, the problem is of great practical interest as it has been reported that a 0.5% savings in fuel costs per year would result in operational savings of millions of dollars to a large utility (Choen and Sherkat 1987). Increases in the cost of fuels, as well as competition due to deregulation of the utility industry, further enhances the economic significance. System security has become an even greater issue now that nearly all of the electrical systems are interconnected throughout the U.S. In addition, objectives other than minimizing cost need to be taken into account. These include the reduction of pollution caused by generation (particularly fossil fuels), the conservation of particular fuels, and in the case of hydroelectric generation, meeting the objectives for other uses of the water (i.e., consumptive use for agriculture and municipalities, as well as recreational and environmental needs).

In its most general form, the STRS problem is a large, non-linear, non-convex, stochastic optimization problem which can not be solved to optimality for systems of any realistic size. However, due to its practical importance, a great deal of effort has been expended over the past two decades to achieve approximate solutions to simplified formulations of the general problem. This research follows this basic philosophy and extends the work in two main areas:

1. A methodology is developed based on linear systems and adaptive filtering theory which can be used to model the flow of water in river channels without the need for more complicated hydraulic measurements and modeling techniques. This methodology can be used to provide surrogate constraints to the STRS model which allows optimal scheduling of the hydro resources within these constraints while ensuring that other water objectives are met, and
2. A methodology (based on Lagrangian relaxation and decomposition theory) is developed that produces tighter lower bounds than previously achieved on the STRS problem. When coupled with a simple rounding heuristic, feasible solutions (upper bounds) for the STRS problem are generated which can be used with the lower bounds to bracket the unknown, optimal solution.

The presentation of this research is as follows: first, a more complete description of the STRS problem and the modeling strategies that have been applied to date are presented in Chapter 2. The particular model formulation considered in this research is then given in Chapter 3, including a discussion of the simplifications made. The new relaxation methodology for producing improved lower bounds is developed in Chapter 4 and the relationship to previous relaxation techniques is discussed. In Chapter 5, the methodology used to compute the best bound from the relaxed problem is presented and its performance is discussed. Computational experiments are presented that show the value of the new relaxation technique. Two heuristics are also presented: a rounding heuristic that can be applied to the solution of the relaxed problem that produces good, feasible solutions to the original problem and an additional heuristic methodology that can be used to produce a feasible initial solution for computing the best bound. In Chapter 6, details of the enhanced water model are presented. Finally, conclusions are drawn and some recommendations for further work are suggested in Chapter 7.

Chapter 2

PROBLEM DESCRIPTION AND CURRENT MODELING STRATEGIES

Planning and operation of an integrated power system may be viewed as a hierarchy of decisions, from long-term planning and policy-making to real-time generation and transmission control. This research focuses on the intermediate problem of short-term resource scheduling (STRS).

In this chapter, the STRS problem is defined in general terms and its position in this hierarchy of decisions is clarified. The most common types of resources (thermal and hydro) available for STRS are discussed from the perspectives of costs and constraints. In addition, the common system-level constraints are presented. Current modeling strategies for the STRS problem are then reviewed. This chapter concludes with the motivation for and contributions of this research.

2.1 Planning and Operation of an Integrated Power System

The hierarchy of decisions facing an integrated power utility range from long-term planning and policy-making (i.e., projecting the growth in electrical and water demand and deciding to construct new power plants), through mid-term operational planning (i.e., consummating fuel contracts, forecasting water supplies, and scheduling maintenance outages), to short-term scheduling and operations (forecasting electrical and water demands and STRS), and finally, to real-time generation and transmission dispatch. A variety of models may be used to support these decisions that require various time steps and time horizons, as shown in Figure 2.1. For example, water release targets may be determined on a monthly time step over a horizon of one to two years, based on monthly forecasts of inflow and target storage levels of reservoirs for flood control. Conversely, the decision within a week to use hydro generation instead of starting a steam turbine to meet an anticipated peak load requires an hourly forecast of load and a model that operates at that time step.

The STRS problem focuses on determining an optimal operational schedule for each generating resource at each time step (typically one hour) over the planning horizon (typically 24 to 168 hours). The overall objective is to minimize the total cost of meeting the electrical demand while meeting a variety of other constraints. These constraints include those particular to each set of resources, as well as constraints affecting the operation of both the water and electrical systems. There may also be competing objectives, particularly for the water system. In addition, many of the variables are stochastic, particularly the electrical demand and hydrologic inflows.

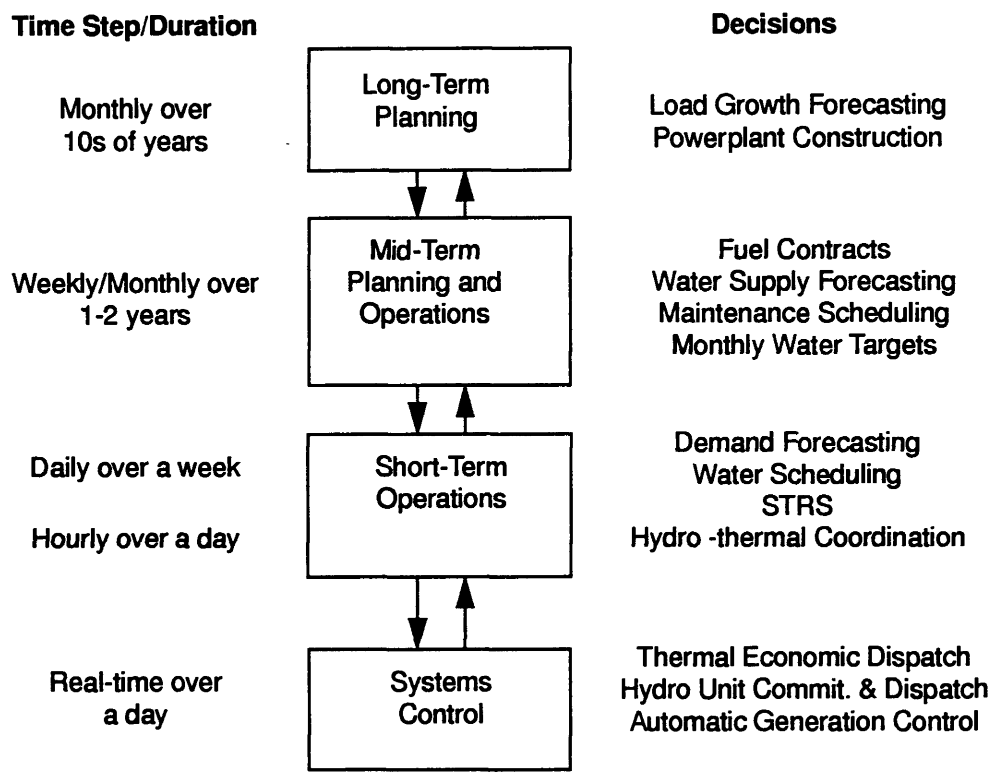

Figure 2.1. Hierarchy of Decisions and Typical Modeling Time Steps and Horizons for the Planning and Operation of an Integrated Power System

2.2 Thermal Resources

Thermal resources heat a working fluid (the most common being water and steam) to drive a turbine generator. The input energy comes from a variety of fuels including fossil fuels (oil, gas, and coal) and nuclear fuels (uranium). The common types of thermal units used in the U.S. include steam turbine generators, internal combustion generators, combined cycle units (those that use both internal combustion and steam units to increase efficiency), and light-water nuclear reactors. Typical unit sizes range from a few Mega-

Watts (MW) for some combustion turbines to 1200 MW for newer steam and nuclear turbines.

2.2.1 Costs Associated with Thermal Resources

The major cost of producing power from thermal generators is due to the cost of the fuel. Figure 2.2 (adapted from Wood and Wollenberg 1984) shows an idealized cost curve for an individual steam generator and a linear approximation to it. Such curves are derived from the energy input-output characteristics of the unit; however, other costs associated with the maintenance and operation of the individual generators and the power plant infrastructure may be included. This type of curve captures the salient features of most thermal generators—a non-linear, convex relationship in the normal operating range given by the minimum (m) and maximum (M) generation levels. Note that there is a fixed cost to operating at the minimum generation level, shown in the figure as f.

For some generators, however, these curves may not be convex (i.e., steam generators with multiple steam admission valves and combined cycle plants), particularly at the low end of the generation when the unit is being started. For nuclear-fueled reactors, the situation is made more complex if the costs include storing or reprocessing the core assembly after it is used (Wood and Wollenberg 1984).

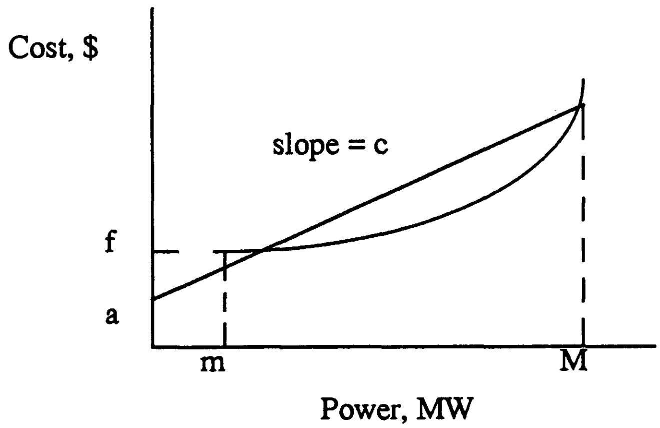

Figure 2.2. Typical Cost Curve for a Thermal (Steam) Generator and its Linear Approximation (Adapted from Wood and Wollenberg 1984)

For many thermal generators, there is an additional cost associated with starting the unit. This start-up cost, b, (not shown in Figure 2.2) is due to the energy used to bring the temperature and pressure of the unit up to its operating level (and again may include the crew costs associated with the start-up process and the maintenance costs due to cycling the unit) and can be a function of how long the unit has been down when it is restarted (Wood and Wollenberg 1984). These costs are often much greater (an order of magnitude greater) than the fixed cost, f, and are often under-estimated (Grimsrud et al. 1995).

2.2.2 Constraints Imposed on Thermal Resources

In addition to having minimum and maximum generating capacities that limit their generation as were shown in Figure 2.2, large thermal generators may also have constraints that limit how fast generation can increase or decrease from hour to hour (referred to as ramping constraints). Due to complexities in operation, nuclear plants are generally not allowed to ramp at all (i.e., they are "base-loaded"). In addition, units with start-up costs typically have constraints on the start-ups as well, including the minimum time a unit must stay up or down if turned on or off. Other units may be constrained by the availability of fuel (their total generation is limited over a certain time period) as well as units that must use a certain amount of fuel in a given time span.

2.3 Hydro Resources

Hydro resources utilize the potential and kinetic energy of water to drive the turbine generators. The common types of hydro facilities are storage (with a dam, reservoir, and power plant that generates when water is released), pumped storage (with a dam, reservoir, and plant with reversible pump-turbines), and run-of-river (with negligible storage and turbines that utilize water flowing in the channel). Individual hydro turbines typically range in size from a few MW to 200 MW and total plant capabilities can be greater than 1500 MW. For example, the total installed capacity at Hoover Dam on the Colorado River is 1930 MW (U.S. Bureau of Reclamation 1991).

2.3.1 Costs Associated with Hydro Resources

Since there is not a direct cost of the fuel, hydro generation is typically much cheaper than thermal generation. Since the water supply is limited, however, there is a trade-off between using the water now and keeping it in storage for future use. There are several ways in which hydro generation can be valued and this trade-off incorporated into a STRS model and these are discussed in the subsequent section on current modeling strategies.

For any of these strategies, however, a non-linear mapping must be made from the state variables used to describe the hydro system (flows and volumes) to the power system state variables (power and energy). On the turbine level, the energy output from a hydroelectric turbine is proportional to the rate of flow of water through the turbine and the operating head. Operating head is the difference in the elevation between the upstream side (the forebay) and the downstream side (the tailbay) and is a representation of the potential energy of the stored water. The power output of a typical hydroelectric turbine is shown in Figure 2.3 (adapted from Wood and Wollenberg 1984). Although similar to the cost curve for the thermal resources (Figure 2.2), the y-axis is now the rate of water flowing through the turbine, not the cost of producing the power. There are a family of these curves for each turbine in the plant over the range of operating head. Each turbine is limited by a minimum (m) and maximum capacity (M), which may also be a function of the operating head. The power output is therefore a non-linear function of the flow and the operating head.

Although Figure 2.3 shows the relationship between flow and power to be convex, some turbines may have non-convex relationships, particularly at the low end of the generation range (Rux 1994). Furthermore, there is a "fixed cost" in terms of water used to bring the units "up to speed" prior to generating. Individual hydro units may also be utilized in a "motoring state" where the turbine is kept on-line for use in meeting system spinning reserve constraints (defined in the next section). In such a state, the generator uses no water but operates as a motor by utilizing energy from other generators to maintain its synchronization to system speed. Finally, the power curves for particular units may be affected by which other units in the plant are operating due to the effect on the tailbay of the other units and also if some units share a common penstock (Cohen and Sherkat 1987).

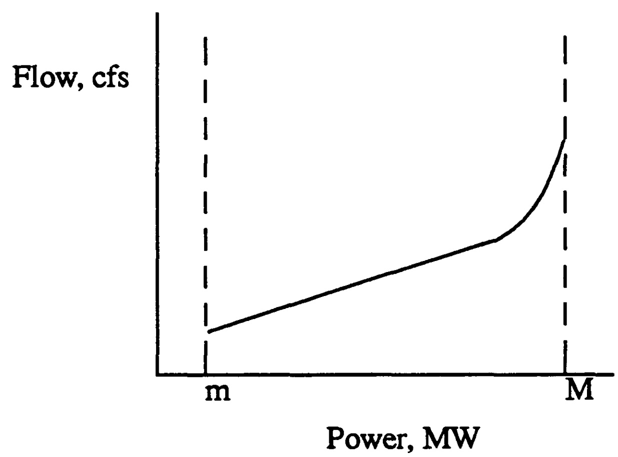

Figure 2.3. Typical Hydroelectric Input-output Curves for Constant Operating Head (adapted from Wood and Wollenberg 1984)

2.3.2 Constraints Imposed on Hydro Resources

In general, hydro generators typically do not have constraints on their minimum up and down times but penalties may be applied to discourage extremely rapid starts and stops. Individual hydro units may have constraints on their ramping rates and may also have generation ranges (known as "rough zones") which should be avoided due to mechanical instabilities.

2.4 System-Level Constraints

The primary system-level constraint is to meet the electrical demand. Because the demand is subject to random fluctuations and to provide backup in case units go down unexpectedly, additional constraints are added to ensure that there is sufficient generating reserve in the system. Typically, these requirements are represented in the form of two additional system level constraints: spinning reserve and supplemental reserve. Spinning reserve refers to the total amount of generation available from all units that are synchronized ("spinning") on the system minus the current load and losses (Wood and Wollenberg, 1984). The supplemental reserve also includes units that are not on-line but could be brought up in some short period of time (e.g., 10 minutes) (Wood and Wollenberg 1984). The actual rules that determine the amount of reserves needed are set by regional reliability councils and vary from area to area. For example, the spinning reserve might be expressed as a percentage of the forecasted load or as a function of the probability of not meeting demand (Wood and Wollenberg 1984).

In addition, large utilities and power pools may have other system-level constraints including the constraints on the distribution of the reserve within their area (perhaps due to transmission limits), as well as limits on the levels of interchange between adjacent areas (Ferreira et al. 1988).

The hydro system is limited by the constraints imposed by the physical network or watershed and often by the policies or operating rules that govern the operation of the entire water system. The physical constraints may be expressed by employing the principles of conservation of mass and energy (or momentum). For the simulation of large reservoirs, the mass balance equation simply states that the storage in the reservoir is a function of the previous storage, the inflows into the reservoir, and the outflows and other losses (i.e., evaporation and seepage) from the reservoir. Depending upon the accuracy desired, however, modeling the flow of water in river channels is much more difficult. In its most general form, flow in an open channel is described by a system of non-linear partial differential equations, the solution of which is difficult and requires a large amount of time-variable physical data.

The operating rules for a water system can be quite simple yet represent the satisfaction of many objectives not directly related to meeting electrical demand. For example, a reservoir's target storage for a given time may result from the desire to minimize the risk of flooding downstream while still providing sufficient storage to meet recreational and consumptive use objectives, under the most likely inflow scenarios. In addition, there may be policy-driven minimum and maximum limits on the reservoir releases and therefore, the power output. There may also be constraints on the flow at various points in the river system necessary to meet biological and other ecosystem needs, as well as providing sufficient water to meet consumptive use demands.

2.5 Current Modeling Strategies

In its most general form, the STRS problem is a large, complex, stochastic, non-linear, and non-convex optimization problem that can not currently be solved to optimality for realistic problem sizes. The primary sources of uncertainty are in the electrical demand, the availability of each generating resource at a given time, and the hydrologic inflows. The primary sources of the non-linearities are the thermal costs curves and the mapping of the hydrologic variables to the power variables. The primary source of the non-convexity arises from the discrete decision of which thermal units should be on- or off-line for each time step, which is necessary to capture the non-trivial fixed and start-up costs associated with the thermal units. A major source of complexity results from the need to simulate the movement of the water throughout the physical network to meet the water system objectives and constraints.

To date, the general STRS problem has been simplified by:

1. assuming that the problem is deterministic by using appropriate forecasts of the stochastic variables, and
2. decomposing the problem into subproblems which are easier to solve and then combining those solutions in some way to yield a solution to the original problem. Such decompositions have historically been along the lines of separating the STRS problem into thermal and hydro subproblems.

Even with these simplifications, the STRS problem is still a difficult one. For large, complex systems, the resulting thermal and hydro subproblems may still be impossible to solve to optimality. In addition, the coordination of the thermal and hydro subproblem solutions is not guaranteed to produce an optimal solution to the overall STRS problem.

2.5.1 The Thermal Subproblem

The thermal subproblem itself can be thought to consist of two parts: deciding which thermal units should be on-line (the unit commitment decision) and the assigning of the generation levels to each on-line unit (the economic dispatch decision). The major complication lies in the necessity to represent the fixed and start-up costs associated with the thermal units. The trade-off is in having too many units committed versus incurring unnecessary start-up costs. The unit commitment decision is generally represented by binary variables in the model formulation that denote whether the particular unit is off- or on-line.

It is natural to view the thermal subproblem (and the overall STRS problem as well) as a sequential decision problem where current decisions affect future costs. Therefore, dynamic programming has been investigated for application to these problems, but the size of the necessary state space severely limits the size of the problems that can be solved (Wood and Wollenberg 1984).

Some entirely heuristic approaches to solving the unit commitment problem have been proposed (Wood and Wollenberg 1984; Sheble 1990). These approaches use priority lists to mimic the real-time decision making of a power operator but do not provide any information about how close to optimality the resulting solution may be. Other work has used similar heuristics to limit the decision space to a tractable size so that a dynamic programming solution could then be found (Erwin et al. 1991), but again, with the loss of an estimate of how good the solution may be.

Similarly, mixed-integer programming using branch and bound also suffers in its ability to solve problems of realistic size (Choen and Sherkat 1987). For example, a system with 100 units scheduled over a week results in a model containing over 16,000 binary decision variables. The linear programming relaxations implemented in traditional branch and bound may not generate sufficiently tight bounds to effectively limit the search space and this observation has directed the more recent work to methodologies which can generate tighter bounds for branch and bound methodologies.

Based on the work of Geoffrion and others (Geoffrion 1974; Fisher 1975), Lagrangian relaxation was first applied to a simple unit commitment problem (Muskadt and Koenig 1977) as a technique to generate tighter bounds within a branch and bound strategy. This idea was advanced in 1982 when it was shown that under certain conditions, the tighter bounds obtained with a particular Lagrangian relaxation resulted in solutions within 0.5% of optimality by examining a single node of the branch and bound tree (Lauer, et al. 1982). More recent work has used the same relaxation strategy to generate tight, albeit infeasible, lower bound solutions that can be used in a procedure to derive feasible solutions, without the use of a branch and bound methodology (Shaw and Bertsekas 1985; Ferreira et al. 1988).

2.5.2 The Hydro Subproblem

Within the STRS problem, it is generally acceptable to model the hydro resources at the plant level rather than on the individual turbine level, using curves similar to Figure 2.3 which represent the output of the plant as a whole (Cohen and Sherkat 1987). By modeling at the plant level, many constraints on the individual turbines (such as avoiding rough zones) can be ignored. These constraints are met in practice by using a near real-time hydro unit commitment/economic dispatch program as is shown in Figure 2.1. Such programs are used to maximize the efficiency of the plant using the hydro plant generation targets from the STRS (Choen and Sherkat 1987).

The hydro subproblem then consists of determining the optimal generation from each hydro plant, while meeting the constraints imposed by the watershed network and various other objectives and policies as previously discussed. The major complications lie in determining the value of the hydro generation, especially in the case of multiple objectives, and in the simulation of the watershed in sufficient detail to ensure that the water system objectives and constraints are met.

A common method to determine the economic value of the hydro energy at each time step is based on application of the first order Kuhn-Tucker (K-T) necessary conditions for optimality in a constrained optimization problem. This determination is easily explained by considering a simplified economic dispatch thermal subproblem where the unit commitment decision is fixed and only the equality demand constraint and min/max capacity constraints for each unit are considered. Then the K-T conditions simply state that the marginal cost of generation for all units, not at their minimum or maximum capacity, must be equal. The marginal value at that time step is given by the Lagrange multiplier on the demand constraint and is often referred to as the "system lamda" in the literature (Cohen and Sherkat 1987). The hydro subproblem objective function can then be expressed as maximizing the value of the hydro over the time horizon resulting in utilization of the hydro generation at those times when the system lamda is relatively high.

More complicated economic value functions for hydro generation may be derived which also incorporate the fuel-constrained nature of the hydro. These functions are based on the particular utility's resource mix, estimates of the future electrical demand, and the firm water availability. For example, the Tennessee Valley Authority uses extensive forecasting and analysis of available resources and system requirements to determine the mean incremental value of hydro energy in storage on a weekly time step over several months (Shane and Boston 1983). The hydro subproblem then considers both the system lamda and the value of the hydro in storage when determining the optimal generation schedule. Recent work has used this value function in a goal programming approach to model the competing demands on the operation of reservoir systems (Magee et al. 1995).

Once the value function has been established, the general hydro subproblem can be viewed as a non-linear optimal control problem. The non-linearities arise from the mapping of flow to energy and from the non-linear equations necessary to simulate the movement of the water throughout the watershed. Dynamic programming has been applied to the optimal control problem with limited success only on small systems (two or three reservoirs) (Choen and Sherkat 1987). Successive approximation methods have also been applied to break the multi-reservoir systems down into smaller subsystems to reduce the state size for the dynamic programs (Giles and Wunderlich 1981).

Most work has concentrated on linearizing the constraint set by greatly simplifying the modeling of the flow of water in the river channels. These methods typically model the flow with simple time lags or assume no time lag whatsoever (Shaw and Bertsekas 1985) so that the watershed may be represented as a pure network, allowing for fast solutions using network algorithms. Given recent advances in solving generalized networks (Glover et al. 1992), it may be possible to achieve reasonable solutions for networks with time lags and gain/loss coefficients which will improve the accuracy of the water model.

The non-linear objective function is usually linearized about a nominal operating point or approximated with a piecewise-linear convex approximation (Ferreira et al. 1988), although reduced gradient methods have also been applied to the problem with a non-linear objective function and linear constraint sets (Lafond 1981; Rosenthal et al. 1981). Other complexities arise if non-network constraints must be included in the hydro subproblem, such as a minimum hydro generation constraint (Ea and Monti 1985).

As noted, the advantage of simplifying the modeling of the flow in river channels is that the hydro subproblem can often be solved with standard optimization methodologies; the disadvantage is that the ability to predict water flows downstream of a controlled facility (which may be critical to meeting the objectives of the water system) is greatly impaired.

2.5.3 Coordination of the Thermal and Hydro Subproblem Solutions

The problem in coordinating the subproblem solutions is most clearly seen by considering the earlier discussion of the system lamda value. Since the thermal subproblem solution depends on knowing the hydro generation and conversely, the hydro subproblem solution must know the system lamda value, some iteration between the two subproblems is necessary (conceptually depicted in Figure 2.4).

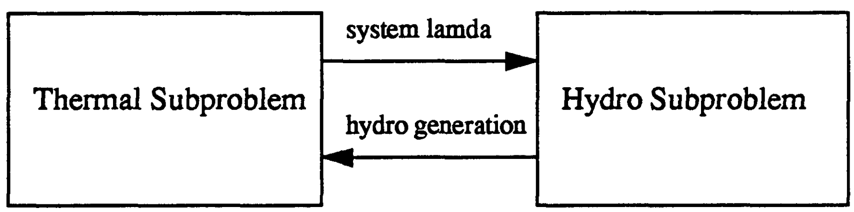

Figure 2.4. Conceptual View of the Hydro-thermal Coordination Procedure

Heuristic decomposition schemes follow such sequential coordination procedures (Duncan et al. 1985) and can suffer from convergence problems (Ferreira et al. 1988). Recently, however, this coordination has been accomplished in parallel (with a guarantee of convergence) by applying large-scale optimization decomposition techniques, such as Lagrangian relaxation and Benders decomposition (Shaw and Bertsekas 1985; Habibollahzadeh and Bubenko 1986; Ferreira et al. 1988). These techniques, however, must simplify the physical watershed constraints to practically solve the hydro subproblem in parallel.

2.6 Motivation and Contributions of this Research

Given that the STRS problem has not been solved to optimality for realistic problem sizes and the potential for significant cost savings for large utilities, the opportunity for improvement remains and is well worth pursuing. This research continues and extends the basic philosophy of decomposition of the STRS problem. In particular, the focus is in two areas:

1. To overcome the limitations imposed by simplifying the water model in current decomposition strategies, constraints on the total amount of energy available from the hydro plants are generated from a separate water model and then used in the STRS model, and
2. To obtain tighter lower bounds on the STRS problem than current relaxation strategies, a new relaxation based on Lagrangian decomposition theory has been developed.

The focus on the water model is driven by the need to better represent the constraints and non-electrical objectives for many water systems, particularly in the Western U.S. In many of these systems, objectives such as meeting water deliveries for consumptive use, providing minimum stream flows for endangered species, and creating a more natural habitat in the river systems are the primary objectives of the system and generating electrical power, although important, is secondary. Detailed water models can be used to set target releases for variable time horizons to ensure that these complex water objectives and constraints are met. The energy that will be generated from these target releases can then be provided as constraints to the STRS model.

The contributions of this area of focus are in the simplified formulation of the STRS model and the development of a methodology based on linear systems and adaptive filtering theory which for some systems, can be used to more accurately model the flow of water in river channels without the need for more complicated hydraulic measurements and modeling techniques.

The focus on obtaining tighter lower bounds on the minimal cost solution is motivated by the inability to currently solve even the thermal subproblem to optimality for large systems. These tighter bounds are valuable in three ways:

1. Tighter lower bounds can greatly improve the performance of branch and bound search,
2. Good lower bounds provide an evaluation of heuristic solutions since optimal solutions are unattainable (the question is how good are these solutions and how much can they be improved), and
3. Solutions to relaxed problems can often lead to solutions to the original problem via perturbation heuristics.

The contributions are two related methods to generate better bounds. First, a new Lagrangian relaxation for the STRS problem has been developed, that under certain conditions, can generate tighter bounds compared to the previously published relaxation (Lauer et al. 1982; Ferreira et al. 1988). Second, a Lagrangian decomposition has been developed for the STRS problem, which achieves the benefits of both the new relaxation and the traditional relaxation. This decomposition is guaranteed to generate bounds at least as tight as the individual relaxations, and for some problems, generates strictly greater bounds. In addition, Lagrangian decomposition generates better bounds with computational effort similar (on the same order of magnitude) to the individual Lagrangian relaxations.

Chapter 3

MODEL FORMULATION

Many different simplifications to the general STRS problem may be made depending upon the particular application and solution methodology applied and this research follows that philosophy. In this chapter, appropriate notation is developed and the model formulation is presented in terms of the thermal resources, the hydro resources, and the remaining system-level constraints. A summary of the overall model is then given. The chapter concludes with a review of the simplifications made and a discussion of the possible extensions to handle more general versions of the STRS problem.

3.1 Thermal Resources

The non-linear cost curves for each thermal resource (shown in Figure 2.2) are parameterized with linear functions represented by parameters c and a, where c is the slope of the line (variable cost of generation), and a is the intercept of the line at zero generation level. Start-up costs are included; however, it is assumed that those costs are constant over the scheduling horizon (i.e., they do not depend upon how long the unit has been down).

The index on time is denoted by t = {1,2,3,... T} and the index on resources (units) as i = {1,2,3,...,I}. Let continuous, non-negative variables $g_{i,t}$, represent the generation level of each resource in each hour. To account for the fixed costs, binary unit commitment variables, $x_{i,t}$ for each unit i and for each hour t (where $x_{i,t} = 1$ if unit i is operating in hour t and 0 if not) are also defined. Let the set K contain all units with start-up costs and for $i \in K$, define binary start-up variables, $y_{i,t}$, where $y_{i,t} = 1$ if unit i was started in hour t and 0 if not.

With these definitions, the thermal cost function can be written as

$$\sum_t \left\{ \sum_{i \in K} (b_i \cdot y_{i,t}) + \sum_i (a_i \cdot x_{i,t} + c_i \cdot g_{i,t}) \right\} \qquad \text{(Equation 3.1)}$$

Internal combustion turbines can also be represented by this notation but typically have negligible startup costs since very little fuel is used to start them (i.e., b_i is zero).

The minimum (m_i) and maximum (M_i) limits on the generation levels of each thermal resource can be written as

$$\forall i \in K, \forall t: \qquad -g_{i,t} + M_i \cdot x_{i,t} \geq 0 \qquad \text{(Equation 3.2)}$$

$$\forall i \in K, \forall t: \qquad g_{i,t} - m_i \cdot x_{i,t} \geq 0 \qquad \text{(Equation 3.3)}$$

Writing the capacity constraints on the $g_{i,t}$ in this manner limits the generation between the minimum and maximum levels, but also forces the correct behavior of the unit commitment variables, $x_{i,t}$. When a unit generates, $g_{i,t} \geq 0$, forcing $x_{i,t} = 1$.

To handle the constraints on ramping rates, let set J contain all units that have ramping constraints, and for $i \varepsilon J$, define r_i to be the maximum rate that the unit can be ramped down in an hour, and R_i as the maximum rate that the unit can be ramped up in an hour. Then the ramping constraints can be written as

$$\forall i \in J, \forall t: \qquad g_{i,t} - g_{i,t-1} + r_i \geq 0 \qquad \text{(Equation 3.4)}$$

$$\forall i \in J, \forall t: \qquad -g_{i,t} + g_{i,t-1} + R_i \geq 0 \qquad \text{(Equation 3.5)}$$

An additional constraint to relate the unit commitment and start-up variables, forcing the correct behavior of the start-up variable $y_{i,t}$, is written as

$$\forall i \in K, \forall t: \qquad y_{i,t} - x_{i,t} + x_{i,t-1} \geq 0 \qquad \text{(Equation 3.6)}$$

With the $x_{i,t}$ defined as binary variables, these start-up constraints limit the $y_{i,t}$ to be binary variables, without explicitly defining them as such.

Units with variable startup costs typically have additional constraints on the start-ups, including the minimum time a unit must stay up or down if turned on or off. Since the resources with variable start-up costs have been ignored, the assumption is also made that units do not have minimum up or down time constraints.

3.2 Hydro Resources

For the applications of interest in this research, the hydro subproblem is modeled in a separate application that provides surrogate constraints to the STRS model. The hydro model is typically a daily time step model that accounts for the multiple objectives of the water system, including meeting consumptive use demands for irrigation and municipalities, meeting minimum flow constraints for recreational and environmental requirements, and minimizing damage due to flooding by meeting monthly target reservoir contents. To achieve the required accuracy, these models are often detailed simulation models which may be run with several alternative input scenarios.

The hydro model produces target water releases (which are then converted to target energy values) over appropriate time horizons that ensure that the water objectives are met. The time horizons over which the target energies must be met are allowed to vary at each hydro plant. For example, some plants may be constrained over each day or for others, the constraint horizon may be an entire week. By providing target energy values to the STRS model, short-term flexibility in the hydro generation is retained, while still meeting the water objectives.

Let set H contain all units with these cumulative energy constraints and then for i ε H, define N_i to be the number of time sub-horizons over which a cumulative constraint exists for plant i, n_i to be the index over those horizons, $T_i^{n_i}$ to be the set containing the time periods for each sub-horizon, n_i, and $G_i^{n_i}$ to be the target energy level for that sub-horizon. The resulting constraints can be written as

$$\forall i \in H, \quad n_i = \{1, \ldots, N_i\}: \qquad \sum_{t \in T_i^{n_i}} g_{i,t} = G_i^{n_i} \qquad \text{(Equation 3.7)}$$

An example for a plant with constraints over the first day, the next 4 days, and the final 2 days of a one-week horizon ($N_i = 3$) is shown in Figure 3.1.

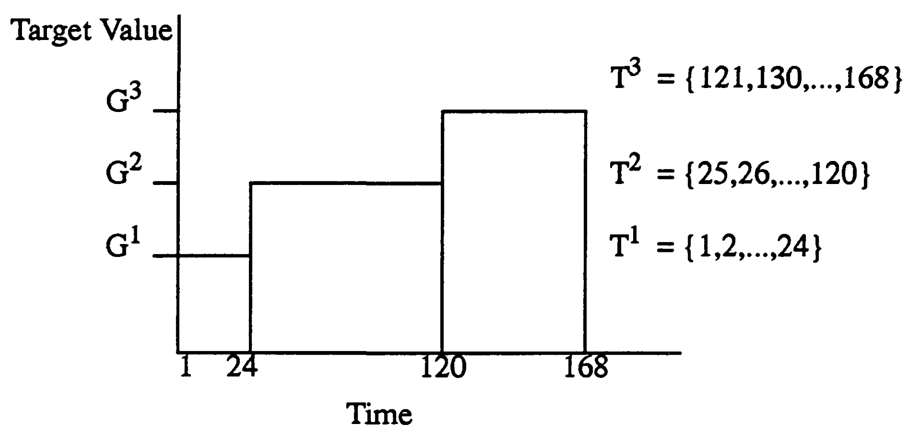

Figure 3.1. Definition of Sets for a Unit i with Cumulative Constraints Over N = 3 Time Sub-horizons

Each hydro plant is limited to a minimum and maximum capacity of generation, the limits being determined by physical and/or policy constraints. With the assumption that these limits are not dependent on operating head, the capacity constraints for hydros may be written as

$$\forall i \in H, \forall t: \qquad -g_{i,t} + M_i \geq 0 \qquad \text{(Equation 3.8)}$$

$$\forall i \in H, \forall t: \qquad g_{i,t} - m_i \geq 0 \qquad \text{(Equation 3.9)}$$

Given the cumulative energy constraints from the hydro model, the marginal cost of the hydro resources is assumed to be negligible within the horizon ($c_i = 0$, $i \in H$).

3.3 System-level Constraints

Recall that the primary system-level constraint is to meet the electrical demand, although additional constraints may be added to ensure that there is sufficient generating reserve in the system at any given time. In this formulation, only constraints for spinning reserve are included and transmission constraints are ignored.

Let d_t be the forecasted demand for each time step t. Then the requirement that system demand be met is given by

$$\forall t: \qquad \sum_i g_{i,t} = d_t \qquad \text{(Equation 3.10)}$$

Let s_t be the spinning reserve requirement for each time step t. For units with ramping limits, the contribution to spinning reserve may be limited by the ability of the unit to ramp up within the hour. Therefore, continuous, non-negative variables, $p_{i,t}$, for i εJ, are defined. These variables serve to dynamically represent the contribution to spinning reserve for each unit i εJ for each time step t. The maximum contribution to spinning reserve of any unit is assumed to be its maximum capacity, M_i. The spinning reserve constraint and the capacity constraints on the $p_{i,t}$ are written as

$$\forall t: \quad \sum_{i \varepsilon J} p_{i,t} + \sum_{i \notin J} M_i \cdot x_{i,t} \geq s_t \qquad \text{(Equation 3.11)}$$

$$\forall i \in J, \forall t: \quad -p_{i,t} + M_i \geq 0 \qquad \text{(Equation 3.12)}$$

$$\forall i \in J, \forall t: \quad -p_{i,t} + g_{i,t} + R_i \geq 0 \qquad \text{(Equation 3.13)}$$

3.4 The STRS Model

The version of the STRS problem considered in this research can now be stated formally as the following mixed-integer linear program (Problem MIP):

$$z_{MIP} = \min \sum_t \left\{ \sum_{i \in K} (b_i \cdot y_{i,t}) + \sum_i (a_i \cdot x_{i,t} + c_i \cdot g_{i,t}) \right\} \qquad \text{(MIP)}$$

Subject to

$\forall i, \forall t: \quad -g_{i,t} + M_i \cdot x_{i,t} \geq 0$

$\forall i, \forall t: \quad g_{i,t} - m_i \cdot x_{i,t} \geq 0$

$\forall i \in J, \forall t: \quad -p_{i,t} + M_i \cdot x_{i,t} \geq 0$

$\forall i \in J, \forall t: \quad -p_{i,t} + g_{i,t} + R_i \cdot x_{i,t} \geq 0$

Block 1:
capacity constraints

$\forall t: \quad \sum_i g_{i,t} = d_t$

$\forall t: \quad \sum_{i \in J} p_{i,t} + \sum_{i \notin J} M_i \cdot x_{i,t} \geq s_t$

Block 2:
demand and spinning reserve constraints

$\forall i \in J, \forall t: \quad g_{i,t} - g_{i,t-1} + r_i \cdot x_{i,t-1} \geq 0$

$\forall i \in J, \forall t: \quad -g_{i,t} + g_{i,t-1} + R_i \cdot x_{i,t} \geq 0$

$\forall i \in K, \forall t: \quad y_{i,t} - x_{i,t} + x_{i,t-1} \geq 0$

$\forall i \in H, \ n_i = \{1, \ldots, N_i\}: \quad \sum_{t \in T_i^{n_i}} g_{i,t} = G_i^{n_i}$

Block 3:
ramping, start-up, and
cumulative constraints

where

$$\forall i, \forall t: \quad x_{i,t} \in \{0, 1\}$$

$$\forall i, \forall t: \quad g_{i,t} \in \Re^+$$

$$\forall i \in J, \forall t: \quad p_{i,t} \in \Re^+$$

$$\forall i \in K, \forall t: \quad y_{i,t} \in \{\Re : 0 \le y_{i,t} \le 1\}$$

$\Re$, $\Re^+$ denote the sets of real and non-negative real numbers

A few comments are in order. The capacity constraints have been written for all units in term of the unit commitment variables, $x_{i,t}$. For the hydro resources, it is assumed that $x_{i,t} = 1$. The formulation of the spin capacity constraints (Equation 3.12 and Equation 3.13) was strengthened by multiplying the constant terms (M_i and R_i) by the commitment variable, $x_{i,t}$. The ramping constraints (Equation 3.4 and Equation 3.5) were also strengthened in a similar way (Magee, 1996).

The reasons for associating the various constraints into blocks will become apparent in the next chapter where alternative relaxations of this problem are developed by relaxing particular blocks of constraints.

3.5 Limitations of this Model and Possible Extensions

The STRS model formulated in this chapter represents the heart of the problem and the simplifications do not affect the primary goal of illustrating the power of the new approach to obtain tighter bounds.

To summarize, the following simplifications have been made:

1. The non-linear cost curves for each thermal unit have been parameterized with linear functions. This assumption allows the use a linear programming-based branch and bound program to solve the decomposed subproblems associated with the relaxation methodologies. This was done for ease in implementation. Non-linear cost functions could be included at the expense of using a different solution methodology for the subproblems, such as dynamic programming.
2. The start-up costs for the thermal units have been assumed to be constant and minimum up and down time constraints have been ignored. Several authors have presented formulations that include variable start-up costs and constraints on up and down times (Lauer et al. 1982; Ferreira et al. 1988). These effects were ignored primarily because the necessary data to model them was unavailable and could be included in the future.
3. Only spinning reserve constraints have been considered. Other reserve constraints may be important for real applications and could be included in this formulation. These may include other classes of reserve and limitations that force the reserve to

be distributed throughout the system to avoid transmission limitations (Ferreira et al. 1988).

4. The hydro resources have been modeled on a plant level via cumulative (over time) energy constraints. For the applications of interest in this research, the water systems are modeled in a separate application to meet the water constraints and objectives first, providing surrogate constraints for the electrical system. For areas where detailed water models are not necessary, a simpler water model could be embedded, at some additional computational expense at the subproblem level.
5. Interchange, transmission, or multiple control area constraints have been ignored. Interchange and control area constraints could be expressed in terms of capacity limits on particular resources. Transmission effects (including line losses) can be included using penalty factors that depend on the relative locations of the generating plants within the control area (Choen and Sherkat 1987). This formulation could be extended to handle these constraints if necessary.

Even with these simplifications, the resulting STRS model is still a very challenging problem. Application of straightforward branch and bound fails to solve even this simplified problem to optimality for large systems, as one might expect based on the size of the decision space.

Chapter 4

LAGRANGIAN RELAXATION AND LAGRANGIAN DECOMPOSITION APPLIED TO THE STRS PROBLEM

The motivation for solving relaxed versions of difficult optimization problems is that the relaxed problem is easier to solve and yet the optimal value of the relaxed problem provides an optimistic bound on the optimal solution of the original problem. As one might expect, there is a trade-off in solving the relaxed problem to optimality (to get the tightest bound) versus the time and effort in finding that solution. When coupled with heuristic methods that generate feasible solutions, relaxations and feasible solutions can be used to bracket the range of possible solutions to the original problem. Such bracketing is crucial to search processes such as branch and bound.

In this chapter, a brief background on Lagrangian relaxation is given. Using the model formulated in Chapter 3, the Lagrangian relaxation that has traditionally been applied to the STRS problem is outlined and a competing relaxation is described. The Lagrangian decomposition methodology is then presented which gains the benefits of both of the Lagrangian relaxations. The chapter concludes with an exploration of the relationships between these approaches.

4.1 Lagrangian Relaxation

Of the many relaxation strategies, Lagrangian relaxation is the most prominent, in part because of the elegant theory that underlies it. It has also been proven to be a valuable computational tool and has been successfully used to solve some difficult mixed-integer programming problems (Fisher 1981). The approach is based on the theories set forth by Everett and further developed for discrete problems by Geoffrion (Everett 1963; Geoffrion 1974).

A problem is generally amenable to solution via Lagrangian relaxation if the constraint set is essentially easy to solve except for some subset of "complicating" constraints. The problem is then relaxed in those complicating constraints by eliminating them from the constraint set, but also penalizing any infeasibility in them in the objective function. The key theoretical result is that the optimal solution to the Lagrangian relaxed problem dominates (gives a bound that is no worse than) the bound achieved from the linear programming relaxation (Geoffrion 1974).

4.2 The Traditional Lagrangian Relaxation of the STRS Problem

With this background, the STRS model may be viewed in terms of defining the set of complicating constraints. To date, the traditional Lagrangian relaxation for the STRS problem (Ferreira et al. 1988; Shaw and Bertsekas 1985; Lauer et al. 1982) has been to relax the constraints that couple units (Block 2). This relaxation for the MIP formulation of STRS problem is given by (Problem LRU)

$$z_{LRU}(u, v) = min \sum_t \left\{ \sum_{i \in K} (b_i \cdot y_{i,t}) + \sum_i (a_i \cdot x_{i,t} + c_i \cdot g_{i,t}) \right. \qquad \text{(Equation 4.1)}$$
$$\left. - \left(\sum_i g_{i,t} - d_t \right) \cdot u_t - \left(\sum_{i \varepsilon J} p_{i,t} + \sum_{i \notin J} M_i \cdot x_{i,t} - s_t \right) \cdot v_t \right\}$$

$$u_t, v_t \in \Re, v \geq 0$$

subject to the remaining constraints (Block 1 and Block 3) and the restrictions on the variables given in the original problem. The new coefficients, u_t, and v_t, are the Lagrange multipliers (or penalties) for violations in the relaxed constraints and are constant in the minimization.

Notice that the relaxed problem is separable by *unit*: a problem for each unit over all time steps can be solved independently. Each unit subproblem contains binary variables equal to the total number of time steps. Methodologies for solving the unit subproblems include dynamic programming and branch and bound.

An obvious question is what are the best values of the multipliers (the values that will give the tightest lower bound to the original problem). Formally, this problem is the Lagrangian dual (Nemhauser and Wolsey 1988), given by

$$z^{*}_{LRU} = \max_{u,v}\{z_{LRU}(u,v)\} \qquad \text{(Equation 4.2)}$$

Methodologies to solve such dual problems are discussed in Chapter 5.

4.3 A Competing Relaxation of the STRS Problem

From the original formulation, one can see that there is another competing relaxation: the relaxation with respect to the constraints that couple time steps (Block 3). This new relaxation is given by (Problem LRT)

$$z_{LRT}(u,v,w,q) = \min \sum_{t}\left\{\sum_{i\in K}(b_i \cdot y_{i,t}) + \sum_{i}(a_i \cdot x_{i,t} + c_i \cdot g_{i,t})\right. \qquad \text{(Equation 4.3)}$$

$$-\sum_{i\in J}[(g_{i,t} - g_{i,t-1} + r_i \cdot x_{i,t-1}) \cdot u_{i,t} + (-g_{i,t} + g_{i,t-1} + R_i \cdot x_{i,t}) \cdot v_{i,t}]$$

$$\left.-\sum_{i\in K}(y_{i,t} - x_{i,t} + x_{i,t-1}) \cdot w_{i,t}\right\} - \sum_{i\in H}\left\{\sum_{n_i=1}^{N_i}\left[\sum_{t\in T_i^{n_i}}\left(g_{i,t} - G_i^{n_i}\right) \cdot q_{i,n_i}\right]\right\}$$

$$u_{i,t}, v_{i,t}, w_{i,t}, q_{i,ni} \in \Re,\ u, v, w \geq 0$$

subject to the remaining constraints (Block 1 and Block 2), and the restrictions on the variables given in the original problem. The Lagrange multipliers, u, v, w, and q, (which are not related to those in the previous relaxation) are again constant in the minimization.

Notice that this relaxed problem is now separable by *time*: a problem for each time step for all units can be solved independently. Each time subproblem contains binary variables equal to the total number of units. Methodologies for solving the time subproblems also include dynamic programming and branch and bound.

The multipliers that give the best bound for this relaxation can also be found by solving another dual problem given by

$$z^*_{LRT} = \max_{u, v, w, q} \{z_{LRT}(u, v, w, q)\} \qquad \text{(Equation 4.4)}$$

A natural question now arises: for the STRS problem, which of the Lagrangian relaxations is the "best" (i.e., which relaxation gives the tightest bounds for reasonable computational effort)? This question is unanswerable in general, but fortunately, the benefit of both of these relaxations can be gained via Lagrangian decomposition.

4.4 Lagrangian Decomposition of the STRS Problem

As discussed in the previous section, the STRS problem has two distinct structures that lead to two competing Lagrangian relaxations. Problems containing such structures may be exploited using the Lagrangian decomposition (or layering) strategy developed independently in the late 1980s (Guignard and Kim 1987; Glover and Kingman 1988). Both groups proved that the bound obtained from the optimal solution of the Lagrangian decomposition dominates the bound achieved from the optimal solution of any of the individual Lagrangian relaxations. The interest in this technique stems from this fact, and that often the relaxed solution to the decomposed problem can be reliably converted to good, feasible solutions to the original problem.

Lagrangian decomposition takes advantage of the competing relaxations (and therefore generates a better bound) by converting the original problem into a new, but equivalent problem and then applying a new Lagrangian relaxation. The decomposition stage is developed by first defining duplicate variables for each layer and then adding additional constraints that the variables from each layer must be equal. A Lagrangian relaxation on those equality constraints completes the development (Magee and Glover 1996). In this sense, Lagrangian decomposition is a type of Lagrangian relaxation and the new decomposed problem can be solved using techniques appropriate for solving other Lagrangian relaxations.

First define duplicate variables for each layer

$g^U_{i,t}, x^U_{i,t}$ to be the generation and commitment variables for the unit layer ,

$g^T_{i,t}, x^T_{i,t}$ to be the generation and commitment variables for the time layer, and write an equivalent problem for (MIP), where the true costs have arbitrarily been attached to the unit subproblems and the other variables are marked by superscripts U and T to denote in which layer they appear:

$$z_{MIP} = min\sum_t \left\{ \sum_{i \in K} \left(b_i \cdot y_{i,t}^U\right) + \sum_i \left(a_i \cdot x_{i,t}^U + c_i \cdot g_{i,t}^U\right) \right\} \qquad \text{(MIP)}$$

Subject to

$\forall i, \forall t$: $\quad -g_{i,t}^U + M_i \cdot x_{i,t}^U \geq 0$

$\forall i, \forall t$: $\quad g_{i,t}^U - m_i \cdot x_{i,t}^U \geq 0$

$\forall i, \forall t$: $\quad -g_{i,t}^T + M_i \cdot x_{i,t}^T \geq 0$

$\forall i, \forall t$: $\quad g_{i,t}^T + m_i \cdot x_{i,t}^T \geq 0$

$\forall i \in J, \forall t$: $\quad -p_{i,t}^T + M_i \cdot x_{i,t}^T \geq 0$

$\forall i \in J, \forall t$: $\quad -p_{i,t}^T + g_{i,t} + R_i \cdot x_{i,t}^T \geq 0$

Block 1:
capacity constraints

$$\forall t: \quad \sum_i g^T_{i,t} = d_t$$

$$\forall t: \quad \sum_{i \in J} p^T_{i,t} + \sum_{i \notin J} M_i \cdot x^T_{i,t} \geq s_t$$

Block 2:
demand and spinning reserve constraints

$$\forall i \in J, \forall t: \quad g^U_{i,t} - g^U_{i,t-1} + r_i \cdot x^U_{i,t} \geq 0$$

$$\forall i \in J, \forall t: \quad -g^U_{i,t} + g^U_{i,t-1} + R_i \cdot x^U_{i,t} \geq 0$$

$$\forall i \in K, \forall t: \quad y^U_{i,t} - x^U_{i,t} + x^U_{i,t-1} \geq 0$$

$$\forall i \in H, \; n_i = \{1, \ldots, N_i\}: \quad \sum_{t \in T^{n_i}_i} g^U_{i,t} = G^{n_i}_i$$

Block 3:
ramping, start-up, and cumulative constraints

$$\forall i, \forall t: \quad x^U_{i,t} - x^T_{i,t} = 0$$

$$\forall i, \forall t: \quad g^U_{i,t} - g^T_{i,t} = 0$$

Block 4:
Equality constraints on duplicate variables

where

$$\forall i, \forall t: \quad x^U_{i,t}, x^T_{i,t} \in \{0, 1\}$$

$$\forall i, \forall t: \quad g^U_{i,t}, g^T_{i,t} \in \Re^+$$

$$\forall i \in J, \forall t: \quad p^T_{i,t} \in \Re^+$$

$$\forall i \in K, \forall t: \quad y^U_{i,t} \in \left\{\Re : 0 \le y^U_{i,t} \le 1\right\}$$

This equivalent problem is known as a layered representation of the original problem (Glover and Klingman 1988), where the problem has been decomposed into two layers (the unit, **U**, and time, **T**, layers) with variables for each layer. The capacity constraints on the units appear with both sets of variables since they do not complicate either layer and will, in fact, give better solutions for each. Note also that Block 4 constraints (equality of the variables from each layer) have now been added. If the Block 4 constraints are now relaxed in the layered representation, the Lagrangian decomposition problem can be written as

$$z_{LD}(u, v) = min \sum_t \left\{ \sum_{i \in K} \left(b_i \cdot y^U_{i,t}\right) + \sum_i \left(a_i \cdot x^U_{i,t} + c_i \cdot g^U_{i,t}\right) + \sum_i \left[-\left(x^U_{i,t} - x^T_{i,t}\right) \cdot u_{i,t} - \left(g^U_{i,t} - g^T_{i,t}\right) \cdot v_{i,t} \right] \right\} \quad \text{(Equation 4.5)}$$

where

$$u_{i,t}, v_{i,t} \in \Re$$

subject to the remaining constraints (Blocks 1, 2, and 3) and the restrictions on the variables previously given. The $u_{i,t}$ are the Lagrange multipliers for the equality constraints on the unit commitment variables and $v_{i,t}$ are the multipliers for the equality constraints on the generation variables. These u and v are not related to the variables in either Equation 4.1 or Equation 4.3.

The Lagrangian decomposition problem can now be separated into two subproblems: the *unit* layer and the *time* layer. The objective function (Equation 4.5) can be re-written as

$$z_{LD} = z_{LDU}(u, v) + z_{LDT}(u, v)$$

where

$$z_{LDU}(u, v) = \min \sum_t \left\{ \sum_{i \in K} \left(b_i \cdot y_{i,t}^U \right) + \sum_i \left[(a_i - u_{i,t}) \cdot x_{i,t}^U + (c_i - v_{i,t}) \cdot g_{i,t}^U \right] \right\}$$

and

$$z_{LDT}(u, v) = \min \sum_t \sum_i \left\{ u_{i,t} \cdot x_{i,t}^T + v_{i,t} \cdot g_{i,t}^T \right\}$$

Then the unit layer subproblem can be written as (Problem LDU)

$$z_{LDU}(u,v) = min\sum_t \left\{ \sum_{i \in K} \left[b_i \cdot y_{i,t}^U \right] + \sum_i \left[(a_i - u_{i,t}) \cdot x_{i,t}^U + (c_i - v_{i,t}) \cdot g_{i,t}^U \right] \right\}$$

Subject to

$\forall i, \forall t$: $\quad -g_{i,t}^U + M_i \cdot x_{i,t}^U \geq 0$

$\forall i, \forall t$: $\quad g_{i,t}^U - m_i \cdot x_{i,t}^U \geq 0$

$\forall i \in J, \forall t$: $\quad g_{i,t}^U - g_{i,t-1}^U + r_i \cdot x_{i,t}^U \geq 0$

$\forall i \in J, \forall t$: $\quad -g_{i,t}^U + g_{i,t-1}^U + R_i \cdot x_{i,t}^U \geq 0$

$\forall i \in K, \forall t$: $\quad y_{i,t}^U - x_{i,t}^U + x_{i,t-1}^U \geq 0$

$\forall i \in H,\ n_i = \{1, \ldots, N_i\}$: $\quad \sum_{t \in T_i^{n_i}} g_{i,t}^U = G_i^{n_i}$

where

$$\forall i, \forall t: \quad x_{i,t}^{U} \in \{0, 1\}$$

$$\forall i, \forall t: \quad g_{i,t}^{U} \in \Re^{+}$$

$$\forall i \in K, \forall t: \quad y_{i,t}^{U} \in \left\{\Re : 0 \le y_{i,t}^{U} \le 1\right\}$$

and the time layer subproblem is given by (Problem LDT)

$$z_{LDT}(u, v) = min \sum_{t} \sum_{i} \left\{u_{i,t} \cdot x_{i,t}^{T} + v_{i,t} \cdot g_{i,t}^{T}\right\}$$

Subject to

$$\forall i, \forall t: \quad -g_{i,t}^{T} + M_i \cdot x_{i,t}^{T} \ge 0$$

$$\forall i, \forall t: \quad g_{i,t}^{T} - m_i \cdot x_{i,t}^{T} \ge 0$$

$$\forall i \in J, \forall t: \quad -p_{i,t}^{T} + M_i \cdot x_{i,t}^{T} \ge 0$$

$$\forall i \in J, \forall t: \quad -p^T_{i,t} + g_{i,t} + R_i \cdot x^T_{i,t} \geq 0$$

$$\forall t: \quad \sum_i g^T_{i,t} = d_t$$

$$\forall t: \quad \sum_{i \varepsilon J} p^T_{i,t} + \sum_{i \notin J} M_i \cdot x^T_{i,t} \geq s_t$$

where

$$\forall i, \forall t: \quad x^T_{i,t} \in \{0, 1\}$$

$$\forall i, \forall t: \quad g^T_{i,t} \in \Re^+$$

$$\forall i \in J, \forall t: \quad p^T_{i,t} \in \Re^+$$

In addition, each layer can be further decoupled by unit and time respectively just as they were in the previous Lagrangian relaxations. Consequently, the unit layer subproblem can be solved by solving each individual unit subproblem and the time layer subproblem can be solved by solving each individual time subproblem.

The task now is to find the optimal values for the multipliers (i.e., the values that will give the maximum lower bound for (MIP)). As in any Lagrangian relaxation, this can be done by solving the Lagrangian dual problem given by

$$z^*_{LD} = \max_{u,\,v} \left\{ z_{LDU}(u, v) + z_{LDT}(u, v) \right\} \qquad \text{(Equation 4.6)}$$

4.5 Relationships of the Three Relaxations

The key to understanding the relationships of the three relaxations lies in the following theorem (Nemhauser and Wolsey 1988): the solution to a Lagrangian dual problem can be found by solving a particular linear program consisting of the original objective function, subject to the complicating constraints and the *convex hull* of the remaining constraints. Applying this theorem to Problem LRU (the relaxation with respect to the Block 2 constraints), the Lagrangian dual (Equation 4.2) can be written as

$$z^*_{LRU} = \sum_t \left\{ \sum_{i \in K} (b_i \cdot y_{i,\,t}) + \sum_i (a_i \cdot x_{i,\,t} + c_i \cdot g_{i,\,t}) \right\}$$

Subject to

$$\forall t: \quad \sum_{i} g_{i,t} = d_t$$

$$\forall t: \quad \sum_{i \in J} p_{i,t} + \sum_{i \notin J} M_i \cdot x_{i,t} \geq s_t$$

Block 2:
demand and spinning reserve constraints

$$\text{conv}\left\{ -g_{i,t} + M_i \cdot x_{i,t} \geq 0,\ g_{i,t} - m_i \cdot x_{i,t} \geq 0,\ -p_{i,t} + M_i \cdot x_{i,t} \geq 0, \right.$$

$$-p_{i,t} + g_{i,t} + R_i \cdot x_{i,t} \geq 0,\ g_{i,t} - g_{i,t-1} + r_i \cdot x_{i,t-1} \geq 0,$$

$$\left. -g_{i,t} + g_{i,t-1} + R_i \cdot x_{i,t} \geq 0,\ \sum_{t \in T_i^{n_i}} g_{i,t} = G_i^{n_i} \right\}$$

Convex hull of Block 1
and Block 3 constraints

where

$$\forall i, \forall t: \quad x_{i,t} \in \{\Re{:}0 \le x_{i,t} \le 1\}$$

$$\forall i, \forall t: \quad g_{i,t} \in \Re^+$$

$$\forall i \in J, \forall t: \quad p_{i,t} \in \Re^+$$

$$\forall i \in K, \forall t: \quad y_{i,t} \in \{\Re{:}0 \le y_{i,t} \le 1\}$$

and *conv*{ } denotes the convex hull of the constraints within { }, defined over their respective sets.

Note that this is a linear program (the integer restriction on the $x_{i,t}$ has been relaxed); however, the convex hull of the constraints that were not relaxed (Block 3) must now be known. This linear program is depicted in Figure 4.1 (adapted from Guignard and Kim 1987). The theorem says that the optimal solution to the relaxation can be found on the intersection of the linear programming relaxation of these relaxed constraints and the convex hull of the remaining constraints (Block 1 and Block 3), shown as the point LRU in Figure 4.1.

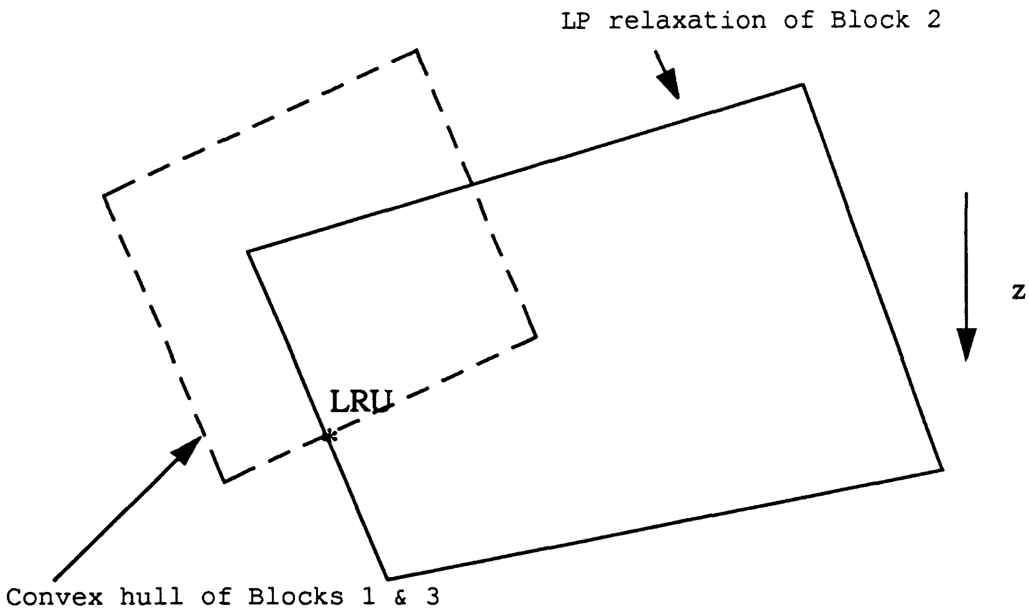

Figure 4.1. A Two-Dimensional Conceptual View of the Linear Programming Solution Equivalent to the Lagrangian Dual for Relaxation in the Block 2 Constraints

An analogous theorem exists for the Lagrangian decomposition: the optimal solution to the Lagrangian dual (where the relaxation is in the equality constraints) lies on the intersection of the convex hulls of the competing constraint sets (Guignard and Kim 1987). That solution (LD) is shown in Figure 4.2. Also shown are the optimal solution to the competing Lagrangian relaxation on the constraints that couple time steps (Block 3) (denoted LRT in Figure 4.2) as well as the solutions to the original problem (MIP) and the linear programming relaxation of the original problem (LP).

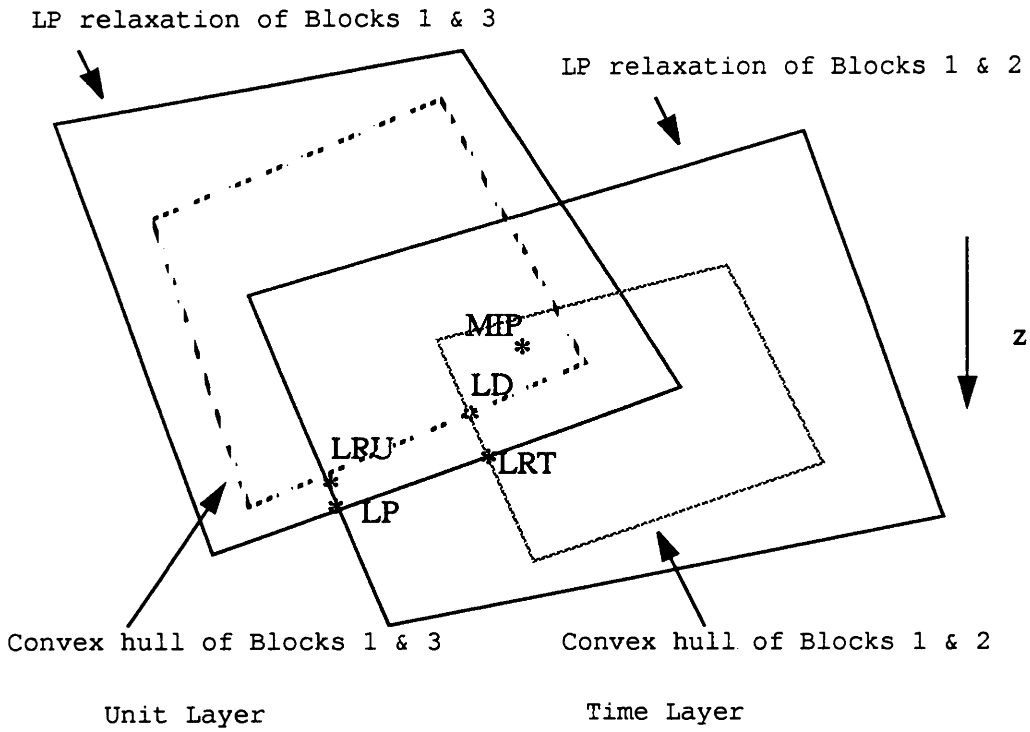

Figure 4.2. A Two-dimensional Conceptual View of the Relaxations and their Optimal Solutions With Respect to the MIP Optimum.

In summary, the key to finding the tightest bound to the STRS problem from a relaxation approach is in solving the Lagrangian dual of the decomposition problem (Equation 4.6). Clearly if the extreme points of the intersection of the convex hulls of the competing constraint sets were known, the tightest bound could be found. Enumerating all extreme points, however, would be prohibitive. This suggests a constructive methodology where only a subset of extreme points necessary to find the optimal solution is generated. In the next chapter, such a methodology is described.

Chapter 5

SOLUTION METHODOLOGY

As previously discussed, the best bound of any Lagrangian relaxation may be found by solving the associated Lagrangian dual problem. In this chapter, a brief review of the techniques for solving these dual problems is given. The methodology chosen (column generation) is detailed for the STRS model. Two heuristics are presented: a priority-based scheme that can be used to produce initial feasible solutions for the column generation technique, and a rounding heuristic that produces good feasible solutions to the original problem from the relaxed solution. Finally, computational results for two test problems are presented and the performance of the methodology is discussed.

5.1 Solving the Dual Problems

Historically, three techniques have been used to solve the dual problems: column generation, subgradient optimization, and multiplier adjustment methods (Magee and Glover 1996). All methodologies are iterative techniques that solve the relaxed problem (given values of the multipliers), adjust the multipliers, and resolve the relaxed problems until some stopping criterion is met.

Column generation solves the Lagrangian dual by generating a "master" linear program. The method used is analogous to the Dantzig-Wolfe decomposition scheme applied to large linear programs (Nemhauser and Wolsey 1988). The linear program has a variable for each subproblem solution (i.e., for each iteration) which represents the contribution of that subproblem to the overall solution. The objective function coefficient for each variable is the original objective function value for that variable. There is one constraint corresponding to each of the complicating (relaxed) constraints and the coefficients for each variable in the constraints reflect the "contribution" of the subproblem to meeting each constraint. Finally, the master problem is constrained to select a convex combination of the subproblem solutions, indirectly enforcing the subproblem constraints.

The subgradient method recognizes that the objective function of the dual problem is a piece-wise linear, convex optimization problem in the multipliers (Nemhauser and Wolsey 1988) and uses a method analogous to gradient search to adjust the multipliers, where the concept of the gradient is replaced by that of a subgradient.

Similar in concept to subgradient methods, multiplier adjustment methods use specific problem structure and knowledge (rather than the subgradient direction) to adjust the multipliers in an ad-hoc manner. Since these methods are specific to a particular problem, some computational efficiencies may be expected (Magee and Glover 1996).

Each method has its distinct advantages and disadvantages. Column generation retains the information from each iteration (which can be an advantage in the adjustment of the multipliers) and it is relatively easy to implement, given the availability of high level algebraic languages for optimization modeling (Magee and Glover 1996). It can be slow to converge on certain problems, however. Subgradient optimization, on the other hand, should be faster since it does not have to solve a linear program on each iteration (Magee and Glover 1996). It's main disadvantage is that it does not retain any information from past iterations and therefore may oscillate for some problems (Magee and Glover 1986). Multiplier adjustment methods require custom implementation since they depend upon the particular problem being solved and do not offer theoretical guarantees of convergence as in column generation and subgradient optimization. However, they have been shown to provide better performance on certain problems (Fisher 1981). For these computational experiments, column generation was chosen primarily for its ease of implementation and its guarantee of convergence.

5.2 Solving the Lagrangian Decomposition Dual Problem for the STRS Model

To solve the dual problem for the Lagrangian decomposition of the STRS model (Equation 4.6) via column generation, the following master linear program (see Magee and Glover 1996 for details) is formed (Problem MP)

$$z_{MP}^{N} = min\sum_{i}\left\{\sum_{n=1}^{N}[Z_{n,i}\cdot\Upsilon_{n,i}]+\sum_{t}[(S_{i,t}^{G+}+S_{i,t}^{G-})\cdot P^{G}+(S_{i,t}^{X+}+S_{i,t}^{X-})\cdot P^{X}]\right\}$$

(Equation 5.1)

Subject to

$$\forall i, \forall t: \sum_{n=1}^{N} [\, g_{i,t,n}^{U} \Upsilon_{n,i}] - \sum_{n=1}^{N} [\, g_{i,t,n}^{T} \lambda_{n,t} \,] + S_{i,t}^{G+} - S_{i,t}^{G-} = 0 \quad \text{(Equation 5.2)}$$

$$\forall i, \forall t: \sum_{n=1}^{N} [\, x_{i,t,n}^{U} \Upsilon_{n,i}] - \sum_{n=1}^{N} [\, x_{i,t,n}^{T} \lambda_{n,t} \,] + S_{i,t}^{X+} - S_{i,t}^{X-} = 0 \quad \text{(Equation 5.3)}$$

$$\forall t: \quad \sum_{n=1}^{N} \lambda_{n,t} = 1 \quad \text{(Equation 5.4)}$$

$$\forall i: \quad \sum_{n=1}^{N} \Upsilon_{n,i} = 1 \quad \text{(Equation 5.5)}$$

where

$$\forall i, \forall t: \quad S_{i,t}^{G+}, S_{i,t}^{G-}, S_{i,t}^{X+}, S_{i,t}^{X-} \geq 0$$

$$\forall n, \forall t: \quad \lambda_{n,t} \geq 0$$

$$\forall n, \forall i: \quad \Upsilon_{n,i} \geq 0$$

In this formulation, z_{MP}^{N} is the master problem objective function optimal value after N iterations. The $\lambda_{n,t}$ are the optimal weights for each time subproblem solution and the

$\Gamma_{n,i}$ are the optimal weights for each unit subproblem solution (for each iteration n). The objective is to find the best convex combination (after N iterations) of the true costs (without penalty terms) of each iteration unit subproblem solution, given by

$$Z_{n,i} = min\sum_t \left\{ \sum_{i \in K} \left(b_i \cdot y^U_{i,t,n} \right) + \sum_i \left(a_i \cdot x^U_{i,t,n} + c_i \cdot g^U_{i,t,n} \right) \right\}$$

Artificial variables, $S^{G+}_{i,t}$, $S^{G-}_{i,t}$, $S^{X+}_{i,t}$, $S^{X-}_{i,t}$, have been introduced to ensure feasibility in the equality constraints in the early iterations. These variables are penalized in the objective function, where the penalties, P^G, P^X, are chosen high enough to drive the artificial variables to zero, while not too high to avoid distorting the final solutions (Magee and Glover 1996). The need for artificial variables can be avoided by introducing an initial, feasible solution as a starting point.

The methodology proceeds iteratively as shown in Figure 5.1. Initial values are assigned to the Lagrange multipliers (u and v) for the equality constraints (Equation 5.2 and Equation 5.3) and are passed to the subproblems specified in the previous chapter (Problems LDU and LDT). Solving each set of subproblems generates new extreme points which are new columns in the master problem. These new columns can then used to find the best convex combination of all solutions (columns) currently known. The procedure continues until some stopping criterion is met.

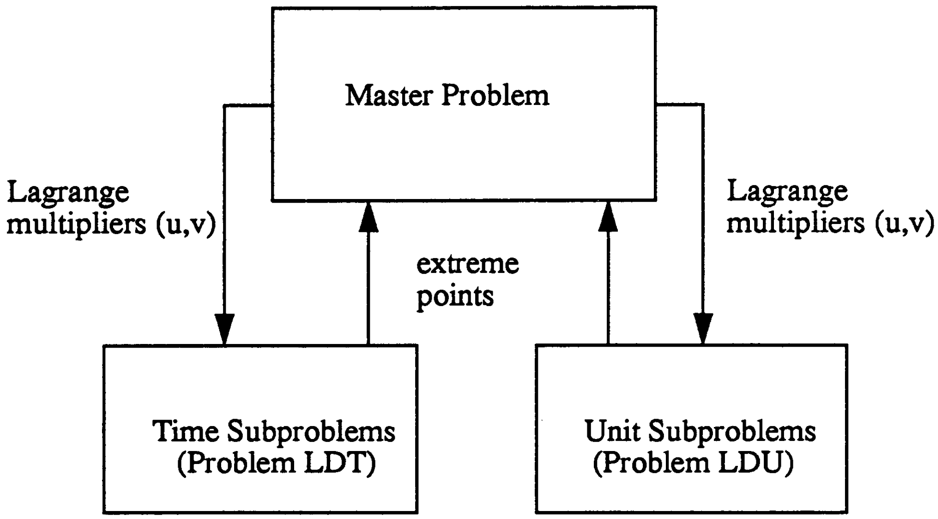

Figure 5.1. Iterative Column Generation Solution Methodology to Solve the Dual Problem for Lagrangian Decomposition of the STRS Model.

Possible stopping criteria include reaching a user-specified number of iterations, or that the absolute, relative difference between the master program upper bound, z_{MP}^{N}, and the master problem lower bound is less than some user-specified tolerance. The master problem lower bound is given by

$$z_{MPLB}^{N-1} = z_{LDU}(u^{N-1}, v^{N-1}) + z_{LDT}(u^{N-1}, v^{N-1})$$

where $z_{LDU}(u^{N-1}, v^{N-1})$, $z_{LDT}(u^{N-1}, v^{N-1})$ are the objective values from each subproblem using the dual prices u^{N-1},v^{N-1} from iteration $N-1$. The stopping criterion is then given by

$$abs\left\{\left(z_{MP}^{N}-z_{MPLB}^{N-1}\right)/z_{MP}^{N}\right\}\leq tolerance\,.$$

5.3 Feasible Solutions

A drawback to any Lagrangian relaxation approach is that the optimal solution to the Lagrangian dual problem is not guaranteed to be feasible for the original problem. However, column generation does produce a primal solution at each iteration and a simple rounding heuristic can be applied to achieve an integer feasible solution to the original problem. The heuristic is as follows: given the master problem solution, the unit commitment is derived by setting $x_{i,t} = 1$ for all units where $\sum_n g_{i,t} \cdot \Upsilon_{n,i} \geq tolerance$. The tolerance is determined empirically and may be zero. Using this commitment, the economic dispatch problem is then solved (via linear programming) to determine the $g_{i,t}$.

To generate an initial feasible solution for the column generation methodology, a priority order scheme is used. The methodology is as follows:

1. Rank all available thermal units with respect to their incremental cost and capacity.
2. Compute the total energy available from all hydros over the time horizon.
3. Commit a thermal unit in priority order.

4. Compute the energy available from all committed thermal units over the time horizon and the remaining energy needed.
5. Can hydro supply the remaining energy? If yes, go to 6. If no, go to 3.
6. Input the thermal unit commitment and solve for the remaining commitments and generation levels. Is the solution feasible? If yes, stop. If no, go to 3.

This methodology is not guaranteed to achieve a feasible solution but in practice has proven to be effective in many cases.

5.4 Computational Results

The column generation methodology was implemented using the algebraic modeling language, GAMS (Brooke et al. 1991) with the CPLEX solver (CPLEX 1994). Details of this implementation are given in Appendix A.

To evaluate the Lagrangian decomposition methodology, several scenarios were developed using electrical system data published by the Electrical Power Research Institute (EPRI 1977). The details of two of these scenarios are also given in Appendix A. Bounds from the linear programming relaxation (LP), the Lagrangian relaxation of constraints that couple units (LRU), the Lagrangian relaxation of constraints that couple time (LRT), the Lagrangian decomposition (LD), and the true solution (MIP) have been generated for each scenario.

5.4.1 Scenario 1

This scenario was developed so that the demand and spinning reserve constraints are tight at optimality. In Table 5.1, the various bounds are compared. As expected, the new

Method	Bound	Relative Gap
MIP	71045	----
LD	71045	0.0%
LRT	71045	0.0%
LRU	69554	2.10%
LP	68824	3.12%

Table 5.1. Computed Bounds for Scenario 1.

Lagrangian relaxation (LRT) gives a tighter bound than the traditional relaxation (LRU), since LRU relaxes these constraints. Both the decomposition (LD) and the new relaxation find the optimal solution. This phenomenon can be illustrated by visualizing that the points corresponding to MIP, LD, and LRT all coincide in Figure 4.2.

5.4.2 Scenario 2

This scenario was developed so that the ramping constraints are very tight at optimality. The bounds computed are given in Table 5.2. As expected, the traditional relaxation (LRU) gets a slightly better bound that the new relaxation (LRT). However, the Lagrangian decomposition (LD) theoretically must dominate both bounds and this is verified.

Method	Bound	Relative Gap
MIP	94203	----
LD	93995	0.22%
LRU	93974	0.24%
LRT	93967	0.25%
LP	91535	2.83%

Table 5.2. Computed Bounds for Scenario 2.

5.5 Performance of the Methodology

The progression of the master problem for Scenario 1 is shown in Figure 5.2. Displayed are the linear programming objective value (LP), the optimal value (MIP), the traditional Lagrangian relaxation value (LRU), and both the upper and lower bounds produced from the column generation algorithm for each iteration. For this scenario, the stopping criterion was set to 100 iterations.

The progression is characterized by several stages. The initial feasibility stage (where the artificial variables are being driven from the solution) is completed in 15 iterations. The algorithm reaches the correct commitment decision (as determined by the rounding heuristic) by iteration 30. Although difficult to see in the figure, by iteration 59 the master problem objective value has reached the optimal value. The solution identical to the MIP solution is found at iteration 87.

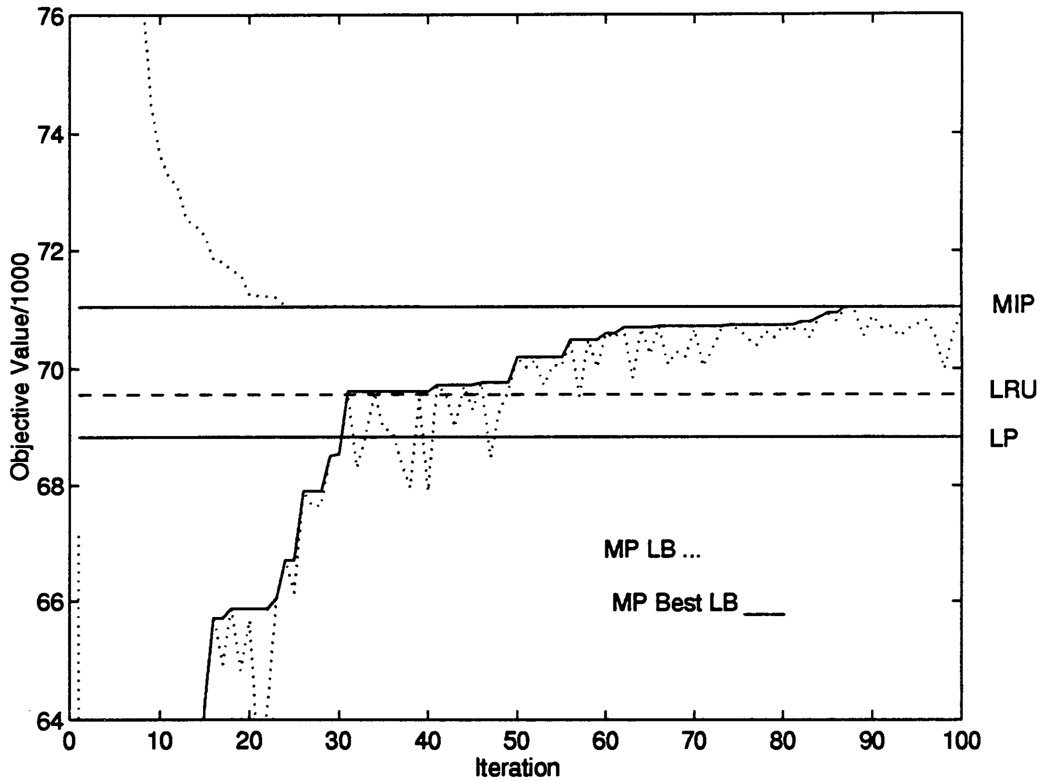

Figure 5.2. Progression of the Master Problem for Scenario 1.

5.5.1 Initial Feasible Solution

The heuristic to find an initial feasible solution was used and provided to the algorithm to eliminate the need of the artificial variables. The performance is shown in Figure 5.3. The progression of the best lower bound is only slightly improved when using the initial feasible solution.

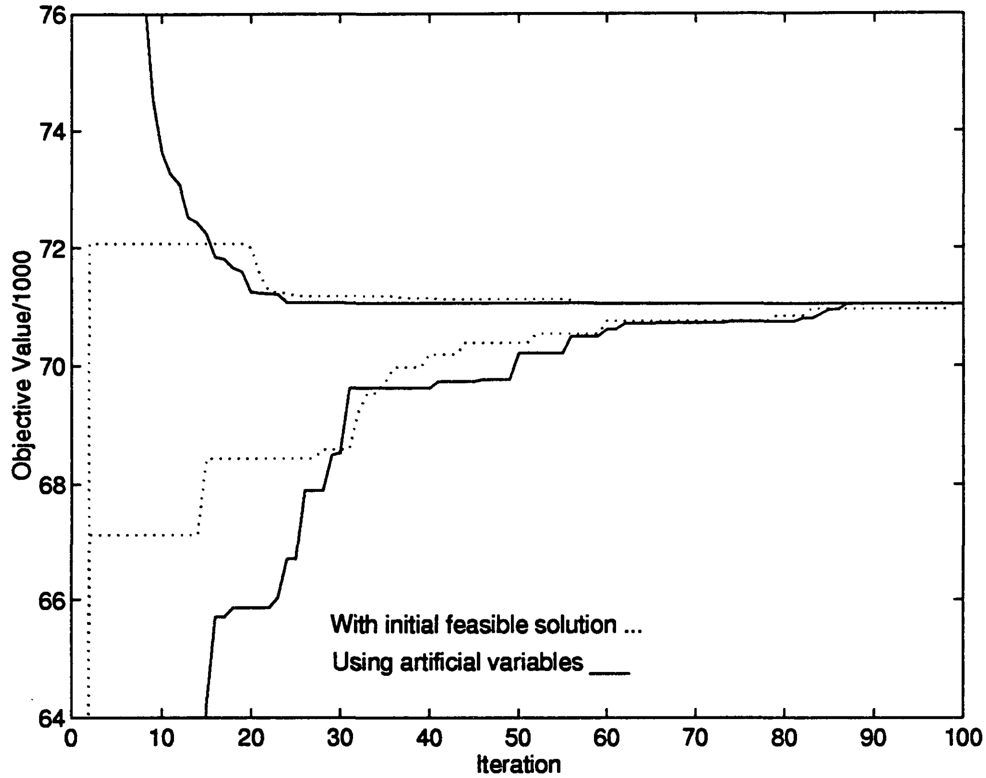

Figure 5.3. Progression of the Master Problem with an Initial Feasible Solution for Scenario 1.

5.5.2 A Restriction Strategy

Previous work in using Lagrangian decomposition has shown promise in using restriction strategies to improve performance (Glover et al. 1984; Glover and Klingman 1988). These strategies incorporate additional constraints in each subproblem layer to compensate for the relaxed equality constraints by effectively "restricting" the change of the subproblem variables between iterations.

To test this idea, a restriction constraint was added to the unit subproblems of the following form:

$$\forall i: \quad \sum_t g_{i,t,n}^U \geq \left(m_i \cdot \sum_t x_{i,t,n}^T\right) \cdot \alpha^n$$

where α^n is a user-specified weighting factor. By letting this factor decay exponentially, the effect of the restriction is greatest in the early iterations. Progression of the best lower bound is shown in Figure 5.4 for three values of α, compared to the progression without

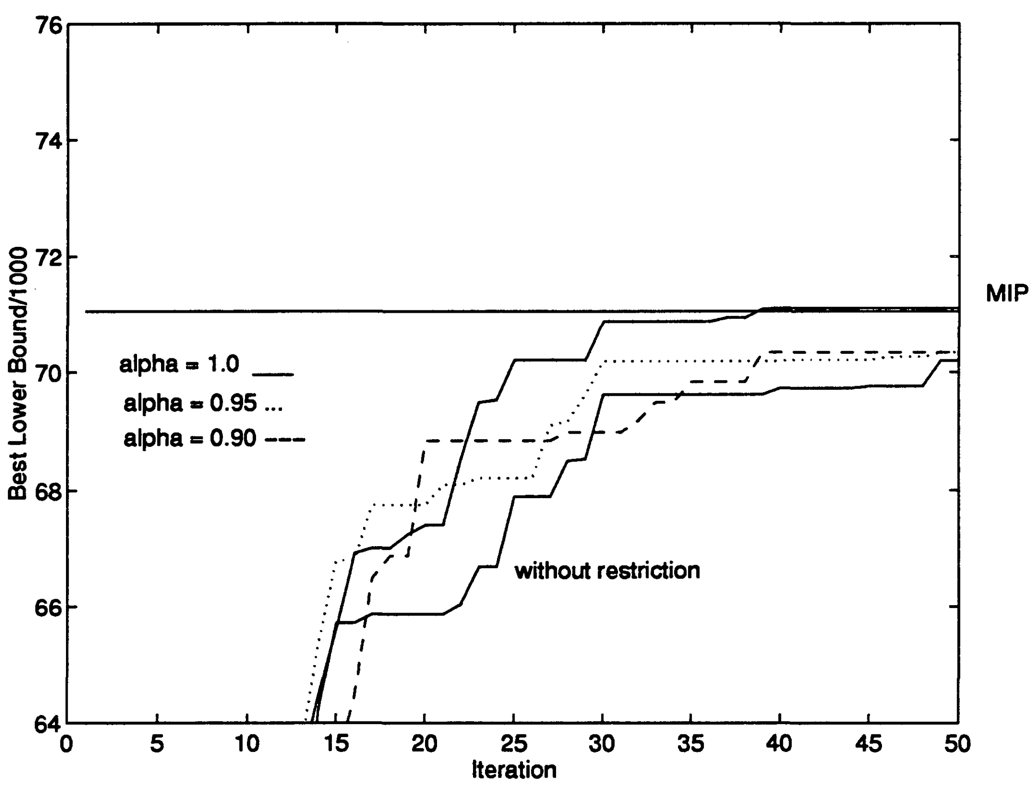

Figure 5.4. Progression of the Master Problem using Restriction on the Unit Commitment Subproblems.

restriction. Some improvement is seen and further investigation of such strategies is warranted.

5.6 Summary

Even for many classes of linear programming problems, it has been observed that the tail-end convergence of column generation techniques may be slow (Bazaraa et al. 1990). The nature of the STRS problem further adds to this difficulty. The STRS problem is characterized by a very flat valley due to the relative similarities in the generation resources. This may also explain the relative small duality gaps that have been observed, both in this study and by other researchers (Lauer et al. 1982). Furthermore, the problem has alternate optima due to the ability to distribute the hydro energy in many ways, once the total contribution of the thermal energy for each resource is known.

To graphically illustrate the presence of a very flat valley, examples of various unit commitments and the resulting objective function values for Scenario 1 are shown in Figure 5.5. For each sub-plot, the horizontal axis is the time step (1-8) and the vertical axis is the thermal unit (1-4). Units that are committed are shown by a darkened cell. Particularly note that substantially different commitments can produce optimal solutions that are very close (in one case, within 0.001% of each other).

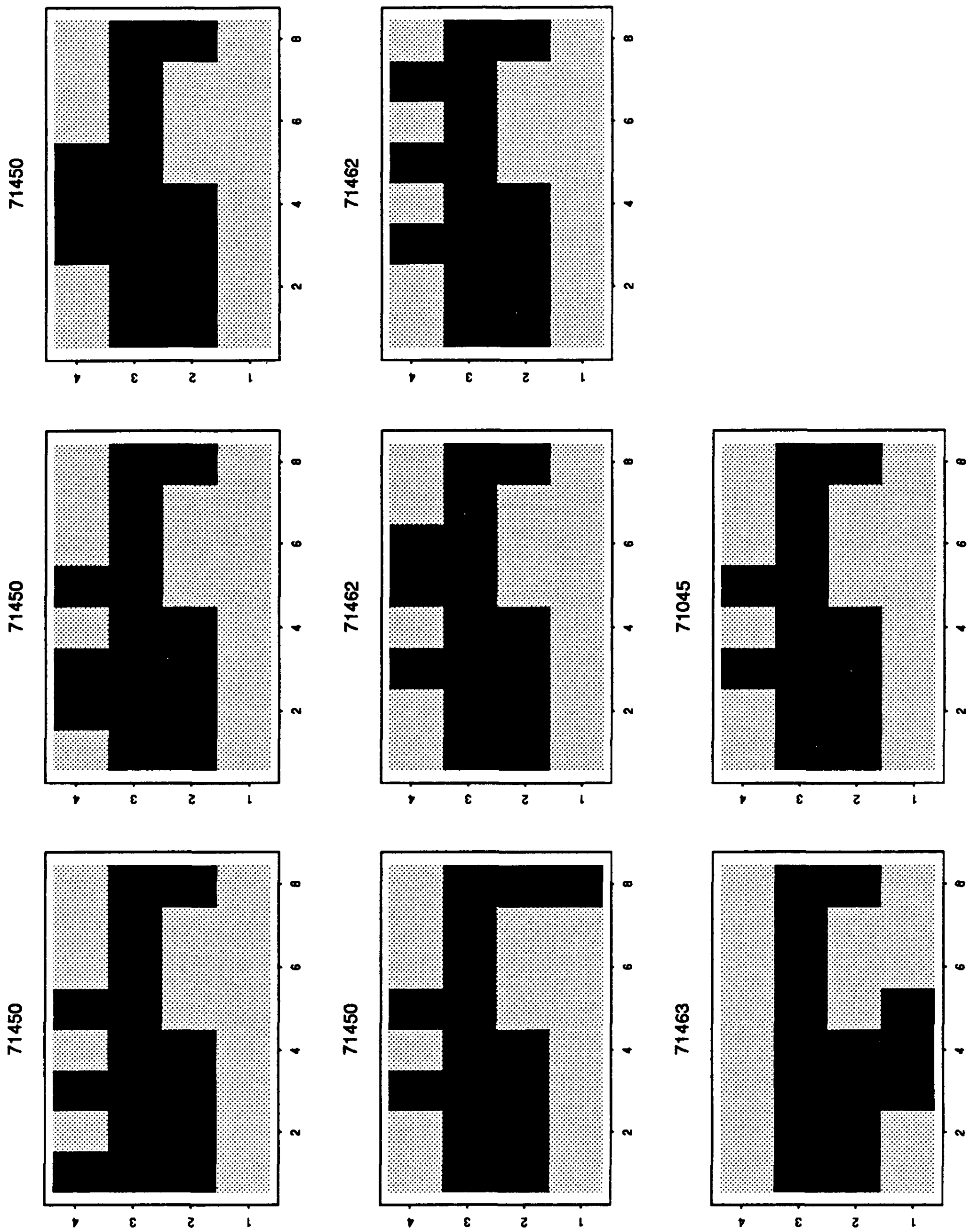

Figure 5.5. Examples of Unit Commitment Solutions and the Resulting Objective Values for Scenario 1.

Chapter 6

ENHANCED MODELING OF THE HYDRO COMPONENT

Modeling the flow of water in river channels is critical for the efficient operation of reservoir and hydroelectric systems. Although river flow can be described by a system of partial differential equations, solution of that system of equations is difficult and requires detailed physical data which vary in time. Systems identification is a statistical modeling approach that provides models which can often depict the essential behavior of a dynamic system. Given real-time, telemetered data, this approach also supports an adaptive methodology where the model parameters are updated automatically. Such updates capture time-varying system responses and often can be used to model non-linear system responses.

In this chapter, an adaptive, systems identification methodology for modeling flow in river channels is presented. First, techniques for modeling river flow are reviewed. Next, an overview of the systems identification approach is presented and a methodology to identify a suitable system model is discussed. An adaptive mechanism for time-varying systems is then developed. These techniques are applied to a river system using data observed on a stretch of the Lower Colorado River, in the southwestern U.S. An analysis of the methodology and its appropriateness for operational decision making is also presented.

6.1 Review of Modeling Flow in River Systems

The efficient operation of reservoir and hydroelectric systems requires that multiple objectives and constraints be considered. These include the control of floods, providing water for consumptive uses (municipal, industrial, and agricultural), meeting requirements for biological, environmental, and recreational interests, as well as meeting the electrical demand. The ability to predict or model the propagation of water throughout the watershed (often termed routing) is critical to meeting these objectives and constraints. Of particular interest is the modeling of the propagation of water downstream of a controlled facility.

Modeling the propagation of water in an open channel is a complex problem. Open channel flow is a distributed process since the flow rate, velocity, and depth of the water vary in space and time. Physical laws (conservation of mass and momentum) lead to a system of nonlinear, hyperbolic, partial differential equations, known as the Saint-Venant equations, having no closed-form or analytical solution (Chow 1959). This system of equations can be solved numerically, usually to capture the spatial variability in the direction of the channel only (one-dimensional). Individual terms in the system of equations can be neglected leading to the common classification of distributed models that includes kinematic wave, diffusion wave, and the full dynamic wave models (Chow et al. 1988).

All distributed flow models require knowledge of the detailed geometrical and hydraulic characteristics of the channel. This detailed information is expensive to collect and varies in time due to erosion and deposition in the channel, as well as man-made effects such as dredging. Furthermore, in an operational environment, decisions must be made in a short period of time as the state of the system and forecasts of stochastic variables evolve. Often the water model must be embedded in another decision support system (such as an optimization program), requiring a simpler, more efficient model. To date, these requirements have limited the use of distributed models in the operational environment.

Due to these difficulties, lumped-parameter models have been utilized in the desire to obtain sufficient accuracy at reasonable effort and expense (Dooge et al. 1982). Lumped-parameter models result from a simplification of the system of partial differential equations to a system of ordinary differential equations with only a time derivative. For the problem of water flow in on open channel, one can think of the "lumping" as an averaging of system properties in the spatial dimension. The major drawback is the loss of spatial information, particularly information about the depth of the water, critical for the analysis of floods and prediction of water quality. It is possible, however, (by assuming a particular type of flow regime at the site of interest) to predict water depths for flood analysis once the flow rate has been predicted (Chow et al. 1988). Common lumped parameter routing methodologies include the Muskingum method (McCarthy 1938) and the river routing component of the

Streamflow Synthesis and Reservoir Regulation (SSARR) model (U.S. Army Corps of Engineers 1987).

Lumped-parameter models can be generalized using systems theory. A system can be thought of simply as a mapping of input functions to produce output functions. Given observation of the outputs and inputs, the characteristics of a system may be deduced. The systems identification problem then is to determine a suitable representation (or model) of the system consistent with the observed data (Ljung 1987). Such representations can be, but are not required to be, based on any physical insight or knowledge about the actual mechanisms of the natural system; i.e., the system may be treated as a "black-box".

This simple idea has found usefulness in a variety of fields, including electrical engineering, economics, geophysics, and water resources. The unit hydrograph method of predicting the runoff response to a rainfall event is an example (Chow et al. 1988), as is the discrete kernel method of computing the drawdown in a well field due to pumping (Illangasekare 1985). Recently, linear functional models were applied to predict daily flows on the Aiakmon River in Greece (Papamichail and Papazafiriour 1992).

Nonlinear effects can often be handled by applying a priori knowledge of the nonlinear character or by assuming that the system behaves linearly about a certain trajectory (Ljung 1987). The latter concept has been applied to a canal modeling and control problem by a priori computing the parameters of a linear model for each trajectory of interest (Zagona 1992).

For many operational applications, a lumped-parameter model is appropriate and the systems identification methodology provides the best approach to determining a model that is suitable to support operational decisions. This contention is supported in that:

1. The resulting models are easy to understand and use while still capturing the essential behavior of the river system
2. Estimation of the model parameters is computationally fast
3. The necessary input-output data can be collected automatically and telemetered to the control center, eliminating the need for costly and time-consuming manual data collection
4. The approach easily supports an adaptive methodology, and when coupled with real-time data collection, allows for the model to automatically adjust its parameters to account for time-varying and non-linear effects

6.2 System Identification of Linear, Time-invariant Systems

An input-output system is depicted in Figure 6.1.

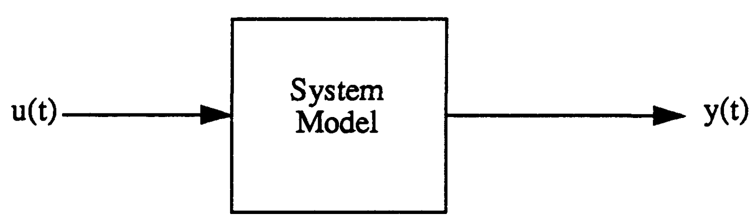

Figure 6.1. Depiction of an Input-Output Dynamic System Model.

Assume that observations are made in discrete time at equal sampling intervals Δt so that $t_k = k \cdot \Delta t$, k=1,2,3,... Without loss of generality, assume that $\Delta t = 1$ and drop the subscript k, so that in Figure 6.1, u(t) is the input to the system and y(t) is the output from the system at time step t. Then assuming a linear, time-invariant, causal system, the system output may be written as

$$y(t) = \sum_{k=1}^{\infty} g(k)u(t-k), t = 0, 1, 2, \ldots \qquad \text{(Equation 6.1)}$$

where g(k) is the impulse response function of the system. By introducing the backward shift operator, z^{-k}, where

$$z^{-k}u(t) = u(t-k) \qquad \text{(Equation 6.2)}$$

Equation 6.1 can be written as

$$y(t) = \sum_{k=1}^{\infty} g(k)z^{-k}u(t), t = 0, 1, 2, \ldots \qquad \text{(Equation 6.3)}$$

By defining the z-transform of g(k) as

$$G(z) = \sum_{k=1}^{\infty} g(k)z^{-k} \qquad \text{(Equation 6.4)}$$

then Equation 6.1 becomes

$$y(t) = G(z)u(t), t = 0, 1, 2, \ldots \quad \text{(Equation 6.5)}$$

G(z) is called the system's transfer function. These developments are given in many signal processing books (Oppenheim and Schafer 1975).

The systems identification problem can now be stated as follows: given observed input and output sequences (u(t) and y(t)), determine the system response (either the impulse response or transfer function) of the dynamic system. Without noise or other disturbances and assuming that the system is stable, the determination of the system response is mathematically straightforward.

If the assumption is made that disturbances, v(t), are additive, then the system response equation can be modified to

$$y(t) = G(z)u(t) + v(t), t = 0, 1, 2, \ldots \quad \text{(Equation 6.6)}$$

The disturbance v(t) includes all sources of noise, both in the measurement of u(t) and y(t), as well as in the process itself. An example of process noise would be unmeasured inputs, which for the problem of modeling river flow, include unmeasured side inflows, diversions, and return flows.

Since v(t) is very seldom known a priori, the system identification problem is made much more difficult. A rich theory exists to deal with disturbances based on the premise that information about past disturbances is valuable to predicting future ones (Ljung 1987). A common approach is to model the disturbances as a system and write

$$v(t) = H(z)e(t), t = 0, 1, 2, \ldots \qquad \text{(Equation 6.7)}$$

where H(z) is the transfer function of the noise and e(t) is a sequence of independent, identically distributed random variables with a certain probability function. Although this model of the noise cannot represent all possible disturbances, it has been shown to be sufficient for most practical applications (Ljung 1987).

The systems identification problem now becomes one of estimating G(z) and H(z).

6.3 Estimation of G(z) and H(z)

Approaches to estimating the transfer functions fall into two main categories: parametric and non-parametric methods. Non-parametric methods determine these functions by direct analysis without assuming a confined set of possible models; i.e., they do not explicitly search for values of a finite-dimensional parameter vector (Ljung 1987). Transient-response analysis, correlation analysis, frequency-response analysis, and spectral analysis fall into this category.

Although parametric methods require us to choose a model structure (a parameter vector to be estimated), in practice this is not a major limitation as a wide variety of structures can be represented. Most parametric models can be classified as either black-box or state-space.

Black-box models include the familiar finite impulse response and autoregressive models. With state-space models, it is easier to incorporate insights into the physical mechanisms of the dynamic system as the relationship between input, output, and noise signals is written as a system of first-order difference equations using an auxiliary state vector (Ljung 1987). The Kalman "filter" is a good example of a model that uses this form (Gelb et al. 1974).

6.4 The Methodology

Given the observed flows into and out of the reach (u(t) and y(t) respectively), the systems identification methodology is shown in Figure 6.2 (adapted from Ljung, 1987).

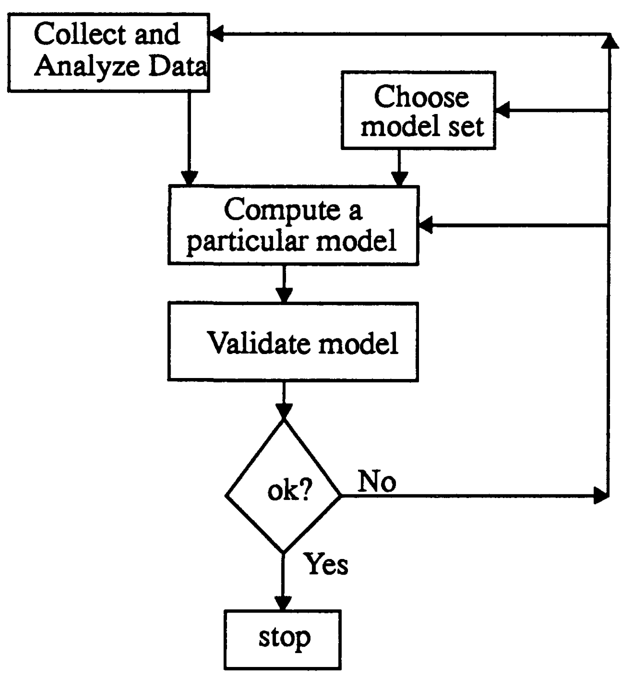

Figure 6.2. The Systems Identification Methodology

6.5 Data Collection and Analysis:

Most often, the system identification procedure is restricted to using data observed from the normal operation of the river system; however, it may be possible and useful to design a particular experiment where the input signals are chosen to better understand the system response. For those circumstances, references exist that will aid in the experimental design (see Ljung 1987).

6.6 Choice of Model Set

By representing G and H as rational functions (polynomials in z), a general parametric representation of Equation 6.6 is given by (Ljung 1987)

$$A(z)y(t) = \frac{B(z)}{F(z)}u(t) + \frac{C(z)}{D(z)}e(t) \qquad \text{(Equation 6.8)}$$

Equation 6.8 represents a family of parameterizations or model structures that covers a large range of possible models (some 32 different model sets), including finite impulse response, autoregressive, autoregressive moving average, output error, and Box and Jenkins models with or without exogenous input (Ljung 1987).

Choosing which model set(s) to investigate from this large range can be a formidable problem. The best advice is to first choose models as simple as possible. It is recommended to start with the equation error model structure (where C(z) = D(z) = F(z) = 1 in Equation 6.8), which, in summation form, is given by

$$y(t) + \sum_{i=1}^{na} a_i \cdot y(t-i) = \sum_{j=1}^{nb} b_j \cdot u(t-nk-j+1) + e(t) \quad \text{(Equation 6.9)}$$

where *na* and *nb* are the number of coefficients (parameters) in the polynomials and *nk* is the nominal delay of the system.

It is also noted that Equation 6.9 easily handles multiple input-output models, in which case *u* and *y* will be matrices with each column representing a single input or output respectively, and there will be corresponding polynomial coefficients (*a* and *b*) for each input and output. Such models may be important for areas with multiple tributaries and large diversions and return flows.

6.7 Computation of a Particular Model

The prediction-error method based on a least-squares criterion is used. To find *d* model parameters with *N* observations, the prediction is given by

$$\hat{y}(t|\theta) = \Phi^T(t)\theta \quad \text{(Equation 6.10)}$$

where θ is the vector of model parameters of dimension dx1 and Φ(t) is the vector containing the observed data for time t, also of dimension dx1. The prediction error is then given by

$$\varepsilon(t|\theta) = y(t) - \hat{y}(t|\theta) \quad \text{(Equation 6.11)}$$

By forming the dxd variance matrix R,

$$R(N) = \frac{1}{N} \cdot \sum_{t=1}^{N} \Phi(t) \cdot \Phi^T(t)$$

and the dx1 data vector

$$f(N) = \frac{1}{N} \cdot \sum_{t=1}^{N} \Phi(t) \cdot y(t)$$

then the least-squares estimate of θ can be written as

$$\hat{\theta} = \min_{\theta} \sum_{t=1}^{N} \varepsilon(t, \theta)^2 = R^{-1}(N) \cdot f(N) \qquad \text{(Equation 6.12)}$$

It should be noted that these results strictly apply only to the equation error model structures described by Equation 6.9, However, for the other model structures, a similar development results in "pseudolinear" regressions that can be solved with iterative search methods (Ljung 1987; Ljung 1991).

6.8 Model Validation

Once a set of models has been computed, the validation step involves answering two questions: 1) how well do the model predictions agree with the observed data, and 2) does the model meet the intended need (i.e., are the predictions "good enough"). The first question can be answered in some quantifiable sense; however, the second question is somewhat philosophical and is generally left to the practitioner to answer.

If possible, it is best to apply the models to a different set of observations (say of length M) from those used in the computation (i.e., a cross-validation). Under those circumstances, a simple comparison of the standard error of the model

$$\text{model std error} = \left\{\frac{1}{M}\sum_{t=1}^{M}\varepsilon(t,\theta)^2\right\}^{\frac{1}{2}} \qquad \text{(Equation 6.13)}$$

is usually sufficient. If additional data are not available, a different comparison criterion other than Equation 6.13 must be used to account for the fact that models with more parameters will naturally perform better when applied to the same data. Measures that have been developed to account for this include the Akaike's Information Theoretic Criterion (AIC) (Ljung 1987).

Additional validation techniques can be applied to specific model structures. For example, a check of the correlation between the errors and the signal input for equation error models (Equation 6.9) is helpful in determining if it is necessary to use more complicated structures.

6.9 Time-varying Linear Systems

If the system's response is time varying (due to physical changes or perhaps non-linearities), one may wish to update the estimate of the parameter vector as new data arrive. Furthermore, it may be prudent to weight the more recent measurements in lieu of the older measurements. The weighted least squares version of Equation 6.12 is given by

$$\hat{\theta}_t = \min_\theta \sum_{k=1}^{t} \beta(t, k) \cdot \varepsilon(k, \theta)^2 \qquad \text{(Equation 6.14)}$$

where the subscript t denotes the estimate based up to sample t. A particular form of the weighting sequence may be written as

$$\beta(t, k) = \lambda(t) \cdot \beta(t-1, k) \qquad \text{for } 1 \le k \le t-1 \qquad \text{(Equation 6.15)}$$

where

$$\beta(t, t) = 1 \qquad \text{(Equation 6.16)}$$

and

$$\lambda(t) = \lambda \quad \text{where } \lambda < 1 \qquad \text{(Equation 6.17)}$$

Then the weighting sequence is given by

$$\beta(t,k) = \lambda^{t-k} \qquad \text{(Equation 6.18)}$$

and the old measurements are damped exponentially. In Equation 6.18, λ is a user-specified parameter called the forgetting factor which is discussed in more detail in the applications section. It can then be shown that a recursive solution of Equation 6.14 is given by

$$\hat{\theta}_t = \hat{\theta}_{t-1} + R^{-1}(t) \cdot \Phi(t) \cdot \{y(t) - \Phi^T(t) \cdot \hat{\theta}_{t-1}\} \qquad \text{(Equation 6.19)}$$

where the update of R is given by

$$R(t) = \lambda(t) \cdot R(t-1) + \Phi^T(t) \cdot \Phi(t) \qquad \text{(Equation 6.20)}$$

Similar to the comment concerning pseudolinear regression, these results strictly apply only to the equation error model structures described by Equation 6.9. However, for the other model structures, similar recursive schemes may be developed (Ljung 1987).

6.10 Application to River Systems: Lower Colorado River

The section of the Colorado River of interest for this study covers approximately 120 river miles and is shown in Figure 6.3. At the top of the section is a storage reservoir and dam (Lake Havasu and Parker Dam) and at the bottom of the reach is a diversion structure (Imperial Dam).

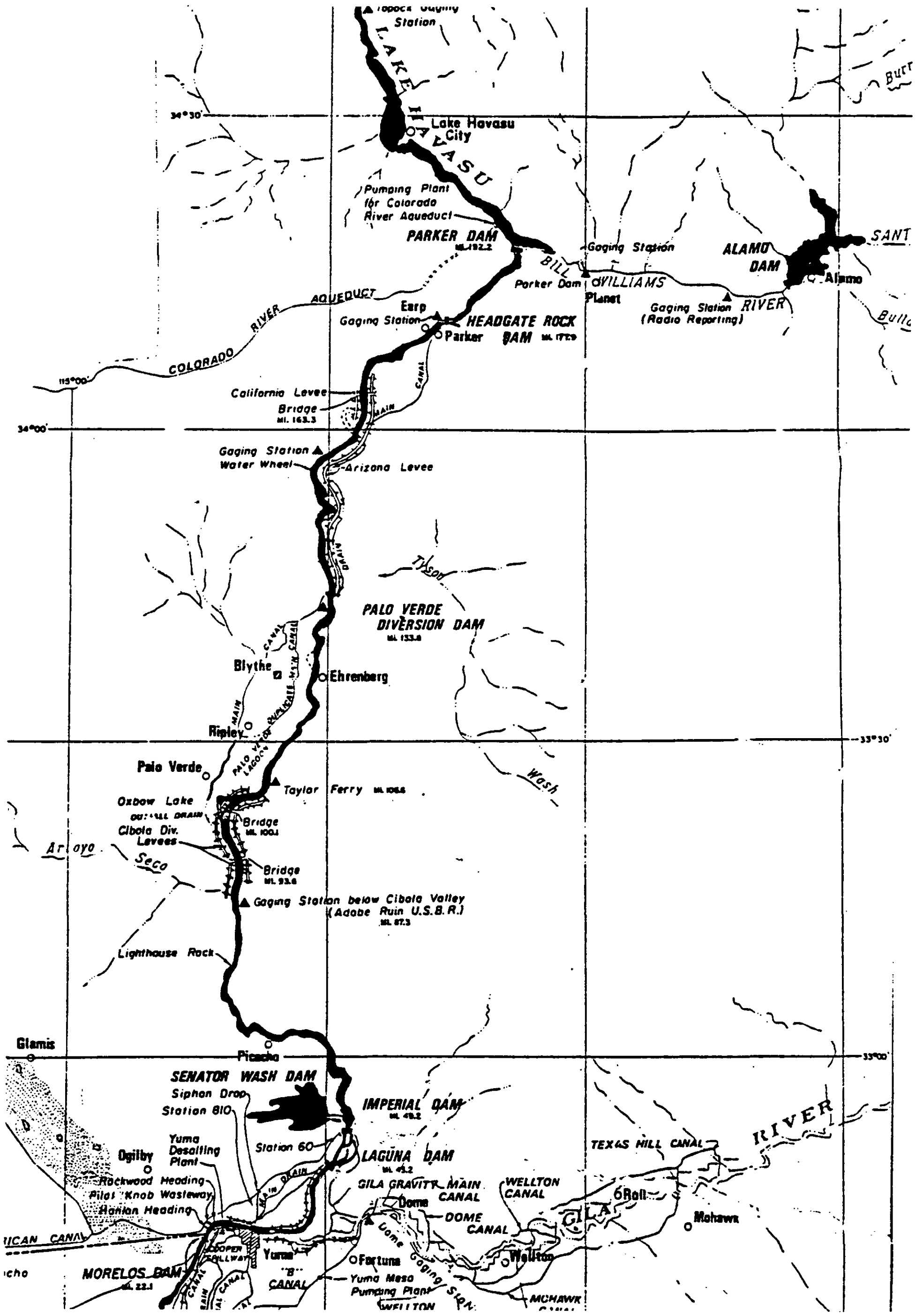

Figure 6.3. Map of the Stretch of the Lower Colorado River Used in this Study

Lake Havasu represents the last major storage facility on the river (capable of storing approximately 640,000 acre-feet of water) and releases are made primarily to meet water demands within the section, at Imperial Dam, and below. Secondary objectives for water releases from Parker Dam include meeting target storage levels in the lake for recreation and flood control and the generation of power.

There are two main diversions, the Colorado River Indian Reservation (CRIR) and the Palo Verde Irrigation District (PVID), and three primary river gaging stations (Waterwheel, Taylor Ferry, and Cibola) within the reach, locations of which are also shown in Figure 6.3. Imperial Dam forms the diversion pool to deliver water to two major canals serving consumptive uses in California and Arizona (the All-American and Gila Gravity Main Canals). There is also a pumped storage facility (Senator Wash Dam) just upstream of Imperial that is used to store excess water arriving at Imperial or to provide additional water as needed. The approximate distance downstream, travel time, range of operation, and data recording interval for each of these facilities is given in Table 6.1.

Facility	Approx. Distance, miles	Approx. Travel Time, hours	Capacity	Data Interval
Lake Havasu/ Parker Dam	0	0	Storage: 640000 acre-feet Release: 20000 cfs	Hourly
Headgate Rock Diversion Dam	10	3	Diversion: 2000 cfs	Daily
WaterWheel Gage	30	10	N/A	Hourly
Palo Verde Diversion Dam	45	20	Diversion: 1800 cfs	Daily
Taylor Ferry Gage	65	26	N/A	Hourly
Cibola Gage	80	35	N/A	Hourly
Imperial Diversion Dam	120	55	Diversion: 17400 cfs	Hourly
Senator Wash Reservoir and Dam	120	55	Storage: 13850 acre-feet Pump/Rel: 960/1200 cfs	Hourly
All-American Canal	120	55	Diversion: 15150 cfs	Hourly
Gila Gravity Main Canal	120	55	Diversion: 2200 cfs	Hourly

Table 6.1. Pertinent Data for Lower Colorado Example

6.11 Inputs and Outputs of the System

The primary goal of the water model is to predict what releases should be made from Parker Dam to meet the forecasted downstream demands. The problem is complicated by the relatively long travel time (approximately 55 hours) from the major storage facility (Lake Havasu) to the major diversions at Imperial Dam. Typically, the Parker release is specified as a mean daily release and the facility is allowed to peak within the day to help meet power demands. In Figure 6.4 the release from Parker Dam and the subsequent flows at the three river gages downstream on an hourly basis over a period of one week in April, 1991 are shown. One can easily see the oscillatory nature of the Parker Dam releases within a day due to the demand for power.

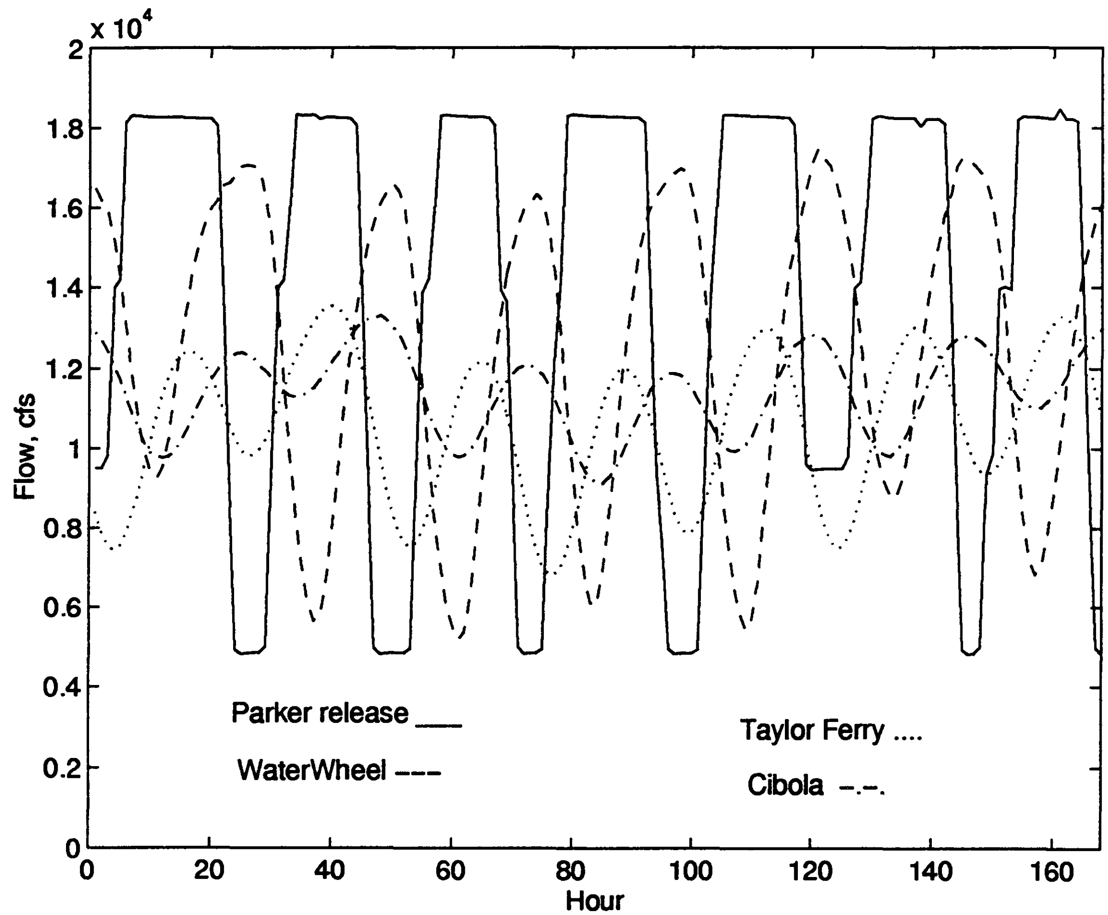

Figure 6.4. Propagation of Water from Parker Dam Downstream

The first investigation involved breaking the river section into reaches corresponding to the telemetered river gaging stations, identifying hourly time step models within each of those reaches, and predicting the arrival at Imperial by cascading the models. Although this approach was successful in identifying models that could predict the hourly arrivals at each section, several difficulties were encountered when using the models to predict the arrival at Imperial in the operational environment. These included:

1. The river gaging stations were fairly unreliable, with substantial periods of bad or missing data,
2. The diversion and return flow records for the diversions within the section were not available on an hourly basis, and
3. There is no direct measurement of the flow arriving at Imperial Dam.

As shown in Table 6.1, the last gaging station (Cibola) is some 40 miles upstream of Imperial Dam. Placing additional stations closer to Imperial is complicated by a lack of reasonably good sites and the backwater effect caused by the dam (Stevens 1992). Currently, the flow arriving at Imperial (I_t) is computed via a mass balance given by:

$$I_t = S_t - S_{t-1} + SW_t + O_t \qquad \text{(Equation 6.21)}$$

where S_t is the volume (or storage) at the end of time step t, SW_t is the net flow out of Imperial due to Senator Wash operations (pumping and releases) over the time step, and O_t is the net outflow from the dam gates and to each diversion canal over the time step. The effects of evaporation, seepage, and other factors have been ignored. As in any water storage facility, the direct measurement of volume is not possible. What is measured is the elevation of the water and therefore, the change in storage may be written as some function of the change in elevation,

$$S_t - S_{t-1} = f(E_t - E_{t-1}) \quad \text{(Equation 6.22)}$$

For Imperial, this functional relationship is difficult to determine due to large sediment loads, vegetation growth, and meandering channel location. It is currently assumed to be linear. Although the data needed in Equation 6.21 and Equation 6.22 are available on an hourly basis, inflow computed at that time step is quite noisy (see Figure 6.5). This is due to the differentiation implied by Equation 6.22 which enhances the noise inherent in the elevation measurement (due to wind, waves, etc.). For these reasons and the fact that the objective is to target Parker's release on a mean daily basis, a daily time step was chosen for this study.

6.12 Diversions and Return Flows

Diversions and return flows within the section to be modeled can be handled in one of three ways:

1. If measurements are not available, the model must assume that these signals are either a part of the system response or the noise. Since the diversions and returns are usually seasonal, a time-varying, adaptive model may be able to adjust the system response parameters accordingly.

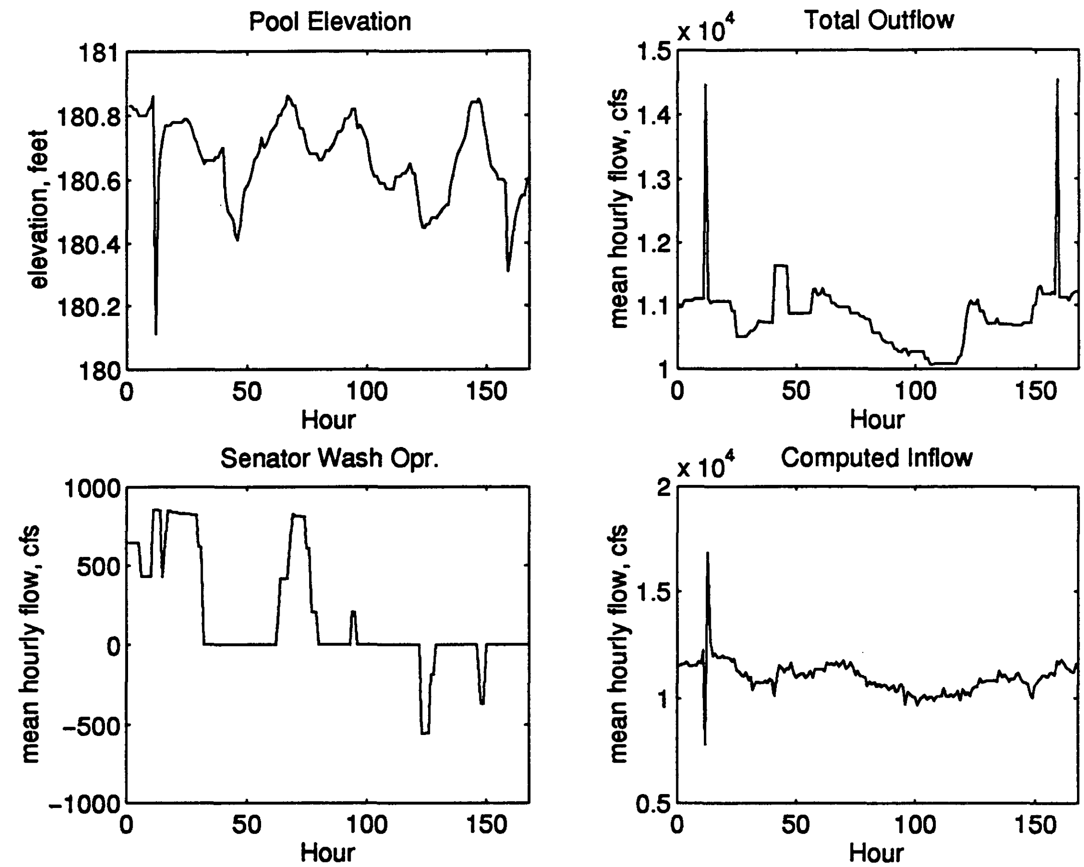

Figure 6.5. Computed Inflow Into Imperial on an Hourly Time Step

2. If measurements are available, they can be modeled as separate inputs to the system since the methodology supports multiple input signals.

3. If measurements are available, the magnitude of the signals is small relative to the input and output signals, and under the assumption of simple dynamics (i.e., known time lags between the input or output and the diversions and returns), the diversions

and returns can be subtracted from the input and output signals prior to model computation.

For this example, the third technique was chosen, although all three were tested. It was found that the time-varying model could discern the seasonal nature of the diversions without having prior knowledge of the signals. However, since records for the two major diversions within the section (CRIR and PVID) are available on a daily basis, one of the explicit methods is better. Modeling the diversions as separate inputs, however, did not provide substantially better results than just subtracting the net (diversion - returns) signals from the input and output records, assuming 0 and 1 day lags from the input for CRIR and PVID respectively. Figure 6.6 shows the net input and output records used in the subsequent analysis.

6.13 High and Low Flow Seasons

Since downstream water demand is the primary objective for the daily Parker Dam releases, the seasonal effect of the water demand can been seen. This effect corresponds to summer and winter irrigation cycles ("high and low flow seasons") for this area. It was anticipated that a particular model might be valid only for each season due to the potential non-linear behavior of the river system. In Figure 6.6, vertical lines have been overlain which denote the extent of each season and the high flow seasons of each year as H89, H90, etc. are labeled. The seasonal boundaries were determined by taking a running 15-day average of the net Parker release data and marking the onset of the high flow season when

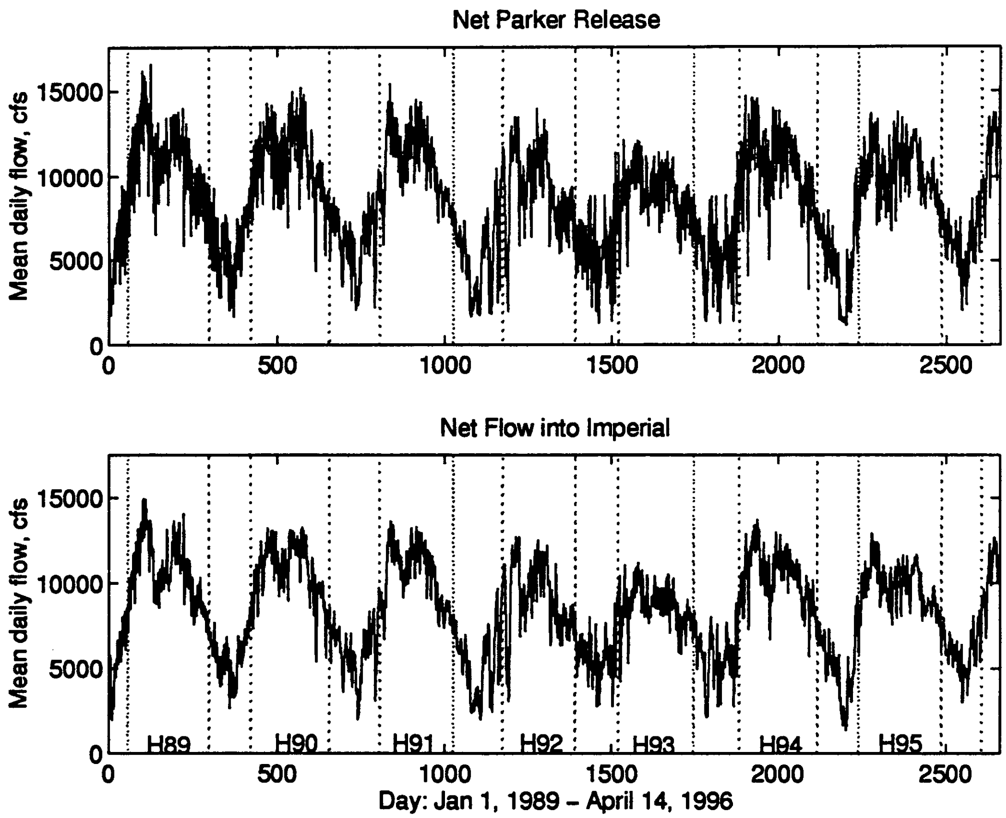

Figure 6.6. Mean Daily Parker Releases and Flow into Imperial After Subtraction of Net Diversions Within the River Segment

the average exceeded 8000 cfs. Conversely, when the average was less than 8000 cfs marked the beginning of the low flow season. If the transitional period oscillated around the 8000 cfs marker (and it did in two of the seven years of data), the earliest crossing was taken when entering the high season and the latest crossing when exiting it. Actual marker dates for the seasons are given in Table 6.2.

Year	Start of High Flow Season	Start of Low Flow Season	Length of High Flow Season, days	Length of Low Flow Season, days
1989	Feb 26	Oct 24	240	124
1990	Feb 25	Oct 17	234	151
1991	Mar 17	Oct 23	220	147
1992	Mar 18	Oct 22	218	128
1993	Feb 27	Oct 13	228	137
1994	Feb 27	Oct 19	234	123
1995	Feb 19	Oct 23	246	120
1996	Feb 20	N/A	N/A	N/A

Table 6.2. Extent of High and Low Flow Seasons for the Period of Study

6.14 Model Validation and Predictions

Recall the error equation model structure given by Equation 6.9:

$$y(t) + \sum_{i=1}^{na} a_i \cdot y(t-i) = \sum_{j=1}^{nb} b_j \cdot u(t-nk-j+1) + e(t)$$

where *na* and *nb* are the number of coefficients (parameters) in the polynomials and *nk* is the nominal delay of the system. A cross-validation test was used to evaluate the performance of all combinations of models ranging from $\{na = 0,1,\ldots,5\}$, $nb = \{1,2,\ldots,5\}$, and $nk = \{1,2,\ldots,5\}$, using data from the 1989 and 1990 high flow seasons for the computation and validation respectively. This totals some 125 different models, but is made easy using the MATLAB software (Ljung 1991). In Figure 6.7, the standard errors for several of the models versus the number of model parameters are plotted. Although the models using

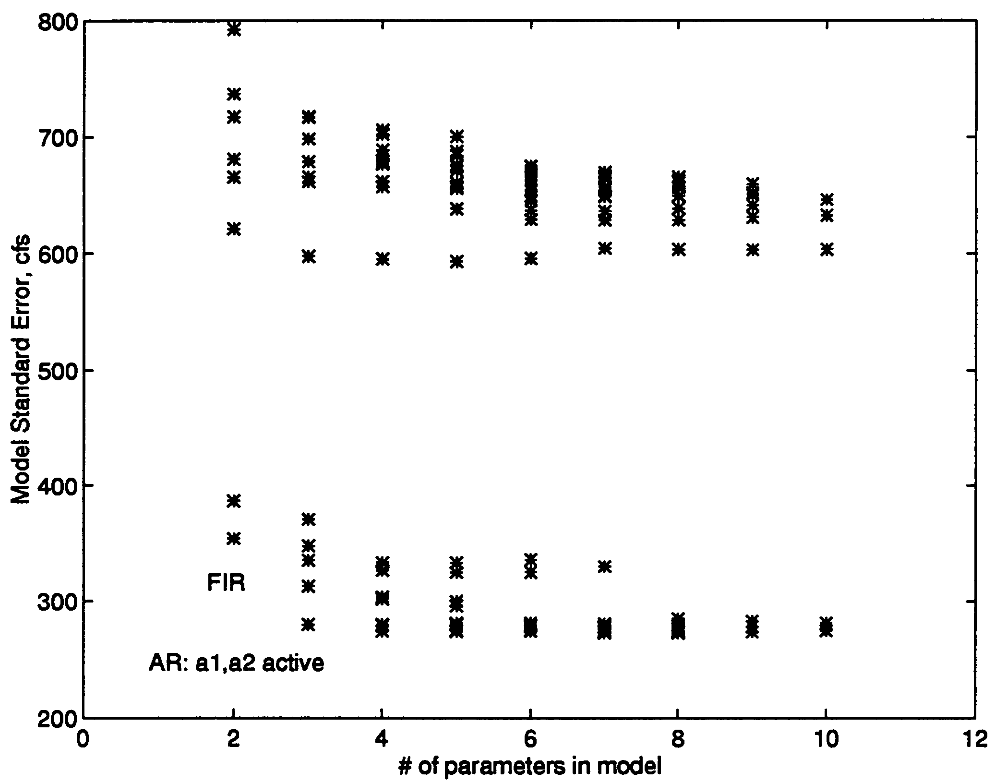

Figure 6.7. Standard Error of Some of the Models Tested Versus the Number of Model Parameters as Described in the Text

active autoregressive coefficients a_1 and a_2 in Equation 6.9 give somewhat better results (the first tier of points in Figure 6.7 marked AR: *a1*, *a2* active), a model that uses those coefficients cannot be used in practice. This is due to the large travel time between the input (Parker Dam) and the output (Imperial Dam). Water orders for water arriving at Imperial must be made 2 to 3 days in advance. Therefore at the time of placing the order, the outputs *y(t)*, *y(t-1)*, and *y(t-2)* are unknown. The next tier of points in Figure 6.7 are all finite impulse response models, (marked FIR in the figure), and the model selection focused on one of those models

In Table 6.3 the values of the computed coefficients, the standard error of each parameter estimate, and the model standard error for the three best FIR models using the same data sets for computation and validation are tabulated. Based on these results, the model given by $nk = 2$ and $nb = 4$ was chosen for this application and is given by

$$y(t) = b_1 \cdot u(t-2) + b_2 \cdot u(t-3) + b_3 \cdot u(t-4) + b_4 \cdot u(t-5) + e(t).$$

	nk=2, *nb*=3	*nk*=2, *nb*=4	*nk*=2, *nb*=5
b1, std error	0.4469, 0.0145	0.4394, 0.0144	0.4361, 0.0146
b2, std error	0.3966, 0.0147	0.3768, 0.0155	0.3752, 0.0155
b3, std error	0.1203, 0.0145	0.0993, 0.0155	0.0924, 0.0163
b4, std error	N/A	0.0483, 0.0144	0.0399, 0.0156
b5, std error	N/A	N/A	0.0202, 0.0147
model std error, cfs	311	300	297

Table 6.3. Computed Coefficients and Standard Errors for Three Finite Impulse Response Models

As a final check of the ability to capture the essential behavior of the system with this FIR model, the autocorrelation of the residuals (prediction error) and the cross-correlation of the residuals with the input signal was computed. These are shown in Figure 6.8. Essentially no correlation as a function of the lag (up to 25 days) is seen so that further modeling of the noise signal is not necessary.

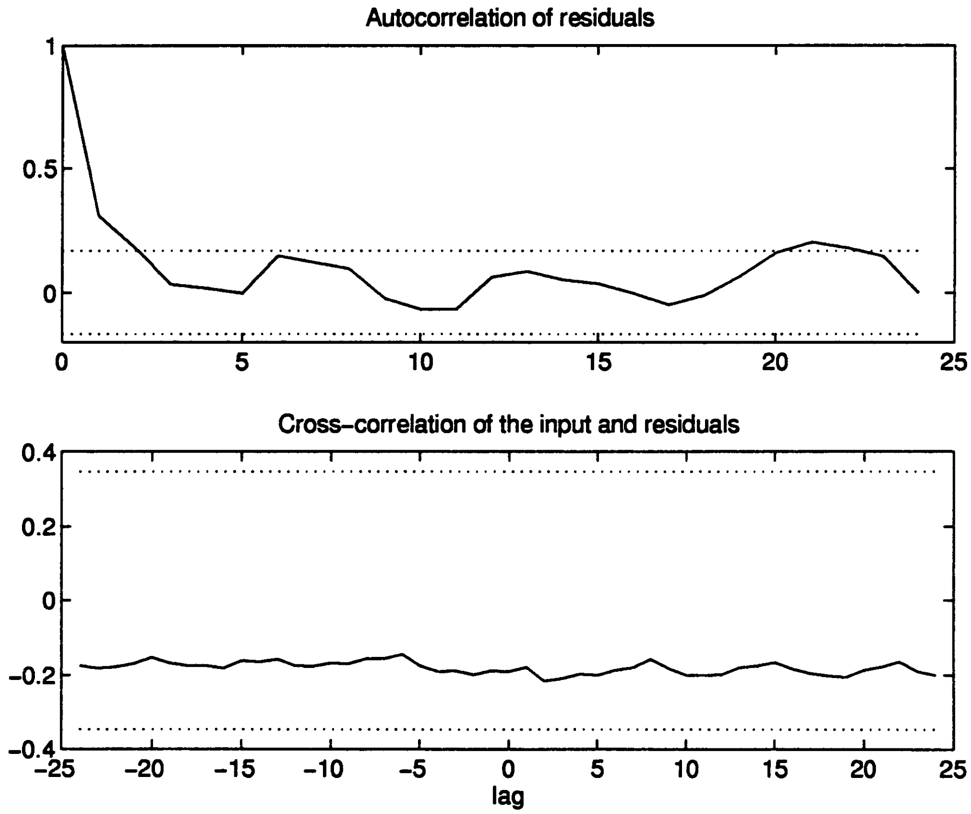

Figure 6.8. Autocorrelation and Cross-correlation of the Residuals and the Input Signal for the FIR Model

In Figure 6.9, the observed inflow, the predicted inflow, and the errors for entire data record using the 4-parameter FIR model derived from the 1989 high flow season data are shown. The model standard error over the entire record is 392 cfs.

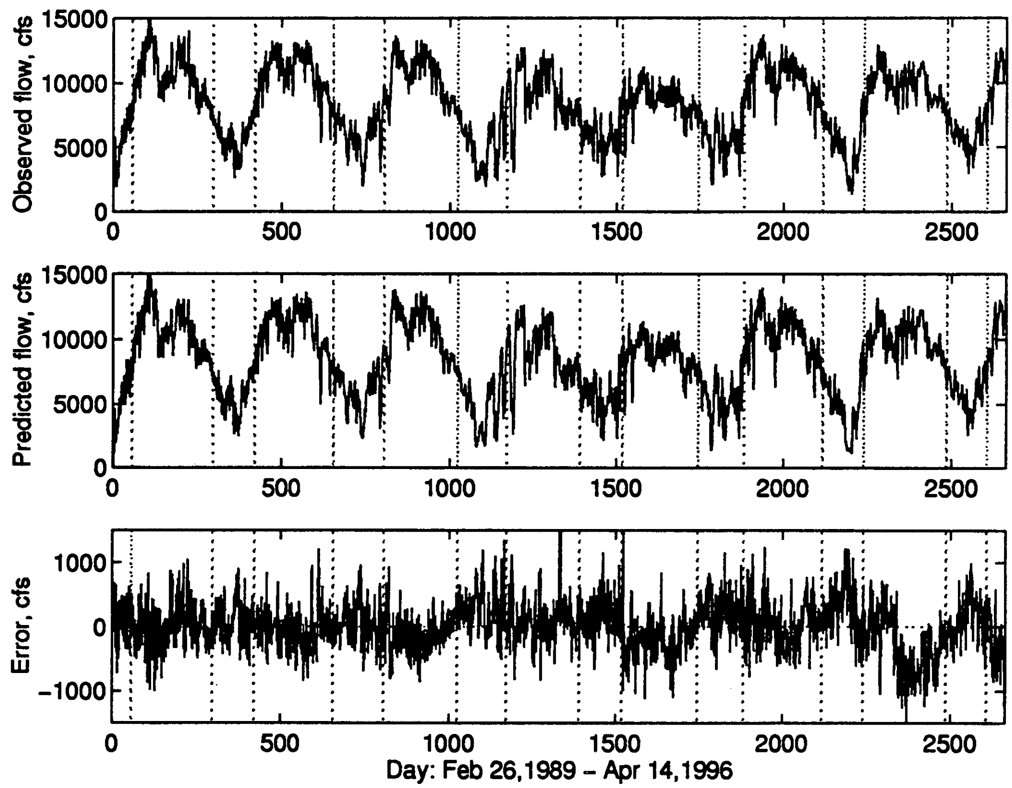

Figure 6.9. Observed Flows, Predicted Flows, and the Error for the FIR Model Computed from the 1989 High Flow Season.

6.15 Model Adaptation

Both the autocorrelation of the residuals and the cross-correlation with the input exhibit some periodicity for long lags (on the order of a year) as shown in Figure 6.10. These trends can be seen in the error signal itself as the model tends to under-predict the inflow during the high flow season and over-predict it during the low flow seasons. Although not all of the low frequency behavior of the error signal is explained in this way (after all, the division of low and high flows is somewhat subjective), the investigation of a model which is adaptive over time is warranted.

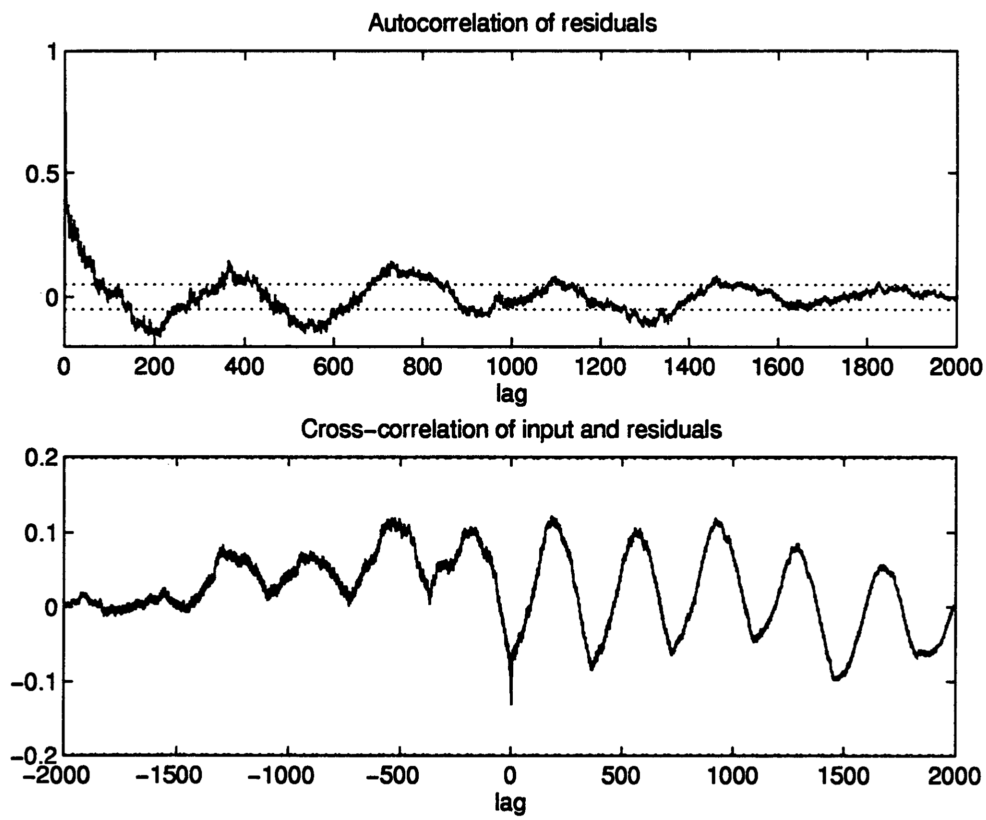

Figure 6.10. Autocorrelation of the Residuals and Cross-correlation of Residuals with the Input for the Four-parameter FIR Model Over the Entire Data Record

Recall the discussion of the weighting for the adaptive scheme (Section 6.9), where λ is the forgetting factor which results in an exponential weight on past errors. In Figure 6.11 the weights as a function of time in the past for a range of λ values are shown.

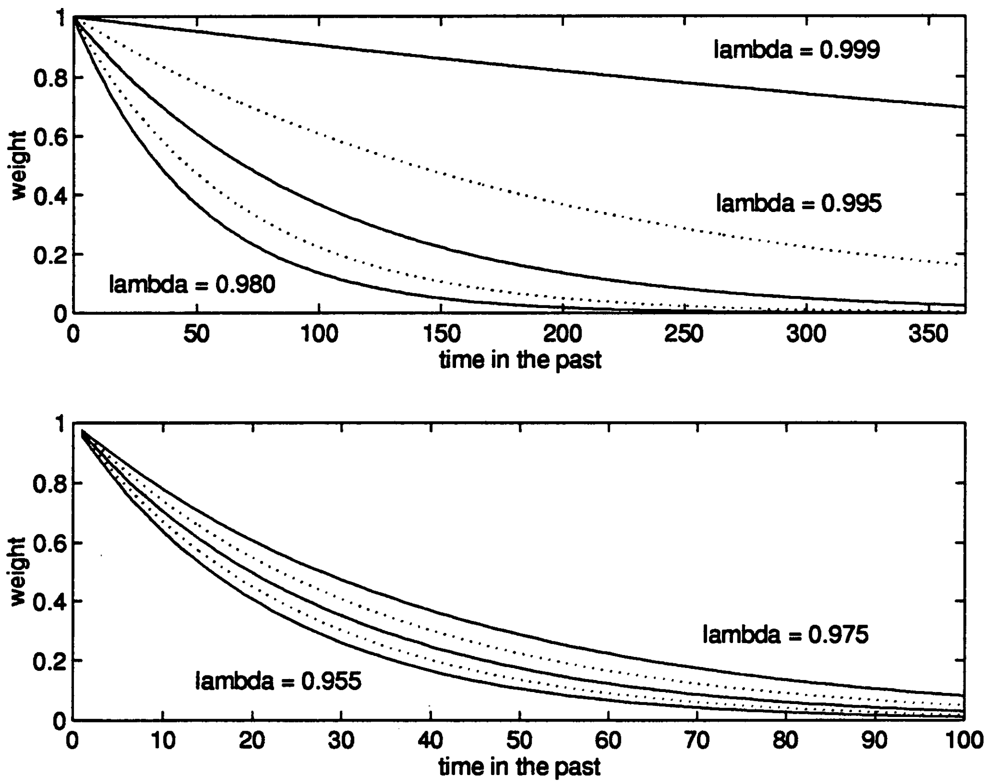

Figure 6.11. Value of Weights Over Time in the Past for Various Forgetting Factors

The model with no adaptation corresponds to $\lambda = 1$. The trade-off is in choosing a λ value that will allow the adaptive scheme to respond fast enough to real changes in the system response while not responding to too fast to noise. In Table 6.4, the model standard errors over the entire record using the range of λ values from 1.0 to 0.955 are given. Decreasing λ past 0.955 did not improve the standard error.

λ	model standard error, cfs-day
1.0	392
0.999	381
0.995	372
0.990	364
0.985	359
0.980	354
0.975	351
0.970	349
0.965	347
0.960	346
0.955	345

Table 6.4. Adaptive Model Standard Errors Over the Entire Record for Various Values of Lambda.

In Figure 6.12, the progression of the weights for the adaptive model for $\lambda = 0.975$ are shown, again with the seasonal boundaries overlain. Two conclusions can be drawn: 1) the adaptive model captures some of the non-linear response of the system due to the high and low flow seasons, and 2) the adaptive model "sees" a major change in the system response beginning in the low flow season of 1990 and extending through the low flow season of

1991. In fact, the FIR model has gone from being "front-loaded" where the b_1 is greater than b_2 to the opposite case and continues to exhibit that behavior throughout the rest of the data record.

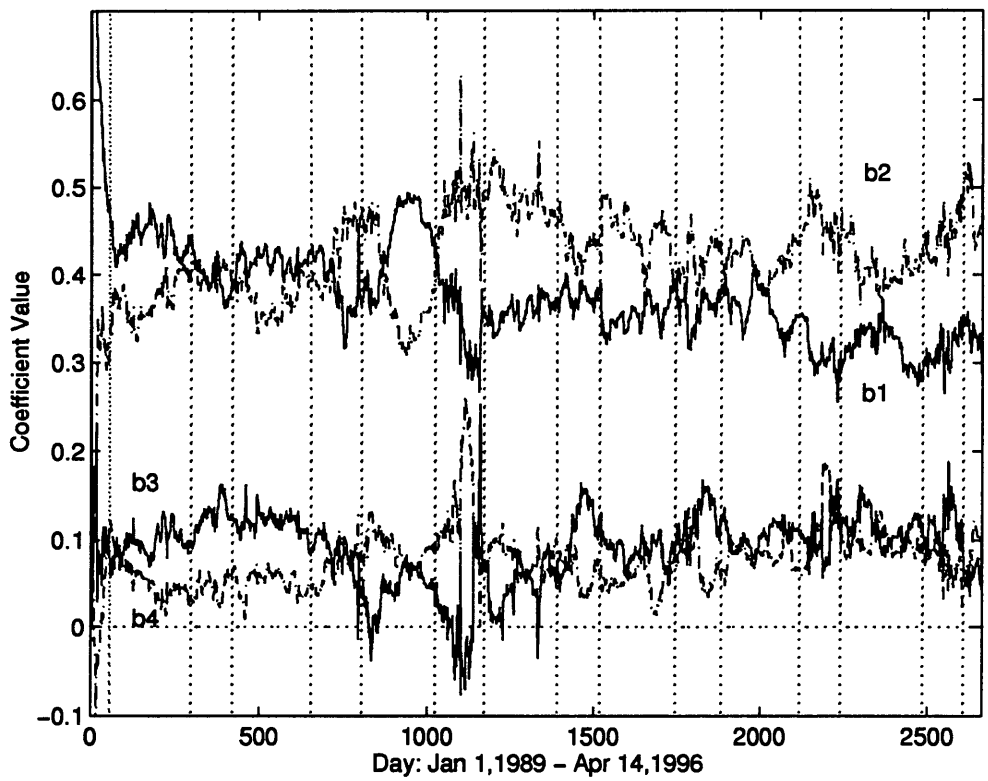

Figure 6.12. Adaptive FIR Model Coefficients for Lambda = 0.975

Although the reason for this long-term shift in system response has not been determined, records prior to a major flood on the entire Colorado river in the late spring of 1983 indicated a similar response. High flows from Parker Dam due to this event persisted into 1987. In Figure 6.13, the observed inflow into Imperial, the predicted inflows, and the adaptive modeling coefficients used for the prediction are shown. By comparing the system response coefficients prior to the large flows to those after, a distinct change in the system response is observed. This is indicative of the changes in the river channel location and geometry and the movement of sediment due to an event of this magnitude and duration.

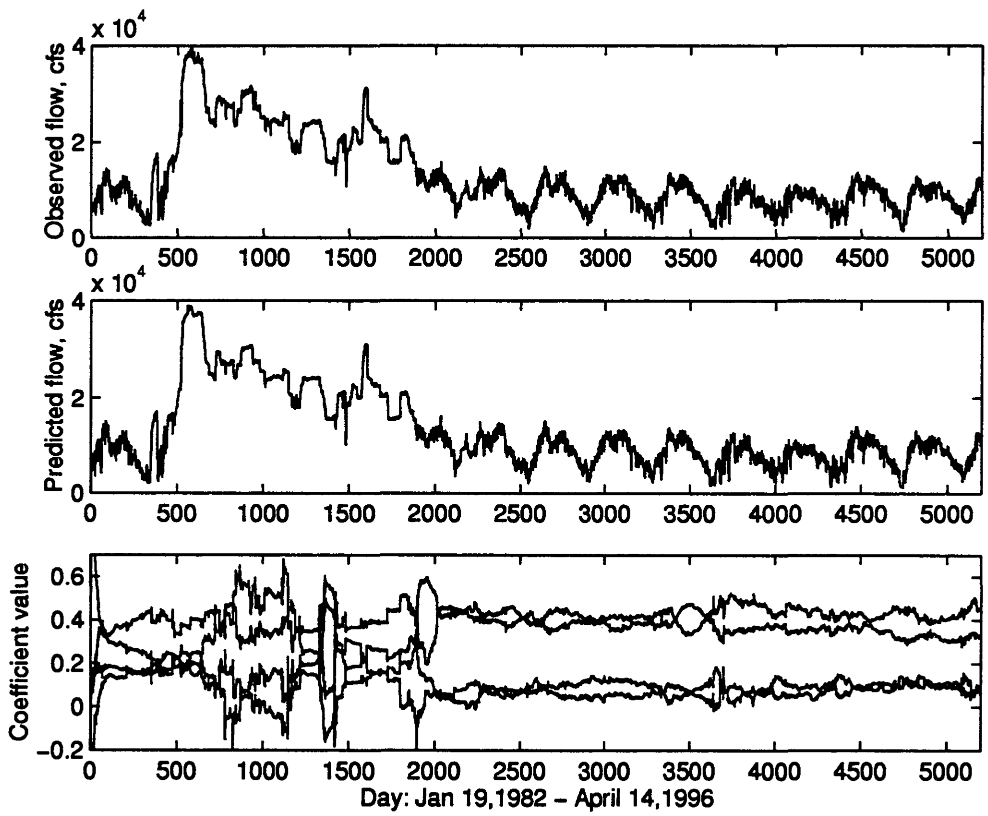

Figure 6.13. Observed Flows, Predicted Flows, and the Adaptive FIR Model Coefficients for the Extended Period of Record Using Lambda = 0.975

6.16 Analysis of Model Standard Errors

The adaptive model improved the model standard error by about 10% (from 392 cfs to 350 cfs). The question is whether it is possible to do better for this example. A quick analysis shows that unless the uncertainty of the data used in the study is decreased, no further improvement in modeling accuracy is expected. Recall the FIR model is given by

$$y(t) = \sum_{j=1}^{nb} b_j \cdot u(t-nk-j) + e(t)$$

Assume first that the model is exact; i.e., that the system responds in exactly this way and the values of the coefficients are known precisely. Then postulate that there is uncertainty $\delta u(t)$ in the measurement of the input signal for each time step and that these measurement errors are independent and random. The uncertainty in the predicted inflow as (see Taylor 1982 for details) can be written as

$$\delta y(t) = \sqrt{\sum_{j=1}^{nb} \{b_j \cdot \delta u(t-nk-j)\}^2}$$

Assuming a constant net Parker release for each day of 10000 cfs and a measurement uncertainty of ±5 %, then the prediction uncertainty for the 4-parameter FIR model given in Table 6.3 would be $\delta y = \pm 295$ cfs. Although the uncertainty in the net Parker release is probably less than ±5 %, given the other uncertainties (particularly the computation of the inflow into Imperial reservoir), significant improvement in the performance of the adaptive model is not possible without significant improvements in the accuracy of the data.

Chapter 7

CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE WORK

In its most general form, the STRS problem is a large, non-linear, non-convex, stochastic optimization problem which cannot be solved to optimality for systems of any realistic size. Given the potential for significant cost savings for large utilities, the problem continues to be of general interest. This study focused on two research areas: 1) formulation of the hydro subproblem as a set of surrogate constraints in the STRS model, and 2) solving a relaxed version of the STRS model to achieve better bounds, while providing feasible solutions using a simple rounding heuristic.

The adaptive, systems identification methodology developed for modeling flow in river channels has been shown to capture the essential behavior of the system. Since the model coefficients are estimated using standard least squares methods, computation of the model parameters is fast. Coupled with an automatic telemetry system to collect the input/output data, the methodology is well suited for practical application in control situations. Finally, the models can adaptively update their parameters to capture time-varying and some non-linear effects inherent in river systems. This model can be used to derive the target energy constraints for facilities governed by downstream water demands.

The relaxation methodology developed for computing tighter bounds for the resulting STRS model has been shown to achieve better bounds than previously used relaxation strategies; however, some work remains to improve the performance of the algorithm. Further research into restriction strategies appears to be the most promising. Investigation of other techniques for deriving the Lagrange multipliers, such as subgradient optimization or other multiplier adjustment methods, may also be warranted. Given an improvement in computational speed, the bounding algorithm could be embedded in a branch and bound procedure to find near-optimal solutions in reasonable times.

A further area of research that may be fruitful is the generation of valid inequalities for the STRS problem. Recent research has shown that problem-specific knowledge can often be applied to generate inequalities that reduce the LP feasible region, providing tighter bounds for branch and bound strategies (Wolsey 1989).

REFERENCES CITED

Bazarra, M.S., Jarvis, J.J., and Sherali, H.D., 1990. *Linear Programming and Network Flows*, John Wiley and Sons, New York, NY.

Brooke, A., Kendrick, D., and Meeraus, A., 1992. *Release 2.25 GAMS: A User's Guide*, The Scientific Press, South San Francisco, CA.

CPLEX, 1994. *Using the CPLEX Callable Library,* CPLEX Optimization, Inc., Incline Village, NV.

Chow, V.T., 1959. *Open Channel Hydraulics,* McGraw-Hill, Inc., New York, NY.

Chow, V.T., Maidment, D.R., and Mays, L.W., 1988. *Applied Hydrology,* McGraw-Hill, Inc., New York, NY.

Cohen, A.I., and Sherkat, V.R., 1987. Optimization-based Methods for Operations Scheduling, in *Proceedings of the IEEE*, Vol. 75, No. 12.

Dooge, J.C.I., Strupczewski, W.G., and Napiorkowski, J.J., 1982. Hydrodynamic derivation of storage parameters of the Muskingum model, *Journal of Hydrology,* 54:371-387.

Duncan, R.A., Seymore, G.E., and Streiffert, D.L., 1985. Optimal Hydrothermal Coordination of Multiple Reservoir River Systems, *IEEE Trans. Power App. System,* Vol. PAS-104, No. 5, May.

Ea, K. and Monti, M., 1985. Daily Operations Planning of the EDF Plantmix—Proposal for a New Method, *Proc. Power Industry Computer Applications,* San Francisco, CA.

EPRI 1977. Synthetic Electric Utility Systems for Evaluating Advanced Technologies, *Electric Power Research Institute Report EM-285*, February.

Erwin, S.R., Griffith, J.S., Wood, J.T., Le, K.D., Day, J.T., Yin, C.K., 1991. Using an Optimization Software to Lower Overall Production Costs for Southern Company, *Interfaces,* Vol. 21, January-February 1991:27-41.

Everett, H.E., 1963. Generalized Lagrange Multiplier Method for Solving Problems of Optimum Allocation of Resources, *Operations Research,* Vol. 11.

Ferreira, L., Andersson, T., Imparato, C.F., Miller, T.E., Pang, C.K., Svoboda, A., and Vojdani, A.F., 1988. *Short-term Resource Scheduling in Multi-area Hydrothermal Power Systems*, Butterworth & Co., Ltd.

Fisher, M.L., 1975. Using Duality to Solve Discrete Optimization Problems: Theory and Computational Experience, *Mathematical Programming Study 3,* North-Holland Publishing Company.

Fisher, M.L., 1981. The Lagrangian Relaxation Method for Solving Integer Programming Problems, *Management Science,* Vol. 27, No. 1.

Gelb, A., Kasper, J.F., Nash, R.A., Price, C.F., and Sutherland, A.A., 1974. *Applied Optimal Estimation,* The MIT Press, Cambridge, MA.

Geoffrion, A.M., 1974. Lagrangean Relaxation for Integer Programming, *Mathematical Programming Study 2,* North-Holland Publishing Company.

Giles, J.E. and Wunderlich, W.O., 1981. Weekly Multipurpose Planning Model for TVA Reservoir System, *Journal of Water Resources,* Planning and Management Division—ASCE, Vol. 107, No. WR2:495-511, October.

Glover, F., Klingman, D., and Phillips, N.V., 1992. *Network Models in Optimization and Their Applications in Practice,* John Wiley and Sons.

Glover, F., Glover, R., and Martinson, F.K., 1984. A Netform System for Resource Planning in the U.S. Bureau of Land Management, *Journal of the Optimization Research Society,* Vol. 35, No. 7.

Glover, F. and Klingman, D., 1988. Layering Strategies for Creating Exploitable Structure in Linear and Integer Programs, *Mathematical Programming,* Vol. 40.

Grimsrud, G.P., Lefton, S.A., and Steury, J., 1995. True Cost of Cycling Enhances the Value of Dispatchable Hydro, *Proceedings of the WaterPower '95 Conference*, ASCE, New York, NY.

Guignard, M., and Kim, S., 1987. Lagrangean Decomposition: A Model Yielding Stronger Lagrangean Bounds, *Mathematical Programming,* Vol. 39.

Habibollahzadeh, H., and Bubenko, J.A., 1986. Application of Decomposition Techniques to Short-term Operation Planning of Hydrothermal Power System, *IEEE Transactions on Power Systems*, Vol. PWRS-1, No. 1.

Illangasekare, T.H., 1985. A Linear Systems Method for Management Modeling of Groundwater Systems, *Hydrological Science and Technology: Short Papers,* Vol. 1, No. 1:25-29.

Lafond, L., 1981. *On the Deterministic Production Planning of a Hydro-electric System,* Ph.D. dissertation, Stanford University, Stanford, CA, December.

Lauer, G.S., Bertsekas, D.P., Sandell, N.R., and Posbergh, T.A., 1982. Solution of Large-scale Optimal Unit Commitment Problems, *IEEE Transactions on Power Apparatus and Systems,* Vol. PAS-101, No. 1, January:79-85.

Ljung, Lennart, 1987. *System Identification—Theory for the User,* Prentice-Hall, Inc., Englewood Cliffs, NJ.

Ljung, Lennart, 1991. *MATLAB: System Identification Toolbox Users Manual,* The MathWorks, Inc., Natick, MA.

Magee, T.M., 1996. Personal communication.

Magee, T.M., and Glover, F., 1996. Integer Programming, in *Mathematical Programming for Industrial Engineers*, edited by Mordecai Avriel and Boaz Golany, published by Marcel Dekker, Chapter 3:123-269.

Magee, T.M., Eschenbach, E.A., and Zagona, E.A., 1995. Object Oriented Optimization for Multi-Reservoir Management, *Proceedings of the WaterPower '95 Conference*, ASCE, New York, NY.

McCarthy, G.T., 1938. *The Unit Hydrograph and Flood Routing*, unpublished manuscript presented at North Atlantic Division conference, U.S. Army Corps of Engineers.

Muckstadt, J.A., and Koenig, S.A., 1977. An Application of Lagrangian Relaxation to Scheduling in Power-Generation Systems, *Operations Research,* Vol. 25, No. 3.

Nemhauser, G.L. and Wolsey, L.A., 1988. *Integer and Combinatorial Optimization,* John Wiley and Sons, New York.

Oppenheim, A.V., and Schafer, R.W., 1975. *Digital Signal Processing,* Prentice-Hall, Inc., Englewood Cliffs, NJ.

Papamichail, D.M., and Papazafiriour, Z.G., 1992. Multiple input-single output linear functional models for river flow routing, *Journal of Hydrology,* 133:365-377.

Rosenthal, R.E., 1981. A Non-linear Network Flow Algorithm for Maximization of Benefits in a Hydro Electric Power System, *Operations Research,* Vol. 29, No. 4, July-August.

Rux, L.M., 1994. An Incremental Economic Dispatch Method for Cascaded Hydroelectric Powerplants, *Bureau of Reclamation Report,* Lakewood, CO.

Shane, R.M., and Boston, W.T., 1983. TVA's Weekly Scheduling Model: Overview and Implementation, *Proceedings of the WaterPower '83 Conference*, ASCE, New York, NY.

Shaw, J.J., and Bertsekas, D.P., 1985. Optimal Scheduling of Large Hydrothermal Power Systems, *IEEE Transactions on Power Apparatus and Systems,* Vol. PAS-104, No. 2.

Sheble, G.B., 1990. Solution of the Unit Commitment Problem by the Method of Unit Periods, *IEEE Transactions on Power Systems,* Vol. 5, No. 1.

Sielvgrem. D. and Dillon, T.S., 1983. Optimal Operations Planning in a Large Hydro-thermal Power System, *IEEE Trans. Power App. System,* Vol. PAS-102, No. 11, November.

Stevens, D., 1992. Personal Communication.

Taylor, J.R., 1992. *An Introduction to Error Analysis: The Study of Uncertainties in Physical Measurements,* University Science Books, Mill Valley, CA.

U.S. Army Corps of Engineers, 1987. Streamflow Synthesis and Reservoir Regulation Model (SSARR), *User Manual,* Army Corps of Engineers, North Pacific Division, Portland, OR.

U.S.Bureau of Reclamation, 1991. *Hydropower 2002: Reclamation's Energy Initiative,* November, Lakewood, CO.

Wolsey, L.A., 1989. *Strong Formulations for Mixed Integer Programming: A Survey,* Mathematical Programming, Vol. 45, pp. 173-191.

Wood and Wollenberg, 1984. *Power Generation Operation and Control,* John Wiley and Sons, New York, NY.

Zagona, E.A., 1992. *Optimal Model-Predictive Supervisory Control of Automated Canals,* Ph.D. Dissertation, University of Colorado, Boulder, CO.

APPENDIX A

COMPUTER CODE AND SCENARIO DATA

```
* GAMS Code for STRS Problem Solved by Column Generation
$OFFUPPER
$OFFSYMXREF OFFSYMLIST
$TITLE STRS model solved by Lagragean Decomposition.
$ONTEXT
Written By: Terry Fulp, USBR
Date: originally written in 1995; modified Summer, 1996
Description:
  Master Problem solved via D-W decomposition.
  Subproblems in units and in time.
  Incremental costs are linear.
  Intercept and start-up costs are included.
  Hourly load must be met.
  Spinning reserve is included.
  Ramping constraints are included.
  Hydros are included with cummulative energy constraints.
$OFFTEXT
* set relative tolerance option (optcr) (default=0.10)
OPTION optcr=0.001;
* more iterations
OPTION iterlim=50000;
* set resource limit
OPTION reslim=20000;
* set number of rows of constraints to list out: default is 3
OPTION limrow = 0
* set number of cols of variables to list out: default is 3
OPTION limcol = 0
* reduce print out for production runs
OPTION SYSOUT= OFF;
OPTION SYSOUT= OFF;
$OFFSYMXREF
$OFFSYMLIST

*  read in data for the scenario
```

```
$INCLUDE '../SCENARIOS/SCENARIO3/DATA'

* the sets are:
*   T           Hours
*   M           Iterations
*   I           List of units
*   K(I)        List of units WITH startup costs
*   NOTK(I)     List of units WITHOUT startup costs
*   J(I)        List of units WITH ramping constraints
*   NOTJ(I)     List of units WITHOUT ramping constraints
*   NH          Set of possible cumm. contraints
*   H(I)        List of hydros
*   TN(H,NH,T)  List of hours in each cum constraint
*   TNH(H,T)    Same as TN for each H to be used in looping
* these correspondence sets are necessary since GAMS doesn't
*  give us a function "is (not) a member of"
*   CORRK(I,K)      correspondence between sets I and K
*   CORRNOTK(I,NOTK) correspondence between sets I and NOTK
*   CORRJ(I,J)      correspondence between sets I and J
*   CORRNOTJ(I,NOTJ) correspondence between sets I and NOTJ
*   CORRH(I,H)      correspondence between sets I and H

*
* TIME Sub-Problems: SPT
*
POSITIVE VARIABLES
  Gt(I)   generation setting for resource I for current time T
  Pt(J)   variable spinning reserve cap for each resource with ramping contraints;

BINARY VARIABLES
  Xt(I)   commitment variable for each resource I (1 if on in time T);

FREE VARIABLES
  StCost  TIME sub-problem cost;

PARAMETERS  computed parameters
  StCostT(M,T)  cost of each TIME subproblem
  LTx(I)      marg. cost from master prob. equality constraint for X
  LTg(I)      marg. cost from master prob. equality constraint for G;

SCALARS
```

```
 Ginit    initial generation
 Xinit    initial commitment
 Load      load for current hour
 Spin     spinning reserve for current hour;

EQUATIONS
 EQST0       sub problem cost function
 EQST1       load constraint for each hour
 EQST2A      spinning reserve constraint for each hour
 EQST2B(J)   spinning reserve capacity limited by maxcap
 EQST2C(J)   spinning reserve capacity limited by ramping
 EQST3A(I)   hourly max gen cap for resource i
 EQST3B(I)   hourly min gen cap for resource i;


EQST0..      SUM(I,(LTx(I)*Xt(I)+LTg(I)*Gt(I))) =E= StCost;

EQST1..      SUM(I,Gt(I)) =E= Load;

EQST2A..     SUM(J,Pt(J))+SUM(NOTJ,Maxcap(NOTJ)*Xt(NOTJ)) =G= Spin;

EQST2B(J)..  -Pt(J)+Maxcap(J)*Xt(J) =G= 0;

EQST2C(J)..  -Pt(J)+Gt(J)+Rampup(J)*Xt(J) =G= 0;

EQST3A(I)..  -Gt(I)+MaxCap(I)*Xt(I) =G= 0;

EQST3B(I)..  Gt(I)-MinCap(I)*Xt(I) =G= 0;

MODEL SPT / EQST0,EQST1,EQST2A,EQST2B,EQST2C,EQST3A,EQST3B / ;
* use CPLEX option file for the TIME Sub-Problems
SPT.optfile=1;

*
* UNIT Sub-Problems: SPU
*

POSITIVE VARIABLES
 Gu(T)      generation setting for time T for current resource I
 Y(T)       startup variable for time T (1 if started in time T);
```

```
FREE VARIABLE
 SH(NH)      slack variable for hydro cum. constraints to turn them on and off;

BINARY VARIABLES
 Xu(T)      commitment variable for time T (1 if on in time T);

PARAMETERS
 Zu(M,I)      UNIT sub problem cost minus dual variable contributions
 SuCostU(M,I) UNIT sub problem cost including dual variable contributions
 CSI          start up cost for unit I
 CII          fixed cost for unit I
 COI          operating cost for unit I
 RampUpI      ramp up limit for unit I
 RampDnI      ramp down limit for unit I
 MaxCapI      max cap for unit I
 MinCapI      min cap for unit I
 GtotH(NH)    cummlative energy values for each constraint for hydro H
 CoeffH       coefficient on unsigned variable to "turn on and off" hydro constraints
 YReal(I,T)   true startup variable (copied from Y(T))
 LUx(T)       marg. cost from master prob. equality constraint for X
 LUg(T)       marg. cost from master prob. equality constraint for G;

FREE VARIABLES
 SuCost       UNIT sub-problem cost;

EQUATIONS
 EQSU0        UNIT sub problem cost function
 EQSU2A(T)    ramp down contraint from original problem
 EQSU2B(T)    ramp up contraint from original problem
 EQSU3A(T)    hourly max gen cap from original problem
 EQSU3B(T)    hourly min gen cap from original problem
 EQSU4(T)     startup constraint
 EQSU5(NH)    cummlative hydro constraints;


EQSU0..      SUM(T, CSI*Y(T)+(CII-LUx(T))*Xu(T)
           +(COI-LUg(T))*Gu(T)) =E= SuCost;

*
* the ORD business adds the extra terms only for the first time period
*   to put in the initial conditions
```

```
*
EQSU2A(T)..   Gu(T)-Gu(T-1)+RampDnI*Xu(T-1)+(RampDnI*Xinit)$(ORD(T) eq 1)
          -Ginit$(ORD(T) eq 1) =G= 0;

EQSU2B(T)..  -Gu(T)+Gu(T-1)+RampUpI*Xu(T)+Ginit$(ORD(T) eq 1) =G= 0;

EQSU3A(T)..  -Gu(T)+MaxCapI*Xu(T) =G= 0;

EQSU3B(T)..   Gu(T)-MinCapI*Xu(T) =G= 0;

EQSU4(T)..   Y(T)-Xu(T)+Xu(T-1)+Xinit$(ORD(T) eq 1) =G= 0;

EQSU5(NH)..   SUM(TNH(NH,T),Gu(T))+CoeffH*SH(NH) =E= GtotH(NH);

MODEL SPU    / EQSU0,EQSU2A,EQSU2B,EQSU3A,EQSU3B,EQSU4,EQSU5 / ;
* use CPLEX option file for the TIME Sub-Problems
SPU.optfile=1;

*
* MASTER PROBLEM
*

POSITIVE VARIABLES
 Lamda(M,T)   weighting variables for each TIME sub-problem solution
 Gamma(M,I)   weighting variables for each UNIT sub-problem solution
 Pg(I,T)      positive deviation variable for equality constraint on the G
 Ng(I,T)      negative deviation variable for equality constraint on the G
 Px(I,T)      positive deviation variable for equality constraint on the X
 Nx(I,T)      negative deviation variable for equality constraint on the X;

FREE VARIABLES
 Mcost        master problem cost;

PARAMETERS
 GMU(I,T,M)    generation levels from UNIT subproblem
 XMU(I,T,M)    commitment from UNIT subproblem
 YMU(I,T,M)    startup from UNIT subproblem
 GMT(I,T,M)    generation levels from TIME subproblem
 XMT(I,T,M)    commitment from TIME subproblem
 LTMg(I,T,M)   marginal costs of equality constraint on G used in TIME subproblem
 LTMx(I,T,M)   marginal costs of equality constraint on X used in TIME subproblem
```

```
  LUMg(I,T,M)    marginal costs of equality constraint on G used in UNIT subproblem
  LUMx(I,T,M)    marginal costs of equality constraint on X used in UNIT subproblem
  McostM(M)      cost of each Master problem
  LboundT(M)     total cost at each iteration from SPT including dual costs
  LboundU(M)     total cost at each iteration from SPU including dual costs
  LboundM(M)     total cost at each iteration (not convex combo) from SPU
  bestLB(M)      best LB found so far at each iteration
  PgM(I,T,M)     all positive deviation for equality constraint on the G
  NgM(I,T,M)     all negative deviation for equality constraint on the G
  PxM(I,T,M)     all positive deviation for equality constraint on the X
  NxM(I,T,M)     all negative deviation for equality constraint on the X
  LamdaM(M,M,T)  saved weights for time subproblems
  GammaM(M,M,I)  saved weights for unit subproblems
  HL(M)          on-off variable if Lamda exists for iteration M
  HG(M)          on-off variable if Gamma exists for iteration M
  GENFT(I,T,M)   convex combo of time subproblem generation
  XFT(I,T,M)     convex combo of time subproblem commitment
  GENFU(I,T,M)   convex combo of unit subproblem generation
  XFU(I,T,M)     convex combo of unit subproblem commitment;

PARAMETERS
  DGENFT(M)      sum of absolute diffs between iterations of genft
  DXFT(M)        sum of absolute diffs between iterations of xft
  DGENFU(M)      sum of absolute diffs between iterations of genfu
  DXFU(M)        sum of absolute diffs between iterations of xfu
  DGMT(M)        sum of absolute diffs between iterations of gmt
  DXMT(M)        sum of absolute diffs between iterations of xmt
  DGMU(M)        sum of absolute diffs between iterations of gmu
  DXMU(M)        sum of absolute diffs between iterations of xmu;

EQUATIONS
  EQM0           master problem cost function
  EQM1           equality in generation (G) constraint
  EQM2           equality in commitment (X) constraint
  EQM3           convex combination for Lamda
  EQM4           convex combination for Gamma;

EQM0..     SUM(I,SUM(M,Gamma(M,I)*Zu(M,I))
         +SUM(T,(Pg(I,T)+Ng(I,T))*PENG + (Px(I,T)+Nx(I,T))*PENX))
         =E= Mcost;
```

```
EQM1(I,T)..   SUM(M,GMU(I,T,M)*Gamma(M,I))-
SUM(M,GMT(I,T,M)*Lamda(M,T))
            +Pg(I,T)-Ng(I,T) =E= 0;

EQM2(I,T)..   SUM(M,XMU(I,T,M)*Gamma(M,I))-
SUM(M,XMT(I,T,M)*Lamda(M,T))
            +Px(I,T)-Nx(I,T) =E= 0;

EQM3(T)..     SUM(M,Lamda(M,T)*HL(M)) =E= 1;

EQM4(I)..     SUM(M,Gamma(M,I)*HG(M)) =E= 1;

MODEL MPROBLEM  / EQM0,EQM1,EQM2,EQM3,EQM4 / ;

*
* Initialize
*
EQM1.m(I,T) = 0.0;
EQM2.m(I,T) = 0.0;
bestLB('1') = 0.0;
*
* set the H's to zero to start
*
LOOP(M, HL(M) = 0);
LOOP(M, HG(M) = 0);

* NOW WE FINALLY SOLVE IT:
* NOTE: data is saved and for output after completion

* LOOP OVER ITERATIONS
LOOP(N,

* LOOP OVER TIME SUBPROBLEMS
  LOOP(T,
*
* copy marginal price info for the subproblems
    LOOP(I,
      LTg(I) = EQM1.m(I,T)$(ORD(N) ne 1) + C(I)$(ORD(N) eq 1);
      LTx(I) = EQM2.m(I,T)$(ORD(N) ne 1) + CI(I)$(ORD(N) eq 1);
    );
```

```
* solve the subproblem for this time
    Load = GR(T);
    Spin = GSR(T);
    SOLVE SPT MINIMIZING StCost USING MIP;

* save the answer for this iteration
    LOOP(I,
      GMT(I,T,N) = Gt.L(I);
      XMT(I,T,N) = Xt.L(I);
      LTMg(I,T,N) = LTg(I);
      LTMx(I,T,N) = LTx(I);
    );
* save the cost for each time and iteration
    StCostT(N,T) = StCost.L;
  );
*
*  LOOP OVER UNIT SUBPROBLEMS
*
  LOOP(I,
*
* copy marginal price info for the subproblems
    LOOP(T,
      LUg(T) = EQM1.m(I,T)$(ORD(N) ne 1) + 0.0$(ORD(N) eq 1);
      LUx(T) = EQM2.m(I,T)$(ORD(N) ne 1) + 0.0$(ORD(N) eq 1);
    );

* copy the data
    COI = C(I);
    CII = CI(I);
    CSI = 0;
    LOOP(K,
      CSI$CORRK(I,K) = CS(K);
    );
    RampUpI = 10000;
    RampDnI = 10000;
    LOOP(J,
      RampUpI$CORRJ(I,J) = RampUp(J);
      RampDnI$CORRJ(I,J) = RampDn(J);
    );
    CoeffH = 1;
    LOOP(H,
```

```
      LOOP(NH,
        GtotH(NH) = Gtot(H,NH);
        TNH(NH,T) = YES$TN(H,NH,T);
        CoeffH$CORRH(I,H) = 0;
      );
    );
*     DISPLAY GtotH,TNH;
    MaxCapI = MaxCap(I);
    MinCapI = MinCap(I);
    Xinit = XI(I);
    Ginit = GI(I);
*     DISPLAY I,COI,CII,CSI,RampUpI,RampDnI,MaxCapI,MinCapI,GtotH;

* solve the subproblem for this unit
    SOLVE SPU MINIMIZING SuCost USING MIP;

* compute the true cost and save the penalized cost
    Zu(N,I) = SUM(T,CSI*Y.L(T)+CII*Xu.L(T)+COI*Gu.L(T));
    SuCostU(N,I) = SuCost.L;
*     DISPLAY Zu;

* save the answer
    LOOP(T,
      GMU(I,T,N) = Gu.L(T);
      XMU(I,T,N) = Xu.L(T);
      YMU(I,T,N) = Y.L(T);
      LUMg(I,T,N) = LUg(T);
      LUMx(I,T,N) = LUx(T);
    );
  );

* set the multiplier to 1 for this iteration
  HG(N) = 1;
  HL(N) = 1;

* solve the Master Problem
  SOLVE MPROBLEM MINIMIZING Mcost USING LP;

* save the answer
  McostM(N) = Mcost.L;
  LOOP(T,
```

```
    LOOP(I,
      PgM(I,T,N) = Pg.L(I,T);
      NgM(I,T,N) = Ng.L(I,T);
      PxM(I,T,N) = Px.L(I,T);
      NxM(I,T,N) = Nx.L(I,T);
    );
   );
* compute the total penalized costs for reporting
* and the lower bound from the master problem
  LboundU(N) = SUM(I,SuCostU(N,I));
  LboundT(N) = SUM(T,StCostT(N,T));
  LboundM(N) = LboundU(N)+LboundT(N);
* skip the first bound since it has real costs in it
  bestLB(N) =  MAX(LboundM(N),bestLB(N-1))$(ORD(N) GT 2) +
           LboundM(N)$(ORD(N) EQ 2) ;

* save the convex combo variables
  LOOP(M$(ORD(M) LE ORD(N)),
    LOOP(T,
      LamdaM(N,M,T) = Lamda.L(M,T);
    );
    LOOP(I,
      GammaM(N,M,I) = Gamma.L(M,I);
    );
   );
*
*  compute the final solution and differences
  LOOP(I,
    LOOP(T,
     GENFT(I,T,N) = SUM(M$(ORD(M) LE ORD(N)),GMT(I,T,M)*LamdaM(N,M,T));
      XFT(I,T,N) = SUM(M$(ORD(M) LE ORD(N)),XMT(I,T,M)*LamdaM(N,M,T));
      GENFU(I,T,N) = SUM(M$(ORD(M) LE ORD(N)),GMU(I,T,M)*Gam-
maM(N,M,I));
      XFU(I,T,N) = SUM(M$(ORD(M) LE ORD(N)),XMU(I,T,M)*GammaM(N,M,I));
    );
  );
);

* output the data
$INCLUDE './MP.bounds.gms'
*$INCLUDE './MP.diffVars.gms'
```

```
*$INCLUDE './SP.diffVars.gms'
*$INCLUDE './MP.dualPrices.gms'
*$INCLUDE './MP.devVariables.gms'
*$INCLUDE './MP.wts.gms'
*$INCLUDE './unitSP.output.gms'
*$INCLUDE './timeSP.output.gms'
```

```
* Author: Terry Fulp, February, 1996
* File: MP.bounds.gms
* this code outputs the bound data from the Master Problem
*   to be read into Lotus
* this file is for the Lagrangian Decomposition Program
*
* output filenames have been defined in the DATA file
* output5 = MP objective function for each iteration
* output7 = MP lower bound each iteration
* output8 = the BEST LB from the MP at each iteration
* define each file, set up for formatted output, quoted text,
*  comma-delimited, set page width to 1000 characters,
PUT output5;
output5.PC = 5;
output5.PW = 1000;
PUT 'Master Problem Objective Function' /;
Put 'SCENARIO',scenario:0:0 /;
PUT output7;
output7.PC = 5;
output7.PW = 1000;
PUT 'Master Problem Objective Function' /;
Put 'SCENARIO',scenario:0:0 /;
PUT output8;
output8.PC = 5;
output8.PW = 1000;
PUT 'Best LB from MP ' /;
Put 'SCENARIO',scenario:0:0 /;

* LOOP OVER ITERATIONS
LOOP(N,
* append to the file
  output5.AP = 1;
  PUT output5;
  PUT ORD(N):0:0, @30, McostM(N) /;
* append to the file
  output7.AP = 1;
  PUT output7;
  PUT ORD(N):0:0, @30, LboundM(N) /;
* append to the file
  output8.AP = 1;
  PUT output8;
```

```
  PUT ORD(N):0:0, @30, bestLB(N) /;
* end loop over iterations
);
PUTCLOSE output5;
PUTCLOSE output7;
PUTCLOSE output8;
```

```
* Author: Terry Fulp, October, 1996
* File: MP.diffVars.gms
* this code computes the sum of abs values in the differences
*   in the solutions from the Master Problem between each iteration
*   to be read into Lotus
* this file is for the Lagrangian Decomposition Program
*
* output filenames have been defined in the DATA file
* output12 = MP.diffVars.out
* define each file, set up for formatted output, quoted text,
*  comma-delimited, set page width to 1000 characters,
PUT output12;
output12.PC = 5;
output12.PW = 255;
PUT 'Master Problem Variable Differences' /;
Put 'SCENARIO',scenario:0:0 /;

* append to the file
output12.AP = 1;
PUT output12;
* compute and output the Master Problem diffs
PUT 'Diffs in MP variables:' /;
PUT @2, 'ITER', @17, 'GENFT', @32,'XFT',@47,'GENFU',@62,'XFU'/;

* LOOP OVER ITERATIONS
LOOP(N,
  Dgenft(N) = 0.0;
  Dxft(N) = 0.0;
  Dgenfu(N) = 0.0;
  Dxfu(N) = 0.0;
  LOOP(I,
    LOOP(T,
      Dgenft(N) = Dgenft(N) + ABS(GENFT(I,T,N) - GENFT(I,T,N-1));
      Dxft(N) = Dxft(N) + ABS(XFT(I,T,N) - XFT(I,T,N-1));
      Dgenfu(N) = Dgenfu(N) + ABS(GENFU(I,T,N) - GENFU(I,T,N-1));
      Dxfu(N) = Dxfu(N) + ABS(XFU(I,T,N) - XFU(I,T,N-1));
    );
  );

  PUT ORD(N):0:0,@15,Dgenft(N),@30,Dxft(N),@45,Dgenfu(N),@60,Dxfu(N)/;
```

```
* end loop over iterations
);
PUTCLOSE output12;
```

```
* Author: Terry Fulp, October, 1996
* File: SP.diffVars.gms
* this code computes the sum of abs values in the differences
*   in the solutions from the subproblems between each iteration
*   to be read into Lotus
* this file is for the Lagrangian Decomposition Program
*
* output filenames have been defined in the DATA file
* output13 = SP.diffVars.out
* define each file, set up for formatted output, quoted text,
*   comma-delimited, set page width to 1000 characters,
PUT output13;
output13.PC = 5;
output13.PW = 255;
PUT 'Sub Problem Variable Differences' /;
Put 'SCENARIO',scenario:0:0 /;

* append to the file
output13.AP = 1;
PUT output13;
* compute and output the diffs
PUT 'Diffs in subproblem variables:' /;
PUT @2, 'ITER', @17, 'GMT', @32,'XMT',@47,'GMU',@62,'XMU'/;

* LOOP OVER ITERATIONS
LOOP(N,
  Dgmt(N) = 0.0;
  Dxmt(N) = 0.0;
  Dgmu(N) = 0.0;
  Dxmu(N) = 0.0;
  LOOP(I,
    LOOP(T,
      Dgmt(N) = Dgmt(N) + ABS(GMT(I,T,N) - GMT(I,T,N-1));
      Dxmt(N) = Dxmt(N) + ABS(XMT(I,T,N) - XMT(I,T,N-1));
      Dgmu(N) = Dgmu(N) + ABS(GMU(I,T,N) - GMU(I,T,N-1));
      Dxmu(N) = Dxmu(N) + ABS(XMU(I,T,N) - XMU(I,T,N-1));
    );
  );

  PUT ORD(N):0:0,@15,Dgmt(N),@30,Dxmt(N),@45,Dgmu(N),@60,Dxmu(N)/;
```

```
* end loop over iterations
);
PUTCLOSE output13;
```

```
* Author:  Terry Fulp, July, 1996
* File: MP.dualPrices.gms
* this code outputs the dual prices from the Master problem
*   for each iteration to be read into Lotus
* this file is for the Lagrangian Decomposition Program
*

* open and setup the file:  output has already been defined by the data file
PUT output6;
* set up for formatted output:  quoted text, comma-delimited columns
output6.PC = 5;
* set page width to the max: 255 characters
output6.PW = 255;

PUT 'Master Problem Dual Prices' /;
Put 'SCENARIO',scenario:0:0 /;

* Generation
LOOP(T,
  PUT 'Dual prices of equality constraint on Generation for Hour =', @70, ORD(T):0:0 /;
  PUT 'Itr' ;
  LOOP(I,
    PUT @(BLANK*(ORD(I)+1)), ORD(I):0:0;
  );
  PUT /;
  LOOP(M,
    PUT ORD(M):0:0;
    LOOP(I,
      PUT @(BLANK*(ORD(I)+1)), LTMg(I,T,M) ;
    );
    PUT /;
  );
);
* Commitment
LOOP(T,
  PUT 'Dual prices of equality constraint on Commitment for Hour =', @70, ORD(T):0:0
/;
  PUT 'Itr' ;
  LOOP(I,
    PUT @(BLANK*(ORD(I)+1)), ORD(I):0:0;
  );
```

```
  PUT /;
  LOOP(M,
    PUT ORD(M):0:0;
    LOOP(I,
      PUT @(BLANK*(ORD(I)+1)), LTMx(I,T,M) ;
    );
    PUT /;
  );
);
PUTCLOSE output6;
```

```
* Author:  Terry Fulp, February, 1996
* File: MP.devVariables.gms
* this code outputs the deviations from the Master Problem
*    to be read into Lotus
* this file is for the Lagrangian Decomposition Program
*
* output filenames have been defined in the DATA file
* output11 = MP.deviations.out
* define each file, set up for formatted output, quoted text,
*   comma-delimited, set page width to 1000 characters,
PUT output11;
output11.PC = 5;
output11.PW = 2000;
PUT 'Master Problem Deviations' /;
Put 'SCENARIO',scenario:0:0 /;

* LOOP OVER ITERATIONS
LOOP(N,
* append to the file
  output11.AP = 1;
  PUT output11;
  PUT 'Iteration ', @30, ORD(N):0:0 /;

* Output the Master Problem Deviation variables
  PUT 'Positive Violation on Generation Equality:' /;
  PUT @8, 'Hour-Unit';
  LOOP(I,
   Put @(BLANK*(ORD(I)+1)), ORD(I):0:0;
  );
  PUT /;
  LOOP(T,
    PUT ORD(T):0:0;
    LOOP(I,
      PUT @(BLANK*(ORD(I)+1)), PgM(I,T,N);
    );
    PUT /;
  );

* Output the Master Problem Deviation variables
  PUT 'Negative Violation on Generation Equality:' /;
  PUT @8,'Hour';
```

```
 LOOP(I,
  Put @(BLANK*(ORD(I)+1)), ORD(I):0:0;
 );
 PUT /;
 LOOP(T,
   PUT ORD(T):0:0;
   LOOP(I,
     PUT @(BLANK*(ORD(I)+1)), NgM(I,T,N);
   );
   PUT /;
 );

* Output the Master Problem Deviation variables
 PUT 'Positive Violation on Commitment Equality:' /;
 PUT @8, 'Hour';
 LOOP(I,
  Put @(BLANK*(ORD(I)+1)), ORD(I):0:0;
 );
 PUT /;
 LOOP(T,
   PUT ORD(T):0:0;
   LOOP(I,
     PUT @(BLANK*(ORD(I)+1)), PxM(I,T,N);
   );
   PUT /;
 );

* Output the Master Problem Deviation variables
 PUT 'Negative Violation on Commitment Equality:' /;
 PUT @8,'Hour';
 LOOP(I,
  Put @(BLANK*(ORD(I)+1)), ORD(I):0:0;
 );
 PUT /;
 LOOP(T,
   PUT ORD(T):0:0;
   LOOP(I,
     PUT @(BLANK*(ORD(I)+1)), NxM(I,T,N);
   );
   PUT /;
 );
```

```
* end loop over iterations
);
PUTCLOSE output11;
```

```
* Author:  Terry Fulp, February, 1996
* File: MP.wts.gms
* this code outputs the weights from the Master Problem
*    to be read into Lotus
* this file is for the Lagrangian Decomposition Program
*
* output filenames have been defined in the DATA file
* output4 = lamdas and gammas from each iteration
* define each file, set up for formatted output, quoted text,
*   comma-delimited, set page width to 1000 characters,
PUT output4;
output4.PC = 5;
output4.PW = 1000;
PUT 'Master Problem Weights' /;
Put 'SCENARIO',scenario:0:0 /;

* LOOP OVER ITERATIONS
LOOP(N,
* append to the file
  output4.AP =1;
  PUT output4;
*  Output the weights for the Time Subproblems
  PUT 'Lamdas used to weight the time Subproblems:' /;
  PUT 'Itr are across the page; time is down the page' /;
  PUT  'Hour';
  LOOP(M $ (ORD(M) LE ORD(N)),
   Put @(BLANK*(ORD(M)+1)), ORD(M):0:0;
  );
  PUT /;
  LOOP(T,
    PUT ORD(T):0:0;
    LOOP(M $ (ORD(M) LE ORD(N)),
      PUT @(BLANK*(ORD(M)+1)), LamdaM(N,M,T);
    );
    PUT /;
  );

*  Output the weights for the Unit Subproblems
  PUT 'Gammas used to weight the unit Subproblems:' /;
  PUT 'Itr are across the page; units are down the page' /;
  PUT 'Unit';
```

```
  LOOP(M $ (ORD(M) LE ORD(N)),
   Put @(BLANK*(ORD(M)+1)), ORD(M):0:0;
  );
  PUT /;
  LOOP(I,
    PUT ORD(I):0:0;
    LOOP(M $ (ORD(M) LE ORD(N)),
      PUT @(BLANK*(ORD(M)+1)), GammaM(N,M,I);
    );
    PUT /;
  );
* end loop over iterations
);
PUTCLOSE output4;
```

```
* Author:  Terry Fulp, February, 1996
* File: unitSP.output.gms
* this code outputs the data for the unit subProblems
*    to be read into Lotus
* this file is for the Lagrangian Decomposition Program
*

* open and setup the file:  output has already been defined by the data file
PUT output2;
* set up for formatted output:  quoted text, comma-delimited columns
output2.PC = 5;
* set page width to 1000 characters
output2.PW = 1000;

PUT 'Problem Solved with L. Decomposition Scheme' /;
Put 'SCENARIO',scenario:0:0 /;
PUT 'Units are across the page; time is down the page' /;

* LOOP OVER ITERATIONS
LOOP(N,
  PUT 'Iteration ', @30, ORD(N):0:0 /;

* Output the generation for each UNIT subproblem
  PUT 'Unit Subproblem Generation:' /;
  PUT @8,'Hour';
  LOOP(I,
   PUT @(BLANK*(ORD(I)+1)), ORD(I):0:0;
  );
  PUT /;
  LOOP(T,
    PUT ORD(T);
    LOOP(I,
      PUT @(BLANK*(ORD(I)+1)), GMU(I,T,N);
    );
    PUT /;
  );

* Output the Commitment for each UNIT subproblem
  PUT 'Unit Subproblem Commitment:' /;
  PUT @8, 'Hour';
  LOOP(I,
```

```
  Put @(BLANK*(ORD(I)+1)), ORD(I):0:0;
 );
 PUT /;
 LOOP(T,
   PUT ORD(T):0:0;
   LOOP(I,
     PUT @(BLANK*(ORD(I)+1)), XMU(I,T,N);
   );
   PUT /;
 );

* Output the Startup for each UNIT subproblem
 PUT 'Unit Subproblem StartUp:' /;
 PUT @8, 'Hour';
 LOOP(I,
  Put @(BLANK*(ORD(I)+1)), ORD(I):0:0;
 );
 PUT /;
 LOOP(T,
   PUT ORD(T):0:0;
   LOOP(I,
     PUT @(BLANK*(ORD(I)+1)), YMU(I,T,N);
   );
   PUT /;
 );

* end loop over iterations
);
PUTCLOSE output2;
```

```
* Author: Terry Fulp, February, 1996
* File: timeSP.output.gms
* this code outputs the data for the time SubProblems
*   to be read into Lotus
* this file is for the Lagrangian Decomposition Program
*

* open and setup the file: output has already been defined by the data file
PUT output1;
* set up for formatted output: quoted text, comma-delimited columns
output1.PC = 5;
* set page width to 255 characters
output1.PW = 255;

PUT 'Problem Solved with L. Decomposition Scheme' /;
Put 'SCENARIO',scenario:0:0 /;
PUT 'Units are across the page; time is down the page' /;

* LOOP OVER ITERATIONS
LOOP(N,
  PUT 'Iteration ', @30, ORD(N):0:0 /;

* Output the generation for each TIME subproblem
  PUT 'Time Subproblem Generation:' /;
  PUT @8, 'Hour';
  LOOP(I,
   Put @(BLANK*(ORD(I)+1)), ORD(I):0:0;
  );
  PUT /;
  LOOP(T,
    Put ORD(T):0:0;
    LOOP(I,
      Put @(BLANK*(ORD(I)+1)), GMT(I,T,N);
    );
    PUT /;
  );

* Output the Commitment for each TIME subproblem
  PUT 'Time Subproblem Commitment:' /;
  PUT @8,'Hour'
  LOOP(I,
```

```
    PUT @(BLANK*(ORD(I)+1)), ORD(I):0:0;
  );
  PUT /;
  LOOP(T,
    PUT ORD(T):0:0;
    LOOP(I,
      Put @(BLANK*(ORD(I)+1)), XMT(I,T,N);
    );
    PUT /;
  );

* end loop over iterations
);
PUTCLOSE output1;
```

```
*
* Data for Scenario 1
* the test problem with ramping, tight spin, and 1 hydro
*

* define the output file names for all solutions
FILE output0 /'../SCENARIOS/SCENARIO1/MIP.out'/;
FILE output9 /'../SCENARIOS/SCENARIO1/LP.out'/;
FILE output1 /'../SCENARIOS/SCENARIO1/timeSP.out'/;
FILE output2 /'../SCENARIOS/SCENARIO1/unitSP.out'/;
FILE output3 /'../SCENARIOS/SCENARIO1/MP.variables.out'/;
FILE output4 /'../SCENARIOS/SCENARIO1/MP.wts.out'/;
FILE output5 /'../SCENARIOS/SCENARIO1/MP.obj.out'/;
FILE output6 /'../SCENARIOS/SCENARIO1/MP.dualPrices.out'/;
FILE output7 /'../SCENARIOS/SCENARIO1/MP.lowerBound.out'/;
FILE output8 /'../SCENARIOS/SCENARIO1/MP.bestLB.out'/;
FILE output10 /'../SCENARIOS/SCENARIO1/FEASIBLE.out'/;

* for output
SCALAR BLANK number of blanks between data /12/;

* define the scenario number
SCALAR
  scenario /1/;

* define the sets
* NOTE: we are assuming that all units have an intercept cost
*     (some may be zero); however, not all units must have startup costs
SETS T        Hours                              /1*8/
    M        Iterations                          /1*50/
    I        List of units                       /1*5/
    K(I)     List of units WITH startup costs            /1*3/
    NOTK(I)  List of units WITHOUT startup costs         /4*5/
    J(I)     List of units WITH ramping constraints      /1*4/
    NOTJ(I)  List of units WITHOUT ramping constraints   /5/
    NH       Set of possible cumm. contraints            /1/
    H(I)     List of hydros                              /5/
    TN(H,NH,T) List of hours in each cum constraint      / 5.1.1*8 /
    TNH(NH,T)  Dynamic set to hold the TN for each H;

* this alias needed since gams can't handle the loops very well
```

```
*
ALIAS(N,M);

* these sets needed since gams doesn't give us an "is a member" function
*  too bad that we have to build these
SETS
  CORRK(I,K) correspondence between sets I and K
    /1.1, 2.2, 3.3 /
  CORRNOTK(I,NOTK) correspondence between sets I and NOTK
    /4.4, 5.5 /
  CORRJ(I,J) correspondence between sets I and J
    /1.1, 2.2, 3.3, 4.4 /
  CORRNOTJ(I,NOTJ) correspondence between sets I and NOTJ
    /5.5 /
  CORRH(I,H) correspondence between sets I and H
    /5.5 /;

* define and set the penalty variables for the master problem
SCALARS
 PENG     value of penalty for equality in G violations in MP
 PENX     value of penalty for equality in X violations in MP;
PENG = 500;
PENX = 3000;

* the Data for the units
PARAMETERS  data
 MaxCap(I) maximum capacity of unit i
      / 1 80, 2 250, 3 300, 4 60, 5 100/
 MinCap(I) minimum capacity of unit i
      / 1 25, 2 60, 3 75, 4 20, 5 0/
 RampUp(J) ramp up limits for unit i
      / 1 40, 2 200, 3 200, 4 20/
 RampDn(J) ramp down limits for unit i
      / 1 40, 2 200, 3 200, 4 20/
 C(I)    incremental generation cost of unit i
      / 1 23.54, 2 20.34, 3 19.74, 4 28.00, 5 0.00/
 CI(I)   intercept cost of unit i
      / 1 213.00, 2 585.62, 3 684.74, 4 252.00, 5 0.00/
 CS(K)    startup cost of unit i
      / 1 350.00, 2 400.00, 3 1100.00 /;
*
```

```
* load data
*
PARAMETERS
 GR(T)    hourly loads
       / 1 450.00, 2 530.00, 3 600.00, 4 540.00,
         5 400.00, 6 280.00, 7 290.00, 8 500.00/

  GSR(T) spinning reserve;

TABLE GTot(H,NH) cummulative energy targets for each hydro

     1
 5   500 ;

SCALAR
  spinFactor spinning reserve factor  /1.10/;

LOOP(T,
  GSR(T) = GR(T)*spinFactor;
);

*
* initial data
*
PARAMETERS
 XI(I)  initial unit commitment
       / 1 0, 2 1, 3 1, 4 0, 5 0/
 GI(I)  initial generation
       / 1 0.0, 2 100.0, 3 300.0, 4 0.0, 5 0.0/;
```

```
*
* Data for Scenario 2
* the test problem with tight ramping, spin, and 1 hydro
*

* define the output file names for all solutions
FILE output0 /'../SCENARIOS/SCENARIO2/MIP.out'/;
FILE output9 /'../SCENARIOS/SCENARIO2/LP.out'/;
FILE output1 /'../SCENARIOS/SCENARIO2/timeSP.out'/;
FILE output2 /'../SCENARIOS/SCENARIO2/unitSP.out'/;
FILE output3 /'../SCENARIOS/SCENARIO2/MP.variables.out'/;
FILE output4 /'../SCENARIOS/SCENARIO2/MP.wts.out'/;
FILE output5 /'../SCENARIOS/SCENARIO2/MP.obj.out'/;
FILE output6 /'../SCENARIOS/SCENARIO2/MP.dualPrices.out'/;
FILE output7 /'../SCENARIOS/SCENARIO2/MP.lowerBound.out'/;
FILE output8 /'../SCENARIOS/SCENARIO2/MP.bestLB.out'/;
FILE output10 /'../SCENARIOS/SCENARIO2/FEASIBLE.out'/;

* for output
SCALAR BLANK number of blanks between data /12/;

* define the scenario number
SCALAR
  scenario /2/;

* define the sets
* NOTE: we are assuming that all units have an intercept cost
*     (some may be zero); however, not all units must have startup costs
SETS T        Hours                          /1*8/
    M        Iterations                     /1*100/
    I        List of units                  /1*6/
    K(I)     List of units WITH startup costs        /1*5/
    NOTK(I)  List of units WITHOUT startup costs     /6/
    J(I)     List of units WITH ramping constraints     /1*5/
    NOTJ(I)  List of units WITHOUT ramping constraints  /6/
    NH       Set of possible cumm. contraints       /1/
    H(I)     List of hydros                  /6/
    TN(H,NH,T) List of hours in each cum constraint     / 6.1.1*8 /
    TNH(NH,T)  Dynamic set to hold the TN for each H;

* this alias needed since gams can't handle the loops very well
```

```
*
ALIAS(N,M);

* these sets needed since gams doesn't give us an "is a member" function
*  too bad that we have to build these
SETS
  CORRK(I,K) correspondence between sets I and K
    /1.1, 2.2, 3.3, 4.4 , 5.5/
  CORRNOTK(I,NOTK) correspondence between sets I and NOTK
    /6.6 /
  CORRJ(I,J) correspondence between sets I and J
    /1.1, 2.2, 3.3, 4.4, 5.5 /
  CORRNOTJ(I,NOTJ) correspondence between sets I and NOTJ
    /6.6 /
  CORRH(I,H) correspondence between sets I and H
    /6.6 /;

* define and set the penalty variables for the master problem
SCALARS
 PENG     value of penalty for equality in G violations in MP
 PENX     value of penalty for equality in X violations in MP;
PENG = 500;
PENX = 3000;

* the Data for the units
PARAMETERS  data
 MaxCap(I) maximum capacity of unit i
      / 1 80, 2 250, 3 300, 4 60, 5 100, 6 100/
 MinCap(I) minimum capacity of unit i
      / 1 20, 2 30, 3 30, 4 20, 5 20, 6 0/
 RampUp(J) ramp up limits for unit i
      / 1 20, 2 75, 3 75, 4 20, 5 20/
 RampDn(J) ramp down limits for unit i
      / 1 20, 2 75, 3 75, 4 20, 5 20/
 C(I)     incremental generation cost of unit i
      / 1 23.54, 2 20.34, 3 19.74, 4 28.00, 5 26.00, 6 0.00/
 CI(I)    intercept cost of unit i
      / 1 213.00, 2 585.62, 3 684.74, 4 252.00, 5 300.00, 6 0.00/
 CS(K)    startup cost of unit i
      / 1 350.00, 2 400.00, 3 1100.00, 4 100.00 , 5 200.00 /;
*
```

```
* load data
*
PARAMETERS
 GR(T)    hourly loads
       / 1 500.00, 2 600.00, 3 650.00, 4 450.00,
         5 350.00, 6 520.00, 7 650.00, 8 400.00/
*        / 1 450.00, 2 530.00, 3 600.00, 4 540.00,
*          5 400.00, 6 280.00, 7 290.00, 8 500.00/

  GSR(T) spinning reserve;

TABLE GTot(H,NH) cummulative energy targets for each hydro

      1
 6   100.0 ;

SCALAR
  spinFactor spinning reserve factor  /1.00/;

LOOP(T,
  GSR(T) = GR(T)*spinFactor;
);

*
* initial data
*
PARAMETERS
 XI(I)  initial unit commitment
       / 1 0, 2 1, 3 1, 4 0, 5 0, 6 0/
 GI(I)  initial generation
       / 1 0.0, 2 100.0, 3 300.0, 4 0.0, 5 0.0, 6 0.0/;
```