Capital Budgeting Decision Making:

Database, Aggregation and Disaggregation

Methods for a Large Scale Problem

By

Yi Liang

Supervised by

Professor Daniel Lane

Systems Science Programme

University of Ottawa

ISBN 0-612-00479-1

Acknowledgment

I wish to express my gratitude to Dr. D. Lane for his supervision and encouragement. Special appreciation is expressed to Dr. J. M. Thizy, Dr. S. Rawat and Dr. J. Doutriaux for their help.

Contents

List of Tables and Figures

Tables

Figures

Appendices

Appendix D

Capital Allocation Summaries

Appendix E

GAMS Input and Output

Chapter 1

Introduction

Capital budgeting is the process of allocating scarce resources among alternative capital projects so as to meet company objectives in the best possible way. All companies are faced with capital budgeting problems that arise when different and competing alternative activities are to be funded out of a limited budget and not all alternatives can be totally funded. This problem is particularly acute in these years of fiscal and economic restraint. Therefore, decision-makers must decide which capital activities contribute most to the company's objectives. Such activities should be given higher priority to the limited funds.

Capital budgeting problems have usually been solved in unstructured analyses that typically take into account the following steps. First, decision-makers determine the total amount of capital to be allocated and what activities should be considered for funding. Next, qualitative statement aspects such as mission of the company are used directly or indirectly to define several criteria to measure the contribution of each activity. Capital allocation must also take into account organizational and operational constraints. According to the overall contribution of each activity to the mission of company and constraints, the decision-maker can rank alternatives for allocation and make decisions regarding the use of capital funds.

This thesis considers the annual capital budgeting problem of a large Canadian telecommunications company where decisions about what sets of alternative projects will be funded each year must be made. Problem difficulty increases when the number of candidate projects is large and there are competing and conflicting objectives for the use of capital funds in the company. This thesis proposes the development of a database system and an aggregation/disaggregation deterministic model solution procedure to aid decision makers to analyze and resolve the annual capital budgeting problem for an initial set of over 100 alternative capital projects.

1.1 Problem Definition

In medium sized to large companies the annual capital budgeting process can involve hundreds of millions of dollars and an ongoing commitment from dedicated staff in dealing with the capital budget allocation directly. The large Canadian telecommunication company that serves as the test case for this thesis has an annual capital budget of several hundred million dollars for modernizing its telecommunication network and adding new services. Moreover, dedicated managerial staff are continually motivated with and responsible for capital allocations to over 100 discretionary projects each year.

The capital budgeting decision-maker faces a number of difficulties. First, how to compare different projects and evaluate their priorities on several objectives. Second, when there are large numbers of projects that need to be funded, it becomes very difficult to evaluate each project's priority at project level based on the overall company mission. If projects are aggregated, the problem is how to evaluate the priority of these aggregated project groups. As well, decision-makers must consider organizational and operational constraints in capital allocation process. It is important to find an effective aggregation procedure over many projects to make allocation decisions that do not violate particular company restrictions.

In this thesis, quantitative procedures in support of capital allocation decisions over many candidate capital projects are presented. These procedures include aggregation/disaggregation modeling and aggregate mathematical programming formulations examined on a set of over 100 capital projects. Project information and allocation results are organized in a computerized database that permit feasible review and analysis of alternative approaches and results. The size and information content of the model presented in the thesis are directly comparable to the annual capital allocation problem for the telecommunications company from which the database was derived. The results provide a management tool for the structured analysis of capital allocations that has potential to save resources and improve capital budgeting decision making.

1.2 Plan of the Thesis

Chapter 2 presents a survey of literature relevant to the thesis. It consists of brief reviews on: (1) Mathematical programming methods for capital budgeting, (2) Multiple criteria decision making models for capital allocations, and (3) Aggregation and disaggregation techniques for problem solution.

Chapter 3 describes the proposed procedure to solve the large scale capital budgeting problem. The problem is presented as a large scale multiple criteria decision making problem. The methodology includes a structured hierarchy decomposition of the large problem into functional or organizational groups. The Analytical Hierarchy Process (AHP) is used to provide key impacts to the multicriteria linear programming problems. As an alternative approach, a large scale linear programming formulation is developed with additional constraints for relative comparison to the decomposition approach.

Chapter 4 presents the results of the methodologies developed in previous chapter. The methods are applied to a complete database based on information from a large Canadian telecommunication company's annual capital budget allocation problem. Database software development and capital allocation results are included in this chapter.

Chapter 5 presents the conclusions, and summaries of the applied methods. Several extensions to the current research are discussed toward implementation of the methodology.

Chapter 2

Literature Review

The large scale capital budgeting problem involves a set of coordinated analyses including research in the following three areas:

(1) mathematical programming problem formulation;

(2) multiple criteria decision making objective functionals;

(3) problem aggregation and disaggregation methods.

Studies previously presented in the literature and directly related to the current analysis are considered in turn below. Relevant publications in each of these areas as pertains to the capital budgeting problem are discussed.

2.1 Mathematical Programming Methods

Charnes, Cooper and Miller (1959) and Weingartner (1963) were the first to formulate a linear programming model to solve capital budgeting problems in a deterministic model structure. Charnes, Cooper and Miller formulated a dynamic linear programming model to assist the firm in performing capital budgeting considering both operating decisions and financial planning. Weingartner formulated a multi-period single objective linear programming model to maximize the total net present value given the capital requirements of the projects for each year. The formulation below presents the model proposed by Weingartner:

$$\text{Max} \quad \sum_{j=1}^{N} b_j x_j$$

$$\text{subject to} \quad \sum_{j=1}^{N} c_{tj} x_j \le C_t \quad \text{for } t=1,\cdots,T$$

$$0 \le x_j \le 1 \quad \text{for } j=1,\cdots,N$$

where

b_j is the net present value of the project j.

c_{tj} is the capital required by project j at period t.

C_t is the maximum total permissible expenditure during period t.

x_j is the fraction of project funded.

N is the number of projects.

T is the number of time periods.

The optimal solution to this problem can be obtained using the linear programming simplex algorithm (Dantzig 1963). To illustrate the general approach, consider the following four period capital budgeting problem with four projects:

Project, j	b_j (NPV)	c_{1j}	c_{2j}	c_{3j}	c_{4j}
1	90	15	20	20	15
2	40	10	15	20	5
3	10	10	0	0	4
4	37	15	10	10	10
Budget available ($) at time t:		C_1=40	C_2=50	C_3=40	C_4=35

Table 2.1 Capital Budgeting Linear Programming Example

The linear programming formulation is as follows:

Formulation 2.1

$$\text{Max} \quad 90x_1 + 40x_2 + 10x_3 + 37x_4$$

$$\begin{aligned} \text{subject to} \quad & 15x_1 + 10x_2 + 10x_3 + 15x_4 \le 40 \\ & 20x_1 + 5x_2 + 0x_3 + 10x_4 \le 50 \\ & 20x_1 + 20x_2 + 0x_3 + 10x_4 \le 40 \\ & 15x_1 + 5x_2 + 4x_3 + 10x_4 \le 35 \\ & 0 \le x_i \le 1 \qquad \text{for } i = 1, 2, 3, 4 \end{aligned}$$

where x_i denotes the fractional investment in each project

The above linear program is solved using Lindo/PC (Schrange 1986) and the optimum solution is $x_1=1$, $x_2=0.5$, $x_3=0.5$, $x_4=1$, with a maximum total net present value of \$152.

In this example, the decision variables represent fractional allocations of the candidate projects. However, sometimes it is not possible to define fractional projects. To select nonfractional projects, integer programming is required, where the decision variables are taken to be $x_i=0$ or 1, indicating that the *i*th project is either rejected or accepted. Weingartner presented a integer programming formulation by forcing the decision variables x_i to be either zero or one, i.e.,

$$x_i \in \{ 0, 1 \} \text{ for } i=1...n$$

Consider the addition of this extra constraint to Example 2.2. The problem is solved again using Lindo/PC and the optimum solution is $x_1=1$, $x_2=1$, $x_3=1$, $x_4=0$ with maximum total net present value of \$140. Note that this solution could not have been discovered by simple rounding the linear programming solution. The best solution obtained by rounding Formulation 2.1 is $x_1=1$, $x_2=0$, $x_3=1$, $x_4=1$ with total net present value of \$137.

Weingartner's work has been adapted and extended in various ways. Other constraints like capacity, liquidity, and manpower have been added, and different single objective functions have been proposed, e.g., minimize total annualized cost or maximize total rate of return. However, to meet the complexity of realistic capital budgeting problems, other criteria for optimization must be included simultaneously to obtain multiple objective capital budgeting problems.

2.2 Multiple Criteria Decision Making Model

The capital budgeting problem that motivates this thesis is a multiple criteria decision making problem. Multiple criteria decision making (MCDM) is concerned with the methods and procedures by which multiple criteria can be formally incorporated into the analytical process. MCDM has two distinct areas: multiattribute decision analysis and multiple objective mathematical programming. Multiple objective mathematical programming is most often applied to deterministic problems in which the number of feasible alternatives is large. Multiattribute decision analysis is most often applicable to problems with a small number of alternatives in an environment of uncertainty.

Section 2.2.1 examines multiple objective mathematical programming. Section 2.2.2 presents the Analytic Hierarchy Process as a methodology for dealing with multiattribute decision analysis.

2.2.1 Multiple Objective Mathematical Programming

Methods for solving single objective mathematical programming problems for capital budgeting (such as Weingartner's formulation described above) have been studied extensively for the past 30 years. However, decision-makers have found that complex real-world problems involve more than one objective, each of which represents a criterion that is important to the company (Steuer 1986). For example, consider the following capital budgeting problem with five objectives:

(1) max { net present value }

(2) min {capital investment }

(3) min { annual operating expenses }

(4) max { investment in projects related to environmental protection }

(5) max { investment in projects in a given geographical area }

A traditional method of solving the problem with k objectives is first to assess the decision-maker's utility function U and then formulated and solve the linear programming problem:

$$\max \{ U(z_1, z_2, \ldots, z_k) \}$$

$$\text{with} \quad c_i x = z_i \quad 1 \leq i \leq k$$

subject to $\mathbf{x} \in S$

where

k is the number of objectives.

$\mathbf{c}_i$ is the vector of size n of linear objective function coefficients for the *i*th objective.

$\mathbf{x}$ is the decision variable vector of size n which defines a point in decision space.

z_i is the criterion value, the objective function value of the *i*th objective.

S is the feasible region defined by a set of linear constraints.

The main difficulty with this approach is how to define the decision-maker's utility function U. As noted in the example from Weingartner the linear weights may have a significant impact on the nature of the derived solution. Without explicit knowledge of the decision-maker's utility function the problem can still be solved by using a different formulation and implicit information from the decision-maker. For example, a multi-objective linear programming can be defined as:

$$\begin{aligned}
&\max \{ \mathbf{c}_1\mathbf{x} = z_1 \} \\
&\max \{ \mathbf{c}_2\mathbf{x} = z_2 \} \\
&\qquad\quad \vdots \\
&\max \{ \mathbf{c}_k\mathbf{x} = z_k \} \\
&\text{s.t.} \quad \mathbf{x} \in S
\end{aligned}$$

A frequently used method in multiple objective linear programming is the point estimate weighted sums approach where each objective $\mathbf{c}_i\mathbf{x}$ is multiplied by a strictly positive scalar λ_i. Then, the k weighted objectives are summed to form a composite single objective function. the composite objective function is written as $\boldsymbol{\lambda}^{\tau}\,\mathbf{C}\mathbf{x}$, where $\mathbf{C}$ is the $k \times n$ criterion matrix whose rows are the $\mathbf{c}_i$. Each weighting vector $\boldsymbol{\lambda} \in \mathbf{R}^k$ is normalized so that its elements remain non-negative and sum to one. Define Λ, the set of all such weighting vectors by:

$$\Lambda = \{ \lambda \in R^k \mid \lambda_i \geq 0, \sum_{i=1}^{k} \lambda_i = 1 \}$$

Therefore the problem reduces to:

$$\max \{ \lambda^T Cx \mid x \in S \}$$

The point estimate weighted sums method converts a multiple objective linear programming into a single objective linear programming that can be solved using standard linear programming algorithms. However, the difficulty lies in finding a "good" weighting vector. The following is an example with 3 objectives and 4 variables (Farguhar 1977).

Example 2.1

$$\begin{aligned}
&(1)\ \max\ 3x_1 + x_2 + 2x_3 + x_4 = z_1 \\
&(2)\ \max\ x_1 - x_2 + 2x_3 + 4x_4 = z_2 \\
&(3)\ \max\ x_1 + 5x_2 + x_3 + 2x_4 = z_3 \\
&\text{subject to:} \\
&\qquad 2x_1 + x_2 + 4x_3 + 3x_4 \leq 60 \\
&\qquad 3x_1 + 4x_2 + x_3 + 2x_4 \leq 60 \\
&\qquad\qquad\qquad\qquad\qquad x_j \geq 0
\end{aligned}$$

Suppose that z_1 is twice as important as z_2 and that z_3 is equally important to z_1. Thus, choose $\lambda = (0.4, 0.2, 0.4)$. The composite objective function is:

$$\max\ 1.8x_1 + 2.2x_2 + 1.6x_3 + 2x_4$$

The single objective mathematical programming is solved and the optimum solution is:

$$x_1 = 0, \quad x_2 = 6, \quad x_3 = 0, \quad x_4 = 18$$

giving

$$z_1 = 24, \quad z_2 = 66, \quad z_3 = 66$$

This system is sensitive to "small" change in the weighting vector λ. For example, if we choose $\lambda = (0.5, 0.1, 0.4)$, the composite objective function is:

$$\max \ 2x_1 + 2.4x_2 + 1.6x_3 + 1.5x_4$$

The optimum solution is:

$$x_1 = 0, \quad x_2 = 12, \quad x_3 = 12, \quad x_4 = 0$$

giving

$$z_1 = 36, \quad z_2 = 12, \quad z_3 = 72$$

Thus the importance of determining "appropriate" weights in the capital budgeting problem is noted.

2.2.2 Multiattribute Decision Analysis

Multiattribute decision analysis addresses problems where there are several criteria applicable over a number of decision alternatives to be selected. Each of the criteria is given a weight based on its importance relative to the problem at hand. For example, consider the choice of a new car where the color criterion is twice as important as gas mileage criterion. Multiattribute decision analysis has been applied successfully to difficult public policy problem such as nuclear power plant site selection, location of airports and types of drug rehabilitation programs (Saaty 1980).

The analytic hierarchy process (AHP), developed by Thomas L. Saaty, is designed to support the solution of complex problems involving multiple criteria. It is a comprehensive framework which is designed to cope with the intuitive, the rational, and the irrational when we make multiobjective, multicriterion and multiactor decisions with and without certainty for any number of alternatives. It is a method for deriving ratio scales used to integrate our procedure for representing the elements of any problem. It organizes the basic rationality by breaking down a problem into its smaller constituent parts and then calls for only simple pairwise comparison judgments to develop priorities in each hierarchy (Harker and Vargas 1987).

There are three principles which one can recognize in problem solving: (1) Decomposition, (2) Comparative judgments, and (3) Synthesis of priorities. The decomposition principle calls for structuring the hierarchy to capture the basic elements of the problem. The principle of comparative judgment calls for setting up a matrix to carry out pairwise comparisons of the relative importance of the elements in a level with respect to the elements in the level immediately above it. This matrix is used to generate a ratio scale. Finally, the principle of synthesis of priorities is used to generate the global or composite priority of the elements at lowest level of the hierarchy. The output of the AHP is the prioritized ranking of the decision options indicating the overall preference for each of the decision alternatives.

In addition, the AHP contains an intrinsic measure of inconsistency for each matrix and for the whole hierarchy. Knowledge of inconsistency enables one to determine those judgments that need reassessment. In this sense, the AHP deviates radically from the more traditional decision analytic methods in that the latter have no formal way of dealing with inconsistency.

To illustrate AHP, consider the problem to purchase a new car (Anderson 1992). After a preliminary analysis of the makers and models available, three cars are being considered in the list of decision alternatives. These are cars A, B, and C. Table 2.2 provides a summary of the information has been collected regarding these cars.

Criteria	Car A	Car B	Car C
Price	$13,100	$11,200	$9,500
MPG	18	23	29
Interior	Deluxe	Above average	Standard
Body	4-door midsize	2-door sport	2-door compact
Radio	AM/FM	AM/FM	AM
Engine	6-cylinder	4-cylinder turbo	4-cylinder

Table 2.2 Information for the Car-Selection Example

There are several criteria that the decision maker needs to consider in making the purchase decision. In this example, purchase price, miles per gallon (MPG), comfort, and style are selected as four criteria to be considered. Quantitative data regarding the purchase price and MPG criteria are provided in Table 2.2. However, measures of comfort and style cannot be specified so easily. Decision makers need to consider factors such as car interior, type of radio, easy of entry and exit, seat-adjustment feature, etc., in order to determine the comfort level for each car. The style criterion will need to be measured in terms of the decision maker's subjective evaluation of each car.

Even when dealing with a criterion as easily measured as purchase price, however, subjectivity becomes an issue whenever a particular decision maker indicates personal preferences. For instance, car A cost $3600 more than car C; this difference might represent a great deal of money to one person, but not very much money to another person. Thus, whether car A is considered extremely more expensive than car C or only moderately more expensive than car C is a subjective judgment that will depend primarily on the financial status of the person making the comparison. An advantage of the AHP is that it is designed to handle the subjective judgments of individuals as an important part of the decision process.

The first step in AHP is to develop a graphical representation of the problem in terms of the overall goal, the criteria, and the decision alternatives. Figure 2.1 depicts the hierarchy for the for the car -selection problem. Note that the first level of the hierarchy shows that the overall goal is to select the best car. At the second level, four criteria (purchase price, MPG, comfort, and style) will contribute to the achievement of the overall goal.

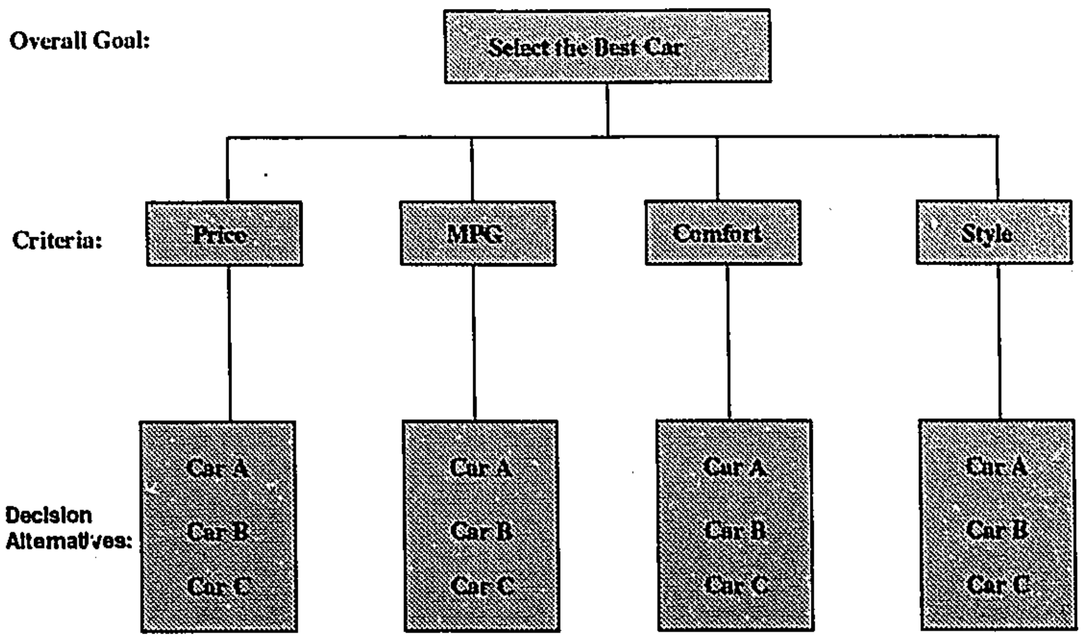


Figure 2.1 Hierarchy for the Car-selection Problem

The priorities that need to be determined in the car-selection problem are follows:

1. The priorities of the four criteria in terms of the overall goal
2. The priorities of the three cars in terms of the purchase-price criterion
3. The priorities of the three cars in terms of the MPG criterion
4. The priorities of the three cars in terms of the comfort criterion
5. The priorities of the three cars in terms of the style criterion

Pairwise comparisons of alternatives criteria measures are the fundamental building blocks of the AHP. In establishing the priorities for the three cars in terms of comfort, car A, car B, car C will be compared in pairs using a scale of values from 1 to 9 to rate relative preferences. Table 2.3 provides the numerical ratings recommended for the verbal preferences expressed by the decision maker. According to Saaty (1980), research and experience have confirmed the nine-unit scale as a reasonable basis for discriminating between the preferences for two items.

Value	Definition
1	Equal importance
3	Moderate importance
5	Strong importance
7	Demonstrated importance
9	Extreme importance
2,4,6,8	Intermediate values between two adjacent judgments
Reciprocals of above	If an item has one of the above number(e.g. 3) compared with a second item, then the second item has the reciprocal value (e.g. 1/3) when compared to the first

Table 2.3 Numerical Ratings of AHP

The values a_{ij} for the pairwise comparison of alternative i versus j for a particular criteria may be represented as a matrix such as the pairwise comparison matrix of the car for the comfort criteria measure:

Comfort	Car A	Car B	Car C
Car A	1	2	8
Car B	1/2	1	6
Car C	1/8	1/6	1

Table 2.4 Pairwise Comparison Matrix for 'Comfort'

We note that $a_{ij} = 1 / a_{ji}$ and $a_{ii} = 1$ i.e., the matrix has reciprocal properties.

Once the matrix of pairwise comparisons has been developed, priority of the elements being compared can be calculated by computing the eigenvector of the matrix. First sum the values in each column of the pairwise comparison matrix. Then divide each element in the pairwise comparison matrix by its column total; the resulting matrix is referred to as the normalized pairwise comparison matrix. Finally compute the average of the elements in each row of the normalized matrix; these averages provide an estimate of the relative priorities of the elements being compared. This procedure is called synthesizing judgments. Using this procedure, the relative priorities of car A, car B, and car C with respect to the comfort criterion is written as follows:

$$[\,0.593 \quad 0.341 \quad 0.066\,]$$

An important consideration in terms of the quality of the ultimate decision related to consistency of judgment that decision maker demonstrated during the series of pairwise comparisons. The AHP provides a measure of the consistency of pairwise comparison judgments by computing a consistency ratio. This ratio is designed in such a way that values of the ratio exceeding 0.1 are indicative of inconsistent judgments; in such case the decision maker would probably want to reconsider and revise the original values in the pairwise comparison matrix. Value of the consistency ratio of 0.1 or less are considered to indicate a reasonable level of consistency in the pairwise comparison.

The consistency ratio (CR), which is defined as follows:

$$CR = \frac{CI}{RI}$$

$$CI = \frac{\lambda_{max} - n}{n - 1}$$

where

n= the number of items being compared

where λ_{max} is the largest (positive) eigenvalue , and RI, the random index, is the consistency index of a randomly generated pairwise comparison matrix. RI takes on constant values that increase as n increases. For the car-selection example of Table 2.4 n=3 , RI=0.58. The consistency ratio for this example is 0.017.

In same way, as for the comfort criterion, pairwise comparison matrix for purchase Price, MPG, and Style for the car-selection example can be obtained. The set of relative priority vectors are:

Criteria	Car A Car B Car C
Price	[0.123 0.320 0.557]
MPG	[0.087 0.274 0.639]
Comfort	[0.593 0.341 0.066]
Style	[0.265 0.655 0.080]

Table 2.5 Criteria Priority Weights for Car

Finally, pairwise comparison matrix for the four criteria in car-selection problem:

Criteria	Price	MPG	Comfort	Style
Price	1	3	2	2
MPG	1/3	1	1/4	1/4
Comfort	1/2	4	1	1/2
Style	1/2	4	2	1

Table 2.6 Pairwise Comparison Matrix for the Main Criteria

Yields priorities for the four criteria are obtained by calculating the eigenvector of the Table 2.6 matrix. The results are:

Price	0.398
MPG	0.085
Comfort	0.218
Style	0.299

Table 2.7 Priority Weight for the Criteria

The resulting overall priority ranking is obtained for each car by combining the criteria weights (Table 2.7) with the corresponding car criteria weights (Table 2.5) using simple vector multiplication. The car priority rankings are given in Table 2.8 bellow.

Alternative	Priority
Car A	0.265
Car B	0.421
Car C	0.314
Total	1.000

Table 2.8 Alternative Weights

These results provide a basis to support the decision regarding the purchase of a car B.

Pissarides (1992) uses AHP in a capital budgeting Decision Support System — Optimal resource allocator (DSS ORA) for small scale (less than 15 projects) problem. AHP was used to set unknown criteria weights as a starting point in a mechanized multiattribute linear programming formulation. Sensitivity analysis allowed decision makers to explore the solution space in a flexible mouse driven windows environment. It has been used to evaluate capital allocation alternatives at the departmental organization level for a large Canadian telecommunications firm.

2.2.3 Incomplete Hierarchy Structure

In Figure 2.2, the hierarchy consists of three levels, the objective being the overall welfare of a nation, the first level a set of possible future scenarios of that nation, the second level the province of that nation, and the third are transport projects which are to be implemented in the province. Note that not every province affects each scenario nor does each project affect every province. The hierarchy in Figure 2.2 is not a complete one. The object is to determine the priorities of the projects as they impact on the overall objective. Here one must weight the priorities of each comparison set by the ratio of the number of elements in that set to the total number of elements in the third level. The incomplete hierarchy structure is important for the practical problem since real world problem solving usually has many alternatives to be evaluated. When the number of alternatives become large, it is more difficult to maintain pairwise comparison response consistency. If the incomplete hierarchy structure is constructed, the number of compare could be decreased significantly.

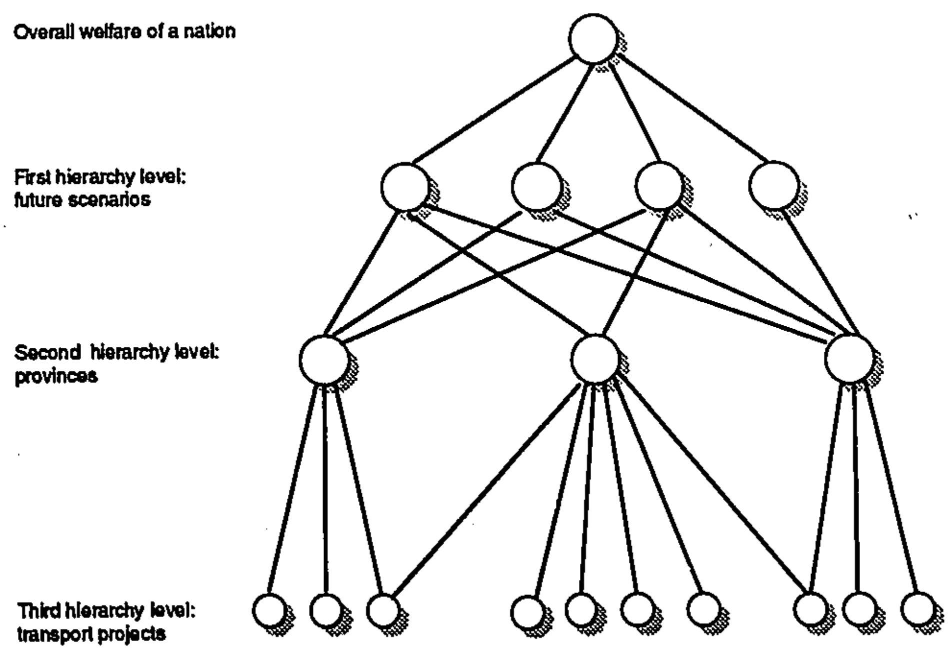


Figure 2.2 A Hierarchy for Priorities of Transport Projects in National Planning (Saaty 1980)

2.3 Aggregation and Disaggregation Techniques

Direct optimization of practical applications of the capital budgeting problem is generally difficult due to the large number of capital projects. A fundamental issue in the use of optimization models is the tradeoff between the level of detail and the ease of using and solving the model. Aggregation and disaggregation techniques have been proven to be valuable tools for manipulating data and determining the appropriate policies to employ for this tradeoff.

Furthermore, aggregation and disaggregation techniques offer promise for solving large-scale optimization models. Promising methodologies for studying the underlying structure of both univariate and multivariate data sets are provide along with a set of tools for manipulating data for different levels of decision makers (Rogers, Plante, Wong and Evans 1990). The process of using aggregation /disaggregation methods to solve larger problem is presented in Figure 2.3.

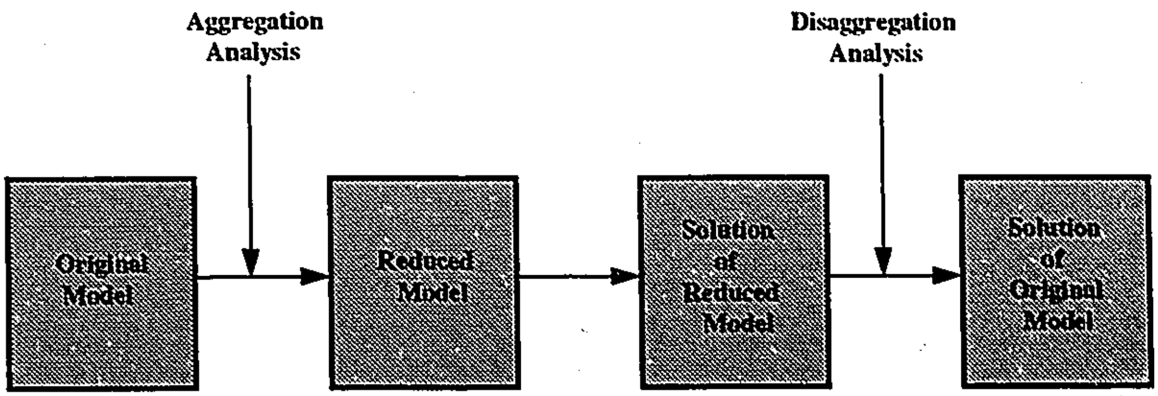


Figure 2.3 The Aggregation/Disaggregation Process

The problem is how to aggregate individual elements of the original large model and the degree of aggregation. "The problem of aggregation is very difficult, and no definitive treatment of the subject has yet been given ... The method of aggregation can have an important bearing on the answers obtained..." (Hadley 1971). The common criteria used to identify activities to be aggregated together are substitutability, complementarity, and similarity of alternatives. In practice, it is usually determined informally and/or intuitively when the formulation is developed according to the given problem. Raine, Flavell and Salkin (1978) developed a model for the annual budgeting problem of a light engineering company. The aggregation process employed in the data aggregation stage is essentially concerned with defining what the linear programming (LP) model will regard as a 'product'. Two kinds of 'product' appear in the LP model:

1. The most important products in the range appear explicitly in the LP as a model 'product'.The criterion for importance was the products forecasted total net sales value in the coming year.
2. The remaining products are represented by a number of aggregate products. The attributes of the aggregate product are calculated as a weighted average of the attributes of the original products which the aggregate product represents.

2.3.1 Aggregation and Disaggregation in Hierarchy Structure

The hierarchy structure breaks things down into large groupings or clusters and then decomposes each of these into smaller clusters and so on. The object is to obtain the priorities of all the elements together by means of clustering. This is by far more efficient process than treating all the elements individually to determine the priorities of each (Saaty 1980).

The usual way to deal with a larger number of elements which fall in a level of a hierarchy is to group them into clusters according to their relative importance. Thus, one would have one cluster of the most important (most similar, or closest) elements, another of those of moderate importance, and another of those of low importance. Then one compares in pairwise fashion the relative impact of the cluster on the relevant criteria of the next upper level. The clustering may differ from criterion to criterion. After this analysis of clusters, the elements in each cluster are then compared pairwise according to their relative importance in that cluster. If there are too many, again they may be subdivided into clusters. In this way each element, since it belongs to several clusters, would receive several weights from the different clusters. These weights can be used to evaluate the overall weight of the individual elements.

The example of distances between cities can be structured into hierarchy for treatment in AHP (Saaty 1980). Instead of comparing all these cities directly, the cities are grouped into clusters according to their falling in nearly equivalent distances from Philadelphia. Three classes are compared in the following matrix.

Philadelphia	Chicago Montreal	London San Francisco	Cairo Tokyo	Eigenvector
Chicago Montreal	1	1/7	1/9	0.056
London San Francisco	7	1	1/4	0.26
Cairo Tokyo	9	4	1	0.68

Table 2.9 Group Comparison Matrix

Now the cities in each cluster are compared separately according to their relative distance from Philadelphia.

Philadelphia	Chicago	Montreal	Eigenvector
Chicago	1	2	0.67
Montreal	1/2	1	0.33

Philadelphia	Cairo	Tokyo	Eigenvector
Cairo	1	1/1.5	0.4
Tokyo	1.5	1	0.6

Philadelphia	San Francisco	London	Eigenvector
San Francisco	1	1/1.3	0.43
London	1.3	1	0.57

Table 2.10 Cities Comparison Matrix in Groups 1-3

Now multiply the first eigenvector by 0.056, the second by 0.26 and the third by 0.68 to obtain the overall relative distance vector of each city from Philadelphia.

	Cairo	Tokyo	Chicago	San Francisco	London	Montreal
Result	0.27	0.41	0.037	0.11	0.15	0.019
Actual result	0.278	0.361	0.032	0.132	0.177	0.019

Table 2.11 Overall Weight of all Cities

It is interesting to note that in Table 2.11, the results obtained from hierarchy aggregation/disaggregation are very close to the actual distance vector relative measures.

The solution methodology that is presented in this thesis for the large scale capital budgeting problem uses aggregation/disaggregation techniques within an incomplete hierarchy structure. The AHP model is used on the defined hierarchy to determine priority weights from user responses to pairwise comparison of clusters of projects and projects. The following section defines the hierarchy structure methodology for the capital budgeting problem.

Chapter 3

Methodology

This chapter describes the procedure to solve the large scale capital budgeting problem. It is based on aggregation/disaggregation methods. The hierarchy structures and weighting methods are defined.

The proposed methodology for the allocation of capital to projects proceeds as follows:

(1) construction of a database of information on all individual projects;

(2) description of hierarchy of decision problem based on database information arranged according to company project categories and criteria;

(3) assignment of weights to company criteria using pairwise comparison with AHP and project aggregation according to the defined database structure and problem hierarchies;

(4) solution of mathematical programming problem for aggregated allocation using linear programming;

(5) allocation of capital to groups of projects and continue applying steps (3) and (4) for suballocation in problem hierarchy (disaggregation process).

The details of the methodology are presented below.

3.1 Database Construction

A database management system (DBMS) consists of a collection of interrelated data and a set of programs to access that data. The collection of data, usually referred to as the database, contains information about one particular enterprise. The primary goal of a DBMS is to provide an environment that is both convenient and efficient to use in retrieving, storing, and using database information.

A comprehensive database system is developed in this thesis. It contains information about the company's individual capital projects including company department membership, capital type and categories and user defined capital project groupings, e.g., see Figure 3.1. Database information is based on company project data for the fiscal year 1990 - 1991. These data are contained, managed, and organized within the database system prior to analyzing alternative capital allocation options. The function of the database is to facilitate aggregation of many individual projects into logical groups for analyses using AHP and LP methods. Aggregate suballocations are then broken down in a series of steps in the database manager for reallocation down to the individual projects level. Alternative aggregation options in the database allow decision makers to explore a variety of allocation constructs and results. The database is developed using the Paradox for Windows (V1.0) relational database system with object-based programming language ObjectPAL. Aspects of the database are described in more detail in Chapter 4 - Application.

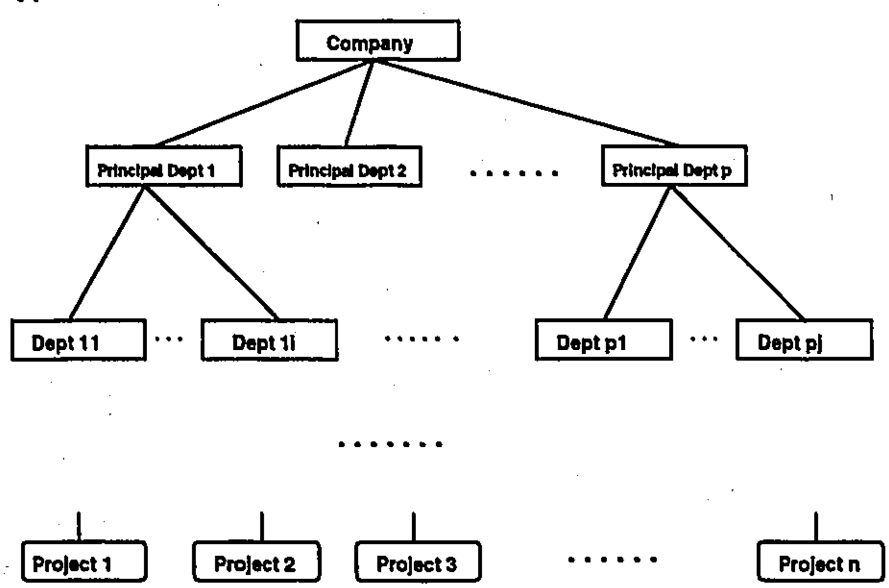


Figure 3.1 Company Structure

3.2 Problem Hierarchy Procedures

A hierarchy is a particular description of the capital projects system, which is based on the assumption that lower level entities, e.g. individual project (lowest level entities), can be grouped into disjoint sets, with the entities of one level influencing the entities of only one other level, and being influenced by the entities of only one other level. The entities in each level of the hierarchy are assumed to be independent. Figure 3.2 illustrates a multi-level problem hierarchy for the capital budgeting problem of the company.

The following subsection presents two alternative procedures for defining problem hierarchy: (1) a multi-level aggregation/disaggregation procedure, and (2) a direct allocation procedure.

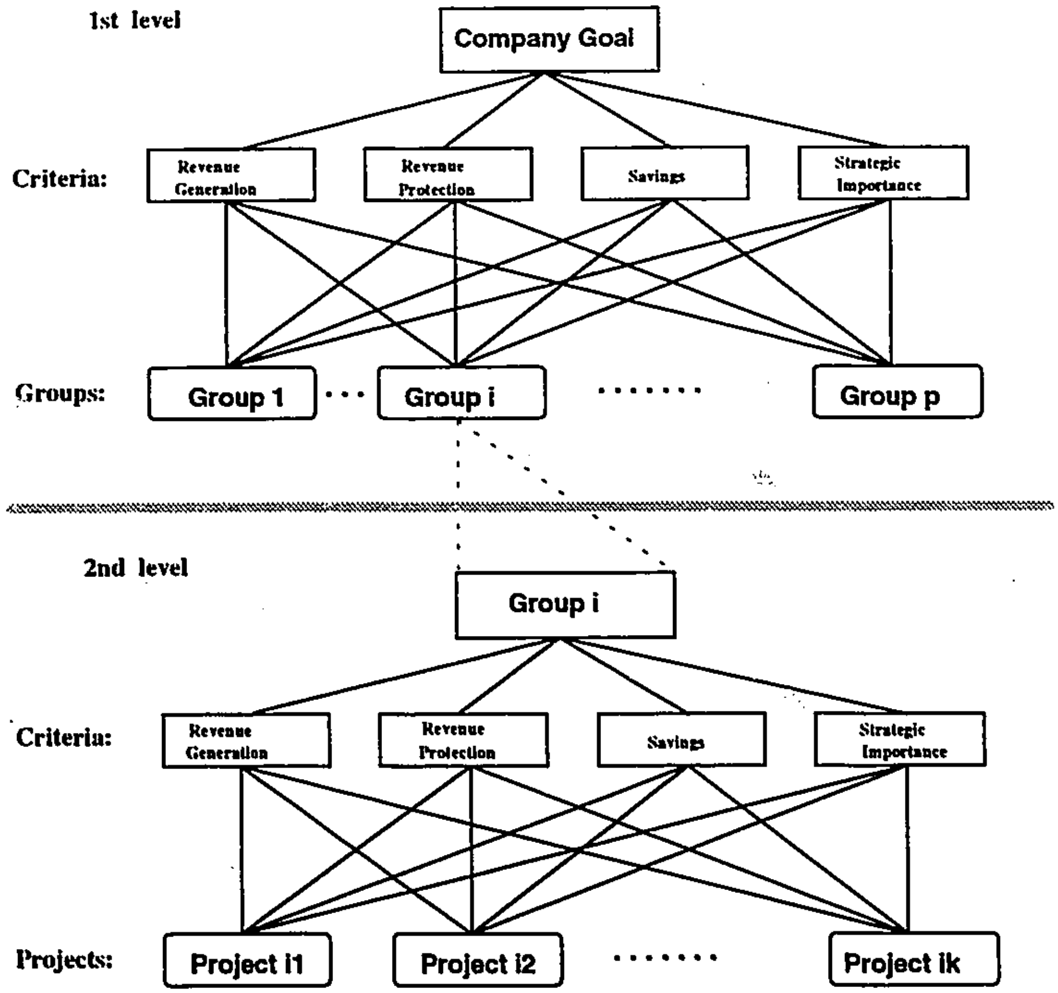


Figure 3.2

Multi-level Problem Hierarchy Structure

3.2.1 Multi-level Problem Hierarchies

Using this method, funds are allocated to groups from the highest level (company level) to the lowest level (project level) in a sequential step by step manner.

Because the funds are initially allocated to groups, the grouping procedure must consider the relationship across different groups. In this thesis, joint groups of projects are constructed. In the second level, relationship within groups can be described by constraints of the mathematical programming program.

Information in the project database provides possibilities for the grouping of projects according to: (1) the department code of the project; (2) the classification of the project into various capital types and categories used by the company; and (3) the user-defined project groups.

Figure 3.2 shows the "first level" as the user-defined group hierarchy. Each group i includes a number of individual capital projects. Using the AHP pairwise comparison, we can obtain the relative weight of each group in terms of the criteria that reflect the company's stated mission which are described by: (1) Revenue generation, (2) Revenue protection, (3) Savings, (4) Strategic importance. The "second level" of the hierarchy explodes the grouped project cluster. Using pairwise comparison as in AHP, we can obtain the weights of the projects within each group relative to the company's mission criteria.

Once the weight is determined for each element of the problem hierarchy, allocations can be made for the aggregate sets at each level through the mathematical programming system. The multi-level hierarchy allocation procedure is most effective in resolving a capital budgeting problem that has a well-defined hierarchy structure with independent groups at each level and little project relationships and dependencies defined across groups. The following section describes an alternative procedure that is better able to consider problem that have extensive across group dependencies.

3.2.2 Direct Project Weighting Scheme

An alternative method to determine project weights is achieved by dividing the projects into user defined groups. Let X_{ij} denote the funds allocated to the jth project in the ith group. Suppose we have N projects divided into p groups $\{X_{11}, X_{12}, \cdots X_{1k_1}\}, \cdots, \{X_{p1}, X_{p2}, \cdots X_{pk_p}\}$ where $k_1 + k_2 + \cdots + k_p = N$. These p groups could be dependent. These dependency across groups can be defined by constraints affecting projects from different groups.

Using AHP within each group, obtain the weights for each project by pairwise comparison. They are $\{W_{11}, W_{12}, \cdots, W_{1k_1}\}, \cdots, \{W_{p1}, W_{p2}, \cdots, W_{pk_p}\}$. Also using AHP on group 1, group 2, ... group p, obtain weights for each group. Denote these group weights as $W_1, W_2, \ldots W_p$. Then the overall weight of jth project in group i, $W_{P_{ij}}$ can be obtained in following way:

$$W_{P_{ij}} = W_{ij} \times W_i \times k_i \qquad j=1,2,\cdots, k_i \; , i=1,2,\cdots, p$$

k_i is used for normalization. These $W_{P_{ij}}$ now can be used as individual weights of each project in a mathematical programming problem that includes all individually weighted projects together as a single large scale formulation.

Direct allocation can be useful in solving the capital budgeting problem when there are significant dependencies and relationships (e.g., supporting functions or allocation constraints) among projects across groups. In this case, the multi-level hierarchies method cannot directly incorporate such dependencies because it is very difficult to group projects into independent groups. This is a "cost" of the aggregation/disaggregation approach. In contrast, the additional cost of assigning weights to each project in the direct weighting method permits the explicit inclusion of across group project dependencies. The impact of this issue for the application problem in this thesis is presented in Chapter 4.

3.3 Capital Allocation

3.3.1 Project Types

In this thesis, all projects are assumed to belong to one of four types: stand alone, supported, supporting and interdependent. These can be expressed graphically as below:

Type 1: Stand alone projects

Figure 3.3 Stand Alone Projects

Projects P1 and P2 are independent projects. These projects may be members of the same groups or different groups. They require no supporting projects (see below) and contribute uniquely to the constraining factors, e.g., depreciation and personnel limits according to relevant unique technological coefficients.

Type 2 , 3: Supported and supporting projects

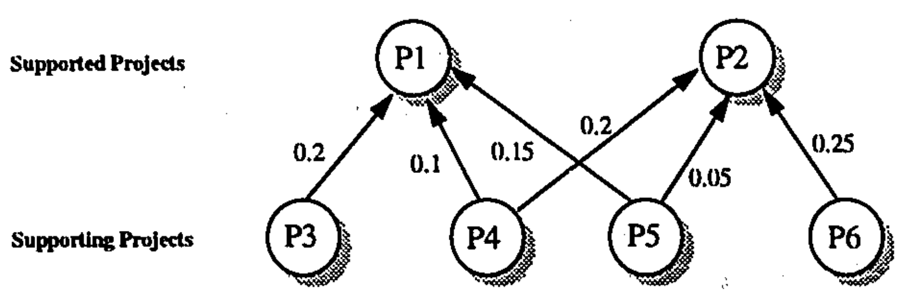


Figure 3.4 Supported and Supporting Projects

In this example projects P1 and P2 must be supported with capital funds allocated to projects P3 through P6. Projects P3 through P5 support P1 and P4 through P6 support P2. The required allocation to the supporting projects is determined by the allocation to the supported project and the fraction indicated. If P1 were to be allocated 100 capital budget units then the minimum allocation to supporting projects P3 through P5 would be 100*0.2=20, 100*0.1=10 and 100*0.15=15 respectively. Note that P5 supports both P1 and P2. If P2 were allocated 200 capital budget units P5 must be max {100*0.15, 200*0.05}=150. It is assumed that P5 could simultaneously support both P1 and P2. As well, the stated upper and lower bounds on supporting projects (as recorded in the database) must be such that required allocations for support are feasible.

Type 4: Interdependent projects

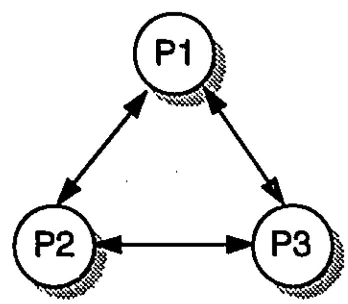


Figure 3.5 Interdependent Projects

In this case, project P1 through P3 are mutually dependent. This subgroup of individual projects can be treated as a single "megaproject" with corresponding total upper and lower bounds and combined constraint factors e.g., depreciation and personnel limits defined. If P1 is only allocated a fraction of its total capital requirement, then the scale of the dependent projects within the megaproject are directly adjusted accordingly.

Within the hierarchy structure, for type 1 stand alone projects, we can compare these projects directly with other projects (stand alone projects, supported projects and megaprojects) using AHP. After obtaining weights for these projects, funds can be allocated accordingly from the combined results of the mathematical programming problem.

For type 2 and 3, supported projects are compared with other projects (stand alone projects, supported projects, megaprojects) using AHP. After obtaining weights, funds can be allocated to these projects after solving the mathematical program. Supporting projects are not compared to other projects. They are considered as constraints according to their relation to the project they are supporting. As a result of the supported project allocation, funds are automatically assigned to the supporting projects.

For type 4, we treat all projects together as a megaproject. These megaprojects can then be compared with other projects (stand alone projects, supported projects, other megaprojects) using AHP. After obtaining weights, funds can be allocated to these megaprojects and accordingly to the individual member projects within each megaproject.

3.3.2 Multi-level Problem Hierarchy Method

After obtaining the weight of each group from the problem hierarchy, funds are allocated to the groups using linear programming as in Weingartner (1963) and Pissarides (1992). Then within each group, funds can be suballocated down to the level of individual projects in the same way.

The multi-level hierarchy approach assumes that independent project groups can be defined. In fact, project dependencies across groups may exist for defining problem constraints. Such dependency constraints are not explicitly accounted for in this aggregation approach. The direct project weighting method described in 3.3.2 above is an alternative weight scheme that can incorporate cross-group dependency constraints directly.

The method used to solve the multi-level hierarchy problem procedes by allocating the capital budget among the levels of the hierarchy as follows:

(a) First Level Allocation

The algorithm for allocating funds to the first level groups is a 6 step procedure as follows:

(1) Divide all projects into the highest functional groups, e.g., Departments. Using AHP on these groups, obtain weights of each group.

(2) If projects defined in different group are independent, then all groups are independent. Allocate funds to these groups using linear programming. Then go to the next level allocation. If projects in different groups are dependent, then construct new joint groups. These joint groups contain projects which are dependent on each other. The result is a newly defined set of independent project groups. For example, consider the following set of 15 capital projects from three organizational groups, e.g., departments.

e.g.

Original groups:

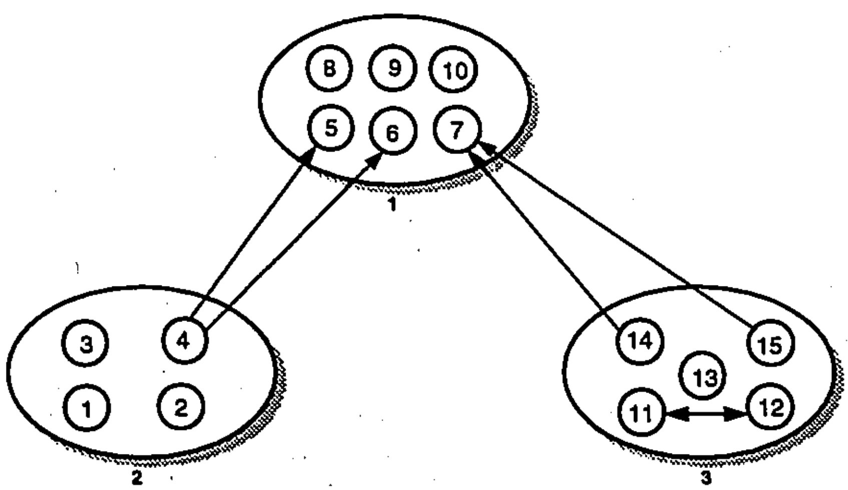


Figure 3.6 Original Groups

Constructed groups:

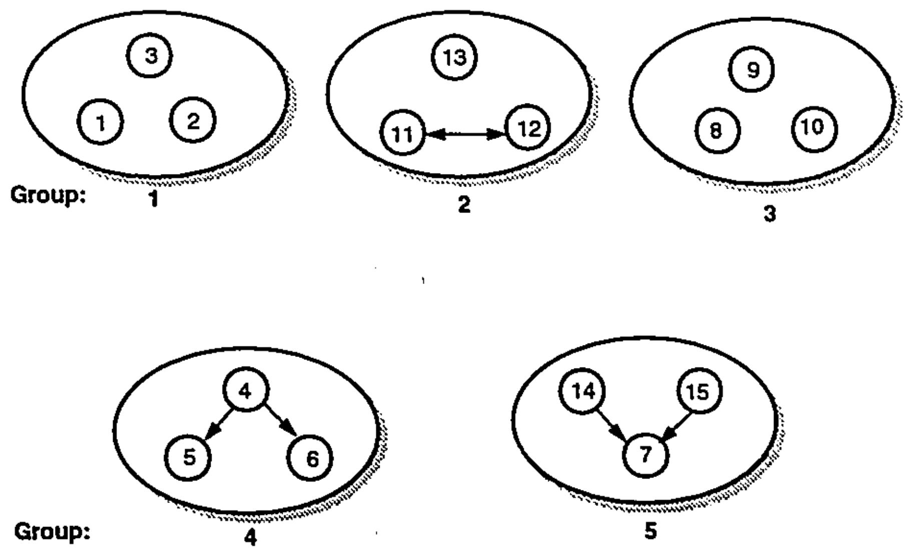


Figure 3.7 Constructed Groups

(3) Two methods to evaluate the weights of the new groups are as follows:

a. Weights for the 5 newly constructed groups may be obtained by pairwise comparison in AHP relative to the first level company criteria (Figure 3.2) to yield parameters W_i, $i=1,2, \ldots, 5$.

b. Alternatively, the weights of the 3 original groups are obtained by AHP pairwise comparison across these groups and relative to the first level company criteria. The weights of the new groups are calculated from the weights of the original groups:

$$W_1 = W_{D1}, \quad W_2 = W_{D2}, \quad W_3 = W_{D3}$$

$$W_4 = \frac{U_4 \times W_{D1} + U_5 \times W_{D3} + D_6 \times W_{D3}}{U_4 + U_5 + U_6}$$

$$W_5 = \frac{U_7 \times W_{D3} + U_{14} \times W_{D2} + U_{15} \times W_{D2}}{U_7 + U_{14} + U_{15}}$$

W_i: Weight of group i from AHP pairwise comparisons

U_i: Capital requirement (upper bound) of project i

(4) Evaluate the depreciation index (i.e., for the depreciation maximum constraint technological coefficients) of the groups:

$$D_i = \frac{\sum_{j=1}^{k_i} d_{ij} \times U_{ij}}{\sum_{j=1}^{k_i} U_{ij}} \qquad i=1,2,\cdots, p$$

D_i : Depreciation index of group i

d_{ij} : Depreciation index of jth project in group i (given)

U_{ij} : Upper bound of jth project in group i

k_i : number of projects in group i

(5) Evaluate manpower requirements or "equivalent employee" (EQE) demands for each of the groups. The EQE index (i.e., for the EQE maximum constraint technological coefficients) of the groups is:

a. EQE index of project j in group i:

$$e_{ij} = \frac{EQE_{ij}}{U_{ij}}$$

e_{ij}: EQE index of jth project in group i

EQE_{ij} : EQE requirement of jth project in group i (given)

U_{ij} : Upper bound of jth project in group i

b. EQE index of the groups:

$$E_i = \frac{\sum_{j=1}^{k_i} e_{ij} \times U_{ij}}{\sum_{j=1}^{k_i} U_{ij}} = \frac{\sum_{j=1}^{k_i} EQE_{ij}}{\sum_{j=1}^{k_i} U_{ij}}$$

E_i : EQE index of group i

(6) Linear programming formulation — the following represents the form of the general LP model for the allocation of capital to groups:

Let X_i denote capital funds allocated to group i

Objective function: $\max \sum_{i=1}^{p} W_i X_i$

Subject to:

(1) Budget constraint: $\sum_{i=1}^{p} X_i \le$ Total budget

(2) Depreciation constraint: $\sum_{i=1}^{p} D_i X_i \le$ Total depreciation

(3) EQE (manpower) constraint: $\sum_{i=1}^{p} E_i X_i \le$ Total EQE

(4) Bounds constraints: $\sum_{j=1}^{k_i} L_{ij} \le X_i \le \sum_{j=1}^{k_i} U_{ij}$ i=1,2,...,p.

Solving the above linear program determines funds allocated to all projects within each group.

(b) Second Level Allocation

Once the first level allocations have been made the group funds are suballocated to the project level. Within group i, all projects can be classified into one of four mutually exclusive types, namely:

(1) Megaproject
(2) Stand alone project
(3) Supported project
(4) Supporting project

Let j=1,...,l_{i_1} denote the index for each megaproject in group i. As previously defined, megaprojects contain more than one individual project. The index j denotes the megaproject and represents all member projects as a whole. At this allocation level, capital funds are assigned to the megaprojects. All allocations to individual member projects for each megaproject are made at the third (final) level of allocation also described below.

Let $j = l_{i_1} + 1,\ldots,l_{i_2}$ denote the index for each stand alone project in group i.

Let $j = l_{i_2} + 1,\ldots,l_{i_3}$ denote the index for each supported project in group i.

Let $j = l_{i_3} + 1,\ldots,l_{i_4}$ denote the index for each supporting project in group i.

Let X_{ij} denote the capital funds allocated to project j in group i, i=1,2,...,p; j=1,2,...,l_{i_4}. Then the mathematical linear programming formulation for second level allocation to project i is:

Objective function:

$$\max \sum_{j=1}^{l_{i_3}} W_{ij} X_{ij}$$

where W_{ij} are weights obtained from pairwise comparison across j=1,2,....,l_{i_1} in AHP.

Subject to:

(1) Budget constraint:

$$\sum_{j=1}^{l_{i_4}} X_{ij} \leq X_i \qquad \text{(from first level group allocation)}$$

(2) Depreciation constraint:

$$\sum_{j=1}^{l_{i_4}} d_{ij} X_{ij} \leq \text{Depreciation i}$$

(3) EQE constraint:

$$\sum_{j=1}^{l_{i_4}} e_{ij} X_{ij} \leq \text{EQE i}$$

(4) Bounds constraints:

$$L_{ij} \leq X_{ij} \leq U_{ij}$$

(5) Supporting project feasibility constraints:

For all project pairs with supported project j=s and supporting project j=t and given support fraction $r_{ist} \geq 0$ then

$$X_{is} \geq r_{ist} X_{it}$$

Solving the second level allocation problem results in capital funds assigned to each of the four project types. At this level all individual project allocations are made to Stand alone, Supported and supporting projects. It remains to make final allocations to individual project members of the megaprojects within each group.

(6) Megaproject allocations:

For group i, let m_{ij} denote the number of individual projects in megaproject j. $j = 1,2,\ldots,l_{i}$. Total allocation to the megaproject is $X_{ij} \le \sum_{n=1}^{m_{ij}} U_{ijn}$, the total maximum allocation allowable from the individual projects upper bounds. The allocation of capital to individual projects is determined as a function of the bounds U_{ijn}, L_{ijn} n=1,2,...,mj as follows:

$$X_{ijn} = L_{ijn} + f \times \frac{R_{ijn}}{\sum_{n=1}^{m_j} R_{ijn}}$$

where $f = X_{ij} - \sum_{n=1}^{m_j} L_{ijn}$, the funds remaining for the megaproject after all lower bounds have been assigned; and $R_{ijn} = U_{ijn} - L_{ijn}$, the range of bounds for each individual project.

This completes the allocation methodology for the multi-level hierarchy approach.

3.3.3 Direct Project Weighting Method

For the alternative weighting scheme, the weights for the projects are obtained using the incomplete hierarchy structure in AHP as described in section 3.2.2 The constraint structure defined in two ways may be used to allocate capital to the projects:

(1) Mathematical programming — include all projects, constraints (including dependency constraints that define restrictions on projects across grouping), bounds on individual project allocations, etc., in one large linear programming problem formulation;

(2) Determine an efficient aggregation scheme for the mathematical programming problem defined in (1), taking into account the dependency constraints, etc., assuming such aggregations may exist, e.g., independent megaprojects.

For each of these approaches allocation to individual projects occurs as the solution to the linear programming formulation presented below:

Let X_i denote funds allocated to project i, i=1,2,...,N

Objective function:

max $\sum_{i=1}^{N} W_i X_i$ $\quad$ W_i : weight of project i determined from AHP incomplete hierarchy approach (section 3.2.2); i=1,2,...,N

Subject to:

(1) Budget constraint: $\sum_{i=1}^{N} X_i \leq$ Total funds available

(2) Total depreciation constraints: $\sum_{i=1}^{N} d_i X_i \leq$ Total depreciation $\quad$ d_i : depreciation rate of project i

(3) Total EQE constraints: $\sum_{i=1}^{N} e_i X_i \leq$ Total EQE $\quad$ e_i : EQE index of project i

(4) Bounds constraints: $L_i \leq X_i \leq U_i$ $\quad$ i= 1,2,...,N $\quad$ L_i : lower bound of project i

U_i: upper bound of project i

(5) Dependency constraints:

1.Supported and supporting projects:

$X_i \geq r_{ij} X_j$

X_i: Fund allocated to the supporting project.

X_j: Fund allocated to the supported project.

r_{ij}: Supporting fraction.

2. Megaproject:

$$X_{M,j} = \sum_{j=1}^{k_i} L_{M,j} + \left(\sum_{j=1}^{k_i} X_{M,j} - \sum_{j=1}^{k_i} L_{M,j}\right) \times \frac{U_{M,j} - L_{M,j}}{\sum_{j=1}^{k_i} (U_{M,j} - L_{M,j})}$$

$X_{M,j}$: Fund allocated to the jth project in ith megaproject.

$L_{M,j}$: Lower bound of the jth project in ith megaproject.

$U_{M,j}$: Upper bound of the jth project in ith megaproject

k_i: Number of projects in ith megaproject.

For any set $J_D \subset \{1,2,...,103\}$, define the following classes of dependency constraints, where membership in J_D across group is unrestricted.

3. EQE constraints:

$$\sum_{j \in J_D} e_j X_j \le E(J_D) \quad \text{or} \quad \sum_{j \in J_D} e_j X_j \ge E(J_D)$$

4. Funds availability constraints:

$$\sum_{j \in J_D} X_j \le F_j(J_D) \quad \text{or} \quad \sum_{j \in J_D} X_j \ge F_j(J_D)$$

5. Depreciation constraints:

$$\sum_{j \in J_D} d_j X_j \le D(J_D) \quad \text{or} \quad \sum_{j \in J_D} d_j X_j \ge D(J_D)$$

The two basic methodologies described here are applied to data from a large Canadian telecommunication company. The allocation results for (1) the multi-level hierarchies method, and (2) the direct weighting approach are determined and compared in the following chapter.

Chapter 4

Application

This chapter describes the development of the database system and the problem solving procedure for the case of the annual capital budgeting problem for the large Canadian telecommunication company.

4.1 Database Construction

Data for 103 projects representative of the company's capital project alternatives for 1990-1991 are used to construct the database. The database presented in the thesis is based on actual company data that has been recoded and changed to protect the proprietary nature of this information. As well data responses (such as AHP weights) are derived and used for illustration purposes of the methodology only. ObjectPAL of Paradox for Windows V 1.0 is used to develop a menu-driven graphic user interface, automatic aggregation, disaggregation, and input/output report facilities.

4.1.1 PARADOX for Windows

Paradox for Windows (1992) is a relational database management system developed by Borland International. It is used to store, sort, and produce reports on data for a wide range of applications. In addition to standard database reporting capabilities, Paradox for Windows also includes integrated business graphics so users may generate bar graphs, pie charts, and other graphics automatically from data defined in the active database.

Paradox for Windows allows user control over how information is entered, stored, and manipulated. This program uses an interface similar to Lotus 1-2-3 and other spreadsheets for interactive manipulation of data, complemented by a programming language for more complex procedures. Paradox for Windows also takes advantage of the ease of use offered by Windows applications and the inherent support for graphics and communication among Windows software packages.

Paradox for Windows has adopted object-oriented programming technology. Object-oriented programming can condense hundreds of lines of program code into dozens and includes functions impossible to perform in low-level programming languages. ObjectPAL is the integrated programming language for Paradox for Windows. ObjectPAL is a high-level, event-driven, object-based, visual programming language. It can be used to create completely customized applications, one with entirely new buttons, menus, dialog boxes, prompts, warnings, and help. In this application ObjectPAL is used to develop specialized reports and user interface to the capital projects database system.

4.1.2 Table Structure Design

Data are stored in tables within Paradox. In this application five tables are created. These are:

1. Project data table: description of the basic information of the project record. (Appendix A.1)
2. Economic indicator table: description of several economic indices of the project. (Appendix A.2)
3. Supporting and supported project table: description of the fraction of the supporting and supported project. (Appendix A.3)
4. Project code table: description of the project code of every project. (Appendix A.4)
5. Department code table: description of every department. (Appendix A.5)

The database structures of these tables are listed in Appendix A. Some fields of these tables are actual data of the company, e.g., Project name is the name of the actual project. Other fields are defined for system usage, e.g., Group name are used to define alternative hierarchy structures. Project code is an identifier of individual project (one to one relationship). Projects may have different economic indicator data in different years. Project codes are used as the key link to all tables containing project data.

4.1.3 Form Design

A Paradox form is a special screen that is used to access one or more tables for entering, viewing, or editing data. In this thesis, an edit form was developed. It is used to input data into the database and edit data already entered in database. Figure 4.1 presents the input/edit screen for a single project (project number 28).

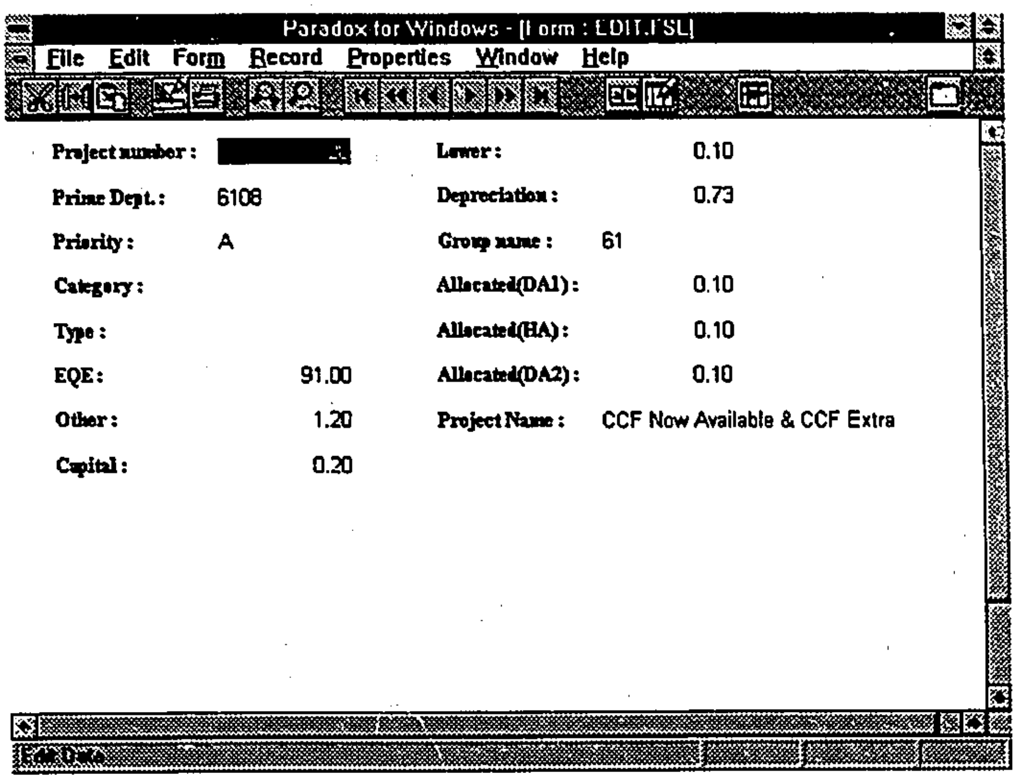


Figure 4.1 Edit Form

4.1.4 Query Design

The Paradox Query allows for information retrieval from the database according to conditions defined by the user. The Answer tables of the Query results are used to prepare required information for capital budgeting allocation. e.g., project groups and clusters data. It can also be used to convert data to other software by using the export function in Paradox.

4.1.5 Report Design

Reports are used to retrieve information in the database. A report may contain any combination of fields and draw on information from one or several tables as described above. In this application, several reports were developed. Complete report for the application are presented in the Appendices. The reports are:

1. Project Code Data: list of all project codes. (see Appendix B.1)
2. Department Code Data: list of all department codes. (see Appendix B.2)
3. Group Code Data: list of all group codes. (see Appendix B.3)
4. Project Data: list of selected data fields of all projects. (see Appendix B.4)
5. Supporting and Supported Project Data: list of all data for supporting and supported projects definitions. (see Appendix B.6)
6. Allocation Results: list of the results of the three different allocations for the capital budgeting problem. (see Appendix D.1, D.2, D.3)

4.1.6 User Interface Design

ObjectPAL allows for customizing access and use of the Paradox database. For this application a series of pull-down menus were developed to facilitate use of the database. The menu system is designed to show the specific functions of : (1) Edit Forms; (2) Query database for group information; and (3) Report - the required data from tables. Figure 4.2 displays the main menu of the capital allocation database. The command bar menu allows the user to move within the database (as in Figure 4.1 described previously) or to carry out other tasks such as generating reports ("Report function") or grouping data into user defined results ("Query function").

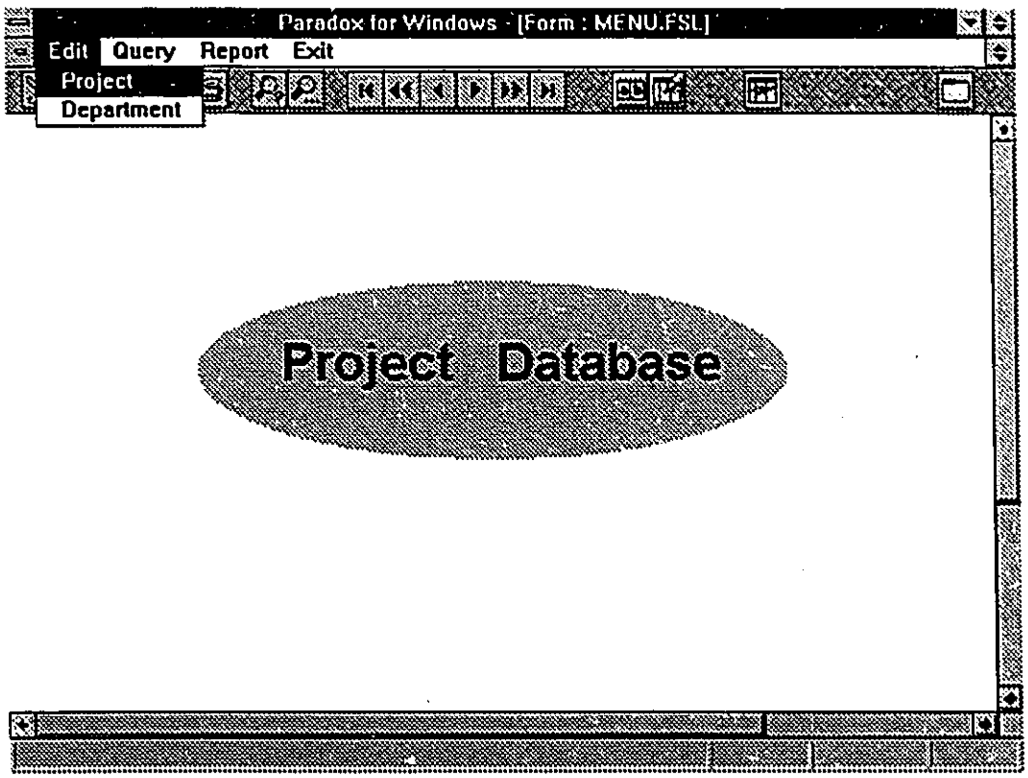


Figure 4.2 Main Menu

4.2 Hierarchy Allocation

4.2.1 Problem Hierarchy Definition

The first step in the hierarchy allocation approach is to define hierarchies for problem structure based on the database information. In this application, first, all projects are grouped according to the principal department to which they belong. Next new independent groups are constructed using the methodology described in Chapter 3. Ten principal departments were defined in this application. Twelve independent groups were defined after regrouping according to the projects's relationship across the groups. The project linkages by group for the 103 capital projects in the database are presented in Appendices B.3 to B.5. The total available funds were first allocated to these groups ("first level" allocation) then down to the individual project ("second level" allocation).

4.2.2 Weight Evaluation

(a) First level weight evaluation

Using AHP on the 12 groups, the following weights were obtained through the pairwise comparison exercise relative to the company goals (see also Figure 3.2).

Group	28	41	49	4951	51	52	61	6163	63	64	80	90
Weight	0.1067	0.1241	0.0645	0.0645	0.0645	0.0943	0.0554	0.0744	0.0711	0.0893	0.0951	0.0959

Table 4.1 Weights of Level 1 Groups

(b) Second level weight evaluation

Within each group, pairwise compare megaprojects, supported projects, and independent projects, and determine the weights of these projects for use as objective function coefficients in the mathematical programming formulation for each of these group allocation problems.

4.2.3 Linear Programming Results

(a) First level linear programming problem

The weights of each group are used to construct the objective function. The EQE index of the group, and the Depreciation rate of the group are calculated from individual projects in the group as described in Chapter 3. Total funds available, the EQE limit value, and the Depreciation limit value define the right-hand side resource constraints of the problem.

In this application, Borland's spreadsheet system, Quattro Pro 5.0 is used to solve the LP formulation for the first level allocation. It consists of 12 variables, denoting funds to be allocated to 12 different groups; 4 constraints sets: (1) total funds available, (2) EQE limits, (3) depreciation limits, and (4) upper/lower allocation bounds). For each capital group, these data are presented below.

(a) Resource availabilities

Available Funds	EQE Limit	Depreciation Limit
450	1700	55

(b) Grouped Data

Group	Weight	EQE	Depreciation	Capital requirement	Lower bound
28	0.1067	12	0.55	0.2	0.1
41	0.1241	18	0.61	0.6	0.3
49	0.0645	694	0.2	61.4	40.2
4951	0.0645	208	0.35	11.2	6.5
51	0.0645	12	0.36	5.4	3.4
52	0.0943	124	0.77	0.4	0.2
61	0.0554	717	0.22	69.17	38.8
6163	0.0744	307	0.14	116	74.9
63	0.0711	279	0.08	334.3	207.1
64	0.0893	219	0.19	8.2	5
80	0.0951	8	0.85	0.2	0.1
90	0.0959	59	0.54	0.4	0.2

Table 4.2 First Level Problem Formulation Data

The results of the first level allocation are presented in Appendix C.1 and are summarized below in Table 4.3. Details of the mathematical formulation results from the QPRO models are presented in Appendix C.3. In summary, it is noted that constraints for total funding and EQE and depreciation limits were binding. Three groups were assigned intermediate allocations between the lower and upper ("capital requirement") bounds.

(c) Solution

Group	Allocation
28	0.1
41	0.6
49	40.2
4951	6.5
51	3.4
52	0.2
61	38.8
6163	78.50485
63	275.4433
64	5.9519
80	0.1
90	0.2
Total	**450**

Table 4.3 First Level Allocation To Groups

(b) Second level linear programming formulation

Once the first level allocation has occurred, the funds to be allocated to each group may be in one of the three cases: 1. Group upper bound; 2. Group lower bound; 3. Intermediate between lower and upper bound allocation. In case 1, all projects in the group automatically receive upper bound funding. In case 2, all projects in the group receive lower bound funding. In case 3, a second level allocation is required. In this application, after the first level allocation only Group 63 , Group 64, and Group 6163 require second level allocation.

For the Group 64, it is noted that it is comprised only of a single project, project # 8 (Appendix B.5). Thus, the intermediate group allocation is assigned directly to this project without need for further analysis. In second level allocation, megaprojects, supported projects, and independent projects are required to be weighted in the objective function. Supporting projects are considered as fractional constraints and have no weights assigned to them. QPRO 5.0 is again used to solve the LP problems at the second level (see Appendix C.2). Similarly the results of the second level allocations to projects may be upper, lower or intermediate allocations. For the case of Group 6163 (which required second level allocation analysis), results were as follows:

Project	Weight	EQE Index	Depreciation Rate	Capital requirement	Lower bound
Mega 1	0.2347	2.2	0.1161	80.8	55.2
54	0.3061	10.22	0.17	9.2	5
34	0.2551	7.97	0.21	7.4	4
32	0.2041	0.05	0.18	9.1	5
43	0	1.43	0.26	5.6	3
62	0	10.8	0.34	2.5	2
90	0	10	0.5	1	0.5
97	0	340	0.63	0.2	0.1
29	0	0	0.74	0.2	0.1

Table 4.4 Second Level Problem Formulation Data (Group 6163)

Allocation:

Project	Allocation
Mega 1	57.5244
54	5
34	4.748
32	5
43	3
62	2
90	0.95
97	0.1329
29	0.15
Total	**78.505**

Megaproject suballocation:

Project	Allocation
63	5.381
92	0.109
89	20.563
96	20.908
93	10.454
95	0.109
Total	**57.5244**

Table 4.5 Second Level Allocations to Group 6163 for Stand Alone, Supporting , Supported and Megaprojects.

For the Group 63, the results was as follows:

Project	Allocation
31	0.2
44	0.2
53	16
59	0.2
65	0.2
83	0.2
84	1.2
88	0.2
71	0.2
7	6.5
72	0.2
9	26.5
30	7.2
68	2.5
33	15
56	0.1
79	0.1
80	192.643
86	2
87	0.1
81	4
Total	**275.4433**

Table 4.6 Second Level Allocation to Group 63

4.3 Direct Allocation

In the direct allocation approach, weights of all projects are calculated through the hierarchy structure analysis as described in Chapter 3, Section 3.2.2. A LP formulation was constructed according to the weights of all projects and the several additional operational constraints. GAMS (the General Algebraic Modeling System) was used to solve the single LP problem with 103 variables and 275 total constraints (Brooke and Kendrick 1989).

4.3.1 Weight Evaluation

There are ten principal departments in this application (See Appendix B.2 — first two digits of department codes). Weights of these principal departments are obtained by using pairwise comparison relative to the four company criteria (Figure 3.2). Weights for the individual projects within each principal department are determined by pairwise comparison against the same company criteria. The overall weights of the individual project are obtained by multiplying principal department weight with individual project weight in the group and the number of projects in the group. As for the case of incomplete hierarchies (Saaty 1980), the weights are determined by the weight of the principal department the project belong, the weight of the project in the principal department which it belong, and the number of projects in that principal department. (See also Section 3.2.2)

4.3.2 Dependency Constraints

In the direct allocation approach, any allocation restriction across the groups can be represented by dependency constraints. These may including funding dependency constraints, EQE dependency constraints, depreciation dependency constraints and other arbitrary across groups constraints. Each dependency constraint contains more than one project. Moreover, the projects belong to more than one group. In the application, 5 additional dependency constraints are considered. They are:

1. Capital dependency constraint:

$$X_{78}^{(28)} + X_{27}^{(41)} + X_{12}^{(49)} + X_{38}^{(52)} + X_{4}^{(61)} + X_{33}^{(63)} + X_{8}^{(64)} + X_{1}^{(80)} + X_{54}^{(6163)} \geq 60$$

2. Capital dependency constraint:

$$X_{64}^{(90)} + X_{91}^{(90)} + X_{25}^{(51)} + X_{39}^{(51)} \geq 45$$

3. Capital dependency constraint:

$$X_{17}^{(49)} + X_{98}^{(61)} + X_{81}^{(63)} \geq 10$$

4. EQE dependency constraint:

$$30X_{71}^{(63)} + 1.5X_{7}^{(63)} + 1.45X_{72}^{(63)} + 0.34X_{9}^{(63)} + 4.8X_{68}^{(63)} + 50X_{22}^{(61)} + 1.45X_{26}^{(61)} + 130X_{45}^{(61)} \geq 90$$

5. EQE dependency constraint:

$$0.65X_{15}^{(41)} + 0.47X_{60}^{(41)} + 0.29X_{66}^{(49)} + 0.76X_{38}^{(52)} + 0.77X_{40}^{(52)} + 0.89X_{3}^{(61)} + 0.14X_{4}^{(61)} + 0.5X_{86}^{(63)} \geq 5$$

6. Depreciation dependency constraint:

$$0.26X_{43}^{(6163)} + 0.74X_{29}^{(6163)} + 0.96X_{11}^{(4951)} + 0.85X_{56}^{(63)} \geq 1.5$$

where $X_j^{(i)}$ denote the decision variables of funds allocated to project j in group i. In the EQE dependency constraints, the coefficient of the decision variable is the EQE index of that project (See Section 3.3.2 for calculation of the EQE indices). In the depreciation dependency constraints, the coefficient of the decision variable is the depreciation rate of that project. (See Appendix B.4 for project depreciation rates and EQE values.)

4.3.3 Linear Programming Problem and Results

Using these overall weights of all projects, the LP objective function can be constructed. Total funds available, total EQE available, total depreciation acceptable, supporting fractions, and dependency requirements constitute the constraints of the LP formulation using the methodology developed in Chapter 3. In this application, GAMS was used to solve the LP problem. The LP consists of 103 variables, 275 constraints for the problem without dependency constraints, 281 constraints for the problem with dependency constraints. (Appendix E provides a complete listing of the GAMS input and output for the analysis of the direct weighting problem with dependency constraints)

4.4 Comparison of Hierarchy and Direct Allocation

Table 4.7 presents the total funds allocated, total EQE, total depreciation, number of projects receiving upper bound funding, number of projects receiving lower bound funding and number of projects receiving intermediate funding allocations. For the hierarchy allocation procedure, the first level allocation depreciation rate and EQE index were calculated from the weighted average of the projects in the group. In the second level allocation the total EQE and the total depreciation therefore do not necessarily satisfy the constraints set in the first level. However, the constraint violation noted in Table 4.7 is not significant.

For the direct allocation with dependency constraints, not all funding was allocated because the dependency constraints used in this application were restrictive. After checking the results of the hierarchy allocation and the direct allocation without dependency constraints, we found that all six dependency constraints (see section 4.3.2) were violated. Moreover, all six constraints in the model which included these were binding. The slight difference between the results of direct allocation and the right hand side parameters (Funds $\leq$ 450, EQE $\leq$ 1700, and Depreciation $\leq$ 55) in some cases were attributed to round-off errors from manual transfer of the LP allocation to the Paradox database.

	Hierarchy Allocation	Direct Allocation without Dependencies	Direct Allocation with Dependencies
Total Funds Allocated	449.999	450.002	436.166
Total EQE	1755.673525	1701.100354	1700.930418
Total Depreciation	55.05117014	54.52419	55.00075
# Upper Bound Allocation	17	9	10
# Lower Bound Allocation	74	82	80
# Intermediate Allocation	12	12	13

Table 4.7 Comparison of Different Allocation Results

For large scale capital budgeting problems involving on the order of 100 separate projects, the direct allocation method would require well over 500 pairwise comparisons to be done in order to determine project level weights (assuming over six projects per group). This amount of detail and the increased risk of inconsistency from individual and group responses makes this approach less attractive for determining project weights for the linear program.

By comparison, the hierarchy allocation for a similarly sized problem in our experience requires less than 300 comparisons to determine project allocations. In the presence of dependency constraints, an efficient approach would use the initial hierarchy allocation to check for feasibility and capital reallocation where necessary to accommodate violated constraints.

Chapter 5

Conclusions and Extensions

5.1 Conclusions

This thesis addressed the capital budgeting problem of a large Canadian telecommunication company. This capital budgeting problem formulated here had following three characteristics:

1. Large number of alternatives: Since there are lot of projects need to be funded, it is very difficult to solve the problem directly at the project level. Aggregation/disaggregation techniques were used to reduce the model scale and to simplify the analysis and results.

2. Multiple Criteria: It is not possible to compare all projects directly. However, the importance of the projects needs to be evaluated using criteria that reflect the company's mission. Therefore, the decision model has to define these criteria to evaluate the contribution of each project to the company's mission.

3. Multiple Constraints: Resource restrictions requires that capital budgeting group consider a number of organizational and operational constraints in the allocation problem.

A database system was developed for organization of all problem data including aggregation/disaggregation option, multiple criteria measures, allocation limits and resource restriction.

AHP was used to obtain user defined objective function weights for the mathematical programming problem formulation. The multiple criteria problem aspect was considered through the adoption of multi-level problem hierarchy definition.

In this thesis, two methods are used to solve the large scale capital budgeting problem. The first method uses problem hierarchy allocation. The other is direct project allocation. Both methods make use of the information database and problem hierarchy structure for aggregation modelling. Hierarchy allocation is a functional, group oriented approach. It is effective for the capital allocation problem that has clear hierarchy structure and where there are not many interrelationships between different entities across the hierarchy. Projects can be grouped into independent groups and these groups are used as entities of a multi-level hierarchy structure. There are two steps in the grouping procedure. The first step is functional grouping (e.g. principal department) then secondly, regrouping these groups into smaller independent groups. Moreover, the pairwise comparison requirements are drastically reduced with this approach when a incomplete hierarchy structure is constructed. Allocation are assigned within each group down to the project level using a linear programming problem formulation.

Direct project allocation is a project oriented approach. It is effective when there are many important interrelationships among different projects and projects can't be grouped into smaller independent groups. All dependency restrictions are considered directly as constraints in large scale the linear programming allocation problem.

5.2 Extensions

Visual Basic or Visual C++ can be used to develop a completely integrated Decision Support System for the large scale capital budgeting problem. This could provide decision makers with a friendly user interface in the Windows environment and create automatic links among database, weights evaluation procedure, and linear programming. Object oriented programming techniques can be used in enhancements to system development (e.g. Microsoft Foundation Class in Visual C++).

The system can give decision makers more flexibility to define different hierarchy such as groups, supporting and supported project relationships, and more relationship between groups and projects using a set of well-defined rules. After all necessary information is provided, it could be used to generate automatically the hierarchy structure and the allocation results. In itself the database modelling approach identifies specific information needs and streamlines the use and effectiveness of data collected on a routine basis.

A set of standards for comparison of the results needs to be defined. The system can use these standards to compare alternative approaches (including sequential storing of results for repeated trials). More data information including other objectives (e.g., NPV : net present value, PWAC: present worth of annualized costs, etc.) should be clearly defined. Variable data elements in LP, RHS , constraint and objective function coefficients may be used as the basis to generate and run simulation trials for random input data and present (graphical) results of I/O distributions designed at exploring the sensitivity of capital allocations to changes in key input variables.

In the course of the research, it was noted that many data elements for each project were unavailable, or not consistently collected. The development of the database focuses the data needs toward more coordinated information use in this problem setting.

Actual testing and validation of alternative approaches for a particular capital budgeting problem would be a logical next step in this research area. Information about system performance selective to actual allocations would provide useful feedback toward successful implementation of the structured allocation methods and more effective and less costly capital budgeting decision making in future.

Bibliography

Anderson, D.R., Sweeney, D.J. and Williams, T.A. 1992. *Quantitative Methods for Business.* Fifth Edition. West Publishing Company

Balas, E. 1963. Solution of large-scale transportation problems through aggregation. *Operations Research.* 13, 82-93.

Bierrman, H. Jr. 1988. *Implementing Capital Budgeting Techniques.* Ballinger Publishing Company, Cambridge, Massachusetts.

Borland International, Inc. 1992. *Paradox for Windows.*

Borland International, Inc. 1993. *Quattro Pro for DOS 5.0.*

Bromwich, M. 1979. *The Economic of Capital Budgeting.* Pitman Publishing Limited.

Brooke, A., Kendrick, D., Meeraus, A. 1989. *GAMS: A User's Guide.* The Scientific Press.

Charnes, A., Cooper, W. and Miller, M. H. 1959. An application of linear programming to financial budgeting and cost funds. *The Journal of Business.* 20-46.

Dantzig, G. B. 1963. *Linear Programming and Extensions*, Princeton University Press, Princeton, New York.

Farguhar, P.H. 1977. A survey of multiattribute utility theory and application. *Studies in Management Science.* North Holland publications Amsterdam.

Forsyth, J.D. and Owen, D. O. 1981. Capital Rationing Methods. In Roy L. Crum, Frans G.J. Derkinderen. *Capital Budgeting Under Conditions of Uncertainty.* Martinus Nijhoff Publishing, pp. 213

Hadley, G. *Linear Programming* (Addison-Wesley Publication Co., Reading, MA, 1971) 194.

Hall, N.G., Hershey, J.C., Kessler, L.G. and Stotts, R.C. 1992. A model for making project funding decisions at the national cancer institute. *Operations Research.* 40(6), No. 6, November-December.

Harker, P.T. and Vargas, L.G. 1987. The Theory of Ratio Scale Estimation: Saaty's Analytic Hierarchy Process. *Management Science.* Vol 33. No.11 November 1987.

Kroenke, D. 1992. *Management Information Systems. Second Edition.* Mitchell McGRAW—HILL.

Peerenboom, J.P.,Buehring, W.A.,Joseph, T.W. 1989. Selecting a portfolio of environmental programs for a synthetic fuels facility. *Operations Research.* 37(5), Sep.-Oct.

Pissarides, S. 1992. *Interactive multiple criteria optimization for capital budgeting.* Master thesis. Systems Science Programme. University of Ottawa.

Rawat, S. 1992. *Decision Support System for Optimal Resource Allocation: Overview.* Bell Canada. Internal document.

Raine, P.S., Flavell, R.B. and Salkin, G.R. 1978. Determining appropriate levels of data aggregation in a linear programming model. *European Journal Operational Research.* 2,26-31.

Rogers, D.F., Plante, R.D., Wong, R.T. and Evans, J.R. 1991. Aggregation and disaggregation techniques and methodology in optimization. *Operations Research*. 39(4), July-August.

Saaty, T.L. 1980. *Multicriteria decision making-The Analytic hierarchy process*. McGraw-Hill, Inc.

Saaty, T.L. and Marinaro, R.S. 1979. Rationing Energy to Industries; Priorities and Input-Output Dependency, *Energy Systems and Policy*.

Salo, A. and Hamalainen, R.P. 1989. A Modelling and Decision Aid for Supporting Telecommunications Investments, *IEEE International Conference on the Systems, Man and Cybernetics,* Conference proceeding, 1, 115-18

Shrange, L. 1986. *Linear, Integer, and Quadratic Programming with LINDO*, The Scientific Press.

Shetty, C.M. and Taylor, R.W. 1987. Solving large-scale linear programs by aggregation. Comput. *Operations Research*. 14(5), 385-393.

Steuer, R.E. 1986. *Multiple Criteria Optimization: Theory, Computation, and Application.* John Willey & Sons Inc., New York.

Towndend, J. J., Lindsay, J., 1993. *Using Paradox for Windows.* QUE Corporation.

Weingartner, H. M. 1963. *Mathematical programming and analysis of the capital budgeting problems*. Prentice-Hall, Englewood Cliffs, New Jersey.

Appendix A

Database Structure

This appendix present the Paradox table structure for the capital budgeting problem. The appendix is composed of five elements. They are:

A.1 Project data table: description of the basic information of the project record.

A.2 Economic indicator table: description of several economic index of the project.

A.3 Supporting and supported project table: description of the fraction of the supporting and supported project.

A.4 Project code table: description of the project code of every project.

A.5 Department code table: description of every department.

A.1. Project data table:

	Name	Type	Size	Example
1	Project code	N	*	1
2	Stage	A	1	I
3	Department code	A	6	8004
4	Contact	A	20	R.E.Mizon
5	Priority	A	1	A
6	Category	A	1	2
7	Type	A	1	2
8	EQE (Employees)	N	*	10 person years
9	A/S	N	*	$0
10	Salary	N	*	$0
11	Other	N	*	$3.5
12	Capital requirement	N	*	$0
13	Lower bound	N	*	$2
14	Depreciation rate	N	*	20%
15	PWAC	N	*	—
16	NPV	N	*	—
17	IROR	N	*	—
18	Group code	A	6	Variable
19	Allocated fund	N	*	$1.5

All dollar values in millions of current dollars

Priority: A— High, B—Medium, C—Low **Stage:** D=Development, I=Implementation

Category:
1: XYX (1)
2: Major contribution improvement
3: Customer satisfaction / service provisioning
4: Network / Operations infrastructure
5: Benchmarking
6: Other

Type:
1: Development (1)
2: Operations
3: Benchmarkiing
4: Others

Size: * denotes free format

(1) Actual name are confidential

A.2. Economic indicator table:

	Name	Type	Size	Example
1	Project code	N	*	1
2	Expense 1	N	*	$0
3	Expense 2	N	*	$0
4	Expense 3	N	*	$0
5	Savings 1	N	*	$0
6	Savings 2	N	*	$0
7	Savings 3	N	*	$0
8	Revenue 1	N	*	$0
9	Revenue 2	N	*	$0
10	Revenue 3	N	*	$0
11	Payback period	N	*	2 years

A.3. Supporting and supported table:

	Name	Type	Size	Example
1	Supporting project code	N	*	1
2	Supported project code	N	*	2
3	Fraction	N	*	0.5

A.4. Project code table:

	Name	Type	Size	Example
1	Project code	N	*	1
2	Project name	A	60	416 Exhaust

A.5. Department code table:

	Name	Type	Size	Example
1	Department code	A	6	8004
2	Department name	A	60	Regional Performance OR

Appendix B

Data Values

This appendix present the data values of the capital budgeting problem. The appendix is composed of six elements. They are:

B.1 Project code data: list all projects code.

B.2 Department code data: list all department code.

B.3 Group code data: list all group code.

B.4 Project data: list some data of all project.

B.5 Project linkages by group: all the relation between projects are presented by graph.

B.6 Supporting and supported project data: list all data of supporting and supported project.

In order to keep sensitive information confidential, the following tables do not depicit actual project or department names.

B.1 Project Code Table

Project number	Name
1	P1
2	P2
3	P3
4	P4
5	P5
6	P6
7	P7
8	P8
9	P9
10	P10
11	P11
12	P12
13	P13
14	P14
15	P15
16	P16
17	P17
18	P18
19	P19
20	P20
21	P21
22	P22
23	P23
24	P24
25	P25
26	P26
27	P27
28	P28
29	P29
30	P30
31	P31
32	P32
33	P33
34	P34
35	P35
36	P36
37	P37

Project Code Table (continued)

Project number	Name
38	P38
39	P39
40	P40
41	P41
42	P42
43	P43
44	P44
45	P45
46	P46
47	P47
48	P48
49	P49
50	P50
51	P51
52	P52
53	P53
54	P54
55	P55
56	P56
57	P57
58	P58
59	P59
60	P60
61	P61
62	P62
63	P63
64	P64
65	P65
66	P66
67	P67
68	P68
69	P69
70	P70
71	P71
72	P72
73	P73
74	P74

Project Code Table (continued)

Project number	Name
75	P75
76	P76
77	P77
78	P78
79	P79
80	P80
81	P81
82	P82
83	P83
84	P84
85	P85
86	P86
87	P87
88	P88
89	P89
90	P90
91	P91
92	P92
93	P93
94	P94
95	P95
96	P96
97	P97
98	P98
99	P99
100	P100
101	P101
102	P102
103	P103

B.2 Department Code Table

Code	Dept name
2804	Special Assignment & R/ Matters
41001	Personnel Resources-1
4106	Personel Resources -2
4903	Corporate Performance -Operations
4904	Operations Planning & Capital Management
4906	Customer Service Operation/Client Services
4907	Cust.Serv.Opns.Plng./Access Netwk.Infopro & Serv. Agreement
5101	Acquisitions & Quality Assurance
5102	Logistics -1
5104	Logistics -2
5207	Systems & Methodes(I,C,M&M)
5208	Systems & Methods(Cust.BLG)
6102	Business Development / Marketing Management
6105	Business Development / Public Networks
6106	Business Development -1
6108	Business Development / Rates
6109	Business Development -2
6303	Standards Research
6304	Operations Planning Research
6305	Operation System & Operator Service
6308	Technology Research
6400	ABC Co.
8004	R / Performance -1
9003	R / Performance -2

B.3 Group Code Table

Group Name	Group Code
Reg. Matters	28
Personnel	41
Cust Serv/Corp. Ops	49
Operations and Quality Assurance	4951
Logistics	51
Methods	52
Bus. Development	61
Bus Development and Planning	6163
Planning	63
ABC Co.	64
R -1	80
R -2	90

B.4 Project Data (selected fields)

Project number	Prime Dept.	EQE	Other	Capital	Lower	Depreciation	Group name
1	8004	8	5	0.2	0.1	0.85	80
2	6105	19	0	0.2	0.1	0.86	61
3	6105	7	0	0.2	0.1	0.89	61
4	6105	77	11.5	14.8	10	0.14	61
5	6105	3	0	0.2	0.1	0.9	61
6	6108	5	0	0.2	0.1	0.92	61
7	6308	10	11.5	6.5	4	0.24	63
8	6400	219	15.6	8.2	5	0.19	64
9	6308	9	1.7	26.5	15	0.1	63
10	5101	14	0	0.2	0.1	0.94	4951
11	5101	14	0	0.2	0.1	0.96	4951
12	4906	42	6	27.8	20	0.08	49
13	6109	54	4	5.8	3	0.25	61
14	4907	69	2.3	2	1	0.38	4951
15	41001	3	0.4	0.2	0.1	0.65	41
16	4903	1	0	0.2	0.1	0.66	49
17	4903	5	1	1	0.5	0.44	49
18	4904	4	1.7	0.2	0.1	0.67	49
19	4903	8	0.6	1	0.5	0.43	49
20	4906	6	0.6	0.6	0.4	0.45	49
21	4907	2	0.16	0.2	0.1	0.68	4951
22	6105	10	0	0.2	0.1	0.69	61
23	6108	2	0	0.2	0.1	0.7	61
24	4906	11	0.8	10	6	0.15	49
25	5102	0	0	0.2	0.1	0.71	51
26	6105	29	0.8	2	1	0.37	61
27	4106	8	0.99	0.2	0.1	0.72	41
28	6108	91	1.2	0.2	0.1	0.73	61
29	6303	0	0	0.2	0.1	0.74	6163
30	6308	0	0	7.2	4	0.22	63
31	6304	4	0	0.2	0.1	0.75	63
32	6303	0.5	3.2	9.1	5	0.18	6163
33	6303	7	17.7	26.5	15	0.1	63
34	6109	59	9.4	7.4	4	0.21	6163

Project number	Prime Dept.	EQE	Other	Capital	Lower	Depreciation	Group name
35	6106	6	0.25	7.92	4	0.2	61
36	4906	18	0.1	1	0.5	0.42	49
37	6108	143	6.3	6.3	4	0.25	61
38	5207	0	0	0.2	0.1	0.76	52
39	5102	0	0	4.6	3	0.3	51
40	5208	124	0	0.2	0.1	0.77	52
41	4906	14	0	0.2	0.1	0.78	49
42	4906	10	0.9	2.3	1	0.35	49
43	6109	8	0	5.6	3	0.26	6163
44	6304	31	0.4	0.2	0.1	0.79	63
45	6105	26	2	0.2	0.1	0.8	61
46	5104	2	0.7	0.2	0.1	0.81	51
47	6105	32	1.4	9.3	5	0.16	61
48	5104	4	0.2	0.2	0.1	0.82	51
49	4906	30	0.9	2.2	1	0.36	49
50	6106	3	0	0.2	0.1	0.83	61
51	4906	7	0.8	0.4	0.3	0.45	49
52	5101	73	1.3	3	2	0.32	4951
53	6304	24	4	16	10	0.12	63
54	6109	94	6.3	9.2	5	0.17	6163
55	6105	55	2.4	0.2	0.1	0.84	61
56	6303	0	0	0.2	0.1	0.85	63
57	6106	5	0	0.2	0.1	0.45	61
58	6108	1	0	0.2	0.1	0.45	61
59	6304	36	1	0.2	0.1	0.46	63
60	41001	7	0.01	0.2	0.1	0.47	41
61	4906	87	1.5	3	2	0.31	49
62	6109	27	6.8	2.5	2	0.34	6163
63	6108	11	5.7	9.2	5	0.2	6163
64	9003	44	0	0.2	0.1	0.48	90
65	6304	33	0	0.2	0.1	0.49	63
66	4906	31	0	4.6	3	0.29	49
67	6102	5	0.1	0.15	0.1	0.98	61
68	6308	12	0	2.5	1	0.33	63
69	6106	31	0	10.8	5	0.15	61
70	6106	39	0	0.2	0.1	0.5	61
71	6305	6	0.7	0.2	0.1	0.5	63
72	6305	29	0	0.2	0.1	0.51	63
73	6105	16	0	0.2	0.1	0.52	61
74	4906	323	5.4	1.1	1	0.4	49

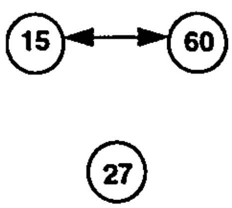
Project number	Prime Dept.	EQE	Other	Capital	Lower	Depreciation	Group name
75	5104	6	0	0.2	0.1	0.53	51
76	4907	23	1.9	5.2	3	0.27	4951
77	4907	9	0.4	0.2	0.1	0.54	4951
78	2804	12	9.9	0.2	0.1	0.55	28
79	6303	0	0	0.2	0.1	0.55	63
80	6303	1	19.6	235.6	150	0.05	63
81	6303	0	0	6.6	4	0.23	63
82	4906	27	0.9	0.2	0.1	0.56	49
83	6304	3	0	0.2	0.1	0.57	63
84	6304	2	0.1	1.2	1	0.39	63
85	4907	4	0	0.2	0.1	0.58	4951
86	6304	53	1.2	3.5	2	0.3	63
87	6304	8	0	0.2	0.1	0.59	63
88	6304	11	0	0.2	0.1	0.6	63
89	6303	1	0	26.2	20	0.11	6163
90	6109	10	0	1	0.5	0.41	6163
91	9003	15	0.1	0.2	0.1	0.6	90
92	6108	2	0	0.2	0.1	0.61	6163
93	6303	1	0	15	10	0.13	6163
94	6102	12	0.5	2.2	1	0.35	61
95	6108	25	2.2	0.2	0.1	0.62	6163
96	6303	1	0	30	20	0.07	6163
97	6109	68	0.5	0.2	0.1	0.63	6163
98	6106	6	1.75	6.5	4	0.24	61
99	4906	44	0.6	1	0.6	0.4	49
100	4906	26	2.7	4.6	3	0.28	49
101	6105	11	0	0.2	0.1	0.64	61
102	6105	21	0	0.2	0.1	0.64	61
103	6106	8	0	0.2	0.1	0.65	61

B.5 Project Linkages by Group

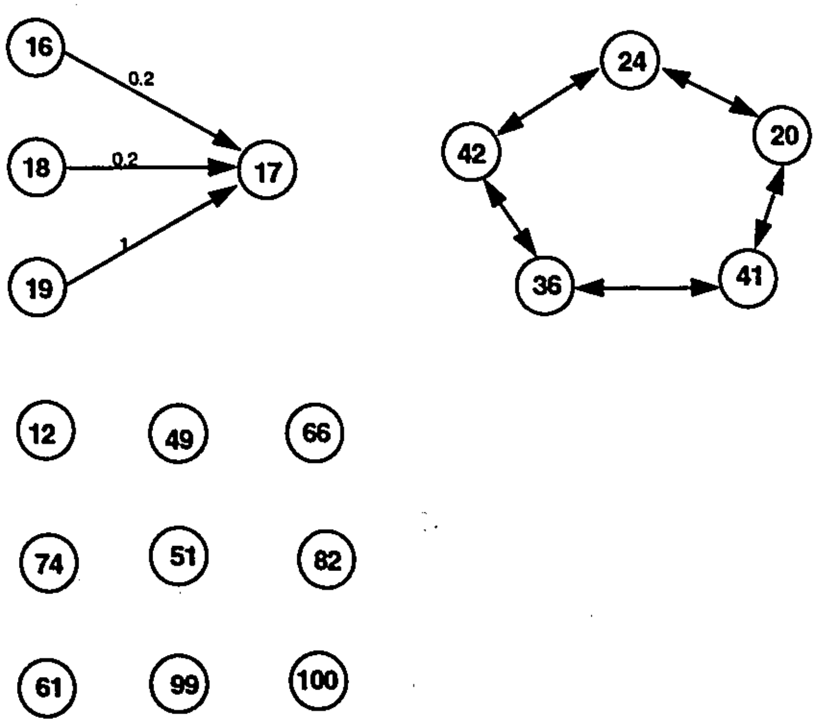
Numbers in nodes refer to project numbers (see appendix B.4). Arrows denote directional interproject dependency. Fraction on arcs denote lower limit allocation proportions of supported projects. Single nodes denote stand alone projects.

Group 28

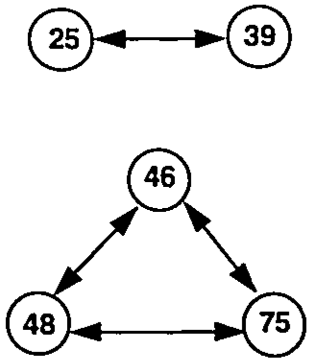
Group 41



Group 49

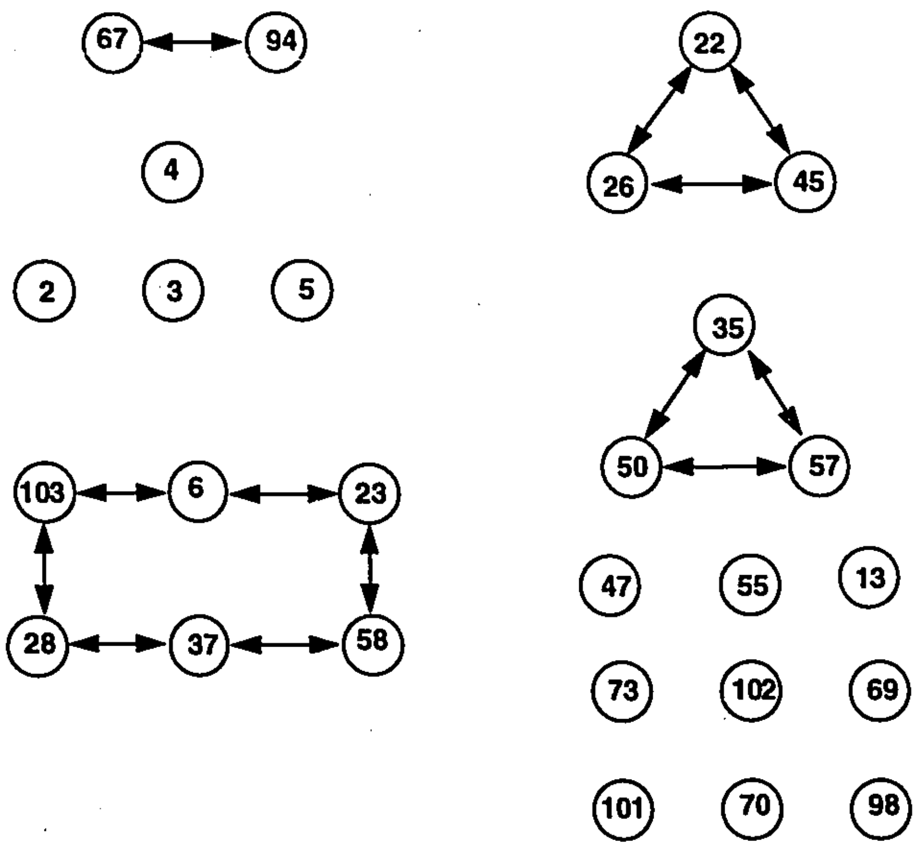


Group 51

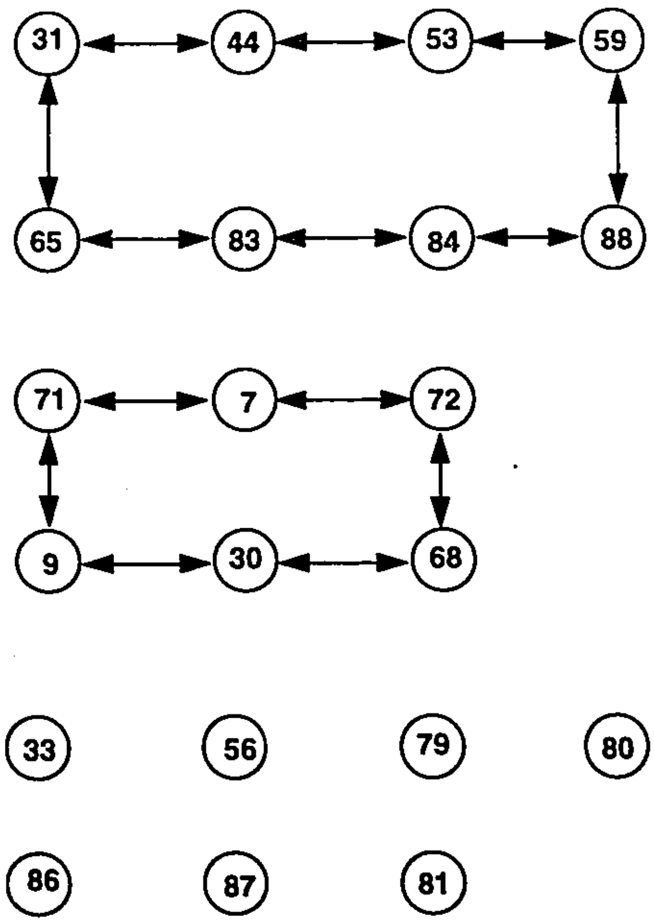


Group 52

Group 61



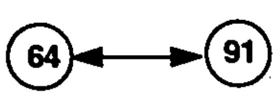
Group 63



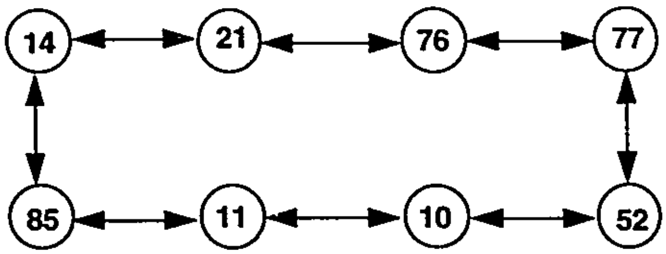
Group 64

Group 80

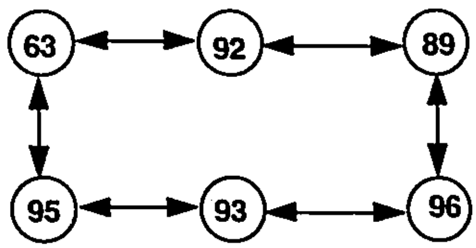
Group 90

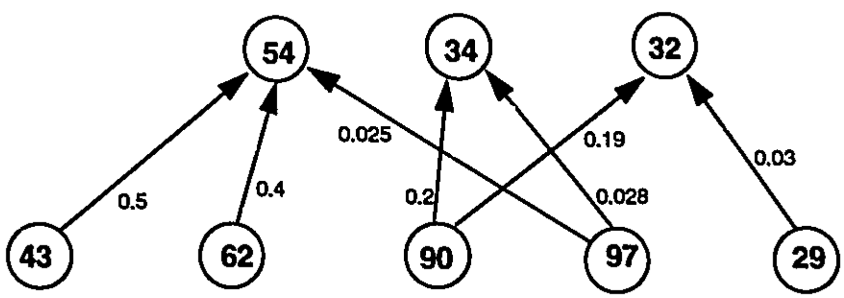


Group 49-51



Group 61-63





B.6 Supporting and Supported Project Data

The fraction value equal to 1 mean supported and supporting projects are in same megaproject.

Supported Project	Supporting Project	Fraction
6	23	1
6	28	1
6	37	1
6	58	1
6	103	1
7	9	1
7	30	1
7	68	1
7	71	1
7	72	1
9	30	1
9	68	1
9	71	1
9	72	1
10	11	1
10	14	1
10	21	1
10	52	1
10	76	1
10	77	1
10	85	1
11	14	1
11	21	1
11	52	1
11	76	1
11	77	1
11	85	1
14	21	1
14	52	1
14	76	1
14	77	1
14	85	1

Supported Project	Supporting Project	Fraction
15	60	1
16	17	0.2
18	17	0.2
19	17	0.1
20	24	1
20	36	1
20	41	1
20	42	1
21	52	1
21	76	1
21	77	1
21	85	1
22	26	1
22	45	1
23	28	1
23	37	1
23	58	1
23	103	1
24	36	1
24	41	1
24	42	1
25	39	1
26	45	1
28	37	1
28	58	1
28	103	1
29	32	0.03
30	68	1
30	71	1
30	72	1
31	44	1
31	53	1

Supported Project	Supporting Project	Fraction
31	59	1
31	65	1
31	83	1
31	84	1
31	88	1
35	50	1
35	57	1
36	41	1
36	42	1
37	58	1
37	103	1
41	42	1
43	54	0.5
44	53	1
44	59	1
44	65	1
44	83	1
44	84	1
44	88	1
46	48	1
46	75	1
48	75	1
50	57	1
52	76	1
52	77	1
52	85	1
53	59	1
53	65	1
53	83	1
53	84	1
53	88	1
58	103	1
59	65	1
59	83	1
59	84	1

Supported Project	Supporting Project	Fraction
59	88	1
62	54	0.4
63	89	1
63	92	1
63	93	1
63	95	1
63	96	1
64	91	1
65	83	1
65	84	1
65	88	1
67	94	1
68	71	1
68	72	1
71	72	1
76	77	1
76	85	1
77	85	1
83	84	1
83	88	1
84	88	1
89	92	1
89	93	1
89	95	1
89	96	1
90	32	0.19
90	34	0.2
92	93	1
92	95	1
92	96	1
93	95	1
93	96	1
95	96	1
97	34	0.028
97	54	0.025

Appendix C

Results of Hierarchy Allocation

This appendix present the results of the hierarchy allocation of the capital budgeting problem. The appendix is composed of two elements. They are:

C.1 Level one results: present the results of the group level.

C.2 Level two results: present the allocation results of individual project level.

C.1. Level 1

(a) Resource availabilities

Available Funds	EQE Limit	Depreciation Limit
450	1700	55

(b) Grouped Data

Group	Weight	EQE	Depreciation	Capital requirement	Lower bound
28	0.129	12	0.55	0.2	0.1
41	0.15	18	0.61	0.6	0.3
49	0.078	694	0.2	61.4	40.2
4951	0.078	208	0.35	11.2	6.5
51	0.078	12	0.36	5.4	3.4
52	0.114	124	0.77	0.4	0.2
61	0.067	717	0.22	69.17	38.8
6163	0.09	307	0.14	116	74.9
63	0.086	279	0.08	334.3	207.1
64	0.108	219	0.19	8.2	5
80	0.115	8	0.85	0.2	0.1
90	0.116	59	0.54	0.4	0.2

(c) Solution

Group	Allocation
28	0.1
41	0.6
49	40.2
4951	6.5
51	3.4
52	0.2
61	38.8
6163	78.50485
63	275.4433
64	5.9519
80	0.1
90	0.2
Total	450

C.2 Level 2

Group 6163

(a) Data

Project	Weight	EQE Index	Depreciation Rate	Capital requirement	Lower bound
Mega 1	0.23	2.2	0.1161	80.8	55.2
54	0.3	10.22	0.17	9.2	5
34	0.25	7.97	0.21	7.4	4
32	0.2	0.05	0.18	9.1	5
43	0	1.43	0.26	5.6	3
62	0	10.8	0.34	2.5	2
90	0	10	0.5	1	0.5
97	0	340	0.63	0.2	0.1
29	0	0	0.74	0.2	0.1

(b) Allocation

Project	Allocation
Mega 1	57.5244
54	5
34	4.748
32	5
43	3
62	2
90	0.95
97	0.1329
29	0.15
Total	78.505

(c) Megaproject suballocation

Project	Allocation
63	5.381
92	0.109
89	20.563
96	20.908
93	10.454
95	0.109
Total	57.5244

Group 63

Project	Allocation
31	0.2
44	0.2
53	16
59	0.2
65	0.2
83	0.2
84	1.2
88	0.2
71	0.2
7	6.5
72	0.2
9	26.5
30	7.2
68	2.5
33	15
56	0.1
79	0.1
80	192.643
86	2
87	0.1
81	4
Total	**275.4433**

Group 64

Project	Allocation
8	5.9519
Total	**5.9519**

Lower bound allocation groups

Group	Project	Allocation
28	78	0.1
49	16	0.1
	18	0.1
	19	0.5
	17	0.5
	42	0.1
	24	6
	20	0.4
	41	0.1
	36	0.5
	42	1
	12	20
	49	1
	66	3
	74	1
	51	0.3
	82	0.1
	61	2
	99	0.6
	100	3
4951	14	1
	21	0.1
	76	3
	77	0.1
	85	0.1
	11	0.1
	10	0.1
	52	2
51	25	0.1
	39	3
	46	0.1
	48	0.1
	75	0.1

Lower bound allocation groups

Group	Project	Allocation
52	38	0.1
	40	0.1
61	67	0.1
	94	1
	22	0.1
	26	1
	45	0.1
	35	4
	50	0.1
	57	0.1
	103	0.1
	6	0.1
	23	0.1
	28	0.1
	37	4
	58	0.1
	47	5
	55	0.1
	73	0.1
	102	0.1
	69	5
	101	0.1
	70	0.1
	98	4
	4	10
	2	0.1
	3	0.1
	5	0.1
80	1	0.1
90	64	0.1
	91	0.1

Upper bound group allocation

Group 41

Project	Allocation
15	0.1
60	0.1
27	0.1

C.3 QPRO 5.0 Linear Programming Results

QPRO 5.0 Linear Programming Output for Level 1 Results

Max	38.06416170											
Group Code	**28**	**41**	**49**	**4951**	**51**	**52**	**61**	**6163**	**63**	**64**	**80**	**90**
Allocation	0.1	0.6	40.2	6.5	3.4	0.2	38.8	78.504849	275.44326	5.9519	0.1	0.2
Weight	0.1067	0.1241	0.0645	0.0645	0.0645	0.0943	0.0554	0.0744	0.0711	0.0893	0.0951	0.0959
Upper bound	0.2	0.6	61.4	11.2	5.4	0.4	69.17	116	334.3	8.2	0.2	0.4
Lower bound	0.1	0.3	40.2	6.5	3.4	0.2	38.8	74.9	207.1	5	0.1	0.2
Capital Total:	450.0000059	**Bound:**	450									
Depreciation rate	0.55	0.61	0.2	0.35	0.36	0.77	0.22	0.14	0.08	0.19	0.85	0.54
Depreciation value	0.055	0.366	8.04	2.275	1.224	0.154	8.536	10.990679	22.035461	1.130861	0.085	0.108
Total:	55.00000040	**Bound:**	55									
EQE index	60	30	11.3	18.57	2.22	310	10.37	2.65	0.83	26.71	40	147.5
EQE value	6	18	454.26	120.705	7.548	62	402.356	208.03785	228.6179	158.97525	4	29.5
Total:	1700.000002	**Bound:**	1700									

QPRO 5.0 Linear Programming Output for Second Level (group 6163) Results

Max	16.9174916538								
Project code	**Megaproject**	**54**	**34**	**32**	**43**	**62**	**90**	**97**	**29**
Allocation	57.52441605872	5.00000002	4.7475038	5	3	2	0.95	0.1329301	0.15
Weight	0.2347	0.3061	0.2551	0.2041					
EQE Index	2.2	10.22	7.97	0.05	1.43	10.8	10	340	0
EQE Value	126.5537153292	51.1000002	37.837605	0.25	4.29	21.6	9.5	45.196236	0
Total EQE	296.3275573159			**EQE Bound**	500				
Depreciation rate	0.1161	0.17	0.21	0.18	0.26	0.34	0.41	0.63	0.74
Depreciation Value	6.678584704418	0.85	0.9969758	0.9	0.78	0.68	0.3895	0.083746	0.111
Total Depreciation	11.46980647701		**Depreciation Bound:**		50				
Upper bound	80.8	9.2	7.4	9.1	5.6	2.5	1	0.2	0.2
Lower bound	55.2	5	4	5	3	2	0.5	0.1	0.1
Total capital		78.50485	**Bound:**	78.50485					

43 supporting	**54 fraction:** 0.5	
62 supporting	**54 fraction:** 0.4	
90 supporting	**34 fraction:** 0.2	**32 fraction:** 0.19
97 supporting	**54 fraction:** 0.025	**34 fraction:** 0.028
29 supporting	**32 fraction** 0.03	

QPRO 5.0 Results for Megaproject suballocation (group 6361)

Total fund	57.5244					
Project	**63**	**92**	**89**	**95**	**93**	**96**
Allocation	5.3813469	0.1090797	20.562941	0.1090797	10.453984	20.907969
Upper bound	9.2	0.2	26.2	0.2	15	30
Lower bound	5	0.1	20	0.1	10	20
Ri	4.2	0.1	6.2	0.1	5	10
Ri/SUM(Ri)	0.1640625	0.0039063	0.2421875	0.0039063	0.1953125	0.390625
f=Total_fund-SUM(lower bound)			2.3244			

QPRO 5.0 LP Results for Group63 Allocation

Max	580.9866								
Project Code	**M1**	**M2**	**33**	**56**	**79**	**80**	**86**	**87**	**81**
Allocation	18.4	43.1	15	0.1	0.1	192.643	2	0.1	4
Weight	0.1851	0.2222	0.0741	0.0741	0.0741	0.1481	0.0741	0.0741	0.0741
Upper bound	18.4	43.1	26.5	0.2	0.2	235.6	3.5	0.2	6.6
Lower bound	11.5	24.2	15	0.1	0.1	150	2	0.1	4
Capital									
Total:	275.4433	Bound:	275.4433						
Depreciation rate	0.163	-0.158	0.1	0.85	0.55	0.05	0.3	0.59	0.23
Depreciation value	2.9992	6.8098	1.5	0.085	0.055	9.63217	0.6	0.059	0.92
Total:	23.436165	**Bound:**	40						
Eqe index	22.28	7.9	0.26	0	0	0	15	40	0
Eqe value	409.952	340.49	3.9	0	0	0	30	4	0
Total:	849.342	**Bound:**	1250						

Appendix D

Capital Allocation Summaries

This appendix present the capital bugeting allocation results of three different approaches. They are:

D.1 Hierarchy allocation results: using hierarchy allocation method to solve the problem.

D.2 Direct allocation results: using direct weight method to solve the problem.

D.3 Direct allocation with dependency constraints: using direct weight method and adding dependency constraints to the capital budgeting problem.

D.4 Comparison the results of three different capital budgeting allocation methods.

D.1 Results of Hierarchy Allocation

Project number	Allocation	Depreciation	EQE
1	0.1	0.085	4
2	0.1	0.086	9.5
3	0.1	0.089	3.5
4	10	1.4	52.02702703
5	0.1	0.09	1.5
6	0.1	0.092	2.5
7	6.5	1.56	10
8	5.9519	1.130861	158.9592805
9	26.5	2.65	9
10	0.1	0.094	7
11	0.1	0.096	7
12	20	1.6	30.21582734
13	3	0.75	27.93103448
14	1	0.38	34.5
15	0.2	0.13	3
16	0.1	0.066	0.5
17	0.5	0.22	2.5
18	0.1	0.067	2
19	0.5	0.215	4
20	0.4	0.18	4
21	0.1	0.068	1
22	0.1	0.069	5
23	0.1	0.07	1
24	6	0.9	6.6
25	0.1	0.071	0
26	1	0.37	14.5
27	0.2	0.144	8
28	0.1	0.073	45.5
29	0.15	0.111	0
30	7.2	1.584	0
31	0.2	0.15	4
32	5	0.9	0.274725275
33	15	1.5	3.962264151
34	4.747504	0.99697584	37.85172108

Project number	Allocation	Depreciation	EQE
35	4	0.8	3.03030303
36	0.5	0.21	9
37	4	1	90.79365079
38	0.1	0.076	0
39	3	0.9	0
40	0.1	0.077	62
41	0.1	0.078	7
42	1	0.35	4.347826087
43	3	0.78	4.285714286
44	0.2	0.158	31
45	0.1	0.08	13
46	0.1	0.081	1
47	5	0.8	17.20430108
48	0.1	0.082	2
49	1	0.36	13.63636364
50	0.1	0.083	1.5
51	0.3	0.135	5.25
52	2	0.64	48.66666667
53	16	1.92	24
54	5	0.85	51.08695652
55	0.1	0.084	27.5
56	0.1	0.085	0
57	0.1	0.045	2.5
58	0.1	0.045	0.5
59	0.2	0.092	36
60	0.2	0.094	7
61	2	0.62	58
62	2	0.68	21.6
63	5.38135	1.07627	6.434222826
64	0.1	0.048	22
65	0.2	0.098	33
66	3	0.87	20.2173913
67	0.1	0.098	3.333333333
68	2.5	0.825	12
69	5	0.75	14.35185185
70	0.1	0.05	19.5
71	0.2	0.1	6
72	0.2	0.102	29
73	0.1	0.052	8
74	1	0.4	293.6363636

Project number	Allocation	Depreciation	EQE
75	0.1	0.053	3
76	3	0.81	13.26923077
77	0.1	0.054	4.5
78	0.1	0.055	6
79	0.1	0.055	0
80	192.643	9.63215	0.817669779
81	4	0.92	0
82	0.1	0.056	13.5
83	0.2	0.114	3
84	1.2	0.468	2
85	0.1	0.058	2
86	2	0.6	30.28571429
87	0.1	0.059	4
88	0.2	0.12	11
89	20.5629	2.261919	0.784843511
90	0.95	0.3895	9.5
91	0.1	0.06	7.5
92	0.10908	0.0665388	1.0908
93	10.454	1.35902	0.696933333
94	1	0.35	5.454545455
95	0.10908	0.0676296	13.635
96	20.908	1.46356	0.696933333
97	0.13293	0.0837459	45.1962
98	4	0.96	3.692307692
99	0.6	0.24	26.4
100	3	0.84	16.95652174
101	0.1	0.064	5.5
102	0.1	0.064	10.5
103	0.1	0.065	4
Total	**449.999744**	**55.05117014**	**1755.673525**

D.2 Results of Direct Allocation

Project number	Allocation	Depreciation	EQE
1	0.1	0.085	4
2	0.1	0.086	9.5
3	0.1	0.089	3.5
4	10	1.4	52.02702703
5	0.1	0.09	1.5
6	0.1	0.092	2.5
7	6.5	1.56	10
8	5	0.95	133.5365854
9	26.5	2.65	9
10	0.1	0.094	7
11	0.1	0.096	7
12	20	1.6	30.21582734
13	3	0.75	27.93103448
14	1	0.38	34.5
15	0.1	0.065	1.5
16	0.1	0.066	0.5
17	0.5	0.22	2.5
18	0.1	0.067	2
19	0.5	0.215	4
20	0.4	0.18	4
21	0.1	0.068	1
22	0.1	0.069	5
23	0.1	0.07	1
24	6	0.9	6.6
25	0.1	0.071	0
26	1	0.37	14.5
27	0.1	0.072	4
28	0.1	0.073	45.5
29	0.15	0.111	0
30	7.2	1.584	0
31	0.158	0.1185	3.16
32	5	0.9	0.274725275
33	15	1.5	3.962264151
34	5	1.05	39.86486486

Project number	Allocation	Depreciation	EQE
35	4	0.8	3.03030303
36	0.5	0.21	9
37	4	1	90.79365079
38	0.2	0.152	0
39	3	0.9	0
40	0.1	0.077	62
41	0.1	0.078	7
42	1	0.35	4.347826087
43	3	0.78	4.285714286
44	0.158	0.12482	24.49
45	0.1	0.08	13
46	0.1	0.081	1
47	5	0.8	17.20430108
48	0.1	0.082	2
49	1	0.36	13.63636364
50	0.1	0.083	1.5
51	0.3	0.135	5.25
52	2	0.64	48.66666667
53	13.463	1.61556	20.1945
54	5	0.85	51.08695652
55	0.1	0.084	27.5
56	0.1	0.085	0
57	0.1	0.045	2.5
58	0.1	0.045	0.5
59	0.158	0.07268	28.44
60	0.1	0.047	3.5
61	2	0.62	58
62	2.5	0.85	27
63	5	1	5.97826087
64	0.1	0.048	22
65	0.158	0.07742	26.07
66	3	0.87	20.2173913
67	0.1	0.098	3.333333333
68	2.5	0.825	12
69	5	0.75	14.35185185
70	0.1	0.05	19.5
71	0.2	0.1	6
72	0.2	0.102	29
73	0.1	0.052	8
74	1	0.4	293.6363636

Project number	Allocation	Depreciation	EQE
75	0.1	0.053	3
76	3	0.81	13.26923077
77	0.1	0.054	4.5
78	0.1	0.055	6
79	0.1	0.055	0
80	198.186	9.9093	0.841196944
81	4	0.92	0
82	0.1	0.056	13.5
83	0.158	0.09006	2.37
84	1.115	0.43485	1.858333333
85	0.1	0.058	2
86	2	0.6	30.28571429
87	0.1	0.059	4
88	0.158	0.0948	8.69
89	20	2.2	0.763358779
90	1	0.41	10
91	0.1	0.06	7.5
92	0.1	0.061	1
93	10	1.3	0.666666667
94	1	0.35	5.454545455
95	0.1	0.062	12.5
96	20	1.4	0.666666667
97	0.14	0.0882	47.6
98	4	0.96	3.692307692
99	0.6	0.24	26.4
100	3	0.84	16.95652174
101	0.1	0.064	5.5
102	0.1	0.064	10.5
103	0.1	0.065	4
Total	**450.002**	**54.52419**	**1701.100354**

D.3 Results of Direct Allocation with Depedency Constraints

Project number	Allocation	Depreciation	EQE
1	0.1	0.085	4
2	0.1	0.086	9.5
3	0.1	0.089	3.5
4	14.5	2.03	75.43918919
5	0.1	0.09	1.5
6	0.1	0.092	2.5
7	6.5	1.56	10
8	5	0.95	133.5365854
9	26.5	2.65	9
10	0.1	0.094	7
11	0.1	0.096	7
12	20	1.6	30.21582734
13	3	0.75	27.93103448
14	1	0.38	34.5
15	0.1	0.065	1.5
16	0.1	0.066	0.5
17	0.5	0.22	2.5
18	0.1	0.067	2
19	0.5	0.215	4
20	0.4	0.18	4
21	0.1	0.068	1
22	0.125	0.08625	6.25
23	0.1	0.07	1
24	6	0.9	6.6
25	0.171	0.12141	0
26	1.246	0.46102	18.067
27	0.1	0.072	4
28	0.1	0.073	45.5
29	0.2	0.148	0
30	7.2	1.584	0
31	0.1	0.075	2
32	5	0.9	0.274725275
33	15	1.5	3.962264151
34	4	0.84	31.89189189

Project number	Allocation	Depreciation	EQE
35	4	0.8	3.03030303
36	0.5	0.21	9
37	4	1	90.79365079
38	0.2	0.152	0
39	4.129	1.2387	0
40	0.1	0.077	62
41	0.1	0.078	7
42	1	0.35	4.347826087
43	4.177	1.08602	5.967142857
44	0.1	0.079	15.5
45	0.125	0.1	16.25
46	0.1	0.081	1
47	5	0.8	17.20430108
48	0.1	0.082	2
49	1	0.36	13.63636364
50	0.1	0.083	1.5
51	0.3	0.135	5.25
52	2	0.64	48.66666667
53	10	1.2	15
54	5	0.85	51.08695652
55	0.1	0.084	27.5
56	0.2	0.17	0
57	0.1	0.045	2.5
58	0.1	0.045	0.5
59	0.1	0.046	18
60	0.1	0.047	3.5
61	2	0.62	58
62	2.5	0.85	27
63	5	1	5.97826087
64	0.1	0.048	22
65	0.1	0.049	16.5
66	3.18	0.9222	21.43043478
67	0.1	0.098	3.333333333
68	2.5	0.825	12
69	5	0.75	14.35185185
70	0.1	0.05	19.5
71	0.2	0.1	6
72	0.2	0.102	29
73	0.1	0.052	8
74	1	0.4	293.6363636

Project number	Allocation	Depreciation	EQE
75	0.1	0.053	3
76	3	0.81	13.26923077
77	0.1	0.054	4.5
78	0.1	0.055	6
79	0.1	0.055	0
80	179.102	8.9551	0.760195246
81	5.5	1.265	0
82	0.1	0.056	13.5
83	0.1	0.057	1.5
84	1	0.39	1.666666667
85	0.1	0.058	2
86	3.236	0.9708	49.00228571
87	0.1	0.059	4
88	0.1	0.06	5.5
89	20	2.2	0.763358779
90	0.95	0.3895	9.5
91	0.1	0.06	7.5
92	0.1	0.061	1
93	10	1.3	0.666666667
94	1	0.35	5.454545455
95	0.1	0.062	12.5
96	20	1.4	0.666666667
97	0.125	0.07875	42.5
98	4	0.96	3.692307692
99	0.6	0.24	26.4
100	3	0.84	16.95652174
101	0.1	0.064	5.5
102	0.1	0.064	10.5
103	0.1	0.065	4
Total	**436.166**	**55.00075**	**1700.930418**

D.4 Comparison the Results of the Three Different Allocations

Project Number	Upper Bound	Lower Bound	Hierarchy Allocation	Direct Allocation	Direct Allocation with Constraints
1	0.2	0.1	0.1	0.1	0.1
2	0.2	0.1	0.1	0.1	0.1
3	0.2	0.1	0.1	0.1	0.1
4	14.8	10	10	10	14.5
5	0.2	0.1	0.1	0.1	0.1
6	0.2	0.1	0.1	0.1	0.1
7	6.5	4	6.5	6.5	6.5
8	8.2	5	5.9519	5	5
9	26.5	15	26.5	26.5	26.5
10	0.2	0.1	0.1	0.1	0.1
11	0.2	0.1	0.1	0.1	0.1
12	27.8	20	20	20	20
13	5.8	3	3	3	3
14	2	1	1	1	1
15	0.2	0.1	0.2	0.1	0.1
16	0.2	0.1	0.1	0.1	0.1
17	1	0.5	0.5	0.5	0.5
18	0.2	0.1	0.1	0.1	0.1
19	1	0.5	0.5	0.5	0.5
20	0.6	0.4	0.4	0.4	0.4
21	0.2	0.1	0.1	0.1	0.1
22	0.2	0.1	0.1	0.1	0.125
23	0.2	0.1	0.1	0.1	0.1
24	10	6	6	6	6
25	0.2	0.1	0.1	0.1	0.171
26	2	1	1	1	1.246
27	0.2	0.1	0.2	0.1	0.1
28	0.2	0.1	0.1	0.1	0.1
29	0.2	0.1	0.15	0.15	0.2
30	7.2	4	7.2	7.2	7.2
31	0.2	0.1	0.2	0.158	0.1
32	9.1	5	5	5	5
33	26.5	15	15	15	15
34	7.4	4	4.747504	5	4

Project Number	Upper Bound	Lower Bound	Hierarchy Allocation	Direct Allocation	Direct Allocation with Constraints
35	7.92	4	4	4	4
36	1	0.5	0.5	0.5	0.5
37	6.3	4	4	4	4
38	0.2	0.1	0.1	0.2	0.2
39	4.6	3	3	3	4.129
40	0.2	0.1	0.1	0.1	0.1
41	0.2	0.1	0.1	0.1	0.1
42	2.3	1	1	1	1
43	5.6	3	3	3	4.177
44	0.2	0.1	0.2	0.158	0.1
45	0.2	0.1	0.1	0.1	0.125
46	0.2	0.1	0.1	0.1	0.1
47	9.3	5	5	5	5
48	0.2	0.1	0.1	0.1	0.1
49	2.2	1	1	1	1
50	0.2	0.1	0.1	0.1	0.1
51	0.4	0.3	0.3	0.3	0.3
52	3	2	2	2	2
53	16	10	16	13.463	10
54	9.2	5	5	5	5
55	0.2	0.1	0.1	0.1	0.1
56	0.2	0.1	0.1	0.1	0.2
57	0.2	0.1	0.1	0.1	0.1
58	0.2	0.1	0.1	0.1	0.1
59	0.2	0.1	0.2	0.158	0.1
60	0.2	0.1	0.2	0.1	0.1
61	3	2	2	2	2
62	2.5	2	2	2.5	2.5
63	9.2	5	5.38135	5	5
64	0.2	0.1	0.1	0.1	0.1
65	0.2	0.1	0.2	0.158	0.1
66	4.6	3	3	3	3.18
67	0.15	0.1	0.1	0.1	0.1
68	2.5	1	2.5	2.5	2.5
69	10.8	5	5	5	5

Project Number	Upper Bound	Lower Bound	Hierarchy Allocation	Direct Allocation	Direct Allocation with Constraints
70	0.2	0.1	0.1	0.1	0.1
71	0.2	0.1	0.2	0.2	0.2
72	0.2	0.1	0.2	0.2	0.2
73	0.2	0.1	0.1	0.1	0.1
74	1.1	1	1	1	1
75	0.2	0.1	0.1	0.1	0.1
76	5.2	3	3	3	3
77	0.2	0.1	0.1	0.1	0.1
78	0.2	0.1	0.1	0.1	0.1
79	0.2	0.1	0.1	0.1	0.1
80	235.6	150	192.643	198.186	179.102
81	6.6	4	4	4	5.5
82	0.2	0.1	0.1	0.1	0.1
83	0.2	0.1	0.2	0.158	0.1
84	1.2	1	1.2	1.115	1
85	0.2	0.1	0.1	0.1	0.1
86	3.5	2	2	2	3.236
87	0.2	0.1	0.1	0.1	0.1
88	0.2	0.1	0.2	0.158	0.1
89	26.2	20	20.5629	20	20
90	1	0.5	0.95	1	0.95
91	0.2	0.1	0.1	0.1	0.1
92	0.2	0.1	0.10908	0.1	0.1
93	15	10	10.454	10	10
94	2.2	1	1	1	1
95	0.2	0.1	0.10908	0.1	0.1
96	30	20	20.908	20	20
97	0.2	0.1	0.13293	0.14	0.125
98	6.5	4	4	4	4
99	1	0.6	0.6	0.6	0.6
100	4.6	3	3	3	3
101	0.2	0.1	0.1	0.1	0.1
102	0.2	0.1	0.1	0.1	0.1
103	0.2	0.1	0.1	0.1	0.1
Total	**607.47**	**377**	**449.999744**	**450.002**	**436.166**

Appendix E

GAMS input and output

This appendix present the GAMS input and output of the capital budgeting problem.

E.1 GAMS Input File of Direct Allocation with Constraints Problem

```
SET
  I     /1*103/;

PARAMETERS
 W(I)
/1 0.1149
 2 0.0679
 3 0.0679
 4 0.0679
 5 0.0679
 6 0.0679
 7 0.1314
 8 0.1079
 9 0.1314
 10 0.0779
 11 0.0779
 12 0.0779
 13 0.0679
 14 0.0779
 15 0.1099
 16 0.0779
 17 0.0779
 18 0.0779
 19 0.0779
 20 0.0779
 21 0.0779
 22 0.0679
 23 0.0679
 24 0.0779
 25 0.0779
 26 0.0679
 27 0.1099
 28 0.0679
 29 0
 30 0.1314
 31 0.1095
 32 0.0876
 33 0.0438
 34 0.1697
 35 0.0679
 36 0.0779
 37 0.0679
 38 0.1139
 39 0.0779
 40 0.1139
 41 0.0779
 42 0.0779
 43 0
 44 0.1095
 45 0.0679
 46 0.0779
 47 0.0679
 48 0.0779
 49 0.0779
 50 0.0679
 51 0.0779
 52 0.0779
 53 0.1095
 54 0.2036
 55 0.0679
 56 0.0438
 57 0.0679
 58 0.0679
 59 0.1095
 60 0.1099
 61 0.0779
 62 0
 63 0.0779
 64 0.1159
 65 0.1095
 66 0.0779
 67 0.0679
 68 0.1314
 69 0.0679
 70 0.0679
 71 0.1314
 72 0.1314
 73 0.0679
 74 0.0779
 75 0.0779
 76 0.0779
 77 0.0779
```

```
78 0.1289
79 0.0438
80 0.0876
81 0.0438
82 0.0779
83 0.1095
84 0.1095
85 0.0779
86 0.0438
87 0.0438
88 0.1095
89 0.0438
90 0
91 0.1159
92 0.0679
93 0.0779
94 0.0679
95 0.0679
96 0.0438
97 0
98 0.0679
99 0.0779
100 0.0779
101 0.0679
102 0.0679
103 0.0679   /,

   UP(I)
/1   0.2
2   0.2
3   0.2
4   14.8
5   0.2
6   0.2
7   6.5
8   8.2
9   26.5
10   0.2
11   0.2
12   27.8
13   5.8
14   2
15   0.2
16   0.2
17   1
18   0.2
19   1
20   0.6
21   0.2
22   0.2
23   0.2
24   10
25   0.2
26   2
27   0.2
28   0.2
29   0.2
30   7.2
31   0.2
32   9.1
33   26.5
34   7.4
35   7.92
36   1
37   6.3
38   0.2
39   4.6
40   0.2
41   0.2
42   2.3
43   5.6
44   0.2
45   0.2
46   0.2
47   9.3
48   0.2
49   2.2
50   0.2
51   0.4
52   3
53   16
54   9.2
55   0.2
56   0.2
57   0.2
58   0.2
59   0.2
60   0.2
61   3
62   2.5
63   9.2
64   0.2
65   0.2
66   4.6
67   0.15
68   2.5
69   10.8
70   0.2
71   0.2
72   0.2
73   0.2
74   1.1
75   0.2
76   5.2
77   0.2
78   0.2
79   0.2
80   235.6
```

```
81   6.6
82   0.2
83   0.2
84   1.2
85   0.2
86   3.5
87   0.2
88   0.2
89   26.2
90   1
91   0.2
92   0.2
93   15
94   2.2
95   0.2
96   30
97   0.2
98   6.5
99   1
100  4.6
101  0.2
102  0.2
103  0.2/,

LO(I)
 /1  0.1
  2  0.1
  3  0.1
  4  10
  5  0.1
  6  0.1
  7  4
  8  5
  9  15
 10  0.1
 11  0.1
 12  20
 13  3
 14  1
 15  0.1
 16  0.1
 17  0.5
 18  0.1
 19  0.5
 20  0.4
 21  0.1
 22  0.1
 23  0.1
 24  6
 25  0.1
 26  1
 27  0.1
 28  0.1
 29  0.1
 30  4
 31  0.1
 32  5
 33  15
 34  4
 35  4
 36  0.5
 37  4
 38  0.1
 39  3
 40  0.1
 41  0.1
 42  1.0
 43  3
 44  0.1
 45  0.1
 46  0.1
 47  5
 48  0.1
 49  1
 50  0.1
 51  0.3
 52  2
 53  10
 54  5
 55  0.1
 56  0.1
 57  0.1
 58  0.1
 59  0.1
 60  0.1
 61  2
 62  2
 63  5
 64  0.1
 65  0.1
 66  3
 67  0.1
 68  1
 69  5
 70  0.1
 71  0.1
 72  0.1
 73  0.1
 74  1
 75  0.1
 76  3
 77  0.1
 78  0.1
 79  0.1
 80  150
 81  4
 82  0.1
 83  0.1
```

```
84   1
85   0.1
86   2
87   0.1
88   0.1
89   20
90   0.5
91   0.1
92   0.1
93   10
94   1
95   0.1
96   20
97   0.1
98   4
99   0.6
100  3
101  0.1
102  0.1
103  0.1 /,
R(I),
DE(I)
/1   0.85
2    0.86
3    0.89
4    0.14
5    0.9
6    0.92
7    0.24
8    0.19
9    0.1
10   0.94
11   0.96
12   0.08
13   0.25
14   0.38
15   0.65
16   0.66
17   0.44
18   0.67
19   0.43
20   0.45
21   0.68
22   0.69
23   0.7
24   0.15
25   0.71
26   0.37
27   0.72
28   0.73
29   0.74
30   0.22
31   0.75
32   0.18
33   0.1
34   0.21
35   0.2
36   0.42
37   0.25
38   0.76
39   0.3
40   0.77
41   0.78
42   0.35
43   0.26
44   0.79
45   0.8
46   0.81
47   0.16
48   0.82
49   0.36
50   0.83
51   0.45
52   0.32
53   0.12
54   0.17
55   0.84
56   0.85
57   0.45
58   0.45
59   0.46
60   0.47
61   0.31
62   0.34
63   0.2
64   0.48
65   0.49
66   0.29
67   0.98
68   0.33
69   0.15
70   0.5
71   0.5
72   0.51
73   0.52
74   0.4
75   0.53
76   0.27
77   0.54
78   0.55
79   0.55
80   0.05
81   0.23
82   0.56
83   0.57
84   0.39
```

```
85   0.58
86   0.3
87   0.59
88   0.6
89   0.11
90   0.41
91   0.6
92   0.61
93   0.13
94   0.35
95   0.62
96   0.07
97   0.63
98   0.24
99   0.4
100  0.28
101  0.64
102  0.64
103  0.65/,

EQE(I)
 /1  40
  2  95
  3  35
  4  5.2
  5  15
  6  25
  7  1.54
  8  26.71
  9  0.34
 10  70
 11  70
 12  1.51
 13  9.31
 14  34.50
 15  15
 16  5
 17  5
 18  20
 19  8
 20  10
 21  10
 22  50
 23  10
 24  1.1
 25  0
 26  14.5
 27  40
 28  455
 29  0
 30  0
 31  20
 32  0.05
 33  0.26
 34  7.97
 35  0.76
 36  18
 37  22.7
 38  0
 39  0
 40  620
 41  70
 42  4.35
 43  1.43
 44  155
 45  130
 46  10
 47  3.44
 48  20
 49  13.64
 50  15
 51  17.5
 52  24.33
 53  1.5
 54  10.22
 55  275
 56  0
 57  25
 58  5
 59  180
 60  35
 61  29
 62  10.8
 63  1.2
 64  220
 65  165
 66  6.74
 67  33.33
 68  4.8
 69  2.87
 70  195
 71  30
 72  145
 73  80
 74  293.64
 75  30
 76  4.42
 77  45
 78  60
 79  0
 80  0
 81  0
 82  135
 83  15
 84  1.67
 85  20
 86  15.14
 87  40
```

```
     88  55
     89  0.04
     90  10
     91  75
     92  10
     93  0.07
     94  5.45
     95  125
     96  0.03
     97  340
     98  0.92
     99  44
     100 5.65
     101 55
     102 105
     103 40/ ;

R(I)=UP(I)-LO(I);


VARIABLES
   X(I)
   Z

EQUATIONS
   OBJECTIVE
   FUND
   UPPER(I)
   LOWER(I)
   EQECONS
   DEPREC
   SUPPORT1
   SUPPORT2
   SUPPORT3
   SUPPORT4
   SUPPORT5
   SUPPORT6
   SUPPORT7
   MEGA61631
   MEGA61632
   MEGA61633
   MEGA61634
   MEGA61635
   MEGA61636
   MEGA6141
   MEGA6142
   MEGA6143
   MEGA6144
   MEGA6145
   MEGA6146
   MEGA6321
   MEGA6322
   MEGA6323
   MEGA6324
   MEGA6325
   MEGA6326
   MEGA49511
   MEGA49512
   MEGA49513
   MEGA49514
   MEGA49515
   MEGA49516
   MEGA49517
   MEGA49518
   MEGA6311
   MEGA6312
   MEGA6313
   MEGA6314
   MEGA6315
   MEGA6316
   MEGA6317
   MEGA6318
   MEGA4921
   MEGA4922
   MEGA4923
   MEGA4924
   MEGA4925
   MEGA5121
   MEGA5122
   MEGA5123
   MEGA6121
   MEGA6122
   MEGA6123
   MEGA6131
   MEGA6132
   MEGA6133
   MEGA411
   MEGA412
   MEGA6111
   MEGA6112
   MEGA5111
   MEGA5112
   MEGA901
   MEGA902
   CAP1
   CAP2
   CAP3
   EQE1
   DEP1
   DEP2
   SUPPORT8
   SUPPORT9
   SUPPORT10;
```

```
OBJECTIVE.. Z =E= SUM(I,W(I)*X(I));
FUND.. SUM(I,X(I)) =L= 450;
UPPER(I).. X(I) =L= UP(I);
LOWER(I).. X(I) =G= LO(I);
EQECONS.. SUM(I,X(I)*EQE(I)) =L= 1700;
DEPREC.. SUM(I,X(I)*DE(I)) =L= 55;
SUPPORT1.. X('43') =G= 0.5*X('54');
SUPPORT2.. X('62') =G= 0.5*X('54');
SUPPORT3.. X('90') =G= 0.2*X('34');
SUPPORT4.. X('90') =G= 0.19*X('32');
SUPPORT5.. X('97') =G= 0.025*X('54');
SUPPORT6.. X('97') =G= 0.028*X('34');
SUPPORT7.. X('29') =G= 0.03*X('32');
MEGA61631.. X('63') =E= (X('63')-LO('63')+X('92')-LO('92')+X('89')
-LO('89')+X('95')-LO('95')+X('93')-LO('93')+X('96')-LO('96'))*R('63')
/(R('63')+R('92')+R('89')+R('95')+R('93')+R('96'))+LO('63');
MEGA61632.. X('92') =E= (X('63')-LO('63')+X('92')-LO('92')+X('89')
-LO('89')+X('95')-LO('95')+X('93')-LO('93')+X('96')-LO('96'))*R('92')
/(R('63')+R('92')+R('89')+R('95')+R('93')+R('96'))+LO('92');
MEGA61633.. X('89') =E= (X('63')-LO('63')+X('92')-LO('92')+X('89')
-LO('89')+X('95')-LO('95')+X('93')-LO('93')+X('96')-LO('96'))*R('89')
/(R('63')+R('92')+R('89')+R('95')+R('93')+R('96'))+LO('89');
MEGA61634.. X('95') =E= (X('63')-LO('63')+X('92')-LO('92')+X('89')
-LO('89')+X('95')-LO('95')+X('93')-LO('93')+X('96')-LO('96'))*R('95')
/(R('63')+R('92')+R('89')+R('95')+R('93')+R('96'))+LO('95');
MEGA61635.. X('93') =E= (X('63')-LO('63')+X('92')-LO('92')+X('89')
-LO('89')+X('95')-LO('95')+X('93')-LO('93')+X('96')-LO('96'))*R('93')
/(R('63')+R('92')+R('89')+R('95')+R('93')+R('96'))+LO('93');
MEGA61636.. X('96') =E= (X('63')-LO('63')+X('92')-LO('92')+X('89')
-LO('89')+X('95')-LO('95')+X('93')-LO('93')+X('96')-LO('96'))*R('96')
/(R('63')+R('92')+R('89')+R('95')+R('93')+R('96'))+LO('96');
MEGA6141.. X('103') =E= (X('103')-LO('103')+X('6')-LO('6')+X('23')
-LO('23')+X('28')-LO('28')+X('37')-LO('37')+X('58')-LO('58'))*R('103')
/(R('103')+R('6')+R('23')+R('28')+R('37')+R('58'))+LO('103');
MEGA6142.. X('6') =E= (X('103')-LO('103')+X('6')-LO('6')+X('23')
-LO('23')+X('28')-LO('28')+X('37')-LO('37')+X('58')-LO('58'))*R('6')
/(R('103')+R('6')+R('23')+R('28')+R('37')+R('58'))+LO('6');
MEGA6143.. X('23') =E= (X('103')-LO('103')+X('6')-LO('6')+X('23')
-LO('23')+X('28')-LO('28')+X('37')-LO('37')+X('58')-LO('58'))*R('23')
/(R('103')+R('6')+R('23')+R('28')+R('37')+R('58'))+LO('23');
MEGA6144.. X('28') =E= (X('103')-LO('103')+X('6')-LO('6')+X('23')
-LO('23')+X('28')-LO('28')+X('37')-LO('37')+X('58')-LO('58'))*R('28')
/(R('103')+R('6')+R('23')+R('28')+R('37')+R('58'))+LO('28');
MEGA6145.. X('37') =E= (X('103')-LO('103')+X('6')-LO('6')+X('23')
-LO('23')+X('28')-LO('28')+X('37')-LO('37')+X('58')-LO('58'))*R('37')
/(R('103')+R('6')+R('23')+R('28')+R('37')+R('58'))+LO('37');
MEGA6146.. X('58') =E= (X('103')-LO('103')+X('6')-LO('6')+X('23')
-LO('23')+X('28')-LO('28')+X('37')-LO('37')+X('58')-LO('58'))*R('58')
/(R('103')+R('6')+R('23')+R('28')+R('37')+R('58'))+LO('58');
MEGA6321.. X('71') =E= (X('71')-LO('71')+X('7')-LO('7')+X('72')
-LO('72')+X('9')-LO('9')+X('30')-LO('30')+X('68')-LO('68'))*R('71')
/(R('71')+R('7')+R('72')+R('9')+R('30')+R('68'))+LO('71');
MEGA6322.. X('7') =E= (X('71')-LO('71')+X('7')-LO('7')+X('72')
```

-LO('72')+X('9')-LO('9')+X('30')-LO('30')+X('68')-LO('68'))*R('7')
/(R('71')+R('7')+R('72')+R('9')+R('30')+R('68'))+LO('7');

646 MEGA6323.. X('72') =E= (X('71')-LO('71')+X('7')-LO('7')+X('72')
-LO('72')+X('9')-LO('9')+X('30')-LO('30')+X('68')-LO('68'))*R('72')
/(R('71')+R('7')+R('72')+R('9')+R('30')+R('68'))+LO('72');

647 MEGA6324.. X('9') =E= (X('71')-LO('71')+X('7')-LO('7')+X('72')
-LO('72')+X('9')-LO('9')+X('30')-LO('30')+X('68')-LO('68'))*R('9')
/(R('71')+R('7')+R('72')+R('9')+R('30')+R('68'))+LO('9');

648 MEGA6325.. X('30') =E= (X('71')-LO('71')+X('7')-LO('7')+X('72')
-LO('72')+X('9')-LO('9')+X('30')-LO('30')+X('68')-LO('68'))*R('30')
/(R('71')+R('7')+R('72')+R('9')+R('30')+R('68'))+LO('30');

649 MEGA6326.. X('68') =E= (X('71')-LO('71')+X('7')-LO('7')+X('72')
-LO('72')+X('9')-LO('9')+X('30')-LO('30')+X('68')-LO('68'))*R('68')
/(R('71')+R('7')+R('72')+R('9')+R('30')+R('68'))+LO('68');

650 MEGA49511.. X('14') =E= (X('14')-LO('14')+X('21')-LO('21')+X('76')
-LO('76')+X('77')-LO('77')+X('85')-LO('85')+X('11')-LO('11')+X('10')
-LO('10')+X('52')-LO('52'))*R('14')/(R('14')+R('21')+R('76')+R('77')
+R('85')+R('11')+R('10')+R('52'))+LO('14');

651 MEGA49512.. X('21') =E= (X('14')-LO('14')+X('21')-LO('21')+X('76')
-LO('76')+X('77')-LO('77')+X('85')-LO('85')+X('11')-LO('11')+X('10')
-LO('10')+X('52')-LO('52'))*R('21')/(R('14')+R('21')+R('76')+R('77')
+R('85')+R('11')+R('10')+R('52'))+LO('21');

652 MEGA49513.. X('76') =E= (X('14')-LO('14')+X('21')-LO('21')+X('76')
-LO('76')+X('77')-LO('77')+X('85')-LO('85')+X('11')-LO('11')+X('10')
-LO('10')+X('52')-LO('52'))*R('76')/(R('14')+R('21')+R('76')+R('77')
+R('85')+R('11')+R('10')+R('52'))+LO('76');

653 MEGA49514.. X('77') =E= (X('14')-LO('14')+X('21')-LO('21')+X('76')
-LO('76')+X('77')-LO('77')+X('85')-LO('85')+X('11')-LO('11')+X('10')
-LO('10')+X('52')-LO('52'))*R('77')/(R('14')+R('21')+R('76')+R('77')
+R('85')+R('11')+R('10')+R('52'))+LO('77');

654 MEGA49515.. X('85') =E= (X('14')-LO('14')+X('21')-LO('21')+X('76')
-LO('76')+X('77')-LO('77')+X('85')-LO('85')+X('11')-LO('11')+X('10')
-LO('10')+X('52')-LO('52'))*R('85')/(R('14')+R('21')+R('76')+R('77')
+R('85')+R('11')+R('10')+R('52'))+LO('85');

655 MEGA49516.. X('11') =E= (X('14')-LO('14')+X('21')-LO('21')+X('76')
-LO('76')+X('77')-LO('77')+X('85')-LO('85')+X('11')-LO('11')+X('10')
-LO('10')+X('52')-LO('52'))*R('11')/(R('14')+R('21')+R('76')+R('77')
+R('85')+R('11')+R('10')+R('52'))+LO('11');

656 MEGA49517.. X('10') =E= (X('14')-LO('14')+X('21')-LO('21')+X('76')
-LO('76')+X('77')-LO('77')+X('85')-LO('85')+X('11')-LO('11')+X('10')
-LO('10')+X('52')-LO('52'))*R('10')/(R('14')+R('21')+R('76')+R('77')
+R('85')+R('11')+R('10')+R('52'))+LO('10');

657 MEGA49518.. X('52') =E= (X('14')-LO('14')+X('21')-LO('21')+X('76')
-LO('76')+X('77')-LO('77')+X('85')-LO('85')+X('11')-LO('11')+X('10')
-LO('10')+X('52')-LO('52'))*R('52')/(R('14')+R('21')+R('76')+R('77')
+R('85')+R('11')+R('10')+R('52'))+LO('52');

658 MEGA6311.. X('31') =E= (X('31')-LO('31')+X('44')-LO('44')+X('53')
-LO('53')+X('59')-LO('59')+X('65')-LO('65')+X('83')-LO('83')+X('84')
-LO('84')+X('88')-LO('88'))*R('31')/(R('31')+R('44')+R('53')+R('59')
+R('65')+R('83')+R('84')+R('88'))+LO('31');

659 MEGA6312.. X('44') =E= (X('31')-LO('31')+X('44')-LO('44')+X('53')
-LO('53')+X('59')-LO('59')+X('65')-LO('65')+X('83')-LO('83')+X('84')
-LO('84')+X('88')-LO('88'))*R('44')/(R('31')+R('44')+R('53')+R('59')
+R('65')+R('83')+R('84')+R('88'))+LO('44');

```
660 MEGA6313.. X('53') =E= (X('31')-LO('31')+X('44')-LO('44')+X('53')
    -LO('53')+X('59')-LO('59')+X('65')-LO('65')+X('83')-LO('83')+X('84')
    -LO('84')+X('88')-LO('88'))*R('53')/(R('31')+R('44')+R('53')+R('59')
    +R('65')+R('83')+R('84')+R('88'))+LO('53');
661 MEGA6314.. X('59') =E= (X('31')-LO('31')+X('44')-LO('44')+X('53')
    -LO('53')+X('59')-LO('59')+X('65')-LO('65')+X('83')-LO('83')+X('84')
    -LO('84')+X('88')-LO('88'))*R('59')/(R('31')+R('44')+R('53')+R('59')
    +R('65')+R('83')+R('84')+R('88'))+LO('59');
662 MEGA6315.. X('65') =E= (X('31')-LO('31')+X('44')-LO('44')+X('53')
    -LO('53')+X('59')-LO('59')+X('65')-LO('65')+X('83')-LO('83')+X('84')
    -LO('84')+X('88')-LO('88'))*R('65')/(R('31')+R('44')+R('53')+R('59')
    +R('65')+R('83')+R('84')+R('88'))+LO('65');
663 MEGA6316.. X('83') =E= (X('31')-LO('31')+X('44')-LO('44')+X('53')
    -LO('53')+X('59')-LO('59')+X('65')-LO('65')+X('83')-LO('83')+X('84')
    -LO('84')+X('88')-LO('88'))*R('83')/(R('31')+R('44')+R('53')+R('59')
    +R('65')+R('83')+R('84')+R('88'))+LO('83');
664 MEGA6317.. X('84') =E= (X('31')-LO('31')+X('44')-LO('44')+X('53')
    -LO('53')+X('59')-LO('59')+X('65')-LO('65')+X('83')-LO('83')+X('84')
    -LO('84')+X('88')-LO('88'))*R('84')/(R('31')+R('44')+R('53')+R('59')
    +R('65')+R('83')+R('84')+R('88'))+LO('84');
665 MEGA6318.. X('88') =E= (X('31')-LO('31')+X('44')-LO('44')+X('53')
    -LO('53')+X('59')-LO('59')+X('65')-LO('65')+X('83')-LO('83')+X('84')
    -LO('84')+X('88')-LO('88'))*R('88')/(R('31')+R('44')+R('53')+R('59')
    +R('65')+R('83')+R('84')+R('88'))+LO('88');
666 MEGA4921.. X('24') =E= (X('24')-LO('24')+X('20')-LO('20')+X('41')
    -LO('41')+X('36')-LO('36')+X('42')-LO('42'))*R('24')/(R('24')+R('20')
    +R('41')+R('36')+R('42'))+LO('24');
667 MEGA4922.. X('20') =E= (X('24')-LO('24')+X('20')-LO('20')+X('41')
    -LO('41')+X('36')-LO('36')+X('42')-LO('42'))*R('20')/(R('24')+R('20')
    +R('41')+R('36')+R('42'))+LO('20');
668 MEGA4923.. X('41') =E= (X('24')-LO('24')+X('20')-LO('20')+X('41')
    -LO('41')+X('36')-LO('36')+X('42')-LO('42'))*R('41')/(R('24')+R('20')
    +R('41')+R('36')+R('42'))+LO('41');
669 MEGA4924.. X('36') =E= (X('24')-LO('24')+X('20')-LO('20')+X('41')
    -LO('41')+X('36')-LO('36')+X('42')-LO('42'))*R('36')/(R('24')+R('20')
    +R('41')+R('36')+R('42'))+LO('36');
670 MEGA4925.. X('42') =E= (X('24')-LO('24')+X('20')-LO('20')+X('41')
    -LO('41')+X('36')-LO('36')+X('42')-LO('42'))*R('42')/(R('24')+R('20')
    +R('41')+R('36')+R('42'))+LO('42');
671 MEGA5121.. X('46') =E= (X('46')-LO('46')+X('75')-LO('75')+X('48')
    -LO('48'))*R('46')/(R('46')+R('75')+R('48'))+LO('46');
672 MEGA5122.. X('75') =E= (X('46')-LO('46')+X('75')-LO('75')+X('48')
    -LO('48'))*R('75')/(R('46')+R('75')+R('48'))+LO('75');
673 MEGA5123.. X('48') =E= (X('46')-LO('46')+X('75')-LO('75')+X('48')
    -LO('48'))*R('48')/(R('46')+R('75')+R('48'))+LO('48');
674 MEGA6121.. X('22') =E= (X('22')-LO('22')+X('45')-LO('45')+X('26')
    -LO('26'))*R('22')/(R('22')+R('45')+R('26'))+LO('22');
675 MEGA6122.. X('45') =E= (X('22')-LO('22')+X('45')-LO('45')+X('26')
    -LO('26'))*R('45')/(R('22')+R('45')+R('26'))+LO('45');
676 MEGA6123.. X('26') =E= (X('22')-LO('22')+X('45')-LO('45')+X('26')
    -LO('26'))*R('26')/(R('22')+R('45')+R('26'))+LO('26');
677 MEGA6131.. X('35') =E= (X('35')-LO('35')+X('57')-LO('57')+X('50')
    -LO('50'))*R('35')/(R('35')+R('57')+R('50'))+LO('35');
678 MEGA6132.. X('57') =E= (X('35')-LO('35')+X('57')-LO('57')+X('50')
```

```
            -LO('50'))*R('57')/(R('35')+R('57')+R('50'))+LO('57');
MEGA6133.. X('50') =E= (X('35')-LO('35')+X('57')-LO('57')+X('50')
            -LO('50'))*R('50')/(R('35')+R('57')+R('50'))+LO('50');
MEGA411.. X('15') =E= (X('15')-LO('15')+X('60')-LO('60'))*R('15')
                       /(R('15')+R('60'))+LO('15');
MEGA412.. X('60') =E= (X('15')-LO('15')+X('60')-LO('60'))*R('60')
                       /(R('15')+R('60'))+LO('60');
MEGA5111.. X('25') =E= (X('25')-LO('25')+X('39')-LO('39'))*R('25')
                       /(R('25')+R('39'))+LO('25');
MEGA5112.. X('39') =E= (X('25')-LO('25')+X('39')-LO('39'))*R('39')
                       /(R('25')+R('39'))+LO('39');
MEGA6111.. X('67') =E= (X('67')-LO('67')+X('94')-LO('94'))*R('67')
                       /(R('67')+R('94'))+LO('67');
MEGA6112.. X('94') =E= (X('67')-LO('67')+X('94')-LO('94'))*R('94')
                       /(R('67')+R('94'))+LO('94');
MEGA901.. X('64') =E= (X('64')-LO('64')+X('91')-LO('91'))*R('64')
                       /(R('64')+R('91'))+LO('64');
MEGA902.. X('91') =E= (X('64')-LO('64')+X('91')-LO('91'))*R('91')
                       /(R('64')+R('91'))+LO('91');
CAP1.. X('78')+X('27')+X('12')+X('38')+X('4')+X('33')+X('8')+X('1')
                              +X('54') =G= 60;
CAP2.. X('64')+X('91')+X('25')+X('39') =G= 4.5;
CAP3.. X('17')+X('98')+X('81') =G= 10;
EQE1.. 30*X('71')+1.5*X('7')+145*X('72')+0.34*X('9')+4.8*X('68')
            +50*X('22')+1.45*X('26')+130*X('45') =G= 90;
DEP1.. 0.65*X('15')+0.47*X('60')+0.29*X('66')+0.76*X('38')
      +0.77*X('40')+0.89*X('3')+0.14*X('4')+0.5*X('86') =G= 5;
DEP2.. 0.26*X('43')+0.74*X('29')+0.96*X('11')+0.85*X('56') =G= 1.5;
SUPPORT8.. X('16') =G= 0.02*X('17');
SUPPORT9.. X('18') =G= 0.02*X('17');
SUPPORT10.. X('19') =G= X('17');

MODEL BUDGET /ALL/;
SOLVE BUDGET USING LP MAXIMIZING Z;
DISPLAY X.L,X.M;
```

E.2 GAMS Output File of Direct Allocation with Constraints Problem

S O L V E   S U M M A R Y

MODEL BUDGET   OBJECTIVE Z
TYPE LP   DIRECTION MAXIMIZE
SOLVER MINOS5   FROM LINE 699

**** SOLVER STATUS   1 NORMAL COMPLETION
**** MODEL STATUS   1 OPTIMAL
**** OBJECTIVE VALUE   36.3735

RESOURCE USAGE, LIMIT   7.470   1000.000
ITERATION COUNT, LIMIT   96   1000

M I N O S 5.3 (Nov 1990)   Ver: 225-386-02
= = = = =

B. A. Murtagh, University of New South Wales
and
P. E. Gill, W. Murray, M. A. Saunders and M. H. Wright
Systems Optimization Laboratory, Stanford University.

Work space allocated   -- 0.12 Mb

EXIT -- OPTIMAL SOLUTION FOUND

	LOWER	LEVEL	UPPER	MARGINAL
---- EQU OBJECTIVE	.	.	.	1.000
---- EQU FUND	-INF	436.165	450.000	.

---- EQU UPPER

	LOWER	LEVEL	UPPER	MARGINAL
1	-INF	0.100	0.200	.
2	-INF	0.100	0.200	.
3	-INF	0.100	0.200	.
4	-INF	14.500	14.800	.
5	-INF	0.100	0.200	.
6	-INF	0.100	0.200	.
7	-INF	6.500	6.500	.
8	-INF	5.000	8.200	.
9	-INF	26.500	26.500	0.010
10	-INF	0.100	0.200	.
11	-INF	0.100	0.200	.
12	-INF	20.000	27.800	.
13	-INF	3.000	5.800	.
14	-INF	1.000	2.000	.
15	-INF	0.100	0.200	.
16	-INF	0.100	0.200	.
17	-INF	0.500	1.000	.
18	-INF	0.100	0.200	.
19	-INF	0.500	1.000	.
20	-INF	0.400	0.600	.
21	-INF	0.100	0.200	.
22	-INF	0.125	0.200	.
23	-INF	0.100	0.200	.
24	-INF	6.000	10.000	.
25	-INF	0.171	0.200	.
26	-INF	1.246	2.000	.
27	-INF	0.100	0.200	.
28	-INF	0.100	0.200	.
29	-INF	0.200	0.200	0.301
30	-INF	7.200	7.200	.

31	-INF	0.100	0.200	.
32	-INF	5.000	9.100	.
33	-INF	15.000	26.500	.
34	-INF	4.000	7.400	.
35	-INF	4.000	7.920	.
36	-INF	0.500	1.000	.
37	-INF	4.000	6.300	.
38	-INF	0.200	0.200	1.328
39	-INF	4.129	4.600	.
40	-INF	0.100	0.200	.
41	-INF	0.100	0.200	.
42	-INF	1.000	2.300	.
43	-INF	4.177	5.600	.
44	-INF	0.100	0.200	.
45	-INF	0.125	0.200	.
46	-INF	0.100	0.200	.
47	-INF	5.000	9.300	.
48	-INF	0.100	0.200	.
49	-INF	1.000	2.200	.
50	-INF	0.100	0.200	.
51	-INF	0.300	0.400	.
52	-INF	2.000	3.000	.
53	-INF	10.000	16.000	.
54	-INF	5.000	9.200	.
55	-INF	0.100	0.200	.
56	-INF	0.200	0.200	0.389
57	-INF	0.100	0.200	.
58	-INF	0.100	0.200	.
59	-INF	0.100	0.200	.
60	-INF	0.100	0.200	.
61	-INF	2.000	3.000	.
62	-INF	2.500	2.500	.
63	-INF	5.000	9.200	.
64	-INF	0.100	0.200	.
65	-INF	0.100	0.200	.
66	-INF	3.180	4.600	.
67	-INF	0.100	0.150	.
68	-INF	2.500	2.500	.
69	-INF	5.000	10.800	.
70	-INF	0.100	0.200	.
71	-INF	0.200	0.200	.
72	-INF	0.200	0.200	.
73	-INF	0.100	0.200	.
74	-INF	1.000	1.100	.
75	-INF	0.100	0.200	.
76	-INF	3.000	5.200	.
77	-INF	0.100	0.200	.
78	-INF	0.100	0.200	.
79	-INF	0.100	0.200	.
80	-INF	179.102	235.600	.
81	-INF	5.500	6.600	.
82	-INF	0.100	0.200	.
83	-INF	0.100	0.200	.
84	-INF	1.000	1.200	.
85	-INF	0.100	0.200	.
86	-INF	3.236	3.500	.
87	-INF	0.100	0.200	.
88	-INF	0.100	0.200	.
89	-INF	20.000	26.200	.
90	-INF	0.950	1.000	.
91	-INF	0.100	0.200	.
92	-INF	0.100	0.200	.
93	-INF	10.000	15.000	.
94	-INF	1.000	2.200	.
95	-INF	0.100	0.200	.
96	-INF	20.000	30.000	.
97	-INF	0.125	0.200	.
98	-INF	4.000	6.500	.
99	-INF	0.600	1.000	.
100	-INF	3.000	4.600	.
101	-INF	0.100	0.200	.
102	-INF	0.100	0.200	.
103	-INF	0.100	0.200	.

---- EQU LOWER

	LOWER	LEVEL	UPPER	MARGINAL
1	0.100	0.100	+INF	-4.215
2	0.100	0.100	+INF	-8.454
3	0.100	0.100	+INF	-1.228
4	10.000	14.500	+INF	.
5	0.100	0.100	+INF	-2.617
6	0.100	0.100	+INF	.
7	4.000	6.500	+INF	.
8	5.000	5.000	+INF	-2.084
9	15.000	26.500	+INF	.
10	0.100	0.100	+INF	.
11	0.100	0.100	+INF	.
12	20.000	20.000	+INF	-0.060
13	3.000	3.000	+INF	-1.058
14	1.000	1.000	+INF	-8.992
15	0.100	0.100	+INF	.
16	0.100	0.100	+INF	-1.448
17	0.500	0.500	+INF	-0.703
18	0.100	0.100	+INF	-2.573
19	0.500	0.500	+INF	-1.266
20	0.400	0.400	+INF	.
21	0.100	0.100	+INF	.
22	0.100	0.125	+INF	.
23	0.100	0.100	+INF	.
24	6.000	6.000	+INF	-1.027
25	0.100	0.171	+INF	.
26	1.000	1.246	+INF	.
27	0.100	0.100	+INF	-3.992
28	0.100	0.100	+INF	.
29	0.100	0.200	+INF	.

30	4.000	7.200	+INF	.
31	0.100	0.100	+INF	.
32	5.000	5.000	+INF	-0.508
33	15.000	15.000	+INF	-0.037
34	4.000	4.000	+INF	-0.787
35	4.000	4.000	+INF	-0.468
36	0.500	0.500	+INF	.
37	4.000	4.000	+INF	-4.012
38	0.100	0.200	+INF	.
39	3.000	4.129	+INF	.
40	0.100	0.100	+INF	-44.556
41	0.100	0.100	+INF	.
42	1.000	1.000	+INF	.
43	3.000	4.177	+INF	.
44	0.100	0.100	+INF	.
45	0.100	0.125	+INF	.
46	0.100	0.100	+INF	.
47	5.000	5.000	+INF	-0.466
48	0.100	0.100	+INF	.
49	1.000	1.000	+INF	-1.560
50	0.100	0.100	+INF	.
51	0.300	0.300	+INF	-2.003
52	2.000	2.000	+INF	.
53	10.000	10.000	+INF	.
54	5.000	5.000	+INF	-2.087
55	0.100	0.100	+INF	-21.712
56	0.100	0.200	+INF	.
57	0.100	0.100	+INF	.
58	0.100	0.100	+INF	.
59	0.100	0.100	+INF	.
60	0.100	0.100	+INF	-1.851
61	2.000	2.000	+INF	-2.607
62	2.000	2.500	+INF	.
63	5.000	5.000	+INF	-1.160
64	0.100	0.100	+INF	-22.465
65	0.100	0.100	+INF	.
66	3.000	3.180	+INF	.
67	0.100	0.100	+INF	.
68	1.000	2.500	+INF	.
69	5.000	5.000	+INF	-0.407
70	0.100	0.100	+INF	-15.208
71	0.100	0.200	+INF	.
72	0.100	0.200	+INF	.
73	0.100	0.100	+INF	-6.751
74	1.000	1.000	+INF	-22.307
75	0.100	0.100	+INF	-7.981
76	3.000	3.000	+INF	.
77	0.100	0.100	+INF	.
78	0.100	0.100	+INF	-5.152
79	0.100	0.100	+INF	-0.920
80	150.000	179.102	+INF	.
81	4.000	5.500	+INF	.
82	0.100	0.100	+INF	-10.873
83	0.100	0.100	+INF	.
84	1.000	1.000	+INF	-31.705
85	0.100	0.100	+INF	.
86	2.000	3.236	+INF	.
87	0.100	0.100	+INF	-3.944
88	0.100	0.100	+INF	.
89	20.000	20.000	+INF	.
90	0.500	0.950	+INF	.
91	0.100	0.100	+INF	.
92	0.100	0.100	+INF	.
93	10.000	10.000	+INF	-0.074
94	1.000	1.000	+INF	-1.119
95	0.100	0.100	+INF	.
96	20.000	20.000	+INF	.
97	0.100	0.125	+INF	.
98	4.000	4.000	+INF	-0.061
99	0.600	0.600	+INF	-3.872
100	3.000	3.000	+INF	-0.830
101	0.100	0.100	+INF	-5.115
102	0.100	0.100	+INF	-8.807
103	0.100	0.100	+INF	.

	LOWER	LEVEL	UPPER	MARGINAL
---- EQU EQECONS	-INF	1700.000	1700.000	0.074
---- EQU DEPREC	-INF	55.000	55.000	1.752
---- EQU SUPPORT1	.	1.677	+INF	.
---- EQU SUPPORT2	.	.	+INF	-1.393
---- EQU SUPPORT3	.	0.150	+INF	.
---- EQU SUPPORT4	.	.	+INF	-1.457
---- EQU SUPPORT5	.	.	+INF	-26.212
---- EQU SUPPORT6	.	0.013	+INF	.
---- EQU SUPPORT7	.	0.050	+INF	.
---- EQU MEGA61631	-4.056	-4.056	-4.056	0.880
---- EQU MEGA61632	-0.116	-0.116	-0.116	-1.658
---- EQU MEGA61633	6.631	6.631	6.631	-0.071
---- EQU MEGA61634	-0.116	-0.116	-0.116	-10.168

---- EQU MEGA61635	-0.781	-0.781	-0.781	.
---- EQU MEGA61636	-1.562	-1.562	-1.562	.
---- EQU MEGA6141	-0.061	-0.061	-0.061	-5.991
---- EQU MEGA6142	-0.061	-0.061	-0.061	-5.356
---- EQU MEGA6143	-0.061	-0.061	-0.061	-3.863
---- EQU MEGA6144	-0.061	-0.061	-0.061	-36.777
---- EQU MEGA6145	0.304	0.304	0.304	.
---- EQU MEGA6146	-0.061	-0.061	-0.061	-3.056
---- EQU MEGA6321	-0.028	-0.028	-0.028	2.041
---- EQU MEGA6322	0.799	0.799	0.799	-0.132
---- EQU MEGA6323	-0.028	-0.028	-0.028	12.616
---- EQU MEGA6324	0.275	0.275	0.275	.
---- EQU MEGA6325	-0 097	-0.097	-0.097	-0.232
---- EQU MEGA6326	-0.921	-0.921	-0.921	0.018
---- EQU MEGA49511	-0.383	-0.383	-0.383	6.578
---- EQU MEGA49512	-0.038	-0.038	-0.038	-1.130
---- EQU MEGA49513	-0.043	-0.043	-0.043	.
---- EQU MEGA49514	-0.038	-0.038	-0.038	-3.470
---- EQU MEGA49515	-0.038	-0.038	-0.038	-1.694
---- EQU MEGA49516	-0.038	-0.038	-0.038	-3.980
---- EQU MEGA49517	-0.038	-0.038	-0.038	-6.017
---- EQU MEGA49518	0.617	0.617	0.617	-1.558
---- EQU MEGA6311	-0.071	-0.071	-0.071	-2.470
---- EQU MEGA6312	-0.071	-0.071	-0.071	-12.509
---- EQU MEGA6313	-0.235	-0.235	-0.235	.
---- EQU MEGA6314	-0.071	-0.071	-0.071	-13.777
---- EQU MEGA6315	-0.071	-0.071	-0.071	-12.722
---- EQU MEGA6316	-0.071	-0.071	-0.071	-1.785
---- EQU MEGA6317	0.659	0.659	0.659	31.219
---- EQU MEGA6318	-0.071	-0.071	-0.071	-4.792
---- EQU MEGA4921	0.754	0.754	0.754	.
---- EQU MEGA4922	0.138	0.138	0.138	-2.210
---- EQU MEGA4923	-0.031	-0.031	-0.031	-7.219
---- EQU MEGA4924	-0.156	-0.156	-0.156	-2.748
---- EQU MEGA4925	-0.705	-0.705	-0.705	-1.617
---- EQU MEGA5121	.	.	.	0.756
---- EQU MEGA5122	.	.	.	7.751
---- EQU MEGA5123	.	.	.	.
---- EQU MEGA6121	.	.	.	4.875
---- EQU MEGA6122	.	.	.	12.051
---- EQU MEGA6123	.	.	.	.
---- EQU MEGA6131	0.004	0.004	0.004	.
---- EQU MEGA6132	-0.002	-0.002	-0.002	-2.696
---- EQU MEGA6133	-0.002	-0.002	-0.002	-2.623
---- EQU MEGA411	-1.39E-17	-1.39E-17	-1.39E-17	-0.114
---- EQU MEGA412	-1.39E-17	-1.39E-17	-1.39E-17	.
---- EQU MEGA6111	0.056	0.056	0.056	-4.282
---- EQU MEGA6112	-0.056	-0.056	-0.056	.
---- EQU MEGA5111	-0.082	-0.082	-0.082	-0.718
---- EQU MEGA5112	0.082	0.082	0.082	.
---- EQU MEGA901	-1.39E-17	-1.39E-17	-1.39E-17	11.968
---- EQU MEGA902	-1.39E-17	-1.39E-17	-1.39E-17	.
---- EQU CAP1	60.000	60.000	+INF	-0.113
---- EQU CAP2	4.500	4.500	+INF	-0.490

---- EQU CAP3	10.000	10.000	+INF	-0.359
---- EQU EQE1	90.000	90.000	+INF	-0.166
---- EQU DEP1	5.000	5.000	+INF	-3.200
---- EQU DEP2	1.500	1.500	+INF	-2.158
---- EQU SUPPORT8	.	0.090	+INF	.
---- EQU SUPPORT9	.	0.090	+INF	.
---- EQU SUPPORT10	.	.	+INF	.

---- VAR X

	LOWER	LEVEL	UPPER	MARGINAL
1	-INF	0.100	+INF	.
2	-INF	0.100	+INF	.
3	-INF	0.100	+INF	.
4	-INF	14.500	+INF	.
5	-INF	0.100	+INF	.
6	-INF	0.100	+INF	.
7	-INF	6.500	+INF	.
8	-INF	5.000	+INF	.
9	-INF	26.500	+INF	.
10	-INF	0.100	+INF	.
11	-INF	0.100	+INF	.
12	-INF	20.000	+INF	.
13	-INF	3.000	+INF	.
14	-INF	1.000	+INF	.
15	-INF	0.100	+INF	.
16	-INF	0.100	+INF	.
17	-INF	0.500	+INF	.
18	-INF	0.100	+INF	.
19	-INF	0.500	+INF	.
20	-INF	0.400	+INF	.
21	-INF	0.100	+INF	.
22	-INF	0.125	+INF	.
23	-INF	0.100	+INF	.
24	-INF	6.000	+INF	.
25	-INF	0.171	+INF	.
26	-INF	1.246	+INF	.
27	-INF	0.100	+INF	.
28	-INF	0.100	+INF	.
29	-INF	0.200	+INF	.
30	-INF	7.200	+INF	.
31	-INF	0.100	+INF	.
32	-INF	5.000	+INF	.
33	-INF	15.000	+INF	.
34	-INF	4.000	+INF	.
35	-INF	4.000	+INF	.
36	-INF	0.500	+INF	.
37	-INF	4.000	+INF	.
38	-INF	0.200	+INF	.
39	-INF	4.129	+INF	.
40	-INF	0.100	+INF	.
41	-INF	0.100	+INF	.
42	-INF	1.000	+INF	.
43	-INF	4.177	+INF	.
44	-INF	0.100	+INF	.
45	-INF	0.125	+INF	.
46	-INF	0.100	+INF	.
47	-INF	5.000	+INF	.
48	-INF	0.100	+INF	.
49	-INF	1.000	+INF	.
50	-INF	0.100	+INF	.
51	-INF	0.300	+INF	.
52	-INF	2.000	+INF	.
53	-INF	10.000	+INF	.
54	-INF	5.000	+INF	.
55	-INF	0.100	+INF	.
56	-INF	0.200	+INF	.
57	-INF	0.100	+INF	.
58	-INF	0.100	+INF	.
59	-INF	0.100	+INF	.
60	-INF	0.100	+INF	.
61	-INF	2.000	+INF	.
62	-INF	2.500	+INF	.
63	-INF	5.000	+INF	.
64	-INF	0.100	+INF	.
65	-INF	0.100	+INF	.
66	-INF	3.180	+INF	.
67	-INF	0.100	+INF	.
68	-INF	2.500	+INF	.
69	-INF	5.000	+INF	.
70	-INF	0.100	+INF	.
71	-INF	0.200	+INF	.
72	-INF	0.200	+INF	.
73	-INF	0.100	+INF	.
74	-INF	1.000	+INF	.
75	-INF	0.100	+INF	.
76	-INF	3.000	+INF	.
77	-INF	0.100	+INF	.
78	-INF	0.100	+INF	.
79	-INF	0.100	+INF	.
80	-INF	179.102	+INF	.
81	-INF	5.500	+INF	.
82	-INF	0.100	+INF	.
83	-INF	0.100	+INF	.
84	-INF	1.000	+INF	.
85	-INF	0.100	+INF	.
86	-INF	3.236	+INF	.

87	-INF	0.100	+INF	.	96	-INF	20.000	+INF	.
88	-INF	0.100	+INF	.	97	-INF	0.125	+INF	.
89	-INF	20.000	+INF	.	98	-INF	4.000	+INF	.
90	-INF	0.950	+INF	.	99	-INF	0.600	+INF	.
91	-INF	0.100	+INF	.	100	-INF	3.000	+INF	.
92	-INF	0.100	+INF	.	101	-INF	0.100	+INF	.
93	-INF	10.000	+INF	.	102	-INF	0.100	+INF	.
94	-INF	1.000	+INF	.	103	-INF	0.100	+INF	.
95	-INF	0.100	+INF	.					

	LOWER	LEVEL	UPPER	MARGINAL
---- VAR Z	-INF	36.374	+INF	.

**** REPORT SUMMARY : 0 NONOPT
0 INFEASIBLE
0 UNBOUNDED

---- 700 VARIABLE X.L

1 0.100, 2 0.100, 3 0.100, 4 14.500, 5 0.100
6 0.100, 7 6.500, 8 5.000, 9 26.500, 10 0.100
11 0.100, 12 20.000, 13 3.000, 14 1.000, 15 0.100
16 0.100, 17 0.500, 18 0.100, 19 0.500, 20 0.400
21 0.100, 22 0.125, 23 0.100, 24 6.000, 25 0.171
26 1.246, 27 0.100, 28 0.100, 29 0.200, 30 7.200
31 0.100, 32 5.000, 33 15.000, 34 4.000, 35 4.000
36 0.500, 37 4.000, 38 0.200, 39 4.129, 40 0.100
41 0.100, 42 1.000, 43 4.177, 44 0.100, 45 0.125
46 0.100, 47 5.000, 48 0.100, 49 1.000, 50 0.100
51 0.300, 52 2.000, 53 10.000, 54 5.000, 55 0.100
56 0.200, 57 0.100, 58 0.100, 59 0.100, 60 0.100
61 2.000, 62 2.500, 63 5.000, 64 0.100, 65 0.100
66 3.180, 67 0.100, 68 2.500, 69 5.000, 70 0.100
71 0.200, 72 0.200, 73 0.100, 74 1.000, 75 0.100
76 3.000, 77 0.100, 78 0.100, 79 0.100, 80 179.102
81 5.500, 82 0.100, 83 0.100, 84 1.000, 85 0.100
86 3.236, 87 0.100, 88 0.100, 89 20.000, 90 0.950
91 0.100, 92 0.100, 93 10.000, 94 1.000, 95 0.100
96 20.000, 97 0.125, 98 4.000, 99 0.600, 100 3.000
101 0.100, 102 0.100, 103 0.100

---- 700 VARIABLE X.M

(ALL 0.000)

EXECUTION TIME = 2.310 SECONDS VERID MW2-00-051

E.3 Results of Direct Allocation Without Dependency Constraints

1	0.100,	2	0.100,	3	0.100,	4	10.000,	5	0.100
6	0.100,	7	6.500,	8	5.000,	9	26.500,	10	0.100
11	0.100,	12	20.000,	13	3.000,	14	1.000,	15	0.100
16	0.100,	17	0.500,	18	0.100,	19	0.500,	20	0.400
21	0.100,	22	0.100,	23	0.100,	24	6.000,	25	0.100
26	1.000,	27	0.100,	28	0.100,	29	0.150,	30	7.200
31	0.158,	32	5.000,	33	15.000,	34	5.000,	35	4.000
36	0.500,	37	4.000,	38	0.200,	39	3.000,	40	0.100
41	0.100,	42	1.000,	43	3.000,	44	0.158,	45	0.100
46	0.100,	47	5.000,	48	0.100,	49	1.000,	50	0.100
51	0.300,	52	2.000,	53	13.463,	54	5.000,	55	0.100
56	0.100,	57	0.100,	58	0.100,	59	0.158,	60	0.100
61	2.000,	62	2.500,	63	5.000,	64	0.100,	65	0.158
66	3.000,	67	0.100,	68	2.500,	69	5.000,	70	0.100
71	0.200,	72	0.200,	73	0.100,	74	1.000,	75	0.100
76	3.000,	77	0.100,	78	0.100,	79	0.100,	80	198.186
81	4.000,	82	0.100,	83	0.158,	84	1.115,	85	0.100
86	2.000,	87	0.100,	88	0.158,	89	20.000,	90	1.000
91	0.100,	92	0.100,	93	10.000,	94	1.000,	95	0.100
96	20.000,	97	0.140,	98	4.000,	99	0.600,	100	3.000
101	0.100,	102	0.100,	103	0.100				