

OPTIMAL REGIONAL WATER QUALITY MANAGEMENT BY
MULTILEVEL APPROACH AND THE DISCRETE MAXIMUM PRINCIPLE

by

SUDHAKAR REDDY PAIDY

B.Tech.(Chemical Engineering), Osmania University,
Hyderabad, India, 1973.

A MASTER'S THESIS

submitted in partial fulfillment of the
requirements for the degree

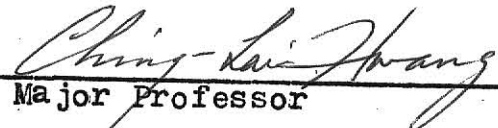
MASTER OF SCIENCE

Department of Industrial Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

1976

Approved by:


Major Professor

LD
2668
TY
1976
P34
C. Z
Document

TABLE OF CONTENTS

206

	PAGE
LIST OF TABLES	
LIST OF FIGURES	
ACKNOWLEDGEMENT	
1. INTRODUCTION	1
2. WATER RESOURCES MANAGEMENT MODELING	4
3. DECENTRALIZED DECISION PROCESSES	10
3.1 Formulation of the problem	
3.2 Multilevel Optimization Approach	
3.2.1 Pseudo decentralized process	
3.2.2 Complete decentralized process	
4. OPTIMIZATION BY DISCRETE MAXIMUM PRINCIPLE	30
4.1 Formulation of the problem	
4.2 Computational Procedure and Results	
5. THREE LEVEL APPROACH TO INTRODUCE A REGIONAL TREATMENT PLANT	38
5.1 Problem definition	
5.2 Problem formulation	
5.3 Analysis of three level structure	
6. CONCLUSION	52
REFERENCES	54
APPENDIX I. Comparison of the multilevel approach and discrete maximum principle	58
APPENDIX II. Computer program for pseudo decentralized process by two level approach	76
APPENDIX III. Algorithm of the complete decentralized process	86

APPENDIX IV. Computer program for pseudo
decentralized process by the discrete maximum
principle

102

LIST OF TABLES

TABLE

1. Matrix representation of the constraint set for Miami river
2. Coefficients in quadratic cost function (3-2) for the fifteen Miami river polluters
3. Computational results for the Pseudo Decentralized model
4. Polluters' response for some tax structures
5. Computational results for the Complete Decentralized model
6. Comparison of results for the Pseudo Decentralized model
7. Description of variables in equation (5-3)
8. Comparison of three level vs. two level solutions
9. Explanation of Fortran symbols in the computer program for pseudo decentralized process by two level approach
10. Explanation of Fortran symbols in computer program for the complete decentralized process
11. Explanation of Fortran symbols in the computer program for pseudo decentralized process by the discrete maximum principle.

LIST OF FIGURES

FIGURE

1. Schematic representation of a Water Resources System
2. A typical subsystem of Water Resources System
3. The Miami River (OHIO) decomposition structure
4. Multilevel structure of the problem
5. The cost functions for various treatment plant sizes
6. Regional plant location for Miami river
7. Three level structure
8. Pseudo decentralized process by two level approach
9. Complete decentralized process
10. Pseudo decentralized process by discrete maximum principle

ACKNOWLEDGEMENT

The author wishes to express his sincere gratitude to his major professor, Dr. C. L. Hwang, for his advice, encouragement, constructive criticism and personal interest taken in the preparation of this thesis.

The author is grateful to Drs. Stephan Konz and Dean Eckhoff for serving on his advisory committee.

1. INTRODUCTION

The objective pursued in water quality management is to secure water qualities such that the marginal costs of cleaning the water equals the accruing marginal benefits. A system of effluent charges or taxes can be used to induce polluters to reduce the waste discharge.

The problem considered in this study is simultaneously finding an optimal waste water treatment configuration for meeting water quality standards along a river basin and determining the optimal pollution taxes to achieve this configuration by using decomposition and the multilevel approach.

Literature Survey

A thorough review and systematic classification of literature on optimal decision models of regional water quality management is given by Bundgaard-Nielsen and Hwang (1976).

The application of effluent charges to regulate waste discharge essentially is based upon a decomposition of the decision hierarchy involved. The decomposition idea is developed into the hierarchical multilevel decision models presented by Haines (1971, 1972), Haines, et al. (1968, 1971, 1972a, b) and Foley and Haines (1973).

The multilevel approach (Lasdon, 1970; Measarovic, et al., 1970; Wismer, 1971) is based on the decomposition of large and complex systems, and then the reconstruction of a new system composed of independent subsystems. This decomposition is accomplished by introducing pseudo variables into the system. Each subsystem is separately and independently optimized (first level solution). The optimized subsystems

are joined together by coupling variables that are manipulated by second level controllers to arrive at the optimal solution of the whole system (second level solution). This coordination may be structured into more than two levels.

Computational results to implement the multilevel algorithms are presented by Haimes and his coworkers for linear constraints (Haimes, et al., 1972a), nonlinear constraints (Foley and Haimes, 1973), and for the possibility of using a centralized waste treatment plant and user charges (Haimes, et al., 1972b).

Objectives

The objective of this study is to apply the hierarchical multilevel approach to a basin wide problem. A two-level hierarchy has been applied to solve a basin wide problem, by at-source treatment and by augmenting the river flow, using effluent charges. Two types of second level coordination schemes are presented for the decentralized decision making process. One assumes a knowledge of the local treatment cost functions by the regional authority (pseudo decentralized process); the other assumes no such knowledge (complete decentralized process). The data for the Miami River in Ohio, with fifteen biological oxygen demand (BOD) dischargers, are used. The system model is given in terms of linear inequality constraints and a quadratic objective function. A three level approach to introduce a regional treatment plant into the river basin also is illustrated.

The mathematical analogy between the two level approach and the discrete maximum principle is presented. To illustrate

the analogy, the Miami river basin problem is solved again for the case of the pseudo decentralized process. The analogy is used to deduce the effluent charges from the solution by the discrete maximum principle.

2. WATER RESOURCES MANAGEMENT MODELING

In the modeling and control of the pollution of a water resources system (WRS) various pollution inputs have to be considered. Suspended organic and inorganic materials, algae, and temperature, are a few of the factors that affect the water characteristics and properties. Each pollution input, municipal or industrial, has its own 'quality' or characteristics and accordingly contributes to the quality of total water resources system. It is the consideration of all these inputs to the WRS where each input consists of a vector of pollution characteristics that makes the modeling and control difficult and often intractable. In addition, the consideration of the environmental, economical, social, political and other non-commensurate aspects is inescapable.

The evolution of mathematical models for water quality control systems has posed two major problems. The first is the lack of data critical in the estimation of model parameters. Data availability usually determines the model form used and its appropriateness. Moreover, the significant variations among the characteristics of different streams and estuaries necessitate the collection of real data in every specific case (Thomann, 1972). The second problem is the complexity of the mathematical representation of the system and therefore the difficulties in finding the solution to the model equations. A mathematical model is an abstraction of the real world, and as such, incapable of taking all factors into account. However, it should be a fair representation of the real system with sufficient generality and flexibility.

The extent to which the mathematical formulation can become complex is limited by the available techniques to solve the model equations.

A general mathematical model is presented here. Each case where the pollution effluent is discharged directly into the river, into the lake, or into a bypass pipe leading to an advanced waste water treatment (AWT) plant constitutes a special case of the general model. Specific models to represent a section of Miami river in Ohio are dealt in the later sections.

It is convenient to represent the water resources system as a chain of N small segments or subsystems, where the beginning of each segment or reach has a source of pollution associated with it and/or a change in the hydrological characteristics of the stream. These changes might arise because of surface area exposure, turbidity, or temperature changes, or a tributary inflow. So, mathematically, this is a sequential process of N stages. To simplify the methodology, no branching in the WRS is assumed to occur. A schematic representation of such system is shown in Figure 1. Figure 2 depicts a typical subsystem of the water resources system.

Let U^n be the n^{th} pollution input to the WRS with N reaches. The input vector $(U^n)^T$ is $(U_1^n, U_2^n, \dots, U_m^n)$ where U_1^n represents the water quantity and $U_2^n, U_3^n, \dots, U_m^n$ represent different water characteristics such as BOD, DO, pH, conductivity, temperature, algae, phosphates, nitrates, etc.

Let S^n be the output vector of the n^{th} subsystem leaving the WRS. Examples of such output are heat loss, reoxygenation by aeration, intakes for industries, etc. Let W^n

**THIS BOOK
CONTAINS
NUMEROUS PAGES
WITH DIAGRAMS
THAT ARE CROOKED
COMPARED TO THE
REST OF THE
INFORMATION ON
THE PAGE.**

**THIS IS AS
RECEIVED FROM
CUSTOMER.**

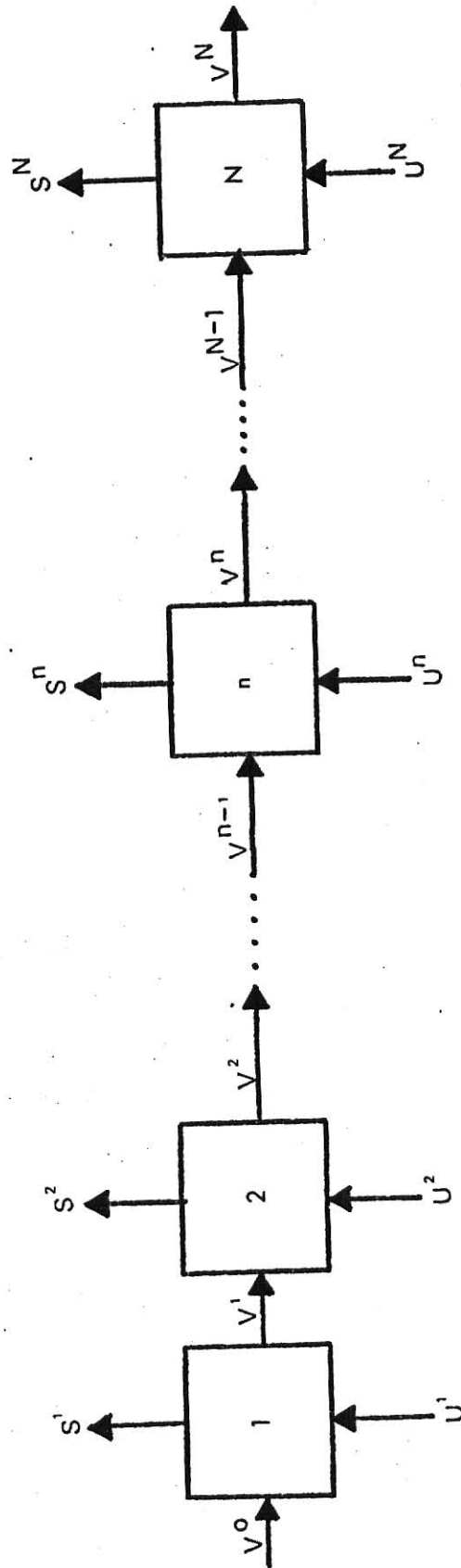


Figure 1. Schematic representation of a water resources system

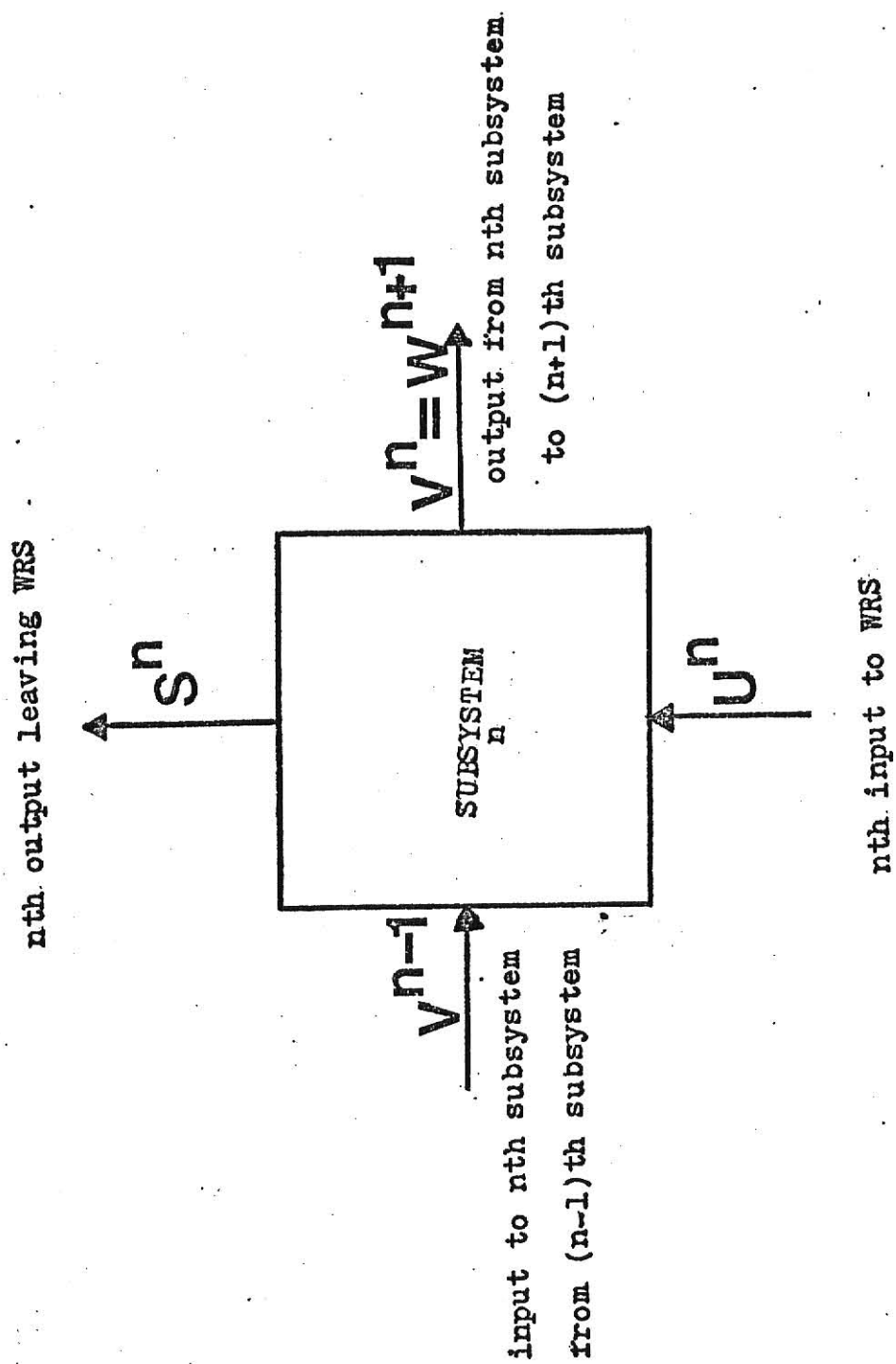


Figure 2. A typical subsystem of water resources system

and V^n be the input and output vectors, respectively, entering and leaving the n th subsystem from and to other subsystems. Since no branching is assumed, the output vector of one subsystem is the input vector only to the following subsystem. The vectors S^n , W^n and V^n are of the same dimension (m) as U^n . The corresponding elements in all the vectors represent the same characteristic. The vector sum $W^n + U^n$ is meaning less and does not equal to $V^n + S^n$. The subsystems' outputs are assumed to be represented by:

$$V^n = T^n(U^n, W^n)$$

$$S^n = \phi^n(U^n, W^n)$$

where

$$\begin{aligned} (T^n)^T &= (T_1^n, T_2^n, \dots, T_m^n) \\ (\phi^n)^T &= (\phi_1^n, \phi_2^n, \dots, \phi_m^n) \end{aligned}$$

Let

$$R^n(U^n, W^n, S^n) \geq 0, \quad n = 1, 2, \dots, N$$

be the vector of constraints to be satisfied by the n th subsystem, where R^n 's are k -dimensional continuous functions. These constraints include the equality constraints, if any, and represent the physical, legal, economical and other system constraints.

The overall WRS treatment cost function, F is the sum of the treatment cost functions of each subsystem $F^n(U^n, S^n)$, i.e.,

$$F = \sum_{n=1}^N F^n(U^n, S^n)$$

where the cost function F^n for subsystem n is assumed to be an explicit function of the effluent input U^n and the output S^n . A typical optimization problem associated with such water resources pollution problem can, now, be stated as:

$$\text{Minimize } F = \sum_{n=1}^N F^n(U^n, S^n) \quad (2-1)$$

subject to:

$$V^n = T^n(U^n, W^n) \quad (2-2)$$

$$S^n = Q^n(U^n, W^n) \quad (2-3)$$

$$W^{n+1} = V^n \quad (2-4)$$

$$R^n(U^n, W^n, S^n) \geq 0 \quad (2-5)$$

$$n = 1, 2, \dots, N$$

where $W^{N+1} = V^N$ is the vector representing the characteristics of the water leaving the WRS.

The problem can be viewed as finding a set of decision vectors U^n such that the total cost F is minimized subject to the constraints R^n . However, the process is described by the performance equations (2-2), (2-3) and (2-4). The variables V^n (or W^n) can be said to be the state variables with respect to U^n being the decision variables. The problem of finding an optimal decision vector U^n which satisfies at least the necessary conditions for a maximum or minimum is called the integrated problem.

Appendix I gives the general solution procedures to the optimization problem, as given by equations (2-1) through (2-5), by the multilevel approach and by the discrete maximum principle.

3. DECENTRALIZED DECISION PROCESSES

3.1 Formulation of the problem

The following formulation of the mathematical model, for the Miami river in Ohio, with linear constraints and quadratic objective function is based on the work by Hass (1970) and as described by Haimes, et al. (1972 a).

The 58 mile portion of the Miami river above New Baltimore is segmented into twenty seven "reaches", fifteen of which are associated with inputs discharging organic wastes into the stream (Figure 3). Since data are available for the dissolved oxygen standards only, the model is a relatively simple one. The Streeter-Phelps equation (Streeter and Phelps, 1925) is utilized to transform the minimum dissolved oxygen standards for all the reaches into a set of linear inequality constraints. In this model, the concentrations of dissolved oxygen and biological oxygen demand are assumed homogeneous within each reach.

The constraints require that the supply of available oxygen for the organic decomposition process in each reach (that available above the quality standards requirement) must be equal to or exceed the demand imposed by BOD loads discharged into the reach and all reaches preceding it. Thus for each reach i it is required that

$$d_{i1} w_1(1-x_1) + d_{i2} w_2(1-x_2) + \dots + d_{ii} w_i(1-x_i) \leq Y b_i \quad (3-1)$$

where

w_j = gross BOD load introduced at the beginning of the j th reach that has a polluting input (lbs/day)

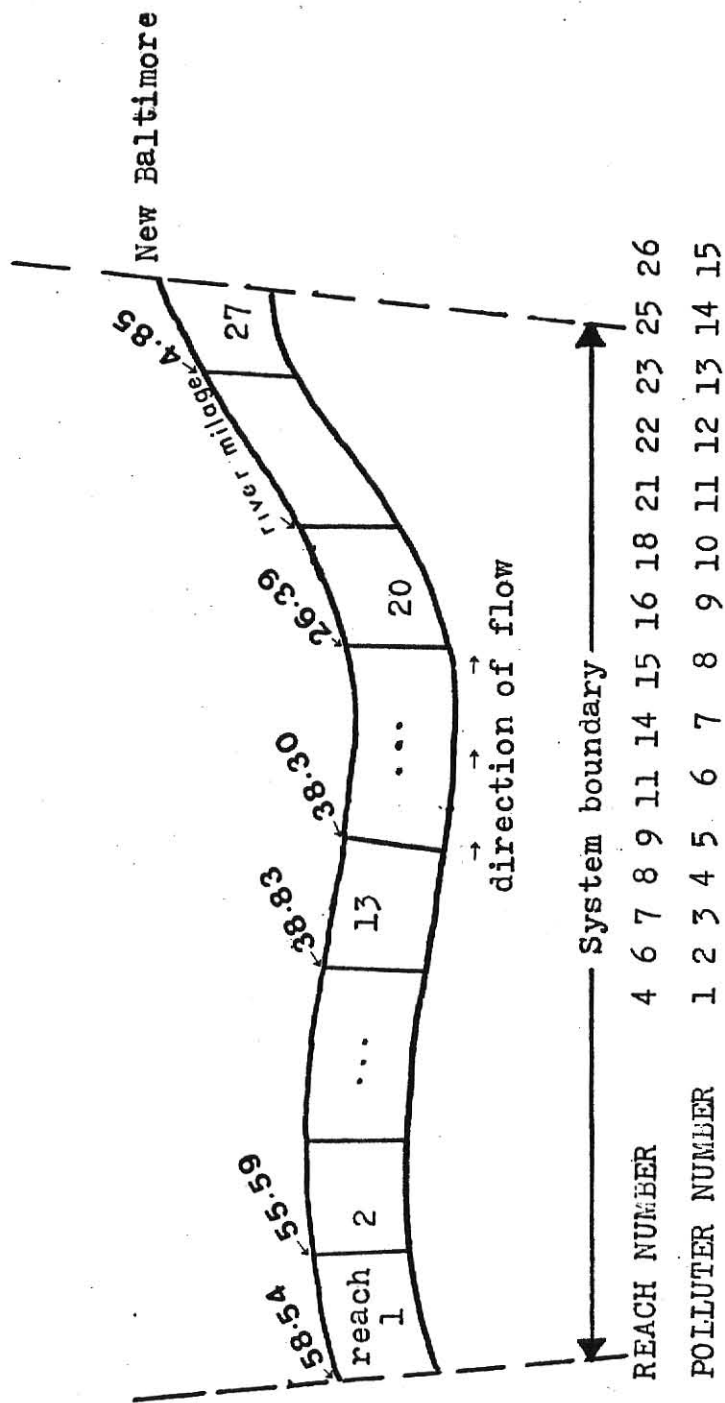


Figure 3. The Miami river (OHIO) decomposition structure

x_j = fraction of w_j removed through treatment by the j th polluter

d_{ij} = pounds of oxygen demanded in reach i by the decomposition of a pound of BOD discharged by the j th polluter

b_i = amount of DO available for decomposition process (total available less standard requirement) in reach i per unit of flow, pounds

Y = augmentation rate, ratio

The system of inequality constraints can be rewritten more compactly as follows:

$$a_{i1} x_1 + a_{i2} x_2 + \dots + a_{ij} x_j \geq S_i - Y b_i$$

where

$$a_{ij} = d_{ij} w_j$$

and

$$S_i = a_{i1} + a_{i2} + \dots + a_{ii}$$

In addition, other restrictions on x_j 's are

$$0.45 \leq x_j \leq 0.99, \quad j = 1, 2, \dots, 15$$

which insure that all discharges be subjected to at least primary treatment. In general, primary treatment involves chlorination, filtering, and settling to reduce the bacteria and remove bulky solids. Such sewage treatment usually removes about 45% of the gross BOD load. 99% is assumed to be the upper limit of treatment level. For the Miami river problem, the constants a_{ij} and b_i 's are available (Hass, 1970). Table 1 depicts the matrix representation of the constraint set.

Let $f_j(x_j)$ denote the treatment cost function of the j th polluter. The following cost functions for the fifteen polluters

Table 1. Matrix representation of constraint set for Miami river (Hass, 1970)

	z ₁	z ₂	z ₃	z ₄	z ₅	z ₆	z ₇	z ₈	z ₉	z ₁₀	z ₁₁	z ₁₂	z ₁₃	z ₁₄	z ₁₅	z ₁₆	z ₁₇	z ₁₈	z ₁₉	z ₂₀	z ₂₁	z ₂₂	z ₂₃	z ₂₄	z ₂₅	z ₂₆	z ₂₇	z ₂₈	z ₂₉	z ₃₀	z ₃₁	z ₃₂	z ₃₃	z ₃₄	z ₃₅	z ₃₆	z ₃₇	z ₃₈	z ₃₉	z ₄₀	z ₄₁	z ₄₂	z ₄₃	z ₄₄	z ₄₅	z ₄₆	z ₄₇	z ₄₈	z ₄₉	z ₅₀	z ₅₁	z ₅₂	z ₅₃	z ₅₄	z ₅₅	z ₅₆	z ₅₇	z ₅₈	z ₅₉	z ₆₀	z ₆₁	z ₆₂	z ₆₃	z ₆₄	z ₆₅	z ₆₆	z ₆₇	z ₆₈	z ₆₉	z ₇₀	z ₇₁	z ₇₂	z ₇₃	z ₇₄	z ₇₅	z ₇₆	z ₇₇	z ₇₈	z ₇₉	z ₈₀	z ₈₁	z ₈₂	z ₈₃	z ₈₄	z ₈₅	z ₈₆	z ₈₇	z ₈₈	z ₈₉	z ₉₀	z ₉₁	z ₉₂	z ₉₃	z ₉₄	z ₉₅	z ₉₆	z ₉₇	z ₉₈	z ₉₉	z ₁₀₀	z ₁₀₁	z ₁₀₂	z ₁₀₃	z ₁₀₄	z ₁₀₅	z ₁₀₆	z ₁₀₇	z ₁₀₈	z ₁₀₉	z ₁₁₀	z ₁₁₁	z ₁₁₂	z ₁₁₃	z ₁₁₄	z ₁₁₅	z ₁₁₆	z ₁₁₇	z ₁₁₈	z ₁₁₉	z ₁₂₀	z ₁₂₁	z ₁₂₂	z ₁₂₃	z ₁₂₄	z ₁₂₅	z ₁₂₆	z ₁₂₇	z ₁₂₈	z ₁₂₉	z ₁₃₀	z ₁₃₁	z ₁₃₂	z ₁₃₃	z ₁₃₄	z ₁₃₅	z ₁₃₆	z ₁₃₇	z ₁₃₈	z ₁₃₉	z ₁₄₀	z ₁₄₁	z ₁₄₂	z ₁₄₃	z ₁₄₄	z ₁₄₅	z ₁₄₆	z ₁₄₇	z ₁₄₈	z ₁₄₉	z ₁₅₀	z ₁₅₁	z ₁₅₂	z ₁₅₃	z ₁₅₄	z ₁₅₅	z ₁₅₆	z ₁₅₇	z ₁₅₈	z ₁₅₉	z ₁₆₀	z ₁₆₁	z ₁₆₂	z ₁₆₃	z ₁₆₄	z ₁₆₅	z ₁₆₆	z ₁₆₇	z ₁₆₈	z ₁₆₉	z ₁₇₀	z ₁₇₁	z ₁₇₂	z ₁₇₃	z ₁₇₄	z ₁₇₅	z ₁₇₆	z ₁₇₇	z ₁₇₈	z ₁₇₉	z ₁₈₀	z ₁₈₁	z ₁₈₂	z ₁₈₃	z ₁₈₄	z ₁₈₅	z ₁₈₆	z ₁₈₇	z ₁₈₈	z ₁₈₉	z ₁₉₀	z ₁₉₁	z ₁₉₂	z ₁₉₃	z ₁₉₄	z ₁₉₅	z ₁₉₆	z ₁₉₇	z ₁₉₈	z ₁₉₉	z ₂₀₀	z ₂₀₁	z ₂₀₂	z ₂₀₃	z ₂₀₄	z ₂₀₅	z ₂₀₆	z ₂₀₇	z ₂₀₈	z ₂₀₉	z ₂₁₀	z ₂₁₁	z ₂₁₂	z ₂₁₃	z ₂₁₄	z ₂₁₅	z ₂₁₆	z ₂₁₇	z ₂₁₈	z ₂₁₉	z ₂₂₀	z ₂₂₁	z ₂₂₂	z ₂₂₃	z ₂₂₄	z ₂₂₅	z ₂₂₆	z ₂₂₇	z ₂₂₈	z ₂₂₉	z ₂₃₀	z ₂₃₁	z ₂₃₂	z ₂₃₃	z ₂₃₄	z ₂₃₅	z ₂₃₆	z ₂₃₇	z ₂₃₈	z ₂₃₉	z ₂₄₀	z ₂₄₁	z ₂₄₂	z ₂₄₃	z ₂₄₄	z ₂₄₅	z ₂₄₆	z ₂₄₇	z ₂₄₈	z ₂₄₉	z ₂₅₀	z ₂₅₁	z ₂₅₂	z ₂₅₃	z ₂₅₄	z ₂₅₅	z ₂₅₆	z ₂₅₇	z ₂₅₈	z ₂₅₉	z ₂₆₀	z ₂₆₁	z ₂₆₂	z ₂₆₃	z ₂₆₄	z ₂₆₅	z ₂₆₆	z ₂₆₇	z ₂₆₈	z ₂₆₉	z ₂₇₀	z ₂₇₁	z ₂₇₂	z ₂₇₃	z ₂₇₄	z ₂₇₅	z ₂₇₆	z ₂₇₇	z ₂₇₈	z ₂₇₉	z ₂₈₀	z ₂₈₁	z ₂₈₂	z ₂₈₃	z ₂₈₄	z ₂₈₅	z ₂₈₆	z ₂₈₇	z ₂₈₈	z ₂₈₉	z ₂₉₀	z ₂₉₁	z ₂₉₂	z ₂₉₃	z ₂₉₄	z ₂₉₅	z ₂₉₆	z ₂₉₇	z ₂₉₈	z ₂₉₉	z ₃₀₀	z ₃₀₁	z ₃₀₂	z ₃₀₃	z ₃₀₄	z ₃₀₅	z ₃₀₆	z ₃₀₇	z ₃₀₈	z ₃₀₉	z ₃₁₀	z ₃₁₁	z ₃₁₂	z ₃₁₃	z ₃₁₄	z ₃₁₅	z ₃₁₆	z ₃₁₇	z ₃₁₈	z ₃₁₉	z ₃₂₀	z ₃₂₁	z ₃₂₂	z ₃₂₃	z ₃₂₄	z ₃₂₅	z ₃₂₆	z ₃₂₇	z ₃₂₈	z ₃₂₉	z ₃₃₀	z ₃₃₁	z ₃₃₂	z ₃₃₃	z ₃₃₄	z ₃₃₅	z ₃₃₆	z ₃₃₇	z ₃₃₈	z ₃₃₉	z ₃₄₀	z ₃₄₁	z ₃₄₂	z ₃₄₃	z ₃₄₄	z ₃₄₅	z ₃₄₆	z ₃₄₇	z ₃₄₈	z ₃₄₉	z ₃₅₀	z ₃₅₁	z ₃₅₂	z ₃₅₃	z ₃₅₄	z ₃₅₅	z ₃₅₆	z ₃₅₇	z ₃₅₈	z ₃₅₉	z ₃₆₀	z ₃₆₁	z ₃₆₂	z ₃₆₃	z ₃₆₄	z ₃₆₅	z ₃₆₆	z ₃₆₇	z ₃₆₈	z ₃₆₉	z ₃₇₀	z ₃₇₁	z ₃₇₂	z ₃₇₃	z ₃₇₄	z ₃₇₅	z ₃₇₆	z ₃₇₇	z ₃₇₈	z ₃₇₉	z ₃₈₀	z ₃₈₁	z ₃₈₂	z ₃₈₃	z ₃₈₄	z ₃₈₅	z ₃₈₆	z ₃₈₇	z ₃₈₈	z ₃₈₉	z ₃₉₀	z ₃₉₁	z ₃₉₂	z ₃₉₃	z ₃₉₄	z ₃₉₅	z ₃₉₆	z ₃₉₇	z ₃₉₈	z ₃₉₉	z ₄₀₀	z ₄₀₁	z ₄₀₂	z ₄₀₃	z ₄₀₄	z ₄₀₅	z ₄₀₆	z ₄₀₇	z ₄₀₈	z ₄₀₉	z ₄₁₀	z ₄₁₁	z ₄₁₂	z ₄₁₃	z ₄₁₄	z ₄₁₅	z ₄₁₆	z ₄₁₇	z ₄₁₈	z ₄₁₉	z ₄₂₀	z ₄₂₁	z ₄₂₂	z ₄₂₃	z ₄₂₄	z ₄₂₅	z ₄₂₆	z ₄₂₇	z ₄₂₈	z ₄₂₉	z ₄₃₀	z ₄₃₁	z ₄₃₂	z ₄₃₃	z ₄₃₄	z ₄₃₅	z ₄₃₆	z ₄₃₇	z ₄₃₈	z ₄₃₉	z ₄₄₀	z ₄₄₁	z ₄₄₂	z ₄₄₃	z ₄₄₄	z ₄₄₅	z ₄₄₆	z ₄₄₇	z ₄₄₈	z ₄₄₉	z ₄₅₀	z ₄₅₁	z ₄₅₂	z ₄₅₃	z ₄₅₄	z ₄₅₅	z ₄₅₆	z ₄₅₇	z ₄₅₈	z ₄₅₉	z ₄₆₀	z ₄₆₁	z ₄₆₂	z ₄₆₃	z ₄₆₄	z ₄₆₅	z ₄₆₆	z ₄₆₇	z ₄₆₈	z ₄₆₉	z ₄₇₀	z ₄₇₁	z ₄₇₂	z ₄₇₃	z ₄₇₄	z ₄₇₅	z ₄₇₆	z ₄₇₇	z ₄₇₈	z ₄₇₉	z ₄₈₀	z ₄₈₁	z ₄₈₂	z ₄₈₃	z ₄₈₄	z ₄₈₅	z ₄₈₆	z ₄₈₇	z ₄₈₈	z ₄₈₉	z ₄₉₀	z ₄₉₁	z ₄₉₂	z ₄₉₃	z ₄₉₄	z ₄₉₅	z ₄₉₆	z ₄₉₇	z ₄₉₈	z ₄₉₉	z ₅₀₀	z ₅₀₁	z ₅₀₂	z ₅₀₃	z ₅₀₄	z ₅₀₅	z ₅₀₆	z ₅₀₇	z ₅₀₈	z ₅₀₉	z ₅₁₀	z ₅₁₁	z ₅₁₂	z ₅₁₃	z ₅₁₄	z ₅₁₅	z ₅₁₆	z ₅₁₇	z ₅₁₈	z ₅₁₉	z ₅₂₀	z ₅₂₁	z ₅₂₂	z ₅₂₃	z ₅₂₄	z ₅₂₅	z ₅₂₆	z ₅₂₇	z ₅₂₈	z ₅₂₉	z ₅₃₀	z ₅₃₁	z ₅₃₂	z ₅₃₃	z ₅₃₄	z ₅₃₅	z ₅₃₆	z ₅₃₇	z ₅₃₈	z ₅₃₉	z ₅₄₀	z ₅₄₁	z ₅₄₂	z ₅₄₃	z ₅₄₄	z ₅₄₅	z ₅₄₆	z ₅₄₇	z ₅₄₈	z ₅₄₉	z ₅₅₀	z ₅₅₁	z ₅₅₂	z ₅₅₃	z ₅₅₄	z ₅₅₅	z ₅₅₆	z ₅₅₇	z ₅₅₈	z ₅₅₉	z ₅₆₀	z ₅₆₁	z ₅₆₂	z ₅₆₃	z ₅₆₄	z ₅₆₅	z ₅₆₆	z ₅₆₇	z ₅₆₈	z ₅₆₉	z ₅₇₀	z ₅₇₁	z ₅₇₂	z ₅₇₃	z ₅₇₄	z ₅₇₅	z ₅₇₆	z ₅₇₇	z ₅₇₈	z ₅₇₉	z ₅₈₀	z ₅₈₁	z ₅₈₂	z ₅₈₃	z ₅₈₄	z ₅₈₅	z ₅₈₆	z ₅₈₇	z ₅₈₈	z ₅₈₉	z ₅₉₀	z ₅₉₁	z ₅₉₂	z ₅₉₃	z ₅₉₄	z ₅₉₅	z ₅₉₆	z ₅₉₇	z ₅₉₈	z ₅₉₉	z ₆₀₀	z ₆₀₁	z ₆₀₂	z ₆₀₃	z ₆₀₄	z ₆₀₅	z ₆₀₆	z ₆₀₇	z ₆₀₈	z ₆₀₉	z ₆₁₀	z ₆₁₁	z ₆₁₂	z ₆₁₃	z ₆₁₄	z ₆₁₅	z ₆₁₆	z ₆₁₇	z ₆₁₈	z ₆₁₉	z ₆₂₀	z ₆₂₁	z ₆₂₂	z ₆₂₃	z ₆₂₄	z ₆₂₅	z ₆₂₆	z ₆₂₇	z ₆₂₈	z ₆₂₉	z ₆₃₀	z ₆₃₁	z ₆₃₂	z ₆₃₃	z ₆₃₄	z ₆₃₅	z ₆₃₆	z ₆₃₇	z ₆₃₈	z ₆₃₉	z ₆₄₀	z ₆₄₁	z ₆₄₂	z ₆₄₃	z ₆₄₄	z ₆₄₅	z ₆₄₆	z ₆₄₇	z ₆₄₈	z ₆₄₉	z ₆₅₀	z ₆₅₁	z ₆₅₂	z ₆₅₃	z ₆₅₄	z ₆₅₅	z ₆₅₆	z ₆₅₇	z ₆₅₈	z ₆₅₉	z ₆₆₀	z ₆₆₁	z ₆₆₂	z ₆₆₃	z ₆₆₄	z ₆₆₅	z ₆₆₆	z ₆₆₇	z ₆₆₈	z ₆₆₉	z ₆₇₀	z ₆₇₁	z ₆₇₂	z ₆₇₃	z ₆₇₄	z ₆₇₅	z ₆₇₆	z ₆₇₇	z ₆₇₈	z ₆₇₉	z ₆₈₀	z ₆₈₁	z ₆₈₂	z ₆₈₃	z ₆₈₄	z ₆₈₅	z ₆₈₆	z ₆₈₇	z ₆₈₈	z ₆₈₉	z ₆₉₀	z ₆₉₁	z ₆₉₂	z ₆₉₃	z ₆₉₄	z ₆₉₅	z ₆₉₆	z ₆₉₇	z ₆₉₈	z ₆₉₉	z ₇₀₀	z ₇₀₁	z ₇₀₂	z ₇₀₃	z ₇₀₄	z ₇₀₅	z ₇₀₆	z ₇₀₇	z ₇₀₈	z ₇₀₉	z ₇₁₀	z ₇₁₁	z ₇₁₂	z ₇₁₃	z ₇₁₄	z ₇₁₅	z ₇₁₆	z ₇₁₇	z ₇₁₈	z ₇₁₉	z ₇₂₀	z ₇₂₁	z ₇₂₂	z ₇₂₃	z ₇₂₄	z ₇₂₅	z ₇₂₆	z ₇₂₇	z ₇₂₈	z ₇₂₉	z ₇₃₀	z ₇₃₁	z ₇₃₂	z ₇₃₃	z ₇₃₄	z ₇₃₅	z ₇₃₆	z ₇₃₇	z ₇₃₈	z ₇₃₉	z ₇₄₀	z ₇₄₁	z ₇₄₂	z ₇₄₃	z ₇₄₄	z ₇₄₅	z ₇₄₆	z ₇₄₇	z ₇₄₈	z ₇₄₉	z ₇₅₀	z ₇₅₁	z ₇₅₂	z ₇₅₃	z ₇₅₄	z ₇₅₅	z ₇₅₆	z ₇₅₇	z ₇₅₈	z ₇₅₉	z ₇₆₀	z ₇₆₁	z ₇₆₂	z ₇₆₃	z ₇₆₄	z ₇₆₅	z ₇₆₆	z ₇₆₇	z ₇₆₈	z ₇₆₉	z ₇₇₀	z ₇₇₁	z ₇₇₂	z ₇₇₃	z ₇₇₄	z ₇₇₅	z ₇₇₆	z ₇₇₇	z ₇₇₈	z ₇₇₉	z ₇₈₀	z ₇₈₁	z ₇₈₂	z ₇₈₃	z ₇₈₄	z ₇₈₅	z ₇₈₆	z ₇₈₇	z ₇₈₈	z ₇₈₉	z ₇₉₀	z ₇₉₁	z ₇₉₂	z ₇₉₃	z ₇₉₄	z ₇₉₅	z ₇₉₆	z ₇₉₇	z ₇₉₈	z ₇₉₉	z ₈₀₀	z ₈₀₁	z ₈₀₂	z ₈₀₃	z ₈₀₄	z ₈₀₅	z ₈₀₆	z ₈₀₇	z ₈₀₈	z ₈₀₉	z ₈₁₀	z ₈₁₁	z ₈₁₂	z ₈₁₃	z ₈₁₄	z ₈₁₅	z ₈₁₆	z ₈₁₇	z ₈₁₈	z ₈₁₉	z ₈₂₀	z ₈₂₁	z ₈₂₂	z ₈₂₃	z ₈₂₄	z ₈₂₅	z ₈₂₆	z ₈₂₇	z ₈₂₈	z ₈₂₉	z ₈₃₀	z ₈₃₁	z ₈₃₂	z ₈₃₃	z ₈₃₄	z ₈₃₅	z ₈₃₆	z ₈₃₇	z ₈₃₈	z ₈₃₉	z ₈₄₀	z ₈₄₁	z ₈₄₂	z ₈₄₃	z ₈₄₄	z ₈₄₅	z ₈₄₆	z ₈₄₇	z ₈₄₈	z ₈₄₉	z ₈₅₀	z ₈₅₁	z ₈₅₂	z ₈₅₃	z ₈₅₄	z ₈₅₅	z ₈₅₆	z ₈₅₇	z ₈₅₈	z ₈₅₉	z ₈₆₀	z ₈₆₁	z ₈₆₂	z ₈₆₃	z ₈₆₄	z ₈₆₅	z ₈₆₆	z ₈₆₇	z ₈₆₈	z ₈₆₉	z ₈₇₀	z ₈₇₁	z ₈₇₂	z ₈₇₃	z ₈₇₄	z ₈₇₅	z ₈₇₆	z ₈₇₇	z ₈₇₈	z ₈₇₉	z ₈₈₀	z ₈₈₁	z ₈₈₂	z ₈₈₃	z _{884</}
--	----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	------------------	-----------------------

were developed by Hass (1970) from data provided by Frankel (1965):

$$\begin{aligned} f_j(x_j) &= 160.8 + 26.7 q_j + 640.7 (x_j - 0.45)^2 \\ &\quad + 255.7 q_j (x_j - 0.45)^2, \text{ \$/day} \\ &= c_{j1} x_j^2 + c_{j2} x_j + c_{j3}, \quad x_j \geq 0.45 \quad (3-2) \end{aligned}$$

where q_j is the plant size in million gallons per day (mgd), and x_j is the fraction BOD removed by treatment. Table 2 shows the plant size and the constants c_{jk} for the fifteen polluters.

Based on data provided by the Miami Conservancy District and the engineering firm employed by the conservancy in designing and operating a low flow augmentation reservoir, Hass developed the augmentation cost function $\phi(Y) = 726.6 Y (\text{\$/day})$, $1 < Y \leq 4.47$, where Y is the augmentation rate. The augmentation rate is the ratio of water flow rate at the time of consideration to that averaged per month of August in which lowest flow for this basin occurs. When there is no augmentation $Y=1$, and the flow rate is 180 cfs. The reservoir capacity constrains the augmentation to 4.47.

Then, the optimization problem for the Miami river basin can be written as follows:

$$\begin{aligned} \text{Minimize } F &= \sum_j f_j(x_j) + \phi(Y) \\ \text{subject to: } x_j & \\ \sum_{j=1}^i a_{ij} x_j + Y b_i &\geq S_i \\ i &= 1, 2, \dots, 27 \\ 0.45 &\leq x_j \leq 0.99 \\ j &= 1, 2, \dots, 15 \end{aligned} \quad (3-3)$$

and

$$1 \leq Y \leq 4.47$$

Table 2. Coefficients in quadratic cost function (3-2)
for the fifteen Miami river polluters.

Polluter Number	Reach Number	Plant size. (mgd)	COST COEFFICIENTS, \$/day		
			c _{j1}	c _{j2}	c _{j3}
1	4	45.2	12198.34	-10978.51	3837.80
2	6	4.7	1842.49	-1658.24	659.39
3	7	4.2	1714.64	-1543.18	620.15
4	8	3.6	1561.22	-1405.10	573.07
5	9	0.5	768.55	- 691.70	329.78
6	11	1.2	947.54	- 852.79	384.72
7	14	0.8	845.26	- 760.73	353.33
8	15	0.6	794.12	- 714.71	337.63
9	16	0.5	768.55	- 691.70	329.78
10	18	3.2	1458.94	-1313.05	541.68
11	21	8.4	2788.58	-2509.72	949.77
12	22	2.7	1331.09	-1197.98	502.44
13	23	0.6	794.12	- 714.71	337.63
14	25	12.1	3734.67	-3361.21	1240.14
15	26	8.4	2788.58	-2509.72	949.77

3.2 Multilevel Optimization Approach

Two types of decentralized decision making processes can be considered. One assumes a knowledge of the treatment cost function by the Regional Authority; the other assumes no such knowledge. Augmentation facility is considered only in the later case.

3.2.1 Pseudo Decentralized Process

In this process, it is assumed that the regional authority (RA) has a complete knowledge of the local treatment cost functions, $f_j(x_j)$, $j=1, 2, \dots, 15$. No augmentation facility is considered, so $Y=1$ and $\phi(Y)=0$.

The duality in nonlinear programming, along with the saddle point concepts (Lasdon, 1970), are the basis of the multilevel scheme utilized in this section. The optimization problem posed by (3-3) will be called the primal problem. Form the Lagrangian function, $L(\underline{X}, \underline{\lambda})$, for the primal problem as follows:

$$L(\underline{X}, \underline{\lambda}) = \sum_j f_j(x_j) + \sum_i \lambda_i (S_i - Y b_i - a_{i1} x_1 - a_{i2} x_2 - \dots - a_{i15} x_{15}) \quad (3-4)$$

where λ_i are Lagrange multipliers, $i=1, 2, \dots, 27$

$$\underline{\lambda} = [\lambda_1, \lambda_2, \dots, \lambda_{15}]$$

$$\underline{X} = [x_1, x_2, \dots, x_{15}]$$

Define a dual function $H(\underline{\lambda})$ as follows:

$$H(\underline{\lambda}) = \min_{\underline{X} \in S} L(\underline{X}, \underline{\lambda})$$

where

$$S = [\underline{X}: 0.45 \leq x_j \leq 0.99, j = 1, 2, \dots, 15]$$

The dual function $H(\underline{\lambda})$ is defined over the domain D , where

$$D = \{ \underline{\lambda} : \underline{\lambda} \geq 0, H(\underline{\lambda}) \text{ exists} \}$$

Since, for all $\lambda_i \geq 0$, $L(\underline{X}, \underline{\lambda})$ is continuous in $\underline{X}$ for all $\underline{X} \in S$, and S is closed and bounded, by Weierstraass theorem, $H(\underline{\lambda})$ exists. Thus

$$D = [\underline{\lambda} : \underline{\lambda} \geq 0]$$

Let $F(\underline{X}) = \sum_j f_j(x_j)$.

The dual problem can be written as

$$\max_{\underline{\lambda} \in D} H(\underline{\lambda}) \quad (3-5)$$

The Lagrangian $L(\underline{X}, \underline{\lambda})$ possesses a saddlepoint at $(\underline{X}^0, \underline{\lambda}^0)$ if and only if $\underline{X}^0$ solves the primal problem and $\underline{\lambda}^0$ solves the dual problem and

$$F(\underline{X}^0) = H(\underline{\lambda}^0) \quad (3-6)$$

When the Lagrangian $L(\underline{X}, \underline{\lambda})$ of the primal is separable in $\underline{X}$, the preceding argument suggests that we can decompose the problem into subsystems, solve them independently (1st level solutions), and coordinate these subsystems by choosing the multipliers, $\underline{\lambda}$, to solve the dual (2nd level solutions) so that their combined solutions solve the primal. The coordinator has the task of maximizing H subject to $\underline{\lambda} \in D$.

Now returning to $L(\underline{X}, \underline{\lambda})$, equation (3-4), since each $f_j(x_j)$ is a function of a single variable x_j and the coefficient terms are linear, $L(\underline{X}, \underline{\lambda})$ is additively separable with respect to x_j s and is decomposed readily into fifteen independent subsystems:

$$L(\underline{X}, \underline{\lambda}) = \sum_j L_j(x_j; \underline{\lambda}) + L_0(\underline{\lambda}) \quad (3-7)$$

where

$$L_j(x_j, \underline{\lambda}) = f_j(x_j) - \left(\sum_{i=1}^{27} \lambda_i a_{ij} \right) x_j \quad (3-8)$$

$$j = 1, 2, \dots, 15$$

and

$$L_o(\underline{\lambda}) = \sum_{i=1}^{27} \lambda_i (S_i - Y b_i) \quad (3-9)$$

Note that $L_o(\underline{\lambda})$ is a function of $\underline{\lambda}$ only, and is taken care of at the second level.

The multilevel structure of the optimization problem is outlined in Figure 4. At the first level $\underline{\lambda}$ is assumed to be known, and thus the first level optimization consists of obtaining $x_j(\underline{\lambda})$ which minimizes the corresponding j th subsystem Lagrangian, L_j , subject to $0.45 \leq x_j \leq 0.99$.

At the second level, with known values of $x_j(\underline{\lambda})$ s from the first level, $H(\underline{\lambda})$ is evaluated, and new multipliers $\underline{\lambda}$ are chosen so that $H(\underline{\lambda})$ is increased. Until $H(\underline{\lambda})$ is maximized, the same procedure is repeated. If, upon termination of this procedure, $\underline{\lambda}^o$ solves the dual, some $\underline{x}^o$ is primal feasible and the primal value is the same as the dual's, then $(\underline{x}^o, \underline{\lambda}^o)$ is the optimal solution to the system, that is, a saddle point exists.

Existence of a saddle point cannot be guaranteed for general classes of problems. However, problems with convex (or concave) objective and constraint functions under generous conditions should have a saddle point. Our problem here is one of these cases.

Computational algorithm. Although other techniques such as the Fletcher and Powell (1963) and Fletcher and Reeves (1964) methods are more efficient in computation, the gradient projection algorithm has been adopted for ease of programming and for theoretical interest. A listing of the computer program

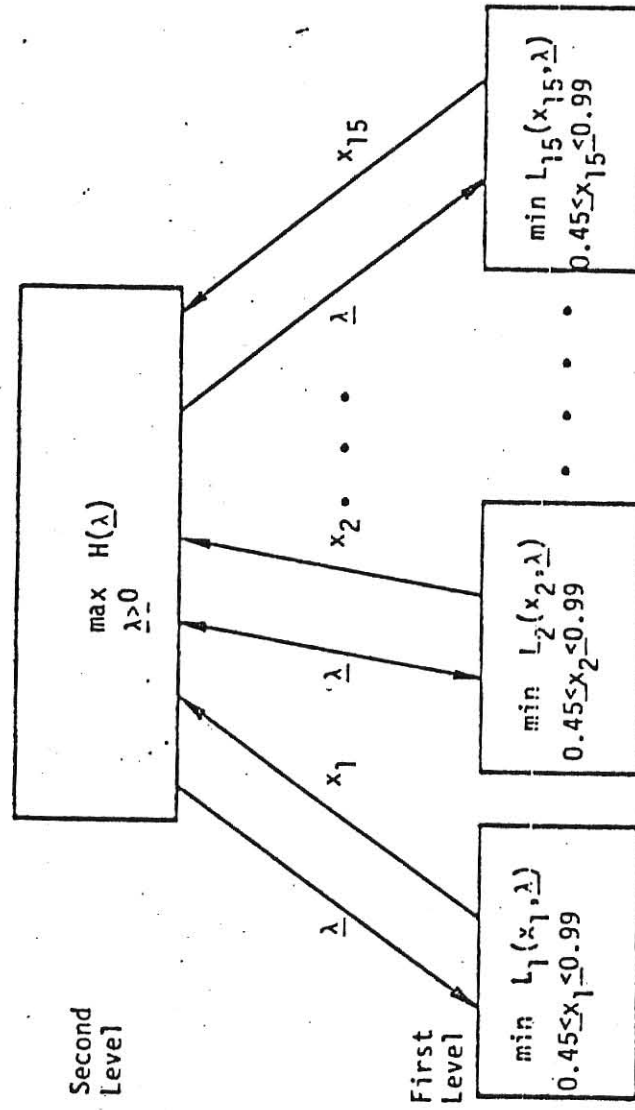


Figure 4. Multilevel structure of the problem.

used, written in FORTRAN IV, is given in appendix II.

The basic algorithm is as follows:

1. Choose initial values $\underline{\lambda}^0 \geq 0$.
2. Solve the first level with $\underline{\lambda} = \underline{\lambda}^k$, obtaining the solution $\underline{x}(\underline{\lambda}^k)$ subject to $0.45 \leq x_j \leq 0.99$ (k is the iteration number, $k = 0, 1, 2, \dots$).
3. Form the dual function $H(\underline{\lambda}) = L(\underline{x}(\underline{\lambda}^k); \underline{\lambda}^k)$ by adding the first level sub-Lagrangians and $L_0(\underline{\lambda}^k)$. Calculate the gradient $\nabla_{\underline{\lambda}} H(\underline{\lambda}^k)$ whose elements are

$$\frac{\partial H}{\partial \lambda_i} = S_i - b_i - a_{i1} x_1 - a_{i2} x_2 - \dots - a_{i15} x_{15} \quad (3-10)$$

4. Define a direction of search $\underline{d}^k$, by

$$d_i^k = \begin{cases} \frac{\partial H}{\partial \lambda_i} \Big|_{\underline{\lambda}^k}, & \text{if } \lambda_i^k > 0 \\ \max \left\{ 0, \frac{\partial H}{\partial \lambda_i} \Big|_{\underline{\lambda}^k} \right\}, & \text{if } \lambda_i^k = 0 \end{cases} \quad (3-11)$$

$i=1,2,\dots,27.$

5. Choose a new vector $\underline{\lambda}^{k+1}$ by

$$\underline{\lambda}^{k+1} = \underline{\lambda}^k + t_k \underline{d}^k$$

The step size is selected so that it maximizes $r(t) = H(\underline{\lambda}^k + t_k \underline{d}^k)$ subject to $t_k \geq 0$ and $\underline{\lambda}^{k+1} \geq 0$. For this selection of optimal t_k , a Fibonacci search technique has been employed in the interval 0 to 1. The direction vector $\underline{d}^k$ is normalized to avoid too small values for t .

6. Return to 2, stopping when $H(\underline{\lambda}^{k+1}) - H(\underline{\lambda}^k) \leq \epsilon$ (a small number)

The flow diagram for the above algorithm is given in appendix II.

Economic interpretation of the Multilevel procedure.

The gradient algorithm in the two level optimization procedure has an interesting economic interpretation. The first level subsystems represent the costs on the part of each local polluter and the second level represents the total social cost for a river basin-wide regional authority (RA).

According to the tax structure (λ^*) announced by the RA, the local polluters plan their treatment levels (x_j s) while minimizing their costs and report them to the RA. Upon receiving the responses by the local polluters, the RA forms the dual function which is the sum of the first level minima;

$$H(\lambda^*) = \sum_j L_j(x_j(\lambda^*); \lambda^*) + L_0(\lambda^*) \quad (3-13)$$

The gradient of H at λ^* has components

$$\frac{\partial H}{\partial \lambda_i^*} = \left[b_i - (a_{i1} x_1 + a_{i2} x_2 + \dots + a_{i15} x_{15}) \right],$$

$$i = 1, 2, \dots, 27$$

which are actually the slacks of the inequality constraints given in (3-3) and may be viewed as excess demands for oxygen in each reach.

By step 4, the algorithm requires that λ_i^* be increased if excess demand is positive and decreased otherwise, unless $\lambda_i^* = 0$. This is a familiar price-adjustment rule of classical economics. Thus, upon solving the second level, the regional authority (RA) has a new measure of the economic value (marginal cost) associated with supplying the i th reach with an additional pound of oxygen per day. These are the Lagrangian

multipliers λ^* . Note that λ_i^* are non-negative and, that for those constraints that are binding, they are positive.

Since $a_{ij} = d_{ij} w_j$ represents the oxygen demand in the i th reach to meet the waste water effluent load incurred by the j th polluter, the term $\sum_{i=1}^{27} \lambda_i a_{ij}$ may be viewed to represent the cost, for which the j th polluter is responsible, in improving the water quality. Hence, the term

$$\sum_{i=1}^{27} \lambda_i \alpha_{ij}$$

is the price (tax) imposed by the regional authority for the pollution caused by the j th polluter, if the polluter discharges the whole polluting load untreated. The tax for which the j th polluter is responsible is given by

$$\sum_{i=1}^{27} \lambda_i a_{ij} (1-x_j).$$

Each polluter is able to determine his own policy as to pay the price specified by the regional authority for causing the degradation of the quality of water or to invest money in developing the local waste treatment plant to decrease his taxes.

Computational results

(i) The process started with initial Lagrange multipliers,

$$\lambda_1 = \lambda_2 = \dots = \lambda_{26} = 0$$

and

$$\lambda_{27} = 5.$$

The initial primal and dual costs were 14607 and -36470 (\$/day) respectively. The initial values for the tax structure are chosen arbitrarily.

(ii) Within 17 iterations, the solution reached a point which was fairly near optimum with primal and dual costs being 8164 and 8331 respectively. The iterations thereafter were oscillating near the optimum. They showed a slow convergence in maximizing the dual H. The maximum H was found in 63rd iteration with primal and dual costs being 8304 and 8504. Thereafter, the convergence towards the optimal was insignificant.

(iii) The slow convergence after a few initial iterations is the characteristic of the steepest descent method used. Other accelerated gradient techniques are recommended for the practical calculation. In particular, since the dual problem is quadratic in the Lagrange multipliers, the gradient methods of Fletcher and Powell (1963) and Fletcher and Reeves (1964), applied at the second level would find the optimal solution of the problem in possibly less iterations.

(iv) It took 5 sec computer time of an IBM 370/158 for program compilation and 20 sec for computation in reaching the optimum. Note that the large number of iterations which was needed for convergence did not consume much computer time. The results obtained in present study are shown in Table 3 as compared with the results obtained by Haimes, et al. (1972 a) and Hass (1970).

3.2.2 Complete Decentralized Process

In this decentralized decision process, it is assumed that the regional authority has no knowledge of the local treatment cost functions. Augmentation facility is introduced into this scheme. While describing the complete decentralized process, Haimes, et al. (1972 a) did not emphasize on such

Table 3. Computational results for the Pseudo
Decentralized model.

TREATMENT FACILITY	DANTIZ-WOLFE HASS(1970)	M U L T I L E V E L	
		HAIMES, et al.(1972a)	PRESENT
1	86.8	87	88.8
2	70.5	71	71.7
3	99.0	99	99.0
4	99.0	99	99.0
5	54.8	55	55.2
6	57.8	57	58.2
7	47.7	48	47.2
8	51.6	52	50.4
9	48.1	48	47.6
10	60.2	60	57.5
11	51.9	52	50.7
12	51.0	51	49.9
13	48.3	48	47.7
14	52.8	53	51.4
15	51.4	51	50.3
SHADOW PRICES λ_{12} λ_{26}	0.4132 0.2118	0.41 0.21	0.442 0.174
TAX \$/day	2856	2816	2566
TREATMENT COST, \$/day	8312	8323	8504
TOTAL COST \$/day	11168	11139	11070

facility in their model. The results reported were compared with results as given by Hass (1970) for augmented flow. In present work such facility is introduced directly into the model. The formulation of the problem, to be solved by the multilevel approach, proceeds as follows:

Form the system Lagrangian $L(\underline{X}, \underline{\lambda})$ from the objective and constraint functions as given by (3-3).

$$L(\underline{X}, \underline{\lambda}) = \sum_j f_j(x_j) + \sum_i \lambda_i (S_i - \sum_j a_{ij} x_j - Y b_i) \quad (3-14)$$

This may be rearranged to yield:

$$L(\underline{X}, \underline{\lambda}) = \sum_j f_j(x_j) - \sum_j [x_j \sum_i a_{ij} \lambda_i] + \sum_i \lambda_i (S_i - Y b_i)$$

Summations over j are introduced:

$$L(\underline{X}, \underline{\lambda}) = \sum_j [f_j(x_j) - x_j \sum_i a_{ij} \lambda_i + (1/15) \sum_i \lambda_i (S_i - Y b_i)] \quad (3-15)$$

Now, the Lagrangian is in decomposable form. Subsystem Lagrangians L_j are readily formulated as:

$$L_j(x_j, \underline{\lambda}) = f_j(x_j) - x_j \sum_i a_{ij} \lambda_i + (1/15) \sum_i \lambda_i (S_i - Y b_i) \quad (3-16)$$

$j = 1, 2, \dots, 15$

The above L_j is viewed by the j th polluter as his objective function to be minimized at the first level. The second level (RA) coordinates the first level minima to bring out the overall optimal. Mathematically speaking, this is maximizing the dual

$$H(\underline{\lambda}) = L(\underline{X}^*(\underline{\lambda}); \underline{\lambda})$$

As shown in the Appendix I from the Kuhn-Tucker conditions for the optimal L , all the necessary conditions are satisfied at the first level solutions, except

$$\lambda_i g_i = 0, \quad i = 1, 2, \dots, 27 \quad (3-17)$$

where

$$\xi_i = S_i - \sum_j a_{ij} x_j - Y b_i$$

Equation 3-17 will be the coordinating criterion for the second level.

Computational algorithm. The computational logic, various steps and criterion used are described in the Appendix III. The algorithm is essentially a steepest ascent description of the law of supply and demand. The following gives an outline of the computational procedure.

The second level (RA) initially specifies zero tax rates, i.e., $\lambda_i = 0$, for all polluters. The fifteen dischargers respond from the first level with the minimum level of 45% removal of BOD. The RA receives this information and checks for violations by substituting the first level minimizing values of x_j s into the constraint set. Twenty two violations occur and the related components of the tax vector $\underline{\lambda}$ are increased and transmitted to the first level for the second iteration. Table 4 shows the responses of polluters for some tax structures.

The tax for each polluter is affected by the total vector of tax rates, $\underline{\lambda}$. A single increase associated with the last (27th) reach will affect all upstream decisions. Hence, twenty two changes should bring about a dramatic response. This occurs as seven dischargers respond with near maximum levels of treatment efficiency. The constraints are satisfied. Since the feasibility has been attained, the tax rates are decreased. The procedure is to determine the reach k which is the closest to violate the constraints by examining the constraint slacks for each reach, then reduce the tax rate at the k th reach to

Table 4. Polluters' response for some tax structures

TAX STRUCTURE 1	POLLUTERS' RESPONSE, %	TAX STRUCTURE 2	POLLUTERS' RESPONSE, %	TAX STRUCTURE 3	POLLUTERS' RESPONSE, %
REACH λ_i	POLLU- TER x_j	REACH λ_i	POLLU- TER x_j	REACH λ_i	POLLU- TER x_j
1 0	45	1 0	99	1 0	99
2 0	45	2 0	99	2 0	99
3 0	45	3 0	99	3 0	99
4 0	45	4 0	99	4 0	99
5 0	45	5 0.1	73	5 0	66
6 0		6 0.1		6 0	
7 0	45	7 0.1	86	7 0	71
8 0	45	8 0.1	67	8 0	45
9 0	45	9 0.1	91	9 0	45
10 0	45	10 0.1	65	10 0	45
11 0		11 0.1	99	11 0	
12 0	45	12 0.1	58	12 1.0	45
13 0	45	13 0.1	56	13 0	45
14 0	45	14 0.1	51	14 0	45
15 0	45	15 0.1	49	15 0	45
16 0	45	16 0.1	48	16 0	45
17 0		17 0.1		17 0	
18 0		18 0.1		18 0	
19 0		19 0.1		19 0	
20 0		20 0.1		20 0	
21 0		21 0.1		21 0	
22 0		22 0.1		22 0	
23 0		23 0.1		23 0	
24 0		24 0.1		24 0	
25 0		25 0.1		25 0	
26 0		26 0.1		26 0	
27 0		27 0		27 0	
AUGMENTATION, Y=1		AUGMENTATION, Y=1		AUGMENTATION, Y=1.086	

zero. New tax rates are transmitted to the first level which then responds with new treatment levels. The second level (RA) checks for violations for every response. The convergence criterion that all $\lambda_i g_i$ products be zero is next checked. This procedure of forcing tax rate components to zero is successively repeated until the constraints are again violated. Recovery steps are made until the critical constraint (i.e., the constraint that upon the recovery steps is the last to become feasible) is found to be the twelfth. This tax rate then is raised selectively. The constraints once again are satisfied and the algorithm attempts to force all $\lambda_i g_i$ products to zero.

Then, the augmentation facility is introduced to see if the overall cost can be minimized. The increment given in augmentation rate was just enough to bring back the violated constraints, because of change in λ , into the feasible set. The two conflicting decisions during the final fine tuning stage, increase in the augmentation cost and decrease in the tax rate, are balanced such that the tax revenue is maximized and the overall cost (sum of treatment cost, augmentation cost and the tax) is minimized.

Computational results. The final results are compared with previously published results (Hass, 1970; and Haines, et al., 1972 a) in Table 5.

It took only 1.1 sec computer time for compilation of the program and 9.5 sec for computations on an IBM 370/158. The programming of the multilevel logic is straight forward and minimizes debugging time. But, the most difficult task was to select the step sizes for the different stepsize options.

Table 5. Computational results for the Complete
Decentralized model.

TREATMENT FACILITY	DANTIZ-WOLFE HASS(1970)	M U L T I L E V E L	
		HAIMES, et al. (1972 a)	PRESENT
1	70.7	70.6	70.7
2	60.6	60.5	60.5
3	99.0	99.0	99.0
4	80.1	80.0	80.1
5	50.8	50.7	50.8
6	53.0	52.0	52.1
7	45.0	45.0	45.0
8	45.0	45.0	45.0
.	.	.	.
.	.	.	.
.	.	.	.
14	45.0	45.0	45.0
15	45.0	45.0	45.0
SHADOW, λ_{12} PRICE	0.278	0.277	0.278
AUGMENTA- TION RATE	2.4000	*	2.3727
COST \$/day	1744	*	1724
TAX \$/day	2410	2411	2410
TREATMENT COST, \$/day	6548	6536	6544

* not mentioned

4. OPTIMIZATION BY DISCRETE MAXIMUM PRINCIPLE

There exists a close relation between the discrete maximum principle and the two level structure of the multilevel system theory. Appendix I documents comparison of these techniques for a generalized river basin-wide water pollution abatement problem. In this section, the Miami river problem solved in preceding section is solved by the discrete maximum principle and the results are compared.

4.1 Formulation of the problem

In order to bring the problem posed by equations (3-3) into a form convenient to solve by the discrete maximum principle, X_1^n , a state variable is defined as the total dissolved oxygen required at reach n . Since the DO required in reach n is given by

$$\sum_{j=1}^n a_{ij} (1-\theta^n)$$

we can define X_1^n as

$$\begin{aligned} X_1^n &= X_1^{n-1} + \sum_{j=1}^n a_{ij} (1-\theta^n) \\ &= T_1^n(X_1^{n-1}, \theta^n), \quad n = 1, 2, \dots, 27 \end{aligned} \quad (4-1)$$

where

a_{ij} = DO required in reach i to decompose total BOD generated by the j th polluter, $i = 1, 2, \dots, 27$ and $j = 1, 2, \dots, 27$, pounds

θ^n = decision variable, the fraction of BOD treated at source in reach n ; if there is no polluter in reach k , $\theta^k = 0$, otherwise $0.45 \leq \theta^n \leq 0.99$, $n = 1, 2, \dots, 27$.

Recall that the objective is to minimize the overall cost which is the sum of treatment costs, tax revenue and augmentation cost. Mathematically, for the no augmentation case, the objective is to minimize

$$F = \sum_{n=1}^{N-27} f^n(\theta^n) + G^n(\theta^n) \quad \$/\text{day} \quad (4-2)$$

where

$$\begin{aligned} f^n(\theta^n) &= \text{cost of treatment at reach } n \\ &= c_{n1}(\theta^n)^2 + c_{n2}(\theta^n) + c_{n3} \quad \$/\text{day} \end{aligned} \quad (4-3)$$

and

$$\begin{aligned} G^n(\theta^n) &= \text{tax payable by the polluter in reach } n \\ &= \sum_{k=1}^N \lambda_k \alpha_{kn} (1 - \theta^n) \quad \$/\text{day} \end{aligned} \quad (4-4)$$

λ_k , $k = 1, 2, \dots, 27$, is the tax rate imposed by the authorities. It is the marginal cost of adding an additional pound of oxygen into reach k .

By introducing a new state variable x_2^n such that

$$\begin{aligned} x_2^n &= x_2^{n-1} + [f^n(\theta^n) + G^n(\theta^n)] \quad \$/\text{day} \\ &= T_2^n(x_2^{n-1}, \theta^n), \quad n = 1, 2, \dots, 27 \\ x_2^0 &= 0 \end{aligned} \quad (4-5)$$

the objective function can be reduced to the standard form of the discrete maximum principle as

$$\begin{aligned} F &= \sum_{n=1}^N [f^n(\theta^n) + G^n(\theta^n)] \\ &= x_2^N \\ &= c_1 x_1^N + c_2 x_2^N \end{aligned} \quad (4-6)$$

with

$$c_1 = 0 \quad \text{and} \quad c_2 = 1$$

The constraint on the state variable X_1^n is that the DO demanded by the BOD in reach n should not exceed the available DO in that reach. Mathematically,

$$R^n = -a_{mj}(1-\theta^j) + Y b^n \geq 0 \quad (4-7)$$

where b^n is DO available in reach n for unit of flow.

The procedure for such an optimization problem, a process with constraints on decision and/or state variables, is to introduce two dimensional (since there are two state variables) adjoint vector $\underline{Z}^n$ and a Hamiltonian function satisfying

$$H^n = (\underline{Z}^n)^T (\underline{X}^n) + U^n R^n \quad (4-8)$$

$$\underline{Z}^{n-1} = \frac{\partial H^n}{\partial \underline{X}^{n-1}} \quad (4-9)$$

with boundary conditions

$$Z_i^N = c_i \quad (4-10)$$

U^n is the penalty rate imposed by the authorities if the constraint R^n in reach n is violated. Recall that the constraint R^n is the requirement that the DO demanded by the BOD introduced by the polluters in reach n should not exceed the available DO in that reach. So, the penalty on reach n has to be shared by the polluters responsible for pollution in that reach, but proportional to the DO demanded by BOD for which each one is responsible. Such a penalty rate can be to any extent, but a reasonable decision is to penalize the polluters only to the extent of cost that authorities incur to satisfy the additional DO demand. In a free economy such a decision is more acceptable. With this penalty rate being assumed, U^n can only be the marginal cost of adding an additional pound of oxygen into reach n. So, the marginal cost λ^k

defined in equation (4-4) and U^n in equation (4-8) are identical.

Returning to equations (4-8) through (4-10), for our problem these are represented as

$$z_1^N = 0$$

$$z_2^N = 1$$

$$z_1^{n-1} = \frac{\partial H^n}{\partial x_1^{n-1}}$$

$$= z_1^n, \quad n = 1, 2, \dots, N$$

thus,

$$z_1^n = 0, \quad n = 1, 2, \dots, N$$

$$z_2^{n-1} = \frac{\partial H^n}{\partial x_2^{n-1}}$$

$$= z_2^n$$

$$= 1, \quad n = 1, 2, \dots, N$$

$$H^n = z_1^n x_1^n + z_2^n x_2^n + U^n R^n$$

$$= T_2^n(x_2^{n-1}, \theta^n) + U^n R^n(\theta^n)$$

Hence,

$$\frac{\partial H^n}{\partial \theta^n} = 2 c_{n1} \theta^n + c_{n2} - \sum_{k=1}^N \lambda_{kn} a_{kn} + U^n \sum_{j=1}^n a_{nj} \quad (4-11)$$

The above equations are the essence of necessary conditions for optimum H^n . Combining the Kuhn-Tucker conditions with the algorithm of the discrete maximum principle, the following equations, in addition, have to be satisfied for optimal H^n .

$$U^n \geq 0 \quad (4-12)$$

$$U^n R^n \geq 0 \quad (4-13)$$

$$R^n \geq 0 \quad (4-14)$$

Thus, equations (4-9) through (4-11) are the necessary conditions for a saddle point in $\bar{\theta}^n$ and $\bar{U}^n$; that is, H^n is a minimum with respect to $\bar{\theta}^n$ and a maximum with respect to $\bar{U}^n$.

4.2 Computational procedure and Results

One computational scheme suggested is a gradient search on the surface of H^n , which simultaneously descends in θ^n (the minimizing variable) and ascends in U^n (the maximizing variable). The computational procedure is as follows:

- a) Choose initial values for θ^n and U^n .
- b) Find the state variables X_1^n and X_2^n from equations (4-1) and (4-5) respectively.
- c) Evaluate $\frac{\partial H^n}{\partial \theta^n}$ from equation (4-11).
- d) Evaluate DU^n as follows:

$$DU^n = \begin{cases} 0 & \text{if } U^n = 0 \text{ and } R^n > 0 \\ -R^n, & \text{otherwise} \end{cases}$$

- e) Update U^n and θ^n as follows:

$$\theta^n = \theta^n - k \frac{\partial H^n}{\partial \theta^n}$$

$$U^n = U^n + k DU^n$$

where k is a suitable positive constant. The stopping criterion are

$$\frac{\partial H^n}{\partial \theta^n} \leq \epsilon_1$$

for all n with $\theta^n = 0$ and $0.45 < \theta^n < 0.99$, and

$$DU^n \leq \epsilon_2$$

ϵ_1 and ϵ_2 are small numbers. This algorithm is guaranteed to converge to global optimum of the objective F (appendix I). Another

computational scheme involves adjusting the adjoint variables Z^n instead of θ^n . In our problem, Z^n s are invariant, so such scheme is not applicable here.

Computational results. i) The iterations started with $U^n = 0.1$ for all n and $\theta^n = 0.99$ for all stages (or reaches) with polluting inputs. Note that $\theta^n = 0$ for non-polluting reaches.

ii) The programming is simpler than that for multi-level structure. The only difficulty was with choosing values for step sizes and stopping criterion for $\frac{\partial H^n}{\partial \theta^n}$ and DU^n . A compromise had to be struck between large computer time required with the smaller values and accuracy of the results for the problem. However, as a result of extensive search, a reasonable step size (0.01) along with the following modifications in the algorithm was found appropriate to force the algorithm to the near optimal.

- a) Let the greatest value of $\frac{\partial H^n}{\partial \theta^n}$ be unity, and the other $\frac{\partial H^n}{\partial \theta^n}$ have the values equal to their ratios to the greatest one.
- b) Let the greatest value of DU^n be unity, and the other DU^n have the values equal to their ratios to the greatest one.

iii) It took 0.6 sec computer time of an IBM 370/158 for compilation of the program and 52 sec for computation. The minimum total cost obtained with this, in 394 iterations, is 11,176 \$/day. Note that large number of iterations did not require much computer time. The cost of computations is \$16.43.

iv) These results are compared with those obtained by multilevel approach in Table 6. The results differ by about 5%.

v) We concluded that the Lagrangian multipliers in multilevel approach correspond to U^n s in the Hamiltonian function

Table 6. Comparison of results for the Pseudo
Decentralized model.

TREATMENT FACILITY	MULTILEVEL MODEL	DIS. MAX. PRIN. MODEL
1	88.8	86.7
2	71.7	70.5
3	99.0	99.0
4	99.0	99.0
5	55.2	56.4
6	58.2	58.6
7	47.2	49.2
8	50.4	53.5
9	47.6	50.2
10	57.5	62.0
11	50.7	52.9
12	49.9	51.8
13	47.7	50.2
14	51.4	53.9
15	50.3	45.0
SHADOW PRICES [λ_{12} λ_{26}	0.442 0.174	0.3787 0.2337
TAX \$/day	2566	2836
TREATMENT COST, \$/day	8504	8340
TOTAL COST \$/day	11070	11175

of this algorithm (appendix I). We arbitrarily have chosen the penalty rates, U^n , as the marginal cost of adding an additional pound of DO into each reach. From this numerical example such assumption is justified.

5. THREE-LEVEL APPROACH TO INTRODUCE A REGIONAL TREATMENT PLANT

A system of effluent charges was illustrated in chapter 3. Such a hierarchical system need not necessarily be limited to just a two-level structure. A physical, economical or political aspect of a system could result in a structure demanding more levels. In this section, a three-level approach for the regional water quality management problem is described.

For ease in comparison, the configuration of Miami river basin in Ohio is chosen again. The formulation of the problem is based on the work by Hass (1970) and Haimes, et al. (1972 b). A hierarchical system of central authority (CA) and regional authority (RA) is postulated. A regional treatment facility is considered as a possibility.

5.1 Problem Definition

In reviewing the data on the Miami river basin, it was noted that the majority of the waste water treatment plants were small in size (less than 10 mgd), see Table 2. By using a quadratic treatment cost function generated by Hass for the computer simulation of the problem, Figure 5 shows that these small plants have a greater per unit treatment cost than the larger capacity plants treating to the same level. Thus, if the central authority built a regional water treatment plant for treatment of the waste water currently treated at the small local plants, a reduction in the overall treatment cost could be obtained. The tax revenues may be used by the central authority to provide a regional treatment facility; polluters may make sewer connections with the regional plant to reduce

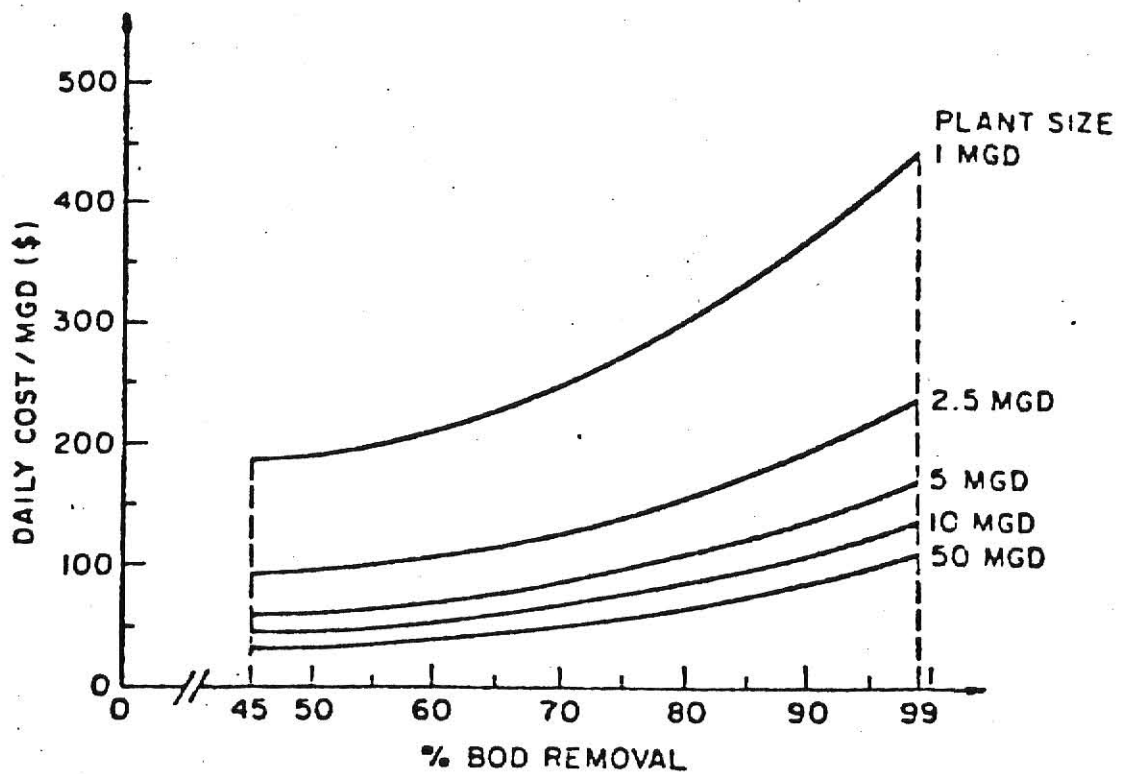


Figure 5. The cost functions for various treatment plant sizes (Haimes, et al., 1972 b).

their waste treatment costs. Considerable savings over local treatment alone may be realized. The new regional water quality problem can be stated as:

" Find an optimal treatment configuration for meeting water quality standards along a river basin (considering the possibility of a regional waste water treatment plant) and simultaneously determine the optimal effluent charges necessary to achieve this configuration when the central authority does not necessarily know the treatment cost functions."

The application of the Dantzig-Wolfe procedure used by Hass (1970) to the solution of the regional problem is not possible any more because of the additional complexity introduced by the possibility of using a regional waste water treatment plant. A multilevel approach is so proposed in which the solution is obtained by a three level optimization procedure.

The river basin model used in chapter 3 has been modified by Haines, et al. (1972 b) from Hass (1970 , microfiche appendix) to take into account the effect of waste water treatment at the regional plant,. In the computational procedure, the waste water from Dayton (reach 4) was not permitted to be shipped to the regional plant, since the plant at Dayton was quite large already (45.2 mgd). In addition, no river flow augmentation was considered. Data from Anderson (1967) were used to determine the location of the regional waste water treatment plant. The location was determined as the centroid of the waste water load-distance product for all polluters except Dayton. Figure 6 shows the location of the regional plant (reach 19) along with the proposed piping routes to the regional plant. The treatment cost functions were as given by equation

Δ Industrial loads
 O Municipal loads
 # Reach

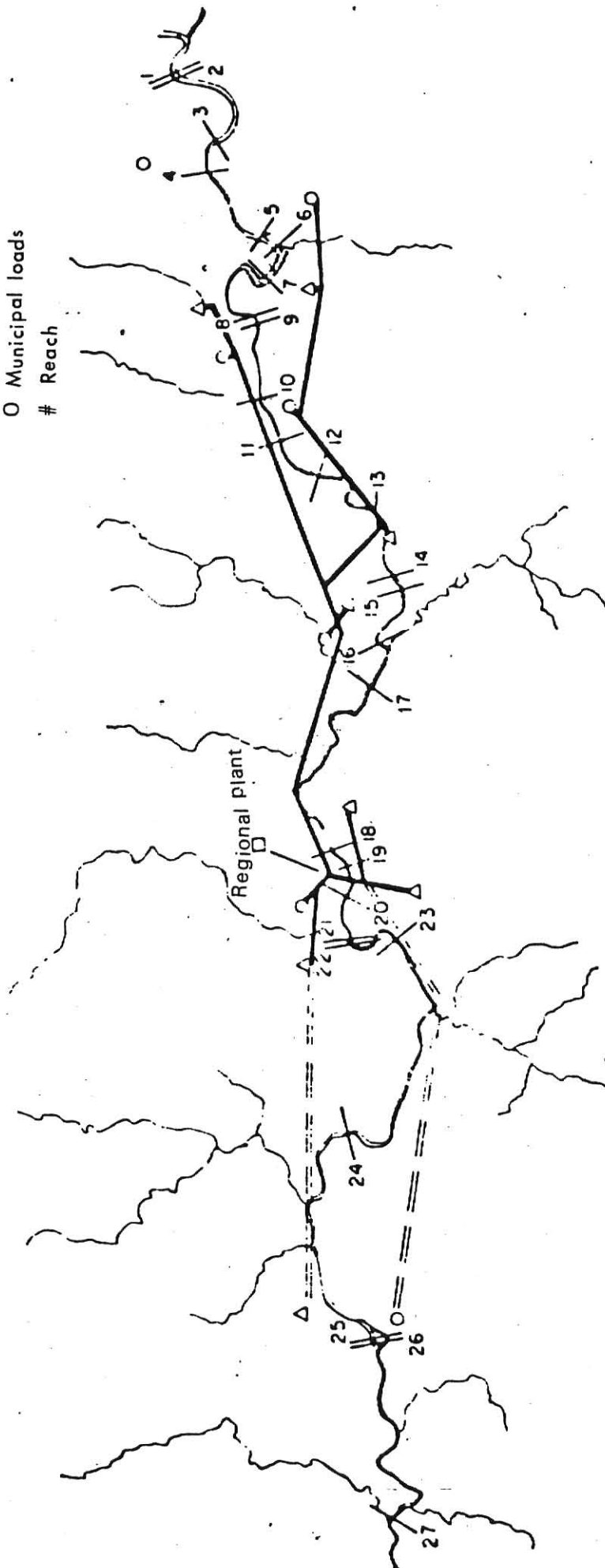


Figure 6. Regional plant location for Miami river (Haines, et al., 1972 b)

(3-2). The piping costs include both operating and maintenance costs and the cost of capital.

5.2 Problem Formulation

The objective is to minimize the total cost. Costs of treatment, at regional and local plants, piping costs and the tax revenue constitute the total cost in our problem. The overall optimization problem is to minimize

$$\sum_{j=1}^{15} [p_j F_j(x_j) + (1-p_j) c_j] + F_R(q_R, x_R) \quad (5-1)$$

subject to

$$\begin{aligned} \text{AWPX} + f_o(ZC_o + BS - Q) Y \geq Vw_o - Z(C^T + PC^E + C^R) \\ + \text{AWPI} + B(\Gamma^T + \Gamma^E)S + (\Gamma^T + \Gamma^E)Q \end{aligned} \quad (5-2)$$

$$0.45 \leq x_j \leq 0.99 \quad (5-3)$$

$$Y = 1 \quad (5-4)$$

$$j = 1, 2, \dots, 15$$

where p_j is 1 if the j th polluter treats the waste water locally and 0 if he ships it to regional plant; c_j and $F_j(x_j)$ are its piping costs and the cost of treatment at the plant, respectively. F_R is the cost of treating all the waste received at the regional plant with a plant size of q_R and a treatment level of x_R . F_R is computed, as other F_j s, from equation (3-2).

The constraints (5-3) are the limits on the treatment levels. Y is the augmentation rate, since no augmentation is considered, $Y=1$. Constraints (5-2) require that, as before, the DO demand in each reach should not exceed the DO available in that reach. The details of such formulation is given by Hass (1970, microfiche appendix). Then, Haimes, et al. (1972 b) extended it to account for added flow and DO from tributaries, local plants, and the regional plant separately. Table 7 depicts the physical meaning of the variables grouped in these

Table 7. Description of variables in equation (5-2)

VARIABLE	DIMENSION	DESCRIPTION
n		number of reaches (27)
A	n x n triangular matrix	DO required per pound of BOD, pounds
W	n x n diagonal matrix	Gross BOD loads, pounds/day
P	n x n diagonal matrix	0,1 decision matrix of treatment location
X	n x 1 vector	treatment levels
f ₀	scalar	unaugmented flow, at the begining of reach 1, cubic feet per sec.
Z B	n x n triangular matrices	functions of deoxygenation and reoxygensation rates.
V	n x 1 vector	... do ...
C ₀	n x 1 vector	initial DO levels, mg/l
S	n x 1 vector	DO saturation levels, mg/l
Q	n x 1 vector	quality standards on DO, mg/l
w ₀	scalar	BOD at reach 1, pounds/day

Table 7. (continued)

VARIABLE	DIMENSION	DESCRIPTION
C^T	$n \times 1$ vector	DO contents of tributaries, pounds
C^E	$n \times 1$ vector	DO contents of discharges, pounds
C^R	$n \times 1$ vector	DO added due to regional discharge, pounds
I	$n \times 1$ vector	$(1 \ 1 \ \dots \ 1)^T$
Γ^T	$n \times n$ diagonal matrix	added flow due to tributaries, cfs
Γ^E	$n \times n$ diagonal matrix	added flow to discharges, cfs

constraints (5-2). Note that change in flow due to regional plant discharge changes the characteristics of water (flow rate, DO and BOD profiles, etc.) in the river.

5.3 Analysis of Three Level Structure

Since we have assumed that the central authority does not know the treatment cost functions of the individual polluters, the problem must be reconstructed accordingly into subproblems that facilitate its solution. The three levels of the hierarchical system are

- (1) the individual polluters,
- (2) the regional authority (RA), and
- (3) the central authority (CA).

To illustrate how this system works, we analyze how information is exchanged among the levels and how this information is updated and coordinated to obtain the optimal solution to the overall system (see Figure 7).

Third Level Problem

The function of the third level, the central authority, is to propose a tax structure to the lower levels. This tax structure is based on the marginal cost to society of additional pound of DO in each reach of the river. These marginal costs are the shadow prices, Lagrangian multipliers, of the binding constraints in the overall optimal solution. The BOD load discharged by the regional plant is taxed in the same manner as the loads of the local plants.

The central authority begins the solution process by estimating all the shadow prices, and thus a tax structure. This tax structure is sent down to the lower levels for processing.

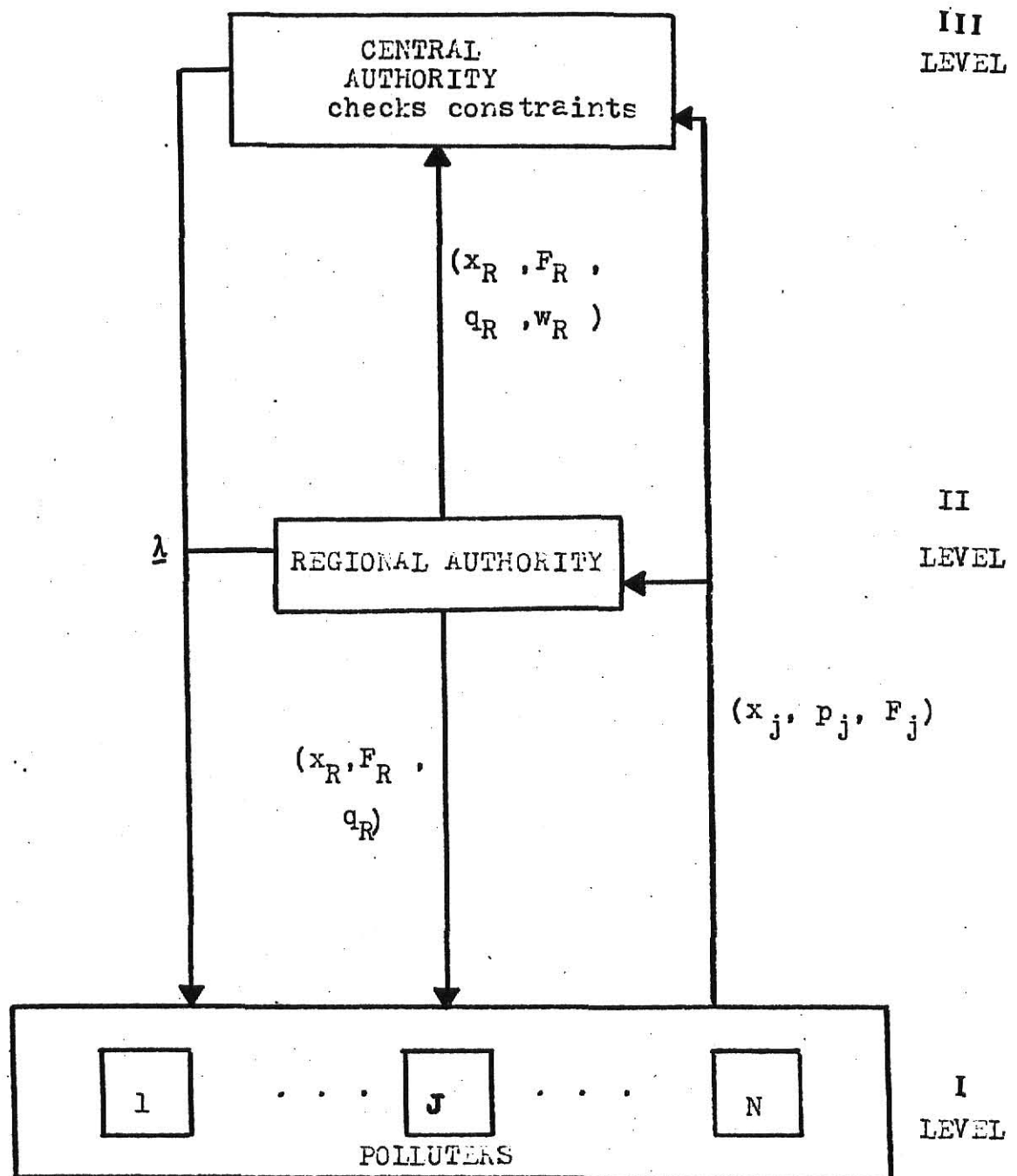


Figure 7. Three Level Structure

The two levels respond with their optimal treatment levels. Using this information, the central authority checks the river quality constraints (5-2) to determine if the previous taxing structure was too high (no binding constraints), too low (some constraints violated) or optimal (no constraints violated, some binding constraints). If the previous structure is not optimal, a new tax structure is developed on the basis of the results of the previous structure. A gradient search technique (Lasdon, 1970) can be used to update the step sizes to converge toward the optimal solution.

Second Level Problem

The function of second level, the regional authority, is to determine an optimal regional plant size and the treatment level, and to issue charges based on the regional cost incurred to the polluters who are shipping their waste water to the regional plant.

The optimization problem at the second level is to minimize the sum of cost of regional treatment and tax on regional plant.. Mathematically, this is to

$$\min [(F_R(q_R, x_R) + (1-x_R) T_R) \quad (5-5)$$

subject to

$$0.45 \leq x_R \leq 0.99 \quad (5-6)$$

$$q_R = \sum_j (1-p_j) q_j \quad (5-7)$$

$$w_R = \sum_j (1-p_j) w_j \quad (5-8)$$

where p_j is the decision from the first level (0 if the j th polluter ships waste water to regional plant, 1 if he treats it locally); x_R is the regional treatment level; q_R and w_R are the capacity (mgd) and total BOD load (lbs/day) at the

regional plant. T_R is the tax imposed on regional plant.

This is computed as the product of shadow price (row) vector, λ and the vector of DO demands (a_R) in each reach due to BOD in regional effluent.

Given a tax structure determined by the third level, a second level assumes a given regional plant size based on a prediction of which polluters will ship their waste water to the regional plant. Then it computes an optimal regional treatment level and the resulting regional treatment cost. The regional treatment cost is divided among the polluters assumed to be shipping their waste water to the regional plant on the basis of the total waste water shipped. The tax on the regional plant is divided among those polluters on the basis of the total BOD load shipped to the regional plant. This cost information then is sent down to the first level, where it is used to determine which polluters actually would ship their waste water to the regional plant. If the prediction which polluters ship to the regional plant is correct, the process stops, and all the information is passed to the third level. If the prediction is not correct, the prediction is updated on the basis of the results obtained, a new regional plant size is determined and the process is repeated until predicted plant size and the actual numbers of polluters that ship waste water to the regional plant agree. In this problem, a modified Gauss-Siedel routine (Haines, et al., 1968) can be used to update the prediction of the regional plant size.

First Level Problems

The function of the first level is to determine the optimal treatment cost for each polluter. This involves determining the optimal local treatment level and whether to ship to the regional treatment plant. The optimal treatment policy can be determined by (i) minimizing the sum of cost of treatment and local tax; and then, (ii) comparing with the sum of piping cost and the polluters share of the regional cost if he decides to ship his waste water to the regional plant. Mathematically, this is to

$$\min \phi_j(x_j, p_j) = p_j [F_j(x_j) + (1-x_j)T_j] + (1-p_j) [(q_j/q_R) F_R(q_R, x_R) + c_j (w_j/w_R)(1-x_R)T_R] \quad \$/\text{day} \quad (5-9)$$

subject to

$$0.45 \leq x_j \leq 0.99 \quad (5-10)$$

$$p_j = 1 \quad \text{for local treatment} \\ = 0 \quad \text{for regional treatment.}$$

For a notified share of the regional cost, the individual polluter will determine (i) the optimal treatment level, (ii) the cost of local treatment, and (iii) his decision if he wants to ship his waste water to regional plant, or not. The resulting information is passed to the higher levels. Note that the decision of any individual polluter is independent of other polluters' decisions and no information is exchanged among them.

Conclusion

The computational results (Haimes et al., 1972 b) for the Miami river basin data show that the introduction of a regional waste water treatment plant results in a overall cost of \$8,493 per day, yielding a savings of \$2,577 per day in comparison to

the cost estimated in the two level structure with no augmentation (\$11, 070 per day). 12 of 15 polluters' decision is to ship their waste water to the regional plant. Savings due to introduction of regional plant instead of augmentation facility results in a savings of \$2, 185 per day. Table 8 depicts the comparison of all costs involved in these three types of water quality management structures.

Table 8. Comparison of three level vs. two level solutions

	TWO LEVEL SYSTEMS		THREE LEVEL
	No Aug. 1	Augmntn. 2	SYSTEM 3
Shadow prices λ_{12}	0.442	0.278	0.11243
λ_{26}	0.174		0.10686
λ_{13}			0.28077
Tax revenue, \$/day	2566	2410	2481
Augmentation cost, \$/day		1724	
Treatment cost, \$/day	8504	6544	6012
Total cost, \$/day	11,070	10,678	8493
Savings over plan 1, \$/day		392	2577
plan 2, \$/day			2185

The attempt to formulate and solve a specific problem by numerous techniques is not merely a mathematical experimentation, but the different approaches and methodologies aim at providing decision makers with more information, and alternate feasible options for their bases of decisions. Although many techniques can solve any specific problem, the physical structure of real systems forces the researcher to look into various methodologies which represent the real system as precisely as possible.

The mathematical analogy between the multilevel approach and the discrete maximum principle could be exploited. The solutions of a simple model for a river basin-wide water quality problem by these two techniques confirm such a relation. Hence, in a situation where the decomposition and coordination of two level approach is complex, the discrete maximum principle algorithm can be used to solve the problem, and then from its relation to multilevel approach, the tax structure can be interpreted. Although these two techniques are proved to be mathematically equivalent, the way of approach to an optimization problem according to each method is different. By means of multilevel approach we can decompose the whole system into subsystems with smaller dimensions. The subsystems then may be solved by any of the existing optimization techniques. Thus, it is plausible to construct, by combining the two techniques, a powerful method for optimizing the highly dimensional complex system. This is an area for future work.

In a complex system solution, neither the shadow prices nor penalty functions introduced can have a direct physical

interpretation. It was plausible to conclude, in case of a simple model with linear constraints for Miami river basin, that the shadow prices or the Lagrangian multipliers are the marginal cost of adding an additional pound of DO into each reach. The penalty could be shared by the polluters in proportion to the DO demands of their waste discharges. Such a distribution is not plausible when there are more than one type of pollutant. The penalty imposed on violating certain constraints can not be weighed equally with other constraints. However, a hypothetical tax structure with different penalizing levels for different pollutants can be introduced. The lack of pertinent data restricts such experimentation at this time. This is left as an area for future work.

REFERENCES

- Anderson, M.W., Regional water quality management in the Miami basin, Ph.D. thesis, Carnegie-Mellon University, Pittsburg, Pennsylvania, 1967.
- Arrow, K.J., L. Hurtticz, and R. Pfants, Essays in Economics and Econometrics, 34, Univ. of North Carolina Press, Chapel Hill, 1963.
- Bundgaard-Nielson, M. and C. L. Hwang, A review on decision models in Economics of Regional Water Quality Management, Water Resources Bulletin, 12(3), June 1976.
- Brosilo, C.B. and L. S. Lasdon, A two level optimization technique for recycle processes, presented at I.Ch.E.-A.I.Ch.E. Joint Meeting, London, June 1964.
- Chen, Y.C., A comparative study of the Maximum principle and the multilevel system theory and their application, M.S. thesis, Kansas State Univ., Manhattan, Kansas, 1966.
- Dantiz, G. and P. Wolfe, The decomposition algorithm for linear programs, Econometrica, 29(4), 767-778, 1961.
- Fan, L.T., The continuous Maximum Principle, Wiley, New York, 1966.
- Fan, L.T. and C. S. Wang, The discrete maximum principle, Wiley, New York, 1964.
- Fletcher, R. and M. J. D. Powell, A rapidly convergent descent

- method for minimization, Brit. Comp. J., 6:163-168, 1963.
- Fletcher, R. and C. M. Reeves, Function minimization by conjugate gradients, Comp. J., 6:163-168, 1964.
- Foley, J.W. and Y. Y. Haines, Multilevel control of multi-pollutant system, Journal of the environmental engineering division, ASCE, 99(EE3), Proc. paper 9795, 253-268, 1973.
- Frankel, R. J., Water quality management: An engineering economy model for domestic waste disposal, Ph. D. University of California, Berkeley, California, 1965.
- Haines, Y. Y., Modelling and control of the pollution of water resources systems via the multilevel approach, Water Resources bulletin, 7(1), 93-101, 1971.
- Haines, Y. Y., Decomposition and Multilevel Techniques for Water Quality Control, Water Resources Research, 8(3), 779-784, 1972.
- Haines, Y. Y., R.L. Perrine, and D. A. Wismer, Identification of aquifer parameters by decomposition and multilevel optimization, Israel J. of Technology, 6(5), 322-329, 1968.
- Haines, Y. Y., M. Kaplan, M. Husar, and L. R. Bronner, Comparison of solution methods for optimal regional water quality management, Systems Research Center Report 71-9, Case Western Reserve Univ., Cleveland, Ohio, 1971.

Haimes, Y.Y., J. Foley, and W. Yu, Computational results for water pollution taxation using multilevel approach, Water Resources Bulletin, 8(4), 761-772, 1972a.

Haimes, Y.Y., M. A. Kaplan, and M. A. Husar, A multilevel approach to determining optimal taxation for the abatement of water pollution, Water Resources Research, 8(4), 851-860, 1972b.

Hass, J. E., Optimal taxing for the abatement of water pollution, Water Resources Research, 6(2), 353-365, 1970.

Kuhn, H.W. and A. W. Tucker, Nonlinear Programming, Proc. Second Berkely Symposium on mathematical statistics, and probability, Univ. of California Press, Berkeley, California, 481-492, 1950.

Lasdon, L.S., A multilevel technique for optimization, Ph.D. thesis, Case Institute of Technology, 1964

Lasdon, L.S., Optimization theory for large systems, 396-434, Mac Millan, New York, 1970.

Letov, A.M., Stability in Nonlinear systems, Princeton Univ. Press, Princeton, New Jersey, 1961.

Merriam, C.W., Optimization theory and the Design of Feedback Control System, McGraw-Hill, New York, 1964.

Mesarovic, M.D., D. Macko and Y. Takahara, Theory of Multi-level Hierarchical Systems, Academic Press, New York, 1970.

Noton, A. R. M., Optimal control and the two point boundary problem, presented at the advances in Automatic Control Symposium sponsored by the Institute of Mechanical Engineers, Nottingham, Britain, Apr 1965.

Samuelson, P.A., Market Mechanisms and Maximization, The Rand Corporation, Santa Monica, 1949.

Saaty, T.L. and J. Brown, Nonlinear Mathematics, 237, McGraw-Hill, 1964.

Streeter, H.W. and E. B. Phelps, A study of the pollution and natural purification of the Ohio river, Public Health Bulletin, 146, Feb 1925.

Takahara, Y., Multilevel approach to dynamic optimization, Report No. SRC-59-A-64-21, Case Institute of Technology, May 1964.

Thomann, R. V., Systems analysis and water quality management, Environmental Research and Applications, Inc., New York, 1972.

Wisner, D.A., Optimization methods for Large-Scale Systems, McGraw-Hill, New York, 1971.

APPENDIX I

COMPARISON OF THE MULTILEVEL APPROACH AND THE DISCRETEMAXIMUM PRINCIPLE

INTRODUCTION

There exists a close relation among the maximum principle, dynamic programming and the calculus of variations (Fan, 1966). Similarly, the two level structure of the multilevel approach and the discrete maximum principle are interrelated. Although these two techniques can be proved to be mathematically equivalent, the two ways of approach to an optimization problem are quite different.

To illustrate the similarities between the two level structure of the multilevel approach and the discrete maximum principle a specific problem, the water pollution control model, is considered here. There is, however, a little loss of generality in establishing the relations from any specific problems, the extension to more general situations seems difficult but not impossible. Chen (1966) discussed these similarities for a case of simple feed back multi-stage processes. Following is the modified version of such comparison, but as applied to our problem.

Recall that a water resources system, as formulated in section 2, is simple sequential process of N stages (Figure 1). The optimization problem is to

$$\text{Minimize } F = \sum_{n=1}^N F^n(U^n, S^n) \quad (\text{A-1})$$

subject to:

$$V^n = T^n(U^n, W^n) \quad (\text{A-2})$$

$$S^n = \phi^n(U^n, W^n) \quad (A-3)$$

$$W^{n+1} = V^n \quad (A-4)$$

$$R^n(U^n, W^n, S^n) \geq 0 \quad (A-5)$$

$$n = 1, 2, \dots, N.$$

the notation used in section 2 is being used here

SOLUTION BY MULTILEVEL APPROACH

The concept of multilevel approach is based on the decomposition of the overall system, and then reconstruction of the system into independent subsystems. This decomposition is accomplished by introducing pseudo variables, if necessary, into the system. Each subsystem is separately and independently optimized. This is called the first level solution. The optimized subsystems are joined together by second (or higher) level controllers in order to arrive at the solution for the overall system. The coordination can be structured into more than two stages. The present problem can be solved by two level structure.

As shown in Figure 1, the water resources system is a chain of N subsystems. View each subsystem as buying and selling the input V^{n-1} (i.e., W^n) and output V^n from and to other subsystems. Associate with each V^n a vector of prices (real numbers) of the same dimension, p^n , at which these transactions take place. Since the first level solution requires the independent subsystems, let the input V^n to the $(n+1)$ th subsystem be independent of the output $T^n(U^n, V^{n-1})$ of the n th subsystem, that is, the performance equations (A-2) be ignored. The penalty associated with this is the price p^n

I. The Subproblem (First Level)

Treating p^n as parameters, the nth subproblem is described by the relations

$$S^n = \phi^n(U^n, V^{n-1}) \quad (A-3)$$

$$R^n(U^n, S^n, V^{n-1}) \geq 0 \quad (A-5)$$

in which a set of $U^n = \tilde{U}^n$ and $V^n = \tilde{V}^n$ that optimizes the subobjective function

$$A^n = F^n(U^n, S^n) + (p^n)^T T^n(U^n, V^{n-1}) - (p^{n-1})^T V^n \quad (A-6)$$

is found.

The subproblem solutions U^n, V^n are, of course, functions of the prices p^n . Then there exist values of $\bar{p}^n$ at which the subproblem solutions are the optimal solution of the overall problem. This criteria can be proved as follows:

Let L be defined as the Lagrangian for the overall problem, so that

$$L = \sum_{n=1}^N \left[F^n(U^n, S^n) + (p^n)^T (T^n - V^n) + (\lambda^n)^T R^n \right] \quad (A-7)$$

and

$$p^0^T V^0 = p^N^T V^N \quad (A-8)$$

where the vectors p^n , $n = 1, 2, \dots, N$ are m -dimensional lagrange multipliers for the equality constraints, equations (A-2), and the vectors λ^n , $n = 1, 2, \dots, N$ are k -dimensional generalized Lagrange multipliers (or Kuhn-Tucker multipliers) for the inequality constraints, equations (A-5).

From the definition of p^n and equation (A-8), it is seen that

$$\sum_{n=1}^N (p^n)^T V^n = \sum_{n=1}^N (p^{n-1})^T V^{n-1} \quad (A-9)$$

(sum of output prices = sum of input prices)

Substituting (A-9) into (A-7), the Lagrangian becomes

$$L = \sum_{n=1}^N \left[F^n(U^n, S^n) + (p^n)^T T^n(U^n, v^{n-1}) - (p^{n-1})^T v^{n-1} \right. \\ \left. (\lambda^n) R^n(U^n, S^n, v^{n-1}) \right] \quad (A-10)$$

$$= \sum_{n=1}^N L^n$$

According to Kuhn-Tucker (1950), the necessary conditions for an extremum are that there exist $\lambda^n = \bar{\lambda}^n$ and $p^n = \bar{p}^n$ such that

$$\frac{\partial L}{\partial U^n} = \frac{\partial F^n}{\partial U^n} + (\bar{p}^n)^T \frac{\partial T^n}{\partial U^n} + (\bar{\lambda}^n)^T \frac{\partial R^n}{\partial U^n} = 0 \quad (A-11)$$

$$\frac{\partial L}{\partial v^{n-1}} = \frac{\partial F^n}{\partial v^{n-1}} + (\bar{p}^n)^T \frac{\partial T^n}{\partial v^{n-1}} - \bar{p}^{n-1} + (\bar{\lambda}^n)^T \frac{\partial R^n}{\partial v^{n-1}} = 0 \quad (A-12)$$

$$\bar{\lambda}^n \geq 0 \quad (A-13)$$

$$(\bar{\lambda}^n)^T R^n = 0 \quad (A-14)$$

$$\frac{\partial L}{\partial \lambda^n} = R^n \geq 0 \quad (A-15)$$

$$\frac{\partial L}{\partial p^n} = v^n - T^n(U^n, v^{n-1}) = 0 \quad (A-16)$$

$$n = 1, 2, \dots, N$$

The problem of finding an optimal decision vector which satisfies at least the necessary conditions for a maximum or minimum is called the integrated problem (Takahara, 1964).

All quantities in equations (A-13) through (A-16) are evaluated at the optimum point, U^n . But the conditions, equations (A-11) through (A-15), are equivalent to the Kuhn-Tucker conditions for the subobjective function, A^n subject to $R^n \geq 0$ at the point $(\bar{U}^n, \bar{S}^n, \bar{v}^{n-1})$. Thus, regarding p^n as prices, we see that these prices the subproblems satisfy the necessary conditions for an extremum at any point at which the integrated (overall) problem satisfies the same conditions.

II. The Second Level Problem

The fact that there exist prices p^n that decouple an integrated problem is utilized to derive an iterative procedure for optimization.

The task of finding the optimal parameters p^n is delegated to second level, and the solution of the subproblems for a given set of p^n s is the responsibility of the first level. Note that since all the subproblems are independent, the first level units need not communicate with each other; that is, the performance equations, equations (A-2) are ignored.

With the subproblem solution U^n and V^{n-1} substituted into the performance equations, (A-2), we can get the output from the n th subsystem, i.e., $T^n(U^n, V^{n-1})$. Then from the difference between the amount V^n (i.e., input) demanded by the $(n+1)$ th subsystem and the amount $T^n(U^n, V^{n-1})$ supplied by the n th subsystem, we can define the vector of excess demands for V^n as

$$E^n(P) = \tilde{V}^n - T^n(\tilde{U}^n, \tilde{V}^{n-1}) \quad (A-17)$$

so that at $P = \bar{P}$, $E^n(P) = 0$, $n = 1, 2, \dots, N$. Note that P is $[p^1, p^2, \dots, p^N]$. If $E^n(P) = 0$ for all n , then the conditions given by equations (A-11) through (A-16) are satisfied, which implies $P = \bar{P}$. Hence, it can be stated that $P = \bar{P}$ if and only if $E^n(P) = 0$ for all n .

The second level adjusts the parameters P by a price adjustment rule suggested by Samuelson (1949):

$$\frac{dp^n}{dt} = E^n(P) \quad (A-18)$$

With excess demands, $E^n(P)$, formed from the lower levels,

the higher level can apply a finite difference approximation to the price adjustment rule given by the above equation (A-18), which will yield convergence to the optimal prices $\bar{P}$ from the feasible initial guess P_0 .

Now the operation of the multilevel scheme can be described. Sequentially it proceeds as follows:

1. The second level sends to the first level an initial set of parameters $p^1, p^2, \dots, p^N$.
2. Each of the first level units optimizes its subproblem using these parameters.
3. Inputs and outputs of the first level units are transmitted back to the second level which forms the excess demands $E^n(P)$.
4. If these excess demands $E^n(P)$ are non zero, the second level adjusts the parameters P by the price adjustment rule such that the difference is reduced and transmits these new parameters back to the first level units.
5. The process is repeated until the excess demands are all zero, at which the solution is optimal.

Feasible and Infeasible decompositions

As described earlier, the first phase of multilevel approach is to decompose the overall system into independent subsystems and is accomplished either relaxing the subsystem interconnecting constraints, that is, the performance equations such as equations (A-2), or considering the state variables V^n , in our problem, as variable parameters. This isolates each subsystem from other subsystems and creates independent problems.

In such a method, the overall system equations are not satisfied except at the solution of the integrated problem. This method is named as the infeasible decomposition method in contrast to the feasible decomposition method in which the system equations of the subproblems always satisfy the integrated system equations. Following discussion shows the contrast between these two methods clearly.

The algorithm described on the preceding page is the essence of an infeasible decomposition method. Recall that equation (A-16) was relaxed and the subproblems were optimized for given p^n . Only at the final iteration, when $E^n(P)$ given by equation (A-17) is zero, the system Lagrangian satisfies all the necessary conditions.

In feasible decomposition method, equation (A-14) instead of (A-16) is relaxed. Therefore, the subsystems are optimized as unconstrained problems and the constraints are satisfied at the coordinating (second) level by adjusting the Kuhn-Tucker multipliers λ^n . Note that equations (A-11), (A-12), (A-15) and (A-16) are the necessary conditions for both L^n (the subsystem Lagrangian) as well as L (the system Lagrangian) to be stationary with respect to all arguments. These necessary conditions are satisfied at all iterations during coordination. A computational algorithm for this two level structure proceeds as follows:

1. Assume values for V^n and n , and send them to the first level (subproblems).
2. Solve each of the first level problems. This yields $\tilde{U}^n$, $\tilde{p}^n$ and $\tilde{S}^n$.

3. Send the first level solutions to the second level which forms the quantities $\frac{\partial L}{\partial v^n}$ and $\frac{\partial L}{\partial \lambda^n}$, i.e., it checks equations (A-14) and (A-15).
4. If λ^n and R^n do not satisfy equations (A-14) and (A-15), a new set of V^n and λ^n is generated as follows:

$$V^n = V^n - k \frac{L}{V^n}$$

$$\lambda^n \text{ if } \lambda^n = 0 \text{ and } \frac{\partial L}{\partial \lambda^n} \geq 0$$

$$\lambda^n + k \frac{\partial L}{\partial \lambda^n} \text{ if } \lambda^n > 0$$

k is a suitable positive constant.

5. Send new values of V^n and λ^n to the first level and iterate from step 2 until $\frac{\partial L}{\partial v^n} = 0$ and $\frac{\partial L}{\partial \lambda^n} > 0$.

The above two algorithms, for infeasible and feasible decomposition methods respectively, are guaranteed to converge to a global minimum of the objective function for convex F and concave R^n . The details are given elsewhere (Lasdon, 1964 and Arrow, et al., 1963). The only other requirement for the feasible decomposition method is that the decision vector U^n has at least as many components as the state vector V^n , and the Jacobian matrix for $T^n(U^n, V^{n-1})$ is of full rank for all U^n and V^n .

SOLUTION BY DISCRETE MAXIMUM PRINCIPLE

Recall that the problem is given by equations (A-1) through (A-5). In this section we attempt to solve the same problem using the algorithm of extended version of the discrete maximum principle, and then this methodology will be compared

to that of multilevel approach in the next section.

To bring this problem into a format that can be solved by the discrete maximum principle, a new state variable V_{m+1}^n is introduced such that

$$V_{m+1}^n = V_{m+1}^{n-1} + F^n(U^n, S^n) \quad (B-1)$$

with

$$V_{m+1}^0 = 0$$

so equation (A-1) can be written as:

$$\begin{aligned} F &= \sum_{n=1}^N \left[V_{m+1}^n - V_{m+1}^{n-1} \right] \\ &= V_{m+1}^N \\ &= \sum_{j=1}^{m+1} c_j V_j^N \end{aligned} \quad (B-2)$$

$$\begin{aligned} \text{with } c_j &= 0, \quad j = 1, 2, \dots, m \\ &= 1, \quad j = m+1 \end{aligned} \quad (B-3)$$

Equation (B-2) will be the objective function to be optimized.

For a process with constraints on state and/or decision variables, as given by equation (A-5), the necessary conditions for optimum can be obtained by combining the Kuhn-Tucker conditions (Kuhn and Tucker, 1950) with the algorithm of the discrete maximum principle (Fan and Wang, 1964). The procedure for solving such an optimization problem, is then to introduce an $(m+1)$ dimensional adjoint vector $\underline{Z}^n$ and a Hamiltonian function H^n as

$$H^n = (\underline{Z}^n)^T \underline{V}^n + (\lambda^n)^T R^n \quad (B-4)$$

satisfying

$$Z_j^{n+1} = \frac{\partial H^n}{\partial V_j^{n-1}} \quad (B-5)$$

$$z_j^N = c_j \quad (B-6)$$

$$\frac{\partial H^n}{\partial U^n} = 0 \text{ or } H^n = \text{minimum} \quad (B-7)$$

$$\frac{\partial H^n}{\partial z^n} = v^n = T^n(U^n, v^{n-1}) \quad (B-8)$$

$$n = 1, 2, \dots, N$$

$$j = 1, 2, \dots, m+1$$

Note that $\underline{z}^n$ and $\underline{v}^n$ are $(m+1)$ dimensional vectors. The necessary conditions for a saddle point in $\bar{U}^n$ and $\bar{\lambda}^n$ ($\bar{\lambda}^n \geq 0$), that is, H^n be a minimum with respect to U^n and a maximum with respect to λ^n , in addition to the equations (B-5) through (B-8) are the Kuhn-Tucker conditions:

$$\lambda^n \geq 0 \quad (B-9)$$

$$R^n \geq 0 \quad (B-10)$$

$$(\lambda^n)^T R^n \geq 0 \quad (B-11)$$

From equation (B-5)

$$\begin{aligned} z_{m+1}^{n-1} &= \frac{\partial H^{n-1}}{\partial v_{m+1}^{n-1}} \\ &= \frac{\partial}{\partial v_{m+1}^{n-1}} \left[(\underline{z}^n)^T \underline{v}^n + (\lambda^n)^T R^n \right] \\ &= \frac{\partial (z_{m+1}^n v_{m+1}^n)}{\partial v_{m+1}^{n-1}} \\ &= z_{m+1}^n \frac{\partial (v_m^{n-1} F^n(U^n, v^{n-1}))}{\partial v_{m+1}^{n-1}} \\ &= z_{m+1}^n \end{aligned} \quad (B-12)$$

From equations (B-3) and (B-6)

$$Z_{m+1}^N = 1$$

$$Z_{m+1}^n = 1, \quad n = 1, 2, \dots, N.$$

From equations (A-2)

$$\begin{aligned} H^n = & (Z^n)^T T^n(U^n, V^{n-1}) + V_{m+1}^{n-1} + F^n(U^n, S^n) \\ & + (\lambda^n) R^n(U^n, S^n, V^{n-1}) \end{aligned} \quad (B-13)$$

Working with this form of the Hamiltonian function, the necessary conditions for optimal H^n as given by equations (B-5) through (B-8), can be rewritten as

$$\begin{aligned} Z^{n-1} &= \frac{\partial H^n}{\partial V^{n-1}} \\ &= (Z^n)^T \frac{\partial T^n}{\partial V^{n-1}} + (\lambda^n)^T \frac{\partial R^n}{\partial V^{n-1}} + \frac{\partial F^n}{\partial V^{n-1}} \end{aligned}$$

or

$$\frac{\partial F^n}{\partial V^{n-1}} + (Z^n)^T \frac{\partial T^n}{\partial V^{n-1}} - Z^{n-1} + (\lambda^n)^T \frac{\partial R^n}{\partial V^{n-1}} = 0 \quad (B-14)$$

From (B-3) and (B-6)

$$Z_j^N = 0, \quad j = 1, 2, \dots, m \quad (B-15)$$

$$\frac{\partial H^n}{\partial U^n} = (Z^n)^T \frac{\partial T^n}{\partial U^n} + \frac{\partial F^n}{\partial U^n} + (\lambda^n)^T \frac{\partial R^n}{\partial U^n} = 0 \quad (B-16)$$

Since U^n is constrained by R^n , U^n need to be determined from (B-16) and then checked for the validity of constraints, or searching the boundary of the feasible region

to satisfy

$$H^n = \text{minimum}$$

Thus, the solution by the discrete maximum principle is simplified to solving the set of simultaneous equations (B-14) and (B-16) along with (A-2) and (B-15).

One of the difficulties in such methodology is the so solution of the so called two point (or split) boundary value problem, i.e., solving a set of simultaneous equations with mixed boundary conditions. It represents, in fact, the major difficulty and, although numerical analysts have given considerable attention to the two point boundary problem, the case that arises in the study of optimization tends to be particularly difficult (Noton, 1965). One way of solving this problem is by relaxation method (Merriam, 1964).

Recall that according to the discrete maximum principle, U^n is chosen to optimize H^n as a function of U^n . This requires, however, knowledge of the correct Z^n . Thus, there are two ways in approaching the optimal point: one starts from guessing the decision variable U^n (Steepest ascent of the Hamiltonian function), and other starts from guessing the adjoint variables Z^n (method of adjusting the adjoint variables) or the state variable V^n .

The method of adjusting the decision variable U^n is called the method of Steepest ascent or descent of the Hamiltonian depending on whether the objective is to maximize or minimize the objective function F .

I. Steepest descent of the Hamiltonian

This computational scheme as suggested by Arrow, et al. (1963) is a gradient search on the surface of H^n , which simultaneously descends in U^n (the minimizing variable) and ascends in λ^n (the maximizing variable). Note that such scheme is used in section 4 for the solution of Miami river problem by the discrete maximum principle. In this method, the variables U^n and λ^n are updated as

$$U^n = U^n - k \frac{\partial H^n}{\partial U^n} \quad (B-17)$$

$$\lambda^n = \lambda^n + k \delta^n \quad (B-18)$$

where k is a suitable positive constant, and

$$\delta_i^n = \begin{cases} 0 & \text{if } \lambda_i^n = 0 \text{ and } R_i^n > 0 \\ -R_i^n, & \text{otherwise} \end{cases} \quad (B-19)$$

The computation proceeds as follows:

1. Choose initial values for U^n and λ^n ,
 $n = 1, 2, \dots, N$.
2. Compute V^n by forward solution, i.e.,
 $n = 1, 2, \dots, N$, of performance equations (A-2)
 with specified initial conditions.
3. Obtain the values of the adjoint variables Z^n
 by backward solution, i.e., $n = N, N-1, \dots, 1$,
 of equation (B-14) with the boundary conditions
 (B-15).

4. Evaluate $\frac{\partial H^m}{\partial U^n}$ and δ^m from equations (B-16) and (B-19) respectively. If these values are nonzero, adjust U^n and λ^n by equations (B-17) and (B-18), and then return to step 2.
5. The process is repeated until $\frac{\partial H^m}{\partial U^n}$ and δ^m are zero (or acceptably small) at which the solution is optimal.

The above algorithm is guaranteed (Brosilo and Lasdon, 1964) to converge to global optimum of the objective F , provided F is convex and R^n are concave functions of U^n , $n = 1, 2, \dots, N$. It converges to at least a local optimum of F provided F is locally convex and R^n are locally concave.

To prove these assertions, it is necessary to prove that the gradient of the Hamiltonian without inequality constraints is the same as the gradient of the objective function F with respect to U^n . Such proof is given by Saaty and Bran (1964). Once we have desired proof, we can then fall back on the well known proofs (Letov, 1961) that the Lagrangian differential gradient method converges under the above conditions.

II. Adjusting the Adjoint variables

This computational scheme is based on the relaxation of the performance equations. The computations start by assuming a set of values for Z^n and λ^n . λ^n is adjusted according to equations (B-18) and (B-19) as before. The adjustment of the adjoint variables depend on the amount that the performance equations, (A-2), did not satisfy. That is

$$Z^n = Z^n + k [V^n - T^n(U^n, V^{n-1})]$$

or in other form

$$Z^n = Z^n - k \frac{\partial H^n}{\partial Z^n} \quad (B-20)$$

where k is a suitable positive constant.

The iterative procedure is as follows:

1. Initialize Z^n and λ^n , $n = 1, 2, \dots, N$.
2. Compute, according to equations (B-16) and (B-14), the values of U^n and V^{n-1} .
3. Compute $T^n(U^n, V^{n-1})$.
4. Adjust Z^n by equation (B-20) and λ^n by equation (B-18), and return to step 2.
5. The process is repeated till both $\frac{\partial H^n}{\partial Z^n}$ and δ^n are zero.

The above algorithm has been proved to converge asymptotically to the correct set of Z^n and λ^n , if such subobjective function H^n is atleast locally optimized for each set of Z^n (Lasdon, 1964).

COMPARISON OF THE TWO LEVEL APPROACH AND THE DISCRETE MAXIMUM PRINCIPLE

The multilevel approach as described is based on the decomposition of the Lagrangian of an integrated system. The necessary condition for the system to be extremum is that the Lagrangian be stationary with respect to all its arguments including the Lagrangian multipliers. By decomposing the Lagrangian of the original integrated problem, the whole system is decomposed into small subsystems. Then a coordination algorithm manipulates the Lagrangian multipliers of the subproblems to the point where the solutions of the subproblems

correspond to the solution of the original integrated problem.

On the other hand, by introducing the adjoint variables, the discrete maximum principle decomposes the overall extremum condition into the cascaded extremum conditions. It has been shown that for a system to attain at least its local extremum value, it is necessary to choose a set of decision vectors such that Hamiltonian for each unit is stationary or extremum. Therefore, if we can show that the Lagrange multipliers, p^n , in the multilevel system theory are the same as the adjoint variables, Z^n , in the discrete maximum principle, and that the adjoint functions, i.e.,

$$Z^{n-1} = \frac{\partial H^n}{\partial v^{n-1}}$$

are exactly the same as the necessary conditions that the Lagrangian be stationary with respect to its arguments, then the discrete maximum principle can be said to be equivalent to the multilevel theory.

However, the multilevel approach employs a price-adjustment rule for adjusting the multipliers to achieve the optimum of the original integrated problem. It also gives an economic interpretation to the subproblems resulting from the decomposition. The discrete maximum principle employs a similar method for adjusting the adjoint variables to achieve the optimum of the problem but makes use of no economic interpretation to the subproblem formulation.

By comparing the necessary conditions derived from the discrete maximum principle, equations (B-16), (B-14), (B-9), (B-11) and (B-10) respectively with those derived from the multilevel approach, equations (A-11) through (A-15), we see

that the Lagrange multipliers, p^n are equivalent to the adjoint variables Z^n in the discrete maximum principle. And, the adjoint function, $Z^{n-1} = \frac{\partial H^n}{\partial V^{n-1}}$, is actually a necessary condition that the Lagrangian be stationary with respect to its arguments V^{n-1} (equations (B-5) and (A-12) respectively).

By comparing the computational schemes, it is easy to recognize that the price adjustment rule, equation (A-17), used in the second level problem of the multilevel approach is exactly the same as the rule for adjusting the adjoint variables, equations (B-20).

In the feasible decomposition method of multilevel approach it is assumed that the decision vector U^n has at least as many components as the state vector V^n , and the Jacobian matrix of $T^n(U^n, V^{n-1})$ is of full rank of all U^n and V^n . It is easy to recognize that if the components of vector U^n and V^n are exactly the same, the feasible decomposition method and the Steepest descent of the Hamiltonian method suggested in the preceding section will be identical. However, if the components of the decision vector U^n are fewer than the components of the state vector V^n , then the feasible decomposition fails. This appears to be the weak point of the feasible decomposition method. On the other hand, if the components of the decision vector U^n are more numerous than the components of the state vector V^n , the steepest descent of Hamiltonian method fails. This means that the system will be over-determined when all the decision variables are specified, that is, the number of decision variables will be greater than the number of state variables. In this case, we should use the feasible decomposition method. Or, if the state variables are interchanged with

the decision variables, the method of the steepest descent of the Hamiltonian can be used.

APPENDIX II

COMPUTER PROGRAM FOR PSEUDO DECENTRALIZED PROCESS
BY TWO LEVEL APPROACH

A listing of the computer program, written in FORTRAN IV, follows Table 9 in which the Fortran symbols used are explained. Figure 8 shows the flow diagram for the computational procedure described in section (3.2.1). The program was compiled on G-level Fortran compiler (4.68 sec). The execution time for reaching the maximum (63 iterations) is 19.2 sec on IBM 370/158.

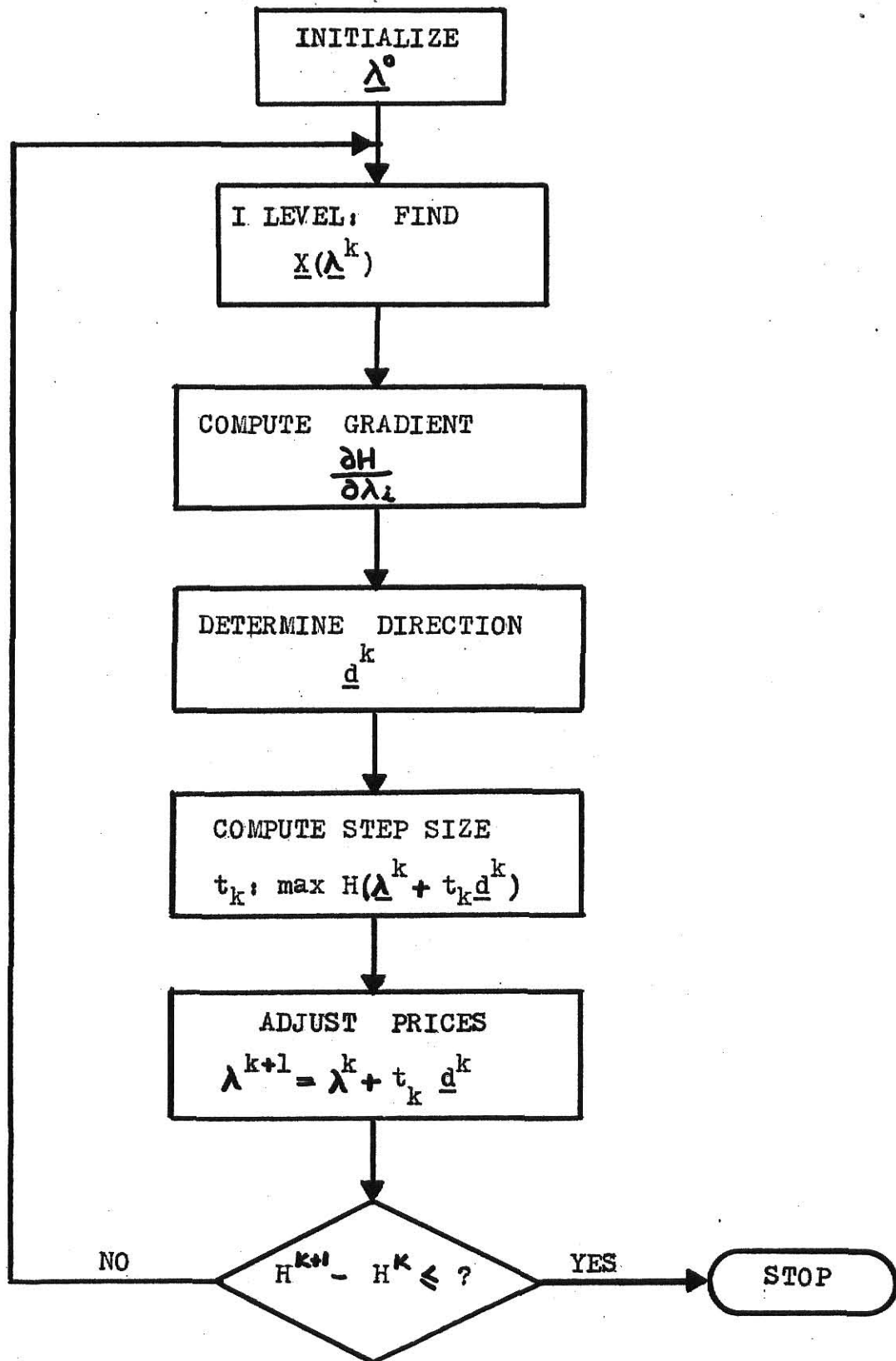


Figure 8. Pseudo Decentralized Process by two level approach

Table 9. Explanation of Fortran symbols in the computer program for pseudo decentralized process by two level approach

FORTTRAN SYMBOL	EXPLANATION	MATHEMATICAL SYMBOL
A	pounds of DO required in reach i per pound of BOD discharged by jth polluter	a_{ij}
B	RHS value in the constraint set, $a_{ij} x_j \leq S_i - Y b_i, Y \leq 1$	$S_i - Y b_i$
C	coefficients in quadratic cost function	C_{jk}
P	shadow prices	λ_i
D	gradient components	d_i
X	treatment levels	x_j
NREACH	number of reaches	(27)
NPOL	number of polluters	(15)
CODE	reach number of each polluter	
BL	upper limit on step size	
CL	inializes step size upper limit	(1)
T	step size	t_k
L	sub-Lagrangians	L_j
H1, H2	dual function values	
P1, P2	primal function values	
IDATA	0 if no data output	(1)
ITER	maximum number of iterations	(100)

FORTRAN SYMBOL	EXPLANATION	MATHEMATICAL SYMBOL
<u>SUBPROGRAMS</u> PRIMAL DUAL TREAT CHECK STEP TECHNQ S	<u>FOR COMPUTING:</u> primal dual treatment levels upper limit on step size normalized gradient vector step size by Fibonacci search (supplementing program for TECHNQ subprogram)	

```

$JOB          PSR,PAGES=50,TIME=(1,20)
C             PSEUDO DECENTRALIZED PROCESS
C             NO AUGMENTATION MODEL
C             DATA ON MIAMI RIVER DATA, HASS(1970)
C             REPRODUCTION OF RESULTS REPORTED BY HAIMES(1972B)
C             DR. C. L. HWANG KANSAS STATE UNIVERSITY
C             SUDHAKAR REDDY PAIDY SPRING 1976
C
1             COMMON A(50,50), B(50), C(50,3), NREACH, NPCL, X(50)
2             REAL P(50), D(50), AP(50)
3             INTEGER CODE(50)
4             READ 100, NREACH, NPCL, ITER, IDATA
5             100 FORMAT (18X,I2,18X,I2,17X,I3, 9X,I1)
6             READ 101, (CODE(J), J=1,NPCL)
7             101 FORMAT(40I2)
8             DO 10 I=1,NREACH
9             READ 102, ( A(I,J), J=1,NPOL)
10            102 FORMAT( 8F10.2 )
11            10 READ 103, B(I)
12            103 FORMAT(F10.2)
13            DO 20 J=1,NPOL
14            20 READ 104, (C(J,K), K=1,3)
15            104 FORMAT (4F20.2)
16            READ 105, (P(I), I=1,NREACH)
17            105 FORMAT(16F5.3)
18            CALL TREAT(P)
19            IF(IDATA.EQ.0) GOTO 701
20            PRINT 178
21            178 FORMAT('1')
22            DO 55 I=1,NREACH
23            PRINT 599, (A(I,J), J=1,NPOL)
24            55 PRINT 598, B(I)
25            PRINT 666
26            DO 44 J=1,NPOL
27            44 PRINT 597, (C(J,K), K=1,3)
28            PRINT 666
29            599 FORMAT(' ', 'AVALUES', 10F12.2)
30            598 FORMAT(' ', 'BVALUE ', F12.2/ )
31            597 FORMAT(' ', 'CVALUES', 3F12.2)
32            701 PRINT 178
33            DO 90 K=1, ITER
34            CL=1.
35            PRINT 692, (X(J), J=1,NPOL)
36            692 FORMAT(' ', 'TREAT', 10F10.4)
37            H1 = DUAL(P)
38            P1=PRIMAL(0.)
39            CALL STEP(P,D)
40            IF(K.EQ.1) PRINT 604, (P(I), I=1,NREACH)
41            PRINT 691, (D(I), I=1,NREACH)
42            691 FORMAT(' ', 'STEPS', 10F10.4)
43            IF(K.EQ.1) PRINT 600, H1, P1
44            600 FORMAT(' ', 'INITIAL DUAL H1', F15.3, 5X, 'INITIAL PRIMAL', F15.3)
45            604 FORMAT(' ', 'PRICES', 15F8.3)
46            CALL CHECK(BL,CL,D,P)
47            PRINT 633, BL
48            633 FORMAT(' ', 'UPPER LIMIT ON STEP SIZE', F10.5)
49            CALL TECHNQ (P,T,BL)
50            DO 53 I=1,NREACH
51            P(I)=P(I)+T*D(I)

```

```
52      IF( P(I).LT.0.001) P(I)=0.
53 53 CONTINUE
54      CALL TREAT(P)
55      H2 = DUAL(P)
56      P2=PRIMAL(0.)
57      ITR = K
58      PRINT 602, K, H2, P2
59      PRINT 603, (I,I=1,NREACH)
60      PRINT 576, (P(I), I=1,NREACH)
61 576 FORMAT(' ', 'CHECK P', 15F8.3)
62      PRINT 605, (CODE(N), N=1, NPOL)
63      PRINT 606, (J, J=1, NPCL)
64      PRINT 607, (X(J), J=1, NPCL)
65      PRINT 666
66 50 CONTINUE
67 602 FORMAT(' ', 'ITERATION', I4, 5X, 'DUAL', F15.3, 5X, 'PRIMAL', F15.3, '+',
        *100(' ', ' '))
68 603 FORMAT(' ', 'REACHES:', 30I4)
69 605 FORMAT(' ', 'POLLUTING REACH', 20I5)
70 606 FORMAT(' ', 'POLLUTER NUMBER', 20I5)
71 607 FORMAT(' ', 'TREATMENT LEVEL', 20F5.2, '////')
72 608 FORMAT(' - ', 'THIS IS THE SOLUTION AFTER ', I4, 2X, 'ITERATIONS')
73 666 FORMAT(' - ')
74 999 PRINT 608, ITR
75 900 STOP
76      END
```

```

77      FUNCTION PRIMAL(DUMMY)
78      COMMON A(50,50), B(50), C(50,3), NREACH, NPOL, X(50)
79      REAL F(50)
80      PRIMAL = DUMMY
81      DO 10 J=1,NPOL
82      F(J) = C(J,1)*X(J)**2 + C(J,2)*X(J) + C(J,3)
83      PRIMAL=PRIMAL+F(J)
84  10 CONTINUE
85      RETURN
86      END

```

C
C

```

87      FUNCTION DUAL(P)
88      COMMON A(50,50), B(50), C(50,3), NREACH, NPOL, X(50)
89      REAL AP(50), P(50), L(50)
90      BP=0.
91      DO 10 I=1,NREACH
92  10 BP =BP +P(I)*B(I)
93      DUAL=BP
94      DO 40 J=1,NPOL
95      AP(J) =0.0
96      DO 20 I=1,NREACH
97      AP(J)=AP(J)+P(I)*A(I,J)
98  20 CONTINUE
99      L(J)=C(J,1)*X(J)**2 + C(J,2)*X(J) + C(J,3) - AP(J)*X(J)
100     DUAL = DUAL + L(J)
101  40 CONTINUE
102     RETURN
103     END

```

```

104      SUBROUTINE TREAT(P)
105      COMMON A(50,50), B(50), C(50,3), NREACH, NPOL, X(50)
106      REAL AP(50),P(50)
107      DO 40 J=1,NPOL
108      AP(J)=0.
109      BP=0.
110      DO 30 I=1,NREACH
111      AP(J)=AP(J)+P(I)*A(I,J)
112      30 BP =BP +P(I)*B(I)
113      X(J) = (AP(J)-C(J,2)) / (2.*C(J,1))
114      IF (X(J).LE.0.45) X(J) = 0.45
115      IF (X(J).GT.0.99) X(J)=0.99
116      40 CONTINUE
117      RETURN
118      END

```

C
C

```

119      SUBROUTINE CHECK(BL,CL,D,P)
120      COMMON A(50,50), B(50), C(50,3), NREACH, NPOL, X(50)
121      REAL D(50), P(50), PP(50),PR(50)
122      DO 200 I=1,NREACH
123      200 PP(I) = P(I)
124      201 CLIN=0.
125      DO 202 I=1,NREACH
126      PR(I)=PP(I)+CL*D(I)
127      IF (PR(I).LT.0.) CL= -PP(I) / D(I)
128      IF( PR(I).LT.0.) GOTO 201
129      202 CONTINUE
130      IF (CL.LE.0.) PRINT 209
131      209 FORMAT(' ','ERRCR - CL IS NONPOSITIVE')
132      BL=CL
133      RETURN
134      END

```

```

135      SUBROUTINE STEP(P,D)
136      COMMON A(50,50), B(50), C(50,3), NREACH, NPCL, X(50)
137      REAL C(50), DH(50), P(50)
138      DO 10 I=1,NREACH
139      DH(I) = B(I)
140      DO 20 J=1,NPCL
141      DH(I) = DH(I) - A(I,J) * X(J)
142      IF(P(I).GT.0.00 ) C(I) = DH(I)
143      IF (P(I).EQ.0..AND.DH(I).LT.0.) D(I)=0.0
144      IF (P(I).EQ.0..AND.CH(I).GE.0.) D(I)=DH(I)
145      10 CONTINUE
146      ANORM = 0.
147      DO 30 I=1,NREACH
148      30 ANCRM = ANCRM + C(I)**2
149      RNCRM = SQRT(ANCRM)
150      DO 40 I=1,NREACH
151      40 D(I)=D(I)/RNCRM
152      RETURN
153      END

```

C
C
C

```

154      FUNCTION S(P ,D,T)
155      COMMON A(50,50), B(50), C(50,3), NREACH, NPCL, X(50)
156      REAL P (50), C(50),PD(50)
157      DO 10 I=1,NREACH
158      PD(I)=P(I)
159      PD(I)=PD(I)+T*D(I)
160      IF (PD(I).LE.0.) PD(I)=0.0
161      10 CONTINUE
162      S = DLAL(PD)
163      RETURN
164      END

```

```

165      SUBROUTINE TECHNQ (P,T,BL)
166      COMMON A(50,50), B(50), C(50,3), NREACH, NPCL, X(50)
167      REAL P(50), D(50)
168      CALL STEP(P,D)
169      AL=0.
170      X1=AL+C.618C3444E*(BL-AL)
171      X2=AL+BL-X1
172      FX1=S(P,D,X1)
173      FX2=S(P,D,X2)
174      EX=0.001 * (BL-AL)
175      N=0
176      1 DL=FX1-FX2
177      N=N+1
178      IF(DL.GE.0.) GOTO 2
179      BL=X1
180      X1=X2
181      X2=AL+BL-X1
182      FX1=S(P,D,X1)
183      FX2=S(P,D,X2)
184      GOTO 3
185      2 AL=X2
186      X2=X1
187      X1=AL+BL-X2
188      FX1=S(P,D,X1)
189      FX2=S(P,D,X2)
190      3 EPSI = ABS(BL-AL)
191      IF (EPSI.GT.EX) GOTO 1
192      IF (FX1.GE.FX2) GOTO 5
193      4 T=X2
194      H2=FX2
195      PRINT 610,T
196      PRINT 600,H2
197      600 FORMAT(' ','TECHCUAL',F15.5)
198      GOTO 6
199      5 T=X1
200      H2=FX1
201      PRINT 610,T
202      PRINT 600,H2
203      610 FORMAT(' ','INCREMENT',F20.6)
204      6 RETURN
205      END

```

\$ENTRY

APPENDIX III

ALGORITHM OF THE COMPLETE DECENTRALIZED PROCESS

This algorithm is essentially a steepest ascent description of the law of supply and demand. The coding of program was done in FORTRAN IV and compiled on WATFIV compiler (1.12). The execution time in reaching the optimal solution was 9.54 sec on IBM 370/158. Table 10 explains the Fortran symbols coded into the program. Figure 9 is the flow diagram for the algorithm. The numbers encircled within each block refer to the details of the process within that block as described below:

Statement of the problem

$$\underset{\underline{X}}{\text{Minimize}} \quad F = \sum_{j=1}^{15} F_j(x_j) + \phi(Y)$$

subject to

$$\sum_j a_{ij} x_j + Y b_i \geq S_i$$

$$i = 1, 2, \dots, 27$$

$$0.45 \leq x_j \leq 0.99, \quad j = 1, 2, \dots, 15$$

where

$$F_j(x_j) = c_{j1} x_j^2 + c_{j2} x_j + c_{j3}$$

The constants a_{ij} , b_{ii} and S_i are listed in Table 1 and c_{jk} in Table 2. $\phi(Y)$ is the augmentation cost.

First level problem

$$\underset{x_j}{\text{Minimize}} \quad L_j(x_j, \underline{\lambda})$$

subject to

$$0.45 \leq x_j \leq 0.99$$

where L_j is given by equation (3-16). From

$$\frac{\partial L_j}{\partial x_j} = 0,$$

$$x_j = \frac{\sum_{i=1}^{27} \lambda_i a_{ij} - c_{j2}}{2 c_{j1}} \quad (1)$$

$$0.45 \leq x_j \leq 0.99$$

Second level problem

$$\text{Maximize } H(\underline{\lambda})$$

subject to

$$\lambda_i G_i = 0,$$

$$i = 1, 2, \dots, 27$$

where

$$H(\underline{\lambda}) = \min_{\underline{x}} L(\underline{x}, \underline{\lambda})$$

and is given by equation (3-15). G_i is the slack in the constraint i and is given by

$$G_i = S_i - Y b_i - \sum_j a_{ij} x_j \quad (5)$$

The critical reach is defined as that reach which becomes the last to satisfy the condition

$$\lambda_i G_i = 0 \quad (6)$$

Step options

(2)

Option 1. Increase λ_i for all violating reaches. The step size $sl = 0.1$. This process continues till there are no constraint violations.

Option 2 (Feasible region, i.e., no constraint violations). Drive $\lambda_k \rightarrow 0$ where k is the reach for which $|G_k| \leq |G_i|$ for all i . If this step forces into infeasible region, chose step option 4 (one violation) or option 3 (more than one violation).

Option 3 (Infeasible region, more than one violations) Increase λ_i for all violating reaches to reduce the violations. In this process, the step size s_3 is reduced (by a factor 0.9) for each repeat of this option. When there exists only one constraint violation, chose option 4.

Option 4 (Infeasible region, critical constraint, c , i.e., the constraint that becomes feasible at last). Selectively increase λ_c . The step size $s_4 = 1$. Repeat till it forces the constraint set into feasible region, then chose option 5.

Option 5 (Feasible region, critical constraint known). Drive all $\lambda_i = 0$ except λ_c , the multiplier on critical reach. If there exists any violations after this choose option 4 again. Otherwise, the option 5 results in a critical reach.

Augmentation facility

The augmentation facility is introduced when the algorithm has obtained the critical reach. In the process of reducing the price λ_c , if violations occur, the augmentation rate is increased. The stopping criterion are depicted in Figure 9. The increment in augmentation rate is computed as

$$Y = G_H / B_H, \quad 1 \leq Y \leq 4.47$$

(3)

where G_H is the highest positive slack, in reach H , and B_H is the coefficient of Y in the constraint H .

Associated costs/revenue

④

$$\text{Augmentation cost (CA), } \phi(Y) = 726.6 Y, \text{ \$/day} \\ 1 < Y \leq 4.47$$

$$\text{Treatment cost (CT)} = \sum_{j=1}^{15} F_j(x_j) \quad \text{\$/day}$$

$$\text{Tax revenue (CR)} = \sum_i \sum_j \lambda_i a_{ij} (1-x_j) \quad \text{\$/day}$$

$$\text{Total cost (TC)} = \text{CA} + \text{CT} + \text{CR} \quad \text{\$/day.}$$

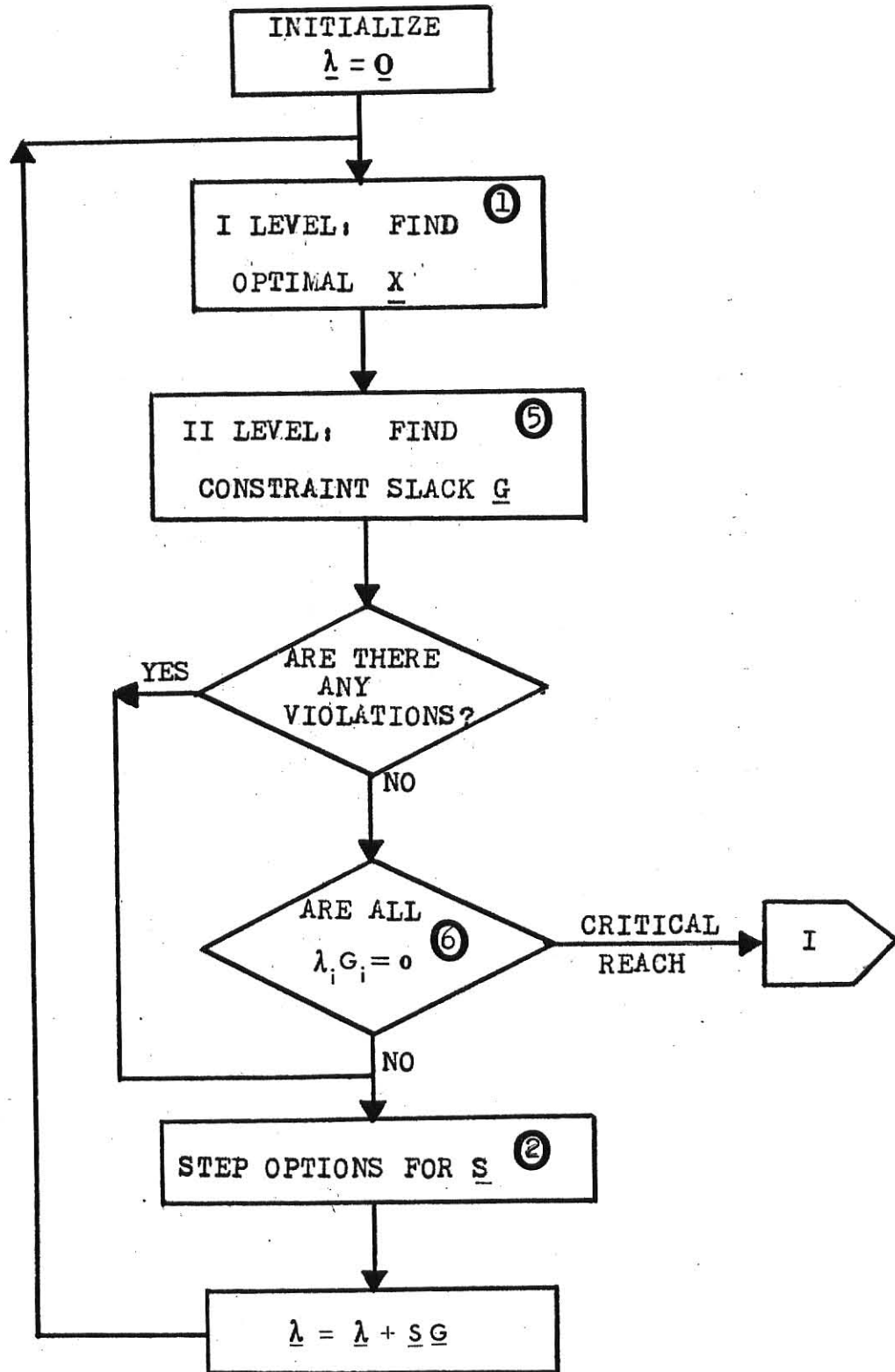


Figure 9. Complete Decentralized Process

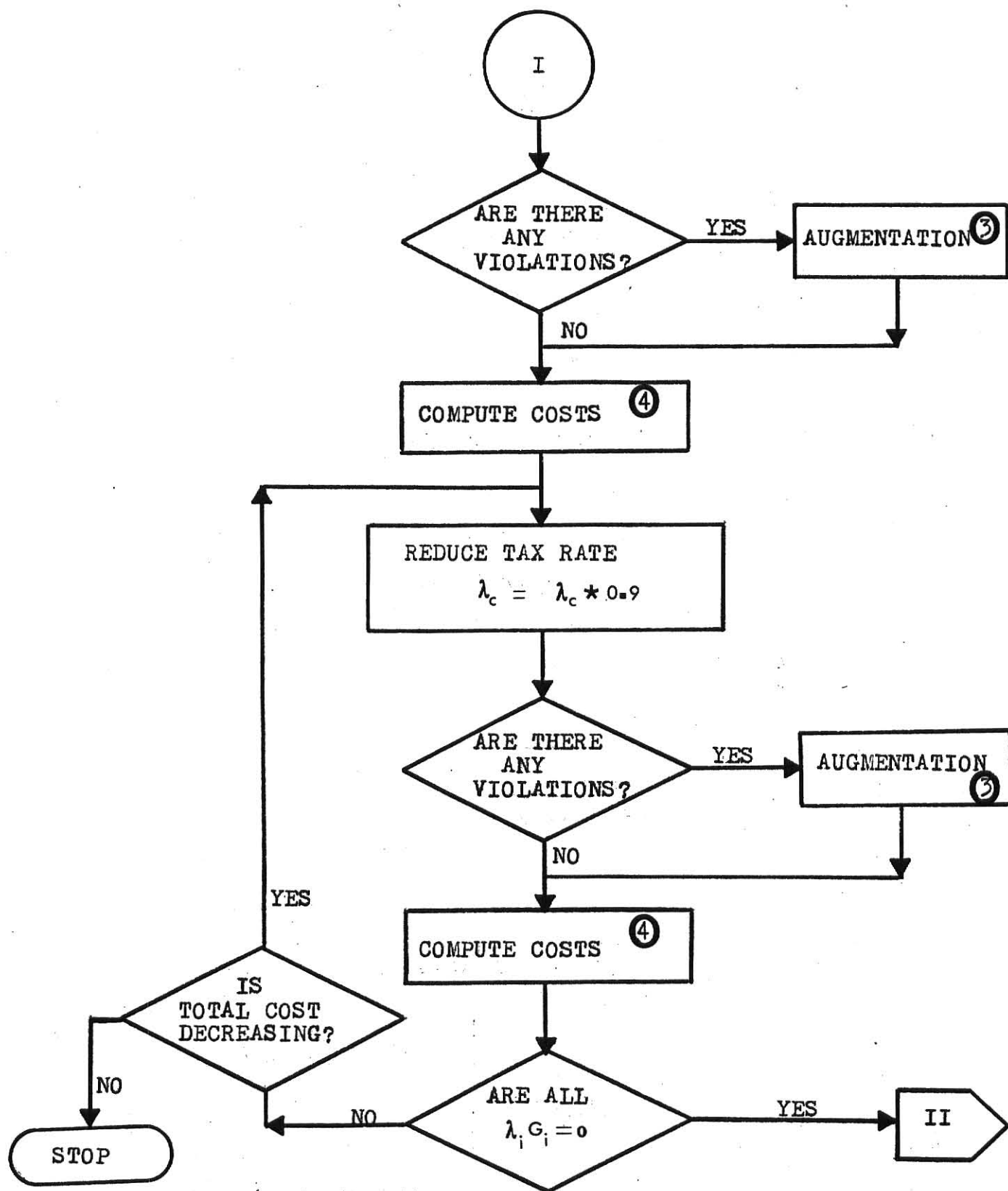


Figure 9 (continued)

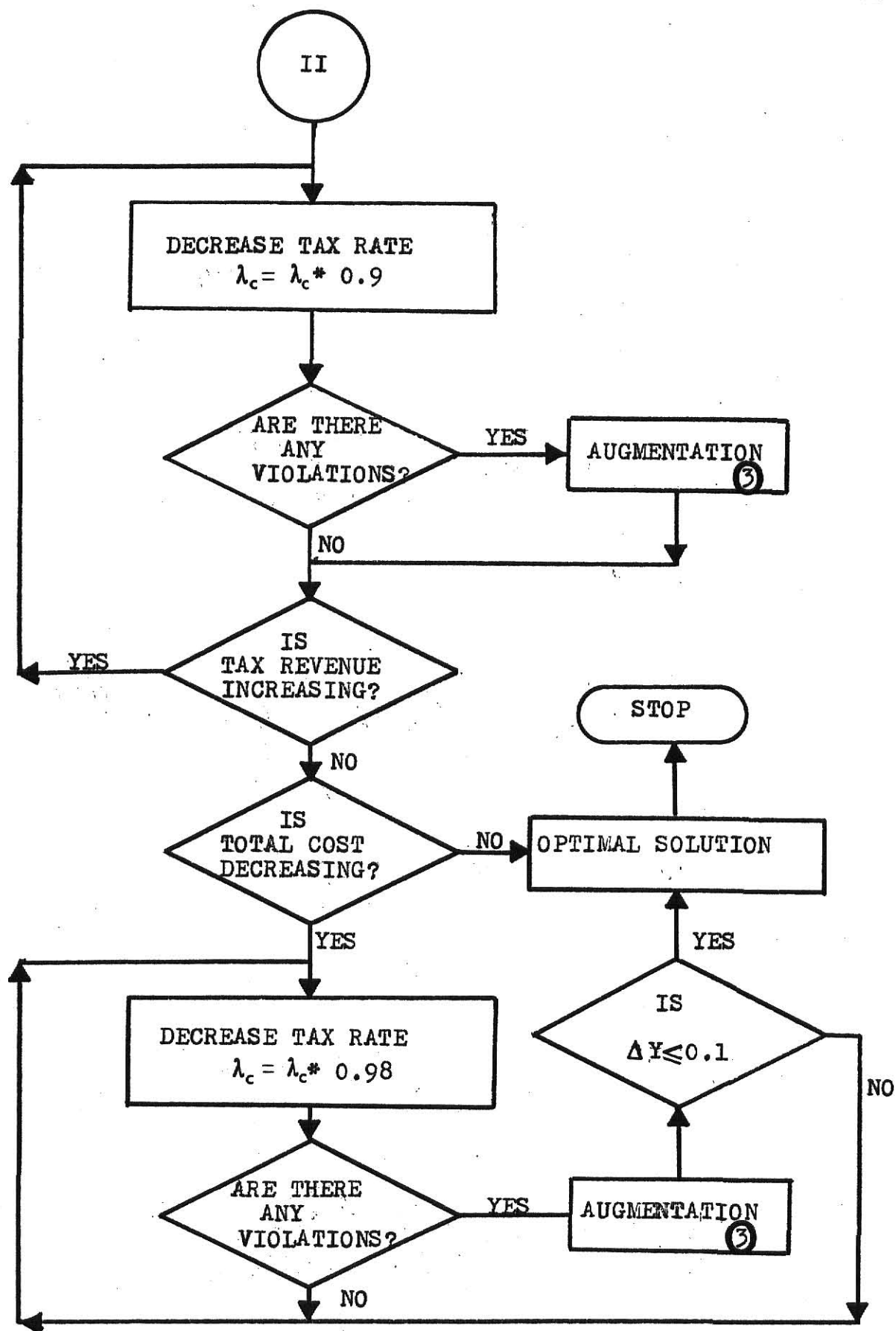


Figure 9 (continued).

Table 10. Explanation of Fortran symbols in computer program
for the complete decentralized process.

FORTRAN SYMBOL	EXPLANATION	MATHEMATICAL SYMBOL
NREACH	number of reaches	(27)
NPOL	number of polluters	(15)
CODE	reach number of each polluter	
ITER	maximum number of iterations	
IDATA	0 if no data out put	(1)
A	pounds of DO required in reach i per pound of BOD discharged by jth polluter	a_{ij}
B	17th column in constraint matrix	S_i
YC	16th column in constraint matrix	b_i
Y	increment in augmentation	ΔY
AGY	augmentation rate	Y
C	coeffients in quadratic cost function	c_{jk}
X	treatment levels	x_j
G	constraint slack	
IVLT	number of constraint violations	
ICOUNT	number of ($\lambda_i g_i = 0$) violations	
ICD	critical reach	
LL, LL1, AP, BP, GN	} internal variables	

Table 10.. (continued)

FORTRAN SYMBOL	EXPLANATION	MATHEMATICAL SYMBOL
TLCST1, TLCST2, . C1	Total cost	
TRCST1, TRCST2, C2	Treatment cost	
TAX1, TAX2, C3	Tax revenue	
AGCST1, AGCST2, C4	augmentation cost	
P	shadow price	λ_i
<u>SUBPROGRAMS</u>	<u>FOR COMPUTING</u>	
STEP	step options, step sizes	
VLTN	constraint violations	
CRITER	$(\lambda_i \xi_i = 0)$ violations	
ECONO	costs and revenue	
AGMTN	increments in augmentation rate.	

```

$JOB          PSR,PAGES=50,TIME=(1,20)
C      WATER QUALITY MANAGEMENT:DATA ON MIAMI RIVER (HASS,1970)
C      COMPLETE DECENTRALIZED MODEL OF HAIMES,1972
C      INTRODUCTION OF AUGMENTATION INTO THE MODEL
C      DR. C. L. HWANG   KANSAS STATE UNIVERSITY
C      SUDHAKAR REDDY PAICY   JAN 1976
C
1      COMMON NREACH, NPCL,LL(50), P(50), G(50),S1,S3
2      COMMON      A(50,50), B(50), C(50,3),  AP(50), X(50)
3      COMMON LL1(50), Y, YC(50), AGY
4      INTEGER CODE(50)
5      READ 100, NREACH, NPOL, ITER,  IDATA
6      100  FORMAT (18X,I2,18X,I2,17X,I3, 9X,I1)
7      READ 101, (CODE(J), J=1,NPOL)
8      101  FORMAT(40I2)
9      DO 11 I=1,NREACH
10     READ 102, ( A(I,J), J=1,NPOL)
11     102  FORMAT( 8F10.2 )
12     11  READ 103, B(I),  YC(I)
13     103  FORMAT (F10.2,F5.0)
14     DO 22 J=1,NPOL
15     22  READ 104, (C(J,K), K=1,3)
16     104  FORMAT (4F20.2)
17     READ 105, (P(I), I=1,NREACH)
18     105  FORMAT(16F5.3)
19     IF(IDATA.EQ.0) GOTO 701
20     PRINT 178
21     178  FORMAT('1')
22     PRINT 491, (J,J=1,NPOL)
23     PRINT 492, (CODE(N),N=1,NPCL)
24     492  FORMAT(' ','POLLUTING REACH',20I5//)
25     491  FORMAT(' ','POLLUTER NUMBER',20I5)
26     DO 55 I=1,NREACH
27     PRINT 599, (A(I,J), J=1,NPOL)
28     55  PRINT 598, B(I),  YC(I)
29     598  FORMAT(' ','BVALUES',2F12.2/ )
30     666  FORMAT(' - ')
31     PRINT 178
32     DO 44 J=1,NPOL
33     44  PRINT 597, (C(J,K), K=1,3)
34     599  FORMAT(' ','AVALUES', 10F12.2)
35     597  FORMAT('0','CVALUES', 3F12.2)
36     701  PRINT 178
C      NO AUGMENTATION AT THE BEGINING
37     AGY=1.
38     Y=AGY
39     DO 111 I=1,NREACH
40     111  B(I)=B(I)-Y*YC(I)
41     PRINT 606, (B(I), I=1,NREACH)
42     606  FORMAT(' ','VERBS*', 10F10.2)
43     DO 888 K=1,ITER
44     PRINT 699,K
45     699  FORMAT(' - ', 44('* ') /' ','I T E R A T I O N ',15//)
46     702  START=0.
C      COMPUTE X
47     DO 66 J=1,NPOL
48     AP(J)=0.
49     BP=0.
50     DO 33 I=1,NREACH

```

```

51      AP(J)=AP(J)+P(I)*A(I,J)
52      BP  =BP  +P(I)*B(I)
53      X(J) = (AP(J)-C(J,2)) / (2.*C(J,1))
54      IF (X(J).LE.0.45) X(J) = 0.45
55      IF (X(J).GT.0.99) X(J)=0.99
56      66 CONTINUE
57      DO 10 I=1,NREACH
58      SUM = 0.
59      DO 20 J=1,NPOL
60      SUM=SUM+A(I,J) * X(J)
61      G(I) = B(I) - SUM
62      10 CONTINUE
63      PRINT 604, (X(J),J=1,NPOL)
64      PRINT 605, (P(I),I=1,NREACH)
65      PRINT 603, (G(I),I=1,NREACH)
66      605 FORMAT(' ', 'PRICES',10F10.4)
67      604 FORMAT(' ', 'TREATS',10F10.4)
68      603 FORMAT(' ', 'SLACKS',10F12.2)
69      378 DUMMY=0.
C      CHECK THE VIOLATIONS
70      IVLT=C
71      DO 30 I=1,NREACH
72      IF(G(I).GT.0.) IVLT=IVLT+1
73      IF(G(I).GT.0.) IM=I
74      IF(IVLT.GT.0) LL(IVLT)=IM
75      30 CONTINUE
76      PRINT 621, IVLT
77      621 FORMAT(' ', 'VIOLATIONS', I5)
78      IF(IVLT.GT.0) PRINT 622, (LL(N),N=1,IVLT)
79      622 FORMAT(' ',30I4)
80      ICOUNT=0
81      IF(IVLT.GT.0) GOTO 705
82      DO 50 I=1,NREACH
83      IF ((P(I)*G(I)).NE.0.) ICOUNT=ICOUNT+1
84      IF ((P(I)*G(I)).NE.0.) IN=I
85      IF(ICOUNT.GT.0) LL1(ICOUNT) = IN
86      50 CONTINUE
87      PRINT 623, ICOUNT
88      623 FORMAT(' ', 'VIOLATED CRITERION', I5)
89      IF (ICOUNT.EQ.0) GOTO 991
90      705 CALL STEP(ICOUNT,IVLT,K,&702,&991)
91      888 CONTINUE
92      PRINT 696, ITER
93      696 FORMAT(' ', '##### NO SOLUTION IN', I4, ' ITERATIONS #####')
94      GOTO 999
95      991 PRINT 691
96      691 FORMAT('-', 'CONVERGENCE ATTAINED * THE SOLUTION FOLLOWS'////)
97      PRINT 604, (X(J),J=1,NPOL)
98      PRINT 666
99      PRINT 605, (P(I),I=1,NREACH)
100     PRINT 666
101     PRINT 603, (G(I),I=1,NREACH)
102     PRINT 178
103     999 STOP
104     END

```

```

105      SUBROUTINE STEP(ICOUNT,IVLT,K,*,*)
106      COMMON NREACH, NPOL,LL(50), P(50), G(50),S1,S3
107      COMMON      A(50,50), B(50), C(50,3), AP(50), X(50)
108      COMMON LL1(50), Y, YC(50), AGY
109      DIMENSION GN(50)
110      C      THE JUNCTION FOR THE STEP OPTIONS
111      IF (ICOUNT.EQ.1) GOTO 5
112      IF (IVLT.EQ.0) GOTO 2
113      IF (K.GT.1) GOTO 3
114      C      STEP1: ATTAIN THE FEASIBLE REGION
115      S1=0.1
116      DO 10 N=1,IVLT
117      10 P(LL(N)) = P(LL(N)) + S1
118      RETURN1
119      2 PRINT 661
120      661 FORMAT(' ',T2,40(' '), T53, 'FEASIBLE REGION')
121      C      REDUCE THE TAX RATES
122      C      STEP2: FORCE THE P CT. LEAST POSITIVE G TO 0
123      IGC=0
124      DO 20 I=1,ICOUNT
125      IF(G(LL1(I)).LT.0.) IGC=IGC+1
126      IF(G(LL1(I)).LT.0.) GN(IGC) = -G(LL1(I))
127      20 CONTINUE
128      Q1 = GN(1)
129      KP=1
130      DO 30 NS=1,IGC
131      IF( GN(NS).GE.Q1 ) GOTO 30
132      Q1=GN(NS)
133      KP=NS
134      30 CONTINUE
135      DO 40 I=1,NREACH
136      IF(-GN(KP).EQ.G(I)) KK=I
137      PRINT 662, KK
138      662 FORMAT(' ', 'CRITICAL PRICE', I5)
139      STORE=P(KK)
140      P(KK)=0.
141      881 CALL VLTN(IVLT)
142      IF(IVLT.EQ.0) GOTO 99
143      IF(IVLT.EQ.1) GOTO 4
144      IF(IVLT.GT.1) P(KK)=STORE*0.9
145      STORE=P(KK)
146      GOTO 881
147      99 RETURN
148      C      STEP3: RECOVERY STEP PROCESS
149      3 S3=S1* 0.9
150      IF(S3.LT.0.01 ) S3=0.01
151      IF(IVLT.EQ.1) GOTO 4
152      DO 50 N=1,IVLT
153      50 P(LL(N)) = P(LL(N)) + S3
154      S1=S3
155      RETURN1
156      4 PRINT 179, LL(IVLT)
157      179 FORMAT(' ', 'CRITICAL CONSTRAINT IS', I5)
158      ICD=LL(IVLT)
159      S4=1.
160      P(LL(IVLT)) = P(LL(IVLT)) + S4
161      CALL VLTN(IVLT)
162      IF(IVLT.EQ.0) GOTO 6
163      IF( P(LL(IVLT)).GT.10 ) PRINT 663
164      IF( P(LL(IVLT)).GT.10 ) RETURN2

```

```

160      IF(IVLT.EQ.1)   GOTO 4
161      IF(IVLT.GT.1)   PRINT 663
162      663 FORMAT(' ', 'CHECK THE VALUE OF S4')
163      RETURN2
164      6   SPR=P(ICD)
165      DO 60 I=1,NREACH
166      60  P(I)=0.
167      P(ICD)=SPR
168      175 CALL VLTN(IVLT)
169      PRINT, 'CURRENT VALUES'
170      PRINT 604, (X(J), J=1, NPOL)
171      PRINT 605, (P(I), I=1, NREACH)
172      PRINT 603, (G(I), I=1, NREACH)
173      603 FORMAT(' ', 'SLACKS', 10F10.4)
174      604 FORMAT(' ', 'TREATS', 10F10.4)
175      605 FORMAT(' ', 'PRICES', 10F10.4)
176      IF(IVLT.EQ.0)   GOTO 411
177      CALL AGMTN
178      GOTO 175
179      411 CALL ECONO(TLCST1, TRCST1, TAX1, AGCST1, ICD)
180      STPR=P(ICD)
181      STY=AGY
182      P(ICD)=P(ICD)*0.9
183      PRINT 667, P(ICD)
184      667 FORMAT(' ', 'REDUCED TAX', F10.4)
185      CALL VLTN(IVLT)
186      IF(IVLT.GT.0)   CALL AGMTN
187      IF(AGY.GT.4.47) RETURN2
188      CALL ECONO(TLCST2, TRCST2, TAX2, AGCST2, ICD)
189      PRINT 682, TLCST2, TLCST1, AGY
190      CALL CRITER(85)
191      IF(TLCST2.LT.TLCST1) GOTO 411
192      682 FORMAT(' ', 'TLCST2', F13.2, 'TLCST1', F13.2, 'AGMTN', F10.4)
193      P(ICD)=STPR
194      AGY=STY
195      CALL VLTN(IVLT)
196      RETURN2
197  C   NOW INCREASE THE TAX AND PRINT ALL VALUES
198      5   PRINT 182, ICD, P(ICD), G(ICD)
199      182 FORMAT(' ', 'FINE TUNING STAGE', I10, 2F10.4)
200      441 CALL ECONO(TLCST1, TRCST1, TAX1, AGCST1, ICD)
201      STPR=P(ICD)
202      STY=AGY
203      P(ICD)=P(ICD)*0.9
204      PRINT 667, P(ICD)
205      CALL VLTN(IVLT)
206      IF(IVLT.GT.0)   CALL AGMTN
207      IF(AGY.GT.4.47) RETURN2
208      CALL ECONO(TLCST2, TRCST2, TAX2, AGCST2, ICD)
209      PRINT 682, TLCST2, TLCST1, AGY
210      IF(TAX2.LT.TAX1) GOTO 443
211      IF(TLCST2.LT.TLCST1) GOTO 441
212      RETURN2
213      443 P(ICD)=STPR
214      473 STPR=P(ICD)
215      P(ICD)=P(ICD)*0.98
216      CALL VLTN(IVLT)
217  C   IF(IVLT.EQ.0)   GOTO 473
218      C   CONSTRAINING THE ICR. IN AGMTN
219      CALL AGMTN

```

```
218      IF(Y.GT.0.1 ) GOTO 473
219      CALL ECONO (C1,C2,C3,C4,ICD)
220      PRINT 689, C1,C2,C3,C4,AGY,ICD,P(ICD)
221      689 FORMAT('---','THE OPTIMAL DAILY CCSTS AND THE POLICY ARE AS FOLLOW'/
1'0','TOTAL COST $',T25,F13.2/'0','COST OF TREATMENT $',T25,F13.2//
2' TAX REVENUE $',T25,F13.2/'0', 'AUGMENTATION COST $',T25,F13.2,8X
3,'AUGMENTATION',F8.4/'0','CRITICAL CONSTRAINT',I4,10X,'SHADOW PRIC
4E',F8.4)
222      CALL VLTN(IVLT)
223      RETURN2
224      END
```

```

225      SUBROUTINE VLTN(IVLT)
226      COMMON NREACH, NPOL, LL(50), P(50), G(50), S1, S3
227      COMMON A(50,50), B(50), C(50,3), AP(50), X(50)
228      COMMON LL1(50), Y, YC(50), AGY
229      DO 66 J=1, NPOL
230      AP(J)=0.
231      BP=0.
232      DO 33 I=1, NREACH
233      AP(J)=AP(J)+P(I)*A(I,J)
234      33 BP =BP +P(I)*B(I)
235      X(J) = (AP(J)-C(J,2)) / (2.*C(J,1))
236      IF (X(J).LE.0.45) X(J) = 0.45
237      IF (X(J).GT.0.99) X(J)=0.99
238      66 CONTINUE
239      DO 10 I=1, NREACH
240      SUM = 0.
241      DO 20 J=1, NPOL
242      20 SUM=SUM+A(I,J) * X(J)
243      G(I) = B(I) - SUM
244      10 CONTINUE
C      CHECK THE VIOLATIONS
245      IVLT=0
246      DO 30 I=1, NREACH
247      IF(G(I).GT.0.) IVLT=IVLT+1
248      IF(G(I).GT.0.) IM=I
249      IF(IVLT.GT.0) LL(IVLT)=IM
250      30 CONTINUE
251      PRINT 602, IVLT
252      602 FORMAT('0','CHECK VIOLATIONS',I5)
253      IF(IVLT.GT.0) PRINT 601, (LL(IS), IS=1, IVLT)
254      601 FORMAT(' ', 'CHECKS', 20I5)
255      RETURN
256      END

257      SUBROUTINE CRITER(*)
258      COMMON NREACH, NPOL, LL(50), P(50), G(50), S1, S3
259      COMMON A(50,50), B(50), C(50,3), AP(50), X(50)
260      COMMON LL1(50), Y, YC(50), AGY
261      ICRIT=0
262      DO 10 I=1, NREACH
263      IF((P(I)*G(I)).EQ. 0.) ICRIT=ICRIT+1
264      10 CONTINUE
265      PRINT 20, ICRIT, NREACH
266      20 FORMAT(' ', I2, ' SATISFIED CRITERION IN', I4)
267      IF(ICRIT.EQ.NREACH) RETURN
268      RETURN
269      END

```

```

270      SUBROUTINE ECONO ( TLCOST,TRCOST,TAX,AGCOST,ICD)
271      COMMON NREACH, NPOL,LL(50), P(50), G(50),S1,S3
272      COMMON      A(50,50), B(50), C(50,3), AP(50), X(50)
273      COMMON LL1(50), Y, YC(50), AGY
274      C      COMPUTE TAXES AND ALL KINDS OF COSTS INVOLVED
275      TAX=0.
276      DO 110 J=1,NPOL
277      110 TAX=TAX+A(ICD,J) * (1-X(J))
278      TAX=P(ICD)*TAX
279      AGCOST=726.6 * AGY
280      TRCOST=0.
281      DO 120 J=1,NPOL
282      120 TRCOST=TRCOST + C(J,1)*X(J)**2 + C(J,2)*X(J) + C(J,3)
283      PLCOST=TRCOST+TAX
284      TLCOST=TRCOST+TAX+AGCOST
285      PRINT 180,TLCOST,TRCOST,TAX,AGCOST
286      180 FORMAT(' ','COSTS:  TOTAL, TREATMENT, TAX, AUGMTN'/' ',4F13.3)
287      RETURN
288      END

288      SUBROUTINE AGMTN
289      COMMON NREACH, NPOL,LL(50), P(50), G(50),S1,S3
290      COMMON      A(50,50), B(50), C(50,3), AP(50), X(50)
291      COMMON LL1(50), Y, YC(50), AGY
292      C      DRIVE THE HIGHEST G TO ZERO
293      420 DUMMY=0.
294      DO 177 I=1,NREACH
295      IF(G(I).GT.0.) Q7=G(I)
296      IF(G(I).GT.0.) GOTO 167
297      177 CONTINUE
298      167 IQ=1
299      DO 133 I=1,NREACH
300      IF(G(I).LE.0.) GOTO 133
301      IF(G(I).LT. Q7) GOTO 133
302      Q7=G(I)
303      IQ=I
304      133 CONTINUE
305      C      FLOW AUGMENTATION
306      C      Y IS THE CHANGE IN AUGMENTATION
307      Y = G(IQ)/YC(IQ)
308      AGY=AGY+Y
309      PRINT 631, AGY
310      IF(AGY.GT.4.47) RETURN
311      631 FORMAT(' ','AUGMTN',F6.3)
312      DO 122 I=1,NREACH
313      122 B(I)=B(I)-Y*YC(I)
314      CALL VLTN(IVLT)
315      PRINT,'FINAL VOILATION',IVLT
316      PRINT 632,(G(I),I=1,NREACH)
317      632 FORMAT(' ','NETSSS',10F10.2)
318      IF(IVLT.GT.0) GOTO 420
319      RETURN
320      END

```

\$ENTRY

APPENDIX IV

COMPUTER PROGRAM FOR PSEUDO DECENTRALIZED PROCESSBY THE DISCRETE MAXIMUM PRINCIPLE

The algorithm as described in section (4.2) and shown in Figure 10, is coded into computer program in FORTRAN IV. It is compiled on WATFIV compiler (0.60 sec). The execution time, on IBM 370/158, is 51.52 sec for 400 iterations. The Fortran symbols used are explained in Table 11.

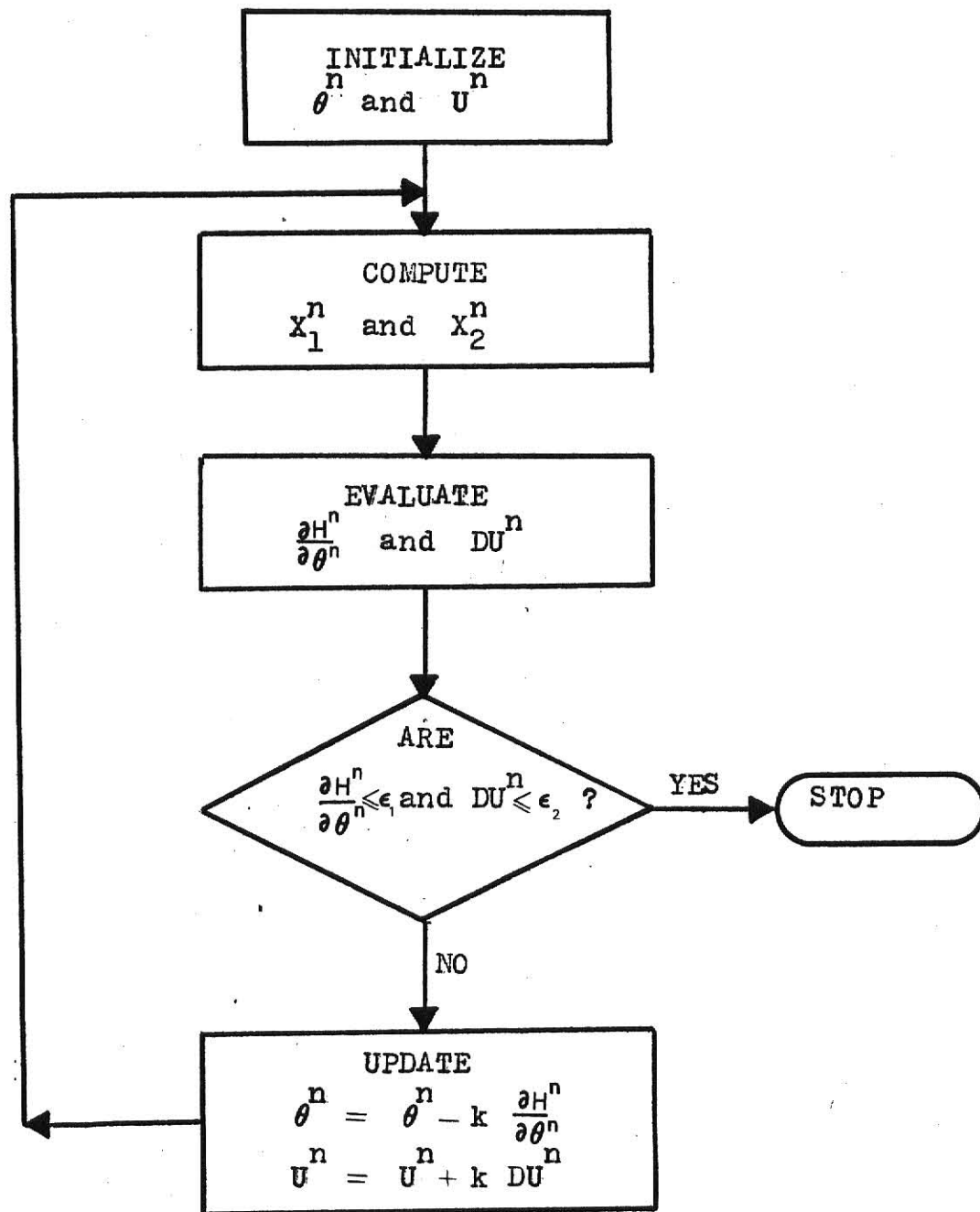


Figure 10. Pseudo Decentralized Process by discrete maximum principle

Table 11. Explanation of Fortran symbols in computer program for the pseudo decentralized process by the discrete maximum principle.

FORTTRAN SYMBOL	EXPLANATION	MATHEMATICAL SYMBOL
A	pounds of DO required in reach i per pound of BOD discharged by jth polluter	a_{ij}
B	17th column of constraint matrix	S_i
S	16th column of constraint matrix	b_i
X_1	DO available upto reach n	x_1^n
X_2	Total cost upto reach n	x_2^n
Z	adjoint variable	z_2^n
TH	treatment level	θ^n
U	shadow price	U^n
R	constraint value	R^n
IPDEX	(0,1) index to show if $\theta^n=0$ or not	
DU	increment for U	DU^n
DHN	increment for TH	$\frac{\partial H^n}{\partial \theta^n}$
c	coefficients in quadratic cost function	c_{jk}
TRCOST	cost of treatment	
TSUM	(internal variable)	
IDATA, IFR ISKIP, IJUMP, ITM	Output controllers	
KSTEP	step size	k

Table 11. (continued)

FORTRAN SYMBOL	EXPLANATION	MATHEMATICAL SYMBOL
EPSI1	stopping criteria on DHN	
EPSI2	stopping criteria on DU	
ISTU	satisfied criterion on DU	
ISTH	satisfied criterion on DHN	
DMAX	maximum DU	
DHMAX	maximum DHN	
TTCOST	total treatment cost	
TAX	total tax revenue	
AUGMTN	augmentation rate, $Y = 1$	

```

$JOB          PSR,TIME=(2,00),PAGES=100
1  REAL KSTEP
2  DIMENSION  A(27,27), B(27), C(27,3), X1(27), X2(27), Z(27),
*             TH(27), U(27), R(27), IPDEX(27), DU(27), DHN(27),
*             S(27), TRCOST(27), TSUM(27)
C  OUTPUT CONTROL PARAMETERS
3  READ 502, ICATA, IFR, ISKIP, IJUMP, ITM
4  502 FORMAT(5I5)
5  KSTEP=0.01
6  EPSI1=100
7  EPSI2=25
8  DO 500 I=1,27
9  READ 501, (A(I,J),J=1,27)
10 READ 501, S(I),B(I)
11 READ 501, (C(I,K),K=1,3)
12 IPDEX(I)=1
13 IF(C(I,1).EQ.0.) IPDEX(I)=0
14 500 CONTINUE
15 IF(IDATA.EQ.0)GOTO 101
16 DO 600 I=1,27
17 PRINT 503,I
18 PRINT 504,(A(I,J),J=1,27)
19 PRINT 505,B(I),S(I)
20 PRINT 506,(C(I,K),K=1,3)
21 600 CONTINUE
22 101 ITER=0
C  INITIALIZE THETA AND U VALUES
23 DO 210 I=1,27
24 TH(I)=0.
25 U(I)=C.1
26 IF(IPDEX(I).NE.0)TH(I)=0.99
27 210 CONTINUE
28 AUGMTN=1.
29 102 DUMMY=0.
C  EVALUATE CCNSTRAINT MAGNITUDE
30 DO 20 I=1,27
31 SUM=0.
32 DO 10 J=1,27
33 SUM=SUM+A(I,J)*TH(J)
34 20 R(I)=SUM+B(I)-S(I)
35 IVLT=0
36 DO 22 I=1,27
37 IF(R(I).LT.0.)IVLT=IVLT+1
38 22 CONTINUE
C  COMPUTE X2 BY FORWARD SOLUTION
39 DO 84 I=1,27
40 TSUM(I)=0.
41 DO 83 K=1,27
42 83 TSUM(I)=TSUM(I) + A(K,I) * U(K)
43 84 CONTINUE
44 TRCOST(I)=0.
45 X2(I)=0.
46 DO 50 I=2,27
47 TRCOST(I)= C(I,1)*TH(I)**2 + C(I,2)*TH(I) + C(I,3)
48 X2(I)=X2(I-1) + TRCOST(I) + TSUM(I) * (1.-TH(I))
49 50 CONTINUE
C  COMPUTE DU
50 DO 70 I=1,27
51 IF(U(I).EQ.0..AND.R(I).GE.0.)DU(I)=0.
52 IF(U(I).EQ.0..AND.R(I).GE.0.)GOTO 70

```

```

53      DU(I)=-R(I)
54      70 CONTINUE
55      ISTU=0
56      DO 230 I=1,27
57      IF(ABS(DU(I)).LE.EPSI2)ISTU=ISTU+1
58      230 CONTINUE
C      MAXIMUM DU IS 1.
59      DMAX= ABS(DU(1))
60      DO 100 I=1,27
61      IF( ABS(DU(I)).LE.DMAX)GOTO 100
62      DMAX= ABS(DU(I))
63      100 CONTINUE
64      IF(DMAX.LE.1.)GOTO111
65      DO 110 I=1,27
66      DU(I)=      DU(I)/DMAX
67      110 CONTINUE
68      111 DUMMY=C
69      DO 80 I=1,27
70      IF(IPDEX(I).NE.0)GOTC 81
71      TH(I)=0.
72      DHN(I)=0.
73      GOTO 80
74      81 DHN(I)= U(I)*A(I,I) + 2.*C(I,1)*TH(I) + C(I,2) - TSUM(I)
75      80 CCNTINUE
C      STOPPING CRITERION ON THETA
76      162 ISTH=C
77      DO 160 I=1,27
78      IF(TH(I).LT.0.99.AND.TH(I).GT.0.45)GOTO 161
79      ISTH=ISTH+1
80      GOTO 160
81      161 DUMMY=0.
82      IF(ABS(DHN(I)).LE.EPSI1)ISTH=ISTH+1
83      160 CONTINUE
C      MAXIMUM DHN IS 1.
84      DHMAX=ABS (DHN(1))
85      DO 120 I=1,27
86      IF(ABS (DHN(I)).LE.DHMAX)GOTO 120
87      DHMAX=ABS (DHN(I))
88      120 CONTINUE
89      IF(DHMAX.LE.1.)GOTO138
90      DO 130 I=1,27
91      DHN(I)=DHN(I)/DHMAX
92      138 TTCOST=0.
93      DO 140 I=1,27
94      140 TTCOST=TTCOST + TRCOST(I)
95      TAX = X2(27) - TTCOST
C      OUTPUT CONTROL BY PARAMETERS: IFR,ISKIP,IJUMP,ITM
96      IF(ITER.EQ.0)IPND=0
97      INC=ITER/IFR
98      IF(ITER.LE.1)GOTC901
99      IF(ITER.LT.IJUMP)GCTC116
100     IF(IVLT.GT.C)GCTC116
101     IF(INC.EQ.IPND)GOTO 900
102     901 DUMMY=0.
103     PRINT 601,ITER
104     IF(ITER.LT.ISKIP)GOTC663
105     PRINT 602,(U(I),I=1,27)
106     PRINT 608
107     PRINT 604,( DU(I),I=1,27)
108     PRINT 608

```

```

109      PRINT 605,(DHN(I),I=1,27)
110      PRINT 608
111      PRINT 606,( TH(I),I=1,27)
112      PRINT 609,ISTU,DMAX,ISTH,DHMAX
113      PRINT 660, X2(27),TTCOST,TAX
114      663 DUMMY=0.
115      IF(IVLT.GT.0)PRINT 661,IVLT
116      IF(IVLT.EQ.0)PRINT 662
117      900 IPND=IND
118      116 DUMMY=0.
C      UPDATE THETA AND U VALUES
119      702 DO 90 I=1,27
120          IF(IPDEX(I).EQ.0)GOTO 91
121          TH(I)=TH(I)-KSTEP*DHN(I)
122          IF(TH(I).GE.0.99)TH(I)=0.99
123          IF(TH(I).LE.0.45)TH(I)=0.45
124          91 U(I)=U(I)+KSTEP*DU(I)
125          IF(U(I).LE.0.)U(I)=0.
126          50 CCNTINUE
127          IF(ISTH.EQ.27.AND.ISTU.EQ.27)GOTO 991
128          ITER=ITER+1
129          IF(ITER.LT.ITM)GOTO 102
130      991 PRINT 690
131          PRINT 601,ITER
132          PRINT 603,(R(I),I=1,27)
133          PRINT 608
134          PRINT 602,(U(I),I=1,27)
135          PRINT 608
136          PRINT 604,( DU(I),I=1,27)
137          PRINT 608
138          PRINT 605,(DHN(I),I=1,27)
139          PRINT 608
140          PRINT 606,( TH(I),I=1,27)
141          PRINT 609,ISTU,DMAX,ISTH,DHMAX
142          PRINT 660, X2(27),TTCOST,TAX
143          PRINT 670
144          STOP
145      501 FCRMAT (8F10.2)
146      503 FORMAT('-', 'REACH=', I4 //)
147      504 FORMAT(' ', 'AVALUES', 8F10.2)
148      505 FORMAT(' ', 'BVALUES', 8F10.2)
149      506 FORMAT(' ', 'CVALUES', 8F10.2)
150      601 FORMAT('-', 'ITERATION=', I4 //)
151      602 FCRMAT(' ', 'SHADCK', 10F12.4)
152      603 FORMAT(' ', 'DOSLAK', 10F12.4)
153      604 FORMAT(' ', 'SHAINC', 10F12.4)
154      605 FORMAT(' ', 'HGRADT', 10F12.4)
155      606 FORMAT(' ', 'TREATS', 10F12.4)
156      608 FCRMAT(' ')
157      609 FORMAT('-', 'SSC CN  U', I6, 5X, 'DU MAX', F10.4/'0',
*          'SSC CN  R', I6, 5X, 'DH MAX', F10.4/)
158      660 FORMAT('0', ' TOTAL COST', F12.2, ' TREATMENT COST', F12.2,
*          ' TAX REVENUE', F12.2)
159      661 FORMAT('0', 'VIOLATED CONSTRAINTS', I5)
160      662 FORMAT('0', 8X, 49('*'), ' FEASIBLE REGION ', 49('*'))
161      670 FORMAT('1')
162      690 FCRMAT('1', ' THE FINAL SCLUTION FOLLOWS')
163      END

```

\$ENTRY

OPTIMAL REGIONAL WATER QUALITY MANAGEMENT BY
MULTILEVEL APPROACH AND THE DISCRETE MAXIMUM PRINCIPLE

by

SUDHAKAR REDDY PAIDY

B.Tech.(Chemical Engineering), Osmania University,
Hyderabad, India, 1973.

AN ABSTRACT OF A MASTER'S THESIS

submitted in partial fulfillment of the

requirements for the degree

MASTER OF SCIENCE

Department of Industrial Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

1976

The problem of simultaneously finding an optimal waste water treatment configuration for meeting water quality standards along a river basin and determining the optimal pollution taxes to achieve this configuration is solved by decomposition and the multilevel approach. A two level hierarchy has been applied to solve the basin wide problem, by at-source treatment and by augmenting the river flow, using effluent charges. Two types of second level coordination schemes are presented for the decentralized decision making process. One assumes a knowledge of the local treatment cost functions by the regional authority (pseudo decentralized process); the other assumes no such knowledge (complete decentralized process). The Miami River in Ohio is considered as an example. A three level approach to introduce a regional treatment plant into the river basin also is illustrated.

The mathematical analogy between the two level approach and the discrete maximum principle is presented. The computational algorithms and the similarities are illustrated using the Miami river data for the pseudo decentralized process.