LINEAR SYSTEM CONTROL AND OPTIMIZATION, A RIGOROUS APPROACH BY MEANS OF THE TRICOTYLEDON THEORY OF SYSTEM DESIGN

by

Choi Fat Lee

A Thesis Submitted to the Faculty of the

DEPARTMENT OF SYSTEMS AND INDUSTRIAL ENGINEERING

In Partial Fulfillment of the Requirements
For the Degree of

MASTER OF SCIENCE
WITH A MAJOR IN SYSTEMS ENGINEERING

In the Graduate College

THE UNIVERSITY OF ARIZONA

1 9 7 4

STATEMENT BY AUTHOR

This thesis has been submitted in partial fulfillment of requirements for an advanced degree at The University of Arizona and is deposited in the University Library to be made available to borrowers under rules of the Library.

Brief quotations from this thesis are allowable without special permission, provided that accurate acknowledgment of source is made. Requests for permission for extended quotation from or reproduction of this manuscript in whole or in part may be granted by the head of the major department or the Dean of the Graduate College when in his judgment the proposed use of the material is in the interests of scholarship. In all other instances, however, permission must be obtained from the author.

SIGNED: Choi F. Lee

APPROVAL BY THESIS DIRECTOR

This thesis has been approved on the date shown below:

A. Wayne Wymore
A. WAYNE WYMORE
Professor of Systems and
Industrial Engineering

29 MAR 74
Date

ACKNOWLEDGMENTS

The author wishes to express his appreciation for being in the graduate program of the Systems and Industrial Engineering Department which broadens his viewpoint on various aspects of Systems Engineering, and which offers him a fascinatingly friendly and warm atmosphere.

The author would like to express his sincere and utmost gratitude to Dr. A. Wayne Wymore of the department, without whose guidance and help, this thesis could not have been accomplished. The author is grateful to Dr. D. G. Schultz of the Electrical Engineering Department for clearing the doubts in the control aspect of the thesis.

The author would also like to thank Mrs. Meta Anderson for her patience and professionalism in typing this thesis.

TABLE OF CONTENTS

TABLE OF CONTENTS--Continued

LIST OF ILLUSTRATIONS

LIST OF TABLES

ABSTRACT

This thesis rigorously defines a linear control problem in the framework of Tricotyledon Theory of System Design (TTSD), which leads the designer to an algorithm for solving the problem and which merges the classical and optimal control specifications on a mathematical basis.

Since TTSD is a new theory that aims at the designing of large scale systems which need interdisciplinary teams from all related fields, it is hoped that this thesis would serve as an introduction to TTSD to those from the control field participating in a large scale system design.

CHAPTER 1

INTRODUCTION

The field of systems control evolved just prior to the second world war; its infant age could be characterized by unity feedback closed loop control (9), the frequency response methods of Nyquist (12) and Bode (4) and the graphical root locus method of Evans (5). At that stage, the design work was done by trial and error. As the 1960's approached, the introduction of the concepts of controllability and observability by Kalman (6, 7), and the emphasis on matrix algebra brought on the adolescence of the control field. The need for precision and optimization in the space age motivated the early maturity of the control field. The expansion of the old mathematics, for example (1, 2), the second method of Liapunov, along with the introduction of new ideas in mathematics, for example, the maximum principle of Pontryagin (in 13) and the dynamic programming technique by Bellman (in 3), have triggered the modern approach to the control field. The optimal control design to maximize or minimize a certain performance index is gradually gaining popularity.

The early stages of the control field could be described as arithmetic or algebraic. The mathematical tools used were nothing more than Laplace or Fourier transforms and complex variables. The system under study was usually described by linear differential equations. The brute-force approach by the control engineers was despised by the mathematicians. Practical engineers and mathematicians could not be reconciled. Those between, namely, engineers with mathematical concepts or mathematicians who tend to be practical, strained to close the gap. The harvest of this endeavor was the birth of system theory.

The gap has been closed somewhat, yet another drawback has not been overcome. The applications of the various aspects of system theory are too limited to their individual purposes. When it comes to an operative, large-scale project that interrelates two or more fields, a lack of communication is usually felt. The need for communication among all professionals motivated the development of a widely applicable, rather universal system design methodology, namely, the Tricotyledon Theory of System Design (abbreviated TTSD) (15, 16). TTSD has been shown to be applicable to many fields (15, 16); control systems are not exempted.

This paper not only attempts to demonstrate that TTSD indeed encompasses control systems, but also the design methodology of TTSD can be used to achieve an optimal design of a control system so as to meet a given performance criterion. A classical problem in control theory is attacked within the framework of the system design methodology based on the TTSD. The object is to demonstrate the specializations necessary in the system design methodology in order that the optimal solution of the classical problem is also the optimal solution in the framework of system design methodology. The development of this paper is done at such a level that knowledge of TTSD and optimal control design is presumed. References provide the necessary background if needed. The remainder of this chapter presents the problem in the form of a state variable model. The example is quoted directly from Schultz and Melsa (14). The performance index is obtained by means of the model response (14). The given plant is to be controlled to meet a certain input-output criterion. It is assumed that the ordinary differential equation describing the dynamics of the plant is given. In Chapter 2, the plant is modeled by TTSD, the desired input-output characteristic is described. The plant is tested to see if it satisfies the input-output specifications. Chapter 3 specifies the technology from which the components of the system to be built can be obtained. The remainder of the paper demonstrates the design methodology of TTSD to obtain an optimal design with respect to the performance index.

The example presented tends to be straightforward since the objective is to demonstrate the methodology in a simple yet rewarding manner.

<u>The Given Plant in the Form of a Differential Equation and a State Variable Model</u>

$$\frac{d^3 y}{d \tau^3}(t) + 7*\frac{d^2 y}{d \tau^2}(t) + 10*\frac{d y}{d \tau}(t) = 10*u(t) + 10*\frac{d u}{d \tau}(t).$$

The above differential equation describes the plant with all initial conditions zero, where y is the output function of the plant and u is the input function. Taking Laplace transforms, the equation is:

$$s^3*Y(s) + 7*s^2*Y(s) + 10*s*Y(s) = 10*U(s) + 10*s*U(s).$$

In the frequency domain, the transfer function of the plant is given by:

$$\frac{Y(s)}{U(s)} = \frac{10*(s+1)}{s*(s+2)*(s+5)}.$$

For a unitstep input function, U(s) = 1/s, then, in the frequency domain, the response is

$$Y(s) = \frac{10*(s+1)}{s^2*(s+2)*(s+5)},$$

$$= \frac{0.3}{s} + \frac{1}{s^2} - \frac{8.3}{s+2} + \frac{0.53}{s+5},$$

or, in the time domain, the response is:

$y(t) = 0.3 + t - 8.3*e^{-2*t} + 0.53*e^{-5*t}$, for all $t \geq 0$.

For a unitramp input function, $U(s) = 1/s^2$, then in the frequency domain, the response is:

$$Y(s) = \frac{10*(s+1)}{s^3*(s+2)*(s+5)},$$

$$= \frac{0.31}{s} + \frac{0.3}{s^2} + \frac{1}{s^3} + \frac{0.417}{s+2} - \frac{0.107}{s+5},$$

and the response in the time domain:

$$y(t) = 0.31 + 0.3*t + 0.5*t^2 + 0.417*e^{-2*t} - 0.107*e^{-5*t},$$

for all $t \geq 0$.

The state variable model of the plant can be obtained from Fig. 1 directly. Let:

$$X_2(s)*\frac{1}{s} = X_1(s),$$

$$X_3(s)*\frac{(s+1)}{(s+2)} = X_2(s),$$

$$U(s)*\frac{10}{(s+5)} = X_3(s).$$

Then, after some algebraic manipulations, the following set of equations is obtained:

$$s*X_1(s) = X_2(s),$$

$$s*X_2(s) = -2*X_2(s) \quad -4*X_3(s) + 10*U(s),$$

$$s*X_3(s) = \quad -5*X_3(s) + 10*U(s).$$

Taking inverse Laplace transforms, the following set of first order linear equations is obtained:

$$\dot{x}_1(t) = x_2(t),$$

$$\dot{x}_2(t) = -2*x_2(t), \quad -4*x_3(t) + 10*u(t),$$

$$\dot{x}_3(t) = \quad -5*x_3(t) + 10*u(t),$$

where $\dot{x}(t) = \frac{d\ x}{d\ \tau}(t)$. Hence in matrix notation,

$$\dot{x}(t) = A*x(t) + B*u(t),$$

$y(t) = C^T x(t)$, where * is matrix multiplication,

$$x(t)= \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix},$$

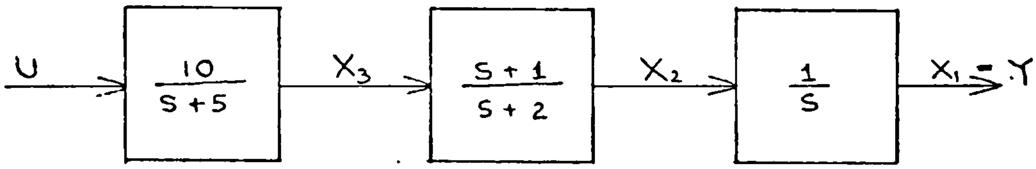

Fig. 1. Block Diagram Representation of the Plant

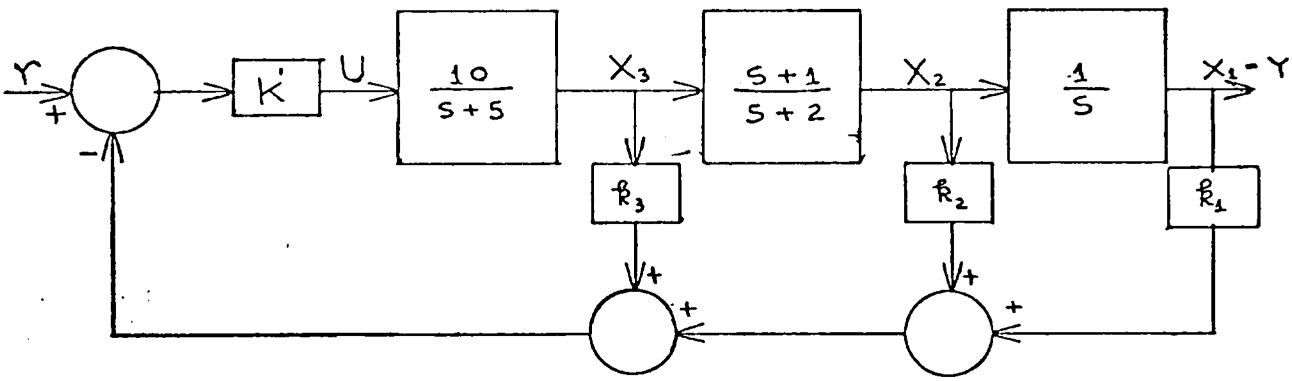

Fig. 2. Configuration of the Optimal System

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -2 & -4 \\ 0 & 0 & -5 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 10 \\ 10 \end{bmatrix}, \quad C = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix},$$

and

$$C^T = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}.$$

Figure 1 is a block diagram representation of the given plant; it is an open-loop system.

Figure 2 gives the closed-loop representation. The parameters K', k_1, k_2, and k_3 are to be sought such that the closed loop system would resemble a first order system described by the equation $3*y(t) + \dot{y}(t) = 0$, with a time constant of 1/3 second.

The desired model response can be translated into a performance index (14),

$$PI(x,u) = \int_0^\infty ((3*x_1(\tau) + x_2(\tau))^2 + u^2(x,\tau))*d\tau .$$

The optimal system with respect to the above performance index can be obtained by several optimization schemes; to name a few: the Hamilton-Jacobi procedure, the matrix Riccati equation, and the Kalman equation in the frequency domain. Schultz and Melsa (14) have presented a detailed synthesis of the optimal system by the Kalman equation with the aid of the root locus method. In the remainder of this chapter, the optimal system will be found by matrix Riccati equation, and in the next chapter, an analysis by TTSD to obtain the same optimal system will be described.

In general, the performance is described by:

$$\int_0^{t'} (x^T*Q*x + u^T*R*u)*dt,$$

where R and Q are symmetric matrices, R is positive definite and Q is positive semi-definite. Then the matrix Riccati equation is:

$$\dot{P}(t) + Q - P(t)BR^{-1}B^T P(t) + P(t)A + A^T P(t) = \overline{0},$$

where the multiplication operator is omitted, A, B, and C are the matrices from the state variable model. The optimal solution is $u_0(x,t) = - R^{-1}B^T P(t)x(t) = -K^T(t)x(t)$, where K is the matrix representing the feedback coefficients. Figure 2 is a single-input, single-output case where K is a three-dimensional vector. Note that the amplifying constant K' is assumed to be 1 in the matrix Riccati equation analysis. The only unknown for finding $u_0(x,t)$ is P(t) which is also a positive definite matrix. If t' is sufficiently large, a steady solution for P(t), $P_0 = \lim_{t\to\infty} P(t)$ is obtained. The steady state matrix Riccati equation is:

$$A^T P_0 + P_0 A - P_0 BR^{-1}B^T P_0 + Q = \overline{0}.$$

The steady state equation is what is useful for this particular problem, where A, B, and C are obtained previously,

$$Q = \begin{bmatrix} 0 & 3 & 0 \\ 3 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

as can be obtained from the equation,

$$x^T Q x = (3*x_1 + x_2)^2, \text{ also } R = R^{-1} = [1].$$

The computer analysis of this optimal control problem is illustrated in Appendix B. The steady state solution is:

$$P = \begin{bmatrix} 1.37 & 0.54 & -0.24 \\ 0.54 & 0.22 & -0.11 \\ -0.24 & -0.11 & -0.077 \end{bmatrix},$$

$$K^T = \begin{bmatrix} 3.0 & 1.1 & -0.32 \end{bmatrix}.$$

In the synthesis of this problem, there is an extra degree of freedom of choosing the parameters, K', k_1, k_2, and k_3. To achieve zero steady error, k_1 is chosen to be unity and hence K' is set to be 3 and $k^T = [1.0, 0.367, -0.11]$ to give the same response.

The control aspect of the problem, both classical and modern, has been presented in quite some detail. Those that are familiar with TTSD but have not been exposed to the field of control may find the introduction helpful in understanding the following chapters; on the other hand, those who are familiar with the field of control but not with TTSD may find this introduction boring, their main difficulties lie ahead. Appendix A hopefully will clear the clouds over their heads.

It is seen that the modern or optimal control solution obtained through a performance index does not meet the classical specification of zero steady error to a unistep input function and classical solution does not minimize the performance index. It will be shown how TTSD could define the problem precisely and rigorously as a guide to a solution to solve the dilemma. The task begins in the next chapter.

CHAPTER 2

DESIGN METHODOLOGY

The Plant Modeled in TTSD

In Chapter 1, the plant to be controlled was presented as a state variable model. Since the design procedure is intended to be presented in TTSD, it is necessary that the plant be describable in TTSD. This is done as follows:

Let PLANT = (S,P,F,T,σ) be a system, where:

$S = R^3$,

$P = R$,

$F = \{f : f \in \mathcal{F}(R^{++},P),\ \int_s^t f(t)dt \in R \text{ for every } s,t \in R^{++}\}$,

$T = R^{++}$, and for every $f \in F$, $t \in T$, $x \in S$,

$$\sigma(f,t)(x) = \exp(A*t)*x + \int_0^t \exp(A*(t-\tau))*B*f(\tau)*d\tau, \text{ where}$$

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -2 & -4 \\ 0 & 0 & -5 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 10 \\ 10 \end{bmatrix}, \quad x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix},$$

or (x_1,x_2,x_3), where appropriate, and $\exp(A*t) = \sum_k \{\frac{t^k}{k!}*A^k : k \in I^{++}\}$, R^{++} is the set of non-negative real numbers. Define the output of PLANT such that if $x \in S$, $\zeta(x) = \pi_1(x) = x_1$. It is not difficult to show that PLANT indeed satisfies the conditions for a system (Appendix A); this has been done in Chapter 3 of Wymore (15).

Expanding the above equation, then

$$\sigma(f,t)(x) = \exp(A*t)*x+\int_0^t \exp(A(t-\tau))*B*f(\tau)*d\tau,$$

$$= \begin{bmatrix} 1 & \frac{1}{2} - \frac{1}{2}*\exp(-2*t) & -\frac{4}{10} + \frac{2}{3}*\exp(-2*t) - \frac{4}{15}*\exp(-5*t) \\ 0 & \exp(-2*t) & -\frac{4}{3}*\exp(-2*t) + \frac{4}{3}*\exp(-5*t) \\ 0 & 0 & (-5*t) \end{bmatrix} *x$$

$$+ \int_0^t \begin{bmatrix} 1 + \frac{5}{3}*\exp(2*(\tau-t)) - \frac{8}{3}*\exp(5*(\tau-t)) \\ -\frac{10}{3}*\exp(2*(\tau-t)) + \frac{40}{3}*\exp(5*(\tau-t)) \\ 10*\exp(5*(\tau-t)) \end{bmatrix} *f(\tau)*d\tau.$$

Let $\underline{\text{unitstep}} = 1$, if $t \geq 0$;

$= 0$, if $t < 0$, and

$\underline{\text{unitramp}} = t$, if $t \geq 0$;

$= 0$, if $t < 0$,

let $x = \overline{0}$, the zero vector, then $\sigma(\underline{\text{unitstep}},t)(\overline{0})$

$$= \begin{bmatrix} t + 0.33 - 0.833*\exp(-2*t)+0.53*\exp(-5*t) \\ -1 + \frac{5}{3}*\exp(-2*t) - \frac{2}{3}*\exp(-5*t) \\ 2 - 2*\exp(-5*t) \end{bmatrix} \text{, also}$$

σ(<u>unitramp</u>,t)($\overline{0}$)

$$= \begin{bmatrix} 0.5*t^2+0.3*t-0.31+0.417*\exp(-2*t)-0.107*\exp(-5*t) \\ - t+\frac{7}{10} - \frac{5}{6}*\exp(-2*t)+ \frac{2}{15}*\exp(-5*t) \\ 2*t-\frac{2}{5} + \frac{2}{5}*\exp(-5*t) \end{bmatrix} .$$

The system to be controlled being defined in TTSD, the formal procedure of the design methodology can now be discussed. The system design methodology is defined (17) by a 6-tuple, P = ($\mathcal{S}$, $\mathcal{T}$, α, β, γ, $\mathcal{D}$), where

$\mathcal{S}$ is an input/output specification,

$\mathcal{T}$ is a technology,

$\alpha \in$ MERITORDERINGSIOCOTYLEDON ($\mathcal{S}$),

$\beta \in$ MERITORDERINGSTECHOTYLEDON ($\mathcal{T}$),

$\gamma \in$ MERITTRADEOFFORDERINGS ($\mathcal{S}$, $\mathcal{T}$, α, β),

$\mathcal{D} \in$ TESTPLANS ($\mathcal{S}$, $\mathcal{T}$, γ).

The definitions of the above notations are defined in Appendix A; redundancy is avoided here. Each of the six entries will be developed along with the discussion.

IOSPECIFICATION

The objective of the design is described by , the input/output specification. For the given model, let IOSPECPLANT = (P, F, Q, G, η) be the input/output specification for the desired system, where

P is the set of inputs of IOSPECPLANT, P = R,

F is the set of input functions of IOSPECPLANT consisting of the two functions: unitstep and unitramp,

Q is the set of outputs of IOSPECPLANT, the output set is the same as the input set, Q = R,

G = $\mathfrak{F}(R^{++}, R)$, is the set of output functions of IOSPECPLANT,

η is the matching function of IOSPECPLANT such that if the input function is unitstep, the output function does not differ from the input function by 0.05, and if the input function is the unitramp, the difference between the input value and the output value is less than 1.

The input/output specification defined here is quite different from those that are usually defined in the classical control field in which specifications such as time delay, time rise, peak response, etc. are usually given (11). Only the zero steady state error is specified by the matching function. In the design methodology of TTSD the other specifications are taken care of in the course of defining a performance index or a figure of merit or, equivalently, a merit ordering. It also may seem unreasonable that the input functions are only the unitstep and the unitramp, but for linear systems the response to the unitstep function determines the characteristics of the system. Besides, these two input functions well demonstrate the methodology and, at the same time, avoid complication.

Satisfaction of the IOSPECPLANT

Before the other entries of the design methodology are developed, it would be interesting to know whether PLANT satisfies the IOSPECPLANT (see Appendix A for the definition of satisfaction). Previously, motion and output function of PLANT were defined; the solutions for the two input functions are now found.

Let the initial state of PLANT be $x_0 = (0, 0, 0)$, then

$\zeta(\sigma(f, t)(x_0))$

$= \pi_1(\sigma(f, t)(x_0))$,

$= t + 0.3 - 0.833*\exp(-2*t) + 0.53*\exp(-5*t)$, if f = unitstep;

$= 0.5*t^2 + 0.3*t - 0.31 + 0.417*\exp(-2*t) - 0.107*\exp(-5*t)$, if f = unitramp,

where $t \geq 0$.

$|1-\zeta(\sigma(f, t)(x_0))|$

$< 0.05,$ if $f =$ unitstep and $t \in R(0.87-\delta, 0.87+\delta)$ for some small $\delta \in R^{++}$,

$|\zeta(\sigma(f,t)(x_0)) - t|$

$< 1,$ if $f =$ unitramp and $0 \leq t < 2.0$ sec.

Apparently, PLANT satisfies (x_o, ζ, T) IOSPECPLANT where

$x_0 = (0, 0, 0),$

$\zeta(x)$

$= \pi_1(x)$, and

$T = R(0.87-\delta, 0.87+\delta).$

Although PLANT does satisfy$((0, 0, 0,) \zeta, R(0.87-\delta, 0.87+\delta))$ IOSPECPLANT, it cannot be considered an optimal system. The obvious reason is that the time set in which it satisfies IOSPECPLANT is too small to be useful. That PLANT is less than optimal will become more obvious when the figure of merit or the merit ordering is defined.

Another System

At this point, it would be interesting to define another system that satisfies IOSPECPLANT. Let DUMYZ = (S, P, F, T, σ) be defined as:

$S = R,$

$P = R,$

F = ADMISSIBLESET({ f : there exists $r \in R$ such that

f = unitstep→r or f = unitramp→r}),

T = R^{++}, and for every $f \in F$, $x \in S$, $t \in T$,

σ(unitstep→r,t)(x)

= x,

σ(unitramp→r,t)(x)

= x+t. Then

σ(f,0)(x)

= x, if f = unitstep→r or f = unitramp→r.

To prove that DUMYZ is a system, suppose f = unitstep→r,

$s \in T$, then

σ(f→s,t)σ(f,s)(x)

= σ(f→s,t)(x),

= x,

= σ(f,s+t)(x). If f = unitramp→r, then

σ(f→s,t)σ(f,s)(x)

= x+s+t,

= σ(f,s+t)(x).

Suppose f, $g \in F$, $t \in T$, res(f,R[0,t))=res(g,R[0,t)), $f \neq g$, then $f \neq$ unitramp→r $\neq g$ for all r. Hence f = unitstep→r and g = unitstep→s and hence

σ(f,t)(x)

= x,

= σ(g,t)(x). Then Z satisfies the conditions of a system

(Appendix A). Then IOSPECPLANT is satisfied $(0.96, \omega, R^{++})$ by DUMYZ. Although the time range that DUMYZ satisfies IOSPECPLANT is optimal, DUMYZ is still not desirable. The reasons will be obvious in the next chapter.

CHAPTER 3

TECHNOLOGY

In this chapter, the second entry of the 6-tuple design methodology is developed in detail. $\mathcal{T}$ is a technology which contains the components of any systems that are realizable. The components must also be systems. Technology is usually constrained by the clients' requests, the hardware available and also the interests of the design engineers. In this development, it is assumed that the technology being considered contains systems having only a one-dimensional state, summers and attenuators. These are named and defined, respectively, as ISTATE, nSUMMER and KXATTENUATOR as follows:

ISTATE/a

In the linear system control field, a complicated system block diagram can be decomposed into elementary block diagram, containing only summers, attenuators and integrators or, equivalently, 1st order systems. It is in this spirit that the technology here is defined.

Figure 3 is a block diagram of a 1st order system, in state variable model; it is represented as $\dot{x} = -a*x + u$, $\frac{1}{a}$ is the time constant, $a \in R^{++}$.

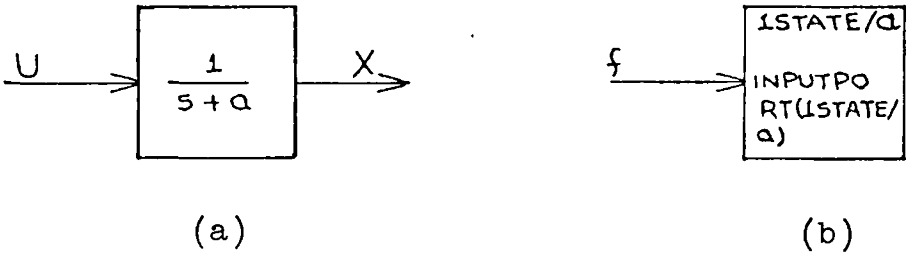

Fig. 3. First Order System, a) in Block Diagram, b) in TTSD Technology

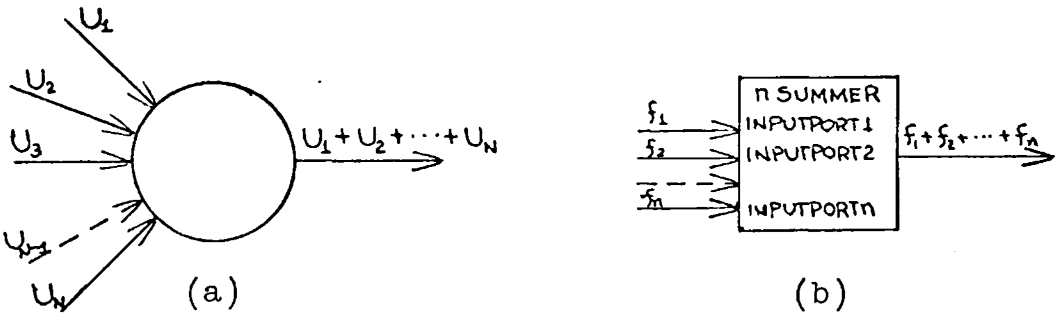

Fig. 4. Summer Representation, a) in Block Diagram, b) in TTSD Technology (nSummer)

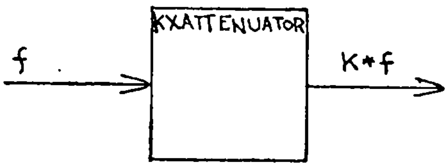

Fig. 5. Attenuator in TTSD Technology

or, $x(t) = \exp(-a*t)*x(0) + \int_0^t \exp(a*(\tau-t))*u(\tau)*d\tau$,

assuming the initial condition at time 0 is x(0). As it was done previously, this must be defined in TTSD for the development. Let ISTATE/a = (S,P,F,T,σ) be a system where:

$S = R^{++}$,

$P = R$,

$F = \{f: f\in \mathcal{F}(R^{++},P), \int_s^t f(\tau)d\tau \in R \text{ for every } s,t\in R^{++}\}$,

$T = R^{++}$, and for every $f\in F$, $t\in T$, $x\in S$,

$\sigma(f,t)(x)$

$= \exp(-a*t)*x + \int_0^t \exp(a*(\tau-t))*f(\tau)*d\tau$, where $0 \leq a < \infty$.

ISTATE/a thus defined is very similar to PLANT; that it is a system is again not difficult to prove.

nSUMMER

The second item on the list of the technology is the summer. Figure 4 is a block diagram representation. Let nSUMMER = (S,P,F,T,σ) be defined such that nSUMMER has n input ports, namely, INPUTPORT1, INPUTPORT2 . . . and INPUTPORTn, each of the n input ports accepts inputs from the set of real number, R,

$S = R^n$,

$P = R^n$,

$F = \{f: f\in \mathcal{F}(R^{++},P), f(t^-)\in P \text{ for every } t\in R\}$,

$T = R^{++}$, and for every $f\in F$, $t\in T$, $x\in S$,

$\sigma(f,t)(x)$

$= x,$ if $t = 0$;

$= f(t^-)$, if $t > 0$, where $f(t^-)$ is the conventional notation for the value of the limit of the function from the left.

Define the output function <u>naddition</u> of nSUMMER: for every $x \in S$, <u>naddition</u>(x)

$= \sum_i \{x_i : i \in I[1,n]\}$, if $x = (x_1, x_2, \ldots, x_n) \in R^n$.

That nSUMMER is a system has been proved by Wymore (15).

KXATTENUATOR

The last item that is listed in the technology is the attenuator. Figure 5 is the block diagram for the attenuator. Let KXATTENUATOR = (S,P,F,T,σ) be a system, where

$S = R^{++}$,

$P = R$,

$F = \{f: f \in \mathcal{F}(R^{++},P),\ f(t^-) \in P \text{ for every } t \in R\}$,

$T = R^{++}$, and for every $f \in F$, $t \in T$, $x \in S$,

$\sigma(f,t)(x)$

$= x,$ if $t = 0$;

$= f(t^-)$, if $t > 0$.

Define the output function <u>Kmultiply</u> of KXATTENUATOR: for every $x \in S$,

<u>Kmultiply</u>(x)

$= K*x$, where $K \in R$.

Now, formally, the definition of the technology for this development is completed. Let PLANTTECHNOLOGY be the technology containing any system isomorphic to any system in the following set of systems:

$$\{1STATE/a:\ 0 \leq a < \infty\} \bigcup \{nSUMMER:\ n \in I^{++},\ n < \infty\} \bigcup \{KXATTENUATOR:\ K \in R^{++},\ K < \infty\}.$$

<u>Buildability and Implementability</u>

After the technology is defined, buildability of a system in a technology can now be discussed. A system Z is said to be buildable in the technology $\mathcal{T}$ if and only if there exists a coupling recipe K such that COMPONENTS-(K) $\subset \mathcal{T}$ and Z = RESULTANT(K). A system Z is said to be implementable in the technology $\mathcal{T}$ if and only if there exists a system Z' buildable in $\mathcal{T}$ and a subsystem, Z" of Z' such that Z is a homomorphic image of Z". Definitions of coupling recipes, components, and resultants are given in Appendix A. To illustrate the definition of buildability, it is interesting to see if PLANT is implementable in PLANTTECHNOLOGY. The answer should be very obvious.

Figure 6 is an equivalent block diagram representation of the plant.

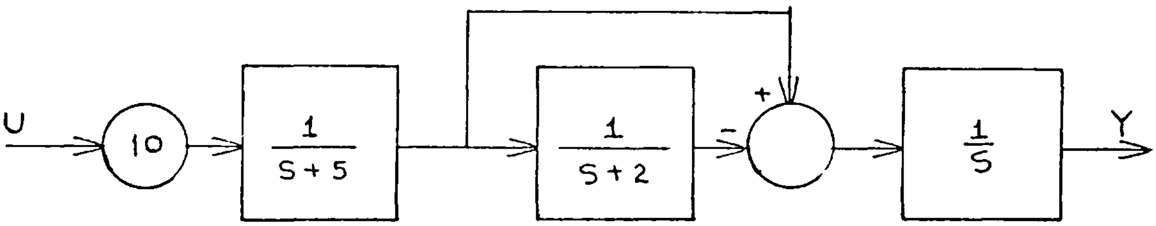

Fig. 6. Equivalent Block Diagram Representation of the Plant

Figure 7 shows the coupling recipe and resultant of PLANT. Let PLANTCOUPLE = $(\mathfrak{z},\alpha,\beta)$ be the coupling recipe of PLANT such that

$\mathfrak{z}$ = {10XATTENUATOR, 1STATE/5, 1STATE/2(1), 1STATE/2(2), -1XATTENUATOR, 3SUMMER, 1STATE/0}.

α is defined on $\mathfrak{z}^2$ as such that:

α(Z',Z")

= PORTISTATE/5,	if Z'=1STATE/5, Z"=10XATTENUATOR;
= 1PORT3SUMMER,	if Z'=3SUMMER, Z"=1STATE/5;
= PORT-1XAT,	if Z'=-1XATTENUATOR, Z"=1STATE/2(1);
= 2PORT3SUMMER,	if Z'=3SUMMER, Z"=-1XATTENUATOR;
= 3PORT3SUMMER,	if Z'=3SUMMER, Z"=1STATE/2(2);
= PORT2XAT(1),	if Z'=1STATE/2(1), Z"=1STATE/5;
= PORTISTATE/0,	if Z'=1STATE/0, Z"=3SUMMER;
= <u>none</u>,	otherwise ;

β(Z',Z")

= <u>10multiply</u>,	if Z'=10XATTENUATOR, Z"=1STATE/5;
= ω,	if Z'=1STATE/5, Z"=3SUMMER, or
	if Z'=1STATE/5, Z"=1STATE/2(1), or

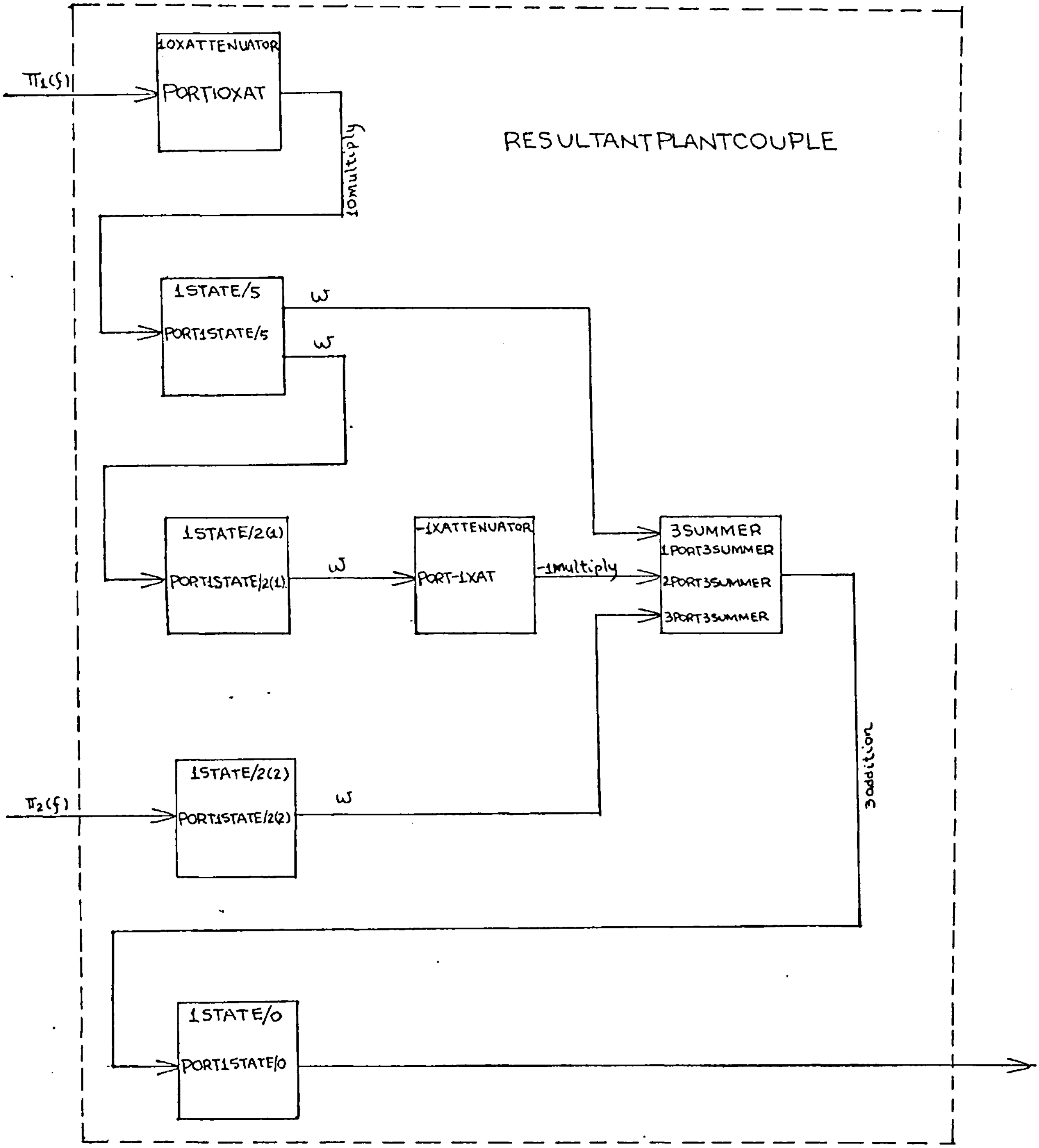

Fig. 7. Coupling Recipe and Resultant of PLANT

if Z'=1STATE/2(1), Z"=-1XATTENUATOR, or

if Z'=1STATE/2(2), Z"=3SUMMER;

= -1multiply, if Z'=-1XATTENUATOR, Z"=3SUMMER;

= 3addition, if Z'=3SUMMER, Z"=1STATE/0;

= φ, otherwise, where

ω is the identity function, φ is the null set and none is the alternate form of φ.

Having defined PLANTCOUPLE, it must be shown that the resultant of PLANTCOUPLE simulates PLANT. This implies that PLANT must be shown to be a homomorphic image of a subsystem of the resultant of PLANTCOUPLE. The formal definition of the resultant of a coupling recipe involves a coupling function (17). But since the configuration of PLANTCOUPLE involves only cascade coupling, the resultant can be obtained without the aid of the coupling function. Let RESULTANTPLANTCOUPLE = (S,P,F,T,σ) be defined as:

S = STATES(10XATTENUATOR) X STATES(1STATE/5) X STATES-(1STATE/2(1)) X STATES(1STATE/2(2)) X STATES(-1XATTEN-UATOR) X STATES(3SUMMER) X STATES(1STATE/0),

P = INPUTS(10XATTENUATOR) X INPUTS(1STATE/2(2)),

F = {f: f$\in \mathcal{F}(R^{++},P)$, such that f = projection(PORT10XAT, PORT1STATE/2(2))$\circ$g, for some g$\in$TOTALINPUTFUNCTIONS(PLANTCOUPLE)},

T = R^{++}, and for every f$\in$F, t$\in$T, $(x_1,x_2,x_3,x_4,x_5,x_6,x_7)\in S$,

$\pi_i(\sigma(f,t)(x_1,x_2,x_3,x_4,x_5,x_6,x_7))$

= $\underline{\text{motion}}$(1OXATTENUATOR)($\pi$(PORT1OXAT)$\cdot$f,t)($x_1$),

= a(t), if i=1;

= $\underline{\text{motion}}$(1STATE/5)(a,t)($x_2$),

= b(t), if i=2;

= $\underline{\text{motion}}$(1STATE/2(1))(b,t)($x_3$),

= c(t), if i=3;

= $\underline{\text{motion}}$(1STATE/2(2))($\pi$(PORT1STATE/2(2))$\cdot$f,t($x_4$),

= d(t), if i=4;

= $\underline{\text{motion}}$(-1XATTENUATOR)(c,t)($x_5$),

= e(t), if i=5;

= $\underline{\text{motion}}$(3SUMMER)(b+e+d ,t)($x_6$),

= g(t), if i=6;

= $\underline{\text{motion}}$(1STATE/0)(g,t)($x_7$),

= h(t), if i=7.

The definition of motion of RESULTANTPLANTCOUPLE seems very lengthy; nevertheless, closed form expressions can be obtained by solving the equations. This will become clear when homomorphism is discussed. To show that PLANT is implementable in PLANTTECHNOLOGY, the lemmas in Appendix C are required.

Corresponding to the lemmas, let U$\subset$STATES(SUBPLANT-COUPLE), such that

U = $\{(0,x_3,0,x_2-x_3,0,(x_3,0,x_2-x_3),x_1)$:
$x_1 \in$STATES(1STATE/0), $x_2 \in$STATES(1STATE/2(1)),
$x_3 \in$STATES(1STATE/5)$\}$; and let

SUBPLANTCOUPLE = (S',P',F',T',σ') be defined as:

S' = {x: $\in$x STATES(RESULTANTPLANTCOUPLE), there exists some y$\in$U, t$\in$T, g$\in$F, such that x=σ(g,t)(y)},

P' = INPUTS(RESULTANTPLANTCOUPLE),

F' = {f: f$\in$INPUTFUNCTIONS(RESULTANTPLANTCOUPLE), and zerofunction = (PORT1STATE/2(2))$\cdot$f, where zerofunction(t) = 0, for all t$\in$T},

T' = TIMESCALE(RESULTANTPLANTCOUPLE), and for every f$\in$F, t$\in$T, x$\in$S,

σ'(f,t)(x) = res(σ(f,t),S).

By lemma 1, SUBPLANTCOUPLE is indeed a subsystem of RESULTANTPLANTCOUPLE. The last step is to show that PLANT is an homomorphic(ρ,μ,θ) image of SUBPLANTCOUPLE. Let ρ=ω, if f$\in$INPUTFUNCTIONS(SUBPLANTCOUPLE), μ(f) = π(PORT10XAT)$\circ$f, and if x$\in$STATES(SUBPLANTCOUPLE),

θ(x)

= $(\pi_7(x), \pi_1(\pi_6(x))+\pi_2(\pi_6(x))+\pi_3(\pi_6(x)), \pi_2(x))$.

It must be shown that if g$\in$INPUTFUNCTIONS-(SUBPLANTCOUPLE), t$\in$TIMESCALE(SUBPLANTCOUPLE), and

x = $(0,x_3,0,x_2-x_3,0,(x_3,0,x_2-x_3),x_1)$$\in$STATES(SUBPLANTCOUPLE),

θ(motion(SUBPLANTCOUPLE)(g,t)(x))

= motion(PLANT)(μ(g),ρ(t))(θ(x)),

= motion(PLANT)(f,t)(x_1,x_2,x_3), according to lemma 2,

where μ(g) = f$\in$INPUTFUNCTIONS(PLANT). The following computations will show the identity:

Let σ denote motion(SUBPLANTCOUPLE), then

$\pi_1(\sigma(g,t)(x)$

$= f(t^-);$

$\pi_2(\sigma(g,t)(x))$

$= \exp(-5*t)+\int_0^t \exp(-(5*(t-\tau))*10*f(\tau)*d\tau$

$= \pi_3(\theta(\sigma(g,t)(x);$

$\pi_3(\sigma(g,t)(x))$

$= \int_0^t \exp(-2*(t-\tau))*(\exp(-5*\tau)*x_3+\int_0^t \exp(-5*(\tau-\delta))*10*$
$f(\delta)*d\delta)*d\tau,$

$= \int_0^t \exp(-2*t-3*\tau)*x_3*d\tau+\int_0^t \exp(-2*(t-\tau))*\int_0^\tau \exp(-5*(\tau-\delta))*$
$10*f(\delta)*d\delta*d\tau,$

$= \exp(-2*t)*x_3 \int_0^t \exp(-3*\tau)*d\tau+\exp(-2*t)*\int_0^t \exp(2*\tau)*$
$\int_0^\tau \exp(-5*(\tau-\delta))*10*f(\delta)*d\delta*d\tau,$

$= \exp(-2*t)*x_3*(\frac{1}{3} - \frac{1}{3}(\exp(-3*t)))+\exp(-2*t)*\int_0^t \exp(-3*\tau)*$
$\int_0^\tau \exp(5\delta)*10*f(\delta)*d\delta*d\tau,$

$= \frac{1}{3}*(\exp(-2*t)-\exp(-5*t))*x_3+\exp(-2*t)*([- \frac{1}{3}*\exp(-3\tau)*$
$\int_0^\tau \exp(5\delta)*10*f(\delta)*d\delta]_0^t - \int_0^t - \frac{1}{3}*(\exp(-3*\tau))* \exp(5*\tau)*$
$10*f(\tau)*d\tau),$ [integrating by parts],

$= \frac{1}{3}*(\exp(-2*t)-\exp(-5*t))*x_3+\exp(-2*t)*(- \frac{1}{3}*\exp(-3*t)*$
$\int_0^t \exp(5*\tau)*10*f(\tau)*d\tau + \frac{1}{3}*\int_0^t \exp(2*\tau)*10*f(\tau)*d\tau,$

$$= \frac{1}{3}*(\exp(-2*t)-\exp(-5*t)) - \frac{1}{3}\int_0^t \exp(-5(t-\tau))*10*f(\tau)*d\tau$$

$$+ \frac{1}{3}*\int_0^t \exp(-2*(t-\tau))*10*f(\tau)*d\tau;$$

$\pi_4(\sigma(g,t)(x))$

$$= \exp(-2*t)*(x_2-x_3);$$

$\pi_5(\sigma(g,t)(x))$

$$= \pi_3(\sigma(g,t^-)(x);$$

$\pi_1(\pi_6(\sigma(g,t)(x))$

$$= \pi_2(\sigma(g,t)(x));$$

$\pi_2(\pi_6(\sigma(g,t)(x))$

$$= -\ \pi_3(\sigma(g,t^-)(x));$$

$\pi_3(\pi_6(\sigma(g,t)(x))$

$$= \pi_4(\sigma(g,t^-)(x));$$

then $\pi_2(\theta(\sigma(g,t)(x))$

$$= \exp(-5*t)*x_3+\int_0^t \exp(-5*(t-\tau))*10*f(\tau)*d\tau\ -$$

$$\frac{1}{3}*(\exp(-2*t)-\exp(-5*t)) + \frac{1}{3}\int_0^t \exp(-5*(t-\tau))*10*f(\tau)*d\tau\ -$$

$$\frac{1}{3}\int_0^t \exp(-2*(t-\tau))*10*f(\tau)*d\tau+\exp(-2*t)*(x_2-x_3),$$

$$= \exp(-2*t)*x_2 + \frac{4}{3}*(\exp(-5*t)-\exp(-2*t))*x_3\ +$$

$$\frac{4}{3}\int_0^t \exp(-5*(t-\tau))*10*f(\tau)*d\tau - \frac{1}{3}*\int_0^t \exp(-2*(t-\tau))*10*$$

$$f(\tau)*d\tau;$$

$\pi_7(\sigma(g,t)(x))$

$= x_1 + \int_0^t (\exp(-2*\tau)*x_2 + \frac{4}{3}*(\exp(-5*\tau)-\exp(-2*\tau)*x_3 +$

$\int_0^\tau \exp(-5(\tau-\delta))*10*f(\delta)*d\delta - \frac{1}{3}*\int_0^\tau \exp(-2*(\tau-\delta))*10*$

$f(\delta)*d\delta)*d\tau,$

$= x_1 + \frac{1}{2}*(1-\exp(-2*t))*x_2+(- \frac{4}{10} + \frac{2}{3}*\exp(-2*t)- \frac{4}{15}*$

$\exp(-5*t))*x_3 + \frac{4}{3}*(- \frac{1}{5}*[\exp(-5*\tau)*\int_0^\tau \exp(5*\delta)*10*f(\delta)*d\delta]_0^t$

$+ \frac{1}{5}*\int_0^t \exp(-5*\tau) *\exp(5*\tau)*10*f(t)*d\tau) - \frac{1}{3}*$

$([- \frac{1}{2}*\exp(-2\tau)*\int_0^\tau \exp(2*\delta)*10*f(\delta)*d\delta]_0^t +$

$\int_0^t \frac{1}{2}*\exp(-2*\tau)*\exp(3*\tau)*10*f(\tau)d\tau),$

$= x_1 + \frac{1}{2}*(1-\exp(-2*t))*x_2 + (- \frac{4}{10} + \frac{2}{3}*\exp(-2*t) -$

$\frac{4}{15}*\exp(-5*t))*x_3- \frac{4}{15} \int_0^t \exp(-5*(t-\tau))*10*f(\tau)*d\tau+$

$\frac{4}{15} \int_0^6 10*f(\tau)*d\tau + \frac{1}{6} \int_0^t \exp(-2*(t-\tau))*10*f(\tau)*d\tau -$

$\frac{1}{6}*\int_0^t 10*f(\tau)*d\tau,$

$= x_1 + \frac{1}{2}(1-\exp(-2*t))*x_2+(- \frac{4}{10} + \frac{2}{3}*\exp(-2*t) - \frac{4}{15}*$

$\exp(-5*t))*x_3 + \int_0^t f(\tau)*d\tau + \frac{5}{3}*\int_0^t \exp(-2*(t-\tau))*f(\tau)*d\tau$

$$- \frac{8}{3} * \int_{0}^{t} \exp(-5*(t-\tau))*f(\tau)*d\tau,$$

$= \pi_3(\theta(\sigma(g,t)(x)))$. Hence

$\theta(\sigma(g,t)(x))$

= motion(PLANT)$(\mu(g),t)(\theta(x))$

= motion(PLANT)$(f,t)(x_1,x_2,x_3)$, then by Lemma 2 of Appendix C, PLANT is an homomorphic(ρ,μ,θ) image of SUBPLANTCOUPLE, and then PLANT is implementable in PLANT-TECHNOLOGY.

Recall, at the end of the last chapter, DUMYZ was a system that satisfies IOSPECPLANT, but it was not desirable. The reason is that DUMZ is not implementable in PLANTTECHNOLOGY thus defined. It is completely ruled out in the sense of implementability.

Thus far, the I/O specification, the technology of the plant, the concepts of satisfaction of I/O specification buildability and implementability have been introduced, the groundwork of the system design methodology has been laid. The job ahead is to compare the systems that are picked and to consider the cost-benefit criteria. These will be discussed in the following chapters.

CHAPTER 4

IOCOTYLEDON AND TECHOTYLEDON

Out of the six - tuple of the design methodology, two have been defined. These two are the problem setting that involves both the client and the systems engineers. The rest of the design methodology tends to be technical in the sense that decisions must be made by experts participating on an interdisciplinary team, in general. This latter part of the design methodology deals with orderings of systems that satisfy the I/O specifications, that are implementable in the defined technology, and that are within the intersection of the before-mentioned domains. The development of an algorithm for choosing a good system needs professional participation; it may be asking too much from the systems engineers alone who may not be as expert as those personnel who deal with the very field for their whole professional lives. This is the reason that an interdisciplinary team with personnel from all related fields is essential.

The development here is done by a team of only one person with the help of the advisor. The criteria for orderings or comparison are chosen only for illustrative purposes; they might not necessarily be the best. The system that is to be picked will be optimal only with respect to the criteria chosen. The philosophy of choosing the best criteria is out of the scope of this development.

IOCOTYLEDON

In Chapter 2, IOSPECPLANT was defined, PLANT and DUMYZ were found to satisfy IOSPECPLANT in a small and an optimal time domain, respectively. It was mentioned that PLANT was by no means an optimal system. Other systems must be considered. But all other systems to be considered must be chosen from a cotyledon of systems that satisfy IOSPECPLANT with respect to an initial condition, an output function and a time set. This set of systems is called the IOCOTYLEDON in TTSD. Formally, define:

IOCOTYLEDONPLANT

= IOCOTYLEDON(IOSPECPLANT);

= $\{(Z, x, \zeta, T)$: Z is a system and IOSPECPLANT is satisfied-(x, ζ, T) by Z$\}$.

The approach here is to choose a system and then test to see if it satisfies the IOSPECPLANT, and, if it does, the range of the time set is evaluated. Hopefully, the time set is the non-negative reals for the system that is finalized. The detail of picking a system will be presented after the TECHCHOTYLEDON and the orderings over the IOCO-TYLEDON and TECHOTYLEDON are defined.

TECHOTYLEDON

At this moment, definitions are given without explanations. But illustrative examples will be shown to clarify the definitions. Again, define:

TECHOTYLEDONPLANT

= TECHOTYLEDON(PLANT),

= $\{(K,Z)$: K is a coupling recipe, COMPONENTS(K) $\subset$ PLANT-TECHNOLOGY, Z = RESULTANT(K)$\}$.

MERITORDERINGSIOCOTYLEDON(ISOPECPLANT)

The orderings over IOCOTYLEDON and over TECHO-TYLEDON are first defined. Then any system designed could be used to compare with PLANT; the better one could be kept as the reference system and the less desirable one abandoned. An iterative search method could be used here based on the same principle as in some optimization schemes.

meritorderingsIOSPECPLANT$\in$ $\mathcal{F}$(IOCOTYLEDONPLANT^2, $\{0,1\}$)

such that if (Z,x,ζ,T), $(Z',x',\zeta',T')\in$IOCOTYLEDONPLANT,

meritorderingsIOSPECPLANT$((Z',x',\zeta',T'),(Z,x,\zeta,T))$

= 1, if $T'\subseteq R(1.5,\infty)\subseteq T$ or if $R(1.5,\infty)\subseteq T'$ and $R(1.5,\infty)\subseteq T$; and

if both $\zeta(\underline{x}) = 0.63$ and $\zeta'(\underline{x}') = 0.63$ occur at $t, t'\in R(0.25,0.5)$ where ζ and ζ' are the output functions of Z and Z', and

$\underline{x}$ = motion(Z)(unitstep,t)(x),

$\underline{x}$'= motion(Z')(unitstep,t')(x');

= 0, otherwise.

The conditions of orderings are related to classical control specifications as follows:

$T' \subseteq R(1.5,\infty) \subseteq T$ implies that a better system must have a long range of time for satisfying IOSPECPLANT (zero steady state error), and the rise time must be smaller than 1.5 seconds; the other condition indicates that the time constant must be in the range of R(0.25,0.5). Recall the ideal system is to follow a first-order system with time constant of 1/3 second. Since normally it is impossible to have an ideal system, some tolerance must be allowed.

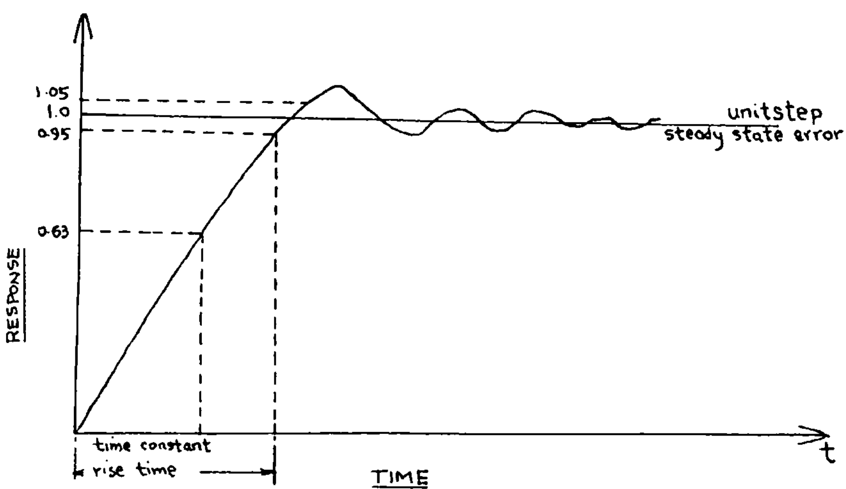

Fig. 8. Classical Control Specifications

MERITORDERINGSTECHOTYLEDON(PLANTTECHNOLOGY)

The definition here would involve experts from the field of optimal control and the client. When the client specifies the overall system, the experts could translate the specifications into a figure of merit that would describe the ideal system. Experiments can be performed on systems actually built; a performance index can be computed (17). Let Z be a system that is the resultant of a coupling recipe in TECHOTYLEDONPLANT, $x \in$ STATES(Z), $t \in$ TIMESCALE(Z), σ=motion(Z), $f \in$ INPUTFUNCTIONS(Z), then define:

$PI_Z(f,x,t)$

$$= \int_0^t ((3*\pi(\mathrm{STATES}(Z_{11}))(\sigma(f,\tau)(x)) + \pi(\mathrm{STATES}(Z_6))(\sigma(f,\tau)(x)))^2 + (K*\pi(\mathrm{STATES}(Z_3))(\sigma(f,\tau)(x)))^2)*d\tau,$$

where Z_i, i=1,2,...,13 are the system components as shown in Fig. 9.

The motivation for the performance index defined on systems of the configuration of Fig. 9 is that from the merit-ordering over TECHOTYLEDONPLANT which will be defined later, only systems of this configuration will be considered good. From the performance index, a figure of merit is defined (17).

Let

figomPLANT(OPTIMALCOUPLEi)

$= PI_Z(\mathrm{zerofunction}, x, \infty)$, where

Z=RESULTANT(OPTIMALCOUPLEi),

$x \in$ STATES(Z), $\pi_{4,6,11}(x)$ = some fixed positive values,

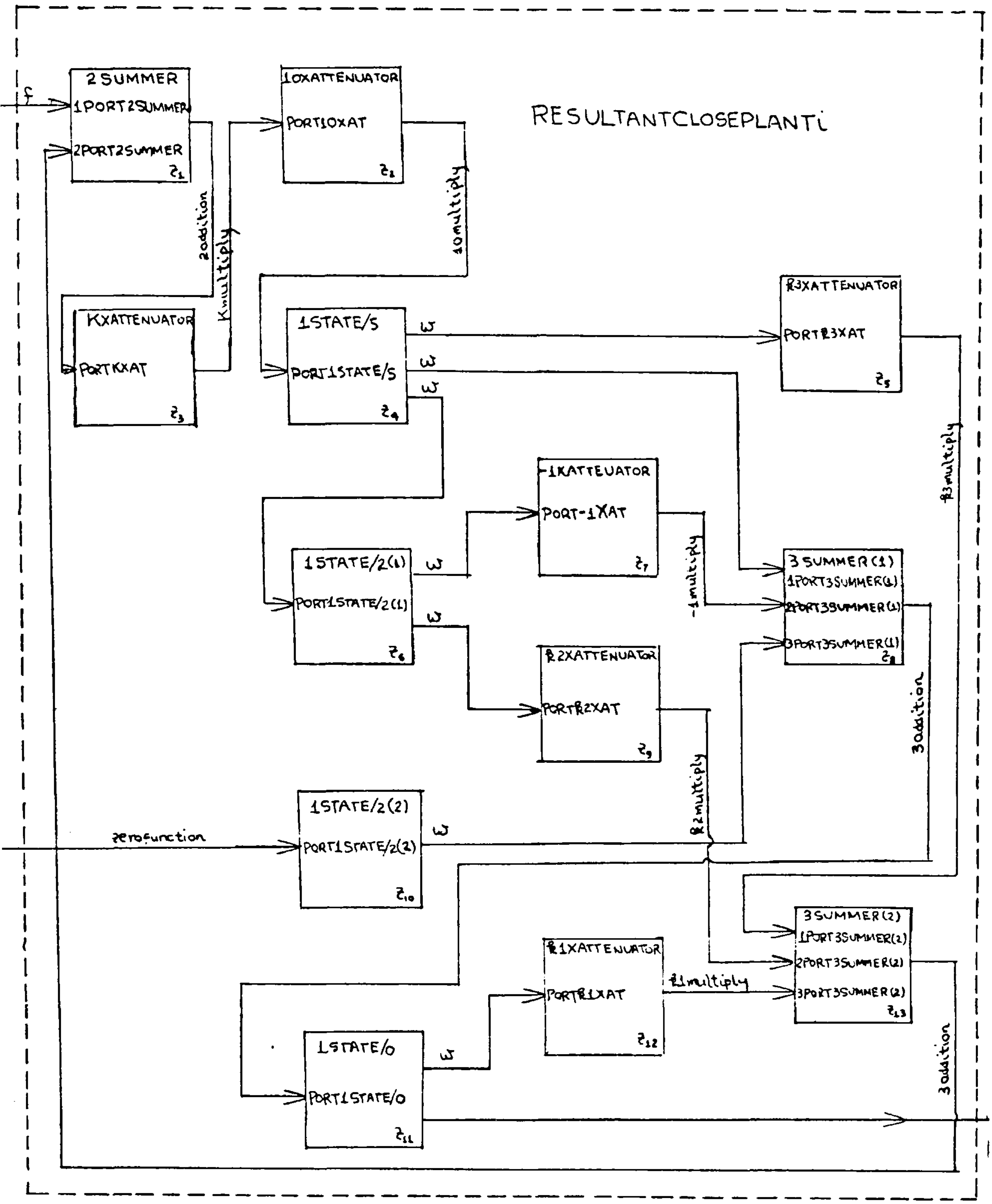

Fig. 9. Optimal Closed Loop Configuration

with the guide of figomPLANT, experts in the field of optimal control realize that in order to have figomPLANT minimum, the system must be in the configuration of Fig. 9, that is, all states of Z_4, Z_6, and Z_{11} must be fed back (8,14). Note that the figure of merit thus defined is analogous to the quadratic performance index defined in the control field, hence Matrix Riccati equation is applicable here. Hence define:

meritorderingsTECHOTYLEDON $\in$ (TECHOTYLEDONPLANT^2, {0,1}),

such that if (K,Z), (K',Z') $\in$ TECHOTYLEDONPLANT,

meritorderingsTECHOTYLEDON((K,Z),(K',Z'))

= 1, if K' is in the configuration of OPTIMALCOUPLEi as shown in Fig. 9, but K is not; or if both K and K' are in OPTIMALCOUPLEi form, but figomPLANT(K') $\leq$ figomPLANT(K),

= 0, otherwise.

The meritordering over TECHOTYLEDONPLANT thus defined has ruled out the classical approach of adding a combination of 1STATE/a's, or equivalently, compensators and unity feedback to control the plant as shown in Fig. 10, since OPTIMALCOUPLEi does not allow addition of 1STATE/a.

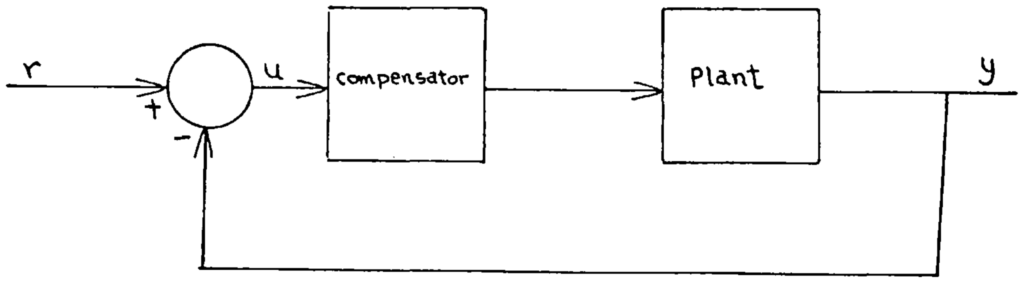

Fig. 10. Classical Control with Compensation and Unity Feedback

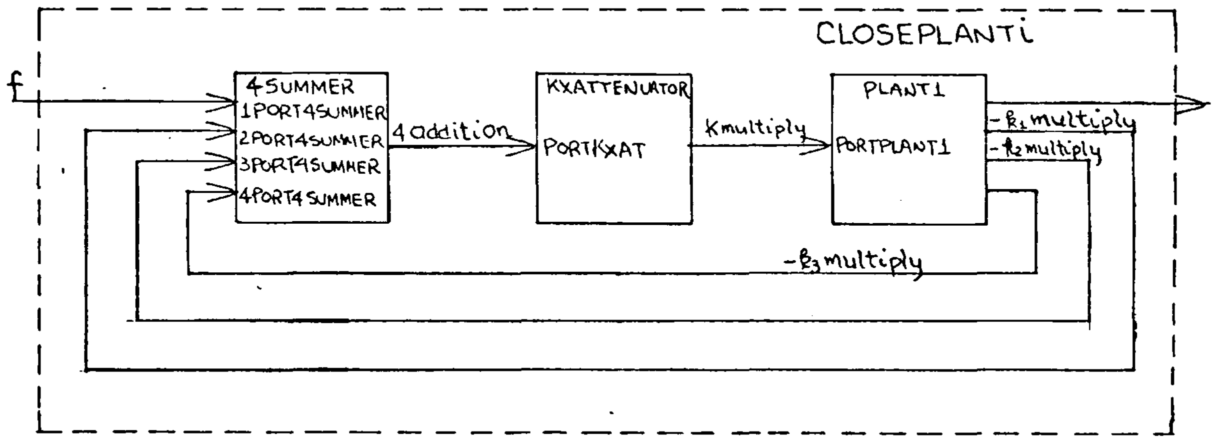

Fig. 11. Configuration for Search of Optimal System

CHAPTER 5

A PROPOSED DESIGN

Thus far, the criteria of comparisons of systems in the IOCOTYLEDON and the TECHOTYLEDON have been defined; there is still another criterion to be considered, namely, the trade-off criterion in the FEASIBILITYCOTYLEDON. Systems which are in the intersection of the IOCOTYLEDON and the TECHOTYLEDON are compared by means of the trade-off criterion. But before doing so, it would be clearer to have a proposal of a design and have it compared in the IOCOTYLEDON and the TECHOTYLEDON.

Let the parameters K_i, k_{i1}, k_{i2} and k_{i3} in OPTIMALCOUPLEi be 3, -1, -0.367 and 0.107, respectively. Denote the coupling recipe by OPTIMALCOUPLE1, denote the resultant of OPTIMALCOUPLE1 by RESULTANTCLOSEPLANT1 and denote its subsystem by SUBCLOSEPLANT1. Let CLOSEPLANT1=(S,P,F,T,σ) denote the system that is simulated(ρ,μ,θ) by SUBCLOSEPLANT1 for some ρ,μ,θ which can be determined in a similar manner as was done in Chapter 3. Hence for every $f \epsilon F$, $x \epsilon S$, $t \epsilon T$,

<u>motion</u>(CLOSEPLANT1)(f,t)(x)

$$= \exp(A''*t)*x + \int_0^t \exp(A''*(t-\tau))*B''*f(\tau)*d\tau, \text{ where}$$

$$A'' = \begin{bmatrix} 0 & 1 & 0 \\ -30 & -13 & -7.2 \\ 30 & -11 & -8.2 \end{bmatrix}, \quad B'' = \begin{bmatrix} 0 \\ 30 \\ 30 \end{bmatrix}.$$

It is readily seen that CLOSEPLANT1 satisfies(x_0', ζ', T') IOSPECPLANT where $x_0' = \overline{0}$, $T' = R(1.4,\infty)$, $\zeta(x) = \pi_1(x)$, hence

$\zeta'(\sigma(f,t)(x_0'))$

$= \pi_1(\exp(A''*t)*x + \int_0^t \exp(A''*(t-\tau))*B''*f(\tau)*d\tau)$, where if f=<u>unitstep</u>, then

$|1-\zeta'(\sigma(f,t)(x_0'))|$

$= |1-(1-0.086*\exp(-0.95*t)-1.25*\exp(-2.89*t)+0.337*\exp(10.96*t))|$

≤ 0.05 for all $t \in T' = R(1.32,\infty)$, and if

f=<u>unitramp</u>,

$\zeta'(\sigma(f,t)(x_0'))$

$= t-0.493+0.095*\exp(-0.95*t)+0.43*\exp(-2.89*t)-0.031*\exp(-10.96*t)$

$|t-\zeta'(\sigma(f,t)(x_0'))|$

$= |0.493-0.095*\exp(-0.93*t)-0.43*\exp(-2.89*t)+0.031*\exp(-10.96*t)|$

< 1, for all $t \in R^{++}$.

Then it may be concluded that

(CLOSEPLANT1, x_0', ζ', T')$\in$IOCOTYLEDONPLANT.

With the proposed design, comparisons can be made in IOCOTYLEDONPLANT and TECHOTYLEDONPLANT. The results are tabulated in Tables 1 and 2. Three systems are involved, namely: PLANT, DUMYZ and CLOSEPLANT1.

TABLE 1

Analysis of the Three Systems

Output	PLANT	CLOSEPLANT1	DUMYZ
0.63	0.57 sec.	0.48 sec.	0^{+} sec.
0.95	0.87 sec.	1.32 sec.	0^{+} sec.

Evidently satisfyingtime(PLANT) $\subset R(1.5,\infty) \subset$ satisfyingtime (CLOSEPLANT1) $\subset$ satisfyingtime(DUMYZ); $0^{+}, 0.57 \in R(0.25,0.5)$, and $0.48 \in R(0.25,0.5)$.

TABLE 2

Ordering of DUMYZ, PLANT and CLOSEPLANT1

	MOIO*	MOTECO**
(PLANT,DUMYZ)	1	-
(PLANT, CLOSEPLANT1)	1	1
(DUMYZ, CLOSEPLANT1)	0	-
(DUMYZ, PLANT)	0	-
(CLOSEPLANT1, PLANT)	0	1
(CLOSEPLANT1, DUMYZ)	0	-

* MOIO = meritorderingsIOSPECPLANT
** MOTECO = meritorderingsTECHOTYLEDON

The dashes in Table 2 indicate no comparison made since DUMYZ is not included in TECHOTYLEDONPLANT. In other words, the table simply says that PLANT and CLOSEPLANT1 are equally good in the sense of implementation, but CLOSEPLANT1 is better than PLANT in satisfying IOSPECPLANT. Also CLOSEPLANT1 is better than DUMYZ in comparison with IOSPECPLANT, but DUMYZ is ruled out in the sense of implementation.

The use of the two merit orderings has been demonstrated; as it was mentioned before, there is yet another cotyledon and hence another ordering or comparison criterion. This will be done in the next chapter.

CHAPTER 6

TRICOTYLEDON

The two cotyledons, namely, IOCOTYLEDONPLANT and TECHOTYLEDONPLANT have been defined; two systems that are both in IOCOTYLEDONPLANT and TECHOTYLEDONPLANT have been described. To complete the tricotyledon theory, the intersection of IOCOTYLEDONPLANT and TECHOTYLEDONPLANT must be defined; that would naturally introduce another comparison criterion. This is done as follows.

FEASIBILITYCOTYLEDON(IOCOTYLEDONPLANT, TECHOTYLEDONPLANT)

The heading denotes the cotyledon determined by IOCOTYLEDONPLANT and TECHOTYLEDONPLANT. Every entry in this cotyledon both satisfies IOCOTYLEDONPLANT and TECHOTYLEDONPLANT. Let this be named and defined as FEASIBILESYSTEMS

$= \{(Z,x,\zeta,T,K,Z',Z',\rho,\mu,\theta)$:

$(Z,x,\zeta,T) \in$ IOCOTYLEDONPLANT,

$(K,Z') \in$ TECHOTYLEDONPLANT,

Z is simulated(Z'',ρ,μ,θ) by $Z'\}$, where the statement that Z is simulated(Z'',ρ,μ,θ) by Z' implies that Z'' is a subsystem of Z' and Z is homomorphic(ρ,μ,θ) image of Z''. It has been found that $(\text{PLANT},x_0,\zeta,T) \in$ IOCOTYLEDONPLANT,

PLANT is simulated(SUBPLANTCOUPLE,ρ,μ,θ) by RESULTANT-PLANTCOUPLE, where

$x_0 = \overline{0}$,

$\zeta(x) = \pi_1(x)$, T=R(0.87-δ,0.87+δ), and

ρ,μ,θ were defined previously.

Hence (PLANT,x_0,ζ,T,PLANTCOUPLE,RESULTANTPLANTCOUPLE, SUBPLANTCOUPLE,ρ,μ,θ)$\in$FEASIBLESYSTEMS.

Also (CLOSEPLANT1,x_0',ζ,T)$\in$IOCOTYLEDONPLANT, where

$x_0' = \overline{0}$,

$\zeta'(x) = \pi_1(x)$,

$T' = R^{++}$,

(OPTIMALCOUPLE1,RESULTANTCLOSEPLANT1)$\in$TECHOTYLEDONPLANT, CLOSEPLANT1 is simulated(SUBCLOSEPLANT1,ρ',μ',δ') by RESULTANTCLOSEPLANT1 for some ρ',μ',θ', and SUBCLOSE-PLANT1. Hence (CLOSEPLANT1,x_0',ζ',T',OPTIMALCOUPLE1, RESULTANTCLOSEPLANT1,SUBCLOSEPLANT1,ρ',μ',θ')$\in$FEASIBLE-SYSTEMS. Now there are two systems at hand that are both in FEASIBLESYSTEMS, the comparisons have been made in IOCOTYLEDONPLANT and TECHOTYLEDONPLANT, but these comparison criteria are still not sufficient to give a final decision. The trade-off criterion comes into play at this stage; it is defined on FEASIBLESYSTEMS.

The Trade-off Merit Orderings

Let u,v$\in$FEASIBLESYSTEMS, then define

<u>fexmerIO</u>(u,v)

= feasibilityextension(meritorderingsIOSPECPLANT)(u,v),

= 1, if and only if

meritorderingsIOSPECPLANT($\pi_{1,2,3,4}(u)$,$\pi_{1,2,3,4}(v)$) = 1.

Also define

fexmerTECH(u,v)

= feasibilityextension(meritorderingsTECHOTYLEDON) (u,v)

= 1, if and only if

meritorderingsTECHOTYLEDON($\pi_{5,6}(u)$,$\pi_{5,6}(v)$) = 1.

Note that fexmerIO and fexmerTECH are essentially the same as meritorderingsIOSPECPLANT and meritorderingsTECHOTYLEDON, respectively (17). Only the domains of the functions are changed. Then let

tradeoffcriterion

= meritorderings(IOCOTYLEDONPLANT,TECHOTYLEDONPLANT, fexmerIO,fexmerTECH),

= {γ: $\gamma \in$ORDERINGS(FEASIBLESYSTEMS), γ is consistent with fexmerIO and fexmerTECH}.

The requirement of consistency is best illustrated by Table 3. Z' and Z" are the two systems to be compared. The table illustrates the definition of consistency, namely, tradeoffcriterion is consistent with fexmerIO and fexmerTECH if and only if for every (Z',Z") $\in$FEASIBLESYSTEMS^2,

(1) if fexmerIO(Z',Z") = 1 and fexmerTECH(Z',Z") = 1, then tradeoffcriterion(Z',Z") = 1 (cases 1,2,3,4),

(2) if <u>fexmerIO</u>(Z",Z') = 1 and <u>fexmerTECH</u>(Z",Z') = 1, but either <u>fexmerIO</u>(Z',Z") = 0 or <u>fexmerTECH</u>(Z',Z") = 0, then <u>tradeoffcriterion</u>(Z',Z") = 0 (cases 5,6,7).

TABLE 3

Consistency of tradeoffcriterion

Case	IO (Z',Z")*	IO (Z",Z')	TECH (Z',Z")**	TECH (Z",Z')	trade (Z',Z")***
1	1	1	1	1	1
2	1	1	1	0	1
3	1	0	1	1	1
4	1	0	1	0	1
5	1	1	0	1	0
6	0	1	1	1	0
7	0	1	0	1	0
8	1	1	0	0	?(0)
9	1	0	0	1	?(0)
10	1	0	0	0	?(0)
11	0	1	1	0	?(0)
12	0	1	0	0	?(0)
13	0	0	1	1	?(0)
14	0	0	1	0	?(0)
15	0	0	0	1	?(0)
16	0	0	0	0	?(0)

*IO(Z',Z") = <u>fexmerIO</u>(Z',Z"),
**TECH(Z',Z") = <u>fexmerTECH</u>(Z',Z"),
***trade(Z',Z") = <u>tradeoffcriterion</u>(Z',Z").

Since it is hopted that the system will satisfy both classical specifications as shown by meritorderings-IOSPECPLANT and performance indices of optimal control as shown by meritorderingsTECHOTYLEDON, both concepts will be weighed equally heavy. One obvious and simple choice of tradeoffcriterion is the product of meritorderingsIOSPEC-PLANT and meritorderingsTECHOTYLEDON. Then define

tradeoffcriterion

= meritorderingsIOSPECPLANT * meritorderingsTECHOTYLEDON.

Essentially, this says that considerations will be given only to systems with all states fed back as shown in Fig. 11, that satisfy IOSPECPLANT. Cases 8 through 16 are then marked by (0). That the product of MERITORDERINGS (IOCOTYLEDON) * MERITORDERINGS(TECHOTYLEDON) gives a consistent ordering has been proved by Wymore (16).

Solution to the Problem

The problem has now been rigorously defined in TTSD. tradeoffcriterion has led the designer to look into the desirable forms of the systems. The solution using classical control techniques done by Schultz and Melsa (14) satisfies tradeoffcriterion which decides it is optimal. Alternatively, since an efficient algorithm exists defined by meritorderingsTECHOTYLEDON, the optimal system with respect to this ordering could be obtained by solving the Matrix Riccati equation as shown in Chapter 1. But the Matrix

Riccati solution is not optimal with respect to tradeoff-criterion, since it does not even satisfy IOSPECPLANT. Fortunately, since from classical control, with an integrator leading to the output, k_1 must be 1 to achieve zero steady state error, then by dividing the parameters k_1, k_2 and k_3 by k_1 and adding a 3XATTENUATOR, the optimal system is obtained, which is just CLOSEPLANT1.

Concluding Remarks

It was pointed out that the design methodology of TTSD is defined by the 6-tuple. The first five of them have been discussed. The last entry will not be discussed here since it involves building the system. Computers, analog or digital may be used to test the system built.

The paper demonstrated, to some extent, TTSD, and its application to defining rigorously a design problem in control field. Essentially, three different representations of the system have been presented, namely, the differential equation approach, the state variable model and the representation in TTSD. Those who are in the field of control may wonder why TTSD is used in a control system design problem, since so many jobs have been done without misunderstandings. It is very true in this regard. But the application of TTSD in the control field is only one of the many circumstances that TTSD is applicable (15,16). Using TTSD in the field of control is just like "killing a mouse with a big cannon." The usefulness of TTSD is best appreciated when it is applied to any large-scale system, for instance, social systems, health systems, etc. Besides, as it was mentioned earlier, the more areas that TTSD can be applied, the better is the need of communication satisfied. Even in this example, TTSD has achieved a modern control design to meet a classical control specification. This cannot be achieved by using Riccati's or other modern control methods. In addition to that, optimal control engineers often find it the most difficult to obtain a satisfactory performance index. It is believed that by TTSD, by the effort of an interdisciplinary team, a satisfactory performance index or merit ordering could be obtained. TTSD is only at its infant age, only the groundwork has been laid. Some time, some day, when TTSD begins to mature, efficient algorithms will be developed for its various applications.

APPENDIX A

TTSD; DEFINITIONS

All the definitions in Appendix A are quoted directly from (17). The definitions are arranged in alphabetical order.

Buildability

A system Z is buildable in the technology $\mathcal{T}$ if and only if there exists a coupling recipe K such that COMPONENTS(K)$\subset \mathcal{T}$ and Z=RESULTANT(K).

Consistency of Orderings

Let A be a set not empty and let $\alpha,\zeta,\gamma\in$ORDERINGS(A). Then γ is consistent with α and β if and only if: for $a,b\in A$

1) $\alpha(a,b)=\beta(a,b)=1$ imply $\gamma(a,b)=1$;
2) $\alpha(b,a)=\beta(b,a)=1$, and, $\alpha(a,b)=0$ or $\beta(a,b)=0$, imply $\gamma(a,b)=0$.

Coupling Functions

Let C=(ζ,i,o) be a coupling recipe. A function k is a coupling function with respect to the coupling recipe C if and only if:

1) k$\in$FUNCTIONS(RESULTANTINPUTFUNCTIONS(C) $\times$ RESULTANTSTATESET(C), TOTALINPUTFUNCTIONS(C));

2) for every f∈RESULTANTINPUTFUNCTIONS(C), x∈RESULTANTST-ATESET(C), t∈RESULTANTTIMESCALE(C), and V∈TOTALINPUTP-ORTS(C):

((k(f,x))(t))(V)

=(f(t))(V) if V∈TOTALUNOCCUPIEDPORTS(C);

=((o(Z,Z'))(((motion(Z))((projection(INPUTPORTS(Z)))∘k(f,x), t))(x(STATES(Z)))))(V)

if V∉TOTALUNOCCUPIEDPORTS(C), V∈i(Z',Z) for some Z,Z'∈Z;

3) if g∈TOTALINPUTFUNCTIONS(C), x∈RESULTANTSTATESET(C), s∈RESULTANTTIMESCALE(C), and (g(t))(V)

=((o(Z,Z'))((motion(Z))((projection(INPUTPORTS(Z)))∘g, t))(x(STATES(Z)))))(V) for every V∈i(Z',Z), for every Z,Z'∈ and for every t∈RESULTANTIMESCALE(C)[o,s) if s≥o, or for every t∈RESULTANTTIMESCALE(C)[s,o) if s<o, then (k((projection(TOTALUNOCCUPIEDPORTS(C))∘g,x))(t)=g(t) for every

t∈RESULTANTTIMESCALE(C)[o,s) if s≥o, or for every t∈RESULTANTTIMESCALE(C)[s,o) if s<o;

4) if f, g∈RESULTANTINPUTFUNCTIONS(C), s∈RESULTANTTIMESCALE(C), s≥o, and restriction(f,RESULTANTIMESCALE(C)[o,s))

=restriction(g,RESULTANTTIMESCALE(C)[o,s)), then restriction(k(f,x), RESULTANTTIMESCALE(C)[o,s))

=restriction(k(g,x), RESULTANTTIMESCALE(C)[o,s)).

5) if f,g∈RESULTANTINPUTFUNCTIONS(C), s∈RESULTANTTIMESSC-ALE(C), s<o, and <u>restriction</u>(f,RESULTANTTIMESCALE(C)[o,s)) =<u>restriction</u>(g,RESULTANTTIMESCALE(C)[s,o)), then <u>restriction</u>(k(f,x), RESULTANTTIMESCALE(C)[s,o)).

<u>Coupling Recipes</u>

A coupling recipe is a triple C=(ζ,i,o) where:

1) ζ is a set not empty of systems such that for every Z,Z'∈ζ:

TIMESCALE(Z) = TIMESCALE(Z');

INPUTPORTS(Z) $\bigcap$ INPUTPORTS(Z') = ∅;

2) i is a function defined on ζ^2 such that for every Z,Z',Z"∈ζ, i(Z,Z')⊂ INPUTPORTS(Z), i(Z,Z')$\bigcap$i(Z,Z)=∅, if Z'≠Z", and $\bigcup$\{INPUTPORTS(Z)~i(Z,Z'): Z'∈ζ, Z∈ζ\}≠∅;

3) o is a function defined on ζ^2such that for every Z,Z'∈Z, o(Z,Z')∈FUNCTIONS(STATES(Z), $\times$i(Z',Z)) if i(Z',Z)≠∅; o(Z,Z')=∅ if i(Z',Z)=∅.

If C=(ζ,i,o) is a coupling recipe and Z∈ζ then the set of system components of C, the input port assignments of C, the output function assignments of C, the set of input ports of Z designated as occupied by C, the set of input ports of Z left unoccupied by C, the set of states of the resultant determined by C, the total set of input ports managed by C, the total input determined by C, the set of total input functions determined by C, the set of unoccupied input ports of the resultant determined by C, and the time scale of the resultant determined by C are denoted, respectively, COMPONENTS(C), inputports(C), outputs(C), OCCUPIEDPORTS(C), UNOCCUPIEDPORTS(C), RESULTANTSTATESET(C), TOTALINPUTPORTS(C), TOTALINPUT(C), TOTALINPUTFUNCTIONS(C), TOTALUNOCCUPIEDPORTS(C), RESULTANTINPUTS(C), and RESULTANTTIMESCALE(C), and are defined, respectively, as follows:

COMPONENTS(C)=$\mathcal{Z}$;

inputports(C)=i;

outputs(C)=o;

OCCUPIEDPORTS(Z,C)= $\bigcup$ {i(Z,Z'):Z'$\in\mathcal{Z}$)};

UNOCCUPIEDPORTS(Z,C)=INPUTPORTS(Z) $\sim$ OCCUPIEDPORTS(Z,C);

RESULTANTSTATESET(C)= $\times$ {STATES(Z'): Z'$\in\mathcal{Z}$};

TOTALINPUTPORTS(C)= $\bigcup$ {INPUTPORTS(Z'): Z'$\in\mathcal{Z}$};

TOTALINPUT(C)= $\times$ TOTALINPUTPORTS(C);

TOTALINPUTFUNCTIONS(C) = {f: f$\in$FUNCTIONS(REALS,TOTALINPUT(C)), (projection(INPUTPORTS(Z')))$\cdot$f$\in$INPUTFUNCTIONS(Z') for every Z'$\in\mathcal{Z}$};

TOTALUNOCCUPIEDPORTS = $\bigcup$ {UNOCCUPIEDPORTS(Z',C): Z'$\in\mathcal{Z}$};

RESULTANTINPUTS = $\times$ TOTALUNOCCUPIEDPORTS(C);

RESULTANTINPUTFUNCTIONS = {f:f$\in$FUNCTIONS(REALS, RESULTANTINPUTS(C)), there exists g∈TOTALINPUTFUNCTIONS(C) such that f=(projection(TOTALUNOCCUPIEDPORTS(C)))•g}

RESULTANTTIMESCALE(C)=TIMESCALE(Z).

The Feasibility Cotyledon

Let $\mathcal{S}$ be an input/output specification and let $\mathcal{T}$ be a technology. Then the feasibility cotyledon determined by $\mathcal{S}$ and $\mathcal{T}$ is denoted FEASIBILITYCOTYLEDON($\mathcal{S}$,$\mathcal{T}$) and is defined as follows:

FEASIBILITYCOTYLEDON($\mathcal{S}$,$\mathcal{T}$)

={(Z,x,ζ,T,K,Z',Z" ρ,μ,θ):

(Z,x,ζ,T)∈IOCOTYLEDON($\mathcal{S}$),

(K,Z')∈TECHOTYLEDON($\mathcal{T}$), Z is simulated(Z",ρ,μ,θ) by Z'}.

Let u=(Z,x,ζ,T,K,Z',Z",ρ,μ,θ), u∈FEASIBILITYCOTYLEDON($\mathcal{S}$,$\mathcal{T}$),

then u is said to be an $\mathcal{S}/\mathcal{T}$ feasible solution and

the satisfying system artifact of the feasible solution u,

the satisfying initial state artifact of the feasible solution u,

the satisfying output function artifact of the feasible solution u,

the satisfying time scale artifact of the feasible solution u,

the blueprints in the feasible solution u,

the system simulating model artifact of the feasible solution u,

the subsystem artifact of the feasible solution u, the time scale matching artifact of the feasible solution u,

the input-function matching artifact of the feasible solution u,

the state matching artifact of the feasible solution u,

are denoted, respectively: SATISFYINGSYSTEM(u), satisfyinginitialstate(u), satisfyingoutputfunction(u), SATISFYINGTIMESCALE(u), BLUEPRINTS(u), SYSTEMODEL(u), SUBSYSTEM(u), timescalematch(u), inputfunctionmatch(u), and statematch(u), and defined, respectively, as follows:

SATISFYINGSYSTEM(u) = Z;

satisfyinginitialstate(u) = x;

satisfyingoutputfunction(u) = ζ;

SATISFYINGTIMESCALE(u) = T;

BLUEPRINTS(u) = K;

SYSTEMODEL(u) = Z';

SUBSYSTEM(u) = Z";

timescalematch(u) = ρ;

inputfunctionmatch(u) = μ; and

statematch(u) = θ.

Figures of Merit

If C is an input/output cotyledon and $u \in C$, then the system coordinate of u is (projection(1))(u). If C is a technology cotyledon and $u \in C$, then the system coordinate of u is (projection(2))(u). If C is a feasibility cotyledon and u∈C, then the system coordinates of u are (projection(1))(u), (projection(6))(u), and (projection (7))(u).

Let C be a cotyledon. Let $\mathcal{Z}$ be a set of systems (coordinates of C) defined as follows:

= RANGE(projection(1)) if C is an input/output cotyledon,

= RANGE(projection(2)) if C is a technology cotyledon;

$\mathcal{Z}$ is any one of RANGE(projection(1)), RANGE(projection(6)) or RANGE(projection(7)) if C is a feasibility cotyledon. Let E be a vector space. For every Z∈$\mathcal{Z}$, let index(Z) be a performance index for Z with values in VECTORS(E), and let flip(Z) be a finite probability distribution over SYSTEMEXPERIMENTS(Z). Then the figure of merit over C determined by index and flip is denoted figureofmerit(C, index,flip) and is defined as follows: If u∈C, and if Z=(projection(k))(u) where k∈INTEGERS such that $\mathcal{Z}$ = RANGE (projection(k)), then

(figureofmerit(C,index,flip))(u)

=expectedvalue(index(Z),SYSTEMEXPERIMENTS(Z),flip(Z)).

Implementation

A system Z is implementable in a technology $\mathcal{T}$ if and only if there exists a system Z' buildable in the technology $\mathcal{T}$ such that Z' simulates Z.

Input/Output Specifications

An input/output specification is a 5-tuple:

$\mathcal{S} = (P, F, Q, G, \eta)$ where:

P is a set not empty;

F FUNCTIONS(REALS,P), $F \neq \emptyset$;

Q is a set not empty;

G FUNCTIONS(REALS,Q) and $G \neq \emptyset$;

If $\mathcal{S} = (P, F, Q, G, \eta)$ is an input/output specification, then: the set of inputs specified by $\mathcal{S}$, the set of input functions specified by $\mathcal{S}$, the set of outputs specified by $\mathcal{S}$, the set of output functions specified by $\mathcal{S}$, the function specified by $\mathcal{S}$ for matching input functions with subsets of output functions, are denoted, respectively, INPUTS($\mathcal{S}$), INPUTFUNCTIONS($\mathcal{S}$), OUTPUTS($\mathcal{S}$), OUTPUTFUNCTIONS($\mathcal{S}$), matchingfunction($\mathcal{S}$), and are defined, respectively, as follows:

INPUTS($\mathcal{S}$) = P;

INPUTFUNCTIONS($\mathcal{S}$) = F;

OUTPUTS($\mathcal{S}$) = Q;

OUTPUTFUNCTIONS($\mathcal{S}$) = G;

matchingfunction($\mathcal{S}$) = η.

The I/O Cotyledon

Let $\mathcal{S}$ be an input/output specification. Then the input/output cotyledon generated by $\mathcal{S}$ is denoted

IOCOTYLEDON($\mathcal{S}$) and is defined as follows: IOCOTYLEDON($\mathcal{S}$) = {(Z,x,ζ,T): Z is a system and $\mathcal{S}$ is satisfied(x,ζ,T) by Z}.

If s=(Z,x,ζ,T), s$\in$IOCOTYLEDON($\mathcal{S}$), then: s is said to be an assertion of the satisfaction of $\mathcal{S}$, and the system satisfying artifact of the assertion s, the initial state artifact of the assertion s, the output function artifact of the assertion s, and the timescale artifact of the assertion s, are denoted, respectively: SATISFYINGSYSTEM(s), satisfyinginitialstate(s), satisfyingoutputfunction(s), and SATISFYINGTIMESCALE(s), and defined, respectively, as follows: SATISFYINGSYSTEM(x)=Z; satisfyinginitialstate(s) = x; satisfyingoutputfunction(s) = ζ; SATISFYINGTIMESCALE(s) = T.

Merit Orderings

Let $\mathcal{S}$ be an input/output specification and $\mathcal{T}$ be a technology. Then the set of merit orderings over the input/output cotyledon determined by $\mathcal{S}$, the set of merit orderings over the technology cotyledon determined by $\mathcal{T}$, and the set of merit orderings over the feasibility cotyledon determined by $\mathcal{S}$ and $\mathcal{T}$, are denoted, respectively, MERITORDERINGSIOCOTYLEDON($\mathcal{S}$), MERITORDERINGSTECHOTYLEDON($\mathcal{T}$), and MERITORDERINGSFEASIBILITYCOTYLEDON($\mathcal{S}$,$\mathcal{T}$), and defined, respectively, as follows:

MERITORDERINGSIOCOTYLEDON($\mathcal{S}$),

=ORDERINGS(IOCOTYLEDON($\mathcal{S}$)),

MERITORDERINGSTECHOTYLEDON($\mathcal{T}$),

=ORDERINGS(TECHOTYLEDON($\mathcal{T}$)), and

MERITORDERINGSFEASIBILITYCOTYLEDON($\mathcal{S},\mathcal{T}$)

=ORDERINGS(FEASIBILITYCOTYLEDON($\mathcal{S},\mathcal{T}$)).

Orderings

Let A be a set. The set of orderings over A is denoted ORDERINGS(A) and is defined as follows: ORDERINGS(A)

$=\{\alpha:\alpha\in$FUNCTIONS$(A^2,\{0,1\})$ such that for every $a\in A, b\in A$, and $c\in A$, $\alpha(a,a)=1$, and, if $\alpha(a,b)=1$ and $\alpha(b,c)=1$, then $\alpha(a,c)=1\}$.

Performance Indices

If Z is a system, then a performance index for Z is any function defined on SYSTEMEXPERIMENTS(Z) with values in a non-empty set.

Projection, π

If $\mathcal{A}$ is a set not empty of sets not empty, then the Cartesian product of the sets in $\mathcal{A}$ is denoted $\times\mathcal{A}$ and is defined as follows:

$\times\mathcal{A}$

$=\{x:x\in$FUNCTIONS$(\mathcal{A},\bigcup\mathcal{A}), x(Z)\in A$ for every $A\in\mathcal{A}\}$ if $\mathcal{A}\supset\{A,B\}$ for some sets A and B, $A\neq B$;

= A if $\mathcal{A} = \{A\}$ for some set A.

If $\mathcal{A}$ is a set not empty of sets not empty and $\mathcal{B}\in$SUBSETS($\mathcal{A}$), $\mathcal{B}\neq\emptyset$, then the projection of $\mathcal{A}$ on the subspace determined by $\mathcal{B}$ is denoted projection($\mathcal{B}$) and is defined as follows:

π(B)

=projection($\mathcal{B}$)

= $\{(x,y): x\in \times\mathcal{A},\ y\in\times\mathcal{B},\ y=$restriction$(x,\mathcal{B})\}$.

RANGE

If f$\in$FUNCTIONS(A,B), then the range of f is denoted RANGE(f) and is defined as follows: RANGE(f) = $\{b: b\in B$, there exists $a\in A$ such that $b=f(a)\}$.

Restriction

If A and B are sets not empty and f$\in$FUNCTIONS(A,B) and A'$\in$SUBSETS(A), then the restriction of f to A' is dnoted restriction(f,A') and is defined as follows:

restriction(f,A')

= $\{(a,b): a\in A', b\in B, b=f(a)\}$.

Resultants

If C is a coupling recipe and k is a coupling function with respect to C, then the resultant of C is an assemblage denoted RESULTANT(C) and is defined as follows:

STATES(RESULTANT(C))

=RESULTANTSTATESET(C);

INPUTS(RESULTANT(C))

=RESULTANTINPUTS(C);

INPUTFUNCTIONS(RESULTANT(C))

=RESULTANTINPUTFUNCTIONS(C);

BEHAVIOR(RESULTANT(C))

=RANGE(<u>motion</u>(RESULTANT(C)));

TIMESCALE(RESULTANT(C))

=RESULTANTTIMESCALE(C);

if f∈INPUTFUNCTIONS(RESULTANT(C)),

t∈TIMESCALE(RESULTANT(C)), and

x∈STATES(RESULTANT(C)), then

(((<u>motion</u>(RESULTANT(C)))(f,t))(x))(STATES(Z))

=((<u>motion</u>(Z))((<u>projection</u>(INPUTPORTS(Z)))∘k(f,x),t))

(x(STATES(Z))) for every Z∈$\mathcal{Z}$.

A coupling recipe is a system couple if and only if there exists a coupling function k with respect to C.

<u>Satisfaction</u>

An input/output specification S is satisfied (x,ζ,T) by an assemblage Z if and only if:

1) x∈STATES(Z);

2) INPUTS(Z) = INPUTS(S);

3) INPUTFUNCTIONS(Z) ⊃ INPUTFUNCTIONS(S);

4) ζ∈FUNCTIONS(STATES(Z),OUTPUTS(S));

5) T⊂TIMESCALE(Z), T≠∅;

6) for every f∈INPUTFUNCTIONS(S), there exists

$g \in ((\underline{\text{matchingfunction}}(S))(f))$ such that for every $t \in T$, $\zeta(((\underline{\text{motion}}(Z))(f,t))(x)) = g(t)$.

Subsystems

Let Z and Z' be systems. Then Z is a subsystem of Z' if and only if:

STATES(Z) $\subset$ STATES(Z');

INPUTS(Z) $\subset$ INPUTS(Z');

INPUTFUNCTIONS(Z) $\subset$ INPUTFUNCTIONS(Z');

TIMESCALE(Z) $\subset$ TIMESCALE(Z');

and for every $f \in$ INPUTFUNCTIONS(Z) and $t \in$ TIMESCALE(Z);

($\underline{\text{motion}}$(Z))(f,t)

=$\underline{\text{restriction}}$(($\underline{\text{motion}}$(Z'))(f,t),STATES(Z)).

System Design

A system design project is a 6-tuple $P=(S,T,\alpha,\beta,\gamma,D)$ where S is an input/output specification;

T is a technology;

$\alpha \in$ MERITORDERINGSIOCOTYLEDON(S);

$\beta \in$ MERITORDERINGSTECHOTYLEDON(T);

$\gamma \in$ MERITTRADEOFFORDERINGS(S,T,α,β);

$D \in$ TESTPLANS(S,T,γ).

The input/output specification of the system design project P, the technology of the system design project P, the input/output merit ordering of the system design project P, the technology merit ordering of the system design project $\mathcal{P}$, the feasibility merit trade off ordering of the system design project $\mathcal{P}$, the test plan of the system design project $\mathcal{P}$, and the set of end items of the system design project $\mathcal{P}$, are denoted, respectively: IOSPEC($\mathcal{P}$), TECHNOLOGY($\mathcal{P}$), IOMO($\mathcal{P}$), TECHMO($\mathcal{P}$), TRADEOFFMO($\mathcal{P}$), TESTP-LAN($\mathcal{P}$), and ENDITEMS($\mathcal{P}$), and are defined, respectively, as follows:

IOSPEC($\mathcal{P}$) $= \mathcal{S}$;

TECHNOLOGY($\mathcal{P}$) $= \mathcal{T}$,

IOMO($\mathcal{P}$) $= \alpha$;

TECHMO($\mathcal{P}$) $= \beta$;

TRADEOFFMO($\mathcal{P}$) $= \gamma$;

TESTPLAN($\mathcal{P}$) $= \mathcal{D}$;

ENDITEMS($\mathcal{P}$)

$=\{(u,Z^R,$<u>outcome</u>$(u,Z^R,$<u>systemtests</u>$(\mathcal{D})))$:

$(u,Z^R)\in$(<u>setinverse</u>(<u>systemtests</u>($\mathcal{D}$)))($\{u\in M$:

$M\in$PARTITIONS(FEASIBILITYCOTYLEDON($\mathcal{S},\mathcal{T}$), <u>testdecision-framework</u>($\mathcal{D}$))$\}\times\{$(<u>projection</u>(6))(u) <u>is an adequate model for</u> $Z^R\}\times\{Z^R$ <u>is acceptable</u>$\})\}$.

<u>System Homomorphisms</u>

Let Z and Z' be assemblages, Z=(S,P,F,T,σ) and Z=(S',P',F',T',σ). Then Z is the homomorphic(ρ,μ,θ) image of Z' if and only if:

1) $\rho\in$FUNCTIONS(T',onto,T);

2) $\mu\in$FUNCTIONS(F',onto,F)

such that for every f, $g \in F'$, and $r \in$ REALS, μ(segmentation-(f,g))=segmentation(μ(f), μ(g)) and μ(translation(f,r)) =translation(μ(f),ρ(r));

3) $\theta \in$ FUNCTIONS(S',onto,S) such that for every $f \in F'$, $t \in T'$, and $x \in S'$, $\theta(\sigma'(f,t)(x))=\sigma(\mu(f),\rho(t))(\theta(x))$.

A system Z is an homomorphic image of a system Z' if and only if there exist ρ,μ, and θ such that Z is the homomorphic(ρ,μ,θ) image of Z'.

System Simulations

A system Z' simulates(Z",ρ,μ,θ) a system Z if and only if Z" is a subsystem of Z' and Z is an homomorphic-(ρ,μ,θ) image of Z".

A system Z' simulates a system Z if and only if there exist Z",ρ,μ, and θ such that Z is simulated(Z",ρ, μ,θ) by Z'.

Systems

A system is an assemblage Z=(S,P,F,T,σ) such that:

1) σ(f,o)= identity(S) for every $f \in F$;

2) σ(translation(f,s),t)$\cdot\sigma$(f,s+t) for every $f \in F$, s, $t \in T$ such that $s+t \in T$;

3) $\sigma(f,r)= \sigma(g,r)$ for every f, $g \in F$ and $r \in$ REALS if restriction(f,T[o,r)) = restriction(g,T[o,r)) when $r \geq 0$, or if restriction(f,T[r,o)) = restriction(g,T[r,o)) when $r<0$.

Let the set of all systems be denoted SYSTEMS.

<u>Systems - Admissible Sets of Input Functions</u>

Let P be a set not empty; let f,g$\in$FUNCTIONS(REALS, P); let r$\in$REALS. Then: the translation of the function f by the amount r is denoted <u>translation</u>(f,r) and is defined as follows: <u>translation</u>(f,r) = $\{(t,y):(t,y)\in$REALS $\times$ P, $y=f(r+t)\}$; the segmentation of the functions f and g is denoted <u>segmentation</u>(f,g) and is defined as follows: <u>segmentation</u>(f,g)={f,g}=$\{(t,y):(t,y)\in$NEGATIVEREALS $\times$ P, $y=f(t)\}$ $\bigcup$ $\{(t,y):(t,y)\in$NONNEGATIVEREALS $\times$ P, $y=g(t)\}$.

A set F is an admissible set of input functions with values in P if and only if F$\subset$FUNCTIONS(REALS,P); F$\neq\emptyset$; if f,g$\in$f, and r$\in$REALS, then <u>translation</u>(f,r)$\in$F and <u>segmentation</u>(f,g)$\in$F.

The set of all admissible sets of input functions with values in P is denoted ADMISSIBLES(P).

Let G$\subset$FUNCTIONS(REALS,P), G$\neq\emptyset$. Then the smallest admissible set of input functions with values in P that contains G is denoted ADMISSIBLESET(G) and is defined as follows:

ADMISSIBLESET(G)=$\bigcap$$\{F:F\in$ADMISSIBLES(P),F$\supset G\}$.

<u>Systems - Assemblages</u>

An assemblage is a 5-tuple, Z=(S,P,F,T,σ) where:

1) S is a set not empty;

2) P is a set not empty;

3) F$\in$ADMISSIBLES(P);

4) $T \subset$ REALS and $0 \in T$;

5) $\sigma \in$ FUNCTIONS($F \times T$,FUNCTIONS(S,S)).

If Z is an assemblage and Z=(S,P,F,T,σ), then the states of Z, the inputs of Z, the set of input ports of Z, the set of input functions of Z, the set of state transitions of Z, the time scale of Z, the state transition function of Z and the set of system experiments on Z are denoted, respectively, STATES(Z), INPUTS(Z), INPUTPORTS(Z), INPUTFUNCTIONS(Z), BEHAVIOR(Z), TIMESCALE(Z), motion(Z) and SYSTEMEXPERIMENTS(Z), and are defined, respectively, as follows:

STATES(Z)=S;

INPUTS(Z)=P;

INPUTPORTS(Z)

=I if P= $\times$ I for some set I not empty of sets not empty;

={ψ Ii):i$\in$I} if P= $\times${ψ(i):i$\in$I} for some index set I and indexing function ψ;

INPUTFUNCTIONS(Z)=F;

BEHAVIOR(Z)=RANGE(σ);

TIMESCALE(Z)=T;

motion(Z)=σ;

SYSTEMEXPERIMENTS(Z)=F $\times$ S $\times$ T.

If f$\in$F, t$\in$T, and x$\in$S, then the state of the assemblage Z at time t given the input function f and the initial state x is (σ(f,t))(x).

If $f \epsilon F$ and $x \epsilon S$, then the state trajectory of Z determined by f and x is denoted statetrajectory(Z,f,x) and is defined as follows:

statetrajectory(Z,f,x) = $\{(t,y):(t,y)\epsilon T \times S, y=(\sigma(f,t))(x)\}$.

An output function for Z is any $\zeta \epsilon$FUNCTIONS(S,Q) where Q is any set not empty.

If $f \epsilon F$, $x \epsilon S$ and $\zeta \epsilon$FUNCTIONS(S,Q), then the output trajectory of Z determined by ζ,f, and x is denoted outputtrajectory(Z,ζ,f,x) and is defined as follows: outputtrajectory(Z,ζ,f,x)

= $\zeta \circ$statetrajectory(Z,f,x).

Let the set of all assemblages be denoted ASSEMBLAGES.

Technology

A technology is a set $\mathcal{T}$ not empty of systems, closed under isomorphism, that is, if $Z \epsilon \mathcal{T}$ and Z' is isomorphic to Z, then $Z' \epsilon \mathcal{T}$.

The Technology Cotyledon

If $\mathcal{T}$ is a technology, then the technology cotyledon determined by is denoted TECHOTYLEDON($\mathcal{T}$) and is defined as follows: TECHOTYLEDON($\mathcal{T}$)

=$\{$(K,Z): K is a coupling recipe, COMPONENTS(K)$\subset \mathcal{T}$, Z is a system,

Z=RESULTANT(K)$\}$.

If b=(K,Z), b∈TECHOTYLEDON($\mathcal{T}$), then b is said to be a buildability assertion and the blueprints for the buildability assertion b and the model for the buildability assertion b are denoted, respectively, BLUEPRINTS(b) and SYSTEMODEL(b), and defined, respectively, as follows:

BLUEPRINTS(b)=K;

SYSTEMODEL(b)=Z.

The Tradeoff Merit Orderings

Let $\mathcal{S}$ be an input/output specification; let $\mathcal{T}$ be a technology; let α be a merit ordering over IOCOTYLEDON(S); let β be a merit ordering over TECHOTYLEDON($\mathcal{T}$). Then the feasibility extension of α and the feasibility extension of β are denoted, respectively, feaibilityextension(α), and feasibilityextension(β), and are defined, respectively, as follows: if u,v∈FEASIBILITYCOTYLEDON($\mathcal{S}$,$\mathcal{T}$) then

(feasibilityextension(α))(u,v)

=1 if and only if α(projection({1,2,3,4}))(u),

(projection({1,2,3,4}))(v)) = 1;

(feasibilityextension(β))(u,v)

=1 if and only if β((projection({5,6}))(u),

(projection({5,6}))(v))=1.

The set of merit tradeoff orderings over FEASIBILITYCOTYLEDON($\mathcal{S}$,$\mathcal{T}$) determined by α and β is denoted MERITRADEOFFORDERINGS($\mathcal{S}$,$\mathcal{T}$,α,β) and is defined as follows:

MERITRADEOFFORDERINGS(S,T,α,β)

$=\{\gamma:\gamma\in$ORDERINGS(FEASIBILITYCOTYLEDON(S,T), γ is consistent with <u>feasibilityextension</u>(α) and <u>feasibilityextension</u>(β)$\}$.

APPENDIX B*

A MATRIX RICCATI PROGRAM

```
      PROGRAM RICATI(INPUT, OUTPUT)
C     RICCATI EQUATION PROGRAM (RICATI)
      DIMENSION A(10,10),B(10,10),C(10,10),R(10,10),Q(10,10),
     * K(10,10),U(10,10),NAME(5),F(10,10),E(10,10),
     *  G(10,10),H(10,10),S(10,10),GG(10,10)
      REAL K
      INTEGER OPTION,BLANK
      DATA ICC,IFF,BLANK,ISS/1HC,1HF,1H ,1HS/
 1000 FORMAT (1H1,5X,37HOPTIMAL CONTROL/KALMAN FILTER PROGRAM/)
 1001 FORMAT (5A4,3I2)
 1002 FORMAT (6X,25HPROBLEM IDENTIFICATION = ,5A4)
 1003 FORMAT (1H0,5X,13H THE A MATRIX/)
 1004 FORMAT (1H0,5X,24H THE B MATRIX TRANSPOSED/)
 1005 FORMAT(1H0,5X,13H THE C MATRIX/)
 1006 FORMAT (8F10.3)
 1007 FORMAT (A1,9X,2F10.3,I3)
 1008 FORMAT(1H0,45(1H*))
 1009 FORMAT (1H0,5X,21H*** FILTER OPTION ***/)
 1010 FORMAT (1H0,5X,22H*** CONTROL OPTION ***/)
 1011 FORMAT(6(1PE20.8))
 1012 FORMAT(1H0,5X,13H THE R MATRIX/)
 1013 FORMAT(1H0,5X,13H THE Q MATRIX/)
 1014 FORMAT(1H0,5X,19H INITIAL CONDITIONS/)
 1015 FORMAT(1H0,5X,8H TIME = ,1PE20.8/6X,5HGAINS)
 1016 FORMAT(1H0,5X,21HSTEADY STATE SOLUTION//
     *   6X,6H GAINS/)
 1017 FORMAT (1H0,5X,13H THE P MATRIX/)
  100 READ 1001,(NAME(I),I=1,5),N,M,L
      PRINT 1000
      PRINT 1002,(NAME(I),I=1,5)
      PRINT 1008
      PRINT 1003
      DO 110 I=1,N
      READ 1006,(A(I,J),J=1,N)
  110 PRINT 1011, (A(I,J),J=1,N)
      PRINT 1004
      DO 120 I=1,M
      READ 1006,(B(J,I),J=1,N)
```

* This program is from Melsa and Jones (10) with some modifications.

```
120 PRINT 1011,(B(J,I),J=1,N)
    PRINT 1005
    DO 130 I=1,L
    READ 1006,(C(I,J),J=1,N)
130 PRINT 1011,(C(I,J),J=1,N)
150 READ 1007,OPTION,T1,T2,NPT
    PRINT 1008
    IF(OPTION.EQ.BLANK) GO TO 100
    IF(OPTION.EQ.ICC) GO TO 300
    IF (OPTION .EQ. ISS) STOP
    PRINT 1009
    NR=L
    NQ=M
    DO 230 I=1,N
    DO 210 J=1,L
210 E(I,J)=C(J,I)
    DO 220 J=1,M
220 D(I,J)=B(I,J)
    DO 230 J=1,N
230 F(I,J)=A(I,J)
    DO 320 J=1,L
320 D(I,J)=C(J,I)
    GO TO 400
300 PRINT 1010
    NR=M
    NQ=N
    DO 330 I=1,N
    DO 310 J=1,M
310 E(I,J)=B(I,J)
    DO 330 J=1,N
330 F(I,J)=A(J,I)
400 PRINT 1012
    DO 410 I=1,NR
    READ 1006,(R(I,J),J=1,NR)
410 PRINT 1011,(R(I,J),J=1,NR)
    PRINT 1013
    DO 420 I=1,NQ
    READ 1006,(Q(I,J),J=1,NQ)
420 PRINT 1011,(Q(I,J),J=1,NQ)
    DO 11 I=1, NR
    DO 11 J=1, NR
 11 H(I,J)=R(I,J)
    CALL MATRIX(10,NR,NR,2,H,NR,DET,0,0,0)
    DO 440 I=1,NR
    DO 440 J=1,N
    R(I,J)=0.0
    DO 440 II=1,NR
440 R(I,J)=R(I,J)+H(I,II)*E(J,II)
    IF(OPTION.EQ.ICC) GO TO 500
    DO 450 I=1,N
    DO 450 J=1,NQ
    G(I,J)=0.0
```

```
    DO 450 II=1,NQ
450 G(I,J)=G(I,J)+D(I,II)*Q(II,J)
    DO 460 I=1,N
    DO 460 J=1,N
    Q(I,J)=0.0
    DO 460 II=1,NQ
460 Q(I,J)=Q(I,J)+G(I,II)*D(J,II)
500 IF(NPT.GT.0) GO TO 530
    DO 520 I=1,N
    DO 510 J=1,N
510 G(I,J)=0.0
520 G(I,I)=0.1
    EPS=0.01
    ID=0.5
    GO TO 570
530 PRINT 1014
    DO 540 I=1,N
    READ 1006,(G(I,J),J=1,N)
540 PRINT 1011,(G(I,J),J=1,N)
    PRINT 1008
    TIME=ABS(T2-T1)
    PTS=200.0*TIME
    XX=NPT
    XX=PTS/XX
    ID=XX
    DI=ID
    YY=ABS(XX-DI)
    IF (YY.GT.0.05) ID=ID+1
    IT=PTS
    EPS=0.005
    TIME=T1
    IF(OPTION.EQ.ICC) TIME=T2
    PRINT 1015,TIME
    DO 560 I=1,NR
    DO 550 J=1,N
    K(I,J)=0.0
    DO 550 II=1,N
550 K(I,J)=K(I,J)+R(I,II)*G(II,J)
560 PRINT 1011,(K(I,J),J=1,N)
570 LC=0
    ICH=ID
575 DO 580 I=1,NR
    DO 580 J=1,N
    K(I,J)=0.0
    DO 580 II=1,N
580 K(I,J)=K(I,J)+R(I,II)*G(II,J)
```

```
      DO 590 I=1,N
      DO 590 J=1,N
      H(I,J)=0.0
      DO 590 II=1,NR
  590 H(I,J)=H(I,J)+E(I,II)*K(II,J)
      DO 610 I=1,N
      DO 610 J=1,N
      D(I,J)=Q(I,J)
      DO 600 II=1,N
  600 D(I,J)=D(I,J)+F(I,II)*G(II,J)+G(I,II)*F(J,II)-G(I,II)*
     * H(II,J)
  610 S(I,J)=G(I,J)+D(I,J)*EPS
      IF(NPT.LE.0) GO TO 640
      LC=LC+1
      IF(LC.LT.ICH) GO TO 625
      ICH=ICH+ID
      ALC=LC
      T=ALC*EPS
      TIME=T1+T
      IF (OPTION.EQ.ICC) TIME=T2-T
      PRINT 1017
      DO 615 I=1,N
  615 PRINT 1011, (G(I,J),J=1,N)
      PRINT 1015,TIME
      DO 620 I=1,NR
  620 PRINT 1011,(K(I,J),J=1,N)
      IF(LC.GE.IT) GO TO 150
  625 DO 630 I=1,N
      DO 630 J=1,N
  630 G(I,J)=S(I,J)
      GO TO 575
  640 SUM=0.0
      DO 650 I=1,N
      DO 650 J=1,N
      SUM=SUM+ABS(D(I,J))
      GG(I,J)=G(I,J)
  650 G(I,J)=S(I,J)
      IF (SUM.GT.0.0001) GO TO 575
      PRINT 1017
      DO 670 I=1,N
  670 PRINT 1011, (GG(I,J),J=1,N)
      PRINT 1016
      DO 660 I=1,NR
  660 PRINT 1011,(K(I,J),J=1,N)
      GO TO 150
      END
```

```
OPTIMAL CONTROL/KALMAN FILTER PROGRAM

PROBLEM IDENTIFICATION = CLOSEP_ANT1 SOLUTION

*******************************************

 THE A MATRIX

0.                   1.00000000E+00       0.
0.                  -2.00000000E+00      -4.00000000E+00
0.                   0.                  -5.00000000E+00

 THE B MATRIX TRANSPOSED

0.                   1.00000000E+01       1.00000000E+01

 THE C MATRIX

1.00000000E+00       0.                   0.

*******************************************

*** CONTROL OPTION ***


 THE R MATRIX

1.00000000E+00

 THE Q MATRIX

9.00000000E+00       3.00000000E+00       0.
3.00000000E+00       1.00000000E+00       0.
0.                   0.                   0.

 THE P MATRIX

1.36989231E+00       5.39101606E-01      -2.39101946E-01
5.39101606E-01       2.18572213E-01      -1.08849542E-01
-2.39101946E-01     -1.08849542E-01       7.68200173E-02

STEADY STATE SOLUTION

 GAINS

2.99999661E+00       1.09722571E+00      -3.20295248E-01

*******************************************
```

```
  10/22/73 UNIV. OF ARIZONA 3.4 LVL 355-H R.D 14.59
19.34.36.SJ  CHOLE8Z  FROM **   10/22/73
19.34.37.JOBCARD-CHOLEE,3N4152112 ,CM50000,T10.
19.34.37.FTN.
19.34.49.     3.778 CP SECONDS COMPILATION TIME
19.34.49.LGO.
19.36.10.     365753 CM REQUIRED FOR LOADING.
19.36.11.    STOP
19.36.11.     1.140 CP SECONDS EXECUTION TIME
19.36.11.DISPOSE,OUTPUT,*P2=CWN.
19.36.11.DMP(4700,5200)
19.36.11.DMP-ARG OUTSIDE FL
19.36.12.IP  00000012 BLOCKS, FILE INPUT  , DC 00
19.36.12. COST=     $.05
19.36.12.OP  00000039 BLOCKS, FILE OUTPUT , DC 42
19.36.12. COST=     $.09 ST= 00 FORM=WN COPIES=01
19.36.12.CP       7.317 SEC.        3.121 ADJ.
19.36.12.IO       6.368 SEC.         .954 ADJ.
19.36.12.CM     126.103 KWS.        1.538 ADJ.
19.36.13.SS                         5.616
19.36.13.COST     $.72/TOTAL       $.58/SS     $.14/UR
19.36.13.EJ  END OF JOB, **, MAX CM USED= 44000B
```

APPENDIX C*

TWO LEMMAS

Lemma 1:

Let $Z=(S,P,F,T,\sigma)$ be a system such that $T \subset R^{++}$ and T is an additive semigroup, let $U \subset S$. Assume that $P=V\times W$ where $V=W=R$ and $C_o \in \pi(W) \circ F$ where $C_o(t)=0$ for all $t \in R$. Then let $Z^*=(S^*,P^*,F^*,T^*,\sigma^*)$ be defined as follows:

$S^*=\{x: x \in S$, there exists some $y \in U$, $s \in T^*$, $g \in F^*$ such that $x=\sigma(g,s)(y)\}$,

$P^*=P$,

$F^*=\{f: f \in F, \pi(W) \circ f=C_o\}$,

$T^*=T$, and if $f \in F^*$, $t \in T^*$, $x \in S^*$,

$\sigma^*(f,t)=\text{res}(\sigma(f,t),S^*)$.

Then Z^* is a subsystem of Z.

Proof: The points worth mentioning are:

1. To show that $\sigma^*(f,t) \in \mathfrak{F}(S^*,S^*)$; if $x \in S^*$, then there exists $y \in U$, $g \in F^*$, $s \in T^*$, such that $x=\sigma(g,s)(y)$. Then, if $f \in F^*, t \in T^*$,

---

* The lemmas in this appendix originated from the work of Dr. Wymore.

$\sigma^*(f,t)(x)$

$=\sigma^*(f,t)(\sigma(g,s)(y));$

$=\sigma(f,t)\sigma(g,s)(y);$

$=\sigma(((g\to s)/f)\to -S,S+t)(y)$, hence

$\sigma^*(f,t)(x)$ is reachable from y.

2. $F^*\in$ADMISSIBLES(P^*) which follows through easily.

Lemma 2:

Let Z and Z* be defined as in Lemma 1. Let $Z_o=(S_o,P_o,F_o,T_o,\sigma_o)$ be a system where

$T = T_o$,

$\pi(V)\circ F^*=F_o$, let

$\theta\in \mathcal{F}(S^*,\text{onto},S_o)$, $\mu\in \mathcal{F}(F^*,F_o)$ such that $\mu(f)=\pi(V)\circ f$ for every $f\in F^*$. Furthermore suppose that for every $f\in F^*$, $t\in T^*$ and $x\in U$, there is a θ such that $\theta(\sigma^*(f,t)(x))=\sigma_o(\mu(f),t)(\theta(x))$. Then Z_o is an homomorphic-(ω,μ,θ) image of Z*.

Proof: The only thing that is needed to prove is that

$\mu(f\to s)=\mu(f)\to s$, $\mu(f/g)=\mu(f)/\mu(g)$ and that for every $y\in S^*$, $t\in F^*$, $t\in T^*$,

$\theta(\sigma^*(f,t)(y))=\sigma_o(\mu(f),t)(\theta(y))$.

Since $y\in S^*$, there exists $x\in U$ and $g\in F^*$, $s\in T^*$, such that $y=\sigma^*(g,s)(x)$. Then

$\theta(\sigma^*(f,t)(y))$

$=\theta(\sigma^*(f,t)\sigma^*(g,s)(x))$

$=\theta(\sigma^*(((g\to s)/f)\to -s,s+t)(x));$

$=\sigma_o(\mu(((g\to s)/f)\to -s),s+t)(\theta(x))$, [by assumption since $x\in U$ and $((g\to s)/f)\to s\in F^*$];

$=\sigma_o(((\mu(g)\to s)/\mu(f))\to -s,s+t)(\theta(x)))$;

$=\sigma_o(\mu(f),t)\sigma_o(\mu(g),s)(\theta(x))$;

$=\sigma_o(\mu(f),t)\theta(\sigma^*(g,s)(x))$, by assumption;

$=\sigma_o(\mu(f),t)\theta(y)$.

REFERENCES

1. Antosiewicz, H., "A Survey of Liapunov's Second Method," in S. Lefschetz (ed.), Contributions to the Theory of Non-linear Oscillations, Vol. 4, pp. 147-166, Princeton University Press, Princeton, N. J., 1958.

2. Bellman, R. E., "Kronecker Products and the Second Method of Lyapunov," Math. Nachr., 1959.

3. Bellman, R. E. and S. Dreyfus, Applied Dynamic Programming, Princton University Press, Princeton, N. J., 1962.

4. Bode, H. W., "Feedback Amplifier Design," Bell System Tech. J., 19 (1940).

5. Evans, W. R., "Graphical Analysis of Control Systems," Trans. AIEE, 67 (1948).

6. Kalman, R. E., "On the General Theory of Control Systems," Proceedings IFAC Moscow Congress, Vol. 1, pp. 481-492, Butterworth, Inc., Washington, D. C., 1960.

7. Kalman, R. E., "Mathematical Description of Linear Dynamical Systems," J. Soc. Ind. Appl. Math., Ser. A, Control, Vol. 1, No. 2, pp. 152-192, 1963.

8. Kalman, R. E., P. L. Falb and M. A. Arbib, Topics in Mathematical System Theory, McGraw-Hill, New York, 1969.

9. Maxwell, J. C., "On Governors," Proc. Roy. Society (London), 16 (1868).

10. Melsa, J. L. and S. K. Jones, Computer Programs for Computational Assistance in the Study of Linear Control Theory, McGraw-Hill, New York, 1973.

11. Melsa, J. L. and D. G. Schultz, Linear Control Systems, McGraw-Hill, New York, 1969.

12. Nyquist, H., "Regeneration Theory," Bell System Tech. J., 11 (1932).

13. Pontryagin, L. C., et al., The Mathematical Theory of Optimal Processes, John Wiley & Sons, Inc., New York, 1962.

14. Schultz, D. G. and Melsa, J. L., State Functions and Linear Control Systems, McGraw-Hill, New York, 1967.

15. Wymore, A. W., A Mathematical Theory of Systems Engineering: The Elements, John Wiley & Sons, Inc., New York, 1967.

16. Wymore, A. W., Systems Engineering Methodology for Interdisciplinary Team, University of Arizona, (unpublished manuscript).

17. Wymore, A. W., Heuristics, Linguistics & Mathematics of the Tricotyledon Theory of System Design, University of Arizona, (unpublished manuscript).