

https://dspace.org/

An approach to the study of organised systems

Kambanis, John

2015-04-22
http://hdl.handle.net/10539/17517

Downloaded from a DSpace repository. https://dspace.org/

AN APPROACH TO THE STUDY OF ORGANISED SYSTEMS

JOHN KAMBANIS

DISSERTATION SUBMITTED TO THE FACULTY OF SCIENCE,
UNIVERSITY OF THE WITWATERSRAND, JOHANNESBURG,
FOR THE DEGREE OF MASTER OF SCIENCE.

JOHANNESBURG 1983

ABSTRACT

The objective of this dissertation is to present some guidelines and some tools for the study of organised systems.

The basic concepts are initially developed as well as a language which is suitable for the description of diverse physical systems. Based on these the following two goals are pursued.

Firstly the fundamental mechanisms which are present in a class of organised systems are investigated and some of the main components and processes identified. This indicates that there is a fundamental similarity between these systems and provides guidelines for more detailed studies.

Secondly a technique is presented for analysing and modelling such systems. This technique is tested by constructing a computer model of an aspect of the economic system.

The above two goals are in line with the General Systems approach in that they identify common patterns in systems which are normally studied by different disciplines, and present certain tools for the extraction, representation and study of these patterns, independently of the physical nature of these systems.

DECLARATION

I declare that this dissertation is my own, unaided work. It is being submitted for the degree of Master of Science in the University of the Witwatersrand, Johannesburg. It has not been submitted before for any degree or examination in any other University.

John Kambanis

30th day of April, 1983

DECLARATION

I declare that this dissertation is my own, unaided work. It is being submitted for the degree of Master of Science in the University of the Witwatersrand, Johannesburg. It has not been submitted before for any degree or examination in any other University.

John Kambanis

30th day of April, 1981

ACKNOWLEDGEMENTS

I would like to convey my sincere appreciation to my supervisor, Mr D.B. Wilson, for his encouragement, especially at the initial stages of this work, and for his guidance through to the completion of this dissertation. I would like to thank Dr T.J. Wright for his valuable suggestions.

I would also like to thank my wife, my parents and my sister for their moral support and Mrs E. Schultz for undertaking the typing of this dissertation.

d.f.

CONTENTS

Page

INTRODUCTION 1

CHAPTER

1. SURVEY OF RELEVANT TOPICS 5

1.1 GENERAL SYSTEMS THEORY 5

1.1.1 The need for a General Systems Theory 5
1.1.2 Foundations of General Systems Theory 6
1.1.3 What is General Systems Theory 7
1.1.4 Philosophy 8
1.1.5 Isomorphies 10
1.1.6 Methodology 11
1.1.7 Axiomatic approach 17
1.1.8 An empirical approach 20

1.2 OTHER RELEVANT FIELDS 22

1.2.1 Dynamical Systems 22
1.2.2 Living systems 27
1.2.3 Organized systems and the second law of thermodynamics 32

2. BASIC CONCEPTS 34

2.1 APPROACH TO THE NATURE OF KNOWLEDGE 34

2.2 THE OBSERVATION SPACE 34

2.3 THE PERCEPT SET 35

2.4 THE RESOLUTION LEVEL 36

2.5 STATE OF AN ENTITY 38

2.6 INTERACTION 38

2.7 SCHEMATIC REPRESENTATION OF INTERACTIONS 41

2.8 INTERACTION NETWORKS 43

2.9 SIMPLIFIED DESCRIPTION OF NETWORKS 49

2.10 PROCESSES 50

2.11 CATALYSTS 51

2.12 INFORMATION 54

3. SELF-MAINTAINING ORGANISED SYSTEMS 60

3.1 INTRODUCTION 60

3.2 THE PHYSICAL ENTITIES TO BE CONSIDERED 60

3.3 BASIC CHARACTERISTICS OF SELF-MAINTAINING ORGANISED SYSTEMS 62

3.4 STABILITY OF IDENTIFYING VARIABLES 67

3.5 THE SELF-MAINTAINING CYCLIC PROCESS 67

3.6 THE ELABORATION AND ENHANCEMENT OF THE CYCLIC PROCESS 69

3.7 SPECIALISATION AND DIFFERENTIATION 70

4. ANALYSIS OF INTERNAL STRUCTURE AND PROCESSES 74

4.1 INTRODUCTION 74

4.2 INPUT PROCESSES AND INPUT GATES 75

4.3 THE TRANSPORT PROCESS - THE TRANSPORT SUBSYSTEM 77

4.4 STORAGE PROCESSES - STORAGE SUBSYSTEMS 78

4.5 METABOLISM - METABOLIC SUBSYSTEM 79

4.6 OUTPUT PROCESSES - OUTPUT GATES 86

4.7 INFORMATION PROCESSES - INFORMATION SUBSYSTEM .. 88

5. INTERACTION WITH THE ENVIRONMENT 92

5.1 INTRODUCTION 92

5.2 NEED SATISFACTION 92

5.3 SURVIVAL MARGINS 94

5.4 ENVIRONMENTAL CHANGES 96

5.5 A NOTE ON REPRODUCTION 96

5.6 THE ORGANISED ENVIRONMENT 97

5.7 MICRO-MEGA NETWORK RELATIONSHIPS 98

5.8 MICRO-MEGA NEEDS 99

5.9 RELATIONSHIP BETWEEN ORGANISATIONAL COMPLEXITY AND MICRO-MEGA NEED DIFFERENTIAL 100

6. MODELS OF INTERACTION NETWORKS 102

6.1 INTRODUCTION 102

6.2 MODELS 102

6.3 COMPUTER MODELS OF INTERACTION NETWORKS 103

6.4 THE BASIC ALGORITHM 103

6.5 STEPS FOR CONSTRUCTING A MODEL 105

6.6 ANALYSIS AT THE RIGHT RESOLUTION LEVEL 107

6.7 TEMPORAL ANALYSIS 109

6.8 INFORMATION AND THE MODEL DIAGRAM 111

6.9 THE STANDARD PROGRAM MODEL 113

7. A DETAILED MODEL 120

7.1 INTRODUCTION 121

7.2 THE OBJECT TO BE STUDIED 121

7.3 THE INTERACTION NETWORK AT A COARSE RESOLUTION LEVEL 122

7.4 ANALYSIS AT A FINER RESOLUTION LEVEL 124

7.5 INVENTORY 126

7.6 DISTRIBUTOR 127

7.7 PRICE 128

7.8 CONSUMER 130

7.9 ENVIRONMENT 132

7.10 PRODUCER 132

7.11 THE MODEL DIAGRAM 139

7.12 THE COMPUTER MODEL 139

7.13 CHECK OF RESULTS 144

7.14 CONCLUSIONS 145

8. MODELS OF SYSTEMS INVOLVING CONTINUOUS VARIABLES.... 147

8.1 INTRODUCTION 147

8.2 REPRESENTATION OF HEIMNETS' MODEL OF A CELL AS AN INTERACTION NETWORK 148

8.3 INTERACTION NETWORKS AND DIFFERENTIAL EQUATIONS. 162

8.4 RELATION OF INTERACTION NETWORKS AND NUMERICAL METHODS 166

8.5 CONSIDERATIONS OF STABILITY IN USING DIFFERENTIAL EQUATIONS AND NUMERICAL METHODS 173

8.5.1 Stiffness of a set of differential equations 175
8.5.2 Convergence of the numerical method 177
3.5.3 Numerical stability of the standard program model 179

8.6 USING A NUMERICAL METHOD 180

8.7 COMPARISON OF OUTPUT 182

9. CONCLUSIONS 187

APPENDIX A 190

APPENDIX B 193

APPENDIX C 201

APPENDIX D 209

REFERENCES 218

BIBLIOGRAPHY 220

LIST OF ILLUSTRATIONS

Page

1.1 "Production-possibility" curves 9
1.2 Activity of the system 12
1.3 Typical modes of representing ST-structure 14
1.4 Typical mode of representing UC-structure 15
1.5 Relation of general systems theory to other disciplines 16
1.6 Hierarchical, multilevel, goal-seeking system 19
1.7 Process of model building 20
1.8 Graph of population of species 27

2.1 Time slices 34
2.2 Spatiotemporal object 35
2.3 Perception mapping 36
2.4 Spatial resolution 37
2.5 Temporal resolution 37
2.6 State of an entity 38
2.7 Three time slices of an interaction 39
2.8 Example of interaction 40
2.9 Example of interactions 42
2.10 Schematic representation of an interaction 43
2.11 Examples of schematic representation of interactions 44
2.12 A factory as a spatiotemporal object 45
2.13a Department A as an interaction 45
2.13b Department B as an interaction 46
2.14 Department D as an interaction 46
2.15 Example of interaction network 47
2.16 Interaction network for chemical reactions 48
2.17 Simplified diagram of interaction network 49
2.18 Interaction network example 49
2.19 Synthetic interactions 50
2.20 Synthetic interaction network 51
2.21 Interaction involving a catalyst 51

2.22 Examples of catalysts 52
2.23 The attribute of "catalyst" is relative 53
2.24 Difference in information states 55
2.25 Information as a change of one object to another . 56
2.26 Information channel 57
2.27 Information transfer by transportation 58
2.28 Information transfer by transmission 59

3.1 Interaction with the environment 64
3.2 State spaces of entity and environment 65
3.3 Survival by continuous adjustment 66
3.4 Self-maintaining cyclic process 67

4.1 Input gate (model) 75
4.2 Input gate (cell) 76
4.3 Transport subsystem (model) 77
4.4 Storage subsystem 78
4.5 Water reservoirs 78
4.6 Metabolism (general model) 80
4.7 Metabolism (organism) 81
4.8 Metabolism (society) 82
4.9 Money as a measure of activity 85
4.10 Output gates (model) 86
4.11 Output gates (cell) 87
4.12 Output gates (organisms) 87
4.13 Output gates (society) 88
4.14 Internal communication in cells (biochemistry) ... 89
4.15 Internal communication in cells (as an interaction network) 90
5.1 Response to the environment 96
5.2 Interaction with entities of an organised environment 98

6.1 Report on the changes of states 104
6.2 Interaction as a sequence of time slices 104
6.3 Interaction affected by a number of entities 104
6.4 Entity affected by a number of interactions 105
6.5 General algorithm of the interaction network model 106
6.6 Interaction network at a coarse resolution level .. 107
6.7 Interaction network at a finer resolution level ... 108

6.8 Omitting of redundant information in interaction networks 109
6.9 Projections of the spacetemporal object 110
6.10 The operation aspect of the network 111
6.11 Example of a model diagram 112
6.12 An interaction network for a system of chemical reactions 116
6.13 Model diagram of the chemical reactions 116
6.14 Graph of the behaviour of the system 120

7.1 producer-distributor-consumer network at a coarse resolution level 122
7.2 The producer-distributor-consumer interaction network at the optimum resolution level 125
7.3 Illustration of ECAC 128
7.4 Illustration of DCOV 129
7.5 PRICE as a function of DCOV 130
7.6 EPCC as a function of PRICE 131
7.7 EPCA' as a function of PRICE 131
7.8 The producer-distributor-consumer model diagram ... 140
7.9 An aspect of producer-distributor-consumer 144
7.10 DYNAMO model statements 145

8.1 Interaction network representing the cell 154
8.2 Temporal analysis 156
8.3 Interaction network for chemical reaction 167
8.4 Interaction network of component assembly 168
8.5 Interaction network for component assembly and chemical reaction 168
8.6 Program model for component assembly and chemical reactions 169
8.7 Communication and control structure of Runge-Kutta method 172
8.8 Communication and control structure
8.9 Graph of difference in predicted behaviour between standard program model and Runge-Kutta 184
8.10 Graph of processing time vs. DT for the program model 186

6.8 Omitting of redundant information in interaction networks 109
6.9 Projections of the spatiotemporal object 110
6.10 The space/time aspect of the network 111
6.11 Example of a model diagram 112
6.12 An interaction network for a system of chemical reactions 117
6.13 Model diagram of the chemical reactions 121
6.14 Graph of the behaviour of the system 122

7.1 A producer-distributor-consumer network at coarse resolution level 144
7.2 The producer-distributor-consumer interaction network at the optimum resolution level 145
7.3 Illustration of ECAB 148
7.4 Illustration of DCOV 150
7.5 PRICE as a function of DCOV 151
7.6 EPCC as a function of PRICE 152
7.7 DPCAP as a function of PRICE 153
7.8 The producer-distributor-consumer model diagram 154
7.9 An aspect of producer-distributor-consumer 161
7.10 DYNAMO model statements 167

8.1 Interaction network representing the cell 180
8.2 Temporal analysis 190
8.3 Interaction network for chemical reaction 193
8.4 Interaction network of component assembly 195
8.5 Interaction network for component assembly and chemical reaction 196
8.6 Program model for component assembly and chemical reactions 199
8.7 Communication and control structure of Runge-Kutta method 211
8.8 Communication and control structure
8.9 Graph of difference in predicted behaviour between standard program model and Runge-Kutta 244
8.10 Graph of processing time vs. DT for the progress model 250

INTRODUCTION

This dissertation is an investigation into the structure and function of organised systems, with special interest in those possessing the property of self-maintenance.

If entities such as atoms, molecules, cells, organisms and societies are considered, the following two fundamental attributes can be distinguished.

Firstly they possess a self-maintained stability of characteristics. Despite changes in the environment these entities preserve their basic structure and behaviour. For example a molecule of carbon dioxide preserves its structure despite deformities caused by collisions with other molecules.

Secondly they are organised. If their internal details are observed and identified they are found to be spatially and temporally structured; i.e. they have a specific anatomy consisting of identifiable components which change, move, and interact with each other in a specific way.

The question now arises: do these entities have anything in common? Can a model of the general self-maintaining organised system be formulated?

This dissertation will investigate the existence of common characteristics in physical systems such as cells, organisms and societies. Each has been studied by the relevant specialised science which has analysed the entity into its components and has constructed a model which represents its internal organisation and explains the stability of

its characteristics and behaviour.

Despite the differences of opinion on phenomena not yet fully investigated, there is a well-recognised body of knowledge in each of the specialised fields studying these entities, which constitutes a model representing the relevant entity as we now understand it. The model of a cell for example is the collection of the generally accepted constructs which describe a cell with its nucleus membrane enzymes etc. which function together as an organised system surviving in a changing environment. Similarly with other entities such as an organism or an entire country, with its government, industry, economic subsystem etc.

This investigation aims firstly at showing that there are a large number of common characteristics in all these entities which display behaviour and structure which one would characterise as organised and able to survive in a hostile environment.

When a large class of such important systems has a significant number of characteristics in common, it is (at least) economically justifiable to study these characteristics separately as a field in its own right. This avoids duplication of effort since the results obtained by studies in such a field are applicable to several areas without having to be re-invented every time. This is the premise of the General Systems approach which will be adopted here and explained in more detail later. It is therefore the first aim of this dissertation to show that such an approach is justifiable in the study of these class of systems and that it is useful to have a field which studies "organisation for the purpose of survival" as a subject in its own right, separate from the specific physical forms which it takes in nature.

Secondly this dissertation aims at presenting a language, a form of representation, and a medium for modelling of such entities or parts of such entities. This is necessary because the problem in comparing knowledge on such diverse phenomena is the different terminology of the special sciences and the different way in which specialised knowledge is classified. To overcome this problem a basic set of concepts will have to be first established which provides the appropriate framework for analysing and classifying observations. This framework having been established provides a common language into which the various specialised observations, concepts or models can be translated. If it is thus possible for diverse phenomena to be described by one form of representation then it should be feasible for a field of study as described above to function and produce useful results.

Thus this dissertation can be seen as a methodological investigation. It defines a field which consists of a collection of diverse specialised knowledge and methods of study as well as speculations and loose analogies and attempts to establish its validity as well as present a basic set of representational tools which this field can use in order to proceed on a scientific basis.

Such an investigation I believe belongs in the same class as mathematics, computer and information science in their broadest sense. All these disciplines are concerned with developing systems of symbols of the world which are used as a tool for representing the world of observation and ultimately providing information which leads to man's better response to the environment.

This study will proceed as follows:

Chapter 1: A survey of related fields. The general systems approach as well as other areas of work which are relevant to this investigation will be presented.

Chapter 2: The conceptual and representational tools will be suggested and developed.

Chapter 3: Using the above approach and tools, the fundamental characteristics of self-maintaining systems will be described.

Chapter 4: The internal structures of three such systems (cells, organisms, societies) will be compared to demonstrate that there is a general scheme which applies to all.

Chapter 5: The same systems will be investigated from the point of view of their interaction with their environment.

Chapter 6: The technique of computer modelling which results it will be presented.

Chapter 7: A detailed analysis and modelling using the described approach will be given.

Chapter 8: A special consideration for systems of continuous variables, and a second example of the application of the described approach will be given. The proposed methodology will be related to modeling by means of differential equations and using standard numerical methods.

Chapter 9: Conclusions.

CHAPTER 1

SURVEY OF RELEVANT TOPICS

The subject matter of this dissertation is interdisciplinary. Since it is concerned with building models of phenomena which are normally studied by different fields, there is a wide range of topics in the literature which are in some way related to this work. A review will be given in this chapter of the most important areas of study which are related to this work and from which information is drawn.

1.1 GENERAL SYSTEMS THEORY

The relationship of General Systems Theory to this dissertation is that it provides the philosophical background and the basic approach of study.

The following survey explores the need for a General Systems Theory, its purpose, foundations, and the main schools of thought and work which exist at the moment. While surveying the field, emphasis will be placed on the areas which are most relevant to this dissertation.

1.1.1 The need for a General System Theory

The past centuries of scientific endeavour have brought science to a certain level of maturity where a great store of specialised knowledge is available on most things that affect human life. Now we are entering a phase where complexity in itself becomes a real enough challenge.

In Weinburg's[13] words: "science and engineering have been catalysts for the unprecedented speed and magnitude of change. The physicist shows us how to harness the power of the nucleus; the chemist shows us how to increase the quantity of our food; the geneticist shows us how to improve the quality of our children. But science and engineering have been unable to keep pace with the second-order effects produced by their victories. The excess heat from the nuclear generator alters the spawning pattern of fish, and before adjustments can be made, other species have produced irreversible changes in the ecology of the river and its borders. The pesticide eliminates one insect only to the advantage of others that may be worse, or the herbicide clears the rain forest for farming, but the resulting soil changes make the land less productive than it was before." Looking at these problems in a positive and constructive way, he concludes that: "...the general systems approach is needed because science has been such a success. Science and technology have colonized the planet, and nothing in our lives is untouched. In this change, they have revealed a complexity with which they are not prepared to deal. The general systems movement has taken up the task of helping scientists to unravel complexity, technologists to master it, and others to learn to live with it..."

1.1.2 Foundations of General Systems Theory

The concept of a system is not new. Aristotle's statement: "the whole is more than the sum of its parts" indicates that the basic concept of "system" has been appreciated for a long time: a collection of interrelated, interacting entities exhibits properties which go beyond the straight forward summation of the properties of the components.

However General Systems theory as a field which studies systems in their own right, independent of the particular

In Weinburg's[15] words: "science and engineering have been catalysts for the unprecedented speed and magnitude of change. The physicist shows us how to harness the power of the nucleus; the chemist shows us how to increase the quantity of our food; the geneticist shows us how to improve the quality of our children. But science and engineering have been unable to keep pace with the second-order effects produced by their victories. The excess heat from the nuclear generator alters the spawning pattern of fish, and before adjustments can be made, other species have produced irreversible changes in the ecology of the river and its borders. The pesticide eliminates one insect only to the advantage of others that may be worse, or the herbicide clears the rain forest for farming, but the resulting soil changes make the land less productive than it was before." Looking at these problems in a positive and constructive way, he concludes that: "...the general systems approach is needed because science has been such a success. Science and technology have colonized the planet, and nothing in our lives is untouched. In this change, they have revealed a complexity with which they are not prepared to deal. The general systems movement has taken up the task of helping scientists unravel complexity, technologists to master it, here to learn to live with it..."

1.1.2 Foundations of General Systems Theory

The concept of a system is not new. Aristotle's statement: "the whole is more than the sum of its parts" indicates that the basic concept of "system" has been appreciated for a long time: a collection of interrelated, interacting entities exhibits properties which go beyond the straight forward summation of the properties of the components.

However General Systems theory as a field which studies systems in their own right, independent of the particular

nature of the components, is a relatively new field. It was introduced in the 1940's by Von Bertalanffy. It was established and publicized later (in 1954) with the formation of the "Society for General Systems Research". It's program is:

(1) to investigate the isomorphy of concepts, laws, and models in various fields, and to help in useful transfers from one field to another;

(2) to encourage the development of adequate theoretical models in fields which lack them;

(3) to minimize duplication of theoretical effort in different fields;

(4) to promote the unity of science through improving communications among specialists.

1.1.3 What is General Systems Theory

Knowledge can be regarded as the construction of conceptual models, which result in economy of thought. By combining a large set of observations into a conceptual model of "reality" it is possible thereafter to use that model in responding to a great variety of situations.

The different disciplines of science study phenomena which belong to certain broad classes: mechanics, organic chemistry, atomic physics, etc. Models (or theories) are built (such as the atom, molecules, the cell, the theory of relativity) which result in economy of thought and enable one to understand a (theoretically) infinite number of individual phenomena and respond to them.

General Systems Theory is concerned with building models which are applicable to several disciplines because they cover phenomena which are normally classified in different classes. The main benefit from this is (as in all knowledge) economy of thought. Information incorporated in a general systems model which was obtained from phenomena of one

discipline may be used to answer questions posed by phenomena of another discipline.

Towards the above purpose General Systems Theory enlists different types of knowledge of other fields, and attempts to integrate them into one tool.

Since this field is still young and developing, the integration of the contributing components to the theory has not matured yet. As a result General Systems Theory consists at the moment of a collection of principles, concepts, problems, methods, tools, some adapted from other disciplines and some developed within this field [Klir [4]]. This diversity of components in the theory, however, although not yet fully integrated and formalised with a specific terminology, does not detract from the clarity of purpose of the field which is as given above. A good indication of the diversity and broadness of scope of General Systems Theory can be seen by reference to the contributions in the General Systems Yearbooks [12]. Integration of General Systems Theory into such a unified field is in fact an important trend at present.

In the following pages the most important aspects and the main currents of thought on General Systems Theory which are relevant to this dissertation, will be presented.

1.2.4 Philosophy

Above all, General Systems Theory is a way of looking at the world. Weinburg [13] takes the view that General Systems training will equip the specialist with a way of looking at his subject which will improve the ease and completeness of his understanding.

Suppose one, with some training in General Systems Theory, is studying economics, and he is faced with the "production-

possibility" curves as in fig. 1.1

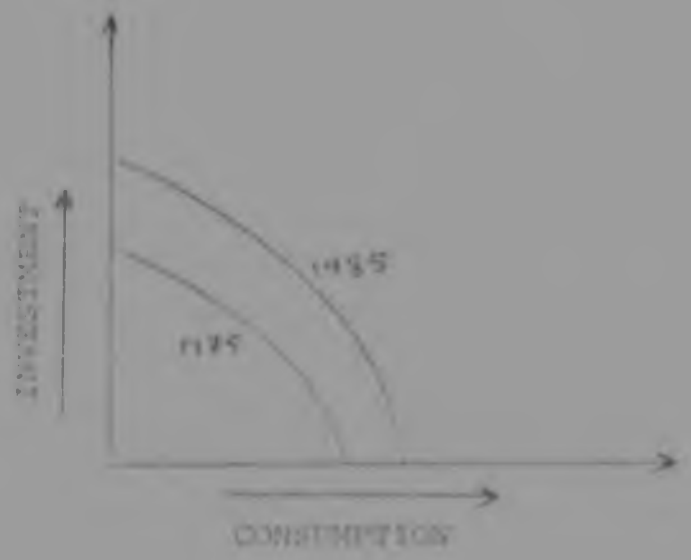


FIG. 1.1. "Production-possibility" curves

"Almost without explanation, he recognises these as special cases of a more general representation, the state space. What the economist calls 'the production-possibility frontier' the generalist recognises as a family of systems all possessing certain property, and any point on that curve representing a particular system within that family. He recognises that the movement from one point to another in this state-space is what he calls a line of behaviour, so that all he knows about such lines may immediately be transferred to this situation ... He has special words in his vocabulary, words such as stability, behaviour, state space, structure, regulation, noise and adaptation which he can relate to the words of the specialist ... When the generalist encounters laws in the special field he will often be able to relate them to the general system "laws" he knows. He identifies the special assumptions that have made general systems laws into laws of economics of whatever. For instance he will immediately recognise the economist's Law of Diminishing Returns as a special

case of the Law of Limiting Factors" (Weinberg [19, p.44-45]).

1.1.3 Isomorphism

General systems theory [illegible] models, each applying to a wide class of phenomena. At the foundation of this characteristic is the concept of mathematical isomorphism as presented by Bangert in [illegible].

A mathematical system can be defined as a set of elements and the relations between them. Two mathematical systems are isomorphic if it is possible to establish a one-to-one correspondence between their elements and if the relations defined [illegible] elements hold true for the corresponding elements [illegible].

This mathematical isomorphism induces a conceptual isomorphism. Two concrete systems are conceptually isomorphic if they are represented by isomorphic mathematical systems, i.e. if they can be represented by the same mathematical model.

This leads to the classification of systems which provides a logical advantage. The classification of concrete systems becomes the image of the classification of mathematical models. But when a mathematical system can be seen as a special case of another (more general one), this relationship can be induced to the corresponding concrete systems, and thus their classification displays one class as including another.

The practical significance of the above was first understood as various kinds of system similarities (e.g. geometric, kinematic, thermodynamic). Two systems were considered similar if their variables were of the same physical nature and their values were proportional at corresponding times. Later this was extended to systems analogy. Two systems are analogous if their variables are

of different physical nature but [illegible] values were proportional at corresponding [illegible]. Analogy is based on the similarity of the algebraic or differential equations describing the system. [illegible] is the analogy of a mechanical harm[illegible] with inductance, [illegible].

1.1.6 Methodology

Klir's approach to G[illegible] a great extent on meth[illegible]

Klir emphasises that it [illegible] theories, each one depending on [illegible] He did a thorough investigation [illegible] "system" given in [14, appendix I] [illegible] "system" are chosen in such a way as to allow general applicability (ability to be used in all disciplines). Only the systems theory based on [illegible] have general applicability. Consequently, Klir undertook to define precisely a set of [illegible] applicable to all disciplines.

He starts from the basic epistemological [illegible] of a phenomenon (or object) we cannot know its complexity in either its full simplicity or [illegible]. What we do is the following: We measure values of certain quantities associated with certain attributes of the object in terms of a space-time reference frame. The values may be non-numerical, as long as they are meaningful in terms of the accuracy and frequency of measurement in the [illegible] level. As a result we obtain the activity of the system (fig. 1.2).

By observing this activity we attempt to understand the [illegible] invariant relations existing [illegible].

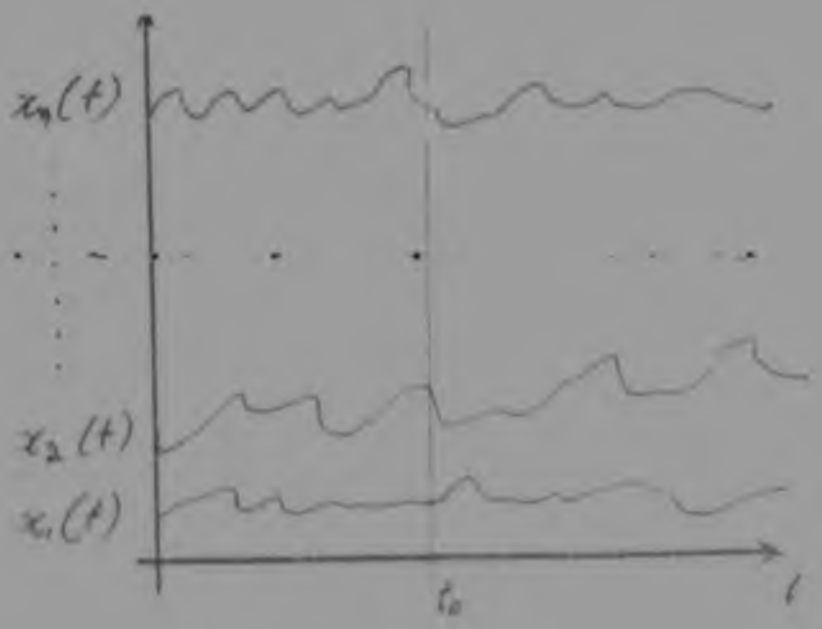


FIG. 1.2. Activity of the system

The process above is termed "defining a system or the object from a distinct point of view". Once a system is so defined the object of the study becomes the system, not the object itself.

Obviously there are many ways of choosing the quantities, recording the activity and defining the system as was done above. Therefore "defining a system on the object" is defining a particular point of view. These vary from one problem to the other and from one discipline to the other.

Are there any fundamental traits of systems in terms of which any point of view can be defined? Klir thinks there are: Experimental branches of science define a system on the object in terms of:

(i) the set of quantities measured;
(ii) the resolution level;
(iii) the time invariant relations between quantities;
(iv) the properties that determine the relations.

The next step is to formulate definitions based on these fundamental traits alone. The following five definitions form the backbone of Flü's approach.

Definition 1: The set of external quantities and the resolution level

A system S is defined by a given set of quantities regarded at a given resolution level.

Definition 2: Activity

The system S is defined by a given ensemble of variations in time of some quantities under consideration.

Definition 3: The permanent behaviour

The system S is defined by a given time invariant relation between instantaneous and/or past and/or future values of the external quantities; every element of the relation may (but need not) be associated with a probability of its occurrence.

Definition 4: The real UC-structure (Structure of the Universe and Couplings)

The system S is defined by a given set of elements together with their permanent behaviours, and a set of couplings between the elements on the one hand and between the elements and environment on the other.

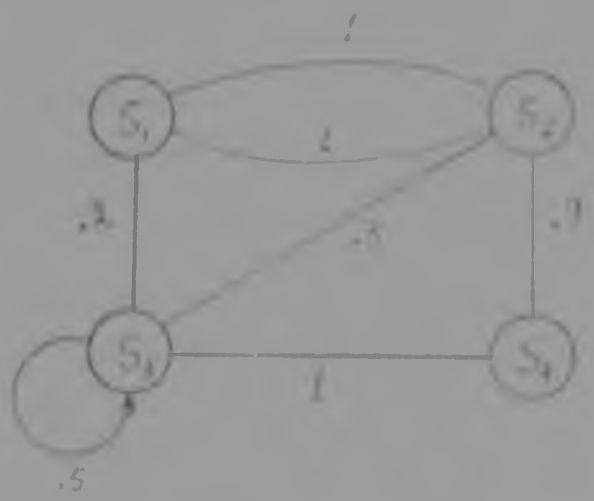
Definition 5: The real ST-structure (State-Transition Structure)

The system S is defined by a given set of states together with a set of transitions between the states; every transition may, but need not, be associated with a probability of its occurrence.

The above definitions have also been formulated mathematically [14, p.55].

These definitions can now be used for defining a system on an object under study. Klir illustrates this by many examples [14] on objects purposely drawn from diverse fields such as bacterial metabolism, a heating system, a musical composition, a traffic light system, a legal system, birth control, a potted rosebush, a model of the instinct of self-preservation, and other.

Some modes of systems representations which he uses are shown in fig. 1.3 and fig. 1.4 below.



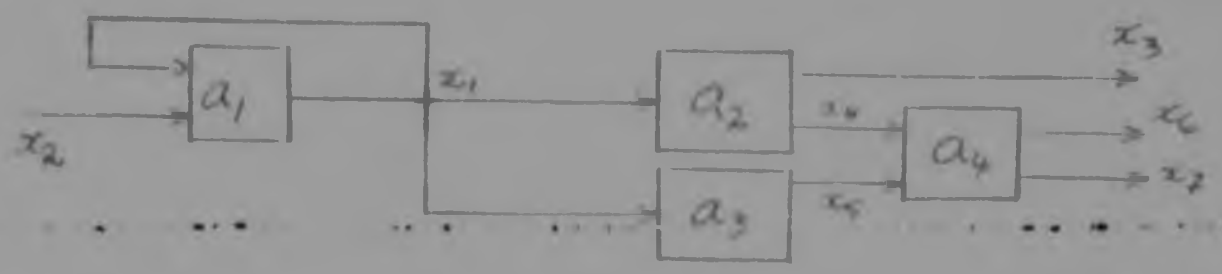
·	S_1	S_2	S_3	S_4
S_1	0	1	0	0
S_2	.1	0	0	.9
S_3	.2	.3	.5	0
S_4	0	0	1	0

w/probabilities

·	S_1	S_2	S_3	S_4
S_1	0	1	0	0
S_2	1	0	0	1
S_3	1	1	1	0
S_4	0	0	1	0

w/o probabilities

FIG. 1.3. Typical modes of representing ST-structure



a_1, a_2, a_3, a_4 are relations

x_1, ..., x_n are variables

FIG. 1.4. Typical mode of representing UC-structure

In specialised disciplines, the basic problem which we try to solve is the following: Some traits of an object (or phenomenon) are given and we are required to define other traits.

Equipped with the above definitions General Systems Theory would proceed to solve the above problem as shown in fig. 1.5.

By defining the object of a specialised discipline in terms of General Systems Theoretical definition, it enables the investigator to draw from the body of general systems knowledge and thus produce information about the object which was not hitherto available.

The problems which General Systems Theory can attempt to solve are classified in three categories, according to Klir[14]: systems analysis, systems synthesis and systems investigation.

Analysis is the process of determining the behaviour of ST-structure of the system, given its UC-structure (i.e. given the system components and their interrelations

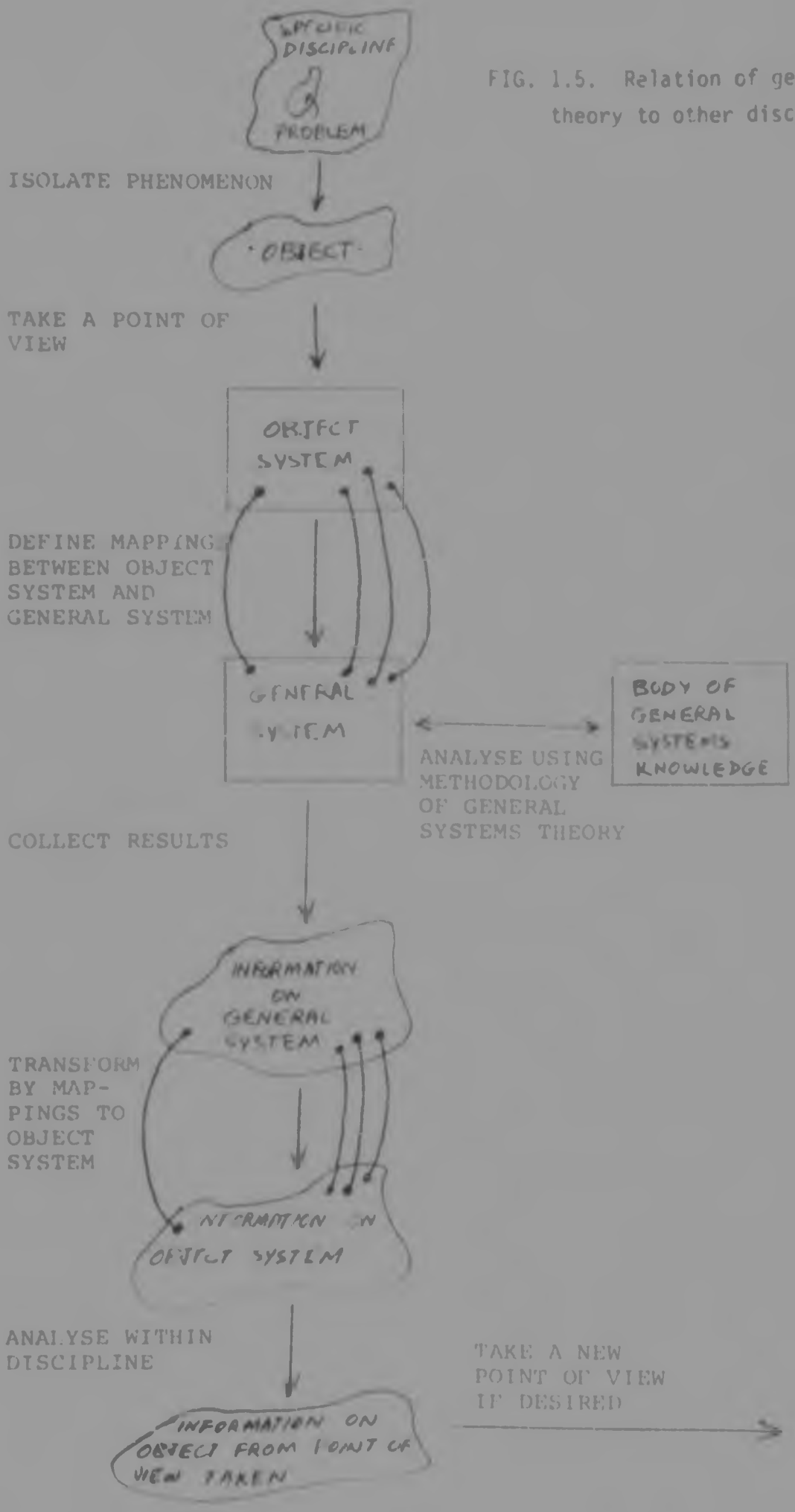


FIG. 1.5. Relation of general systems theory to other disciplines.

derive the system behaviour).

Synthesis is the process of finding a UC-structure which realises a given ST-structure or permanent behaviour or activity (i.e. given the specifications of the performance of (e.g.) a machine, derive a design for the construction of such a machine).

Investigation (or the black box problem) is the process of acquiring knowledge of the system given its definition as quantities at a given resolution level.

Klir elaborates on the applications of general systems approach on these three classes of problems and detailed examples can be found in [14, chapters 7,8,9].

1.1.7 Axiomatic approach

Another school of thought, which can be termed axiomatic, attempts to establish precise rigorous mathematical foundations. Examples of this approach are seen in the approaches of Mesarovic, Takahara[10] and Wymore[8].

Mesarovic starts with the definition of a general system as a relation on abstract sets

$$S = X(V_1 \times V_2 \times \ldots \times V_n)$$

where V_i is one of the systems objects (components) and it is the set of all alternative ways in which the particular object is experienced (or in different terminology, it is the set of all possible values of a certain variable of the system).

From this definition he proceeds along the path of formalisation, from the more general to the specific, along two approaches: the input-output approach and the goal-seeking approach.

In the input-output approach the objects are grouped into two categories:

inputs(stimuli) $X = \{(V_i : i \in I_x)\}$

and outputs(responses) $Y = \{(V_i : i \in I_y)\}$

where $\{I_x, I_y\}$ is a partition of I. The system S is then a relation on inputs and outputs $S \subset X \times Y$.

In the goal-seeking approach (which is of more interest here) the above partition is also defined and then the system described in terms of a goal-seeking process:

A decision object M and a value object V such that every subset has a minimum are introduced as well as two functions.

the outcome(process) function $P: X \times M \to Y$
the performance function $G: M \times Y \to V$

then a system $S \subset X \times Y$ can be defined:

$\forall x \in X$ and $\forall y \in Y$, $(x,y) \in S$

if $\exists\ m_x \in M$ such that $\forall m \in M$

$G[m_x, P(x,m_x)] \leq G[m,P(x,m)]$

and $y = P(x,m_x)$. The goal of the system is then minimisation of G.

A system problem which is of special interest to the Mesarovic group is the behaviour and structure of hierarchical, multilevel, goal-seeking systems. A typical example of such a system is the organisation chart of a company. Mesarovic formulates this type of problem as follows (fig. 1.6):

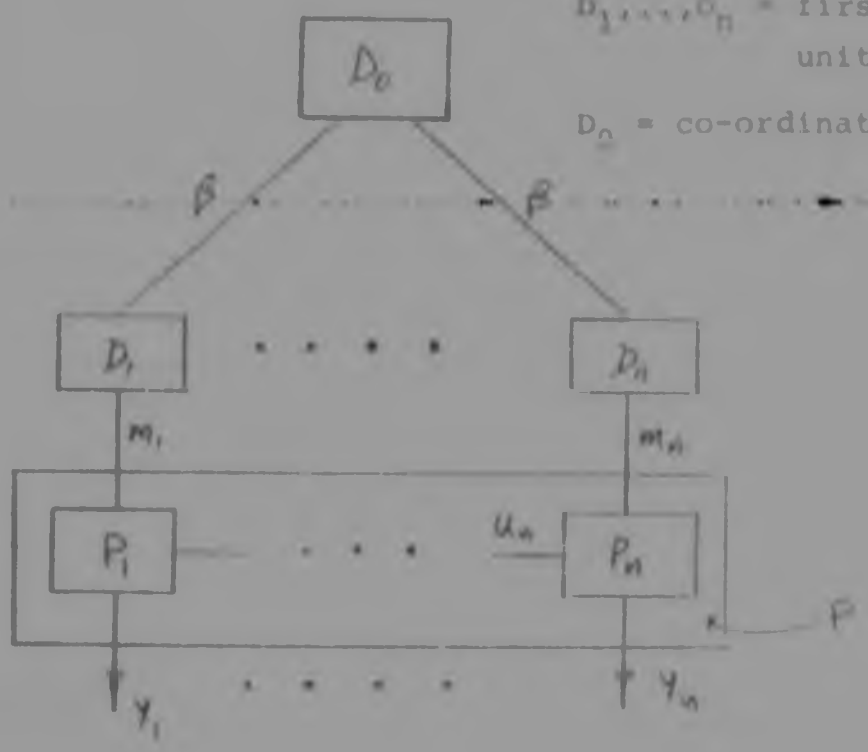


FIG. 1.6. Hierarchical, multilevel, goal-seeking system

The basic subsystems of the above diagram are the following. A controlled process, $P:M \cdot Y$ is decomposed into n interacting subprocesses, $P_i:M_i \times U_i \to Y_i$ where $M = M_1 \times \ldots \times M_n$ are controls, $Y = Y_1 \times \ldots \times Y_n$ are outputs, and $U = U_1 \times \ldots \times U_n$ are the interactions which can be expressed in terms of controls by interaction functions, $K_i:M \to U_i$. There are n first-level control subsystems defined by the respective subprocess models $P_i:M_i \times U_i \to Y_i$ and a local performance function, $G_i:M_i \times Y_i \times |\beta| \to V$, where β is the co-ordination term while V is the value set. Finally, there is a second-level subsystem, termed a co-ordinator whose task is to select the co-ordination term β so that the first-level units, while being concerned with their local control problems, will

actually satisfy the overall control objective for the total system defined by the total process, P:M → Y, and an overall performance function G:M × Y → V.

Starting with the above Mesarovic derives many theorems for a wide range of specific aspects of systems as given in [5] and [16] thus establishing a precise rigorous and formal foundation for systems theory.

Mesarovic sees the position of general systems theory in the process of systems analysis and model building as given in fig. 1.7 [5, p.4].

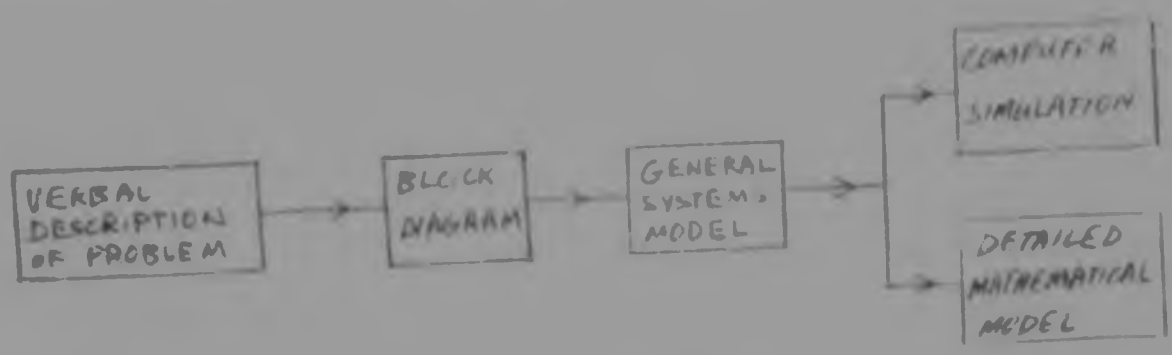


FIG. 1.7. Process of model building

In building a model of the system one moves from the more general (high level of abstraction) to the detailed level by stepwise moving to more specific representations. The initial verbal description of the system is substituted by a block diagram where the overall composition of the system is revealed by showing the components and their interrelativity. Then a general systems theory model introduces precision by mathematical formalisation. Finally the detailed mathematical model or computer simulation model is constructed, based on the general systems model.

1.1.8 An empirical approach

Now to come closer to some of the real phenomena that General Systems Theory is meant to investigate, von Bertalanffy's

approach is presented.

Since General Systems Theory does not study the nature of the specific events which constitute a phenomenon but is rather interested in the pattern of the interrelationships of components, it is not a surprise that this field was introduced by Von Bertalanffy, a biologist.

In his own words: "since the fundamental character of the living thing is its organisation, the customary investigation of the single parts and processes cannot provide a complete explanation of vital phenomena. This investigation gives us no information about the co-ordination of parts and processes."[1] It was this need for a theory which can help study organisation as such which prompted the development of General Systems Theory.

Von Bertalanffy's approach to the application of General Systems theory is essentially empirical. On the basis of experimental results, isomorphic relations between different systems are established and studied. Finally general systems principles applicable to all systems of that class are formulated.

Von Bertalanffy uses the concept of an open system to study complex interrelationships between a set of elements within the system as well as between system and environment.

An open system is one which exchanges material and information with its environment [1]. In a living organism for example there is a continuous inflow and outflow of material, a simultaneous breaking and building of components, thus maintaining itself in a steady state.

Based on the above, Von Bertalanffy goes on to study the main features of organisms. He borrows the classical representation of dynamical systems by means of differential equations,

$$\frac{dx}{dt} = f_i(x_1, x_2, \ldots, x_n) \qquad (i=1,2,\ldots,n)$$

(this will be presented in more detail later in this chapter), we proceed to illustrate how some of the hitherto elusive characteristics of organisms can be formulated and studied. Thus in 11, chapter 11 the investigates stability, growth, competition, wholeness, [illegible], regularity, finality.

1.2 OTHER RELEVANT FIELDS

The exposition above has examined General Systems Theory from various viewpoints. The purpose of this was to show that it is possible and justifiable to study objects from various different disciplines and attempt to represent them by the same model.

The purpose of this dissertation is to present an approach [illegible] of self-maintaining organised systems. General Systems Theory contains the background and the justification for such an approach. We will proceed [illegible] to examine some other relevant fields of work, which are of different scope and more specific to our topic.

1.2.1 Dynamical Systems

Dynamical system is a system [illegible] variables change in time.

Classical approach for the study of dynamical systems for the case of continuous variables is a representation of a system by a set of differential equations.

[illegible] uses this method for the study of systems in [illegible]. Although used specifically in biology it is of [illegible] general interest for this investigation since the [illegible] can be used for socio-economic phenomena.

A representation of a dynamical system has to achieve two things: Firstly it must give an instantaneous description of the system at a particular point in time, and secondly it must represent the manner in which these instantaneous descriptions change in time.

Each of the quantities observed is a state variable and since they change in time, they can be represented by

$$x_i(t), \ldots, x_n(t)$$

In describing a dynamical system we note that the rate of change of a particular state variable in the vicinity of a given state $(x_1, \ldots, x_n)$ depends only on that state. Thus the condition which specifies the dynamical behaviour of the [illegible] is

$$\frac{dx_i}{dt} = f_i(x_1, \ldots, x_n) \quad (i = 1, \ldots, n)$$

The fundamental problem of dynamical system theory is: given the above equation (1) as a condition describing the system, what can be said about the dynamical properties of the above system (1), (2).

Rosen uses the dynamical system framework to investigate the important problems of biology, such as stability, regulation and control. Since a biological system is perhaps the most obvious example of the class of system which this dissertation will show, Rosen's work is of interest here.

Rosen sees stability as the most obvious concept in the study of dynamical systems (9, p.111). The general idea of stability is defined as follows: let

$$\frac{dx_i}{dt} = f_i(x_1, \ldots, x_n) \quad (i = 1, \ldots, n)$$

be the equations of a dynamical system defined in some open region Ω of the partition n-dimensional space.

Assuming this system possesses the unique trajectory $\bar{C}$ property (as defined in [9]), let

$c(t) = \{x_1(t),\ldots,x_n(t)\}$ be a trajectory and let

$c(0) = \{x_1(0),\ldots,x_n(0)\}$ be the state point of the

system at $t=0$. We say that:

(a) $c(t)$ is bounded (on the right) if, for any $\delta > 0$, $\exists$ a positive number M such that

$$|x_i(0) - \bar{x}_i(0)| < \delta \Rightarrow |x_i(t) - \bar{x}_i(t)| < M$$

for all $t \geq 0$ and all $i=1,\ldots,n$

(b) $c(t)$ is weakly stable (on the right) if given any $\varepsilon > 0$, $\exists\ \delta > 0$ such that

$$|x_i(0) - \bar{x}_i(0)| < \delta \Rightarrow |x_i(t) - \bar{x}_i(t)| < \varepsilon$$

for all $t \geq 0$, $i=1,\ldots,n$

(c) $c(t)$ is asymptotically stable (on the right) if $\exists$ a $\delta > 0$ such that

$$|x_i(0) - \bar{x}_i(0)| < \delta \Rightarrow \lim_{t\to+\infty} |x_i(t) - \bar{x}_i(t)| = 0$$

$(i=1,\ldots,n)$

(d) $c(t)$ is unstable (on the right) if $\forall\delta > 0$, a trajectory $\bar{C}(t)$ such that

$$|x_i(0) - \bar{x}_i(0)| < \delta \quad \text{and} \quad \lim_{t\to+\infty} |x_i(t) - \bar{x}_i(t)| \to +\infty$$

for at least one i, $(i=1,\ldots,n)$

The above could be termed stability of behaviour. Another stability can also be considered: that of a stable state: Consider the system $\frac{dx_i}{dt} = f_i(x_1,\ldots,x_n)$ and the traject-

ory defined by the solutions $x_i(t) = C_i$ where C_i = constant $(i=1,\ldots,n)$. Obviously since the rates of change of the variables are zero, once the system arrives at this state it will stay in this state forever. This is the critical point of the system.

The next question is that of stability of critical points: When a system reaches such a state, will it return to it after a perturbation?

A critical point of the system is just a special trajectory (where all its variables are constant); therefore the definitions of stability (a), (b), (c), (d) formulated above are the same here, with the additional special condition $x_i(t) = C_i$ $(i=1,\ldots,n)$.

Rosen explains some powerful criteria (enunciated by Lyaponov) for determining whether a critical point of a dynamical system is stable or not [9, p. 10].

In returning the perturbed values of the state variables to the critical point a system acts as if it were opposing an attempt to push it out of that state. In other words such a system has regulatory properties in appropriate neighbourhoods of any of its asymptotically stable critical points.

What happens if instead of simply perturbing the values of the variables we modify the structure of the system itself; in other words modify the equations of motion themselves?

Rosen gives an example of an oscillator and perturbations of its dumping parameters. Some of his conclusions are that a dynamical system can display particular topological characteristics of trajectories for each specific structure, and a system is structurally stable if the topology of its trajectories is preserved under sufficiently small perturbations of its structure.

Dynamical systems can also be conditionally structurally stable, i.e. it is structurally stable with respect to certain variables and unstable with respect to others. Thus real physical systems may be structurally unstable with respect to certain variables without this being catastrophic, since they are structurally stable with respect to the variables of their natural operating environment.

An example of a dynamical system: Volterra uses this dynamical representation to study ecosystems[11].

In the case of two species one feeding on the other, N_1 and N_2 can represent the number of individuals in the two species. If the second species were not present, the first would grow at a rate of $\gamma_1 > 0$. Also assume that the second species would die out for lack of food if the first were not present so that its death would be $-\gamma_2$. Thus for each species alone we would have

$$\frac{dN_1}{dt} = \gamma_1 N_1 \quad \text{and} \quad \frac{dN_2}{dt} = -\gamma_2 N_2 .$$

But if they are together, species 2 feeds on species 1. Let γ_1 measure the susceptibility of species 1 to predation and that the effects of predation are proportional to the numbers N_2 of the predator species. Similarly γ_2 represents the predatory ability of species 2. We can then represent the system of the two interacting species by

$$\left.\begin{aligned} \frac{dN_1}{dt} &= \gamma_1 N_1 - \gamma_1 N_2 N_1 \\ \frac{dN_2}{dt} &= -\gamma_2 N_2 + \gamma_2 N_1 N_2 \end{aligned}\right\}$$

On the basis of this Volterra derived some interesting characteristics of this system: for example the popu-

lations of the species fluctuates periodically and the period depends on the rates of growth and decline and initial population sizes [11, p. 92] as shown in fig. 1.8, and the average population of each species is independent of its initial value. This system responds in specific ways to outside interference as analysed by Voltera[11;p. 93].

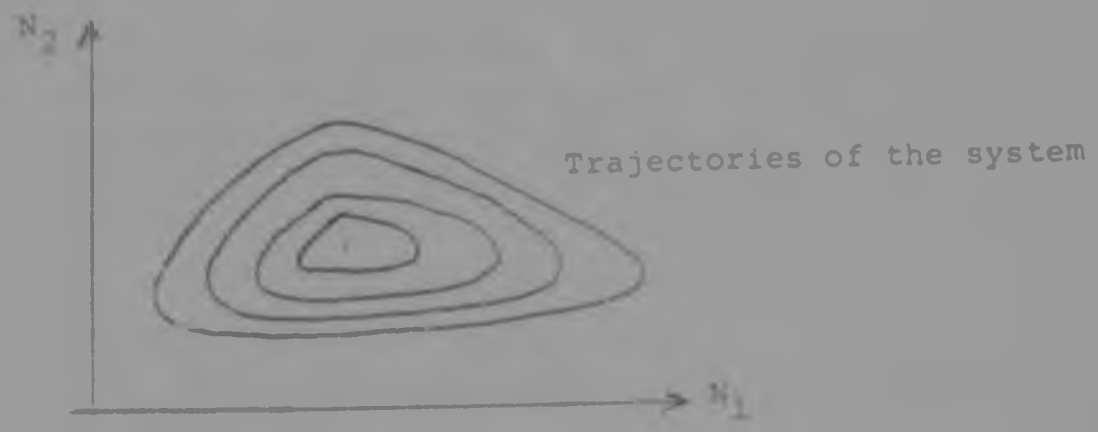


FIG. 1.8. Graph of population of species

The same formulation can be used to study concentrations of substances in chemical reactions, as done by Rosen in [9].

Similarly La Violette used the same model to study the stock marked as a predator-prey situation in [13, p.181].

1.2.2 Living Systems

A work which is the most relevant to this dissertation is Miller's [6].

Miller deals with the special class of living systems. This system comprises monerans, fungi, plants, animals, groups, organisations, societies, and supranational systems.

The general living systems theory which his dissertation

presents is a conceptual framework which attempts to unify the special bodies of knowledge dealing with each of the above systems. Out of a mass of facts regarding specific systems, his research distills certain patterns of structure process and behaviour, which are common to all living systems. This he does, not by describing loose analogies, but rather by a structured comparison of the various aspects of living systems.

Firstly he distinguishes a set of characteristics which are common to all living systems and justify their classification into the same class:

(a) They are open systems. They exchange matter, energy and information with the environment.

(b) They maintain a steady state. By taking in input of foods or fuels which are lower in entropy than the outputs, they replace components which have broken down.

(c) They have more than a certain minimum degree of complexity.

(d) They have an original "blueprint" or "program" which determines their structure and process.

(e) They are composed of aqueous suspension of proteins and include non-living components.

(f) They have some kind of control system which co-ordinates the interaction of their components.

(g) They have a set of specific critical subsystems or have symbiotic or parasitic relationships with other systems.

(h) Their subsystems are integrated together to form self-regulating, developing, unitary systems with purposes and goals.

(i) They can exist only in a specific environment where variables such as temperature, oxygen, radiation etc. do not vary beyond a certain margin. [6,p.18].

presents is a conceptual framework which attempts to unify the special bodies of knowledge dealing with each of the above systems. Out of a mass of facts regarding specific systems, his research distills certain patterns of structure, process and behaviour, which are common to all living systems. This he does, not by describing loose analogies, but rather by a structured comparison of the various aspects of living systems.

Firstly he distinguishes a set of characteristics which are common to all living systems and justify their classification into the same class:

(a) They are open systems. They exchange matter, energy and information with the environment.

(b) They maintain a steady state. By taking in input of foods or fuels which are lower in entropy than the outputs, they replace components which have broken down.

(c) They have more than a certain minimum degree of complexity.

(d) They have an original "blueprint" or "program" which determines their structure and process.

(e) They are composed of aqueous suspension of proteins and include non-living components.

(f) They have some kind of control system which co-ordinates the interaction of their components.

(g) They have a set of specific critical subsystems or have symbiotic or parasitic relationships with other systems.

(h) Their subsystems are integrated together to form self-regulating, developing, unitary systems with purposes and goals.

(i) They can exist only in a specific environment where variables such as temperature, oxygen, radiation etc. do not vary beyond a certain margin. [6,p.18].

Structures which carry out living processes are identified at seven hierarchical levels: cell, organ, organism, group, organisation, society, and supranational system [6, p.1]. Each level is made of systems of lower level [6, p.25].

Miller proposes that at each of these levels a living system will have to carry out certain processes in order to exist. Each of these essential functions is performed by a subsystem within that particular system.

Nineteen such critical subsystems have been put forward, the list of which constitutes the backbone of Miller's work:

1. Reproducer, the subsystem which is capable of giving rise to other systems similar to the one it is in.

2. Boundary, the subsystem at the perimeter of the system that holds together the components that make up the system, protects them from environmental stresses, and excludes or permits entry to various sorts of matter-energy and information.

3. Ingestor, the subsystem which brings matter-energy across the system boundary from the environment.

4. Distributor, the subsystem which carries inputs from outside the system or outputs from its subsystems around the system to each component.

5. Converter, the subsystem which changes certain inputs to the system into forms more useful for the special processes of that particular system.

6. Producer, the subsystem which forms stable associations that endure for significant periods among matter-energy inputs to the system or outputs from

its converter, the materials synthesized being for growth, damage repair, or replacement of components of the system, or for providing energy for moving or constituting the system outputs of products or information markers to its suprasystem.

7. Matter-energy storage, the subsystem which retains in the system, for different periods of time, deposits of various sorts of matter-energy.

8. Extruder, the subsystem which transmits matter-energy out of the system in the forms of products or wastes.

9. Motor, the subsystem which moves the system or parts of it in relation to part or all of its environment or moves components of its environment in relation to each other.

10. Supporter, the subsystem which maintains the proper spatial relationships among components of the system, so that they can interact without weighing each other down or crowding each other.

11. Input transducer, the sensory subsystem which brings markers bearing information into the system changing them to other matter-energy forms suitable for transmission within it.

12. Internal transducer, the sensory subsystem which receives, from subsystems or components within the system, markers bearing information about significant alterations in those subsystems or components, changing them to other matter-energy forms of a sort which can be transmitted within it.

13. Channel and net, the subsystem composed of a route in physical space, or multiple interconnected routes,

by which markers bearing information are transmitted to all parts of the system.

14. Decoder, the subsystem which alters the code of information input to it through the input transducer or internal transducers into a "private" code that can be used internally by the system.

15. Associator, the subsystem which carries out the first stage of the learning process, forming enduring associations among items of information in the system.

16. Memory, the subsystem which carries out the second stage of the learning process, storing various sorts of information in the system for different periods of time.

17. Decider, the executive subsystem which receives information input from all other subsystems and transmits information outputs that control the entire system.

18. Encoder, the subsystem which alters the code of information input to it from other information processing subsystems, from a "private" code used internally by the subsystem into a "public" code which can be interpreted by other systems in its environment.

19. Output transducer, the subsystem which puts out markers bearing information from the system, changing markers within the system into other energy-matter forms which can be transmitted over channels in the system's environment.

Miller believes that this procedure is valid at all resolution levels provided we are studying a living system which displays characteristics (a) to (t) given above, and these are cell, organ, organism, group, organization,

society, and supranational systems. As he points out in (g) they either have this set of specific critical subsystems or have symbiotic or parasitic relationship with other systems.

The above processes are essential to all living systems and are performed by all. However the structure of subsystem used to perform each process may be different for different systems or levels. Thus the gaps in the cell walls, the mouth of an animal, and an airport of a country, although incomparable in structure, perform the same function: that of bringing matter from the environment into the system.

Miller analyses these processes and structures for the proposed seven hierarchical levels. In order to demonstrate this analysis best, the chapters concerned with each level follow the same outline, with similarly titled and numbered divisions. Thus it is possible to compare particular structures and processes at various levels (8, p. xxiii).

1.2.3 Organised systems and the second law of thermodynamics

Another researcher whose work has important relevance to this dissertation is Harold Morowitz [7]. Although his work is in biology, he addresses some important questions which are fundamental to the type of systems that will be studied here.

The second law of thermodynamics asserts that any isolated system tends towards maximum entropy. On the other hand biological systems maintain their order and evolve to more complex configurations.

This superficial contradiction between the two sciences bring forward the following question: How is the order of biological systems explained in terms of matter/energy conservation and the second law of thermodynamics?

Morowitz addresses this question, and his thesis is that the flow of energy through a system can maintain as well as generate order. He supports this by examining chemical systems placed between a source and a sink of matter/energy, and showing how the flow of energy through the system generates work which can be used for the ordering of the system.

CHAPTER 1

BASIC CONCEPTS



PHENOMENON — A SERIES OF TIME SLICES

For example the phenomenon of growth of a particular tree is a spatiotemporal object consisting of a succession of time slices, each describing the tree at that interval in time (fig. 2.2). Note that this is not the general concept of "tree". This is the percept of a particular tree.



GROWTH OF A TREE

FIG. 2.2. Spatiotemporal object

2.3 THE PERCEPT SET

This is a set of elements, each representing a percept, which is identifiable by a word or sequence of words. A set of points of the observation space can now be mapped on to an element of the percept set. This can be defined as the perception mapping. Thus we can represent any observation with an element of the percept set and identify it by the use of language. A phenomenon can consist of one or more entities changing in time or it can consist of one or more entities unchanged during a time interval. For example the specific tree of fig. 2.2 may be seen not as a phenomenon of change but simply a particular entity that we want to identify, ignoring any changes (fig. 2.3).

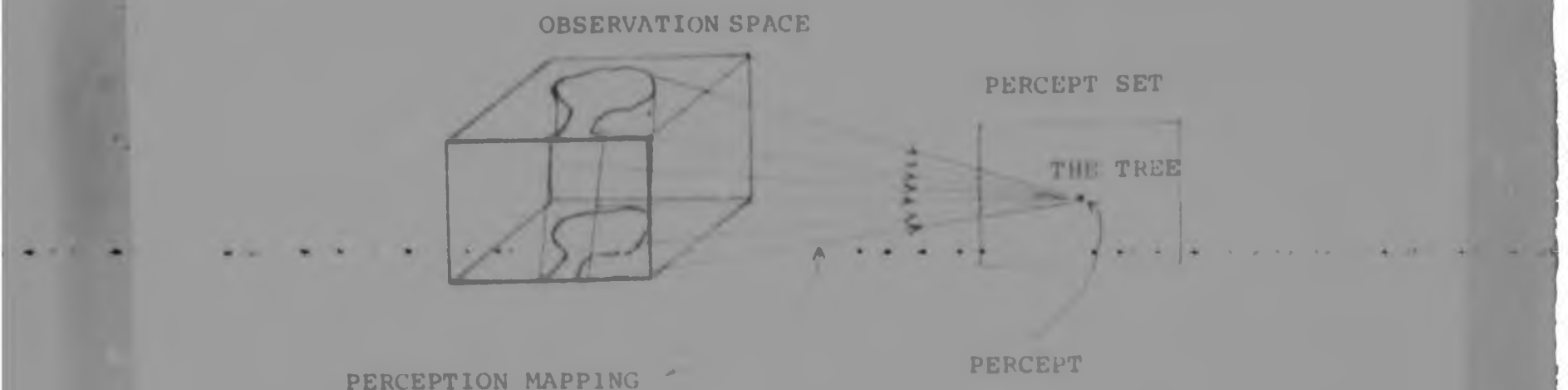


FIG. 2.3. Perception mapping

2.4 THE RESOLUTION LEVEL

In terms of the above concepts we can define resolution level as follows:

Consider an entity e.g. a particular organism. A perception mapping (P_1) maps a set of elements of the observation space on to a point of the percept set identified as (say) John Smith. Now we can consider another perception mapping (P_2), which maps subsets of our spatiotemporal object, each onto a different point of the concept set. E.g. John Smith's stomach, John Smith's heart, etc. (fig. 2.4). We say that P_2 is at a finer resolution level than P_1.

As we move from a coarser to a finer resolution level we distinguish details of our spatiotemporal object. By "distinguish" we mean that while a set in the observation space was mapped onto a single element of the percept space, we now partition the set into subsets and map each onto a different element of the percept space.

We will need to distinquish two types of resolution level differences: in the example of fig. 2.4, mapping P_2 is at a spatially finer resolution level: it distinquishes

parts of our object in space, not in time. The second type of resolution level difference is illustrated in Fig. 2.5. In this case perception mapping P_3 is at a temporally finer resolution level.

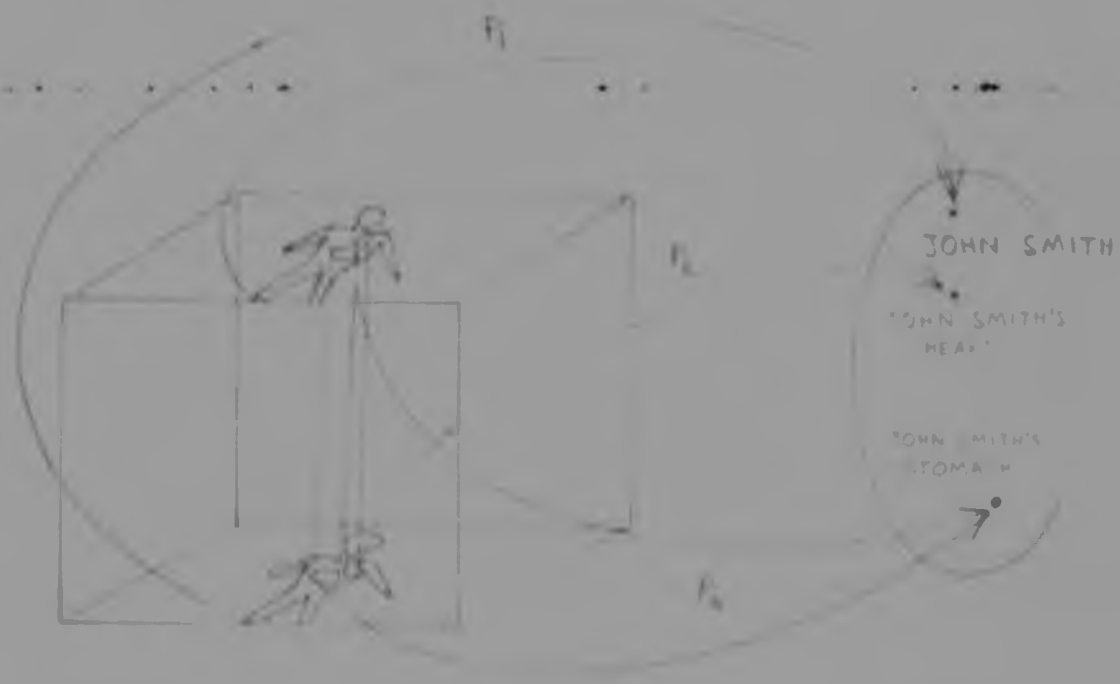


FIG. 2.4. Spatial resolution

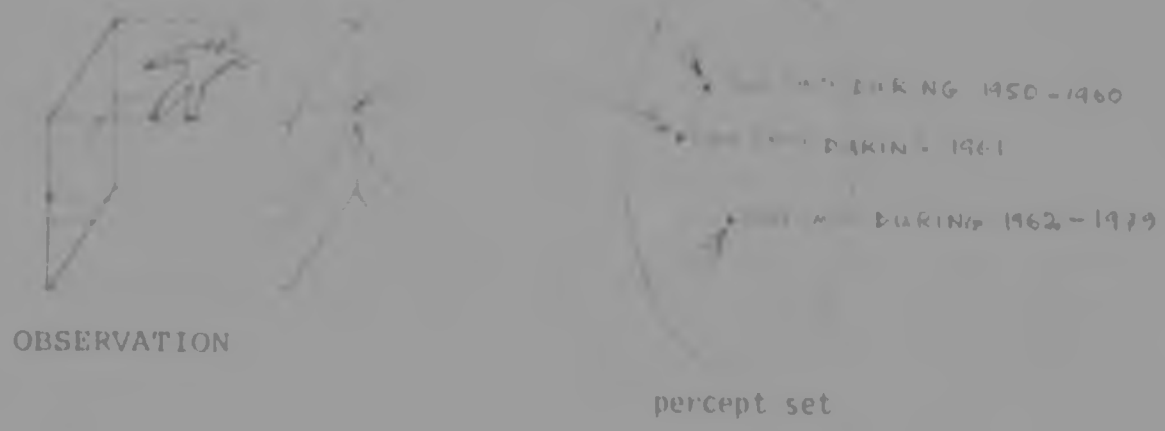


FIG. 2.5. Temporal resolution

parts of our object in space, not in time. The second type of resolution level difference is illustrated in fig. 2.5. In this case perception mapping F_1 is at a <u>temporally finer</u> resolution level.



FIG. 2.4. Spatial resolution



FIG. 2.5. Temporal resolution

The resolution level is a matter of choice of mapping and it can be as fine as possible when analysing a spatio-temporal object structurally or dynamically.

2.5 STATE OF AN ENTITY

State of an entity is an element of the percept set on to which a number of like states of an entity are mapped. This is illustrated in fig. 2.5.



fig. 2.5. State of an entity

2.6 INTERACTION

The concept "interaction" is fundamental to this dissertation.

Since the whole object of this enquiry is to investigate the possibility of constructing a model which applies to a wide variety of phenomena involving cells, organisms, etc., we need to be able to compare phenomena of diverse natures. It is towards this end that the concept of interaction will be used here.

The following examples of interaction will precede the definition.

1. A molecule of H_2 and a molecule of I_2 collide and a chemical reaction takes place. Resulting in two molecules of HI.

2. A parent cell divides into two daughter cells.

3. A child [illegible] a [illegible] with a hammer.

DEFINITION:

A. An interaction is a [illegible], (i.e. a spatiotemporal object) which contains [illegible] entities. For example, H_2, I_2, HI (example 1); parent cell, daughter cell 1, daughter cell 2 (example 2); child, hammer, vase, porcelain pieces (example 3).

B. An interaction can be divided into at least three time slices as shown in fig. 2.7.



FIG. 2.7. Three time slices of an interaction

E.g. For the example (2) above we have fig. 2.8.

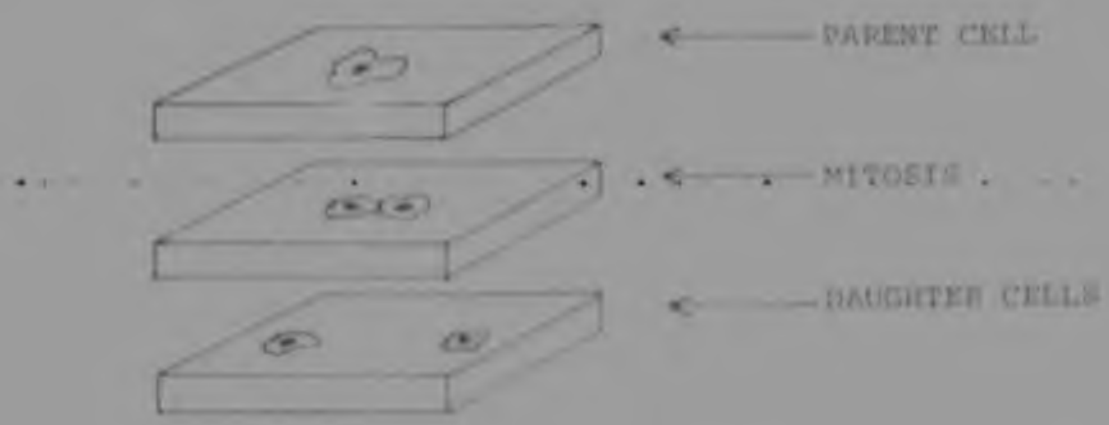


Fig 2.8. Example of interaction

c. We observe that for the chemical reaction in example (i) to take place, the molecules H_2 and I_2 had to collide, i.e. come to within relatively close distance of each other. Furthermore the two resulting molecules of HI were at some time instant relatively close to each other. The same applies for the daughter cells of example (j) as well as for the entities boy, hammer, vase, and key, hammer porcelain pieces of example (j).

There exists at least one time slice in which the entities present are relatively close. The smallest volume enclosing all the entities we will call the <u>interaction site</u>.

It should be noted here that when we consider a simple interaction (as in the examples above), it is assumed that at the present resolution level, the <u>maximum information available</u> is that: "a change has taken place at the intermediate time slice". At a finer resolution level this interaction may consist of many interactions (see interaction networks, section 2.8) and this would constitute the "explanation" of the above interaction.

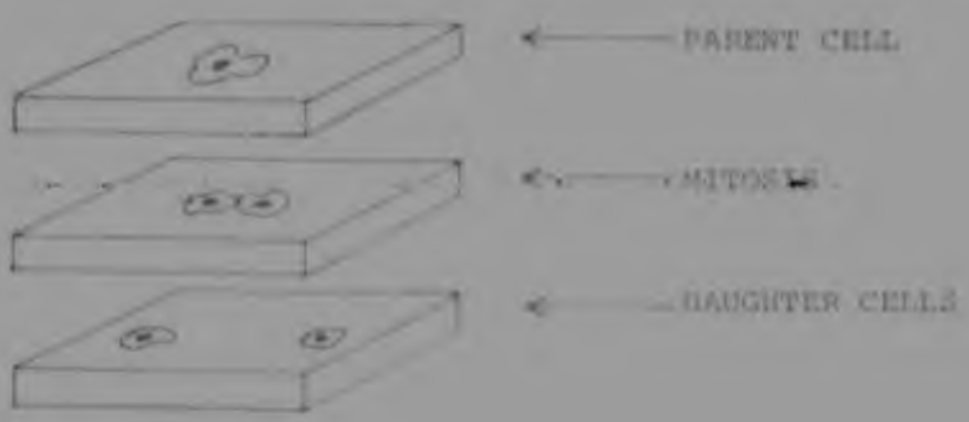


FIG: 2.8. Example of interaction

c. We observe that for the chemical reaction in example (1) to take place, the molecules H_2 and I_2 had to collide, i.e. come to within relatively close distance of each other. Furthermore the two resulting molecules of HI were at some time instant relatively close to each other. The same applies for the daughter cells of example (1) as well as for the entities boy, hammer, vase, and boy, hammer porcelain pieces of example (3).

There exists at least one time slice in which the entities present are relatively close. The smallest volume enclosing all the entities we will call the intersection site.

It should be noted here that when we consider a simple interaction (as in the examples above), it is assumed that at the present resolution level, the maximum information available is that: "a change has taken place at the intermediate time slice". At a finer resolution level this interaction may consist of many interactions (see interaction networks, section 2.0) and this would constitute the "explanation" of the above interaction.

The term "interaction" has been used since in its usual meaning it signifies an arbitrary change involving any number of entities and it is consistent with the uni-directional nature of time. Two people or two molecules come together, "interact" and move on. The people may move away from each other but with new different mental states (due to "information" exchanged), or the molecules may now move as a single new molecule (due to some chemical bond that has been formed). In all cases the sequence initial-intermediate-final (time order) is unidirectional, and a basic component of "interaction", and at that resolution level no further information is provided, other than the identity of all the (before and after) entities involved.

Three examples can now be considered diagrammatically in the light of the above definitions (fig.2.9).

Thus an interaction is an elementary change which at the given resolution level cannot be further analysed. It gives the "before" and the "after" situation in the observation space. It is a pair of sets of information. The first is an identification of the entities existing before the change, and the second an identification of entities existing after the change.

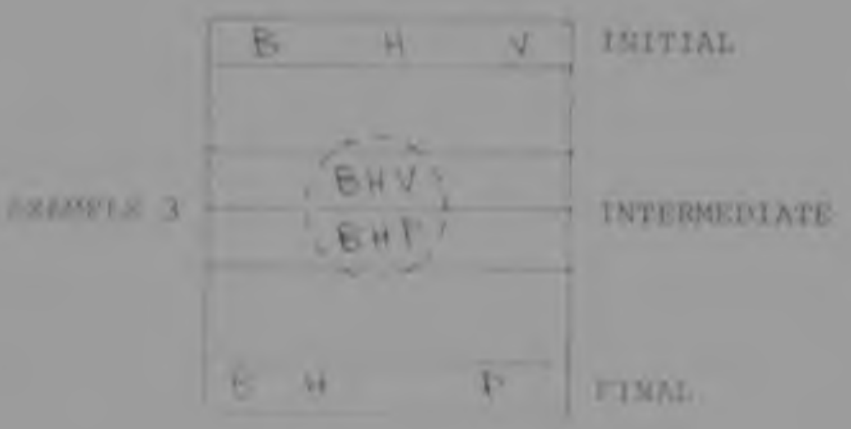
If further information is required regarding the change, then a finer resolution level has to be adopted and the above single interaction replaced by a set of interactions (see section 2.8).

2.7 SCHEMATIC REPRESENTATION OF INTERACTIONS

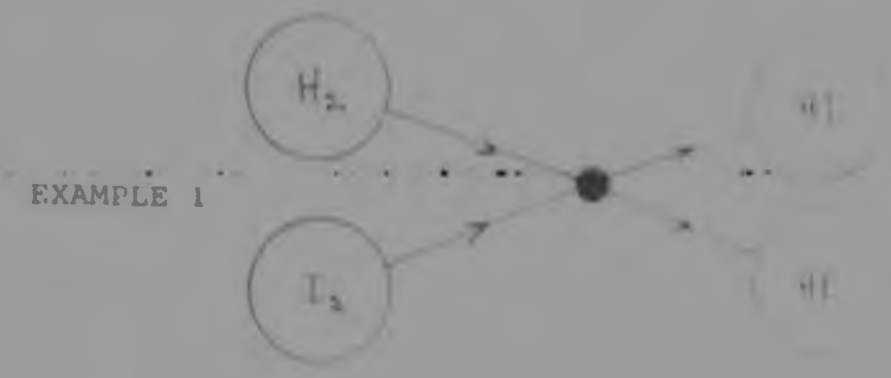
This may be done as shown in fig. 2.10. The above three examples can thus be represented as in figure 2.11.


EXAMPLE 1
INITIAL
INTERMEDIATE
FINAL
HI
HI


EXAMPLE 2
INITIAL
INTERMEDIATE
FINAL


EXAMPLE 3
INITIAL
INTERMEDIATE
FINAL


ENTITY
1
ENTITY
2
INTERACTION
SITE
ENTITY
3





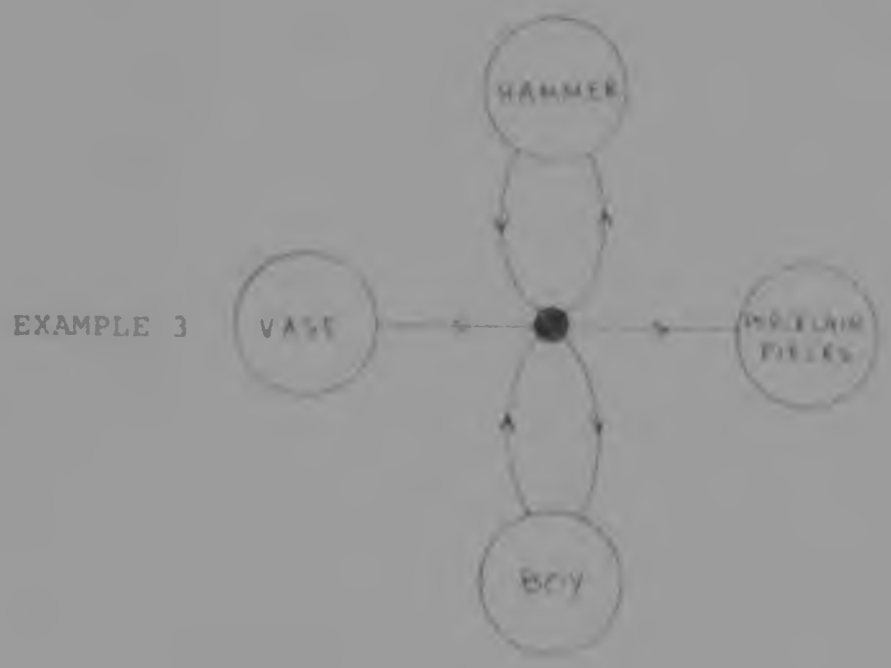


FIG. 3.11. Examples of schematic representation of interaction.

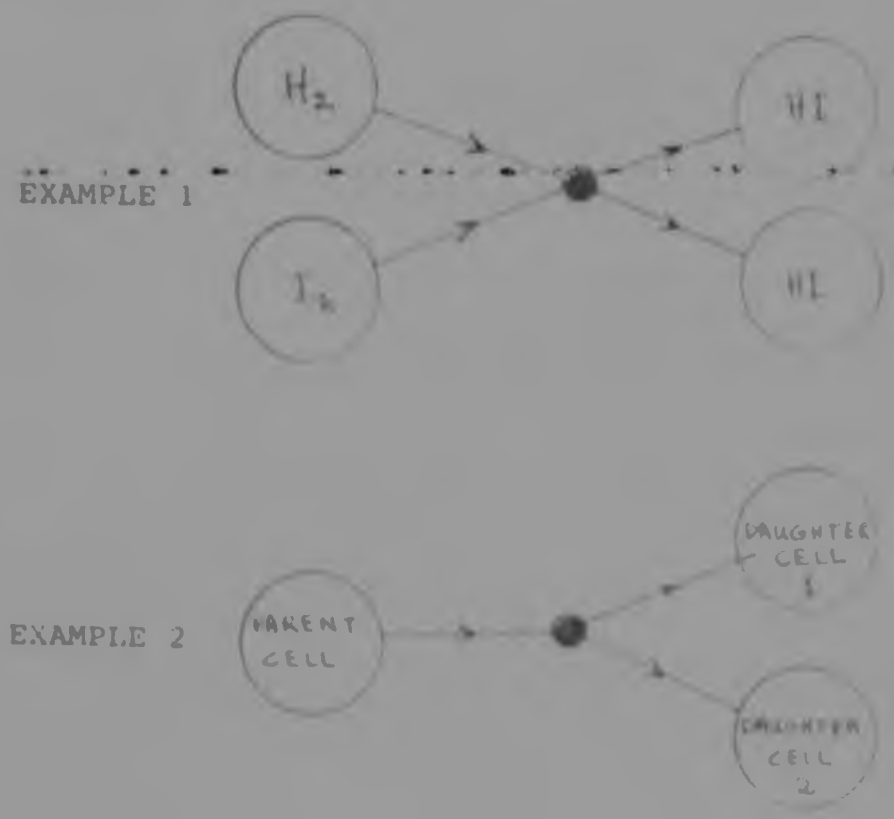


FIG. 2.11. Examples of schematic representation of interactions

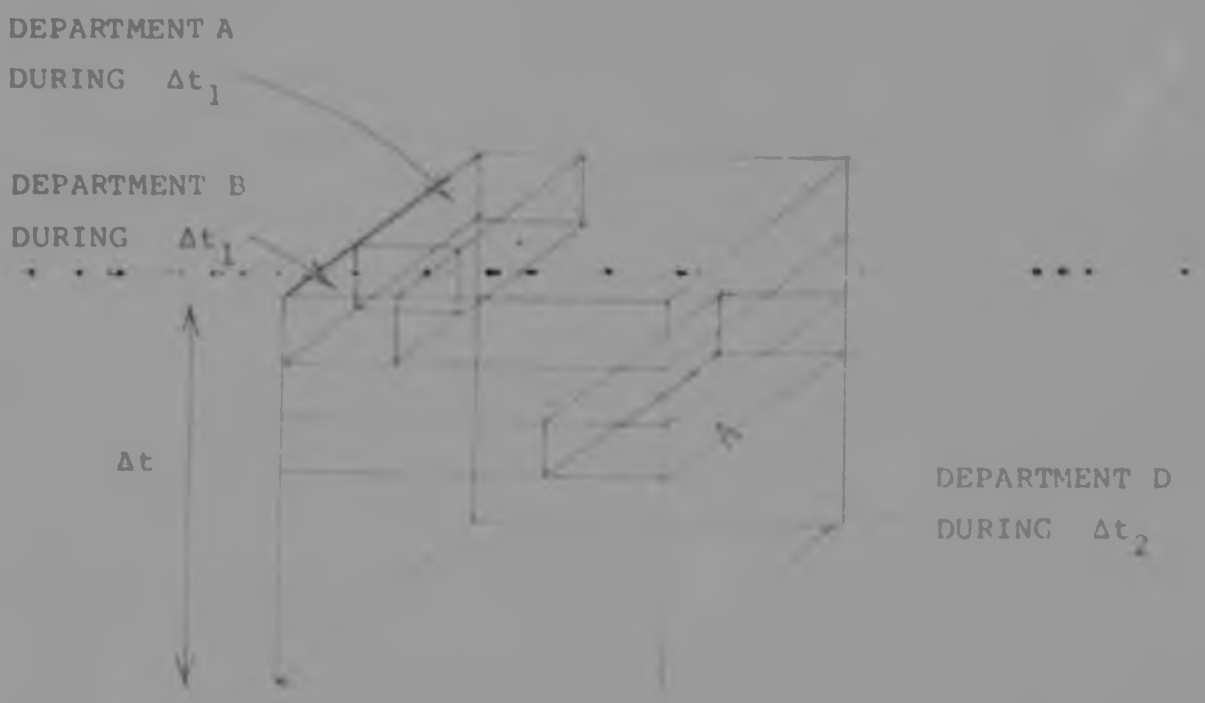


FIG. 2.12. A factory as a spatiotemporal object

Taking a time slice Δt_1 and the factory departments A and B, two interactions can be distinguished (fig. 2.13a and 2.13b).

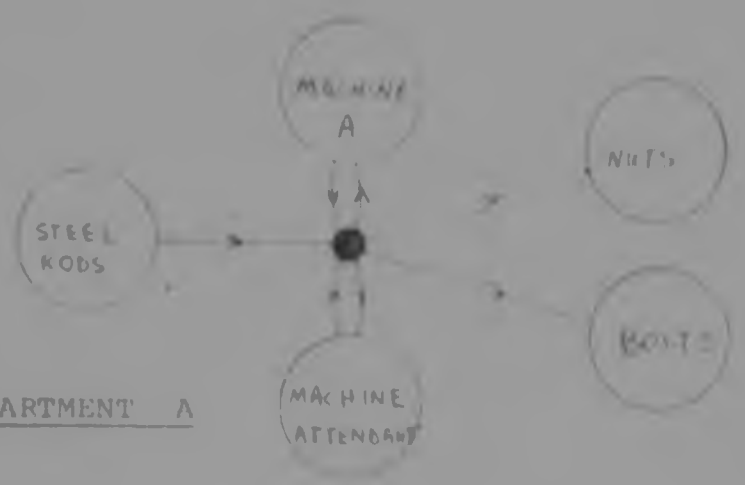


FIG. 2.13a. Department A as an interaction

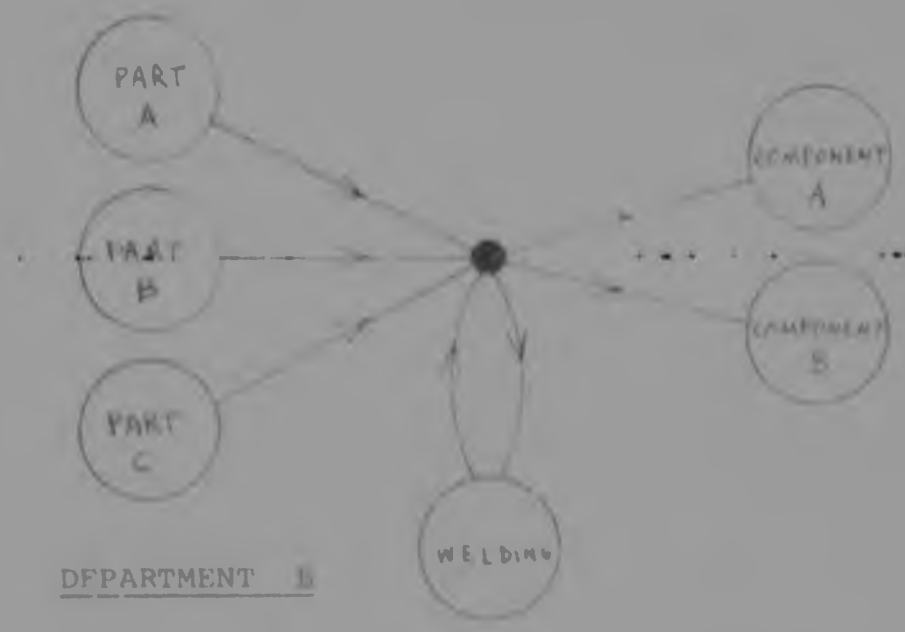


FIG. 2.13b. Department B as an interaction

Now looking at the activities of department D during Δt we have (fig. 2.14)



FIG. 2.14. Department D as an interaction

In this way we can distinquish the various single interactions in the factory during Δt. Finally the whole object can be described in terms of interactions shown in fig. 2.15.



FIG. 2.15. Example of interaction network

In this way we can distinguish the various single interactions in the factory during Δt. Finally the whole object can be described in terms of interactions shown in fig. 2.15.



FIG. 2.15. ARRANGEMENT OF INTERACTION NETWORK

The above example showed the suitability of interaction networks in representing processes which involve entities such as people and objects.

Example 2: Processes involving molecules can equally well be represented as follows.

Consider the reversible reaction

$$H_2 + I_2 \rightleftharpoons 2HI$$

The process taking place in a container can be represented as in fig. 2.16

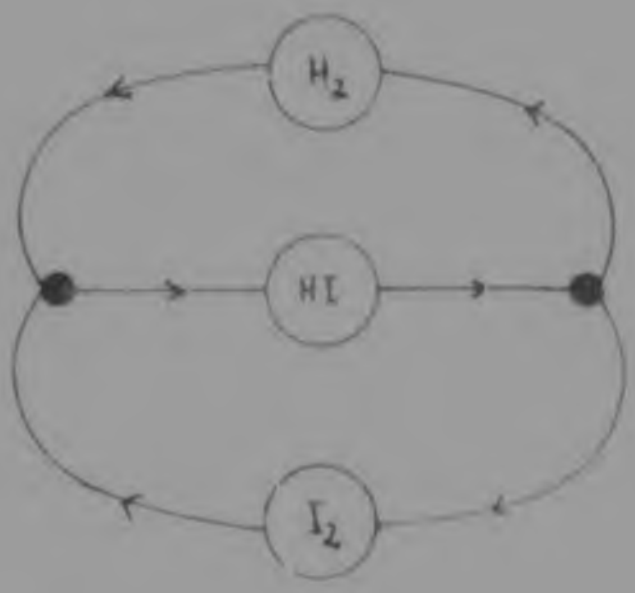


FIG. 2.16. Interaction network for chemical reactions

where H_2 represents the entity "concentration of hydrogen", its various states being the concentration values. Similarly for I_2 and HI.

The above examples suggest that interaction networks can be used to describe processes involving any type of entity.

The above example showed the suitability of interaction networks in representing processes which involve entities such as people and objects.

Example 2: Processes involving molecules can equally well be represented as follows.

Consider the reversible reaction

$$H_2 + I_2 \rightleftharpoons 2HI$$

The process taking place in a container can be represented as in Fig. 2.16

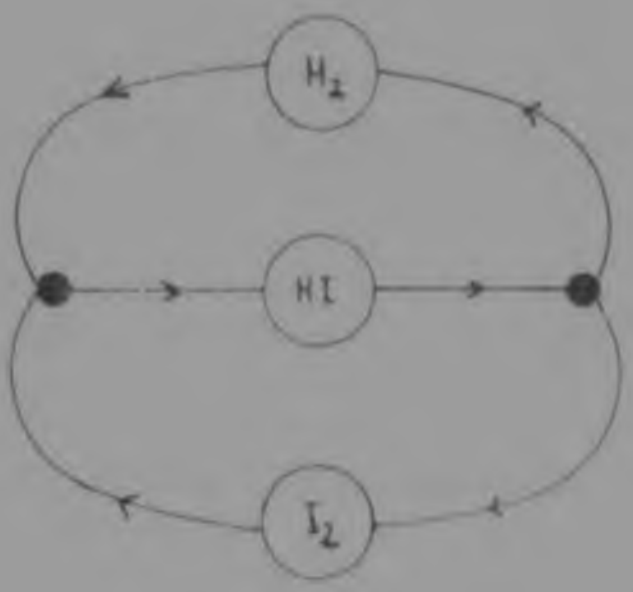


FIG. 2.16. Interaction network for chemical reactions

where H_2 represents the entity "concentration of hydrogen", its various states being the concentration values. Similarly for I_2 and HI.

The above examples suggest that interaction networks can be used to describe processes involving any type of entity.

2.9 SIMPLIFIED DESCRIPTION OF NETWORKS

When a large number of entities and interactions are involved in a network a simpler diagram can be used as shown in fig. 2.17.



FIG. 2.17. Simplified diagram of interaction network

This displays the interactions taking place and the entities which are involved in each interaction and affected by it.

This scheme can also be used when it is desirable to display which entity interacts with which, with the actual nature of the change involved regarding each entity, is described in a mathematical or computer model (see next section). For example the simplified network of fig. 2.17 above, could have been used to represent the network in fig. 2.18 below, which detailed information regarding the

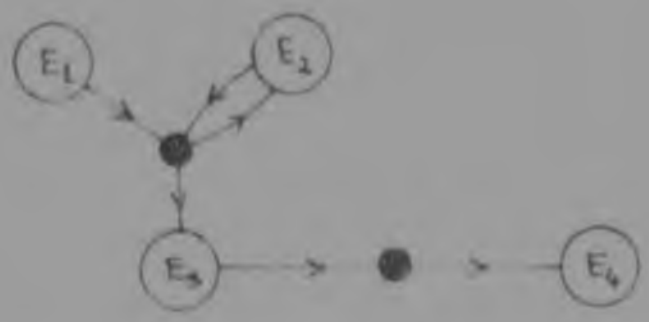


FIG. 2.18. Interaction network example

change of state (which is indicated by the arrows) is represented in another medium.

2.10 PROCESSES

A process is defined here as a subnetwork of interactions. Thus a process can consist of one or many interactions within an interaction network.

The possibility of the existence of interaction characteristics which can be used to partition cells, organisms and societies into processes, will now be investigated. These partitions will constitute the components of our general model and make the comparison of these diverse systems possible.

By the term "interaction characteristics" we mean a relationship between initial and final entities of a process. For example the interactions in this text may be called syn-

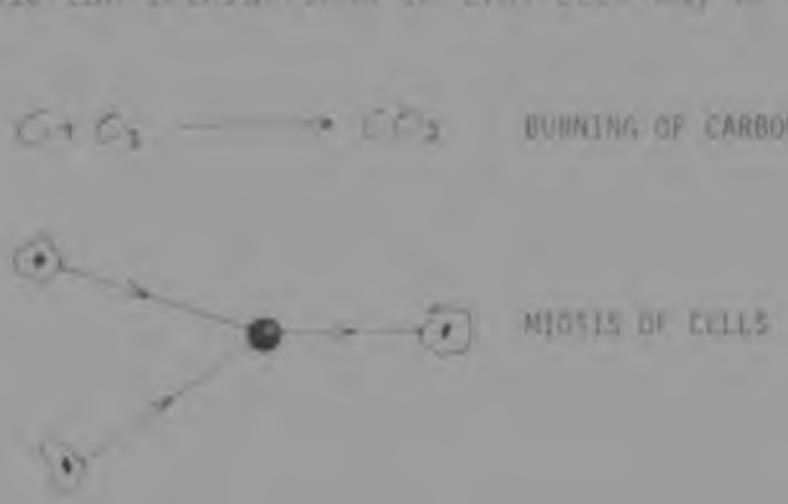


FIG. 2.19. Synthetic interactions

thetic interactions for obvious reasons. In the case of an interaction network, this is shown in fig. 2.20



FIG. 2.20. Synthetic interaction network

2.11 CATALYSTS

Consider the interaction in fig. 2.21. Entities 1 and 2 enter into interaction to produce entities 1 and 3. Therefore entity 1 is unaffected by the interaction but necessary.



FIG. 2.21. Interaction involving a catalyst

An entity can be defined as a catalyst of an interaction if it is present in both initial and final time slices, unchanged, and if the interaction under study does not take place unless the entity is present.



Example (A): The enzyme is strongly known as a catalyst in biochemistry.

Example (B): In the process of the cleaning of the blood to produce clean blood and urine, all the entities considered as change except the kidneys. The kidney is the tool which produces the change and satisfies our definition of catalyst above.

Example (C): As in the example (B) above the artisan plays the role of the agent of change.

As seen in fig. 2.23 the definition of catalyst is

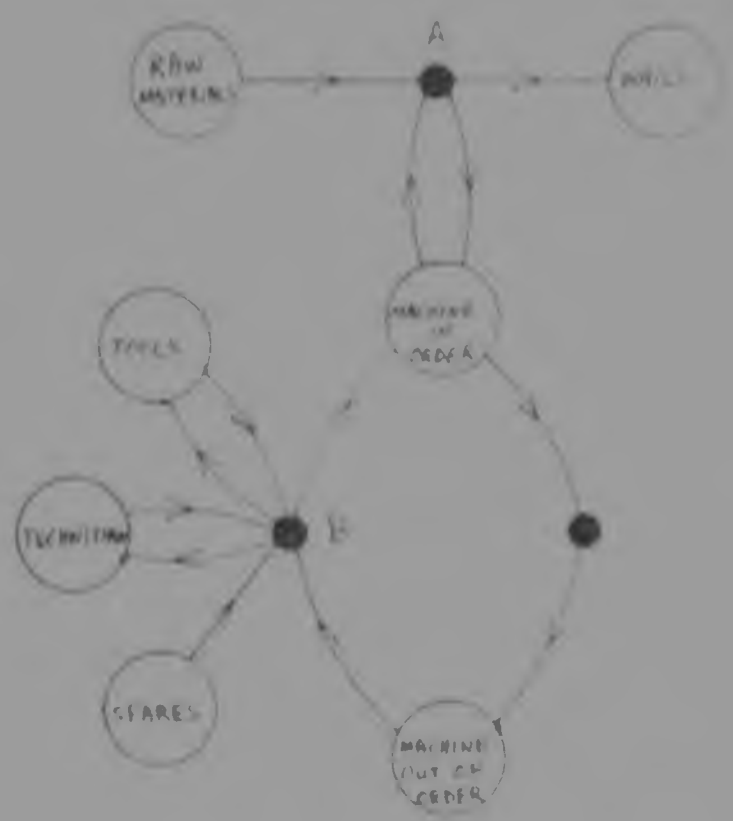


FIG. 2.23. The attribute of "catalyst" in relation

relative characteristic which depends on the differences in the magnitude of changes undergone by the entities involved in the particular interaction. In the above example (fig. 2.23), the machine is a catalyst with respect to interaction A, but it is a product with respect to interaction B. In interaction A its change is negligible; the change of interest takes place in the materials and nails. But its change in interaction B is very significant: here the technician and the tools are the catalysts.

2.1: INFORMATION

Phenomena such as transmission of information, decision making and computer processing are normally (for convenience sake) treated as involving the transmission and processing of something non-material. However these phenomena involve basically changes of (material) entities in our observation-space and can be studied as networks of interactions.

INFORMATION STATE: What is normally considered as a change in the information content of a system (the obtaining or losing of knowledge), will here be treated as a change of state of the entity concerned. For example consider a person (John Smith) who has been taught by a series of demonstrations how to make an apple pie. This can be considered as a change of state from a person who does not know (state X), to person who does know how (state Y). The difference of state is defined by the following two possible interactions illustrated in fig. 2.24.

The essential theme here is that the two objects, "John Smith in state X" and "John Smith in state Y", are being distinguished by mapping on different elements of the percept set (section 2.1). The fact that we consider both objects as being states of the same object "John Smith" by mapping both on to a further element "John Smith"

is irrelevant in the context of our interaction network. The gain of knowledge is a change from one entity to another which is distinguishable because it interacts differently.

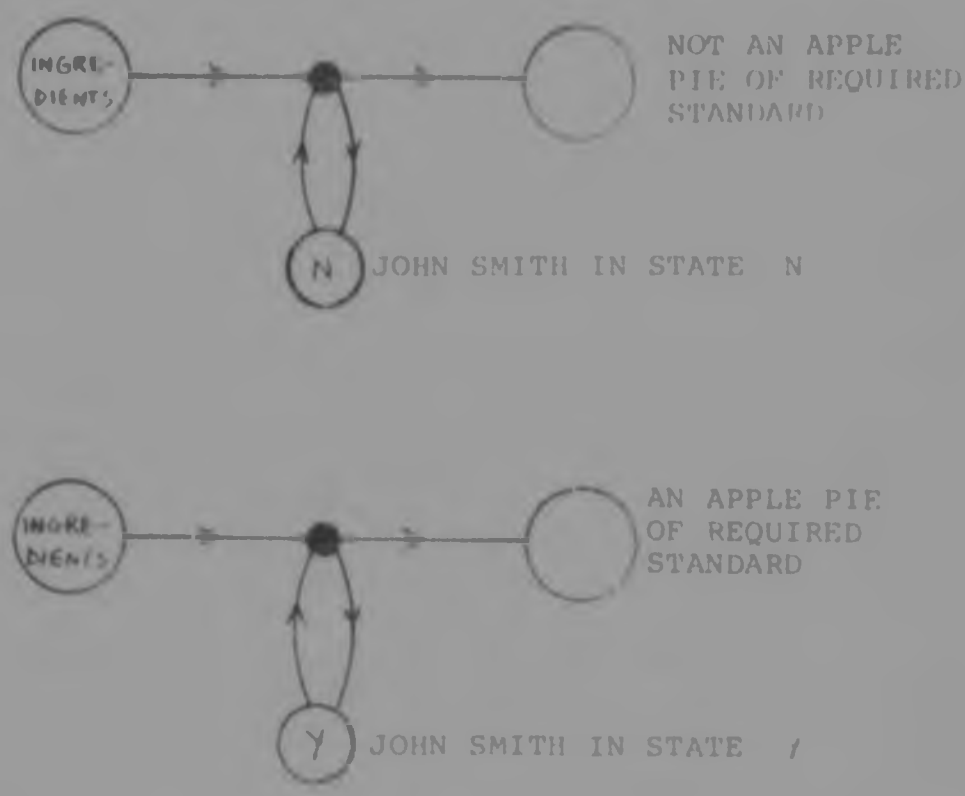


FIG. 2.24. Difference in information states

A more obvious example of information being a change from one object to another is the act of (say) John Smith writing down the recipe (fig. 2.25). Here the object A is changed into object B in the sense that if another person interacted with the object A he would not change his state of knowledge while he would, if he interacted with object B (as described in fig. 2.24).

TRANSFER OF INFORMATION: All changes in information considered above are a result of interactions that took place. This is commonly named transfer of information. What should be emphasised here is that transfer of information being an interaction (or network of interactions) involves an interaction site, i.e. the coming close of a

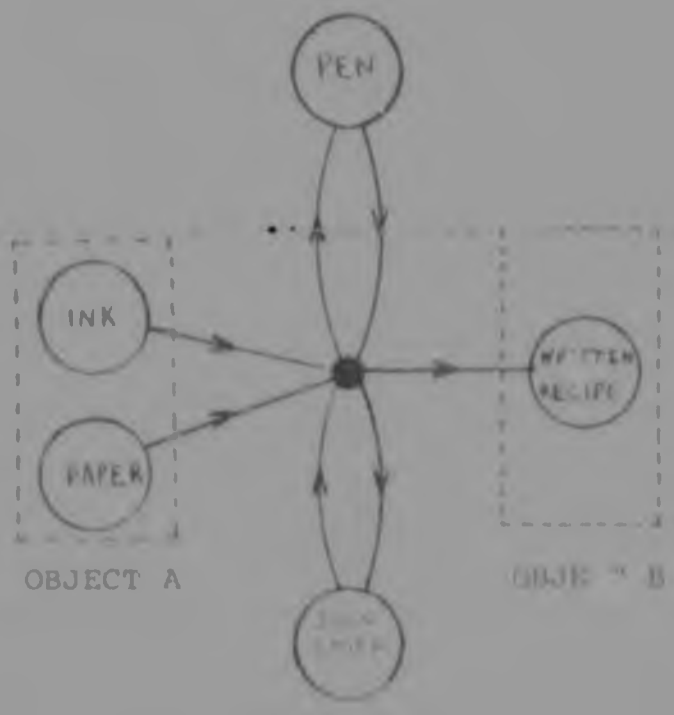


FIG. 2.25. Information as a change of one object to another

"transmitter" and a "receiver" of information. For example in order that John Smith learns how to make an apple pie, he will have to come relatively close to a book or a letter or a telephone receiver or another person, so that the (information transfer) interaction can take place.

INFORMATION CHANNEL: If an information transfer takes place from an entity A through an interaction network to an entity B, then the interaction network between entities of interest (A and B) is the information channel (fig. 2.26).



FIG. 2.26. Information channel

Two types of transfer of information can be distinguished:

1. Information transfer by transportation where the interaction network involves movement of an information carrier in space (fig. 2.27).

   Posting of letters is a typical example of such transfer of information. Similarly the transmission of messages through the circulatory system of animals by releasing hormones in the blood. These are information carriers (or markers) which physically move to the receiver and result in its change of state.

2. Information transfer by transmission where the interaction network involves information carriers which do not move in space but rather change states successively as shown in fig. 2.28.

   Transfer of information by waves in a medium is a typical example. In the case of sound waves the change of A_1 to B_1 represents a change of pressure of a small volume of air. Each small volume of air causes the pressure in the neighbouring volume to change. The chain reaction produced propagates the change of



FIG. 2.26. Information channel

Two types of transfer of information can be distinguished:

1. Information transfer by transportation where the interaction network involves movement of an information carrier in space (fig. 2.27).

   Posting of letters is a typical example of such transfer of information. Similarly the transmission of messages through the circulatory system of animals by releasing hormones in the blood. These are information carriers (or markers) which physically move to the receiver and result in its change of state.

2. Information transfer by transmission where the interaction network involves information carriers which do not move in space but rather change states successively as shown in fig. 2.28.

   Transfer of information by waves in a medium is a typical example. In the case of sound waves the change of A_1 to B_1 represents a change of pressure of a small volume of air. Each small volume of air causes the pressure in the neighbouring volume to change. The chain reaction produced propagates the change of

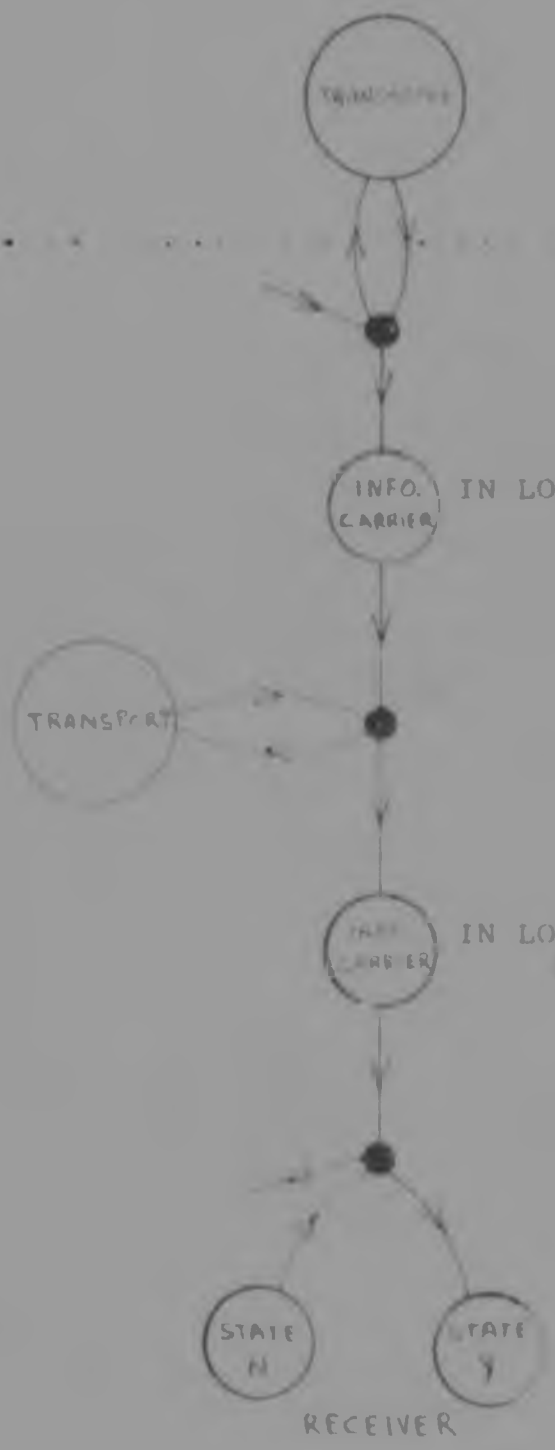


FIG. 2.27. Information transfer by transportation



FIG. 2.27. Information transfer by transportation



FIG. 2.27. Information transfer by communication.

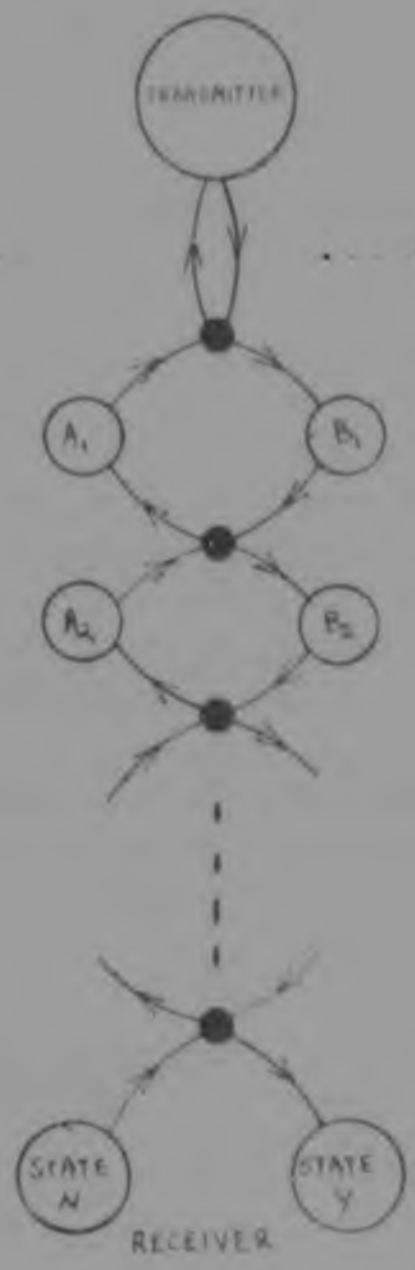


FIG. 2.28. Information transfer by transmission

state (pressure) until it reaches the receiver as shown in fig. 2.28. The same scheme represents the transmission of impulses through a chain of neurons in the nervous system.

CHAPTER 3

SELF-MAINTAINING ORGANISED SYSTEMS

3.1 INTRODUCTION

In the previous chapter the necessary concepts, terms, and means of representation for the description of phenomena have been established. These will be used in this chapter to investigate the characteristics of self-maintaining organised systems.

After identifying and describing the physical systems to be analysed, their common fundamental characteristics will be identified and formulated in terms of interaction networks.

3.2 THE PHYSICAL ENTITIES TO BE CONSIDERED

Following is a quick view of the main classes of identifiable entities in nature.

If we start with the smallest identifiable entities which represent the finest resolution level possible presently, we have subatomic particles - protons, neutrons, electrons, etc.

At a coarser resolution level we have atoms which consist of a network of interactions containing subatomic particles.

Molecules are networks of interactions containing atoms. For example a protein molecule consists of many atoms in an interaction which has the property of preserving their relative spatial positions.

Cells are networks of interactions involving molecules. DNA, RNA, enzymes and other molecules present are in continuous interaction with the result of preserving the identifying variables of the cell, e.g. membrane, nucleus, intake of specific nutrients, etc.

Organisms are networks of interactions involving cells which produce a stable structure and behaviour.

Finally societies (i.e. separate countries) are interaction networks involving organisms.

In this dissertation we will limit our investigation to cells, organisms and societies. Any conclusions reached will therefore also be limited to these three systems and at most suggest a similar system which includes the rest of the systems.

Obviously the analysis of an organised system depends on what is defined as the system and what is defined as its environment. In the following analysis it will be assumed that:

A cell is the totality of entities within the cell membrane.

An organism is the totality of entities within the skin of the animal, including food in its digestive tube, Air in its lungs, etc.

A society is the totality of people, buildings, roads, cultivated land, livestock and anything that is commonly understood as being controlled by man in that particular

country. The ore in the ground, the atmosphere, etc. as well as other countries will be regarded as the environment of the society being studied.

In the above definition the distinction between an organised system and its environment is arbitrary, but once decided it is fixed and the analysis that follows will be based on it. Furthermore we are not going to consider changes in the basic structure or overall behaviour of the systems, i.e. we will study the system over a period of time Δt where there are no significant structural changes associated with what is known as development or decay. In other words the time period Δt will be small compared with what is known as the lifespan of the organised system being studied.

3.3 BASIC CHARACTERISTICS OF SELF-MAINTAINING ORGANISED SYSTEMS

What do cells, organisms and societies have in common? At first sight these entities are so different that a comparison seems hardly possible.

Yet in spite of the difference in size, content, and behaviour, they have the following characteristics in common:

1. They survive in a changing environment, by making the appropriate internal and external adjustments.

   Example:

   Cells: An amoeba draws away from a toxic substance.

   Organism: Perspires when the environment temperature increases.

   Society: Central heating is switched on in winter in most of the buildings.

2. There is a continuous exchange of matter and energy with the environment.

   Example:
   Cells: absorb nutrients.
   Organisms: eat food.
   Societies: use ores, water etc. from the environment.

3. The entities consist of distinguishable parts which are involved in specific processes.

   Example:
   Cells: Structures exist such as chromosomes, mitochondria, ribosomes; and biochemical processes take place, such as the decoding and transmission of the DNA code to produce specific proteins.
   Organisms: They have a specific anatomy (heart, stomach, kidneys), and processes such as digestion, blood circulation etc.
   Societies: Roads, harbours, factories, and people are some of its components; some of the processes involved are clothes manufacturing, telephone communications etc.

Entities which possess these three fundamental characteristics will be referred to as self-maintaining organised systems. These characteristics will now be investigated formally as interaction networks.

3.4 STABILITY OF IDENTIFYING VARIABLES

Any entity, when examined at a finer resolution level, is an interaction network. The entities which we are examining are also interacting with the environment. In terms of our interaction networks, such an entity is represented as in fig. 3.1.

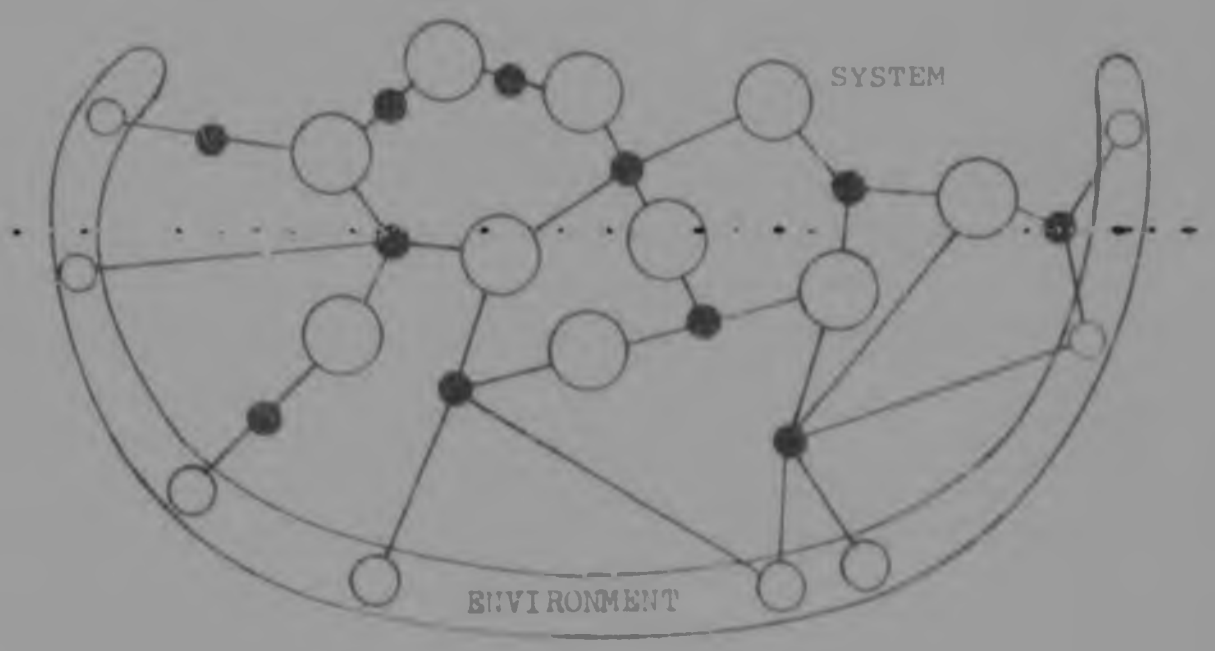


FIG. 3.1. Interaction with
the environment

A special relationship however will have to exist between the interactions in order that survival in a changing environment can be achieved.

It is suggested here that survival is defined as the stability of the identifying variables. In other words if the entity under study is investigated at a finer resolution level so as to distinguish its characteristics and their variation, these vary only within limits which are indistinguishable at a coarser resolution level.

For a specific organism for example survival means the variation of characteristics such as heartrate, temperature, anatomy, etc., within certain limits (it is on these characteristics that the perception mapping is based, as described in sections 2.3 and 2.4). Non-survival or death, is the variation of these characteristics outside these limits and failure to return within these limits.

Suppose that $x_1,\ldots,x_n$ are the component entities of an interaction network, and $y_1,\ldots,y_m$ the component entities of the environment which interact with the network.

All the possible states of the entity can be represented by an n-dimensional space. Similarly the environment can be represented by an m-dimensional space. If the entity under study is a self-maintaining organised system, then the system of equations

$$\left.\begin{aligned} \frac{dx_i}{dt} &= f_i(x_1,\ldots,x_n y_1,\ldots,y_m) \qquad i=1,\ldots,n \\ \frac{dy_j}{dt} &= f_j(x_1,\ldots,x_n,y_1,\ldots,y_m) \qquad j=1,\ldots,m \end{aligned}\right\} \quad (3.4.1)$$

describing the relationships of all the interactions and entities involved, is such that if the entities $y_1,\ldots,y_m$ change states then the trajectory of the states of $x_1,\ldots,x_n$ returns to the same volume of the n-dimensional space. As illustrated in fig. 3.2, when the state of the environment changes in the volume E, then the system after an initial perturbation always returns to a point inside the volume S of its state space.

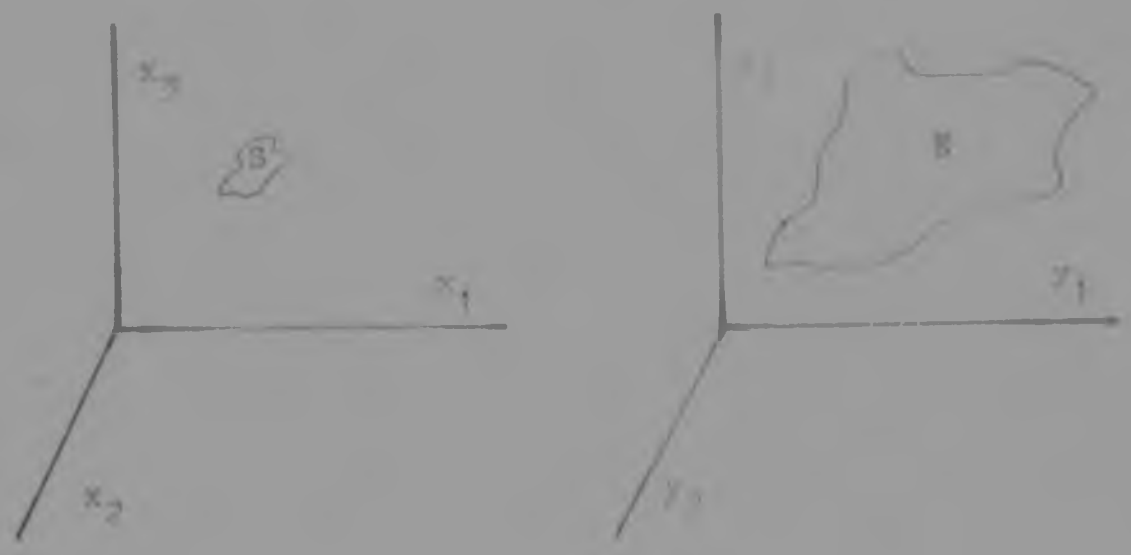


FIG. 3.2. State spaces of entity and environment

There is, however, a boundary around volume E such that if the environment changes to a state outside E, then the system does not return to S.

Having defined the system of equations (3.4.1) which represents the interaction network being studied, and the requirement that it must satisfy, the reader is referred to the work of Rosen [9], from which a more detailed mathematical analysis can be derived. Rosen uses systems of differential equations for the study of properties such as stability, equifinality and regulation in biological systems. His analysis is also applicable to the general class of self-maintaining systems as defined above.

Consider the following two sets of states for an entity:

1. "entity in normal state" = entity in one of the states in volume S (fig. 3.2) of the state space.

2. "entity under stress" = entity state not in S.

Then survival by continuous adjustment can be represented as in fig. 3.3.

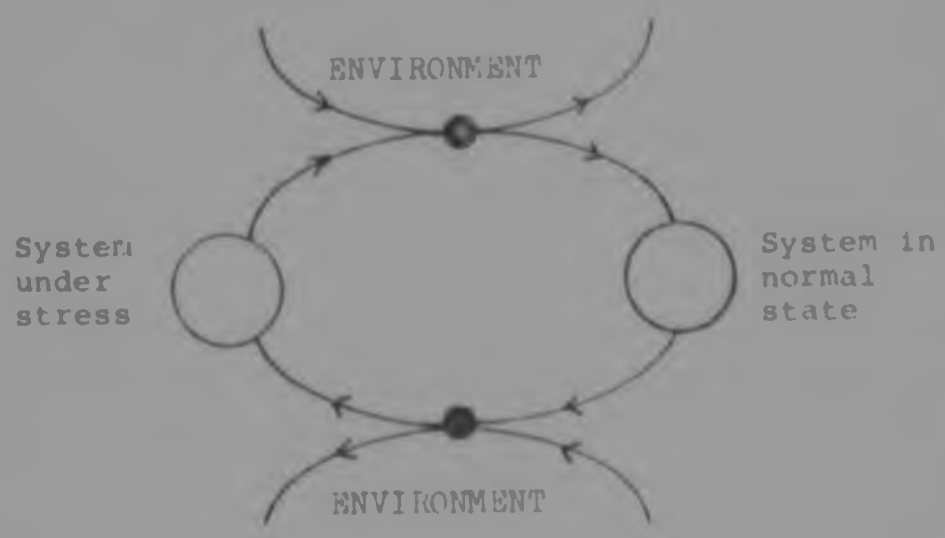


FIG. 3.3. Survival by continuous adjustment

3.5 THE SELF-MAINTAINING CYCLIC PROCESS

The backbone of every self-maintaining organised system is a process which can be termed "self-maintaining cyclic process" or metabolism. This is represented in fig. 3.4 and explained below.

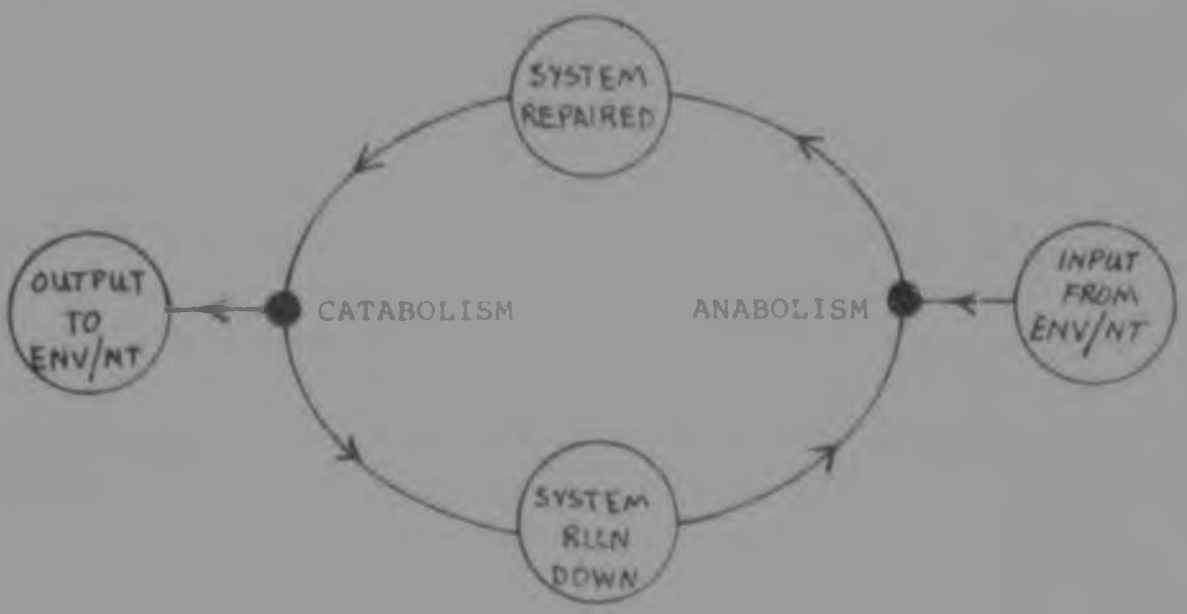


FIG. 3.4. Self-maintaining cyclic process

In considering this process the existence and structure of physical component entities of an interaction network will be considered.

According to the second law of thermodynamics every isolated physical entity disintegrates because irreversible processes change it to a state of higher entropy.

If an interaction network consists totally of physical entities which don't disintegrate this belongs to the class of trivial self-maintaining systems. Trivial in the sense that at the next finer resolution level, the components of the system are themselves stable and constant.

Our interest here lies in the wider realm of interaction networks consisting of physical entities which do disintegrate. In this realm the only possibility of stability of the components of the interaction network is the existence of interactions which are such that they result in the substitution of the disintegrated components by newly constructed ones. Such a process can be summarised at a lower resolution level as the interaction of Anabolism (fig. 3.4).

Anabolism implies, by the law of conservation of matter and energy, that the environment takes part in the interaction thus resulting in the transfer of matter and energy from the environment to the system. This is the amount of matter and energy required to make the new component.

Similarly the interaction of disintegration can be summarised as the interaction of Catabolism and result in the transfer of matter and energy from the system to the environment.

Morowitz [7] has made a study on the flow of matter and energy through biological systems which is very relevant to the above, and the reader is referred to his work for detailed analysis and experimental support of the principles being generalised here.

Examples:

Cells: nutrients from the environment are used to construct enzymes and various organelles (anabolism). These same components disintegrate continuously (catabolism).

Organisms: food is processed to ultimately become part of new cells (anabolism) which replace dead ones (catabolism).

Society: metals, coal, etc. are extracted from the environment to be used in the manufacture of commodities (anabolism) which replace machines

etc. (catabolism).

3.6 THE ELABORATION AND ENHANCEMENT OF THE CYCLIC PROCESS

As described above, metabolism is only the backbone of a self-maintaining organised system. Given a milieu of entities there is an indefinite number of networks that could satisfy the requirements of the above process of metabolism. Furthermore this has to be such that it is relatively unperturbed by the environment (i.e. stable).

Thus one finds in nature many forms of metabolism. In each case its ramifications are the structures and processes which give stability to the metabolism and make the entity relatively independent from the environment. This gives rise to a great number of types of entities in nature for each level, and different types of cells, organisms and societies.

Some aspects of the above ramifications are:

The process of changing materials from the environment to replace system entities can vary in complexity. Similarly for the process of discarding waste to the environment.

Auxiliary processes can also be used to provide stability and control the interactions of the metabolism.

The types of materials used can vary, and likewise the process of obtaining them from the environment.

Finally the interactions with the environment can vary from simple absorption of food to complex movement of the entire entity relative to the environment or parts of it. The information processing involved in controlling such interactions of the entity with the environment can vary from very simple feedback loops to complex information processing subsystems such as the nervous system of animals.

Thus every cell, organism, and society is yet another solution to the problem of self-maintaining cyclic process which is relatively independent from environmental perturbations.

3.7 SPECIALISATION AND DIFFERENTIATION

Consider the different interactions that take place within an interaction network.

In sections 2.11 and 2.12 we saw that interactions in a network can be classified into different types. A classification of such types is given in the next chapter. It is sufficient for the purpose of this section to point out that such a classification is possible, e.g. input processes, output processes, communication etc. From this it is apparent that some component entities are involved in single types of interactions while others are involved in several types of interactions.

This involves the idea of specialisation. Its basic every-day meaning is obvious when two different phases of development of the human society are considered as an example.

In the stone age all men were hunters and all interactions of an individual with the environment were generally the same as those of the other individuals.

In today's society there is a wide variation between interactions of different individuals with the environment. For example the daily interactions of a manager are quite different from those of a crane driver which in turn are quite different from those of a technician.

Now it is necessary to generalise this concept in order to make it applicable to all systems.

Assume that all the interactions of the system have been classified into categories I_1 I_2 I_3 ... I_n. Let the total number of entities in the system be $m : X_1, X_2, \ldots, X_m$. Every catalyst X_i is involved in a number of interactions. These can be classified into categories I_1 I_2 ... I_n.

Thus to every catalyst there corresponds a set of categories $I_{i1}, I_{i2}, \ldots, I_{in}$ which represent the types of interactions in which the catalyst is involved.

It will probably turn out that many catalysts have the same set of interaction categories. We now partition all the entities constituting the organised system into partitions of catalysts which are involved in the same interaction categories. Each of these partitions we call a speciality.

Thus we started with a classification of interactions and arrived at a classification of entities.

Example:

Consider the following classification which will be examined in chapter 4.

Input processes
Transport processes
Output processes
Storage processes
Etc.

If the entities labour and capital within the social system are considered, then it is possible to associate with each entity a set of interaction categories.

For example: John Smith is crane driver (transport);
George Blogg is a businessman (info processing);

Thus the well-known specialities of labour and capital will be arrived at:

Managers	Farming Equipment
Engineers	Transport Equipment
Artisans	Communication Equipment
Farmers	Storage Equipment
Etc.	Etc.

It is obvious that the classification can vary depending the initial interaction and the resolution level. However this is an algorithm for comparing systems on the same fixed frame of reference. For example comparing two countries with respect to their level of specialisation as in section 4.1.

LEVEL OF SPECIALISATION: In order to compare organised systems with respect to specialisation let us define the following concept:

$$\text{Level of specialisation} = \frac{\text{Number of specialities}}{\text{Total number of entities}}$$

The lowest level of specialisation possible occurs when the system defined, consists of entities which perform the same function. For example a diamond may be seen as consisting of elements (carbon atoms) which are involved in the same type of interaction.

$$\text{Level of specialisation} = \frac{1}{\text{No. of atoms in crystal}}$$

The highest level of specialisation would be a system in which every entity performs a very special function which is different than that of the others.

In that case:

Number of specialities = Number of entities

Level of specialisation = 1

to an example of comparing two systems as to the level of specialisation, a sodium chloride crystal can be considered which contains as many atoms as there are in a cell. The specialities in the NaCl crystal under any classification could not be more than two. In the cell however there are more: function of RNA, function of enzymes, function of ribosome, etc. Therefore the cell is at a higher level of specialisation than the NaCl crystal.

The above should be considered as an indication that it is possible to construct a method of measurement of specialisation.

SUBSPECIALISATION. The above applies equally well to a particular process in an interaction network, as it applies to the total network itself. If we identify a subnetwork which is for example, the input process of a self-maintaining organised system, it is possible that this subnetwork may be further specialised to a finer degree and contains its opponents and specialised subprocesses. Thus specialisation can be seen within a hierarchic framework, where each process may have further specialised components.

DIFFERENTIATION. The structural manifestation of specialisation may be termed "differentiation of components". When different processes and the components involved are spatially separated rather than diffuse, thus forming distinguishable structures, then the components of the system are differentiated. The different organs of an animal are an example.

CHAPTER 4

ANALYSIS OF INTERNAL STRUCTURE AND PROCESSES

4.1 INTRODUCTION

This chapter will identify and examine various types of processes taking place in self-maintaining organismic systems. The component entities taking part in these processes will also be considered and an example is given from physical entities in which these occur.

The following format of presentation will be adopted:

A. General model: A description of the characteristics of the process and the entities involved.

B. Examples from cells.

C. Examples from organisms.

D. Examples from societies.

A point that has to be borne in mind is that entities differ in their level of specialisation and differentiation as defined in section 3.7. Thus while the types of processes which are considered in the model exist in all entities, certain subsystems are more differentiated than others. A process which is performed in one system by a number of components which are present in all parts of the system, is performed in another system by a specific specialised structure.

4.2 INPUT PROCESSES AND OUT GATES

MODEL

An input process is one which contains in its initial time slice entities of the environment, which at the final time slice appear as entities of the system, without necessarily having undergone any structural changes. The catalysts of this process will be called the input gate of the system (fig. 4.1).

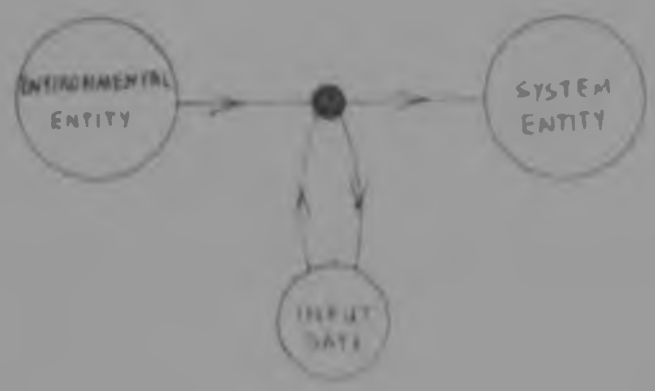


FIG. 4.1. Input gate (model)

Note that the distinction between an environment entity and a system entity depends on the definition of the system.

CELLS

Example: Cell membrane.

Certain molecules outside the cell interact with the cell membrane resulting in these molecules being admitted into the cell and taking part in the metabolism.

The following diagram presents the function of the cell membrane as seen in biochemistry (fig. 4.2). (Orten & Neuhaus [8]).

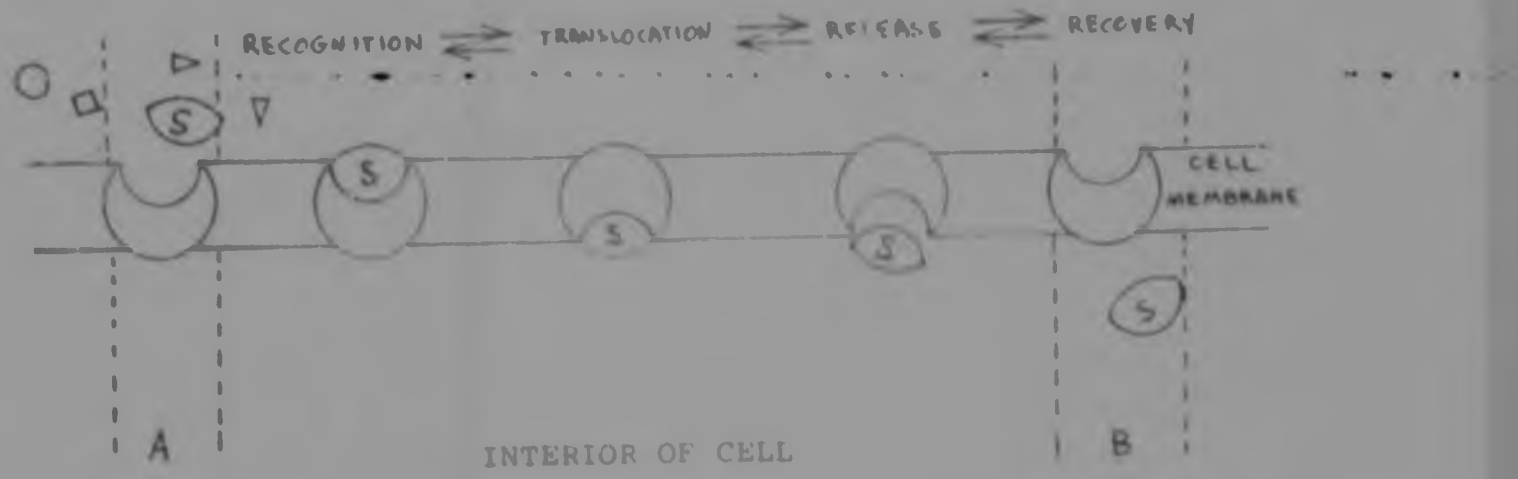


FIG. 4.2. Input gate (cell)

ORGANISMS

Example: Mouth
Nose

Before an apple is eaten it is considered as part of the environment of the animal. From the moment it is introduced into the mouth and chewed it is part of the organism.

SOCIETIES

Example: Mines
Dams
Farms

A mine is an interaction site where the set of entities that constitute the mine (people, machines, etc.), interact with the ore which up to this point in time is considered as part of the environment. This interaction results in the ore becoming part of the socio-economic system (e.g. being transported in a truck to a processing plant).

A farm is an interaction site where environmental entities such as photons from the sun, CO_2 molecules from the atmosphere, and minerals from the soil interact in an input process (involving farmer, tractors, seeds, etc.) to become vegetables (part of society).

4.3 THE TRANSPORT PROCESS
THE TRANSPORT SUBSYSTEM

MODEL

A transport process is one which results in a change of location of the entity being transported. The collection of catalysts in the transport process is the transport subsystem (fig. 4.3).



FIG. 4.3. Transport subsystem (model)

CELLS

Entities in a cell move mainly by means of random collisions therefore a great number of entities act as catalysts of the transport process.

ORGANISMS

Examples: Cardiovascular system
Trachea, Bronchioles
Ureters

The cardiovascular system is a set of entities (veins, arteries, capillaries and heart) which interact with the blood resulting in the change of location of each volume of blood.

Trachea and bronchioles are part of the set of entities which result in air being transported from the nose and mouth (input gates) to the alveoli of the lungs.

Ureters convey the urine from the kidneys to the urinary bladder.

SOCIETY

Examples: Roads
Aeroplanes
Ships

Interactions of entities such as trucks, cars, and people with roads, results in the change of location of these entities.

Aeroplanes and ships interact with passengers and cargo resulting in changing of their location.

4.4 STORAGE PROCESSES
STORAGE SUBSYSTEMS

MODEL

A storage process is an interaction network which results in maintaining certain entities within an area of space where they can interact with other entities of the system at a later time. The set of catalysts involved are the storage subsystem (Fig. 4.4).



IN STORAGE AREA

FIG. 4.4 Storage subsystem



IN RESERVOIR

FIG. 4.5 Water reservoirs

CELL

Examples: Nucleolus

This is the storehouse of mRNA prior to movement of the latter into the cytoplasm.

ORGANISMS

Examples: Urinary bladder
Fat deposits of the body

The bladder stores urine, i.e. interacts with it resulting in space localisation.

SOCIETY

Examples: Oil storage tanks
Water reservoirs

Many large and small pieces of equipment like the above are used for storage of commodities for later use (fig 4.5).

4.5 METABOLISM
METABOLIC SUBSYSTEM

MODEL

An interaction which produces some fundamental change in the entities involved will be defined as a metabolic interaction. This would be the case when some entities exist in the initial but not in the final time slice, and vice versa.

Within the organised system there is a chain of metabolic interactions which changes environmental entities into different system entities. This is the metabolism of the organised system.

The following types of metabolic interactions can be distinguished in an organised system:

(1) Interactions of which some reactants have not undergone any structural changes since they entered the organised system (fig. 4.6 A).

(2) Interactions whose reactants are products of other metabolic interactions, and whose products are reactants of other metabolic interactions (fig. 4.6 B).

(3) Interactions whose products do not undergo any further structural changes before they are released to the environment (fig. 4.6 C).

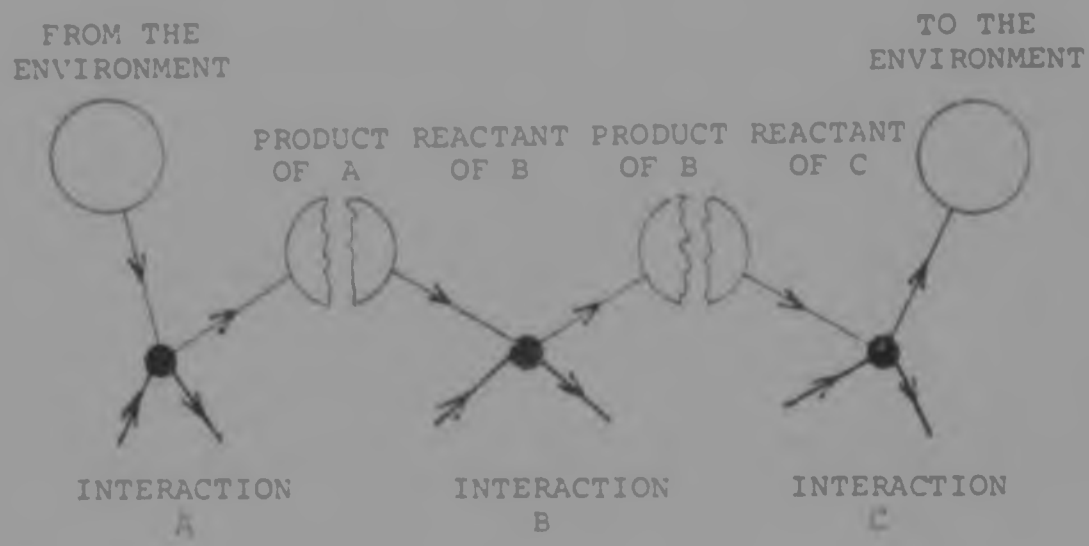


FIG. 4.6. Metabolism (general model)

CELLS

Example: Enzymes and metabolism

The various enzymes interact with the molecules of precursors (raw material) resulting in molecular changes thus producing intermediates and final products.

The metabolism of the cell is a network of biochemical reactions.

ORGANISMS

Examples: Mouth
Stomach
Intestine

In the mouth the environmental entities (food) are modified by chewing and mixing with saliva. The product of that interaction in turn interacts with the stomach (muscles and gastric juices) to produce a different entity (the digested food). This product interacts with the intestines which separates it into the substances which are absorbed in the bloodstream and the waste which is released to the environment (fig. 4.7).

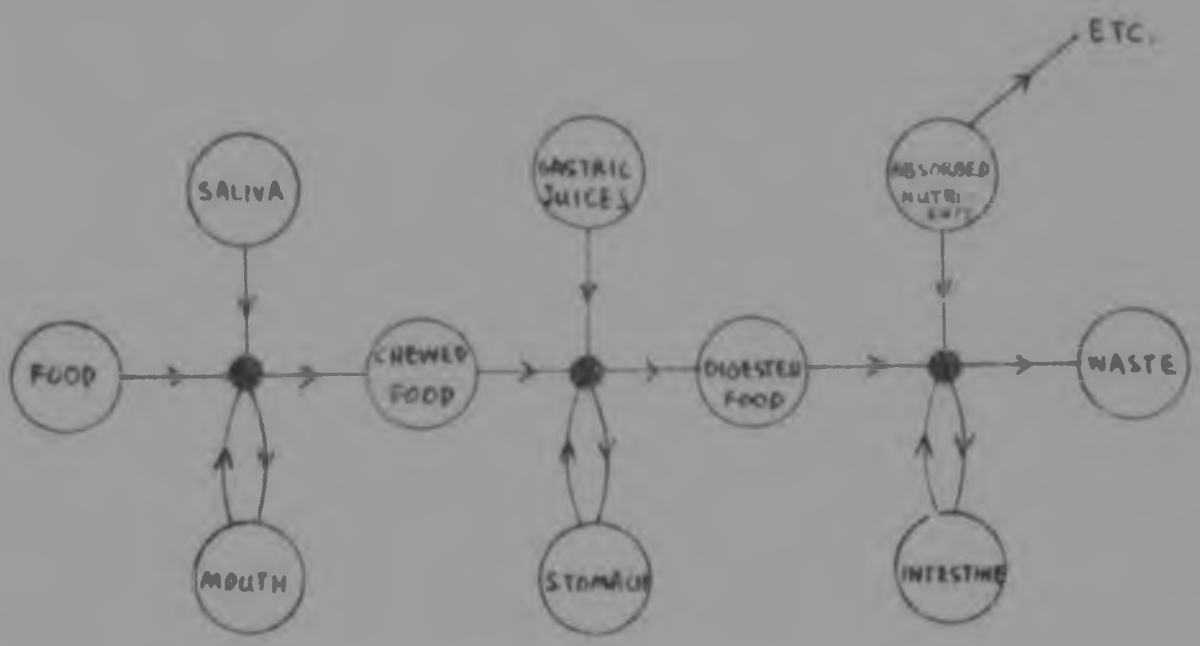


FIG. 4.7. Metabolism (organisms)

SOCIETY

Examples: Industry
Farming

A whole chain of interactions of the factors of production (land, labour, capital) processes materials from the environ-

ment to produce consumer goods or producer goods. Consumer goods interact with people to produce satisfied people and waste. The producer goods replace obsolete capital or add to the existing capital (tools, reserves etc.). This is the network of interactions which constitutes the metabolism of this organised system known as industry, farming, etc. (i.e. all productive + consumption) (fig. 4.8).

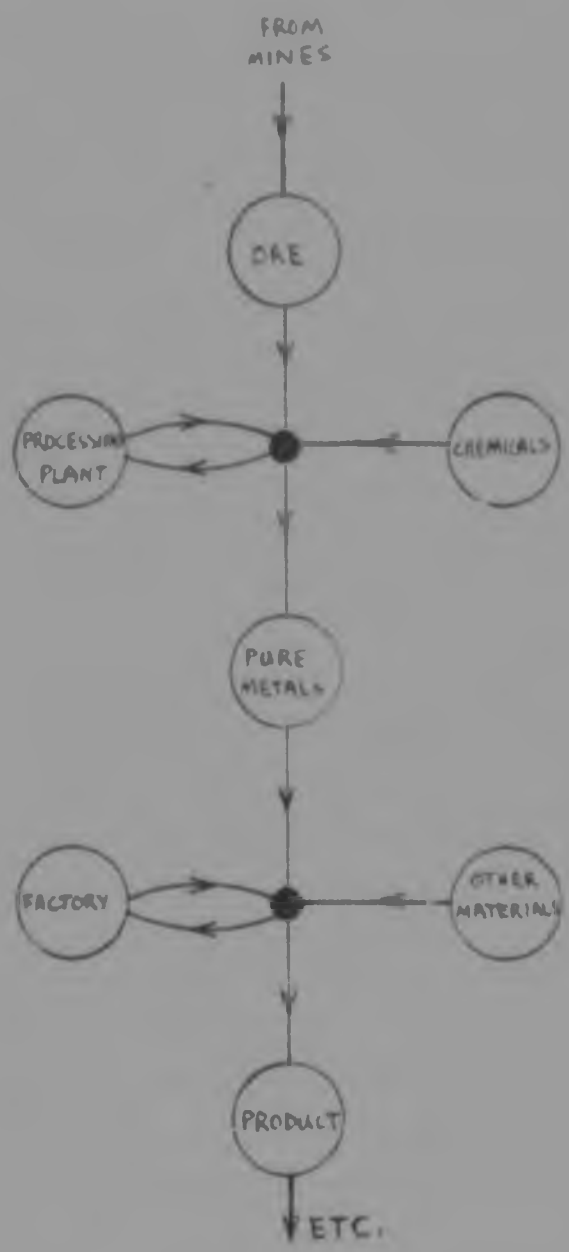


FIG. 4.8. Metabolism (society)

Following are examples of some aspects regarding the metabolism of a system.

STEADY STATE, GROWTH, DECAY:

If the rate of the following interaction:

Catalysts + Materials → Catalysts + Products

is equal to the rate of the interaction:

Catalyst → Waste

then the system will remain in a steady state. If not, then a growth or decay of the quantity of entities in the system occurs.

EXAMPLES:

CELLS: During growth of the cell the enzyme production is higher than the disintegration of enzymes.

ORGANISMS: During growth of an organism the number of cells produced and maintained in the tissues is greater than the decay rate

SOCIETY: When production is greater than consumption, then the extra unconsumed products are extra capital being available as a catalyst for more production. This is the case of a growing economy.

THE MULTIPLIER EFFECT:

Every interaction in a self-maintaining cyclic process will not take place unless the necessary reactants are present in the interaction site.

It thus happens that although the main ingredients of a particular interaction are present, the absence of a necessary entity results in the slow rate, or non occurrence of an interaction. As every interaction in the self-main-

taining cyclic process is linked to other interactions, this results in the whole process running at a low rate due to the lack of a particular entity.

EXAMPLES:

CELLS: Deficiency of some minerals which are .. necessary for the functioning and growth of the cell, does not allow normal development and growth.

ORGANISMS: Vitamin deficiency: This is a well-known phenomenon where the functioning of the entire organism is hindered by the deficiency of certain substances. In injection of a small quantity of these vital substances into the metabolism of the system, resu- t in rApid development of the organism to a state of normal functioning.

SOCIETY: An injection of capital or the right type of skilled labour at the appropriate point into the economy, can result into a many-fold advance in the economy. This is due to the fact that idle factors in the economy have now become productive.

LEVEL OF METABOLISM:

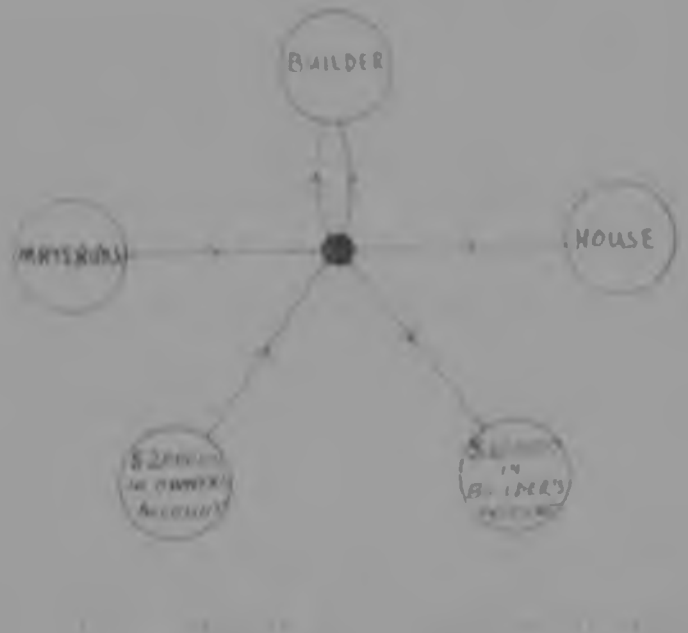
This is a concept of the measure of the activity of the self-maintaining cyclic process. Among the possible formulae that can be used the following is suggested here:

Level of metabolism = quantity of products produced in the cyclic process per unit time.

A problem arises here out of the fact that there are many types of entities produced. What is needed is a method of

mov

of money is
ons each of which
For example
a builder will
$200 000 (note
illustrated



EXAMPLES OF LEVELS OF METABOLISM

CELLS [illegible] unit time.

ORGANISM: As in cells the level of metabolism [illegible] isms can be measured in units of energy per unit time.

SOCIETY: GNP, the gross national product, is a measure of all economic interactions that [illegible] during the period of a year.

4.6 OUTPUT PROCESSES
OUTPUT GATES

MODEL

Similarly to the input processes, this is the [illegible] interactions which result in system [illegible] released to the environment. The [illegible] cess are the output [illegible] (Fig. 4.10).

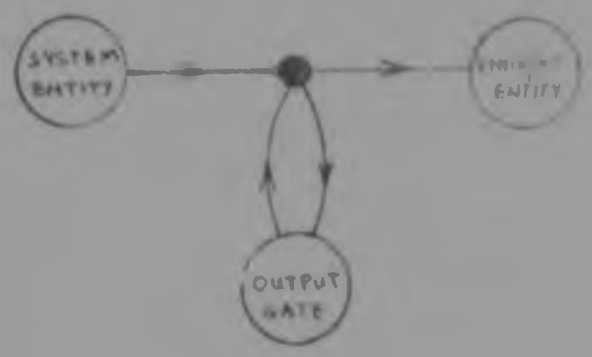


FIG. 4.10. Output gates [illegible]

CELLS

Example: Membrane

As in the process of capturing nutrients from the environment, waste products of the cell's metabolism are released

into the environment (fig. 4.11).

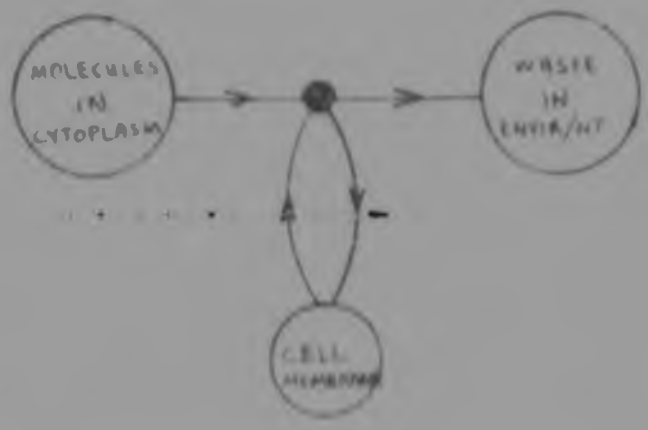


FIG. 4.11. Output gates (cell)

ORGANISMS

Examples: Pores of the skin
Ureter

Waste and other substances (e.g. sweat) are released to the environment by special structures in the anatomy of the organism (fig. 4.12.).

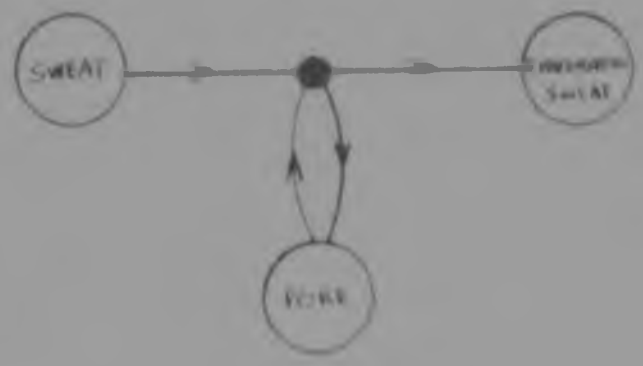


FIG. 4.12. Output gates (organisms)

SOCIETY

Examples: Chimneys
Harbours

At various points in the network of interactions of society there are entities leaving the society to become part of the society's environment.

The smoke being released by chimneys for example, becomes part of the atmosphere which is considered as part of the environment of society.

Waste rejection is of course only part of the output processes. Other countries are part of a society's environment. Therefore goods transport to other countries is an output process. If entities outside the border of a country are considered as part of its environment, then harbours, airports, etc. are output gates of the society (fig. 4.13).

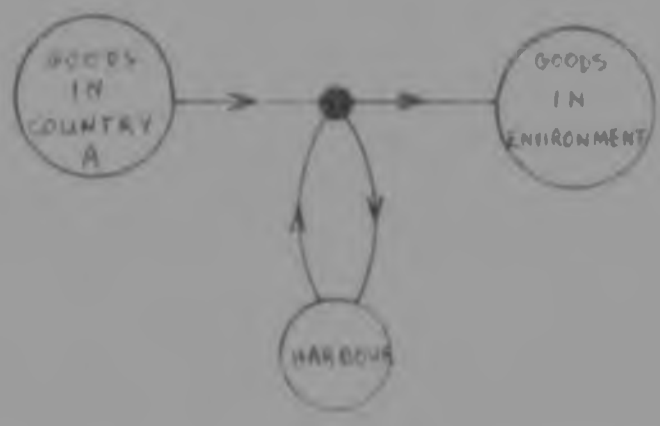


FIG. 4.13. OUTPUT GATES (society)

4.7 INFORMATION PROCESSES
INFORMATION SUBSYSTEM

MODEL

The distinction between communication and information processing is a matter of degree of complexity of the interaction network of this subsystem.

Communication processes are those interaction networks which effect information transfers over relatively long distances by means of a network of information channels. This makes it possible for certain interactions taking place at an input site of the network to result in specific interactions taking place at a certain output site, which

is relatively far in space. The interaction network constituting the communication process is relatively simple.

Information processing on the other hand is a relatively complex interaction network of information transfers. Relatively short distance information transfers take place from each entity to several others and from several entities to one. As in the communication subsystem it has input and output sites. However, the relationship between interactions at an input site and those taking place at an output site is not a one to one correspondence, hence the idea that the information has been processed, i.e. changed.

CELLS

Example: Messenger RNA

An RNA molecule of specific structure is produced by the interaction of DNA with the components of the RNA. The RNA molecule then travels to an interaction site (information transfer by transportation) where it catalyses the production of a specific enzyme (protein).

The phenomenon as seen by biochemistry is shown in fig. 4.14.

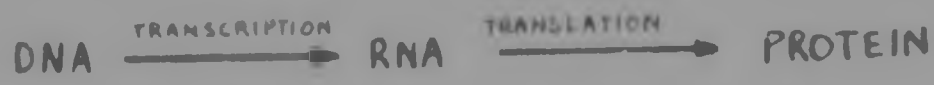


FIG. 4.14. Internal communication in cells (biochemistry)

In terms of our interaction networks this is shown in fig. 4.15.

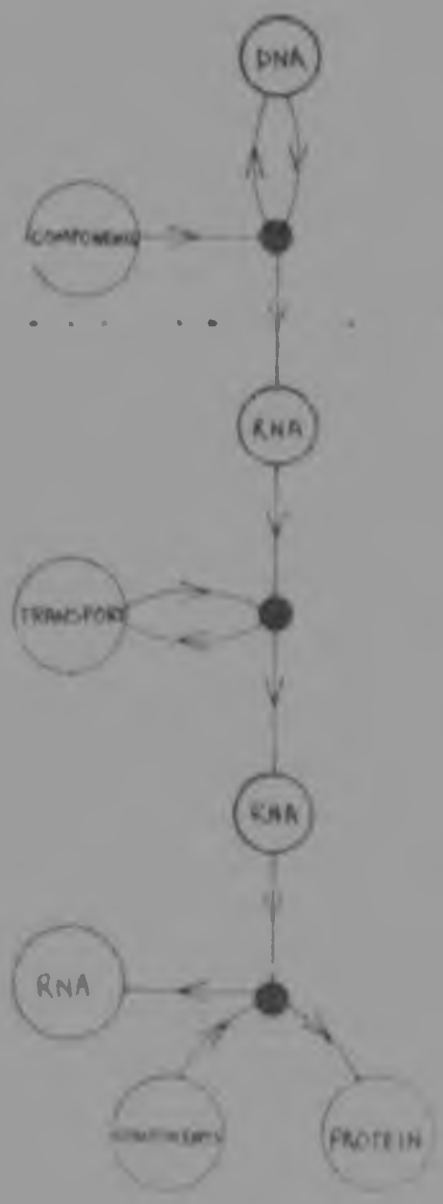


FIG. 4.1[illegible]. Internal communication in cells (as an interaction network)

ORGANISMS

Examples: Endocrine system
Peripheral nervous system

The endocrine system effects communication by transportation. A certain interaction (for example, the function of the pituitary gland) produces a hormone which is trans-

ported by the transportation system (cardiovascular) to other interaction sites (for example other hormone glands). At these sites other hormones are produced which travel further to interact with tissues and result in changes in the states of the relevant organs (for example faster heart rate).

Peripheral nerves on the other hand effect communication by transmission. For example an interaction in a certain part of the spinal cord results in a nerve impulse which via an interaction chain, as that described in fig. 2.28 ultimately results in the interaction between a nerve cell and the cell of a skeletal muscle. This causes a contraction, i.e. a change of state of the muscle.

Information processing takes place in the brain where a complex interaction network exists between neurons in a relatively small volume compared to the distances involved in the peripheral nervous system.

SOCIETY

Examples: Telephone (communication)
Post (communication)
Office block with its contents (information processing)

The post and telephone are part of the communication subsystem of society (by transportation and transmission respectively). In all cases an interaction at an input site of this process results in a corresponding interaction at some output point of the process.

An office block in a city with all its contents (including people) is an example of part of the information processing subsystem of society. Here people talk to each other, read, write, listen to the telephone etc. This is a complex network of information transfers which involve changes of mental states of people.

NOTE: It is beyond the scope of this investigation to study the detailed structure of these subsystems, for example the hierarchical organisation of complex information processing, etc.

CHAPTER 5

INTERACTION WITH THE ENVIRONMENT

5.1 INTRODUCTION

In the previous two chapters the organised system has been analysed and the main processes and entities involved were investigated. The integration of these processes was then considered. It is possible now to explore the subject of the interaction of the organised system as a whole with its environment. The information and communication subsystem co-ordinates the various entities of the organised system so that the whole interacts with the environment in such a way so as to maintain its identifying variables (i.e. survive). The following are some aspects of the interactions between the organised system and the environment.

5.2 NEED SATISFACTION

The whole self-maintaining cyclic process of an organised system depends on:

(1) The intake of raw materials from the environment to replace the constantly disintegrating entities of the system.

(2) The integrity and stability of the interaction network, so that a large critical part of the self-maintaining cyclic process is not suddenly destroyed as a result of a fatal interaction with the environment.

This implies that the right interactions between system and environment must take place in order that the organised system survives. The details of these interactions depend on the particular system and its particular environment.

The obtaining of raw materials can range from simple selective intake to complex processes of locomotion by means of which the system moves in its environment in order to come into contact with the appropriate reactants in the environment.

EXAMPLES

CELLS: Nutrients are taken into the cell by passive selective intake of the appropriate substances.

ORGANISMS: Interaction of the receptors (a specialised part of the communication subsystem) with the environment, followed by information processing in the nervous system, results in the effectors (muscles) interacting with the environment (ground, branches, etc.), so as to move the system and/or the environment until food (or any other requisite reactant) is in contact with the required part of the system (in this case the mouth). For example a tiger pursuing its prey, a man digging a well to obtain water.

SOCIETY: An exploration team moves in the desert looking for oil. Samples are sent to the laboratories, reports are submitted to the decision makers, decisions are made, detailed engineering plans are drawn, equipment and men are sent to the site, and finally the oilfields (consisting of men and equipment) start to draw oil from the environment.

On the other hand ensuring the integrity of the system can range from simple avoidance of certain environmental conditions, to active seeking of favourable conditions, to structuring of the environment so as to make conditions favourable.

EXAMPLES:

CELLS: Amoeba moves away from toxic environment.

ORGANISMS: Escapes from danger or seeks shelter or constructs shelter. All these are performed as a sequence of receptor-environment interaction, information processing, and finally effector-environment interaction.

SOCIETY: A weather station interacts with the atmosphere (receptor-environment interaction). An interaction of technicians, instruments and decision makers follows, constituting the realisation of sensor (information processing). Finally preparation for a hurricane take place.

Walls of castles are BUILT to protect society from wild animals or foreign armies.

5.3 SURVIVAL MARGINS

For the appropriate interactions to take place, the required reactants must be present in the interaction site. In the present study there are two broad reactants involved: the organised system and the environment. The previous sections studied the role of the organised system in the system-environment interaction. These were a sequence of system states which were required for the satisfaction of needs. The environment however being the second reactant must also have the required states, properties or entities. Thus an organised system can proceed with the satisfaction of needs only

within a certain set of environmental conditions, beyond which it cannot survive.

A set of environmental conditions can thus be associated with any organised system, and defined as the survival margin of the relevant system.

EXAMPLES:

CELLS: A certain concentration of nutrients, range of temperatures, absence of poisonous materials, etc. constitute the survival margin of a cell.

ORGANISMS: Availability of a minimum quantity of food, oxygen, noise level, and a certain level of stimulation of the brain are some of the characteristics of an organism's survival margin.

SOCIETY: Minimum and maximum of sun, rain, wind, minerals, temperature, etc. characterise a society's survival margin.

The survival margin could be a criterion for comparing and studying organised systems. For example two societies (a technologically advanced one and a primitive one) could be compared and relationships could be established between organisational structure and survival margin.

As a second example organised systems of different levels (i.e. cells organisms societies) could be compared and the conclusion drawn that as one moves up the hierarchy the survival margins widen, from an aqueous solutions of specific concentrations, to the human society's ability to survive in deep sea, in the desert and in space?

within a certain set of environmental conditions, beyond which it cannot survive.

A set of environmental conditions can thus be associated with any organised system, and defined as the survival margin of the relevant system.

EXAMPLES:

CELLS: A certain concentration of nutrients, range of temperatures, absence of poisonous materials, etc. constitute the survival margin of a cell.

ORGANISMS: Availability of a minimum quantity of food, oxygen, noise level, and a certain level of stimulation of the brain are some of the characteristics of an organism's survival margin.

SOCIETY: Minimum and maximum of sun, rain, wind, minerals, temperature, etc. characterise a society's survival margin.

The survival margin could be a criterion for comparing and studying organised systems. For example two societies (a technologically advanced one and a primitive one) could be compared and relationships could be established between organisational structure and survival margin.

As a second example organised systems of different levels (i.e. cells organisms societies) could be compared and the conclusion drawn that as one moves up the hierarchy the survival margins widen, from an aqueous solutions of specific concentrations, to the human society's ability to survive in deep sea, in the desert and in space?

5.4 ENVIRONMENTAL CHANGE

As the environment changes within the survival margin, the system's behaviour changes, i.e. the state of the system changes thus producing different interactions with the environment. This results in the maintenance of stability of the identifying variables. This change of behavioural state can be called a response to the environment.

The process of responding to the environment can be described by the loop of fig. 5.1.

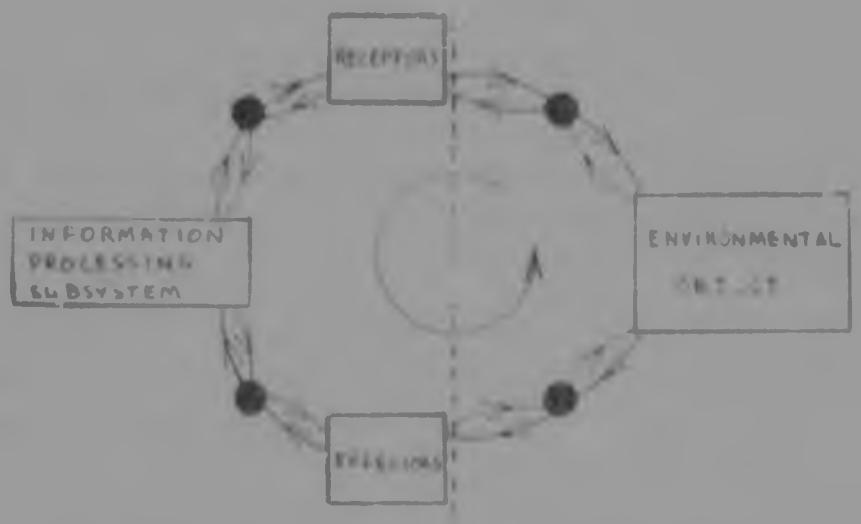


FIG. 5.1. Response to the environment

5.5 A NOTE ON REPRODUCTION

The organised system has been considered as an isolated system in an environment, performing the appropriate activities to ensure its own survival. There is one need however (and the corresponding set of interactions) namely that of reproduction, which is not connected with the survival of the system. This is related with the

fact that an organised system of certain complexity has to be initially produced (born), as well as produce another similar system, and the fact that it is not static but rather that there is a developmental change from birth to maturity to death.

As this dissertation investigates only the functions of the organised system at a certain interval in time (see section 3.2) and it is beyond its scope to study developmental changes, the needs of reproduction have been omitted from consideration since they are needs concerned with the development (birth-death) of the system. It should be borne in mind however as a need, and its satisfaction follows the same principles as the other needs.

5.6 THE ORGANISED ENVIRONMENT

In the previous sections the interactions of an organised system were considered. The only assumption about the environment was that it fluctuates randomly within (or out of) the survival limits while the organised system changes behavioural states as to maintain its identifying variables.

An environment which does not fluctuate randomly but is itself an organised system, can now be considered. Cells have been seen as independent organised systems in an arbitrary environment. Now their behaviour within an organised system (an organism) can be considered. The organism has also been considered within an arbitrary environment. Now it can be considered within a society, i.e. within an organised environment.

Thus having investigated the common characteristics of structure and behaviour of cells, organisms, and societies, the common characteristics in the cells-organism and organisms-society relationships can now be investigated.

5.7 MICRO-MEGA NETWORK RELATIONSHIPS

The statement that an organised system is within an organised environment means that the environment can be described as a network of interactions which possesses the characteristics of an organised system.

Within this mega-network of interactions the organised system is an entity characterised by changes of state, as a result of interaction with other entities of the organised environment. For example an organised system with two states can be described as in fig. 5.2. When in state S_1 interaction with environmental entity (or set of entities) B results in a change to state S_2. When in state S_2, interaction with entity A results to state S_1.



FIG. 5.2. Interaction with entities of an organised environment

Here the entity S_1 and S_2 is a network of interactions itself and the changes of state $S_1 \longrightarrow S_2$ and $S_2 \longrightarrow S_1$ are responses to the environment B and A respectively.

Thus control of a component of a megasystem can be achieved by controlling the immediate environment of that component. In order to maintain its identifying variables the component will then respond by changing to the desired state.

EXAMPLES:

Cell-organism relationship:

Interactions in the brain of an animal result in the movement of a muscle cell. The processes inside the cell which resulted in the contraction of the muscle cell were not directly controlled by the brain. They occurred as a response of the muscle cell to its environment, i.e. the nerve impulse on its membrane. The brain does not control the metabolism inside the cells of the organism but rather controls the environment of these cells. By changing these environments the internal interactions of cells are indirectly controlled.

Organism-society relationship:

A foreman instructs a worker to perform certain tasks and as a result the worker moves a muscle. Here again the foreman sets the environment of the worker. The worker in turn controls the interactions in his body, i.e. move the required muscles in order to do the job.

5.8 MICRO-MEGA NEEDS

If the mega-system investigated in the previous section survives and is not in the process of disintegration, then its needs can be traced to the needs of the micro-systems which constitute it. As regards the input which each micro-system receives from its organised environment (which is the mega-system), it has to ultimately in some (mostly different) form be obtained from the environment of the mega-system.

EXAMPLES:

Cell-organism relationships:

The nutrients required by the cells in an organism which

are obtained by interacting with entities contained in the blood, represent the needs of the microsystem "cell". These are derived via a network of interactions of the mega-system from food, oxygen, sunlight, etc., which are the needs of the mega-system.

Organism-society relationships

A simple need of an individual in society is food. This ultimately comes through a long network of interactions (industry, transport, etc.) from sunlight (on the fruit trees) rain (to fill the dams, to make electricity, to fry the bacon) etc., which are the needs of society.

Thus the needs of the mega-system reflect the needs of the micro-systems but are different, since the mega-system interacts with an environment which is very different than that of the micro-systems.

5.9 RELATIONSHIP BETWEEN ORGANISATIONAL COMPLEXITY AND MICRO-MEGA NEED DIFFERENTIAL

In an organised system of low complexity, the mega-system needs are similar to the micro-system needs, while a highly complex mega-system has needs which are very different from the needs of the micro-systems in it.

This is apparent from the consideration that there is a limited number of transformations that a low complexity system can achieve (by definition of complexity), therefore its needs are similar to the micro-system needs.

A highly complex organised system on the other hand, can achieve complex transformations, and can therefore survive in an environment which is totally alien to the micro-systems. This is because the large network of interactions makes possible the change of totally alien interactions with the environment to be transformed to need satisfaction of the micro-systems.

EXAMPLES:

Cell-organism:

The sea sponge is an organised system of low complexity and lives in the same environment as single cells (i.e. the sea water). Its input and generally its need satisfaction is similar to that of the single cells contained in the sea sponge.

A complex organism on the other hand can live in the desert under the hot sun, where its cells could not survive, and eats food which undergoes many changes before it is transformed into nutrients acceptable by the cells.

Organism-society:

A primitive society lives in a kind of environment which is similar to that of single individuals. Its needs (meat, wood, etc.) are very much the same as those of individuals.

A complex society however has needs such as coal, oil, uranium, which are quite different from the needs of individual members of society.

EXAMPLES:

Cell-organism:

The sea sponge is an organised system of low complexity and lives in the same environment as single cells (i.e. the sea water). Its input and generally its need satisfaction is similar to that of the single cells contained in the sea sponge.

A complex organism on the other hand can live in the desert under the hot sun, where its cells could not survive, and eats food which undergoes many changes before it is transformed into nutrients acceptable by the cells.

Organism-society:

A primitive society lives in a kind of environment which is similar to that of single individuals. Its needs (meat, wood, etc.) are very much the same as those of individuals.

A complex society however has needs such as coal, oil, uranium, which are quite different from the needs of individual members of society.

CHAPTER 6

MODELS OF INTERACTION NETWORKS

6.1 INTRODUCTION

In chapter 2 the basic conceptual tools were described. The concepts of entity, interaction, and interaction network were chosen so that they can be used in describing phenomena from different disciplines. In chapters 3, 4 and 5 a general indication was given of how this approach can be used in expressing features which are common to self-maintaining organised systems of different physical nature (cells, organisms, societies).

In this chapter the technique of modelling which is suitable for this approach will be presented. This will be followed in the next chapter by a detailed example of the application of this technique.

6.2 MODELS

In order to compare different systems or postulate features which are common to a class of systems, it is desirable to have a medium for representing these specific systems, as well as any theoretical postulates about them, with something which is clear, unambiguous and can be examined repeatedly and by different investigators.

A concrete model such as a computer model has the above properties.

Once a phenomenon has been described as an interaction network, such a concrete model of the phenomenon can be

constructed. Some aspects of the model building process are discussed in the following sections.

6.3 COMPUTER MODELS OF INTERACTION NETWORKS

Having identified the entities and interactions in a network, the changes in time can be studied by means of a computer model, formulated as follows:

The state of each entity i in the network is represented by a vector $(x_{i1}, x_{i2}, \ldots, x_{im})$. A vector is used since in the general case the state of an entity can be multi-dimensional. The variables $x_{i1}, \ldots, x_{im}$ are discrete. For example if an entity has a one-dimensional state space and it has four identifiable states in the given resolution-level then it can be represented by $x_i = 1,2,3,4$. In another case the only interest in a certain entity may be its existence or non-existence. That is, under the given resolution level, different states of the entity are of no interest or are not distinguishable. In this case the variable x_i representing entity i can have values 1 and 0 (exists, does not exist).

A computer model of the interaction network can be constructed by a program which iteratively calculates the changes to the states of the entities as one moves through a sequence of time slices.

The output of such a program reports the states of the entities at successive time slices as the example in fig. 6.1.

6.4 THE BASIC ALGORITHM

There are two basic steps for each iteration:

Considering the concept of interaction as a sequence of three time slices as shown in fig. 2.8 and 2.9 and in fig. 6.2 below, it becomes apparent that one needs

to do two computations for each iteration.

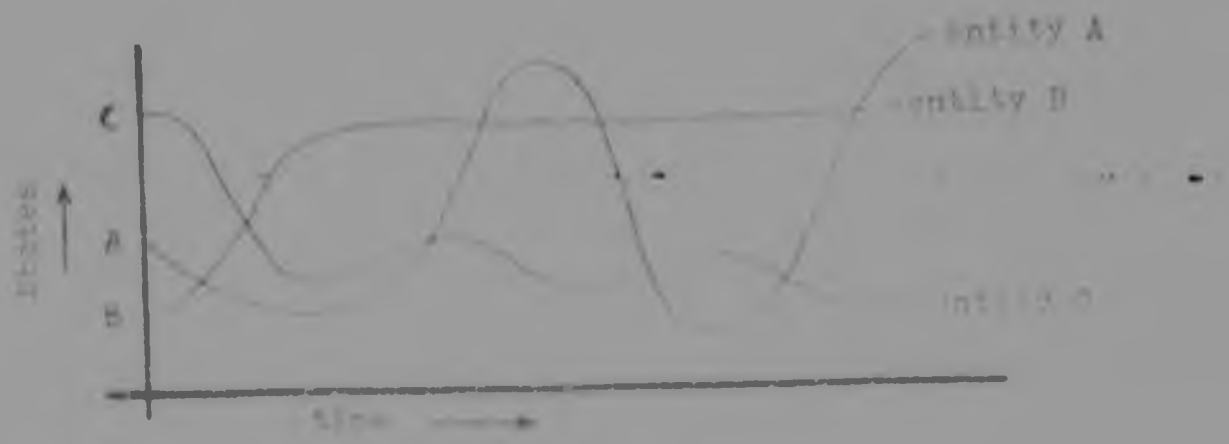




Fig. 5.1. Interaction affected by a number of entities

The interaction mode characterises the type of change which is taking place.

2. Once the type of changes taking place have been identified (in step 1 above), the change in the state of each of the entities can be calculated. In doing this, each entity is considered independently, and the change of state is calculated based in the modes of the interactions in which this entity is involved (fig. 6.4).



FIG. 6.4. Entity affected by a number of interactions

Fig. 6.5 below shows the general algorithm which can be used to implement the above.

Steps 1 and 2 can be calculated in terms of any arbitrary algorithm given by the study of the phenomenon in the special field of study.

6.5 STEPS FOR CONSTRUCTING A MODEL

In order to construct a model of a phenomenon it is necessary to proceed through a sequence of steps. This is illustrated in the next chapter during the modelling of a production/consumption phenomenon. In the following sections of this chapter some of the main considerations involved in these steps will be briefly discussed.

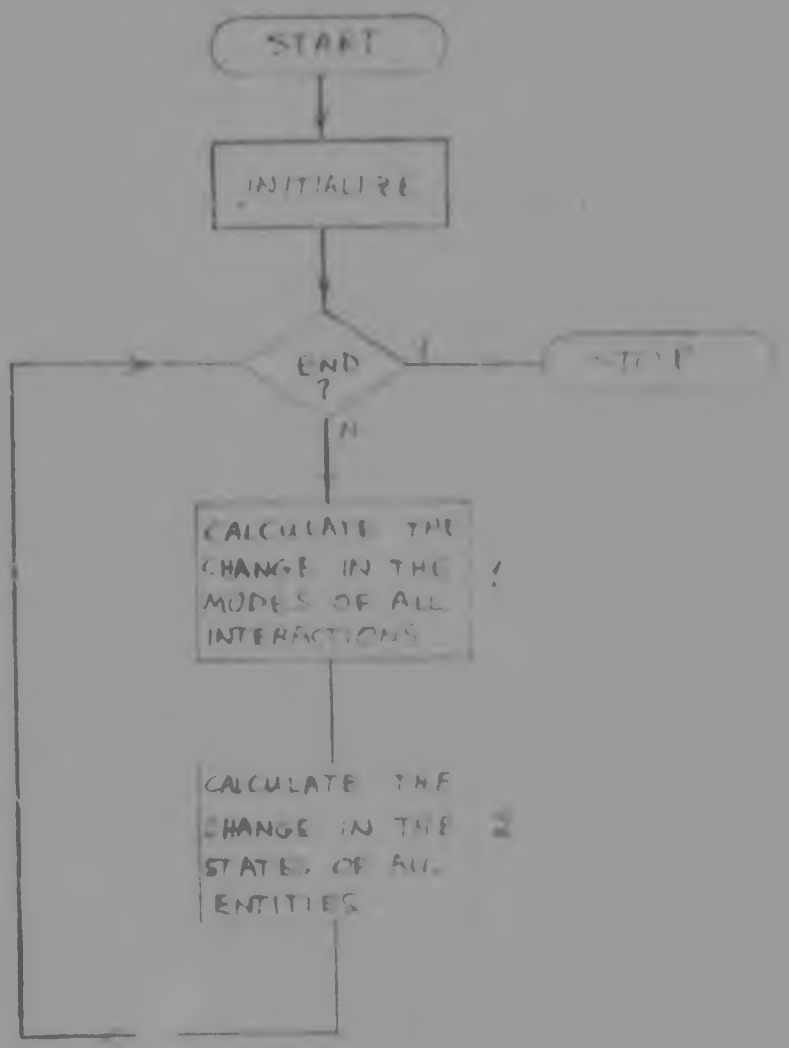


FIG. 6.5. General algorithm of the interaction network model

The main considerations are:

1. Analysis at the right resolution level.

2. Temporal analysis of the network.

3. Information flow aspect and the model diagram.

6.6 ANALYSIS AT THE RIGHT RESOLUTION LEVEL

In defining an interaction network or a phenomenon, and subsequently constructing a model, it is expedient to start at the coarsest resolution level and progressively through finer levels to the optimum resolution level. By "optimum resolution level", we mean that if the resolution level was coarser, then essential information needed for the representation of aspects of the phenomena, would be lost. If it was finer, then effort is lost in processing redundant information.

At every step from the coarsest to the optimum resolution levels, only the entities and the interactions which are of interest are retained.

Consider the interaction network below (fig. 6.6). Each entity is of interest, i.e. the raw materials available, the state of the PRODUCTION catalyst (e.g. production capacity), the quantity of products which are available.

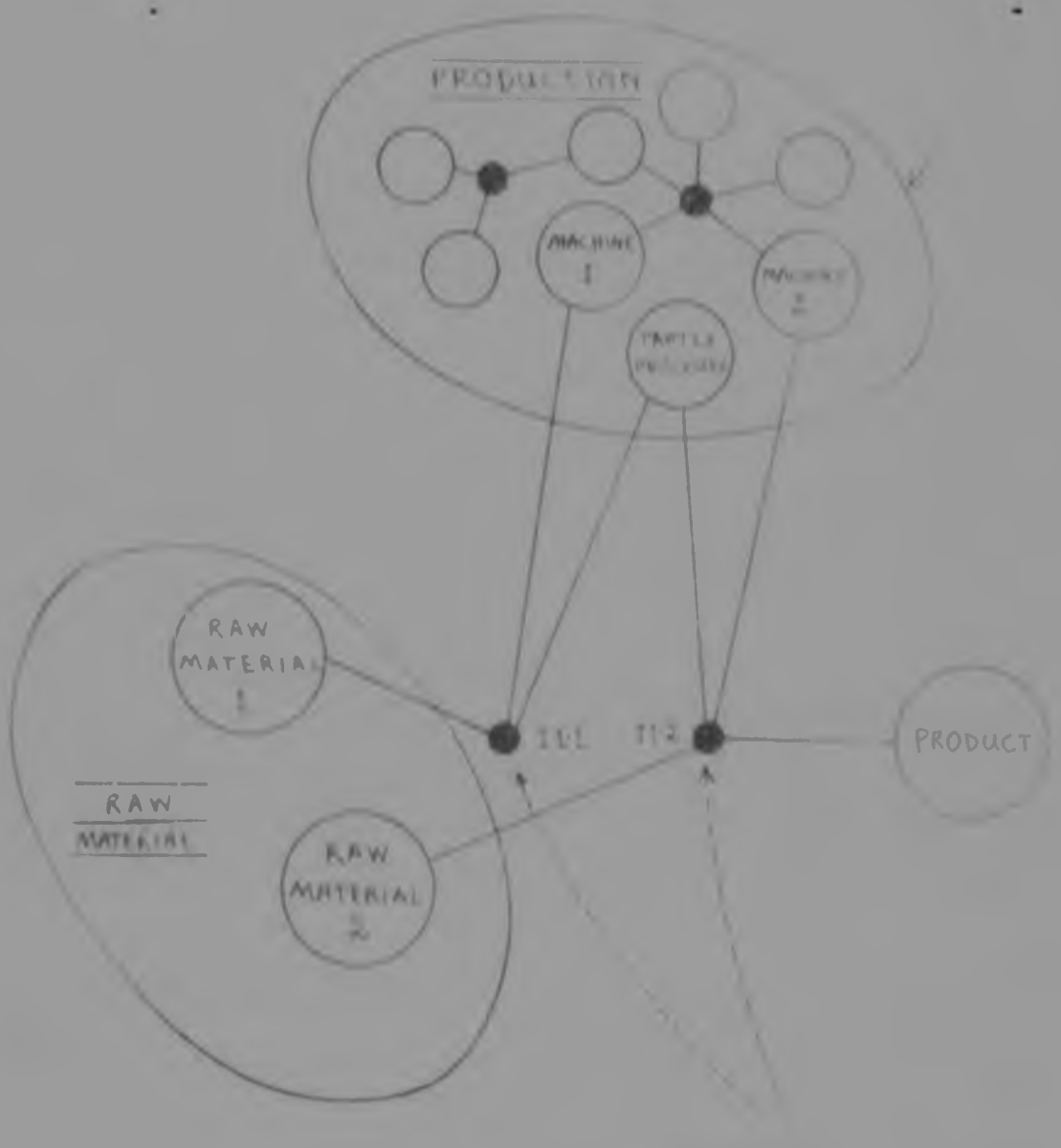


Fig. 6.6: Interaction network at a coarse resolution level

If it is desirable to represent the components and processes of the entity PRODUCTION, this will be replaced by a number of entities and interactions. Its interactions

with other entities may be replaced with several more interactions as shown in Fig. 6.7 below.

PRODUCTION at a finer resolution level



Interaction I1 at a finer resolution level

FIG. 6.7. Interaction network at a finer resolution level

In analysing PRODUCTION at a finer resolution level, the Production Scheduler may be considered as shown in fig. 6.8 below.

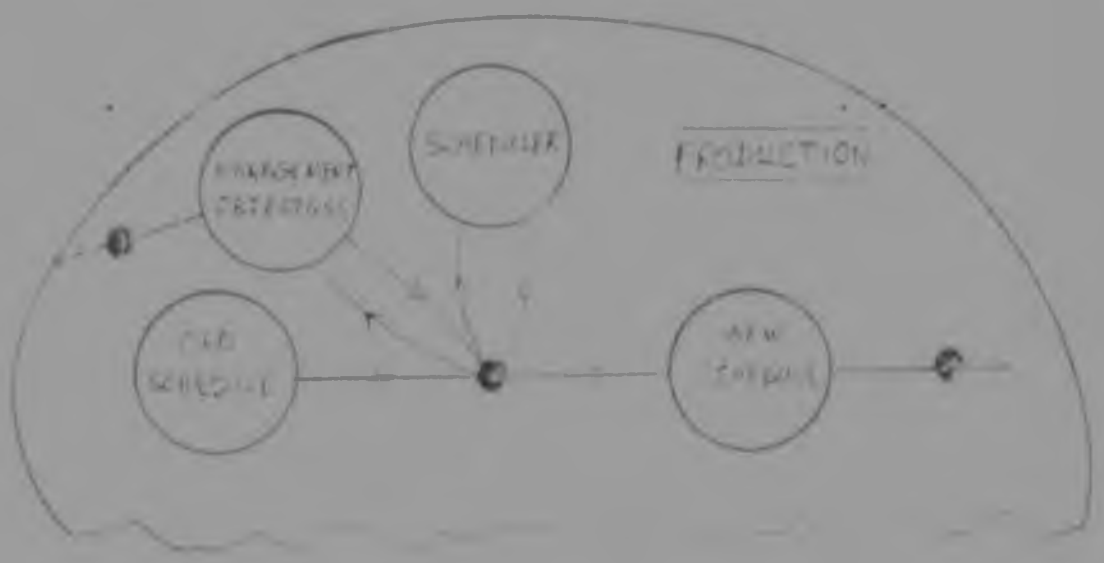


FIG. 6.8. Omitting of redundant information in interaction networks

If the new schedule is produced by a given set of rules and the state of the scheduler is not important, (it could have been important for example if error rate, absenteeism etc. were of interest) then the scheduler shown in the network is a redundant information and should be omitted.

6.7 TEMPORAL ANALYSIS

The analysis of the entity PRODUCTION shown in fig. 6.8 above, is a spatial analysis. As defined in section 2.2, the identifiable phenomenon is a spatictemporal object.

If we consider production as a spatiotemporal object shown in fig. 6.9 below, then it is apparent that our analysis described in fig. 6.7 is essentially a projection of the spatiotemporal object onto space (represented as two-dimensional in fig. 6.7, 6.8, 6.9 and hence does not give any time information. This is however important

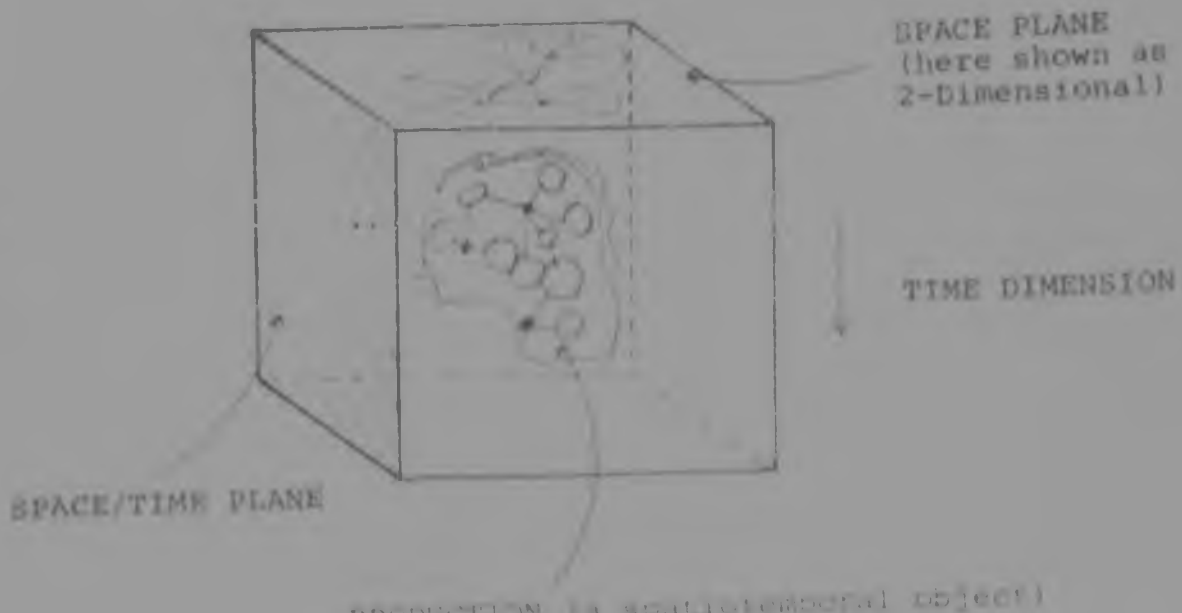


FIG. 6.9. Projections of the spatiotemporal object

information for the purpose of precise description by means of a model.

The next step therefore is the analysis of a phenomenon is to represent the space/time aspect (or plane - fig. 6.9) of the interaction network as in fig. 6.10 below.

Suppose that the network was considered at one chosen temporal resolution level. Then the time slices 1, 2 and 3 would have constituted a single time slice. This view would have given one interaction of the system as a whole which would have consisted of three time slices: initial (time slice 0), intermediate (time slices 1+2+3 where the change in the system is taking place) and final (time slice 4) as per our original concept of interaction in section 2.6. In each a chosen temporal resolution level, the system as a whole changed state, where we define the system state as being the set of states of its component entities.

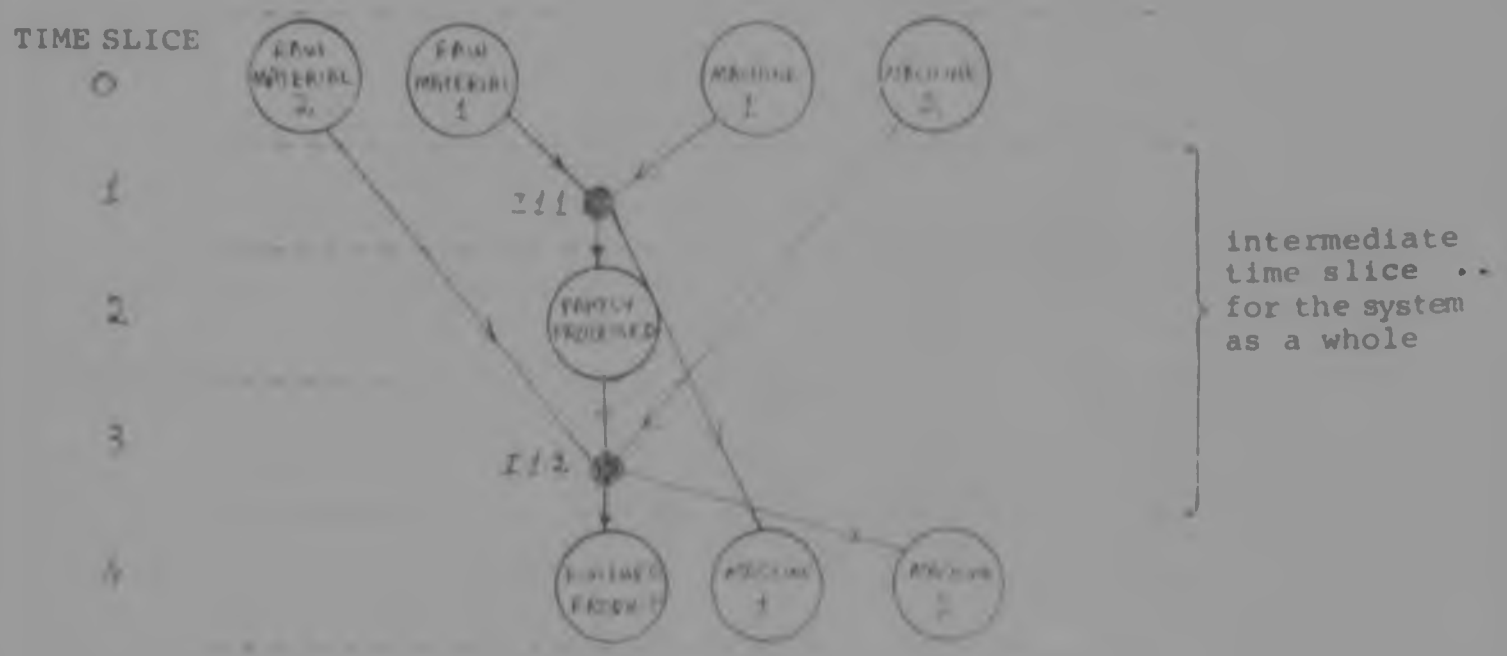


FIG. 6.10. The space/time aspect of the network

The temporal analysis above has analysed the intermediate time slice into three time slices (fig. 6.10) thus revealing just enough temporal information which provides the succession of interactions (I1.1, I1.2) assumed to be significant in this case.

6.8 INFORMATION AND THE MODEL DIAGRAM

At this stage all the required information is available. The last step is to translate this information into a specialised diagram which facilitates the programming of the model on the computer.

This diagram (shown in fig. 6.11 below) preserves the basic structure of the interaction network diagrams, but is specialised for the purpose of representing the information flow and processing which will take place in



FIG. 6.10. The space/time aspect of the network

The temporal analysis above has analysed the intermediate time slice into three time slices (fig. 6.10) thus revealing just enough temporal information which provides the succession of interactions (I1.1, I1.2) assumed to be significant in this case.

6.8 INFORMATION AND THE MODEL DIAGRAM

At this stage all the required information is available. The last step is to translate this information into a specialised diagram which facilitates the programming of the model on the computer.

This diagram (shown in fig. 6.11 below) preserves the basic structure of the interaction network diagrams, but is specialised for the purpose of representing the information flow and processing which will take place in

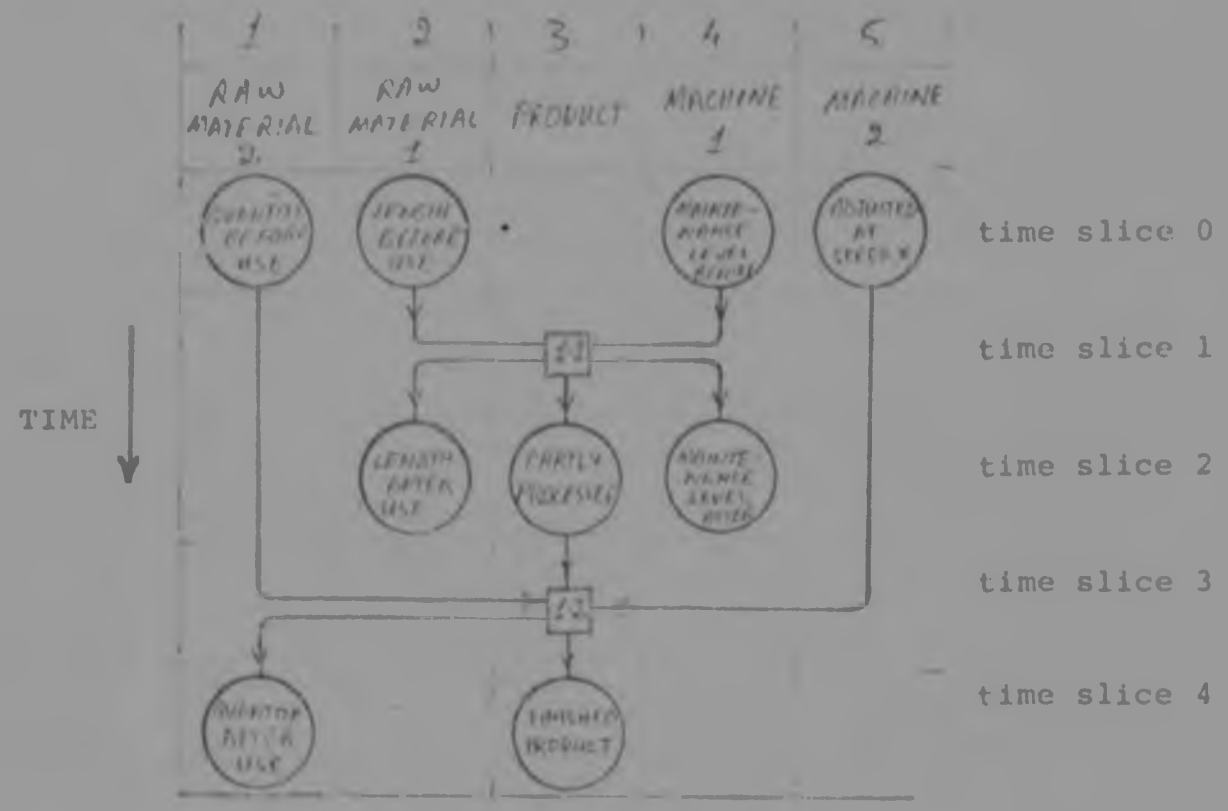


FIG. 6.11. Example of a model diagram

the computer model of the interaction network. It can be distinguished by the term Model Diagram as distinct from the interaction network diagrams used so far.

Its features are as follows:

1. Columns represent entities. Each entity may change as it moves down the column in the time dimension. At the last time slice, it is assumed to have the most recent state acquired during this time movement. For example machine 1 at time slice 4 is in the same state as at time slice 2.

2. Interactions are numbered for identification, and distinguished by small squares.

3. Entities are drawn only at the time slices where a change of state occurs and can show some indication of state (if desired).

4. Lines represent information flow in the computer model. For example, in order to determine the mode of interaction I1.1 at time slice 1, the states of entities 2 and 4 are needed.

Note that machine 2 which in this case is a catalyst, may be changed in the broader network by some other interaction (e.g. it may be adjusted by an attendant). Because it does change, it is not redundant information, and therefore is represented in the broader diagram (of which the diagram in fig. 6.10 is a section). In the diagram of fig. 6.10 it provides information to the algorithm which calculates the mode of I1.2, and therefore only shown in time slice 0.

Note that the model diagram apart from preserving the idea of interaction (see 2.6) and preserving the structure of the interaction network which it represents, it also performs the function of a data flow diagram and a flowchart, since as the former it specifies the flow of data (lines) and as the latter it specifies the flow of control (sequence of time slices).

6.9 THE STANDARD PROGRAM MODEL

If we want to program the above, the computer must be taken through each time slice and calculate the modes of interactions or the state of entities in each time slice. When calculating modes of interactions, information regarding the states of entities involved in this interaction must be used. These entity states should have been calculated during earlier steps corresponding to earlier time slices. Similarly when calculating entity states information on interaction modes (calcu-

lated for previous time slices) will be used. Whichever language is used, the computer program must have the structure indicated by the model diagram (e.g. fig. 6.11). We use the term "program structure" in the same way as used by Yourdon & Constantime in "structured systems design" [27] to which the reader is referred.

The main points are:

(a) A program statement (or subroutine call) represents an entity or an interaction and when executed gives the new interaction mode or the new entity state.

(b) The lines of information flow shown in the model diagram (e.g. fig. 6.11) should be represented in the program by the same communication structure [27]. E.g. a line drawn from an entity to an interaction in the model diagram is represented in the program by the statement representing the interaction, using (only) the state variable which is calculated by the statement representing the entity.

(c) Finally the control structure [27] of the program (sequence of statements) must be the same as that of the model diagram, i.e. the sequence of control (execution) passed from one statement to the next should be in the same sequence as that indicated by the time slice numbers. Furthermore the whole model diagram represents one iteration to be repeated until the time interval of interest has been covered.

Thus by following the above a program can be produced with a two-fold function:

1. It serves as a documentation of the structure of the phenomenon.

2. It produces its behaviour when executed on the computer.

Furthermore the structure is abstract in the sense that by changing the statements but keeping the structure (communication and control) [27] we can model different systems and demonstrate that their structure is the same. This would be the case, for example, if we are studying three different physical systems and have used one interaction network to show that they all have the same structure.

The statements of the program model of this interaction network will all be subroutine calls (and of course the communication and control structure is the same as discussed above). Now we can have a different set of subroutines for each of the three physical systems each calculating the specific entity states and interaction modes relevant to that physical system.

Thus we have a simulation program which separates the structure which is common to a class of systems from the specifics of each physical system.

Where we had three different phenomena simulated by three different programs, and were able to say that these systems are "analogous" in certain respect, we can (by using the above method) extract that analogy (structure) in the form of a concrete standard program.

EXAMPLE

Consider the interaction network shown in fig. 6.12 below. It represents the system of two chemical reactions which was considered in section 2.8.

From this network, the model diagram can be constructed as shown in fig. 6.13 below.

It should be stressed that arrows in a model diagram represent flow of information between the algorithms in the computer program which calculate interaction modes and entity states.

On the other hand arrows in an interaction network diagram (fig. 6.12), if and when used, represent whatever is required

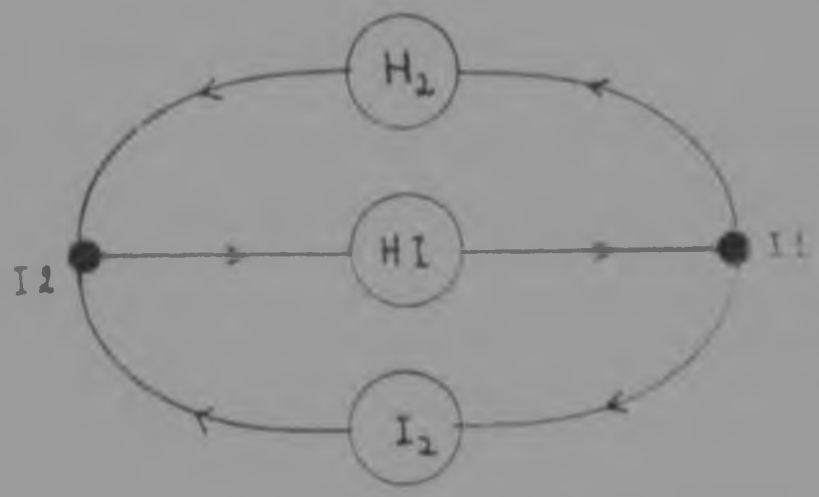


FIG. 6.12. An interaction network for a system of chemical reactions

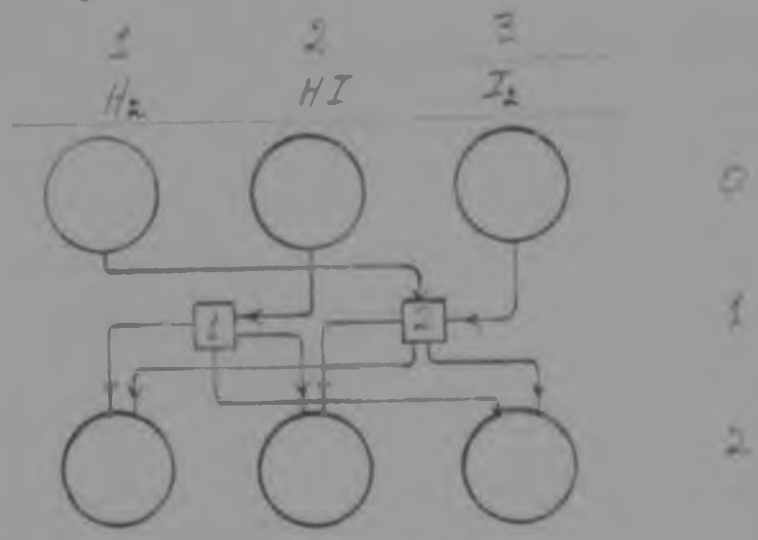


FIG. 6.13. Model diagram of the chemical reactions

according to the definitions for that phenomenon. For example in fig. 6.12 above they represent flow of substances.

Suppose that the following information was given about the phenomenon:

- The concentrations of the substances HI, H_2, I_2 are denoted by [HI], $[H_2]$, $[I_2]$
- The rate of reaction 1 is $R_1 = k_1[HI]^2$, where $k_1 = 0,06$ (1)

- The rate of reaction 2 is $R_2 = k_2[H_2][I_2]$ where $k_2 = 0{,}04$ (2)

- For a given small time interval the rates do not change in the present resolution level (3)

- The change in the concentrations during a time interval are $\Delta[HI] = 2(R_2 - R_1)$, $\Delta[I_2] = \quad H_2 = R_1 - R_2$ (4)

We let the <u>modes</u> of interactions represent the reaction rates and the <u>states</u> of entities HI, H_2, I_2 represent the concentrations. Then using equations (1), (2), (3) and (4) the program shown below is written.

```
C-----------------------------------------------I
C              D A T A                          I
C-----------------------------------------------I
      REAL I1,I2,II,K1,K2
      DATA HH,II,HI,K1,K2,T/1,.8,0,.06,.04,0/
C-----------------------------------------------I
C   S U P E R V I S O R     R O U T I N E       I
C-----------------------------------------------I
      WRITE(6,2)
    2 FORMAT(" TIME    HH        II        HI",/)
   10 WRITE(6,200T,HH,II,HI
   20 FORMAT(I4,3F6.3)
      T=T+1
      IF (T.EQ.5))GO TO 200
C-----------------------------------------------I
C   T I M E    S L I C E  1    (INTERACTIONS)   I
C-----------------------------------------------I
      I1=K1*HI*HI
      I2=K2*HH*II
C-----------------------------------------------I
C   T I M E    S L I C E  2    (ENTITIES)       I
C-----------------------------------------------I
      HH=HH+I1-I2
      II=I1+I1-I2
      HI=HI+2*(I2-I1)
C-----------------------------------------------I
C          E N D                                I
C-----------------------------------------------I
      GO TO 10;
  200 LOCK 5
      STOP
      END
```

This program is in a standard form which we will use to model interaction networks. The DATA and SUPERVISOR ROUTINE sections take care of the initialisation, control of iterations and reporting of the behaviour of the system.

The important parts are the "TIME SLICE 1" and "TIME SLICE 2" sections. These correspond to the time slices shown in our model diagram in fig. 6.13. In TIME SLICE 1 the new interaction modes are calculated as per equations (1) and (2) above. The information used is the entity states $[E_1]$ $[E_2]$ and $[I_2]$ as they are at the 'before' time slice of this interaction (time slice 0 fig. 6.13). In TIME SLICE 2 the new entity states are calculated according to (3) and equation (4) above. The information used is the interaction modes calculated by TIME SLICE 1. What is produced is the state of the entities at the 'after' time slice of this interaction (time slice 2 in fig. 6.13).

Each iteration in the program corresponds to one interaction taking place. Also note that the flow of control (sequence of execution of statements) and flow of data (which variables are used where) are preserved in the program and are as given in the model diagram fig. 6.13.

The output representing the behaviour of the system is shown below in table (...)

TABLE 6.1

The behaviour of the system

TIME	HH	II	HI
0	1.000	0.800	0.000
1	0.966	0.766	0.064
2	0.939	0.739	0.123
3	0.912	0.712	0.177
4	0.888	0.688	0.225
5	0.866	0.666	0.268
6	0.847	0.647	0.305
7	0.831	0.631	0.336
8	0.817	0.617	0.364
9	0.805	0.605	0.390
10	0.794	0.594	0.411
11	0.786	0.586	0.429
12	0.778	0.578	0.443
13	0.772	0.572	0.456
14	0.767	0.567	0.466
15	0.763	0.563	0.475
16	0.759	0.559	0.482
17	0.756	0.556	0.488
18	0.753	0.553	0.493
19	0.751	0.551	0.497
20	0.750	0.550	0.502
21	0.748	0.548	0.504
22	0.747	0.547	0.506
23	0.746	0.546	0.508
24	0.745	0.545	0.510
25	0.745	0.545	0.511
26	0.744	0.544	0.512
27	0.744	0.544	0.513
28	0.743	0.543	0.514
29	0.743	0.543	0.514
30	0.743	0.543	0.515
31	0.742	0.542	0.515
32	0.742	0.542	0.516
33	0.742	0.542	0.516
34	0.742	0.542	0.516
35	0.742	0.542	0.516
36	0.742	0.542	0.517
37	0.742	0.542	0.517
38	0.742	0.542	0.517
39	0.742	0.542	0.517
40	0.742	0.542	0.517
41	0.742	0.542	0.517
42	0.741	0.541	0.517
43	0.741	0.541	0.517
44	0.741	0.541	0.517
45	0.741	0.541	0.517
46	0.741	0.541	0.517
47	0.741	0.541	0.517
48	0.741	0.541	0.517
49	0.741	0.541	0.517

A graph of the results is shown in fig. 6.14 below.

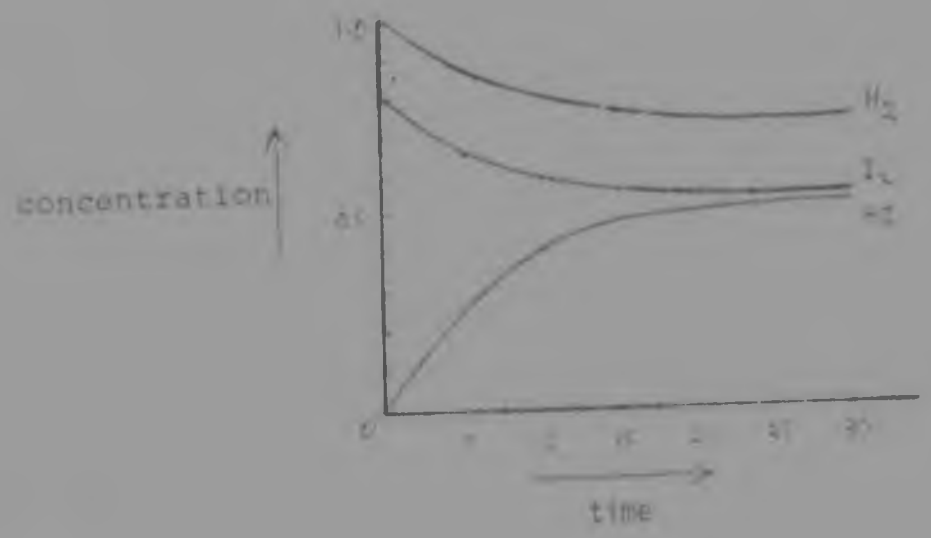


FIG. 6.14. Graph of the behaviour of the system

CHAPTER 7

A DETAILED MODEL

7.1 INTRODUCTION

The aim of this chapter is to give a detailed illustration of the technique outlined in the previous chapter, on a specific phenomenon. A phenomenon has been chosen which has already been analysed and modelled by other more specialised methods, so that the results can be checked thus establishing the validity of this general approach.

7.2 THE OBJECT TO BE STUDIED

This phenomenon is a subsystem of the system "society". It consists of the process of production of commodities and their consumption by the consumer. It is based on a study made by Meadows in his work "The Dynamics of Commodity Production Cycles" [21] in which he analyses factors of production and consumption of commodities for the purpose of studying the mechanisms involved in the commodity cycles. In these cycles production, price and consumption fluctuate regularly and are resistant to stabilisation.

From the point of view of this dissertation the interest is in constructing a model on the same data and the same assumptions as those made by Meadows, and in describing the construction of the model from the initial interaction network to the final run of the computer model which must agree in behaviour with Meadows' model.

Meadows uses the simulation language DYNAMO which is especially designed for simulating this kind of systems.

In the following sections the object will be described in our terminology. For details of Meadows' study the reader is referred to [21]. Since the results of the two models will be compared it is important to use the same formulae and data exactly as used in Meadows' model.

7.3 THE INTERACTION NETWORK AT A COARSE RESOLUTION LEVEL

The entities involved (shown in fig. 7.1 below) are

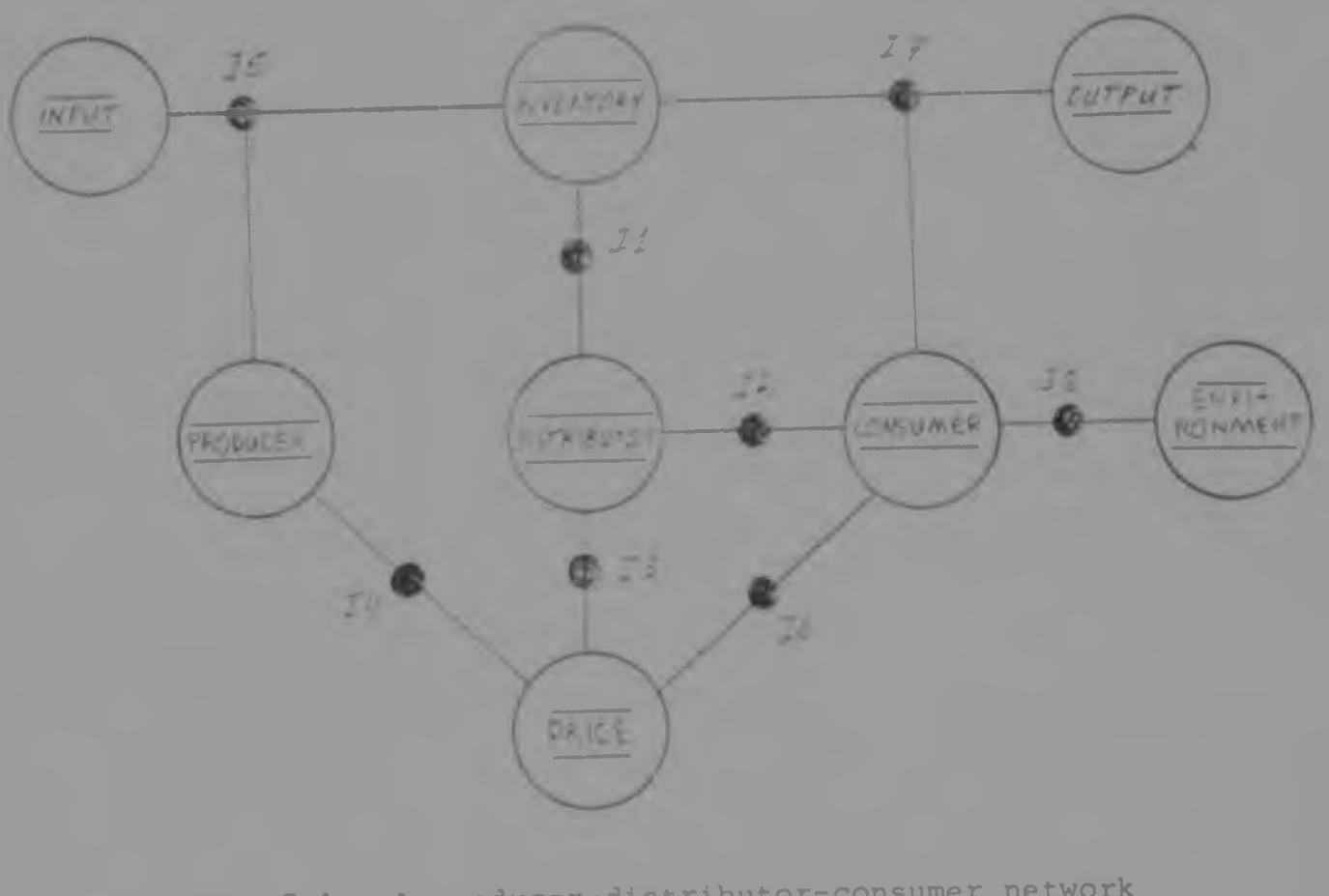


FIG. 7.1. A producer-distributor-consumer network at a coarse resolution level

the following:

1. The price: This is an information carrier which can be visualised as consisting of all the latest pricelists, pricetags etc. associated with that commodity.

2. Inventory: The collection of all commodities stored after production.

3. The Distributor: The collection of all people and equipment involved in establishing the price. Other aspects, such as transportation of commodities are of no interest here.

4. Producer: The collection of all the people and equipment which interact with some kind of Input and produce commodities.

5. Consumer: The collection of all people and equipment which consume the commodities.

6. Input: The collection of all materials or any outside entities used in the production of commodities. This entity is of no significance in Meadows' study and will be ignored here.

7. Output: The collection of all entities generated during consumption (e.g. waste or other commodities). This is also ignored in this study.

8. Environment. The collection of all entities influencing the relevant decisions of the consumer.

The principal interactions are:

I1: The distributor monitors the level of inventory in order to decide any adjustment to prices.

I2: He also monitors the consumer demand in order to decide any expected drops in the level of inventory.

I3: If the distributor realises that he will have low inventory levels, he raises the prices.

I4: The producer monitors the price in order to adjust his production

I5: If the price is higher then he increases production capacity to take advantage of the price raise. If the price is lower he diverts production capacity and capital to other activities.

I6: The consumer also monitors the price which is important for his behaviour.

I7: The consumer tries to maximise the utility obtained from that commodity. Thus if the price is lowered the marginal utility of the commodity in general raises and funds are diverted to increase consumption. The opposite occurs when the price is raised.

I8: There are extraneous influences to every entity and interaction in the system. Here Meadows is interested in the extraneous factors influencing the consumer such as wrong information about the price or miscalculation in the increase of consumption warranted by a decrease in price. In the model studied here this interaction is used to introduce a disturbance in the system by forcing the customer demand to a relatively high level for a short duration of time at the beginning of the simulation.

7.4 ANALYSIS AT A FINER RESOLUTION LEVEL

Fig. 7.2 gives the interaction network at a finer resolution level. This reveals that some of the entities consist of a number of component entities and interactions.

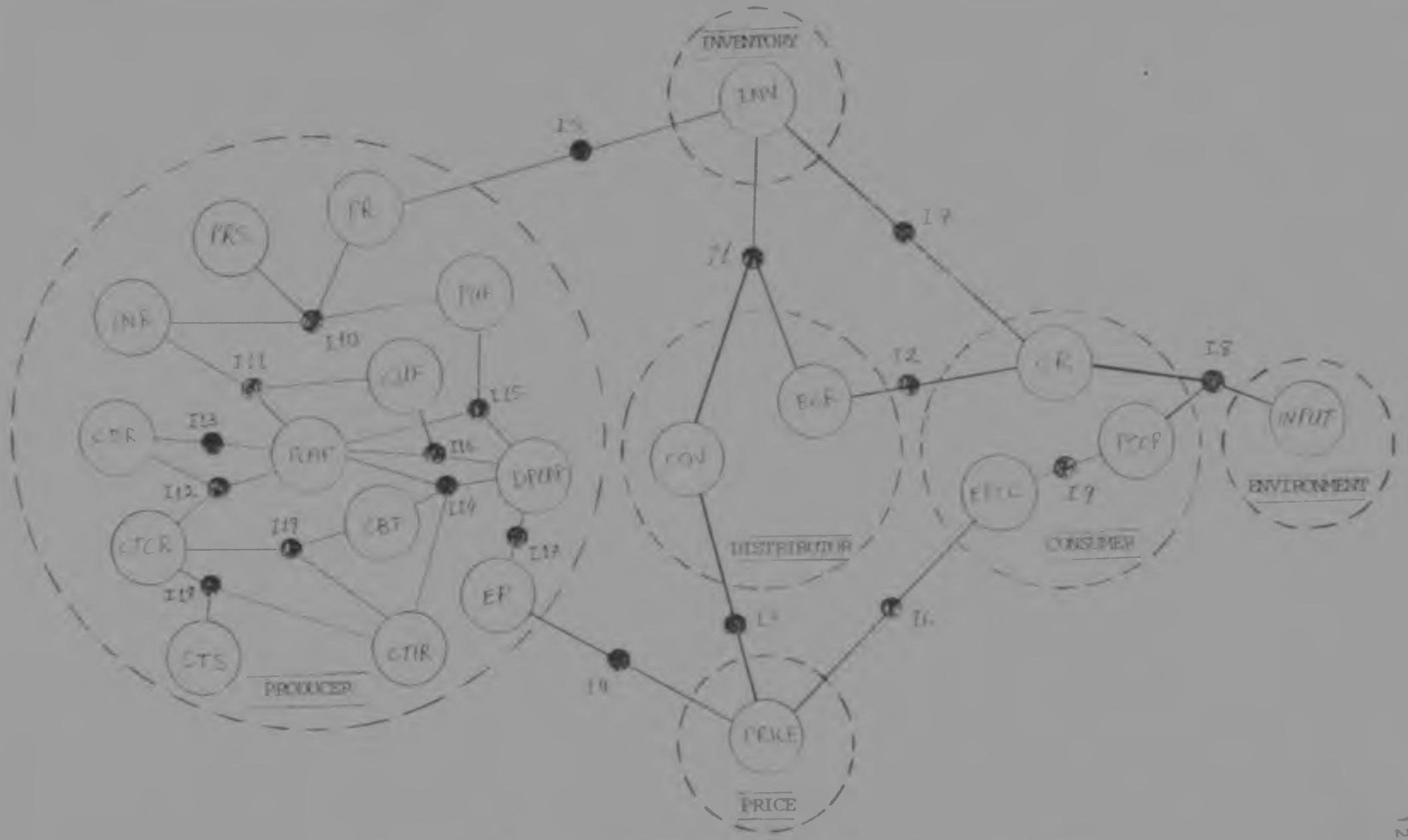


FIG. 7.2. The producer-distributor-consumer interaction network at the optimum resolution level

The following sections will discuss each major entity shown in fig. 7.1 in greater detail. For each, the information provided by Meadows' study will be given first, followed by the formalisation in terms of entities and interactions.

Once the information required to describe all component entities and interactions of fig. 7.2 is available, the model diagram will be constructed followed by the computer program.

7.5 INVENTORY

Information provided:

This entity consists of all stocks of the product in different forms and locations. Meadows assumes in his basic model that only the combined magnitude of all the inventories is important in the long run, and that production and consumption determine the quantity of the commodity in stock. Thus theft, dumping, spoilage, etc. are ignored.

For a small interval of time Δt (time instants t-1 to t), assuming constant production rate (PR) and consumption rate (CR), the inventory (INV in units/month) is given by

$$INV_t = INV_{t-1} + \Delta t \times (PT_{t-1} - CR_{t-1}) \qquad (1)$$

(in the following sections ABC_t will mean the variable ABC at time t).

Interaction network formalisation:

This (see fig. 7.2) is represented by a single entity INV, involved in interactions I5 (production) which increases its state value, and I6 (consumption) which reduces its state value, as per the equation (1) above.

Note: Theoretically the above are continuous variables and the above equation (1) could be written

$$INV_t = INV_{t-1} + \int_{t-1}^{t} (PR-CR)dt$$

However, Meadows makes the simplifying assumption of constant rates of change (linearity) during the time intervals. As stressed in section 7.2 the present model will be based on the same assumptions and use whatever algorithms, formulae or data as Meadows used. Then it will be possible to check the output of this model against that of Meadows. Furthermore the approach to modelling phenomena which are understood as involving continuous variables involve some special considerations. These will be discussed in the next chapter.

7.6 DISTRIBUTOR

Information provided:

This entity consists, from Meadows' point of view, of those who control inventory by adjusting the selling price. They are concerned with the length of time for which the current inventory would satisfy the anticipated demand. This is termed Coverage (COV in months) and is the ratio of current inventory (INV) to the expected consumption rate.

$$COV = INV/ECR \qquad (2)$$

Expected consumption rate (ECR in units/month) is the consumption which the distributor expects in the near future. This is assumed to undergo adjustment which is proportional to the difference between recent actual consumption rate and the previous expected consumption rate.

$$ECR_t = ECR_{t-1} + \Delta t \times (CR_{t-1} - ECR_{t-1}) \times ECAD \qquad (3)$$

Note: Theoretically the above are continuous variables and the above equation (1) could be written

$$INV_t = INV_{t-1} + \int_{t-1}^{t} (PR-CR)dt$$

However, Meadows makes the simplifying assumption of constant rates of change (linearity) during the time intervals. As stressed in section 7.2 the present model will be based on the same assumptions and use whatever algorithms, formulae or data as Meadows used. Then it will be possible to check the output of this model against that of Meadows. Furthermore the approach to modelling phenomena which are understood as involving continuous variables involve some special considerations. These will be discussed in the next chapter.

7.5 DISTRIBUTOR

Information provided:

This entity consists, from Meadows' point of view, of those who control inventory by adjusting the selling price. They are concerned with the length of time for which the current inventory would satisfy the anticipated demand. This is termed Coverage (COV in months) and is the ratio of current inventory (INV) to the expected consumption rate.

$$COV = INV/ECR \qquad (2)$$

Expected consumption rate (ECR in units/month) is the consumption which the distributor expects in the near future. This is assumed to undergo adjustment which is proportional to the difference between recent actual consumption rate and the previous expected consumption rate.

$$ECR_t = ECR_{t-1} + AI \times (CR_{t-1} - ECR_{t-1}) \times ECAD \qquad (3)$$

where ECAD (in months) is the expected consumption rate adjustment delay. This parameter is the reciprocal of the expectation adjustment delay (1/ECAD), the meaning of which is illustrated in fig. 7.3 below.

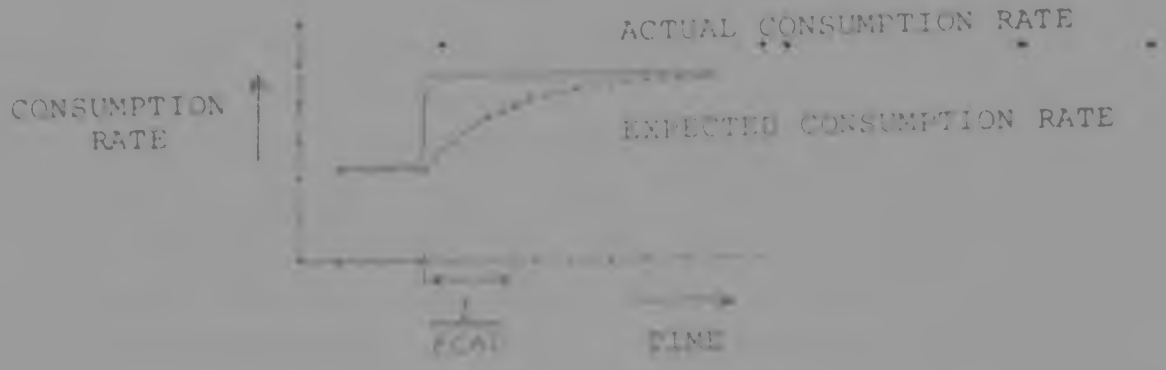


FIG. 7.3. Illustration of ECAD

Interaction network formalisation:

The distributor (fig. 7.2) consists of two entities: ECR and COV which are both information carriers. During interaction I2 involving ECR and C9 (CR is a consumer entity representing consumer demand explained in section 7.8 below) the state of ECR is changed to reflect the new expected consumption rate as per equation (3). During interaction I1 involving ECR, INV and COV, the state of COV is changed to reflect the new coverage as per equation (2). ECAD is constant and therefore not an entity of interest in the network but rather part of the algorithm of I2.

7.7 PRICE

Information provided:

Price is taken by Meadows as being the single variable which the distributor manipulates in order to control the inventory at a desired level. Control of the inventory level is a viable for the following reasons:

The type of commodities which Meadows studies are oil, tin, sugar etc., i.e. the consumer here is an intermediate processor who uses the commodity as feestock into his own manufacturing process. The maintenance of a certain inventory level by the distributor provides a service, in that it protects the intermediate processor from cost of idle capacity should the supply of the commodity be interrupted. They also stabilise the price which the intermediate processors charge to recover the average cost of the product. As inventory increases, the costs of storage, facilities maintenance, personnel expenses, spoilage etc., rise while the protection afforded by each additional unit decreases.

Thus there is an optimum compromise reflected in fig. 7.4.

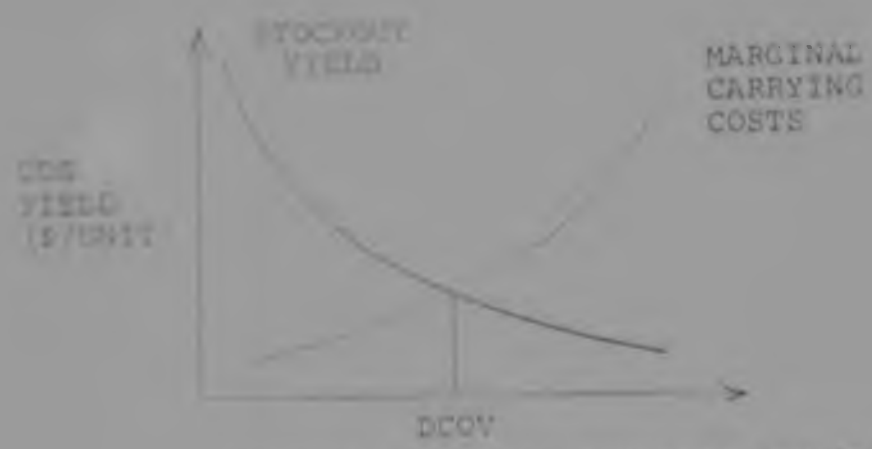


FIG. 7.4. Illustration of DCOV

The optimum coverage just equates the marginal cost and yield of inventory. This is denoted by DCOV (desired coverage in months) and is assumed to be a constant for this model.

DCOV (no units) is given by the ratio

$$DCOV = DCOV/DCOV \qquad (4)$$

and is a measure of the agreement between the actual and the desired coverage. Price then can be determined as function of RCOV. Meadows uses the graph shown in fig. 7.5.

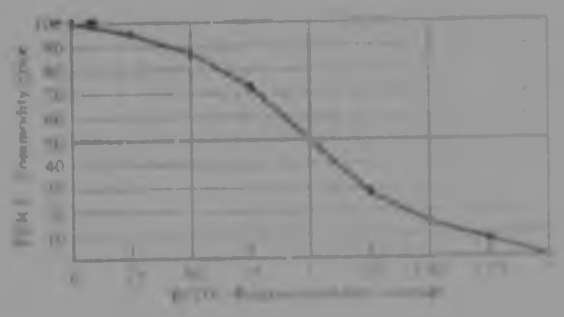


FIG. 7.5. PRICE as a function of RCOV

Interaction network formalisation:

Only the entity PRICE is required (fig. 7.4). Interaction 13 involves entities COV and PRICE and results in a new state for PRICE. The desired inventory coverage (DCOV) is a constant in Meadows model and therefore not required as part of the interaction network. RCOV is also not required as a separate entity since from fig. 7.5 and equation (4) PRICE can be obtained as a function of COV.

7.8 CONSUMER

Information provided:

As the price of a commodity is raised it is replaced in some of its uses by other competing commodities. The opposite occurs for drops in the price. The relationship between the commodity price and the demand used by Meadows is given by the graph shown in fig. 7.6 below. This gives the equilibrium per capita consumption (EPCC in units/man · month) as a function of price. This variable represents the optimum consumption for a given price

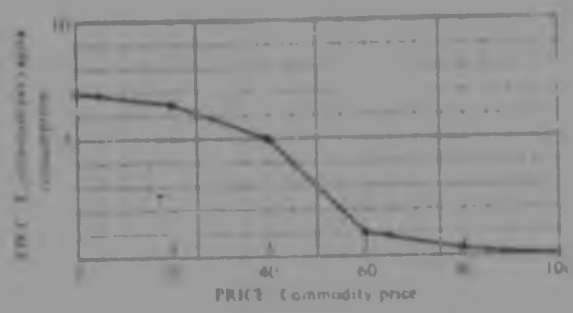


FIG. 7.6. EPCC as a function of PRICE

The consumer however does not respond instantaneously to a change in price. Psychological, physiological, and technological factors give the consumer a type of inertia of behaviour. Thus another variable, per capita consumption requirement (PCCR in units/man.month) reflects the actual requirements per consumer. This variable adjusts continuously towards the equilibrium per capita consumption (EPCC) variable. The speed of adjustment is reflected by the parameter: consumption requirements adjustment delay (CRAD in months). PCCR is given by

$$PCCR_t = PCCR_{t-1} + \frac{\Delta t \times (EPCC_{t-1} - PCCR_{t-1})}{CRAD} \qquad (5)$$

Finally the actual consumption rate (CR in units/month) is given by

$$CR = POP \times PCCR \times INPUT \qquad (6)$$

where POP (in men) is the population and INPUT (no units) is any extraneous noise which can influence the demand.

Interaction network formalisation:

The entities are EPCC, PCCR and CR (fig. 7.2). CR is

seen in Meadows model as "Consumption Rate" but the term "rate" would be appropriate for an interaction mode. However it is desirable to keep the same symbols where possible for comparisons therefore the symbol CR will be used for Demand (an information carrier entity).

Interaction I6 involves PRICE and EPCC and results in change of state of EPCC using the graph in fig. 7.6. Interaction I9 involves EPCC and PCCR and results in change of PCCR as per equation (3). I8 involves PCCR, INPUT and CR and results in change of state of CR as per formula (6).

7.9 ENVIRONMENT

Information provided:

This being the extraneous influences on the consumer (INPUT no units), it was used in this model to artificially increase the demand by 30% for an initial duration of 10 months, thus introducing a disturbance in the system in order to study its response.

Interaction network formalisation:

INPUT is kept by the supervisor routine of the model at the value of 1.3 for the equivalent duration of 10 months after which it is changed to 1, thereafter having no effect on the system, as per equation (6) which describes interaction I8 (see fig. 7.2).

7.10 PRODUCER

Information provided:

This is a more complex entity than the others, involving a more complicated information processing.

At the start of the process is the determination of the expected price and at the end as the final result,

is the adjustment of the physical production level.

The expected price (EP in $/unit) which the producer assumes in his calculations can be given by the formula

$$EP_t = EP_{t-1} + \frac{\Delta t \times (PRICE_t - EP_{t-1})}{EPAD} \qquad (7)$$

where EPAD in months is the expected price adjustment delay. This parameter reflects the fact that the producer tends to discount recent changes in the price thus giving some stability to his plans.

For each expected price there corresponds a unique desired production capacity (DPCAP in units/month). This is established in Meadows' study by using the graph in fig. 7.7.

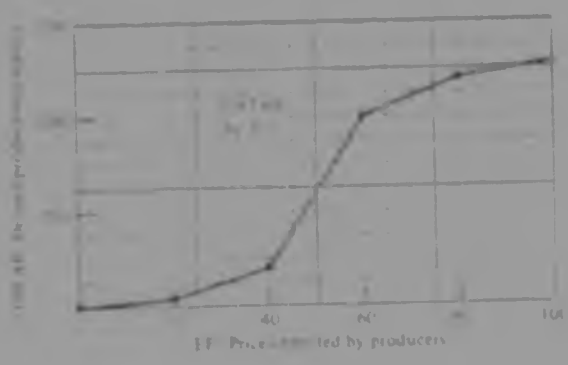


FIG. 7.7. DPCAP as a function of PRICE

By comparing the desired production capacity to the actual production capacity it is possible to determine the capacity transfer in tiation rate (CTIR in units/month^2). This indicates the change which has to be initiated in order to bring the production capacity to what it should be, and is given by

$$CTIR_t = (DPCAP - PCAP - CBT)\ CTID \qquad (8)$$

CTID (in months) in equation (8) is the capacity transfer

initiation delay and reflects the fact that changes to production capacity cannot be initiated immediately after they have been decided. There is internal competition for personnel, finance, physical facilities, etc. Transferring resources to the task of transfer (such as construction, purchase of machinery, etc.) takes time. Hence the parameter CTID is used.

CBT (in units/month) is the capacity being transferred, i.e. the changes in the production capacity which have been initiated and are now being implemented. It is given by

$$CBT_t = CBT_{t-1} + \Delta t \times (CTIR_{t-1} - CTCR_{t-1}) \qquad (9)$$

CTCR is the capacity transfer actually taking place (described below).

Having initiated a production capacity transfer, the process of implementing this is started, and it takes time to be completed (actually construct new plants, order and deliver new equipment, etc.). This is complicated further by the fact that while these changes are being implemented, CTIR also changes, i.e. orders (and the accompanying resources) arrive which initiate the construction of more plants or cancellation of old orders for machinery. As a result the capacity transfer completion rate (CTCR in units/month) is affected by CTIR through a dynamic delay relationship. The DYNAMO equation which Meadows uses is

CTCR ·KL = DELAY3(CTIR · JK, CTD)

DELAY3 is a special DYNAMO [22] curve-shaping intrinsic which generates certain functions transparent to the user, which ensure that the changes of the dependent variable follow those of the independent variable in a special delay pattern.

In terms of the present approach this part of the phenomenon must be analysed in terms of the entities and interactions which comprise the mechanism of the delay. This is done as follows:

In situations such as the one on hand where the capacity transfer initiation rate changes yet the physical implementation of this takes time, there is some kind of a project schedule. Any capacity transfer which has been decided (and the resources provided) is put into a schedule, where it will go through a certain number of phases. For example: order, delivery, installation, testing, or in a different case: design, construction, testrun. This introduces the entity CTS (capacity transfer schedule in units/month). This has a 2-dimensional state CTS = (x_1, x_2) as established in section 6.3 and it represents the capacity transfer schedule consisting of two phases. State component x_1 represents all the capacity transfer being in the first phase of the implementation schedule and x_2 that in the second phase of the implementation schedule. The third phase of the implementation is actually the materialising of change in the capacity.

Any capacity transfer which has been decided on, will have to progress in orderly manner through the three phases until it is finally materialised, i.e. added to the actual production transfer completion rate. This, it is believed, represent the mechanism of the delay between CTIR and CTCR. The three phases chosen represent the order of the DELAY3 intrinsic [22] which has been chosen by Meadow as 3 and represents a 3-compartment delay.

The duration of the implementation capacity transfer initiated is CTD (capacity transfer delay in units/month), therefore the duration of each phase is CTD/2. The equations to be used are:

$$
\left.
\begin{aligned}
CTCR_t &= CTCR_{t-1} + \frac{\Delta t \times (x_{2,t-1} - CTCR_{t-1})}{CTD/3} \\
x_{2,t} &= x_{2,t-1} + \frac{\Delta t \times (x_{1,t-1} - x_{2,t-1})}{CTD/3} \\
x_{1,t} &= x_{1,t-1} + \frac{\Delta t \times (CTIR_{t-1} - x_{1,t-1})}{CTD/3}
\end{aligned}
\right\} \quad (10)
$$

Production capacity (PCAP in units/month) is the production rate which the plants are capable of sustaining at full utilisation. It is given by:

$$PCAP_t = PCAP_{t-1} + \Delta t \times (CTIR_{t-1} - CDR_{t-1}) \quad (11)$$

CDR is the capacity depreciation rate (explained below). Thus the change in PCAP is equal to the net effect of the capacity transfer completion rate and the depreciation.

Capacity depreciation rate CDR (in units/month^2) represents the amount of decrease in production capacity due to depletion, biological aging, physical deterioration etc. This is given by:

$$CDR = PCAP/ALPC \quad (12)$$

where ALPC (in months) is the average life of production capacity.

The available production capacity is not necessarily utilised to the full. The capacity utilisation factor CUF (no units) is a measure of this. The decision regarding what fraction of capacity to utilise depends on the ratio of actual to desired production capacity DPCAP/PCAP. Based on this ratio, CUF can be calculated by a simple or a complex rule. In the model studied here Meadows takes CUF to be 1, i.e. capacity is utilised fully irrespective of the DPCAP/PCAP ratio.

Having a certain production capacity and having decided the capacity utilisation factore, production is initiated. The measure for this is the commodity initiation rate (INR in units/month) given by:

$$INR = PCAP \cdot CUF \tag{13}$$

Production rate PR (in units/month) is the physical production taking place. This materialises the instructions contained in INR. There is here also a delay similar to the one existing between CTIR and CTCR.

We introduce here (as in the case of CTIR and CTCR above) a production schedule ($PRS = (y_1, y_2)$ in units/month) which is defined similarly to the $CTS = (x_1, x_2)$.

Thus we have:

$$\begin{aligned} PR_t &= \left[PR_{t-1} + \frac{\Delta t \cdot (y_{2,t-1} - PR_{t-1})}{PD/3} \right] \cdot PUF \\ y_{2,t} &= y_{2,t-1} + \frac{\Delta t \cdot (y_{1,t-1} - PR_{t-1})}{PD/3} \\ y_{1,t} &= y_{1,t-1} + \frac{\Delta t \cdot (INR_{t-1} - y_{1,t-1})}{PD/3} \end{aligned} \tag{14}$$

PD is the production delay (in months) i.e. the average time it takes for INR to become effective.

PUF is the production utilisation factor and is analogous to CUF. After production is essentially completed producers may have the options of harvesting or refining more intensely or some finished product may be left unharvested or deployed outside the commodity system. This decision depends on the ratio of the actual to desired production capacity and may be seen as some function of PCAP/DPCAP. In the particular case studied by Meadows, PUF has been taken as 1, i.e. all products are added to inventory.

Interaction network formalisation:

The entities used are EP, DPCAP, CTIR, CTS, CTCR, CBT, PCAP, CDR, CUF, INF, PRS, PUF, PR as shown in fig. 7.2.

Interaction I4 involves EP and PRICE and results in a state change of EP as per equation (7). Interaction I17 involves EP and DPCAP and results in a state change of DPCAP as per the rate given in fig. 7.7. I14 involves DPCAP, PCAP and CBT and results in change of state of CTIR as per equation (5). I19 changes state of CBT. In Meadows' model CTIR is a rate variable. Although in the interaction network the same symbol (CTIR) is used, it represents here the information carrier (i.e. a decision or a schedule etc.) which gives instructions for a certain change of capacity. The term "rate" is unsuitable for representing an entity and is suitable for interaction nodes. I18 involving CTIR, CTS, CTCR adds capacity transfer to and schedule CTS at the rate indicated by entity CTIR, and adds capacity to CTCR as indicated by the second phase of CTS according to equations (18). I12 results in change to PCAP as per equation (11). I11 results in change of state of CDR. CDR in our model is an information carrier which interacts with PCAP resulting in a reduction of production capacity. One of the things that CDR may be representing is a schedule to replace a certain machine after its maintenance exceeds a certain value. I10 and I15 change the states of CUF and PUF respectively. I11 results in a new state of INF and finally I16 sets up PR at the right production level. I5 is an interaction between the production plant and input (material etc.) to produce products thus resulting in change of state of INV.

7.11 THE MODEL DIAGRAM

In the previous sections the interaction network was defined at the optimum resolution level. The entities and interactions were defined and the algorithms or formulae, as provided by the specialised discipline (economics) were given. These formulae will be used in establishing the interaction modes and entity states at each time slice.

Before this is programmed for the computer it is necessary to draw the model diagram as explained in sections 6.8 and 6.9 of the previous chapter. This is given in fig. 7.8 below.

Firstly this presents the structural analysis of the network where any existing duplication of interactions is revealed and can be incorporated in the computer model.

Secondly it indicates the information flow between entities and interactions which is used to connect the formulae or algorithms which calculate the changes to interaction modes and entity states.

7.12 THE COMPUTER MODEL

Based on the model diagram of fig. 7.8 A computer model was written in FORTRAN. This is shown below and is of the standard structure established in sections 6.9 and 6.10.

This program was run on a Burroughs B6800 computer giving the output listed in the table 7.1 below.

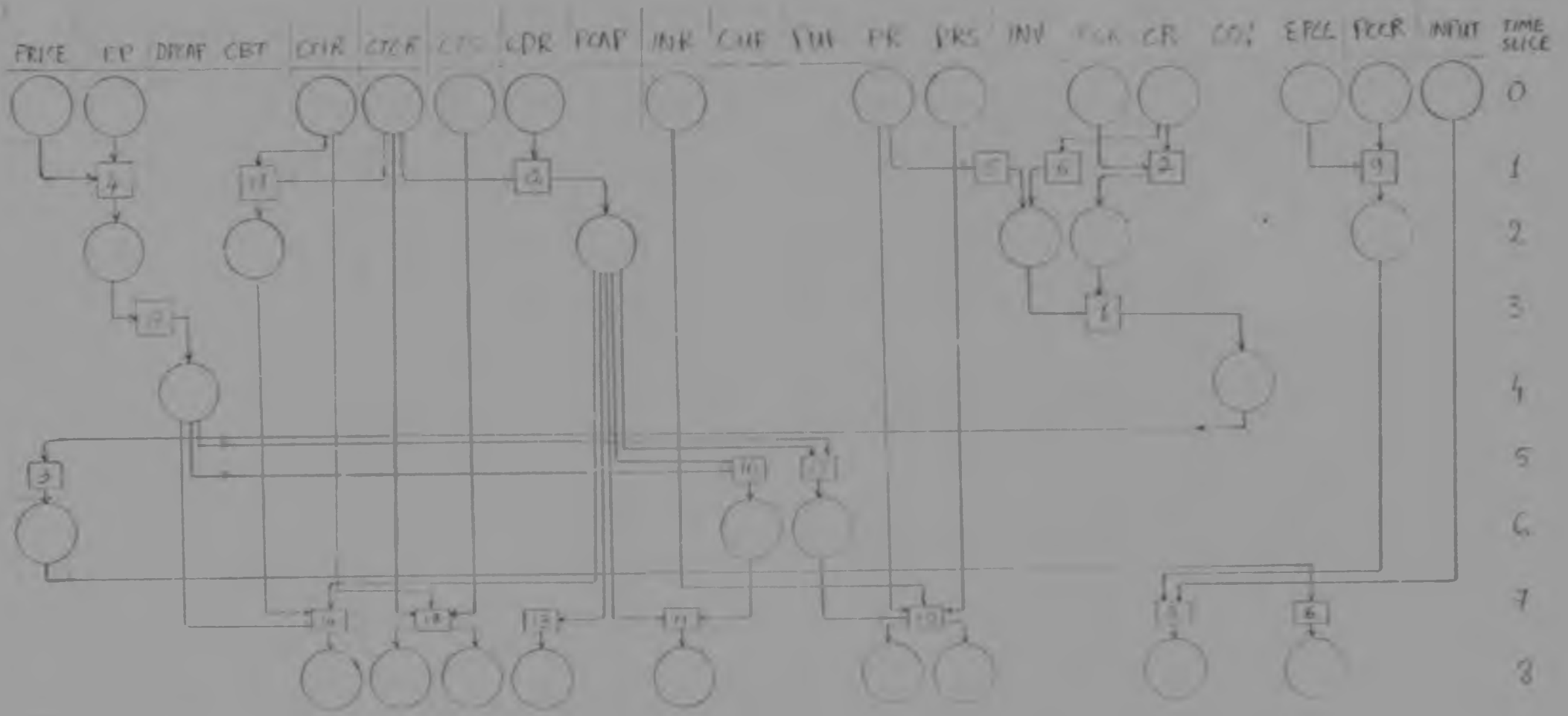


FIG. 7.8. The Producer-distributor-consumer model diagram

The computer model:

```
$REQUEST FREE
FILE  6(TITLE="MODEL",KIND=PRINTER)
C,-----------------------------------------------------------------------I
CI                                  D A T A                               I
CI-----------------------------------------------------------------------I
      REAL I1,I2,I3,I4,I5,I6,I7,I8,I9,I10,I11,I12,I13,I14,I15
      REAL I16,I17,I18,ESI,INPUT
      INTEGER T,RT
      DIMENSION PTAB(9),CATAB(6),CUTAB(6),CTS(2),PRS(2),I10(3),I18(3)
      DATA CUTAB,PUTAB,DCCV,CTID,CID,ALPC,PD,PCP/1,1,10,3,4,200,6,200/
      DATA INPUT,EPAD,ECAD,CRAD /1.3,3,100,3/
      DATA DT,I,RT/0.2,0,0/
      DATA EP,CBT,PCAP,INV,ECR,PCCR,CTIR,CTCR/50,0,600,6000,600,3,0,0/
      DATA CTS(1),CTS(2),INR,PR,PRS(1),PRS(2)/0,0,600,600,600,600/
      DATA EPCC/3/
      DATA PTAB/100,95.5,87,72.5,50,27,15,7.5,0/
      DATA CATAB/0,40,200,1000,1200,1260/CCTAB/7,6.5,5,1,0.3,0/

CI-----------------------------------------------------------------------I
CI          S U P E R V I S O R          R O U T I N E                    I
CI-----------------------------------------------------------------------I
      WRITE(6,20)
   20 FORMAT("  TIME PRICE  INV   INR    PR    CR",/)
      GO TO 150
   50 IF(RT.NE.10) GO TO 100
      WRITE(6,70)T=DT,PRICE,INV,INR,PR,CR
   70 FORMAT(F6.1,5F6.0)
      RT=0
  100 RT=RT+1
      T=T+1
      IF (T.GE.50) INPUT=1
      IF (T.GT.500) GO TO 1000

CI-----------------------------------------------------------------------I
CI              T I M E     S L I C E  1     (INTERACTIONS)               I
CI-----------------------------------------------------------------------I
      I4=DT*(PRICE-EP)/EPAD
      I19=DT*(CTIR-CTCR)
      I12=DT*(CTCR-CDR)
      I5=DT*PR
      I7=DT*CR
      I2=DT*(CR-ECR)/ECAD
      I9=DT*(EPCC-PCCR)/CRAD
CI-----------------------------------------------------------------------I
CI              T I M E     S L I C E  2     (ENTITIES)                  I
CI-----------------------------------------------------------------------I
      EP=EP+I4
      CBT=CBT+I19
      PCAP=PCAP+I12
      INV=INV+I5-I7
      ECR=ECR+I2
      PCCR=PCCR+I9
```

```
CI-----------------------------------------------------------------------I
CI            T I M E     S L I C E  3     (INTERACTIONS)                 I
CI-----------------------------------------------------------------------I
  150 I=INT(E?/20)
      I17=CATAB(I+1)+(CATAB(I+2)-CATAB(I+1))*(EP-I*20)/20
      I1=INV/ECR
CI-----------------------------------------------------------------------I
CI            T I M E     S L I C E  4     (ENTITIES)                    I
CI-----------------------------------------------------------------------I
      DPCAP=I17
      COV=I1
CI-----------------------------------------------------------------------I
CI            T I M E     S L I C E  5     (INTERACTIONS)                 I
CI-----------------------------------------------------------------------I
      I=INT(COV/DCOV/0.25)
      I3=PTAB(I+1)+(PTAB(I+2)-PTAB(I+1))*(COV/DCOV-I*0.25)/0.25
      I16=CUTAB
      I15=PUTAB
CI-----------------------------------------------------------------------I
CI            T I M E     S L I C E  6     (ENTITIES)                    I
CI-----------------------------------------------------------------------I
      PRICE=I3
      CUF=I16
      PUF=I15
CI-----------------------------------------------------------------------I
CI            T I M E     S L I C E  7     (INTERACTIONS)                 I
CI-----------------------------------------------------------------------I
      I14=(DPCAP-PCAP-DT)/CTIE
      I18(1)=DT*(CTS(1)-CTCR)*3/CTE
      I18(2)=DT*(CTS(1)-CTS(2))*3/CTE
      I18(3)=DT*(CTIR-CTS(1))*3/CTE
      I13=PCAP/ALPC
      I11=PCAP*CUF
      I10(1)=(PR*DT*(PRS(2)-PR)*3/FD)*PUF
      I10(2)=DT*(PRS(1)-PRS(2))*3/FD
      I10(3)=DT*(INR-PRS(1))*3/PC
      I8=PUP*PUCR*INPUT
      I=INT(PRICE/20)
      I6=CUTAB(I+1)+(CUTAB(I+2)-CUTAB(I+1))*(PRICE-I*20)/20
CI-----------------------------------------------------------------------I
CI            T I M E     S L I C E  8     (ENTITIES)                    I
CI-----------------------------------------------------------------------I
      CTIR=I14
      CTCR=CTCR+I18(1)
      CTS(2)=CTS(2)+I18(2)
      CTS(1)=CTS(1)+I18(3)
      CDR=I13
      INR=I11
      PR=I10(1)
      PRS(2)=PRS(2)+I10(2)
      PRS(1)=PRS(1)+I10(3)
      CR=I8
      EPCC=I6
CI-----------------------------------------------------------------------I
CI                                E N D                                  I
CI-----------------------------------------------------------------------I
      GO TO 50
 1000 LOCK  6
      STOP
      END
```

TABLE 7.1

OUTPUT FROM THE COMPUTER MODEL

TIME	PRICE	INV	INR	PR	CR
2.0	55.	5879.	594.	600.	706.
4.0	57.	5581.	592.	599.	575.
6.0	55.	5687.	610.	597.	519.
8.0	53.	5821.	651.	598.	547.
10.0	52.	5973.	694.	609.	464.
12.0	48.	6119.	718.	631.	535.
14.0	46.	6241.	718.	659.	632.
16.0	46.	6248.	697.	663.	697.
18.0	47.	6234.	639.	695.	718.
20.0	48.	6160.	591.	688.	705.
22.0	49.	6120.	535.	662.	689.
24.0	50.	6071.	509.	624.	659.
26.0	51.	6038.	502.	584.	616.
28.0	52.	5939.	511.	551.	577.
30.0	53.	5892.	531.	530.	547.
32.0	53.	5877.	558.	523.	519.
34.0	52.	5899.	588.	528.	517.
36.0	51.	5923.	616.	543.	522.
38.0	51.	5963.	635.	565.	547.
40.0	50.	6003.	644.	589.	568.
42.0	49.	6036.	642.	617.	594.
44.0	49.	6057.	631.	625.	618.
46.0	49.	6063.	614.	622.	637.
48.0	49.	6056.	594.	609.	635.
50.0	50.	6037.	577.	622.	631.
52.0	50.	6009.	564.	608.	629.
54.0	51.	5979.	557.	594.	605.
56.0	51.	5954.	557.	589.	588.
58.0	51.	5935.	563.	590.	574.
60.0	51.	5925.	574.	595.	564.
62.0	51.	5925.	585.	595.	561.
64.0	51.	5934.	597.	599.	562.
66.0	51.	5944.	605.	597.	561.
68.0	50.	5962.	610.	596.	575.
70.0	50.	5973.	611.	595.	587.
72.0	50.	5983.	608.	602.	596.
74.0	50.	5987.	602.	606.	603.
76.0	50.	5987.	595.	606.	606.
78.0	50.	5977.	588.	604.	606.
80.0	50.	5967.	582.	599.	602.
82.0	50.	5957.	578.	594.	597.
84.0	50.	5947.	577.	588.	591.
86.0	51.	5937.	579.	584.	586.
88.0	51.	5927.	582.	581.	581.
90.0	51.	5927.	585.	580.	578.
92.0	51.	5927.	591.	581.	577.
94.0	50.	5930.	595.	584.	578.
96.0	50.	5941.	597.	587.	582.
98.0	50.	5940.	599.	591.	587.
100.0	50.	5941.	598.	594.	589.

7.13 CHECK OF RESULTS

A graph of the states of entities INR, PR and PRICE is shown in fig. 7.9. These show the behaviour of the system for a duration of 100 months which is the same as that of Meadows [21,p.33].

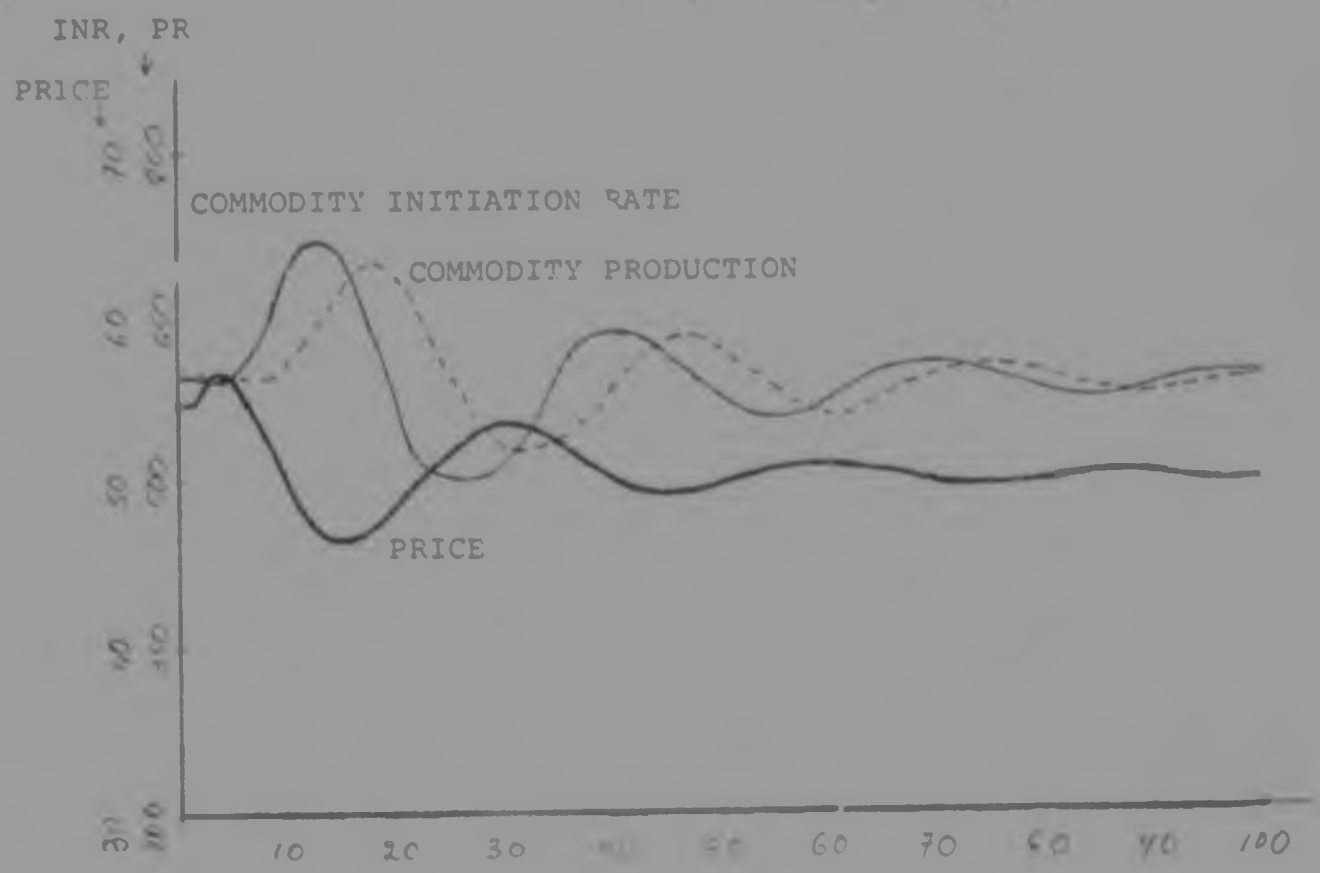


FIG. 7.9. An aspect of producer-distributor-consumer

The DYNAMO equations used by Meadows are reproduced below (fig. 7.10). It is interesting to note that DYNAMO is a package specialised for use in these kind of systems and offers facilities such as the DELAYn intrinsic which save instructions and variables. It is noted however that the two programs have approx. the same number of statements and therefore from the programming point of view, the present general method has not required more effort.

7.13 CHECK OF RESULTS

A graph of the states of entities INR, PR and PRICE is shown in fig. 7.9. These show the behaviour of the system for a duration of 100 months which is the same as that of Meadows [21,p.33].

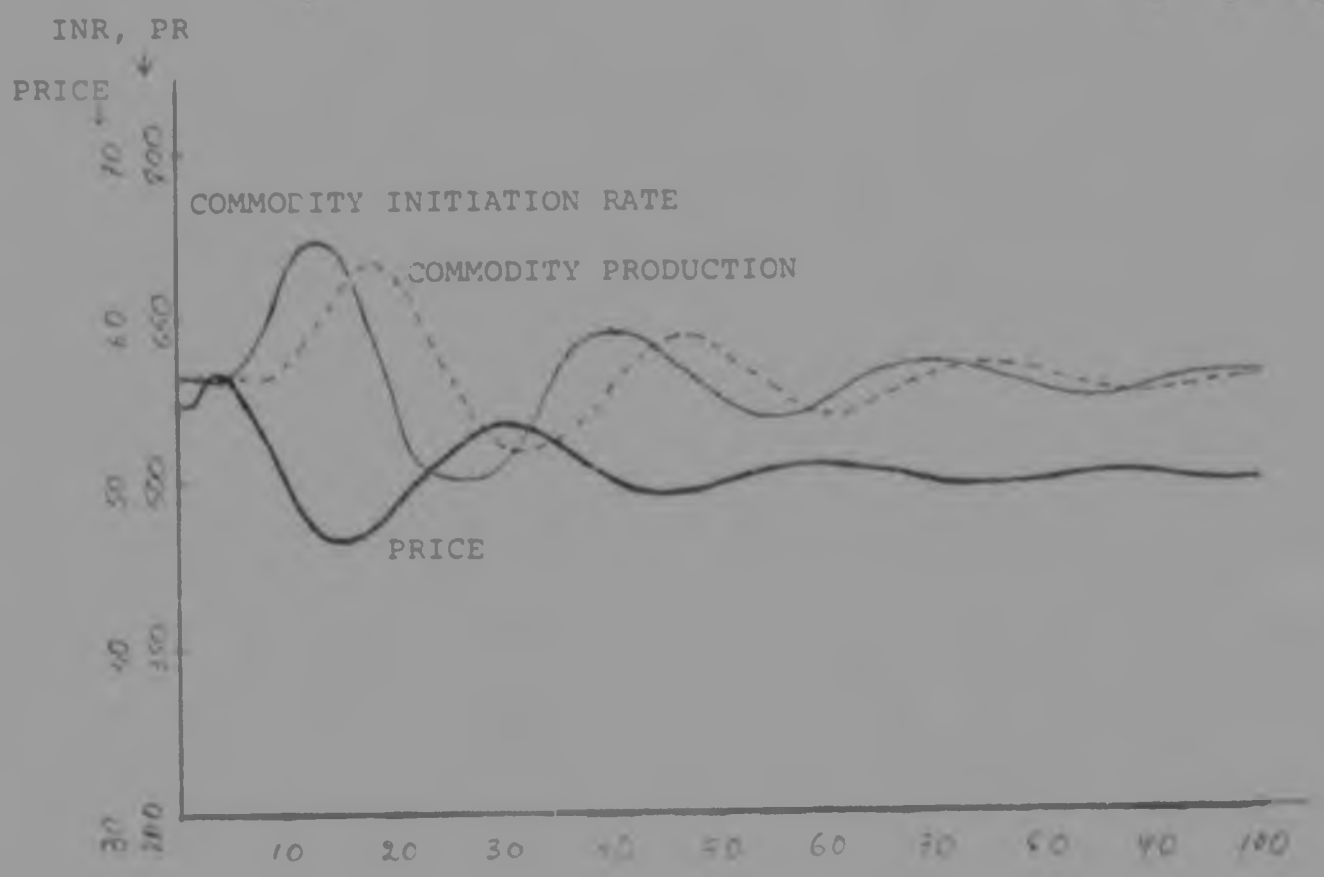


FIG. 7.9. An aspect of producer-distributor-consumer

The DYNAMO equations used by Meadows are reproduced below (fig. 7.10). It is interesting to note that DYNAMO is a package specialised for use in these kind of systems and offers facilities such as the DELAYn intrinsic which save instructions and variables. It is noted however that the two programs have approx. the same number of statements and therefore from the programming point of view, the present general method has not required more effort.

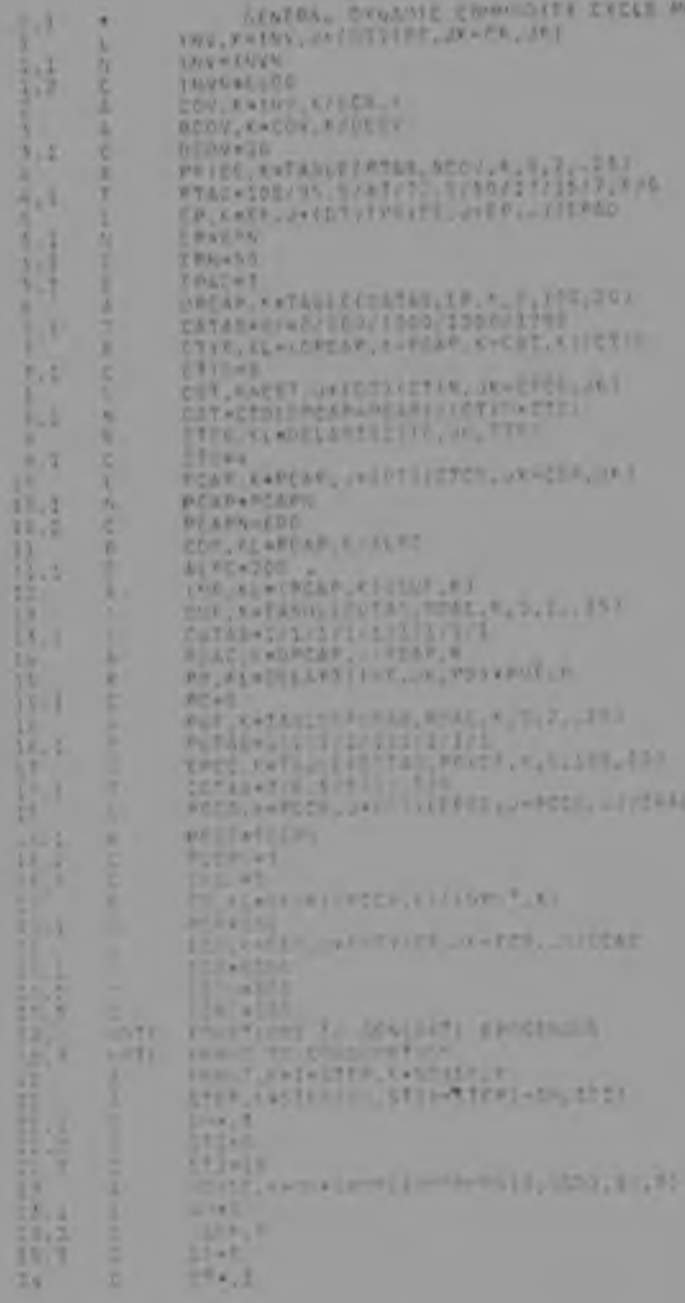


FIG. 7.10. DYNAMO model statements

7.14 CONCLUSIONS

This chapter has illustrated how the present methodology can be used as a standard procedure of modelling an arbitrary phenomenon (taken from the class of self-maintaining organised systems).

The actual program has a structure which is standard and identifies the entities and interactions with the algorithms which give their changes in time.

Because entities and interactions are the basic elements of phenomena as discussed in chapter 2, it is believed that the above type of models can be used as the standard method of representing phenomena thus making comparisons possible between phenomena of different physical nature as well as between theoretical models and groups of phenomena.

CHAPTER 8

MODELS OF SYSTEMS INVOLVING CONTINUOUS VARIABLES

8.1 INTRODUCTION

Chapter 6 presented a technique for constructing models of interaction networks in the form of computer programs which represent the structure of the network as well as produce its behaviour when run on the computer.

Chapter 7 applied this technique in detail to an example. The purpose of this was to demonstrate that the method does not break down under the complexity of details and to compare the results with those obtained on the same problem using a simulation language.

Since we want to use interaction networks for modelling, diverse physical systems, we will in this chapter represent a physically very different entity (a living cell) in the form of an interaction network. We will again (as in chapter 7) attempt to relate this method to existing methods of modelling. Here we will relate interaction networks to the established methodology for modelling systems which involve continuous variables; more specifically to differential equations and numerical methods for their solution. This is done in the following four sections:

Section 8.2 examines the physical system, represents it in the form of an interaction network, and constructs a computer model which will give us the behaviour of the system. This is the same as what we did for the production/consumption system in chapter 7.

Section 8.3 represents the same physical system in the form of a system of differential equations and relates this to interaction networks.

Section 8.4 examines the computer program models which are used to produce the behaviour of interaction networks and relates this to the numerical methods used to solve the system of differential equations.

Section 8.5 discusses the important issue of stability which must be considered when representing a system by using differential equations and numerical methods.

Section 8.6 uses an existing package for the Runge-Kutta method to produce the behaviour of the same system and section 8.7 compares the results with those of section 8.2.

8.2 REPRESENTATION OF HEINMETS' MODEL OF A CELL AS AN INTERACTION NETWORK

A model of a cell as a system of biochemical reactions will now be considered.

Heinmets, [23] and [24], has studied the modelling of biochemical processes in the cell. One of his studies is the modelling of the cellular growth process, and this was implemented on an analogue computer.

A system of 30 biochemical reactions which take place simultaneously in the cell, were simulated. Table 8.1 below shows this system of reactions as given by Heinmets.

TABLE 8.1

SYSTEM OF BIOCHEMICAL REACTIONS

1. $E_p + P_n \xrightarrow{k_1} [E_p P_n]$

2. $B' + E \xrightarrow{k_2} B$

3. $G_B + P_n \xrightarrow{k_3} G_B + B'$

4. $G_C + P_n \xrightarrow{k_4} G_C + C$

5. $G_p + P_n \xrightarrow{k_5} G_p + M_p$

6. $G_E + [E_p P_n] \xrightarrow{k_6} G_E + E_n + M$

7. $B + M \xrightarrow{k_7} N$

8. $B + M_p \xrightarrow{k_8} N_p$

9. $C + P_a \xrightarrow{k_9} [CP_a]$

10. $N + [CP_a] \xrightarrow{k_{10}} B + C + M + E$

11. $N_p + [CP_a] \xrightarrow{k_{11}} B + M_p + C + E_p$

12. $M \xrightarrow{k_{12}} P_n$

13. $M_p \xrightarrow{k_{12}} P_n$

14. $B \xrightarrow{k_{14}} P_1$

15. $C \xrightarrow{k_{15}} P_1$

16. $E \xrightarrow{k_{16}} P_1$

17. $E_p \xrightarrow{k_{17}} P_1$

18. $P_\epsilon + E \xrightarrow{k_{18}} P_1 + E$

19. $P_1 + E \xrightarrow{k_{19}} P_{II} + E$

20. $P_1 + E \xrightarrow{k_{20}} P_\delta + E$

21. $E_p + C \underset{k_{-21}}{\overset{k_{21}}{\rightleftharpoons}} [E_pC]$

22. $E + P_1 \underset{k_{-22}}{\overset{k_{22}}{\rightleftharpoons}} [EP_1]$

23. $E_p + B \underset{k_{-23}}{\overset{k_{23}}{\rightleftharpoons}} [E_pB]$

24. $N_p + s_1 \xrightarrow{k_{24}} N'_p$

25. $N'_p + s'_1 \xrightarrow{k_{25}} N_p$

26. $N + s_1 \xrightarrow{k_{26}} N$

27. $N' + s'_1 \xrightarrow{k_{27}} N$

28. $N \xrightarrow{k_{28}} P_1$

29. $N_p \xrightarrow{k_{29}} P_1$

30. $P_1 \xrightarrow{k_{30}} X$

Each of the above symbols (e.g. E_p, P_n, B' etc) represents one of the biochemical substances which are contained in the cell. These substances take part in chemical reactions where these substances are produced and/or consumed. For example in the chemical reaction 8 shown above

$$B + M_p \xrightarrow{k_8} N_p \quad .$$

Substances B and M_p combine to form the substance N_p.

Table 8.2 contains identification of the symbols used.

TABLE 8.2

TABLE OF TERMINOLOGY AND SYMBOLS

POOLS

P_e - Extracellular nutrient pool
P_i - General intracellular metabolic pool
P_a - Amino acid pool for protein synthesis
P_n - Nucleotide pool for RNA synthesis

ENZYMES

E - Total protein
E_p - RNA polymerase for messenger RNA (M) synthesis

GENES

G_E - Genes for messenger RNA (M) synthesis
G_P - Genes for messenger RNA (M_p) synthesis
G_B - Genes for the synthesis of RNA fraction of ribosome
G_C - Genes for transport RNA (C) synthesis

MESSENGERS

M - Messenger (RNA) for protein (E) synthesis

M_p - Messenger (RNA) for E_p synthesis

B' - RNA fraction of ribosome

B - Ribosome

C - Transport RNA

N - Ribosome and messenger complex for protein (E) synthesis (template)

N_p - Ribosome and messenger complex for E_p synthesis (template)

N' - Inactive state of N

N'_p - Inactive state of N_p

P_a - Metabolic which converts templates N and N_p into inactive state

P_i - Metabolic which converts inactive template N' and N_p into active state

COMPLEXES

$[E_p P_n]$ - complex of E_p and P_n

$[C\ P_a]$ - complex of C and P_a

$[E_p B\ ]$ - complex of E_p and B

$[E\ P_i]$ - complex of E and P_i

$[E_p\ C]$ - complex of E_p and C

$k_1 \ldots k_n$ - Various rate constants

Appendix 4 contains an extract from Heinmet's explanation of these reactions in biochemical terms.

Since we will need to incorporate these symbols in a computer program we will use slightly different symbols as shown in table 8.3 below.

TABLE 8.3

TABLE OF SYMBOL CONVERSION

(new symbol = old symbol replaced

$EP = E_p$	$NP = N_p$	$E = E$
$PN = P_n$	$N1 = N'$	$GB = G_B$
$EPPN = [E_p P_n]$	$PA = P_a$	$GC = G_C$
$B1 = B'$	$CPA = [CP_a]$	$GP = G_p$
$B = B$	$PI = P_i$	$GE = G_E$
$C = C$	$EPC = [E_I C]$	$SI = S_i$
$M = M$	$EPB = [E_p B]$	$SI1 = S'_i$
$MP = M_p$	$EP1 = [EP_1]$	
$N = N$	$NP1 = N'_p$	

The question now arises: can we use the approach to organised systems developed in this dissertation to represent the above system as was done for the production/consumption model in chapter 7?

In order to construct our interaction network we will let each interaction represent a chemical reaction and each entity represent a chemical substance. We thus construct the interaction network shown in fig. 8.1.

Note the following regarding the graphic symbols used:

Entities are represented by large circles containing the symbol of the chemical substance. Interactions are represented by the small circles containing the number of the chemical reaction they represent as per table 8.1 of chemical reactions.

Interactions number 31, 32, 33 represent chemical reactions -21, -22, -23 which are the reverse reactions of 21, 22, 23. As table 8.1 shows these three are reversible reactions and each will be represented by two interactions.

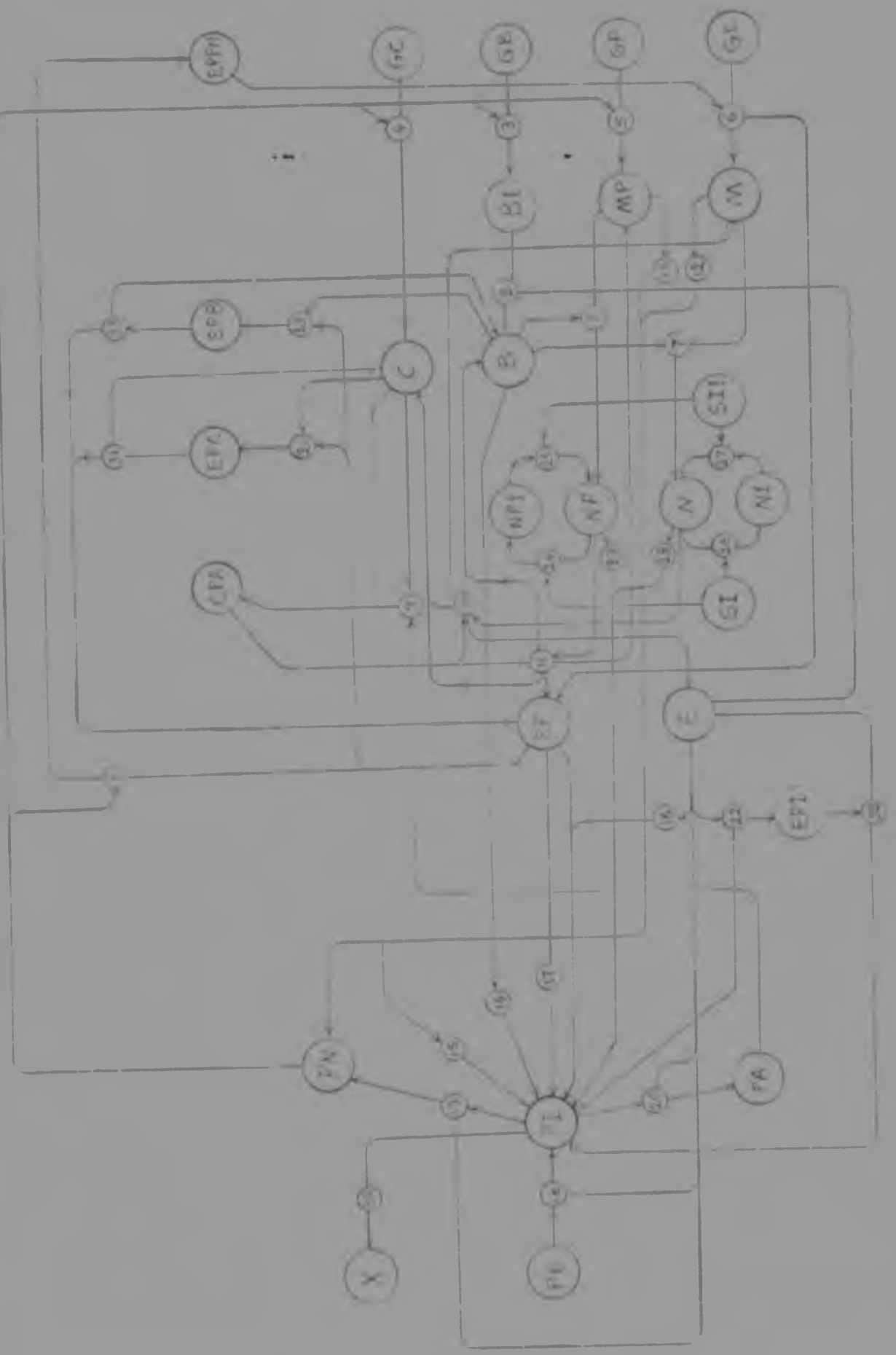


FIG. 8.1. Interaction network representing the cell

The lines represent as before the relationships between entities and interactions which in this physical system indicates that a chemical substance takes part in a certain chemical reaction. As indicated in 2.9 the presentation of these lines can be adjusted to enhance the communication value of the diagram. Here we did the following:

Arrows have been added to provide additional information:

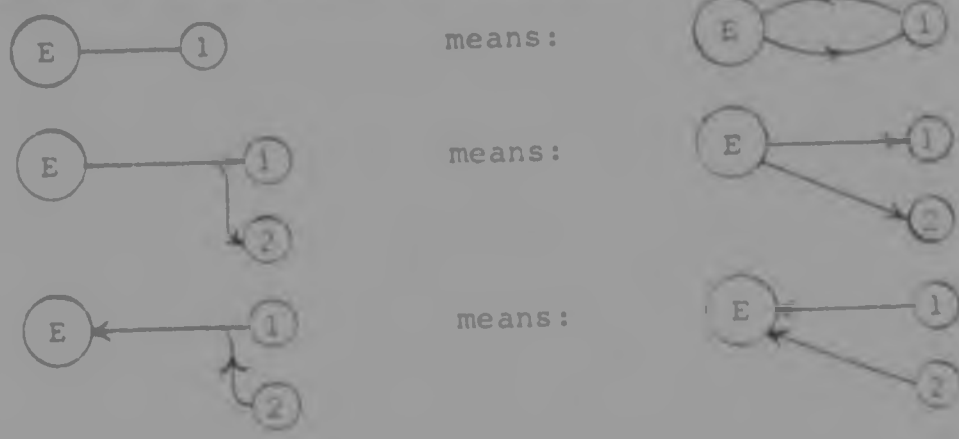


indicates that chemical substance E is a reactant in chemical reaction 1, i.e. it is being consumed.



indicates that E is a product of 1.

The part of the diagram which is covered by limes has been reduced to a minimum by adopting the following convention:



As in the production/consumption system, the next step after constructing the interaction network is temporal analysis. Here we want to establish whether there are any intermediate interactions (with a specific succession) between the initial and the final time slices of the total system.

This consideration in the present system doesn't reveal any succession of interactions. Unlike the production/consumption system of chapter 7, there is no procedure given here whereby certain interactions have to follow certain others in time. All the biochemical reactions given take place at the same time. Thus the model would consider only one interaction as shown in fig. 8.2 below (according to our concept of interaction developed in 2.6 and 2.7).

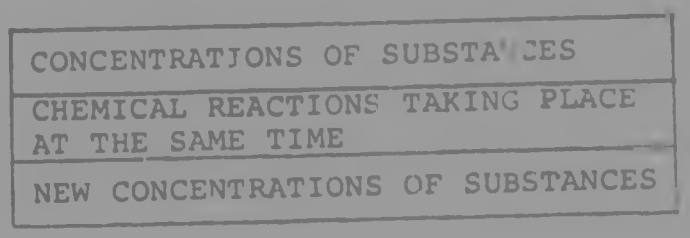


FIG. 8.2. Temporal analysis

We can now proceed to define exactly how each interaction mode and entity state will be calculated.

1. The mode of an interaction will represent the "extent" of the chemical reaction between the initial and the final time slices. If Δt represents the time interval between the initial time slice and the final time slice, then the mode of an interaction tells us how much chemical reaction took place during Δt. If we knew the rate of the chemical reaction during the interval then the interaction mode R_i would be

$$R_i = \Delta t \times C_i$$

2. The state of an entity is the concentration of that chemical substance in the cell. If we knew the change of the concentration during Δt then the state of the entity at the final time slice $X_{j,t}$ would be

$$X_{j,t} = X_{j,t-\Delta t} + \Delta X_j$$

where $X_{j,t-\Delta t}$ is the state of the entity at the initial time slice and ΔX_j is the change of state between initial and final time slices.

Now according to the laws of chemical kinetics [25] the rate of the chemical reaction of the type

$$X_1 + X_2 + X_3 \xrightarrow{k_1} X_4$$

is $C_i = k_1[X_1][X_2][X_3]$ (8.1) where k_1 is the chemical reaction constant and $[X_1][X_2][X_3]$ are the respective concentrations of reactants X_1, X_2, X_3.

Furthermore the change of concentration of a certain chemical substance is the algebraic sum of the changes produced by each chemical reaction in which it takes place (+ve if it is a product and -ve if it is a reactant). The magnitude of each of these changes

$$\Delta X = \Delta t \times C_i$$

where C_i is the rate of the chemical reaction producing the change. Δt is the time duration in which the change in the concentration of chemical substances occurs. In our interaction network this is the time duration between <u>initial</u> and <u>final</u> time slices (fig. 8.2).

Thus we will calculate our <u>interaction modes</u> using:

$$R_i = \Delta t \cdot k_i \cdot \prod_r X_r \qquad (8.2)$$

where X_r is the state of an entity representing a reactant for this chemical reaction.

Our <u>entity states</u> will be calculated using:

$$X_{j,t} = X_{j,t-\Delta t} + \sum_p R_p - \sum_r R_r \qquad (8.3)$$

where R_p and R_r are modes of interactions representing chemical reactions in which entity j is a product or a reactant respectively.

Based on the above formulae, the temporal analysis of fig. 8.2, the interaction network of fig. 8.1, and the table of biochemical reactions of table 8.1, we can produce the following standard

program model of the system. The notes following the program provide explanation of its contents.

```
$RESET FREE
FILE  6(TITLE="CIREP1",KIND=PRINTER)
FILE  7(TITLE="CIREP2",KIND=PRINTER)
FILE  8(TITLE="CIREP3",KIND=PRINTER)
FILE  9(TITLE="CIREP4",KIND=PRINTER)
CI----------------------------------------------------------------------I
CI                                D A T A                                I
CI----------------------------------------------------------------------I
      DIMENSION R(33),RK(33)
      REAL M,MP,N,NP,N1,NP1
      DATA RK/.0450,.0707,.0127,.0210,.0647,.1093,.1271,.0421,.1363,
     1.1533,.0407,.0973,.0263,.0245,.1077,.0103,.0109,.0803,.1007,
     2.1051,.0278,.0389,.0116,.1,.1,.1,.1,.0017,.0054,.03,.2091,.0088,
     3.1278/
      DATA EP,PN,B1,E,NP,CPA,EPC,B,EPH,EPPN,PI,MP,
     1PA,N,EPI,PE,M,GB,GC,C,GP,SI,SI1,N1,NP1,GE
     1/.2,.2,.1,.2,.2,.72,.049,.2,.008,.078,.2,.2,
     2.2,.2,.049,.115,.2,.2,.2,.2,.2,.2,.2,.2,.2,.2/
      DT=1
      T=0
CI----------------------------------------------------------------------I
CI        S U P E R V I S O R        R O U T I N E                       I
CI----------------------------------------------------------------------I
 50   IF (T.GT.100)GO TO 1000
      WRITE(6,100) T,PI,PA,PN,E,N
      WRITE(7,100) T,M,C,MP,EP,B
      WRITE(8,100) T,EPPN,B1,NP,N1,CPA
      WRITE(9,100) T,EPC,EPH,EPI,NP1
  100 FORMAT(1H ,I3,3X,5F10.6)
      T=T+DT
CI----------------------------------------------------------------------I
CI           T I M E    S L I C E   1    (INTERACTIONS)                  I
CI----------------------------------------------------------------------I
      R(1)=DT*RK(1)*EP*PN
      R(2)=DT*RK(2)*B1*E
      R(3)=DT*RK(3)*GB*PN
      R(4)=DT*RK(4)*GC*PN
      R(5)=DT*RK(5)*GP*PN
      R(6)=DT*RK(6)*GE*EPPN
      R(7)=DT*RK(7)*M*M
      R(8)=DT*RK(8)*B*MP
      R(9)=DT*RK(9)*C*PA
      R(10)=DT*RK(10)*N*CPA
      R(11)=DT*RK(11)*NP*CPA
      R(12)=DT*RK(12)*M
      R(13)=DT*RK(13)*MP
      R(14)=DT*RK(14)*B
      R(15)=DT*RK(15)*C
```

program model of the system. The notes following the program provide explanation of its contents.

```
$RESET FREE .
FILE  6(TITLE="CIREP1",KIND=PRINTER)
FILE  7(TITLE="CIREP2",KIND=PRINTER)
FILE  8(TITLE="CIREP3",KIND=PRINTER)
FILE  9(TITLE="CIREP4",KIND=PRINTER)
CI-----------------------------------------------------------------------I
CI                              D A T A                                  I
CI-----------------------------------------------------------------------I
      DIMENSION R(33),RK(33)
      REAL M,MP,N,NP,N1,NP1
      DATA RK/.0450,.0707,.0127,.0210,.0847,.1093,.1271,.0421,.1363,
     1.1533,.0407,.0973,.0263,.0245,.1077,.0103,.0109,.0803,.1007,
     2.1051,.0278,.0389,.0118,.1,.1,.1,.1,.0017,.0054,.03,.2091,.0088,
     3.1278/
      DATA EP,PN,B1,E,NP,CPA,EPC,B,EPB,EPPN,PI,MP,
     1PA,N,EPI,PE,M,GB,GC,C,GP,SI,SI1,N1,NP1,GE
     1/.2,.2,.1,.2,.2,.72,.049,.2,.008,.078,.2,.2,
     2.2,.2,.049,.115,.2,.2,.2,.2,.2,.2,.2,.2,.2,.2/
      DT=1
      T=0
CI-----------------------------------------------------------------------I
CI          S U P E R V I S O R        R O U T I N E                     I
CI-----------------------------------------------------------------------I
 50   IF (T.GT.100)GO TO 1000
      WRITE(6,100) T,PI,PA,PN,E,N
      WRITE(7,100) T,M,C,MP,EP,B
      WRITE(8,100) T,EPPN,B1,NP,N1,CPA
      WRITE(9,100) T,EPC,EPB,EPI,NP1
  100 FORMAT(1H ,I3,3X,5F10.6)
      T=T+DT
CI-----------------------------------------------------------------------I
CI            T I M E    S L I C E   I    (INTERACTIONS)                  I
CI-----------------------------------------------------------------------I
      R(1)=DT*RK(1)*EP*PN
      R(2)=DT*RK(2)*B1*E
      R(3)=DT*RK(3)*GB*PN
      R(4)=DT*RK(4)*GC*PN
      R(5)=DT*RK(5)*GP*PN
      R(6)=DT*RK(6)*GE*EPPN
      R(7) DT*RK(7)*B*M
      R(8)=DT*RK(8)*B*MP
      R(9)=DT*RK(9)*C*PA
      R(10)=DT*RK(10)*N*CPA
      R(11)=DT*RK(11)*NP*CPA
      R(12)=DT*RK(12)*M
      R(13)=DT*RK(13)*MP
      R(14)=DT*RK(14)*B
      R(15)=DT*RK(15)*C
```

```
      R(16)=DT*RK(16)*E
      R(17)=DT*RK(17)*EP
      R(18)=DT*RK(18)*E*PE
      R(19)=DT*RK(19)*E*PI
      R(20)=DT*RK(20)*PI*E
      R(21)=DT*RK(21)*EP*C
      R(22)=DT*RK(22)*E*PI
      R(23)=DT*RK(23)*EP*B
      R(24)=DT*RK(24)*SI*NP
      R(25)=DT*RK(25)*SI1*NP1
      R(26)=DT*RK(26)*SI*N
      R(27)=DT*RK(27)*SI1*N1
      R(28)=DT*RK(28)*N
      R(29)=DT*RK(29)*NP
      R(30)=DT*RK(30)*PI
      R(31)=DT*RK(31)*EPC
      R(32)=DT*RK(32)*EPI
      R(33)=DT*RK(33)*EPB
CI----------------------------------------------------------------------I
CI           T I M E     S L I C E    2     (ENTITIES)                   I
CI----------------------------------------------------------------------I
      EP=EP+R(11)-R(17)-R(21)+R(31)-R(23)+R(33)-R(1)+R(6)
      PN=PN+R(19)-R(1)+R(12)+R(13)-R(4)-R(5)-R(3)
      EPPN=EPPN+R(1)-R(6)
      B1=B1+R(3)-R(2)
      B=B+R(2)-R(7)-R(8)-R(14)+R(10)+R(11)-R(23)+R(33)
      C=C+R(4)-R(9)+R(10)+R(11)-R(15)-R(21)+R(31)
      M=M+R(5)-R(7)-R(12)+R(10)
      MP=MP+R(5)-R(6)+R(11)-R(13)
      N=N+R(7)-R(26)+R(27)-R(10)-R(28)
      NP=NP+R(6)-R(24)+R(25)-R(11)-R(29)
      N1=N1+R(26)-R(27)
      PA=PA+R(20)-R(9)
      CPA=CPA+R(9)-R(10)-R(11)
      PI=PI+R(18)-R(19)-R(20)-R(22)+R(32)+R(14)+R(15)+R(16)
     1      +R(17)+R(28)+R(29)-R(30)
      EPC=EPC+R(21)-R(31)
      EPB=EPB+R(23)-R(33)
      EPI=EPI+R(22)-R(32)
      NP1=NP1+R(24)-R(25)
      E=E+R(10)-R(16)-R(22)+R(32)-R(2)
CI----------------------------------------------------------------------I
CI                              E N D                                   I
CI----------------------------------------------------------------------I
      GO TO 50
 1000 LOCK 6
      LOCK 7
      LOCK 8
      LOCK 9
      END
```

NOTES:

The program is of standard structure (according to the criteria discussed in section 6.9) as was the case with the program for the production consumption model of section 7.12.

In the "DATA" section of the program we set the chemical reaction constants (RK) and the starting entity states. The values used have been obtained from reports on Heinmet's analogue computer simulation, as far as possible. For those where values were not available in Heinmet's reports, hypothetical values were used.

The "SUPERVISOR ROUTINE" reports the entity states after each iteration and decides when the run should be terminated.

The "TIME SLICE 1 (INTERACTIONS)" section of the program represents the INTERMEDIATE time slice of fig. 8.2 and produces the interaction modes. These are stored in variables R(1) to R(33) and correspond to interactions 1 ... 33 (fig. 8.1). The established formula for interaction modes (equation 8.2 above) is used to calculate each. The only information used to calculate each interaction mode is the entity states of the entities involved in that interaction.

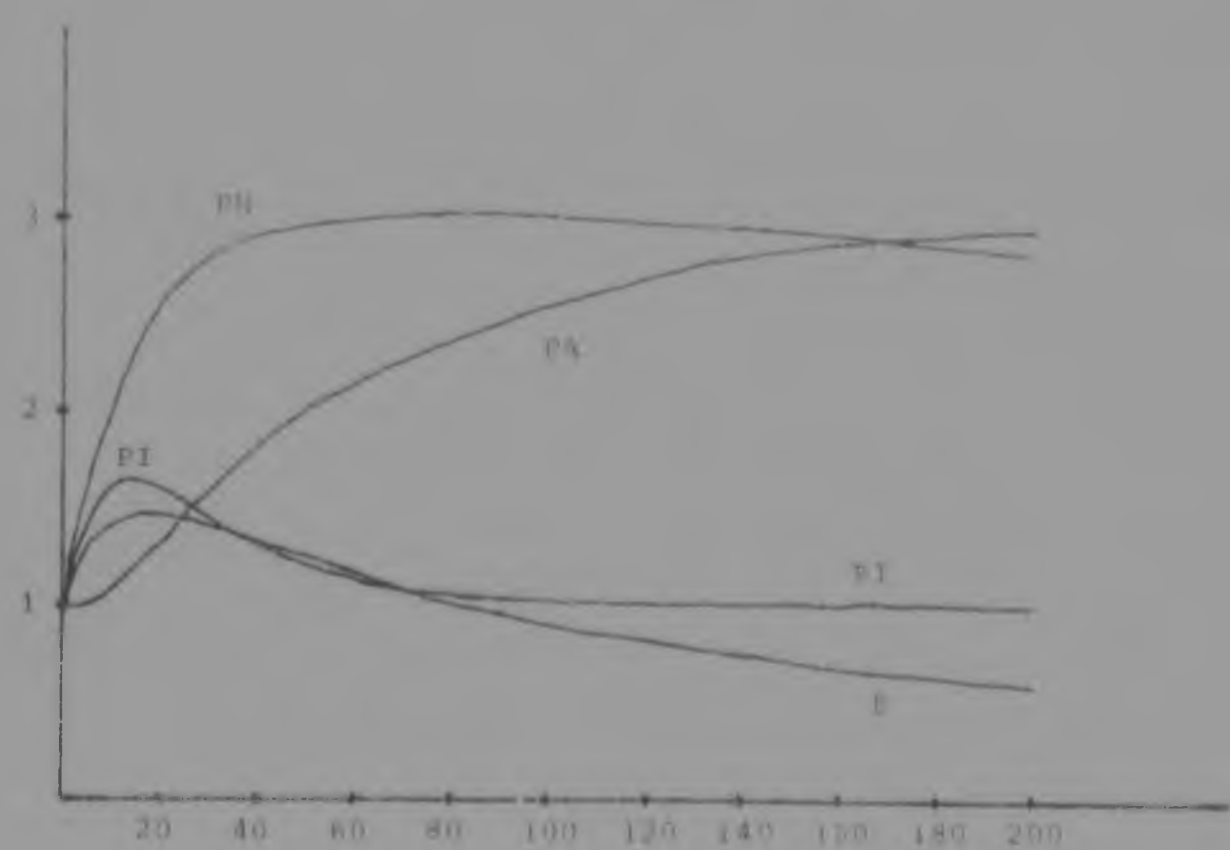
The "TIME SLICE 2 (ENTITIES)" section of the program represents the FINAL time slice of fig. 8.2, and produces the new entity states. These are stored in variables EP, PN,...E (see program) and represents the chemical substances bearing the same name in fig. 8.1. The established formula for entity states (equation 8.3 above) is used to calculate each. The only information used to calculate each entity state is its own present state and the modes of the interactions in which it is involved.

Certain entity states are not being calculated because these are constant and they retain their initial values: Entities GE, GC, GP, GB, GE are catalysts (according to our terminology). The net effect of the interactions on these is nil. Entities SI, SII, PE are specified by Heinmets as constant. Entity A is of no interest (according to Heinmets) since it represents all the disintegration products "dumped" by the cell to the outside world. If we wanted to show these entities in TIME SLICE 2 this would consist of re-assigning the initial values to these variables in the program.

Behaviour of the interaction network:

The output of this program is shown in Appendix 3. Each column of figures represents the state of an entity at different points in time.

The behaviour of four selected entities is shown in the graph below. The horizontal axis represents time in time units used in the program. The vertical axis is dimensionless and represents the ratio of the current entity state to the initial state. An extended run was done to the time units and this output is also shown in the graph.



8.3 INTERACTION NETWORKS AND DIFFERENTIAL EQUATIONS

Systems such as the one being considered here (biochemical) and many other physically different systems are normally modelled in the respective discipline as systems of ordinary differential equations of the form:

$$\frac{dX_i}{dt} = f_i(X_1,\ldots,X_n) \quad \text{for } i=1,\ldots,n$$

We would like to explain here the relationship between the representation of such a system as an interaction network and its representation as a system of differential equations.

Can we produce a system of differential equations to represent an interaction network?

We can under certain conditions: States and modes must be expressible as continuous variables. Then if we can express the modes of interactions as functions of the entity states involved in those interactions and then express the instantaneous rates of change of entity states in terms of the modes of the interactions in which they are involved, we may be able to express these in the form of systems of ordinary first order differential equations.

For simplicity let us consider the reversible chemical reaction

$$H_2 + I_2 \underset{k_1}{\overset{k_2}{\rightleftharpoons}} HI + HI$$

which was presented as an interaction network in section 2.8 and 6.9.

As discussed in section 8.2, the interaction modes according to equation 8.2 will be

$$R_1 = \Delta t \cdot k_1 \cdot HI \cdot HI \qquad (8.4)$$

$$R_2 = \Delta t \cdot k_2 \cdot H2 \cdot I2 \qquad (8.5)$$

where $H1$, $H2$, $I2$ are the entity states representing the chemical concentrations of these substances.

According to equation 8.3 the entity states will be given by

$$H2_t = H2_{t-\Delta t} + R_1 - R_2 \tag{8.6}$$

$$I2_t = I2_{t-\Delta t} + R_1 - R_2 \tag{8.7}$$

$$HI_t = HI_{t-\Delta t} + R_2 + R_2 - R_1 - R_1 \tag{8.8}$$

(Note that R_2 and R_1 appear twice in 8.8. This is due to the peculiarity of this particular chemical reaction which produces two molecules of HI in the right-hand side. In equation 8.3 based on chemical kinetics [25] the terms $\sum_p R_p$ and $\sum_r R_r$, each refers to a single molecule in the chemical reaction irrespective of the fact that we may have two or more of the same molecule which combine to form the concentration of a single substance.)

Now from (8.6), (8.7), (8.8) we see that the change in each entity state is

$$\Delta H2 = R_1 - R_2$$

$$\Delta I2 = R_1 - R_2$$

$$\Delta HI = 2R_2 - 2R_1$$

Substituting (8.4) and (8.5) in the above we have

$$\Delta H2 = \Delta t(k_1 \cdot HI^2 - k_2 \cdot H2 \cdot I2)$$

$$\Delta I2 = \Delta t(k_1 \cdot HI^2 - k_2 \cdot H2 \cdot I2)$$

$$\Delta HI = \Delta t \cdot 2 \cdot (k_2 \cdot H2 \cdot I2 - k_1 \cdot HI^2)$$

Letting $\Delta t \to 0$ for infinite resolution level we have

$$\frac{dH2}{dt} = k_1 \cdot HI^2 - k_{-1} \cdot H2 \cdot I2$$

$$\frac{dI2}{dt} = k_1 \cdot HI^2 - k_{-1} \cdot H2 \cdot I2 \qquad (8.9)$$

$$\frac{dHI}{dt} = 2(k_2 \cdot H2 \cdot I2 - k_1 \cdot HI^2)$$

We have thus arrived at the system of differential equations which represents this particular interaction network and in the process we related these two representations of the physical system.

The system (8.9) is the same well-known mathematical model which would normally be used in chemistry [25] to represent this chemical reaction.

A chemist would interpret the first equation of (8.9) as follows: the rate of change of hydrogen atoms is equal to their rate of production minus the rate of its consumption. The reaction that produces hydrogen is reaction 1 and its rate is $k_1 \cdot HI \cdot HI$ i.e. a constant times the product of the reactants. Similarly for the term $-k_{-1} \cdot H2 \cdot I2$ representing the reverse reaction which consumes hydrogen.

Coming back to our biochemical system (the cell), in exactly the same way we produce the system of differential equations which represents the interaction network (shown in table 8.4 below).

TABLE 8.4

DIFFERENTIAL EQUATIONS

1. $\dot{EP} = k_{11} \cdot NP \cdot CPA - k_{17} \cdot EP - k_{21} \cdot EP \cdot C + k_{31} \cdot EPC - k_{23} \cdot EP \cdot B + k_{33} \cdot EPB - k_1 \cdot EP \cdot PN + k_6 \cdot GE \cdot EPPN$

2. $\dot{PN} = k_{19} \cdot E \cdot PI - k_1 \cdot EP \cdot PN + k_{12} \cdot M - k_{13} \cdot MP - k_4 \cdot GC \cdot PN - k_5 \cdot GP \cdot PN - k_3 \cdot GB \cdot PN$

3. $\dot{EPPN} = k_1 \cdot EP \cdot PN - k_6 \cdot GE \cdot EPPN$

$$\left.\begin{aligned} \frac{dH2}{dt} &= k_1 \cdot HI^2 - k_1 \cdot H2 \cdot I2 \\ \frac{dI2}{dt} &= k_1 \cdot HI^2 - k_2 \cdot H2 \cdot I2 \\ \frac{dHI}{dt} &= 2(k_2 \cdot H2 \cdot I2 - k_1 \cdot HI^2) \end{aligned}\right\} \qquad (8.9)$$

We have thus arrived at the system of differential equations which represents this particular interaction network and in the process we related these two representations of the physical system.

The system (8.9) is the same well-known mathematical model which would normally be used in chemistry [25] to represent this chemical reaction.

A chemist would interpret the first equation of (8.9) as follows: the rate of change of hydrogen atoms is equal to their rate of production minus the rate of its consumption. The reaction that produces hydrogen is reaction 1 and its rate is $k_1 \cdot HI \cdot HI$ i.e. a constant times the product of the reactants. Similarly for the term $-k_1 \cdot H2 \cdot I2$ representing the reverse reaction which consumes hydrogen.

Coming back to our biochemical system (the cell), in exactly the same way we produce the system of differential equations which represents the interaction network (shown in table 8.4 below).

TABLE 8.4

DIFFERENTIAL EQUATIONS

1. $\dot{EP} = k_{11} \cdot NP \cdot CPA - k_{17} \cdot EP - k_{21} \cdot EP \cdot C + k_{31} \cdot EPC - k_{23} \cdot EP \cdot B + k_{33} \cdot EPB - k_1 \cdot EP \cdot PN + k_6 \cdot GE \cdot EPPN$

2. $\dot{PN} = k_{19} \cdot E \cdot PI - k_1 \cdot EP \cdot PN + k_{12} \cdot M - k_{13} \cdot MP - k_4 \cdot GC \cdot PN - k_5 \cdot GP \cdot PN - k_3 \cdot GB \cdot PN$

3. $\dot{EPPN} = k_1 \cdot EP \cdot PN - k_6 \cdot GE \cdot EPPN$

4. $\dot{B1} = k_3 \cdot GB \cdot PN - k_2 \cdot B1 \cdot E$

5. $\dot{B} = k_2 \cdot B1 \cdot E - k_7 \cdot B \cdot M - k_8 \cdot B \cdot MP - k_{14} \cdot B - k_{10} \cdot N \cdot CPA + k_{11} \cdot NP \cdot CPA - k_{23} \cdot EP \cdot B + k_{33} \cdot EPB$

6. $\dot{C} = k_4 \cdot GC \cdot PN - k_9 \cdot C \cdot PA + k_{10} \cdot N \cdot CPA + k_{11} \cdot NP \cdot CPA - k_{15} \cdot C - k_{21} \cdot E_p \cdot C + k_{31} \cdot EPC$

7. $\dot{M} = k_6 \cdot GE \cdot EPPN - k_7 \cdot B \cdot M - k_{12} \cdot M + k_{10} \cdot N \cdot CPA$

8. $\dot{MP} = k_5 \cdot GP \cdot PN - k_8 \cdot B \cdot MP + k_{11} \cdot NP \cdot CPA - k_{13} \cdot MP$

9. $\dot{N} = k_7 \cdot B \cdot M - k_{26} \cdot SI \cdot N + k_{27} \cdot SI1 \cdot N1 - k_{10} \cdot N \cdot CPA - k_{28} \cdot N$

10. $\dot{NP} = k_8 \cdot B \cdot MP - k_{24} \cdot SI \cdot NP - k_{25} \cdot SI1 \cdot NP1 - k_{11} \cdot NP \cdot CPA - k_{29} \cdot NP$

11. $\dot{N1} = k_{26} \cdot SI \cdot N - k_{27} \cdot SI1 \cdot N1$

12. $\dot{PA} = k_{20} \cdot PI \cdot E - k_9 \cdot C \cdot PA$

13. $\dot{CPA} = k_9 \cdot C \cdot PA - k_{10} \cdot N \cdot CPA - k_{11} \cdot NP \cdot CPA$

14. $\dot{PI} = k_{18} \cdot E \cdot PE - k_{19} \cdot E \cdot PI - k_{20} \cdot E \cdot PI - k_{22} \cdot E \cdot PI + k_{32} \cdot EPI + k_{14} \cdot B + k_{15} \cdot C - k_{16} \cdot E + k_{17} \cdot EP + k_{28} \cdot N + k_{29} {}^{+}NP - k_{30} \cdot PI$

15. $\dot{EPC} = k_{21} \cdot EP \cdot C - k_{31} \cdot EPC$

16. $\dot{EPB} = k_{23} \cdot EP \cdot B - k_{33} \cdot EPB$

17. $\dot{EPI} = k_{22} \cdot E \cdot PI - k_{32} \cdot EPI$

18. $\dot{NP1} = k_{24} \cdot SI \cdot NP - k_{24} \cdot SI1 \cdot NP1$

19. $\dot{E} = k_{10} \cdot N \cdot CPA - k_{16} \cdot F - k_{22} \cdot E \cdot PI + k_{32} \cdot EPI - k_2 \cdot E \cdot B1$

This is the same as the mathematical model used by Heinmets which was then run on an analogue computer [23] [24].

Let us check one of these equations (say equation 2). A chemist would interpret it as follows: The rate of change $\dot{PN}$ of substance PN is the sum of its production rates minus the sum of its consumption rates. It is produced by chemical reactions 19, 12, 13, and consumed by reactions 1, 4, 5, 3 as shown in table 8.1, hence the positive and negative terms

respectively in the right-hand side of the equation. Each of these terms is the respective reaction rate and is the product of the reaction constant and the reactant of the respective reaction, e.g. $k_1 \cdot EP \cdot PN$ is the rate of reaction 1.

We have thus seen that given a certain physical system which is represented as an interaction network, this can (under certain conditions) be expressed as a system of ordinary differential equations. In this case we can use existing numerical methods for solving these differential equations. This gives us an alternative tool (to that of our standard program model) for obtaining the behaviour of the system efficiently and accurately. This is discussed in the following sections.

8.4 RELATION OF INTERACTION NETWORKS AND NUMERICAL METHODS

When (if the appropriate conditions hold) an interaction network is expressed as a system of ordinary differential equations it also becomes possible to use the standard numerical methods to derive specific behaviours of the network.

One relationship which we want to investigate between interaction networks and these numerical methods is the following: as we discussed in chapter 6 we can use a program model of the interaction network to obtain its behaviour. What is the relationship between such a program and one which implements a standard numerical method?

As discussed in section 6.9 the purpose of the standard program model was two-fold: to provide a model (or representation) of

the structure of the phenomenon and when executed on the computer to provide a behaviour of the network. The program should also be able to serve as documentation of the structure of a class of phenomena. One more illustration of this follows:

Consider again the chemical reaction

$$H_2 + I_2 \underset{k_1}{\overset{k_2}{\rightleftharpoons}} HI + HI$$

which was discussed in sections 6 9 and 8.3. Its interaction network representation is reproduced here in fig. 8.5.

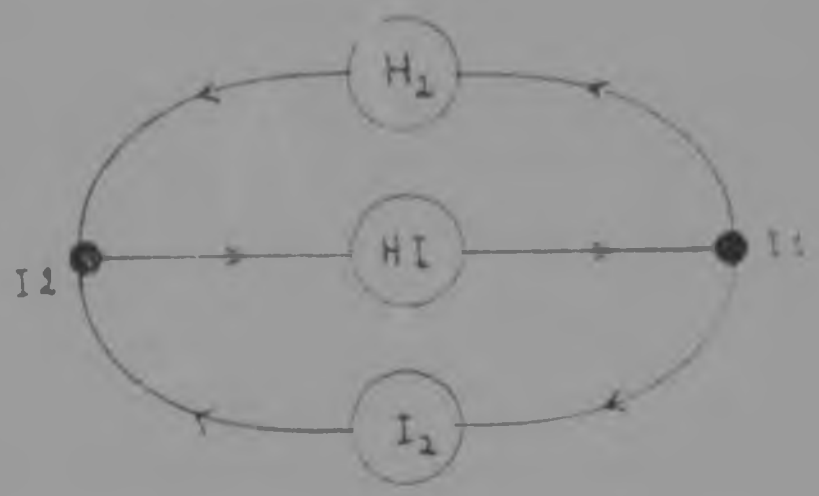


FIG. 8.5. Interaction network for chemical reaction

Now consider the phenomenon shown in fig. 8.6 below. It may represent a phenomenon where machines are assembled (I2) using materials (which may not be of interest here) as well as components 1 and 2. After a useful life the machines are scrapped after components 1 and 2 are extracted for re-use (these may be more durable and expensive than the rest of the components). The assembly (I2) may take place periodically when sufficient components 1 and 2 have accumulated to justify set-up costs. The scrapping (I1) may take place depending on the frequency of breakdowns, and components 1 and 2 re-used if they are in good condition depending on a certain test.

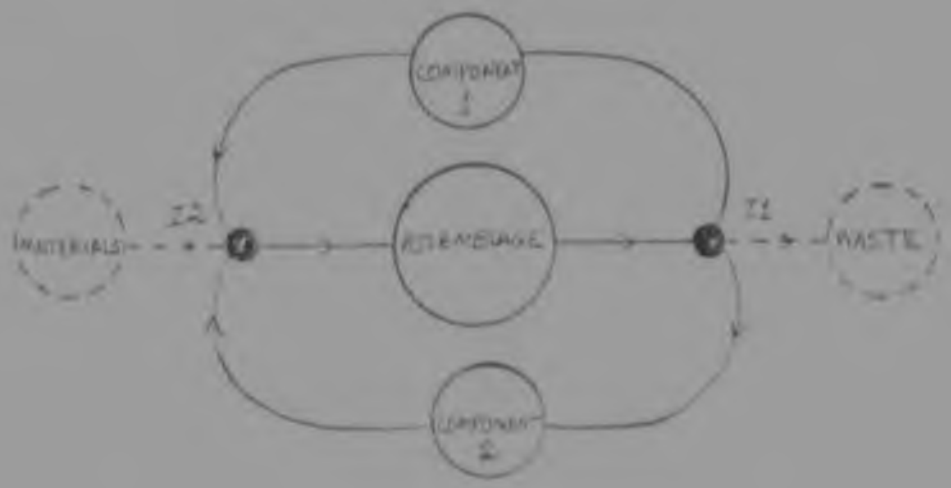


FIG. 8.6. Interaction network of component assembly

The above two systems can be represented by the same interaction network (fig. 8.7) and by the same program model of a specific

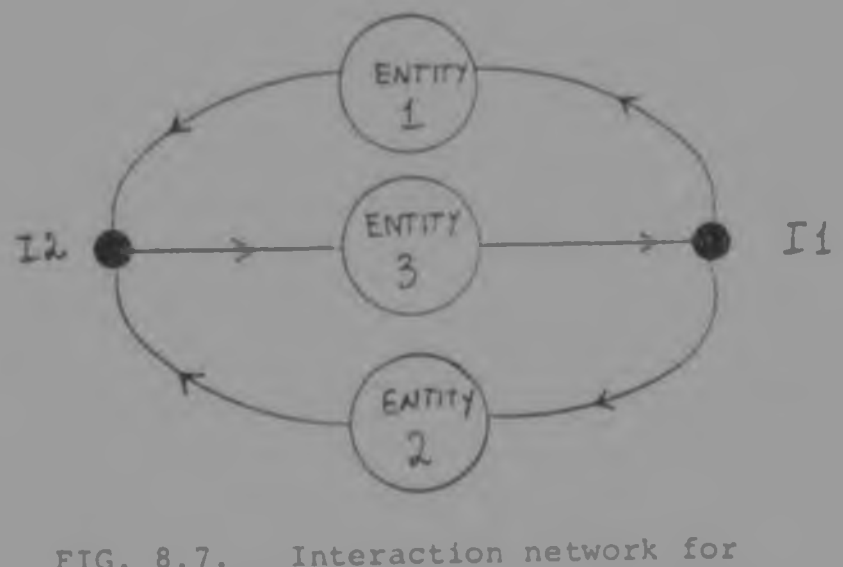


FIG. 8.7. Interaction network for component assembly and chemical reaction

structure. This is presented here in FORTRAN-like syntax (fig. 8.8). The symbol "←" signifies flow of data.

TIME SLICE 1 (interactions)

CALL SUBROUTINE-I2 (I2 ← E1, E2)
CALL SUBROUTINE-I1 (I1 ← E3)

TIME SLICE 2 (entities)

CALL SUBROUTINE-E1 (E1 ← I1, I_2)
CALL SUBROUTINE-E2 (E2 ← I1, I_2)
CALL SUBROUTINE-E3 (E3 ← I1, I2)

FIG. 8.8. Program model for component assembly and chemical reactions

This program represents the common structure of the two systems. It can furthermore produce the behaviour of either one of the two systems by calling the set of subroutines belonging to that particular system. For example, for the chemical reaction we would use equations 8.4 and 8.5 for time slice 1 and equations 8.6, 8.7 and 8.8 for time slice 2 respectively.

We thus have an example of a standard program model which serves the dual purpose of representing the structure of one or more systems as well as produce their behaviours.

Now suppose we could represent the interaction network of fig. 8.8 (chemical reaction) by a system of differential equations and then proceeded to use a certain numerical method to solve this system. The program implementing the algorithm of this particular numerical method will have a certain structure (of communication and control [27]). This structure may or may not be the same as the structure of our standard program (fig. 8.8) representing the interaction network. Thus, if we use a numerical method, the program which will give us the behaviour of the system (by being run on the computer) may not also have the structure of the interaction network.

In order to illustrate this point let us examine thi: problem for the usage of two different numerical methods: Euler's and Runge Kutta.

If we used the mathematical model for the system which we produced in 8.3 (system of ordinary differential equations 8.9) and we used Euler's method to solve it, we could use the following program statements for each iteration (where = is the assignment statement, and DT is the time step).

```
H2 = H2 + DT * (K1 * HI * HI - K2 * H2 * I2)
I2 = I2 + DT * (K1 * HI * HI - K2 * H2 * I2)
HI = HI + DT * 2 * (K2 * H2 * I2 - K1 * HI * HI)
```

On the other hand each iteration in the standard program model for this system is (as discussed above):

```
TIME SLICE 1 (interactions)

R1 = DT * K1 * HI * HI
R2 = DT * K2 * H2 * I2

TIME SLICE 2 (entities)

H2 = H2 + R1 - R2
I2 = I2 + R1 - R2
HI = HI + 2 * R2 - 2 * R1
```

Comparing these two programs we can see that our standard program is mathematically the same as Euler's method. The structural difference between it and the above algorithm is that in the standard program, in the first step (TIME SLICE 1) we calculate the intermediate variables R1, R2 (which are our interaction modes) using only information of the entity states. We then (TIME SLICE 2) calculate the new entity states using the interaction modes.

Let us now consider the compatibility of the structure of our

standard programs with the corresponding program obtained using a higher order numerical method: a Runge Kutta four step method.

We take as an example the following system.

$$\left.\begin{aligned}\frac{dy}{dt} &= f_1(t,y,u)\\ \frac{du}{dt} &= f_2(t,y,u)\end{aligned}\right\}$$

If t is the time at the beginning of the interval h and y_i and y_{i+1} are the values of y at the beginning and end of the interval respectively, then the formulae for the Runge-Kutta fourth-order method are:

$$y_{i+1} = y_i + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

where

$$k_1 = (h) f_1(t, y_i, u_i)$$

$$k_2 = (h) f_1\left(t + \frac{h}{2},\ y_i + \frac{k_1}{2},\ u_i + \frac{q_1}{2}\right)$$

$$k_3 = (h) f_1\left(t + \frac{h}{2},\ y_i + \frac{k_2}{2},\ u_i + \frac{q_2}{2}\right)$$

$$k_4 = (h) f_1(t + h,\ y_i + k_3,\ u_i + q_3)$$

and

$$u_{i+1} = u_i + \frac{1}{6}(q_1 + 2q_2 + 2q_3 + q_4)$$

where

$$q_1 = (h) f_2(t,\ y_i,\ u_i)$$

$$q_2 = (h) f_2\left(t + \frac{h}{2},\ y_i + \frac{k_1}{2},\ u_i + \frac{q_1}{2}\right)$$

$$q_3 = (h) f_2\left(t + \frac{h}{2},\ y_i + \frac{k_2}{2},\ u_i + \frac{q_2}{2}\right)$$

$$q_4 = (h) f_2(t + h,\ y_i + k_3,\ u_i + q_3)$$

The above suggest the following algorithm:

Using the starting entity states we calculate some intermediate values (k_1, q_1). Then using these we obtain changed entity states $y_i + \frac{k_1}{2},\ u_i + \frac{q_1}{2}$. Then using these new states we get a second set of intermediate values (k_2, q_2),... etc., until finally we calculate the final entity states using all the above intermediate values.

The communication and control structure of the above can be represented by fig. 8.9 where solid arrows represent flow of information and broken arrows flow of control.

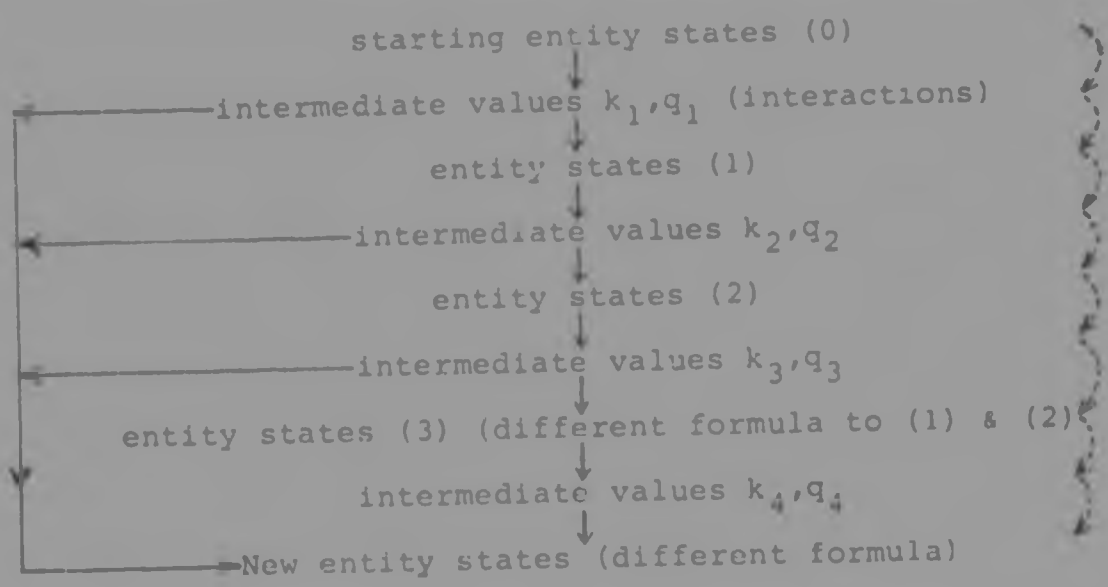


FIG. 8.9. Communication and control structure of Runge-Kutta method

This cannot be reconciled with our standard program structure shown in fig. 8.10 where we calculate in the first step the interaction modes based only on the entity states involved in each interaction and in the second step the new entity states each based only on the modes of the interactions in which it is involved.

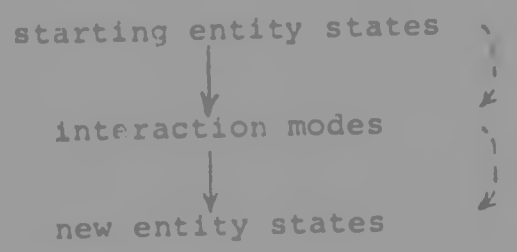


FIG. 8.10. Communication and control structure

We encounter similar structure problems with predictor-corrector methods. For example consider Milne's predictor corrector method:

Given the derivative of x as

$$x' = f(t,z)$$

the process of solution is as follows:

1st: the value x_{i+1} is predicted by using the following predictor equation:

$$\bar{x}_{i+1} = x_{i-3} + \frac{4}{3}h\,[2x' - x'_{i-1} + 2x'_{i-2}]$$

2nd: we use the following corrector equation:

$$x_{i+1} = x_{i-1} + \frac{h}{3}(x'_{i-1} + 4x'_i + \bar{x}'_{i+1})$$

where

$$\tilde{x}_{i+1} = f(t+1,\ \tilde{x}_{i+1})$$

Basically the problem here regarding program structure is that one has to include information from future time slices, and one cannot calculate interaction modes in terms of states of entities in the previous time slice alone. This means the program structure does not reflect the succession of time slices of the spatiotemporal object.

We conclude therefore that our standard program is mathematically equivalent to Euler's method and programs implementing numerical methods such as the Runge-Kutta method and predictor corrector methods are not compatible as models of structure of interaction networks.

However this does not mean that we cannot use them here. Obviously there are many numerical methods that are better than Euler's from the point of view of accuracy, stability and computational economy, therefore it is desirable to use them when these criteria are important. The problem of program structure discussed above can be bypassed by simply not using these programs as models of structure. We can use these methods while we conceptually regard them as black boxes. Having interpreted the interaction network as a system of ordinary differential equations we can use these black boxes purely to derive the behaviour of the system (efficiently and accurately). The documentation of the structure of the system is done by the interaction network and the standard program model. Thus apart from Euler's method they can all be used as special tools to derive behaviour, when accuracy and computational economy are important.

Having now related interaction networks and their standard program models, via ordinary differential equations to numerical methods and having established that there is a place in our methodology for the usage of standard numerical methods, we will now discuss some of the main considerations in choosing such methods (more details in 8.5). Then, in section 8.6 we will actually use one of these methods on our model of a cell.

8.5 CONSIDERATIONS OF STABILITY IN USING DIFFERENTIAL EQUATIONS AND NUMERICAL METHODS

As was pointed out above we can use a system of first order ordinary differential equations and a numerical method to predict the behaviour of an interaction network. There are, however, problems associated with the choice and usage of numerical methods for solving systems of ordinary differential equations.

Since these methods have a place in our methodology and can be used under certain circumstances, it would be appropriate for the sake of completeness to review here the main problem areas of these methods.

In this section we will consider the general case and its two main problem areas: The stiffness of the system of differential equations and the convergence of the numerical method used.

8.5.1 Stiffness of a set of differential equations

Consider the case where the interaction network is represented by the following system of linear first-order differential equations.

$$\left.\begin{aligned} \frac{dy_1}{dt} &= a_{11}y_1 + a_{12}y_2 + \ldots + a_{1n}y_n \\ &\vdots \\ \frac{dy_n}{dt} &= a_{n1}y_1 + a_{n2}y_2 + \ldots + a_{nn}y_n \end{aligned}\right\} \quad \begin{aligned} y(t_0) &= y_{01} \\ &\vdots \\ y_n(t_0) &= y_{0n} \end{aligned}$$

or in matrix representation: (1)

$$y' = \frac{dy}{dt} = Ay \,, \quad y(t_0) = y_0 \quad \text{where} \quad A = (a_{ij})_{n\times n} \text{ is constant,}$$

and $y = \begin{vmatrix} y_1 \\ \vdots \\ y_n \end{vmatrix}$, $y' = \begin{vmatrix} y_1' \\ \vdots \\ y_2 \end{vmatrix}$ and $y_0 = \begin{vmatrix} y_{01} \\ \vdots \\ y_{0n} \end{vmatrix}$

Diagonalizing A (if possible) we have that

$$V^{-1} A V = D = \text{diag}(\lambda_1,\ldots,\lambda_n) \quad \text{say}$$

where $V = [V_1, V_2, \ldots, V_n]$ and the $V_1,\ldots,V_n$ are eigenvectors (for $\lambda_1,\ldots,\lambda_n$) which forms a basis for n-space (2)

Then we can change variables by using z such that

$$y = Vz \tag{3}$$

which satisfies $y' = V\frac{dz}{dt} = Vz'$.

Since the V's are constant vectors, we can write (1) as

$$V\frac{dz}{dt} = A(Vz) \quad \text{and} \quad Vz(t_0) = y_0 \tag{4}$$

$$\Rightarrow \frac{dz}{dt} = V^{-1}AVz = Dz, \quad z(t_0) = z_0$$

where $z_0 = V^{-1}y_0 = [z_{01},\ldots,z_{0n}]$

The solution of t[illegible]ncoupled system is

$$z(t) = \begin{bmatrix} z_{01}\, e^{\lambda_1(t-t_0)} \\ \vdots \\ z_{0n}\, e^{\lambda_n(t-t_0)} \end{bmatrix}$$

and from (2) and (3) we have

$$y(t) = Vz(t) = z_{01}\, e^{\lambda_1(t-t_0)} V_1 + \ldots + z_{0n}\, e^{\lambda_n(t-t_0)} V_n \tag{5}$$

where $z_0 = V^{-1}y_0$.

In other words the solution of (1) is a weighted sum of the eigenvectors of A in which the "weight" of V_i is the

exponential $z_{0i}\, e^{\lambda_i t}$ where

$$A V_i = \lambda_i V_i \quad \text{and} \quad z_0 = V^{-1} y_0$$

and the $\lambda_1, \ldots, \lambda_n$ are the eigenvalues.

If $|\lambda_1|$ is the smallest and $|\lambda_n|$ the largest (dominant) eigenvalue and $|\lambda_1| \ll |\lambda_n|$ then the n-th term in (8) dominates the solution and we have a stiff set of differential equations.

For the general nonlinear set of ordinary differential equations the same considerations of stiffness apply where instead of matrix A we consider the Jacobian matrix

$$\frac{\partial f(y)}{\partial y} = \left[\partial f_i(y)/\partial y_j\right]_{n \times n}$$

Thus the stiffness of the system of ordinary differential equations depends on the eigenvalues of the Jacobian at the relevant instant in time.

In practice obtaining the eigenvalues of such a matrix for a large system is a major computational problem and therefore in practice one does not establish analytically whether the system is stable or not but rather one obtains a numerical solution using one of the packages that can handle unstable (stiff) systems. One such package is the DGEAR routine in the IMSL library of routines [29].

8.5.2 Convergence of the numerical method

The minimal property which a method must possess is that of convergence. For interaction networks, this means that the numerically obtained entity state y and the exact solution $y(t_n)$ of the ordinary differential equations must approach

each other as the temporal resolution becomes finer.

A numerical method for an ordinary differential equation which satisfies the Lipschit's property is convergent if the numerical solution y_j converges uniformly to $y(t_j)$.

for $x_j \in [t_0, e]$ $y_0 = y(t_0)$

for $\Delta t = \frac{(e - t_0)}{N}$ as $N \to \infty$.

The necessary and sufficient conditions for a method to be convergent is that it must be consistent and stable [30].

A numerical method is consistent if its order is $p \geq 1$.

A numerical method is stable for a given size step h and a given ordinary differential equation if the deviation δ in one of the values y_n has the result that the deviation on all subsequent solutions y_m $(m>n)$ is a bounded multiple of δ.

We have absolute stability if the deviations for all the solutions y_m $(m \geq n)$ do not exceed δ.

The solution of $y(t)$ of the differential equation $y'(t) = f(t, y(t))$ is stable with respect to changes in the initial values if for $\varepsilon > 0$ a $\delta > 0$ such that for any other solution $z(t)$ of the differential equation such that

$$\|y(t_0) - z(t_0)\| < \delta$$

implies $\|y(t) - z(t)\| < \varepsilon \ \forall \ t \in (t_0, \infty)$.

We have relative stability

$$\text{if } \|y(t) - z(t)\| \leq \varepsilon \|y(t)\| \quad \forall \quad t \in (t_0, \infty) \text{ .}$$

In modelling of interaction networks we have systems of

differential equations, and this is stable if each solution is stable, i.e. if the above apply for each entity state.

8.5.3 Numerical stability of the standard program model:

As discussed in section 8.4 the standard program model which represents the structure of the interaction network (in addition to producing its behaviour) is mathematically equivalent to Euler's numerical method. Therefore due to its importance in our methodology we should take a look at the stability of this method:

If we have an error ϵ_j at $t = t_i$

and we have $z_{j+1} = z_j + h f(t_j, z_j)$ (6)

instead of $y_{j+1} = y_j + h f(t_j, y_j)$ (7)

where $|z_j - y_j| = |\epsilon_j|$.

The next error will be from (6) and (7)

$$\epsilon_{j+1} = z_j - y_j + h[f(t_j, \epsilon_j) - f(t_j, y_j)]$$

and using the Lipschitz [31] condition which must hold we have

$$|\epsilon_{j+1}| \leq |\epsilon_j|(1 + h\lambda) \quad (5)$$

Repeated application of (5)

gives us $|\epsilon_n| \leq [1 + h\lambda]^{N-J} |\epsilon_j|$

which means that the deviation at $t = t_n$ is a bounded multiple of the deviation at $t = t_i$. Therefore our standard program model is numerically stable.

8.6 USING A NUMERICAL METHOD

In this section we use the system of ordinary differential equations of Table 8.4. These as developed and discussed in section 8.3 are an alternative (mathematical) model of the biochemical system under study. This system was modelled as an interaction network in section 8.2 and its behaviour was derived.

Now we will use a numerical method (Runge-Kutta of order 5) to derive the behaviour of this system of differential equations. As discussed at the end of section 8.4, this will be used as a black box. We have used here a subroutine (DVERK) from an existing library (IMSL [29]) on a Burroughs B6800 mainframe computer (see Appendix D). The source listing of the program is shown below.

```
$RESET FREE
FILE  6(TITLE="CRKREP1",KIND=PRINTER)
FILE  7(TITLE="CRKREP2",KIND=PRINTER)
FILE  8(TITLE="CRKREP3",KIND=PRINTER)
FILE  9(TITLE="CRKREP4",KIND=PRINTER)
CI-----------------------------------------------------------------I
CI                      M A I N    R O U T I N E                   I
CI-----------------------------------------------------------------I
      INTEGER N,IND,NW,IER,K
      EXTERNAL FCN1
      REAL Y(19),C(24),W(19,9),X,TOL,XEND
      NW=19
      N=19
      X=0.0
      DATA Y/.2,.2,.078,.1,.2,.2,.2,.2,.2,.2,
     1 .2,.2,.72,.2,.049,.008,.049,.2,.2/
      TOL=.00001
      IND=1
      DO 10 K=1,100
        XEND=FLOAT(K)
        CALL DVERK(N,FCN1,X,Y,XEND,TOL,IND,C,NW,W,IER)
        IF (IND.LT.0.OR.IER.GT.0) GO TO 20
        WRITE(6,100)X,Y(14),Y(12),Y(2),Y(19),Y(9)
        WRITE(7,100)X,Y(7),Y(6),Y(8),Y(1),Y(5)
        WRITE(8,100)X,Y(3),Y(4),Y(10),Y(11),Y(13)
        WRITE(9,100)X,Y(15),Y(16),Y(17),Y(18)
100     FORMAT(1H ,I3,3X,5F9.6)
10      CONTINUE
      LOCK 6
      LOCK 7
      LOCK 8
      LOCK 9
      STOP
20    CONTINUE
      STOP
      END
```

```
CI-----------------------------------------------------------------I
CI                                                                 I
CI                        S Y S T E M   O F                        I
CI          D I F F E R E N T I A L   E Q U A T I O N S            I
CI-----------------------------------------------------------------I
      SUBROUTINE FCN1(N,X,Y,YPRIME)
      INTEGER N
      REAL Y(N),YPRIME(N),X,RK(33)
      IF (IFIRST.EQ.1) GO TO 50
      DATA RK/.0456,.0707,.0127,.0210,.0847,.1093,.1271,.0421,.1363,
     1.1533,.0407,.0973,.0213,.0245,.0847,.0103,.0109,.0803,.1007,
     2.1051,.0278,.0389,.0118,.1,.1,.1,.1,.0017,.0054,.03,.2091,.0088,
     3.1277/
      DATA GE,GC,GP,GB,SI1,SI,PE/.2,.2,.2,.2,.2,.2,.1157
      IFIRST =1

      YPRIME(1)=RK(11)*Y(10)*Y(13)-RK(17)*Y(1)
     1 -RK(21)*Y(1)*Y(6)+RK(31)*Y(15)
     2 -RK(23)*Y(1)*Y(5)+RK(33)*Y(16)-RK(1)*Y(1)*Y(2)+RK(6)*GE*Y(3)
      YPRIME(2)=RK(19)*Y(18)*Y(14)-RK(1)*Y(1)*Y(2)
     1 +RK(12)*Y(7)*RK(13)*Y(8)
     2 -RK(4)*GC*Y(2)-RK(5)*GP*Y(2)-RK(3)*GB*Y(2)
      YPRIME(3)=RK(1)*Y(1)*Y(2)-RK(6)*GE*Y(3)
      YPRIME(4)=RK(3)*GB*Y(2)-RK(2)*Y(4)*Y(19)
      YPRIME(5)=RK(2)*Y(4)*Y(19)-RK(7)*Y(5)*Y(7)-RK(8)*Y(5)*Y(8)
     1 -RK(14)*Y(5)+RK(10)*Y(9)*Y(13)
     2 +RK(11)*Y(10)*Y(13)-RK(23)*Y(1)*Y(5)+RK(33)*Y(16)
      YPRIME(6)=RK(4)*GC*Y(2)-RK(9)*Y(6)*Y(12)
     1 +RK(10)*Y(9)*Y(13)+RK(11)*Y(10)*Y(13)
     2 -RK(15)*Y(6)-RK(21)*Y(1)*Y(6)+RK(31)*Y(15)
      YPRIME(7)=RK(6)*GE*Y(3)-RK(7)*Y(5)*Y(7)
     1 -RK(12)*Y(7)+RK(16)*Y(9)*Y(13)
      YPRIME(8)=RK(5)*GP*Y(2)-RK(8)*Y(5)*Y(8)
     1 +RK(11)*Y(10)*Y(13)-RK(13)*Y(8)
      YPRIME(9)=RK(7)*Y(5)*Y(7)-RK(26)*SI*Y(9)+RK(27)*SI1*Y(11)
     1 -RK(10)*Y(9)*Y(13)-RK(28)*Y(9)
      YPRIME(10)= RK(8)*Y(5)*Y(8)-RK(24)*SI*Y(10)
     1 +RK(25)*SI1*Y(11)
     2 -RK(11)*Y(10)*Y(13)-RK(29)*Y(10)
      YPRIME(11)= RK(26)*SI*Y(9)-RK(27)*SI1*Y(11)
      YPRIME(12)= RK(20)*Y(14)*Y(19)-RK(9)*Y(6)*Y(12)
      YPRIME(13)=RK(9)*Y(6)*Y(12)-RK(10)*Y(9)*Y(13)-RK(11)*Y(10)*Y(13)
      YPRIME(14)=RK(18)*Y(19)*PE-RK(19)*Y(19)*Y(14)
     1 -RK(20)*Y(14)*Y(19)-RK(22)*Y(19)*Y(14)+RK(32)*Y(17)
     2 +RK(14)*Y(5)+RK(15)*Y(6)+RK(16)*Y(19)+RK(17)*Y(1)+RK(28)*Y(9)
     3 +RK(29)*Y(10)-RK(30)*Y(14)
      YPRIME(15)=RK(21)*Y(1)*Y(6)-RK(31)*Y(15)
      YPRIME(16)= RK(23)*Y(1)*Y(5)-RK(33)*Y(16)
      YPRIME(17)=RK(22)*Y(19)*Y(14)-RK(32)*Y(17)
      YPRIME(18)=RK(24)*SI*Y(10)-RK(25)*SI1*Y(18)
      YPRIME(19)=RK(10)*Y(9)*Y(13)-RK(18)*Y(19)
     1 -RK(22)*Y(19)*Y(14)+RK(32)*Y(17)-RK(2)*Y(4)*Y(19)

      RETURN
      END
CI-----------------------------------------------------------------I
CI        S E T   O F   S T A N D A R D   S U B R O U T I N E S    I
CI                          ( I M S L )                            I
CI-----------------------------------------------------------------I
$INCLUDE     "DVERK"
$INCLUDE     "UERTST"
$INCLUDE     "UGETIO"
```

Notes on the source program:

The MAIN ROUTINE is as specified by the IMSL library manuals [29]. The standard subroutine DVERK is being called which returns the new values of the variables, after each time interval.

The DATA statement in this MAIN ROUTINE sets the initial values of the variables. Here we have changed the names of the variables to the array Y( ) since this is a requirement of the DVERK subroutine. The correspondence between the new names and those used in section 8.2 in the interaction network model, is given by equation numbers of table 8.4, i.e. EP is now Y(1), PN is now Y(2) ,.. E is now Y(19).

The SYSTEM OF DIFFERENTIAL EQUATIONS section is the subroutine containing the system of differential equations. This contains a DATA statement setting the reaction constants and a second one setting the initial conditions for those substances that don't change (E ... PE (explained in section 8.2).

The actual differential equations are given as required by DVERK. For example for substance EPPN or Y(3) we have the equation

```
YPRIME(3) = RK(1) * Y(1) * Y(2) - RK(6) * GE * Y(3)
```

The output from running this program is given in Appendix C.

8.7 COMPARISON OF OUTPUT

In section 8.2 we modelled the cell as an interaction network and derived its behaviour by running the model on the computer. We have discussed in 8.4 that it is possible to model

the interaction network as a system of ordinary differential equations and then derive its behaviour by using existing numerical methods. In 8.5 we discussed convergence and stability with regard to these methods. In the previous section (8.6) we modelled the interaction network using the Runge-Kutta method from a subroutine library.

Here we have some observations comparing the two results.

The difference in behaviour at any point is less than 4%, i.e. if our resolution level is coarser than 4% then the two models produce the same behaviour. For a finer than 4% resolution level the following should be borne in mind:

The time slice for the interaction network here was taken as DT = 1 unit. I.e. Euler's method is using a step size of 1 while the R/K subroutine adjusts automatically to smaller step size and therefore to more accurate results (i.e. acceptable under finer resolution level).

The difference between the two results depends on the second derivative of the state of the entity, i.e. the positive or negative acceleration of the state of the entity.

The graphs in fig. 8.11 show this on the entity PI where the biggest difference between the two models occurs. The two graphs show how the difference changes as the entity state changes in time.

The resolution of entity states of the behaviour of the network can be increased by increasing the temporal resolution: Euler's method increases in accuracy by decreasing the step size. The interaction network model was re-run for different DT and the largest difference between the two models is shown

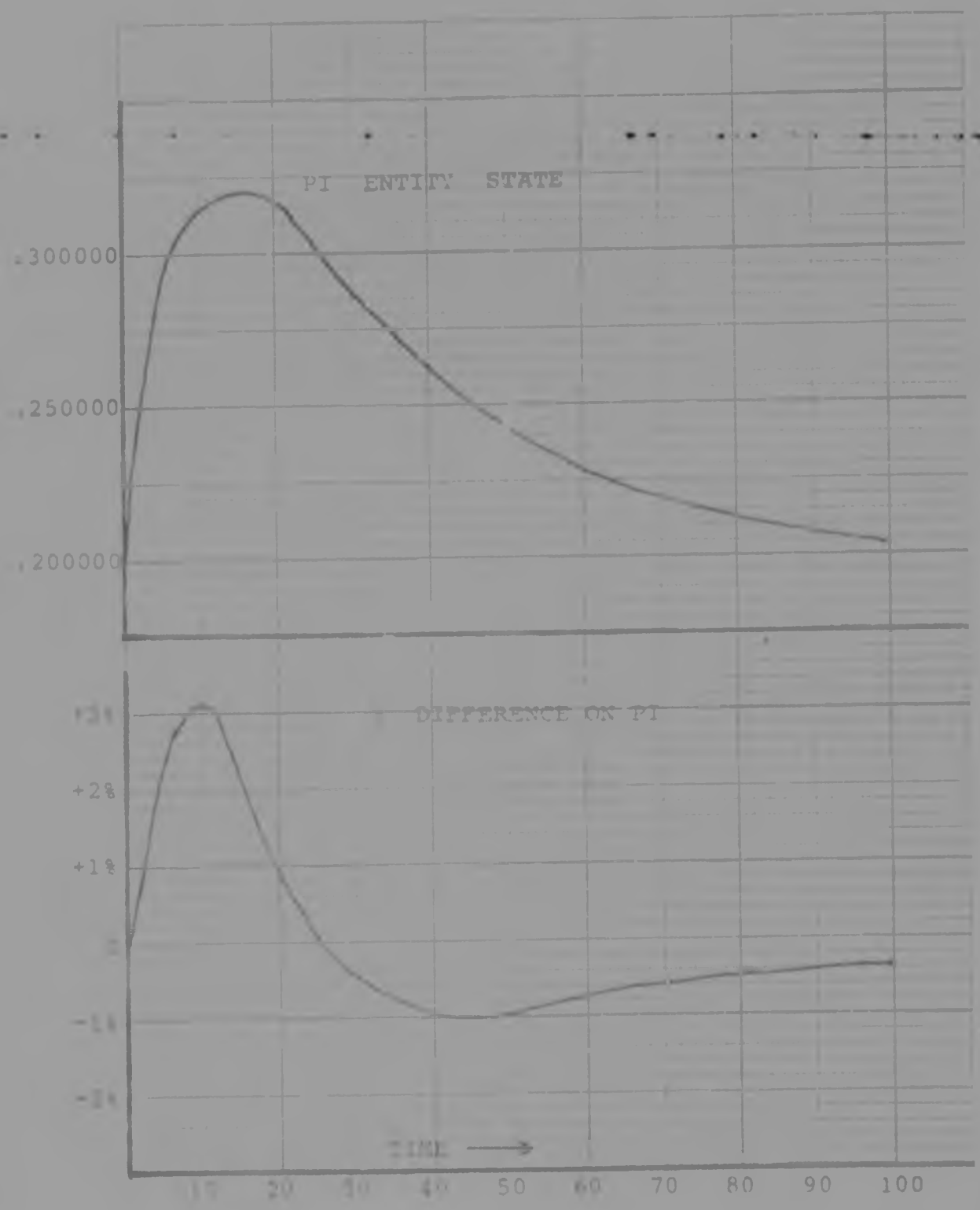


FIG. 8.11. Graph of difference in predicted behaviour between standard program model and Runge-Kutta

against DT in table 8.5 below

TABLE 8.5

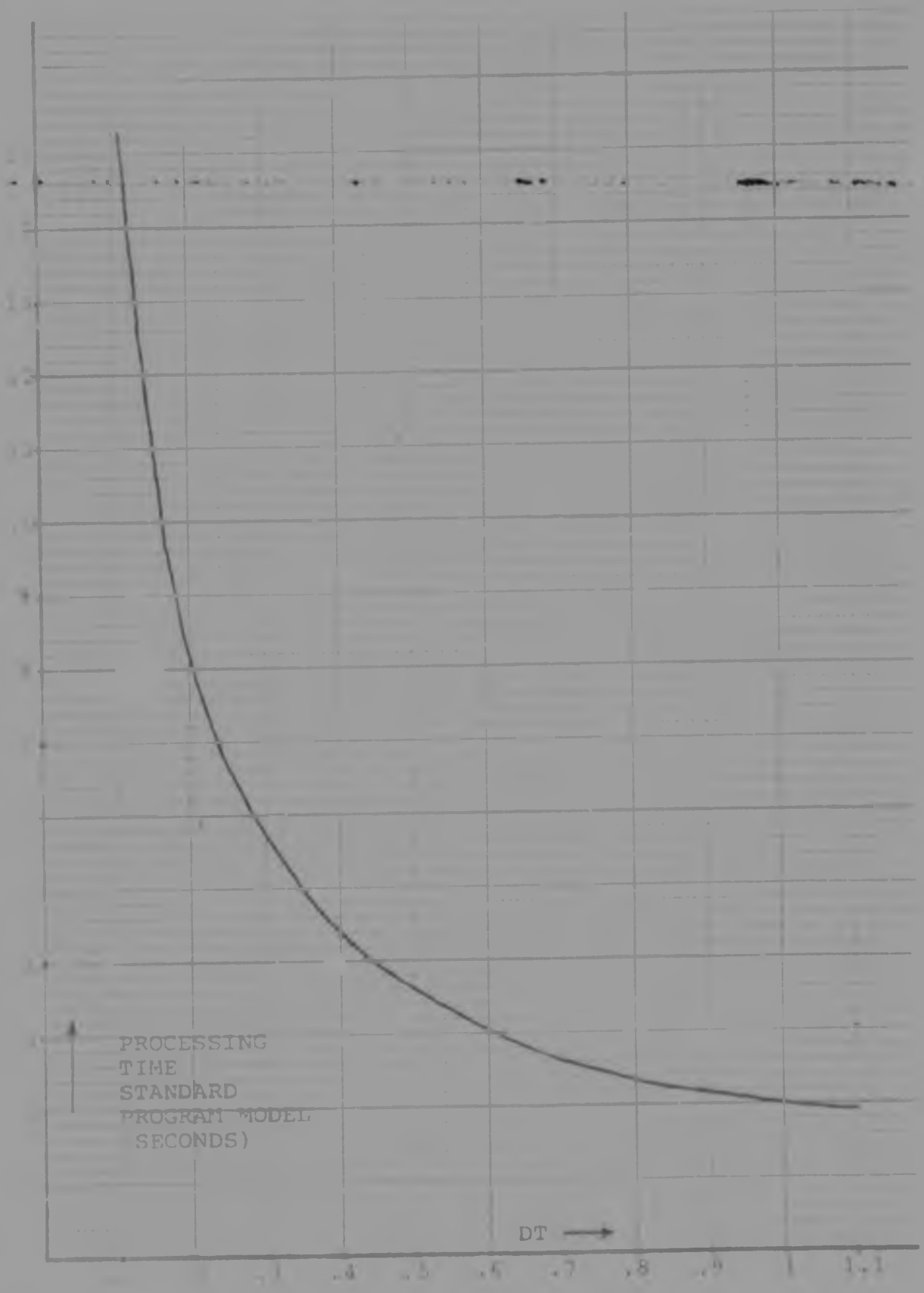
LARGEST % DIFFERENCE BETWEEN MODELS
FOR DIFFERENT DT

DT	MAX % DIFF.
0.2	0,9
0,3	1,2
0,8	2,5
1,0	3,1

This of course cannot be carried to an arbitrary extent due to the limitations of the computer. Below a certain step size the machine round off error will affect the results significantly.

The other limitation is of course economic (in computational time and resource) since this increases as DT decreases. The graph in fig. 8.12 shows the processing time required to run the interaction network model of section 8.2 at different time slice sizes (DT). For comparison, the processing time to run the system of differential equations using DVERK (Runge-Kutta order 5) with TOL = 0,00001 (tolerance of error control) was 6.3 seconds.

As a check for stiffness this system of differential equations was run calling the DGEAR [29] subroutine of IMSL which is for stiff systems. The results were virtually indistinguishable from those of DVERK.



Graph of processing time vs. DT
for the program model

CHAPTER 9

CONCLUSIONS

The objective of this dissertation was:

Firstly to demonstrate that there are basic characteristics common to all organised systems with the need to survive in a changing environment, and that these characteristics are independent of the physical nature of these systems.

Secondly to present a methodology for the representation of these systems and their components, which makes it possible to study these common characteristics in an unambiguous way.

It was first necessary to establish a conceptual framework and a basic language (chapter 2). In order to be able to compare such diverse phenomena, it is necessary to find a language which can describe all these phenomena. Such a language should have as elements representations of the basic elements of all phenomena as they are perceived. The two such fundamental elements were the [illegible] and the [illegible] blocks, the Interaction Network was used to describe a phenomenon. A method of schematic representation of these networks was presented. This would now serve as a diagrammatic model of the phenomenon to the participating [illegible].

Having established the basic tools, this study [illegible] on the two goals.

The first goal was attained in chapters 3, 4 and 5. Using the conceptual framework and language which was established, the main features of the self-maintaining organised system were explored for three physical systems (cells, organisms, societies). Chapter 3 identified some fundamental mechanisms which characterise organised systems with the ability to survive. Chapter 4 explored some of the main components and processes typically found in such systems, and finally in chapter 5, their relationships with their environment were considered.

It is claimed that the above indicated some of the main common characteristics of self-maintaining organised systems.

Second, a study can focus on one of these features (e.g. the circulatory subsystem or the metabolic subsystem and its control or the communication subsystem) and in more explicit detail the way in which these are implemented in the different physical systems. Such studies can be interdisciplinary bridges through which solutions to problems in one field are transferred to another providing a new point of view, a new idea, and possibly a new direction.

The second goal was pursued in chapters 6, 7 and 8. Here dealing with the basic conceptual framework developed in chapter 2, a technique was established for the formal representation of phenomena to any desired resolution level. Modelling of these on the computer provided a (operational representation) which is unambiguously reproducible and testable.

This technique was tested on a phenomenon taken from the economic subsystem of society and a second illustration was given on a physically different system (a cell).

It is suggested that chapters 6, 7 and 8 have demonstrated the applicability of the technique in an unambiguous representation of diverse physical phenomena in such a way that the structure of the phenomenon is clearly represented and can be compared to others.

A problem which was identified for this method of representation was that it is relatively weak in terms of accuracy coupled with computational economy. It was suggested that in cases where this is of importance this method can be supplemented with existing standard numerical methods.

Beyond this point, this technique could be developed by constructing interactive computer software which can aid in the formalisation of interaction networks, model diagrams and computer models, from the definitions of entities and their interactions. Another software could associate representations of a phenomenon at different levels of resolution, in which one entity of a model at a coarser resolution level is associated with the entities and interactions of another model at a finer resolution level. Furthermore the components of the abstract models could be associated to the specific algorithms representing specific physical entities and interactions.

Finally the author believes that, as indicated above, the approach presented in this dissertation provides some guidelines and tools for the study of the class of organised systems.

APPENDIX A

Extract from Heinmets description of the system of bi chemical reactions presented in table 8.1:

G_E represents a group of genes which are required for protein synthesis. Gene G_P is for the synthesis of the enzyme E_P, which is the polymerase for messenger M. Gene G_B produces RNA fraction of the ribosome, while the protein fraction of ribosome is provided by total protein E. Gene G_C produces the transport RNA. The first functional process is the complexing of messenger M with ribosome B yielding the template N. Messenger M is produced by interaction of complex $[E_pP_n]$ with gene G_E. Enzyme synthesis results when templates interact with transport RNA and the amino acid pool complex $[CP_a]$. The total protein E contains three fractions: E_n - enzymes which convert internal pool P_i into RNA precursors, E_a - enzymes which convert internal pool P_i into amino acids, and E_t - enzymes which convert external pool P_e into internal pool P_i. Rate constants (k_n, k_a, and k_t) determine what fraction of the total protein is converted to respective enzymes. [See NOTE below.] Gene G_E is the only gene where polymerase is operating in the messenger synthesis.

Initially it was intended that the same polymerase E_P be operational with all genes. However, competition among various genes for the same polymerase made it difficult to organize a stable functional system on the [analog] computer. It seems that these genes should each have different kinds of polymerases. However, inclusion of four polymerases makes the system extremely complex, and therefore other genes

are considered to interact with pool P_n directly. Nevertheless, participation of a polymerase in forming messenger M for total protein will give system characteristics where a polymerase is part of the gene product. The formation of polymerase E_P follows generally the same line represented by total protein E synthesis. External nutrient pool P_e is converted into a internal pool P_i which in turn is split into an amino acid pool P_a and a nucleotide pool P_n. In addition, part of the pool P_i leaks out from the cell via k_{30}, but the product X will not be associated with the external pool.

A regulatory feature in the system is the complexing of total protein with the internal pool. This complexing is reversible and the ratio between the two rate constants k_{22} and k_{-22} will determine the degree of regulation at that level. Regulatory processes are also produced by the complexing of polymerase with transport RNA (Eq. 21) and the complexing of polymerase with ribosomes (Eq. 23). Furthermore, two additional regulatory steps are introduced into the system. It is visualized that template N can be converted into inactive form N', and template N_P into N_P'. This conversion can result from the action of metabolites or hormones and is considered to be reversible by activation process. All functional entities have different degrees of stability, and they decay int. their respective pools.

For further information see Heinmets [23] & [24].

NOTE

Equations 18, 19, 20 of Table 8.1 were given by Heinmets [24] initially as follows:

18. $P_e + E_t \xrightarrow{k'_{18}} P_i + E_t$

19. $P_i + E_n \xrightarrow{k'_{19}} P_n + E_n$

20. $P_i + E_a \xrightarrow{k'_{20}} P_a + E_a$

together with the following information:

$$E_t = k_t E, \quad E_n = k_n E, \quad E_a = k_a E$$

and $k_{18} = k'_{18} k_t$, $k_{19} = k'_{19} k_t$, $k_{20} = k'_{20} k_a$

The above equations can be substituted by

18. $P_e + E \xrightarrow{k_{18}} P_i + E$

19. $P_i + E \xrightarrow{k_{19}} P_n + E$

20. $P_i + E \xrightarrow{k_{19}} P_a + E$

and these are the equations that heinmets actually implements in his analogue computer (Table 3 in [24] p. 162).

We have thus used these equations in Table 8.1.

18. $P_e + E_t \xrightarrow{k'_{18}} P_i + E_t$

19. $P_i + E_n \xrightarrow{k'_{19}} P_n + E_n$

20. $P_i + E_a \xrightarrow{k'_{20}} P_a + E_a$

together with the following information:

$$E_t = k_t E, \quad E_n = k_n E, \quad E_a = k_a E$$

and $k_{18} = k'_{18} k_t$, $k_{19} = k'_{19} k_t$, $k_{20} = k'_{20} k_a$

The above equations can be substituted by

18. $P_e + E \xrightarrow{k_{18}} P_i + E$

19. $P_i + E \xrightarrow{k_{19}} P_n + E$

20. $P_i + E \xrightarrow{k_{19}} P_a + E$

and these are the equations that Heinmets actually implements in his analogue computer (Table 3 in [24] p. 162).

We have thus used these equations in Table 8.1.

APPENDIX B

OUTPUT FROM THE STANDARD PROGRAM MODEL OF A CELL

TIME	PI	PA	PN	E	N
1	0.218590	0.198752	0.222212	0.217476	0.182669
2	0.236839	0.198035	0.244311	0.231837	0.168699
3	0.253942	0.197987	0.266002	0.245697	0.157271
4	0.269423	0.198666	0.287082	0.253515	0.147763
5	0.283033	0.200072	0.307412	0.261646	0.139787
6	0.294689	0.202170	0.326901	0.268367	0.132951
7	0.304413	0.204903	0.345489	0.273900	0.127025
8	0.312302	0.208291	0.363145	0.278426	0.121823
9	0.318494	0.211990	0.379853	0.282094	0.117203
10	0.323149	0.216200	0.395611	0.285027	0.113059
11	0.326434	0.220781	0.410447	0.287330	0.109307
12	0.328517	0.225608	0.424366	0.289091	0.105883
13	0.329560	0.230685	0.437403	0.290384	0.102739
14	0.329713	0.235940	0.449589	0.291274	0.099835
15	0.329112	0.241328	0.460961	0.291814	0.097141
16	0.327882	0.246809	0.471559	0.292052	0.094631
17	0.326134	0.252351	0.481421	0.292029	0.092286
18	0.323966	0.257924	0.490588	0.291779	0.090088
19	0.321462	0.263504	0.499100	0.291333	0.088024
20	0.318696	0.269071	0.506991	0.290717	0.086082
21	0.315732	0.274607	0.514320	0.289953	0.084252
22	0.312624	0.280099	0.521105	0.289062	0.082526
23	0.309419	0.285536	0.527387	0.288061	0.080895
24	0.306155	0.290908	0.533202	0.286964	0.079353
25	0.302865	0.296207	0.538583	0.285786	0.077893
26	0.299575	0.301429	0.543560	0.284536	0.076510
27	0.296307	0.306569	0.548164	0.283226	0.075199
28	0.293079	0.311624	0.552422	0.281864	0.073955
29	0.289906	0.316591	0.556360	0.280457	0.072774
30	0.286799	0.321469	0.560002	0.279013	0.071652
31	0.283765	0.326257	0.563370	0.277537	0.070584
32	0.280813	0.330955	0.566486	0.276034	0.069568
33	0.277945	0.335564	0.569369	0.274510	0.068600
34	0.275166	0.340084	0.572037	0.272967	0.067677
35	0.272478	0.344516	0.574507	0.271410	0.066795
36	0.269880	0.348860	0.576795	0.269841	0.065954
37	0.267372	0.353120	0.578914	0.268263	0.065149
38	0.264954	0.357295	0.580877	0.266679	0.064378
39	0.262625	0.361387	0.582696	0.265091	0.063639
40	0.260382	0.365399	0.584387	0.263500	0.062931
41	0.258223	0.369332	0.585954	0.261908	0.062250
42	0.256146	0.373188	0.587409	0.260316	0.061595
43	0.254147	0.376968	0.588759	0.258726	0.060965
44	0.252225	0.380674	0.590015	0.257139	0.060357
45	0.250376	0.384308	0.591181	0.255556	0.059771

TIME	PI	PA	PN	E	N
46	0.248597	0.387872	0.592266	0.253977	0.059204
47	0.246886	0.391368	0.593276	0.252403	0.058655
48	0.245239	0.394797	0.594215	0.250935	0.058124
49	0.243654	0.398161	0.595086	0.249273	0.057608
50	0.242128	0.401461	0.595902	0.247719	0.057108
51	0.240658	0.404700	0.596660	0.246171	0.056621
52	0.239241	0.407878	0.597365	0.244631	0.056147
53	0.237875	0.410997	0.598022	0.243099	0.055685
54	0.236558	0.414059	0.598633	0.241574	0.055234
55	0.235286	0.417065	0.599202	0.240058	0.054794
56	0.234059	0.420017	0.599732	0.238551	0.054363
57	0.232873	0.422915	0.600224	0.237052	0.053941
58	0.231727	0.425762	0.600681	0.235561	0.053528
59	0.230619	0.428557	0.601106	0.234080	0.053123
60	0.229546	0.431304	0.601500	0.232607	0.052725
61	0.228507	0.434001	0.601864	0.231143	0.052334
62	0.227501	0.436652	0.602201	0.229688	0.051949
63	0.226526	0.439256	0.602512	0.228242	0.051570
64	0.225581	0.441815	0.602798	0.226805	0.051197
65	0.224663	0.444330	0.603060	0.225377	0.050830
66	0.223772	0.446802	0.603300	0.223957	0.050467
67	0.222906	0.449232	0.603519	0.222547	0.050109
68	0.222064	0.451620	0.603717	0.221145	0.049755
69	0.221246	0.453968	0.603896	0.219752	0.049406
70	0.220450	0.456276	0.604055	0.218368	0.049060
71	0.219675	0.458545	0.604197	0.216992	0.048718
72	0.218920	0.460777	0.604321	0.215626	0.048380
73	0.218184	0.462971	0.604428	0.214267	0.048045
74	0.217467	0.465128	0.604519	0.212918	0.047713
75	0.216767	0.467250	0.604594	0.211576	0.047384
76	0.216084	0.469337	0.604654	0.210243	0.047059
77	0.215418	0.471389	0.604699	0.208919	0.046736
78	0.214767	0.473407	0.604730	0.207603	0.046415
79	0.214131	0.475392	0.604747	0.206295	0.046097
80	0.213510	0.477344	0.604750	0.204995	0.045782
81	0.212903	0.479265	0.604740	0.203704	0.045469
82	0.212308	0.481154	0.604716	0.202420	0.045159
83	0.211727	0.483012	0.604681	0.201144	0.044850
84	0.211158	0.484840	0.604632	0.199877	0.044544
85	0.210601	0.486638	0.604572	0.198617	0.044240
86	0.210056	0.488406	0.604500	0.197365	0.043938
87	0.209522	0.490146	0.604418	0.196121	0.043638
88	0.208998	0.491857	0.604320	0.194884	0.043340
89	0.208485	0.493541	0.604214	0.193656	0.043044
90	0.207982	0.495197	0.604096	0.192434	0.042750
91	0.207489	0.496826	0.603968	0.191221	0.042457
92	0.207005	0.498428	0.603828	0.190014	0.042167
93	0.206530	0.500004	0.603679	0.188815	0.041878
94	0.206064	0.501555	0.603519	0.187624	0.041591
95	0.205607	0.503080	0.603349	0.186440	0.041306
96	0.205159	0.504580	0.603169	0.185263	0.041023
97	0.204718	0.506055	0.602979	0.184093	0.040741
98	0.204285	0.507506	0.602779	0.182930	0.040461
99	0.203860	0.508933	0.602569	0.181775	0.040183
100	0.203442	0.510337	0.602351	0.180626	0.039906

TIME	M	C	MP	EP	B
1	0.199236	0.210918	0.202305	0.213270	0.216232
2	0.195564	0.215568	0.204418	0.223537	0.232488
3	0.193012	0.215860	0.206428	0.231409	0.243695
4	0.183312	0.213165	0.208394	0.237367	0.252541
5	0.175984	0.208467	0.210357	0.241790	0.259543
6	0.168392	0.202475	0.212347	0.244976	0.265087
7	0.160767	0.195695	0.214378	0.247163	0.269464
8	0.153341	0.188489	0.216483	0.248539	0.272899
9	0.146166	0.181112	0.218665	0.249256	0.275560
10	0.139333	0.173749	0.220905	0.249434	0.277580
11	0.132883	0.166513	0.223082	0.249174	0.279061
12	0.126836	0.159499	0.225370	0.248555	0.280084
13	0.121196	0.152759	0.227726	0.247644	0.280713
14	0.115957	0.146326	0.230123	0.246497	0.281001
15	0.111108	0.140217	0.232555	0.245158	0.280989
16	0.106632	0.134440	0.235018	0.243667	0.280713
17	0.102508	0.128993	0.237499	0.242055	0.280231
18	0.098716	0.123870	0.239997	0.240350	0.279474
19	0.095234	0.119060	0.242505	0.238576	0.278567
20	0.092041	0.114551	0.245018	0.236752	0.277484
21	0.089115	0.110327	0.247529	0.234895	0.276247
22	0.086437	0.106373	0.250035	0.233020	0.274870
23	0.083987	0.102674	0.252530	0.231138	0.273366
24	0.081747	0.099215	0.255011	0.229259	0.271747
25	0.079701	0.095979	0.257474	0.227393	0.270023
26	0.077831	0.092952	0.259916	0.225547	0.268205
27	0.076124	0.090120	0.262334	0.223726	0.266301
28	0.074566	0.087469	0.264726	0.221937	0.264321
29	0.073144	0.084986	0.267090	0.220182	0.262272
30	0.071846	0.082660	0.269424	0.218466	0.260161
31	0.070663	0.080478	0.271726	0.216792	0.257995
32	0.069584	0.078430	0.273995	0.215160	0.255781
33	0.068600	0.076506	0.276231	0.213573	0.253524
34	0.067703	0.074698	0.278432	0.212031	0.251231
35	0.066885	0.072995	0.280598	0.210536	0.248906
36	0.066139	0.071392	0.282729	0.209088	0.246555
37	0.065460	0.069879	0.284823	0.207686	0.244182
38	0.064840	0.068451	0.286882	0.206330	0.241791
39	0.064276	0.067100	0.288904	0.205021	0.239386
40	0.063762	0.065822	0.290890	0.203757	0.236971
41	0.063293	0.064611	0.292841	0.202537	0.234550
42	0.062865	0.063461	0.294756	0.201361	0.232125
43	0.062476	0.062369	0.296635	0.200227	0.229700
44	0.062121	0.061330	0.298479	0.199135	0.227278
45	0.061797	0.060341	0.300289	0.198083	0.224860
46	0.061501	0.059397	0.302065	0.197070	0.222450
47	0.061231	0.058496	0.303807	0.196095	0.220050
48	0.060985	0.057634	0.305516	0.195157	0.217661
49	0.060760	0.056809	0.307192	0.194253	0.215285
50	0.060554	0.056019	0.308836	0.193384	0.212925

TIME	M	C	MP	EP	B
51	0.060366	0.055260	0.310448	0.192548	0.210582
52	0.060193	0.054532	0.312029	0.191743	0.208256
53	0.060031	0.053831	0.313580	0.190968	0.205950
54	0.059884	0.053156	0.315100	0.190222	0.203664
55	0.059755	0.052505	0.316591	0.189504	0.201400
56	0.059632	0.051877	0.318053	0.188812	0.199159
57	0.059518	0.051271	0.319487	0.188146	0.196941
58	0.059412	0.050685	0.320892	0.187503	0.194746
59	0.059314	0.050118	0.322271	0.186885	0.192577
60	0.059222	0.049568	0.323622	0.186288	0.190433
61	0.059137	0.049036	0.324947	0.185712	0.188314
62	0.059057	0.048519	0.326246	0.185156	0.186222
63	0.058981	0.048017	0.327520	0.184620	0.184156
64	0.058910	0.047529	0.328768	0.184101	0.182117
65	0.058842	0.047055	0.329992	0.183600	0.180104
66	0.058778	0.046594	0.331193	0.183116	0.178119
67	0.058716	0.046144	0.332369	0.182647	0.176161
68	0.058657	0.045707	0.333522	0.182193	0.174231
69	0.058599	0.045280	0.334652	0.181753	0.172328
70	0.058544	0.044863	0.335760	0.181327	0.170452
71	0.058489	0.044457	0.336846	0.180913	0.168604
72	0.058436	0.044060	0.337910	0.180512	0.166783
73	0.058384	0.043673	0.338953	0.180122	0.164989
74	0.058333	0.043294	0.339975	0.179742	0.163223
75	0.058282	0.042923	0.340976	0.179374	0.161483
76	0.058232	0.042561	0.341957	0.179015	0.159770
77	0.058181	0.042206	0.342918	0.178665	0.158083
78	0.058131	0.041859	0.343859	0.178324	0.156423
79	0.058081	0.041519	0.344780	0.177992	0.154788
80	0.058030	0.041185	0.345683	0.177667	0.153180
81	0.057979	0.040859	0.346567	0.177350	0.151597
82	0.057928	0.040539	0.347432	0.177040	0.150040
83	0.057877	0.040225	0.348279	0.176737	0.148507
84	0.057825	0.039917	0.349108	0.176440	0.146999
85	0.057772	0.039615	0.349920	0.176149	0.145516
86	0.057719	0.039319	0.350713	0.175864	0.144056
87	0.057665	0.039028	0.351490	0.175585	0.142621
88	0.057610	0.038743	0.352250	0.175310	0.141208
89	0.057554	0.038462	0.352992	0.175040	0.139819
90	0.057498	0.038187	0.353718	0.174775	0.138453
91	0.057441	0.037916	0.354428	0.174515	0.137109
92	0.057383	0.037651	0.355122	0.174258	0.135788
93	0.057324	0.037389	0.355800	0.174005	0.134488
94	0.057265	0.037133	0.356462	0.173756	0.133209
95	0.057204	0.036880	0.357109	0.173511	0.131952
96	0.057143	0.036632	0.357740	0.173269	0.130716
97	0.057080	0.036388	0.358356	0.173029	0.129500
98	0.057017	0.036148	0.358958	0.172793	0.128304
99	0.056953	0.035912	0.359544	0.172560	0.127128
100	0.056887	0.035680	0.360116	0.172330	0.125971

TIME					
1	0.078091	0.099094	0.194743	0.200000	0.697516
2	0.078520	0.096134	0.190127	0.199653	0.678169
3	0.079261	0.097147	0.186049	0.199034	0.661201
4	0.080299	0.096119	0.182428	0.198199	0.646078
5	0.081610	0.095155	0.179202	0.197191	0.632416
6	0.083171	0.094175	0.176318	0.196043	0.619936
7	0.084956	0.093219	0.173734	0.194781	0.608432
8	0.086942	0.092291	0.171416	0.193426	0.597747
9	0.089103	0.091397	0.169333	0.191994	0.587762
10	0.091416	0.090539	0.167462	0.190498	0.578384
11	0.093858	0.089719	0.165782	0.188949	0.569538
12	0.096408	0.088939	0.164269	0.187356	0.561161
13	0.099048	0.088199	0.162912	0.185727	0.553206
14	0.101757	0.087499	0.161694	0.184067	0.545628
15	0.104519	0.086839	0.160603	0.182382	0.538392
16	0.107320	0.086219	0.159627	0.180678	0.531467
17	0.110145	0.085636	0.158755	0.178957	0.524827
18	0.112981	0.085091	0.157978	0.177223	0.518448
19	0.115817	0.084582	0.157287	0.175480	0.512309
20	0.118644	0.084107	0.156674	0.173731	0.506392
21	0.121451	0.083666	0.156132	0.171978	0.500682
22	0.124233	0.083258	0.155655	0.170224	0.495163
23	0.126982	0.082883	0.155235	0.168470	0.489823
24	0.129691	0.082531	0.154866	0.166718	0.484649
25	0.132357	0.082211	0.154549	0.164971	0.479633
26	0.134975	0.081918	0.154271	0.163230	0.474764
27	0.137541	0.081651	0.154032	0.161495	0.470033
28	0.140053	0.081408	0.153826	0.159769	0.465434
29	0.142509	0.081189	0.153651	0.158053	0.460958
30	0.144906	0.080992	0.153502	0.156347	0.456600
31	0.147244	0.080817	0.153376	0.154653	0.452354
32	0.149521	0.080662	0.153271	0.152972	0.448214
33	0.151737	0.080527	0.153183	0.151304	0.444176
34	0.153893	0.080410	0.153111	0.149650	0.440235
35	0.155986	0.080312	0.153051	0.148010	0.436387
36	0.158020	0.080230	0.153002	0.146386	0.432627
37	0.159992	0.080164	0.152962	0.144777	0.428954
38	0.161905	0.080114	0.152929	0.143185	0.425363
39	0.163759	0.080079	0.152901	0.141609	0.421851
40	0.165556	0.080058	0.152877	0.140049	0.418415
41	0.167295	0.080051	0.152855	0.138507	0.415053
42	0.168978	0.080057	0.152835	0.136982	0.411763
43	0.170607	0.080076	0.152815	0.135474	0.408541
44	0.172182	0.080107	0.152794	0.133984	0.405387
45	0.173706	0.080149	0.152772	0.132511	0.402297
46	0.175178	0.080202	0.152747	0.131057	0.399270
47	0.176601	0.080267	0.152719	0.129620	0.396304
48	0.177976	0.080341	0.152688	0.128200	0.393398
49	0.179304	0.080426	0.152652	0.126799	0.390549
50	0.180586	0.080520	0.152611	0.125417	0.387757

TIME	EPPN	B1	NP	N1	CPA
51	0.181824	0.0[illegible]	0.152564	0.124049	0.365019
52	0.183019	0.080736	0.152513	0.122700	0.362334
53	0.184173	0.080857	0.152455	0.121369	0.359702
54	0.185286	0.080986	0.152391	0.120055	0.357120
55	0.186360	0.081123	0.152320	0.118759	0.354588
56	0.187396	0.081268	0.152243	0.117480	0.352104
57	0.188395	0.081421	0.152159	0.116217	0.349667
58	0.189358	0.081581	0.152069	0.114972	0.347276
59	0.190287	0.081748	0.151971	0.113743	0.344930
60	0.191183	0.081922	0.151866	0.112531	0.342629
61	0.192046	0.082103	0.151755	0.111334	0.340370
62	0.192878	0.082290	0.151636	0.110154	0.338154
63	0.193679	0.082485	0.151510	0.108990	0.335977
64	0.194451	0.082682	0.151378	0.107842	0.333840
65	0.195194	0.082888	0.151238	0.106709	0.331742
66	0.195909	0.083099	0.151092	0.105592	0.329681
67	0.196598	0.083315	0.150938	0.104490	0.327657
68	0.197261	0.083537	0.150778	0.103401	0.325670
69	0.197899	0.083765	0.150612	0.102328	0.323719
70	0.198512	0.083997	0.150439	0.101270	0.321804
71	0.199101	0.084235	0.150259	0.100226	0.319924
72	0.199668	0.084477	0.150074	0.099196	0.318079
73	0.200212	0.084724	0.149882	0.098179	0.316268
74	0.200734	0.084976	0.149684	0.097177	0.314491
75	0.201236	0.085232	0.149481	0.096187	0.312747
76	0.201717	0.085493	0.149272	0.095211	0.311035
77	0.202178	0.085758	0.149057	0.094248	0.309354
78	0.202620	0.086027	0.148837	0.093298	0.307705
79	0.203044	0.086301	0.148611	0.092360	0.306086
80	0.203449	0.086578	0.148381	0.091435	0.304497
81	0.203837	0.086859	0.148145	0.090522	0.302937
82	0.204207	0.087144	0.147905	0.089621	0.301406
83	0.204561	0.087433	0.147659	0.088732	0.299903
84	0.204898	0.087726	0.147410	0.087854	0.298428
85	0.205220	0.088022	0.147156	0.086988	0.296980
86	0.205526	0.088321	0.146897	0.086133	0.295559
87	0.205817	0.088624	0.146635	0.085289	0.294164
88	0.206094	0.088931	0.146369	0.084456	0.292795
89	0.206356	0.089240	0.146098	0.083634	0.291451
90	0.206604	0.089553	0.145824	0.082822	0.290132
91	0.206839	0.089869	0.145547	0.082020	0.288837
92	0.207061	0.090188	0.145266	0.081229	0.287566
93	0.207269	0.090511	0.144981	0.080448	0.286318
94	0.207465	0.090836	0.144694	0.079677	0.285093
95	0.207649	0.091164	0.144403	0.078915	0.283890
96	0.207821	0.091494	0.144109	0.078163	0.282709
97	0.207981	0.091828	0.143813	0.077420	0.281549
98	0.208129	0.092165	0.143514	0.076686	0.280411
99	0.208267	0.092504	0.143212	0.075962	0.279293
100	0.208393	0.092845	0.142907	0.075246	0.278195

TIME	EPC	EPB	EPI	NP1
1	0.039866	0.007450	0.050125	0.200000
2	0.032781	0.007047	0.051533	0.199895
3	0.027266	0.006759	0.053215	0.199700
4	0.022953	0.006561	0.055194	0.199426
5	0.019560	0.006430	0.057326	0.199067
6	0.016872	0.006349	0.059702	0.198689
7	0.014723	0.006304	0.062253	0.198241
8	0.012969	0.006284	0.064949	0.197751
9	0.011575	0.006281	0.067760	0.197225
10	0.010410	0.006289	0.070658	0.196667
11	0.009438	0.006302	0.073620	0.196083
12	0.008618	0.006317	0.076620	0.195477
13	0.007918	0.006331	0.079641	0.194852
14	0.007314	0.006343	0.082682	0.194214
15	0.006787	0.006349	0.085671	0.193563
16	0.006324	0.006351	0.088653	0.192904
17	0.005912	0.006346	0.091598	0.192238
18	0.005544	0.006336	0.094496	0.191569
19	0.005212	0.006318	0.097342	0.190897
20	0.004912	0.006295	0.100128	0.190225
21	0.004639	0.006266	0.102851	0.189554
22	0.004389	0.006231	0.105508	0.188885
23	0.004161	0.006190	0.108094	0.188221
24	0.003950	0.006145	0.110610	0.187561
25	0.003757	0.006095	0.113056	0.186907
26	0.003578	0.006040	0.115427	0.186260
27	0.003413	0.005982	0.117727	0.185620
28	0.003260	0.005921	0.119955	0.184988
29	0.003118	0.005856	0.122113	0.184365
30	0.002986	0.005789	0.124201	0.183751
31	0.002864	0.005720	0.126221	0.183146
32	0.002750	0.005649	0.128174	0.182551
33	0.002644	0.005576	0.130061	0.181965
34	0.002545	0.005503	0.131885	0.181389
35	0.002453	0.005428	0.133646	0.180824
36	0.002368	0.005353	0.135347	0.180268
37	0.002288	0.005277	0.136989	0.179723
38	0.002213	0.005201	0.138573	0.179188
39	0.002143	0.005125	0.140102	0.178663
40	0.002077	0.005049	0.141578	0.178147
41	0.002016	0.004974	0.143001	0.177642
42	0.001958	0.004899	0.144373	0.177146
43	0.001904	0.004824	0.145697	0.176660
44	0.001853	0.004750	0.146972	0.176183
45	0.001805	0.004677	0.148202	0.175715
46	0.001760	0.004605	0.149387	0.175256
47	0.001717	0.004534	0.150528	0.174806
48	0.001677	0.004464	0.151628	0.174364
49	0.001639	0.004394	0.152686	0.173931
50	0.001603	0.004326	0.153705	0.173505

TIME	EPC	EPB	EPI	NPI
51	0.001569	0.004 59	0.154686	0.173087
52	0.001537	0.004193	0.155629	0.172677
53	0.001506	0.004129	0.156536	0.172274
54	0.001477	0.004065	0.157408	0.171877
55	0.001449	0.004003	0.158246	0.171488
56	0.001423	0.003942	0.159051	0.171104
57	0.001398	0.003882	0.159823	0.170727
58	0.001374	0.003823	0.160564	0.170356
59	0.001351	0.003765	0.161274	0.169990
60	0.001329	0.003709	0.161955	0.169630
61	0.001307	0.003653	0.162607	0.169274
62	0.001287	0.003599	0.163231	0.168924
63	0.001268	0.003546	0.163827	0.168578
64	0.001249	0.003494	0.164396	0.168237
65	0.001231	0.003443	0.164940	0.167900
66	0.001214	0.003393	0.165458	0.167566
67	0.001197	0.003344	0.165952	0.167237
68	0.001181	0.003297	0.166421	0.166911
69	0.001166	0.003250	0.166867	0.166588
70	0.001151	0.003204	0.167290	0.166269
71	0.001136	0.003159	0.167690	0.165952
72	0.001122	0.003116	0.168069	0.165638
73	0.001109	0.003073	0.168426	0.165327
74	0.001096	0.003031	0.168762	0.165018
75	0.001083	0.002990	0.169078	0.164711
76	0.001070	0.002949	0.169374	0.164407
77	0.001058	0.002910	0.169651	0.164104
78	0.001047	0.002871	0.169909	0.163803
79	0.001035	0.002833	0.170148	0.163504
80	0.001024	0.002796	0.170369	0.163206
81	0.001014	0.002760	0.170573	0.162909
82	0.001003	0.002725	0.170759	0.162614
83	0.000993	0.002690	0.170928	0.162320
84	0.000983	0.002656	0.171080	0.162027
85	0.000973	0.002622	0.171217	0.161734
86	0.000964	0.002590	0.171337	0.161443
87	0.000954	0.002558	0.171442	0.161152
88	0.000945	0.002526	0.171532	0.160862
89	0.000936	0.002496	0.171607	0.160572
90	0.000928	0.002465	0.171667	0.160282
91	0.000919	0.002436	0.171713	0.159993
92	0.000911	0.002407	0.171746	0.159704
93	0.000903	0.002379	0.171764	0.159415
94	0.000895	0.002351	0.171770	0.159127
95	0.000887	0.002323	0.171762	0.158838
96	0.000880	0.002297	0.171742	0.158549
97	0.000872	0.002270	0.171709	0.158261
98	0.000865	0.002245	0.171664	0.157972
99	0.000858	0.002219	0.171607	0.157682
100	0.000851	0.002195	0.171538	0.157393

APPENDIX C

OUTPUT FROM MODEL OF A CELL USING RUNGE-KUTTA METHOD

TIME	PI	PA	PN	E	X
1	0.218331	0.199022	0.222101	0.215997	0.164246
2	0.235727	0.198653	0.243825	0.229394	0.171178
3	0.251744	0.198947	0.264993	0.240639	0.160216
4	0.266126	0.199915	0.285477	0.250084	0.150919
5	0.278758	0.201531	0.305186	0.258009	0.142945
6	0.289624	0.203754	0.324063	0.264642	0.136031
7	0.298772	0.206528	0.342070	0.270173	0.129975
8	0.306297	0.209793	0.359188	0.274758	0.124617
9	0.312320	0.213467	0.375412	0.278527	0.119835
10	0.316976	0.217549	0.390749	0.281591	0.115531
11	0.320405	0.221921	0.405212	0.284045	0.111628
12	0.322747	0.226552	0.418825	0.285970	0.108066
13	0.324135	0.231391	0.431612	0.287434	0.104795
14	0.324696	0.236397	0.443604	0.288498	0.101777
15	0.324546	0.241530	0.454833	0.289213	0.098981
16	0.323793	0.246756	0.465335	0.289623	0.096378
17	0.322531	0.252047	0.475146	0.289768	0.093950
18	0.320847	0.257377	0.484300	0.289680	0.091677
19	0.318817	0.262723	0.492835	0.289390	0.089544
20	0.316508	0.268067	0.500785	0.288924	0.087539
21	0.313978	0.273393	0.508186	0.288302	0.085651
22	0.311279	0.278689	0.515072	0.287546	0.083870
23	0.308453	0.283941	0.521475	0.286673	0.082187
24	0.305539	0.289143	0.527426	0.285696	0.080596
25	0.302568	0.294285	0.532956	0.284631	0.079090
26	0.299568	0.299362	0.538092	0.283487	0.077663
27	0.296562	0.304370	0.542862	0.282277	0.076309
28	0.293568	0.309303	0.547292	0.281007	0.075023
29	0.290602	0.314160	0.551405	0.279688	0.073802
30	0.287677	0.318938	0.555223	0.278325	0.072640
31	0.284802	0.323637	0.558768	0.276925	0.071534
32	0.281987	0.328254	0.562060	0.275493	0.070480
33	0.279238	0.332790	0.565117	0.274034	0.069476
34	0.276558	0.337244	0.567956	0.272552	0.068517
35	0.273952	0.341617	0.570593	0.271051	0.067602
36	0.271422	0.345910	0.573043	0.269535	0.066726
37	0.268969	0.350123	0.575320	0.268006	0.065889
38	0.266593	0.354258	0.577436	0.266468	0.065085
39	0.264295	0.358315	0.579404	0.264921	0.064317
40	0.262074	0.362295	0.581234	0.263369	0.063576
41	0.259928	0.366200	0.582936	0.261813	0.062868
42	0.257857	0.370032	0.584521	0.260255	0.062186
43	0.255858	0.373792	0.585995	0.258696	0.061528
44	0.253930	0.377481	0.587368	0.257137	0.060894
45	0.252070	0.381100	0.588647	0.255580	0.060283
46	0.250276	0.384652	0.589837	0.254026	0.059692
47	0.248546	0.388138	0.590947	0.252474	0.059120
48	0.246878	0.391558	0.591981	0.250927	0.058567
49	0.245269	0.394916	0.592945	0.249385	0.058030
50	0.243716	0.398211	0.593843	0.247848	0.057509

TIME	PI	PA	PN	E	N
51	0.242219	0.401447	0.594680	0.248316	0.057003
52	0.240773	0.404623	0.595461	0.244791	0.056510
53	0.239378	0.407741	0.596188	0.243273	0.056031
54	0.238030	0.410803	0.596865	0.241761	0.055563
55	0.236728	0.413810	0.597496	0.240257	0.055107
56	0.235470	0.416763	0.598084	0.238760	0.054661
57	0.234254	0.419664	0.598630	0.237271	0.054225
58	0.233077	0.422513	0.599138	0.235789	0.053798
59	0.231939	0.425312	0.599610	0.234316	0.053380
60	0.230837	0.428062	0.600048	0.232852	0.052970
61	0.229769	0.430765	0.600454	0.231394	0.052568
62	0.228734	0.433420	0.600830	0.229945	0.052173
63	0.227731	0.436029	0.601176	0.228505	0.051784
64	0.226758	0.438594	0.601496	0.227073	0.051401
65	0.225814	0.441114	0.601789	0.225650	0.051025
66	0.224897	0.443592	0.602058	0.224235	0.050654
67	0.224007	0.446027	0.602304	0.222828	0.050288
68	0.223141	0.448422	0.602527	0.221430	0.049927
69	0.222299	0.450776	0.602729	0.220040	0.049571
70	0.221481	0.453090	0.602910	0.218659	0.049219
71	0.220684	0.455366	0.603072	0.217286	0.048872
72	0.219908	0.457603	0.603215	0.215922	0.048528
73	0.219152	0.459804	0.603339	0.214566	0.048188
74	0.218415	0.461968	0.603447	0.213218	0.047851
75	0.217697	0.464096	0.603537	0.211879	0.047518
76	0.216996	0.466190	0.603611	0.210547	0.047189
77	0.216313	0.468248	0.603669	0.209224	0.046862
78	0.215646	0.470273	0.603712	0.207910	0.046538
79	0.214994	0.472265	0.603740	0.206603	0.046217
80	0.214357	0.474224	0.603754	0.205304	0.045899
81	0.213735	0.476151	0.603754	0.204013	0.045584
82	0.213127	0.478046	0.603740	0.202731	0.045271
83	0.212532	0.479911	0.603713	0.201456	0.044960
84	0.211950	0.481745	0.603673	0.200189	0.044652
85	0.211381	0.483550	0.603620	0.198930	0.044346
86	0.210823	0.485325	0.603555	0.197678	0.044043
87	0.210277	0.487071	0.603478	0.196435	0.043741
88	0.209743	0.488789	0.603389	0.195198	0.043442
89	0.209219	0.490479	0.603289	0.193970	0.043145
90	0.208705	0.492141	0.603177	0.192749	0.042850
91	0.208202	0.493776	0.603054	0.191535	0.042557
92	0.207709	0.495385	0.602920	0.190329	0.042265
93	0.207225	0.496967	0.602775	0.189131	0.041976
94	0.206750	0.498524	0.602620	0.187939	0.041689
95	0.206285	0.500055	0.602454	0.186755	0.041403
96	0.205828	0.501561	0.602279	0.185578	0.041119
97	0.205379	0.503043	0.602093	0.184409	0.040838
98	0.204938	0.504500	0.601897	0.183246	0.040557
99	0.204506	0.505933	0.601692	0.182091	0.040279
100	0.204081	0.507343	0.601477	0.180942	0.040002

TIME	M	C	MP	EP	E
1	0.197916	0.206058	0.202221	0.211671	0.216365
2	0.193774	0.211455	0.204324	0.221285	0.229526
3	0.188261	0.211476	0.206362	0.228683	0.240142
4	0.181892	0.209061	0.208378	0.234405	0.248721
5	0.175040	0.204990	0.210398	0.238834	0.255657
6	0.167978	0.199742	0.212442	0.242058	0.261206
7	0.160902	0.193740	0.214523	0.244383	0.265 59
8	0.153949	0.187283	0.216646	0.245939	0.269355
9	0.147216	0.180593	0.218815	0.246858	0.272193
10	0.140764	0.173835	0.221031	0.247249	0.274392
11	0.134633	0.167126	0.223292	0.247203	0.276044
12	0.128845	0.160554	0.225595	0.246795	0.277237
13	0.123410	0.154177	0.227936	0.246089	0.278023
14	0.118326	0.148037	0.230309	0.245138	0.278457
15	0.113587	0.142158	0.232711	0.243987	0.278581
16	0.109183	0.136555	0.235134	0.242673	0.278433
17	0.105098	0.131236	0.237575	0.241230	0.278041
18	0.101318	0.126200	0.240027	0.239684	0.277433
19	0.097824	0.121443	0.242487	0.238059	0.276629
20	0.094601	0.116948	0.244948	0.236375	0.275651
21	0.091629	0.112730	0.247407	0.234649	0.274514
22	0.088892	0.108766	0.249860	0.232895	0.273234
23	0.086374	0.105035	0.252302	0.231127	0.271825
24	0.084059	0.101530	0.254730	0.229354	0.270299
25	0.081931	0.098241	0.257142	0.227586	0.268666
26	0.079977	0.095153	0.259533	0.225830	0.266938
27	0.078183	0.092254	0.261903	0.224092	0.265123
28	0.076536	0.089533	0.264248	0.222379	0.263229
29	0.075025	0.086977	0.266567	0.220693	0.261266
30	0.073639	0.084577	0.268858	0.219040	0.259240
31	0.072369	0.082321	0.271120	0.217422	0.257158
32	0.071204	0.080199	0.273351	0.215841	0.255026
33	0.070136	0.078203	0.275551	0.214299	0.252850
34	0.069157	0.076322	0.277718	0.212798	0.250636
35	0.068260	0.074550	0.279852	0.211338	0.248389
36	0.067438	0.072879	0.281953	0.209919	0.246114
37	0.066685	0.071300	0.284020	0.208543	0.243816
38	0.065995	0.069809	0.286053	0.207210	0.241497
39	0.065363	0.068397	0.288052	0.205918	0.239164
40	0.064783	0.067061	0.290017	0.204668	0.236818
41	0.064252	0.065794	0.291946	0.203460	0.234464
42	0.063766	0.064591	0.293844	0.202292	0.232104
43	0.063320	0.063448	0.295707	0.201163	0.229743
44	0.062911	0.062361	0.297537	0.200074	0.227381
45	0.062536	0.061326	0.299333	0.199022	0.225023
46	0.062192	0.060339	0.301097	0.198007	0.222669
47	0.061877	0.059397	0.302828	0.197029	0.220323
48	0.061587	0.058496	0.304527	0.196085	0.217987
49	0.061321	0.057635	0.306194	0.195175	0.215666
50	0.061076	0.056809	0.307830	0.194297	0.213350

TIME	M	C	MP	EP	B
51	0.060651	0.050018	0.309435	0.193451	0.211053
52	0.060644	0.050258	0.311010	0.192635	0.208771
53	0.060453	0.050452	0.312556	0.191848	0.206507
54	0.060277	0.050625	0.314072	0.191090	0.204262
55	0.060114	0.050779	0.315559	0.190358	0.202036
56	0.059963	0.050917	0.317017	0.189652	0.199830
57	0.059824	0.051040	0.318448	0.188972	0.197646
58	0.059694	0.051150	0.319851	0.188315	0.195484
59	0.059573	0.051248	0.321228	0.187681	0.193345
60	0.059461	0.051335	0.322578	0.187068	0.191229
61	0.059356	0.049552	0.323902	0.186477	0.189138
62	0.059257	0.049018	0.325200	0.185906	0.187071
63	0.059164	0.048500	0.326473	0.185353	0.185028
64	0.059077	0.047998	0.327721	0.184819	0.183011
65	0.058994	0.047509	0.328944	0.184303	0.181020
66	0.058915	0.047034	0.330146	0.183803	0.179054
67	0.058840	0.046572	0.331323	0.183318	0.177114
68	0.058768	0.046122	0.332476	0.182849	0.175200
69	0.058700	0.045684	0.333608	0.182394	0.173312
70	0.058633	0.045257	0.334716	0.181953	0.171450
71	0.058569	0.044841	0.335803	0.181525	0.169614
72	0.058507	0.044434	0.336868	0.181109	0.167804
73	0.058446	0.044038	0.337912	0.180705	0.166021
74	0.058387	0.043650	0.338935	0.180312	0.164263
75	0.058329	0.043271	0.339938	0.179930	0.162532
76	0.058271	0.042901	0.340920	0.179558	0.160826
77	0.058215	0.042539	0.341883	0.179196	0.159146
78	0.058159	0.042185	0.342825	0.178842	0.157491
79	0.058103	0.041838	0.343749	0.178498	0.155861
80	0.058048	0.041498	0.344653	0.178162	0.154256
81	0.057992	0.041166	0.345539	0.177833	0.152677
82	0.057937	0.040840	0.346406	0.177512	0.151121
83	0.057882	0.040521	0.347255	0.177198	0.149590
84	0.057826	0.040208	0.348086	0.176891	0.148083
85	0.057770	0.039901	0.348899	0.176590	0.146600
86	0.057714	0.039600	0.349695	0.176295	0.145141
87	0.057657	0.039304	0.350473	0.176006	0.143704
88	0.057600	0.039014	0.351235	0.175722	0.142291
89	0.057543	0.038730	0.351980	0.175444	0.140900
90	0.057484	0.038450	0.352708	0.175170	0.139531
91	0.057425	0.038176	0.353420	0.174901	0.138185
92	0.057366	0.037906	0.354116	0.174637	0.136860
93	0.057306	0.037641	0.354796	0.174376	0.135556
94	0.057245	0.037381	0.355461	0.174120	0.134274
95	0.057183	0.037125	0.356110	0.173867	0.133012
96	0.057121	0.036874	0.356743	0.173618	0.131771
97	0.057058	0.036627	0.357362	0.173373	0.130550
98	0.056994	0.036384	0.357966	0.173130	0.129349
99	0.056929	0.036144	0.358555	0.172891	0.128167
100	0.056863	0.035909	0.359129	0.172655	0.127005

TIME	EPPN	B1	MP	N1	CPA
1	0.078252	0.099071	0.195046	0.199839	0.696992
2	0.078809	0.098109	0.190648	0.199396	0.680505
3	0.079656	0.097132	0.186721	0.198725	0.664020
4	0.080774	0.096154	0.183207	0.197883	0.649152
5	0.082141	0.095185	0.180054	0.196857	0.635612
6	0.083734	0.094234	0.177218	0.195719	0.623177
7	0.085532	0.093307	0.174665	0.194476	0.611675
8	0.087511	0.092409	0.172364	0.193145	0.600972
9	0.089649	0.091543	0.170287	0.191739	0.590960
10	0.091925	0.090711	0.168413	0.190272	0.581552
11	0.094319	0.089916	0.166721	0.188753	0.572676
12	0.096811	0.089159	0.165195	0.187190	0.564273
13	0.099383	0.088439	0.163818	0.185590	0.556292
14	0.102020	0.087757	0.162577	0.183960	0.548693
15	0.104705	0.087112	0.161459	0.182304	0.541438
16	0.107425	0.086505	0.160453	0.180628	0.534496
17	0.110168	0.085933	0.159550	0.178936	0.527840
18	0.112921	0.085397	0.158740	0.177230	0.521445
19	0.115674	0.084895	0.158015	0.175514	0.515293
20	0.118420	0.084426	0.157366	0.173792	0.509362
21	0.121146	0.083990	0.156788	0.172065	0.503638
22	0.123853	0.083584	0.156273	0.170336	0.498106
23	0.126526	0.083207	0.155816	0.168607	0.492753
24	0.129167	0.082860	0.155412	0.166880	0.487566
25	0.131766	0.082539	0.155054	0.165156	0.482536
26	0.134321	0.082245	0.154739	0.163438	0.477653
27	0.136829	0.081978	0.154463	0.161726	0.472908
28	0.139287	0.081736	0.154221	0.160021	0.468294
29	0.141692	0.081520	0.154009	0.158326	0.463805
30	0.144044	0.081307	0.153825	0.156641	0.459432
31	0.146340	0.081128	0.153665	0.154966	0.455171
32	0.148580	0.080969	0.153526	0.153304	0.451017
33	0.150762	0.080829	0.153406	0.151654	0.446963
34	0.152887	0.080707	0.153302	0.150017	0.443007
35	0.154955	0.080604	0.153212	0.148394	0.439144
36	0.156966	0.080517	0.153134	0.146785	0.435370
37	0.158919	0.080446	0.153065	0.145192	0.431681
38	0.160816	0.080390	0.153005	0.143613	0.428075
39	0.162657	0.080350	0.152957	0.142051	0.424548
40	0.164443	0.080324	0.152903	0.140504	0.421097
41	0.166174	0.080311	0.152859	0.138974	0.417720
42	0.167853	0.080317	0.152817	0.137460	0.414415
43	0.169478	0.080325	0.152776	0.135963	0.411179
44	0.171053	0.080350	0.152736	0.134483	0.408009
45	0.172577	0.080387	0.152695	0.133019	0.404905
46	0.174052	0.080435	0.152653	0.131573	0.401863
47	0.175479	0.080494	0.152609	0.130144	0.398883
48	0.176859	0.080563	0.152562	0.128732	0.395962
49	0.178194	0.080643	0.152513	0.127338	0.393099
50	0.179484	0.080732	0.152459	0.125960	0.390292

TIME	EPPN	B1	NP	N1	CPA
51	0.180732	0.080830	0.152401	0.124599	0.387540
52	0.181937	0.080937	0.152338	0.123256	0.384841
53	0.183101	0.081053	0.152271	0.121930	0.382194
54	0.184226	0.081178	0.152198	0.120620	0.379598
55	0.185312	0.081310	0.152119	0.119327	0.377052
56	0.186361	0.081451	0.152035	0.118051	0.374554
57	0.187373	0.081599	0.151945	0.116792	0.372103
58	0.188351	0.081754	0.151848	0.115548	0.369699
59	0.189294	0.081916	0.151746	0.114322	0.367339
60	0.190203	0.082086	0.151637	0.113111	0.365024
61	0.191081	0.082262	0.151521	0.111916	0.362752
62	0.191927	0.082445	0.151400	0.110737	0.360522
63	0.192743	0.082634	0.151272	0.109573	0.358333
64	0.193529	0.082829	0.151137	0.108425	0.356185
65	0.194287	0.083030	0.150996	0.107292	0.354076
66	0.195017	0.083236	0.150848	0.106174	0.352006
67	0.195721	0.083449	0.150695	0.105071	0.349974
68	0.196398	0.083667	0.150535	0.103983	0.347979
69	0.197050	0.083890	0.150368	0.102909	0.346021
70	0.197677	0.084118	0.150196	0.101849	0.344098
71	0.198281	0.084352	0.150018	0.100804	0.342211
72	0.198861	0.084590	0.149835	0.099772	0.340358
73	0.199419	0.084834	0.149643	0.098754	0.338538
74	0.199955	0.085081	0.149447	0.097749	0.336752
75	0.200471	0.085334	0.149246	0.096758	0.334998
76	0.200964	0.085591	0.149039	0.095780	0.333276
77	0.201438	0.085852	0.148826	0.094814	0.331585
78	0.201893	0.086118	0.148609	0.093862	0.329924
79	0.202329	0.086387	0.148386	0.092921	0.328294
80	0.202747	0.086661	0.148158	0.091993	0.326693
81	0.203148	0.086938	0.147926	0.091077	0.325121
82	0.203532	0.087220	0.147688	0.090174	0.323577
83	0.203899	0.087505	0.147447	0.089281	0.322062
84	0.204242	0.087794	0.147200	0.088401	0.320573
85	0.204575	0.088087	0.146950	0.087531	0.319112
86	0.204892	0.088383	0.146695	0.086673	0.317677
87	0.205193	0.088682	0.146436	0.085826	0.316268
88	0.205480	0.088985	0.146173	0.084990	0.314884
89	0.205752	0.089291	0.145906	0.084164	0.313525
90	0.206011	0.089601	0.145636	0.083349	0.312191
91	0.206255	0.089913	0.145362	0.082544	0.310881
92	0.206486	0.090229	0.145084	0.081749	0.309595
93	0.206704	0.090548	0.144804	0.080965	0.308331
94	0.206909	0.090870	0.144520	0.080190	0.307091
95	0.207101	0.091194	0.144233	0.079425	0.305874
96	0.207282	0.091522	0.143943	0.078669	0.304678
97	0.207450	0.091852	0.143650	0.077923	0.303504
98	0.207607	0.092185	0.143354	0.077185	0.302351
99	0.207752	0.092521	0.143055	0.076457	0.301220
100	0.207887	0.092860	0.142754	0.075738	0.300109

TIME	EPC	EPB	EPI	NPI
1	0.040614	0.007517	0.050259	0.199950
2	0.034257	0.007152	0.051780	0.199830
3	0.028989	0.006881	0.053549	0.199587
4	0.024743	0.006683	0.055544	0.199297
5	0.021304	0.006543	0.057741	0.198946
6	0.018505	0.006448	0.060114	0.198544
7	0.016214	0.006385	0.062637	0.198095
8	0.014326	0.006346	0.065283	0.197609
9	0.012760	0.006324	0.068027	0.197087
10	0.011450	0.006313	0.070846	0.196538
11	0.010346	0.006310	0.073718	0.195964
12	0.009408	0.006310	0.076623	0.195369
13	0.008605	0.006311	0.079544	0.194757
14	0.007912	0.006311	0.082467	0.194133
15	0.007308	0.006309	0.085377	0.193497
16	0.006779	0.006302	0.088263	0.192852
17	0.006312	0.006292	0.091116	0.192201
18	0.005897	0.006277	0.093928	0.191547
19	0.005525	0.006258	0.096692	0.190890
20	0.005192	0.006232	0.099403	0.190232
21	0.004890	0.006202	0.102056	0.189575
22	0.004616	0.006167	0.104648	0.188921
23	0.004367	0.006127	0.107177	0.188270
24	0.004139	0.006084	0.109641	0.187623
25	0.003930	0.006036	0.112038	0.186982
26	0.003738	0.005984	0.114369	0.186346
27	0.003561	0.005929	0.116632	0.185718
28	0.003398	0.005871	0.118829	0.185096
29	0.003247	0.005810	0.120959	0.184483
30	0.003107	0.005747	0.123024	0.183878
31	0.002977	0.005681	0.125024	0.183281
32	0.002857	0.005614	0.126960	0.182693
33	0.002745	0.005545	0.128834	0.182114
34	0.002640	0.005476	0.130646	0.181545
35	0.002543	0.005405	0.132399	0.180985
36	0.002453	0.005333	0.134093	0.180434
37	0.002368	0.005261	0.135731	0.179893
38	0.002289	0.005188	0.137313	0.179361
39	0.002215	0.005116	0.138842	0.178838
40	0.002146	0.005043	0.140318	0.178325
41	0.002081	0.004971	0.141742	0.177821
42	0.002020	0.004898	0.143118	0.177327
43	0.001962	0.004827	0.144445	0.176841
44	0.001909	0.004755	0.145726	0.176364
45	0.001858	0.004685	0.146961	0.175896
46	0.001810	0.004615	0.148152	0.175436
47	0.001765	0.004545	0.149300	0.174984
48	0.001723	0.004478	0.150407	0.174541
49	0.001682	0.004410	0.151473	0.174105
50	0.001644	0.004344	0.152500	0.173677

TIME	EPC	EPS	EPI	NPI
51	0.001608	0.004278	0.153489	0.173256
52	0.001574	0.004214	0.154440	0.172843
53	0.001542	0.004150	0.155356	0.172436
54	0.001511	0.004086	0.156237	0.172036
55	0.001482	0.004027	0.157084	0.171642
56	0.001454	0.003966	0.157898	0.171255
57	0.001427	0.003907	0.158679	0.170874
58	0.001402	0.003849	0.159429	0.170498
59	0.001378	0.003792	0.160149	0.170128
60	0.001355	0.003736	0.160839	0.169762
61	0.001332	0.003681	0.161500	0.169402
62	0.001311	0.003628	0.162133	0.169047
63	0.001291	0.003575	0.162739	0.168696
64	0.001271	0.003523	0.163318	0.168350
65	0.001252	0.003473	0.163871	0.168008
66	0.001234	0.003423	0.164398	0.167670
67	0.001217	0.003374	0.164901	0.167335
68	0.001200	0.003327	0.165380	0.167004
69	0.001184	0.003280	0.165835	0.166676
70	0.001168	0.003234	0.166267	0.166351
71	0.001153	0.003190	0.166676	0.166030
72	0.001139	0.003146	0.167064	0.165711
73	0.001124	0.003103	0.167430	0.165395
74	0.001111	0.003061	0.167776	0.165081
75	0.001098	0.003019	0.168101	0.164769
76	0.001085	0.002979	0.168406	0.164460
77	0.001072	0.002940	0.168692	0.164152
78	0.001060	0.002901	0.168958	0.163847
79	0.001048	0.002863	0.169206	0.163543
80	0.001037	0.002826	0.169436	0.163240
81	0.001026	0.002789	0.169648	0.162939
82	0.001015	0.002753	0.169843	0.162640
83	0.001004	0.002718	0.170021	0.162341
84	0.000994	0.002684	0.170182	0.162044
85	0.000984	0.002651	0.170327	0.161748
86	0.000974	0.002618	0.170456	0.161452
87	0.000965	0.002585	0.170569	0.161157
88	0.000955	0.002554	0.170667	0.160863
89	0.000946	0.002523	0.170751	0.160570
90	0.000937	0.002492	0.170819	0.160277
91	0.000929	0.002462	0.170874	0.159984
92	0.000920	0.002433	0.170915	0.159692
93	0.000912	0.002405	0.170942	0.159400
94	0.000904	0.002376	0.170955	0.159108
95	0.000896	0.002349	0.170956	0.158816
96	0.000888	0.002322	0.170944	0.158524
97	0.000880	0.002295	0.170919	0.158233
98	0.000873	0.002269	0.170882	0.157941
99	0.000866	0.002244	0.170833	0.157649
100	0.000858	0.002219	0.170773	0.157357

APPENDIX D

DOCUMENTATION OF DVERK ROUTINE IN IMSL LIBRARY

IMSL ROUTINE NAME - DVERK

PURPOSE - DIFFERENTIAL EQUATION SOLVER - RUNGE KUTTA-VERNER FIFTH AND SIXTH ORDER METHOD

USAGE - CALL DVERK (N,FCN,X,Y,XEND,TOL,IND,C,NW,W,IER)

[illegible] - NUMBER OF EQUATIONS. (INPUT)

[illegible] - NAME OF SUBROUTINE FOR EVALUATING FUNCTIONS. (INPUT)
THE SUBROUTINE ITSELF MUST ALSO BE PROVIDED BY THE USER AND IT SHOULD BE OF THE FOLLOWING FORM

```
SUBROUTINE FCN(N,X,Y,YPRIME)
REAL Y(N),YPRIME(N)
```

FCN SHOULD EVALUATE YPRIME(1),...,YPRIME(N) GIVEN N,X, AND Y(1),...,Y(N). YPRIME(I) IS THE FIRST DERIVATIVE OF Y(I) WITH RESPECT TO X.
FCN MUST APPEAR IN AN EXTERNAL STATEMENT IN THE CALLING PROGRAM AND N,X,Y(1),...,Y(N) MUST NOT BE ALTERED BY FCN.

X - INDEPENDENT VARIABLE. (INPUT AND OUTPUT)
ON INPUT, X SUPPLIES THE INITIAL VALUE.
ON OUTPUT, X IS REPLACED WITH XEND UNLESS ERROR CONDITIONS ARISE. SEE THE DESCRIPTION OF PARAMETER IND.

Y - DEPENDENT VARIABLES, VECTOR OF LENGTH N. (INPUT AND OUTPUT)
ON INPUT, Y(1),...,Y(N) SUPPLY INITIAL VALUES.
ON OUTPUT, Y(1),...,Y(N) ARE REPLACED WITH AN APPROXIMATE SOLUTION AT XEND UNLESS ERROR CONDITIONS ARISE. SEE THE DESCRIPTION OF PARAMETER IND.

XEND - VALUE OF X AT WHICH SOLUTION IS DESIRED. (INPUT)
XEND MAY BE LESS THAN THE INITIAL VALUE OF X.

TOL - TOLERANCE FOR ERROR CONTROL. (INPUT)
THE SUBROUTINE ATTEMPTS TO CONTROL A NORM OF THE LOCAL ERROR IN SUCH A WAY THAT THE GLOBAL ERROR IS PROPORTIONAL TO TOL. MAKING TOL SMALLER IMPROVES ACCURACY AND MORE THAN ONE RUN, WITH DIFFERENT VALUES OF TOL, CAN BE USED IN AN ATTEMPT TO ESTIMATE THE GLOBAL ERROR.
IN THE DEFAULT CASE (IND=1), THE GLOBAL ERROR IS

```
MAX(ABS(E(1)),...,ABS(E(N)))
```

WHERE E(K)=(Y(K)-YT(K))/MAX(1,ABS(Y(K))) YT(K) IS THE TRUE SOLUTION, AND

```
      Y(K) IS THE COMPUTED SOLUTION AT XEND,
      FOR K=1,2,...,N.
      OTHER ERROR CONTROL OPTIONS ARE AVAILABLE.
      SEE THE DESCRIPTION OF PARAMETERS IND AND
      C BELOW.
IND   - INDICATOR. (INPUT AND OUTPUT)
      ON INITIAL ENTRY IND MUST BE SET EQUAL TO
      EITHER 1 OR 2.
      IND = 1 CAUSES ALL DEFAULT OPTIONS TO BE
        USED AND ELIMINATES THE NEED TO SET
        SPECIFIC VALUES IN THE COMMUNICATIONS
        VECTOR C.
      IND = 2 ALLOWS OPTIONS TO BE SELECTED.  IN
        THIS CASE, THE FIRST 9 COMPONENTS OF C
        MUST BE INITIALIZED TO SELECT OPTIONS AS
        DESCRIBED BELOW.
      THE SUBROUTINE WILL NORMALLY RETURN WITH
        IND = 3, HAVING REPLACED THE INITIAL VALUES
        OF X AND Y WITH, RESPECTIVELY, THE VALUE
        XEND AND AN APPROXIMATION TO Y AT XEND.
      THE SUBROUTINE CAN BE CALLED REPEATEDLY WITH
        NEW VALUES OF XEND WITHOUT CHANGING ANY
        OF THE OTHER PARAMETERS.
      THREE ERROR RETURNS ARE ALSO POSSIBLE, IN
        WHICH CASE X AND Y WILL BE THE MOST
        RECENTLY ACCEPTED VALUES.
        IND = -3 INDICATES THAT THE SUBROUTINE WAS
          UNABLE TO SATISFY THE ERROR REQUIREMENT.
          THIS MAY MEAN THAT TOL IS TOO SMALL.
        IND = -2 INDICATES THAT THE VALUE OF HMIN
          (MINIMUM STEP-SIZE) IS GREATER THAN HMAX
          (MAXIMUM STEP-SIZE), WHICH PROBABLY MEANS
          THAT THE REQUESTED TOL (WHICH IS USED IN
          THE CALCULATION OF HMIN) IS TOO SMALL.
        IND = -1 INDICATES THAT THE ALLOWED MAXIMUM
          NUMBER OF FCN EVALUATIONS HAS BEEN
          EXCEEDED.  THIS CAN ONLY OCCUR IF OPTION
          C(7), AS DESCRIBED BELOW, HAS BEEN USED.
C     - COMMUNICATIONS VECTOR OF LENGTH 24. (INPUT IF
        IND.NE.1, AND OUTPUT).
        C IS USED TO SELECT OPTIONS AND TO RETAIN
          INFORMATION BETWEEN CALLS.  THE USER NEED
          NOT BE CONCERNED WITH THE FOLLOWING
          DESCRIPTION OF THE ELEMENTS OF C WHEN
          DEFAULT OPTIONS ARE USED (IND=1).
          HOWEVER, WHEN IT IS DESIRED TO USE IND=2
          AND SELECT OPTIONS, A BASIC UNDERSTANDING
          OF DVERK IS REQUIRED.  THE FOLLOWING
          PARAGRAPH DESCRIBES, BRIEFLY, THE BASIC
          TERMS.  FOR MORE DETAILS, SEE THE DOCUMENT
          REFERENCE.
          DVERK ADVANCES THE INDEPENDENT VARIABLE
          X ONE STEP AT A TIME UNTIL XEND IS
          REACHED.  THE SOLUTION IS COMPUTED AT
          XTRIAL = X+HTRIAL ALONG WITH AN ERROR
          ESTIMATE EST.  IF EST IS LESS THAN OR
          EQUAL TO TOL (SUCCESSFUL STEP), THE STEP
          IS ACCEPTED AND X IS ADVANCED TO XTRIAL.
```

```
            IF EST IS GREATER THAN TOL (FAILURE)
            HTRIAL IS ADJUSTED AND THE SOLUTION IS
            RECOMPUTED.  HMAG = ABS(HTRIAL) IS NEVER
            ALLOWED TO EXCEED HMAX NOR IS IT ALLOWED
            TO BECOME SMALLER THAN HMIN.  THE FIRST
            TRIAL STEP IS HSTART.  DURING THE
            COMPUTATION, A COUNTER (C(25)) IS
            INCREMENTED EACH TIME A TRIAL STEP FAILS
            TO PROVIDE A SOLUTION SATISFYING THE ERROR
            TOLERANCE.  ANOTHER COUNTER (C(22)) IS
            USED TO RECORD THE NUMBER OF SUCCESSFUL
            STEPS.  AFTER A SUCCESSFUL STEP, C(23) IS
            SET TO ZERO.
          OPTIONS.  IF THE SUBROUTINE IS ENTERED WITH
            IND=2, THE FIRST 9 COMPONENTS OF THE
            COMMUNICATIONS VECTOR MUST BE INITIALIZED
            BY THE USER.  NORMALLY THIS IS DONE BY
            FIRST SETTING THEM ALL TO ZERO, AND THEN
            THOSE CORRESPONDING TO PARTICULAR OPTIONS
            ARE MADE NON-ZERO.
C(1)      - ERROR CONTROL INDICATOR.
            THE SUBROUTINE ATTEMPTS TO CONTROL A NORM
            OF THE LOCAL ERROR IN SUCH A WAY THAT THE
            GLOBAL ERROR IS PROPORTIONAL TO TOL.
            THE DEFINITION OF GLOBAL ERROR FOR THE
            DEFAULT CASE (IND=1) IS GIVEN IN THE
            DESCRIPTION OF PARAMETER TOL.  THE DEFAULT
            WEIGHTS ARE 1/MAX(1,ABS(Y(K))).  WHEN IND=2
            IS USED, THE WEIGHTS ARE DETERMINED
            ACCORDING TO THE VALUE OF C(1).
            IF C(1)=1 THE WEIGHTS ARE 1
                      (ABSOLUTE ERROR CONTROL)
            IF C(1)=2 THE WEIGHTS ARE 1/ABS(Y(K))
                      FOR K=1,2,...,N.
                      (RELATIVE ERROR CONTROL)
            IF C(1)=3 THE WEIGHTS ARE
                      1/MAX(ABS(C(2)),ABS(Y(K)))
                      FOR K=1,2,...,N.
                      (RELATIVE ERROR CONTROL, UNLESS
                      ABS(Y(K)) IS LESS THAN THE FLOOR
                      VALUE,ABS(C(2)))
            IF C(1)=4 THE WEIGHTS ARE
                      1/MAX(ABS(C(K+30)),ABS(Y(K)))
                      FOR K=1,2,...,N.
                      (HERE INDIVIDUAL FLOOR VALUES
                      ARE USED)
                      IN THIS CASE,  THE DIMENSION OF C
                      MUST BE GREATER THAN OR EQUAL TO
                      N+30 AND C(31), C(32),...,C(N+30)
                      MUST BE INITIALIZED BY THE USER.
            IF C(1)=5 THE WEIGHTS ARE 1/ABS(C(K+30))
                      FOR K=1,2,...,N.
                      IN THIS CASE,  THE DIMENSION OF C
                      MUST BE GREATER THAN OR EQUAL TO
                      N+30 AND C(31), C(32),...,C(N+30)
                      MUST BE INITIALIZED BY THE USER.
```

FOR ALL OTHER VALUES OF C(1), INCLUDING C(1)=0 THE DEFAULT VALUES OF THE WEIGHTS ARE TAKEN TO BE 1/MAX(1,ABS(Y(K))) FOR K=1,2,...,N.

C(2) - FLOOR VALUE. USED WHEN THE INDICATOR C(1) HAS THE VALUE 3.

C(3) - HMIN SPECIFICATION. IF NOT ZERO, THE SUBROUTINE CHOOSES HMIN TO BE ABS(C(3)). OTHERWISE IT USES THE DEFAULT VALUE 10*MAX(DWARF,RREB*MAX(NORM(Y)/TOL,ABS(X))) WHERE DWARF IS A VERY SMALL POSITIVE MACHINE NUMBER AND RREB IS THE RELATIVE ROUNDOFF ERROR BOUND.

C(4) - HSTART SPECIFICATION. IF NOT ZERO, THE SUBROUTINE WILL USE AN INITIAL HMAG EQUAL TO ABS(C(4)), EXCEPT OF COURSE FOR THE RESTRICTIONS IMPOSED BY HMIN AND HMAX. OTHERWISE IT USES THE DEFAULT VALUE HMAX*(TOL)**(1/6).

C(5) - SCALE SPECIFICATION. THIS IS INTENDED TO BE A MEASURE OF THE SCALE OF THE PROBLEM. LARGER VALUES OF SCALE TEND TO MAKE THE METHOD MORE RELIABLE, FIRST BY POSSIBLY RESTRICTING HMAX (AS DESCRIBED BELOW) AND SECOND, BY TIGHTENING THE ACCEPTANCE REQUIREMENT. IF C(5) IS ZERO, A DEFAULT VALUE OF 1 IS USED. FOR LINEAR HOMOGENEOUS PROBLEMS WITH CONSTANT COEFFICIENTS, AN APPROPRIATE VALUE FOR SCALE IS A NORM OF THE ASSOCIATED MATRIX. FOR OTHER PROBLEMS, AN APPROXIMATION TO AN AVERAGE VALUE OF A NORM OF THE JACOBIAN ALONG THE TRAJECTORY MAY BE APPROPRIATE.

C(6) - HMAX SPECIFICATION. FOUR CASES ARE POSSIBLE,
IF C(6).NE.0 AND C(5).NE.0, HMAX IS TAKEN TO BE MIN(ABS(C(6)),2/ABS(C(5))).
IF C(6).NE.0 AND C(5).EQ.0, HMAX IS TAKEN TO BE ABS(C(6)).
IF C(6).EQ.0 AND C(5).NE.0, HMAX IS TAKEN TO BE 2/ABS(C(5)).
IF C(6).EQ.0 AND C(5).EQ.0, HMAX IS GIVEN A DEFAULT VALUE OF 2.

C(7) - MAXIMUM NUMBER OF FUNCTION EVALUATIONS. IF NOT ZERO, AN ERROR RETURN WITH IND = -1 WILL BE CAUSED WHEN THE NUMBER OF FUNCTION EVALUATIONS EXCEEDS ABS(C(7)).

C(8) - INTERRUPT NUMBER 1. IF NOT ZERO, THE SUBROUTINE WILL INTERRUPT THE CALCULATIONS AFTER IT HAS CHOSEN ITS PRELIMINARY VALUE OF HMAG, AND JUST BEFORE CHOOSING HTRIAL AND XTRIAL IN PREPARATION FOR TAKING A STEP (HTRIAL MAY DIFFER FROM HMAG IN SIGN, AND MAY REQUIRE ADJUSTMENT IF XEND IS NEAR). THE SUBROUTINE RETURNS WITH IND = 4, AND WILL RESUME CALCULATION AT THE POINT OF INTERRUPTION IF RE-ENTERED WITH IND = 4.

C(9) - INTERRUPT NUMBER 2. IF NOT ZERO, THE SUBROUTINE WILL INTERRUPT THE CALCULATIONS IMMEDIATELY AFTER IT HAS DECIDED WHETHER OR NOT TO ACCEPT THE RESULT OF THE MOST RECENT TRIAL STEP, WITH IND = 5 IF IT PLANS TO ACCEPT, OR IND = 6 IF IT PLANS TO REJECT. Y(*) IS THE PREVIOUSLY ACCEPTED RESULT, WHILE W(*,9) IS THE NEWLY COMPUTED TRIAL VALUE, AND W(*,2) IS THE UNWEIGHTED ERROR ESTIMATE VECTOR. THE SUBROUTINE WILL RESUME CALCULATIONS AT THE POINT OF INTERRUPTION ON RE-ENTRY WITH IND = 5 OR 6. IND MAY BE CHANGED BY THE USER IN ORDER TO FORCE ACCEPTANCE OF A STEP (BY CHANGING IND FROM 6 TO 5) THAT WOULD OTHERWISE BE REJECTED, OR VICE VERSA.

NW - ROW DIMENSION OF THE MATRIX W EXACTLY AS SPECIFIED IN THE DIMENSION STATEMENT IN THE CALLING PROGRAM. (INPUT)

W - WORKSPACE MATRIX. THE FIRST DIMENSION OF W MUST BE NW AND THE SECOND MUST BE GREATER THAN OR EQUAL TO 9.

IER - ERROR PARAMETER. (OUTPUT)
TERMINAL ERROR
- IER = 129, NW IS LESS THAN N OR TOL IS LESS THAN OR EQUAL TO ZERO.
- IER = 130, IND IS NOT IN THE RANGE 1 TO 6.
- IER = 131, XEND HAS NOT BEEN CHANGED FROM PREVIOUS CALL OR X IS NOT SET TO THE PREVIOUS XEND VALUE.
- IER = 132, THE RELATIVE ERROR CONTROL OPTION (C(1)=2) WAS SELECTED AND ONE OF THE SOLUTION COMPONENTS IS ZERO.

PRECISION/HARDWARE - SINGLE AND DOUBLE/H32
- SINGLE/H36,H48,H60

REQD. IMSL ROUTINES - UERTST,UGETIO

NOTATION - INFORMATION ON SPECIAL NOTATION AND CONVENTIONS IS AVAILABLE IN THE MANUAL INTRODUCTION OR THROUGH IMSL ROUTINE UHELP

REMARKS 1. IN A TYPICAL SITUATION, DVERK IS CALLED REPEATEDLY WITH A SEQUENCE OF VALUES FOR XEND. AFTER EACH SUCH CALL, THE USER SHOULD INTERROGATE IND AND IER. ERROR CONDITIONS ARE SIGNALED WHEN IND IS LESS THAN ZERO AND/OR IER IS GREATER THAN ZERO. CORRECTIVE ACTION (SUCH AS CHANGING CERTAIN PARAMETER VALUES) MUST BE TAKEN PRIOR TO RE-ENTRY.

2. WHEN ERROR CONDITIONS ARISE, IT IS OFTEN HELPFUL TO EXAMINE COMPONENTS OF THE COMMUNICATIONS VECTOR C. A SUMMARY FOLLOWS;

PRESCRIBED AT THE OPTION OF THE USER

C(1) ERROR CONTROL INDICATOR
C(2) FLOOR VALUE
C(3) HMIN SPECIFICATION
C(4) HSTART SPECIFICATION
C(5) SCALE SPECIFICATION
C(6) HMAX SPECIFICATION
C(7) MAXIMUM NUMBER OF FCN EVALUATIONS
C(8) INTERRUPT NUMBER 1
C(9) INTERRUPT NUMBER 2

DETERMINED BY THE PROGRAM

C(10) RREB (RELATIVE ROUNDOFF ERROR BOUND)
C(11) DWARF (VERY SMALL MACHINE NUMBER)
C(12) WEIGHTED NORM OF Y
C(13) HMIN
C(14) HMAG
C(15) SCALE
C(16) HMAX
C(17) XTRIAL
C(18) HTRIAL
C(19) EST
C(20) PREVIOUS XEND
C(21) FLAG FOR XEND
C(22) NUMBER OF SUCCESSFUL STEPS
C(23) NUMBER OF SUCCESSIVE FAILURES
C(24) NUMBER OF FCN EVALUATIONS

IF C(1) = 4 OR 5, C(31),C(32),...,C(N+30) ARE FLOOR VALUES.

3. PARAMETER NW GIVES THE ROW DIMENSION OF W EXACTLY AS IT APPEARS IN THE DIMENSION STATEMENT IN THE CALLING PROGRAM. IF ONLY ONE SYSTEM OF EQUATIONS IS BEING SOLVED, NW NORMALLY WILL HAVE THE SAME VALUE AS N. HOWEVER, IF MORE THAN ONE SYSTEM IS BEING HANDLED, AND THEY ARE TO USE A COMMON WORKSPACE, W, ONE AFTER THE OTHER, THE VALUE OF NW (AND HENCE, THE ROW DIMENSION OF W IN THE CALLING PROGRAM) MUST BE AS LARGE AS THE MAXIMUM VALUE OF THE INDIVIDUAL N VALUES.

Algorithm

DVERK finds approximations to the solution of a system of first order ordinary differential equations of the form y'=f(x,y) with initial conditions. It is designed to be easy to use. By setting parameter IND to 1, the user need only provide parameters to describe the problem; everything else is done automatically by the subroutine. Alternatively, the user may set IND to 2 and then select any one of several options, including different kinds of error control, restrictions on step sizes, and interrupts which permit the user to examine the state of the calculations (and perhaps make modifications) during intermediate stages. DVERK attempts to keep the global error proportional to a tolerance

specified by the user. The proportionality depends on the kind of error control that is used as well as the differential equation and the range of integration.

DVERK is efficient for non-stiff systems where derivative evaluations are not expensive and where solutions are not required at a large number of finely spaced points (as might be the case for example with graphical output). See the Chapter D prelude for general guidelines.

The subroutine is based on a code designed by T. E. Hull, W. H. Enright, and K. R. Jackson. It uses Runge-Kutta formulas of orders 5 and 6 that were developed by J. H. Verner.

See references:

1. T. E. Hull, W. H. Enright, and K. R. Jackson, "User's Guide for DVERK - A Subroutine for Solving Non-Stiff ODE's", TR No. 100, Department of Computer Science, University of Toronto, October, 1976.

2. K. R. Jackson, W. H. Enright, and T. E. Hull, "A Theoretical Criterion for Comparing Runge-Kutta Formulas TR101", January, 1977.

Example 1

This example illustrates the basic usage (all default options) of DVERK. A table of solution values for x = 1.0,2.0,...,10.0 is obtained for the predator-prey problem:

$$y_1' = 2y_1(1-y_2) \qquad y_1 = 1$$
$$y_2' = y_2(y_1-1) \qquad y_2 = 3 \qquad \text{at } x = 0$$

```
      INTEGER   N,IND,NW,IER,K
      REAL      Y(2),C(24),W(2,9),X,TOL,XEND
      EXTERNAL  FCN1
      NW    = 2
      N     = 2
      X     = 0.0
      Y(1)  = 1.0
      Y(2)  = 3.0
      TOL   = .0001
      IND   = 1
      DO 10 K=1,10
        XEND=FLOAT(K)
        CALL DVERK(N,FCN1,X,Y,XEND,TOL,IND,C,NW,W,IER)
        IF(IND.LT.0.OR.IER.GT.0) GO TO 20
                     Y(1) and Y(2) are current solution values at X.
                     Insert write statement here.
   10 CONTINUE
      STOP
   20 CONTINUE
```

specified by the user. The proportionality depends on the kind of error control that is used as well as the differential equation and the range of integration.

DVERK is efficient for non-stiff systems where derivative evaluations are not expensive and where solutions are not required at a large number of finely spaced points (as might be the case for example with graphical output). See the Chapter D prelude for general guidelines.

The subroutine is based on a code designed by T. E. Hull, W. H. Enright, and K. R. Jackson. It uses Runge-Kutta formulas of orders 5 and 6 that were developed by J. H. Verner.

See references:

1. T. E. Hull, W. H. Enright, and K. R. Jackson, "User's Guide for DVERK - A Subroutine for Solving Non-Stiff ODE's", TR No. 100, Department of Computer Science, University of Toronto, October, 1976.

2. K. R. Jackson, W. H. Enright, and T. E. Hull, "A Theoretical Criterion for Comparing Runge-Kutta Formulas TR101", January, 1977.

Example 1

This example illustrates the basic usage (all default options) of DVERK. A table of solution values for x = 1.0,2.0,... 10.0 is obtained for the predator-prey problem:

$$y_1' = 2y_1(1-y_2) \qquad y_1 = 1$$
$$y_2' = y_2(y_1-1) \qquad y_2 = 3 \qquad \text{at } x = 0$$

```
      INTEGER   N,IND,NW,IER,K
      REAL      Y(2),C(24),W(2,9),X,TOL,XEND
      EXTERNAL  FCN1
      NW   = 2
      N    = 2
      X    = 0.0
      Y(1) = 1.0
      Y(2) = 3.0
      TOL  = .0001
      IND  = 1
      DO 10 K=1,10
        XEND=FLOAT(K)
        CALL DVERK(N,FCN1,X,Y,XEND,TOL,IND,C,NW,W,IER)
        IF(IND.LT.0.OR.IER.GT.0) GO TO 20
                    Y(1) and Y(2) are current solution values at X.
                    Insert write statement here.
   10 CONTINUE
      STOP
   20 CONTINUE
```

```
                Handle IND.LT.0 or IER.GT.0
                Items that may help diagnose the problem should be
                output here.
                IND,TOL,N,W,Y(1),...,Y(N),XEND, and C(1),...,C(24).
STOP
END
SUBROUTINE      FCN1(N,X,Y,YPRIME)
INTEGER         N
REAL            Y(N),YPRIME(N),X
YPRIME(1) = 2.0*Y(1)*(1.0-Y(2))
YPRIME(2) = Y(2)*(Y(1)-1.0)
RETURN
END
```

Output:

IER = 0

X	Y(1)	Y(2)
1.	0.08	1.46
2.	0.09	0.58
3.	0.29	0.25
4.	1.45	0.19
5.	4.05	1.44
6.	0.18	2.26
7.	0.07	0.91
8.	0.15	0.37
9.	0.65	0.19
10.	3.15	0.35

Example 2

This example shows how IND = 2 is used to select specific options, while using default values for others. The differential equation

$$y' = y \quad , \quad y = 1 \text{ at } x = 0,$$

is solved for x = .1,.2,...,1.0, using the absolute error control option (C(1)=1).

```
INTEGER   N,IND,NW,IER,I,K
REAL      Y(1),C(24),W(1,9),X,TOL,XEND
EXTERNAL  FCN2
NW    = 1
N     = 1
X     = 0.0
Y(1)  = 1.0
TOL   = 0.0005
IND   = 2
                Select all default options, first
DO 5 I=1,9
5 C(I) = 0.0
                 Then specify C(1)=1.0 to select the absolute error
                 control option.
C(1) = 1.0
DO 10 K=1,10
  XEND = FLOAT(K)*0.1
  CALL DVERK(N,FCN2,X,Y,XEND,TOL,IND,C,NW,W,IER)
  IF(IND.LT.0.OR.IER.GT.0) GO TO 20
```

```
C              Y(1) is the current solution value at X.  Insert write
C              statement here.
10 CONTINUE
   STOP
20 CONTINUE
C              Handle IND.LT.0 or IER.GT.0
C              Items that may help diagnose the problem should be
C              output here.
C              IND,TOL,N,X,Y(1),...,Y(N),XEND, and C(1),...,C(24).
   STOP
   END
   SUBROUTINE     FCN2(N,X,Y,YPRIME)
   INTEGER        N
   REAL           Y(N),YPRIME(N),X
   YPRIME(1) = Y(1)
   RETURN
   END
```

Output:

IER = 0

X	Y(1)
.1	1.105
.2	1.221
.3	1.350
.4	1.492
.5	1.649
.6	1.822
.7	2.014
.8	2.226
.9	2.460
1.0	2.718

```
                Y(1) is the current solution value at X.  Insert write
                statement here.
10 CONTINUE
   STOP
20 CONTINUE
                Handle IND.LT.0 or IER.GT.0
                Items that may help diagnose the problem should be
                output here.
                IND,TOL,N,X,Y(1),...,Y(N),XEND, and C(1),...,C(24).
   STOP
   END
   SUBROUTINE      FCN2(N,X,Y,YPRIME)
   INTEGER         N
   REAL            Y(N),YPRIME(N),X
   YPRIME(1) = Y(1)
   RETURN
   END
```

Output:

IER = 0

X	Y(1)
.1	1.105
.2	1.221
.3	1.350
.4	1.492
.5	1.649
.6	1.822
.7	2.014
.8	2.226
.9	2.460
1.0	2.718

```
                Y(1) is the current solution value at X.  Insert write
                statement here.
10 CONTINUE
   STOP
20 CONTINUE
                Handle IND.LT.0 or IER.GT.0
                Items that may help diagnose the problem should be
                output here.
                IND,TOL,N,X,Y(1),...,Y(N),XEND, and C(1),...,C(24).
   STOP
   END
   SUBROUTINE      FCN2(N,X,Y,YPRIME)
   INTEGER         N
   REAL            Y(N),YPRIME(N),X
   YPRIME(1) = Y(1)
   RETURN
   END
```

Output:

IER = 0

X	Y(1)
.1	1.105
.2	1.221
.3	1.350
.4	1.492
.5	1.649
.6	1.822
.7	2.014
.8	2.226
.9	2.460
1.0	2.718

REFERENCES

BERTALANFFY, L. VON. General Systems Theory. Penguin Press (1971).

COUGER, J.D. Systems Analysis Techniques. John Wiley & Sons (1974).

GOLD, H.J. Mathematical Modeling of Biological Systems. N.Y. John Wiley & Sons (1977).

KLIR, G.J. Trends in General Systems Theory. John Wiley & Sons (1972).

MESAROVIC, M.D. General Systems Theory: Mathematical Foundations. Academic Press (1975).

MILLER. J.G. Living Systems. McGraw-Hill (1978).

MOROWITZ, H.J. Energy Flow in Biology. Academic Press (1968).

ORTEN, J.M. & NEUHAUS, O.W. Biochemistry. C.V. Mosby Company (1970).

ROSEN, R. Dynamical System Theory in Biology, Vol. I. John Wiley & Sons (1970).

MESAROVIC, M.D., MACKO, D, TAKAHARA, Y. Theory of Hierarchical Multilevel Systems(1970). Academic Press.

SCUDO, F.M. Golden Age of Theoretical Ecology (1978). Springer.Verlag.

VON BERTALANFFY, L. AND RAPOPORT, A.(ed.). General Systems Yearbook, Vol. 1-19. (1956-1974). Ann. Arbor: Society for General Systems Research.

LA VIOLETTE, P. The Predator-Prey Relationship in Stock Market. General Systems (1974).

KLIR, G.J. An approach to General Systems Theory (1969). Nitton Educational Publishing Inc.

WEINBURG, G.M. An Introduction to General Systems Thinking (1975). John Wiley & Sons, Inc.

ZEIGLER. B.P. Theory of Modeling and Simulation (1976). John Wiley.

17. GOODMAN, M.R. Study notes in System Dynamics (1974). Wright-Allen Press.

18. REITMAN, J. Computer Simulation Applications (1971). John Wiley.

19. BRITSKER, A.A.B and PEGDEN, C.D. Introduction to Simulation and SLAM (1979). A.A.B. Pritsker.

20. YAOHAN, C. Digital Simulation of Continuous Systems (1969). McGraw-Hill.

21. MEADOWS, D.L. Dynamics of Commodity Production Cycles (1970). Wright-Allen Press.

22. Burroughs Machines DYNAMO (User's manual) (1975). Burroughs Machines USA.

23. HEINMETS, F. Quantitative Cellular Biology (1970). Marcel Dekker.

24. HEINMETS, F. (ed.) Concepts and Models of Biomathematics (1969). Marcel Dekker.

25. SAVAGEAU, M.A. Biochemical Systems Analysis (1976). Addison-Wesley.

26. JAMES, M.L., SMITH, G.M. and WOLFORD, J.C. Applied Numerical Methods for Digital Computation (1977). Y. Crowell Company.

27. YOURDON, E. and CONSTANTINE, L.L. Structured Design (Fundamentals of a Discipline of Computer Program and Systems Design) (1979). Prentice-Hall.

28. SEGEL, L.A. (ed.) Mathematical Models in Molecular and Cellular Biology (1980). Cambridge University Press.

29. IMSL edition 9, (1982). IMSL Houston, Texas.

30. LAMBERT, J.D. Computational Methods in Ordinary Differential Equations. John Wiley & Sons (1973).

31. VICHNEVETSKY, R. Computer Methods for Partial Differential Equations, Volume I (1981). Prentice-Hall.

BIBLIOGRAPHY

LERNER, A.Y. Fundamentals of Cybernetics (1972). Chapman & Hall.

GEORGE, F.H. The Brain and the Computer (1973). Pergamon Press.

ILERAL, A.S. Towards a General Science of Viable Systems (1972). McGraw-Hill.

WHYTE, L.L., WILSON, A.G., WILSON, D. (ed.). Hierarchical Structures (1969). American Elsevier Publishing Company.

KAPP, K.W. Towards a Science of Man in Society (1961). The Hague Martinus Mijhoff.

BLALOCK, H.M. Jr. Theory Construction.

TRAPPL, R., PICHLER, F.R. (ed.). Progress in Cybernetics and Systems Research. Vol. I).

MONOD, J. Change and Necessity (1970). Collins.

KALMAN, R.E., FALB, P.L., ARBIB, M.A. Topics in Mathematical Systems Theory. McGraw-Hill.

SIMON, H.A. The Architecture of Complexity, in Proceedings of the American Philosophical Society.

MILSAM, J.H. Biological Control Systems Analysis (1966). McGraw-Hill.

WIENER, N. Cybernetics (1961). MIT and John Wiley.

KILBRIDGE, M., WESTER, L. An Economic model for the Division of Labor.

JIRI, K., MIROSLOV, V. Cybernetic Modeling (1967). Iliffe Books.

HEILBRONER, R.L. Understanding Macroeconomics (1972). Prentice-Hall.

GOSTA, E. Life: Origin and Development (1962). The University of Chicago Press.

BUTLER, S.T., MESSEL, H. Light and Life in the Universe (1965). Pergamon Press.

ROBERTS, F.S. Discrete Mathematical Models (1976). Prentice-Hall.

MAY, R.M. Stability and Complexity in Model Ecosystems (1973/4). Princeton.

GLANSDORFF, P., PRIGOGINE, I. Thermodynamic Theory of Stability and Fluctuations (1971). Wiley Interscience.

LOCKER, A.(ed.). Biogenesis Evolution Homeostasis (1973).

THOM, Réné. Structural Stability and Morphogenesis (1975). W.A. Benjamin.

FORRESTER, J.W. Principles of Systems (1968). Wright-Allen Press.

COUGER, J.D. Systems Analysis Techniques (1974). John Wiley & Sons.

ROSEN, R.(ed.). Foundations of Mathematical Biology. Vol. I, II, III) (1972/3). Academic Press.

FORRESTER, J.W. World Dynamics (1971). Wright-Allen Press.

KLIR, G.J.(ed.). Applied General Systems Research (1978). Plenum Press.

ORTEGA, J.M. and POOLE, W.G. Jr. Numerical methods for Differential Equations (1981). Pitman Publishing Inc.

RALSTON, A. A First Course in Numerical Analysis. McGraw-Hill, Inc. (1965).

GERALD, C.F. Applied Numerical Analysis. Addison-Wesley Publishing Company (1978).

GEAR, C.W. Numerical Initial Value Problems in Ordinary Differential Equations. Prentice-Hall, Inc. (1971).

HENRICI, P. Discrete Variable Methods in Ordinary Differential Equations. John Wiley & Sons, Inc. (1962).

HENRICI, P. Error Propagation for Differential Methods. John Wiley and Sons, Inc. (1963).

Author Kambanis J
Name of thesis An approach to the study of organised systems 1983

PUBLISHER:
University of the Witwatersrand, Johannesburg
©2013

LEGAL NOTICES:

Copyright Notice: All materials on the University of the Witwatersrand, Johannesburg Library website are protected by South African copyright law and may not be distributed, transmitted, displayed, or otherwise published in any format, without the prior written permission of the copyright owner.

Disclaimer and Terms of Use: Provided that you maintain all copyright and other notices contained therein, you may download material (one machine readable copy and one print copy per page) for your personal and/or educational non-commercial use only.

The University of the Witwatersrand, Johannesburg, is not responsible for any errors or omissions and excludes any and all liability for any errors in or omissions from the information on the Library website.