COMPUTERIZE DATA INFORMATION FOR ARCHITECTURE DESIGN

by

Syau-yi Yu

Thesis submitted to the Faculty of the Graduate School of the New Jersey Institute of Technology in partial fulfillment of the requirements for the degree of Master of Science in Architecture 1989

APPROVAL SHEET

Title of thesis : Computerize Data Information for Architecture Design

Name of candidate : Syau-yi Yu
Master of Science in Architecture

Thesis and Abstract Approved : ______________________ ____________

Dr. David L. Hawk — Date
Professor
School of Architecture

Signature of other members of the thesis committee : ______________________ ____________

McGlumphy James — Date
CAD Professor
School of Architecture

: ______________________ ____________

K.L. Wei — Date
Architect
K.L. Wei & Associates

VITA

Name :	Syau-yi Yu		
Permanent address :			
Degree and date to be conferred :	M. S. in Arch July 1989		
Date of birth :			
Place of birth :			
Secondary education :	National Taipei Institute of Technology		
Collegiate institutions attened	date	degree	date
National Taipei institute of Technology	1981-85	B.Arch	June 1985
New Jersey Institute	1986-89	M.S.in Arch	July 1989
Major :	Architecture		

ABSTRACT

Title of Thesis: Computerize Data Information for Architectural Design

Syau-yi Yu, Master of Science in Architecture, 1989

Thesis directed by: Dr. David L. Hawk, Professor of Architecture

Computer-aided architectural design is technically and commercially feasible yet experience in the architectural profession to date is paradoxical. Many who apply computers to architecture appear satisfied but the tangible benefits are seldom obvious and even where computers are used in firms it is not done to the fullest potential. Contents of this thesis are an attempt to help architecture better utilize the advantages of computer applications.To do this it is important to examine two question: what is design and what is the computer? They are similar yet distinct. Architectural design is a special kind of problem solving process; an information processing task. Design is the organization and processing of information. Computers are electronic devices capable of storing data in an internal memory, as well as storing sets of instructions (known as programs) that operate upon data. The minds behind design accomplish similar ends with different means.

The difficulty of designers solving problems is proportional to the volume of data to be handled and the quantity of interconnected relationships between functions. Various types of computers and soft-wares are available to help handle large volumes of signs and data, and help operate them according to instructions. The output of processing may be in written or graphic form and is meant to assist us in analyzing complicated problems and making decisions. It can even lead to new forms of analysis and planning.

Computerization of production-information procedures such as computer selection and combination of standard details, computer produced schedules and drawings, and computer aided information retrieval, should drastically reduce the amount of effort needed in the process. New patterns of organization of the design process must then inevitably arise from such drastic changes.

TABLE OF CONTENTS

INTRODUCTION

Data can represent various kinds of facts, and lead to conclusions. It provides us with a basis for our actions and can be applied to our analysis or calculations. Information is a kind of knowledge or message, a conclusion reached as a result of our review of certain knowledge or experiences. Most data is of a fixed form which we can use when we make choices from its organization. Data in experience has less of a fixed form. Sometimes it may can only be used only after being processed in one way or another. Thus in both senses data can be regarded as processed information. In a general sense, information is organization data.

Design behavior is a continuous process of planning and decision-making. A good design is seldom done by chance, guess work or conjecture. On the contrary, it is based on an appropriate model and achieved through a number of decisions based on relative information. A design depends on the production and utilization of correct information or data. During the design process, we have to have information for various processes and stages to support our decision. The more complete our information is the more reliable and more practical our designs will be.

Better information is not necessarily a cure-all. We shall not be able to determine the direction of design simply by means of better information. It is only one. We must not forget less tangible parts like the designer's experience and judgments needed for choice. From this we can define the major factors contributing to decisions.

The study explores the methods by which needed information is collected and organzed in the design process. It deals with categories, sources of information, and how both can be investigated. The effect of changes in information frequency and duration of use will be examined. Methods of forecasting future use-patterns will be studied, using the quantitative and qualitative methods. Finally, documentation as the storage and management of information will be discussed.

Almost all information or data related to architecture, such as that associated with scale, standards and specifications associated with performance can be incorporated to computer language and indexed according to need. The data base can thus offer a basis for making generalizations about the relations of building form to technology to performance.

I. CLASSIFICATION OF INFORMATION

From the standpoint of an architect, two major types of information are ;1) project specific information, and 2) general information.

(1) Project information: this refers to all information related to the project, such as the design brief, scheme design, production drawings, specification, construction items sheet, estimates, and correspondence. Such information represents information related to the project itself rather than being general in nature. It comes from and is available to architects, surveyors, contractors, customers, and others.

As design of a project progresses, outputs from various parallel procedures accumulate and an extensive, complex, "project data base" is built up. Some of the information in the data-base is simply temporary working data, where the output of one procedure is temporarily recorded so that it can be used as input to another procedure. Sketches, notes, and mental notes fall into this category. Other information must be recorded more permanently, and is eventually incorporated in the working drawings, specifications, etc., which constitute the final design. In one sense, a design project may be thought of as the building up of a body of data which consistently and completely describes a proposed building.

(2) General information: This refers to the information not specific to a given project and can be applied to any project or any person. It includes information derived from former projects, or derived in the office via libraries or other media. It can be put into general service in the libraries and presented in a common form. Those who need such information can obtains through indexes related to the subject.

Design procedures often require data of a general, non project- specific type. Such as data include in catalogues of building materials and components, libraries of standard design details, and description of previous buildings. This may be referred to as "library data" , as distinguished from specific project data.

The above discussion represents general considerations of information processing in a common architects' office. As a matter of fact, the information needed in the design process cannot be defined in a generalized manner, rather we have to explore it from all aspects throughout the process. The information needed in various aspects will be nothing more than the following:

(1) Program Information: plainning organization, personnel organization, decision-making organization, budget, and planning time, etc.

(2) Internal Information: the client's initiatives on the project, the policy, the internal situation within the client's organization,the plan of operation, methods of operation, capital available, the association with other enterprises or processions, the nature of the project, and project requirements, etc.

(3) External Information: the general social situation, political situation, social values, living condition, regional status social changes, technical status and changes, and forecasting for the future, etc.

(4) Initial Information: pastr records, economic status, key points of requirements, intention, conditions for site selection, programe background, policy, forcasting for the future, life cycle, potential direction of project, direction of concept for the future, and social demands, etc.

(5) Planning Information: the information is related to:

- site: site status, climate, folk-ways, communication and road network, service facilities, regulations and codes.

- operation: investment plan, operation plan and policy, business plan, association with other enterprises.

- scale: that stipulated, that planned.

- functions: service conditions, management conditions, organization, and other conditions associated with these

- performance: level of criteria, lasting requirement, internal environment, type of structure, and facility capacity.

- idea: characteristics, futures, and client's propositions.

- time: client's desire, proposed, and planned by phases.

- economy: financial plan, status of expenditures, staging, and local unit prices.

(6) Conceptional Information: it includes information related to:

- environment: situation associated with the city pattern, and social structure.

- society: cultural, customary, historic aspects, and current tendency.

- functions: nature of space, and organization of space.

- psychology: the behavior, and feelings of people going to and out of the building, and of spectators.

- idea: form, vision, intuition, and style.

(7) Physical Information:

the information associated with quality includes:

- performance: lighting, sound, strength, air flow, ventilation, and durability.

- technology: production method, production techniques materials, and method of construction.

the information associated with quantity includes:

- scale: scale of personal requirements, accommodation capacity, activity to be accommodated, devices and appliancies, machinery requirement scale, dimensions of space, and additional scale scheduled.
- functions: number of rooms required, number of buildings, and number of floors.
- regulations and codes: highs limit, coverage, safety, light, and ventilation.

(8) Evaluation Information: information available for establishing the objective at the beginning can be used for evaluative purpose, and all reference and examples may be drawn from past ex-perience.

A useful destination can be drawn between two types of building data descriptions that are commonly employed in architectural design, which are "Geometric Descriptions" and " Non-geometric Description". Geometric descriptions, such as plans, elevation, section, and prespections, which describe the dimensions, shapes, and locations of elements; Non-geometric descriptions, such as accommodation, door, and window schedules, bills of quantitis, etc., which describe quantities, costs, and other mostly non-spatial attributes of elements.

Those various of information we have described, some belong to Non-geometric description, some are Geometric description. Usually for programming mostly we use Non-geometric description; for designing we use Geometric description. Due to the fact that the information required for programming and that required for designing often come from the same source, the collection of information for both purposes may by done at the same time. These information need to be described in both geometric and non-geometric. These two types of descriptions play different and complementary roles in a traditional design process.

1. Non-Geometric Description

In design process geometric descriptions are usually primary, and non-geometric descriptions are used in three kind of supplementary ways: 1) To described building before form decision have been made, and thus before any geometric descriptions exist.The initial schedule of accommodation requirements is an example of a non-geometric descriptions in this role. 2) To record component selection decisions. Door, window, and furniture and fitting schedules are examples of this application. 3) To facilitate tasks of tabulating, aggregating, and checking quantities, costs, areas, etc. bills of quantities, and to a certain extent accommodation schedules, are used in this way.

Construction of a useful non-geometric building description presupposes the existence of an appropriate scheme by means of which elements can be uniquely identified and unambiguously classified.

1.1 S F B system

U C D was an internationally adopted system. It used complicated digits for classification and was found difficult to practice. Now the S f B system is more commonly used internationally. This new system appears to meet our requirements adequately and will be discussed in the following section.

S f B was derived from the Swedish, The Technical Secretariat of the coordinating Committee for Building. This system was developed as an auxiliary to the scope of buildings in U B C system or facilitating review of :

1) estimates, specifications, arrangement of information or data required for our estimation ;

2) technical information, administrative file for information related to manufactured products ;

3) files for working (production) drawings.

Since it was developed in Sweden in 1947-1950, and because of its practicality it has been adopted by the Scandinavian countries. Before 1956 an extensive study was conducted on this classification methods by I B C C (Thee international Building Classification Committee). They found that the U D C covered a wide scope of items while the S f B system was simple yet flexible and practical. It was then proposed to combine both two into a single so-called S f B / U D C System. This was accepted by the C I B (The International Council of Building) Headquarters as an international classification method in May 1959.

With several times revisions to details of the system conducted by R I B A (Royal Institute of British Architects), S f B / U D C Building Filing Manual, 1961 ; CI / S f B-Construction Indexing Manual and Construction Index / S f B, 1966; and CI / S f B Project Manual, 1971 were published successively. Accordingly the criterion of the use of this system was established.

Meanwhile, in Denmark a proprietary classification system known as CBC (Coordinated Building Communication) was developed by the brothers Bjorn and Knud Bindslev. CBC is also directly derived from SfB, but its intended uses are wider. Not only can it be used as a product data and library classification scheme, it can also be used as a coding technique for identifying particular components and materials in working drawings, specifications, and bills of quantities, and for facilitating cross referencing between different documents. Furthermore, it was specifically designed to form a suitable basis for development of computerized construction information management and document production systems. The Bindslevs have developed and marketed a set of proprietary computer programs for this purpose.

In Sweden, a system very closely based on CBC, known as BDC, has also been developed. This is intended to introduce standariized, computer-produced contract documentation at a national scale. It employs a comprehensive "central catalogue." which lists unique codes for all commonly used building materials and components.

C I / S f B is a standard, framework, or a heading list or organizing libraries, collection of design information, and the preparation of reports, cost planning, drawings, specification, and establishing other associated building information files. Due to its flexibility, we can make use of it to the different extents regardless the size of offices. It will also provide a basis for coordination and communication and communication between the construction industries. Regardless of the extent of differences in scale and the method used for organization associaled with the construction industry, information files can be established with the minimum cost in accordance to the inclination of users and their purpose.

The application of the C I / S f B system is divided into two parts, one for project information and the other for the purpose of general information. In the former, we make classification for the matters associated with the project itself, while in the latter, may apply to any projects and contain information required for any one of them. Included are also exprojects which are fed back to the architects' offices.

The classification of general information in this system, we have five tables : Table 0, 1, 2, 3, and 4. We establish for these five different areas with five different headings. The basis of its development was to consolidate classification methods widely accepted internationally, such as :

Table 0 Physical Environment based on U D C classification

Table 1 Elements

Table 2 Constructions, Forms based on S f B basic Table of CIB

Table 3 Materials i.e. S f B original.

Table 4 Activities, Requirements based on Master Table of CIB

At the beginning of each Table, we have a matrix listing items in detail. The divisions of the heading are made in so much details that it can apply to a library or for the collection of information for most projects.

If we calssify the general information according to the S f B system, such as commercial catalogues, leaflets, and pamphlets, etc., documentation preparation will be made simpler than ever before. In fact, most regular documentations can not be calssified in accordance with the S f B system. since two or more tables of S f B system are often be involved in those documentations, including the complete documents, and their parts. For instance, three tables are involved in OUTER RED BRICK WORK, namely, Table 1-Elements and Table 2 and 3-Construction and Materials. The index editor must determine an order, namely, the CITATION ORDER for such documentations, so that he will be able to edit the index in a way appropriately for those documents having two or more tables involved when he beings index edition. (Fig.1.1-1)

Index for the outer red brick work should be as follow

Outer Wall	(21)
Brick work	F
Clay Products (dried, kilned)	G
Kilned Clay Products	G2

Therefore, the whole number will be read as (21) F G 2 and will be shown on Table 1 and Table 2 and Table 3 as Fig.1.1-2(1).

Due to the fact that we have not shown which building where use red brick outer walls and that this have nothing to do with their performance, there are not shown on Table 0 and Table 4. If the price of red brick outer walls is involved in it will go to Table 4 for classification and should be shown under (Y2). Then the indexing will become (21) FG 2 (Y2) as shown in Figure 1.1-2(2).

When such outer walls are those of a hospital, then Table 0 will be involved and its indexing should be 4 (21) FG 2. However, we must draw attention to the facts that such structures can be used in all types of buildings, not necessarily be confined to the hospitals and information seekers will not have to go to the indexing for hospitals. Thus it will be more desirable for us to simplify the index number to read simply (21) GF 2. If the available information was previously indexed, the old index may be shown on the indexing at the same time as shown in Figure 1.1-2(3).

1.2 Applications of Data Base Management System

The capacity of a computer to store, sort, merge, perform arithmetic upon, and print out large sequential files was recognized very early as powerful aid in design documentation and cost estimation. Similar sequential file techniques to those which have been employed in building documentation and cost estimation have also been utilized for handling programmatic data on large projects. Accommodation and equipment files of space management data can also be used as the basis for space management systems.

There is an alternative and more powerful approach, which is now feasible,is to employ the facilities of a general purpose database management system. this approach confers the advantage of at least some degree of system independence from particular physical implementations, allows required special programs to be relatively simply and quickly written in very high-level languages, and provides more sophisticated and flexible file manipulation and report generation capabilities.

The user conceptualizes TABLE files as tables, in which the rows are called items and the columns are called headings. To create a data structure for a particular purpose, the user defines heading names and the types of data each heading may contain: characters integer, and single or double precision floating point numbers. Each cell in the table may contain either a single value or a list of values. Internally, a table is stored as an indexed file accessed by the item names. Facilities are provided by TABLE II for creating, modifying, and deleting tables, sorting tables and columns, extracting data, totaling columns, and printing out reports.

An example of a relatively simple data structure employed in a programmatic data base management application by the Chicago architectural firm of Skidmore, Owing and merrill (Teague and davis 1972). It was implemented using TABLE II, a data base management subsystem of The ICES integrated civil engineering system. Figure 1.2-1

Another type of data structure, in this case hierarchical in character to reflect a hierarchical classification of data. This hierarchical type of structure, which is very suitable for many applications, can be implemented in several different ways. Fig.1.2-2(a), for example, illustrates an implementation in which all records are contained within a single file. This type of implementation is commonly referred to as an integrated data base. While relatively straightforward, an integrated hierarchically structured data always be entered with high-level keys.

For example, a query to the data base illustrated in Fig.1.2-2(a) would have to be stated, "Tell me about roome." An alternative, which overcomes this problem, is modular organization of the data base into several files, each one of which represents a single level of the hierarchy, as shown in Fig.1.2-2(b). Records in each file contain pointer fields, providing necessary cross-referencing between levels. Although it is generally more difficult to implement than an equivalent integrated organization, a modularized data base has several important advantages: up dates can be processed more efficiently, queries can be stated more simply and directly, and processing time for simple queries tends to be lower. Often, a particular application is most efficiently dealt with by some combination of the integrated and modular approaches, for example as shown in Fig.1.2-2(c).

When designing a hierarchically-structured database for a particular application, the hierarchy of data items must be very carefully considered.It is usually relatively easy, in a typicaldata base management system, to make minor changes in data baseorganization such as the addition or deletion of fields within a record; but the addition or deletion of an entire level in animplemented data base is likely to prove difficult and costly.

Use of data base management system requires systematic organization of the data gathering process. The basic steps of the process are:

Preparation of source documents;

Keypunching and verifying;

Input of the data to the system;

Editing and updating the data base.

Examples of source documents include employee questionnaires used to determine space and equipment inventories, and work sheets used in taking off quantities from drawings. Source document formats must be designed with considerable care to minimize ambiguities and chances of respondent error. The format should also be arranged to facilitate the keypunching of data directly from the source documents, without the need to translate information manually into some intermediate form.

Keypunching of data to convert it to machine-readable form can be carried out either by clerical staff within the organization or by a commercial keypunching service. It is usually advisable to verify the punched data in order to eliminate as possible of the errors which are inevitably introduced during the keypunching process. Verification is accomplish by repunching the data using a special machine which indicates any discrepancy between a symbol as originally punched and the verification key-stroke. During the mid-1970's,the cost of keypunching and verfication was around fifty cents per thousand characters. Thus a typical architectural data base of one or two million characters might cost $500 to $1000 to input.

In most applications the initial data base does not remain static; transactions must be executed at intervals to edit and update the data Formats for transactions can be defined using the transaction definition facilities of the data base management system.

The reports that are required from a data base are likely to be of two types: standard reports, the need for which can be anticipated in advance, and special reports, which are not anticipated in advance. The contents and formats for required reports can be specified using the report generation facilities of the data base management system. The production of standard reports can be a routine task, carried out by clerical staff using pre-written programs. However, the production of a special report requires a technically competent programmeror data base manager to write the necessary special program.

Database management systems which can handle non-geometric descriptions of buildings are now finding wide application throughout all phases of design and construction, particularly in organizations which undertake large hospital, school or commericial projects. Typical data bases dealt with include schedules of space and equipment needs, existing space and equipment inventories, finish, furniture, equipment, and component schedules.

Not only are thes data bases used during the design and construction phases of a project, but in many cases they also continue in use after occupancy as an element in the client organization's apace management system. Furthermore, where a design organization repeatedly deals with buildings of a similar character, it is often not necessary to generate new data for each new project; relatively minor editing of an existing data base from an earlier project may be all that is necessary. Thus a library of data bases from previous projects can be a valuable resource.

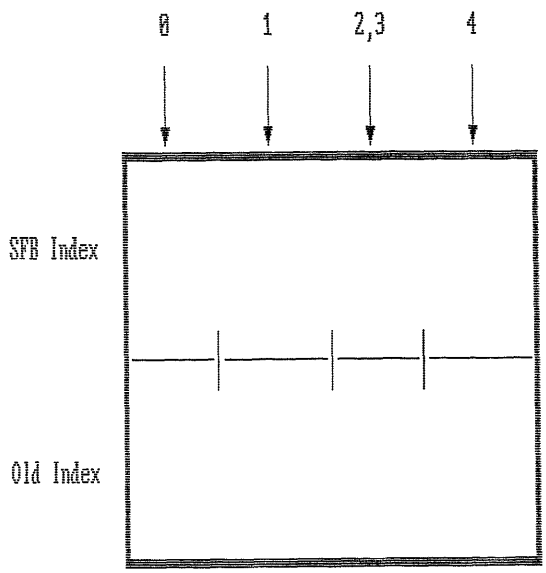
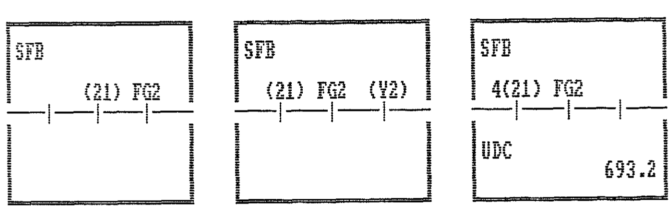


Fig. 1.1-1 Way of index



(1) SFB indexing for outer wall brick work

(2) SFB indexing and outer wall brick work cost

(3) SFB indexing and UDC indexing for outer wall brick work in a hospital

Fig. 1.1-2

Table 1 S F B Table Contents and its Application

Table	Item	Number	Final Result Example
0	Planning area	0	051 Villages
	Utilities, Civl Engineering	1	13 Waterway Communication
	Industrial facilities	2	27 Manufacturing
	Commerce,Administration	3	32 office
	Health,Welfare	4	1 Hospital
	Recreation	5	56 Sporting
	Religion	6	67 Funeral home
	Education,Science, Information	7	75 Display,Exhibition
	Inhibitation	8	81 House Construction
	Coummon Facilities	9	93 Kitchen
1	Site,Project	(1-)	
	Substructure,Structure	(2-)-(4-)	(41) Outside finishing, Painting
	Services	(5-)(6-)	(53) Water supply
	Fittings	(7-)-(8-)	(73) Cooking table
	Other	(9-)	
2	Excavation,fill	C	
	Cast in situ work	E	
	Blocks, Manufactured	F-G	F Brickwork,Bricks

	Blocks		
	Section work	H-J	I Plumbing work Pipes
	Manufactured products		
	Sheet work	K-V	V Film layer
	Component	X	
	Joints	Z	
3	Formed materials	e-o	h Metals
	Formless materials	p-s	s Asphalt
	Functional materials	t-w	u Paint
	Substance	z	
4	Activities,Aids	(A)-(D)	(A) Management
			(B) Contractor,Instrument
	Description	(E)-(G)	(F)Shapes,dimensions,etc.
	Contex,Environment	(H)	(H1)Climate,Cycle of Life
	Performance	(J)-(T)	(K) Fire,Explosion
	Users,Resources	(U)	(U1) Groups,Committee
			(U4)Physical and
			Spiritual factors
	Working factors	(V)	(V2) Demountability
	Operation,Maintenance	(W)	(W7) Restoration
	Changes,Movement	(X)	(X7) Quality Change
	Stability		
	Economy,Commerce	(Y)	(Y2) Costs,Prices
	Peripheral	(Z)	

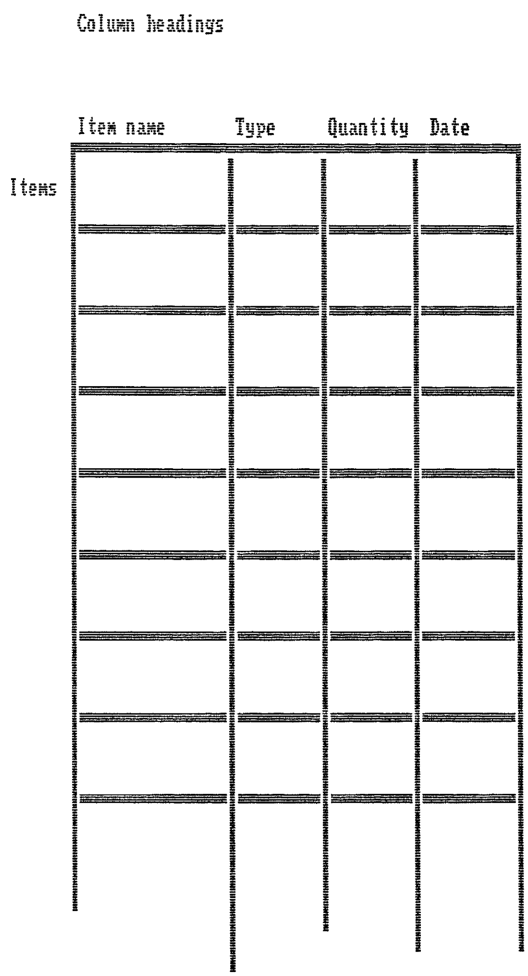
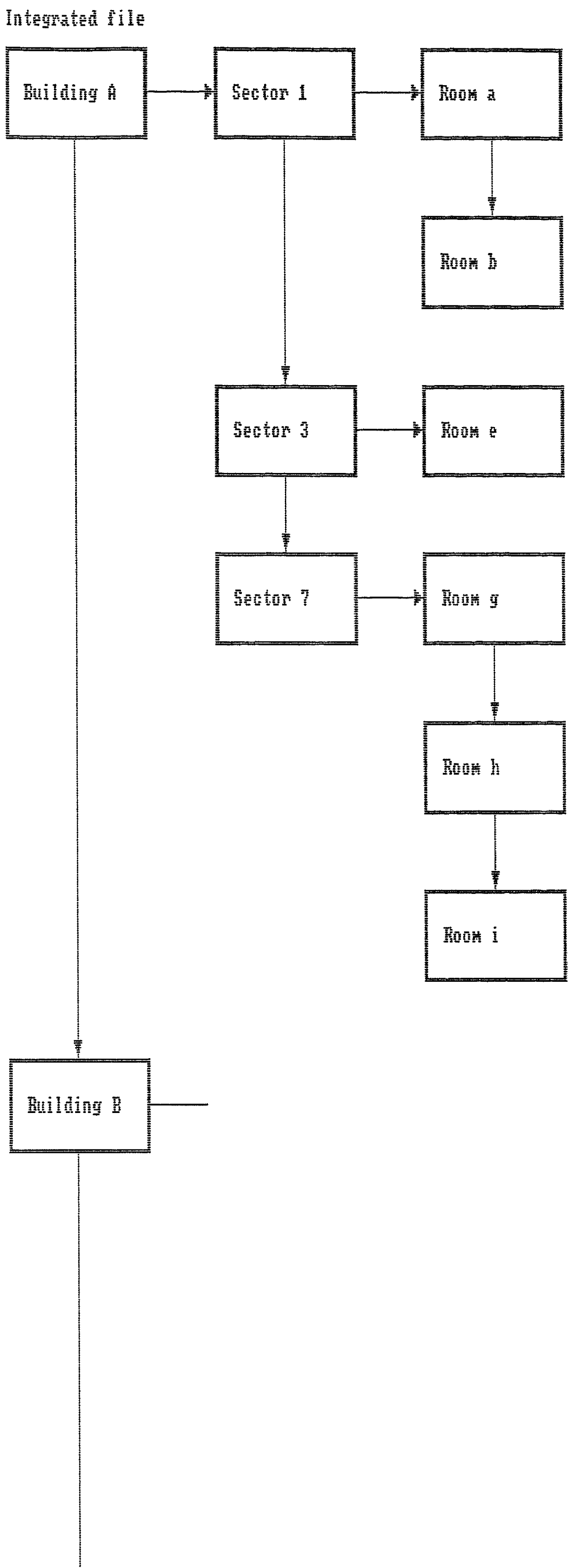


Fig.1.2-1
An example of a TABLE data structure
for storing work station requirements
data.



Integrated file
Building A
Sector 1
Room a
Room b
Sector 3
Room e
Sector 7
Room g
Room h
Room i
Building B

Fig. 1.2-2 (a)

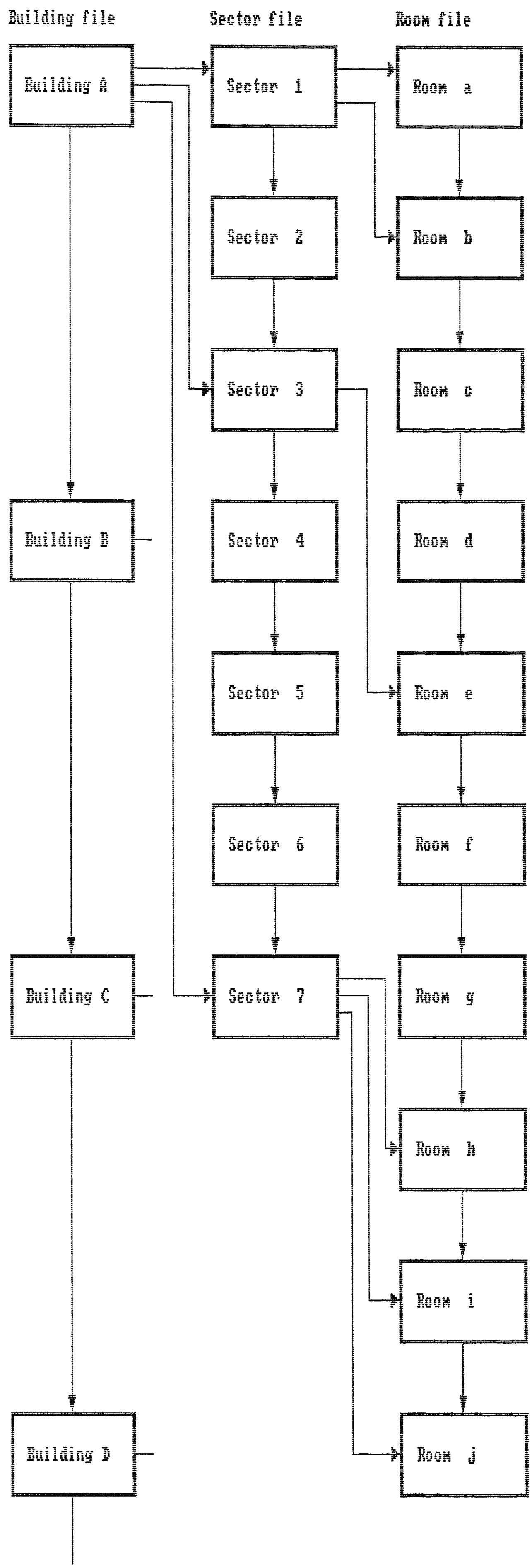

Building file
Sector file
Room file
Building A
Sector 1
Room a
Sector 2
Room b
Sector 3
Room c
Building B
Sector 4
Room d
Sector 5
Room e
Sector 6
Room f
Building C
Sector 7
Room g
Room h
Room i
Building D
Room j

Fig. 1.2-2 (b)

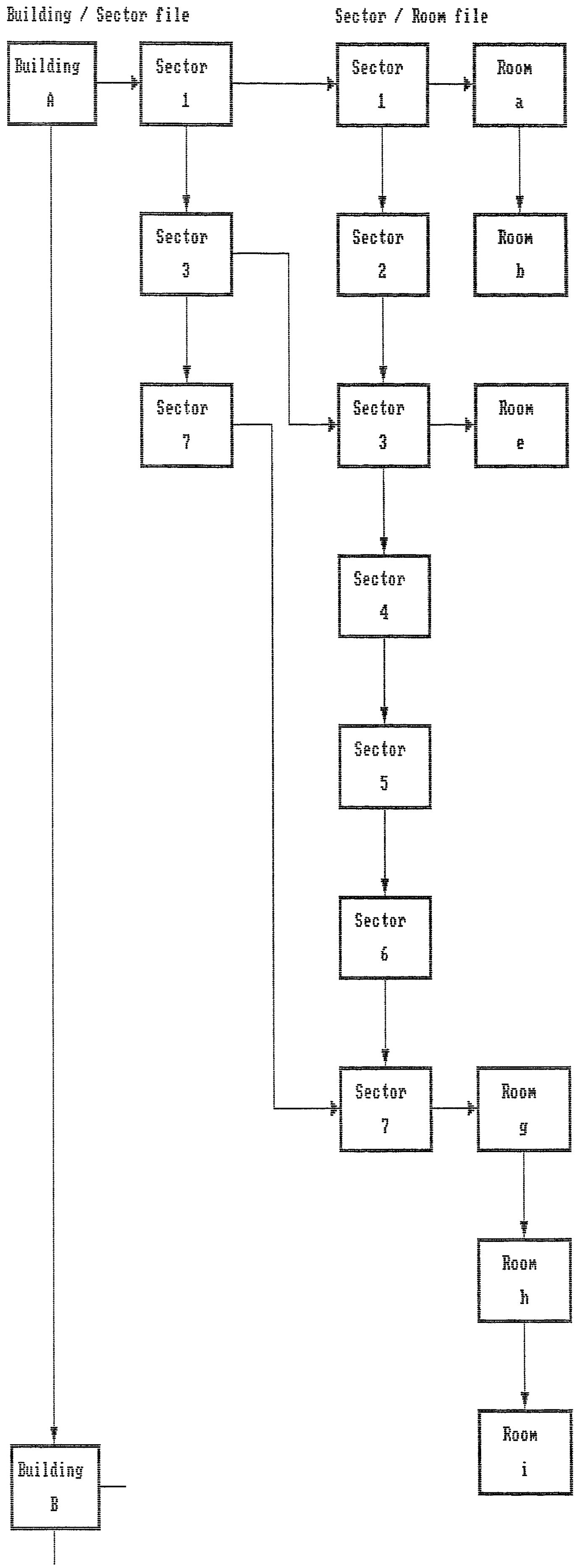

Building / Sector file
Sector / Room file
Building A
Sector 1
Sector 1
Room a
Sector 3
Sector 2
Room b
Sector 7
Sector 3
Room e
Sector 4
Sector 5
Sector 6
Sector 7
Room g
Room h
Room i
Building B

Fig 1.2-2 (c)

2. Geometric Description

Different types of geometric description can be distinguished according to:

1. The type of geometric element upon which the description is based,

2. The particular geometric and topological attributes of entities to be described. For most applications, it is necessary to describe the primary geometric properties of overall length, width, height, and orientation, and location in the building, of each element. Sometimes it is necessary to include detailed descriptions of shape properties. It may be worthwhile to explicitly store some secondary geometric data, like volume, surface area, etc. For engineering computations, it is often necessary to store non-geometric physical attributes like weight, U-vaule, etc.

3. The type of geometric and topological relations between entities to be represented,

4. The level of detail at which the description is made. When considering techniques for computer representation of building geometry, the stage of the design process for which the representation is intended must be kept in mind. A representation which incorporates too much or too little geometric information, or data at too high or too low a level of detail, will not be successful.

5. The geometric assumptions upon which the description method is based. In developing techniques for storage of geometric descriptions of built form in computer memory, it is neces sary to make tradeoffs between the efficiencies that can be attained by adopting particular geometric disciplines and the restrictions on the range of representable forms that result.

In response to the considerations discussed above,diverse approaches to structuring geometric descriptions of buildings have evolved. These can be grouped into three broad categories: 1) Representations based upon variably-dimensioned grids and lattices. 2) Polygon and polyhedron representations. 3) Dual-graph representations.

2.1 Variably-Dimensioned grids and lattices.

Use of rectangles of varying dimension as the fundamental elements of the description. Consider the three rectilinear objects shown in Figure 2.1-1. Although their dimensions and proportions are quite different, they clearly have some shape properties in common: the are all "U-shaped" objects. The observation that dimensionally very different objects can be discerned to have common shape properties suggests that it may be possible to develop descriptive techniques in which shape and dimensions are specified separately. This can be accomplished by use of a dimensionless representation in conjunction with dimensioning vectors.

The dimensional properties of a particular rectilinear shape can be specified by defining X and Y dimensioning vectors, as shown in Figure 2.1-2. the elements of the X dimensioning vector describe the widths of the columns, and the elements of the Y dimensioning vector describe the widths of the rows. Given a dimensionless representation and dimensioning vectors, the dimensioned shape can be constructed. Applying different dimensioning vectors to a dimensionless representation will produce a family of shapes in which the constituent rectangles have different dimensions and areas, but the adjacency relations between these rectangles remain constant.

For example the class of floor plans generated by dissection of a rectangle into retangles. For dissection into any given number of ractangle into ractnagles, the number of distinct dissections is limited, and there exists a sample algorithm for exhaustively enumerating these distinct dissections (Steadman 1976,Sauda 1975,Mitchell steadmen and liggett 1976). All the distinct dissections in top to six rectangles are illustrated, in dimensionless representation.

Non-rectilinear polygons and floor plans may also be reduced to dimensionless representations.However, that in the nonrectiliner case a number of different orientations of the frame with respect to the plan are possible, and that each different orientation will produce a different dimensionless representation. A further difference from the rectilinear case is that vertex angles do not necessarily remain constant as the dimensions of the girting are adjusted. Since walls need not necessarily lie along grating lines, the simple interger array form of representation cannot be employed.

Representations based upon variably-dimensioned grids and lattices appear to have found their widest application in automated floor plan layout programs.

2.2 Polygon and Polyhedron

In traditional architectural drawings,where spatial domains are implicitly represented by drawing-edge lines which define the boundaries of domains. Systems of representation based upon storing edge-lines of forms and spaces may also be utilized for describing buildings in computer memory. These may be classed as polygon and polyhedron representations.

Points are the basic building blocks of polygon and polyhedron representations. A point in space, relative to some given coordinate system, can be represented by its coordinates (x,y) in two-dimensional space or (x,y,z) in three-dimensional space. A straight line is defined by it's end points. it can be represented as a 2X2 matrix composed of the two coordinate pairs which specify the end points:

$$\begin{matrix} x1 & y1 \\ x2 & y2 \end{matrix}$$

Polygons are defined by lines which begin and end at the same point. The faces of polyhedra are polygons. A polygedron can thus be represented by several point-matrices.

Point-matrices can readily be stored in a computer by means of arrays of lists. This type of representation is very widely employed in computer graphics system, since arrays of lists storing point-matrices are readily converted into instructions to a graphics output device to generate a picture. The following type of FORTRAN statement, for example, is often used to instruct a pen plotter to draw a line:

```
CALL LINE (X1,Y1,X2,Y2)
```

where X1,Y1, and X2,Y2, are coordinates describing the startpoint and end-point, respectively, for the line on the plotting surface.

Polygon and polyhedron representations are very flexible and powerful representations. They can describe both spatial domains like rooms, and physical elements like walls, columns, and beams, to any desired level of accuracy. In principle, they are closely related to normal architectural drawings, and translations of these representations into graphic output can be easily accomplished. They may be employed for any purposes which require complete and accurate descriptions of building geometry.

There are some practical difficulties, however. For many types of architectural applications polygon and polyhedron represensations tend to be difficult to implement and complex to manipulate. Certain types of information tend to be rather difficult to extract. In particular, the efficient detection of intrisections, overlaps, and adjacencies of forms and spaces often presents a considerable problem (Comba 1968, Eastman 1970).

Polyhedron representations have been very widely employed in computer graphics and computer-aided design systems. Most of the integrated computer-aided architectural design system that have been developed employ some form of polyhedron representation technique for comprehensive building description.

2.3 Dual-Graph

Certain graphs possess the properrty of planarity. planarity can be defined as (Deo 1974): A graph G is said to be planar if there exists some geometric realization of G which can be drawn on a plane such that no two of its edges intersect.

Incidence matrices and adjacency matrices can be used to numerically encode a graph for computer storage. An incidence matrix representing a given graph can be constructed as shown in Figure 2.3-1(e).Rows represent vertices and columns represent edges. A matrix element aij is set equal to 1 if the jth edges is incident on the ith vertex, and to 0 if it is not. An adjacency matrix for a graph of n vertices is an nxn symmetrical matrix in which both rows and columns represent vertices. A matrix element aij is set equal to 1 if there is an edge between the ith and jth vertices, and to 0 if there is not.Fig. 2.3-1(f). Both incidence matrices and adjacency matrices are readily stored as two dimension arrays.

Room adjacency graphs can be encoded for computer storage by means of incidence and adjacency matrices Figure 2.3-1. Both incidence matrices and adjacency matrices may be readily stored in the computer as two-dimensional arrays. This provides a very convenient and economical way to represent and access data on floor plan topology. Whether or not two space are adjacent can be determined directly by accessing the appropriate entry in the adjacency matrix. Such topological data may be important in space planning applications, and in certain types of structure, thermal, and acoustic analyses.

The dual-graph representation of floor plans has suggested to many researchers a systematic, computer-implementable procedure for generation of floor plans which satisfy sets of adjacency requirements (Levin 1964, Krejcirik 1969, Seppanen and More 1970). The basic steps of this procedure are as follows:

1) Define an adjacency requirements matrix, specifying necessary adjacency relationships between spaces. Figure 2.3-2(a)

2) Test if the dual-graph corresponding to the adjacency requirements matrix is planar. If it is, go on to the next step. If it is not, the problem as defined has no solution.

3) Unfold the dual-graph so that no edges cross. Figure 2.3-2(b)

4) Construct a corresponding floor plan by drawing in wall segments as shown. Figure 2.3-2(c)

5) Adjust the shapes, dimensions, and locations of rooms to comply with specified requirements. Figure 2.3-2(d)

Unfortunately, however, most of these steps involve quite difficult computational tasks. This turns out not to be the case; and rather complex procedures, based upon alternative characterizations of planarity, are required for efficient planarity detection (Whitney 1932,1933, Maclane 1937, Deo 1974) . Due to the computational difficulties involved. this procedure has so far only been showen to be practical for small buildings, of the order of ten or fewer rooms.

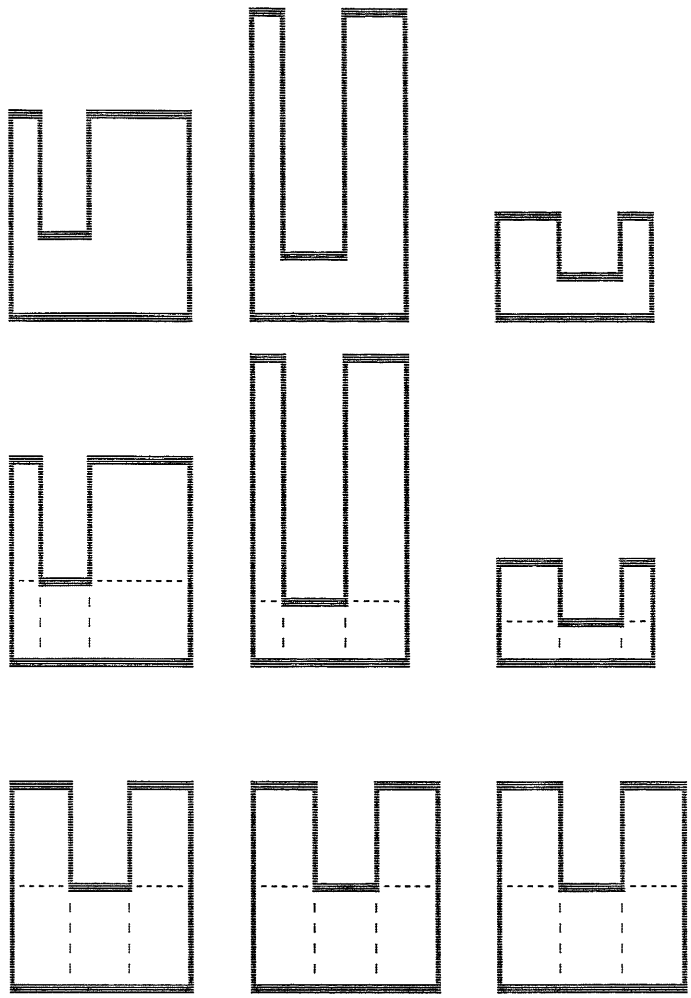


Fig 2.1-1
Dimension representations of similar rectilinear shapes. Several U-shape objects. Imposition of minimun rectangular gratings. Transformation to dimensionless representations

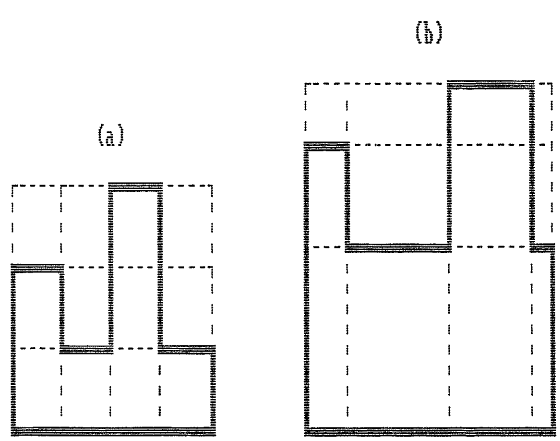
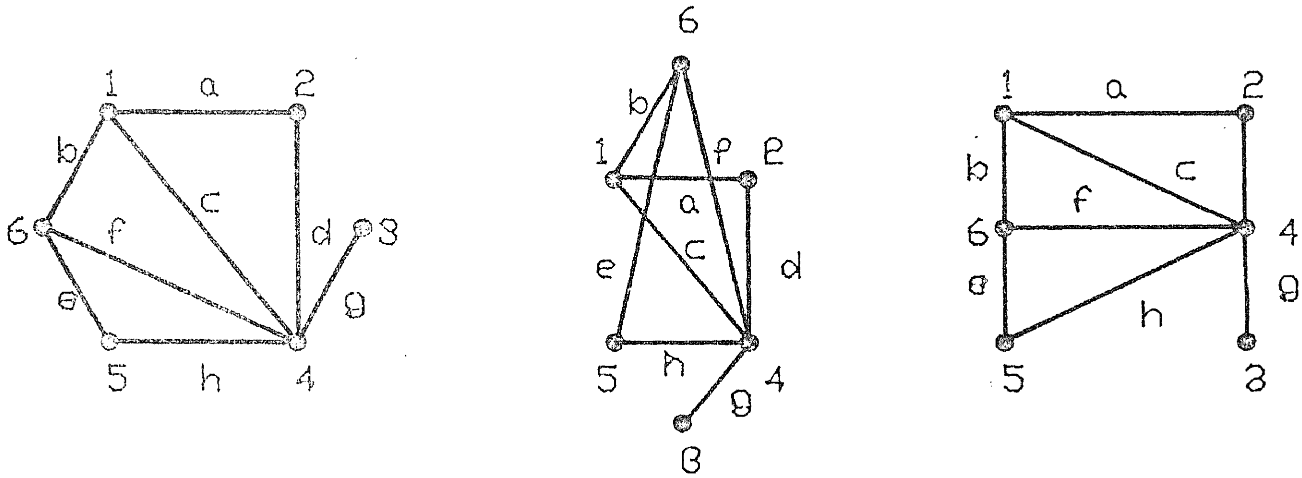
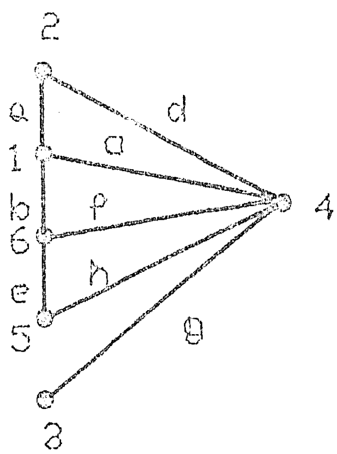


Fig.2.1-2
Description of a shape by a dimensionless repersentation plus dimensioning vectors (a) Dimensionless repersentation





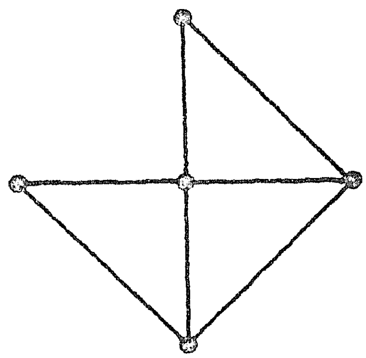
	a	b	c	d	e	f	g	h
1	1	1	1	0	0	0	0	0
2	1	0	0	1	0	0	0	0
3	0	0	0	0	0	0	1	0
4	0	0	1	1	0	1	1	1
5	0	0	0	0	1	1	0	0
6	0	1	0	0	1	1	0	0

	1	2	3	4	5	6
1	0	1	0	1	0	1
2	1	0	0	1	0	0
3	0	0	0	1	0	0
4	1	1	1	0	1	1
5	0	0	0	1	0	1
6	1	0	0	1	1	0

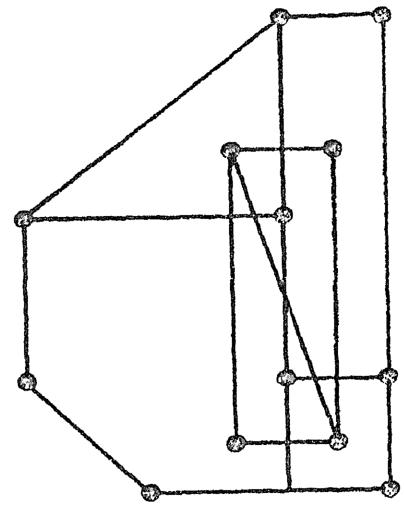
Figure 2.3-1. Several different geometric relizations of a graph. (e) Incidence matrix representation. (f) Adjacency matrix representation.

A B O D

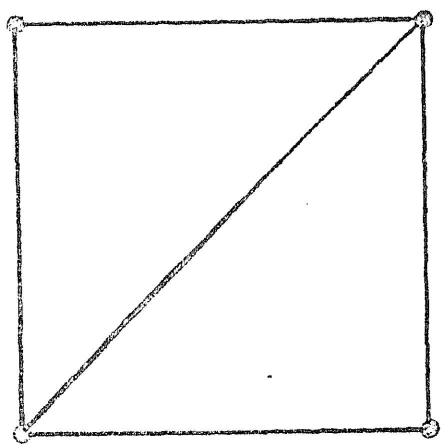
$$\begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix}$$



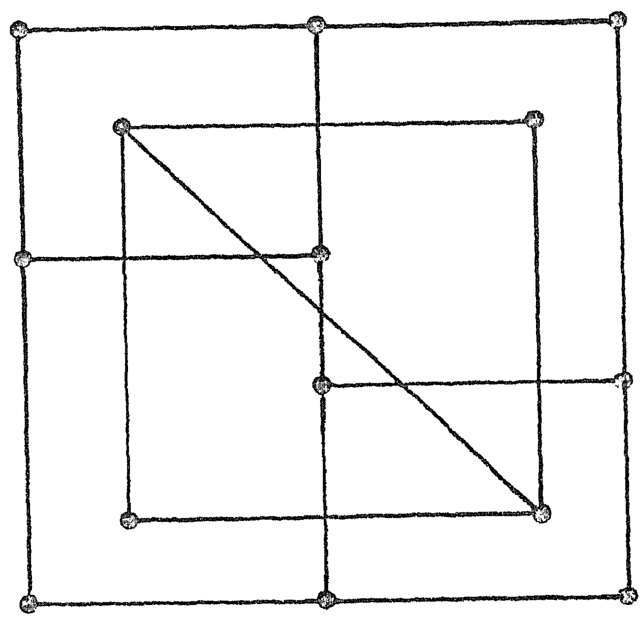
(a)



(d)



(b)



(c)

Figure 2.3-2
Procedure for constructing a floor plan corresponding to a specified adjacency requirements matrix.

(a) Adjacency requirements matrix and its corresponding graph.
(b) Graph unfolded in the plane.
(c) Dual constructed.
(d) Shapes and dimensions of rooms adjusted.

II. INFORMATION AND INVESTIGATION

Data refers to a kind of knowledge of a fixed form and may be obtained by means of collection; while the information does not exist in a fixed form, it is fluid and will become useful data only after it is investigated and processed as discussed earlier. In the progress of research, we have realized the development of various methods of investigation by controlling physical situations. Such methods are primarily based on observation and inquiring.

1. Preparations for Investigation and Analysis

Whenever we have advanced preparations prior to our investigation of problems associated with the physical conditions, we shall certainly have our investigation accomplished better and more effectively. To know our objects better, our preparations for the investigation should begin with our observations and then converge on the problem. We should select two or three methods which have been considered effective in defining our problem. Then we pick out the one having the greatest feasibility and try it.

Besides the methods we use and the scope of our investigation, we must also stress the methods we use for analysis. If one observes solely for his own interest and observes, or inquires about the physical situations of life and does not apply analytical methods to this, little of consequence will be achieved.

With the knowledge accumulated through our preparations for the investigations, we will be able to gain an insight into the
essence of the problem and the relation of cause and effect and existing among the events associated with it. Thus, we shall often be able to forecast the conclusions associated with the points of the problem. However, since such conclusions may not necessarily be per-

suasive, we should first establish hypothesis based on such forecasting and then proceed with our reviewing and verification until our formal investigation begins, or until we start to investigate the key points of the problem.

The task of initial problem structuring in design has been described as follows: the designer first tries to identify all the elements of this problem. These are the goals, requirements, constraints, or performance specifications which his final solution must satisfy. At this point, the situation seems to be in complete chaos. The designer's dilemma is not the lack of information; in fact, he is usually overwhelmed by more knowledge and more data than he can handle. Instead, his difficulty is the lack of structure. His task is to somehow organize all this information into a precise and consistent problem description, and then break it down into manageable sub-problems. By the process of articulating the problem elements and then establishing their relational structure, the designer begins to replace chaos and uncertainty with content and order.

In the early 1960's Christopher Alexander developed a systematic computer-based procedure for performing this initial problem structuring task. Since then, Milne (1971) has continued to improve and refine the technique, and has developed it into an efficient and reliable method that has been widely utilized not only in architecture, but also in planning and administrative
science applications. The version of the technique which is described here is that developed by Milne.

The first step in the process is to write down a list of elementary problem statements. The next step is to define an interaction matrix, which shows interrelations between elements. The element list and the interaction matrix are then input to a computer program which employs an algorithm based upon graph theory to sort the elements into clusters. Each cluster consists of a subset of elements that are tightly interrelated by many links between themselves, but which have relatively few links to elements outside the cluster. Thus each cluster defines a relatively discrete sub problem. In the algorithm employed by Milne, the clusters may overlap to a certain extent. Some other algorithms produce completely non-overlapping cluster.

The algorithm next proceeds to hierarchically recombine small cluster to form successively larger clusters, printing out each new cluster generated, and eventually produces one large cluster which consists of all elements and represents the total problem.

In effect, the algorithm structures the set of initial elementary problem statements into a hierarchy of increasingly larger sub-problems. To show the hierarchical pattern of interrelations between sub-problems, it is convenient to produce a dendrogram.

In common with other techniques for revealing structure in data, for example, factor analysis and multidimensional scaling, the theoretical foundations for this type of analysis are more obscure and controversial than one would like. But there seems to be little doubt that, carefully applied, it can produce very useful results in practical situations. The principal disadvantage is that it is a very laborious and somewhat error-prone technique. It usually takes quite a lot of effort to develop an initial list of elements. Even worse, the number of potential interactions between elements grows as an exponent of the number of elements, and filling out the interaction matrix can take an excessive amount of time. Many important criteria may only be revealed by actually engaging in the search for a solution. Because of these restrictions, problem structuring methods are most appropriate for use in the initial stages of new and complex problems, where previous experience and existing prototype solutions provide little guidance.

2. Investigation of Facts and Information Available

The investigation of real conditions refers to the examples of similar existing facilities to supply ideas about conditions of living and conditions of service. Besides the investigation of the existing facilities, and users or management, the investigators will also have to personally participate. In addition, we must explore the present condition of systems and make use of the statistical data available on a selective basis according to our need and our purpose. Sometimes we may not find statistical data compatible with our purpose, then we shall have to explore them further in other avenues.

Recently computers are available to collate all factors related to a certain phenomenon by means of factor analysis and have them arranged into arrayed for our analysis of their interrelations. We have been able to achieve remarkable progress in the exploration and research into life structure. For the relative analysis of life structure, however, we should not rely on the static analysis of collated relative factors only. It is important for us to uncover the relation between cause and effect in life in life phenomena, so that we can gain in insight into its essence.

Feasibility analysis is becoming an increasingly crucial step in design and construction processes. It is at the feasibility analysis stage that the basic parameters of a project are set, and critical design constraints and objectives are established. Traditionally, feasibility analysis for building construction projects has relied primarily upon experience and informed professional judgment. Today, some very powerful computer-based mathematical techniques are available to assist. They do not reduce or eliminate the need for accurate basic to assist. They do not reduce or eliminate the need for accurate basic information and sound judgment as the basis for decision making, but they do enable the implications of available information to be rigorously and exhaustively explored.

The first step in the mathematical formulation of a feasibility analysis problem for computer solution is to list the key decision variables, parameters and constraints. Next, a mathematical model which interrelates the key decision variables, parameters, and constraints is constructed. The model is then manipulated for the purposes of enumeration of ranges of feasible solutions, evaluation of options against various criteria, optimization of an objective, or sensitivity analysis to explore the effects of variations in the values of key variables, parameters, and constraints.

Although some simple feasibility analysis calculations can be carried out by hand or with the aid of an electronic calculator, larger and more complex analyses necessitate use of a computer. The following subsections describe some important computer-based mathematical feasibility analysis techniques, and discuss examples of their use.

2.1 Morphological Analysis

Undoubtedly the simplest and most general of all useful feasibility analysis techniques is morphological analysis. To perform are first listed, together with the range of potential alternatives for each one. For example, the variable and alternatives for a house might be:

Variable	Alternatives
Wall construction	1. Brick
	2. Concrete block
	4. Timber stud framing
Floor construction	1. Concrete slab
	2. Timber
Roofing	1. Bituminous
	2. Metal

3. Terra-cotta

An exhaustive catalogue of potential solutions (not all of which are necessarily feasible) can then be generated by systematically enumerating and listing combinations of alternatives, as shown in Figure 2.1-1. Of course in this simple example, the enumeration could easily be performed by hand. But where there are more variables and alternatives, it becomes worthwhile to use a computer to generate and print out the catalogue.

The purpose of morphological analysis is to insure that the complete range of potential solutions is brought explicitly to the decision maker's attention, and to suggest possibilities that might otherwise be overlooked. Although it may appear rather simpleminded, and it is obviously very limited by the rapid growth in combinatorial possibilities as the numbers of variables and alternatives increase, it has been widely and successfully employed in a number of fields.

2.2 Method of Incompatibility

A great disadvantage of the morphological analysis technique is that it often generates large numbers of nonsensical or infeasible combinations. It would be a considerable improvement if these could be automatically eliminated. The method of incompatibility matrices provides a powerful and efficient way of doing this.

The basic assumption underlying the method of incompatibility matrices is that most nonsensical or infeasible solutions arise because of the existence of pairwise incompatibilities between alternatives. A pairwise incompatibility exists if an alternative a of variable A cannot be combined with an alternative b of variable B.

Pairwise incompatibilities between alternatives are most conveniently described by means of binary incompatibility matrices. The rows of an incompatibility matrix the alternatives for one variable, and the columns the alternatives for another. An entry of 1 in the matrix indicates that the corresponding alternatives are compatible, and an entry of 0 indicates that they are not. Where there are n variables, then $(n - n)/2$ matrices are required to defineall compatibilities and incompatibilities.

When incompatibility matrices have been defined, the total number of feasible solutions can be found by a simple matrix manipulation procedure (Harary,1965). Of more interest in this context, however, is the possibility of using incompatibility matrices to implicitly eliminate subsets of potential solutions from consideration. The type of procedure shown in Figure 2.2-1(a)can be used.

The state-action graph for this problem, with the portion of the graph actually searched by the procedure, is shown in Figure 2.2-1(b). It can be seen that search along any branch is terminated as soon as a vertex representing an incompatible combination of alternative is encountered, since all combinations spring from that vertex must contain the same incompatible pair. Thus feasible solutions are exhaustively enumerated without searching the entire tree. Where the tree is large, the amount of searching that can be eliminated in this way may be very significant.

2.3 Optimization by Enumeration

Morphological analysis makes explicit the complete range of potential solutions. The method of incompatibility matrices introduces feasibility considerations. Further refinements are to incorporate resource constraints, and to utilize a measure of performance, which is a function of the design variables and the parameters of the problem, and which can be employed to rank feasible solutions in order of desirability.

For example, the procedure shown in Figure 2.3-1(a) might be employed to perform a simple analysis of a high rise office building. The building is assumed to be rectangular in plan, and the design variables are plan length, plan width, and number of stories. The permissible ranges of variation are:

LENGTH 80' - 100'

WIDTH 80' - 120'

STORIES 15 - 50

A four-foot square plan module is to employed. Thus there are 5 x 10 x 35 = 1,750 potential solutions. There is a constraint on total floor area, and the objective is to maximize percentage return on investment, computed as a function of floor area. The procedure essentially consists of three nested loops. If desired, it can be modified to produce only the optimum solution as shown in Figure 2.3-1(b).

Quite a few feasibility analysis programs of this type have been implemented and put into use. Perhaps the best known program of this type is the Building Optimization Program (BOP) developed and employed by the firm of Skidmore, Owings and Merrill (Harper,1968). The BOP model assumes that the design for an urban high-rise office building can be reasonable well described in terms of values for a small number of key design variables; building width, length, floor area per story, number of floors, and service-core dimensions. Permissible ranges of values and increments for these variables are specified, and the program enumerates all possible combinations of values, computers estimated construction cost and other measures for each combination, and prints out the results.

All thought the enumeration technique for finding an optimum solution is very general and flexible in its application, and produces excellent results, it obviously becomes impractical where the number of potential solutions is very large. In this case, more powerful optimization techniques must be employed. One of the best known and most widely employed such techniques is linear programming.

General purpose linear programming software packages suitable for performing this type of analysis are widely available, and some such packages are implemented at most university and commercial computer centers. In addition to linear programming packages, many other types of optimization programs are also available.

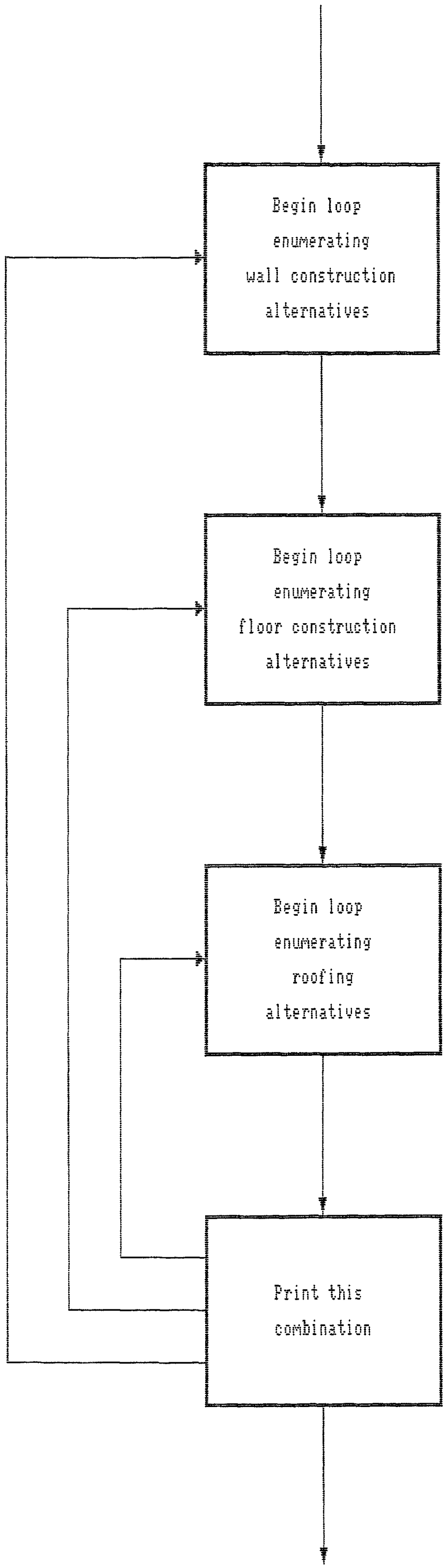


Fig. 2.1-1
A procedure for exhaustive enumeration
of potential house types.

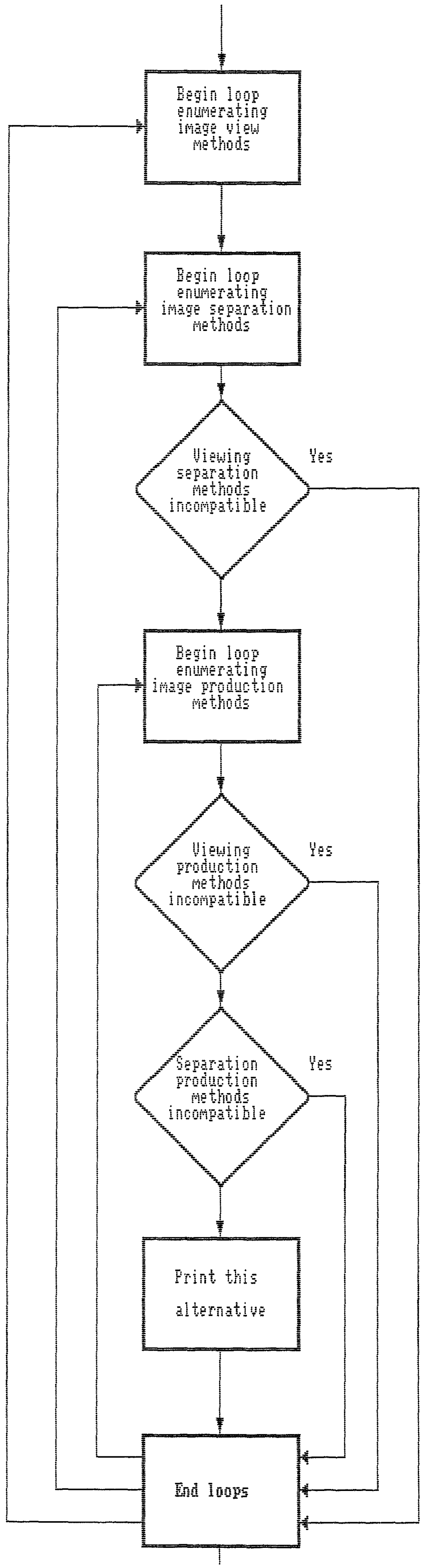


Fig. 2.2-1 (a)
Flow diagram of the enumeration procedure.

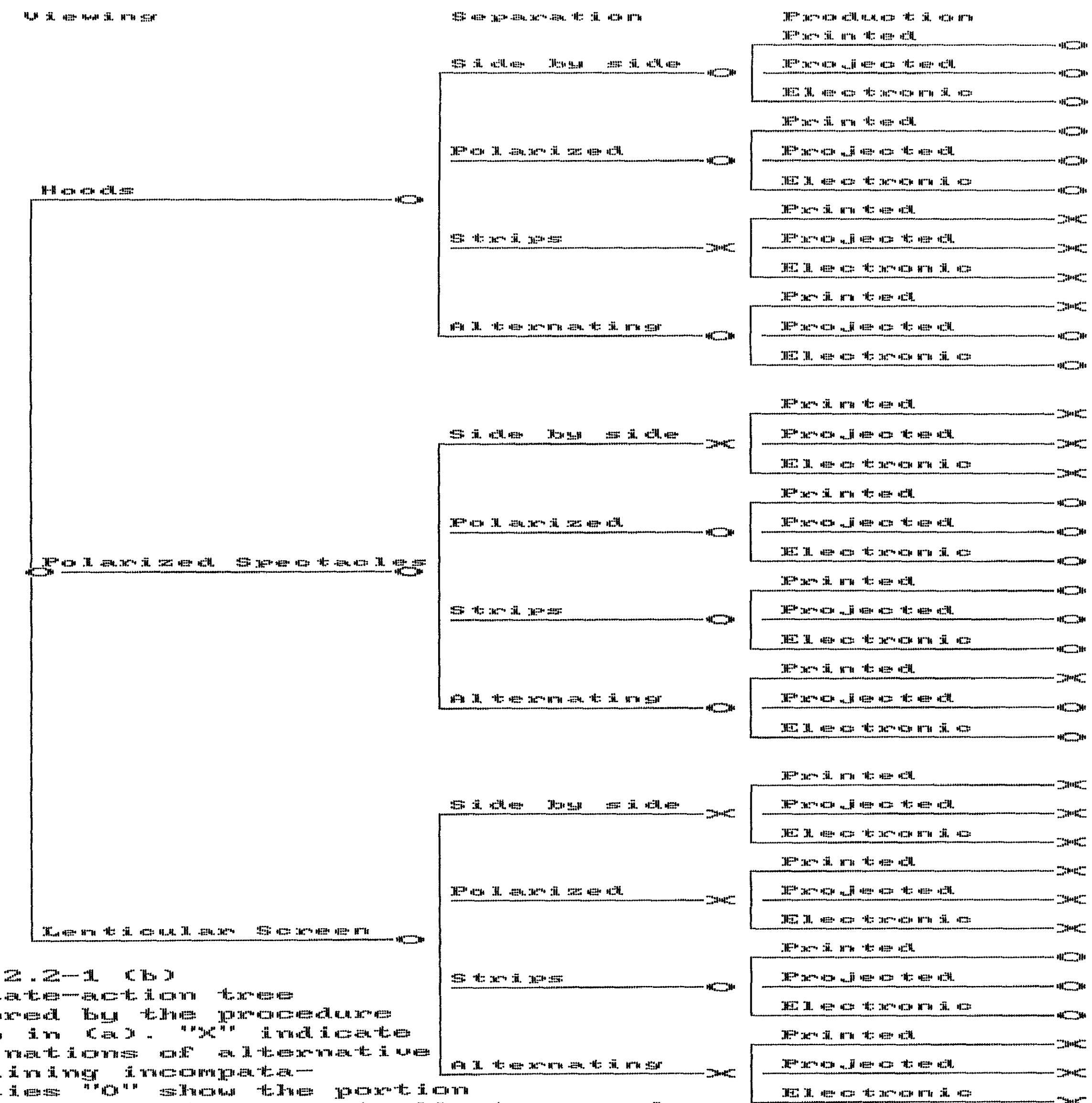


Fig. 2.2-1 (b)
(b)State-action tree explored by the procedure shown in (a). "X" indicate combinations of alternative containing incompatabilities "O" show the portion of the tree which is actually traversed.

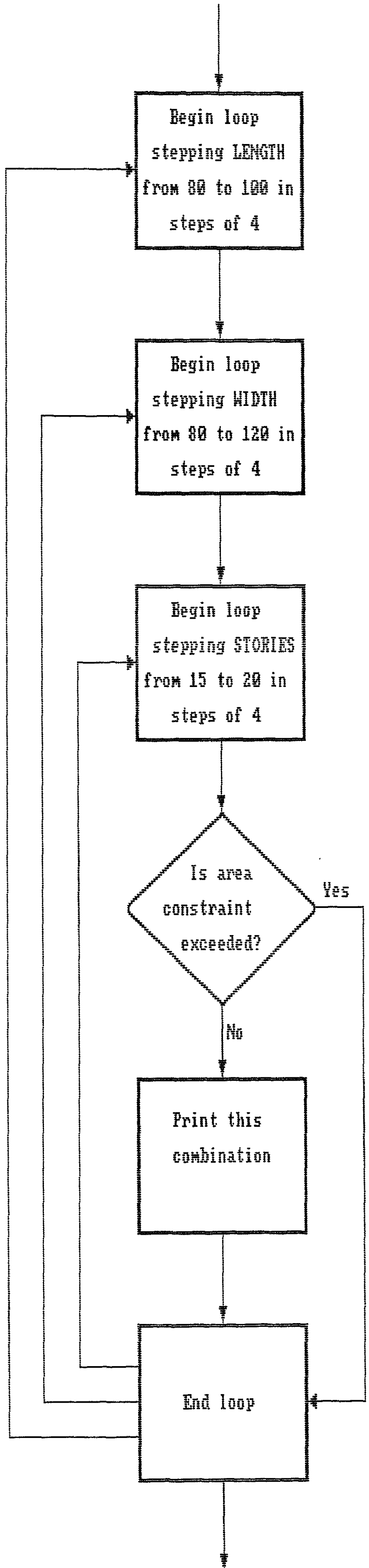


Fig. 2.3-1 (a)
Procedure for enumerating all
feasible solutions.

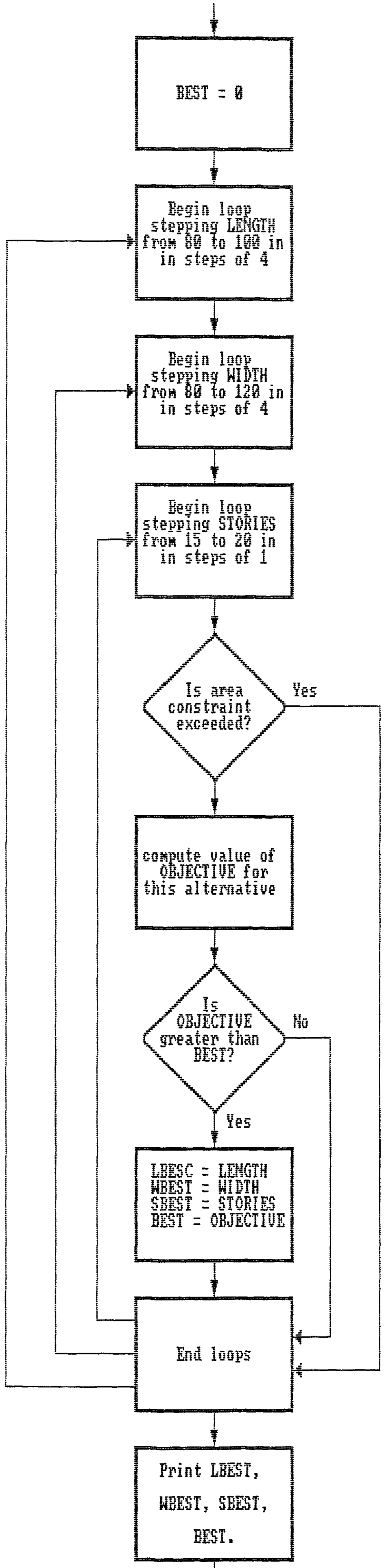


Fig. 2.3-1 (b) procedure for producing only the optimum solution.

3. Investigation By Inquiry

Inquiry of the users of a building will be an effective means for understanding the relation between life and space and can often be used simultaneously. There are two types of enquiry by nature:

1) For the life behaviors which we cannot understand by means of immediate observation.

2) To recognize the ideas, demands or consciousness of users of a building.

By way of enquiries, we shall be able to understand the attitudes and opinion of the people being enquired, their wishes, complaints, different ideas with reasons possibilities for compromise, and further analyze them for deserved solutions. Then we shall be able to clarify their requirements.

We may often have limitations of place, time, ect. so far as the observation of life behavior is concerned. So there may be cases in which we may not be able to conduct such investigation at all. In addition, there may be some parts of them which cannot be understood through observations and can be understood only by means of enquiries. Ideas and attitudes can only be ascertained by personal enquiry, or through interview.

There are two types of inquiry:

1) Direct and person-to-person interview.

2) Enquiry by means of questioner.

Sometimes both can be employed in combination.

3.1 Enquiry

In our interview we shall be allowed to change the contents of our inquiry or ask questions in response to the reaction of the other party. We may also encourage him to raise questions on matters he may be concerned about in addition to his being questioned as scheduled. However, this will require additional manpower and time allowance.

Meanwhile the answer by the other party may vary with the investigator, his feelings. his inductive powers, and the limitation of place or other aspects which may be imposed upon the other party. Then the answers which we may acquire may not necessarily be as complete as desired.

A successful visit depends upon the ability of the planner interviewer. He must first have questions prepared and make the arrangements and preparation in advance before he can ask the right questions. In order to make the talk go on naturally, a recorder may be used as an aid without having to take notes during their talk. However, a recorder may often embarrass the people being visited and distract the inquirer as he has to take care of the operation of the recorder. A natural talk should nerve be interrupted in any way. Our talks are supposed to go on in a peaceful and calm atmosphere just as we chat with our old friends at home. One should never make those being visited feel ill at case.

The skill required for a successful interview may vary with the number of persons participating in the interview. Generally speaking, there are individual interview and group interview.

(1) Individual interview:

In an individual interview, the interviewer is usually one person. However, sometimes he may ask one or two to join him in the interview. If the interviewer conducts the interviewing alone, he had better use a recorder as an aid. On the other hand, he may take notes for key points. This will make the interviews believe that his ideas or views are receiving due attention. If the interviewing party consists of two persons, one of them to make records, the interviewer will be made more free and more natural so that he may attain more information from the interviewee.

(2) Group interview:

The object of a group interview may be a unit or an organization headed by a leader. Most questions will probably be answered by him, and answers may not necessarily be of representative. The interview should pay due attention to reactions or propositions by the other members of the group. If the group consists of a number of organizations or sub-groups, the member of each component may have different ideas or points of view. Therefore, whenever the decision-maker is about to make a decision, they often try to influence or press him for changes favorable to their position. At that point, the interviewer should suggest or hint them in one way or another to re-consider the question.

Prior to our interview, we should direct our attention to the following requirements:

1) TO recognize the decision-maker.

2) TO make our intention know and tell the head of the group or organization the information we need:what; and when.

3) TO explain the ways in which we are trying to explore problems.

4) To explain what data we have available on hand (obtained from individual interview of records on hand).

Generally speaking, the head of the group or organization is the man who makes decisions. The executives of the organization may not necessarily by representative. The executives of the organization may not necessarily be the ones making decisions. In order to facilitate us in conducting our planning operation, authorization by the decision-maker will be essential.

Prior to interview we had better make an introduction,briefly declaring the purpose and direction to our interview in order to guide all participants to concentrate on our main subject.

During the interview, our attention should be directed at the following;

1) At recognizing our psychological goal.

2) At raising questions immediately pertinent out goal.
3)At summarizing and making conclusions or guiding the participants and winding up per talk at a given time.
4)At continuing to return to our main subject until our problems are cleared.
5) At having all speech recorded and insuring nothing is omitted in order to avoid the serious which might be caused by such omission.
6) At preparing a checklist for interviewees, since the client or interviewees may not at one time necessarily be able to think of all problems concerned.

What is said by the interviewees may not necessarily be what they would says their own in mind. This may be made c;ear by raising questions to test whether they have a uniform directive goal and whether they conform in their view of facts, concepts and requirements. The questions asked must be such that they will not fail in eliciting the answers and information associated with the goal established, so that we may establish conformity from the beginning till the ending.

The ideas of management may not necessarily be the same of users. Contrast between the ideas of those in the different standpoint may serve to clarify those ideas inherent to them which will provide a bases for discussing. The man in the street, who may be less adept at decisions, may nevertheless trigger ideas relevant to proposal or problems, which may lead to solutions.

Within a complicated decision-making organization, conflicts can hardly be avoided. The interviewer must be able to recognize and clarify the points which are a key to problems and sort out required information by choice and compromising.

In view of the fact that a multitude of people are involved in the architectural requirements, we must explore and analyze problems in an orderly before we can reach logical conclusions which can withstand future examination. We shall have to classify or correlate the desires and demands of those people and their ideas and attitudes so as to establish effective architectural information.

We must record all concepts and ideas generated from the discussion with the interviewees and have them compiled in an orderly manner, regardless how simple they are and regardless whether they are just expectation or are regarded as practical. As far as possible, we must collate them under proper titles or subtitles, or classify them under headings such as personnel. or charts for a presentation which will be attractive and appealing to the readers.

3.2 Questionaires

Questionaires are design for collection and digestion of the information and data obtained from a multitude of users. The information and data collected by way of questionaires include those relating to personnel, materials, space and activities. First-hand information and data contained in questionaires can be collate and analyzed by computers.

Free from limitations such as distribution and collection, or geographical limitation, we shall be able to obtain a greater number of examples by means of questionaires delivered through the postal system. Meanwhile it is also free from the influenced personal feelings, the information received can be expected with steadiness. for the adoption of questionaires, usually we found out the focus of problems by means of personal inquiry which will be followed by questionaires investigation for further extensive inquiry in order to verify the reliability of information obtained from personal inquiry.

Since those answering the questionaires must read them before they answer them, no reliable outcome can be attained unless those answering questionaires have the ability to read. Thus such a method will not be suitable for the aged or children. When many questions are contained in the questionaire, too much time will be needed for the statistics. Therefor, the contents of questionaire must be limited and their wording must be simple and brief, clear and precise. Otherwise, the questions may be answered in an entirely wrong way.

3.3 Space Needs Analysis

Space needs analysis basically taking data about people, activities, and equipment to be accommodated in a project. and transform it into a set of precise spatial requirements. The initial data might consist of the results of interviews with clients and users, questionnaire survey data, observational studies, standard guidelines, or investigations of pervious similar projects.

Computer based techniques are beginning to revolutionize inquiry of space needs, since they make possible the efficient manipulation of large volumes of data, and the performance of much more extensive and sophisticated analyses than had been possible in the past. In the following, some typical examples are described:

(1) Simple empirical models

The traditional approach to estimation of accommodation requirements for a facility is to utilize simple empirical formulate, based upon previous experience. For example, a typical formula for a number of university lecture room seats which should be provided per head of enrollment is

$$\frac{h * f * \sigma}{\lambda}$$

where

h = average number of hours of teaching per student per week

f = frequency factor

σ = occupancy factor

λ = length of teaching week (hours)

The frequency factor is an empirically derived estimate of the average percentage of the teaching week that a lecture room will be in use, and the occupancy factor is an estimate of the average percentage of available seats actually occupied when a lecture room is in use. It is very straightforward to write simple computer programs which project space needs based upon this type of formula.

Although simple empirical models can be very useful, they have certain limitations. Projection from prior experience carries with it the risk that mistakes of the past will be perpetuated. Furthermore, there is a possibility that new conditions and policies may, unknown to the analyst, invalidate some of the basic assumptions underlying the projection. Perhaps most important of all, this type of model provides no firm basis for optimization of space usage, or for examining the sensitivity of space and facility needs to variations in activity patterns and space use policy.

(2) Timetable models

An alternative approach, which can potentially overcome many of these difficulties, is to develop a detailed computer-based model of the activity system that the facility must accommodate. The parameters of the model can then be varied to examine the effects of different assumptions and policies.

A typical application of this latter approach is in school design. If a proposed weekly timetable for a given subject area is known, then a diagram plotting class size against number of periods taught per week can be plotted as shown in Fig.3.3-1. A second stepped line, representing a proposed distribution of classrooms of different sizes, can then be plotted on the same axes as illustrated. The difference between the two lines represents unused capacity, so the better the match of the two lines the closer the fit between accommodation need and accommodation provision. This graphic method is readily translated into a computer algorithm (Markus 1972,Th'ng and Davies 1975) which can be employed to explore the effects of different enrollment patterns and time table and classroom size policies, and to find optimum distribution of classroom sizes implied by particular activity patterns.

(3) Queuing models

In facilities such as supermarkets, department stores, cafeterias, etc., use of simple queuing models may be an appropriate way to project space needs. A study by Desai and Yanouzas (1970) of an Athens department store is a good example. Data was collected on customer service times at cashier and wrapping stations, and a regression analysis was conducted to find the relation between number of active cashiers or wrappers and service time. The resultant model could be employed to find the number of stations necessitated by setting a waiting time criterion at any particular level.

(4) Discrete Event Digital Simulation Models

In more complex service facilities, such as air terminals, clinics, ect., it may not be possible to employ simple analytical queuing models. In this case, discrete event digital simulation models (Fishman 1973) may be utilized. A discrete event digital simulation model describes processes in terms of states and events. For example, the states and events experienced by a customer approaching a service window might be as follows:

Enter queue	(Event)
Wait in queue	(State)
Begin business at service window	(Event)
Conduct business	(State)

Depart (Event)

It can be seen that occurrence of an event results in a change of state. A discrete event simulation model is a computer program whose execution simulates the occurrence of events (and hence changes of states) in a system over time, and which prints out statistical data on states of the system that are of interest, for example, queue lengths. NOrmally, the facility to be simulated is represented as a flow network, for example, as shown in Fig.3.3-1.

Discrete event digital simulation models can be written using general purpose programming languages, but it is generally much easier to employ one of the many special simulation languages that have been developed. Among the best know and most widely used are GPSS (Schriber 1972), SIMSCRIPTT (Kiviat, Villaneuva, and Markowitz 1969), and SIMULA (Dahl and Nygaard 1966).

An excellent example of a discrete event digital simulation model which can be used to investigate accommodation needs is a multi story industrial parking structure model developed by Broad (1969). The model is programmed in GPSS, and its scope includes movement of people to the structure, vertical transportation within the structure, exit of cars from the structure, and entry of cars to the surrounding street system. The primary output is a graph . Design features such as total capacity of the parking structure, numbers and locations of exit ramps, stairs, elevators, ect., can be varied, and the effects on transit times studied. In this way, a configuration which adequately meets transit time performance standards can be determined.

Extensive studies of the design and utilization of hospital facilities, conducted using SIMSCRIPT discrete event digital simulation models, have been carries out at Yale University (Fetter and Thompson 1965, Thompson and Goldin 1975). A maternity suite model, for example, allows exploration of relations between the capacity of the skillet, patterns of demand, operating polices, and levels of service provided. A model of a surgical pavilion allows experimentation with different configurations of special and general purpose operating rooms, and with different scheduling policies for their use. A model of an outpatient clinic focuses primarily on patient waiting time, and a ward model is defined for analysis of the economics of different mixes of private and multi bed rooms under various conditions.

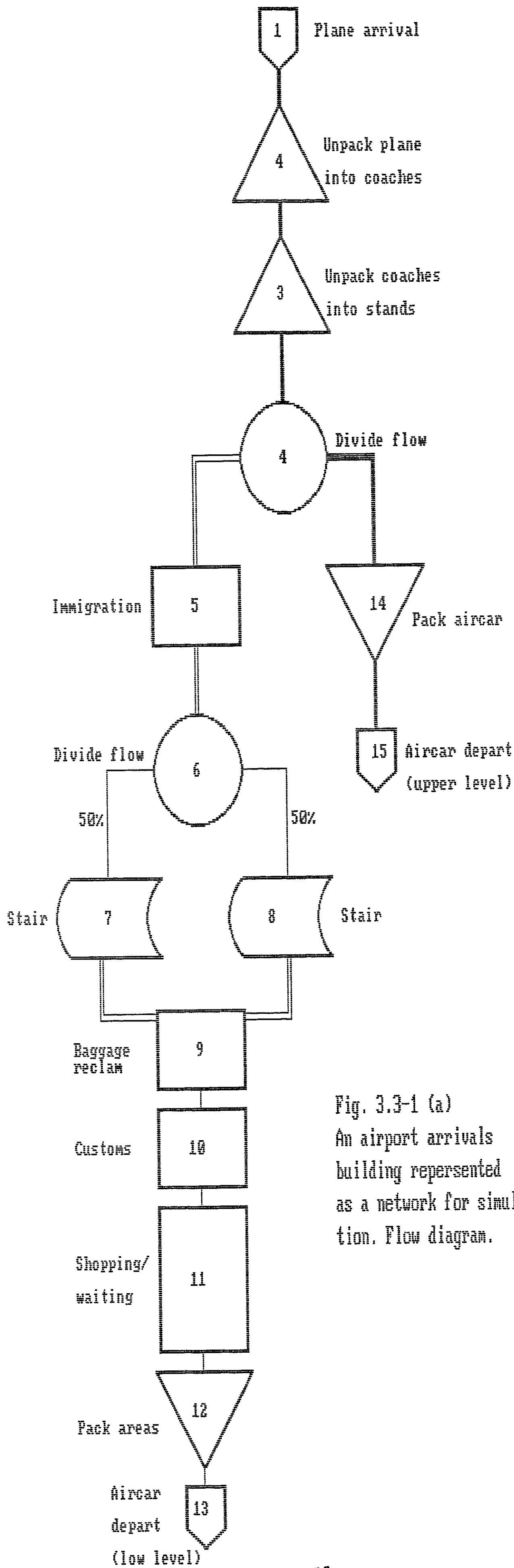


Fig. 3.3-1 (a)
An airport arrivals
building repersented
as a network for simula-
tion. Flow diagram.

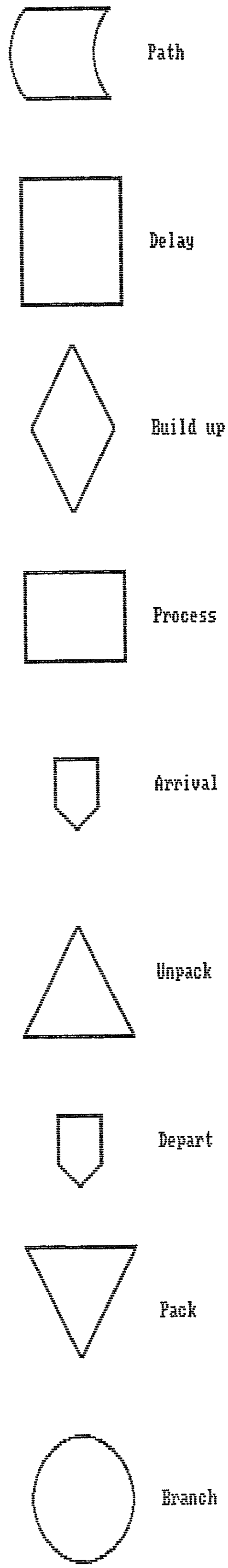


Fig. 3.3-1 (b)
Key to flow diagram symbols

III. CHANGE OF INFORMATION

1. Factors of Changes

1.1 Environmental Factors

Environmental comfort criteria play a critical role in almost all architectural design problems. Numbers of different quantitative measures of environmental performance are employed, but the following are among the most common :

Climatic	Air temperature
	Relative humidity
	Rate of air movement
	Period of insulation
Illumination	Daylight factor
	Artificial lighting illumination level
	Glare index
Acoustic	Noise level
	Reverberation time
	Speech privacy

Analyses with respect to these criteria may be performed for two purpose, either to predict capital and operating costs of keeping criteria within specified limits of acceptability, or to determine whether a specific proposed design can meet criteria of environmental acceptability. In general, well-known existing models may be employed in performing most of these analyses.

The following paragraphs briefly discuss some typical examples of automated environmental analysis techniques.

(1). Thermal Analysis

The types of analyses of thermal performance which might be carried out in a building design process vary somewhat according to building type, design objectives, and the sequence of decision making that is followed. Gupta, Spencer and Muncey (1971) have categorized the main types of analyses into four groups as follows :

Calculation of indoor air temperature in the absence of artificial cooling or heating.

Calculation of indoor air temperature when some heat is being removed or added but the indoor temperature is still variable.

Calculation of heat gains or losses and cooling or heating loads when the indoor air temperature is kept constant at a known value.

Calculation of heat gains or losses and cooling or heating loads when the indoor air temperature is variable and specified the case of temperature swing.

Most program is written in the interactive APL language, and computes the heat loss through the external shell of a building according to the well-known simplified formula :

$$\text{Heat loss} = \sum_{i=1}^{n} u_i a_i$$

where

n = number of surfaces

ui = average thermal transmissions of surface i

ai = area of surface i (sq. ft.)

Since the patterns of heat flow in a building are exceedingly complex, it has in the past been standard practice to employ simplified, approximate formulae like that above in performing thermal analyses (see the ASHRATE Handbook of Fundamentals (1974) and Maver (1971) for discussions of numerous standard formulae and procedures). But the widespread availability of computers has now made it feasible to go beyond these simple methods, and to utilize more sophisticated procedures that do not rely upon such restrictive assumptions (Fitzgerald 1971, Feinberg 1974). In addition, the availability of very detailed, machine-readable meteorological record tapes has made it practical to utilize more detailed and accurate weather data. Concurrently, the increasing concern with minimizing energy cost in buildings has made more accurate and reliable thermal evaluation procedures a pressing necessity. In the late 1960's, ASHRAE demands of buildings, and these were implemented in the form of a very large and complex FORTRAN program for use in design of U. S. postal buildings (Lokmanhekim 1971). In addition, a number of consulting firms and research groups have developed comparable proprietary programs. Among these are AXCESS (Edison Electric Institute, New York), E-CUBE (GATE, San Antonio, Texas), MACE (McDonnell Douglas Automation company, St. Louis, Missouri), and TRACE (Trane Company, La Crosse, Wisconsin).

Integrated computer-aided design systems quite often incorporate thermal analysis facilities, which allowing the user to rapidly evaluate the thermal characteristics of alternative envelope configurations at any early stage in the design process.

(2). Insulation and shadow pattern analysis

The calculation of sunlight and shadow pattern on, in, and around buildings is a straightforward but very laborious task if performed by hand. The following trigonometric formula can be used to compute the sun's azimuth and altitude as functions of location on the face of the earth, time of year, and time of day :

$$\sin Z = \sin t . \cos d . \sec H$$

$$\sin H = \sin L . \sin d . + \cos L . \cos d . \cos t$$

where

Z = sun's azimuth

H = sun's altitude

L = latitude

d = declination of the sun (i.e., the angle formed between a connecting the center of the earth with the center of the sun and the plane of the earth's equator)

t = time of day (expressed in degrees)

From a knowledge of sun location, the position of the shadow projected by any point onto any planar surface can be calculated by means of elementary trigonometry. The outline of the shadow cast by a polygonal object onto a planner surface can be generated by connecting the projected locations of its vertices. (See Givoni 1976 for a concise and complete summary of the principles of sun path and shadow plotting.) Since most high level, general purpose programming languages incorporate trigonometric functions, it is a simple matter to write a computing the shadow cast by a building onto the ground plane, or shadow cast on simple elevations.

A more complicated shadow-casting problem, of considerable architectural importance, is to compute the outline of the patch of sunlight cast through a window into a room. The difficulty arises out of the quite complex conditions generated by overlapping of shadows cast by the wall surface, window reveals, and horizontal and vertical sun shading devices. illustrates the variety of different conditions which can result from overlapping of shadows cast on a rectangular window by overhangs and side fins . Tseng-Yao Sun (1968) has published formulae which may be employed to compute window insulation.

The general problem of computing cast shadows in three-dimensional built environments is quite complex and difficult. It is essentially identical with the problem of producing a perspective projection of a complex scene with hidden surfaces removed. The approach is to first compute a perspective projection, taken from a view-point determined by the sun's altitude and azimuth and at a distance of infinity, then to use a hidden surface removal algorithm to determine the surfaces "seen" by the sun. Those not "seen" are in shadow.

For any shadow pattern computations, whether of the simplest or the most complex type, a full description of relevant building and site geometry is necessary. If this description must be specially input, not be justified by the results obtained. But if the information can be automatically extracted from a project data base, then the analysis can be produced very rapidly and cheaply.

(3) Natural and artificial lighting analysis

For automated computation of daylight factors, the basic formulae are the natural illumination at a point in a room will depend upon the area of the sky vault that is subtended at the point by the location of the reference point. Furthermore, the illumination contributed by each small element of a large light source, such as a window, varies with the position of the element. Additional complications are that the luminous of a patch of sky varies with its altitude, and that the light transmitted through glazing varies with the angle of incidence. To evaluate the sky component of illumination at a point we must, in effect, sum the contributions of the patches of sky which lie within the area subtended by the window at that point. Finally, the sky component must be adjusted by addition of reflected components. The result may be expressed either in some units of illumination or as a daylight factor. The computations involved are straightforward, but extremely tedious to perform by hand.

A similar technique can be employed to plot levels of artificial illumination. The illumination due to a point source at a specified locations given by:

$$\text{illumination} = \frac{I\cos\theta}{D^2}$$

where

I = intensity of the light source in candelas in the direction of the point

θ = angle between the normal at the point and the direction of the light source

D = distance of the point from the source

(4) Acoustical analysis

Reverberation time for a rectangular room is computed according to the following formula:

Reverberation time = 0.49V/a

where

V = volume of room in cubic feet

a = total number of absorption units (Sabins)

The total number of absorption units is given by:

$$\text{Sabins} = \sum_{i=1}^{n} S_i C_i$$

where

n = number of different surface materials used in the room

Si = total surface area (square feet) of material i

Ci = absorption coefficient of material i at the specified frequency.

In the example, absorption coefficient are assumed as follows:

	Floor	Walls	ceiling
125 herts	0.05	0.02	0.2
500 herts	0.05	0.02	0.65
2,000 herts	0.1	0.04	0.75

The room under consideration is assumed to be square, with a ceiling height of 10 feet, and the output is a plot of reverberation times against side dimensions. The graphing facilities of the APL language were employed to produce the results shown.

1.2 Economic factor :

Some production activities and economic activities may bring about seasonal changes, such as the cropping period for fresh foods. For example, when sugar cane matures and the sugar refinery starts producing sugar, the expected effects on the communities nearby will involve traffic and personnel use and economic change etc.

Seasonal changes may be expressed by indexes:

Seasonal Index for Month i = number of users in month i / number o f users in a year * 12000

Generally speaking , the occupation rate of a recreation center is higher in Spring than in Autumn. The reasons why it is so may be that in Winter you may feel closed in while you may feel being liberated in the Spring season and thereafter, most schools will have vacations in the Spring.

The familiar architect's building costing manuals are essentially unit price data files, which are updated at regular intervals. Standard data base management facilities may be employed for maintaining, updating, interrogating, and generating reports from these files as required.

The different types of materials, labor, and equipment implied in building are extremely numerous, their costs are likely to change quite rapidly, and future costs are likely to be difficult to predict. Thus the construction and updating of an accurate cost information file can be a formidable undertaking. A very interesting solution to this problem was employed in the West Sussex County Councils integrated computer-aided design system.

West sussex employs a serial tendering procedure, under which the County Council architects prepare a schedule of unit items of construction work covering every item likely to be used in the County Council building
program over a given period. Contractors are then invited to price these items on the basis that they would be given a guaranteed quantity of work over the specified period, and the lowest tender is accepted based on a theoretical bill of quantities which proportion anticipated types of work. Different contractors are appointed in this way for different areas of the County and for different sized contracts. Files of establish contractors prices were held in the computer, ready to be applied to particular building design descriptions at any time. In principle, the West Sussex system allowed very effective cost control at the design stage, and allowed the time spent at the precontract phase of a project to be very substantially reduced.

If a traditional building procedure is employed, however, files of costs cannot be stabilized in this way. One approach to development of a cost file in the context of traditional bidding procedures is simply to abandon the attempt to predict actual construction costs, and to focus on comparison of the relative costs of different design alternatives. The cost file contains approximate values, which fairly accurately reflect comparative costs of different materials and techniques.

Some attempts have been made to develop large-scale computer based information systems which would begin to allow actual costs to be predicted more accurately in the context of a traditional bidding process.

2. Changes in hours-type characterized of movements

The use of facilities in the cities is mostly concentrated on a specific floor or section of hour in a building. In order to plan the facilities logically, we must forecast the pattern for the changes of numbers and the extent of concentration of the people arriving and the duration of their working hours.

2.1 Pattern of Changes of Hours

A regular pattern of changes of hours is determined mainly by the limitations imposed on service time. There are the following types :

(1) Changes in response to commutation hours

The pattern of commutation refers to the users who come to and go from the facilities on schedule, such as starting and finishing times for the schools, offices, and movie theaters, etc.

From the example of facilities designed for commuters, we know that the pattern of concentration looks mostly like the one shown in the Figure 2.1-1, Its theoretical form is generally in conformity with the index curve for normal distribution as shown in the Fig.2.2-2. Its theory can be demonstrated by the Information Theories. In short, the most desirable time for commuters would be a few minutes before the starting time. The farther it is from then, the greater the loss of time will be.

The major factors contributing to the concentration rate include seasons, weekdays, hours desired by individuals, means of communication desired by individuals, traffic conditions, regulations governing the penalty for those being late, and the nature of building or facilities, etc. Among them the seasons and weekdays may contribute less and the regulations governing the penalty may contribute more. The means for communications may vary with the distance to and from stations, number of buses or trains and their intervals. When we plan, the time recorder an locker facilities will also be major factors.

(2) Complex Change by Hours

This is the same pattern as indicated above but in a duplicated situation. Such a station is the typical one. The pattern concentration very often depends on local condition or the nature of other facilities located in the service area.

(3) Changes in Free Service Hours

The free service time means that the hours may enter or leave the facilities at any time within a given time frame without restriction to any specific hour(s) so that the service demand is satisfied within that given time frame, such activities might be to read in the library, to look at the collection in an art gallery, to do shopping in a shopping center, and to obtain service in a recreational facility, etc. All these are associated with facilities where the living behaviors are characterized by greater freedom. Their concentration rates are generally low. Though there may not be any noticeable concentration, we may also have gathering of a good number of people at certain hours. The service rates are usually low at the two extremes and higher at the mid-hours or the time closer to the off-time.

(4) Changes of People Staying at Different Hours

Of the number of users of facilities in cities, that of significance to planning would be the changes of number of clients at different hours rather than the total number or hours, such as in libraries, restaurants, shopping centaur, parks, art galleries, and exhibition halls, etc. When we plan for such facilities, the number of users we consider would be the maximum number of people there at given time.

As showing Figure 2.1-3, the changes of people arriving and staying at different hours are almost of the same pattern. The only difference is that there is a decrease at peak hours.

By means of the total number of users and the average staying time, we shall be able to forecasting the maximum number of persons to stay is to describe a curve showing the changes of different arrival by using the peak hour as the center and cutting it by the width of average staying time. The information essential to our forecasting includes the pattern for changes of different arrival and the average staying time. Both of them can be calculated by using the information obtained previously. For existing facilities, we can obtain more precise information about the number of staying persons from the data or information recorded for the visitors.

2.2 Circulation system efficiency

Quantitative analysis of circulation system performance has been rare in the past, since although the principles are fairly simple, the computations required are extensive. But in the context of a computer-aided design process, sophisticated circulation system performance analysis is quite feasible.

One common approach is to use the quadratic assignment objective as a formula for measuring overall circulation system efficiency in a proposed design. In this context, it may be stated

$$\text{Total circulation cost} = \sum_{i=1}^{n} \sum_{j=1}^{n} G_{ij} C_{ij}$$

where

n = total number of activity centers between which circulation takes place

Gij = distance between activity centers i and j

Cij = a measure of circulation cost per unit distance between i and j

As before, value for Cij must be obtained from a matrix of circulation flow data. But since a design is now defined, values for Gij may be measured directly and accurately by tracing appropriate routes through the plan. A map meter might be employed on drawings, and the values obtained entered numerically. Alternatively, routes might be traced out using light-pen or tablet. An even more highly automated approach is to employ a "shortest path" algorithm to find routes.

A pioneering comparative study of the efficiency of alternative hospital ward designs employing the quadratic formula was carried out at Yale-New Haven Hospital in 1958 (Thompson and Pelletier 1962, Thompson and Goldin 1975). An extensive survey of traffic flow in wards was conducted, and the resultant data employed to compute circulation costs for a variety of alternative ward designs (single corridor, double corridor, circular, etc.) To provide a basis for comparison of wards containing different numbers of beds, circulation costs were standardized as follows :

Circulation cost per bed = t / b

where

t = total circulation cost as computed by the quadratic formula

b = number of beds in the ward

It was found that circular and square design were generally most of festinate, and (surprisingly) that circulation cost per bed was not directly related to the number of beds in the ward.

A difficulty with use of the equadratic circulation cost formula in this way is that it gives an absolute value for circulation cost, rather than a value relative to some standard scale. Thus it is difficult to tell whether a particular layout is better or worse than might normally be expected. This can be overcome by first summing the circulation cost over all possible arrangements as follow :

$$S = \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{i=1}^{n} \sum_{j=1}^{n} G_{ij} C_{ij}$$

then dividing by n!, then number of possible arrangements, to produce a mean expected value :

mean expected value = S / n!

This provides a standard for comparison.

Another difficulty with the quadratic formula is that it gives no indication of which activity centers are most severely "mislocated" in the plan, thus contributing most to circulation cost. An indication of this can be obtained by assuming that, since in general the greater the value of Cij the closer together spatial elements i an dj should be; relationships of the following form should hold in a well-constructed layout

Gij Cij = K

where K is a constant determined by the size of the building and the magnitude of the different values for Cij. Thus a measure of the "appropriateness" of the distance between a pair of elements in a layout may be derived as follows :

Appropriateness = (Gij Cij - Gm Cm) / Gm Cm

where

Gm is the mean value for G

Cm is the mean value for C

A value of 0 indicates that i and j are probably appropriately located in relation to each other, a high positive value indicates that they are probably too far apart, and a high positive value indicates that they are probably too close together. This type of evaluation is performed by programs described by Willoughby, et al (1970) and Maver (1970).

A good example of a current circulation system performance evaluation program is incorporated in the Harness integrated computer-aided hospital design system (Shirley 1974). This system employees the quadratic circulation cost formula to give a basic measure of circulation efficiency for a proposed layout. Since Harness hospitals are part of an extensive ongoing hospital building program, this can be compared with corresponding values for other hospitals designed to the same brief. The following additional indices are also computed :

Individual department pair circulation costs = Gij Cij

Average circulation cost = T / N

where

T = total circulation cost

N = number of department pairs

Average distance walked per trip = T / C

where

C = total traffic (unweighted)

Coefficient of variation = $\sqrt{\sum_{i<j}(C_{ij} - C_A)^2/(N-1)C_A^2}$

where

Ca = average circulation cost.

Department pair circulation costs are displayed by means of a histogram (Fig.2.2-1(a)). In a satisfactory layout, the histogram should have a peak at the left hand side and no tail, while the coefficient of variation should be low. Values for Cij at plotted against Gij, as shown in Figure 2.2-1(b).In a good layout, the curve should slope downwards indicating a general increase relation between interdepartmental weighted traffic flow and distance. In addition to their uses in general evaluation of layouts, the results of these analyses are also employed as input to an automated elevator placement algorithm.

* Shortest path calculations

In analyzing the efficiency of circulation systems, it is often desirable to compute shortest circulation distances from a point in the plan to other points, rather than to measure these distances by hand. If the circulation system is represented as a network, then this can be accomplished by application of a shortest path algorithm. Numerous different shortest path algorithms have been developed (Pollack and Wiebenson 1960, and Dreyfus 1969), and many of them are suitable for this application. As an example, an efficient algorithm developed by Dijkstra (1959) will be described.

Figure 2.2-2(a) illustrates a simple circulation network,and Figure 2.2-2(b) its description by means of an adjacency matrix in which the non-zero entries represent the distances between adjacent vertices. The vertex from which we wish to compute distances is denoted s. For purposes of this example, we will take vertex 1 as s.

The Dijkstra algorithm, which we can use to find distances from s, employs three lists, called A, B, and C respectively. Each entry in A, B, or C consists of the name of a vertex, together with an associated label giving the minimum distance through the network from s to that vertex. The algorithm is as follows :

1. Place all vertices on list C.

2.Transfer vertex s to list A, call it n, and set its label to 0.

3. Set labels of all other vertices to x.

4.Consider all vertices R connected to vertex n just transferred to list A. If a particular vertex r is on list B, check if the sum of the label on n plus the length of the edge connecting n and r is less than the label of r. If so, replace the label on r with the shorter distance. If a particular vertex r is on list C, transfer it to list B and change the label to the sum of the label on n plus the length of the edge connecting n and r.

5.Transfer the vertex on list B which has the minimum label to list A, and call it n.

6. Unless all vertices are on list A, return to step 4.

The algorithm terminates with all vertices on list A, and with the label of each vertex indicating the minimum distance through the network from s.

The steps in execution are as follows :Table 2

* Circulation system simulation

The circulation flow simulation techniques which were discussed earlier for use in space and facilities needs estimation may also be employed in circulation system performance evaluation. In this case the purpose is to check the adequacy of a specific design proposal rather than to estimate the characteristics which any design should have.

One of the most ambitious circulation simulation systems yet to be proposed for this purpose was also one of the earliest : the COPLANNER system for evaluation of circulation systems in hospitals (Souder, et al 1964). COPLANNER was an interactive graphics system which allowed the designer to vary proposed building layout by using a light-pen to manipulate a cathode ray tube display. The system had access to an extensive and detailed data base of traffic flow information for hospitals. Layout proposals were evaluated by a sophisticated simulation of the operation of the circulation system over a period of time. In many ways COPLANNER was ahead of its time, and it was difficult to justify use of such an evaluations. However, it illustrated the way in which circulation evaluation facilities can be combined with an interactive graphics interface, and it seems likely that COPLANNER-like capabilities will eventually be incorporated as a standard feature in integrated computer-aided design systems.

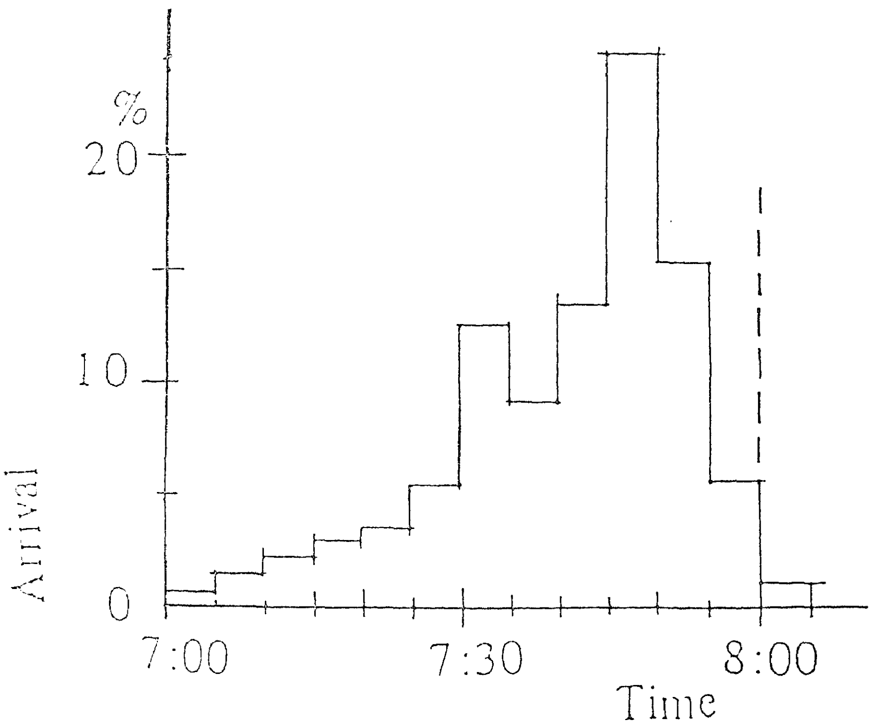


Fig 2.1-1
Arrivals of
commuters

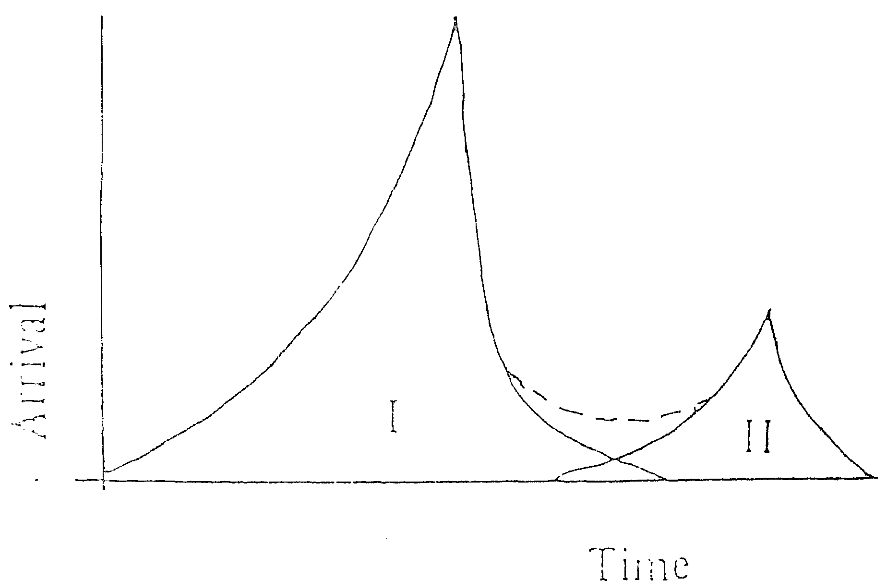


Figure 2.1-2
Model for change of
arriving time.

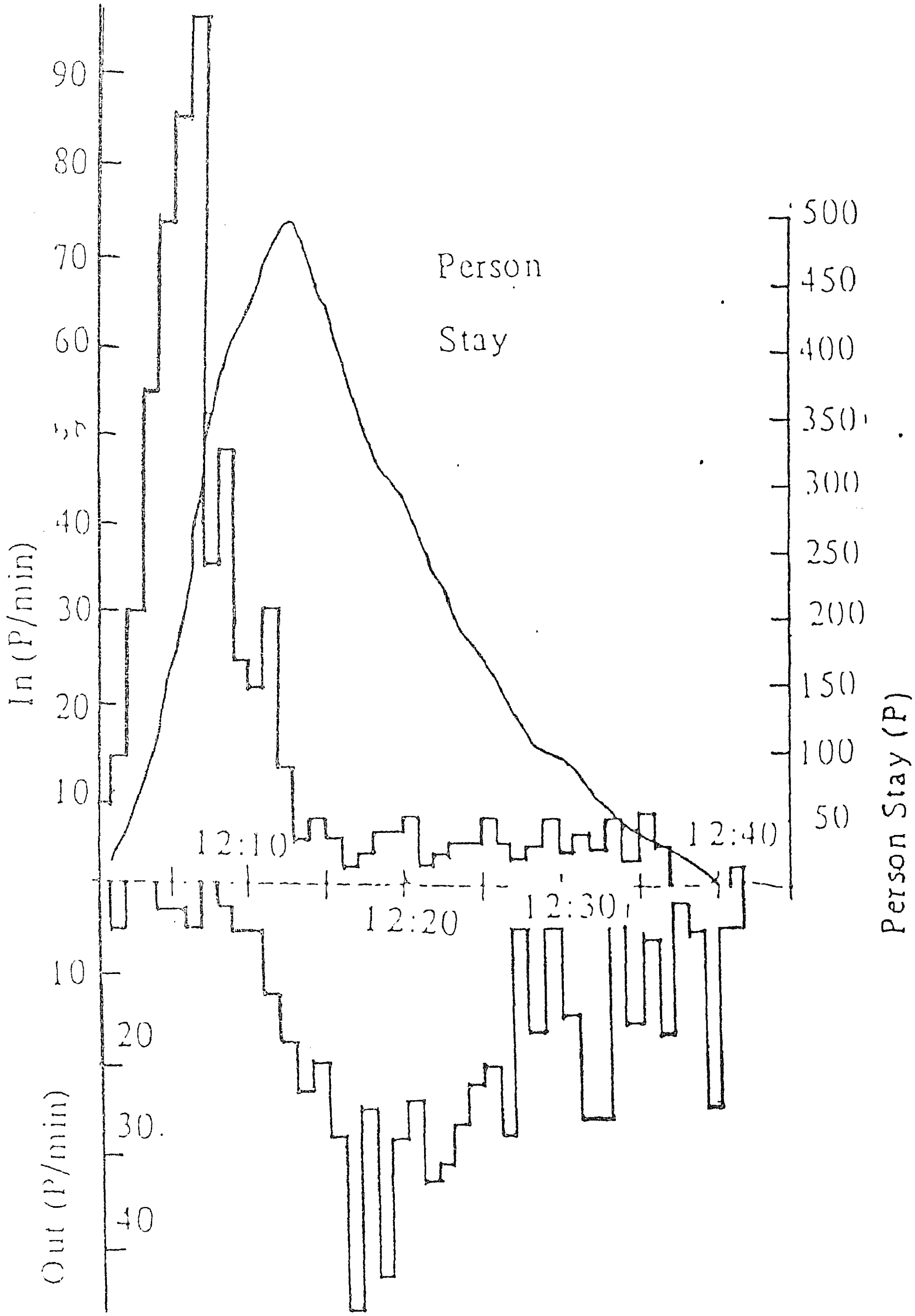
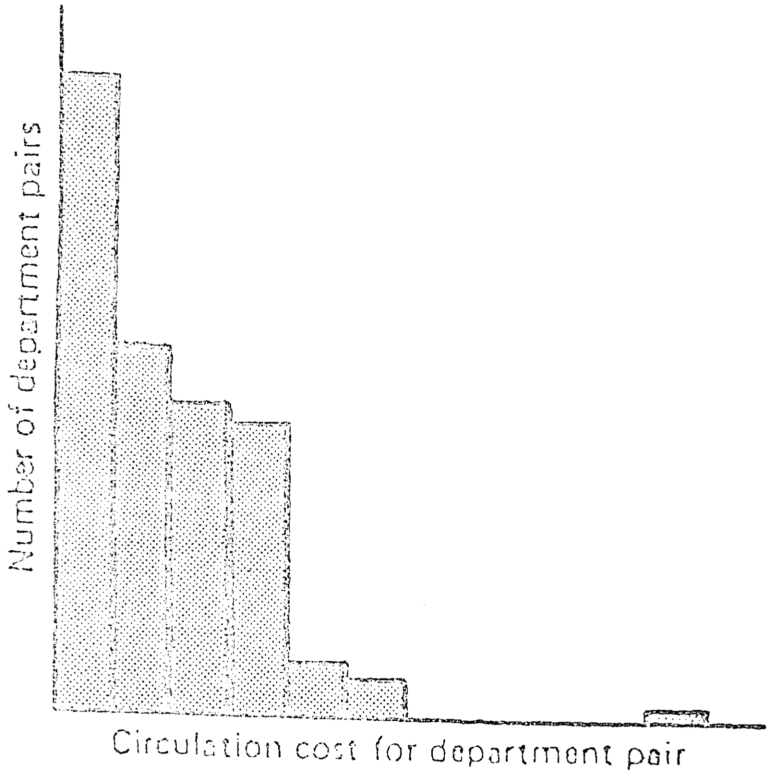


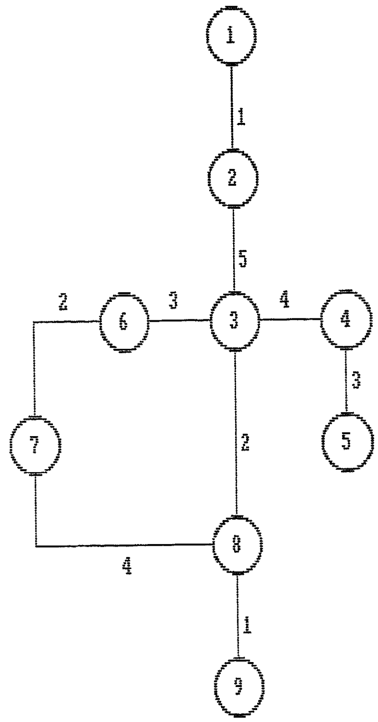
Fig.2.1-3
Changes of partrons
in restaurants at
different hours.



A

B

Figure 2.2-1
Some typical output from the Harness hospital circulation cost analysis program.
(a) Department pair circulation cost histogram.
(b) Interdepartmental traffic flow graph.



(a)

	1	2	3	4	5	6	7	8	9
1		1							
2	1		5						
3		5		4		3		2	
4			4		3				
5				3					
6			3				2		
7						2		4	
8			2				4		1
9								1	

(b)

Fig. 2.2-2
Dijkstra shortest path algorithm example.
(a) Example network.
(b) corresponding adjacency matrix, in which non-zero entrieson indtance betweenvertices (zero entries omitted for clarity).

Table 2

		A	B	C
Step 1	Put all vertices on C and set all labels to (∞)			1(∞) 2(∞) 3(∞) 4(∞) 5(∞) 6(∞) 7(∞) 8(∞) 9(∞)
Step 2	Put 1 on A and set its label to 0	1(0)		2(∞) 3(∞) 4(∞) 5(∞) 6(∞) 7(∞) 8(∞) 9(∞)
Step 3	Put 2 on B and set its label to 0+1	1(0)	2(1)	3(∞) 4(∞) 5(∞) 6(∞) 7(∞) 8(∞) 9(∞)
Step 4	Put 2 on A	1(0) 2(1)		3(∞) 4(∞) 5(∞) 6(∞) 7(∞) 8(∞) 9(∞)
Step 5	Put 3 on B and set its lable to 1+5	1(0) 2(1)	3(6)	4(∞) 5(∞) 6(∞) 7(∞) 8(∞) 9(∞)
Step 6	Put 3 on A	1(0) 2(1) 2(6)		4(∞) 5(∞) 6(∞) 7(∞) 8(∞) 9(∞)
Step 7	Put 4,6,and 8 on B and adjust labels	1(0) 2(1) 3(6)	4(10) 6(9) 8(8)	5(∞) 7(∞) 9(∞)
Step 8	Put 8 on B	1(0) 2(1) 3(6) 8(8)	4(10) 6(9)	5(∞) 7(∞) 9(∞)

		A	B	C
step 9	Put 7 and 9 on B and adjust labels	1(0) 2(1) 3(6) 8(8)	4(10) 6(9) 7(12) 9(9)	5(∞)
Step 10	Put 6 on A	1(0) 2(1) 3(6) 8(8) 6(9)	4(10) 6(9) 7(9) 9(9)	5(∞)
Step 11	Reset label on 7 to 9+2	1(0) 2(1) 2(6) 8(8) 6(9)	4(10) 7(11) 9(9)	5(∞)
Step 2	Put 9 on A	1(0) 2(1) 3(6) 8(8) 6(9) 9(9)	4(0) 7(11)	5(∞)
Step 13	Put 4 on A	1(0) 2(1) 3(6) 8(8) 6(9) 9(9) 4(10)	7(11)	5(∞)
Step 14	Put 5 on B and set its label to 10+3	1(0) 2(1) 3(6) 8(8) 6(9) 9(9) 4(10)	7(11) 5(13)	
Step 15	Put 7 on A	1(0) 2(1) 3(6) 8(8) 6(9) 9(9) 4(10) 7(11)	5(13)	
Step 16	Put 5 on A	1(0) 2(1) 3(6) 8(8) 6(9) 9(9) 4(10) 7(11)		

IV. FORECASTING

Forecasting means to anticipate situations to take place in the future based on some available clear-cut information. Generally speaking, it is fairly reliable.

To conduct an investigation on the use of buildings is to get the physical situations under our control. However, it is also important for us to explore evolution from the past till the present, or a future tendency. Whatever we plan for, we must forcasting and its extent of assistance in our planning varies with the methods we adopt or the physical conditions we face. Sometime our forecast fails due to the environmental conditions changing during the course of planning and such change takes place beyond the expectation of the forecaster or planner. In other cases, the decision made on many not be in line with the outcome anticipated in our forecast. The accuracy of our forecasting will depend upon our control over the varying factors. Varying factors refers to the varying of design requirements, environmental conditions and service conditions. It depends on complex phenomena and we shall never have any panacea which can be universally applied to all situations. We shall have to select means suitable to the type of changes, objectives, and the nature of the problem, etc. Generally speaking, continuous change, may be calculated by quantitative measurements.

1. The Mathmatical Way

1.1 Regression Forecasting

This is a statistical method for forecasting the future and has been one of the methods most commonly used. In this method, all information and data about the past and current situations

are to be transformed into graphic form and extrapolated so to show tendencies. We can say that, with this approach, we can extend past experience so as to grasp the future.

With the method, we transform the various changes of our object into simple geometrical graphs or forms. We call it the Extrapolation Method. It will be effective for a short period of time for objects having a tendency to normal functional changes. However it will not be suitable for long-range forecasting for objects having complicated changes.

When supplemented with some mathematic calculations, it is called the Quantitative Method. In this method, the inter relations existing between the major factors and their quantities with a certain phenomenon are expressed in the form of formulas for anticipating future tendencies. With the extrapolation method, we shall be able to obtain an accurate measurement of the value of changes shown in the slope of the extended line.

In the Regression Method, we forecast the future on the current information available. Such forecasting will be made possible only when the phenomenon changes continuously. It will not apply events which do not take place continuously. The regression line which appears to be in the form of a straight line will be the simplest one. However, it may also appear to be a second order polynomial curve, or become a form of growing curve as discussed later. We may determine the regression line by means of the Ordinary Least Squares Algorithm, the Moving Average Method, and Manual Plotting Method, etc. according to the order of past time.

The Moving Average Method is used to work out a long-term tendency by getting rid of the fluctuation caused by seasonal changes from the whole which presents a radical change of a constant tendency. In manual Plotting Method, we establish the regression line by means of our observation and can work out the dynamic state of regression easily and rapidly. Its disadvantage would be that it dose not have the objectivity adherent to the Ordinary Least Squares.

1.2 Trend Line Method

The trend is theoretically representing the process of growth of a changing quantity in the long run, such as changes in the population and economy. In this theory, we claim that, during the development of events or situations or growth of a life, there must be a pause or retrogression, In the regression method. if which the tendency continues continues mono-directionary and endlessly. This appears to be illogical. We should then justify the growth with a theoretical method.

There are two theoretical model for the trend line, a Logistic Curve and Gomperts Curve are commonly used.

(1) Logistic curve

This is established on the basis of the biological population theory, abbreviated as an L curve. Its basic form can be shown below:

$$P(t) = \frac{N}{1 + K_e^{-\lambda t}}$$

In the above,

P(t): the population at the point of time t.

N: the maximum population at the time of t-> .

K, λ : the constants denoting the environmental conditions.

In the curve as shown in Fig.1.2-1, we note that it rises slowly at the beginning, increases radically at a certain stage, increases constantly before and after the turning point, and lowers its increase rate gradually from that point so as to form a lateral development until it approaches to its limit.Its increase rate S is shown as below:

$$S = \frac{P(t')}{p(t)} = \frac{\lambda}{1 + K_e^{\lambda t}}$$

The dotted line shown in the fig.20 indicates the decrease function.

In a certain limited environment, there will be a definite limit of population. For instance, population increase will not continue when its increase goes up to a certain point governed by the size of land and its productivity. Another example: When the increase rate of black and white television sets reaches situation, the demand for color television sets and stereo audio equipment may still be increasing

Fig.1.2-1 shows it in a theoretically simple form only. In fact, the curve may look like a ripple as environmental conditions may often change during the course of development, such as: the demand will be lowered as the increase of customers reaches a platen. The altitude may often change. If we change the environmental conditions by means of addition or by remodeling the building, or renewing development, the situation may be improved and the curve may become a new S form. This has often been used to forecast the increase of customer demand.

(2) Gomperts curve

This is also an S-shaned growth curve called a G curve as shown in Fig.1.2-2.

The general forn of Gomperts curve is shown in the following:

$$Y = kg^{ct}$$

k,g, and c: constants

when 1 > c > 0, and t -> , y=k. k will be the maximum value.

Its turning point takes place at a lower position in G curve than it dose in an L curve. We forecast development in the future by making use of the information and data shown below the turning point as as shown in Fig.1.2-3. The tendency shown in the curve appears to be stronger than an L curve.

In quantitative forcasting, we do not simply forecast only the objects we wish to forecast. We pay close attention to other indexes associated with those objects so as to some extent enable us to explore the trend of objects to the forecast. Generally it is established by comparing a certain index against the current information and data available in various advanced countries. For instance, wen forecast the situations we are going to face when the per capital is brought up to a certain level. This method is also called Compared Supplementary Method or Preceding Index Method.

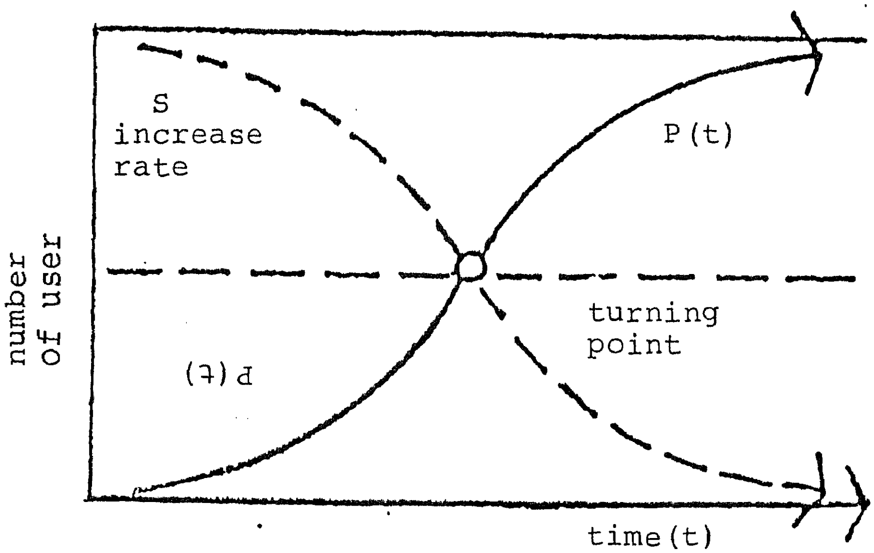


Figure 1.2-1
Logistic curve
(simple form)

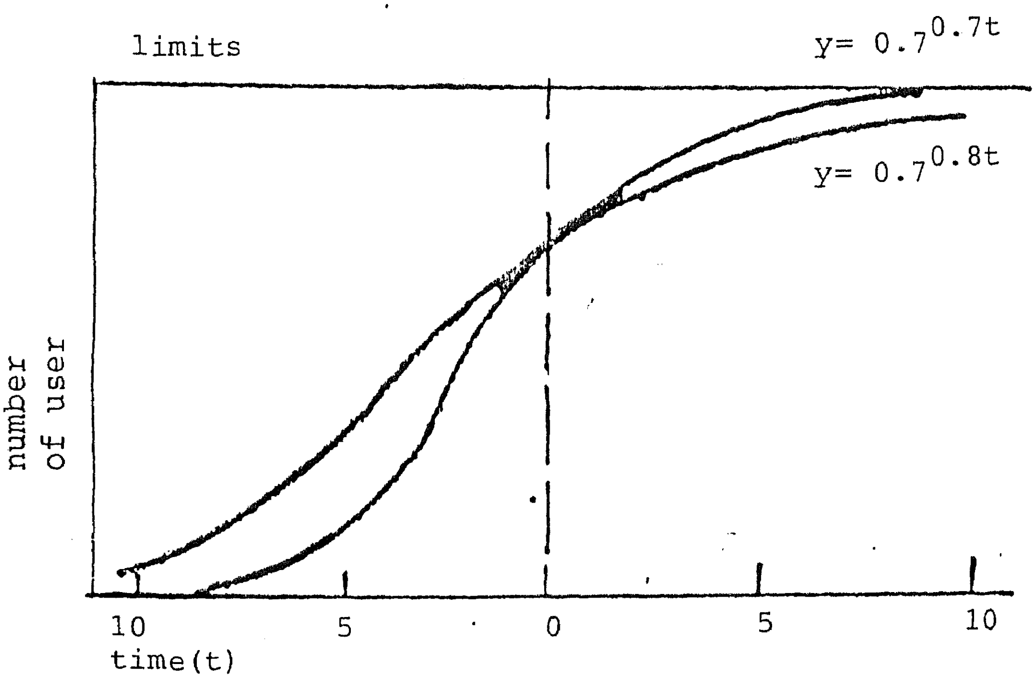


Figure 1.2-2
Gomperts curve

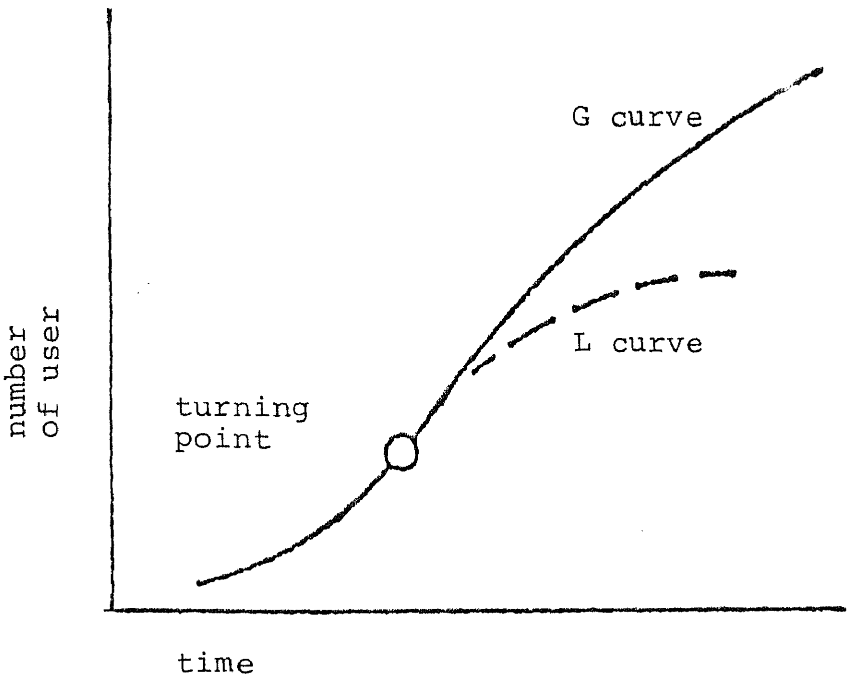


Fig. 1.2-3
Comparison of L curve
and F curve

2. Space Quantity Needs Analysis

Space needs analysis basically involves taking data about people, activities, and equipment to be accommodated in a project. and translating it into a set of precise spatial requirements. The initial data might consist of the results of interviews with clients and users, questionnaire survey data, observational studied, standard guidelines, or investigations of previous similar projects.

Computer-based techniques are beginning to revolutionize forecasting of space needs, since they make possible the efficient manipulation of large volumes of data, and the performance of much more extensive and sophisticated analyses than had been possible in the past. In the following, some typical examples are described.

One of the first heuristic search spatial synthesis programs to incorporate partitioning of the problem into sub-problems was developed by Beaumont (1967). Form analysis of a circulation flow matrix. Beaumont's program first breaks down the list of spaces to be located into a binary tree of spatial subgroups (Fig.2-1(a)). An amoeba-like process is then employed to generate a layout. The first pair of spatial sub-groups is laid out by successively locating one spare module of one, then a module of thee other, and so on (Fig. 2-1(b)). Each of these subgroups is then redivided and so on until all spaces are laid out.

2.1 Nonlinear And Linear Programming

Even stronger, but correspondingly less general techniques than heuristic search are nonlinear programming and linear programming methods. These methods were initially developed for solution of economic, operations research, and engineering problems, but they have found some interesting applications to floor plan synthesis problems.

*Nonlinear Programming

Nonlinear programming has been uses in conjunction with dimensionless representations of floor plans to generate optimum dimensioned layouts with respect to some cost criterion and subject to certain functional constraints (mitchell 1974,1975,Sauda 1975, Mitchell, Steadman and liggett 1976, McGovern 1976). The following is a typical example.

Fig.2.1-1(a) shows, in dimensionless representation, a proposed layout for a small house. Dimensional constraints are specified as follows: The problem is to choose values for the dimensioning vectors such that these constraints are met, and the following objective is minimized:

$$\sum_{r=1}^{7} a_r c_r$$

Where

ar is the total floor area of room r, and

Cr is the construction cost per square foot for room r.

This is a very simple construction-cost formula, A more elaborate objective function could obviously be formulated if desired. Values for the coefficient Cr are as follows:

Room	1	2	3	4	5	6	7
	living	kitchen	bath	hall	bedroom	bedroom	bedroom
Cr	1.0	2.0	2.0	1.0	1.0	1.0	1.0

The problem is nonlinear, due to the presence of multiplicative terms representing areas in the constraints and objectives. Powerful nonlinear programming algorithms now exist for finding solutions to problems of this form. They cannot absolutely guarantee to generate the optimum solution, but experience has shown them to be extremely reliable and efficient. In this case, a program developed by Clasen, Graves and Lu (1974) was employed to find the solution shown in Fig.2.1-1(b) in about two seconds of computation on a 360/91 computer, It would be quite feasible to handle much larger problems, consisting of hundreds of variables and constraints.

*Linear Programming

Standard linear programming techniques are applicable and will very efficiently generate a solution or report infeasibility where this type of dimensioning problem can be formulated in terms of the optimization of a linear objective function subject to linear constraints. Typical linear objectives are maximization or minimization of overall plan length, width, perimeter, or proportion ratio. Typical linear constraints are upper and lower bounds on allowable lengths, widths, perimeters, and proportion ratios of individual rooms and of the overall plan envelope.

An immediately obvious limitation of the linear programming approach is that area constraints are nonlinear, and thus cannot be incorporated in a linear programming formulation. Furthermore, objectives representing such important properties as construction cost,heat loss, ect., are also functions of floor area.

However, problems involving terms representing floor area can be formulated for linear programming solution if values for one of the dimensioning vectors are known at the outset. Consider the plan (in dimensionless representation) of a mobile trails unit illustrated in Fig.2.1-2(a). Severe restrictions on allowable width constrain they dimensioning vector to the values shown. Cnostraints on lengths, widths, areas, and proportions of the various rooms are defined as follow:

The objective is to minimize overall length of the trailer subject to these constraints. Thus the problem can be formulated as:

Minimize $(a+b+c+d+e+f+g)$

subject to:	$g<=10.42$
$a>= 6.25$	$(a+b)>=15.0$
$a<= 6.67$	$(a+b)>=16.67$
$(b+c)>=13.33$	$(c+d+e)<=20.0$

$(b+c)\leq 15.0$	$(f+g)\geq 15.0$
$d\geq 4.17$	$(f+g)\leq 16.67$
$d\leq 6.67$	$c\geq 6.67$
$(e+f)\geq 12.5$	$c\leq 8.89$
$(e+f)\leq 16.67$	$(d+e)\geq 11.11$
$g\geq 8.33$	$(d+e)\leq 13.89$

Fig.2.1-2(b) illustrates the optimum solution, found in a fraction of a second of computation, by a standard linear programming system.

Linear programming procedures are much more powerful than any of the techniques discussed previously. They guarantee to produce the optimum solution, and they do so with great efficiency. The price which must be paid, however, is that their application is restricted to a relatively narrow range of problem types.

2.2 Analytical Procedures

Analytical procedures are the most powerful of all. Non-trivial architectural spatial synthesis problems which can be solved analytically are rare, but contrary to general opinion, they do exist.

One of the most interesting such tasks might be termed the Palladio Problem. Consider a floor plan in dimensionless representation, such as that shown in Fig.2.2-1(a). Let us assume that we wish all the rooms to be squares. This requirement can be expressed by the following set of simultaneous linear equations:

$$X1=Y1,\quad X2=Y2,\quad X1+X2=Y2,\quad X1=1$$

By substitution

$$Y2=2$$

The resultant plan is shown in Fig.2.2-1(b)

If we expand the problem to allow proportioning requirements other than 1:1 (that is, 2:1, 3:2,ect), there is non linger a single solution. If, for example, take the proportion 2:1, there are now two different ways a proportioning requirement can be met by each room. Room length could be twice room width or vice versa. Two proportioning equations for each room, therefore,can be formulated. To proportion a plan, a system of equations composed of one equation for each room must be selected.

There is not, then , a unique set of simultaneous equations for proportioning a plan, In general, there are 2n sets of equations where n is the number of rooms in the plan. Fig.2.2-2(b) shows a portion of a tree diagram which represents exhaustive enumeration of all possible combinations of simultaneous equations for the particular proportioning problem illustrated in Fig.2.2-2(a).

It is, of course. possible to specify proportion requirements which are unrealizable within a given plan. In this case it will be found that the simultaneous equations have no meaningful solution.

The dimensioning problem as formulated in this way may be solved either by hand or by computer, with any of the standard methods * for solving systems of simultaneous linear equations, for example, substitution, elimination, the GAUSSINA method, or the Cramer method. Where the number of rooms is greater than about three or four, the number 2" of different systems of equations which must be considered becomes quite large:

n	2	3	4	5	6	7	8
2n	4	8	16	128	256		

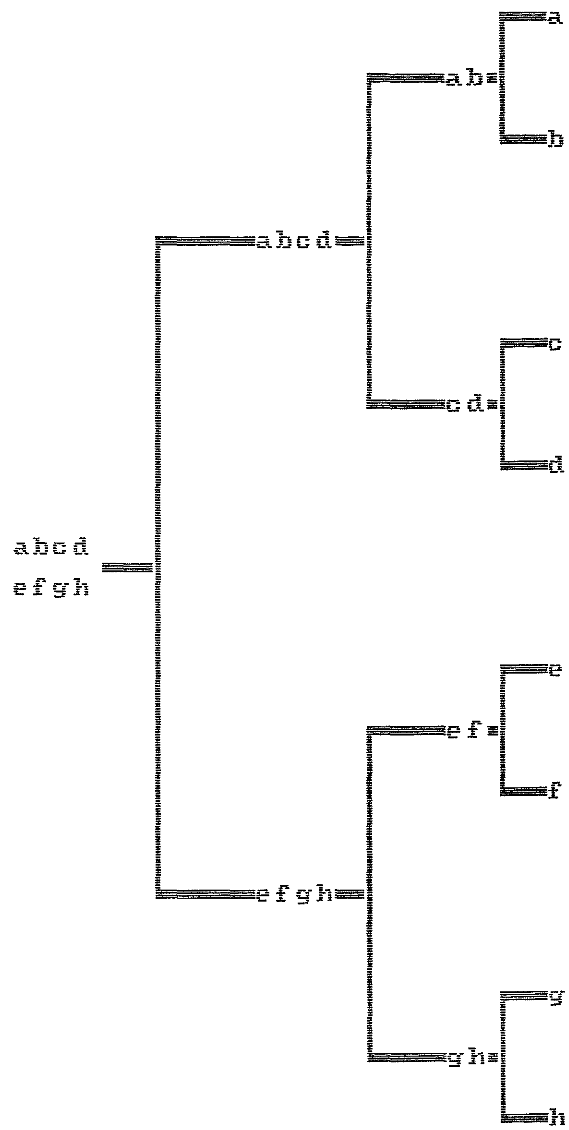


fig 2-1 (a)
Use of planning in Beaumont's (1976) floor plan layout program.
(a) Breakdown into spatial subgroups.

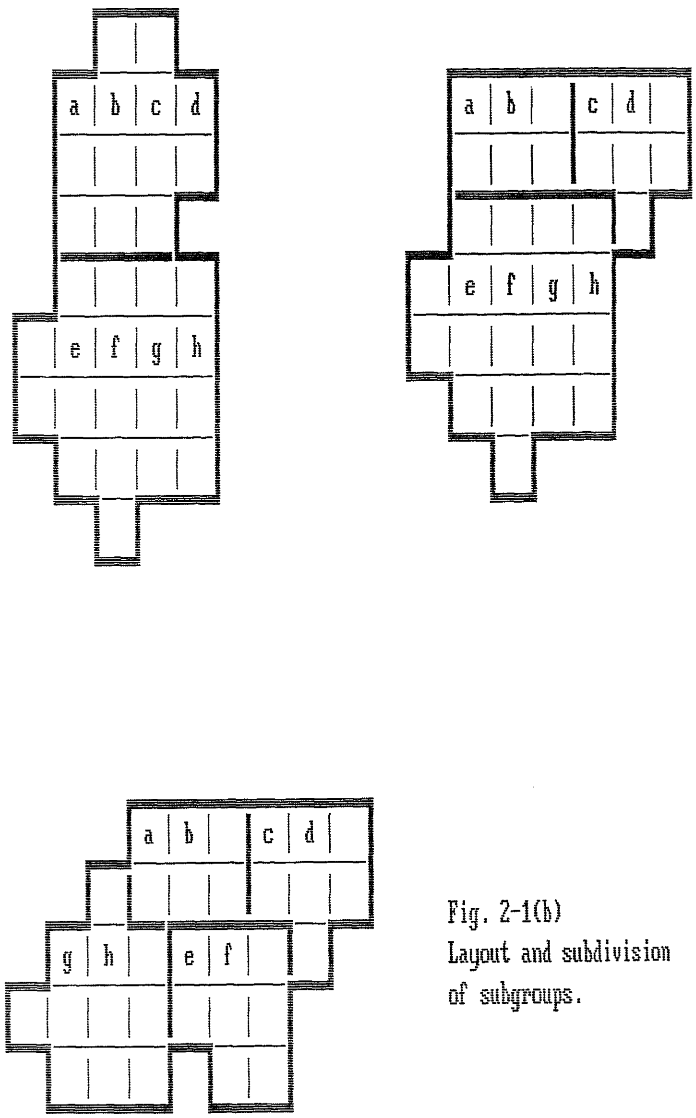
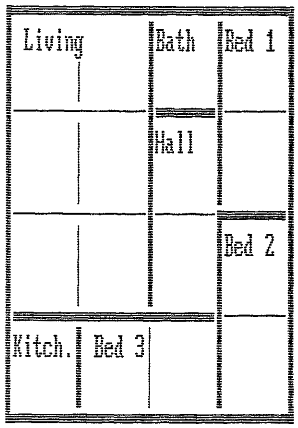
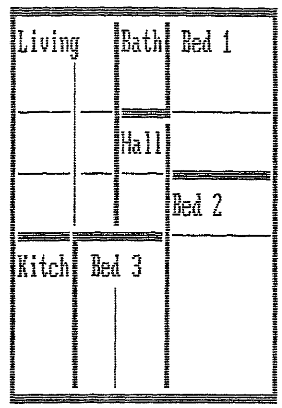


Fig. 2-1(b)
Layout and subdivision
of subgroups.

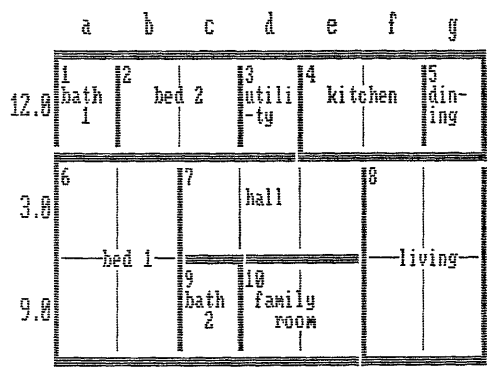


A

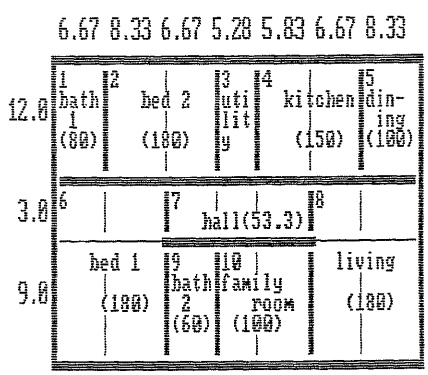


B

Fig. 2.1-1
An example of application of nonlinear programming to a floor plan layout problem
(a) Dimensionless representation of layout.
(b) Optimum dimensions found by nonlinear programming.



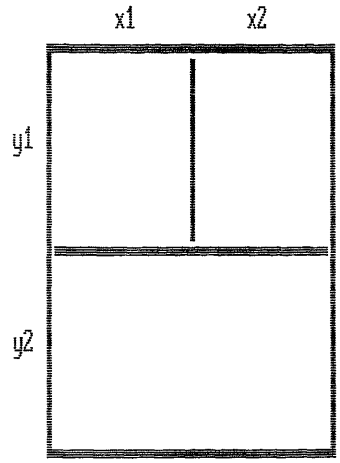
(a)



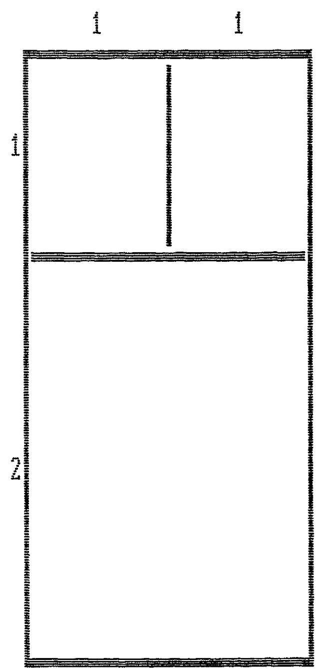
Length = 47.8ft

(b)

Fig. 2.1-2
An example of application of linear programming to a floor plan layout problem.
(a) Dimensionless representation of a "double wide" (25 ft) mobile home unit, with dimensioning vector along short side fixed.
(b) Optimum solution drawn to scale.

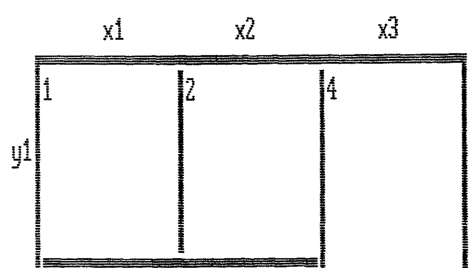


(a)



(b)

Fig. 2.2-1 (a) (b)
Algebraic solution of a simple proportioning problem.



(a)

Fig. 2.2-2 (a)
Proportioning problem for a plan. Where all rooms are to be 2 : 1. Dimensionless plan.

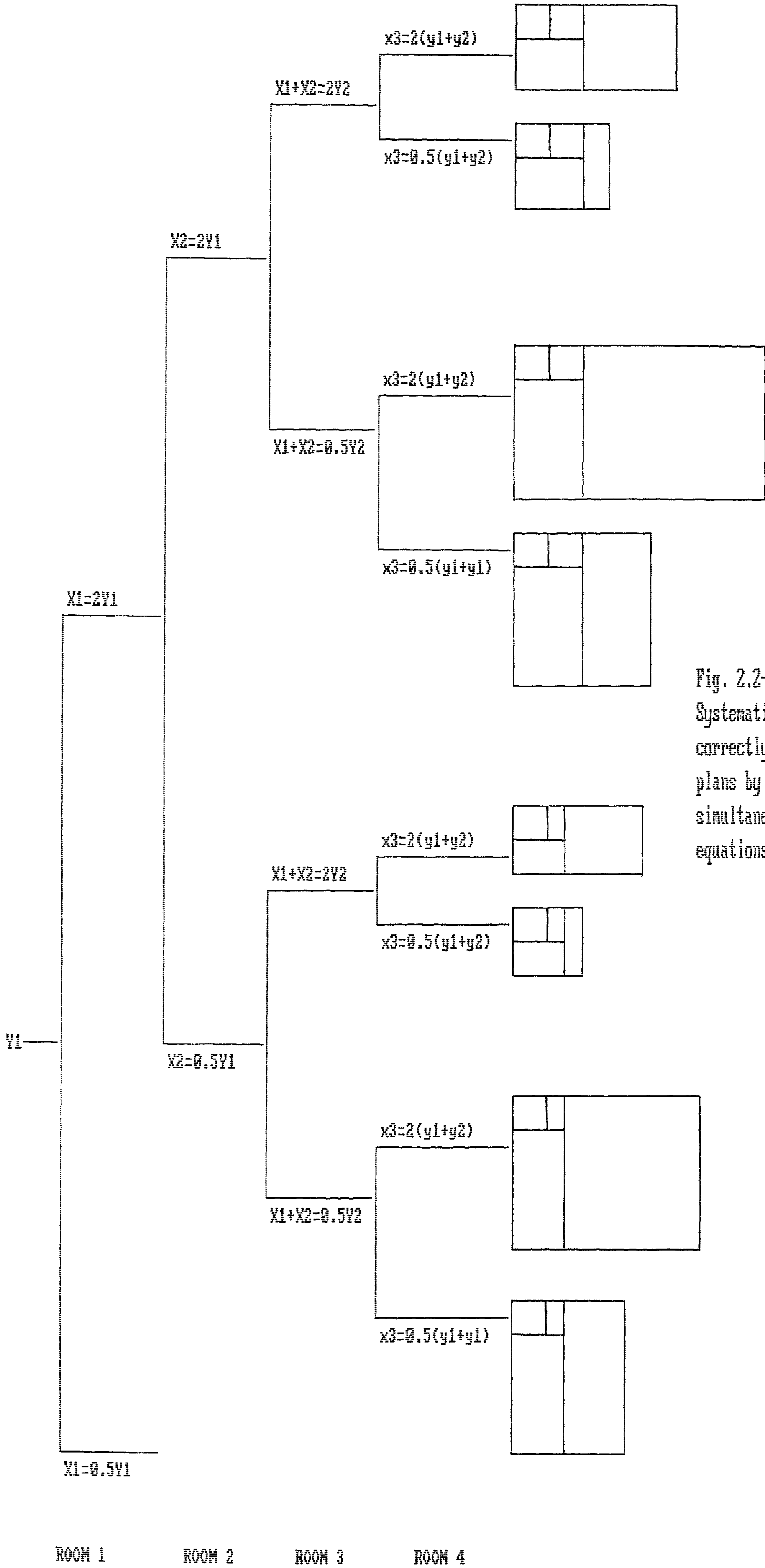


Fig. 2.2-2 (b)
Systematic generation of correctly proportioned plans by solution of simultaneous linear equations.

V. CONCLUSION

Computers are a major technological innovation having a profound impact throughout society. Since the Industrial Revolution, technological change has seemed an inevitable fact of life in Western societies. Recently, however, a series of major criticisms of the modern technological society have come to prominence. In addition to the various concerns with energy, pollution and resources, there are signs of a crises in advanced technology. Technological "progress" has been criticized for its ecological destruction, its deterministic tyranny, its psychological alienation, and its social disruption. This is leading to a comprehensive reassessment of the costs and benefits of technological change.

It is in this critical and crisis-ridden atmosphere that architects are being asked to accept the most striking technological innovation in the history of their profession: the introduction of computers to the architectural design process. Understandably, most architects are variously mystified, dubious, or disturbed about the effects computers will have on the practice of design, despite the enthusiasm of the minority of their fellow architects who are in the vanguard of this particular technological progression. Enthusiasm for the computer as a design aid is somewhat misplaced-at least until the machine's performance has significantly improved. But the growth of computer applications in architecture seems inexorable. Can we anticipate the effects that CAD systems will have?

As with most other major technological innovations (for example, motorcars, aircraft, television) the effects of computers as experienced in various sections of society are probably a mixture of benefits and problems. Because the concept of technological "progress" has been relatively unquestioned, society has tried to cope with the down-side of technological change on the assumption that these are generally outweighed by the benefits. Only recently, in response to the growing criticisms and crises of modern industrial technology, has the idea of "technology assessment" gained some limited favor.

1. Technology Assessment

A technology-assessment program would attempt to forecast the effects contingent upon particular technological developments. These forecasted effects might be ranked as "first-order" ,"second-order" , etc, in recognition of the way in which ripples of effects and "side effects" are generated by technological change.

1.1 Forecasting Change and Effect

In Britain and in the field of computer-aided architectural design, a related concept, but with a different emphasis, was included in a report of the Department of the Environment (1969). , which would be responsible for forecasting the problems raised by introducing computers to architectural design practice. In this case, however, the"look-out" research was intended more to find means of smoothing the way for technological change (that is, CAD), rather than to assess and perhaps question the implications of that change. Thus, a major aim of this "look-out" research was expressed as: "To identify... obstacles to progress that could prove to be critical to the development of computer aids to building design." Obviously the crises and criticisms of modern technology that developed in the early 1970s have cast technological change in a new light. Developments such as computer applications are no longer regarded as the inevitable march of progress, from the path of which all obstacles must be removed and to which people must learn to adapt.

Perhaps fortunately for architects, their profession may well be one of the last sectors of society to undergo "computerization". Computer-aided design systems for architecture are still in their infancy. This could mean that architects will be fortunate enough to avoid some of the major disbenefits of computerization that may be experienced in other trades and professions that succumb earlier. In most cases, however, the full range of effects of computerization are never reported, and it therefore seems unlikely that a "wait-and-see" attitude will on its own prove adequate as an assessment technique; other techniques are also required.

1.2 Systemic Testing

One of the major difficulties that prevents any comprehensive assessment of the effects of technological change is simply the problem of forecasting just what the actual effects and side effects are going to be. Not until the likely effects are known can they be assessed for their desirability or undesirability. This difficulty is particularly acute where a technological change is coupled to a required behavioral change in that relatively unpredictable organism, the human being. As Jones (1967) puts it: "The reactions of users to the existence of radically new facilities cannot be assessed until behavior had adapted to suit the new conditions. The behavioral reactions of so complex an organism as the human being in a social context cannot be foreseen without large scale tests." Jones therefore proposed that there was a need for "test cities" that would incorporate "simulations of new systems that are at the stage of pre-design evaluation" for the volunteer inhabitants to test in real-life situations.

Computer-aided design systems presumably would fall into the category, "radically new facilities" requiring such tests. The man-machine system of designer plus computer is a new system, of which there is only fragmentary experience to date. Simulations of computer-aided design systems, as have in fact been carried out by the author and colleagues (Cross, 1967; Cross et al, 1970' Evans, 1969), could provide a basic forecasting and assessment technique.

There are two main problems which make investigation of the effects of new computer-aided design systems difficult. First, there is the probability that computer aids will so alter normal design processes that speculation based entirely on conventional practice cannot be reliable. Second, there is the prohibitive cost of setting up working experimental systems to provide the necessary experience and feedback.

In attempting to investigate and predict the behavior of a CAD system, it would be misleading to extrapolate from the previous behavior of the system's components in isolation from one another. The most important factor in a working system is the effect of the whole upon its parts. The behavior of the components is affected by their being combined into a system-this being, of course, the very purpose of combining them. It is these systemic behavioral patterns that need to be forecasted and assessed.

The problem is, therefore, to devise a suitable simulation of a computer-aided design system. Current prototype systems consist essentially of a teleprinter console, usually plus a graphic device, through which the designer "converses" with the computer-which may actually be some miles from the console. All that the user perceives of the system is this remote-access console, and the remainder is a black box to him.

Viewing the computer-aided design system in this way leads to an obvious suggestion for a simulation technique-one may as well fill the black box with people as with machinery. Doing so provides a comparatively cheap simulator, with the remarkable advantages of the human operator's flexibility, memory, and intelligence, and which can be reprogrammed to give a wide range of computer roles merely by changing the rules of operation. It sometimes lacks the real computer's speed and accuracy, but a team of experts working simultaneously can compensate to a sufficient degree to provide an acceptable simulation. This was the basic hypothesis of the research described here.

2. Simulation Studies

The first simulation project (Cross, 1967) established the basic methodology for the technique. The form of this project was to carry out experiments in simulating computer-aided design systems, and to observe and record the interactions, that is, the messages sent between user and "machine", and any other relevant factors or occurrences. Much of the effort went into establishing the technique of simulation, in the hope that others might build on the experience gained, but it was also possible to generate some tentative information on the probable behavior of real computer-aided design systems.

The basic technique used in the simulation experiments was to provide a designer (who had a prearranged problem to work on) with a means of communicating with a human "expert computer". Either speech or writing was the medium for communication between designer and "computer".

Verbal communication was used in the earlier, pilot experiments, but the majority of the twenty-three experiments were conducted with the use of written messages transmitted by a closed-circuit television link. The hardware used for communications never reached a very sophisticated level, primarily because the project was concerned with software factors, and increasing sophistication of communication hardware did not appear to after the patterns of man-machine interaction significantly.

The project was primarily orientated towards computer-aided design of buildings, and the designers used as subjects in the experiments were generally architects. They were given the requirements for a small building to work on as a design problem during the experiments. The "computer" was simulated by small interprofessional teams of building specialists.

No restrictions were placed on the content of the messages sent from designer to "computer", in order that some assessment could be made of the pattern of interaction and the facilities called for in an idealized free system. There were no language restrictions imposed either, as this would have meant an undesirable learning time for the subjects and because uninhibited interaction patterns were considered to be probably the most valuable in these exploratory experiments.

How, then, did the designers respond to these idealized opportunities for using a computer-aided design system? Perhaps most important was the fact that they all found the experiments, which lasted about one hour each, very stressful and very hard work.

This feeling of stress experienced by the designers may have been largely induced by the experimental situation itself, in that the designers were aware of being "watched". However, the feeling of stress in working with computer-aided design systems has been confirmed by research workers in the field, with comments such as: "You get the feeling that there is nothing you can do to beat the machine". This is a point which should be worthy of careful attention, as such stress obviously could have adverse effects on the designer's capabilities, as well as limiting the duration of work periods.

2.1 An Information Retrieval System

A second project (Cross et al, 1970) used the simulation technique to investigate some of the possibilities for a computerized system to aid the designer in choosing the correct component in a design assembly. The simulation technique was used as a tool for exploring some of the questions arising from a postulated major computerized information-retrieval system for the construction industry. In particular this project was orientated towards the use of such a system for component selection during project design. Floor finish was the component selected for the project. The information now available on floor finish is in a particularly poor state, and this component often fails in use because the designer selects inappropriately from the many hundreds available.

The questions which were explored in this fairly modest study were:

1. Using "conversational" retrieval systems, can a person who is not trained in the design discipline in which the component selection takes place, select an appropriate component? (For these purposes, such a person is called a "naive designer".)

2. Is it feasible to devise a question-asking procedure for the computer, which leads to the selection of an appropriate component?

3. If so, do significant differences arise between systems in which (a) the designer asks all the questions, and (b) the computer asks all the questions?

To investigate the above questions, three designers were asked to perform two experiments each, in which they would select a floor finish suitable for some design situation of their own choosing, the requirements of which they were familiar with. To aid them in their selection they had a simulated computer with which they could communicate by written messages. The "computer" was actually two people with specialist knowledge in this field who were backed up with a store of data on floor finishes. In the first experiments, each designer asked the"computer" whatever questions he wished, but in the second experiments the "computer" asked all the questions.

The three designers were chosen to satisfy three types required for the study, thus:

(1) "naive" designer with no training in building design or construction;

(2) recently qualified, but inexperienced, architect;

(3) qualified and experienced architect.

For analysis of results, the floor-finish attributes were divided into two groups: structural compatibility and performance requirements. This division was suggested because compatibility with the building structure is essential if a floor finish is to avoid early and serious failure, whereas to meet all of the performance requirements is not usually so essential-failure being, perhaps, only accelerated wear.

There was a significant difference between the results obtained from the "naive" designer and those from the other two designers, in that he failed to investigate any of the structural compatibility attributes. This suggests that "naive" designers could probably not satisfactorily use an information retrieval system which relied entirely on the question-asking initiative coming from the designer.

The other interesting difference between the "naive" designer and the other two was in the time taken and the number of questions asked. The qualified designers spent much longer exploring components' properties and specifying their design situation.

Further experiments suggested that "naive" designers could use a sophisticated system such as was simulated, provided that the computer asks the designer some "fail-safe" questions. The system could in fact be devised as a teaching aid to provide technical information where required.

In the second set of experiments the "computer" had a prepared series of questions which it presented to the designer. These questions covered both the structural compatibility attributes (requiring yes/no answers from the designer) and the performance requirements (requiring graded answers from the designer, on a five-point scale of 0-4 for each requirement).

Thus the"computer" built for itself a design model, or profile, of site conditions and performance requirements, which it matched against the profiles of all available floor finishes. The 'computer' then output to the designer the one(s) which satisfied the design model, together with information on costs, maintenance, suppliers, etc.

There were no significant differences between the types of designer in these latter experiments, as all of them were able to answer the "computer's" questions (although the "naive" designer had some difficulty with building industry terminology, such as "damp-proof membrane").

2.2 An Intelligent System

A further project (Evans, 1969) was concerned with looking ahead to the time when computers can be expected to exhibit reasonable degrees of intelligence. The advent of machine intelligence would lead to the development of intelligent computer-aided design systems. In this situation the designer could perhaps succeed with less professional expertise than is required at present, and hence the whole design process could be accessible to designers who have little or no professional training.

In order to investigate this possibility, an intelligent computer-aided design system was simulated, and the computer functions that evolved between designer and "computer" were analyzed. The future designer was postulated as being a "generalist", that is less specialized and less expert than present designers. ("Naive" designers were used as an approximation to the generalist.) The "computer" was allowed to exercise its intelligence in monitoring the designer's progress and in its responses to messages from the designer.

Ten experiment, in which communication between designer and "computer" was a mixture of verbal and written messages and drawings, were performed. In the first three experiments, the designer and "computer" (problem expert) merely faced each other across a table, but in the remaining experiments closed-circuit television and intercom units were used. The use of verbal messages and the inexperience of the designers could not result in the sort of explicit, structured communication of the earlier projects reported here. It was thought, however, that the resulting increased speed of message flow would reduce the likelihood that delays would interrupt the designers' working.

The designers were given one of two problems: the design of an integrated services duct, or the design of a small exhibition building. These problems had several points of dissimilarity which seemed likely to produce a variety of demands on the design system. The duct design required a lot of information to be gathered quickly, with the designer soon reaching the stage of detailed hardware design. The exhibition building design was much more complex. Both problems were intentionally fairly small, so that a range of design techniques would be required in the one to two hours spent on them.

In general it was found from the experiments that the resulting design work produced solutions which were usable but not very efficient. The simulated computer system was not entirely adequate to support a designer of so little experience as the "naive" designer, but it seems likely that a "generalist" designer (much less specialized that present designers) could be adequately supported by such a system.

3. Effects and Effectiveness

There are three major areas in which one may assess the effects that CAAD systems might have for the architect and the architectural design process. First, there are the effects on the day-to-day working life of the architect - on architectural design as a job. This job could change radically.

Notably, for the individual designer, CAD systems pose threats of stress and anxiety, through intensifying his work rate and making public and assessable his decisionmaking.

Second, there are the areas of change where the computer meshes with and promotes a broad set of related changes. Computer-aided design systems will catalyze these other factors which affect the design process, and will amplify the changes that are leading to a reduction of the design effort applied in some parts of the process. This reduction of effort may well lead to problems of redundancy, particularly at "middle-management" levels.

The third major area where we might draw some conclusions on the effect of CAD is in the structure of the design team. Roles, relationships, and patterns of communication between the architect and other participants in the design process are very likely to change under the impact of computer systems. Generally this change will be towards a more "open" structure. In the long term, CAD systems could also make the design process accessible to a wider range of participants. In particular the explicit aim of some CAD developments is towards user participation in design-even, eventually, user control of the design process.

Regarding the assessment of the effectiveness of CAD systems, it seems that, for the moderately sized design problems used in the experiments reported here, there are no overwhelming differences between human and machine performances. In highly constrained problems, the machine tends to produce results approximately 6% better than those of the humans. In less constrained, more "real world" problems, however, the machine does not perform so well and humans (given adequate but modest time to produce their solutions) tend to produce results approximately 4% better than those of the machine.

Man-machine symbiosis is also still a long way off if the results of the experiments with an "interactive" CAD system are typical. The man-machine mode, as used in these experiments, tended to produce worse results than either humans or machines alone. Man-machine results were approximately 6% worse than those of unaided humans, and approximately 2% worse than machine results. The concept of interactive man-machine problem solving would seem to need to be reorientated significantly from that of the evaluative program used in these experiments.

In general we might conclude that the assessment attempted here has indicated that computer-aided architectural design systems may have wide effects but limited effectiveness.

That wide-ranging effects would be associated with the introduction of computers to the design process is perhaps only to be expected. A lack of effectiveness in terms of the overall efficiency of the design process and the measurable criteria (quite apart from the unmeasurable ones) of the design product is more surprising, and doubtless will be disputed by the change agents. But until further experimental evidence is forthcoming, it seems that architects could justifiable maintain a certain skepticism about the performance of the machine as a designer.

Perhaps, after all, the quantifiable aspects of architectural design are just not as difficult as the computer systems' protagonists would suggest? We might possibly conclude this from a report by Berger et al (1974) of some experiments with a General Decision Model (GDM) computer program and its application to space-planning problems. To establish the relevance of the optimization procedure of the GDM, Berger et al first established some date on random solutions to a modest architectural design problem. They defined this as a general problem in terms of a multistoreyed rectangular building on a rectangular flat site, to be designed within a set on nine variables: site depth and breadth, building depth and breadth, internal corridor width, number of stories, story height, usable floor area, and percentage of glazed area on the facade. To define a particular problem, each of these variables was then given lower and upper bounds (for example, glazed area between 10% and 100%) or constrained in other appropriate ways (for example, fixing the site dimensions and controlling the relationship of glazing area to room depth). In summary the problem might be considered to represent the significant quantifiable variables in a typical outline office-building design.

On the basis of what little evidence there is, designers do seem to be quite good at producing near-optimum solutions. It may be remembered that the architects in the case study that formed the basis of Luckman's (1967) AIDA design method chose a design solution which was later demonstrated by the AIDA method to be the minimum-cost solution.

Overall, then, there seems to be little reliable evidence to support an hypothesis that CAD systems are genuinely effective in improving the standard of design solutions. Even on their own terms of quantifiable measures of the built environment, their effectiveness has not been convincingly demonstrated. On other terms, such as the handling of the qualitative, the machine as yet offers no contest. There still remains an outstanding need for more rigorous research in the evaluation of computerized design.

LIST OF ILLUSTRATION

BIBLOGRAPHY

Alfred M. Kemper, Pioneers of CAD in Architecture, Hurland/Swenson, CA. 1985

Ackoff Russel, The Art of Problem Solving, Wiley , New York , 1976

Broad,E. , "Planning a Multilevel Car Park," Industrial Engineering, 1969

Bross , I. D. J. , Design for Decision , New York , 1965

Comba, P. G. "A Procedure for Detecting Intersections of Three-Dimensional Objects." Journal of the ACM, 1968

Dale/Lilly , Pascal Plus Data Structures, D. C. Heath , 1985

Donal Hearn , M. Pauline , Computer Graphics , Prentice-Hall , 1986

Farradane , J. , Information for Design , The Design Method , London , 1966

Feinberg, K. N. , "Use of Computer Programs to Evaluate Energy Consumption in Large Office Buildings." , ASHRAE Journal , 1974

Fitzgerald, D. , Cooling Loads by Computer: Some Programs Compared." Journal of the IHVE,1973

Gupta, C. L. , Spencer, J. , and Muncey, R. Use of Computer for Environmental Engineering Related to Buildings. Washington D.C. , 1971

Gerhard , Schmitt , Microcomputer Aided Design for Architects and Designs, Wiely , New York, 1988

Kiviat, P. J. , Vilaneuva, R. , and Markowitz, H. The Simscript II Programming Language. Prentice-Hall , 1969

Kuester, J. L. , and Mize, J. H. , Optimization Techniques with FORTRAN, New York , 1973

Levin, P.H. , Use of Graphs to Decide the Optimun Layout of Buildings. Architects' Journal , 1964

Lokmanhekim, M. Use of Computers for Environmental Engineering Related to Buildings. Washington D.C. 1971

Maclane, S. , "A Structure Characterization of Planar Combinatorial Graphs." Duke Mathmatical Journal. 1937

Markus,T.A. , Whyman,P. , Morgan,J. , Whitton,D. , Maver,T. , Canter,D. , and Fleming,J. Building Performance. Halsted Press , New York, 1972

Milne,M.A. , Computer Graphics in Architecture Design. Yale Cooperative Corporation , New Haven, Connecticut, 1969

Mitchell,W.J. , Steadman,P. , and Liggett,R.S."Synthesis and Optimization of Small Rectangular Floor Plans." Environment and Planning B. 1974

Okada,K. , and Kobayashi ,A Decission Method for Architecture Planning. Asakura Book Co. Tokyo , 1972

Pollack, M. , and Wiebenson, W. , "Solutions of the Shortest-Route Problem : A Review. Operations Research, 1960

Schriber,T. , A GPSS Primer. Ulrich's Book, Michigan, 1972

Seppanen, J.J. , and Moore, J.M. , "Facilities Planning with Graph Theory." Management Science,1970

Souder,J.J. , and Clark, W.E. , "computer Technology : A New Tool for Planning ." , AIA Journal , 1963

Thompson,J.D., and Pelletier,R.J. , Privacy vs. Efficiency in the Inpatient Unit.", Hospitals, 1962

Thompson,J.D. , and Goldin,G. , The Hospital : A Social and Architectural History. New Haven , 1975

William J. Mitchell , Computer-Aided Architecture Design . California , 1977