THE UNIVERSITY OF MANITOBA

A COMPUTER MODEL OF PRIMITIVE PLANT MORPHOGENESIS

A THESIS SUBMITTED TO
THE FACULTY OF GRADUATE STUDIES AND RESEARCH
IN PARTIAL FULFILMENT OF THE REQUIREMENT FOR
THE DEGREE OF MASTER OF SCIENCE

by
William George Simpson
December 1975

"A COMPUTER MODEL OF PRIMITIVE PLANT MORPHOGENESIS"

by

WILLIAM GEORGE SIMPSON

A dissertation submitted to the Faculty of Graduate Studies of the University of Manitoba in partial fulfillment of the requirements of the degree of

MASTER OF SCIENCE

ABSTRACT

A COMPUTER MODEL OF PRIMITIVE PLANT MORPHOGENESIS

by

William George Simpson

The hypothesis that cells in a plant are internally and genetically controlled and that morphogenesis is a cooperative activity of cells, rather than a centrally controlled one, is taken as a condition on which to construct a computer model of plant morphogenesis. The main parts of the model are (1) a method of reconciling different cell growth rates within a semi-rigid structure of cells and (2) a set of cell parameters and equations used to define the behaviour of individual cells.

The model, in the form of a computer program, is capable of producing very irregular structures which could be considered to represent very primitive aggregations of cells. The present stage of the model is described in detail. Analogies with real plant growth and possibilities for further development of the model are discussed.

ACKNOWLEDGEMENTS

This project benefited from the assistance and courtesy of members of the Institute for Environmental Studies, the Computer Centre and the Department of Botany of the University of Toronto, where much of this study was done. I am particularly grateful to Miss J. Retèl of the Institute for Environmental Studies and Dr. J. Dainty of the Department of Botany for their interest and help.

I am very grateful to Dr. P.K. Isaac of the University of Manitoba for many reasons, not the least of which is his patience.

I am grateful most of all to Elspeth Simpson, my wife, for her constant support and confidence during this work.

CONTENTS

CHAPTER 1

GENERAL INTRODUCTION

It is a wonderful cycle that causes a seed or a leaf broken from a plant to grow and eventually become a whole plant, like the one from which the seed or leaf fell. These cycles have come to be understood, partially, in two different ways. The first is as the result of evolution, the ancient and continuing series of changes, brought about by many causes, in which the structures and cycles of structures that are unstable break down and in which those that are more stable survive, for a time, to undergo more changes. This relentless series of changes has led, through uncountable numbers of cycles, to the life cycles known as plants and animals. The complexity of the growth and change making up one of these cycles is comprehensible if the cycle is regarded as just slightly changed from the cycle preceding it and this in turn as slightly different from its predecessor and each in turn slightly different back through countless cycles and changes. The second way of understanding the cycles is, so to speak, internally. What happens at each step of the cycle to produce the next step? Considering the extraordinary series of changes that have nevertheless resulted, for the moment, in a particular cycle, one can say that, at each step in the cycle, the next step is almost inevitable (barring misfortune). There is a machinery at work in the plant, continuing the cycle. This machinery has been found to be largely determined by molecules of DNA found in each cell of the plant. (Largely but not wholly, for reasons not critical in this discussion.) The long molecules of DNA are essentially stable, reproducible codes describing other molecules (proteins) and determining, through the proteins, the chemical processes of which the cell is capable. The DNA and proteins also control which part of the potential activity of a cell occurs at a particular time and place in the cell. This control depends on complex and highly specific enzymes, proteins which have the capability of controlling rates of biochemical reactions and which, in turn, have their actions enhanced or inhibited by the presence of other molecules.

(These molecules may be native to the cell or from another cell of the plant or from outside the plant.) The enzymes act in complex patterns at different levels of control and on different time scales. As well as producing other materials, they interact with the DNA, aiding or suppressing its reproduction and its decoding, as well as repairing damage to it. The variety of behaviour of these molecules and their products is enormous and is just beginning to be understood. This variety permits a great number of different types of behaviour in cells and hence in plants and plant cycles. In fact, it apparently accounts for all the plant cycles seen on earth. Thus, in a sense, it is possible to claim that DNA and its associates and their activities explain all plant cycles. However, the explanation, for even one plant, is very complex.

In this study, the attempt is made to understand part of a plant cycle in a way compatible with the genetic explanation. Although the details of the biochemical mechanisms are not considered in this study, they are part of the essential background to it. The study assumes that biochemical control mechanisms are available in the plant cell as required. For example, it might be convenient to assume that, if other factors are constant, the growth rate of a cell is proportional to the square root of the concentration of a certain molecule up to a critical concentration and is constant above that concentration. It is taken for granted in this study that <u>if</u> such a rule were useful to a plant, that species could have evolved a control mechanism to implement that rule. This assumption allows the study to concentrate on the relationship of cellular controls to the shape of the whole plant, rather than on the mechanism of the controls.

The approach used can now be described in general. It is to start with a set of assumptions about how cells grow and divide (expressed in mathematical equations) and to attempt to model the resulting growth and form of a plant made up of such cells. Within the framework of a computer program, the cells are allowed to grow and divide as decided by the equations. The computer program keeps a record of the number, size and positions of the cells and periodically

draws the resultant arrangement of cells (which is called, optimistically, a *computer plant*). After inspection of these plants, the equations can be modified and the process repeated. In the desirable, but by no means certain, event that this series of computer trials results in a computer plant that resembles a known plant, the experiment is regarded as having a positive outcome. This can give some understanding of the ways in which the shape of plants may be controlled by their component cells, both in terms of the behaviour of individual cells and in terms of cell interactions.

In the less desirable event that the computer plants cannot be said to resemble any known plants, it is possible to contend that they represent plants more primitive than any known plants. Although this seems logical, it is not useful by itself. However, if a series of computer plants were found, related through their equations, in which some plants were recognizable, to some degree, as real plants, then the series might be considered as a possible evolutionary series and might give some insight into evolution. This possibility is explored in Chapter 8.

The approach described above is quite general and potentially far-reaching. In order to give the initial stages of the study more definite limits and direction, a more specific objective was chosen. This objective is to produce computer plants which increase their number of cells by division at identifiable meristems and meristematic zones rather than by divisions distributed more or less equally throughout the mass of cells. If this objective is reached, the properties required by computer cells to enable them to form meristems may be analogous to the genetic coding developed at an early stage in plant evolution. (It is bound to be an *early* or *primitive* stage because of the relatively simple nature of the equations used. It is emphasized, however, that this study did not attempt to chose an specific stage of evolution, or any specific plant, as an object to be modelled.)

The general hypothesis of the study can now be stated, although further explanation and sub-hypotheses will be necessary. The hypo-

thesis is that the form and growth of a plant is determined by the cell growth and division and by the nature of the connections between its cells and, furthermore, that the growth and division of each cell is regulated within that cell and, furthermore, that a simple regulation scheme, based on a few properties of a cell and its immediate neighbours, is sufficient to produce a plant with a meristematic division pattern and, furthermore, that such a plant can be represented by a computer model.

This brief outline of the hypothesis and the method used to test it will be expanded in the following chapters. Chapter 2 is a survey of some literature related to the subject. The hypothesis is discussed in detail in Chapter 3. Various matters are included in Chapter 4 under the heading "evolutionary and environmental context". Chapter 5 deals with the cell parameters (that is, the properties of the cells) and with the equations used to calculate growth and division from these parameters. Chapter 6 is a detailed discussion of the method.

To summarize, it has been known for centuries that the seeds of a plant produce plants which are very similar to the parent. This is now known to be determined by the genetic material contained in each cell which controls the biochemical processes of the cell. It is clear from this that cell biochemistry must somehow control plant shape. This control, which operates over a range of magnitude from molecules to giant trees, is a very complex one. This project explores how such a control might operate in simple cases. The method used is to assume a specific set of cellular controls and attempt to derive, through a computer model, the appearance of the resulting plant.

CHAPTER 2

RELATED LITERATURE

Although the basic idea for this study was conceived independently, it has, not with great surprise, since been learned that it is not original. J. Bonner in 1965 suggested the principle of a systematic method by which plant cells might differentiate. It is a relatively formal scheme. It includes the idea of a cell testing whether it is an apical cell and whether it is inside or outside the cell structure. It also tests for the size of the mass of meristematic cells. He concludes generally:

"It has been repeatedly discovered that analysis of a physical problem in terms specific and detailed enough to permit programming of that problem for a computer gives us deeper insight into the nature of the problem itself. The same may well be true of the problem of differentiation. It will be of interest to develop such programmes in much greater depth and detail than the example considered here.

"Since no detailed programme or model of a developmental process has as yet been made, we do not yet know with certainty if the approach will be a fruitful one. It does, however, appear to me that the approach to understanding of the genetic-switching network by conceptual modelling holds promise; that by this approach we may hope in time to generate a sound theory of the nature of development."

Raup (1969) discussed the simulation of morphology, from the point of view of paleontology, in the following terms:

"There are three important justifications for the simulation of morphology: (1) Successful simulation provides confirmation of the underlying models as valid descriptions of the actual biological situation. (2) Unsuccessful simulation shows flaws in the postulated model and may suggest the changes that should be made in the model to correct the flaws. (3) Non-occurring forms, perhaps intermediate

between actual species, may be simulated and this may lead to a better understanding of the relationships between the real forms and of evolutionary or ecological processes that may have transformed one into the other. In other words, the simulation process enables the paleontologist to look at the forms that *could have* evolved as well as those that *did* evolve.

"The ultimate objective of a model used in simulation is to replicate the genetic structures, or genetic "program", used by an organism. This goal is rarely, if ever, attained and the closeness of a given model to the goal is difficult or impossible to test. Still, the attempt to mimic genetic programs provides a useful rationale in selection of models. The best practical criteria presently available for justifying a given model are its relative simplicity, that is, requiring relatively few parameters to describe morphology, and its applicability to a large and varied array of organisms, including ontogenetic stages."

The nature of the cell wall was discussed by Esau (1953):

"Cell walls show varying degrees of *plasticity*(property of becoming permanently deformed when subjected to changes in shape or size), *elasticity* (property of recovery of the original size and shape after deformation), and *tensile strength* in relation to their chemical composition and their microscopic and submicroscopic structure. Plasticity of cells is well illustrated by their permanent extension in certain stages of growth of cells in volume; elasticity, by the reversible changes in volume (30 percent or more in mesophyll cells) in response to changes in turgor pressure. Notable tensile strength is characteristic of mechanical cells, particularly of extraxylary fibers of monocotyledons and dicotyledons.

"Although the cells in the meristematic tissue are generally closely packed, during tissue differentiation the intimate connection between walls of adjacent cells may be partly broken, with the appearance of intercellular spaces. The most common intercellular

spaces result from a separation of cell walls from each other along more or less extended areas of their contact." (References omitted.)

Bonner (1974) considered the basic processes involved in development. From his conclusions:

"In large measure the developmental control of this proportionality (of form) is achieved by chemical signals. Where the living units are separate, as in social insects, certain stages in the life cycle of cellular slime molds, and various fungi, there are pheromones or other chemical messengers that act at a distance. In multicellular organisms the mechanisms of transport of the signals (hormones and inductions) include diffusion, cell movement, active transport, and polar movement. From the very fact that the cells are touching, the chemical messengers can be passed in a greater variety of ways, and the possibilities of control are thereby increased. In the smaller level of single cells we begin to see form-making phenomena that could best be interpreted in terms of a combination of macromolecular interactions, active transport, and chemical signals diffusing over short distances. Note that the order of these three levels of organization is in a rough sense the reverse of that which occurred during evolution."

Many points raised by Wardlaw (1952) are very relevant to this study. A few are quoted from various parts in his book:

"This was a recognition of the fact that the changes in configuration undergone by an organ or tissue, e.g. the vascular system, during development, might be referable not so much to intrinsic factors as to extrinsic factors, e.g. general physiological, physical or other factors which become incident with increase in size."

"These contemporary studies are, and must be, essentially dynamic in outlook. Furthermore, there must be an integration of the data of morphology, physiology and genetics. The hereditary constitution of a species, itself a small fragment of the evolutionary picture, is conceived as being subdivisible into genes. These are involved in all

developmental processes: they find expression in the action of chemical substances which, in conjunction with various factors, are directly or indirectly effective in morphogenetic processes and culminate in the distinctive configuration of the adult. It will be of the utmost importance to accumulate information on genic action in morphogenetic processes. These several investigations should do much to enhance our knowledge of form and structure in plants. The conclusions and inferences derived from this multi-aspect approach may, with due caution, be applied to the organisation shown by fossils from different geological horizons. In essentials, our interpretation of fossil plants is based on our understanding of living ones and, therefore, if an adequate analysis of the factors underlying the organisation and changes in organisation of living plants can be achieved, a fuller and more exact interpretation of the event indicated by the fossil record should also become possible, though it can never be absolute."

"Where a particular form or structure can in the main be referred to the action of physical factors, and to physiological processes common to autotrophic plants in general, it can have little phylogenetic significance, however interesting and distinctive the form or structure may be. The shoot or axiate type of organisation, found in the gametophytes of mosses, in the sporophytes of pteridophytes and seed plants, and perhaps even among the larger algae, may well be of this nature."

The classic "On Growth and Form" by D'Arcy Thompson is, on this subject, more helpful in general than in specific terms:

"The search for differences or fundamental contrasts between the phenomena of organic and inorganic, of animate and inanimate things, has occupied many men's minds, while the search for community of principles or essential similitudes has been pursued by few; and the contrasts are apt to loom too large, great though they may be."

"The waves of the sea, the little ripples on the shore, the sweeping curve of the sandy bay between the headlands, the outline of the hills, the shape of the clouds, all these are so many riddles of form, so many problems of morphology, and all of them the physicist can more or less easily read and adequately solve: solving them by reference to their antecedent phenomena, in the material system of mechanical forces to which they belong, and to which we interpret them as being due. Nor is it otherwise with the material forms of living things. Cell and tissue, shell and bone, leaf and flower, are so many portions of matter, and it is in obedience to the laws of physics that their particles have been moved, moulded and conformed. Their problems of form are in the first instance mathematical problems, their problems of growth are essentially physical problems, and the morphologist is, *ipso facto,* a student of physical science."

BONNER, JAMES, 1965: The Molecular Biology of Development, Clarendon Press, Oxford.

BONNER, JOHN T., 1974: On Development (The Biology of Form), Harvard Univ. Press., Cambridge, Mass.

ESAU, K., 1953: "Plant Anatomy", J. Wiley & Sons, New York.

RAUP, D.M., 1969: Modelling and Simulation of Morphology by Computer. Proc. North American Paleontological Convention: 1969, Part B, pp. 71-83.

THOMPSON, D'ARCY W., 1917, 1942: "On Growth and Form". (Quotation is from the abridged edition, 1961, Cambridge Univ. Press.)

WARDLAW, C.W., 1952: Phylogeny and Morphogenesis, MacMillan, London.

CHAPTER 3

DISCUSSION OF HYPOTHESIS

The main hypothesis was stated in Chapter 1. Now, each part will be restated and discussed in detail.

Control at cell level

The first part of the hypothesis is that the specificity shown in the form and growth of a plant is determined fully by the growth and division of its cells and by the nature of the connections between cells and, furthermore, that the growth and division of each cell is regulated within that cell. This means that plants consist of cells and cell products. It also means that there is no supra-cellular control mechanism operating in the plant. This part of the hypothesis seems to follow from genetic theory. It is emphasized here not because it is in question but because it is so remarkable and yet apparently true. The cooperation between cells which is so apparent in the outward symmetry (Fig. 3-1) and internal organization of the plant is achieved not through a central control but by each cell following explicit instructions contained in its own fabric. The necessary integrative mechanisms act through physical and chemical interactions (or messages) between cells which may, in effect, signal a cell to read a different part of its instructions. In thinking of the highly sophisticated messages and integrative mechanisms used by higher plants, the distinction between centralized control and control residing in each cell may seem to become lost or to involve just choice of terminology, rather than matters of real substance. In primitive plants, however, the autonomy of individual cells was almost certainly much more obvious (though possibly no more real) than it is in modern ones and, in this study, this autonomy is emphasized and protected as much as possible. (A partial compromise of the principle of autonomy is discussed in Chapter 6 under the heading "Reconciliation". The autonomy of cells in primitive plants is considered in a speculative way in Chapter 8.)

This hypothesis implies that the computer model must contain instructions determining the growth rate and division rate of each cell. A cell is instructed to grow, as nearly as it can, at a constant rate until it reaches a certain size and then to stop. A cell is instructed to divide if and only if it meets several criteria. It is emphasized that the instructions given the cells are extremely simple and straightforward. These instructions are expressed in quantitative terms in Chapter 5.

In order to produce a working model, a number of sub-hypotheses and assumptions of several kinds are necessary. The attempt is to frame them all in accordance with the main hypothesis. For example, a sub-hypothesis that the plant will not grow taller than seventeen feet would be inappropriate since it is not a control at the cellular level. On the other hand, a sub-hypothesis that a cell will not grow if no nutrients reach it would be appropriate and, furthermore, in conjunction with other hypotheses, might restrict the height of the plant in a reasonable way.

This hypothesis of cellular control raises the question whether the instructions for growth and division are genetic or direct (or in the language of Chapter 2, intrinsic or extrinsic). As regards the model, this question is not relevant. The instructions are arbitrary and their mode of implementation need not be thought of as genetic or direct. This same question is considered for primitive plants in Chapter 8. (As with the earlier question of autonomy, this question, when applied to higher plants, becomes more complicated and largely a matter of terminology. It is probably true that for higher plants, cell control is partly direct but mostly genetic.)

Meristematic division

The second part of the main hypothesis is that a simple cellular regulation scheme, based on a few properties of a cell and its neighbours, is sufficient to produce a meristematic division pattern. The key words are "simple" and "based on a few properties" for without

them the proposition is obvious, as witness most higher plants. It is possible that with a cellular control scheme of sufficiently great complexity, the growth of a plant through its complete life cycle could be modelled. However, this would be most tedious and unlikely to succeed at the present state of knowledge. The attempt in this study is to model some aspects of a real plant with a few controls. These controls will be as simple and intuitively reasonable as possible. This is not because it is assumed that plants actually operate in a simple, intuitively reasonable way but because it seems potentially more instructive to work this way.

The choice of meristematic division as a specific goal in the study was made for three principal reasons. It is a characteristic of higher plants and, therefore, probably started early in their evolution and, therefore, might be expected to have occurred in some quite primitive plants. Second, a computer plant will be quite different in appearance with meristematic activity than without. That is, it is a fairly clear distinction. Third, it seems plausible that if each cell of a plant can sense if its neighbours are dividing and divide only if they are, then concentrated centres of division should occur. In other words, it seems as if it might be amenable to a simple mathematical treatment.

Computer model

The last part of the hypothesis is that a plant of this general type can be represented by a computer model. If the previous parts of the hypothesis are correct this does not seem to be a fundamental question but rather a test of modelling ability and skill (and good fortune) in choosing methods, equations and numbers. The complexity of the problem requires an experimental or trial-and-error approach, in which the results of one attempt are analysed to give clues as to how the model might be changed to give better results on the next attempt. (The analogy between this process and evolution is considered in Chapter 8.)

CHAPTER 4

DESCRIPTION OF PROTOTYPES; EVOLUTIONARY AND ENVIRONMENTAL CONTEXT

This chapter attempts to describe details of the type of plants visualized as prototypes for the computer plants. It discusses how these details are treated (or ignored) in the model. Some more speculative paths related to this discussion are explored in Chapter 8.

State of evolution

This study is concerned, as its main hypothesis, with multi-celled plants governed by cellular controls. The simplicity of the controls limits the study to *primitive* plants. If more complex controls were used, a similar type of model might well produce more sophisticated plant development. It is not the inherent nature of the model which limits it to primitive plants, but rather the choice of controls. It is noted that the structures being produced in the model might be so simple or so unlike modern plants that they could not be considered as plants at all. Despite this possibility, the term "computer plants" is used for the products of the computer program. In the study, interaction between cells is assumed to take place through chemical messages between *adjacent* cells. This interchange of messages and the resultant cooperation of cells seems to justify referring to these structures as *cooperating multicelled structures*. It may not justify calling them organisms. One possible interpretation of this situation is that of dealing with a group of cells starting the evolutionary stage of learning how to cooperate to their mutual benefit. The question of the place of these plants in evolution is discussed in Chapter 8 at greater length.

Ecological niches

The model is not highly specific in regard to the ecological situation that its plants occupy. It can represent land or water plants, with the following qualifications. The model has no constraints

as to direction of growth. For example, there is no equation representing the direction of gravity. Also there is no simulated light source in the model to add a directional effect. Without such directional guidance, the model probably applies better (in this respect, at least) to free-floating water plants which, first of all, are close to the density of water and, therefore, little affected by gravity and secondly, are free to turn so that all sides are exposed to light. It may apply, though less well, to small primitive land plants that are both small enough that gravity has no important direct or external effect and primitive enough that they have evolved no major internal responses to gravity (if such a state of primitiveness is possible). It would be possible to add to the calculation process a term taking gravity into effect with which plants could be made to lean or to grow downward as much as was desired. Then, however, there would arise the much more difficult problem of modelling the defiance of gravity that is shown by land plants. It seems advisable not to enter this complex area before it becomes necessary to do so and the field of study represented by water plants and small land plants is not a restrictive one. One effect of the lack of gravity is that there is no part of the model which calculates the required strength of structures. Because of this there is no necessary tendency for the lower part of the plant to be thicker or stronger than the upper branches (except that the lower part presumably started to grow first and has had more time to grow larger and stronger).

The model is versatile enough to represent both free-floating (water) plants and plants anchored or rooted in a fixed location (on land or in water). The base cells, (discussed in Chapter 6) which are essential to the calculation method, can be thought of as representing a rigid surface to which the plant is attached, or, in the case of a rooted plant, below which the root system has developed (although there is no specific representation of the root system in the model). If a large number of base cells are used, they will influence and constrain the growth to the extent that the plant is unlikely to grow through the plane of the base cells (although it is possible for it to

do so). If the parameter *nouriture* (which represents nutrition conveyed to the plant from the base) is used with many base cells, this combination clearly corresponds to a plant drawing food through some sort of root system. On the other extreme, if only a few base cells are used (three is the minimum possible), they will have little effect on the plant which is then quite free to grow around them. (They simply provide a fixed position relative to which the positions of other cells are calculated.) In this case, the plant is free to grow to, for example, a spherical shape without external constraints (that is,a free-floating plant). The parameter *surface* might be interpreted as the availability of nutrition through the outer surfaces of the plants in this case. A combination of nutrition through a root system and from the immediate surroundings (air or water) is also possible in the model.

There are many environmental factors which influence the shape of a plant. These include the method of propagation, the supply of nutrients, water supply, amount and type and direction of light, diurnal and annual cycles, atmospheric conditions, wind or wave action. Each of these factors has many different components and many other factors might be listed. Nevertheless, it is true that the species of a plant specimen is usually recognizable despite variations in any or all of these factors. Of course, if the variation is too extreme, the plant may not grow at all but if the combination of factors is suitable to permit growth, the genus, if not species, will normally be recognizable. This is true no matter how the plant is propagated. Since in this project we are concerned with the effects of *internal* control mechanisms, these external environmental factors are ignored if possible and, if not, they are brought into the model in as unobtrusive a manner as possible. For example, the supply of light to a cell may be considered to be included in the parameter SFC. This simple interpretation ignores both the shading of the lower parts of a plant by the upper part and the distinction between the sunny side and the shady side of a plant. Having disposed of a great number of environmental factors in a similar summary manner, it remains to be noted

that one effect of these neglected external influences is to introduce some irregularity into the developing plant. If the computer program produced plants which were very symmetric and regular, they would not appear natural. It seems reasonable, because of this, to insert a variable degree of randomness into the equations which can be used if necessary to disturb the mathematically regularity of the computer plants.

Although environmental influences are ignored as nearly as possible, some of them, could be incorporated into the model at a later stage.

In summary, the prototype for the computer plants is a primitive multicelled plant in a very sheltered environment. There is cooperation between cells but nothing like the degree of integration of a modern plant. To conform to the main hypothesis, it has developed meristems. It may be either a water plant, free-floating or anchored, or a small land plant, depending on which parameters are used in the cellular controls.

CHAPTER 5

CELL PARAMETERS AND EQUATIONS

Control of cell growth and division is hypothesized to be based on "a few properties of a cell and its neighbours". We now define a set of cell parameters which are used in the model to represent the cell properties. To be consistent with the hypothesis and the modelling process, these must meet several criteria. They must be biologically meaningful if the resulting model is to have any biological significance. Related to this, they must be consistent with biochemical control of the cell. That is, the biochemical apparatus of the cell must include these parameters in some way. This does not mean that the parameters will be defined directly in biochemical terms. They must be mathematically definable and calculable in the computer program. They must depend, for each cell, only on it and its immediate neighbours; that is, local control.

Justification for the choice of parameters can only come, in this type of study, by deriving significant results from them. The discussion of each parameter presented here is intended to explain why they were chosen for study.

We now define the parameters. This list includes all parameters presently calculated in the model, though some of them have not been used. Many others could be defined but much more can be done using only the present parameters and variations of them. For each parameter, we discuss the following points. First, a qualitative and quantitative definition is given. (Details of the calculation are in the computer program in Chapter 6, but all essential points are included here including some practical problems regarding their use and behaviour.) Second, why, from a biological point of view, is this parameter expected to affect growth and division; that is, why is it biologically *relevant*? Third, how might this parameter be realized in biochemical and physical terms; that is, how is it biologically *feasible*? It is emphasized that there is no intention of trying to specify in detail

the underlying biochemical mechanisms but merely to make a reasonable case that these could exist.

Linear dimension parameter - LIN

Cell size is well-defined and straightforward in principle but because of the nature of the model it is convenient to calculate it in a somewhat peculiar way. The position of each cell in the structure is specified by the three coordinates (X, Y and Z) of the *centre* of the cell. The linear dimension parameter of a cell, LIN, is the average of the distances between the centre of the cell and the centre of each of its neighbours. (In the rest of this thesis, LIN is used in this specific sense and "size" in a general sense but they refer to the same basic property.) Because of the nature of the definition, the total of the volumes (LIN cubed) of all the cells of a structure may not equal the actual volume of the structure. But if one compares two plants, identical except that one is twice the linear size of the other, then LIN for each cell of the larger will be twice LIN of each cell of the smaller. This relative accuracy is adequate for the purposes to which LIN is put. The distances between cell centres are the more fundamental distances in this model and, compared to them, LIN is a statistical, approximate parameter but it is convenient to use. The main calculation of cell position is done using actual centre-to-centre distances and LIN enters this calculation only in a secondary role in which its relative nature is quite adequate.

The values of LIN in actual studies so far have ranged from about 0.1 to 2.0. Smaller cells cause calculation problems; to prevent these, the minimum size cell produced was limited by the minimum size cell allowed to divide. Cells larger than 2.0 cause no calculation problems; this upper size limit, determined by the growth equation, is an arbitrary one. The size range established by these constraints should be regarded merely as a framework relative to which to experiment. The relative nature of this parameter will become clearer in later discussion.

In a third sense, LIN is relative. No actual length units are specified. The plant shape and its number of cells is modelled but not the cell size. With this limitation, LIN is suitable for its role in the model.

It is clear that cell size is both a relevant and a feasible parameter. There must be some size below which a cell is incapable of division. Size can affect growth rate in many ways. Consider why a cell stops growing at a certain size. It may be because some material produced in limited quantity at cell division becomes used up or because large size introduces great inefficiency into the growth mechanism and stops further growth.

Cell AGE

This parameter is defined for each cell as the time elapsed from the division of the parent cell which produced it. The program runs in loops or time-steps. At each time-step, the AGE of new cells formed by division is set to zero and the AGE of each cell is increased by one. No real time units are specified. AGE is relative to other factors in the model. For example, if cells are not allowed to divide before AGE 20, then 20 is the minimum generation time and this gives a type of scale to the model. If the growth equations are such that a cell can grow from minimum to maximum allowed size in 100 time-steps, this gives a different meaning to time and AGE. The equations must be adjusted so these meanings are somehow compatible. For example, if cells are not allowed to divide with AGE less than 20 or LIN less than 0.5 and if, because of the growth equations, cells do not reach LIN = 0.5 until AGE 40, then the generation time is 40, not 20. Also size at division affects the size of the new cells formed and hence the time required for new cells to grow large enough to divide again. This *parameter interaction* is important in the model. Other parameters cause further complications. Stating specific time and length units would be meaningless unless this was done after the study was completed.

AGE is a relevant parameter since a young cell requires some time to "get organized" before dividing and a very old cell can become incapable of further growth or division (under normal circumstances, at least). AGE could affect cell biochemistry through a substance produced at cell division time which subsequently decayed so that AGE is inversely related to the amount of substance remaining. Thus the parameter is feasible.

Surface effect - SFC

This property called SFC is dependent on the position of the cell in the cell tissue. It is highest for apical cells, intermediate for surface cells and lowest for cells internal to the plant. It is defined as the vector sum of the distances from the cell centre to the centre of each of its neighbours divided by the scalar sum of the same distances (and then multiplied by 10). Without going into great detail, note that if all the neighbours of a cell are in the same direction from it (Fig. 5-1), then it is at an apex. In this case, the vector and scalar sums are nearly the same and SFC is close to 10. If the neighbours of the cell are all around it, then the vectors cancel each other and SFC will be low (Fig. 5-2). (If the neighbours were perfectly symmetrically distributed, SFC would be zero. This is unlikely.) Actual values of SFC are shown in Fig. 5-3.

This parameter is intended as a measure of the degree to which the cell is exposed to the external environment. Many factors in the cell biochemistry could be changed by this exposure so that SFC is an omnibus parameter. Differences in SFC could represent differences in the light level at the cell, in the rate of exchange of gases between cell and environment, in the degree of interaction with other cells and in the water content of the cell. (In a dry environment, cells with high SFC would be dryer than those with low SFC.) The dependence of cell biochemistry on these factors and indirectly on SFC seems beyond question. We conclude that SFC is a feasible parameter (even though the nature of the dependence is obviously potentially very complex). Since, clearly, differences in growth and division rates between cells of high and low SFC would produce differences in form, SFC is a relevant parameter.

Division rate - MER

The parameter *merism* or MER is related to the division rate of the neighbours of a cell; that is, it is a measure of meristematic activity. It is defined as the average for all neighbour cells of

$$1 / (AGE^3/1000 + 1)$$

This is largest for cells with neighbours that have recently divided and decreases as the neighbours get older. The problem of parameter interaction is further complicated by MER.

This parameter is obviously relevant to meristem division. It is feasible through a mechanism such as a substance produced during cell division which diffuses or otherwise moves to neighbouring cells carrying the message of division. It decays corresponding to the decrease of MER.

Growth rate - WAX

The *waxing* or WAX of a cell is defined simply as the average growth rate of its neighbours (expressed as percentage growth per timestep). This parameter seems relevant since it implies growth can either stimulate or depress other growth or division. But this has not yet been tested in the model to see if it is a useful parameter. Waxing is feasible through a substance produced during cell growth and at a rate proportional to the growth rate which then carries the message to surrounding cells in a similar way to that hypothesized for merism.

Nutrient supply - NOU

The parameter *nouriture* (or NOU in the computer program) represents a nutrient and water source at the base of the plant (a root). Nutrient is visualized as being transported through the plant from its base. It is calculated as follows. The NOU of the base cells is constant. The NOU of any other cell in the plant is equal to that of the nearest cell below it less a certain amount. This amount depends on the size of the cell and on its growth rate. That is, NOU represents nutrients moving through the plant and being used so that the supply decreases with distance from the base. This can be regarded either as

diffusion out from a source or as a type of polar transport. This parameter should be useful in the model but it has not been used to date. NOU is obviously relevant to higher plants and might be to primitive ones. It is feasible in the narrow sense that nutrient supply can affect cell biochemistry. Whether it is feasible in the sense that the rate of decrease of NOU throughout a structure is appropriate is not known yet. Much further work could be done with this parameter or one like it.

Further discussion of parameters

One of the requirements of the parameters is that they are local in nature. LIN and AGE are dependent only on the cell for which they are defined. SFC, MER and WAX depend only on the position and activity of the neighbouring cells. NOU is not properly a local parameter since, although it is calculated for each cell from the nearest neighbour, it depends on the activity of many other cells in the plant.

We consider briefly the evolutionary status of these parameters as cellular controls. Size and AGE are applicable to single cells (though the method of calculating LIN is not) and they are very fundamental and probably applied to the earliest cells. SFC is also very fundamental and probably affected the earliest clusters of cells. (In a sense, the SFC concept applies to single cells since a single cell in a very restricted space could be thought of as having a low surface effect and would behave differently than an unconfined cell.) MER and WAX are perhaps less primitive since they imply material exchange between cells. NOU as presently defined does not apply to very primitive cell structures (though a simpler form of transport or diffusion of nutrient would).

During growth and division in the model, the cell parameters change. These changes can be interpreted as *differentiation*. For example, a small rapidly dividing cell (part of a meristem) stops dividing, grows larger, and becomes a stable part of the structure. This seems like a limited type of differentiation. When a cell undergoes these changes, the equations of growth and division change or,

more precisely, different parts of these equations apply to the changed cell. Thus the equations lead to differentiation and with this may come changes in control representing different parts of the genetic material being expressed. In this way the model is fundamentally similar to nature.

If a parameter can be shown through this model to be significant for control, then it is likely that biochemical species roughly corresponding to the parameter have been evolved by cells; thus the parameters may be identifiable as specific biochemical species. By describing in detail the variation of the parameters, the study may suggest critical experiments with real plants. (In the computer program output, the parameter values are periodically listed for each cell so that comparisons can be made.) However, the parameters used are primitive like the rest of the model and may be only distantly related to biochemical species in modern cells.

Growth and division equations

Although the equations governing cell growth and division are essential to the model, they are a small part of the total size of the program. They can be compared to DNA molecules which, though a small part of the cell, require (in both senses) the rest of the cell to fulfill their potential. In the same way, the rest of the program is required to demonstrate the potential expressed in the equations. The workings of the model are described in detail in the next chapter. Here we examine the equations themselves.

The ultimate point of the study is to learn the potential of different equations. Because of this, many sets of equations have been tried. However, one growth equation was used for most of the study so we will describe it. Three different division equations will be described representing stages in the study.

Growth equation

The growth equation tells how much a cell is to grow in one timestep, expressed as a multiplicative factor. (Growth = 1.02 means two percent growth in linear dimension. The corresponding volume growth would be 1.02 cubed or 1.0612 or about six percent.) The growth equation used and a graph of its shape are in Fig. 5-4. The main points to note are that very small cells (LIN less than 0.07) have the highest percentage growth rate (1.20 or twenty percent), large cells (LIN greater than 1.22) have no growth, and the variation of the rate between these extremes is such that the growth as an absolute rate (rather than a percent) is roughly constant. No special properties are claimed for this equation; it was chosen through experiment as one which produced reasonable growth in a set of cells. Thereafter, this equation was used consistently while different forms of division equation were used. Thus we have had little experience with the effects of different growth equations. Figures 5-5, 5-6 and 5-7 illustrate the growth produced by this equation and may be compared to Fig. 5-8 and 5-9 showing the results of a constant growth rate of 1.02 until LIN = 10. (Details necessary to the understanding of the figures are together preceding the figures. Plotting conventions are explained early in Chapter 6.)

Division equations

In this model, a division takes place in one timestep; there is no provision for extending the process over a longer time. Thus the division equation has to determine whether or not a cell divides, not how much or how fast. Three main methods of doing this have been tried. Since the first two were dropped in favour of the third, we outline them without going into full detail. In the first, an equation was used to calculate the *probability* of cell division from the cell parameters. Then a random-number mechanism in the computer program is used to decide, with this probability, whether or not the cell actually divides. This method can be made to work but it creates awkward problems in writing suitable equations.

In the second method, a *class* of cells is defined by a set of parameter ranges. If *all* the parameters fall in their ranges, the cell is in the class. A class and a probability are specified and cells not in the class cannot divide. For those in the class, the random mechanism and the fixed probability determine whether or not the cell divides. The third method is an elaboration of this. Two classes and two probabilities are given. If the cell fits a class then it has the corresponding probability of division. If it doesn't fit either class, it cannot divide. This method could be extended to give very specialized control of division by increasing the number of classes or in other ways. If the probability for a class is 1.0 (that is, certainty), then a cell will always divide if it falls in that class. In this case, no randomness is involved. With high probabilities, smaller classes are needed to prevent excessive division. Many of the later runs in the study were done in this way. For example, in run #153, only cells in the class

LIN 0.5 to 0.8, MER 0.02 to 0.13, SFC 7 to 8.5 and AGE 5 to 100

divided. (This proved to result eventually in too much division and a very confused "computer plant".)

Note that although the method of determining division is not an equation, we continue to refer to "growth and division equations" for simplicity.

To summarize, the growth and division equations are vital to the study but very unimpressive in themselves. Their potential is expressed through the computer program discussed in the next chapter.

CHAPTER 6

COMPUTER PROGRAM AND METHOD

In this chapter, we discuss each part of the program in order, explaining what it does and, in outline form, how it does it. A complete description of the study would include details of the many steps in development of the program to the present form. This would be long and tiresome. Instead, in the discussion, the development of the most important features is given briefly. A listing of the computer program is included in this chapter. It is the FORTRAN language. Following it are technical notes regarding the program. At the end of the chapter is a note on the method of experimenting with this program. In both the discussion and the technical notes we deal with the parts of the program in their normal order except that the last main section of the program, the routine "Plot Plant" used to draw the computer plants, is discussed first. An understanding of what it does will make the figures referred to later more meaningful. (All "plants" referred to in this chapter are computer plants.)

The plots (for example, Fig. 6-9) are intended to show the general shape of plant produced and, when studied carefully, to provide more detailed information about the plant. They show a side view (Z axis up and Y to the right normally). Only the lines joining the centres of neighbouring cells are plotted. There are lines from each cell to at least three neighbours. The centres of cells can be recognized as the points where the lines meet. Since the cell walls are not drawn, the outside surface of the plant is not shown. If it was, it would be larger but the same general shape as the plot. In fact, it cannot be drawn with the information available in the model without making assumptions about cell shape. Figure 6-1 shows one of the computer plots with an outline superimposed showing the outer edges of the plant (assuming the cells are spherical). When looking at the plots, the fact that they are somewhat smaller than the plant they represent is not a serious problem.

Scale, date, run number, number of cells and number of loops are printed on each plot. The scale factor used is chosen to fit the plot into an 8-1/2 by 11 inch area. In comparing drawings it is important to note the scale factors unless only shapes are being compared. If two plants are the same size in the plots but one is scale factor 0.5 and the other 1.0, the one with factor 0.5 represents the larger plant since it is plotted at half its calculated size. With scale 0.5, a plant of calculated height 10 would be plotted 5 inches high. The run number identifies the computer job in which the plant was produced. The number of loops is the number of timesteps that have occurred. Run number together with number of loops uniquely identifies a plot. A plot with zero loops is the initial plant used to start the computer run. Number of cells includes both active cells and the base cells (discussed later) which do not move or divide. Increases in the number of cells during a run represent division of active cells.

Calculations

The first section of the program specifies the storage requirements and the information to be stored about each cell. This is a suitable place to say just what is known and what is not known about the cells in the computer model. Fundamentally, the X, Y and Z coordinates of the centre of the cell, the AGE, and the growth rate are known. All the other parameters are calculated from this data. Only the cell coordinates ("cell coordinates" means the X, Y and Z coordinates of the centre of the cell) and the list of neighbours (which is obtained using cell coordinates) are used by "Plot Plants". There is no information about cell shape. Therefore, the measure of cell size (LIN) based on cell coordinates is only approximate.

The section "Read Parameters and Controls" reads those numbers which are changed frequently during experimentation with the program. The important ones are discussed in the section of the program where they are used.

In "Set-Up Initial Plant", a set of cells, called the *initial*, is specified which is used as the starting point of subsequent growth. It is built up as a series of layers of cells. The number of layers and the spacing, AGE and number of cells in each layer can be independently specified. This allows considerable variety in the initial. Also, a randomness factor can be used to make the initial less regular. If it is zero, the cells on each layer form a square grid. The bottom layer cells are the *base cells*. During the running of the program, these do not change but they are necessary to the calculation (as a set of fixed reference points). The other cells are called *active* cells. "Set-up Initial Plant" is used only once in a computer run. The rest of the program forms the main loop which can be used many times.

The section "Reorder" has two vital functions which are carried out together. The first is to arrange the cells in order starting with the base cells (which are always first). The desired order is one in which cells closest to the base come first and those at the top of the plant (or the tips of branches) come last. That is, they are in order of distance from the base *measured through the plant,* not straight line distance. Another way of expressing this is, if cells at the end of the order were one by one removed from the plant, the plant would shrink but would remain in one piece, but if cells were removed that were early in the order, the plant would start to fall apart. The second function is to obtain for each cell a list of the nearest neighbours which are before it in order. Reordering is done as follows. The first active cell in order after the base cells is the one which is "nearest" to the base cells. The next in order is the one "nearest" to the base cells and the first active cell. The third in order is the one "nearest" to the base cells and the first two active cells and so on throughout the whole plant. The technical meaning of being "nearest" to a group of cells is explained in a technical note. During this process, a record is kept for each cell of the nearest neighbours which precede it in order. These are called its "neighbours below". This is important in understanding the rest of the program. "Neighbours below" a cell may actually be higher, that is, have a larger Z coordinate, if

they are on a downward growing branch. The term "neighbour above" is also used. Cell A is a "neighbour above" cell B if B is a "neighbour below" A. (There is no such thing in this program as neighbours side-by-side. "Reorder" always assigns a definite order to two cells close together and the first in this order is "below" the other.) A cell always has neighbours below it and unless it is a prominent cell on the surface or an apical cell, it will have neighbours above it.

To understand why this process is necessary it is helpful to explain earlier methods used to order the cells. First, they were ordered according to the distance from the plane of base cells; from the ground up, so to speak. However, in the case of a branch growing sideways, cells at the end of the branch could come before those at the base and this order is not correct. The next method used was in order of distance upward and outward from the centre of the base. This works as long as the plant grows upward and outward but in a downward growing branch, the order is wrong. Other more complex methods in which downward growing branches were ordered by a different method than the rest of the plant were tried before the present one was devised. This method deals with branches growing in any direction. Fig. 6-2 shows the cell centres in a vertical section through a computer plant. The exact order calculated using the method is not shown but the first several cells in order are marked "1", the next group "2", and so on. The order is upward and outward in a reasonable way.

The section "Calculate Cell Parameters" is a straightforward calculation of the parameters as defined in the previous chapter. There is one important point to note. The parameters SFC, MER, WAX and LIN depend on neighbour cells. This includes neighbours both above and below (in other words, all around). The parameters MER and WAX are visualized as depending on substances which move from one cell to another. The implication of using neighbours both above and below is that there is transport in all directions (not polar transport).

The section "Calculate Growth and Division" is a straightforward application to each cell of the growth and division equations. Also

the frequency distribution of the parameters and the growth rate is obtained and printed.

In the section "Implement Divisions", those cells which were determined by the division equation are actually divided. In this process, their AGE is set to zero. One new cell retains the coordinates and number of the parent cell. The other is given a new number and a position between the parent position and the average position of the four nearest neighbours below (four unless there are only three such neighbours). Thus, very simply, there are two new cells where there was one older one before. The positions of the new cells relative to the neighbours below are shown in sketches (Fig. 6-3). They fall in line with the centre of the four cells below. One new cell is above the other or further from the base of the plant. Although there is considerable variation depending on which are the four nearest neighbours, the position after dividing may be described roughly as one above the other rather than side-by-side. To explain this in a different way, if the cell that divided was prominent on the surface of the plant, then the division pattern would be periclinal rather than anticlinal. That is, one new cell would be more exposed at the surface than the other one. This is a general pattern in which there can be considerable variation.

The next section "Reconciliation" is the most complex and interesting part of the program. Developing it was the main work of this study.

Consider the situation in the model after all the preceding sections have been run. The previous positions of the cells are known but most of them have now grown and some may have divided. How has the shape of the structure changed because of this growth and division? Consider the following cases. If there is only one active cell resting on the plane of base cells, then if it grows, it will move upward by an amount proportional to its growth rate. In a structure of several layers of cells, if all the cells grow by the same amount, the structure will expand by that amount. If only the lower cells in the structure

grow, the upper cells will nevertheless move because the lower cells will push them. The simpler case is when only the upper cells grow since the lower cells are not affected. The latter is frequently the case in real plants. The cells in the apical meristem are dividing. The cells below this are growing. The lower part of the plant is relatively inactive in growth or division, disregarding lateral meristems which don't affect this analogy. The fact that this is a simpler case involving less relative movement of cells is another reason why meristematic growth was chosen as an objective in the model. Perhaps it is also a reason why plants chose meristem growth.) If the cells in one side of the structure grow, then it will bend to the other side (for example, Fig. 8-2). Considering all these different types of interaction, a general approach was developed for calculating new cell positions and the new shape of a structure due to cell growth. The following general description of this approach is quoted from the proposal prepared prior to the study.

"The method to be tried has the disadvantages that it is arbitrary and does not permit the shape of individual cells to be studied. It has the advantages that it is possible to calculate, allows growth to be governed by internal cell controls and preserves to a large extent the relative positions of cells. The size of a cell is specified by its volume. At each time step in the growth, after the new set of cells have been calculated, another step occurs in which each cell will if necessary have its coordinates adjusted so that it is not "crowding" its neighbours. The routine which carries this out works systematically up the stem and along each branch of the plant. The method will result in small relative movements of neighbouring cells but the cumulative change in shape and size of the plant may be considerable. This is reasonable in terms of real plants."

Aside from the fact that the size of a cell is not specified by its volume, this was accomplished basically as proposed. We will now describe the overall process in more detail. For each cell, the neighbours below are known. Starting with the first cell in the order (we refer to it in this discussion as cell 1), all of whose neighbours below

are base cells which never move, we consider the distance from cell 1 to each of the neighbours below and using the growth rate of cell 1, calculate what these distances should now be. With three such distances, we can calculate the new position of cell 1. (In three-dimensional space, three distances are needed; this is why "Reorder" is set up to find at least three neighbours for each cell.) If we use three other base cells, a different position will usually result. If two different positions are obtained, a compromise position between the two is used. This position is the new position of cell 1. Next the same calculation is done for the second cell and so on for each cell in the order. The calculation becomes more complicated since most cells will not have fixed base cells as their neighbours below. In this case, distances between *old* cell positions are adjusted according to growth and then used with the *new* cell positions of the neighbours below to find the new position of the cell being reconciled.

We now consider the "Reconciliation" in more detail. Figure 6-4 illustrates "Reconciliation" through a simple example. As noted earlier, when only cells in the upper part of the plant are growing and dividing, cells below them will not move. (We are assuming that the lower cells are strong enough to support the upper cells without shape changes to themselves; that is, the effect of gravity is ignored.) To save unnecessary reconciliations, cells that cannot move (that is, cells which have neither grown nor divided and for which all cells below them in order have neither grown nor divided) bypass the reconciliation process and retain their old coordinates.

The calculation of the new distance between neighbouring cells from the old distance and cell growth rates is done as follows. The new distance is the old distance multiplied by an average of the growth rates of the two cells. If the two cells have the same size (that is, the same LIN) a simple average is used. If not, a weighted average is used giving greater weight to the growth rate of the larger cell. (If the cells were spherical, this calculation would be precisely accurate. It is a reasonable calculation for cells of the same general shape.)

The average growth rate of the two cells is used to determine the "allowable variation" in the new distance between the cells. The relationship of the variation to the average growth rate can be changed for experimental purposes. The "allowable variation" that was used in most runs increases with growth rate. This means that in a group of cells with differing growth rates, the slower growing cells are more rigid. The allowable variation enters into the reconciliation calculation in a complex way but briefly what happens is as follows. If only three neighbours below a cell are used, the new position can be calculated exactly. If four or more are used, they will generally not agree (that is, the position calculated from the first three will be too near or too far from the fourth, and so on). In this situation, a compromise is used which takes into account the allowable variations in the distances. If the allowable variations are equal, the cell position is chosen so that the worst actual variation is as small as possible. If one allowable variation is larger than the rest, the cell position is chosen so that the other distances are more closely maintained, at the expense of greater actual variation in the distance with greater allowable variation.

To summarize, "Reconciliation" moves cells in such a way that their calculated growth is taken into account as nearly as possible in their movement.

"Reconciliation" is a compromise of the principle of having all controls applicable at the cellular level because it explicitly takes into account interactions between cells. In cases where there are considerable differences in growth rates between cells, the actual growth achieved is not always equal to the calculated growth. The cells, in plain words, are not just pushing each other around, which is valid and does not violate the main hypothesis, but in some cases they are crowding so much as to prevent some cells from growing according to the dictates of their internal growth equation, and this does violate the hypothesis of strict cellular control. The extent of the compromise can be gauged, in a relative way, using measures of error which are incorporated in the program. Improvements to "Reconciliation" could

reduce this problem to some degree but it cannot be avoided entirely unless growth rates vary only very slowly throughout the structure or the structure is a very thin one so that all cells have room to grow.

The next section of the program "Smoothing" also is a compromise of the cellular control. In it the movements of cells calculated in "Reconciliation", that is, the differences between old and new positions, are averaged between neighbours. This is under the control of a factor which can vary between zero and one. If it is one, the "Smoothing" has no effect at all; the movement of each cell is exactly as calculated in "Reconciliation". Otherwise, the actual movement is a weighted average of the calculated movement and the average movement of the neighbouring cells. The factor 0.75 which has been usually used gives 0.75 of the calculated movement and 0.25 of the neighbour's movement. Cells nearby thus move more nearly the same amount and direction and the structure is more stable. Figures 6-5 to 6-8 show the same stages of growth from the same initial as Figs. 5-7 and 6-9 but with different smoothing factors. The least smoothed case (Figs. 6-5 and 6-6) grows most quickly and stops growing first. In the most smoothed (Figs. 6-7 and 6-8), growth is slowed down considerably. In structures that are wider or that have division taking place, some amount of smoothing is necessary to keep the structure together. A subjective judgement is involved as to how much irregularity in the growth is suitable. Increasing uniformity of growth is at the expense of the principle of cellular control. The general effects of smoothing are to decrease the relative movements of cells and increase their rigidity. It would be desirable to revise "Reconciliation" so that cell rigidity and extent of relative movements were explicitly controlled within it rather than being imposed in a more arbitrary way through "Smoothing".

The rest of the program consists of "Plot Plant" which has already been discussed, "Store Plant" used to store data about a plant for future use, and the control which either starts another loop at "Reorder" or stops the run.

A listing of the computer program follows.

```
C*****************************************************************************
C*                                                                           ***
C*                          PLANT GROWTH                                     ***
C*                                                                           ***
C*****************************************************************************
C
C
C                          "OUTLINE"
C
C     - READ PARAMETERS AND CONTROLS.
C     - SET UP INITIAL PLANT  (EITHER CALCULATE OR READ FROM STORAGE).
C     -      STATEMENT #245 IS START OF MAIN LOOP.
C     - "RECORDER" (ARRANGES CELLS IN ORDER OF INCREASING DISTANCE FROM ORIGIN AND
C                   OBTAINS A LIST OF THE NEAREST NEIGHBOURS TO EACH CELL.)
C     - "CALCULATE CELL PARAMETERS"
C     - "CALCULATE GROWTH AND DIVISION"
C     - "IMPLEMENT DIVISIONS"
C     - "RECONCILE" (OBTAIN MOVEMENT OF EACH CELL DUE TO ITS OWN GROWTH AS WELL
C                                AS THAT OF ITS NEIGHBOURS)
C     - "SMOOTHING" (AVERAGES MOVEMENTS OF NEIGHBOURS TO INCREASE UNIFORMITY OF
C                              MOVEMENT)
C        OPTION:  PLOT PLANT.
C        OPTION:  STORE DATA ON DISC FOR FUTURE USE.
C     - IF MORE LOOPS ARE DESIRED, GO TO 245.
C     - STOP.
C
      DIMENSION LR(11),NR(11),LRNGE(11),ICT(08),DANDRN(3),ISTAT(140),
     1STAT(26),                  COORD(64),RAND(97),MX(3),MN(3)
      DIMENSION LIN(0400),P(0400,3)  ,LING(0400),NN(0400,08),
     1SUMLIN(0400),NLIN(0400),PP1(0400),PP2(0400),PP3(0400),P11(0400),
     2 P21(0400),P31(0400),P12(0400),P22(0400),P32(0400),KLM(0400),
     3 INEXT(0400),IHT(0400),              D1(0400), D2(0400), D3(0400),
     4            MER(0400),WAX(0400),NOU(0400),AGE(0400),SFC(0400),
     5TD(0400),   DN(0400,08)
       EQUIVALENCE(P11(1),P(1)),(P21(1),P( 401)),(P31(1),P( 801))
       EQUIVALENCE (DN(1),P12(1)),(DN( 401),P22(1)),(DN( 801),P32(1)),
     1(DN(1201),PP1(1),SUMLIN(1)),(DN( 1601),PP2(1)),(DN( 2001),PP3(1)),
     2(DN( 2401),NLIN(1)),  (TD(1),SFC(1))
      EQUIVALENCE                                    (STAT(1),NRECON)
      EQUIVALENCE (MER(1),D1(1)),(WAX(1),D2(1)),(NOU(1),D3(1))
      INTEGER OM, XSECT,STATT,PLOTT,              A,B,C, STORE  ,STAT
      REAL LIN,LING,LR,LRNGE,NLIN ,LRT,LRNT,MINLIN,MAXLIN
        REAL NOU,NOUO,MER,MX,MN,LOSS
      DATA DN/3200*0.0/     ,DELTA/0.0001/,ICT/0,0,0,3,12,17,20,23/
      DATA AGE/0400*0.0/,LIN/0400*0.0/,LING/0400*0.0/,P/ 1200*0.0/
       DATA NMAX /  400/,                    INN / 8/                    *
      DATA  NRECON/0/ ,IERROR/0/,NOFP/0/,          NRA/124578157/                  PAGE  1
```

```
      DATA COORD/
     1 0.,0.,1.,0.,0.,1.,1.,1.,   -1., 0., 0.,-1., 1.,-1.,-1., 1.,
     2 2.,0.,0.,2.,1.,2.,2.,1.,    2., 2.,-1.,-1., 2.,-1.,-1.,2.,
     3 0.,3.,1.,3.,3.,1.,3.,0.,    0.,-2., 1.,-2.,-2., 1.,-2., 0.,
     4 2.,3.,3.,2.,3.,-1.,2.,-2.,-1.,-2.,-2.,-1.,-2., 2.,-1., 3./
C
C  * INN IS THE 2ND DIMENSION OF NN AND DN. THE DIMENSION OF LR AND NR IS INN+3.
C
C
C                                                                                *
C*********************************************************************************
C                                                                                *
C                                                                                *
C                        "READ PARAMETERS AND CONTROLS"                          *
C                                                                                *
C                            VERSION NUMBER
      VERSON= 4.00
C
C
      READ(5,250) DANDRN, IINIT,ISFTAD,C9,C10
     1,C11,FFF,IPRNT2,IPRNT3,IPLOT,ISTORE,MRECON,IUNIT,MIUNIT,ISTMIN,
     2 DLIMIT
  250 FORMAT(2X,3A4,2(/2X,I6),4(/3X,F12.5),8(/3X,I6)/3X,F12.5)
C
      F11=1.0-FFF
C
      WRITE(6,285)VERSON
  285 FORMAT('1',130('*')//50X,'"PLANT GROWTH" OUTPUT, VERSION NUMBER',F
     16.2/)
      WRITE(6,248)DANDRN,      NMAX,      IINIT,INN,      ISFTAD,C9,C10
     1,C11,FFF,IPRNT2,IPRNT3,IPLOT,ISTORE,MRECON,IUNIT,MIUNIT,ISTMIN,
     2 DLIMIT
  248 FORMAT(' DATE AND RUN NO. ',3A4/
     1' MAX.NO.OF CELLS DIMENSIONED FOR  ',I6/
     2                 ' CODE FOR ORIGIN OF INITIAL  ',I2,'  MAX.NO.OF
     3NEIGHBOURS PER CELL',I6/' ISFTAD = ',I6,',  COEFFICIENTS TO CALC.L
     4RNGE:  ',3F15.5/'   FRACTIONAL WEIGHT USED IN SMOOTHING  ',F15.5/
     5'  IPRNT2= ',I6,'  IPRNT3= ',I6,'  IPLOT= ',I6,'  ISTORE= ',I6/
     6 '  MRECON= ',I6,'   IUNIT & MIUNIT ARE ',2I7,'  ISTMIN =',I7/
     7 '  DLIMIT = ',F12.5)
      READ(5,249)GL1,  GL2,  GL3,  GL4,  GL5,  GL6,
     1DS1,DA1,DL1,DL2,DS2,DIVIDE,DA2,DM1,DM2,
     2DS3,DA3,DL3,DL4,DS4,DIVID2,DA4,DM3,DM4
  249 FORMAT  (6X,F15.8)
      WRITE(6,620)GL1, GL2,  GL3,  GL4,  GL5,  GL6,
     1DS1,DA1,DL1,DL2,DS2,DIVIDE,DA2,DM1,DM2,
     2DS3,DA3,DL3,DL4,DS4,DIVID2,DA4,DM3,DM4
```

```
  620 FORMAT(' GL1',F15.8,' GL2 ',F15.8,' GL3 ',F15.8,' GL4 ',F15.8/
     1       ' GL5',F15.8,' GL6 ',F15.8,' DS1 ',F15.8,' DA1 ',F15.8/
     1       ' DL1',F15.8,' DL2 ',F15.8,' DS2 ',F15.8,' DIVIDE ',F15.8/
     1       ' DA2',F15.8,' DM1 ',F15.8,' DM2 ',F15.8,' DS3 ',F15.8/
     1       ' DA3',F15.8,' DL3 ',F15.8,' DL4 ',F15.8,' DS4 ',F15.8/
     1    ' DIVID2',F15.8,' DA4 ',F15.8,' DM3 ',F15.8,' DM4 ',F15.8)
C
C
C
C**********   FOLLOWING IS FACSIMILE OF DATA DECK.   **********************
C
C1             /// DANDRN = DATE & RUN NO.
C5          /// IINIT =CODE FOR ORIGIN OF INITIAL (0=CALC.,1=READ FROM DISC.).
C8          /// ISETAD =0 IF ADVICE OF ARB.SHIFTS NOT DESIRED, =1 IF DESIRED.
C9             ///  C9,10,11 ARE COEFFICIENTS IN EQUATION FOR LRNGE(I) :
C10            ///    LRNGE(I)= LR(I)*AMIN1(C9,C10*AVLING+C11)
C11            ///           FORMAT OF C9,10,11 &FFF IS(3X,F12.5).
C12            /// FFF = FRACTIONAL WEIGHT USED IN SMOOTHING.
C14         /// IPRNT2=FREQUENCY OF LISTING LINGS & POSITIONS (0= NO LIST)
C15         /// IPRNT3= FREQUENCY OF LISTING DIST.OF PARAMETERS (0= NO LIST).
C16         /// IPLOT= FREQUENCY OF PLOTTING INTERMEDIATE RESULTS (0= NO PLOT).
C18         /// ISTORE = FREQUENCY OF STORING INTERMEDIATE RESULTS (0= NO STORING.)
C19         /// MRECON = NO.OF RECONS WANTED(INCL. THOSE DONE PRIOR TO THIS RUN.)
C20         /// IUNIT = 1ST UNIT # TO USE TO STORE A DATASET.
C21         /// MIUNIT = LAST UNIT # TO USE TO STORE A DATASET.
C22         /// ISTMIN. STORE ROUTINE STARTS AT NRECON = ISTMIN OR LARGER.
C23            /// DLIMIT IS SQUARE OF DISTANCE LIMIT FOR NEIGHBOURS.
C    LAST CARD OF THIS GROUP IS C23.
C
COMMENT: THESE CARDS ARE IN NUMERICAL ORDER EXCEPT THAT, DUE TO PROGRAM CHANGES,
C           THESE NUMBERS ARE NOT USED: C2,C17,C13,C7,C3,C4,C6
C
C    PARAMETER CARDS FOLLOW,IN STANDARD FORMAT (6X,F15.8). SINCE THESE CHANGE
C   FREQUENTLY, REFER TO READ STATEMENT AND TO "CALCULATE GROWTH & DIVISION"
C   FOR DETAILS OF THE PARAMETERS.
C
C                      ************************************************
C                      **                                            **
C                      ** DATA DECK, AFTER CONTROLS AND PARAMETERS, CONSISTS **
C                      ** OF  (1) CARDS DESCRIBING INITIAL, IF INITIAL IS    **
C                      **          TO BE CALCULATED (IE.IF IINIT = 1),AND    **
C                      **                                            **
C                      **      (2) ONE CARD DESCRIBING PLOT.         **
C                      **                                            **
C                      **                                            **
C                      **   SEE THESE ROUTINES FOR DETAILS.          **
C                      **                                            **
```

```
C                              ***************************************************
C
C                                                                                *
C*********************************************************************************
C                                                                                *
C                                                                                *
C                                                                                *
C                          "SET-UP INITIAL PLANT"                                *
C                                                                                *
      IF(IINIT.GT.0)GO TO 252
C
C--->  EITHER CALCULATE INITIAL PLANT (IF IINIT = 0)
C
C   NLAYER=# OF LAYERS OF CELLS, NPLYER = # OF CELLS PER LAYER, RLIN = AV. SIZE
C    OF CELLS, RANDF = RANDOMNESS FACTOR IN CELL POSITIONING, MORE = 0 IF NO
C          OTHER LAYERS OF CELLS ARE TO BE ADDED.AGE0 IS INITIAL AGE ASSIGNED TO
C          CELLS AND NOU0 IS NOURITURE ASSIGNED TO BASE CELLS.
C        NLAYER ON 1ST CARD IS TAKEN AS 1 (REGARDLESS OF # ON CARD). I.E.,THERE
C         IS ONLY ONE LAYER OF BASE CELLS.
C
       READ(5,258)NLAYER,NPLYER ,RLIN,RANDF,MORE,AGE0,NOU0                READ ***
  258  FORMAT(8X,2I6,2F6.2,I1,2F6.2)
       WRITE(6,452)NLAYER,NPLYER,RLIN,RANDF,AGE0,NOU0
  452  FORMAT('0' ,130('*')/ ' INITIAL PLANT WAS CALCULATED FROM THE FOLL
     1OWING INFORMATION:'/12X,I3, ' LAYERS OF',I3,' CELLS PER LAYER; RLI
     2N=',F6.2,', RANDF=',F6.2,', AGE0=',F6.2,', NOU0=',F6.2)
  453  FORMAT(' ABOVE THIS,',I3,' LAYERS OF',I3,' CELLS PER LAYER; RLIN='
     1    ,F6.2,', RANDF=',F6.2,', AGE0=',F6.2)
        IF(NPLYER .GT.32)NPLYER =32
       NILYER =NPLYER
       DO 253 I=1,NPLYER
       IYC=2*I
       IXC=IYC-1
       LIN(I)=RLIN*1.24
       LING(I)=1.000
       INEXT(I)=I+1
       P31(I)=0.0001
       P32(I)=0.0001
       P11(I)=COORD(IXC)*RLIN
       P12(I)=P11(I)
       P21(I)=COORD(IYC)*RLIN
       P22(I)=P21(I)
       AGE(I)=AGE0
  253  NOU(I)=NOU0
        HEIT=0.0001
       THEIT=0.0001
       MM=NPLYER
       NPREV=0
```

```
  466 IF (MORE.EQ.0)GO TO 263
      READ(5,258)NLAYER,NPLYER ,RLIN,RANDF,MORE,AGE0                    READ ***
      WRITE(6,453)NLAYER,NPLYER,RLIN,RANDF,AGE0
      NLM1=NLAYER
      HEIT=THEIT
      NPREV=MM-NPLYER
      GO TO 257
  257 DO 254 II=1,NLM1
      NRR=3*NPLYER
C
C  RANDOM GENERATES A SET OF RANDOM LOOKING NUMBERS BETWEEN +&- RANDF/2.
       DO 197 I=1,NRR                                                   RANDOM
      NRA=NRA*65539                                                     RANDOM
       IF(NRA.LT.0)NRA=NRA+2147483647+1                                 RANDOM
  197 RAND(I)=( NRA*4.656613E-10)*     RANDF-RANDF/2.                   RANDOM
      CM=1                                                              RANDOM
C
      DO 255 M=1,NPLYER
      MM=M+II*NPLYER +NPREV
      P31(MM)=FLOAT(II)*RLIN+RAND(CM)+HEIT
      CM=CM+1
      IYC=2*M
      IXC=IYC-1
      P11(MM)=COORD(IXC)*RLIN+RAND(CM)
      CM=CM+1
      P21(MM)=COORD(IYC)*RLIN+RAND(CM)
      CM=CM+1
      INEXT(MM)=MM+1
      LIN(MM)=RLIN*1.24
      LING(MM)=1.0005
      AGE(MM)=AGE0
      WRITE(6,952)MM,P11(MM),P21(MM),P31(MM)
  952 FORMAT(1X,I5,3F10.3)
  255 CONTINUE
       THEIT=FLOAT(II)*RLIN+HEIT
  254 CONTINUE
      GO TO 466
  263  N = MM                    /
      INEXT(N)=0
      N1P1=N1LYER+1
      N1L1=N1LYER-1
            GO TO 245
C
C
C--->  OR READ STORED PLANT TO USE AS INITIAL (IF IINIT = 1)
C
C
C
```

```
  252     IQ=10
      READ(IQ,601)N,NRECON,NMX,N1LYER,N1P1,NOU(1)
      READ(IQ,602)(LIN(I),I=1,NMX),(P11(I),I=1,NMX),(P21(I),I=1,NMX),
     1(P31(I),I=1,NMX),(LING(I),I=1,NMX),(AGE(I),I=1,NMX)
  601 FORMAT(5I6,E13.7,37X)
  602 FORMAT(6E13.7,2X)
      DO 606 I=1,N1LYER
      INEXT(I)=I+1
      P12(I)=P11(I)
      P22(I)=P21(I)
      P32(I)=0.0001
  606 NOU(I)=NOU(1)
      N1L1=N1LYER-1
      DO 607 I=N1P1,N
      INEXT(I)=I+1
  607 CONTINUE
       INEXT(N)=0
C
C                                                                              *
C                          END OF SET-UP-INITIAL                               *
C                                                                              *
C*****************************************************************************
C
C
C                           "START OF MAIN LOOP"
C
  245 CONTINUE
C
      DO 145 I=2,26
  145 STAT(I)=0                                                    STAT
C
C        COUNT # OF LOOPS (NRECON).
      NRECON=NRECON+1
C
C
C*****************************************************************************
C                                                                              *
C                                                                              *
C                           "REORDER"                                          *
C                                                                              *
C        ARRANGE CELLS IN ORDER OF INCREASING DISTANCE FROM BASE AND
C            OBTAIN LISTS OF NEAREST NEIGHBOURS OF EACH CELL.
C
C
      DO 201 I=N1P1,N
      DO 393 J=1,8
      NN(I,J)=0
  393 DN(I,J)=0.0
```

```
      TD(I)=0.0
  201 CONTINUE
C
  202  DO 703 I=1,N1L1
       DO 203 J=N1P1,N
      PZ=P31(I)-P31(J)
      PZ=PZ*PZ
      IF(PZ.GT.DLIMIT) GO TO 203
      PY=P21(I)-P21(J)
      PY=PY*PY+PZ
      IF(PY.GT.DLIMIT) GO TO 203
      PX=P11(I)-P11(J)
      PX=PX*PX+PY
      IF(PX.GT.DLIMIT)GO TO 203
           TD(J)=TD(J)+1.0/(PX+1.3)
       IF(PX.GE.DN(J,8).AND.NN(J,8).NE.0)GO TO 203
      DO 329 K=1,INN
      IF(DN(J,K).GT.PX)GO TO 327
      IF(DN(J,K).LT.DELTA)GO TO 328
  329 CONTINUE
  327 IF(INN.EQ.K)GO TO 328
      ITEQ=INN-K
      DO 331 IMX=1,ITEQ
      IM=INN+1-IMX
      NN(J,IM)=NN(J,IM-1)
  331 DN(J,IM)=DN(J,IM-1)
  328 DN(J,K)=PX
      NN(J,K)=I
  203 CONTINUE
  703 CONTINUE
           ISTAR=N1P1
           I=N1LYER
      DO 204 IJK=N1P1,N
      TDMAX=0.0
      J=INEXT(N1LYER)
      L=N1LYER
  315 PZ=P31(I)-P31(J)
      PZ=PZ*PZ
      IF(PZ.GT.DLIMIT)GO TO 205
      PY=P21(I)-P21(J)
      PY=PY*PY+PZ
      IF(PY.GT.DLIMIT)GO TO 205
      PX=P11(I)-P11(J)
      PX=PX*PX+PY
      IF(PX.GT.DLIMIT)GO TO 205
      TD(J)=TD(J)+1.0/(PX+1.30)
       IF(PX.GE.DN(J,8).AND.NN(J,8).NE.0)GO TO 205
      DO 379 K=1,INN
```

```
      IF(DN(J,K).GT.PX)GO TO 381
      IF(DN(J,K).LT.DELTA)GO TO 382
  379 CONTINUE
  381 IF(INN.EQ.K)GO TO 382
      ITEQ=INN-K
      DO 383 IMX=1,ITEQ
      IM=INN+1-IMX
      NN(J,IM)=NN(J,IM-1)
  383 DN(J,IM)=DN(J,IM-1)
  382 DN(J,K)=PX
      NN(J,K)=I
  205 CONTINUE
      IF(TD(J).LE.TDMAX)GO TO 316
      ITDM=J
           ITDML1=L
      TDMAX=TD(J)
  316  L=J
      J=INEXT(J)
      IF(J.NE.0)GO TO 315
      IF(NN(ITDM,3).EQ.0.OR.TDMAX.LE.0.0)GO TO 340
      KLM(ISTAR)=ITDM
      ISTAR=ISTAR+1
      INEXT(ITDML1)=INEXT(ITDM)
      I=ITDM
  204 CONTINUE
C
      J=N1LYER
      IF(IRST=KLM(N1P1)
      DO 338 III=N1P1,N
      D1(III)=0.0
      D2(III)=0.0
      D3(III)=0.0
      SFC(III)=0.0
      I=KLM(III)
      INEXT(J)=I
      J=I
      DO 389 K=1,INN
  389 DN(I,K)=SQRT(DN(I,K))
  338 CONTINUE
      INEXT(I)=0
      GO TO 391
C
  340 DLIMIT=DLIMIT*1.5
      STAT(3)=STAT(3)+1
      INEXT(N1LYER)=N1P1
      DO 341 I=N1P1,N
      TD(I)=0.0
      DO 392 J=1,8
```

```
      NN(I,J)=0
  392 DN(I,J)=0.0
  341 INEXT(I)=I+1
      INEXT(N)=0
      GO TO 202
  391 CONTINUE
C
      IF(NRECON.NE.1)GO TO 142
      NRECON=0
       GO TO 476
  142 CONTINUE
C
C                    END OF REORDER
C                                                                              *
C*****************************************************************************
C                                                                              *
C                  "CALCULATE CELL PARAMETERS"                                 *
C                                                                              *
      DO 417 I=1,INN
      DO 418 II=N1P1,N
      NI=NN(II,I)
      IF(NI.EQ.0)GO TO 418
      TT1=P11(NI)-P11(II)
      TT2=P21(NI)-P21(II)
      TT3=P31(NI)-P31(II)
      D1(II)=D1(II)+TT1
      D2(II)=D2(II)+TT2
      D3(II)=D3(II)+TT3
      SFC(II)=SFC(II)+DN(II,I)
      IF(NI.LT.N1P1)GO TO 418
      D1(NI)=D1(NI)+TT1
      D2(NI)=D2(NI)+TT2
      D3(NI)=D3(NI)+TT3
      SFC(NI)=SFC(NI)+DN(II,I)
  418 CONTINUE
  417 CONTINUE
      DO 420 II=N1P1,N
      SFC(II)=SQRT(D1(II)*D1(II)+D2(II)*D2(II)+D3(II)*D3(II))/SFC(II)*10
     1.0
      AGE(II)=AGE(II)+1.0
      MER(II)=0.0
      WAX(II)=0.0
      NOU(II)=10000.0
      IHT(II)=0
  420 CONTINUE
      DO 450 I=1,INN
      DO 451 II=N1P1,N
      NI=NN(II,I)
```

```
      IF(NI.EQ.0)GO TO 451
      MER(II)=MER(II)+1.0/(AGE(NI)*AGE(NI)*AGE(NI)/1000.+1.)
      WAX(II)=WAX(II)+LING(NI)*100.0-100.0
      IHT(II)=IHT(II)+1
      IF(NI.LT.N1P1)GO TO 451
      MER(NI)=MER(NI)+1.0/(AGE(II)*AGE(II)*AGE(II)/1000.+1.)
      WAX(NI)=WAX(NI)+LING(II)*100.0-100.0
      IHT(NI)=IHT(NI)+1
  451 CONTINUE
  450 CONTINUE
        J=1
      II=IFIRST
  422 MER(II)=MER(II)/FLOAT(IHT(II))
      WAX(II)=WAX(II)/FLOAT(IHT(II))
            KLM(II)=0
      NI=NN(II,1)
      LOSS= 0.10+LIN(NI)*LIN(NI)*(LING(NI)-1.0)*5.0
      LOSS=AMAX1(LOSS, 0.10)
      NCU(II)=AMAX1(NCU(NI)-LOSS,0.0)
      II=INEXT(II)
      IF(II.NE.0)GO TO 422
C
C         END OF CALCULATE CELL PARAMETERS
C                                                                              *
C*****************************************************************************
C                                                                              *
C                        "CALCULATE GROWTH AND DIVISION"                       **
C                                                                              *
      K=0
      DO 449 I=N1P1,N
      IHT(I)=0
      GL=GL5
      IF(LIN(I).LT.GL1)GO TO 446
      GL=GL2/(LIN(I)+GL3)+GL4
  446 GL=AMAX1      (1.00,AMIN1(GL5,GL))
      IF(GL.LT. GL6 )GL=1.000
      LING(I)=GL
      IF(LING(I).GT.1.00)KLM(I)=1
          DIV=0.0
      IF(SFC(I).GT.DS1.AND.SFC(I).LT.DS2.AND.AGE(I).GT.DA1.AND.AGE(I).LT
     1.DA2.AND.LIN(I).GT.DL1.AND.LIN(I).LT.DL2.AND.MER(I).GT.DM1.AND.
     2 MER(I).LT.DM2) DIV=DIVIDE
      IF(SFC(I).GT.DS3.AND.SFC(I).LT.DS4.AND.AGE(I).GT.DA3.AND.AGE(I).LT
     1.DA4.AND.LIN(I).GT.DL3.AND.LIN(I).LT.DL4.AND.MER(I).GT.DM3.AND.
     2 MER(I).LT.DM4) DIV=DIVID2
      NRA=NRA*65539
      IF(NRA.LT.0)NRA=NRA+2147483647+1
      CRAND= NRA*4.65661E-10
```

```
      IF(DRAND.GT.DIV    )GO TO 449
      K=K+1
      IHT(K)=I
            KLM(I)=1
  449  CONTINUE
C
C     OBTAIN DISTRIBUTION OF PARAMETERS AND LING.
C
      IF(IPRNT3.EQ.0)GO TO 455
      IT=MOD(NRECON,IPRNT3)
      IF(IT.NE.0.AND.NRECON.LT.MPECON)GO TO 455
      DO 456 I=1,140
        ISTAT(I)=0
  456 CONTINUE
      DO 457 I=N1P1,N
      II=MIN1(AGE(I)*2.0,19.9)+1
      ISTAT(II)=ISTAT(II)+1
      II=MIN1(LIN(I)*5.0,19.9)+21
      ISTAT(II)=ISTAT(II)+1
      II=MIN1(NOU(I)*2.0  ,19.9)+41
      ISTAT(II)=ISTAT(II)+1
      II=MIN1(MER(I)*20.0  ,19.9)+61
      ISTAT(II)=ISTAT(II)+1
      II=MIN1(WAX(I)*2.0   ,19.9)+81
      ISTAT(II)=ISTAT(II)+1
      II=MIN1(SFC(I)*2.0   ,19.9)+101
      ISTAT(II)=ISTAT(II)+1
      II=MIN1((LING(I)-1.0)*200.0,19.9)+121
      ISTAT(II)=ISTAT(II)+1
  457 CONTINUE
      IT=N-N1LYER
      WRITE(6,458)IT,K,ISTAT
  458 FORMAT('1FOR ',I5,' CELLS, DISTRIBUTION OF PARAMETERS WAS AS FOLLO
     1WS.  ',I5,' CELLS READY TO DIVICE.'/'0AGE: |',4(5I6,'|')/
     2' LIN: |',4(5I6,'|')/' NOU: |',4(5I6,'|')/' MER: |',4(5I6,'|')/
     2' WAX: |',4(5I6,'|')/' SFC: |',4(5I6,'|')/' LING:|',4(5I6,'|')/
     3  '0 II    CALC   ACTUAL   RATIO        X          Y          Z          AGE
     1  LIN      NOU      MER      WAX      SFC    KLM   ')
  455 CONTINUE
C
CC                END OF CALCULATE GROWTH AND DIVISION
C                                                                              *
C*******************************************************************************
C                                                                              *
C                                                                              *
CC                      "IMPLEMENT DIVISIONS."                                 *
C                                                                              *
      IF(K.EQ.0)GO TO 495
```

```
      NK=N+K
      IF(NK.GT.NMAX)GO TO 431
      DO 433 IDI=1,K
      I=IHT(IDI)
      N=N+1
      II=NN(I,1)
      I2=NN(I,2)
      I3=NN(I,3)
      I4=NN(I,4)
           WRITE(6,996)N,I,II,I2,I3,I4
  996 FORMAT(106X,6I4)
      DN(I,1)=AMIN1(LIN(I)*0.7,DN(I,1)*0.5)
      DP1= P11(II)+P11(I2)-P11(I)*3.0+P11(I3)
      DP2= P21(II)+P21(I2)-P21(I)*3.0+P21(I3)
      DP3= P31(II)+P31(I2)-P31(I)*3.0+P31(I3)
      IF(I4.EQ.0)GO TO 467
      DP1=DP1+P11(I4)-P11(I)
      DP2=DP2+P21(I4)-P21(I)
      DP3=DP3+P31(I4)-P31(I)
  467 CONTINUE
      DISTN=DN(I,1)/SQRT(DP1*DP1+DP2*DP2+DP3*DP3)
      P11(N)=P11(I)+DP1*DISTN
      P21(N)=P21(I)+DP2*DISTN
      P31(N)=P31(I)+DP3*DISTN
      AGE(I)=-0.2
      AGE(N)=-0.2
           KLM(N)=1
      LIN(I)=LIN(I)*0.794
        LIN(N)=LIN(I)
      LING(N)=LING(I)
      DO 443 III=1,INN
      IK=NN(I,III)
      NN(N,III)=IK
      IF(IK.NE.0)DN(N,III)=SQRT((P11(N)-P11(IK))**2+(P21(N)-P21(IK))**2+
     1(P31(N)-P31(IK))**2)
  443  CONTINUE
      NN(I,1)=N
      DO 444 IJ=N1LYER,N
      IF(INEXT(IJ).EQ.I)GO TO 482
  444 CONTINUE
  482 INEXT(IJ)=N
      INEXT(N)=I
      IF(IFIRST.EQ.I)IFIRST=N
  433 CONTINUE
  495 CONTINUE
C
C       END OF IMPLEMENT DIVISIONS
C
```

```
C*************************************************************************************
CRRRRR                                                                        RRRRR
CRRRRR                                                                        RRRRR
CRRRRR                                                                        RRRRR
CRRRRR                          "RECONCILE"                                   RRRRR
CRRRRR                                                                        RRRRR
CRRRRR                                                                        RRRRR
CRRRRR                                                                        RRRRR
C*************************************************************************************
C
C
C    CELLS THAT HAVE NEITHER GROWN NOR DIVIDED AND FOR WHICH ALL CELLS BELOW THEM
C                  HAVE NEITHER GROWN NOR DIVIDED DO NOT REQUIRE RECONCILIATION.
C
      II=IFIRST
  472  IF(KLM(II).EQ.1)GO TO 127
      STAT(24)=STAT(24)+1                                                STAT
      P12(II)=P11(II)
      P22(II)=P21(II)
      P32(II)=P31(II)
        II=INEXT(II)
      IF(II.EQ.0) GO TO 473
      GO TO 472
C
C       START RECONCILIATION OF CELL II.
C
  127  J=0
      KLM(II)=1
      STAT(2)=STAT(2)+1                                                  STAT
      DO 100 I=1,INN
      NR(I)=0
      NII=NN(II,I)
       IF(NII.EQ.0)GO TO 101
      J=J+1
      NR(J)=NII
      AVLING=(LIN(II)*LING(II)+LIN(NII)*LING(NII))/(LIN(II)+LIN(NII))
      LR(J)=DN(II,I)*AVLING
      LRNGE(J)=LR(J)*AMIN1(C9,C10*AVLING+C11)
  100 CONTINUE
  101 KJ=J+4                                                             STAT
      STAT(KJ)=STAT(KJ)+1                                                STAT
C
C    FIND SET OF NEAREST NEIGHBOURS SUITABLE FOR USE IN RECONCILIATION.
C
           A=1
              B=2
                 C=3
      ICA=3
```

```
      ICB=3
      ICN=1
      DDMIN=LIN(II)*LIN(II)/9.
C
  188 NR2=NR(B)
      NR1=NP(A)
      DX=P12(NR2)   -P12(NR1)
      DY=P22(NR2)   -P22(NR1)
      DZ=P32(NR2)   -P32(NR1)
      DAB=      DX*DX+DY*DY+DZ*DZ
CCCCCCCCCCHECK THAT POINTS ARE NOT TOO CLOSE TOGETHER.
      I114=3
      IF(DAB.LT.DDMIN)GO TO 114
         DAB=SQRT(DAB)
C
  109  NR3=NR(C)
      DXX=P12(NR3)   -P12(NR1)
      DYY=P22(NR3)   -P22(NR1)
      DZZ=P32(NR3)   -P32(NR1)
      DAC=      DXX*DXX+DYY*DYY+DZZ*DZZ
      I114=2
      IF(DAC.LT.DDMIN)GO TO 114
      DBC=      (P12(NR2)-P12(NR3))**2+(P22(NR2)-P22(NR3))**2
     1+(P32(NR2)-P32(NR3))**2
      I114=1
      IF(DBC.LT.DDMIN)GO TO 114
         DAC=SQRT(DAC)
      I114=4
      ANG=(DX*DXX+DY*DYY+DZ*DZZ)/(DAB*DAC)
         ABANG=ABS(ANG)
CCCCCCCCCCCCCHECK THAT THE THREE POINTS ARE NOT NEARLY IN LINE.
      IF(ABANG.GT.0.90)GO TO 114
      AA=DZ*DYY-DY*DZZ
      BB=DX*DZZ-DZ*DXX
      CC=DY*DXX-DX*DYY
      CCT=SQRT(AA*AA+BB*BB+CC*CC)
      CANG=SQRT(1.-ANG*ANG)
      CQQ=     AA*(P11(II)-P11(NR1))+BB*(P21(II)-P21(NR1))+CC*(P31(II)-
     1  P31(NR1))
         CQ1 = ABS(CQQ)/CCT
      CQ=CQ1/SQRT(DAB*DAC*CANG)
CCCCCCCCCCHECK THAT NEW POINT IS NOT NEARLY IN PLANE OF THE OTHERS.
        IF(CQ.LT.0.20.AND.CQ1.LT.0.20)GO TO 114
C
       XCP=DAC*ANG
      YCP=DAC*CANG
      AQ=LR(A)*LR(A)
      XP=(AQ-LR(B)*LR(B)+DAB*DAB)/(2.*DAB)
```

```
      YP=(AQ+DAC*DAC-LR(C)*LR(C)-2.*XCP*XP)/(2.*YCP)
      ZP=AQ-XP*XP-YP*YP
      IF(ZP.GE.0.0)GO TO 380
      STAT(14)=STAT(14)+1                                              STAT
      GO TO 114
  380 ZP=SQRT(ZP)
        OA=DX/DAB
      OD=DY/DAB
      OG=DZ/DAB
      OB=(DXX-OA*XCP)/YCP
       OE=(DYY-OD*XCP)/YCP
      OH=(DZZ-OG*XCP)/YCP
      CCD=ABS(CC/CCT)
      IF(CCD.LT.0.2)GO TO 291
      OI=SQRT(1.-OG*OG-OH*OH)
      CC=(CA*OG+OB*OH)/(-OI)
      CF=(CD*OG+OE*OH)/(-OI)
      GO TO 294
  291 CCD=ABS(AA/CCT)
      IF(CCD.LT.0.2)GO TO 292
      OC=SQRT(1.-OA*OA-OB*OB)
      CF=(CA*OD+OB*OE)/(-OC)
      CI=(CA*OG+OB*OH)/(-OC)
      GO TO 294
  292 OF=SQRT(1.-OD*OD-OE*OE)
      OC=(CA*OD+OB*OE)/(-OF)
      CI=(CD*OG+OE*OH)/(-OF)
  294           IF(A.EQ.4)GO TO 304
        X=CA*XP+OB*YP+OC*ZP+P12(NR1)
        Y=CD*XP+OE*YP+OF*ZP+P22(NR1)
        Z=CG*XP+OH*YP+OI*ZP+P32(NR1)
      TEST=(AA*(X-P12(NR1))+BB*(Y-P22(NR1))+CC*(Z-P32(NR1)))/CQO
       IF(TEST.GE.0.0)GO TO 371
        X=X-2.0*OC*ZP
        Y=Y-2.0*OF*ZP
        Z=Z-2.0*OI*ZP
  371 GO TO 113
C
  114       IF(A.EQ.4)GO TO 177
      IF(ICN.GT.ICT(J))GO TO 206
      STAT(13)=STAT(13)+1                                              STAT
      IF(ICA.EQ.I114)ICA=ICA-1
      IF(ICA.EQ.0)ICA=3
      IF(ICA.EQ.3)ICB=ICB+1
      IF(ICB.GT.J)ICB=4
      NRT=NR(ICA)
      NR(ICA)=NR(ICB)
      NR(ICB)=NRT
```

```
      LRT=LR(ICA)
      LR(ICA)=LR(ICB)
      LR(ICB)=LRT
      LRT=LRNGE(ICA)
      LRNGE(ICA)=LRNGE(ICB)
      LRNGE(ICB)=LRT
      ICA=ICA-1
      IF(ICA.EQ.0)ICA=3
      ICN=ICN+1
      IF(ICA.EQ.2)GO TO 109
      GO TO 188
C
C
C     ONE SUITABLE SET FOUND, CHECK FOR SECOND SET.
C
  113 IF(ICN.GT.ICT(J))GO TO 120
      ICN=1
      ICA=6
      ICB=6
      ICTT=ICT(J)
        J=J+1
      NR(J)=NR(3)
      LR(J)=LR(3)
      LRNGE(J)=LRNGE(3)
      J=J+1
      NR(J)=NR(2)
      LR(J)=LR(2)
      LRNGE(J)=LRNGE(2)
      J=J+1
      NR(J)=NR(1)
      LR(J)=LR(1)
      LRNGE(J)=LRNGE(1)
         A=4
            B=5
               C=6
        GO TO 188
C
  177 IF(ICN.GT.ICTT)GO TO 120
      STAT(13)=STAT(13)+1                                          STAT
      I114=I114+3
      IF(ICA.EQ.I114)ICA=ICA-1
      IF(ICA.EQ.3)ICA=6
      IF(ICA.EQ.6)ICB=ICB+1
      IF(ICB.GT.J)ICB=7
      NRT=NR(ICA)
      NR(ICA)=NR(ICB)
      NR(ICB)=NRT
      LRT=LR(ICA)
```

```
      LR(ICA)=LR(ICB)
      LR(ICB)=LRT
      LRT=LRNGE(ICA)
      LRNGE(ICA)=LRNGE(ICB)
      LRNGE(ICB)=LRT
      ICA=ICA-1
      IF(ICA.EQ.3)ICA=6
      ICN=ICN+1
      IF(ICA.EQ.5)GO TO 109
           GO TO 188
C
  304   X2=OA*XP+OB*YP+OC*ZP+P12(NR1)
        Y2=OD*XP+OE*YP+OF*ZP+P22(NR1)
        Z2=OG*XP+OH*YP+OI*ZP+P32(NR1)
      TEST=(AA*(X2-P12(NR1))+BB*(Y2-P22(NR1))+CC*(Z2-P32(NR1)))/CQO
      IF(TEST.GE.0.0)GO TO 373
      X2=X2-2.0*OC*ZP
      Y2=Y2-2.0*OF*ZP
      Z2=Z2-2.0*OI*ZP
  373 GO TO 122
C
C
C   COMBINE RESULTS FROM TWO SETS OF CELLS.
C
  122 EA=ABS(SQRT((X2-P12(NR(1)))**2+(Y2-P22(NR(1)))**2+(Z2-P32(NR(1)))*
     1*2)-LR(1))/LRNGE(1)
      EB=ABS(SQRT((X2-P12(NR(2)))**2+(Y2-P22(NR(2)))**2+(Z2-P32(NR(2)))*
     1*2)-LR(2))/LRNGE(2)
      EC=ABS(SQRT((X2-P12(NR(3)))**2+(Y2-P22(NR(3)))**2+(Z2-P32(NR(3)))*
     1*2)-LR(3))/LRNGE(3)
           EA=AMAX1(EA,EB,EC)
      ED=ABS(SQRT((X-P12(NR1)  )**2+(Y-P22(NR1)  )**2+(Z-P32(NR1)  )**2)
     1-LR(4))/LRNGE(4)
      EE=ABS(SQRT((X-P12(NR2)  )**2+(Y-P22(NR2)  )**2+(Z-P32(NR2)  )**2)
     1-LR(5))/LRNGE(5)
      EF=ABS(SQRT((X-P12(NR3)  )**2+(Y-P22(NR3)  )**2+(Z-P32(NR3)  )**2)
     1-LR(6))/LRNGE(6)
C  IN THE ABOVE 3 EQUATIONS, NOTE THAT NR1=NR(4),NR2=NR(5) & NR3=NR(6)
C               WHEN THESE EQUATIONS ARE USED.
           ED=AMAX1(ED,EE,EF,DELTA)
      F=1./(1.+EA/ED)
      IE=MIN1(F*EA+1.0,5.0)+15                                          STAT
      STAT(IE)=STAT(IE)+1                                               STAT
C
      P12(II)  =X+F*(X2-X)
      P22(II)  =Y+F*(Y2-Y)
      P32(II)  =Z+F*(Z2-Z)
      GO TO 121
```

```
C
C
C       USING ONE SET ONLY
C
  120  P12(II)  =X
       P22(II)  =Y
       P32(II)  =Z
       GO TO 121
C
C
C          ARBITRARY RECONCILIATION APPLIED                    * * * * * * * *
C                IF STANDARD RECONCILIATION NOT POSSIBLE.
C                                                                              *
  206  PX=0.                                                                   *
       PY=0.                                                                   *
       PZ=0.                                                                   *
       PNX=0.                                                                  *
       PNY=0.                                                                  *
       PNZ=0.                                                                  *
       RALG=0.
       DO 207 I = 1,J                                                          *
       NBVC=NR(I)                                                              *
       PX=PX+P11(NBVC)                                                         *
       PY=PY+P21(NBVC)                                                         *
       PZ=PZ+P31(NBVC)                                                         *
       PNX=PNX+P12(NBVC)                                                       *
       PNY=PNY+P22(NBVC)                                                       *
       PNZ=PNZ+P32(NBVC)
  207    RALG=RALG+LING(NBVC)                                                  *
         Q=1./FLOAT(J)
       RALG=RALG/2.*Q+LING(II)/2.                                              *
       P12(II)  =(P11(II)  -PX*Q)*RALG+PNX*Q                                   *
       P22(II)  =(P21(II)  -PY*Q)*RALG+PNY*Q                                   *
       P32(II)  =(P31(II)  -PZ*Q)*RALG+PNZ*Q                                   *
       STAT(15)=STAT(15)+1                                          STAT       *
       IF( ISFTAD.EQ.0.OR.STAT(19).GT.6)GO TO 121                              *
       WRITE(6,208)II,J,P11(II)  ,P21(II)  ,P31(II)  ,P12(II)  ,               *
      1  P22(II)  ,P32(II),NRECON
  208  FORMAT(' ARBITRARY SHIFT , II & J =',2I4,3F10.6,6X,3F10.6,I4)           *
       DO 209 I=1,J
       NBVC=NR(I)                                                              *
       WRITE(6,210)NBVC,P11(NBVC)  ,P21(NBVC)  ,P31(NBVC),
      1    P12(NBVC)  ,P22(NBVC)  ,P32(NBVC)                                  *
  209  CONTINUE
  210  FORMAT('     BASED ON ',20X,I4,2(3F10.6,6X) )                           *
C                                                                             *
C                                                                             *
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
```

```
C
C
C          RECONCILIATION OF CELL II COMPLETE. NEXT CELL IS INEXT(II).
C
C
  121   II=INEXT(II)
          IF(II.NE.0)GO TO 127
C                                                                              *
C                                                                              *
C                FINISHED RECONCILIATION.                                      *
C                                                                              *
C*******************************************************************************
C                                                                              *
C                                                                              *
C                           "SMOOTHING"                                        *
C                                                                              *
C
          DO 223 II=N1P1,N
         IF(KLM(II).EQ.0)GO TO 474
       PP1(II)=P12(II)   -P11(II)
       PP2(II)=P22(II)   -P21(II)
       PP3(II)=P32(II)   -P31(II)
       APP1=ABS(PP1(II))
         IF(APP1.LT.DELTA)PP1(II)=0.
       APP1=ABS(PP2(II))
         IF(APP1.LT.DELTA)PP2(II)=0.
       APP1=ABS(PP3(II))
         IF(APP1.LT.DELTA)PP3(II)=0.
       GO TO 475
  474   PP1(II)=0.
        PP2(II)=0.
        PP3(II)=0.
  475 NLIN(II)=0.
       P12(II)=0.0
       P22(II)=0.0
       P32(II)=0.0
  223 CONTINUE
       DO 220 II=N1P1,N
       DO 221 I=1,INN
       J=NN(II,I)
        IF(J.EQ.0) GO TO 220
       P12(II)=P12(II)+PP1(J  )
       P22(II)=P22(II)+PP2(J  )
       P32(II)=P32(II)+PP3(J  )
       NLIN(II)=NLIN(II)+1.0
       IF(J.LE.N1LYER)GO TO 221
       P12(J  )=P12(J  )+PP1(II)
       P22(J  )=P22(J  )+PP2(II)
       P32(J  )=P32(J  )+PP3(II)
```

```
      NLIN(J )=NLIN(J )+1.0
  221 CONTINUE
  220 CONTINUE
      DO 364 II=N1P1,N
      PP1A=FFF*PP1(II)+F11*P12(II)/ NLIN(II)
      PP2A=FFF*PP2(II)+F11*P22(II)/ NLIN(II)
      PP3A=FFF*PP3(II)+F11*P32(II)/ NLIN(II)
      P11(II)  =P11(II)  +PP1A
      P21(II)  =P21(II)  +PP2A
      P31(II)  =P31(II)  +PP3A
      SUMLIN(II)=0.0
  364       NLIN(II)=0.0
      DO 226 I=1,3
      DO 225 II=N1P1,N
      J=NN(II,I)
      T=SQRT((P11(II)  -P11(J))  **2+(P21(II)  -P21(J))  **2
     1+(P31(II)  -P31(J)  )**2)/(LIN(II)*LING(II)+LIN(J)*LING(J))*2.0
      SUMLIN(II)=SUMLIN(II)+T*LIN(II)*LING(II)
      NLIN(II)=NLIN(II)+1.
      IF(J.LE.N1LYER)GO TO 225
      SUMLIN(J )=SUMLIN(J )+T*LIN(J )*LING(J )
      NLIN(J )=NLIN(J )+1.
  225  CONTINUE
  226  CONTINUE
      TRATIO=0.
      TNT=0.
      RMAXR=0.0
       RMINR=100.0
      IP=1
      IF(IPRNT2.EQ.0)GO TO 483
      IP=MOD(NRECON,IPRNT2)
  483 CONTINUE
      II=IFIRST
       DO 227  I=N1P1,N
      DIST=SUMLIN(II)/NLIN(II)
      RLG=DIST/LIN(II)
          RATIO=RLG/LING(II)
      TRATIO=TRATIO+RATIO
       TNT=TNT+1.
      RMAXR=AMAX1(RMAXR,RATIO)
      RMINR=AMIN1(RMINR,RATIO)
       LIN(II)=DIST
      IF(IP    .NE.0)GO TO 227
      WRITE(6,228)II,LING(II),RLG,RATIO,P11(II),P21(II),P31(II),AGE(II),
     1DIST,NOU(II),MER(II),WAX(II),SFC(II),KLM(II)
  228 FORMAT(1X,I4,3F7.3,3X,3F8.2,F8.0,F8.2,3F8.3,F8.2 ,I6)
  227 II=INEXT(II)
      IF(IP.EQ.0)WRITE(6,460)
```

```
  460 FORMAT('0PCN       NN ** NO.OF NEIGHBOURS',13X,'****CLOS    ',
     1                    ' ARB///DIST.OF ERRORS///GROWTH RATIO',
     1 'DCR-PLOTSTOP'/
     2  '     #CELLRECL* 0    1   2   3   4   5   6   7   8*     INCP',
     3                    '       0-1 1-2 2-3 3-4 4+? AV MAX MIN ',
     1 'MANT  ?   ? ')
      TRATIO=TRATIO/TNT
      STAT(21)=INT(TRATIO*100.0)                                          STAT
      STAT(22)=INT( RMAXR*100.0)                                          STAT
      STAT(23)=INT( RMINR*100.0)                                          STAT
C                                                                                *
C                                                                                *
C          END OF SMOOTHING                                                      *
C                                                                                *
C                                                                                *
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC*
C                                                                                *
C                                                                                *
C                "PLOT PLANT"                                                    *
C                                                                                *
C  ONE CARD IS READ THE FIRST TIME "PLOT PLANT" IS USED.
C  KY & KZ DETERMINE THE SIDE FROM WHICH THE PLANT IS "VIEWED" FOR PLOTTING. FOR
C  EXAMPLE,KY = 2 AND KZ = 3 GIVES ABSCISSA Y AND ORDINATE Z.(THIS IS DEFAULT.)
C  KY = 1 AND KZ = 2 GIVES ABSCISSA X AND ORDINATE Y.
C  SUPPLY EITHER ALL OR NONE OF KX, KY & KZ.
C****** IN THIS VERSION, MX,MN AND SCALG ARE CALCULATED REGARDLESS OF GIVEN
C        VALUES. THE FULL PLANT IS PLOTTED ON A SCALE SUCH THAT IT FITS AN
C        8 & 1/2 BY 11 INCH AREA. SCALE FACTOR DECREASES AS THE PLANT GROWS.
C
C
      IF(IPLOT.EQ.0)GO TO 238
      IT=MOD(NRECON,IPLOT)
       IF(IT.NE.0.AND.NRECON.LT.MRECON)GO TO 238
  339 CONTINUE
          STAT(25)=999                                                    STAT
      IF(NCFP.GT.0)GO TO 200
  476 READ(5,186)KY,KZ,KX,MX,MN,SCALG                                     READ ***
  186 FORMAT(3I3,6F6.2,F4.2)
      CALL PLOTST('SIMPSON',7)
      CALL PLOT(1.,4.10,-3)
       IF(KY.NE.0) GO TO 200
      KX=1
      KY=2
      KZ=3
  200 MX(KY)=-10.
      MX(KZ)=-10.
      MN(KY)=10.
```

```
      MN(KZ)=10.
      DO 198 I=1,N
      MX(KY)=AMAX1(MX(KY),P(I,KY))
      MX(KZ)=AMAX1(MX(KZ),P(I,KZ))
      MN(KY)=AMIN1(MN(KY),P(I,KY))
  198 MN(KZ)=AMIN1(MN(KZ),P(I,KZ))
  199 SC1=4.0/AMAX1(MX(KY),-MN(KY))
      SC1=AMIN1(SC1,1.0,10.0/(MX(KZ)-MN(KZ)))
      SC=10.0**(1.0+AINT(-ALOG10      (SC1)))
      SC1=AINT(SC1*SC)/SC
      CALL PLOT(0.0,4.40,3)
      CALL PLOT(0.0,-4.10,2)
        CALL SYMBOL(0.50,3.0,0.16,6HSCALE ,270.,6)
      CALL NUMBER(999.,999.,0.,SC1,270.,2)
      CALL SYMBOL(999.,999.,0.,17H / DATE & RUN #: ,270.,17)
      CALL SYMBOL(999.,999.,0.,DANDRN,270.,12)
      CALL SYMBOL(0.25,3.0,0.16,11H# OF CELLS ,270.,11)
      RN=FLOAT(N)
      CALL NUMBER(999.,999.,0.,RN,270.,-1)
      CALL SYMBOL(999.,999.,0.,13H  # OF LOOPS ,270.,13)
           RN=FLOAT(NRECON)
      CALL NUMBER(999.,999.,0.,RN,270.,-1)
      CALL PLOT(0.8,0.0,-3)
      CALL FACTOR(SC1)
      CALL PLOT(-MN(KZ),0.0,-3)
      DO 191 I=N1P1,N
      J=NN(I,1)
      CALL PLOT(P(J,KZ),-P(J,KY),3)
      CALL PLOT(P(I,KZ),-P(I,KY),2)
      J=NN(I,2)
      CALL PLOT(P(J,KZ),-P(J,KY),2)
      J=NN(I,3)
      CALL PLOT(P(J,KZ),-P(J,KY),3)
      CALL PLOT(P(I,KZ),-P(I,KY),2)
      J=NN(I,4)
      IF(J.NE.0)CALL PLOT(P(J,KZ),-P(J,KY),2)
  191 CONTINUE
      CALL PLOT(MX(KZ),0.0,-3)
      CALL FACTOR(1.0)
      CALL PLOT(0.3,0.0,-3)
      CALL PLOT(0.,-4.10,3)
      CALL PLOT(0., 4.40,2)
      CALL PLOT(0.3,0.0,-3)
      NOFP=NOFP+1
       IF(IERROR.GT.0)GO TO 604
      IF(NRECON.NE.0)GO TO 238
      NRECON=1
      WRITE(6,460)
```

```
      GO TO 142
  238 CONTINUE
C
C     END OF PLOT PLANT
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                            *
C                                                                            *
C                           "STORE PLANT"                                    *
C                                                                            *
      IF(ISTORE.EQ.0)GO TO 242
      IF(NRECON.LT.ISTMIN)GO TO 242
      IT=MOD(NRECON,ISTORE)
      IF(IT.NE.0.AND.NRECON.LT.MRECON)GO TO 242
  604 IF(IUNIT.GT.MIUNIT)GO TO 603
      WRITE(IUNIT,601)N,NRECON,NMAX,N1LYER,N1P1,NOU(1)
      WRITE(IUNIT,602)LIN,P,LING,AGE
      STAT(26)=IUNIT                                                  STAT
         IUNIT=IUNIT+1
  603 IF(IERROR.GT.0)GO TO 605
  242 CONTINUE
C
C     END OF STORE ROUTINE
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     PRINT STATISTICS FOR THIS LOOP.
C
C
C
      WRITE(6,461)STAT
  461 FORMAT(1X,I3,I5,25I4)
C
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     LOOP AGAIN OR STOP
C
C
C
      IF(NRECON.LT.MRECON)GO TO 245
C
C
C
      WRITE(6,246)VERSON
      WRITE(6,261)N,NRECON
  246  FORMAT('0',110('*')//55X,'NORMAL TERMINATION'/50X,'"PLANT GROWTH"
     1, VERSION ',F6.2//1X,110('*'))
  247 FORMAT('0XXX   IERROR= ',I5,',   TERMINATING "PLANT GROWTH", VERSION
     1 ',F6.3//1X,110('X'))
```

```
  261 FORMAT('  PRESENT # OF CELLS =',I8,',   # OF RECONCILIATIONS DONE =
     1',I6)
C
      IF(NCFP.GT.0)CALL PLOTND
       STOP
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     ABNORMAL TERMINATIONS
C
C     NO.OF CELLS WILL EXCEED STORAGE; STOPPING RUN.
  431 IERROR=1
      GO TO 469
C
C     NO CELLS GROWING OR DIVIDING; STOPPING RUN.
  473 IERROR=2
C
  469 NRECON=NRECON-1
       GO TO 339
  605  WRITE(6,461)STAT
      WRITE(6,247)IERROR,VERSON
      WRITE(6,261)N,NRECON
      IF(NCFP.GT.0)CALL PLOTND
      STOP
C
C
         END
```

Technical notes about the computer program

The program listed is dimensioned for up to 400 cells and 8 neighbours per cell.

The option in "Set-up initial" is either to set up a new initial plant or to read from storage a previously stored plant and then continue with it.

Reordering is done with the following algorithm. A weighting factor for each active cell is set at zero. The order starts with the base cells. Then for every combination of a base cell and an active cell, the square of the distance between them is calculated. If this is less than a control number DLIMIT, then the quantity

$$1 / (\text{Distance}^2 + 1.3)$$

is calculated and added to the weighting factor of the active cell. After this has been done for all combinations, the cell with the highest weighting factor is, in a sense, the "nearest cell to the group of base cells" and it is put next in the order. Then, for each of the cells not yet ordered, the same quantity is calculated using distance between the newly ordered cell and the cell not yet ordered and added to the weight factor of the latter. The one with highest weight then becomes the next in order and the process is repeated until all cells have been ordered. The quantity

$$1 / (\text{Distance}^2 + 1.3)$$

gives reasonable results for the cell sizes occurring in this study. Other related forms will also work. This procedure will work with any arbitrary distribution of cells although it is not as good if there is strong clustering. It is always necessary to specify one or more base cells or starting cells.

The parameter LIN is calculated in "Smoothing" instead of in "calculation of parameters" for historical reasons. The numbers in the growth and division equations are read at execution time. This allows experimentation be done without recompiling the program for every change. This means that in some runs part of the equations are not used. For example, if probability of division for a class is 1.0, the random mechanism is redundant.

"Reconciliation" would require a very long note to explain in detail. The explanation given earlier is basically correct but simplified. A set of three neighbours below is not suitable for use in reconciliation if they are co-planar with the cell to be reconciled, collinear or too close together. Sets close to these conditions are not used. When there are more than three "neighbours below" available, the program tries to find two suitable sets of three each. (One neighbour may be in both sets.) Then the exact position calculation is done for each set. Then a compromise position as discussed earlier is obtained between the two calculated positions. In some cases no set of neighbours below is suitable to use in the Reconciliation calculation. Then a more approximate calculation called "Arbitrary Reconciliation" is used.

The plots were drawn on Gould and Calcomp plotters. In them, the centres of the base cells are plotted along a line parallel to the printing and just above it (unless cells have grown below the base cells, in which case the whole plot is moved up to avoid the plant covering the printing). The central base cell (which is always cell number one and the origin of the coordinate system) is found just above and to the right of the "&" in "Date & Run #". This information is useful in making measurements to locate a specific cell in the plot.

Experimental method

This study involves two major parts: development of the computer program and experimenting with it. These two activities have gone on simultaneously for most of the study. Because of the many variables in the program, it is necessary when experimenting with it to change only one or two things at a time and try to understand the results of these changes before making more. However, because of the complexity of the system it is not possible to fully understand it. In some cases, major changes are made to the variables to see what will happen. The useful results are studied further and the others are forgotten. This has been referred to as a trial-and-error method. In many ways it is like a selective breeding experiment. The results of this are discussed in the next chapter.

CHAPTER 7

PROGRESS AND RESULTS OF THE EXPERIMENT

The general plan of this chapter is a review in chronological order of the major steps in development of the program, illustrated by computer plants produced at each step. This leads to the present state of the program and the recent plants which are discussed in more detail. (For ease of reference the explanations of figures in this chapter are included in the text. Figure references are followed by the computer run number for reference purposes. Also, when a new version of the program is first used, the version number and number of statements in the program are given in the form [Run 35, V1.0, S661] representing version 1.0 which contained 661 statements. Only major changes to the program are represented by changes in version number. Countless other changes have been made. On some plots, the terms "# of cells" and "# of loops" are replaced by "N AND NRECON" followed by the number of cells and of loops.)

The first part of the work was writing the basic program and testing it in various ways. Nothing more can be said about this early period. The major achievements marking the end of this period were "Reconciliation" and "Smoothing" routines. An early result is shown in Fig. 7-1 [Run 35, V1.00, S661]. (The cells are drawn as squares of dimension LIN. This gives peculiar results since the squares do not turn as the plant bends.) The initial consists of seven layers with four cells per layer and cells on the right side growing more rapidly. The plant bends to the left as expected. However, the cells at the tip maintain a side-by-side orientation even though the plant has bent over and they should be above one another. In this version of the program, cells retain the same neighbours and growth rates throughout the run.

At the next step (Figs. 7-2, 7-3 and 7-4, Run 38, V1.20, S789) major changes to "Reconciliation" and calculation of neighbours had caused the cells to maintain their side-by-side orientation as the

plant tilted. This is also shown in Fig. 8-2 as lines joining cell centres.

At the stage of Fig. 7-5 (Run 43, V2.00, S882) parameters and growth and division equations had been added. The strange appearance of the figure is due to very rapid division and to the cells remaining in line with each other after division.

By the time of Figs. 7-6 and 7-7 (Run 63, V2.20, S773) after many changes to division and "Reconciliation", reasonable growth was taking place except for a tendency to bulge at the centre for no good reason.

The next stage is Figs. 7-8 and 7-9 (Run 66). (Fig. 7-8 is in two parts as are some later figures. When necessary to refer to them separately, they are 7-8 U and 7-8 L for upper and lower.) The bulging has been cured and division rate is reasonable. However, division is scattered through the structure. The division was controlled by an equation for division probability.

The next step (Figs. 7-10 and 7-11, Run 72) is the first case of reasonable growth with division. The initial was Fig. 7-8 U. The plant appeared to be forming branches when it stopped.

The next (Fig. 7-12, Run 74) shows very rapid division leading to many tiny cells all packed together. The division was limited to cells in the class
LIN 0.05-0.70, AGE 2 or more, SFC 7.0 or more, with probability division 0.20. The "Reconciliation" in the case of so many tiny cells is highly arbitrary.

By Fig. 7-13 (Run 78) which has the same initial (Fig. 7-12 U), the division rate was more reasonable but still too high. This division rate was based on a similar division class with probability 0.10. This shows several regions of concentrated division. The parameter MER is not being used. The concentration of division at the tips is due to the definition of the class by SFC and LIN. Only cells of the right

size and on the surface can divide. This seems encouraging in terms of an apical meristem. However, the area of division spreads and the number of growing points multiply and no pattern emerges.

To try to control this, the condition MER less than 0.31 was added to the division class for the next run (Fig. 7-14, Run 79). The number of cells is a bit less than in the previous run and a well defined branch has developed. This is very encouraging. However, the branch is a bit thin and falls over (Fig. 7-15). Producing thicker branches turns out to be a major problem.

In Fig. 7-16 (Run 81) division class
LIN 0.2-0.7, AGE 4 or more, SFC 7.0-8.0 with probability 0.10 produced a somewhat thicker branch off to the right but the shape of the branch is obscured by lines drawn from it to the lower part of the plant. These occur because the method of reordering and choosing neighbours only chooses neighbours closer in toward the base of the plant. Thus for branches growing downward the "neighbours" chosen are far away. This could be eliminated by not allowing downward branches but this would be very arbitrary. Instead an attempt was made to combine the determination of neighbours with ordering cells into a scheme in which the main part of the plant is treated as before but downward branches are dealt with in a different way.

This was implemented and seemed to work but a series of latent problems in the program appeared. By Figs. 7-17 to 7-19 (Run 96, V2.50, S1026), these problems were finally solved. However, no overhanging branches occur to test the new and much longer "Reorder". The growth is complex. The division class is
LIN 0.2-0.7, AGE 4 or more, SFC 7.0-8.0 with probability 0.10, the same as Fig. 7-16.

The next run used the same initial and the same division class except the range of LIN was smaller (0.2-0.5) than for the previous figures. This prevented division entirely (Fig. 7-20, Run 112). Limiting division without stopping it entirely is a major problem.

Figure 7-21,(Run 113) shows the result of a direct continuation of a previous run (Figs. 7-17 to 7-19). This used the ability of the program to store a plant for later continuation. This run stopped after 420 loops because of a complex fault in the "Reordering" process.

In an attempt to obtain more precise control over division, the definition of MER was revised from the average reciprocal of age to the present form. The reason for this is that in order to have a meristematic continuity using the earlier form of MER (which decreases very rapidly with AGE), the division rate had to be very high and too many cells were formed. The new value MER does not decrease as quickly. Other changes were also made. A series of runs were made with the revised program. Many of these resulted in very few divisions and growth eventually stopping (Fig. 7-22, Run 126, V3:00, S1052). The division class for this was
LIN 0.3-0.8, AGE 5-100, SFC 7-8.5, MER 0.04-0.06 with probability 1.0 for cells in this class. (Hereafter, if probability is not stated it is 1.0.) The fault in "Reorder" was still a problem and the current form of "Reorder" was devised. The program had now reached the form described in the last chapter (Version 4.0). The first results are in Fig. 7-23 to 7-25 (Run 145, V4.0, S870) using division class
LIN 0.3-0.8, AGE 5-100, SFC 7.0-8.5, MER 0.05-0.15 which initially looked promising but which developed a very narrow branch that swung around (Fig. 7-26) and hit the main structure (Fig. 7-27) causing damage which then led to the structure changing its shape entirely (Fig. 7-28). This shows that with too many small cells, the "Reconciliation" method breaks down. The problem which led to the breakdown, however, is the thin branches formed which are too flexible.

The next attempt was to reduce division rate allowing more time for growth of broader, more stable branches. In Run 148, the range of LIN was reduced to 0.5-0.8. This halted division almost entirely. In the next attempt, the division class was the same as Run 145 except for SFC 7.0-8.0. This was promising (Fig. 7-29, Run 149) but the branch later became thin and bent over (Fig. 7-30). This then became totally unstable as in Figs. 7-31 to 7-33.

These are included to illustrate the great instability problem with too many small cells.

In the next run, division was again the same as Run 145 but with MER 0.05-0.13. This led to equally unstable (but perhaps more picturesque) results (Figs. 7-34, 7-35, Run 150).

Several more runs were made with further reductions in the division class. It proved impossible to adjust division class to a condition between too few and too many divisions. At this stage there seemed to be at least four possible approaches. One would be to study further the type of pattern found in Figs. 7-17 to 7-19 and 7-21, where, since MER is not involved, it may be possible to have a low but sustained division rate. There can be no concentrated meristematic activity but just rather random outward growth as in these figures. This might, with low division rates, lead to stable structures which could be considered to represent very primitive growth patterns. Because of the lack of order they are less likely to be very interesting and because of the large numbers of cells involved they require a lot of computer time. Another approach is to use two division classes (which is possible with Version 4.0 of the program), one which would by itself cause some division to take place but not sustained division and a second class which would only cause divisions in conditions of low MER, high AGE and high SFC. In this way, after division due to the first class stopped, the second one would eventually restimulate division at some apex giving continuing, though stop-and-go, division at a controllable rate.

A third approach is to use a single division class, including restrictions on MER, but limit division by a low division probability. This might work but it seems likely to eventually become unstable. A fourth approach (which requires changes to the basic program) is to redefine MER so that the influence of a cell division in stimulating other division would last much longer and perhaps extend over a larger area of the plant. This should improve the possibility of sustained but controlled meristematic areas.

The use of two division classes resulted in Figs. 7-36 and 7-37 (Run 160) in which the limb produced is still too thin and flexible but a definite improvement. The division classes were
LIN 0.5-0.8, AGE 10-100, SFC 7.0-8.0, MER 0.03-0.13 and
LIN 0.5-0.9, AGE 40-100, SFC 8.0-9.0, MER 0.0-0.01.

The next attempt (Run 161) used division classes
LIN 0.5-0.8, AGE 20-100, SFC 7.0-8.0, MER 0.03-0.13 and
LIN 0.5-0.9, AGE 50-100, SFC 8.0-9.0, MER 0.0-0.01. This produced a thicker, more rigid branch which grew downward to one side. The division rate was slower and division eventually stopped.

The next trial used division classes
LIN 0.5-0.8, AGE 20-100, SFC 6.9-8.0, MER 0.03-0.13 and
LIN 0.5-0.9, AGE 50-100, SFC 8.0-9.0, MER 0.0-0.01. The only change is a slightly larger range of SFC in the first class. This resulted in Figs. 7-38 to 7-43 (Run 163). This is a slow growing, slow dividing and quite rigid structure which stopped growing entirely after 552 loops. Most of the division that occurred was in the second division class, that is, there was no stimulation of one division by another that could be called meristematic activity. In fact, similar results might be obtained using only the second division class or one like it and the class could probably be adjusted to give continuing division.

This approach was followed using one division class
LIN 0.5-0.8, AGE 30-100, SFC 8.0-9.0 and MER 0.01-0.06. This (Run 163B) gave only one division. The next run, using AGE 20-100, gave the plant in Fig. 7-44 (Run 164) which did not divide after this point.

More trials were made using two division classes. In Run 172, with
LIN 0.5-0.8, AGE 20-100, SFC 6.9-8.0, MER 0.03-0.13 and
LIN 0.5-0.9, AGE 40-150, SFC 8.0-9.0, MER 0.0-0.01, very little division occurred. Apparently by the time high SFC cells were old enough to be in the second class, they were too large for it. In the next case (Run 175)

LIN 0.6-0.9, AGE 15-100, SFC 6.8-8.0, MER 0.03-0.13 and
LIN 0.5-0.9, AGE 40-100, SFC 8.0-9.0, MER 0.0-0.01, there was too much division. An intermediate case, the same as Run 172 except with AGE 15-100 in the first class, produced slightly too rapid division, that is, the branch is a bit thin (Fig. 7-45, Run 176).

The next run was made using the identical equations and an initial with a very small change in the cells at the tip. This gave the result in Fig. 7-46 (Run 177). At this point no division had occurred for some time.

For the next run, the same classes were used but with an initial which was broader and flatter. This produced division at a reasonable rate with the remarkable results shown in Figs. 7-47 to 7-49.

At this point, the experiments were terminated. Conclusions that can be drawn are considered in Chapter 9.

CHAPTER 8

COMPARISON OF REAL AND COMPUTER PLANTS; EXTENSIONS OF THE MODEL

The first part of this chapter is a comparison of real plants to computer plants and the evolution of real plants to the development of the computer model. The latter part is a speculative discussion arising from these comparisons and from earlier chapters.

Analogies between real and computer plants

The various parts of the computer model are considered in turn. The part known as *set-up-initial* is used to define the structure with which growth starts. This is analogous to sowing a seed or starting a tissue culture. The initial used determines to some extent the future growth but it is not the main factor in this. Because of the nature of the computer program, differences in the initial used will cause large differences in the appearance of the resulting computer plants but these are mainly superficial differences. More fundamental points such as the basic pattern of growth expressed in the distributions of the parameters are much less affected by the initial used, except in the short term. A related point is that if the initial is very small then the plant will grow much more slowly than one started with a larger initial. This is not as trivial a statement as it seems because in most of the equations used, division of one cell encourages division of the others and when there are very few cells, this stimulation will take longer to develop. With some of the equations that have been used, an initial consisting of only one active cell could never have divided. This is a rough analogy to tissue culture experiments where initials below a certain size are less likely to grow. A further analogy is that, depending on the equations used, if the initial was, in effect, a piece of tissue from the older part of the plant, it could not grow at all (because of age and cell size) but if it was young tissue from the same plant, it could grow readily. In other words, the meristematic condition requires continuity; we are modelling its continuation, not its beginning. It should be noted that these analogies are specific to

the equations used to encourage meristematic division and they would not hold with all equations. If the analogies are good, they support the choice of equations.

The next section of the program is *reorder*, an essential technical function. Basically it determines, for each cell, which other cells are supporting it in the sense that they are neighbouring cells and also *below* it. In a real plant, this ordering is implicit in the fact that the cells are attached to one another. The computer program determines, for each cell, which cells are supporting it and if cell A is supporting cell B, then B cannot support A at the same time. However, in a real plant there is a two way interaction. For example, referring to Fig. 8-1, if this represents cells of a real plant and if cell 8 were to grow while the other cells did not, one would expect the growth of cell 8 to cause some movement of cells 5, 6 and 7. In the model, in this situation, cell 8 could grow by moving away from cells 5, 6 and 7 but the positions of cells 5, 6 and 7 would not change. This unidirectional interaction is clearly a limitation of the model. If the relative positions of these cells changed so that cell 8 came before cell 7 after reordering, then the position of cell 7 would be affected by the position of cell 8 and, in the case where only cell 8 was growing, both cells 7 and 8 might be moved. However, this interaction, on a longer time scale, is less than in a real plant and influences the results to some extent.

The next section is *calculation of parameters*. In real plants, the activities which are the prototype for this calculation are not separated in time or space from the activities which are the prototypes of *calculation of growth and division, reconciliation* and *smoothing*. However, it is possible to make a logical distinction for purposes of discussion. The "calculation of parameters", so to speak, in a real plant involves much of the biochemical activity of the cells. This has been discussed for each parameter in Chapter 5. For example, the *merism* is hypothesized to involve the production of a substance which travels to neighbouring cells, effectively bearing the message that a neighbour has divided, and then breaks down rendering the message less

important. In the model, this complex process is represented by a simple calculation having the general type of behaviour wanted. There are several faults in this analogy and as the model is refined, better methods of calculating merism and other parameters can be introduced. However, because of the way the parameters are used in the growth and division equations and because of the trial-and-error approach, the exact formulation is less critical.

The next section, *calculate growth and division,* is a straightforward application of the assumed relations between parameters and growth and division. In a real plant, the analogous "calculation" involves the whole biochemical and physical machinery of the cell.

The section *implement divisions* is straightforward, except in one respect. For each cell that is to divide, coordinates are determined for the daughter cells, their age is set at zero, their linear dimension is set at a fixed fraction of the size of the parent, and neighbour cells are assigned to them. The one very difficult point is the assigning of positions to the two daughter cells. As learned in the study, this has major effects on the resulting structures. One important consideration is that no two cells should be too close together. Many methods could be used consistent with this condition. The method used in the study is that one daughter cell retains the exact position of the parent and the second daughter cell takes a position intermediate between the first and the average centre position of the four neighbour cells below it. In this way, there is no danger that either daughter cell will have the same position as any other cell. The exact relative positions depend in detail on the four nearest neighbour cells below the parent. In a series of divisions, where one or more daughter cells redivide, the centres of the new cells tend to form a line from surface cells to inside cells but, because of the detailed dependence on neighbour cell positions, they are most unlikely to fall in a straight line. This non-collinearity is technically important because the reconciliation method does not work with cells that are precisely in line. The consequences of this division pattern are discussed in Chapter 9 and its possible origin in the latter part of

this chapter. In general, it may be noted that throughout evolution, as plant structures became more rigid, the effect of division pattern on the resulting structures must have become increasingly important. That is, if there is much relative movement of cells, then the division pattern is less critical to the resulting shape. One of the lessons of this study is that, in the model, there is already a high degree of control on relative movement of cells and consequently the division pattern is very important and requires further study.

The next two sections, *reconciliation* and *smoothing* are its largest and most vital part. The prototype of these calculations in a real plant depends fundamentally on how cells are attached to one another and how cells move relative to one another when there is cell growth. Since the relative movement during differential cell growth is a very complicated process, the approach taken was to develop a reasonable but arbitrary calculation method which gave "reasonable" results. The example (Fig. 8-2) is a test of this behaviour. In this example, the starting point of the calculation was a vertical column of seven layers of four cells each. The desired growth of the cells was fixed at the beginning so that the initially right-hand cells grew fastest. (No division was allowed.) The resulting shape change was obtained using the reconciliation calculation and is shown at four times after 6, 12, 18 and 24 reconciliations (all with the same growth rates). Lines connect the centres of the closest cells. (Since it is a three dimensional structure, there is a cell behind each cell indicated. For simplicity these are not shown but front and back cells remained nearly in line with each other throughout.) No justification more fundamental than the reasonable nature of the calculation and the resulting behaviour can be made. Within these two sections, a number of parameters are used. Although the values used have been chosen after much testing, better choices might be made with further study.

The main sections of the model are used in a loop; that is, after reordering, calculation of parameters, calculation of growth and division, implementation of divisions, reconciliation and smoothing, then reordering is carried out again and the loop repeats. This looping

is necessary in the computer model, whereas in real plants all the activities represented in this loop could occur simultaneously and continuously. However, the looping is just a convenient representation of a continuous process and it does not impose any special constraints on the model since the change at each loop is small.

The termination of a computer growth process can occur in various ways in the model. The first is the purely technical reason that the number of cells reaches the limit of storage requested in the computer program. (There is provision in the program for information about the computer plant to be stored and the growth process later continued with a version of the program allowing greater storage.) The second cause of termination is that the parameters and equations controlling growth and division are such that no cell division has occurred for some time and all cells have reached their limiting size. The third cause of termination is an error or inconsistency in the program which results in a situation where the program cannot continue. Although this was very common during program development, it is believed that the program is now without faults of this type. None of the above are analogous to processes limiting real plant growth (except in fanciful ways). The parameter *nouriture* could be used as a limiting factor for cell growth and division, causing the plant to stop growing at some point. The emphasis in the study thus far has been on early growth.

The reorder program finds neighbours for each cell even in the case of a cell or groups of cells far removed from the rest (at the tip of a branch, as it were). An interesting possibility that would require only minor changes to the program would be when a group of cells grew more than some specified distance from any neighbour, that group would, in effect, drop off the plant and either land on the ground (the plane of base cells) or drift entirely away. If it landed on the plane of base cells then it would continue growing as a separate plant. This is quite analogous to the reproduction of some modern plants.

In a general way the development of this computer model can be compared to the process of plant evolution. There is no intention of claiming that this is a close analogy in terms of specific steps; it is merely an analogy as to the type of process. In this view, changes to the basic structure of the program, to the definitions of the parameters and to the growth and division equations may be regarded as major evolutionary steps. Changes in the numbers used in the model, on the other hand, compare with relatively minor evolutionary steps (or perhaps environmentally induced modifications). In the process of experimenting with the program some of the numbers are changed very frequently (and the program is set up so this can be done easily). Their effect on the output of the program is potentially less far reaching than that of changes to the program structure. Changes to several of these numbers may cause a major change in the results that changes to any of them individually could not have produced. This may be compared to a series of evolutionary changes which are individually minor but cumulatively results in a major evolutionary shift. This analogy to evolution brings up the question of the goals of the study and those of evolution. The "goal" or driving power of evolution is the survival, multiplication and change of those structures which are able to survive, multiply and change. The goal of the study is to produce computer plants which share some features of real plants. Consideration of the goal of evolution suggests other objectives (than meristematic division) which could be pursued in further study. One such is the attainment of a large surface area relative to volume, leading to rapid intake of nutrient from the environment and perhaps to more rapid growth. Another is increase in height, which, for plants that propagate by dropping parts of their structure, would increase the area over which these were distributed.

Implications and extensions

Three interrelated topics which arose earlier are here treated in more detail. They are cell autonomy, attachment of one cell to another, and cell division patterns. The discussion deals with the possible evolutionary position of real cell structures having the properties assumed for the computer plants. From this speculative

discussion there arise some ideas for extensions of the model.

The topic of *cell autonomy* was introduced in Chapter 3 where it was suggested that the cells of primitive plant structures may have been more autonomous than those of modern plants. This raises the possibility that some multicellular structures developed from unicellular species through a progression in which initially separate cells discovered advantages in remaining together rather than separating after division and, further, that the longer they remained together, the more and the better the reasons that were developed to stay together and the less ability they retained to live independently. In this hypothetical progression, the cells being modelled would come at an intermediate point. They have developed a degree of cooperation in that they pass messages from one to another; they are informed of the state of growth and division of their immediate neighbours and, in a very primitive way, they cooperatively use this information to determine the overall structure. Also there is interaction in that the division pattern of a cell is affected by the positions of its neighbours, and, further, it seems possible that the nature of this interaction has through selection pressure been adapted to improve the overall structure. Despite this cooperation the cell is still responding only to its own immediate environment. There is no transfer of information for long distances through the structure.

In this discussion the word *structure* is used in preference to *organism* which seems to convey the idea of a more closely integrated group of cells than those being modelled. Also the term structure suggests an object which is mechanical in its operation. Another way of describing these structures is as intermediate between organisms and completely chance groupings of non-cooperating cells. The intermediate condition could be called a cooperating colony of cells to emphasize their individual autonomy. The question arises when in evolution cells might have developed some of the abilities which are assumed in this study; that is, for example, did the dependence of growth and division on each of the various parameters arise while the cell was living as an individual or after cooperating colonies first developed? This question

was briefly considered for each parameter in Chapter 5. Also, when in evolution did cells develop the ability to remain attached to one another? There are various possibilities. As soon as cells developed the ability to divide, they had a logical option; either to divide completely and move away from one another or to divide into two parts which stuck together (either sharing a common wall or having separate but connected walls). Sticking together might sometimes have been good policy. If the mother cell is attached to a surface in a favourable location, then it may be to the advantage of the daughter cells to remain attached to one another and to the surface. (This assumes the cells are capable of adhering to another surface.) Also dividing and sticking together is in a sense an easier process than dividing completely and moving apart. It may be that the condition of daughter cells staying together is as old as the condition of daughter cells separating from each other. (It is not likely to be much older since external forces would tend to separate some daughter cells.) Thus there may have been, from earliest times, some cells which persisted in staying together (called *multicells* in this discussion). On the other hand, it is clear that there are some advantages in not sticking together. Cells which separate completely are not competing as directly for nutrients and, further, they are more likely to achieve wider distribution and find favourable environments. These are called *unicells* in the following. Pointing out advantages both for multicells and unicells clearly does not make a decisive case for either. On the contrary, it suggests the possibility that both might have developed successfully in different environmental niches. Admitting this possibility, for sake of discussion, raises the further question of whether their descendants later converged or diverged to quite different types of organisms. Of course, some cells still live as single-celled organisms so they have clearly not completely merged. The question is rather, assuming that some unicells, highly adapted to that way of life, later evolved into multicelled organisms, are these organisms different from organisms evolved from multicells that had never had the habit of living alone? (To attempt to distinguish between the descendants of these two hypothetical early lines of evolution among present day organisms would be foolish, but it is an interesting speculation.) One approach to

this question is by considering cell division pattern, the third topic. By this is meant the determination of the plane of division of a cell by the positions of neighbouring cells. Clearly for unicells there can be no such specialized relationship developed since cells do not remain closely adjacent for long periods. This suggests that perhaps unicells, not having major constraints on division pattern from their neighbours, were relatively free to evolve more diverse division patterns both as specialized responses to external environmental factors and to achieve particular internal needs and economies. (They would of course evolve in other respects also.) Then, having this variety, when members of this diverse unicell group became associated in colonies or multicelled structures, they would be able to form very diverse types of structures and organisms. Compare this to the course of evolution of multicells which from earliest times remained attached to one another. These, through long experience in living together in colonies or structures, could have developed very elaborate modes of behaviour but might retain a division pattern which, though changed through evolution, was related to the mode of division of their primitive multicell predecessors and, therefore, retained specific differences from the line of development from unicells. What was the nature of the mode of division of those primitive multicells?

Consider this example (Fig. 8-4). If one of the two cells of structure A divides, B and C are two possible results. B is the configuration in which the original plane of division is approximately maintained. Since B is unique whereas C represents only one of many similar-looking results, probability would favour "something that looks like C" over B. However, the presence of the specialized wall between the two original cells of A may well result, even in very primitive cells, in gradients and strains which polarize the cells in such a way that B is the normal result, rather than the exceptional one. This fundamental control on division on the basis of symmetry would seem to be quite weak, but in the absence of any more specialized control, it could have been the determining factor of plane of division in primitive multicells. That is, there may be a very simple but effective control of division direction in this structure. Given this, no additional

control is required for the cells to continue dividing in the same way to form a long chain. Such division in a chain of cells could logically take place at any cell in the chain. In this discussion this division is called *chain division* and the special case of this in which division only occurs at the end cells is referred to as *chain-end division*.

Compare this to the next diagram of part of a hypothetical layer of cells (Fig. 8-5). This is highly organized in that the cells are in regular rows and columns and this layer is only one of many similar layers in a tissue. In principle, such a structure could grow and maintain its shape either if only surface cells divide (all simultaneously and with internal division planes parallel to the surface) or if all cells on one plane divide (simultaneously and with internal division planes parallel to the plane of dividing cells). It seems clear that to produce the near-simultaneity needed and to accurately control the plane of division in each cell, despite the many cells involved, would require a far more sophisticated mechanism than that of the chain division in the first example. If one considers real tissues of cells (Fig. 8-6) the need of a complex control mechanism is also apparent. The conclusion reached from this comparison is that, in the course of evolution, chain division could have, and therefore did, arise long before the controls necessary to produce highly regular tissues of many cells.

We now try to integrate these lines of discussion. The probable early appearance of chain division and chains of cells and the much later appearance of regular multicelled tissues suggests that during the long interim, the possible benefits of a multicelled condition were explored by relatively irregular multicelled structures of diverse types. It seems that during this period, great regularity of structure was not of first importance for selective advantage. (The advantage conferred by regular and symmetric structures would be in greater efficiency but exploitation of new opportunities does not depend on the most efficient exploiters.) Thus relatively irregular or loosely organized structures probably lived quite successfully for a very long time during which they changed into or were replaced by more regular ones. In this view, one of the likely primitive structures ready to take part

in this evolution would have been multicells in the form of chains. We now consider this course of events. Flexible chains of cells would quickly develop into a type of irregular multicelled structure since they would frequently tangle together (most sailors can verify this from experience with ropes). A tangled mass of cell chains is a multicelled structure (even though it consists of non-cooperating or interfering cells). When such tangles form, the opportunity for mutually beneficial changes is present. Since inner and outer cells would automatically differ in exposure to nutrients and waste products, interaction with other cells, room for continued division, and protection from the outer environment, specialization between these cells seems inevitable. The outer parts might specialize from chain division to chain-end division and further specialize to collection, disposal and protective structures. Specialization in this way would require development of exchange mechanisms to pass nutrients between cells and would cause polarization of chains of cells. Other changes would be in the nature of the connections between chains. The initial large scale tangles and small scale cohesion could lead to more formal arrangements such as large scale interweaving or parallel growth and specialized small scale interconnections. In this, the links between cells of the same chain would remain while the links between cells of adjacent chains changed in strength and in sophistication. This seems to be a potentially very fruitful evolutionary path. Also, this seems like a path which could have led to some of the arrangements of cells in modern plants (Fig. 8-6) and of hyphae in fungi (Fig. 8-7) where the appearance of chains is evident. (Of course, there are other possible routes to these structures and this is not a compelling point.)

The type of cell structure produced by the cell ordering and division patterns used in this computer model can be interpreted as a stage in the postulated evolutionary development from cell chains. Cells are ordered in terms of their position relative to other cells and this ordering process imposes *linearity*. Of course, linearity is fundamental to the nature of a chain. (It is interesting that, based on this ordering, the cell arrangement can be interpreted as a group of chains of cells. However, there is no unique interpretation; it may be

considered as many chains of varying lengths or as a lesser number of branched chains; also, the chains may change during growth. This argument is not compelling since any arbitrary set of cells could be so interpreted.) Computer plants can be viewed as a stage in evolution from chains of cells not because a linear ordering is done but because of how it is used in the model. It is used first in the linear order in which reconciliation is carried out. This was chosen as a convenient calculation method but it does affect the type of structure produced. (Furthermore, the method is in some respects mathematically simple and elegant. If the model were of a known structure, mathematical simplicity and elegance would encourage the belief that the model represented the fundamental nature of the structure. In dealing with a computer model of a structure too primitive to recognize, this reasoning would be strained, to say the least. However, the point can be reversed; that is, the elegance of the model may be an argument for it being fundamental and natural and, hence, for the structure produced being a primitive and natural one. At least, such arguments have led to discoveries in physical science and they may have a place in biology.) The second use of the ordering is in division where one cell retains the position of the parent and the other takes a position between the first and the neighbours below. This is analogous to chain division.

To summarize, this model may be interpreted as a stage in an evolution from chains of cells and tangled chains to more organized structures. This interpretation is not essential but viewing the model in this context suggests interesting changes to it. These changes are in the type and importance of the chaining effect. One is a greater strength of chaining where the immediate neighbours, above and below, of each cell are specifically defined as neighbours in a chain and where the bond to these is different from the bond to the other cells. In other words, chains are clearly identified. The bonds between cells in the same chain could be stronger, longer lasting, fixed in length or direction or some combination. Changes in the direction of reduced chaining effect could be reducing the dependence of the process on the ordering or modifying the division process. There are several ways of implementing each of these changes and other changes could be considered,

representing various evolutionary lines both forward and backward from the present stage. A profitable approach might be to define a discrete series of models in terms of strength of chaining and then explore in detail the possibilities of each model. The computer model could accommodate many changes of this type within its basic structure. Computer explorations along this and other primitive evolutionary paths might lead to a better understanding of the actual course of early evolution.

Direct and genetic responses - A digression

This section returns briefly to a topic raised earlier in the thesis. The question posed was, to what degree are the responses of a plant to stimuli direct and to what degree are they genetic? After considerable thought, the question seems further than ever from clarification. As noted earlier, it is not a practical problem in regard to the thesis. It is rather a philosophical question that sometimes seems important and sometimes meaningless. Because of this uncertainty on the subject, no position on this question will be taken here.

CHAPTER 9

CONCLUSIONS

The series of computer experiments carried out has produced a considerable variety of "computer plants". It would be inappropriate to claim that they closely resemble real plants. Greater similarity will have to be produced before such a claim could be taken seriously. Real plants take so many shapes that it would be strange if some resemblances could not be seen. However, there are three general conclusions that can be drawn at this time.

The principle of cellular control of structure has been implemented in a reasonable way in the computer model. The principle is violated to some extent when there are considerable variations in growth rates within a large mass of cells; the range of growth rate is in effect decreased. However, if the cells are not in a solid mass or if the growth rates vary only slowly through the structure, this problem is much reduced. Further improvements to "Reconciliation" could decrease the problem in those cases where it is serious. This development could take place in different ways depending on what in fact one believes will happen when cells in the centre of a cell mass grow more quickly than their neighbours. It is also noted that this general situation may be relatively uncommon in real plants; a reasonable way of describing meristematic growth would seem to be as a system of division and growth evolved by cells to reduce conflicts between one another during plant morphogenesis.

The capacity that has been developed for "reconciling" cell growth could be put to use testing hypotheses about differential growth in a mass of cells. To do this it is not necessary to have growth and division equations. One can simply assign hypothetical growth rates, either to each cell independently or to cells as a function of location and of time, if appropriate. A hypothesized division pattern could also be imposed in this way. Then "Reconciliation" could calculate the changing shape of the whole mass and the relative positions of individual

cells. This is what was done in the run represented by Figs. 7-2, 3, and 4. In that case, the purpose was to see if the results produced by the program conformed to the expected pattern. This could be done in more complex cases where the actual growth rates of cells in different parts of the cell mass were known. This would be a sophisticated test of "Reconciliation". If "Reconciliation" met the test for a particular type of cell, then it could be used with more confidence to test hypotheses about that type of cell.

The second general conclusion is that there has been some encouraging progress toward the objective of modelling meristematic growth and the more general objective of modelling plant morphogenesis. This is based partly on the computer plants produced to date and partly on the wide variety of promising avenues of development that have appeared. (In fact, the writing of the thesis has been hampered by the temptation to keep trying different computer plants.)

These possible developments vary in complexity. The simplest consist of continuations of the developments discussed in the latter part of Chapter 7. It is noted that one "division class" is defined by nine numbers, that there are two division classes and that the growth equation also depends on several numbers. Thus many combinations of changes are possible.

The second level of changes which could be made are in what might be called peripheral parts of the program. The definitions of MER and SFC could be improved and perhaps the definition of LIN. The other major change at this level is in the positions of new cells at division. On the basis of experience with the division pattern in the program, the tentative conclusion has been reached that another division pattern would be more tractable in the production of regular growth and meristematic growth patterns. The present pattern, when division occurs at the plant surface, corresponds roughly to periclinal division. A pattern resembling anticlinal division would be an interesting study and could be made through changes to the section "Implement Divisions". On the other hand, the present pattern has many more possibilities that

could be explored and it also has a plausible (though hypothetical) origin in evolution, discussed in Chapter 8.

The third level of changes are more basic to the structure of the program. The strength of attachments of cells to one another and the rigidity of cells could effectively be varied through changes to "Reconciliation" and "Smoothing". At present, a simple chain of cells in this model would be very flexible and, as noted in Chapter 7, this causes problems which we have attempted to overcome by producing structures wider than simple chains. Another approach would be to stiffen such chains through changes to the program.

The fourth level of changes are those suggested in Chapter 8: attempts to follow possible evolutionary pathways. These should probably wait until after some improvements have been made to the model at the other levels.

It may be objected that the multitude of changes and improvements suggested are evidence of the inadequacy of the study and the incomplete state of the computer program. The incompleteness is acknowledged but it is maintained that the multitude of possible changes is evidence of the basic soundness of the model and its possible fruitfulness.

The last conclusions are philosophical. It was pointed out that the model has a possible interpretation in an evolutionary line leading to primitive plants. This seems to support the model. Also, the model, though complex, has a certain mathematical elegance and simplicity in its basic nature. This writer is inclined to regard this as an argument in its favour.

The overall conclusion is that the main hypothesis has been supported by this study, though not completely established, and, further, that the model as developed has great potential for future testing of the hypothesis and for improving understanding of primitive plant evolution and, perhaps, modern plant morphogenesis.

FIGURES

The figures are numbered and arranged according to the chapter to which they directly relate. Some figures are referred to in more than one chapter. The plots of computer plants, which are the majority of the figures, are described in general at the beginning of Chapter 6. Notes regarding some figures are in this list but most of the figures are described in the text.

Fig. 3-1 is from C. Billington, 1952: Ferns of Michigan. Bulletin No. 32, Cranbrook Institute of Science.

Fig. 5-1 Cell A has a high SFC since its neighbours are all in the same general direction from it.

Fig. 5-2 Cell B has a low SFC since its neighbours are all around it.

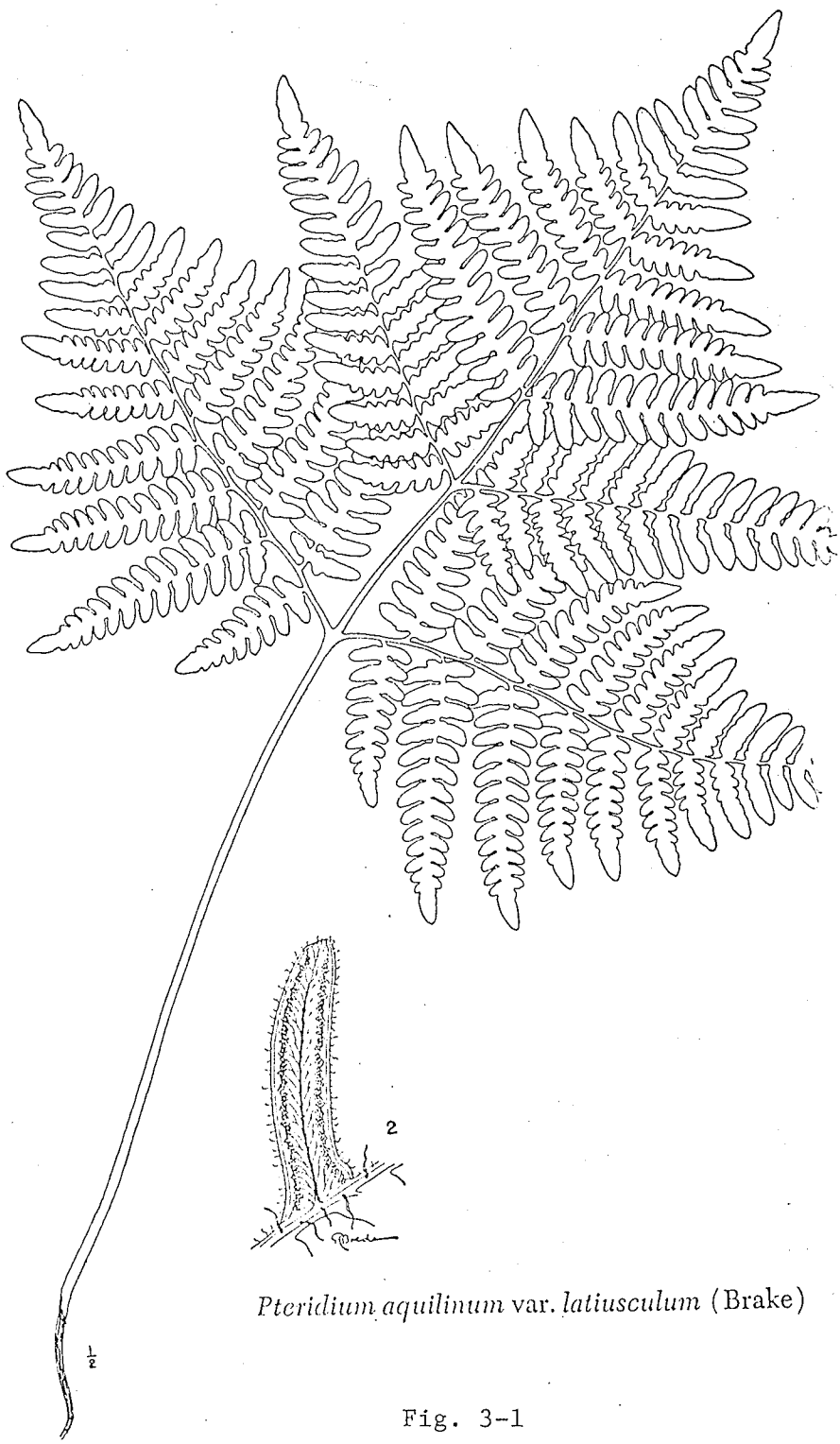
Fig. 5-3 Actual values of SFC. The plant is outlined with a dotted line. Areas of SFC 7.0 or more are marked. They are generally on the periphery of the plant. Only cells in a vertical slice through the centre of the plant are shown for clarity.

Fig. 5-4 Equation of growth and graphs of growth rate (actual and percentage growth) as a function of LIN.

(There is no Fig. 8-3)

Fig. 8-6 is originally from M. Schaffner, 1906: The Embryology of the Shepherd's purse. Ohio Nat. 7:1-8.

Fig. 8-7 is from E.J.H. Corner, 1950: Clavaria and Allied Genera, Clarendon Press, Oxford



Pteridium aquilinum var. *latiusculum* (Brake)

Fig. 3-1

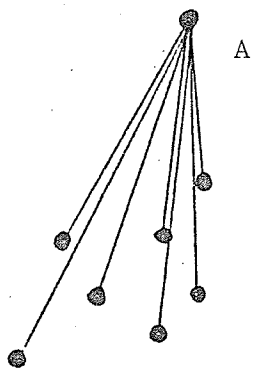


Fig. 5-1

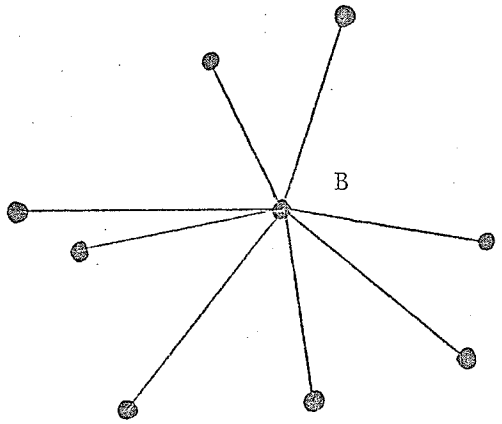


Fig. 5-2

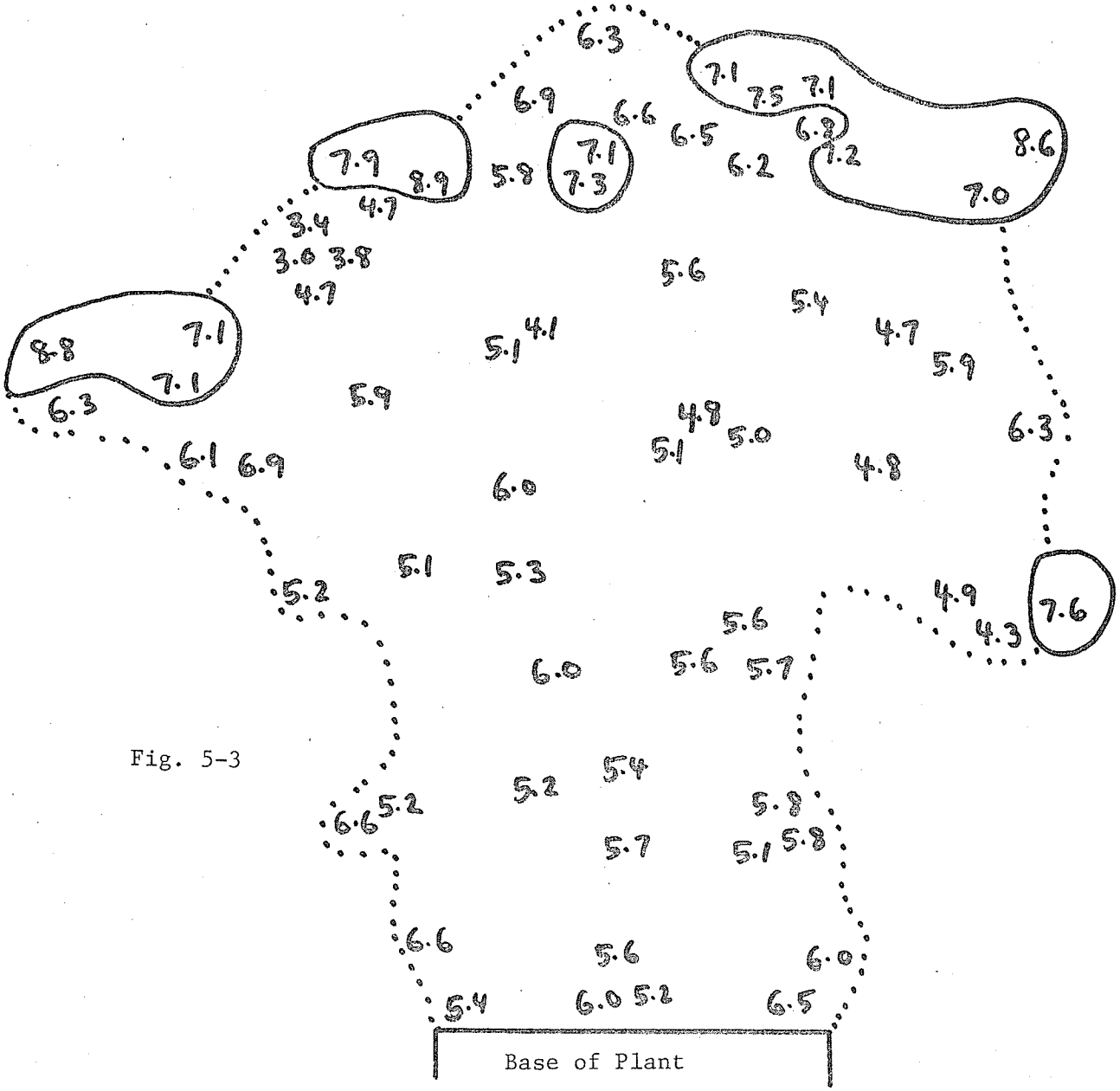


Fig. 5-3

GROWTH EQUATION

If LIN is less than 0.07, GROWTH = 20%.

If LIN is between 0.07 and 1.22,

$$\% \text{ GROWTH} = \frac{1.687}{\text{LIN} + 0.01123} - 0.17 \quad .$$

If lin is greater than 1.22, GROWTH = 0%.

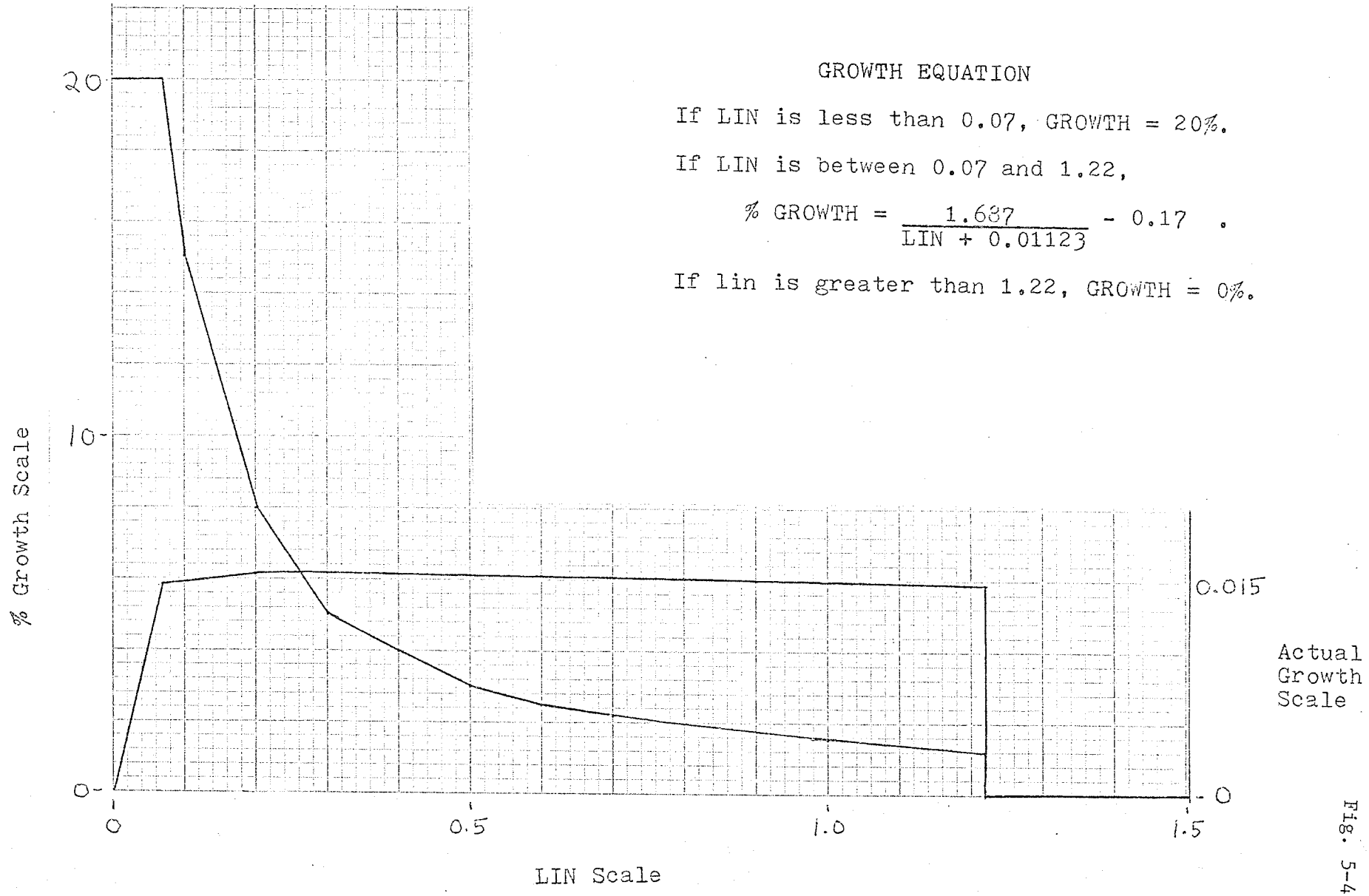
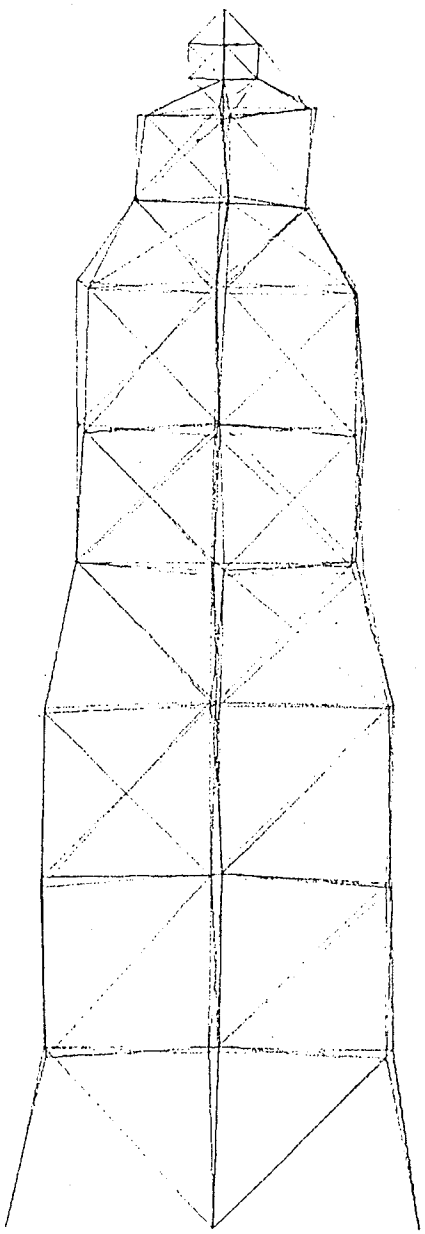


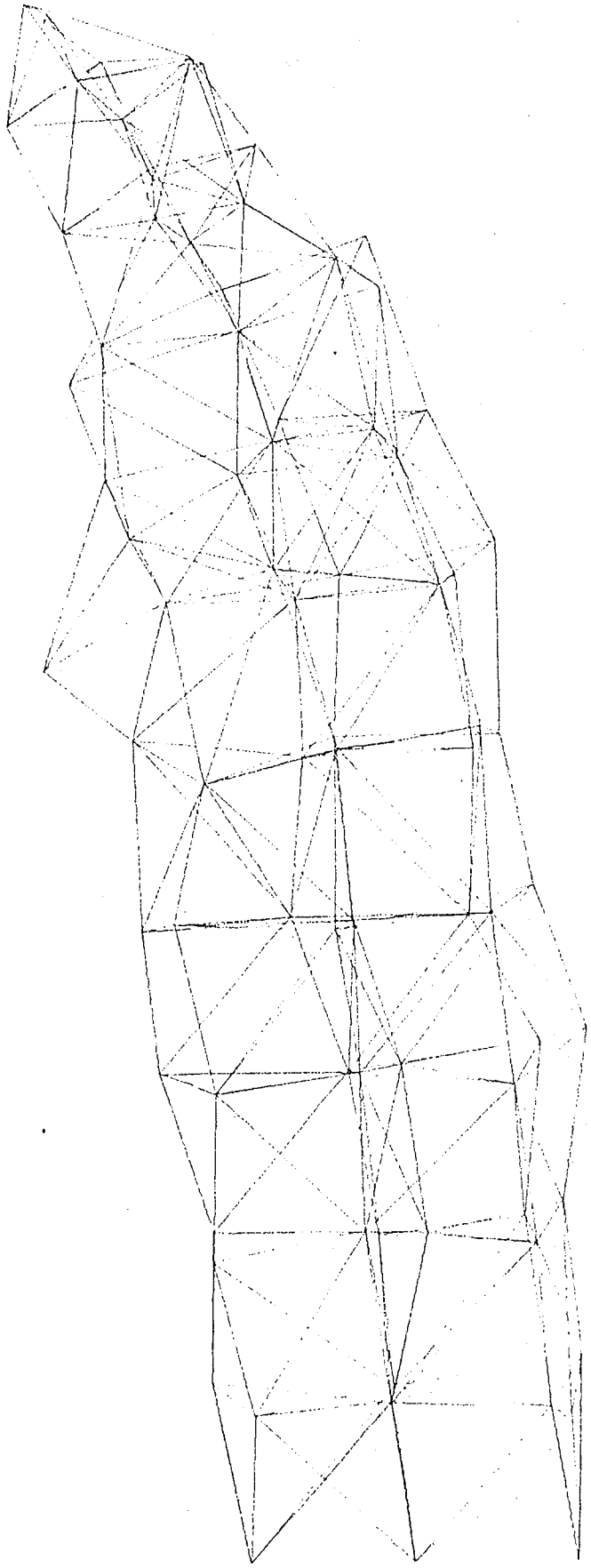
Fig. 5-4

Fig. 5-5



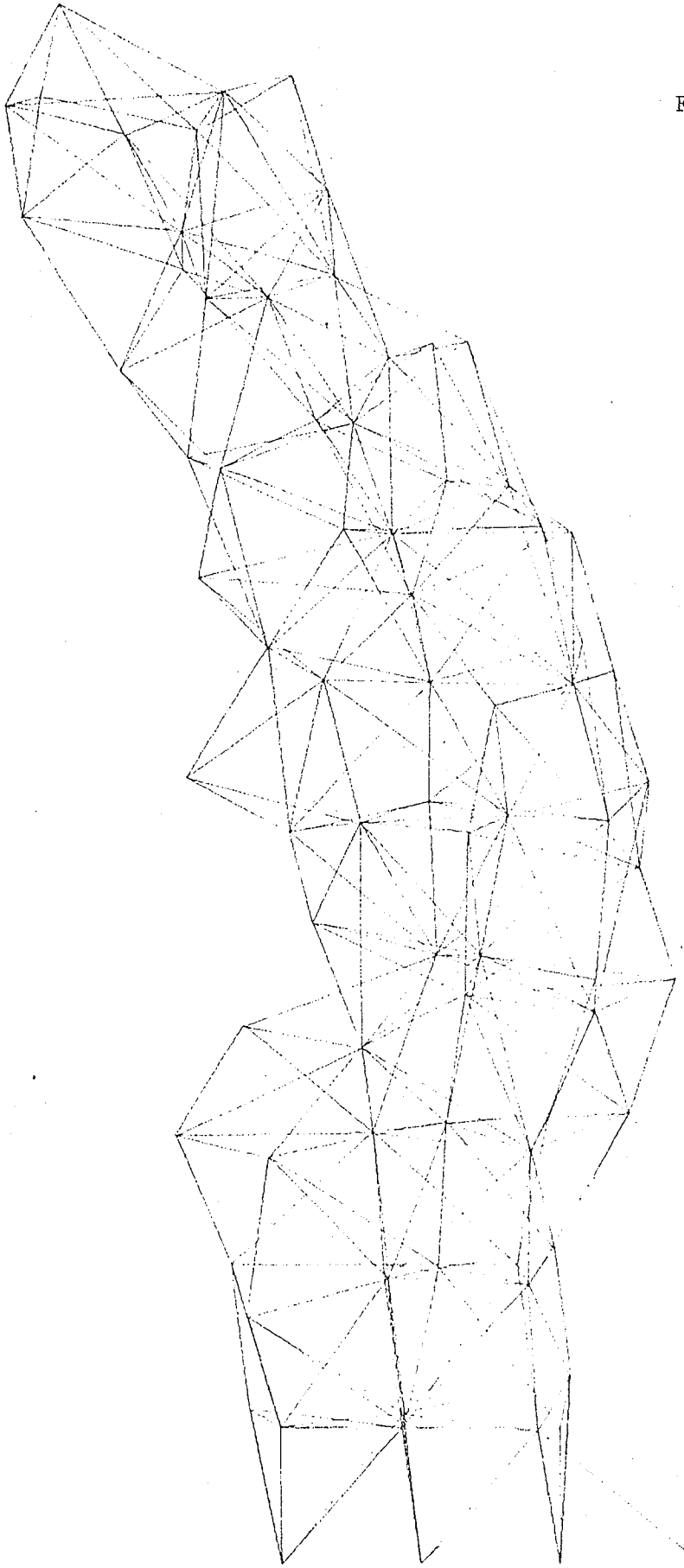
SCALE 0.90 / DATE & RUN =: DEC.10 =165
OF CELLS 94 # OF LOOPS 0

Fig. 5-6



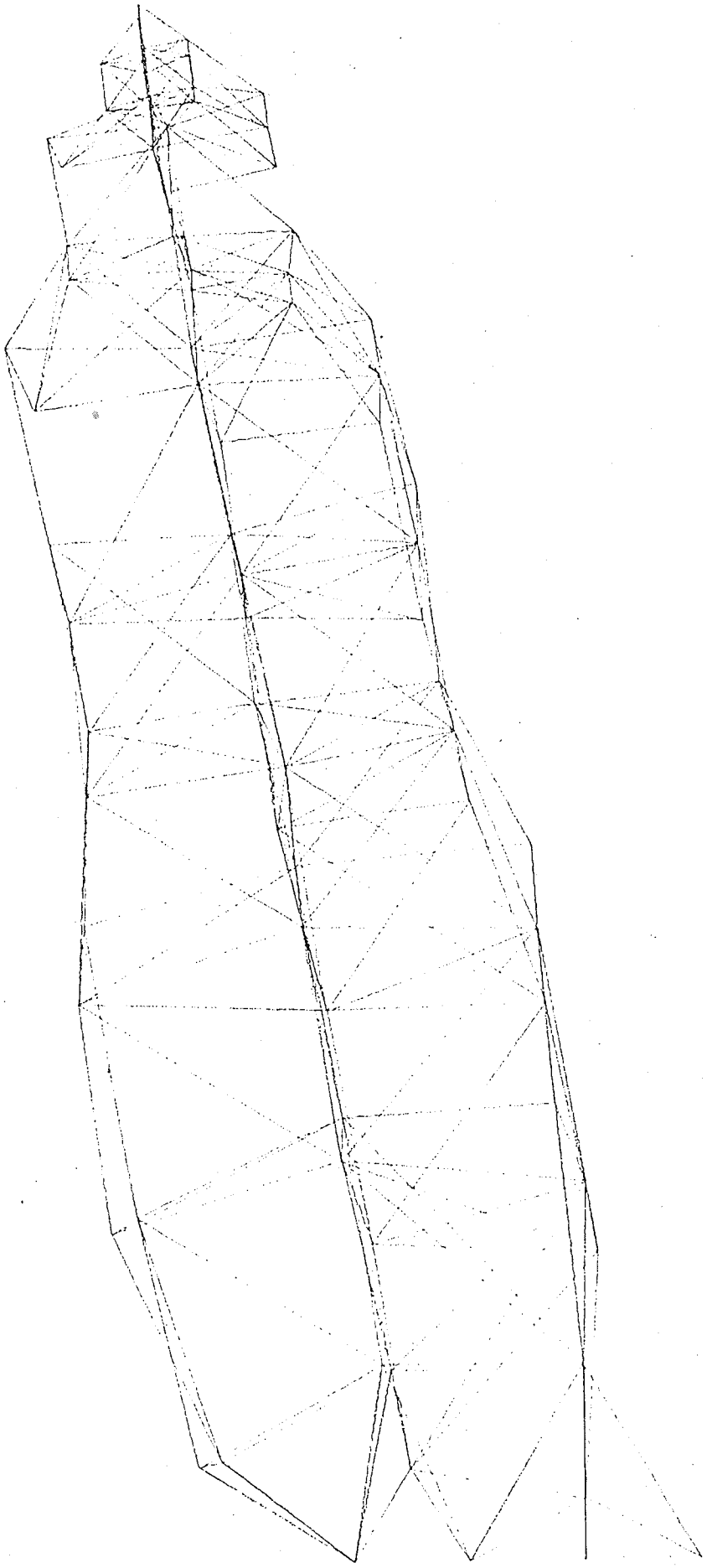
SCALE 0.80 / DATE & RUN #: DEC.10 #165
OF CELLS 94 # OF LOOPS 50

Fig. 5-7



SCALE 0.70 / DATE & RUN #: DEC.10 #165
OF CELLS 94 # OF LOOPS 177

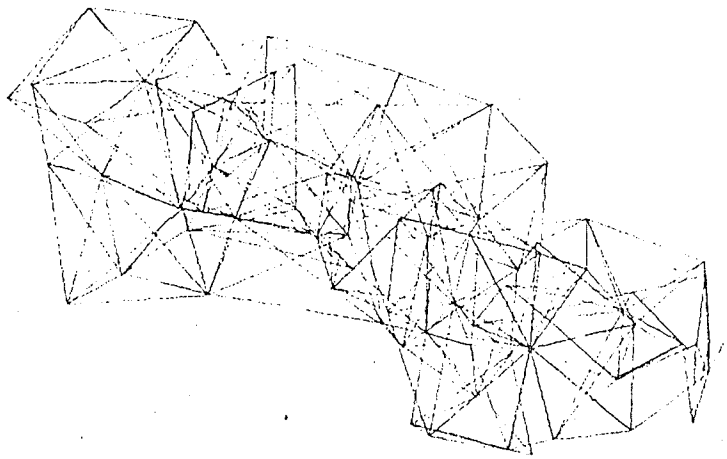
Fig. 5-8



SCALE 0.60 / DATE & RUN #: DEC.10 #166
OF CELLS 90 # OF NODES 50

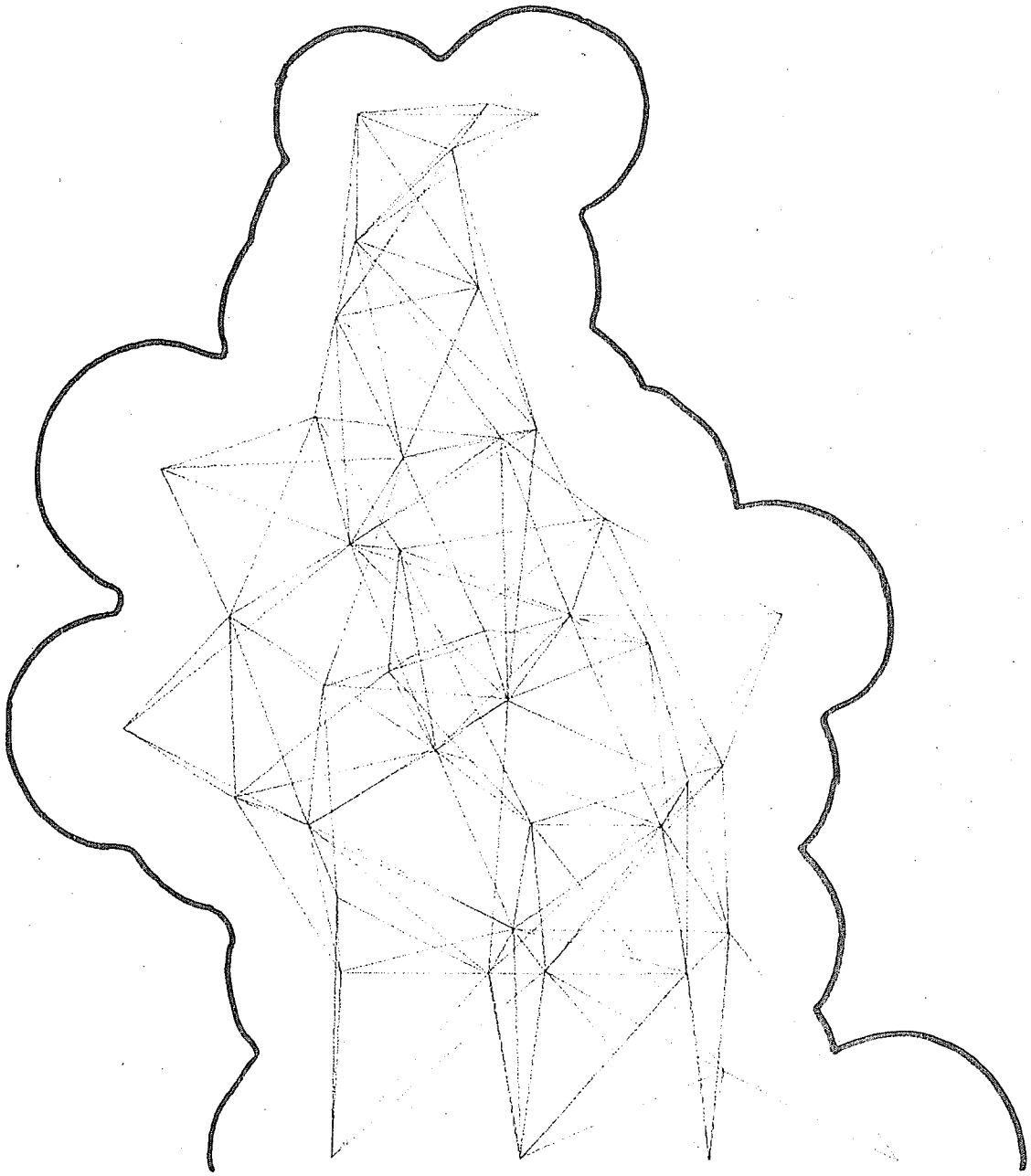
Fig. 5-9

SCALE 0.10 / DATE & RUN #: DEC.10 #166
OF CELLS 94 # OF LOOPS 100

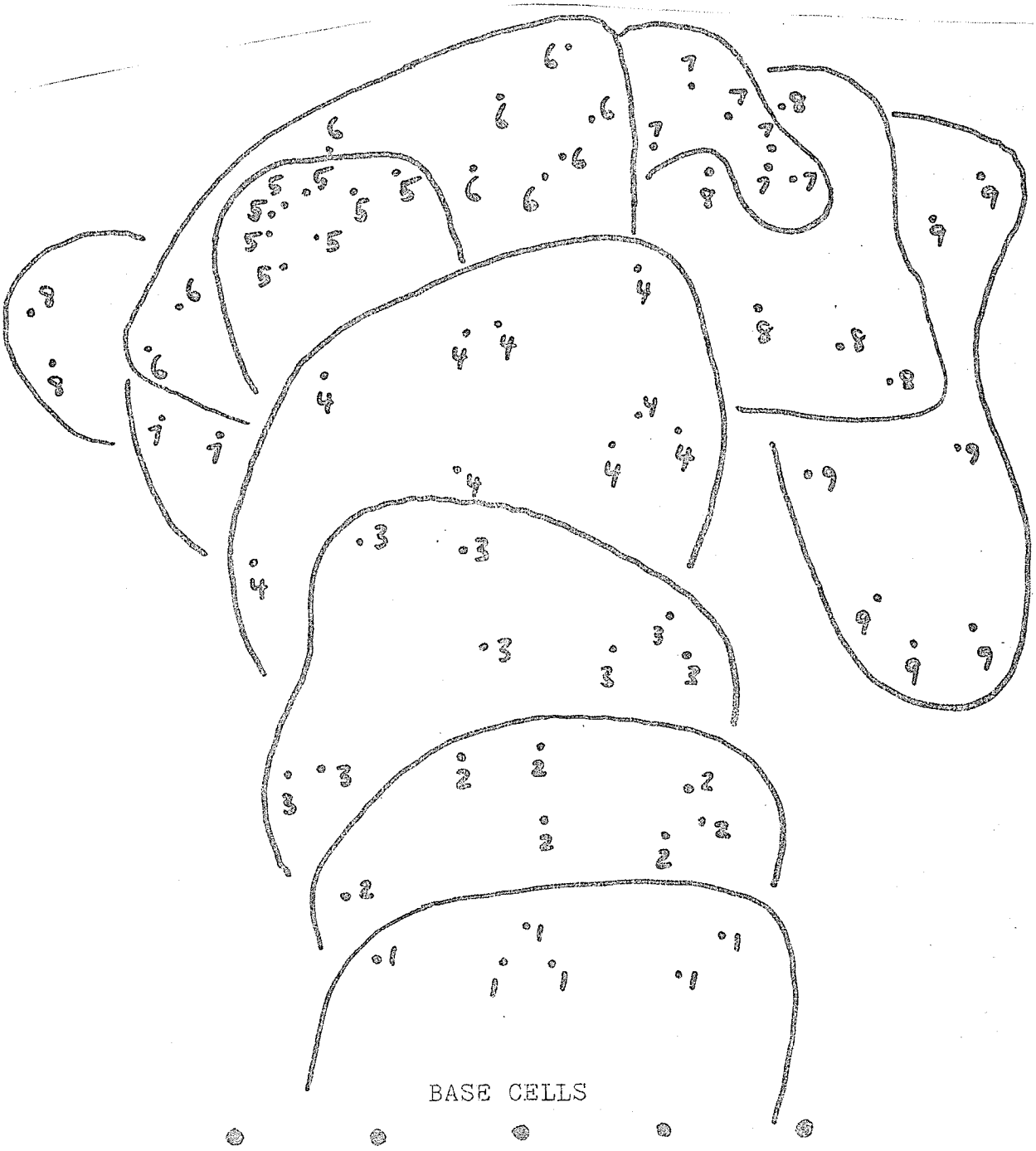


SCALE 0.06 / DATE & RUN #: DEC.10 #166
OF CELLS 94 # OF LOOPS 292

Fig. 6-1



CELL ORDER AS DETERMINED BY "REORDER" Fig. 6-2



The lines are drawn to emphasize the progressively outward ordering.

EXAMPLES OF DIVISION PROCESS

Cell about to divide is "D". The new cells produced by division are "N". Neighbours below are circled in the first two examples.

Fig. 6-3

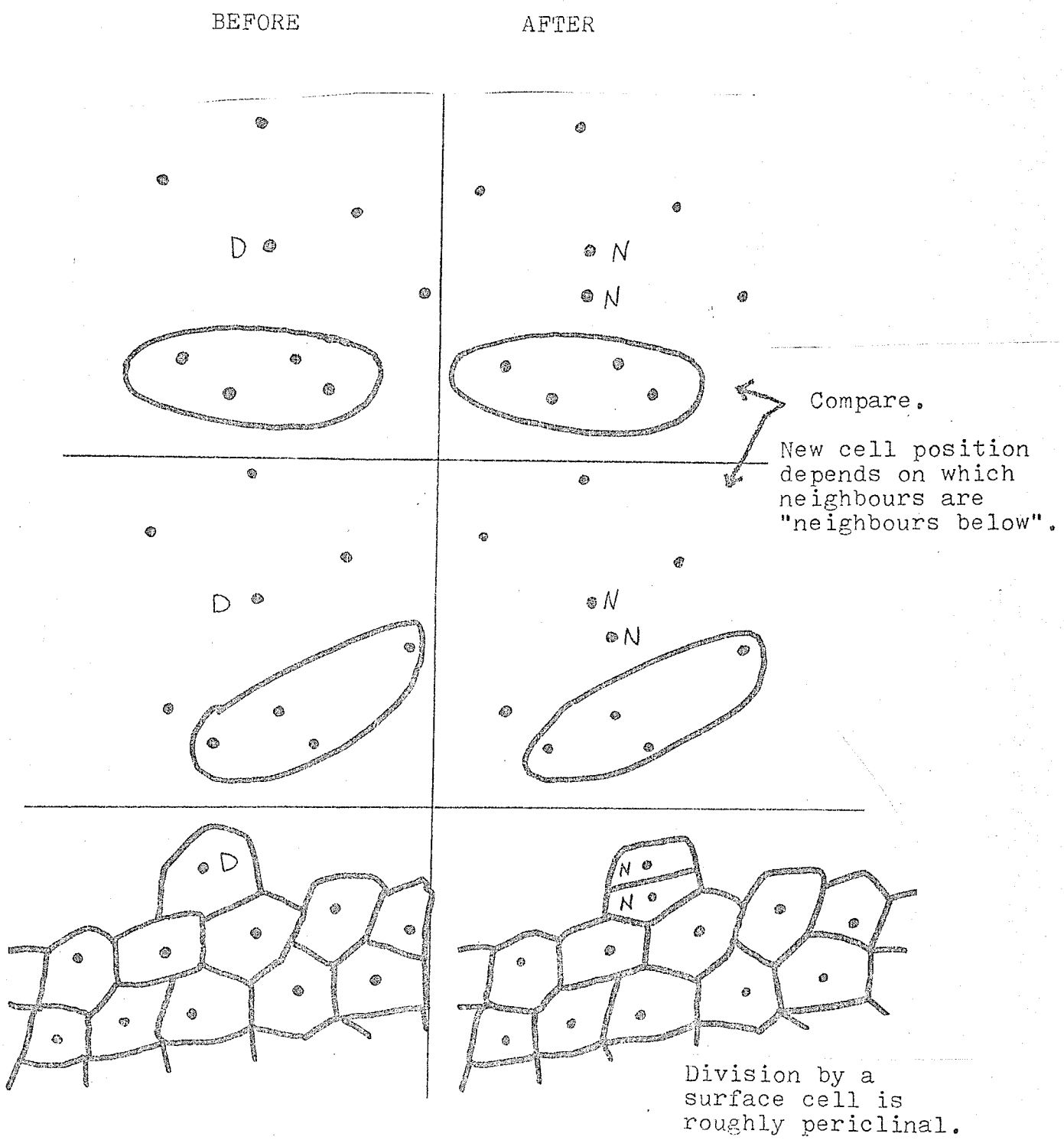
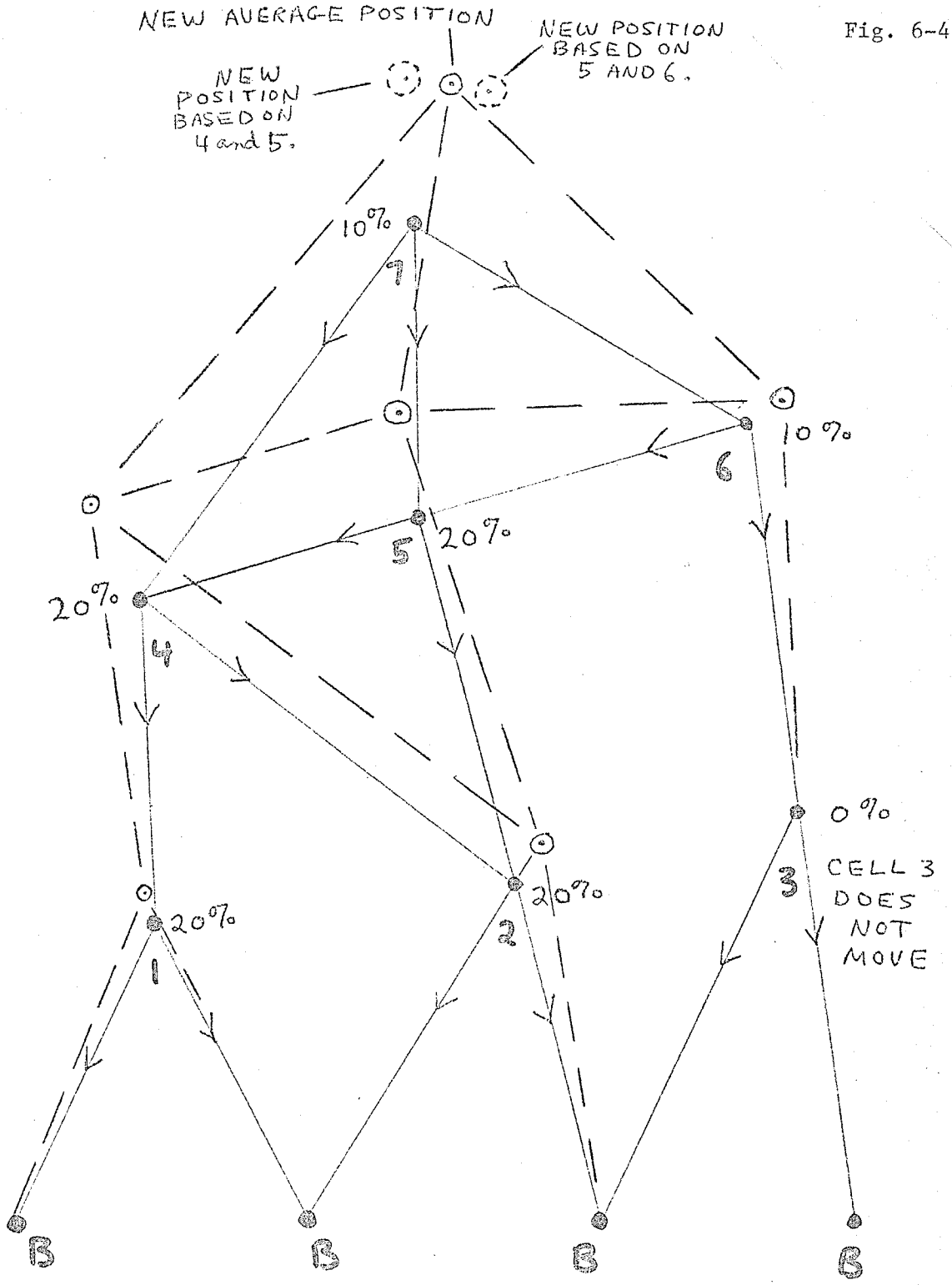


Fig. 6-4



Reconciliation in two dimensions. All cells assumed to have equal LIN and growth rate as marked. Cells are "reconciled" in the order shown. Arrows run from each base cell to its "neighbours below". Since this is in two dimensions, only two"neighbours below" are required. For cell 7,two sets of cells were used. Position before reconciliation is marked "⊙" and after "⊙". Dashed lines join cells at their new positions. "B" represents a Base Cell.

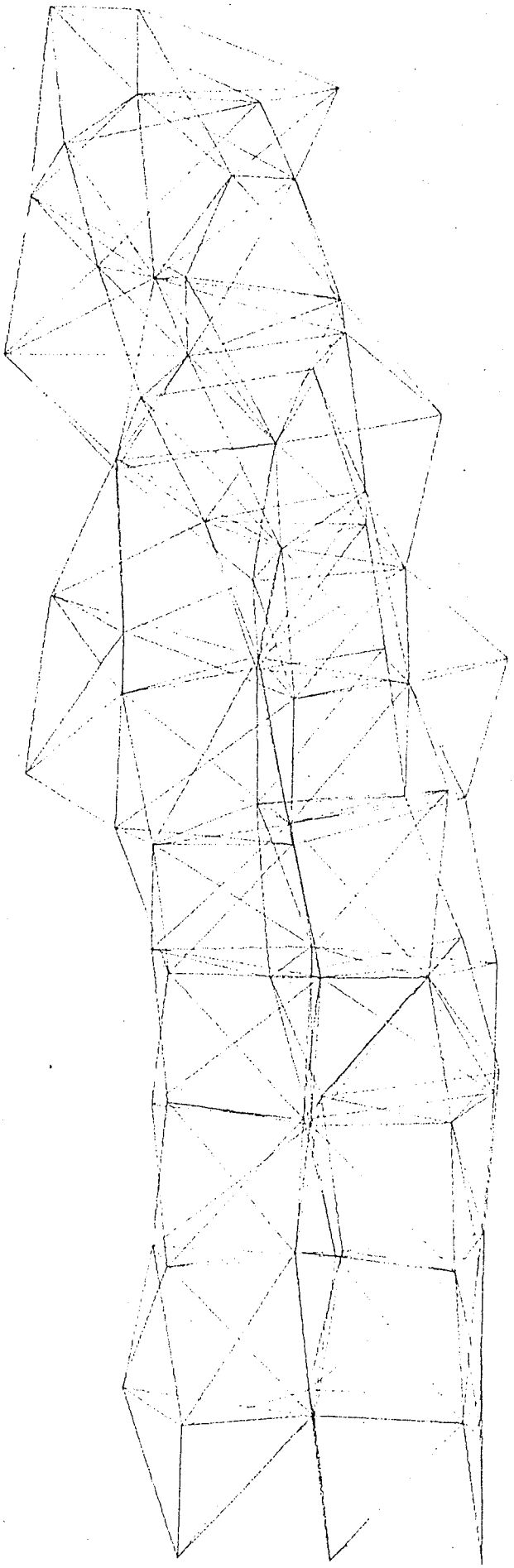


Fig. 6-5

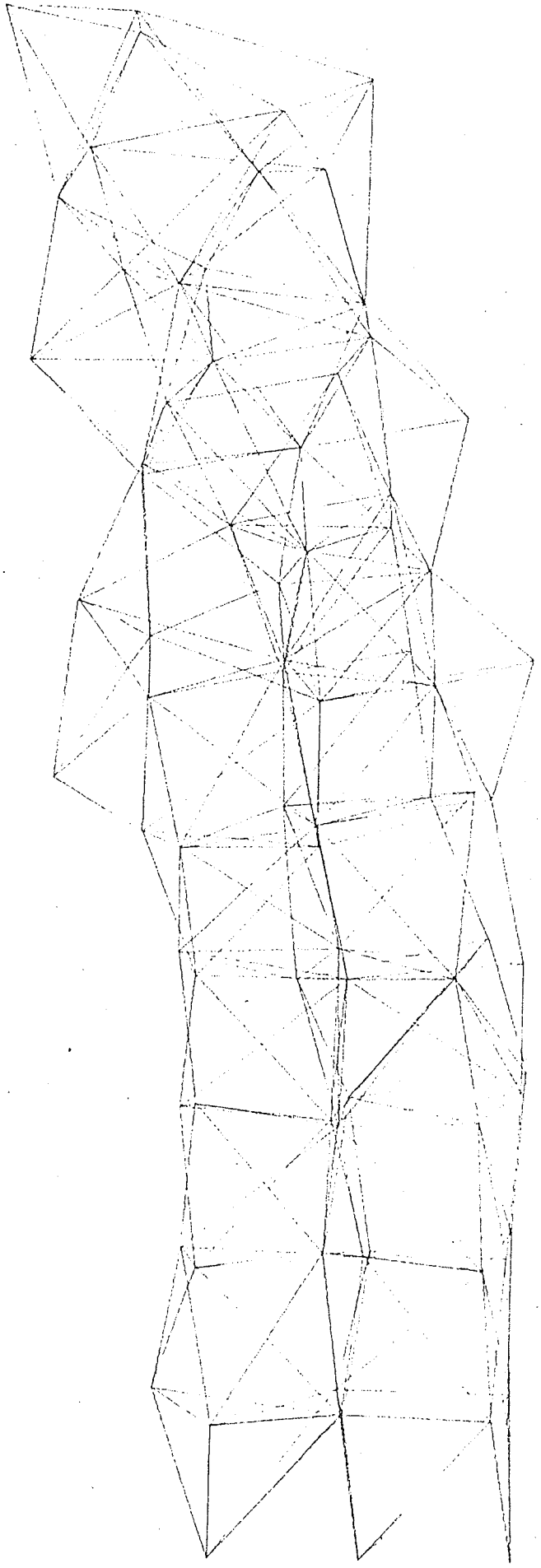
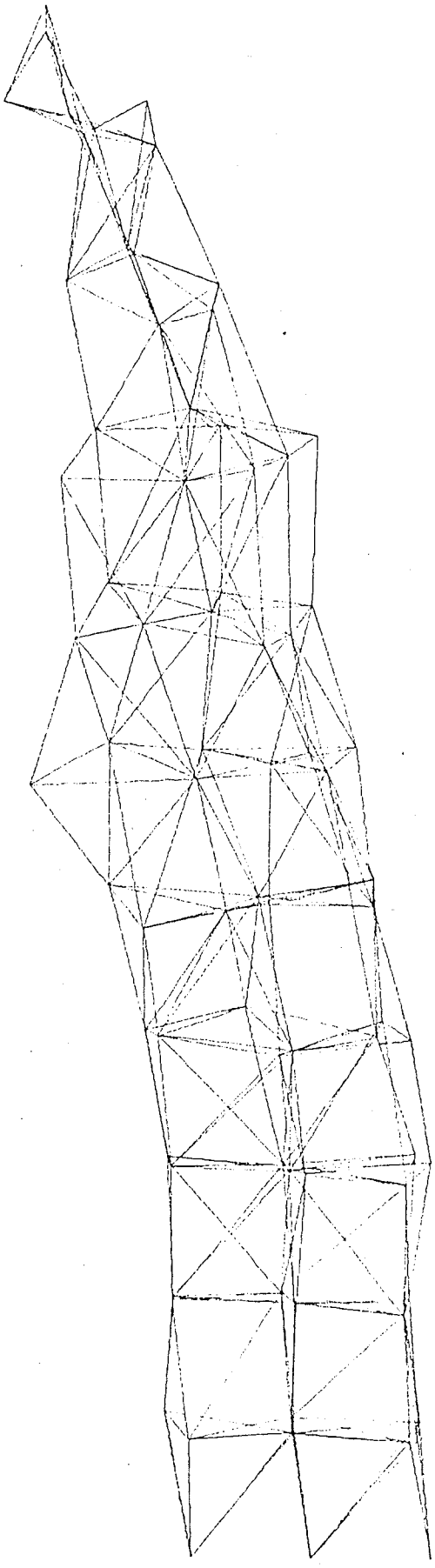


Fig. 6-6

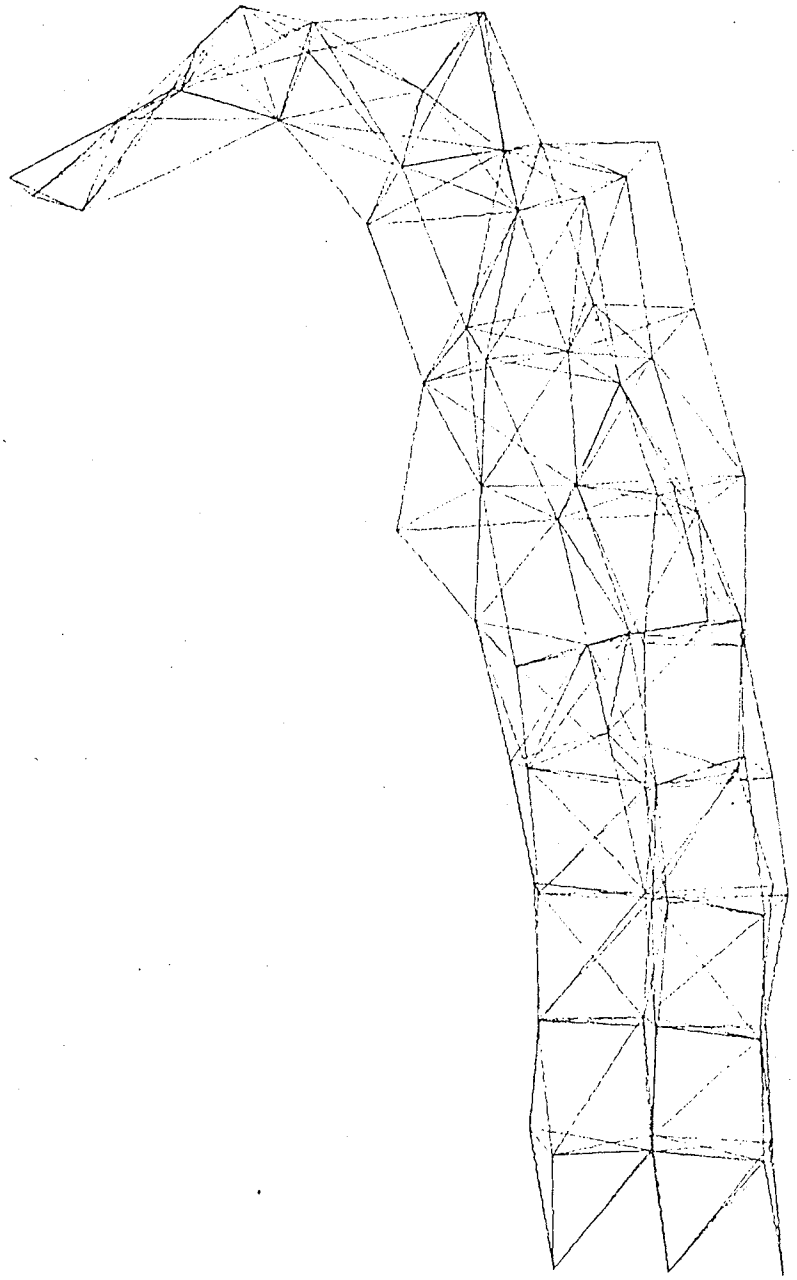
SCALE 0.80 / DATE & RUN # DET 10 #170
LOOPS 147

Fig. 6-7



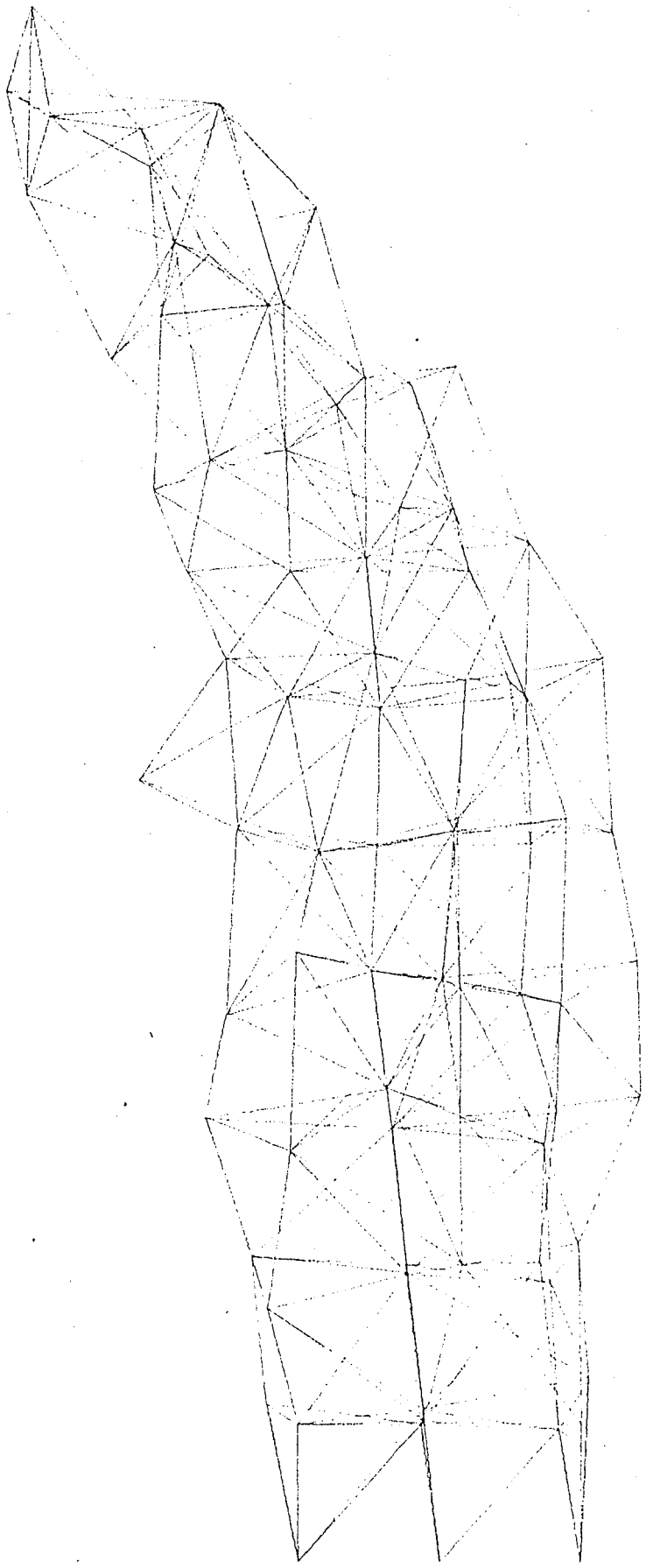
SCALE 0.60 / DATE & RUN #: DEC.10 #169
OF CELLS 94 # OF LOOPS 100

Fig. 6-8



SCALE 0.50 / DATE & RUN #: DEC.10 #169
OF CELLS 94 # OF LOOPS 278

Fig. 6-9

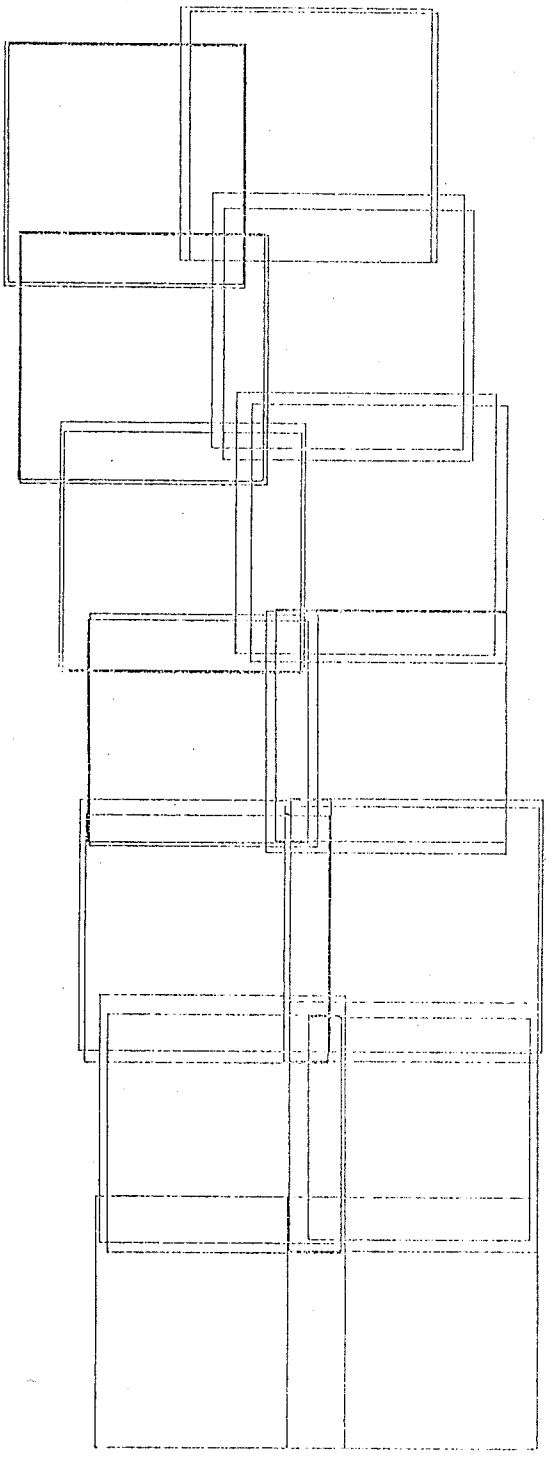


SCALE 0.70 / DATE & RUN # DEC 10 #165
OF CELLS 94 # OF LOOPS 100

SCALE 1.00 / DATE & RUN #: JUNE 29 #35
N AND NRECON 2824

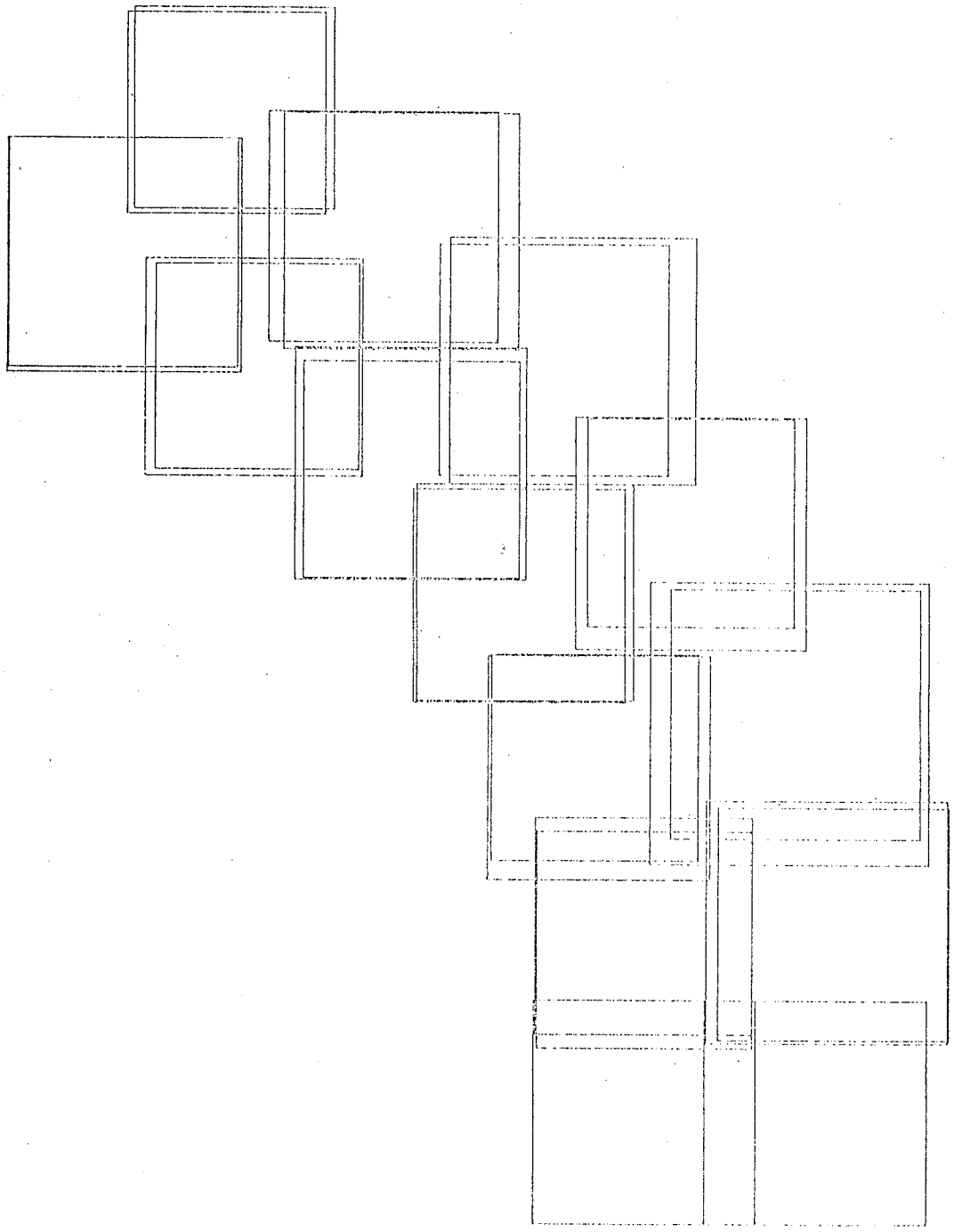
Fig. 7-1

Fig. 7-2

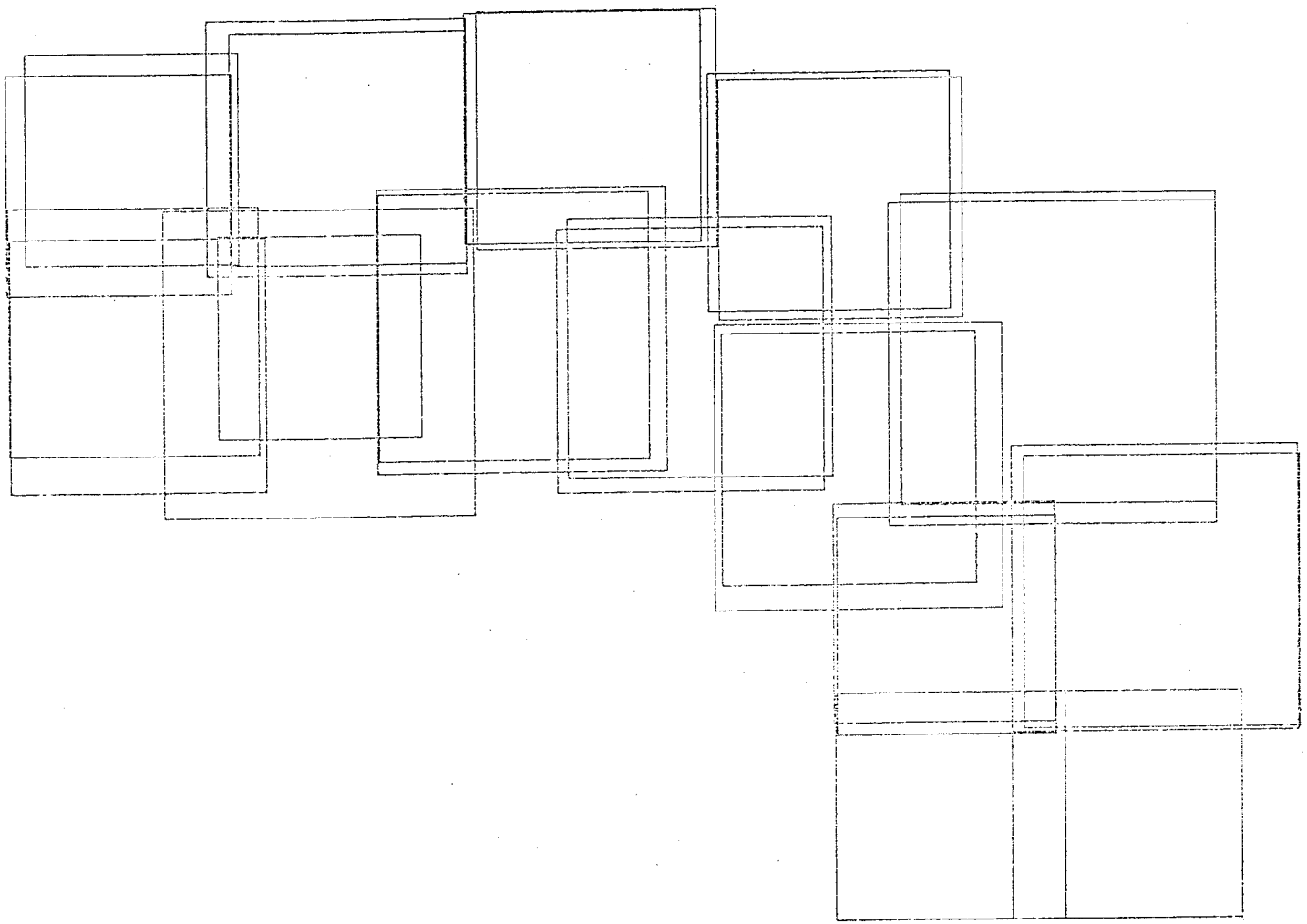


SCALE 1.00 / DATE & RUN #: JUL.17
N AND NRECON 282 #38

Fig. 7-3



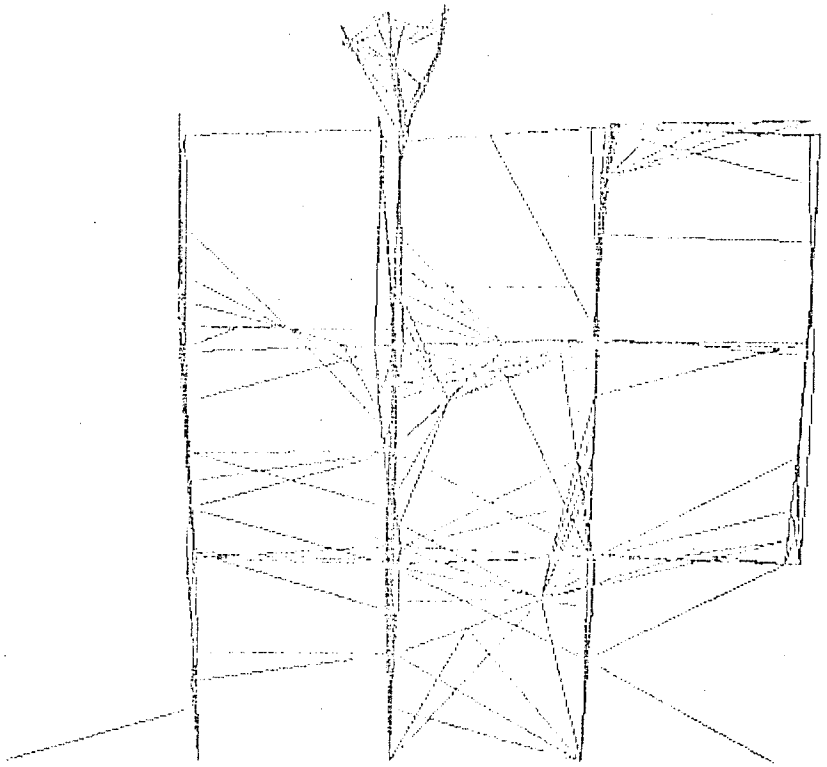
SCALE 1.00 / DATE & RUN #: JUL.17 #38
N AND NRECON 2812



SCALE 1.00 / DATE & RUN #: JUL.17 #38
N AND NRECON 2824

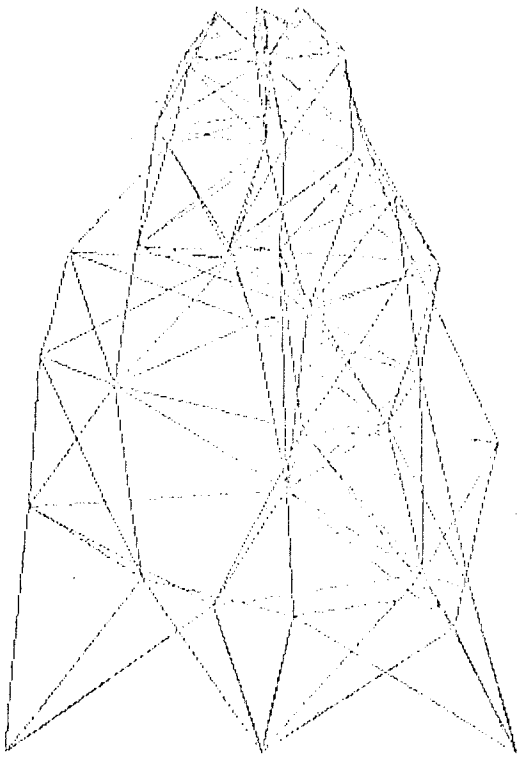
Fig. 7-4

Fig. 7-5



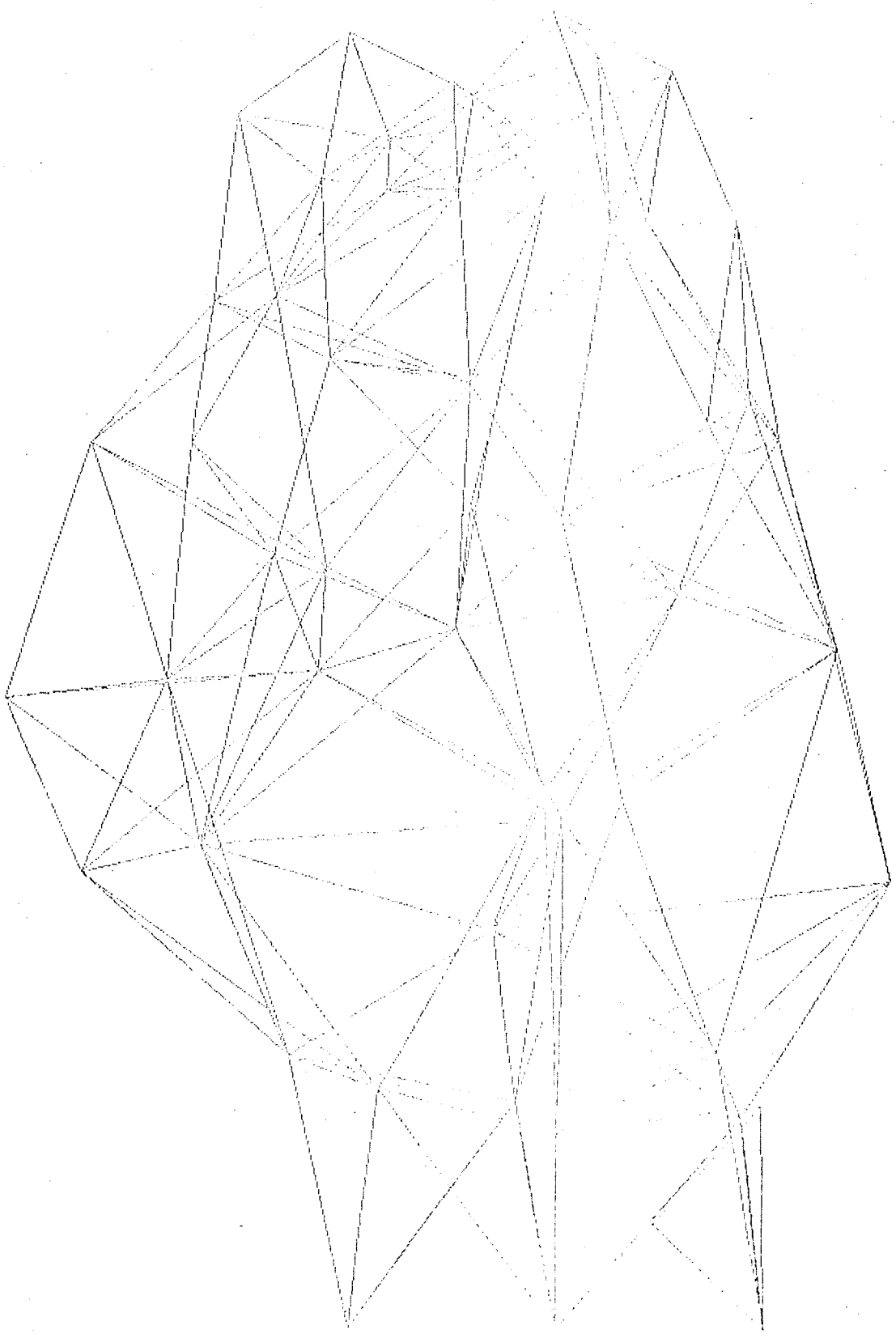
SCALE 1.00 / DATE & RUN #: JULY 27 #43
N AND NRECON 198 4

Fig. 7-6



SCALE 0.89 / DATE & RUN #: AUG. 15 #63
N AND. NRECON 80 5

Fig. 7-7



SCALE 0.89 / DATE & TIME :: AUG. 15 '83
N AND NRECON 82 75

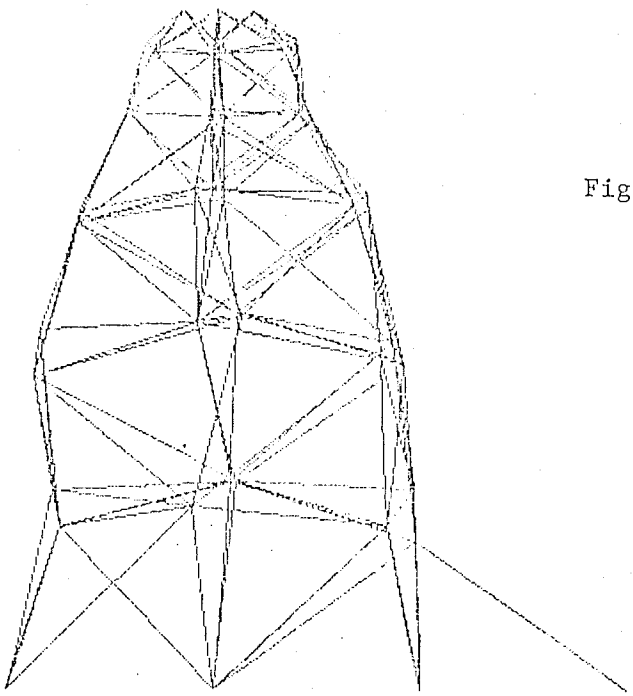
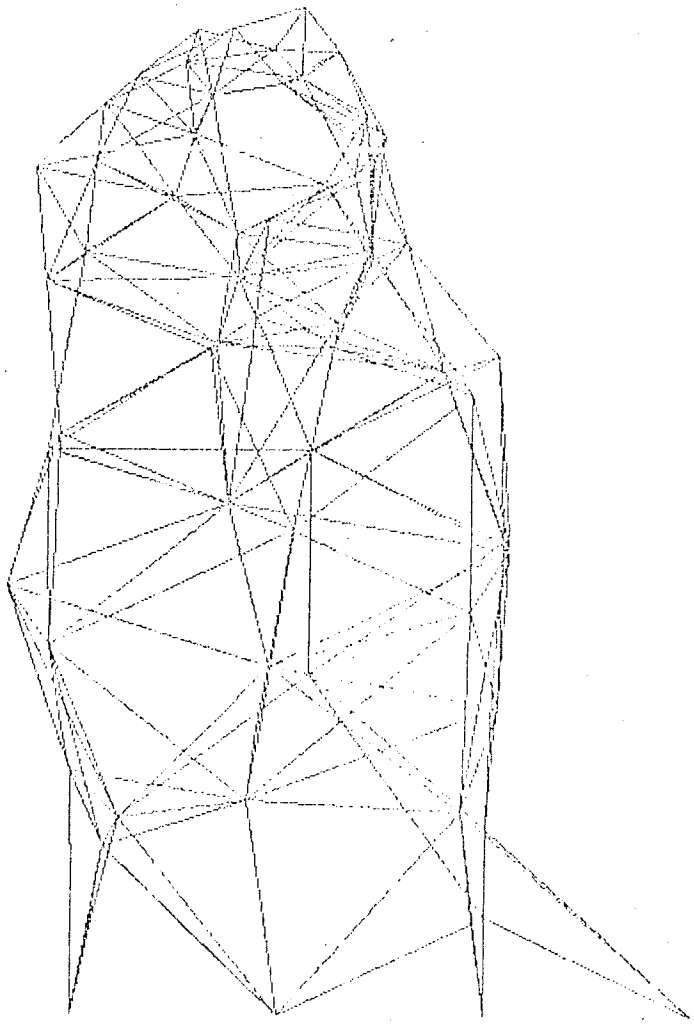


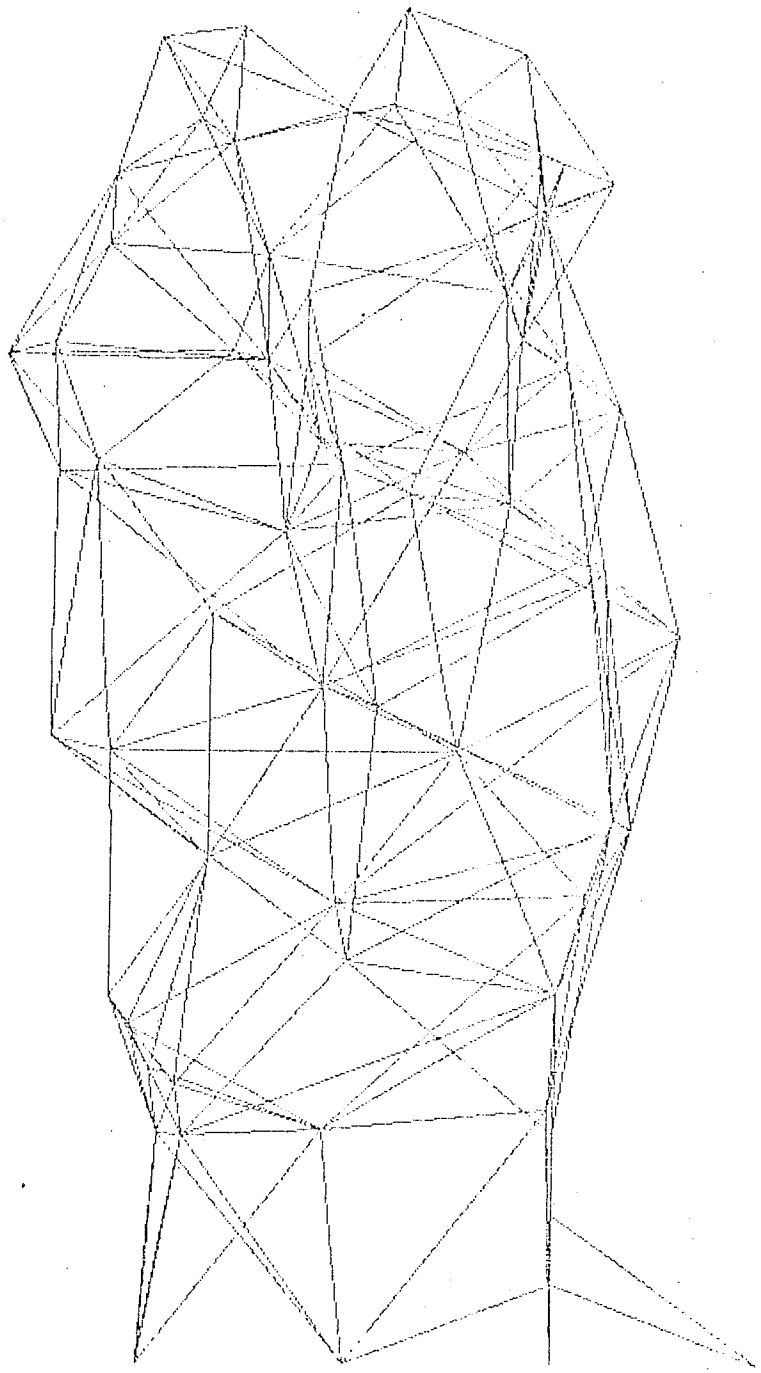
Fig. 7-8

SCALE 0.90 / DATE & RUN #: SEPT.11 #66
N AND NRECON 80 0



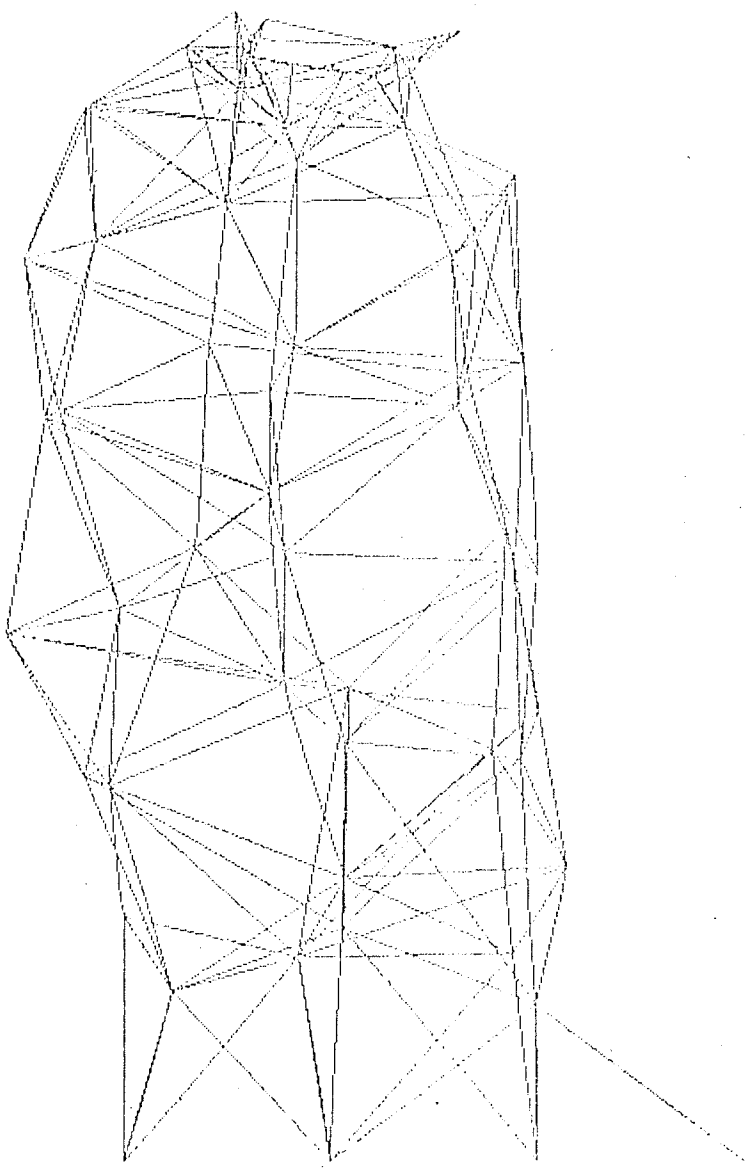
SCALE 0.90 / DATE & RUN #: SEPT.11 #66
N AND NRECON 90 20

Fig. 7-9



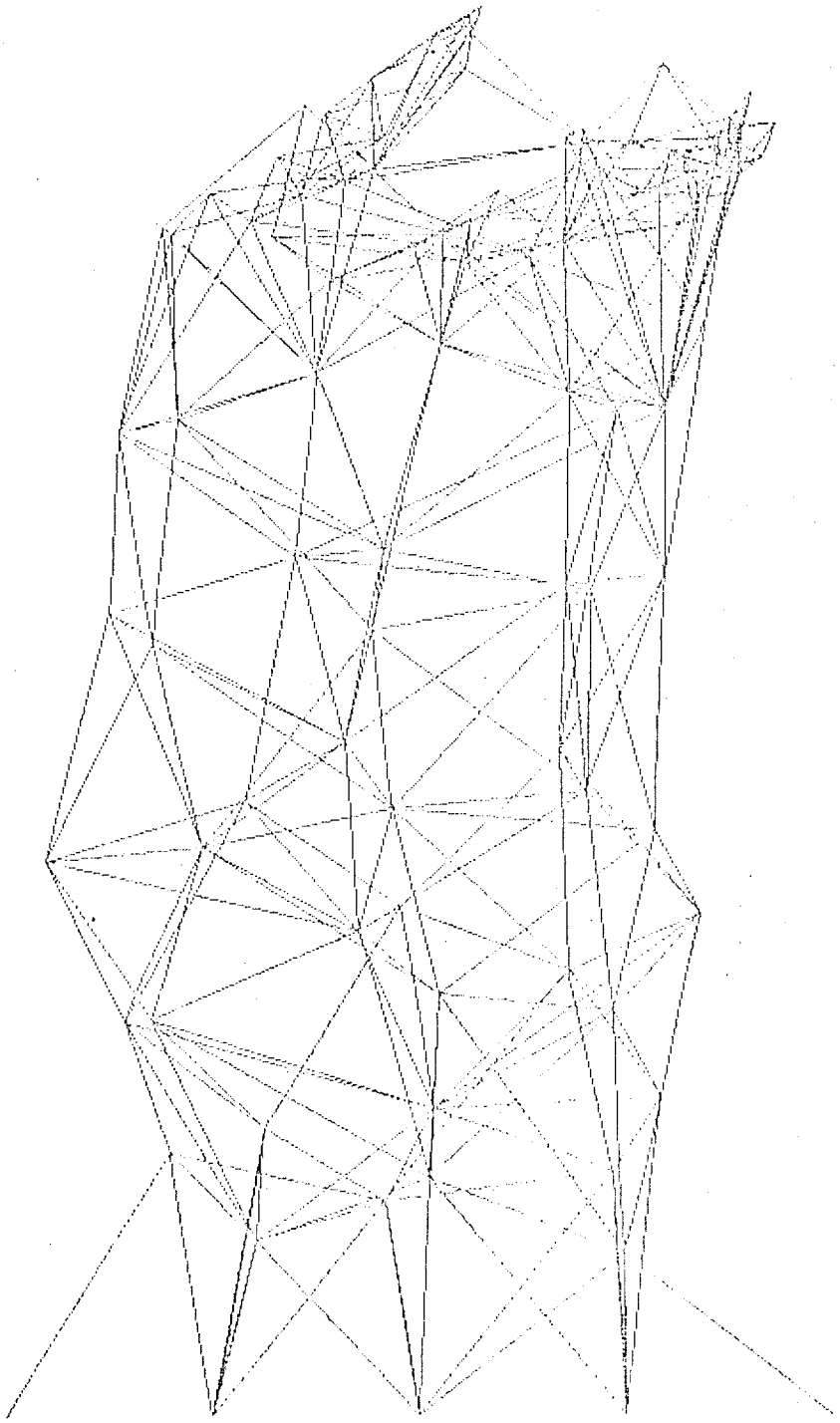
SCALE 0.90 / DATE & RUN #: SEPT.11 #66
N AND NRECON 95 50

Fig. 7-10



SCALE 0.90 / DATE & RUN #: SEPT.15 #72
N AND NRECON 94 50

Fig. 7-11



SCALE 0.90 / DATE & RUN #: SEPT.15 #72
N AND NRECON 132 100

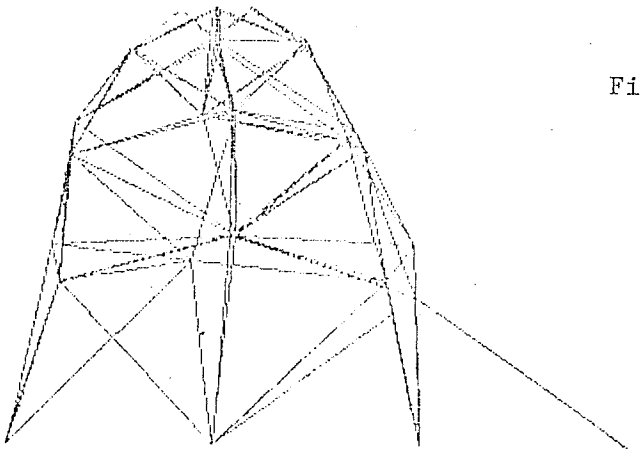
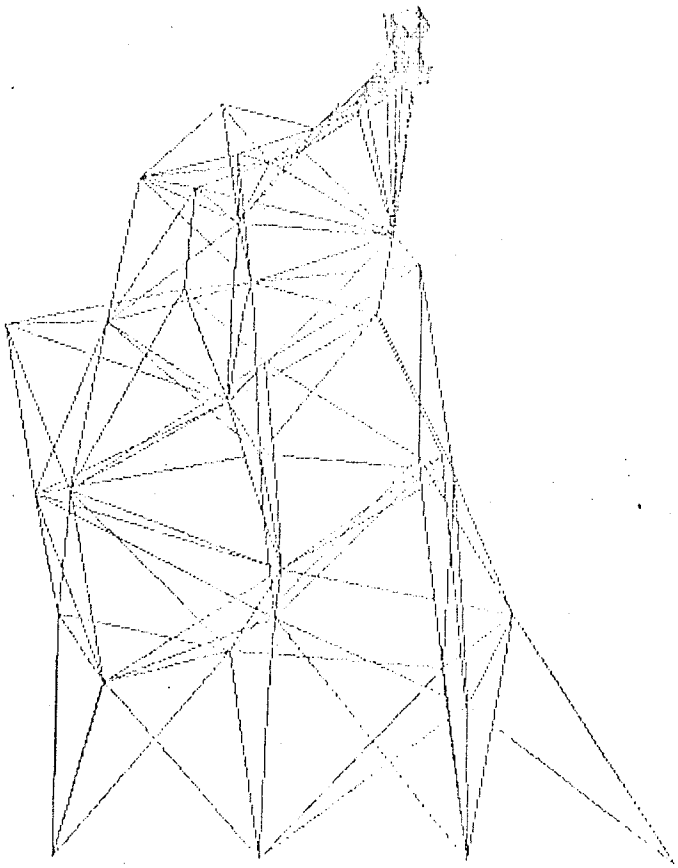


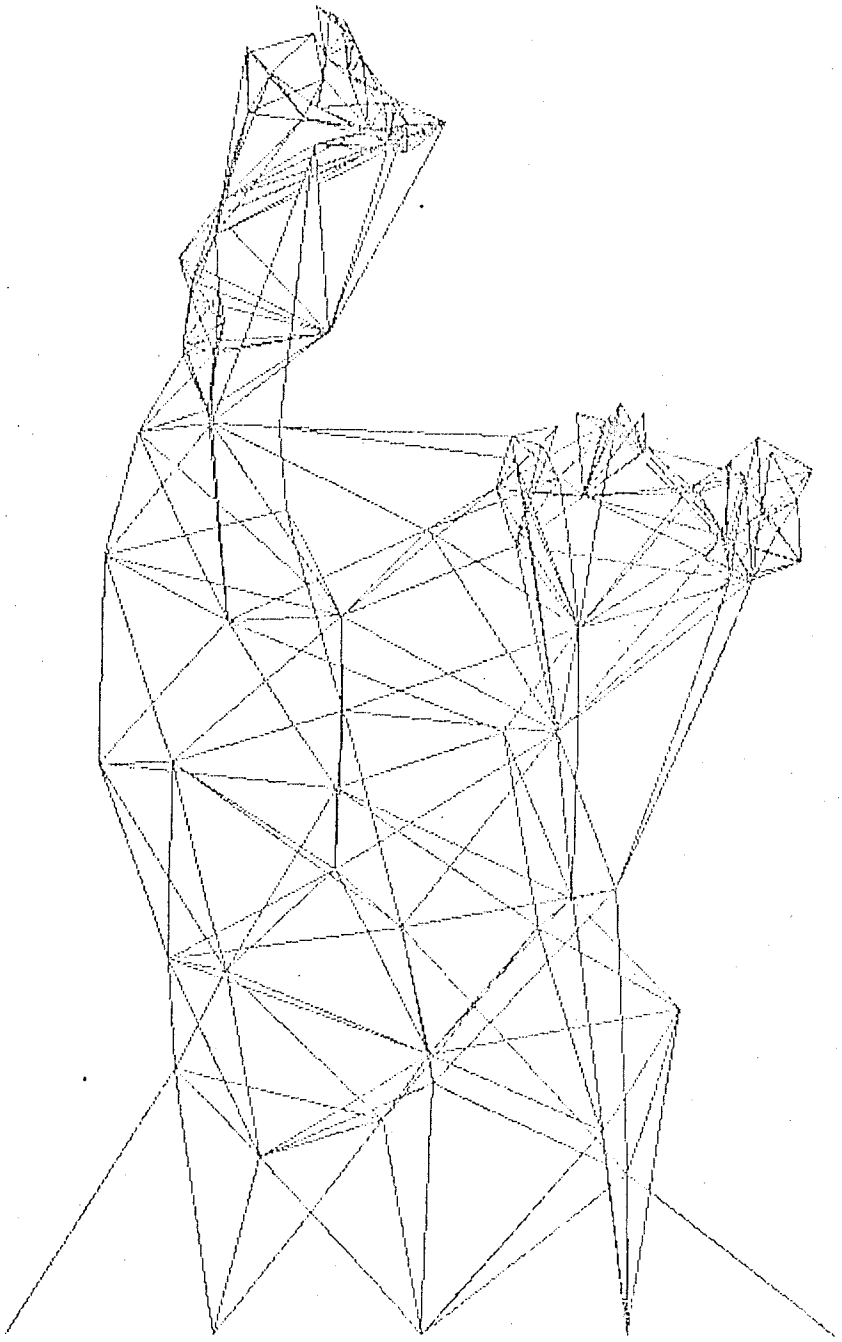
Fig. 7-12

SCALE 0.90 / DATE & RUN #: SEPT.16 #74
N AND NRECON 62 0



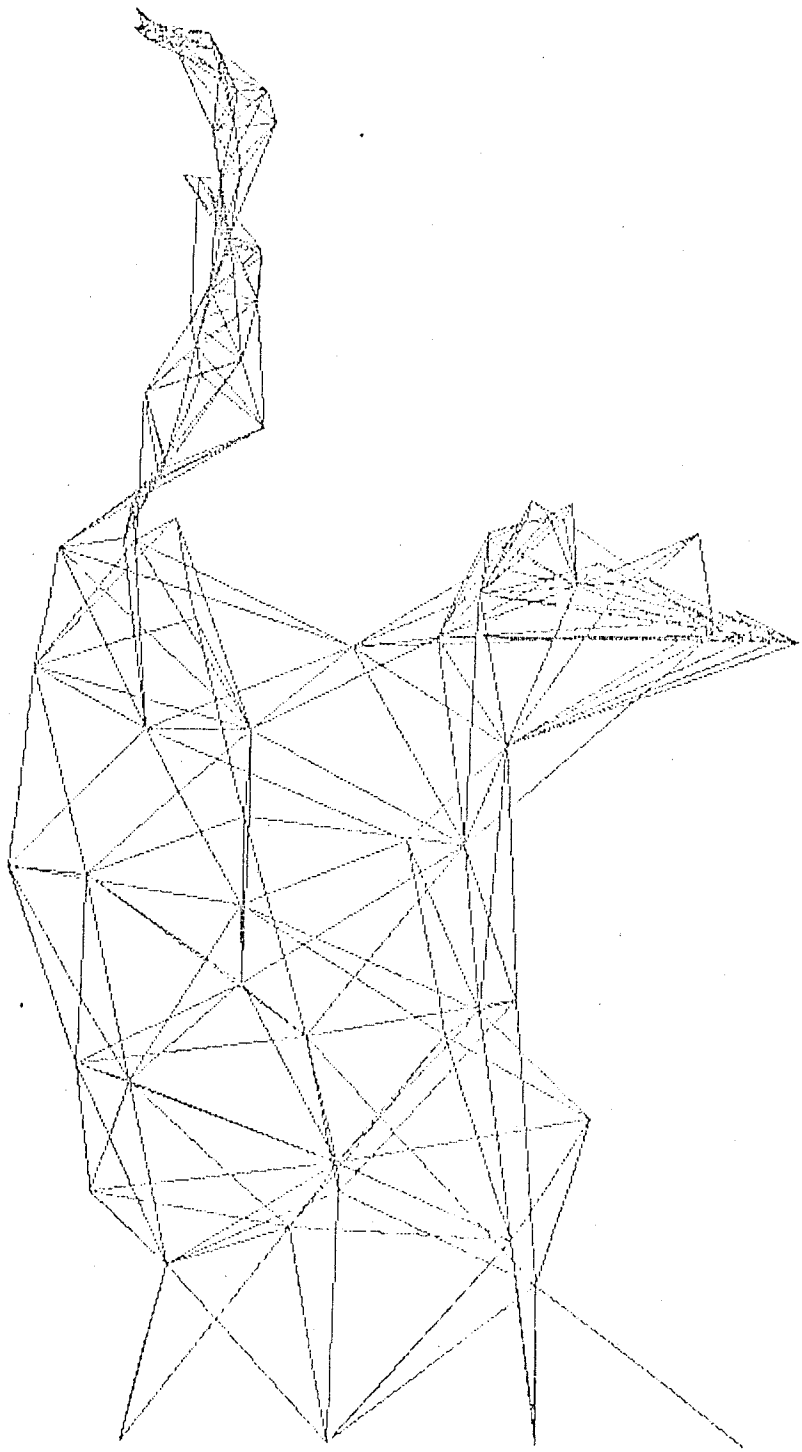
SCALE 0.90 / DATE & RUN #: SEPT.16 #74
N AND NRECON 116 40

Fig. 7-13



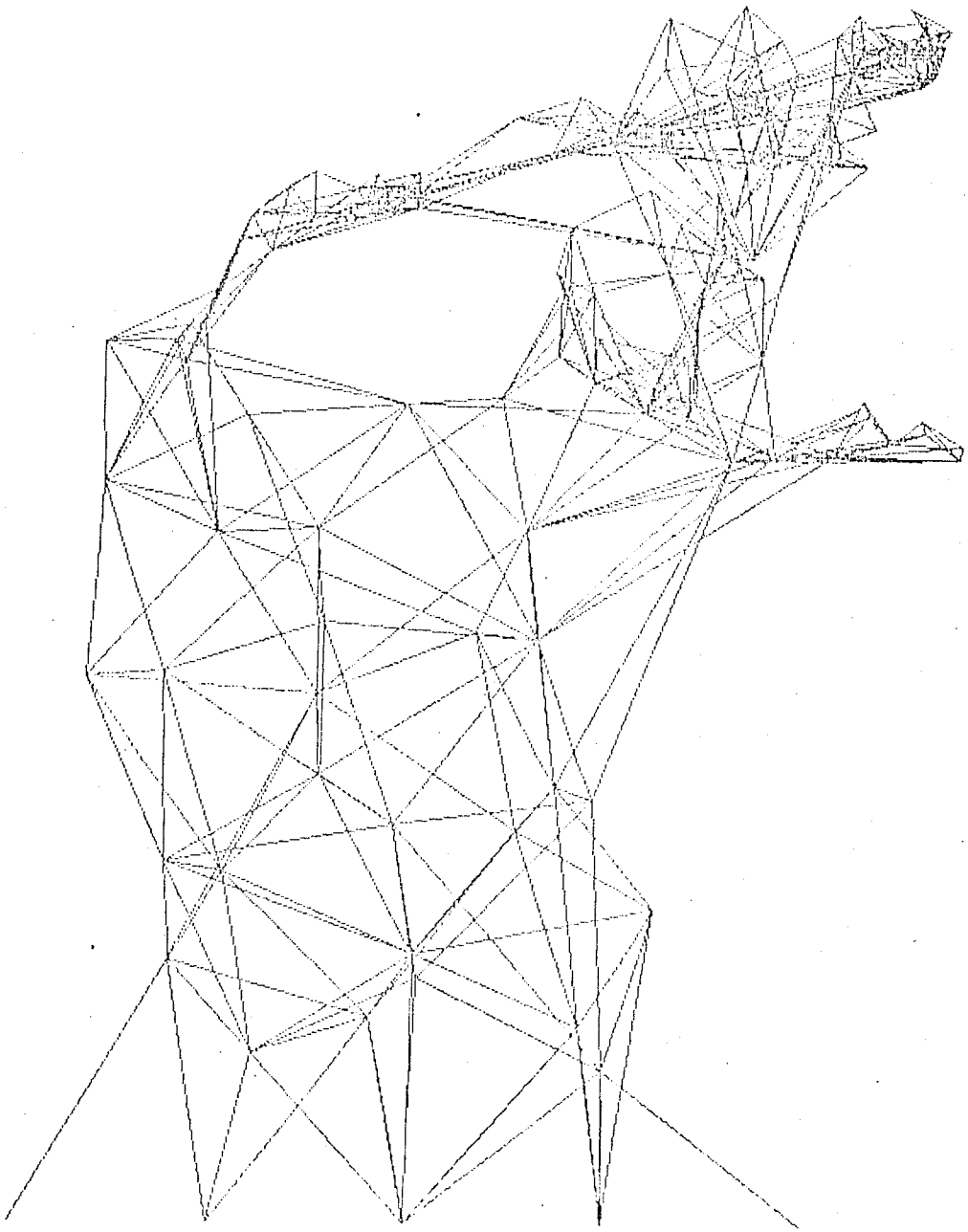
SCALE 0.90 / DATE & RUN #: SEPT.19 #78
N AND NRECON 144 120

Fig. 7-14



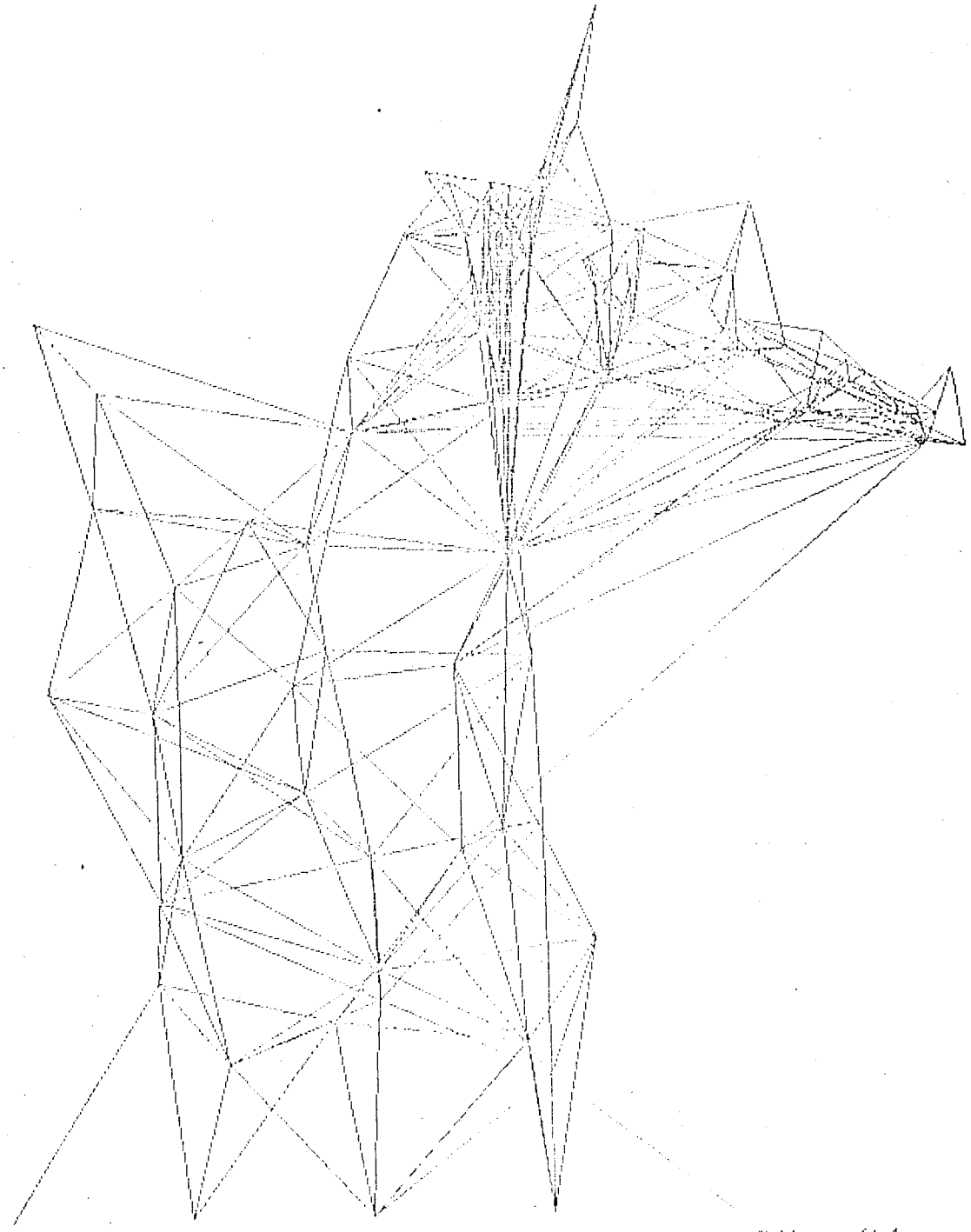
SCALE 0.90 / DATE & RUN #: SEPT.19 #79
N AND NRECON 125 120

Fig. 7-15



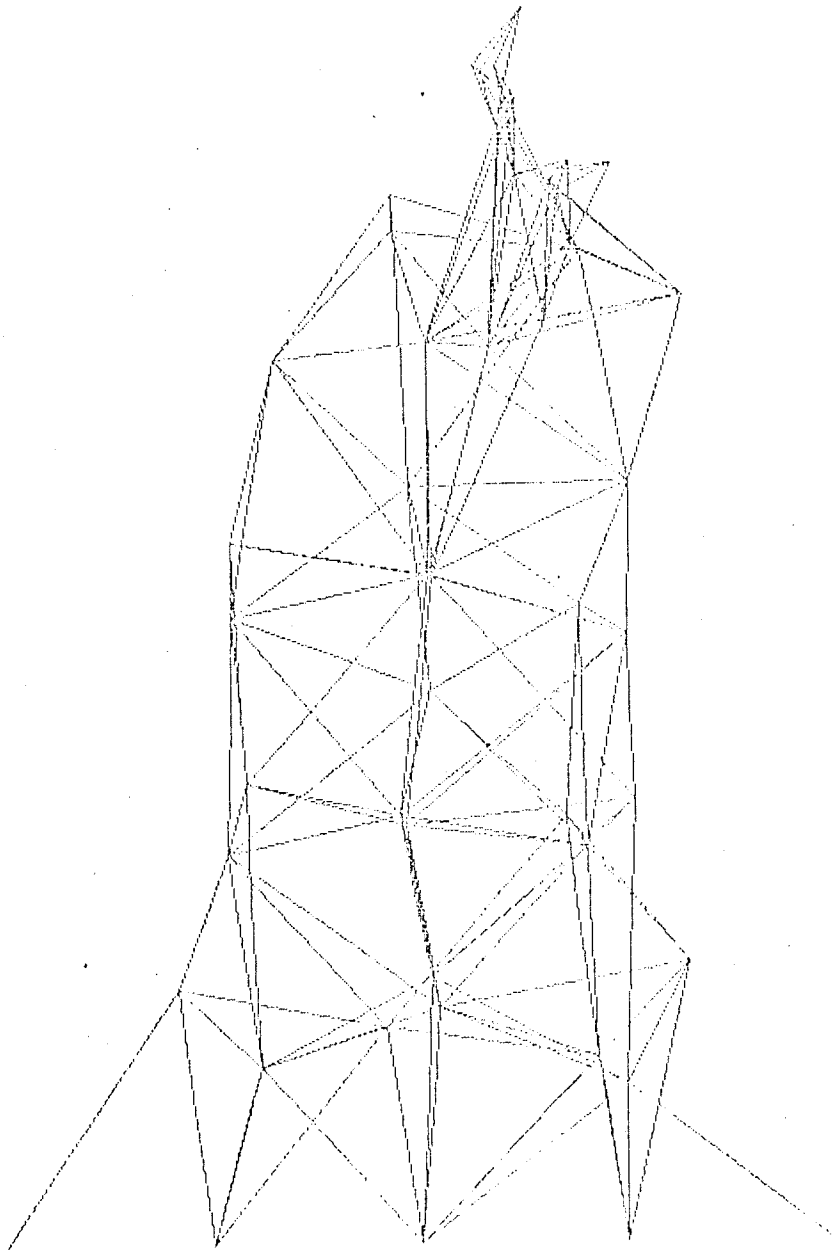
SCALE 0.90 / DATE & RUN #: SEPT.19 #79
N AND NRECON 177 150

Fig. 7-16

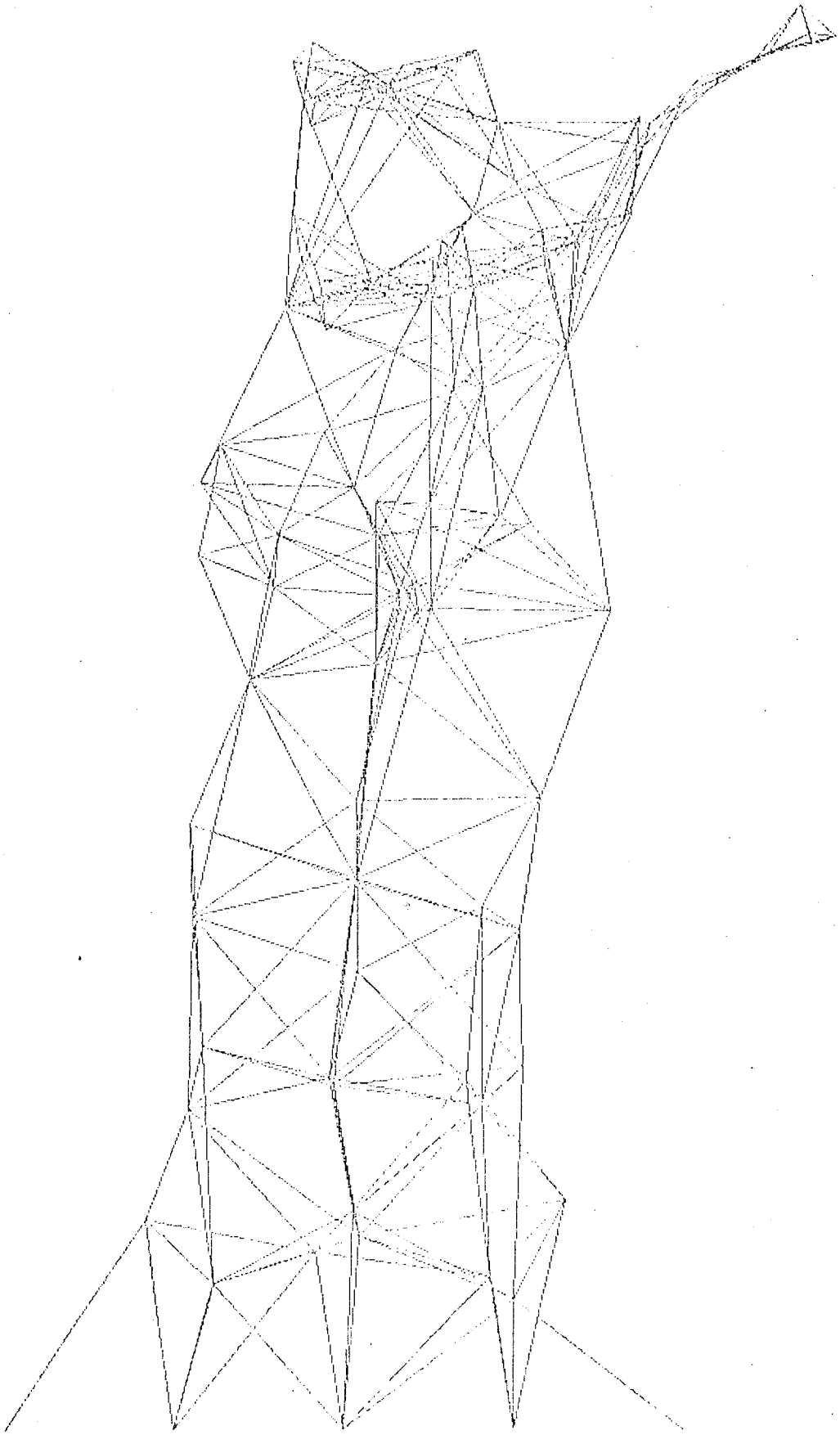


SCALE 0.9m / DATE & RUN #: SEP1.24 #81
N AND NRECON 123 210

Fig. 7-17

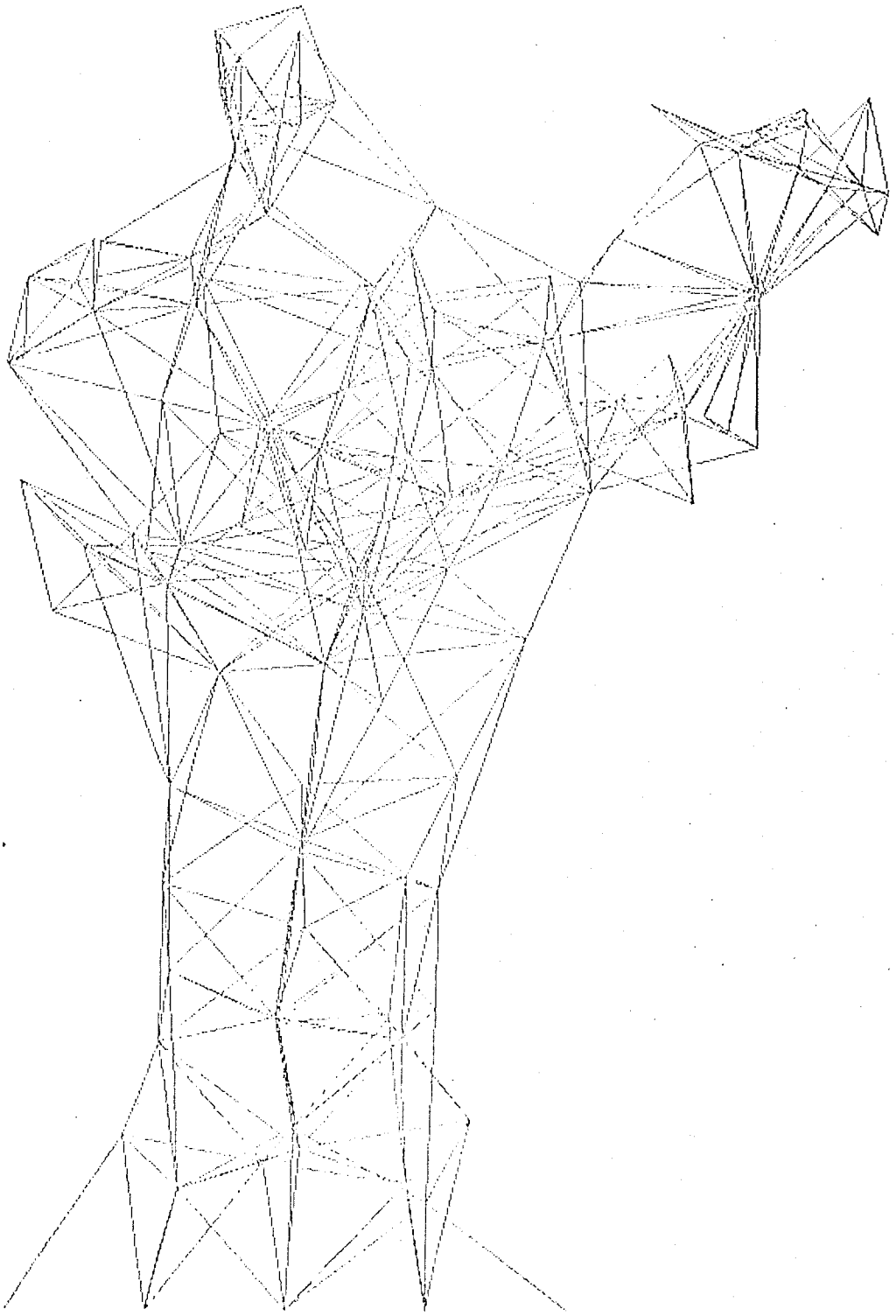


SCALE 0.90 / DATE & RUN #: OCT.31 #96
N AND NRECON 78 150



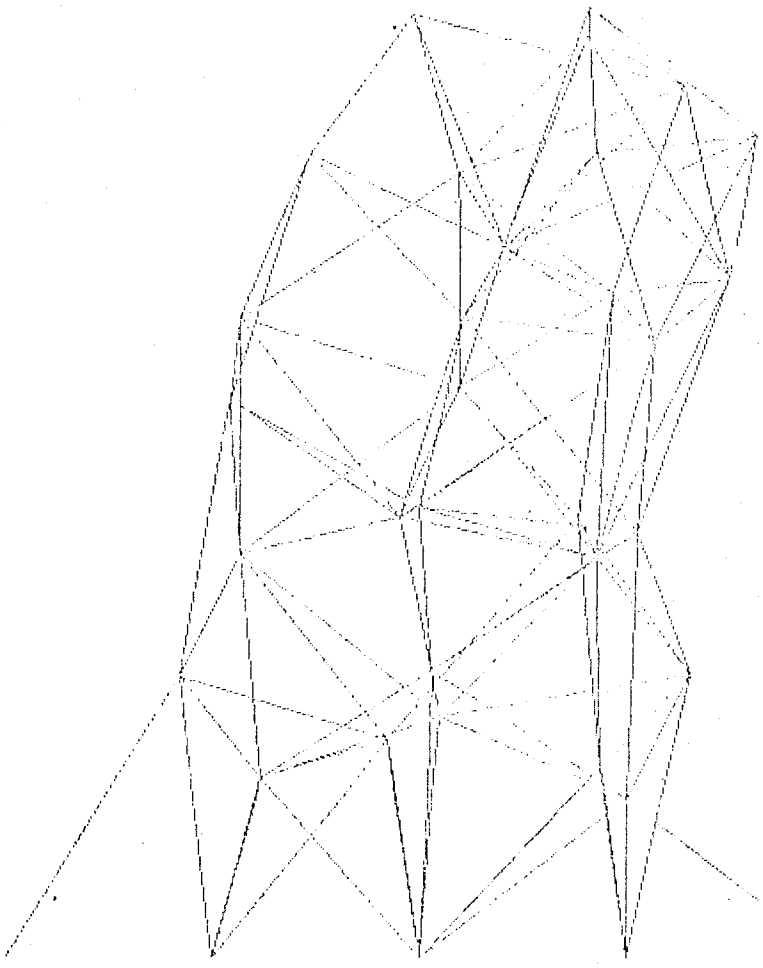
SCALE 0.90 / DATE & RUN #: OCT.31 #96
N AND NRECON 131 240

Fig. 7-19



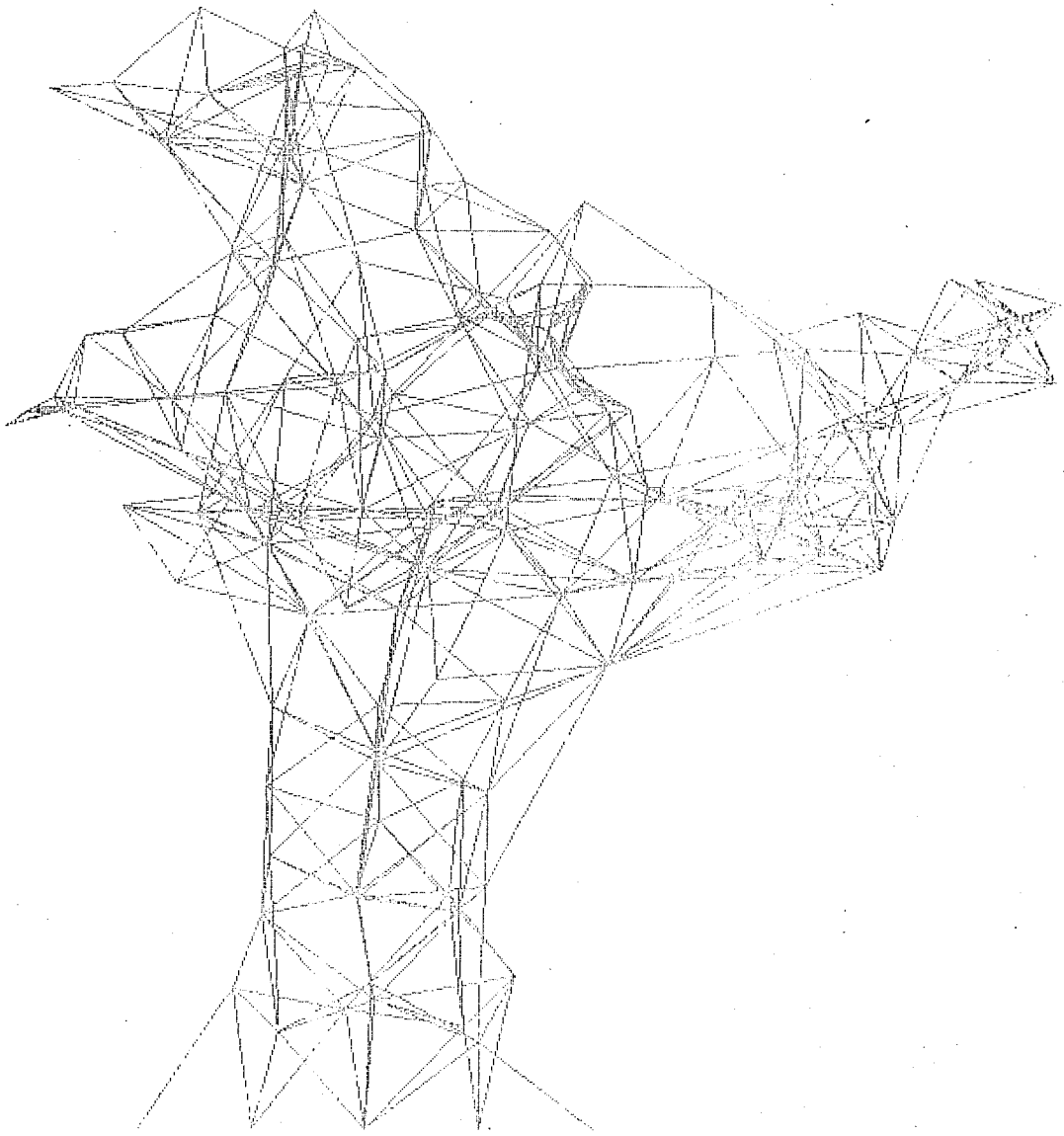
SCALE 0.70 / DATE & RUN #: OCT.31 #96
N AND NRECON 154 300

Fig. 7-20



SCALE 0.90 / DATE & RUN #: NOV 10 #112
OF CELLS 62 # OF LOOPS 210

Fig. 7-21



SCALE 0.50 / DATE & RUN #: NOV 11 #113
OF CELLS 192 # OF LOOPS 400

Fig. 7-22

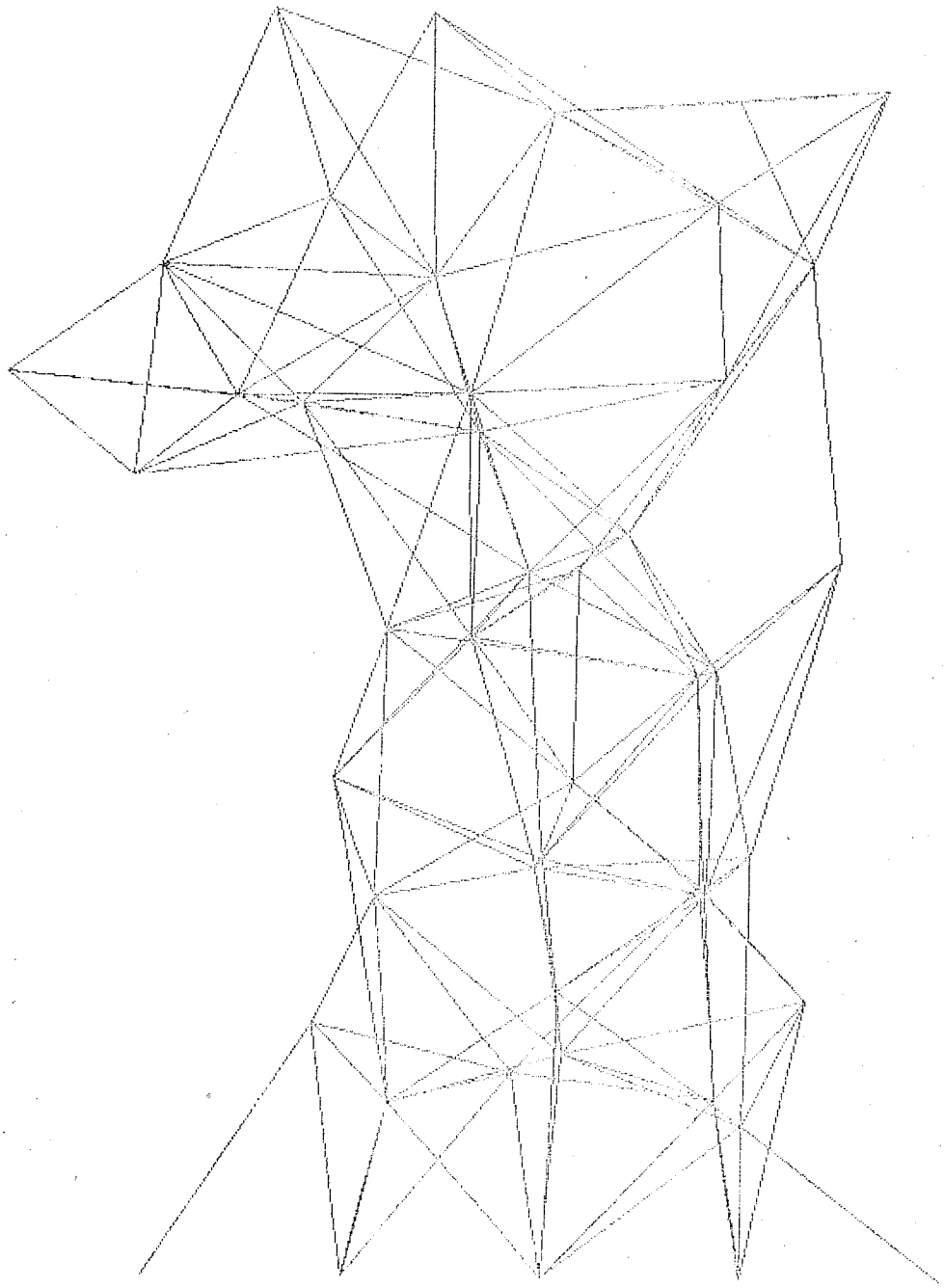
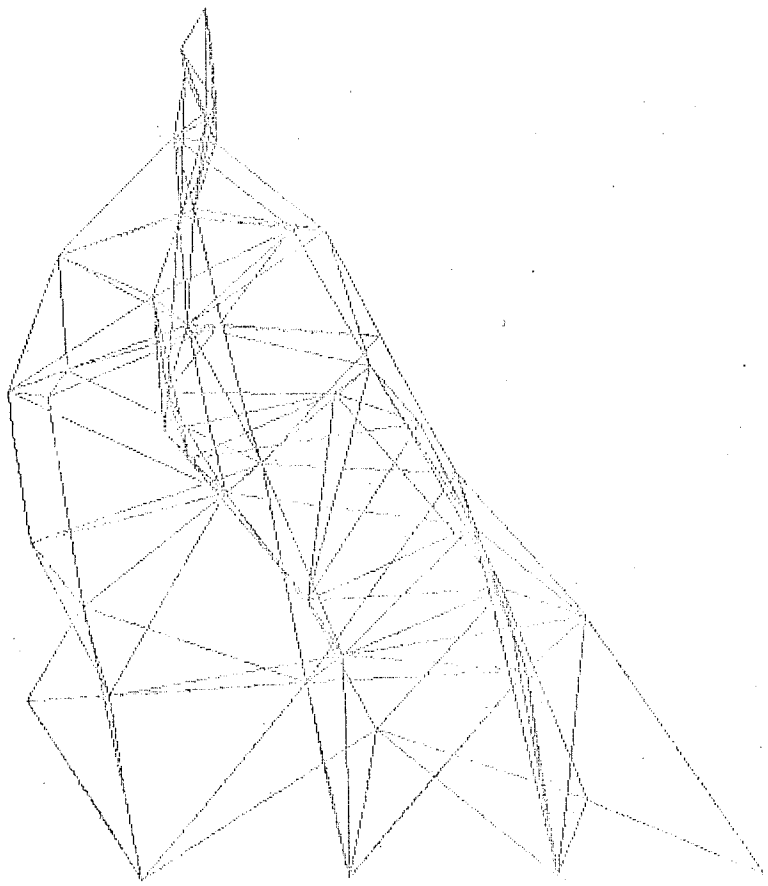
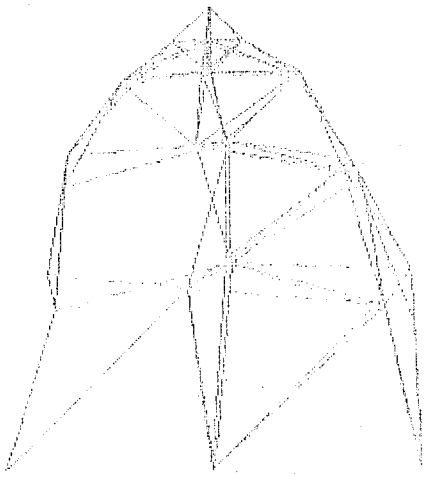


Fig. 7-23

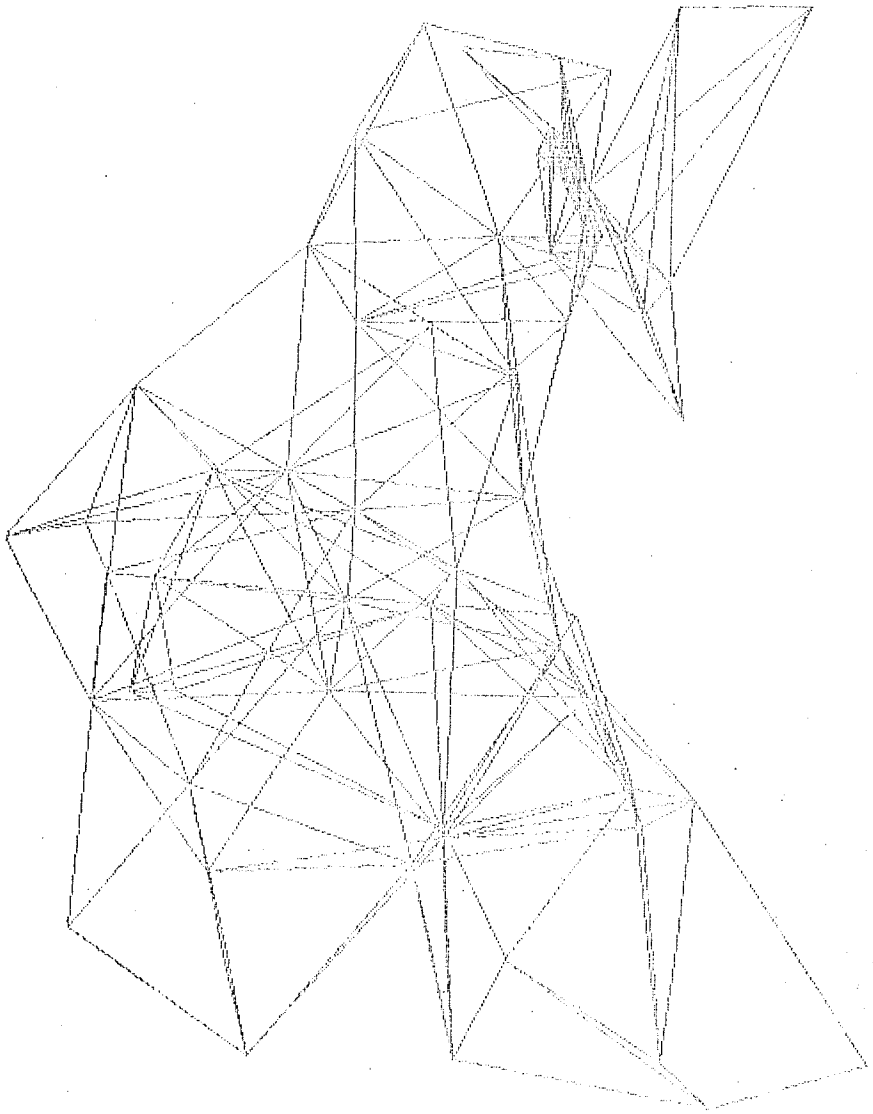


SCALE 0.90 / DATE & RUN #: DEC.2 #145
OF CELLS 57 # OF LOOPS 50



SCALE 0.90 / DATE & RUN #: DEC.2 #145
OF CELLS 58 # OF LOOPS 0

Fig. 7-24



SCALE 0.90 / DATE & RUN #: DEC.2 #145
OF CELLS 76 # OF LOOPS 150

Fig. 7-25

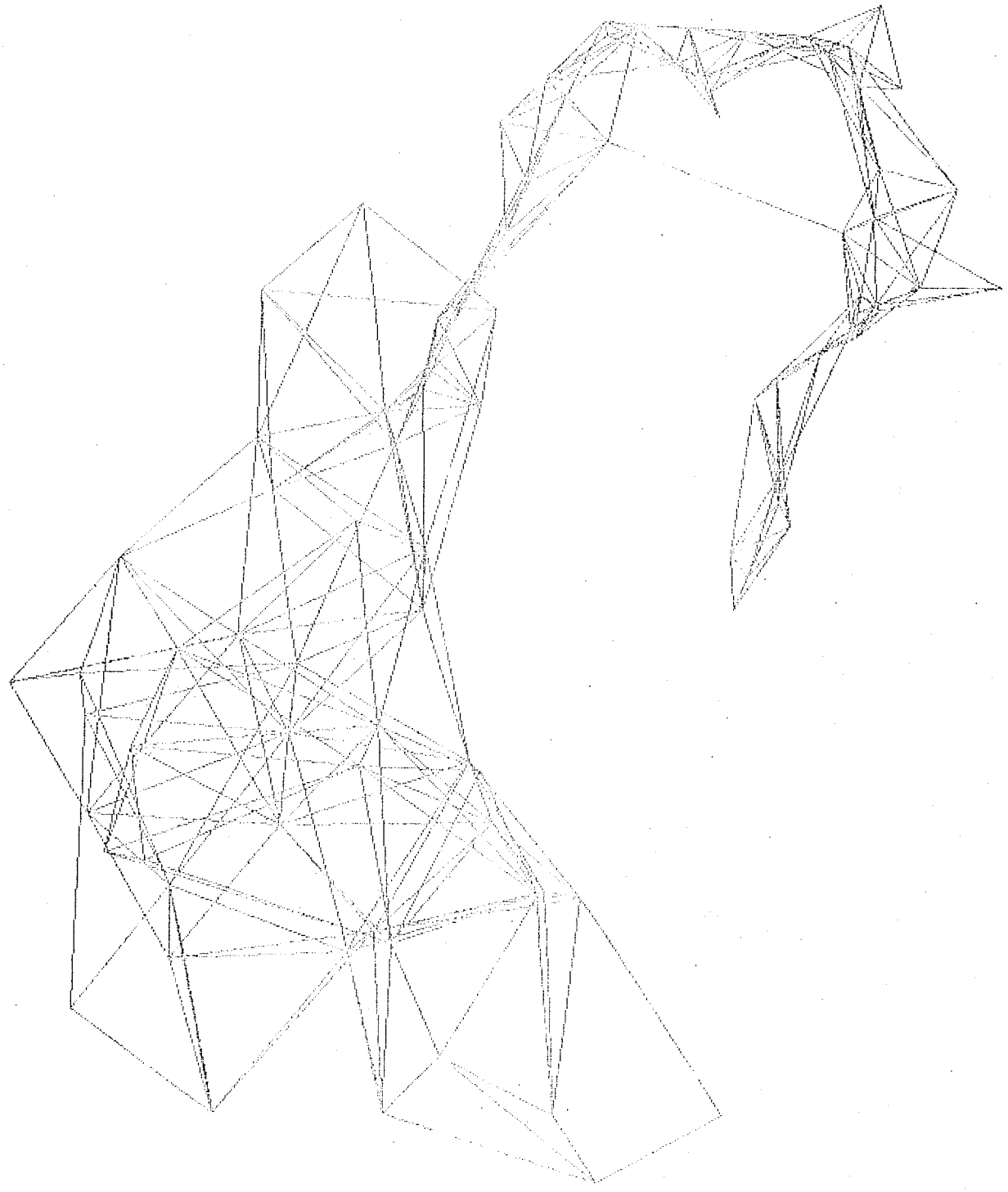
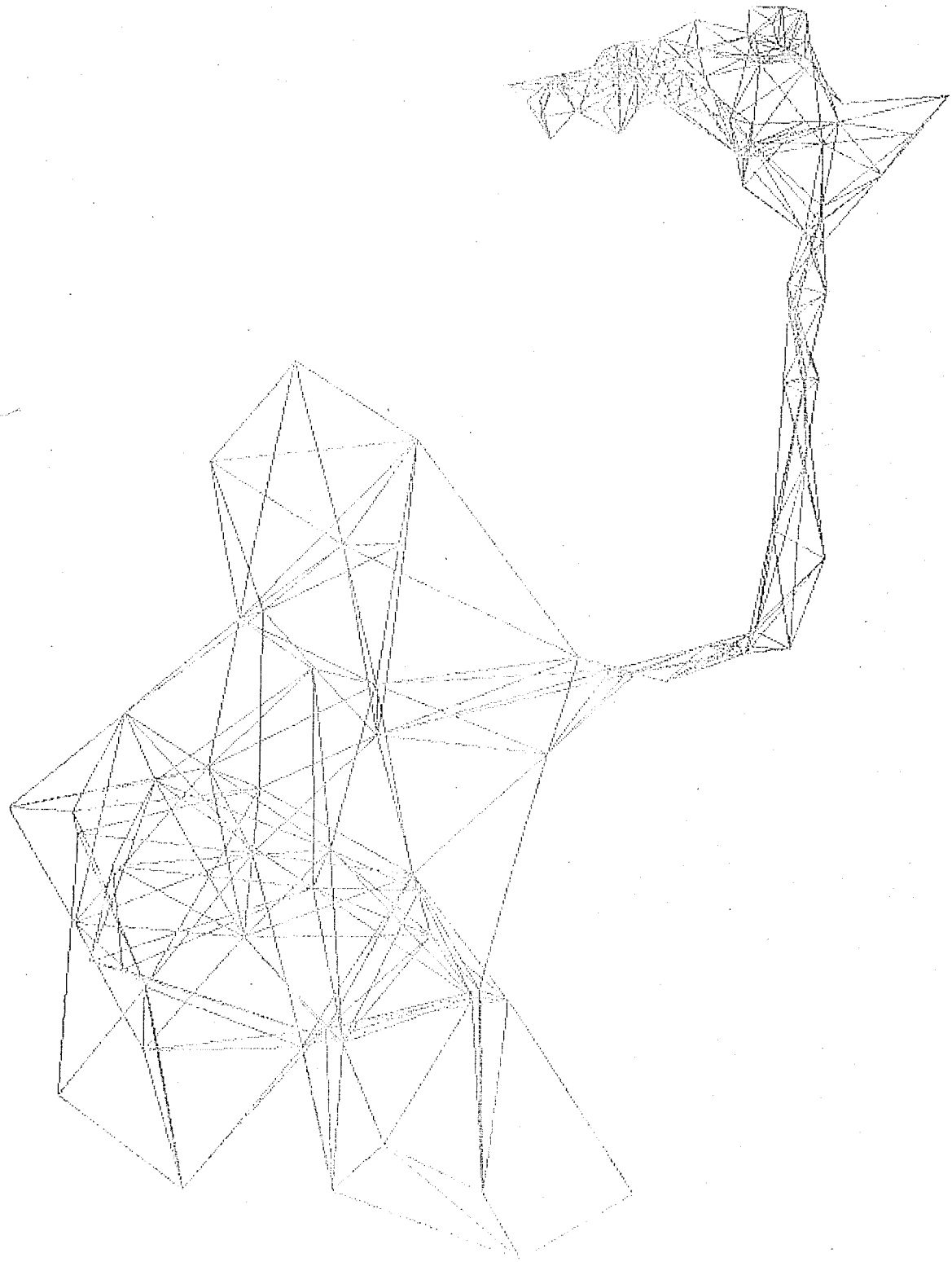
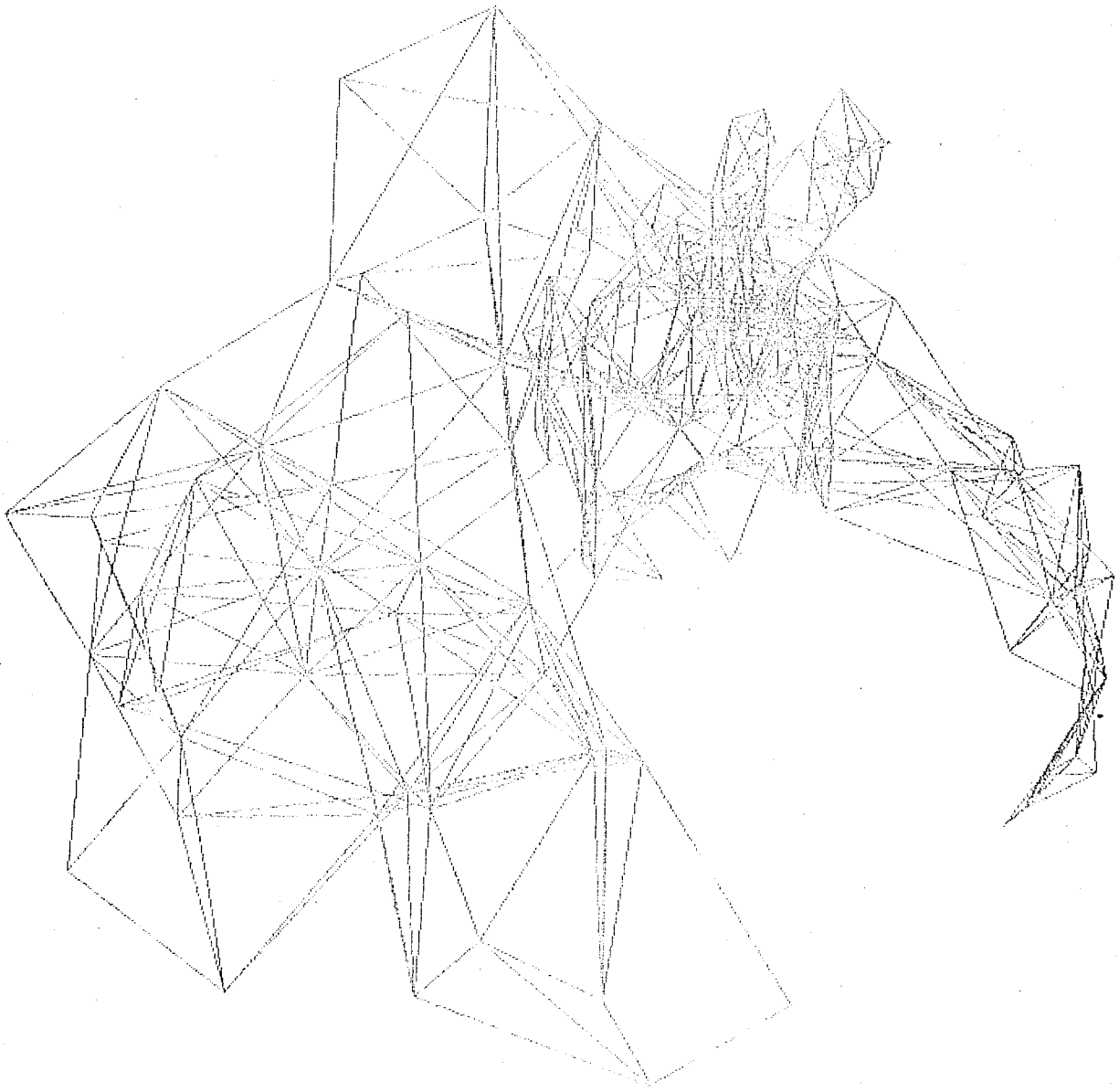


Fig. 7-26



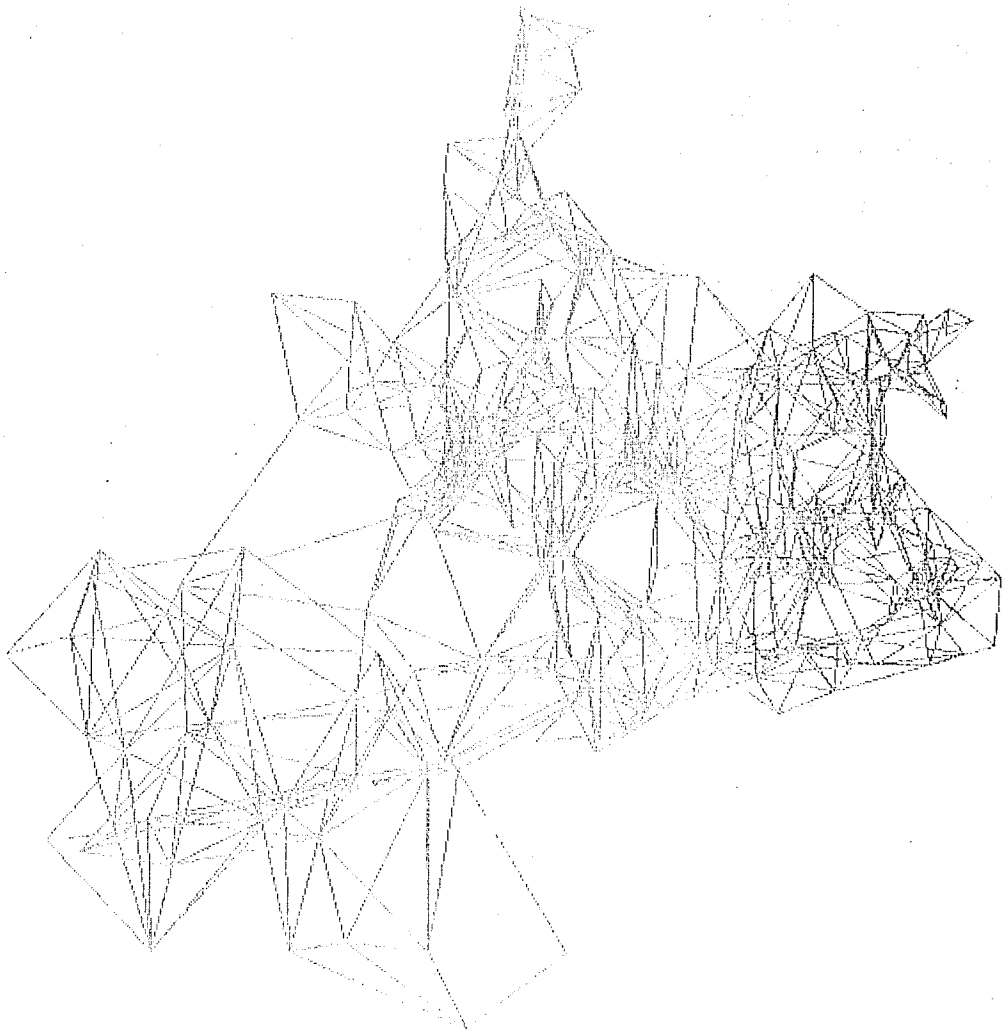
SCALE 0.80 / DATE & RUN #: DEC.2 #145
OF CELLS 147 # OF LOOPS 250

Fig. 7-27



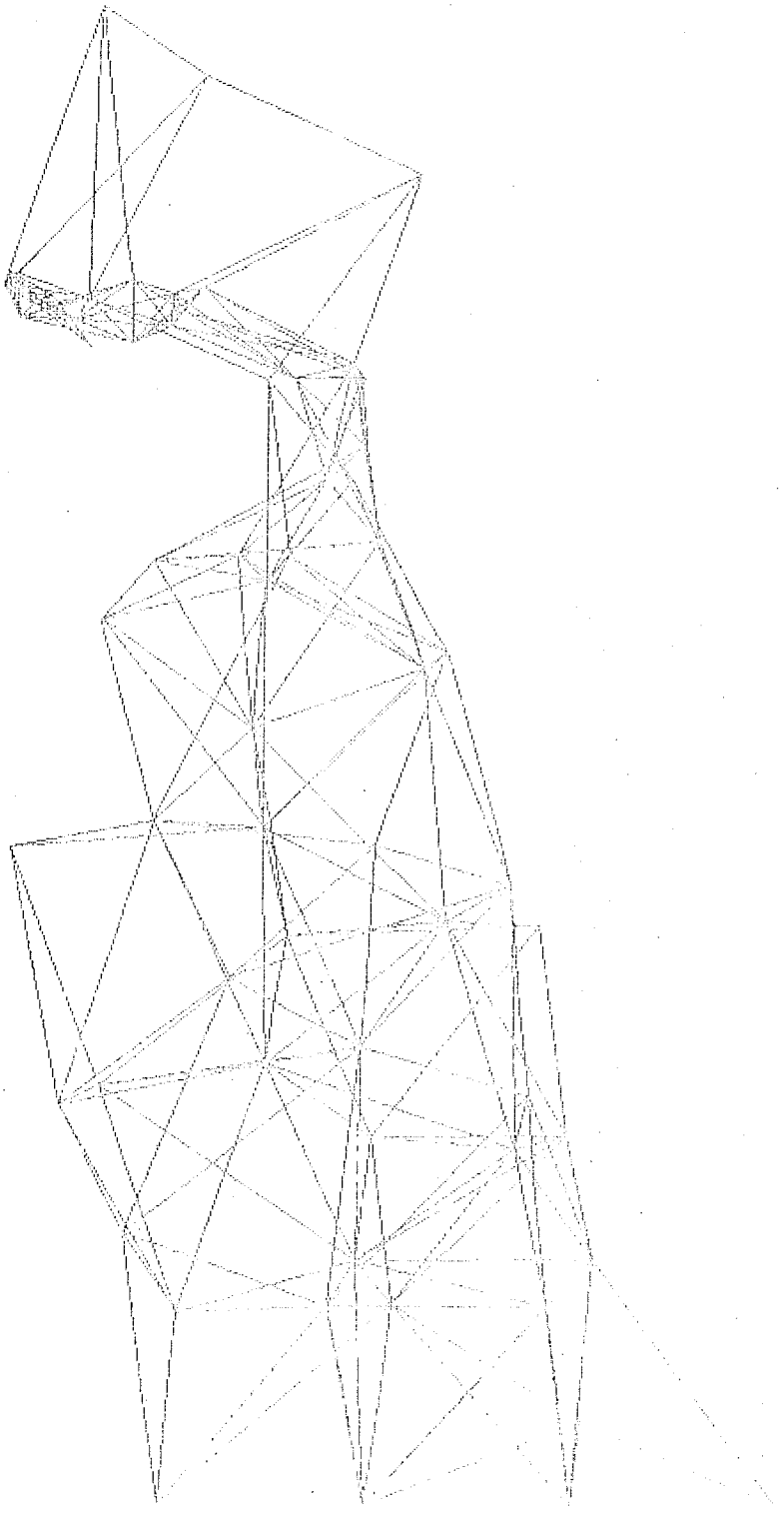
SCALE 0.90 / DATE & RUN #: DEC.2 #145
OF CELLS 201 # OF LOOPS 301

Fig. 7-28



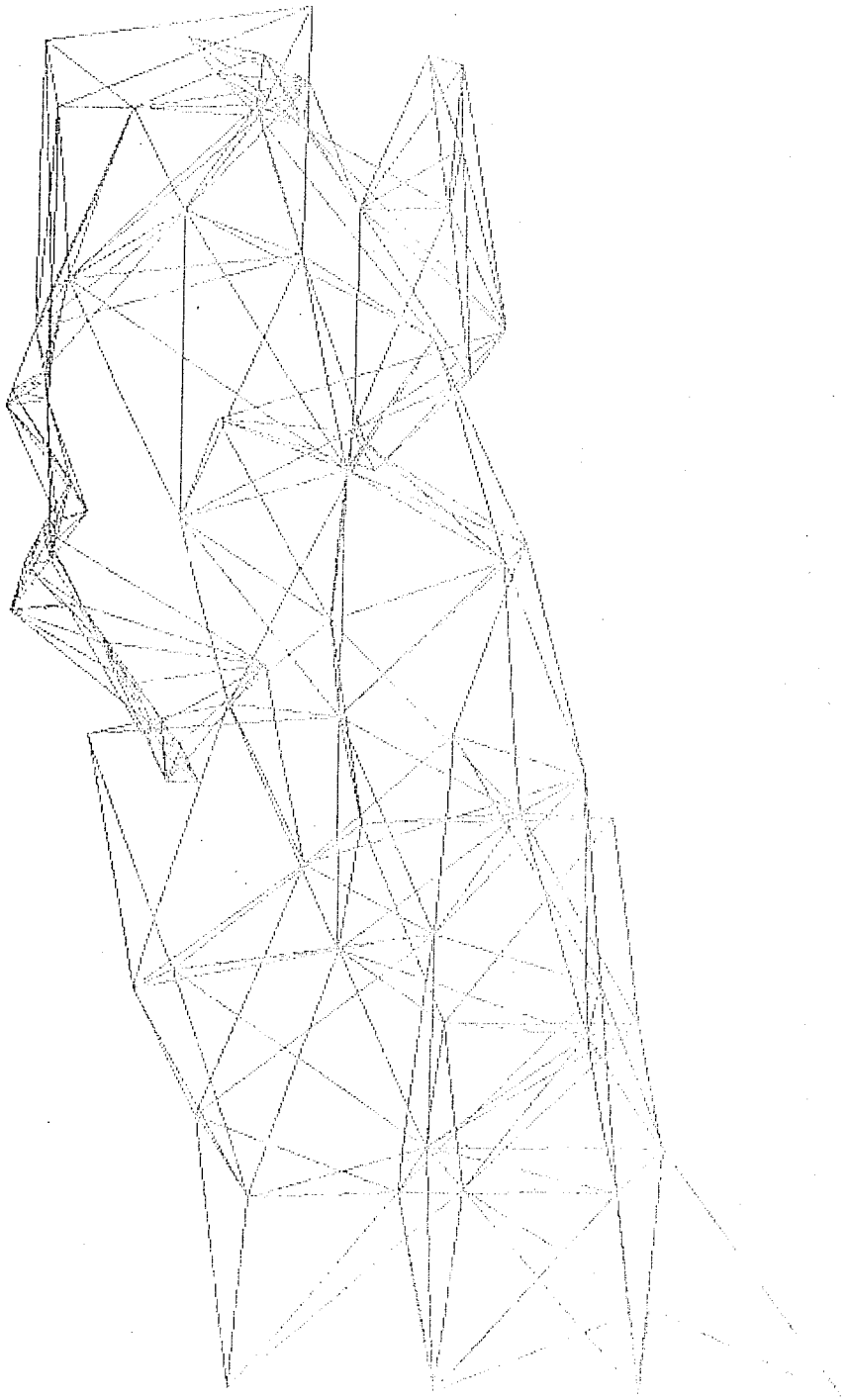
SCALE 0.60 / DATE & RUN #: DEC.2 #145
OF CELLS 333 # OF LOOPS 400

Fig. 7-29



SCALE 0.90 / DATE & RUN #: [illegible] #1169
OF CELLS 80 # OF LAOR'S 150

Fig. 7-30



SCALE 0.90 / UNIT - RUN #: [illegible] #149
OF CELLS 99 # OF LOOPS 200

Fig. 7-31

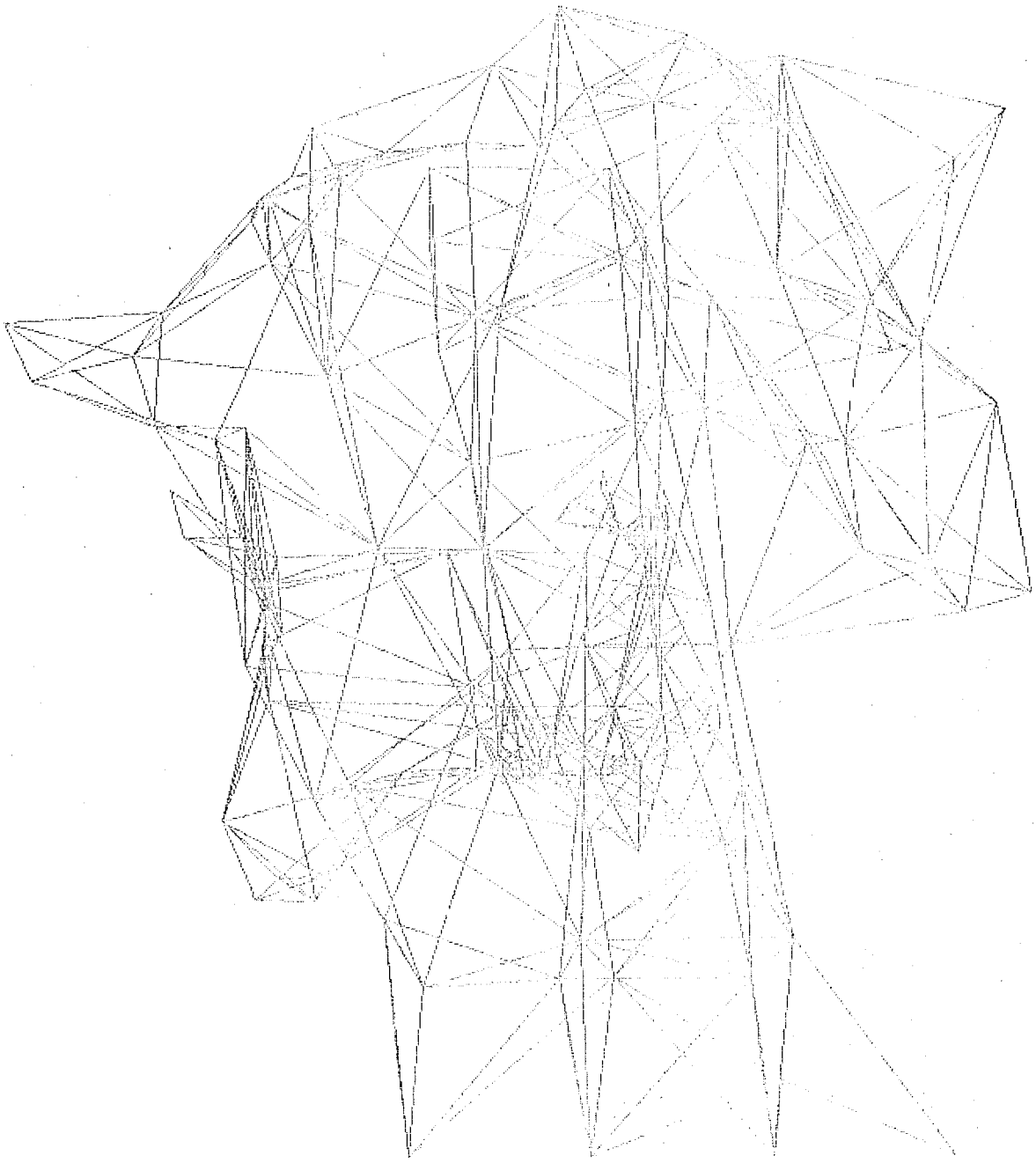
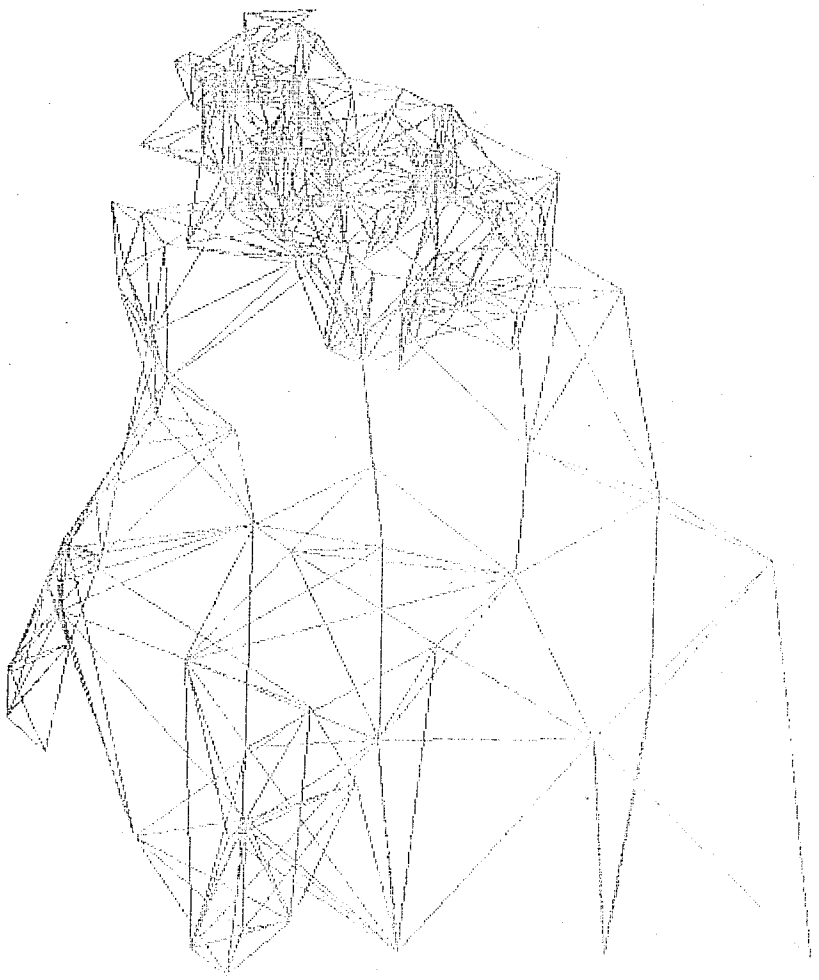


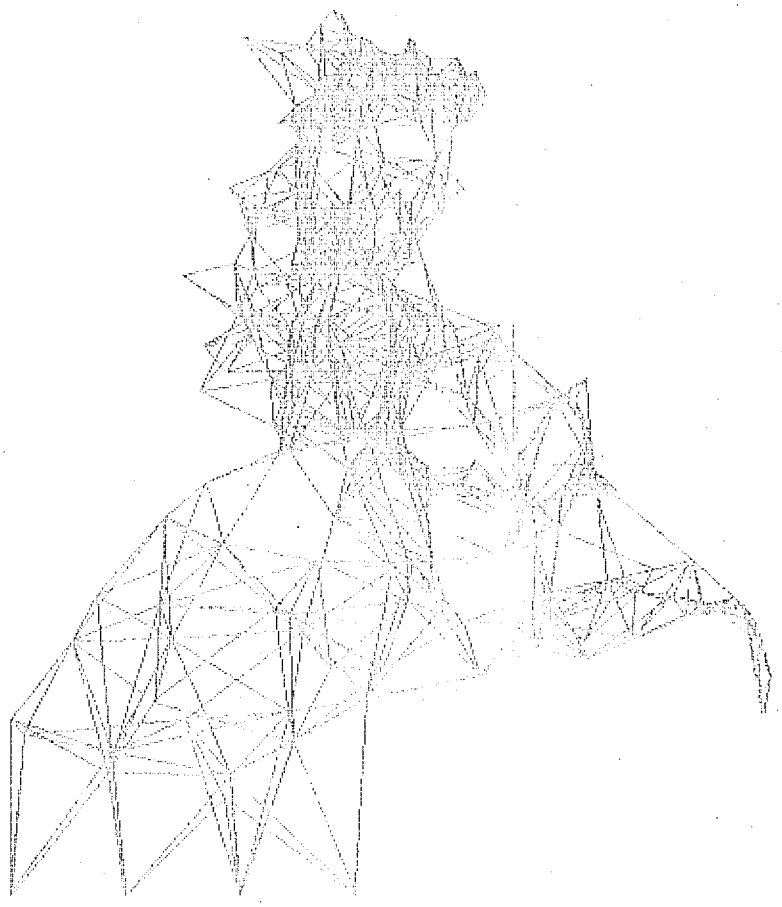
Fig. 7-32



SCALE 0.90 / DATE & RUN #: DEC 3 #149
OF CELLS 209 # OF LOOPS 250

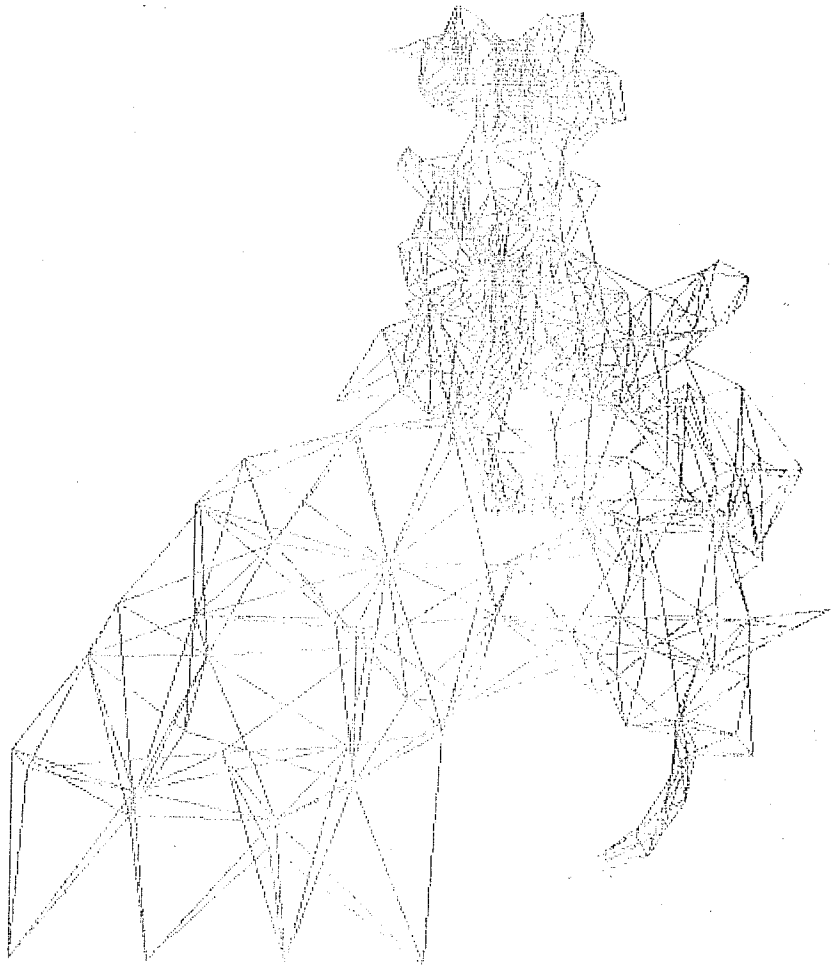
Fig. 7-33

Fig. 7-34



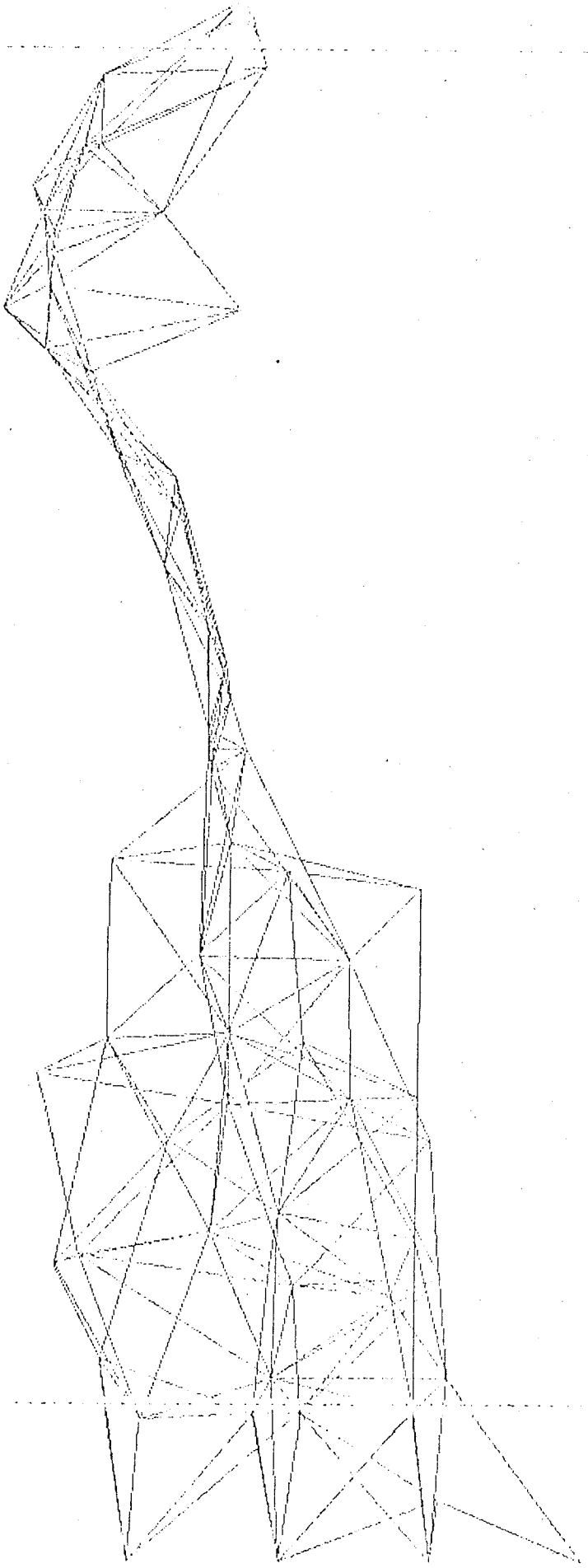
SCALE 0.50 / DATE & RUN #: [illegible] #150
OF CELLS 392 # OF LAMPS 400

Fig. 7-35



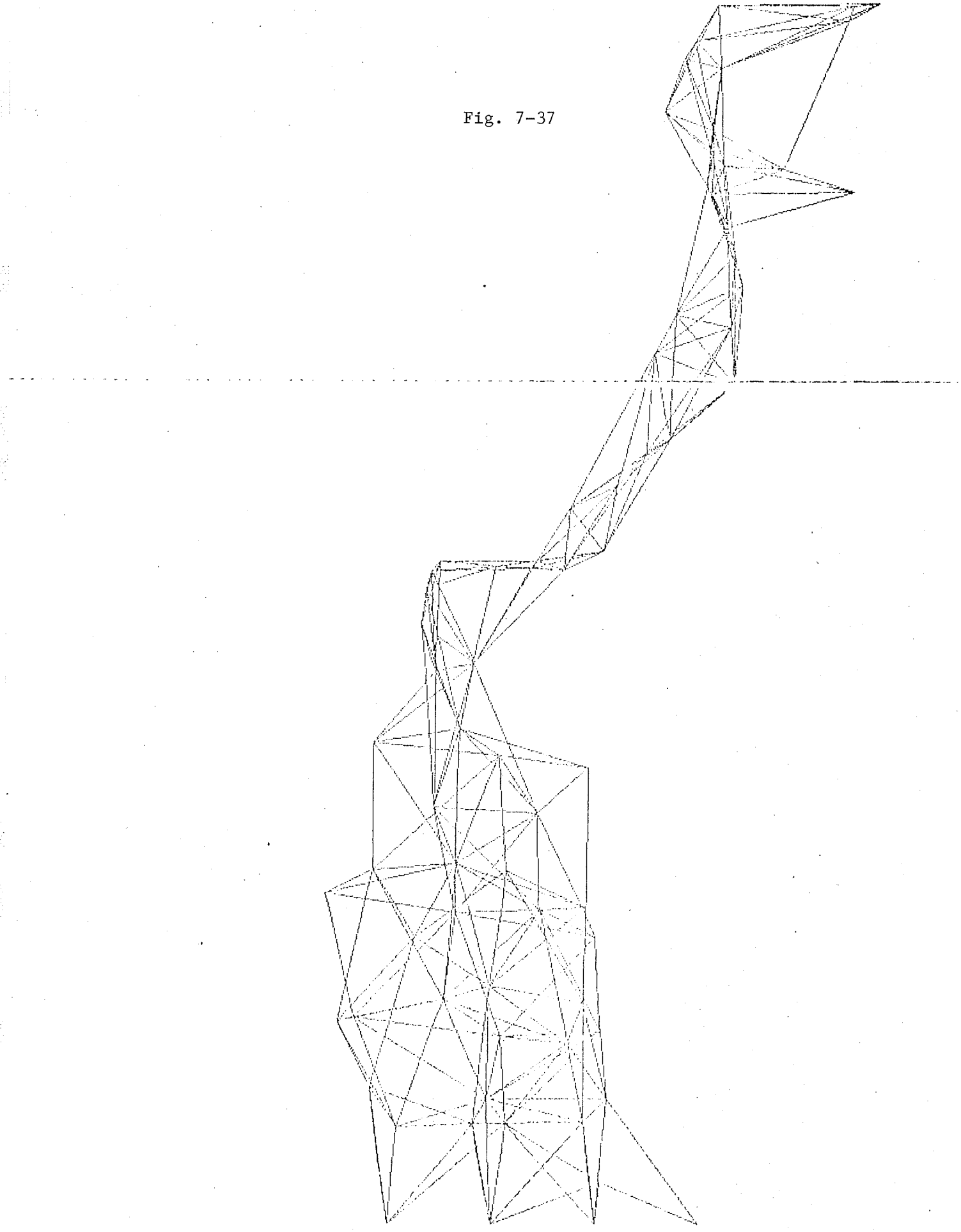
SCALE 0.60 / DATE & RUN #: [illegible] #150
OF CELLS 400 # OF LOOPS 400

Fig. 7-36



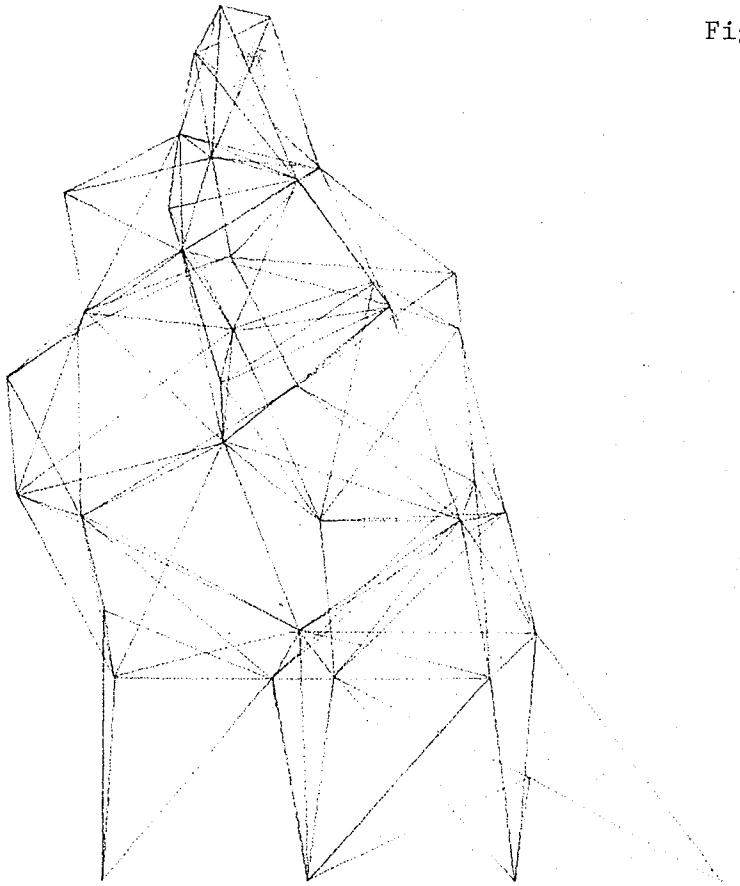
SCALE 0.80 / DATE & RUN #: DEC.8 #160
OF CELLS 75 # OF LOOPS 200

Fig. 7-37

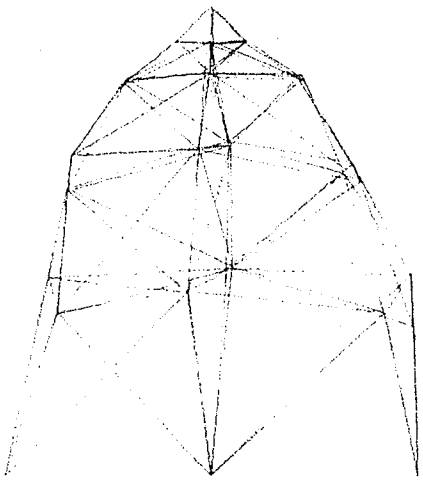


SCALE 0.70 / DATE & RUN #: DEC.8 #160
OF CELLS AU # OF LOOPS 250

Fig. 7-38

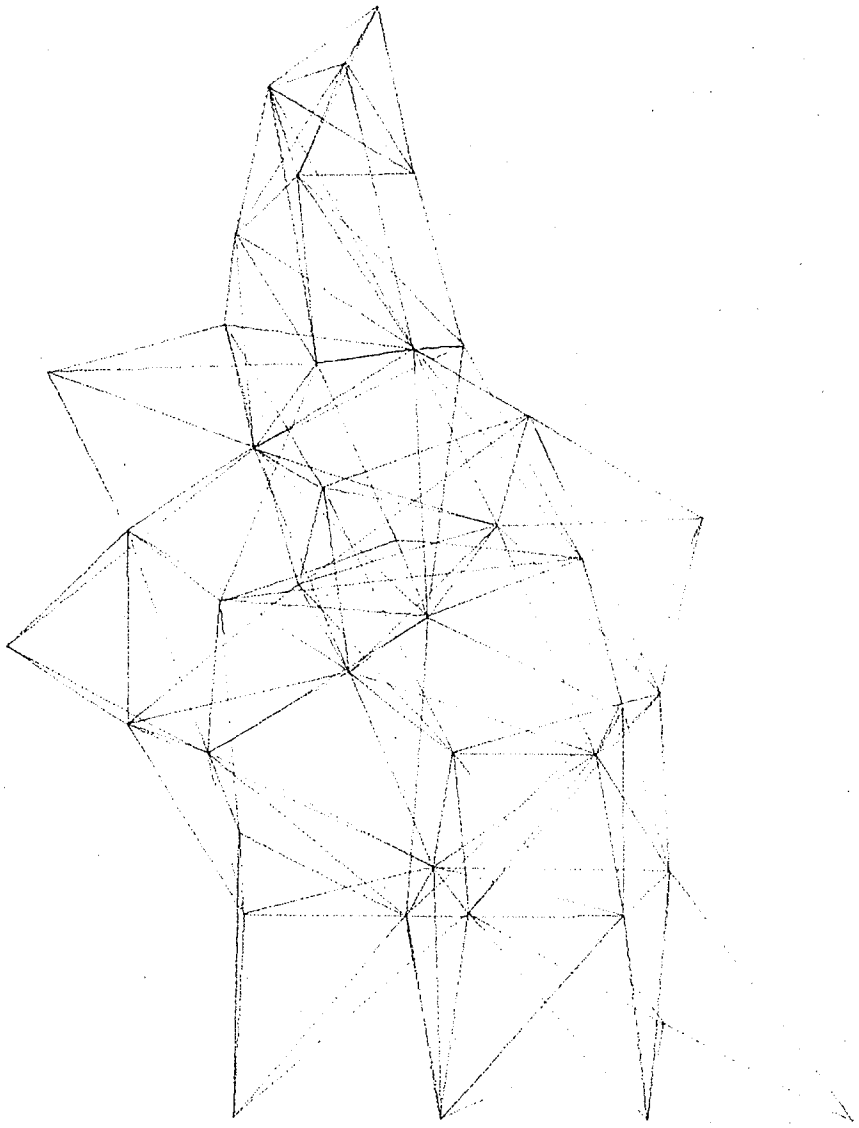


SCALE 0.90 / DATE & RUN #- DEC 9 #163
OF CELLS 52 # OF LOOPS 53



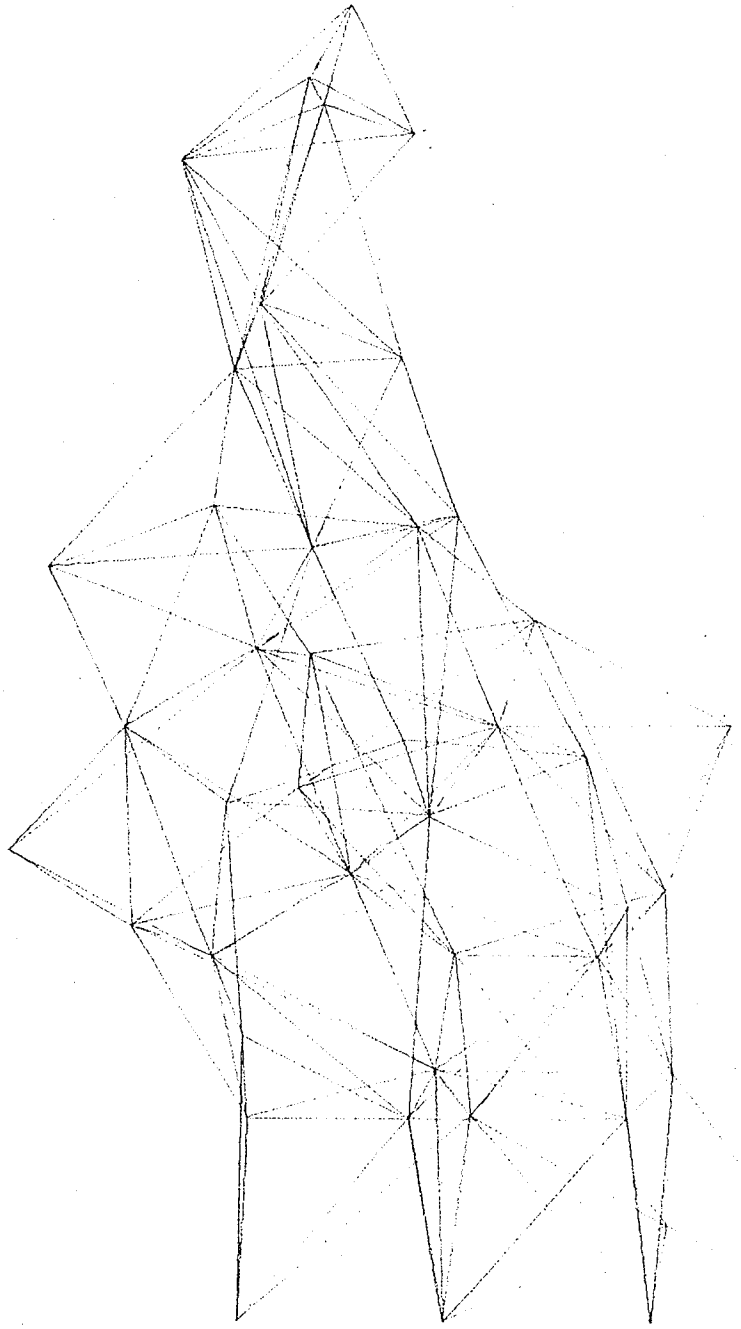
SCALE 0.90 / DATE & RUN #-, DEC 9 #163
OF CELLS 48 # OF LOOPS 0

Fig. 7-39



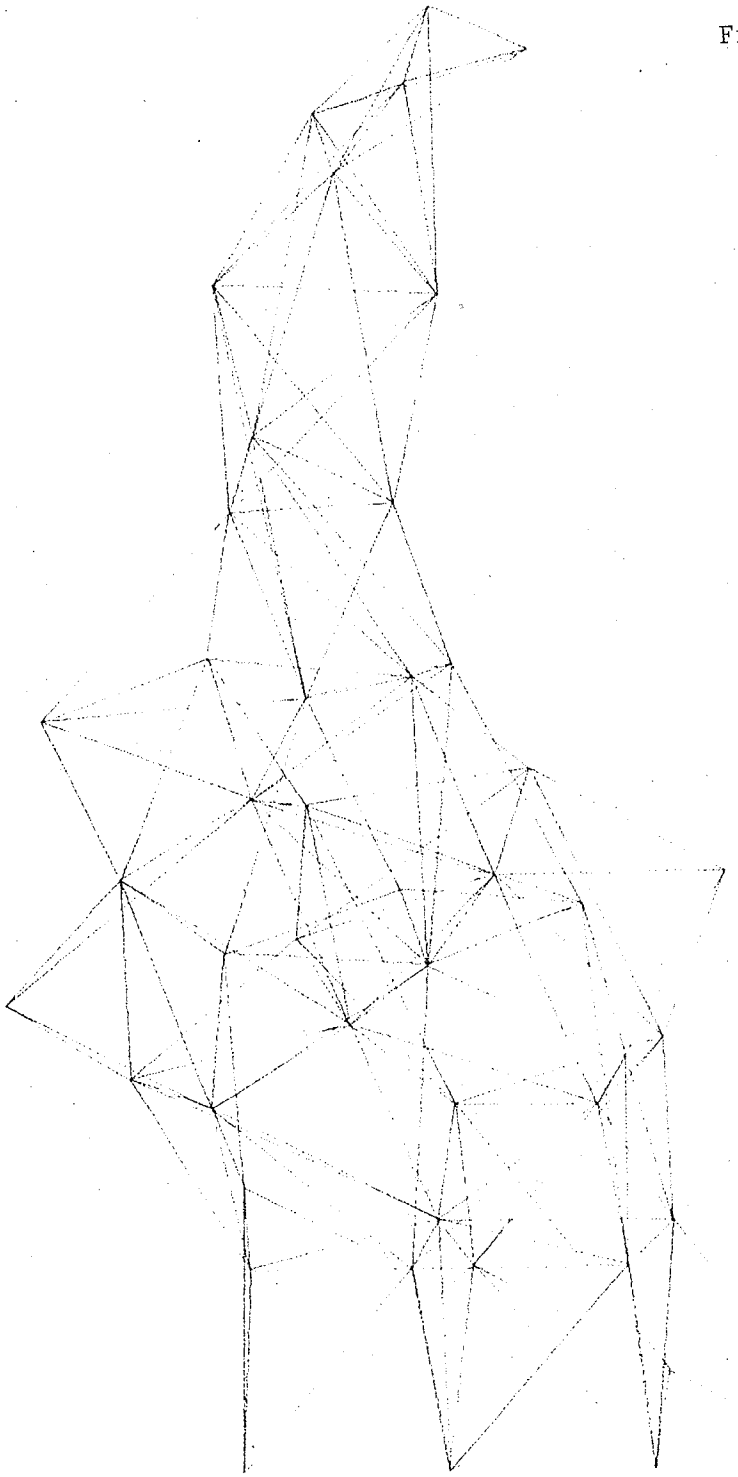
SCALE 0.90 / DATE & RUN #: DEC 9 #163
OF CELLS 54 # OF LOOPS 100

Fig. 7-40



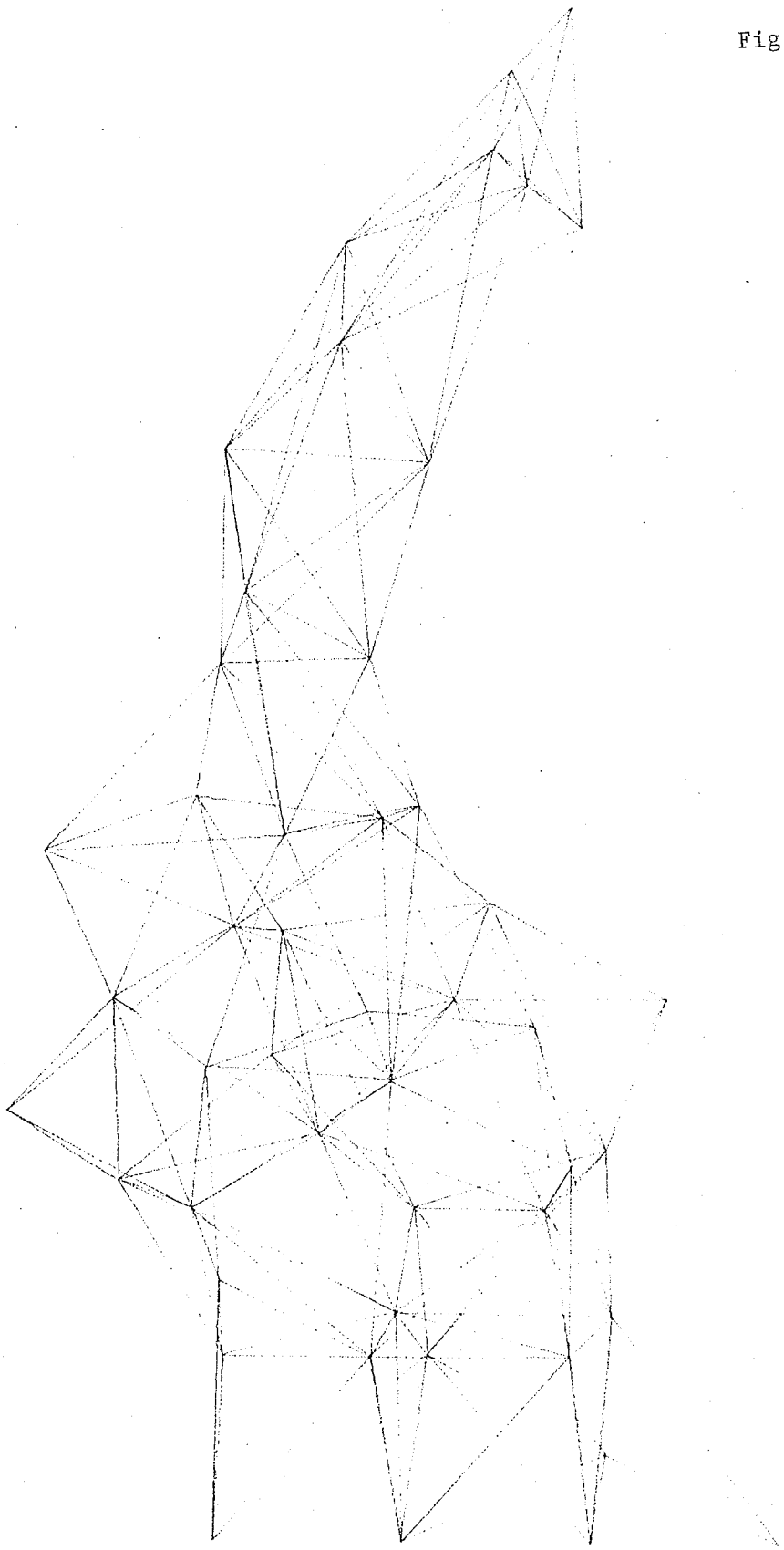
SCALE 0.90 / DATE & RUN .. OCT 9 #163
OF CELLS 65 # OF LOOPS 200

Fig. 7-41



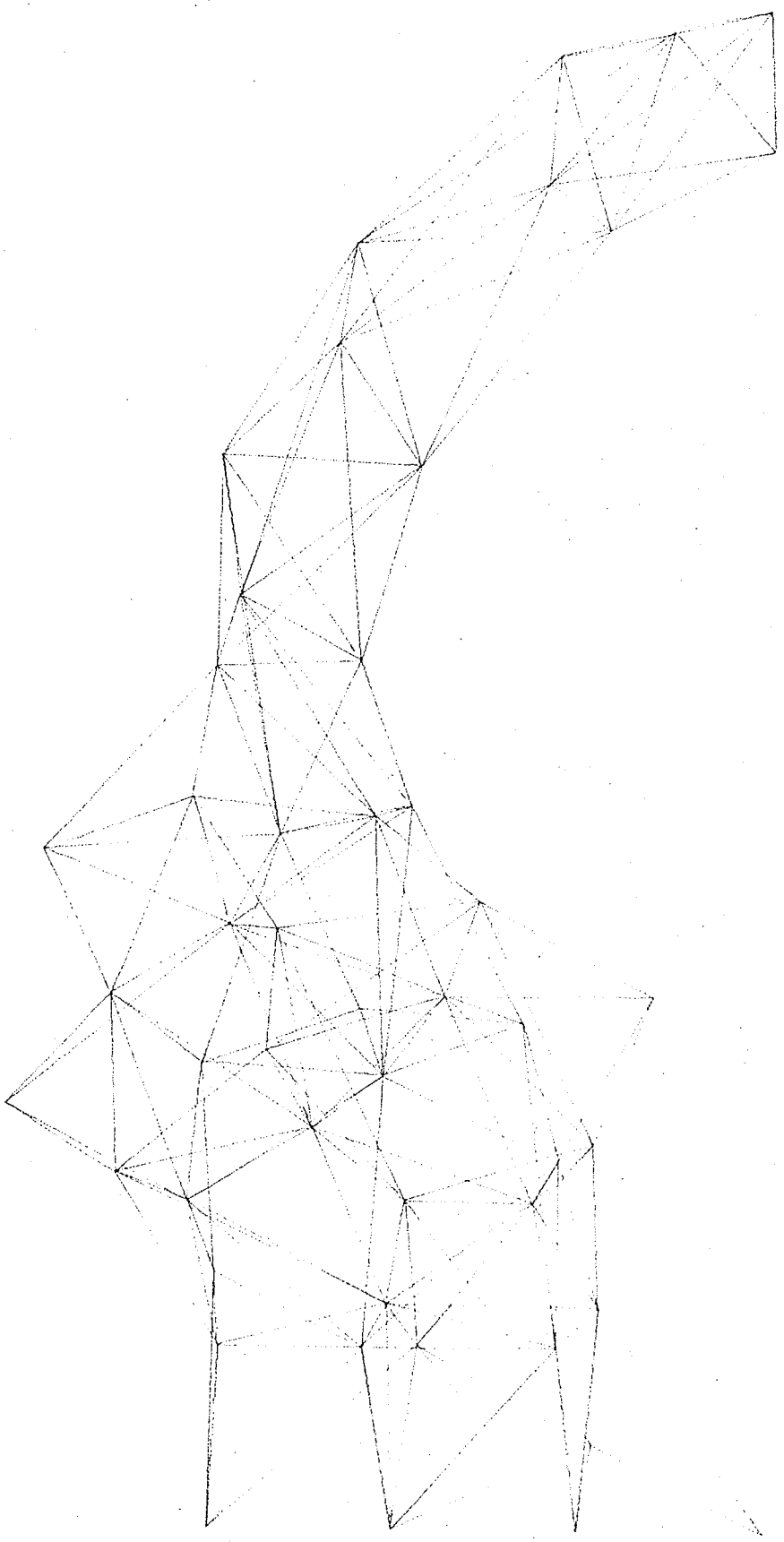
SCALE 0.90 / DATE & RUN #: DEC 9 #163
OF CELLS 58 # OF LOOPS 200

Fig. 7-42



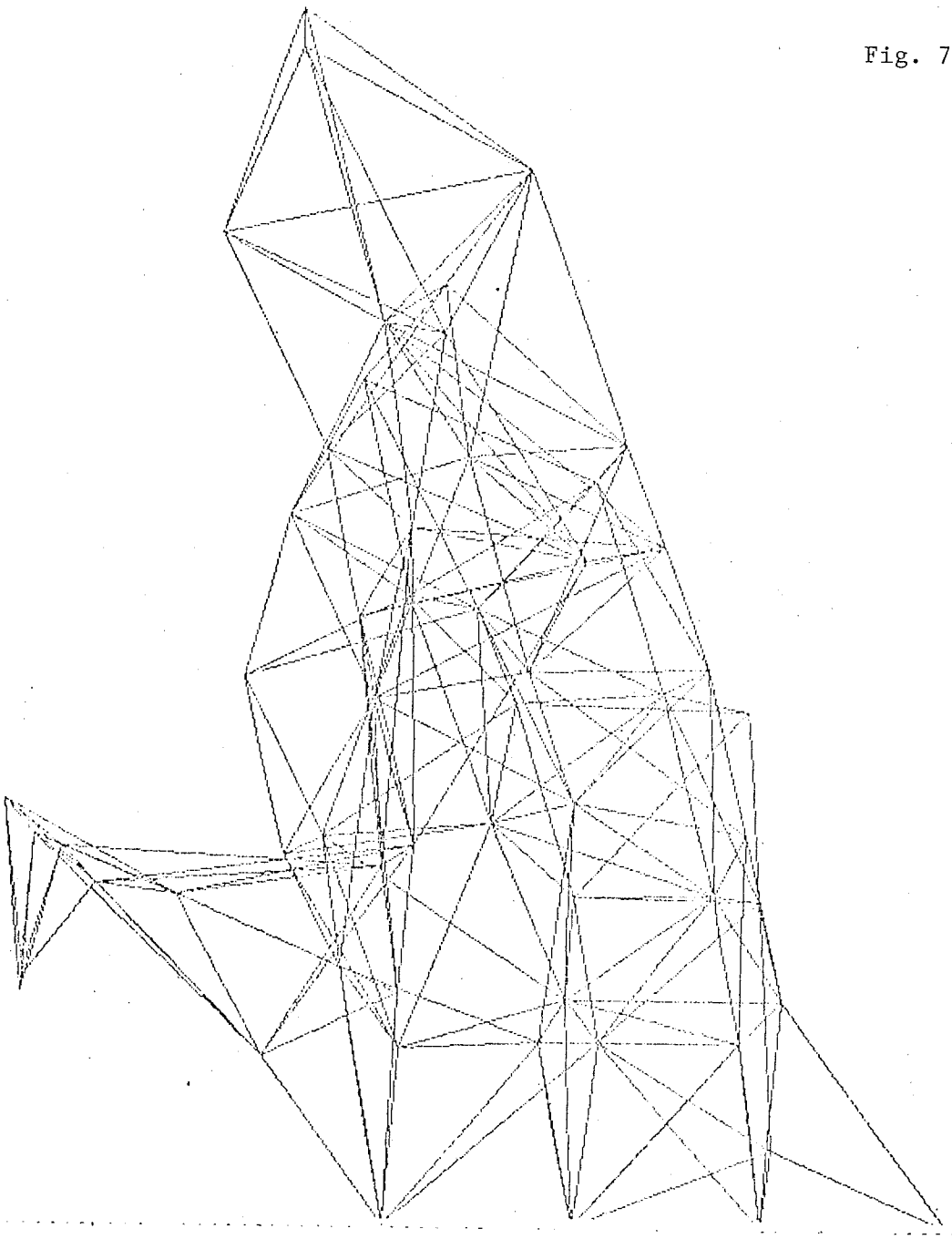
SCALE 0.90 / DATE & RUN # [illegible] #163
OF CELLS 60 # OF LOOPS 100

Fig. 7-43



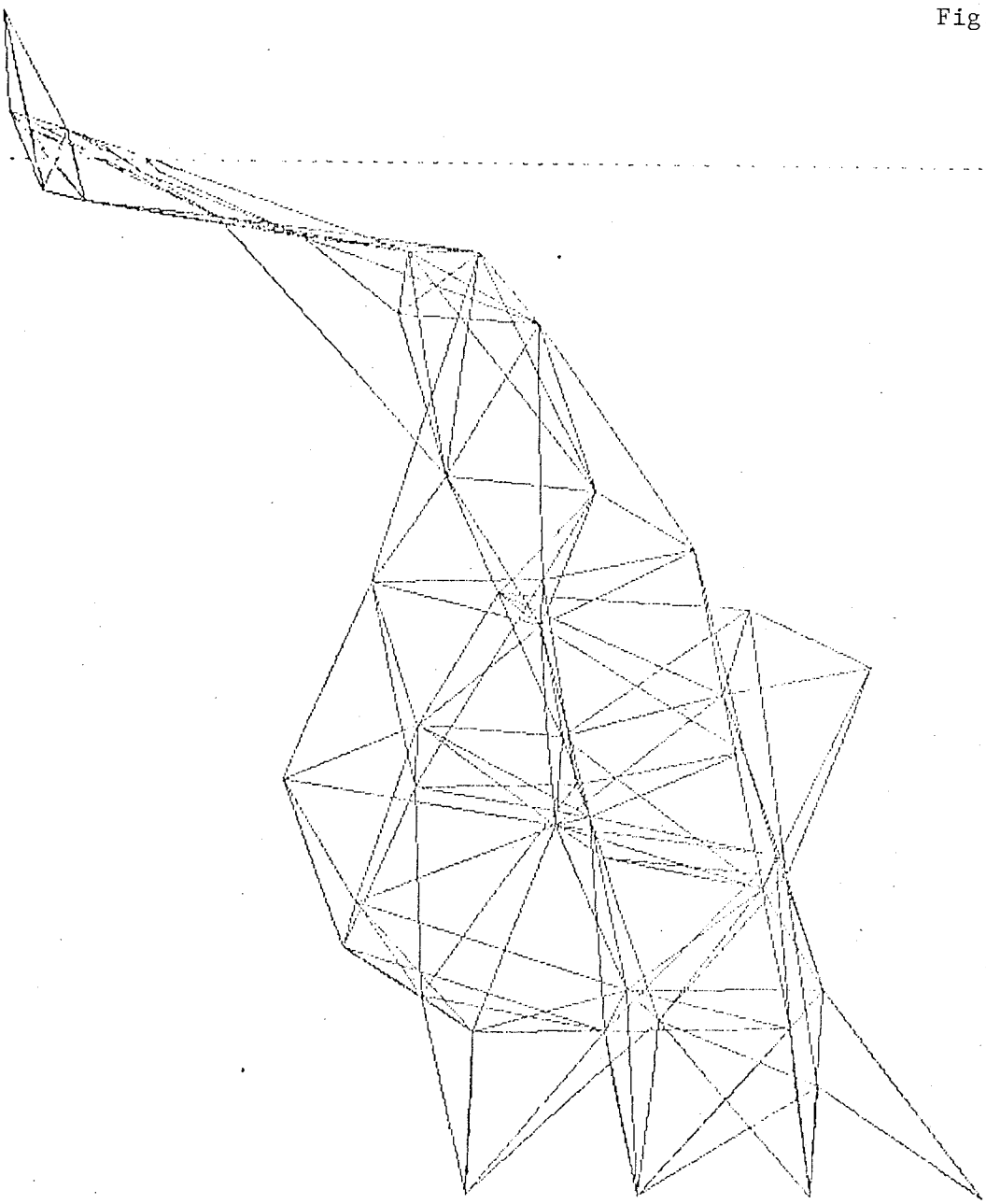
SCALE 0.30 / DATE & RUN #: DEC. 9 #163
OF CELLS 91 # OF LOOPS 100

Fig. 7-44



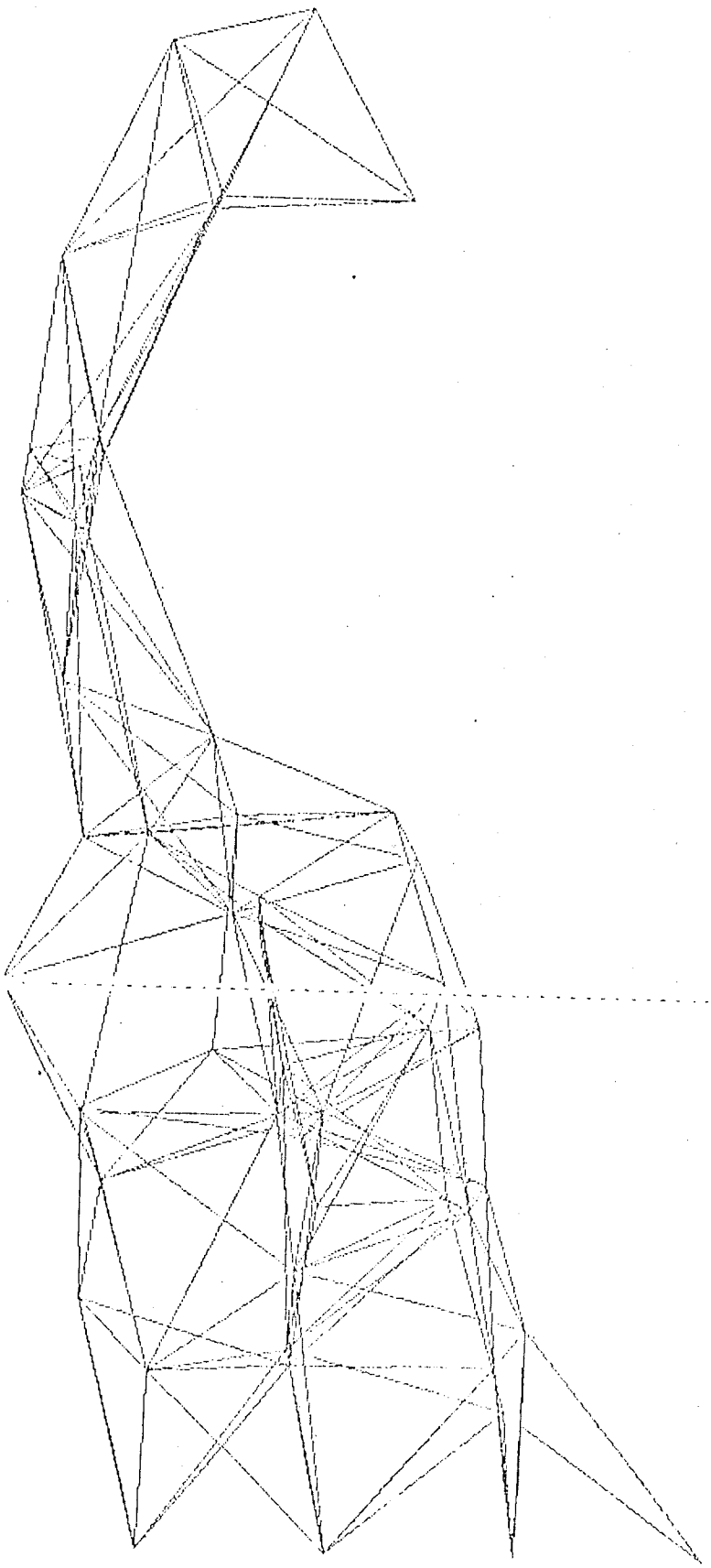
SCALE 0.90 / DATE & RUN #: DEC.10 #164
OF CELLS 69 # OF LOOPS 350

Fig. 7-45



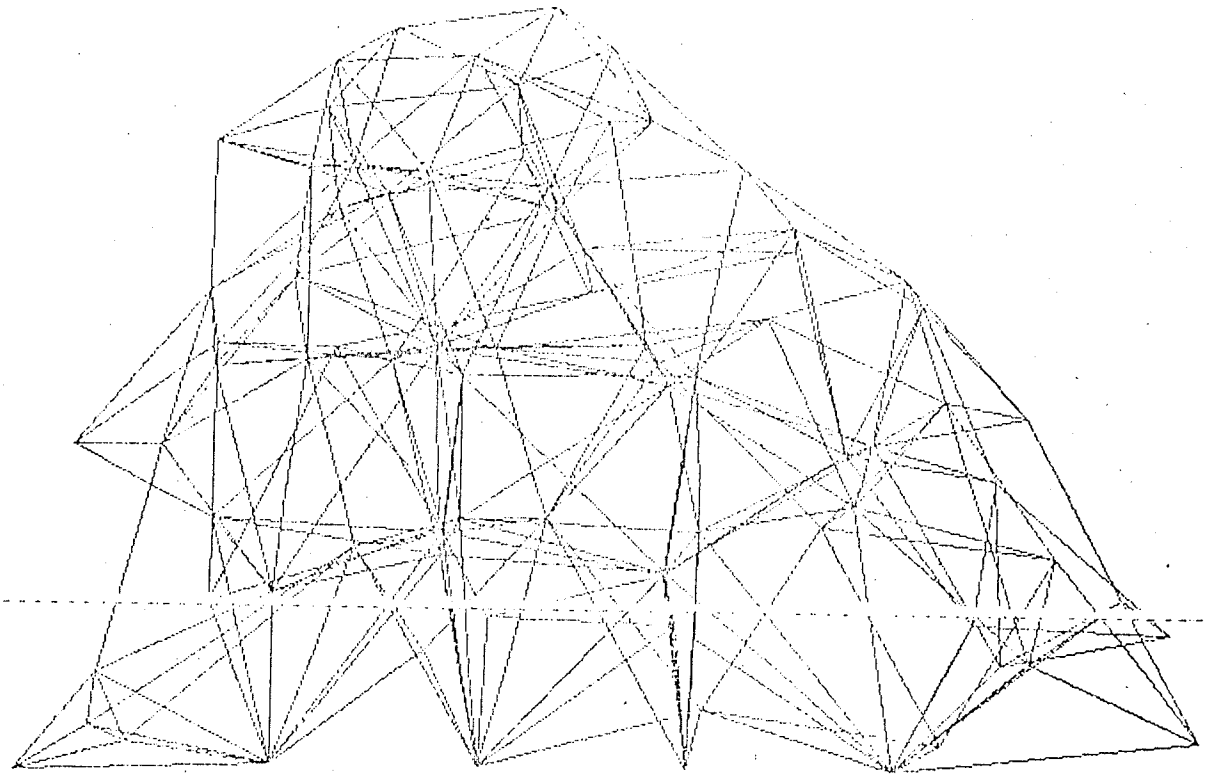
SCALE 0.60 / DATE & RUN #: DEC.10 #176
OF CELLS 60 # OF LOOPS 200

Fig. 7-46

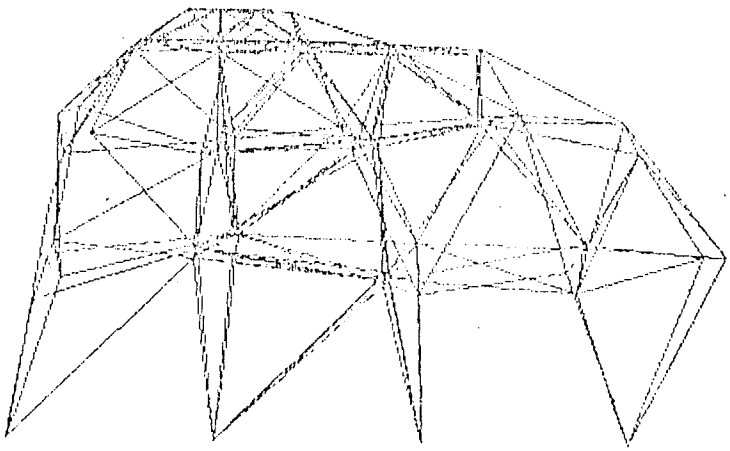


SCALE 0.90 / DATE & RUN #: DEC.10 #177
OF CELLS 63 # OF LOOPS 500

Fig. 7-47

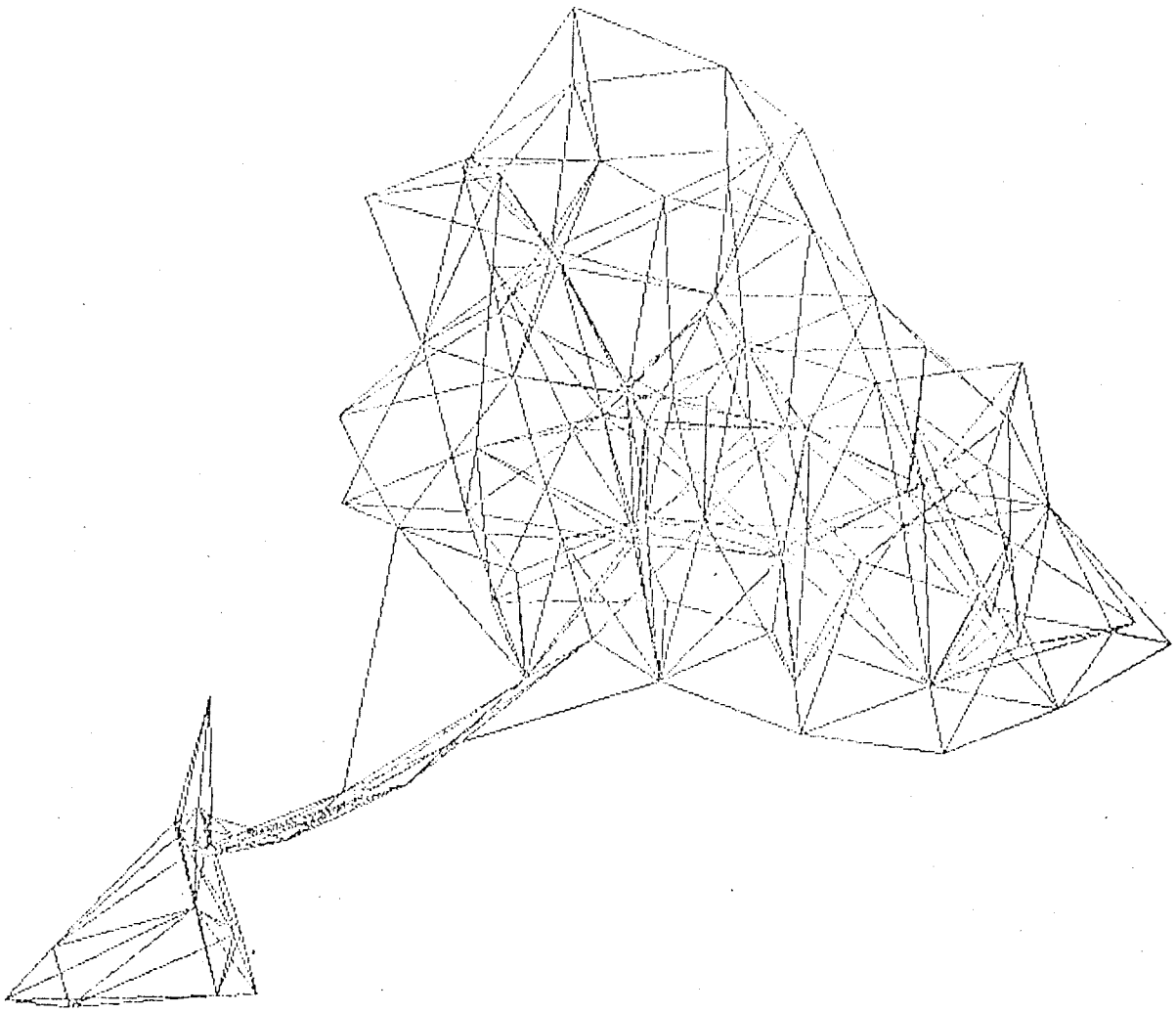


SCALE 0.90 / DATE & RUN #: DEC.10 #178
OF CELLS 92 # OF LOOPS 50



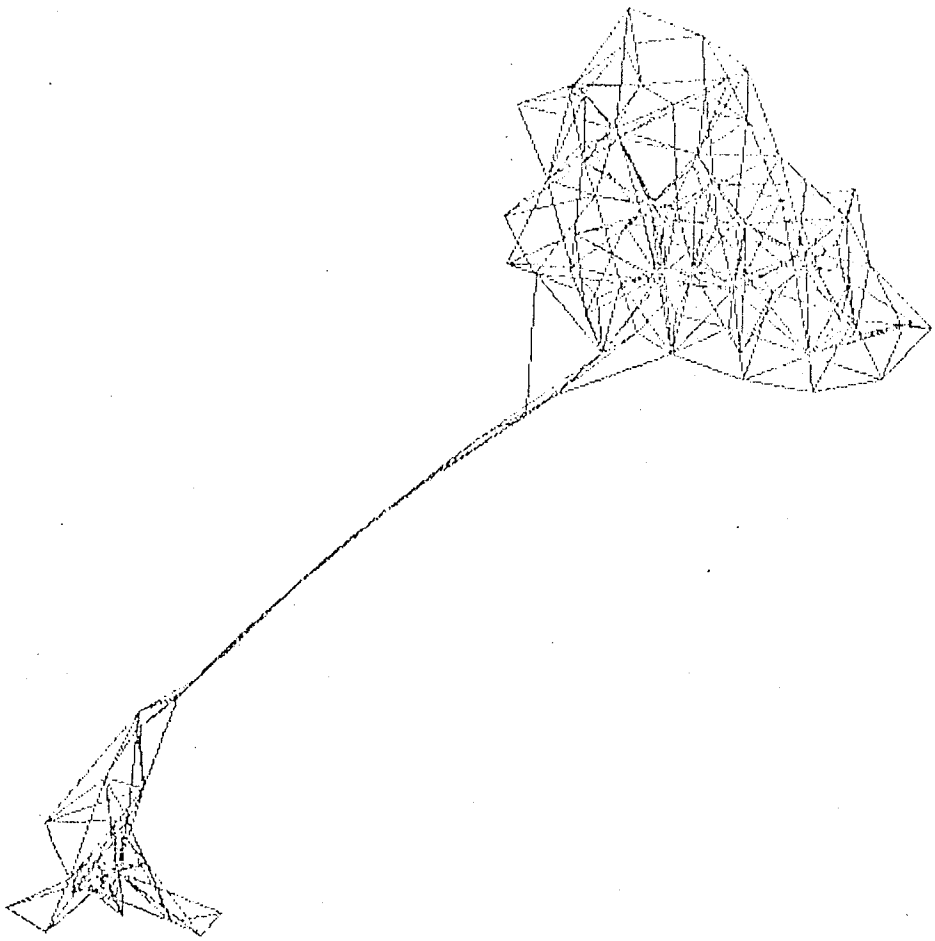
SCALE 0.90 / DATE & RUN #: DEC.10 #178
OF CELLS 86 # OF LOOPS 0

Fig. 7-48



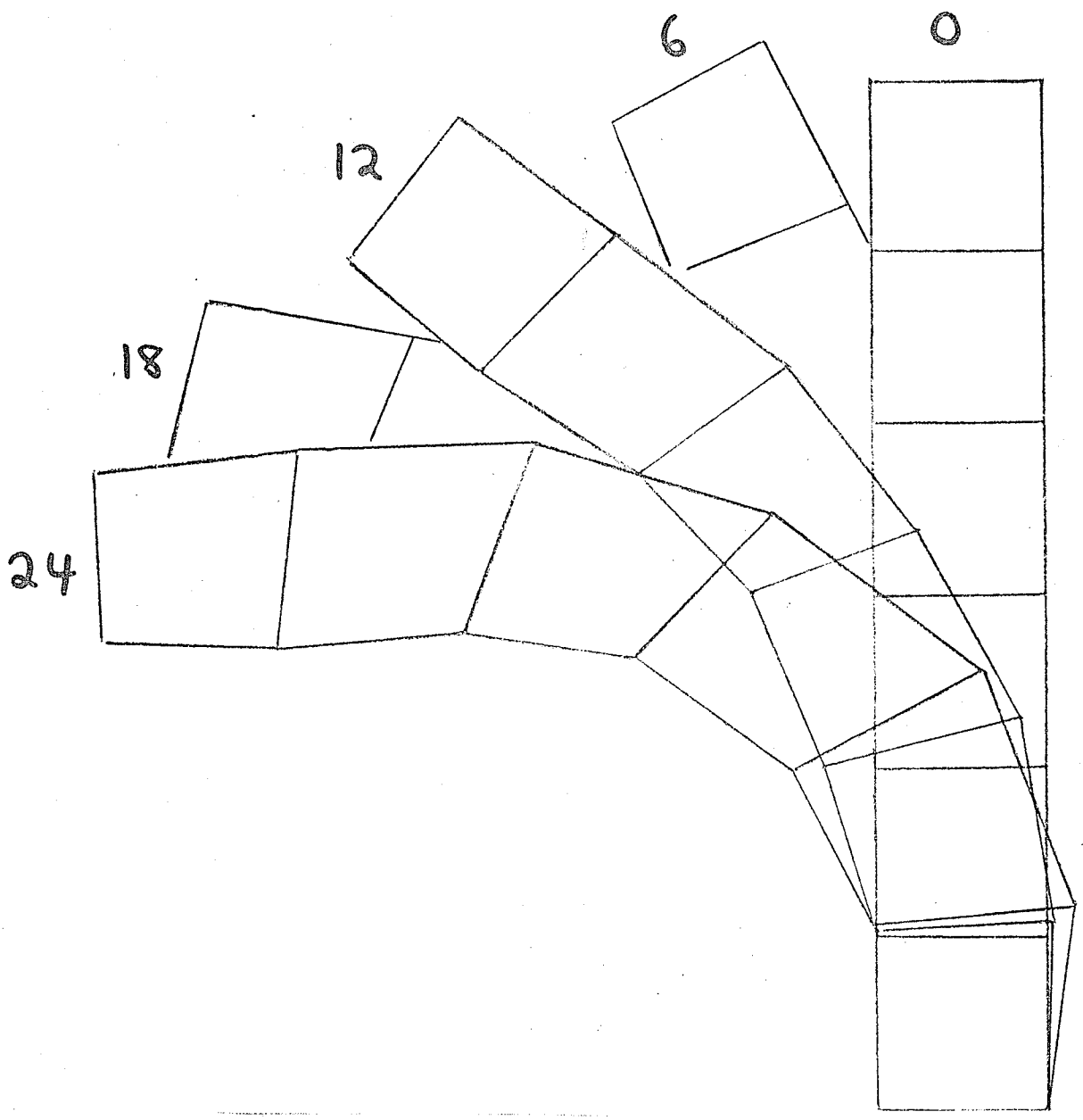
SCALE 0.60 / DATE & RUN #: DEC.10 #178
OF CELLS 110 # OF LOOPS 200

Fig. 7-49



SCALE 0.30 / DATE & RUN #: DEC.10 #178
OF CELLS 130 # OF LOOPS 400

Fig. 8-2



Reconciliation of an initially straight column of cells to a growth rate of about 2% for cells initially on the right and zero for cells on the left. Numbers are the numbers of Reconciliation steps (with 2% growth per step). Lines represent lines between centres of cells. For clarity, the lower section after 6 and 18 steps is not shown. (See also figures 7-2,7-3 and 7-4.)

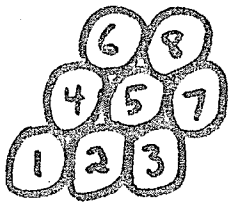


Fig. 8-1

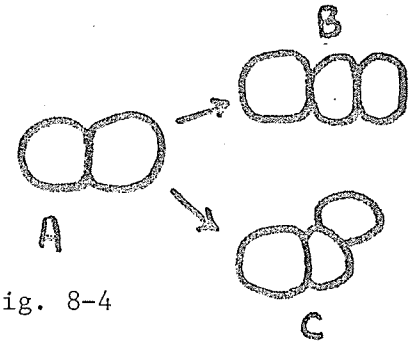


Fig. 8-4

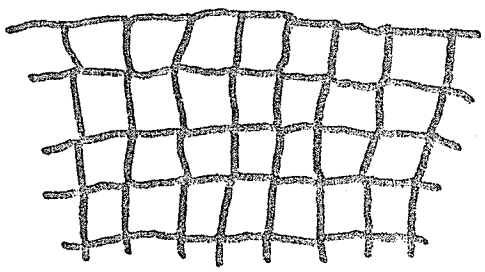
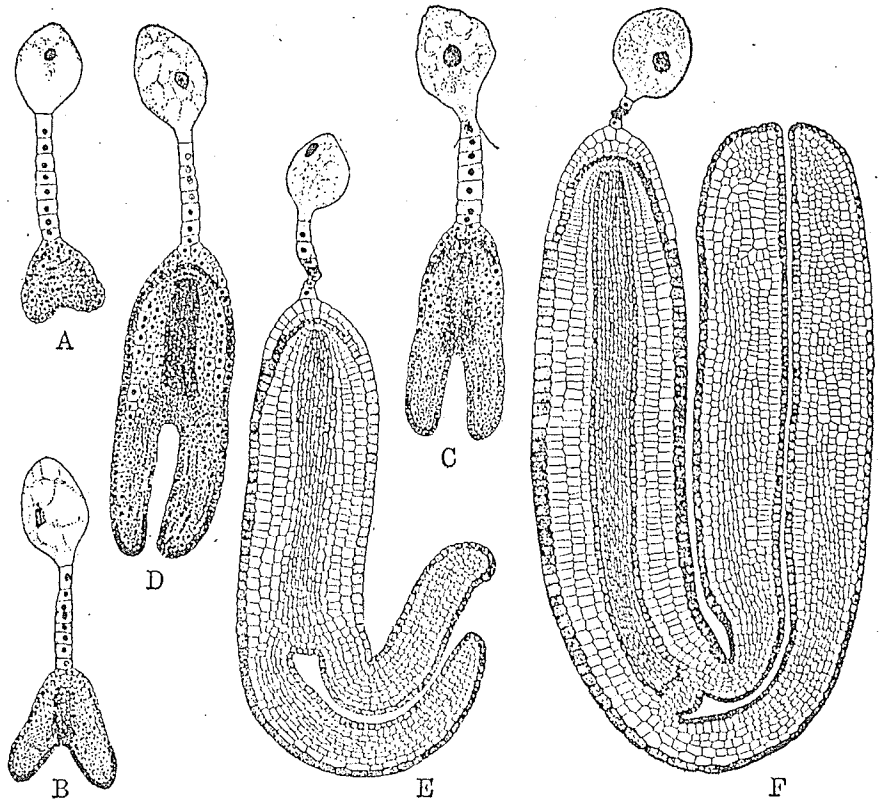
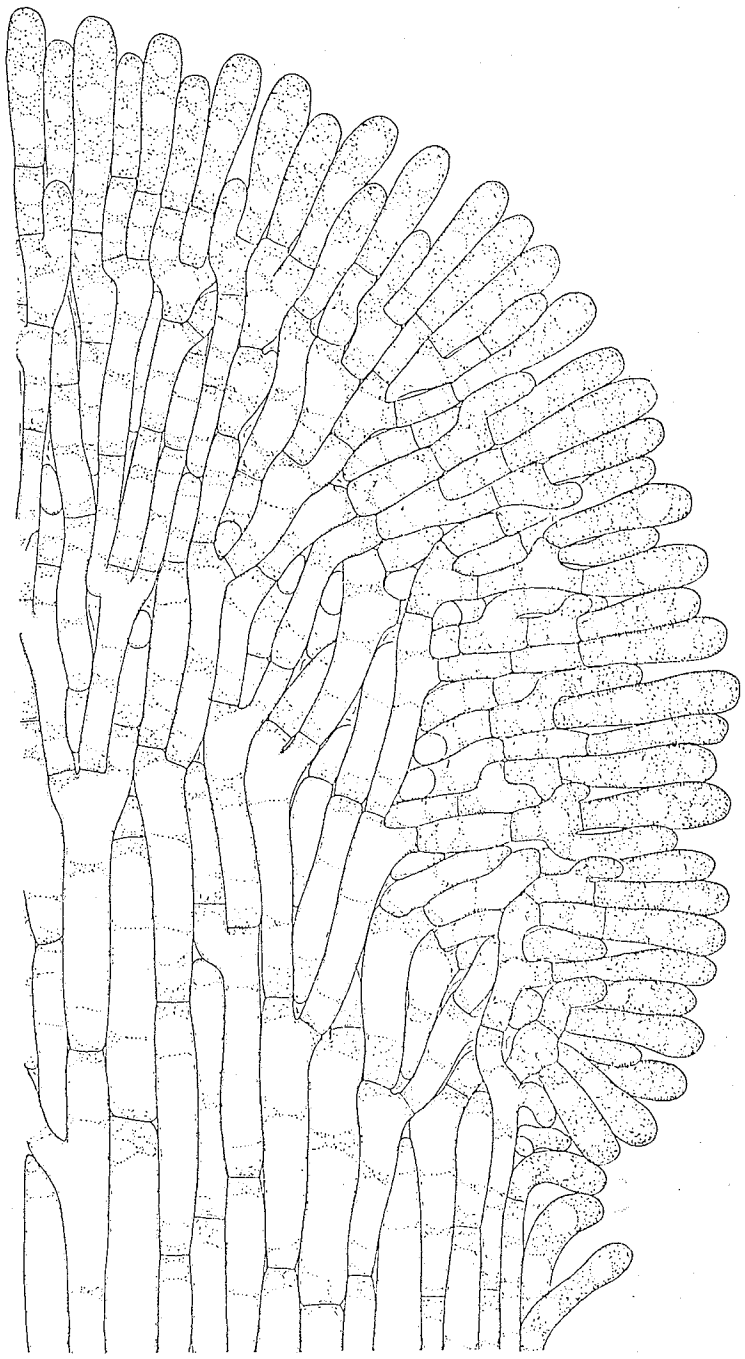


Fig. 8-5

Fig. 8-6



Older stages in development of embryo of *Capsella bursa-pastoris*. (*After Schaffner, 1906.*)



Longitudinal section through the head of a fruiting body of the basidiomycete *Typhula gyrans* at the moment of its cessation of upgrowth. [From E. J. H. Corner, *Clavaria and Allied Genera* (Clarendon Press, Oxford, 1950), Fig. 47.]

Fig. 8-7