(NASA-CR-161289) STANDARD TRANSISTOR ARRAY (STAR). VOLUME 1: PLACEMENT TECHNIQUE Final Report (Auburn Univ.) 150 p HC A07/MF A01 CSCL 09C

N81-32386

Unclas G3/33 37462

NASA CONTRACTOR REPORT

NASA CR-161289

STANDARD TRANSISTOR ARRAY (STAR)
Volume 1: Placement Technique

By G. W. Cox and B. D. Carroll
Electrical Engineering Department
Auburn University
Auburn, Alabama 36830

July 26, 1979

Final Report

Prepared for

NASA - George C. Marshall Space Flight Center
Marshall Space Flight Center, Alabama 35812

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

I. INTRODUCTION

The physical design problem has long been recognized as a critical aspect of the implementation of any system. In its most basic form, the solution of this problem consists of translating a conceptual specification of the system into a physical implementation.

A number of problems may occur in the satisfaction of this translation process. The conceptual system may have been designed without regard to physical limitations which must be overcome in order to actually construct the system. The conceptual specification may be presented at a high, functional, level which is not easily translated to the detailed level required for construction. In addition, due to the complexity of the design, it may not be possible to "solve" the physical design problem in an acceptable amount of time.

Perhaps in no other area do these problems occur more frequently than in the design of digital LSI (Large Scale Integration) circuitry. The digital logic designer has historically worked at a gate level, often with disregard for the physical considerations necessary for implementation of this design at the transistor level. Internal connections between the transistors composing a gate and connections between gates occur with such frequency that "good" physical solutions can easily resemble bowls of spaghetti. Size, weight, electrical noise, and heat dissipation requirements may be ignored at the conceptual level, yet become critical once circuit construction has begun.

To assist in overcoming these inherent difficulties, the digital logic physical design problem has been divided into three specific steps: partitioning, placement, and routing. The first of these steps, partitioning, involves the assignment of circuit elements to modules (or chips) subject to the size limitations of each module and the desired inter-module connectivity. The second step, placement, entails specification of the exact location for each element in a module subject to some optimizing criteria such as ease of interconnection. Once the position for each element is specified, the third step in which the element interconnection paths are located (i.e., the circuit is _routed_) , is performed.

All three steps of t..e design procedure are of equal importance to the realization of a physical circuit implementation, and algorithms have been developed for the satisfaction of each step. These existing techniques have been developed to meet the physical design requirements of particular LSI organizations and manufacturing strategies. Due to the wide variety of LSI development systems, certain techniques are more adaptable to one system than to another. Thus, while the general nature of many of the techniques is similar, specialization of physical design procedures to a single technology is the rule.

In the remainder of this report , a procedure for implementation of the placement step is presented. This procedure is oriented toward a new LSI technology developed at Marshall Space Flight Center. The technology is briefly described in the following section. Objectives and organization of the report are more explicitly stated at the end of this chapter.

The Standard Transistor Array (STAR)

Historically, a low-volume user requiring a special-purpose digital integrated circuit (IC) has been forced to weigh the advantages of integration (such as design secrecy, reliability gain and size reduction) against the high costs associated with the development of a custom IC. These high costs were imparted due to the nature of custom integrated circuit development technology which is geared toward large quantity production.

In the last several years, alternate methods of development of special-purpose integrated devices, suited for low-volume applications, have been achieved. The popularity of these devices, known as semicustom IC's, is reflected by the large number of manufacturers who specialize in them [1].

These manufacturers have attacked the cost problem in several ways. Almost uniformly, the construction of the semicustom circuits consists of forming masks for the interconnection of a standard (perhaps pre-fabricated) understructure of transistors. This organization can be contrasted with that of the custom IC process, in which all active devices and interconnects are specialized to the application and require separate fabrication steps.

A second factor contributing to cost reduction is the use of a standard cell approach. It was very quickly realized that, if the internal connections of common digital logic elements were pre-defined as cells and stored, customization could be achieved by selecting an appropriate set of cells, arranging them on the standard understructure and interconnecting them as specified by the designer. The costs associated with physical design can thus be reduced from those required for transistor-level design to the costs for a cell-level implementation. This cost reduction is due, in great part, to the reduced size (number of elements) of the physical design problem which must be solved.

A third area in which cost reduction techniques have been applied concerns the means by which the physical design problem is solved. Hand design, which might be suited for custom technologies, must be abandoned in favor of more cost-effective automated techniques. Even some computerized procedures must be eliminated due to requirements for excessive time, excessive storage, large computing facilities, or specialized I/O devices.

Based on the need for semicustom integrated circuits and on recognition of the factors discussed, Standard Transistor ARray (STAR) [2] processing technology has been developed at NASA's Marshall Space Flight Center facility. This system uses a matrix of transistors as the understructure and is supplied with a comprehensive library of standard digital logic cells which are implemented by means of a two-level metallization process.

While the STAR organization is adaptable to many diverse technologies, during development of the system a bulk-metal CMOS (complementary MOS) technology is being used. A sketch of the CMOS understructure is shown in Figure 1 with a typical cell shown in Figure 2. A sketch of a typical STAR placement is shown in Figure 3.

The STAR organization shown in these figures can be contrasted with the more common polycell layout organization shown in Figure 4. The most evident difference between these organizations is the lack of routing channels between cell rows in STAR. The paths for cell interconnection (global paths) in the STAR organization are provided internally to each cell (Figure 5). By performing vertical routing on the first level, four global paths are made

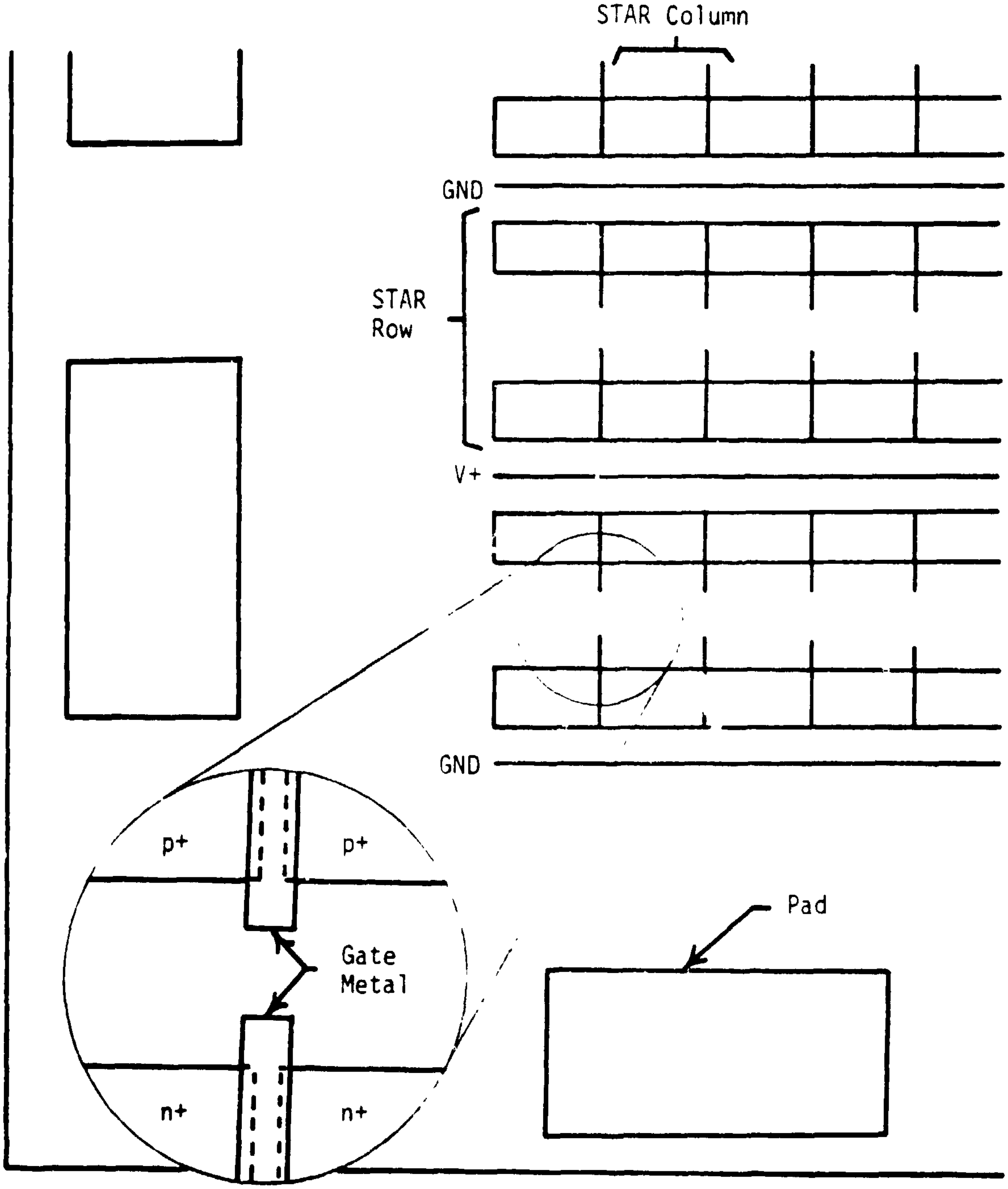
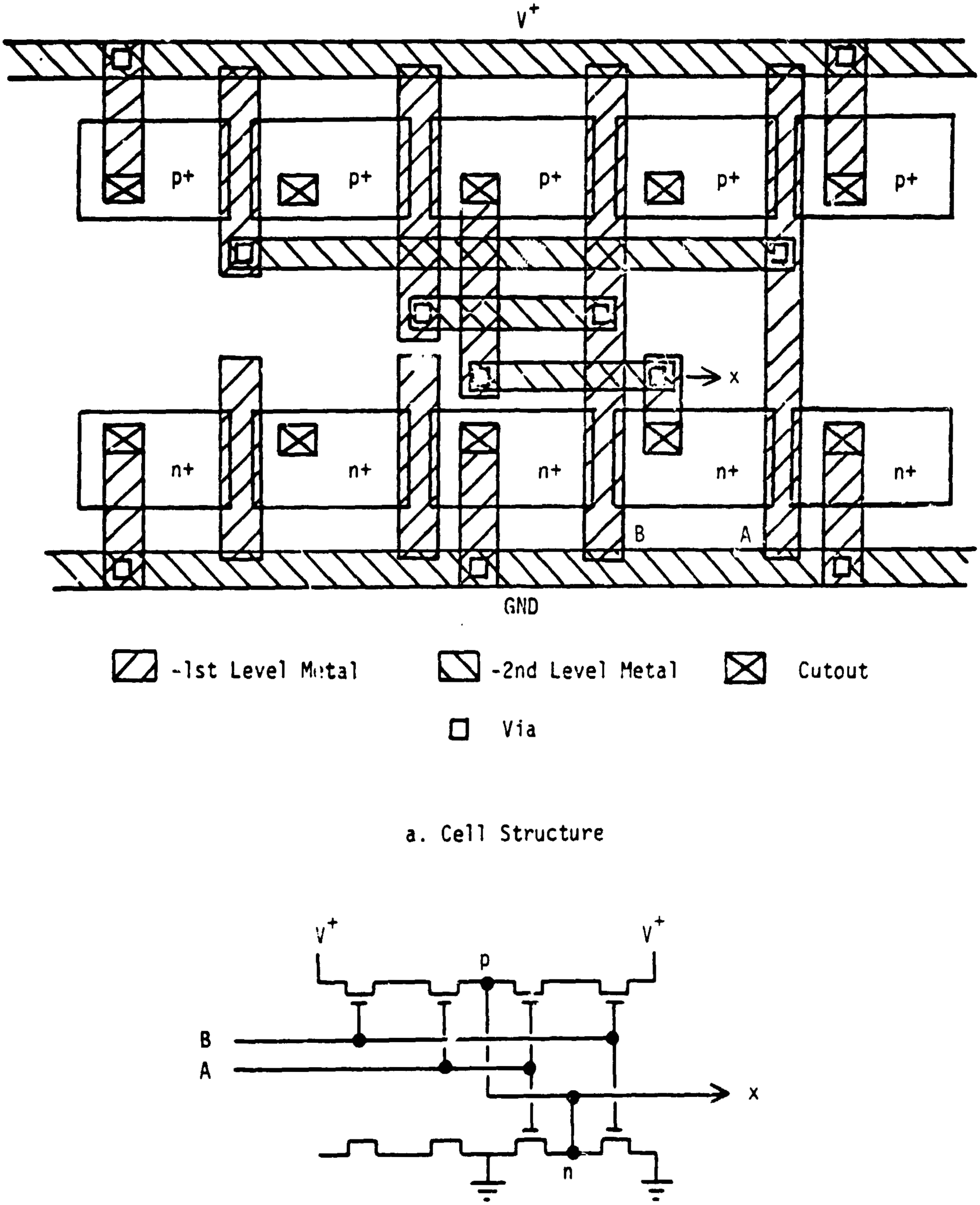


Figure 1. STAR CMOS Bulk-Metal Understructure



a. Cell Structure

b. Circuit Diagram

Figure 2. STAR Logic Cell (NOR)

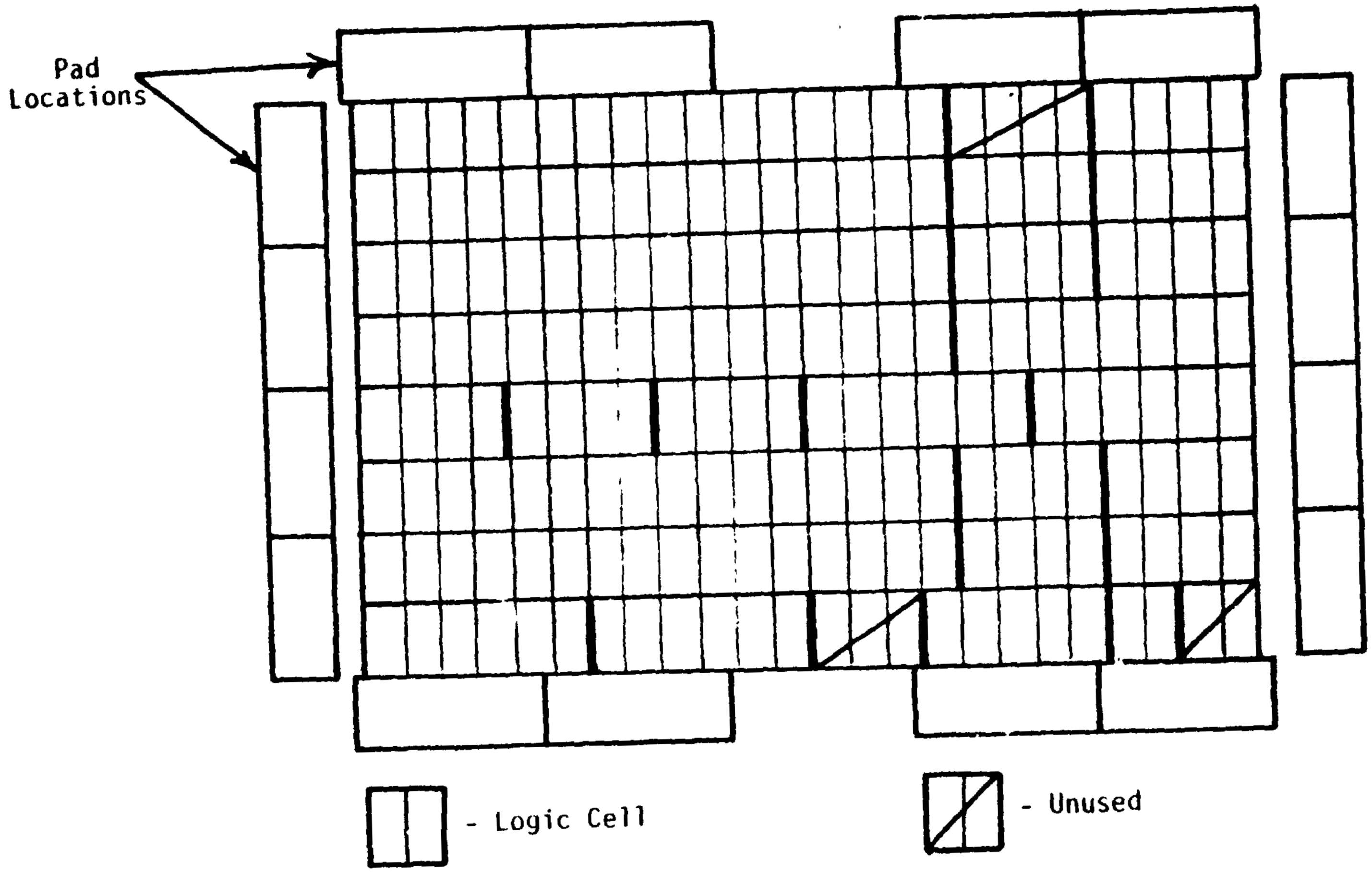


Figure 3. STAR Placement

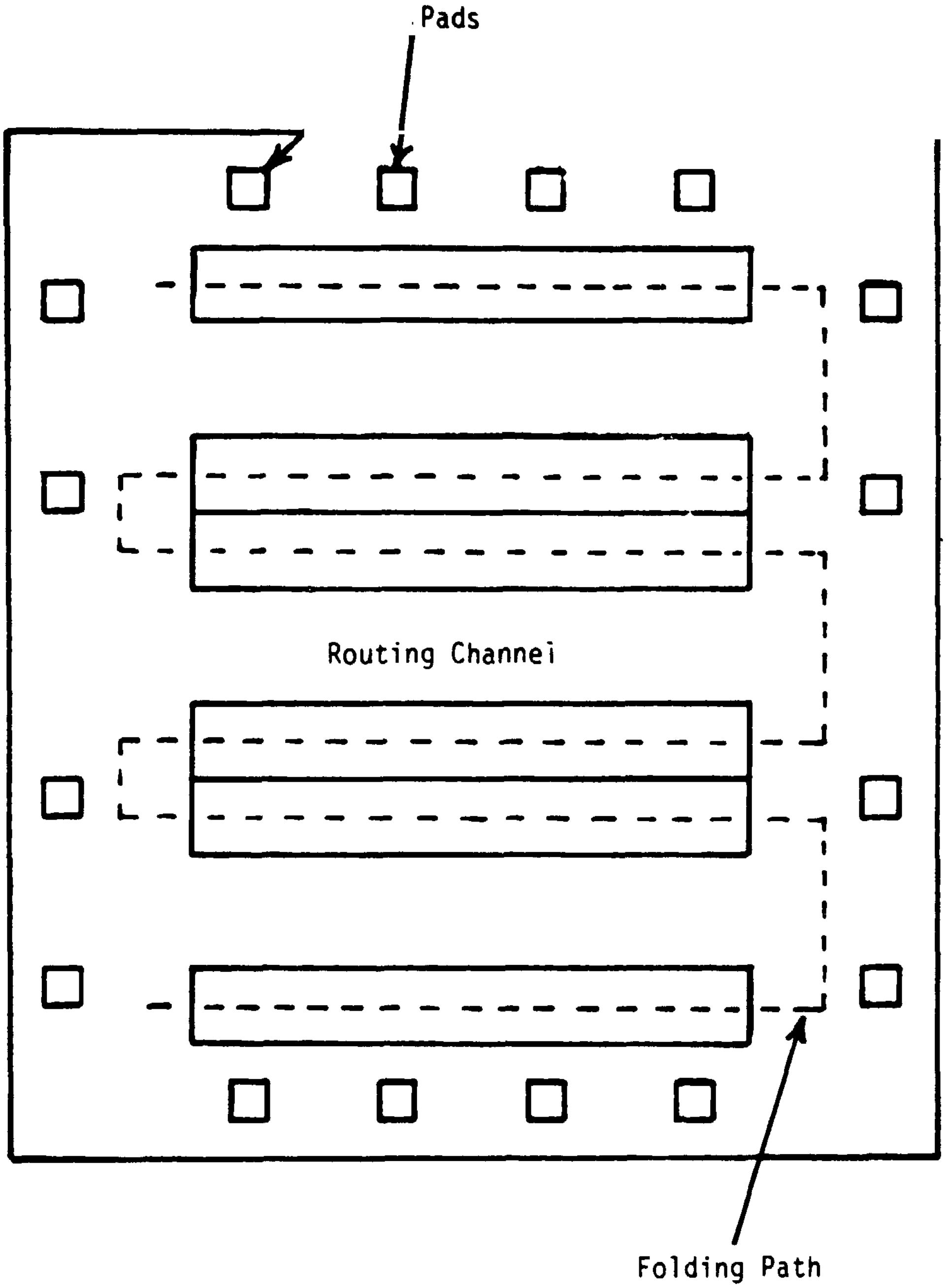
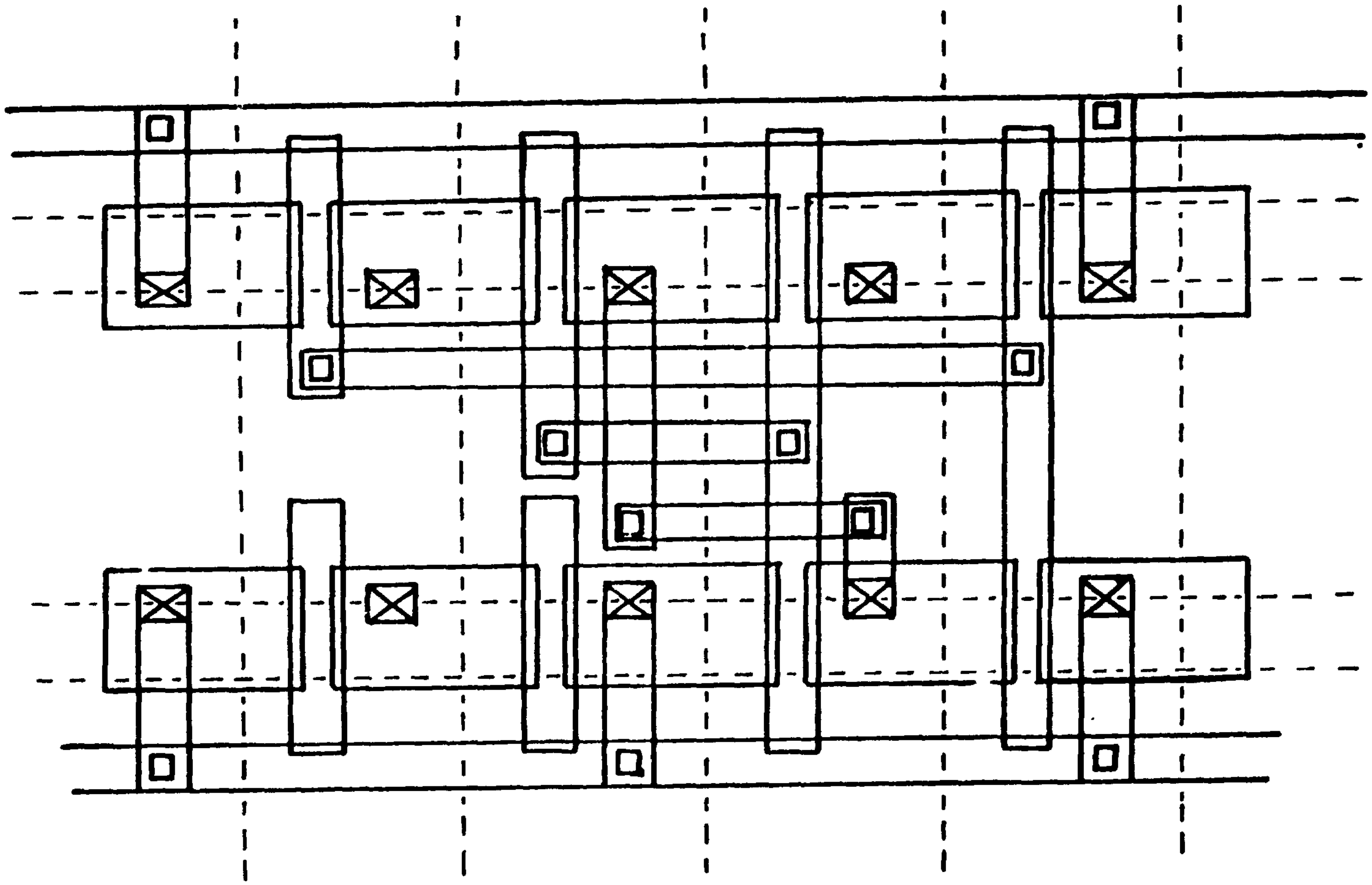


Figure 4. Typical Polycell LSI Organization



— — — — Global Path Centerline

Figure 5. STAR Cell Global Paths

available in the horizontal direction on each double-transistor row (STAR *row*). In addition, vertical passage through the cells is provided by a vertical global path in each transistor column (STAR *column*). A STAR cell which is n transistors in width supplies 4 horizontal and n vertical paths for non-terminating interconnections. There are ROWS x COLS of these horizontal and vertical path segments in a STAR placement, where ROWS is the number of STAR rows and COLS the number of columns. It should be noted that metal paths between levels (*vias*, as illustrated in Figure 6) are provided for the connection of horizontal and vertical segments.

A second innovation present in the STAR organization is indicated in Figure 7. External connections to a cell may enter from any side and can be joined to a cell bus at a number of points. This allows a greater degree of flexibility than the single cell entrance side and connection point provided in most technologies.

The STAR processing methodology has been proven by the fabrication of a number of test circuits. Due to the lack of cell placement facilities, however, cell arrangement on the chips has been performed by manual techniques. In the following section, the objectives of this dissertation (which include description of automated STAR cell placement methods) will be stated.

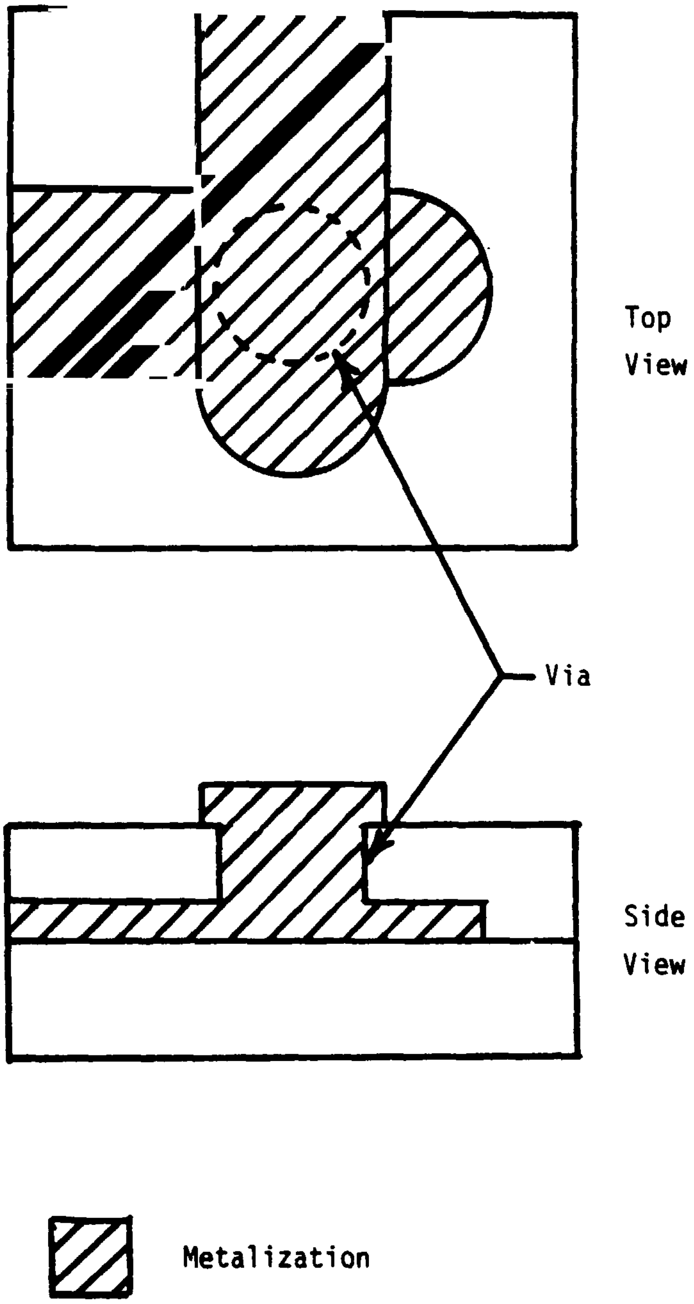
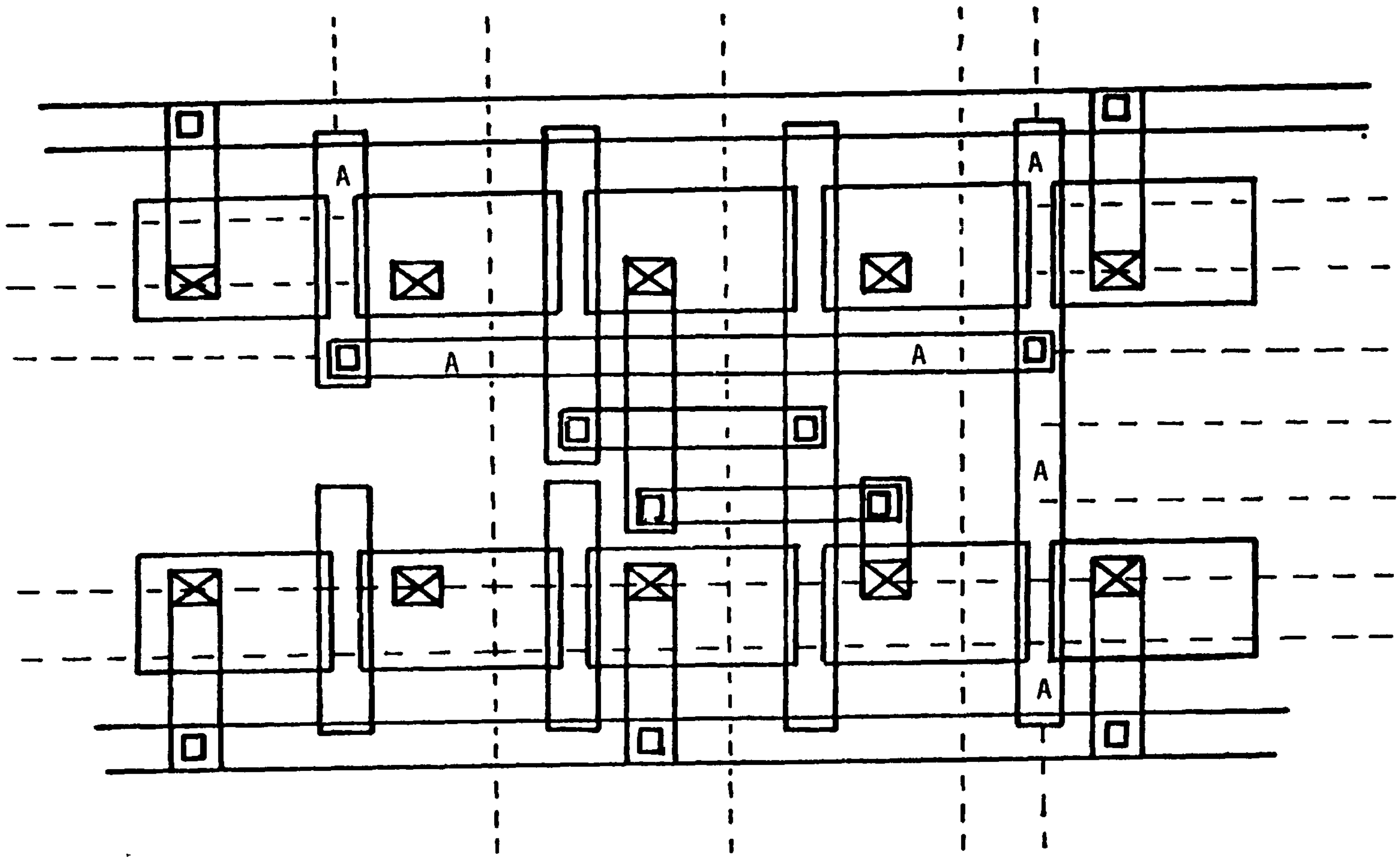


Figure 6. Double-level Metalization and Vias



— — — Possible Path of Connection to Pin 'A'

Figure 7. STAR Cell Pin Range

Objectives and Organization

The overall concern of the work on which this report is based is the development of simple automated STAR cell placement procedures which will enhance the ease of placement routing. The primary objective of the report , then, is the description of the placement techniques developed.

A second objective of the report is to demonstrate that, by modification of the algorithms, a placement procedure adapted to use with traditional LSI organizations can be utilized with non-polycell technologies such as STAR. The final objective is to illustrate STAR placement optimality measurement techniques developed.

In Chapter II, background information regarding the placement problem and existing methods for its solution are given. Chapter III describes the placement techniques developed for use with STAR. Chapter IV contains a discussion of several of the placement optimality measurement techniques developed. Chapter V details the integration of the various techniques into the final placement system. Placement system performance results are shown in Chapter VI. The final chapter presents a summary of results and recomendations for future work.

Equational developments are shown in an Appendix to this report.

II. BACKGROUND

The LSI cell placement problem is discussed in general in this chapter. The first section concerns problem definition, characteristics, and modelling. Previously-developed methods for problem solution are described at the end of the chapter.

The Placement Problem

As stated in the preceding chapter, solution of the placement problem involves identification of optimum locations for circuit elements within a module. A more exact definition of the problem, as applied to LSI technology, is:

> The LSI placement problem consists of identification of the optimum arrangement of elements on the chip with respect to criteria defined on element interrelations.

This statement of the problem, providing more flexibility than many prior definitions (such as that given by Hanan and Kurtzberg [3]) will now be discussed.

Discussion of Problem Statement

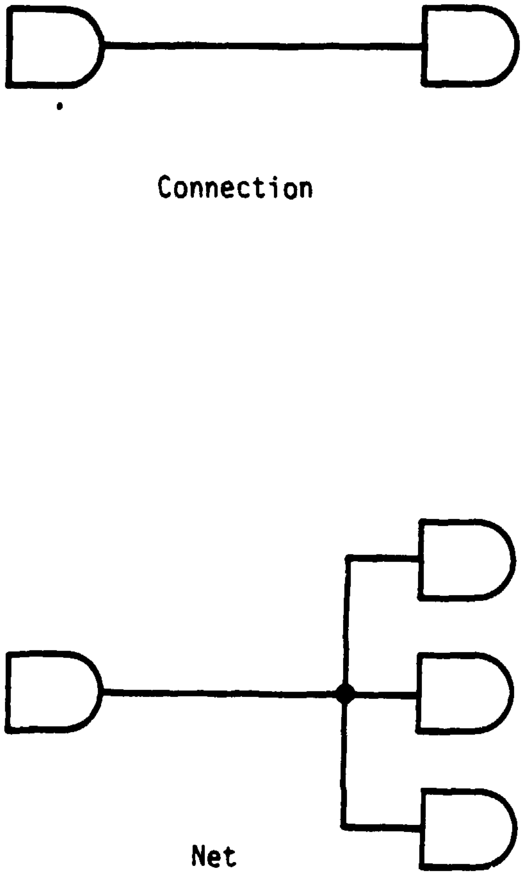
A typical partitioned digital LSI circuit will consist of logic devices (AND gates, OR gates, flip-flops, etc.) and of pads (metal areas used for interconnection to off-chip devices). Each logic device will internally consist of the active and passive components required to perform the desired logic function in a given technology (TTL, PMOS, CMOS, etc.).

It is thus possible to perform placement of the circuit at either of two levels: *component placement* in which the placement routine should assign optimum locations for each resistor, transistor, etc., and *gate placement* (or cell placement) in which the placement routine should form an optimum placement of the circuit components at a gate-description level.

The "elements" referred to in the problem statement, then, may be circuit constituents of various levels of internal complexity. While the level at which a placement routine must work is typically consistent for a single problem, a completely general placement routine should be able to perform efficiently at either level.

A second item requiring consideration in the statement of the placement problem is the meaning of "element interrelations". These relations are typically well-defined for a given problem and are used as a basis for the calculation of the optimality of a placement.

The most immediate element interrelation is that specified by electrical connections between elements. In the simplest case, shown at the top of Figure 8, each



Connection

Net

Figure 8 . Element Interconnections

connection joins only two elements. For this case, the element interrelation is binary in nature and need only specify whether a connection exists between each pair of elements. Almost universally, however, points on several elements are electrically common and a net is used for element interconnection. An example of this structure is also shown in Figure 8. The resultant element interrelation can be modelled as a binary relation (as will be discussed later in this chapter) but is most accurately represented as a relation between elements and nets. For notational convenience in the following sections, the terms "net" and "interconnection" are used to indicate an electrical connection between two or more elements. The term "connection" is used to describe a net between only two elements.

A second possible interrelation occurs due to the variety of power-consumption characteristics of the circuit elements. Since the power consumed by an element is related to the heat dissipated by the element, it is possible to form "hot spots" on the chip by placing high-power elements in proximity to each other. If this is of concern in the chosen LSI technology, an interrelation may be specified which represents the power requirements of the elements. A properly-chosen optimality criterion can then be used to discourage placements in which hot spots occur.

While other interrelations are conceivable, the most commonly used is that based on element interconnections. This is due both to the fact that interconnection data is easily retrieved from a circuit description and to the assumption that this relation can be used to form optimality measures for the most common criteria over all LSI technologies. Further discussion of element interrelations and optimality criteria are deferred to a later section.

The most disturbing (and, in practice, the most difficult to implement) feature of the placement problem statement is the specification of an optimum solution. As will be discussed in a later section, the placement problem cannot be solved in general in a realistic amount of time (at least by known methods). The only known methods for finding exact solutions of the placement problem (i.e., location of the optimum) involve investigation of the complete problem solution space. Since even a small, 13-element placement problem has a solution space containing over 6 billion (13!) placements, the derivation of an optimum solution for any realistic problem requires an unreasonable amount of time on existing computing systems.

The most common placement problem solution methods, then, strive to identify either an optimal (optimum within a restricted region of the solution space) or near-optimum (predictable as lying within a certain percent of the optimum) solution. Placement techniques which can realize these objectives have heen developed and have been shown to perform within computationally feasible time limits. Several of these are described in a later section.

A final clarification of the placement problem statement regards possibilities for element arrangement. In a number of LSI technologies, elements may be placed at any location on the chip as long as minimum inter-element distances are observed. In other organizaticns (gridded technologies), of which STAR is an example, a grid is specified and element positions must be selected to align with the grid. Organizations of the first kind are attractive since they, in general, provide the most dense element packings. The existence of a grid, however, is a great aid to chip modelling and element positioning and tends to simplify placement routines.

A third organization, related to the gridded LSI organization, specifies "slots" on the chip into which the elements are fitted. This technique is based on the characteristics of printed-circuit boards and is very seldom used for placement of elements in the active area of an LSI circuit. Pad locations, however, are typically maintained at fixed positions on the chip periphery and may be viewed as slots into which the circuit pads are fitted.

Since the "goodness" of a placement is judged on the basis of criteria defined on the element interrelations, these criteria must reflect all placement characteristics which are desired in the final element arrangement. Several desirable characteristics and the corresponding criteria are described in the following section.

Optimality Criteria

As previously outlined, the placement optimality crite-ia selected for use with a placement routine will determine the extent to which the results of the routine will possess desired characteristics. Among the characteristics which are most commonly used as the objectives of placement routines are:

1. minimum required interconnection length,
2. minimum required non-linear routing paths,
3. minimum routing channel crowding, and
4. maximum routing ease.

Since the result of the placement step can be viewed as a foundation for the routing step, characteristic 4 is typically the most desired quality for the results of any placement system. However, the translation of "maximum routing ease" into a quantitative measure which can be used for placement optimization is anything but straightforward. First, a multiplicity of routing techniques exist and placements which are "easily" routed by one may not be routeable by use of : other. Second, even if a known routing system is to be used, information regarding routing ease is sketchy - a placement can either be completely routed (easy routing) or not (hard routing) and differences between the "easy" and "hard" cases may be neither apparent nor consistent.

The use of better-defined, more measurable characteristics has thus gained acceptance in modern placement systems. By far, the most popular characteristic used is characteristic 1 (minimum required interconnection length). The wide acceptance of this characteristic as the most important (often the only) driving force behind a placement system is based on several factors:

1. total interconnection length is relatively easy to estimate given only the element placement, the net lists, and a presumed routing scheme,
2. the criterion to be used in the placement procedure for optimality measurement is easily derived from the characteristic (for example, placement A is more nearly optimum than placement B if the total interconnection length of A is less than that of B), and
3. as noted by Hanan and Kurtzberg [3], an LSI placement in which total interconnection length is optimized may be near-optimum in other respects, such as ease of routing.

The rationale behind factor 3 may be summarized as follows: the existence of many long (relative to the chip size) interconnections increases routing difficulty by using a large fraction of the total available routing area and by forming blockages of other interconnections which would optimally use portions of the same paths. By minimizing total interconnection length, many of the elements in a net are assigned locations in proximity to each other, thus reducing the average interconnection path length and (hopefully) the number of long paths, leading to increased routing ease.

No known proof of the relationship between minimum total interconnection length and routing ease has been presented. However, the large number of existing placement routines which optimize with respect to this characteristic indicates satisfactory placement routine performance is obtainable.

Characteristic 3 (minimum routing channel crowding) stems from the typical finite-capacity routing areas (channels) available in gridded technologies and from the desire for maximum chip density in non-gridded technologies. Since interconnection paths occupy space which could otherwise be used for placement of logic elements in the non-gridded applications, the motivation behind this characteristic for these technologies is immediate.

In the gridded LSI organizations, locations and capacities (maximum number of routing paths) of routing channels are typically pre-defined. Since all inter-element connections must be routed by way of these channels, the placement formed must not force their overuse.

Precise calculation of the channel usage requirements for a placement of a given circuit is not feasible without performance of the routing step. However, it is possible to obtain simple estimates of channel crowding. Methods for this estimation will be given in a later chapter.

The second desirable characteristic (minimum non-linear routing paths) has its basis in a technique used in many LSI technologies. In these organizations, all interconnection paths are composed of horizontal and vertical segments. The horizontal segments of all paths reside on one layer of the chip and the vertical segments on another layer. Path segments on the two levels are connected by vias through the insulation layer.

The three levels (horizontal, vertical and insulation) are formed in different processing steps by use of different patterns (masks). At each point at which a via exists, strict alignment between the levels is required. Allowable level alignment tolerances vary inversely with the number of vias. Thus, with increasing vias, the difficulty of chip fabrication will, in general, increase. Since each non-linear interconnection path consists of both horizontal and vertical segments, the minimization of non-linear paths can be expected to aid in reduction of fabrication difficulties and thus, may be a placement technique objective.

In the placement step, the number of non-linear routes can only be minimized by assuring either horizontal or vertical alignment of connection points on each element. For many LSI technologies, in which only one tie point (pin) exists on an element for each incident net, the alignment problem is very difficult to solve. For other LSI organizations, including that with which this dissertation is concerned, a number of tie points exist for each interconnection to an element and element alignment can be achieved in a number of ways. For these latter organizations, the objective of minimizing the number of non-linear routes may be feasible.

In this section, various desirable characteristics of LSI placements have been presented and criteria for achieving the characteristics in the solution of the placement problem have been outlined. The following sections will discuss the placement problem with respect to classical problems and will present modelling techniques commonly used for solution.

Circuit Modelling and Problem Characterization

Since the placement problem is computationally inconvenient in many respects, it is fortunate that simple, yet powerful circuit modelling tools are available. As in many electrical problems, the most facile modelling techniques use circuit representations in the form of graphs or in forms derived therefrom. In the following paragraphs, modelling methods will be outlined and will be used to gain insight into the placement problem.

Figure 9 shows a simple digital circuit and a graph model derivable from it. Several general characteristics of circuit models for the placement problem can be seen from this figure. The mapping of elements to nodes and element interconnections to edges is typical of logic circuit models. While other modelling applications (e.g. simulation) may require directed edges for preservation of signal flow sense, placement routines are, in general, exclusively concerned with the existence of connections between elements so that an undirected graph is satisfactory.

While modelling of connections between elements (such as A - E) is immediate, the mapping of a higher-order net (the 3-element net F) onto the graph model is not. The modelling process in the case of Figure 9 is such that the net F is represented as a complete graph on elements 4, 5, and 6.

Two major problems exist with this net-to-complete graph model. First, since the number of edges in a complete graph on n nodes is

$$\frac{n(n-1)}{2} \qquad \text{(from [4])}$$

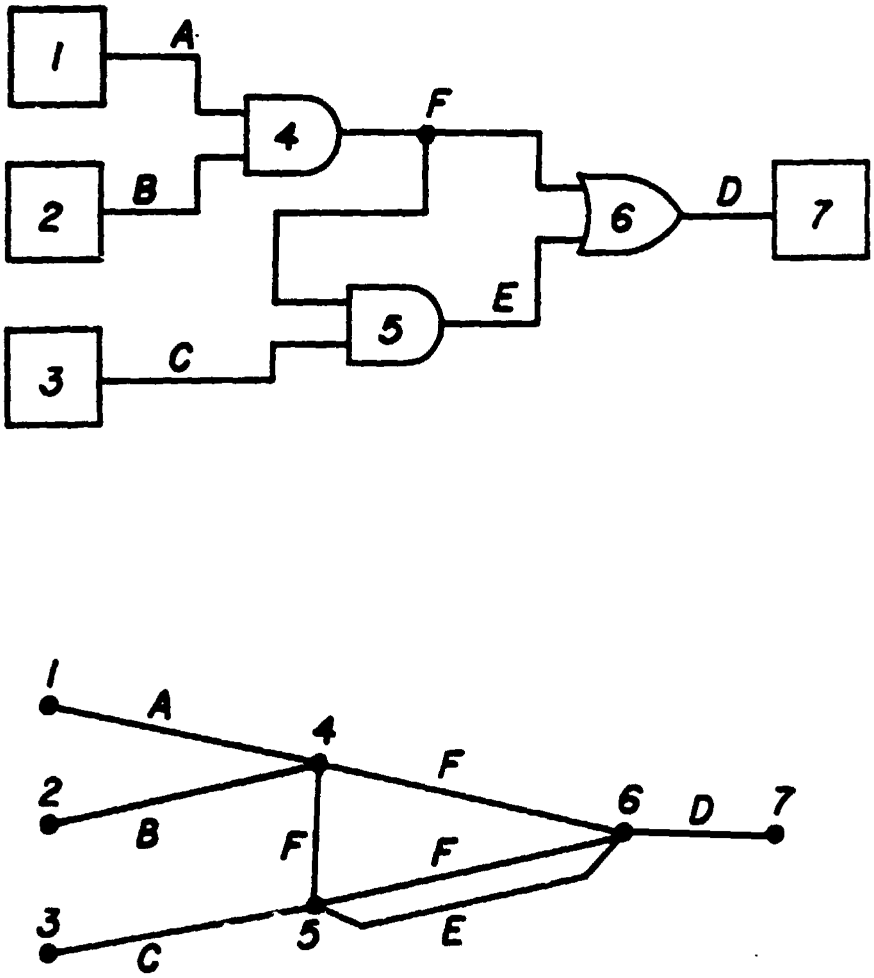
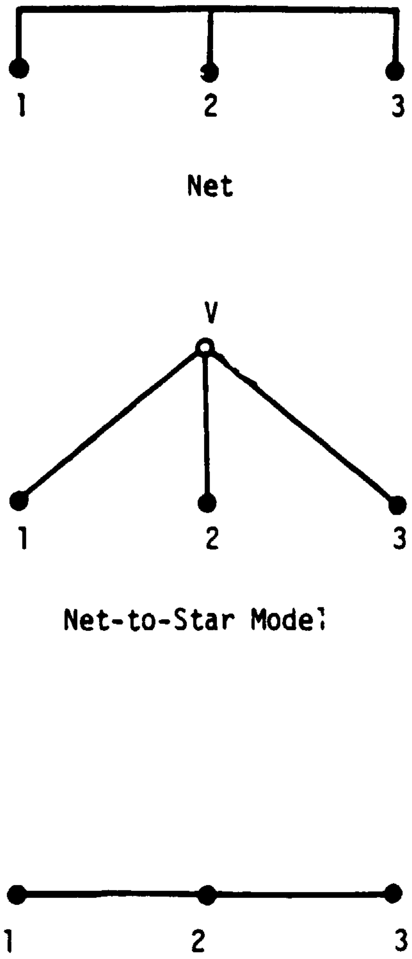


Figure 9. Logic Circuit and Equivalent Graph Model

the data required for element interconnection description increases rapidly for circuits containing large nets. Second, modelling an n-element net as a complete graph on the n nodes implies a high degree of "connectedness" between the elements whereas the elements are actually only joined by one interconnection. Many existing placement algorithms tend to place tightly connected elements in clusters on the chip. The groupings formed, then, may be unjustly biased toward large nets at the sacrifice of smaller, equally important, nets.

Two alternate graph models for a 3-element net are shown in Figure 10. The net-to-star model shown in this figure (proposed by Goldstein and Schweikert [5]) overcomes tne problems of the previously-described technique by modelling the net as a star centered on an artificial vertex V. The net is then correctly represented with a single incidence on each element vertex. However, the connections between elements are inaccurately modelled as indirect (through the vertex V). Special handling of the net vertices in the placement routine is possible, but the requirement for recognizing two vertex types may unnecessarily complicate the placement routine.

The net-to-chain model shown in Figure 10 is, conceptually, the simplest of the modelling techniques shown. In this model, the 3-element net has been represented as two edges forming a chain between the vertices. While



Net

Net-to-Star Model

Net-to-Chain Model

Figure 10. Alternate Graph Models for Net

this is the most computationally convenient of the three methods, it is important to realize that the flexibility inherent in the ordering of the elements in the net has been lost. For example, if the placement routine performs such that connected elements are adjacent in the horizontal direction, each of the 6 placements (123, 132, 213, 231, 312, 321) should be equally acceptable. However, for the net-to-chain model, only (123) or (321) are accepted. Since others of the 6 possibilities might provide a more optimum total placement (due to connections to elements not in the net), the placement routine performance may be poor.

The graph models shown in this section all map circuit elements onto vertices and connections onto edges. Alternate graph models are available which can provide excellent representations of a logic circuit for purposes such as simulation and routing. Since these models are only infrequently proposed for use with the placement problem, they will not be discussed here. Vancleemput and Linders [6] have presented an excellent summary of the forms and characteristics of these models.

While the circuit graph model is useful for human understanding, alternate forms derivable from the graph are more suited to computer implementation. These forms are typically matrices which specify the network structure. Among the matrices most commonly used for the placement problem are the incidence matrix, A, which specifies edge-vertex adjacencies and the adjacency or connection matrix, C, which describes vertex-vertex adjacencies (i.e., connected elements). Of these, the connection matrix (defined by C (I, J) = the number of edges between vertices I and J) is most common.

Unfortunately, the matrix representations may not be computationally feasible for large networks since the C matrix grows with the square of the number of graph vertices (circuit elements) and the A matrix grows with the product of the number of vertices and edges. Because the matrices are typically sparse (a large fraction of the elements are 0) or symmetric (in the case of the C matrix), data reduction techniques might be used to reduce size requirements. However, the main attractiveness of the matrix representation (fast access to circuit connection data) may be lost.

An alternate method of circuit data representation, which requires significantly less storage, is the maintenance of lists detailling, for each circuit net, the elements upon which the net is incident. While no explicit net model is required for this technique, data convenience and access speed are sacrificed by its use. This method has been adopted for use in the placement system which this report describes and further discussion will be deferred to a later section.

As previously mentioned, some of the circuit graph models can be used to provide insight into the placement problem. In particular, by constructing the net-to-complete graph model as described, the general logic circuit placement problem is reduced to a problem in which a simple relation (defined by the graph edges) exists between each pair of vertices (elements). The reduced placement problem then becomes the problem of optimally assigning elements to chip positions with respect to criteria defined on relations between each pair of cells, which is equivalent to the classical quadratic assignment problem [3]. There are no known methods for the solution of the quadratic assignment problem which are computationally feasible with respect to execution time. In fact, exact solution methods for this problem are merely strategies for complete investigation of the solution space.

This apparent lack of promise in the search for efficient methods for the exact solution of the placement problem explains the profusion of heuristic procedures which have been developed for identification of near-optimum solutions. A number of these procedures have proven particularly successful and will be presented in the following section.

Prior Work

As noted in the preceding section, numerous methods for approximate solution of the placement problem have been proposed. The intent in this section is to outline characteristics of various method classes and to present existing solution methods in the framework of this classification.

Classification of Techniques

The placement techniques to be described might be classified in a number of ways. For convenience here, two major divisions of placement problem solution methods will be recognized. These are initial placement (IP) techniques and placement improvement (PI) techniques.

The IP class will consist of all techniques which form an element placement from an unplaced set of elements and interconnections. The PI class will contain methods which modify a given starting placement to produce a more nearly optimum placement.

Since many composite placement systems can be constructed by following an IP technique with one or more PI techniques, no attempt will be made to detail these systems.

Initial Placement Techniques

Conceptually, the simplest of the IP techniques is the Monte Carlo (or "shotgun") placement method [3]. In this procedure, the circuit elements are randomly assigned to locations on the chip on the basis of a uniform distribution. The assignment procedure is repeated a large number of times and the most optimum placement is retained. As noted in [3], the performance of the procedure is poor due to the extremely low probability of randomly selecting a "good" placement from the solution space.

Also noted in [3] is the existence of a Monte Carlo technique in which the probability with which an element is assigned to a particular location is biased by the past experience of "good" assignments for the element. Performance of this technique is limited by the time required for distribution adjustment of each iteration.

The pair-linking IP technique [7] represents a considerable improvement over the Monte Carlo methods. In this procedure, the most highly-connected pair of elements is selected and placed on the chip to form the placement nucleus. In each following iteration of the procedure, the unplaced element which is most connected to a placed element is selected and is placed as near as possible to its partner.

A related technique, cluster development [7,8], begins with a nucleus element which is positioned on the chip. On each succeeding iteration, the unplaced element which is most connected to elements in the nucleus is selected and is placed as near as possible to the center of the positions of the placed elements it is connected to.

Due to their simplicity and relatively good performance characteristics, the pair-linking and cluster development techniques are among the most commonly-used IP routines. However, due to their somewhat restricted view of the global placement characteristics, actual application systems utilizing these techniques almost always follow them with one of the more powerful PI routines.

Branch and bound techniques [9,10] ave been shown to be capable of forming excellent solutions to the quadratic assignment problem, and hence can be applied to the placement problem. These techniques are the only commonly-proposed methods which can be used to find an optimum placement.

The branch and bound methods, in general, give a strategy for partitioning the solution space and for searching for the optimum in each partition. Lower bounds on the non-optimality (cost) of the solutions in each partition are computed and the search for the optimum in a partition is terminated when the lower bound exceeds the cost of some previous solution. An excellent description of the process is shown by Hanan and Kurtzberg [3].

While the bounding strategy eliminates the need for examination of many regions of the solution space, the time requirements of the procedures are too excessive for practical application.

Modifications to the exact branch and bound procedure which allow the isolation of near-optimum solutions have been proposed by several authors (in particular, Gilmore [11] and Hillier and Conners [12]). These approximate branch and bound techniques can be utilized to significantly reduce the solution time required by the exact scheme. However, the complexity of the approximate methods remains high (on the order of the fourth power of the number of elements as compared to the second power for pair-linking and cluster development [3]) and time requirements may remain prohibitively high.

The final IP technique to be discussed will be referred to as the linear ordering-folding (LOF) technique. In this procedure, the placement problem is effectively divided into two parts: formation of a near-optimum one-dimensional placement (linear order) and "folding" of the linear order onto the chip.

This method has been used for a number of LSI technologies in which the final placement can conveniently be organized as rows of elements. Several of these technologies use the MOS complex (or array) organization [13,14,15] in which elements (typically, at the transistor level) are serially interconnected to form the desired logic function. The elements are arranged as required in the linear order and a simple folding operation suffices to arrange the order of the chip. Larsen [14] provides an excellent discussion of the technology and possible layout techniques.

A more general LSI organization in which the LOF technique has seen use is the polycell layout shown in the previous chapter. In this organization, elements (at the logic gate level) are arranged in back-to-back double rows and element interconnections are routed between the rows in interconnect channels.

The LOF technique is easily adapted to this layout organization since the elements can be placed in a one-dimensional order and the rows can be easily obtained by isolating appropriately-sized segments of the order and arranging them on the chip. The use of LOF procedures to obtain these organizations is reported by Mattison [16]. Similar techniques are used in the RCA-developed PRF program [17].

The LOF procedure is attractive due to its relative simplicity and ease of adaption to certain LSI organizations. Application of LOF techniques to an alternate organization will be described in later sections.

Several existing IP methods have been outlined in this section. In the following section, a number of placement improvement techniques are discussed.

Placement Improvement Techniques

Many existing PI procedures can be roughly categorized as interchange techniques. Among the simplest of these is the pair-wise interchange (PWI) placement improvement technique. In this procedure, a pair of elements in the placement is interchanged and the optimality of the resulting layout is calculated. If an improvement results, the new placement replaces the old. Each pair of elements is trial interchanged during an iteration and iterations continue until no further improvement is made or until a desired degree of optimality is obtained.

As noted in [18], due to the large number of interchanges and optimality computations required during an iteration, the PWI technique is excessively time-consuming for large circuits. A variant to the basic PWI procedure which attempts to overcome this problem is the neighborhood-PWI (NPWI) technique. This routine limits the number of trial interchanges in an iteration by considering only element pairs which lie within a distance, D, of each other.

Use of PWI methods (or variants) in application environments has been reported in the BTL NOMAD system [8], Raytheon's IPLACE [19], and in the Circuit Design System developed at ADAGE, Inc. [20].

A much more sophisticated interchange routine is that reported by Steinberg [21]. This procedure achieves higher performance than the PWI techniques by handling groups of elements rather than pairs. The groups for consideration are formed by partitioning the circuit elements into "maximal independent sets" which are the largest partitions that can be formed such that no two elements in a partition are connected.

Placement improvement proceeds by removing a maximal independent set from the placement and re-assigning the elements among the available locations. Since none of the removed elements are connected, the re-assignment problem is linear in nature and allows use of simple linear assignment techniques for solution.

Placement technique comparison [18] has shown that Steinberg procedure performance lies below that of the NPWI technique for many problems. However, the algorithm has seen use in several application systems including the UNIVAC Automated Design System [22] and the Raytheon IPLACE system [19].

A third general type of interchange technique is that typified by the Min-Cut Placement Algorithms described by Breuer [23]. In these procedures, an imaginary cut-line divides the chip and elements are interchanged across it such that the number of interconnection paths crossing the cut line is minimized. On succeeding iterations, other cut lines are drawn and used for swapping (while recognizing the boundaries specified on preceding passes). The process is one of successively refining an estimation of the optimum location for each element.

These algorithms have the added feature that they can act as IP techniques as well as PI (i.e., elements can be assigned to each side of a cut-line without specifically identifying their location). Breuer refers to one application system using this technique (PRANCE, by Automated Systems Inc.), although no data on system performance is available.

Relaxation PI techniques represent a radical departure from the interchange methods just described. These routines effectively model each element as a point source with the interconnections modelled as springs between point sources. Each point source (element), then, has forces applied to it in the directions of and proportional to the distances to all other elements to which it is connected. A target location for the element can then be identified as the location at which the forces on the element are zero.

In the simplest relaxation technique, the forces on each element are calculated in turn and the element is moved to its target location if the location is not occupied. While this might be satisfactory for sparsely-populated (small ratio of elements to element locations) chips, the probability of the target location not being occupied in a densely-populated chip is very low.

To overcome this problem, alternate relaxation techniques have been proposed. One of these, force-directed relaxation [3,18], either selects an available location as near as possible to the target location for placement or displaces the element at the target location, which is then relocated in a similar manner.

Variants of the force-directed relaxation techniques include the force-directed interchange technique [24] which uses force-directed concepts to identify profitable element interchanges in what is, otherwise, an interchange technique.

Problems occur in the use of these techniques due to the use of a point-source for modelling of the finite-size element. In particular, overlapping of elements in the final placement is possible and usual. Generally, then, post-processing routines are called on to eliminate element overlapping without destroying the relative placement formed. Typical of these post-processors is the EXPAND process described by Scanlon [25].

Relaxation techniques appear to be among the most powerful and efficient PI techniques available [18] and, not surprisingly, among the most popular. A number of reports of use of these techniques have been made, among them [8,25,26,27,28,29,and 30].

A discussion of the LSI cell placement problem and a number of methods for its solution have been presented in this chapter. A modification to the LOF placement technique which allows handling of STAR-like structures is presented in the next chapter.

III. THE LINEAR ORDERING-FOLDING (LOF) TECHNIQUE

The preceding chapters outlined the meaning and characteristics of the LSI cell placement problem, methods for its approximate solution, and the organization of a semicustom LSI technology (STAR). The intent in this chapter is to illustrate the development of cell placement routines suitable for use with STAR technology.

In particular, the use of linear ordering-folding techniques for STAR cell placement will be described. The first section will deal with existing linear order formation techniques. In the second section, the development of the folding techniques to be utilized will be given.

The Linear Ordering Procedure

The STAR cell linear ordering problem can be considered as a special case of the STAR cell placement problem in which only one-dimensional placement is performed. Intuitively, this is a simpler problem. In fact, the solution spaces of the two-dimensional problem and an equivalent (same number of grid positions) one-dimensional problem are equal in size and the difficulty of exact solution of either problem is the same.

The advantages of the linear ordering problem are due to the nature of the approximate placement problem solution techniques which, in general, have as their objective the location of connected cells as near as possible to each other. Since nearness in four directions can be achieved for the two-dimensional case (as opposed to two for the one-dimensional case), the nearness decision processes for linear ordering are inherently less complex and near-optimum (one-dimensional) solutions can be more quickly obtained.

While, conceivably, any process suited for approximate solution of the two-dimensional placement problem can be adapted to the linear ordering problem, the linear ordering techniques proposed by Schuler and Ulrich [31] seem to hold the most promise.

These techniques efficiently achieve near-optimum (with respect to total interconnection length) one-dimensional solutions by a two-stage process. The first stage, clustering, combines pairs of interconnected cells or pads until all circuit elements are contained in one cluster. In the second stage, decomposition, the clusters are located in the one-dimensional placement and are iteratively decomposed into their constituent cells.

In the following paragraphs, these processes are detailed. For convenience, the term "cluster" is used to describe any group of one or more combined circuit elements.

The clustering process begins with identification of the "most combinable" pair of circuit elements. This pair is combined to form a cluster and the combination is noted in a record of cluster formation (the CHR). The combined cells are deleted from the set of clusters eligible for further combination and the new cluster is added.

Succeeding iterations of the procedure identify the "most combinable" cluster pair and form new clusters as before. The clustering process terminates when only one cluster remains. The clustering procedure is illustrated in the flow diagram shown in Figure 11.

The results of the clustering step can be visualized in the form of a binary tree such as that shown in Figure 12. The nodes of this tree represent the clusters and the branches show the cluster composition. The lower terminal nodes of the tree are the original circuit elements.

The decomposition procedure can be easily conceptualized by consideration of this tree form. Each of the binary subtrees can be rotated about its root cluster into either of two configurations as shown in Figure 13. If each binary subtree is rotated into its more optimum configuration, (relative to other subtrees) the implication is that the optimality of the terminal node order is improved. Since a binary tree with n terminal nodes contains (n-1) proper binary subtrees, the number of optimality comparisons required is (n-1).

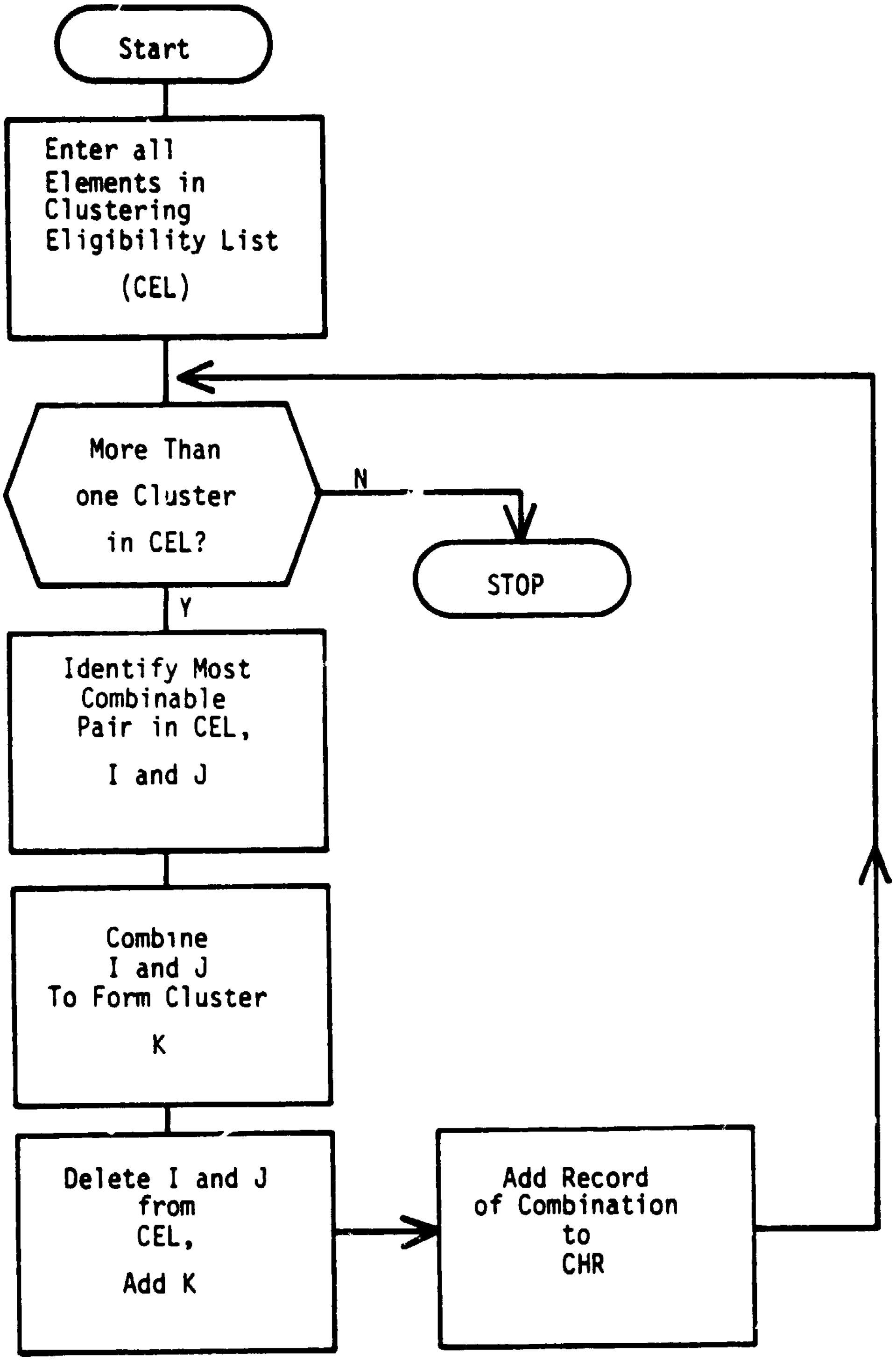

Start
Enter all Elements in Clustering Eligibility List (CEL)
More Than one Cluster in CEL?
N
STOP
Y
Identify Most Combinable Pair in CEL, I and J
Combine I and J To Form Cluster K
Delete I and J from CEL, Add K
Add Record of Combination to CHR

Figure 11. Functional Flow of Clustering Procedure

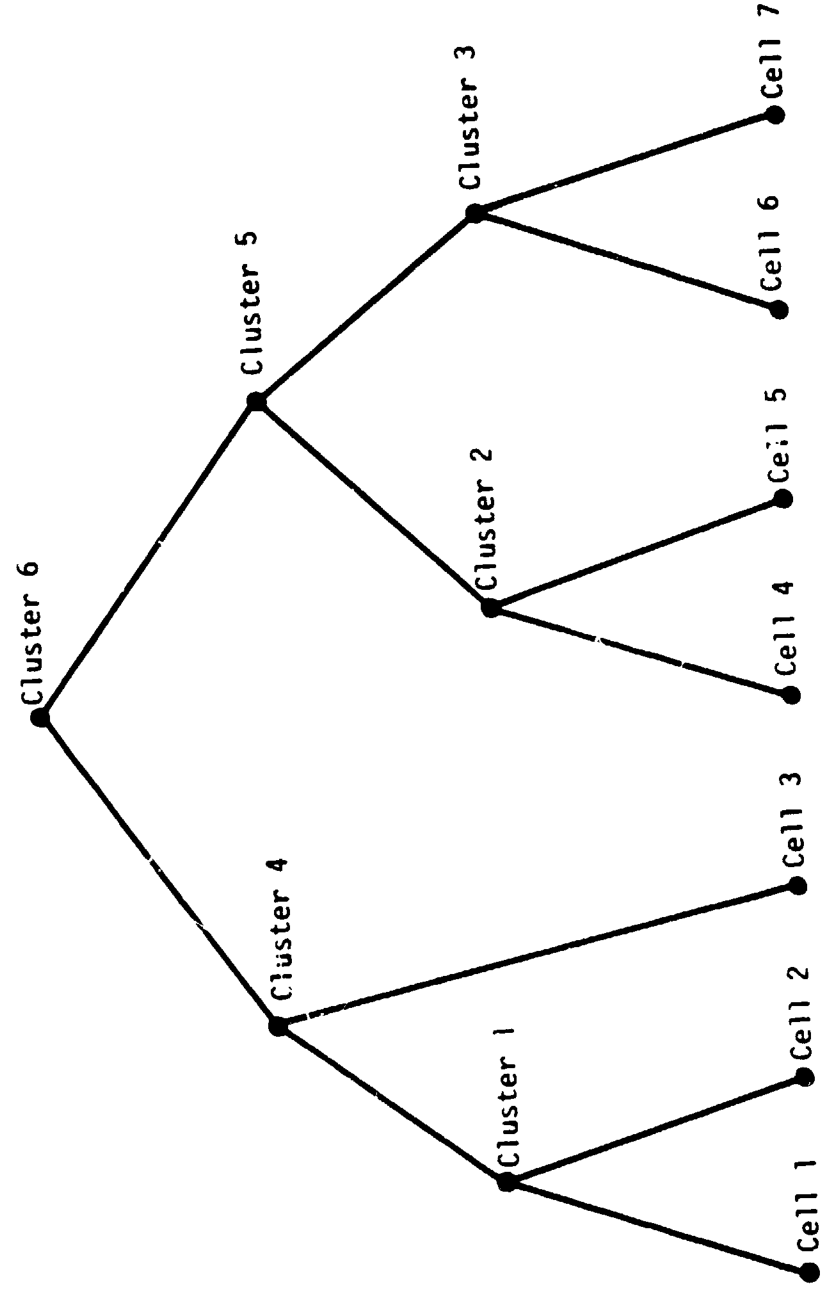
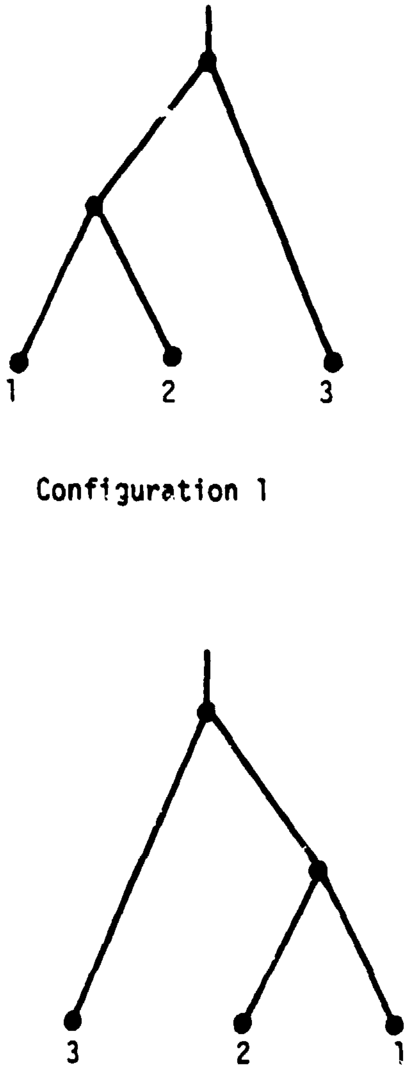


Figure 12. Binary Tree Representation of Results of Clustering Step



Configuration 1

Configuration 2

Figure 13. Subtree Rotation

The decomposition process is implemented by simulating this subtree rotation. The process begins by placing the two constituent clusters of the final cluster formed in an arbitrary sequence in the linear order.

Succeeding iterations of the procedure identify the latest-formed cluster in the linear order and replace it with its constituents. The optimality of each of the two possible orientations of the constituents is calculated and the more optimum configuration selected. The process terminates when all elements of the linear order consist of a single circuit element. The decomposition procedure is shown in the flow diagram of Figure 14.

Two optimality decision processes are required for performance of the linear ordering procedure. The first of these occurs in the clustering step when it is desired to identify the "most combinable" clusters. Schuler and Ulrich propose a method by which the connectivity of a cluster pair is evaluated relative to its connectivity to other clusters. The pair with the highest relative connectivity is then selected for combination. This criterion has the effect of achieving a near-minimum of interconnections between clusters at each clustering step and aids in producing linear orders which are near-optimum with respect to total interconnection length. Use of a similar technique is reported in [32].

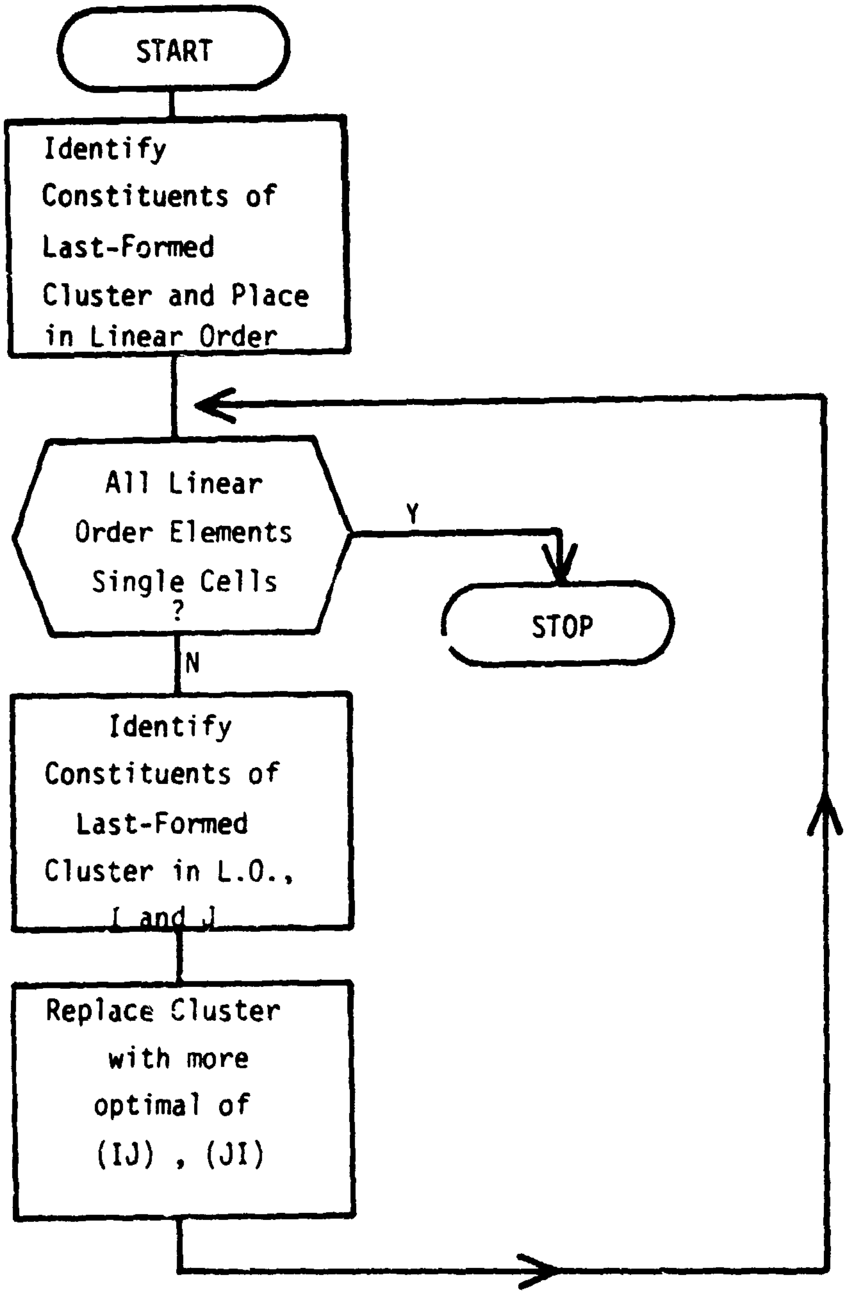

START
Identify Constituents of Last-Formed Cluster and Place in Linear Order
All Linear Order Elements Single Cells ?
Y
STOP
N
Identify Constituents of Last-Formed Cluster in L.O., I and J
Replace Cluster with more optimal of (IJ) , (JI)

Figure 14. Functional Flow of Decomposition Procedure

While the Schuler and Ulrich combination criteria perform well for many application environments, an alternate method, more suited to the STAR technology, is used in the linear ordering procedure deveioped. This combination criterion, developed by J. Gould of the Marshall Space Flight Center, will now be described.

The Gould combination criterion is based on a model which attempts to account for cluster size (sum of the constituent cell widths) in the minimization of average interconnection length. A cluster of size w^2 is modelled as a square with side length w and all interconnections to the cluster are considered to emanate from the square (nets between cells in the same cluster have zero length).

The escape distance, ED, is defined as the average horizontal and vertical distance that a connection must traverse in order to run from the inside to the outside of the square (cluster). From the model, ED can be easily estimated as the sum of one-half the horizontal and one-half the vertical square dimension, or, the square root of the cluster size.

It is now possible to consider three asses of nets on two clusters, A (size x^2) and B (size y^2) and to estimate the effects on interconnection length if the clusters are combined. Each of the three classes is illustrated in Figure 15. The first class, the non-connecting class, is that of an interconnection to A but not B. For this class,

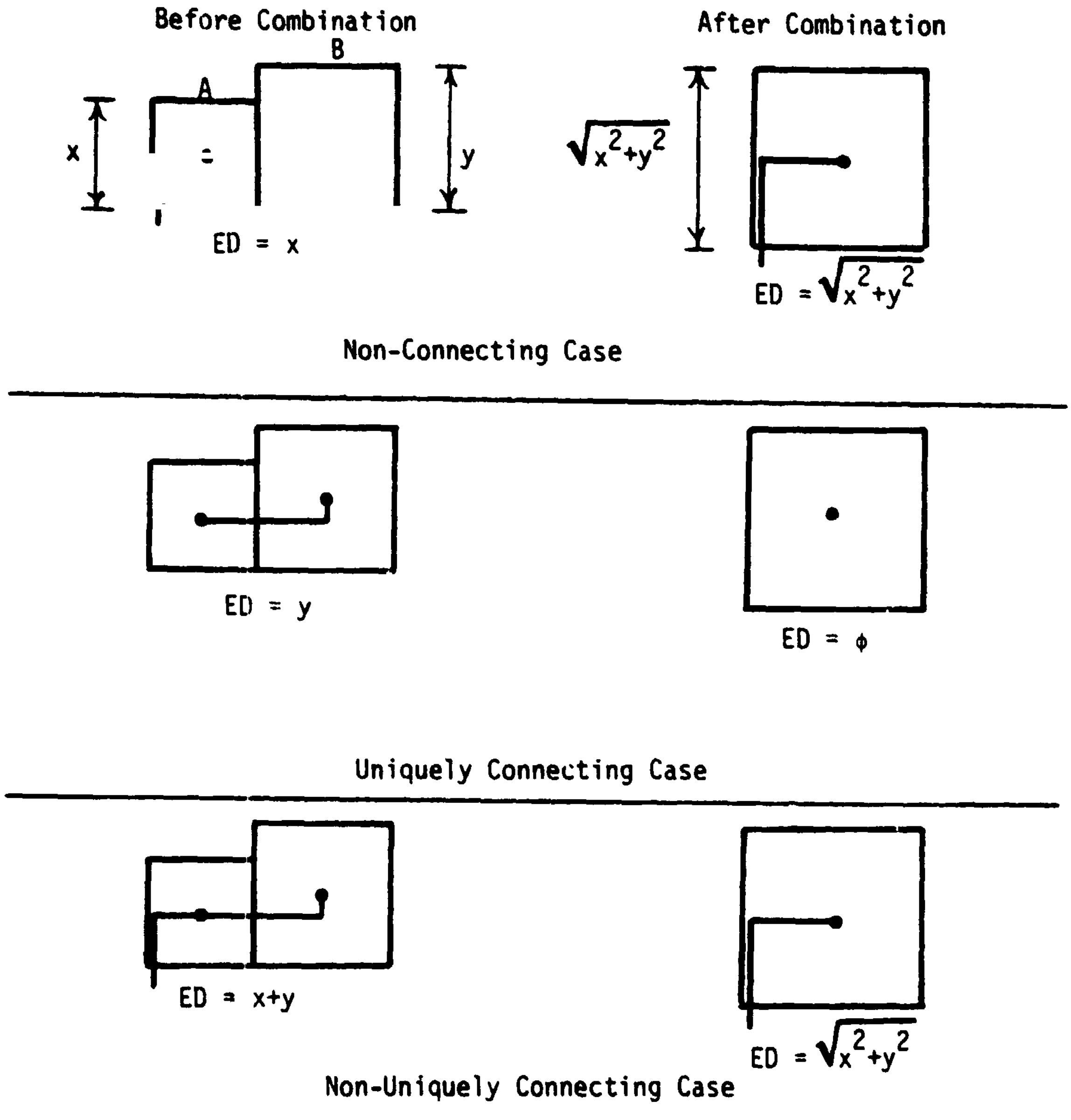


Figure 15. Gould Criteria Clustering Models

the ED for the net before combination is

$$2(0.5x) = x$$

and after is

$$2'.5\sqrt{x^2 + y^2}\ ) = \sqrt{x^2 + y^2}$$

Interconnections from B are treated identically by interchanging x and y.

The second class, the uniquely connecting class, contains those nets which run between A and B, but no other cluster. The ED for this class before combination is

$$0.5x + 0.5y + 0.5y - 0.5x = y$$

and is 0 after combination.

For the third class, containing nets between A and B which also run to other clusters, the total ED before combination is

$$x + y$$

and after combination, is

$$\sqrt{x^2 + y^2}$$

For the pair of clusters A and B, then, the total ED before combination is

$$ED = NC1Ax + NC1By + NC2y + NC3(x + y)$$

where NC1A is the number of class 1 connections to A, NC1B is the number of class 1 connections to B, and NC2 and NC3 are the number of class 2 and 3 connections between A and B. The total ED if the clusters are combined is

$$ED' = (NC1A + NC1B + NC3)(\sqrt{x^2 + y^2}\ \ )$$

The improvement (reduction) in total escape distance expected by combination of the clusters A and B is

$$EDI = ED - ED'$$

The Gould procedure selects a cluster A, and for each cluster B which is connected to it, computes the EDI for the pair. The B which produces the maximum EDI is selected for combination with A.

If cluster B is selected such that its size is equal to or greater than that of A, it can be seen that maximum (positive or negative) improvement for each of the three interconnection classes is attained when the size of B is equal to that of A $(y = x)$. Since all clusters must be eventually combined, maximum improvement can be made by selecting the smallest clusters early in the procedure when other small clusters are available. The rule used for selection of the cluster A, then, consists only of choosing the smallest cluster available.

The second optimality decision in the linear ordering procedure is required in the decomposition step. This decision must isolate one of the two constituent orders possible when a cluster is to be replaced.

The criterion used for this choice is based solely on minimum interconnection length considerations. The number of nets by which each of the constituents is connected to the left and right of the target location is calculated and the orientation is selected which minimizes the total connection distance.

The result of the linear ordering procedure outlined in this section is a one-dimensional placement in which the total interconnection length is near-minimum (considering cell width). In the following section folding techniques which can be used to map the STAR cell linear order onto the STAR will be developed.

The Folding Procedure

As noted in a previous chapter, folding methodologies have been developed for various LSI organizations. For these organizations, however, the required chip layout has made obvious the folding strategies required and, for the most part, the methods developed have been of an extremely simple nature.

In this section, folding techniques suitable for use in STAR and STAR-like organizations are developed. While the complexity of these methods is greater than that of the simpler folding techniques, the desirable qualities of the folding procedures (i.e., speed and relative simplicity) have not been sacrificed.

Following introductory material detailing folding objectives, the methods will be presented in two parts. First, folding techniques will be presented that are suited to placement of circuits consisting of uniform-size cells.

Next, the more complex problem of folding of networks containing cells of non-uniform size will be treated. Since separate chapters are devoted to placement optimality measurement and folding procedure performance, these items will not be discussed in detail. Also pad placement, which is discussed in a later chapter, is ignored here.

Folding Objectives

Briefly stated, the objective of the folding portion of an LOF technique is "map the linear order onto the chip without disturbing the relative cell positions". The simplistic nature of this statement is due to the intended character of the LOF method in which the linear ordering segment is to perform the "hard" work (i.e., relative cell position assignment) and the folding segment merely to lay out the chip in a pre-defined manner while obeying the relations formed.

For STAR organizations, however, it is possible for the folding segment, while regarding the specified linear order, to increase the optimality of the final placement over that present in the one-dimensional case. This improvement can be achieved by use of folding strategies which decrease the distance between connected cells or which place a higher percentage of cells in juxtaposition than specified in the linear order.

In general, the folding methodologies presented rely on the assumption that the linear order represents a near-optimum one-dimensional cell placement with respect to total interconnection distance. The folding procedures, then, are developed and justified on the basis of preserving and augmenting the relations given by the linear order.

Finally, while optimality maximization is desired, the fact that a primary objective of the folding technique is to fit the cells onto the finite STAR cannot be ignored. Since failure by the routines to form a STAR placement may require the use of expensive manual placement techniques, the probability of the folding procedure to find a placement if one exists (regardless of optimality) should be acceptably high.

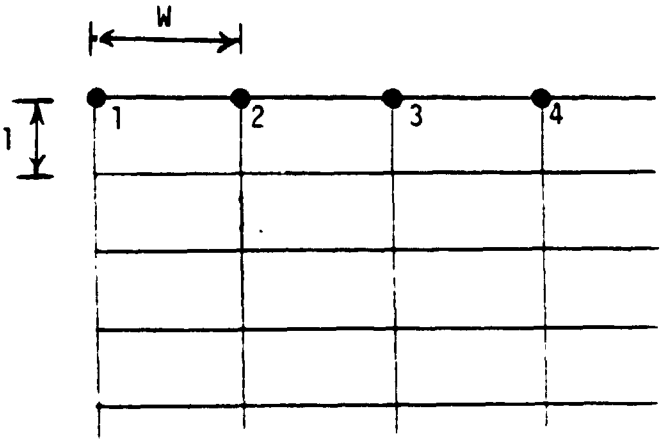
Folding for Uniform-Width Cells

Since cells in the STAR cell library are defined in a wide variety of widths, the probability of the occurrence of a random-logic custom STAR application in which all circuit cells are the same size is extremely low. However, these uniform-width networks are useful for illustration of several folding concepts and are discussed in this section.

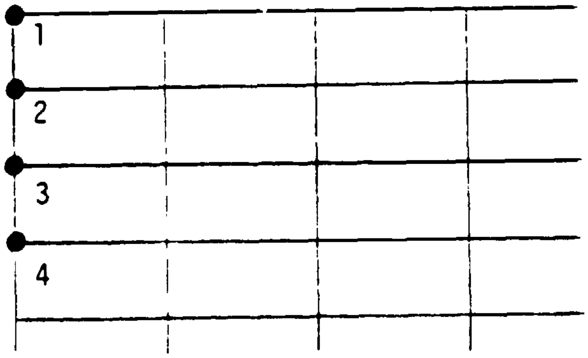
The STAR model used in this section is a reduced form of the normal STAR grid structure. Horizontal grid lines conform to the STAR rows and vertical grid lines correspond to transistor columns 1, 1+W, 1+2W, ..., where W is the uniform width of the cells. Cells are reduced to point sources at their left-hand end (with respect to orientation on the STAR). By restricting point source placement to only those positions at which a horizontal and a vertical grid line intersect, the gridded STAR placement is modelled as a more convenient slotted organization. For the purposes of this section, these simplifying approximations cause no serious loss of generality.

This model is utilized in the following paragraphs as a medium for analysis of various STAR folding strategies. The major criterion to be used for measurement of the quality of a strategy is the minimization of the average distance in the STAR placement between the Ith and (I+k)th cells in the linear order with the objective of minimizing total interconnection length and channel usage. Analyses of results of the linear ordering technique indicate that the majority of connections to a target cell in the order in a net-to-chain graph model of the linear order run to cells no farther removed than four cells from the target. The k in the statement above will thus be restricted to the range 1 to 4.

As a simple starting point, the problem of placement of a C-cell linear order on an infinite STAR will be treated. The two folding methods shown in Figure 16 are immediately suggested. For the horizontal alignment method, the average distance between the Ith and (I+k)th cells is kW. This



Horizontal Alignment



Vertical Alignment

Figure 16. Uniform Cell Folding Methods for Infinite STAR

distance in the vertical alignment method is k. Further, if at any point the vertical alignment is modified so that a horizontal component appears, the average distance is increased from k. Thus, from interconnection length considerations, the vertical alignment folding method is optimum (with respect to the linear order).

From the standpoint of minimizing global channel crowding, however, the vertical alignment method may not be acceptable. To illustrate this fact, a method for estimating channel utilization (density) is now introduced.

The desired utilization estimates should represent the expected fraction of the total available horizontal and vertical global channel area which is used in any region of the STAR. The simple model of a STAR transistor area (the intersection of a row with a transistor column) shown in Figure 17 .s used to quantify this area. If each of the portions of a channel in a single transistor area is called a channel segment, there are exactly 4m horizontal and n vertical channel segments available in an m-row by n-column portion of the STAR. For this case, 4m will be called the horizontal channel area and n, the vertical channel area .

The area occupied by a cell interconnection can also be represented in terms of channel segments. While the area occupied for the connection of cell I to cell J cannot be less than the distance between the point sources corresponding to I and J in the STAR placement model, more

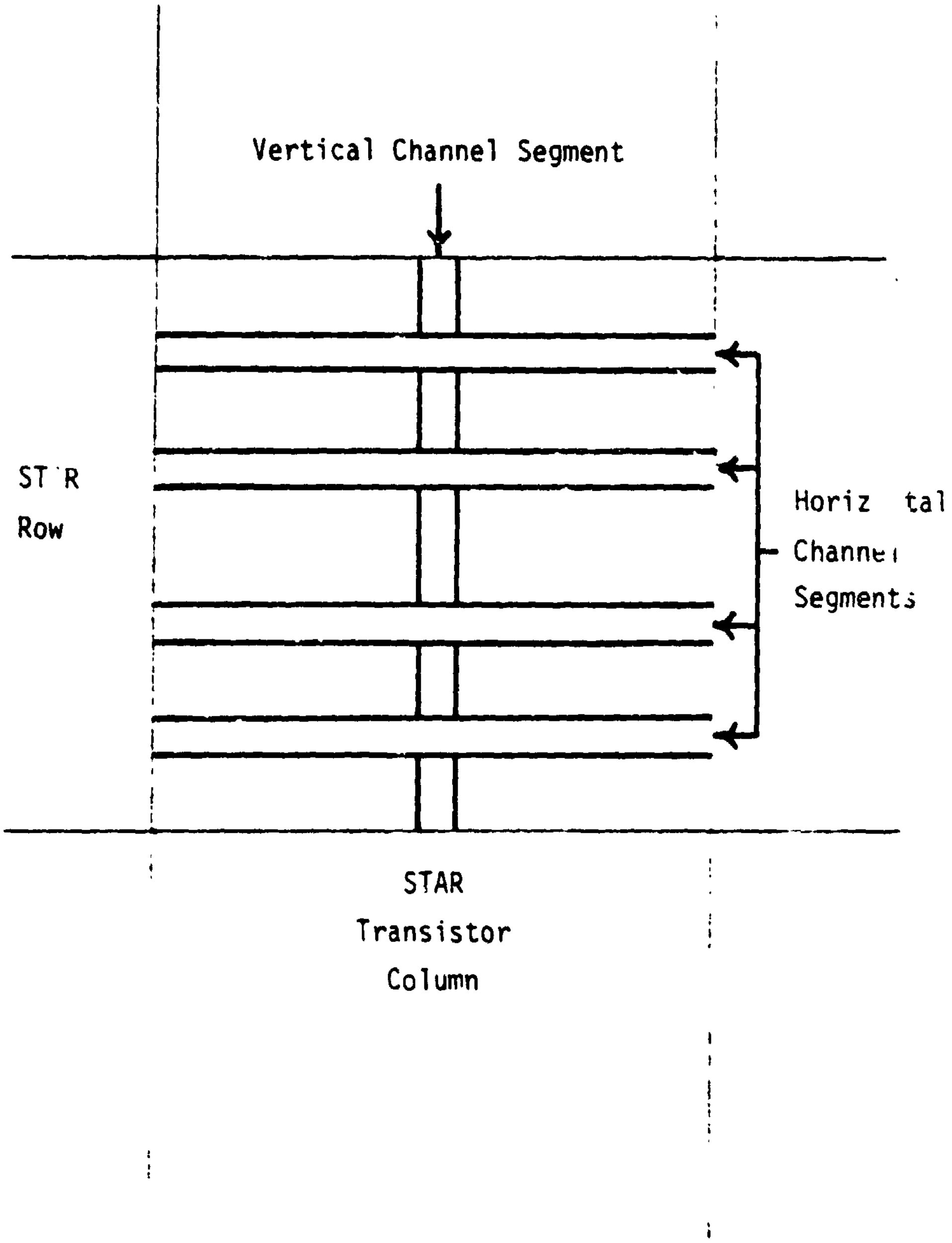


Figure 17. STAR Transistor Area Channel Model

accurate density calculations can be achieved by recognizing that a partially-used channel segment cannot be re-utilized and thus, should be considered as completely used. An interconnection between point sources a distance (d) apart, then, should be charged with the spanning distance (d+1) to account for channel usage in both terminal cells.

By use of the concepts above, channel utilization in an m row by n column area of the STAR can be estimated by

$$U(H) = \frac{L}{4mn}\left[\sum_{J=1}^{4}(H(I,I+J)F(J))\right]$$

$$U(V) = \frac{L}{mn}\left[\sum_{J=1}^{4}(V(I,I+J)F(J))\right]$$

where U(H) and U(V) are the horizontal and vertical utilizations, H(I,I+J) is the average horizontal spanning distance between the Ith cell from the linear order and the (I+J)th, V(I,I+J) is the corresponding vertical spanning distance, F(J) is the fraction of the total number of connections between cells a distance J apart in the linear order, and L is the number of connections within the n-by-m area.

It is now possible to estimate localized channel utilization for the vertical alignment case. The STAR area of interest is the C-row by W-column region in which the cells have been placed. V(I,I+J) = J+1 and H(I,I+1) = Ø for all cases. Then

$$U(V) = \frac{L}{CW}\left[2F(1) + 3F(2) + 4F(3) + 5F(4)\right]$$

As noted, the quantity L in this equation represents the number of connections in a net-to-chain graph model of the linear order. Now, the net-to-chain model of an n-cell net is a spanning tree on n vertices and contains (n-1) edges (connections). L may then be approximated by

$$L = (y - 1)N$$

where y is the average net size (in cells) and N is the number of nets in tne circuit.

To calculate y, the sum of the net sizes is divided by N, or

$$y = \frac{\text{sum of net sizes}}{N}$$

But, the sum of the net sizes is exactly equal to the number of pin.. (connection points) in the circuit, so

$$y = \frac{\text{total number of circuit pins}}{N}$$

or

$$y = \frac{zC}{N}$$

where z is the average number of pins per cell. Thus,

$$N = \frac{zC}{y}$$

and,

$$L = zC\left(\frac{y - 1}{y}\right)$$

Then

$$\frac{L}{CW} \approx \frac{z}{W} \left(\frac{y - 1}{y} \right)$$

The first term in this equation is the average pins per unit cell width. An analysis of cells available in the STAR cell library reveals that this ratio ranges from 0.25 to 1. Accepting 1 as a worst-case (highest utilization) value,

$$\frac{L}{CW} \approx \left(\frac{y - 1}{y} \right)$$

As y increases, this ratio of (y-1) to y approaches 1, so, for the worst case,

$$\frac{L}{CW} \approx 1$$

or,

$$U(V)_{WC} \approx 2F(1) + 3F(2) + 4F(3) + 5F(4)$$

It is easily seen that this quantity is not upper-bounded by 1. In fact, for the typical (experimentally-derived) values,

$$F(1) = 0.5$$

$$F(2) = F(3) = F(4) = 0.1$$

the right-hand side of this equation becomes 2.2. Equivalently stated, approximately 220% of the vertical channel segments available in the placement area are required for worst-case circuit characteristics.

Routing of the vertically-aligned placement is possible by use of the unfilled area to the right of the cells (use of an area 5w in width results in a vertical utilization of Ø.44). However, a placement which requires this type of routing is hardly likely to be classified as "easily-routed". The use of vertically or horizontally-aligned folding is thus rejected for application purposes.

An alternate folding strategy, called block-oriented folding, has been developed for use in the STAR cell placement problem. This technique can provide improved channel utilization characteristics over the simple methods presented in the previous section. In addition, the method allows recognition of the finite STAR size.

The block-oriented technique combines aspects of both horizontal and vertical alignment. Vertically aligned segments of the linear order are replicated horizontally across the STAR to form a block. Blocks are then stacked vertically. In the following discussion, the length of the vertical segments within a block is referred to as the block depth .

The simplest block-oriented folding method is typified in Figure 18. This, in fact, is the same folding structure used in the polycell organization described in an earlier chapter. The blocks in this layout are the horizontal rows of n cells, each.

For derivational convenience, in the remainder of this section, only values of 1 and 2 will be substituted for J in

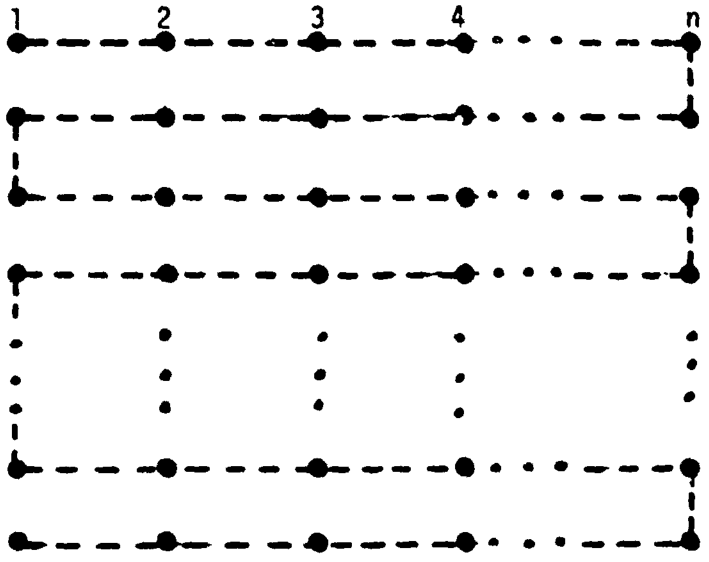


Figure 18. Block-Oriented Folding (Block Depth = 1)

the calculation of the distance to the (I + J)th cell. These average horizontal and vertical distances in this organization can be easily obtained as

$$H(I,I+1) = \frac{(n-1)(W+1)}{n} \quad ; \quad V(I,I+1) = \frac{2}{n}$$

and

$$H(I,I+2) = \frac{(n-2)(2W+1)+2(W+1)}{n} \quad ; \quad V(I,I+2) = \frac{4}{n}$$

The worst-case utilization figures for this organization are

$$U(H)_{WC} \simeq (0.175)W(1-\frac{1}{n}) - (0.125)(\frac{1}{n}) + 1.5 \qquad (3\text{-}1)$$

and

$$U(V)_{WC} \simeq \frac{1.4}{n} \qquad (3\text{-}2)$$

Derivation of equations 3-1 and 3-2 is shown in Appendix A.

Equations 3-1 and 3-2 can be contrasted with expressions for tne same quantities in a different organization (Figure 19). The average horizontal and vertical distances for this layout are

$$H(I,I+1) = \frac{(n-1)(W+1)}{2n} \quad ; \quad V(I,I+1) = \frac{n+1}{n}$$

and

$$H(I,I+2) = \frac{(n-1)(W+1)}{n} \quad ; \quad V(I,I+2) = \frac{2n+1}{n}$$

The worst-case channel utilization figures (developed in

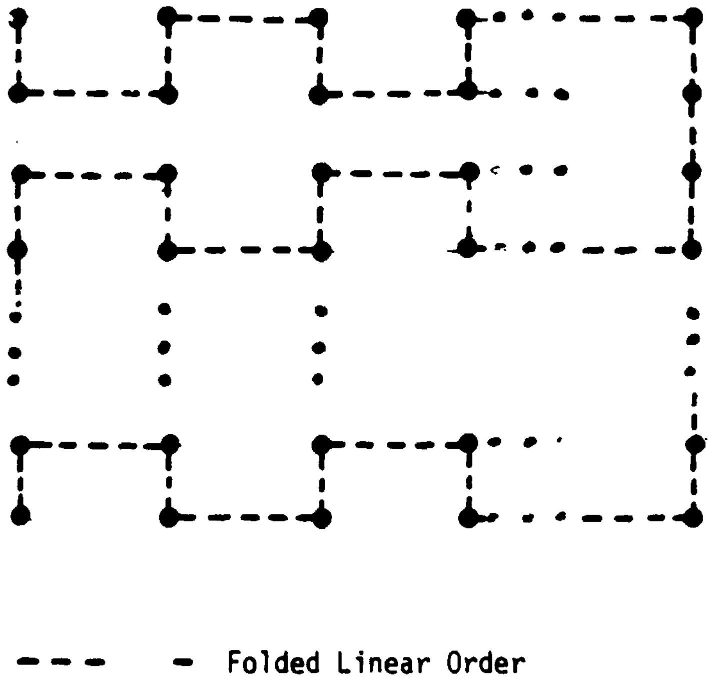


Figure 19. Block-Oriented Folding
(Block Depth = 2)

Appendix A) are

$$U(H)_{wc} \simeq (0.088)\ [(1- \frac{1}{n})W - \frac{1}{n} + 1] \qquad (3\text{-}3)$$

and

$$U(V)_{wc} \simeq 0.7 + \frac{0.6}{n} \qquad (3\text{-}4)$$

As might be expected, a comparison of utilization between the first organization (block depth = 1) and the second (block depth = 2) reveals that for all applicable values of n, the horizontal usage of the block depth = 2 layout is less than that of the layout in which the block depth is 1. The relation between the vertical utilizations is the reverse.

Thus, the anticipated result for block-oriented layouts is that horizontal utilization decreases and vertical utilization increases with an increase in block depth. This is, in fact, the case as shown in Figure 20.

The objective for a placement in which channel crowding is to be minimized must be to hold both horizontal and vertical channel usage as low as possible. Any other criteria, such as minimization of the sum of the utilizations in both directions, is apt to minimize density in one direction at the sacrifice of the other. Since the total interconnection length can be approximated as a linear multiple of this utilization sum, it can be seen that the minimization of channel usage in both directions provides a

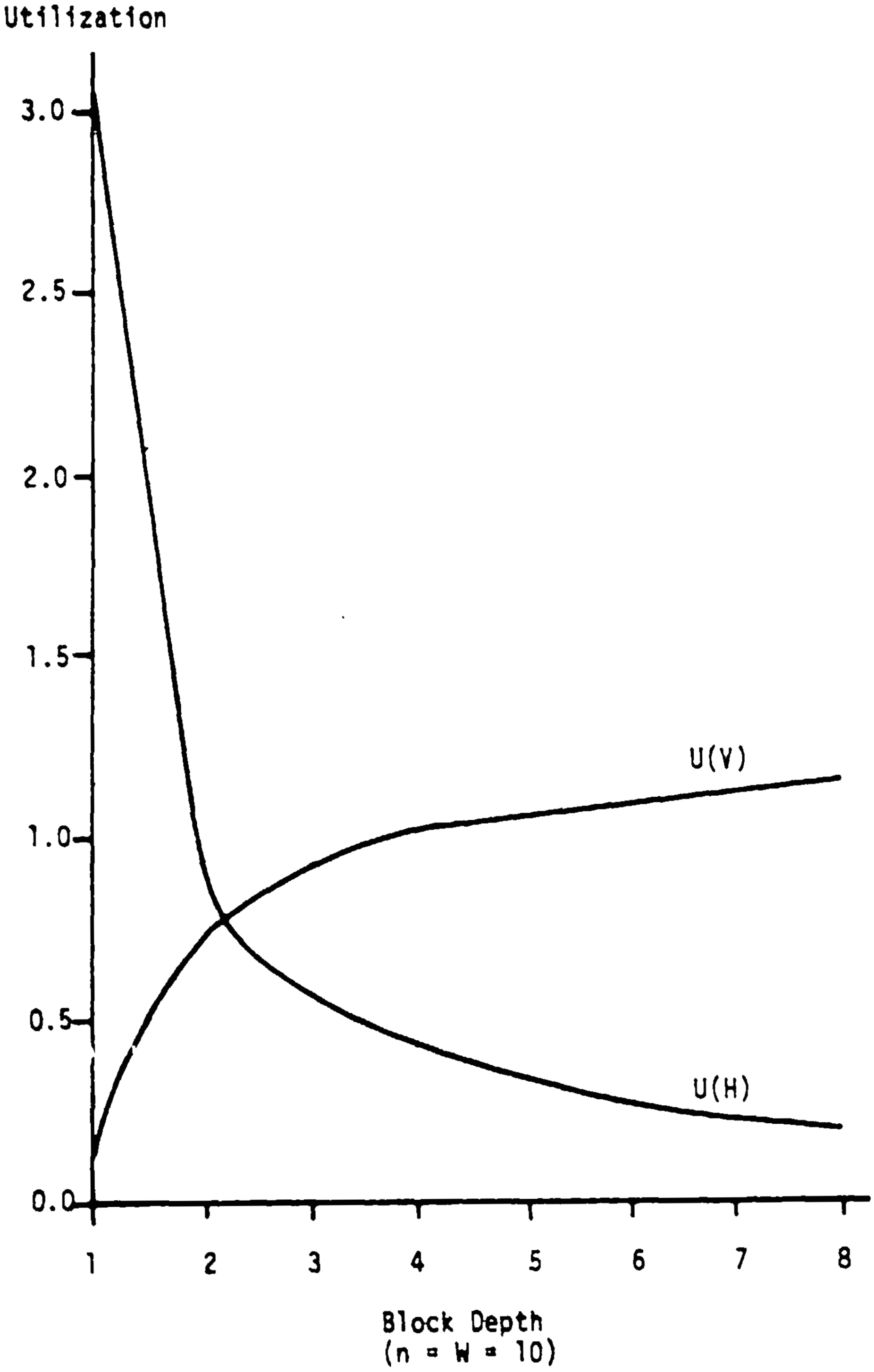


Figure 20. Relation Between Utilization and Block Depth

more powerful optimization criterion than this more common placement objective.

The problem of finding an optimal block-oriented folding of a linear order can thus be approached by identifying a block depth at which both horizontal and vertical channel usage are minimum. From Figure 20, it can be seen that this is an impossible objective since U(H) is minimized at high block depths and U(V) at low depths.

An approximate solution, then, can be obtained by selection of a block depth at which both U(H) and U(V) are as small as possible. A reasonable strategy would seem to be the selection of the block depth at which U(H) is most nearly equal to U(V) since, at this point, either increasing or decreasing the block depth must worsen the utilization in one direction. Unfortunately, a-priori location of this point is difficult due to the unproportional dependence of both U(H) and U(V) on cell width (W) and row length (n).

However, a method for solution can be suggested by noting the regularities present in the block-oriented structures and the resultant computational simplicity of folding. Since a layout for a single block depth can be generated quickly, and since the range of block depths is limited by the number of STAR rows, it is feasible to sweep the entire block depth range and to select the most optimal solution. This is the strategy used in the actual STAR placement routine and will be described in a later section.

Several qualifications for the methods of this section are required. First, it should be noted that the conceivable range of unique folding strategies is effectively unlimited. The use of the block-oriented techniques presented here has been based on performance comparisons with other folding methods and on the simplicity of the procedure. While other strategies may be simpler or produce more optimum solutions, the block-oriented methods have shown satisfactory performance and are in use in the current STAR folding routine.

A second portion of the methods requiring clarification regards modifications to the block-oriented technique which might be necessary for certain STAR sizes. For example, fewer than r rows might be available at the end of the STAR for placement of the last block in a procedure in which the block depth is r. The procedure can be easily modified to sense the case in which fewer than r (say, s) rows remain and to perform only s-block depth folding for the last block.

More serious size constraints occur in the horizontal direction. For the STAR models presented in this section, the row length in cells (n), has been implicitly assumed to be odd. An even n prevents the use of the normal folding strategy by eliminating the possibility for correct matching between the ends of the blocks. For sparsely-populated STARs, it may be possible to preserve the folding pattern by neglecting the final column of slots. For dense STARs, the full row width must be used and the block connection problems are ignored.

Finally, mention should be made of the possibility of a circuit specification which cannot be fitted onto the particular STAR requested. For the uniform cell case, this event can be simply detected prior to cell placement by comparison of the number of W-wide slots available with the number of cells.

Folding techniques applicable to uniform cell STAR placement have been presented in this section. The more general case, in which various cell sizes exist, will be discussed in the following section.

Folding for Non-Uniform-Width Cells

As was noted previously, the occurrence of a STAR application in which uniform cell sizes are requested is an extremely low probability event. The most common STAR applications are those in which cells occur in many different widths.

Several of the simplifying assumptions applied to the uniform cell problem cannot be justified for the non-uniform cell case. In particular, the modelling of the STAR as a slotted organization is not, in general, possible since this model depended on division of the STAR into regions equal in size to the uniform cell width. While modelling of slots which are larger than the largest cell might be satisfactory for a sparsely-populated, non-uniform problem, the STAR space wasted by this approach would preclude solution of dense problems. Thus, the true gridded organization of the STAR must be recognized for the non-uniform cell case.

However, the block-oriented folding techniques applied to the uniform cell problem can be adapted for use in the non-uniform case. The block-oriented technique is used to compute a *base row* for each cell to be placed. An *alternate row*, adjacent to the base row, is also specified. The alternate is selected to be the row which is in the direction of the current vertical placement trend within the block. The column locations to be occupied by a cell are selected as the left-most available positions of the row for left-to-right blocks (*odd blocks*) and the right-most available positions for right-to-left blocks (*even blocks*). A STAR placement formed in this manner is shown in Figure 21.

As can be seen from this figure, it is possible that there is insufficient space on the desired row for placement of a cell (note cell 12). In this eventuality, placement is attempted on the alternate row. If this fails, the next base row is selected.

The folding procedure must allow for the possibility of being unable to completely place the cells in the linear order. This situation can arise in two ways. First, it may

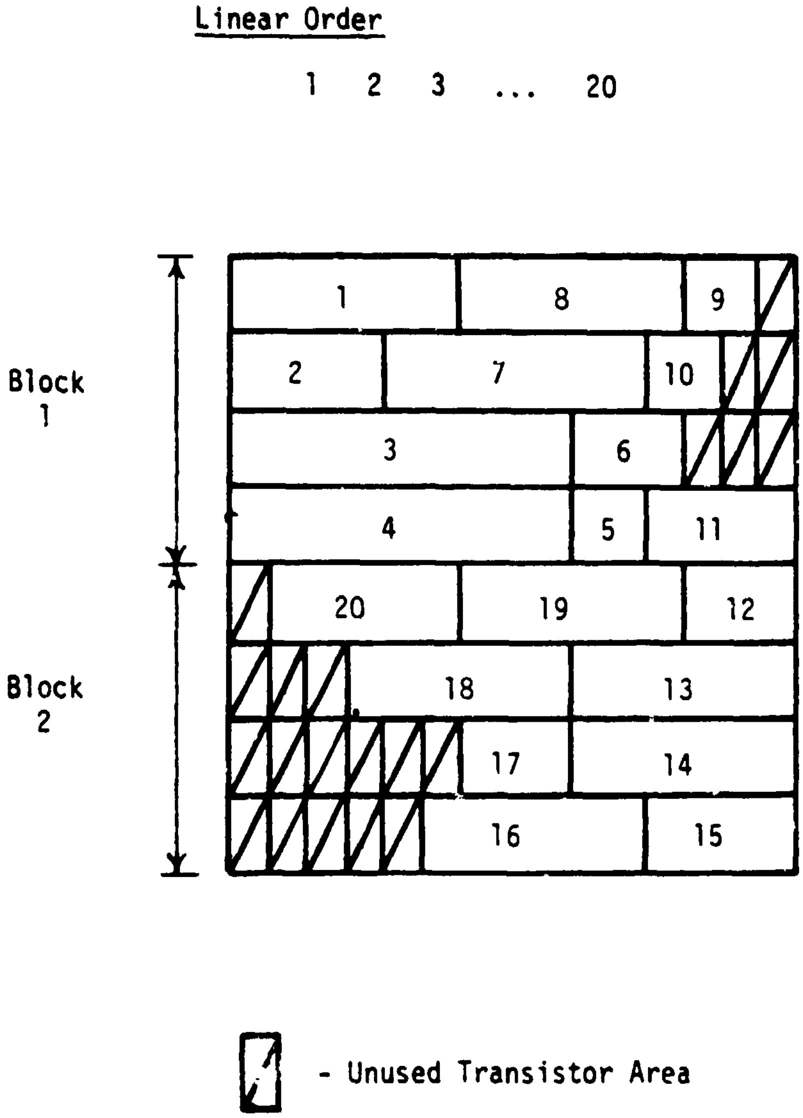


Figure 21. Non-Uniform Cell Placement

be impossible to fit the complete cell set on the STAR selected. Three ways in which this may occur are noted in Figure 22.

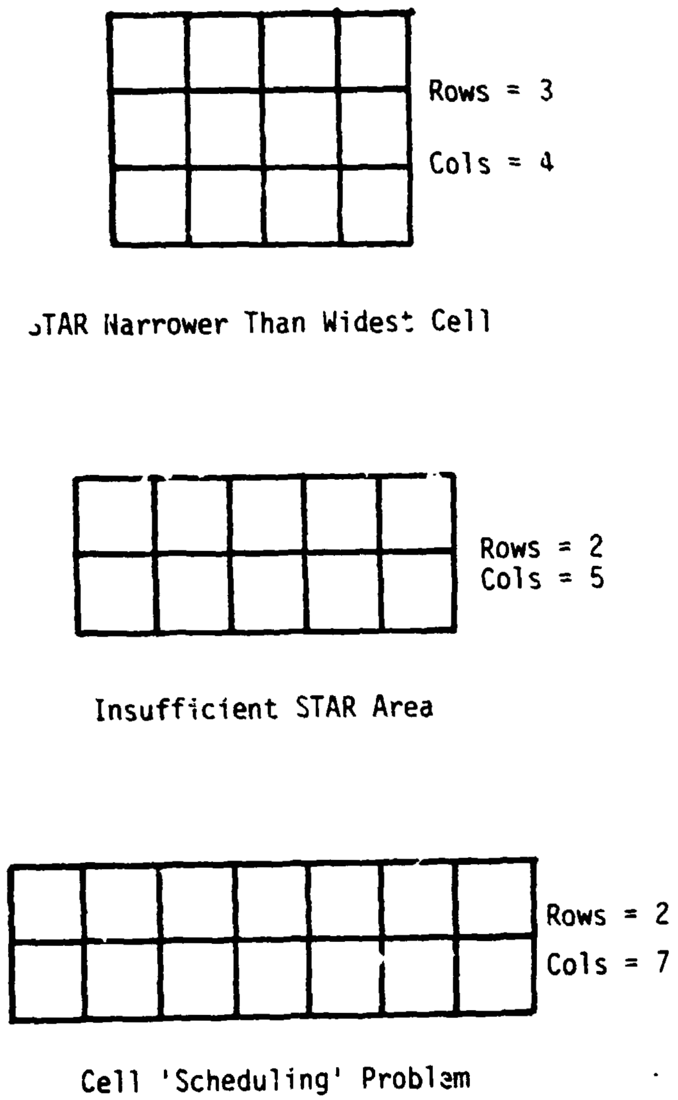
The first two cases shown can be simply isolated by comparison of the cell widths and the STAR size. In the third case shown, none of the possible cell-to-row assignments can result in a complete placement.

This third case bears great similarity to the classical bin packing problem (a special case of the scheduling problem) in which a number of finite-length tasks are to be assigned among several workers and it is desired to determine if all the tasks can be completed in a given amount of time. As noted by Graham [33], the only techniques available for complete solution of this problem involve exhaustive investigation of all possibilities. A-priori knowledge of a cell "scheduling" problem, then, can only be obtained by use of procedures which are on the same order of difficulty as the complete placement problem. No pre-folding tests for the existence of this problem are performed. The assumption is that this is the cause of the problem if the complete folding procedures (to be shown) fail to identify a solution.

A second way in which failure of folding of a linear order may arise is shown in Figure 23. Two linear orders for a given set of cells are shown in this figure. The first linear order cannot be folded at any block depth (BD).

Cell Set

1(width 5), 2(width 4), 3(width 4)



ͻTAR Narrower Than Widest Cell

Insufficient STAR Area

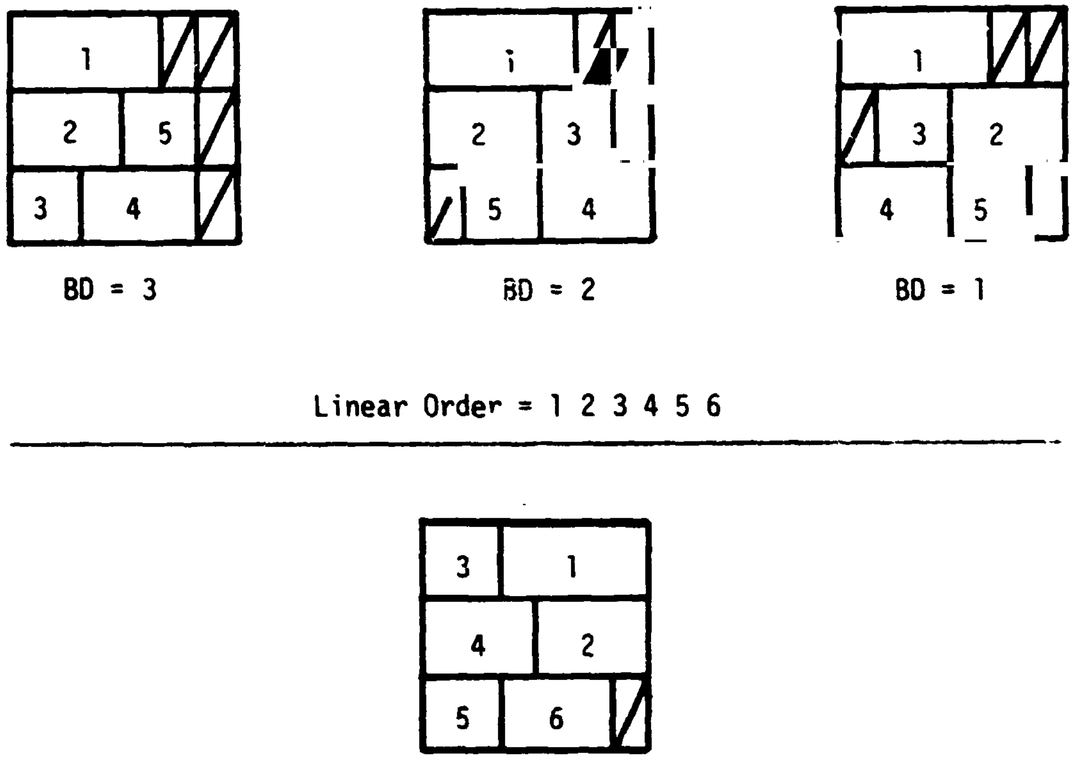
Cell 'Scheduling' Problem

Figure 22. STARs on Which a Given Cell Set Cannot be Placed

Cell Set

1(width 4), 2(width 3), 3(width 2)
4(width 2), 5(width 2), 6(width 3)

Rows = 3, Cols = 6



BD = 3 BD = 2 BD = 1

Linear Order = 1 2 3 4 5 6

BD = 3

Linear Order 3 4 5 6 1 2

Figure 23. Dependence of 'Foldability' on Linear Order

However, the second order is easily folded. Thus, linear order has a non-negligible effect on "foldability".

To take advantage of this fact, the procedures developed f.r the STAP cell folding problem ucilize limited modification to the specified linear order in the event that folding for a particular order cannot be performed.

This modification to the linear order is called rotation and consists of splitting the linear order at a boundary between two cells and reversing the order of the two parts formed. For example, the linear order

1 2 3 4 5

can be rotated about the boundary between cells 3 and 4 to form the new order

4 5 1 2 3 .

Rotation of a linear order has the effect of presenting a different sequence of cell widths to the folding routine and is performed in the hope that the modified order can be folded to fit the STAR.

A rotation disrupts all interconnec.ions which crossed the rotation boundary in the original order. To keep this disruption to a minimum, the cell boundaries to be used as rotation boundaries are selected in their reverse order of connectic. strength. In other words, the first rotation of a linear order is performed at the boundary which is crossed by the fewest connections. Succeeding rotations of the original linear order are performed at boundaries with more connection crossings. Thus, a STAR placement formed by folding an early rotation of the linear order should contain relatively few disturbed connections.

A second operation which improves the probability of complete circuit placement is based on the characteristics of the block-oriented procedure when used with non-uniform cells. The nature of the procedure is such that "holes" or unfilled portions of the STAR, may remain after processing of a row has been completed. A cell occurring later in the linear order may be small enough to fit the "hole" and, if placed there, will relieve crowding conditions on the remainder of t'e chip. Thus, in the event of folding failure with block depth modification and rotation, a lookback operation is performed. This operation scans a limited number of preceding rows for "holes" large enough to contain the cell to be placed. If any are found, the cell is located there. If not, the normal base or alternate row is used for placement.

The general flow of the non-uniform folding technique is shown in Figure 24. Data concerning the performance of this and other procedures shown here are presented in a later chapter. A unified placement system, which is based on the LOF techniques presented here is described in Chapter V.

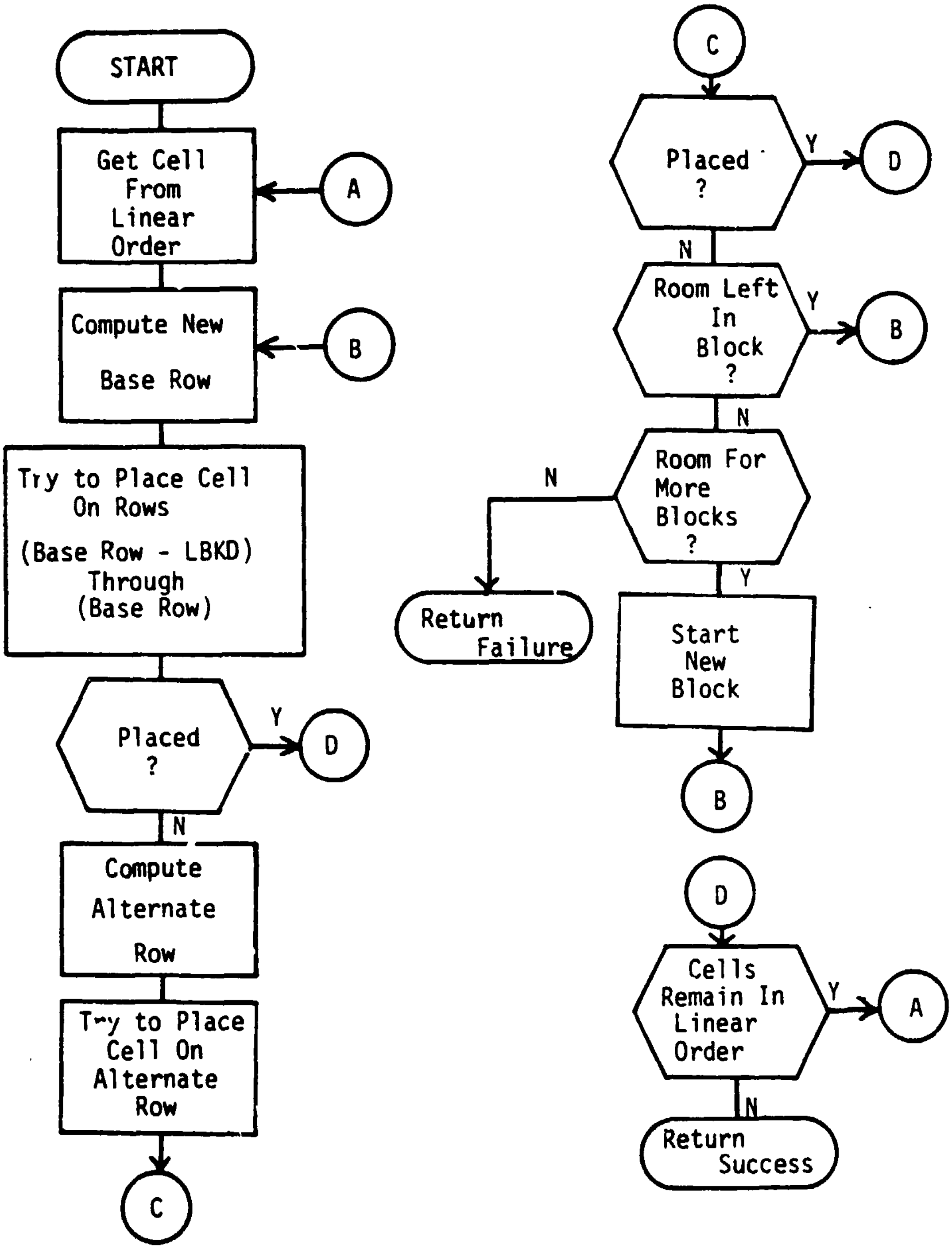


Figure 24. Functional Flow of Foldin] Procedure

IV. STAR PLACEMENT OPTIMALITY MEASUREMENT

In the preceding chapter, general methods for generating a linear order of STAR cells and for folding the order onto the STAR have been presented. The performance of the final placement system depends on these methods and on the exi۶tence of a fast medium for gauging STAR placement optimality (a placement rater) . The development of this optimality measurement technique is discussed in this chapter.

The following section contains a description of criteria concerning optimality measurement. In the next two sections, methods for measuring two optimality characteristics are discussed. The final section of this chapter describes a method by which nearness of a placement rating to the highest expected rating can be estimated.

Criteria for STAR Placement Optimality Measurement

The overall objective for the STAR placement system is to improve the ease with which a STAR placement can be routed. As is shown in the next chapter, the optimality of the placement produced by the STAR placement system depends, to a great extent, on the performance of the STAR placement rating routine.

This rating procedure should satisfy two basic criteria:

1. the procedure should form measures which are proportional to ease of placement routing, and
2. the procedure should operate in as little time as possible so that many repetitions of the technique will not produce an adverse effect on total system speed.

As stated in a previous chapter, the objective of placement to maximize routing ease is not easily expressed in terms of measurable phenomena. Translation of this objective into simple criteria is necessary prior to development of optimality measurement techniques.

For the STAR placement system, two measurable quantities that have been selected for use in placement rating are channel usage and the fraction of linear routing paths. In the following section, methods for estimation of STAR channel usage are presented. Estimation of the fraction of linear routes is discussed in a later section.

Channel Usage Prediction

The necessity for high-speed rating of STAR placements precludes use of many conceivable "exact" channel usage prediction techniques. The method presented here has been developed to allow computational speed. The method is also valuable since it can be used to provide estimates of both horizontal and vertical usage in all areas of the STAR. A third advantage of the method is that n-cell nets can be handled directly and need not be translated into cell-pair connection equivalents.

The estimation procedure can be easily visualized by consideration of the STAR structure. If, as shown in Figure 25, at least one c ll in net I lies to the left of column J and at least one cell to its right, then it is possible to state with certainty that at least one horizontal channel segment in column J must be used for the routing of net I. In a similar manner, it can be seen that at least one vertical channel in row K must by used for net I. Corresponding statements can be made for any of the rows or columns which intersect the rectangle A (the minimum routing boundary) in Figure 25.

In the actual routing of the placement, it is required for the net to extend beyond the boundaries of this rectangle in order to connect to the internal pins of the cells. Exactly how far the net extends is determined by the location of the net entry ranges of the cells at the extremities of .he rectangle. The cell pair shown in Figure 26 illustrates this point.

The best (minimum channel usage) interconnection path for these cells is the path obtained if the cells' pin structures are such that the nearest ends of the cells can be connected. The worst-case path is that required if the farthest ends of the cells must be connected. It is then

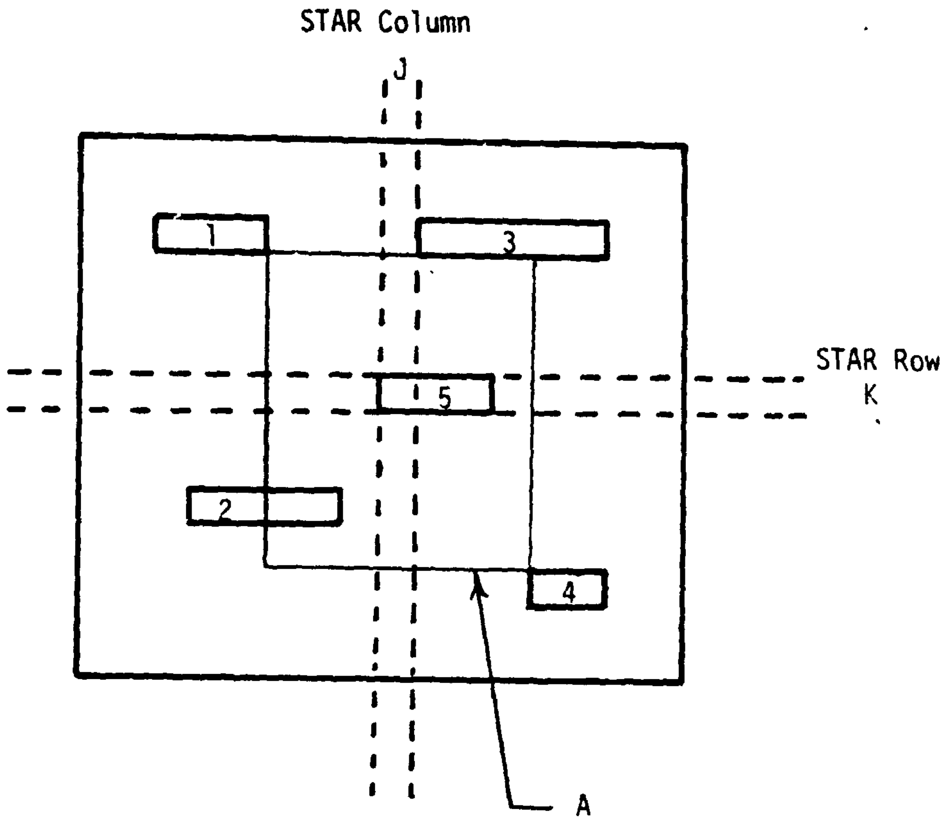


Figure 25. STAR Channel Usage Estimation Concepts

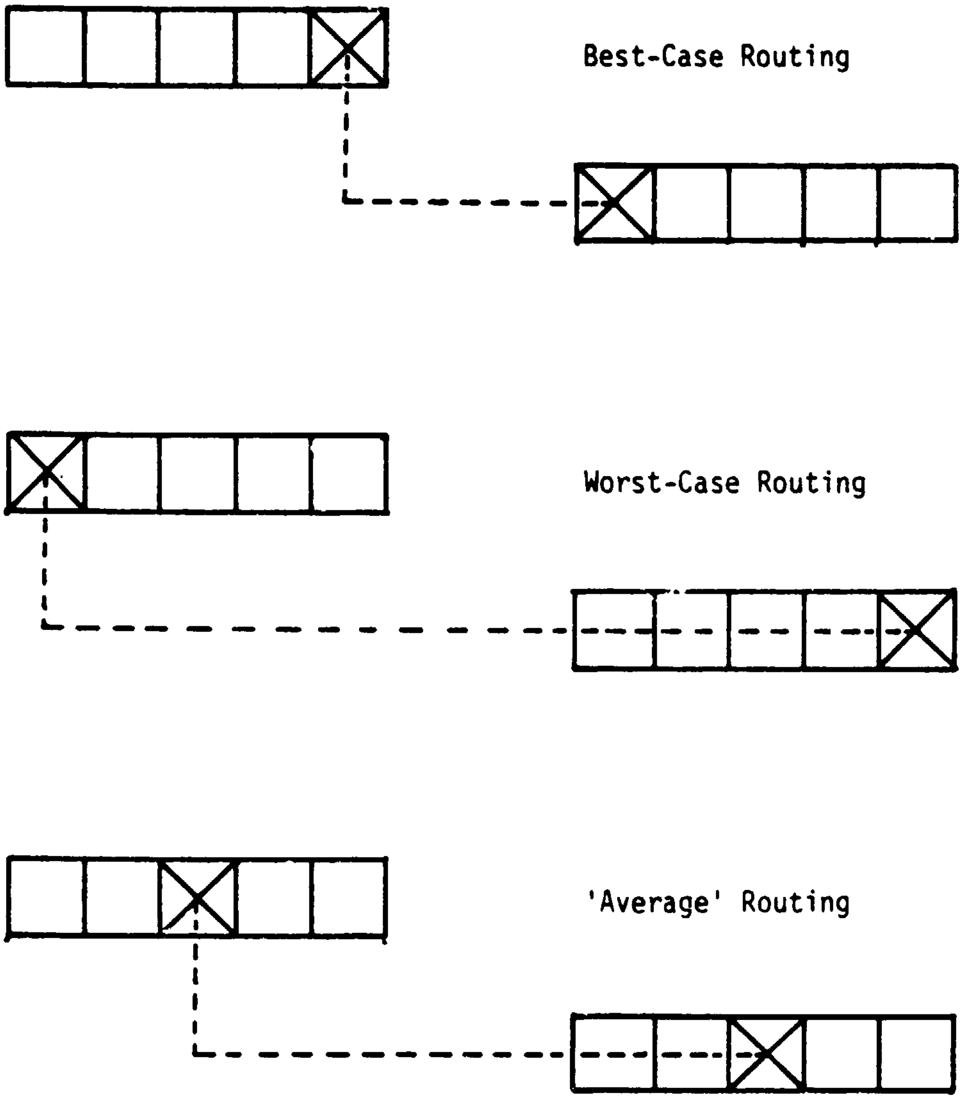


Figure 26. Dependence of Routing on Internal Cell Structure

possible to define the "average" routing of the connection to be such that the number of channel segments charged in each row or column of the STAR is exactly equal to the average of the charges for the best and worst-case routings in the corresponding columns.

The average charges for each net in the placement can be calculated and summed for each STAR row and column. The result of this summation is an estimate of the number of channel segments to be used in the routing of the placement in each STAR row and column. Since the number of channel segments available in a row or column is fixed, the results are easily presented as the expected fraction of the segments to be filled in each row or column (i.e., the channel utilization).

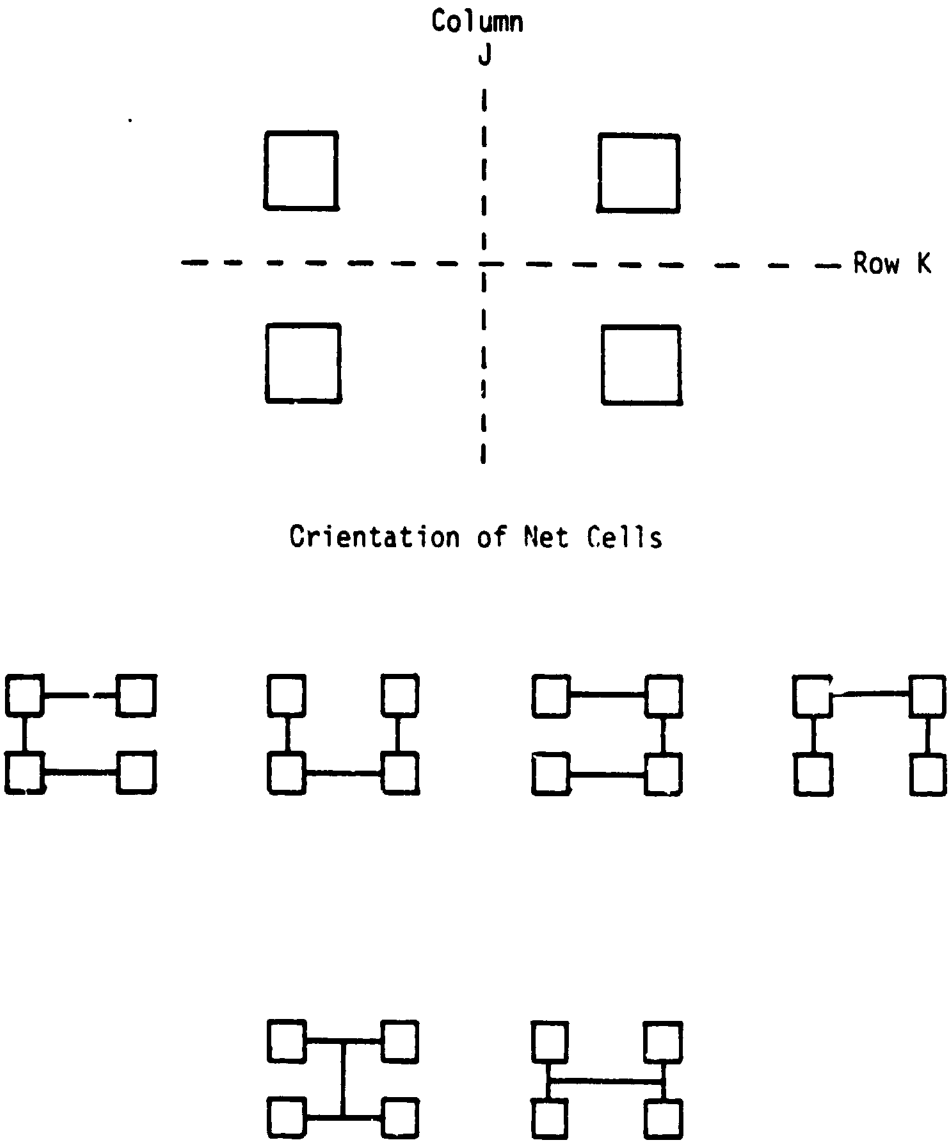
It should be noted that, in the calculation of horizontal charges, the four horizontal routing channels available on each row are considered to be equivalent and equidistant from any other row. Thus, while the best and worst-case charges for the horizontal direction are different, they are the same for the vertical direction. Averaging, then, need only be performed for the horizontal segments of each net.

It should also be noted that, regardless of net size, these simple usage calculations can be made based solely on the cells located at the extremities of the routing boundary. Thus, the assumption of a particular routing structure (such as the minimum spanning tree) is not necessary.

Two previously un-mentioned aspects of this procedure may adversely affect the accuracy of the calculations. First, the implicit assumption that all routing of a net will be conducted within the routing boundary does not accurately reflect the performance of modern routing techniques. While routing of a net within its boundary is a good model for routing under uncrowded circumstances, as channels within the boundary are filled, more and more channel segments outside the boundary must be utilized for the net path. However, the "easiest" routing for typical routers does reside within the routing boundary. Thus, the density calculations reflect the interconnection of the cells under the simplest conditions and can be used as indicators of routing ease.

A more serious limitation of the procedure is its failure to consider the characteristics of a net which must cross either a STAR row or column more than once. An example of this type of net is shown in Figure 27.

Since nets of this type must increase the channel usage in either the horizontal or vertical direction, the usage figures obtained by use of the procedure discussed should be recognized as optimistic views of the actual channel usage. However, results of statistical analysis of typical STAR application circuits indicate that neglecting these multiple



Orientation of Net Cells

Possible Routing Paths

Figure 27. Example of Net with Multiple Crossings of Row or Column

row or column crossings may not produce excessive differences between the calculated and actual usage figures. These statistics, shown in Chapter VI, indicate that average nets contain between two and three cells and, because the minimum possible net size is two, indicate that the majority of nets are simple connections. Since, regardless of relative cell position, it is always possible to route between two cells without use of multiple crossings of a single column or row, it can be seen that the channel usage of the majority of circuit nets is correctly estimated.

The procedure used for calculation of the channel usage estimates will now be given. In this procedure, BH(I) is the best-case number of horizontal channel segments used in column I, WH(I) the worst-case number of horizontal segments used in the column and V(J) the number of vertical segments used in row J. WIDTH(J) is the width in transistors of cell J. The two figures, U(H) and U(V), represent the global horizontal and vertical channel usage as described in a preceding chapter.

The usage estimation procedure is

1. Form vectors S and R such that

   S(J)=STAR column containing the left-most transistor in cell J, and

   R(J)=STAR row containing cell J

   for all cells J in the placement.

2. For each net, I, in the circuit and for each cell, J,
in net I, let
LEFT(I)=MIN(S(J))
RIGHT(I)=MAX(S(J))
TOP(I)=MIN(R(J))
BOTT(I)=MAX(R(J))
S1(I)=MIN(S(J)+WIDTH(J)-1)
S2(I)=MAX(S(J)+WIDTH(J)-1)

3. BH(K)=the number of nets I such that
[(S1(I) $\le$ K) AND (RIGHT(I) > K)]
OR [(S1(I) < K) AND (RIGHT(I) $\ge$ K)]

4. WH(K)=the number of nets I such that
[(LEFT(I) $\le$ K) AND (S2(I) $\ge$ K)]

5. V(L)=the number of nets I such that
[(TOP(I) $\le$ L) AND (BOTT(I) > L)]
OR [(TOP(I) < L) AND (BOTT(I) $\ge$ L)]

6.
$$U(H) = \frac{1}{8ROWS \cdot COLS} \left[\sum_{K=1}^{COLS} (BH(K) + WH(K))\right]$$

$$U(V) = \frac{1}{ROWS \cdot COLS} \left[\sum_{L=1}^{ROWS} V(L) \right]$$

c-2

Linear Routing Prediction

The maximization of the fraction of STAR cell nets which can be routed without bends was selected as a placement objective both to increase routing ease (by providing simple routing paths) and to minimize the number of vias required for routing.

Since, like exact channel usage prediction, exact linear routing prediction requires knowledge of internal cell pin structure, approximate methods for counting linear nets have been selected.

The quantity which is actually measured is the number of potentially linear nets (i.e., the number of nets which can be routed linearly if internal cell structures permit). This can be easily calculated as the sum of the number of nets in which all cells reside on the same row and the number of nets in which all cells share at least one STAR transistor column.

These figures are available as by-products of the channel usage estimation procedure described previously. If a net, I, is linear in the horizontal direction, then TOP(I) = BOTT(I). If the net is linear vertically, then S1(I) $\geq$ RIGHT(I). The fraction of the total number of circuit nets which are potentially linear (FSN) can be used for an indication of optimality with respect to linear routing.

Placement Quality Measurement

While measurement of various placement characteristics, such as channel usage or fraction of linear nets, can be used as indices for the comparison of two placements of a circuit, neither give an estimate of the nearness of a placement to the optimum.

If a large number of random placements are formed, each is rated, and the rating results are plotted in histogram format, then a normal curve is approximated. Standard probabalistic techniques can then be used to estimate the fraction of all possible placement which would have ratings lower than a particular placement (the placement quality) . The assumption is that the closer this fraction approaches to one, the more optimum is the placement.

It should be emphasized that this procedure for estimating placement quality is based on no theoretical study of the placement problem, but rather on the nature of observable results. Quality is not used as a driving force for the STAR cell placement routines, but is presented to the user as additional output data. Further discussion of placement quality is deferred to a later chapter.

V. THE CELL ARRANGEMENT PROGRAM FOR STAR (CAPSTAR)

General techniques for the solution of the STAR cell placement problem and methods for placement rating have been discussed in the preceding chapters. The incorporation of these techniques into a FORTRAN placement program (CAPSTAR) for use in an application environment will be discussed in this chapter.

CAPSTAR was developed to act as an integral part of a system of programs which solve the physical design problem for logic circuits to be implemented by use of STAR technology. As such, several portions of the program are involved with the formating of input and output data necessary for communication with other programs. These portions of CAPSTAR will not be discussed in this chapter.

The features of CAPSTAR which will be presented here are those which deal with the previously-discussed placement and rating techniques and other functions necessary for high-speed identification of ncar-optimum STAR placements. The following sections contain descriptions of high-level program organization, database organization, LOF procedure implementation, placement improvement techniques, placement rating procedures, and a method for placement of circuit pads.

A user's guide for the program described here is given in [35]. The source listing of the program is shown in [34].

Program Organization

The high-level organization of CAPSTAR is illustrated in Figure 28. As shown in this figure, following the performance of the clustering and decomposition portions of the LOF technique, the folding procedure is used repetitively to generate a number of different placements. After a user-entered number of solutions (MAXSOL) has been formed, the best (highest-rated) IMPROVE of the placements are selected and are improved by means of a simple PI routine. The best improved placement is selected as the problem solution.

As previously noted, the folding step solutions are formed using various block depths, rotations, and lookback distances. The procedure is designed so that the earliest-formed placements are usually the highest rated placements that can be formed by folding. Due to the simplicity of the folding procedure, the time requirements for a large number of repetitions is usually not excessive. The variable MAXSOL, then, is typically picked to be a large number to allow the best IMPROVE to be selected from a relatively large sampling of placements.

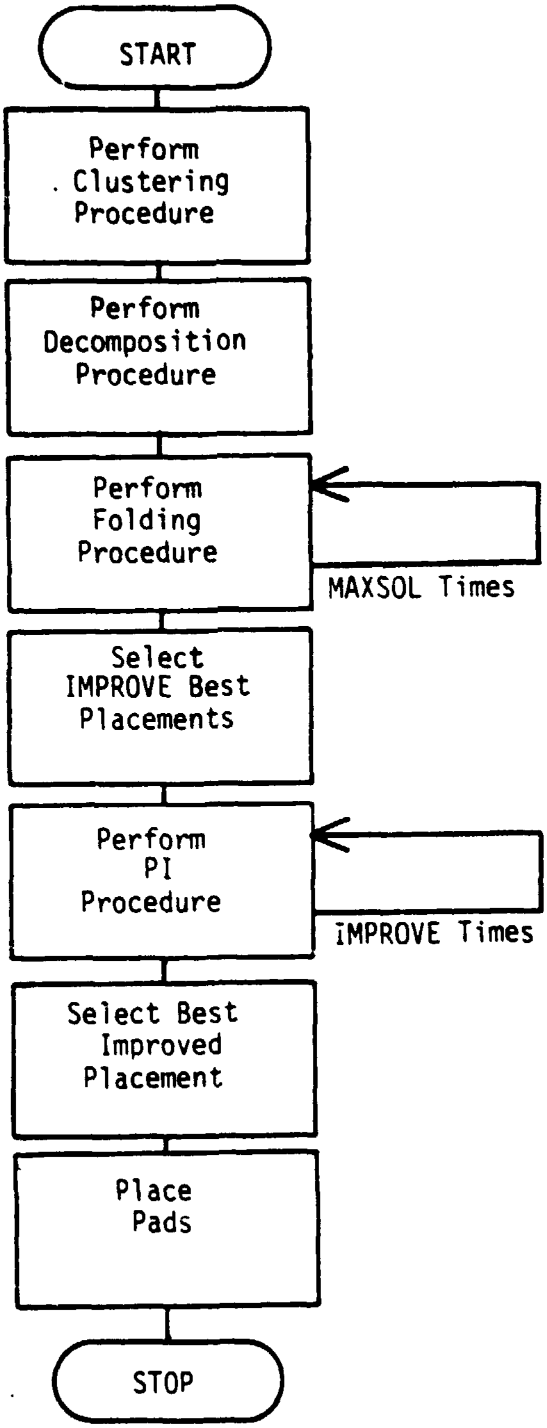

START
Perform Clustering Procedure
Perform Decomposition Procedure
Perform Folding Procedure
MAXSOL Times
Select IMPROVE Best Placements
Perform PI Procedure
IMPROVE Times
Select Best Improved Placement
Place Pads
STOP

Figure 28. CAPSTAR High-Level Flow

A placement improvement routine is then performed on the set of IMPROVE best folded solutions. This routine, which is based on the neighborhood pair-wise interchange technique, is quite simple conceptually but tends to execute rather slowly. The variable IMPROVE, then, is usually selected to be a number in the range, 3 to 5.

It is possible to set IMPROVE to 1 to reduce execution time. However, experimental CAPSTAR runs have indicated that the best folding solution is often not the best solution after placement and sub-nominal placements may result by reduction of IMPROVE from the range noted above.

Following improvement, the highest-rated placement is selected and pad placement is performed. The pad location process is deferred to this point in the program so that this relatively slow procedure need only be performed on one placement.

After the pads are located, the appropriate output files are constructed and results are presented to the user. The program then terminates and control is passed to the successor program, a STAR placement router.

A discussion of the high-level aspects of CAPSTAR has been presented in this section. In the following sections, a more detailed description of the CAPSTAR segments will be given.

Database Organization and Storage

CAPSTAR has been designed for execution on computing systems with limited available program and data storage areas. Thus, the minimization of data storage requirements is a high-priority objective of the program.

In general, the data storage requirement for any placement program is dependent on the maximum circuit size intended for use. For the STAR placement problem, these restrictions have been established so that the largest circuit which can be handled by CAPSTAR is one consisting of 1000 elements and 500 nets.

If the data describing the interconnection structure of a circuit of this size is stored using the connection matrix (C), one million entries must be saved. If each entry is 2 bytes (16-bit integer format) in length, almost 2M (1M = 2^{20}) bytes of storage is required for this array, alone. Storage of only the entries above the diagonal of this (symmetric) matrix would require 1M bytes, which is still in excess of the total s ›rage available in many small computers.

The C-matrix, then, has not been utilized in CAPSTAR. Circuit interconnection data is stored in a vector (NTC) which is a list of the elements in each net with single-entry delimiters between nets. The required length of this vector is a function of the maximum number of nets and the maximum average net size. Selecting an upper bound of 5 elements on average net size, the required length of this vector is 3000 entries, or less than 6K (1K = 2^{10}) bytes of storage at 2 bytes per entry.

This approach has the added advantage of removing the requirement for modelling nets as connections between cell pairs (as is necessary for the C-matrix organization). Thus, the processing time associated with modelling of the input network is not required.

While the placement problem for STAR can be solved by use of the net-to-cell mapping specified by the NTC vector, an alternate organization of the interconnection data is more facile for some portions of CAPSTAR. These portions (primarily, the clustering and decomposition portions of the LOF process) can be more simply structured if a cell-to-net specification of the circuit is available. A second vector (CTN) is provided for use by these segments. This vector is, basically, an ordered list of elements in the network which specifies, for each element, the nets incident to the element.

The length of the CTN vector is the product of the maximum number of elements and the average number of nets per element. If the maximum average net size is 5 elements, a 500-net circuit contains no more than 2500 pins. The average number of nets per element in a 1000 element circuit, then, is 2.5 . The length of the CTN vector can thus safely be set at 3000 entries.

For convenience in the decomposition segment, the clusters formed in the clustering step are treated as elements and are also entered in the CTN array. The actual working dimension of this array, then, is roughly twice the 3000-entry figure stated, or approximately 12K bytes of storage at 2 bytes per entry.

Speed of access to the data in the NTC and CTN vectors can be improved by supplying lists of pointers. For example, if the data for net I begins at location J in the NTC vector, the NTC pointer entry at location I would contain J. While improvement of overall processing speed might occur if this structure was maintained in all segments of CAPSTAR, the current version of the program uses pointer vectors only in the clustering step.

Most other data structures used in the system, such as those containing the cluster formation history and cell width data, are small in comparison to the NTC and CTN vectors. However, the data structures which specify a STAR placement can be larger and will now be described.

The gridded organization of the STAR leads to a matrix model for use in STAR cell placement. The size of this matrix is fixed by the number of rows and columns available in the largest STAR. At the time of this writing, the largest STAR is one consisting of 28 rows and 94 columns. The working dimensions of the STAR model matrix (chip) are thus set at 30 rows and 100 columns, requiring 3000 entries.

The storage required for this array is less than 6K bytes at 2 bytes per entry.

An alternate form of the placement, specifying the row and column position of each element in the circuit is also constructed. Since the maximum element count is 1000, this array contains 2000 entries. The total storage per placement is thus 5000 entries, or, approximately 10K bytes for 2 byte entries.

While this storage requirement may not be excessive for a single placement, the CAPSTAR structure requires the storage of no fewer than IMPROVE placements. If the maximum IMPROVE is selected to be 10, almost 100K bytes of storage are required. Even if only the smaller alternate version of each placement is retained, the storage required is almost 40K bytes.

To alleviate this problem, CAPSTAR maintains these intermediate placements in a disk file rather than in main memory. The storage required is thus reduced to that for a single placement. A time penalty is incurred, however, due to the increased number of disk accesses required.

A final consideration regarding CAPSTAR storage requirements should be noted. The program has been logically separated into functions so that physical separation of program parts can be facilitated. This may be useful in the event that the entire program requires an excessive amount of storage space.

In the current version of CAPSTAR, a physical division has been made between the decomposition segment and the folding segment (see Figure 28). This division was found to achieve a significant reduction in required storage over the case in which the complete program was executed as a unit.

Implementation of LOF Procedures

The clustering and decomposition segments of CAPSTAR are organized as described in Chapter III. The result of these procedures is a linear order which is to be folded onto the STAR. The CAPSTAR implementation of this folding process will now be described.

A fold cycle is defined to be the set of operations required to either successfully fold a line.r order onto the STAR or to determine that possibilities for fold structure modification (block depth modification, rotation, etc.) have been exhausted. The logical organization of a fold cycle is shown in Figure 29. In this figure, FOLD (n,m,p) is defined as the folding operation performed at a block depth of n using linear order m with a lookback distance of p.

On the first fold cycle performed, control is transferred to the primary entry point shown in Figure 29. If a placement is found in the cycle, it is rated and, if it is among the IMPROVE best solutions so far identified, is saved for further use. Control is returned to the fold cycle by use of the alternate entry point. Fold cycles are

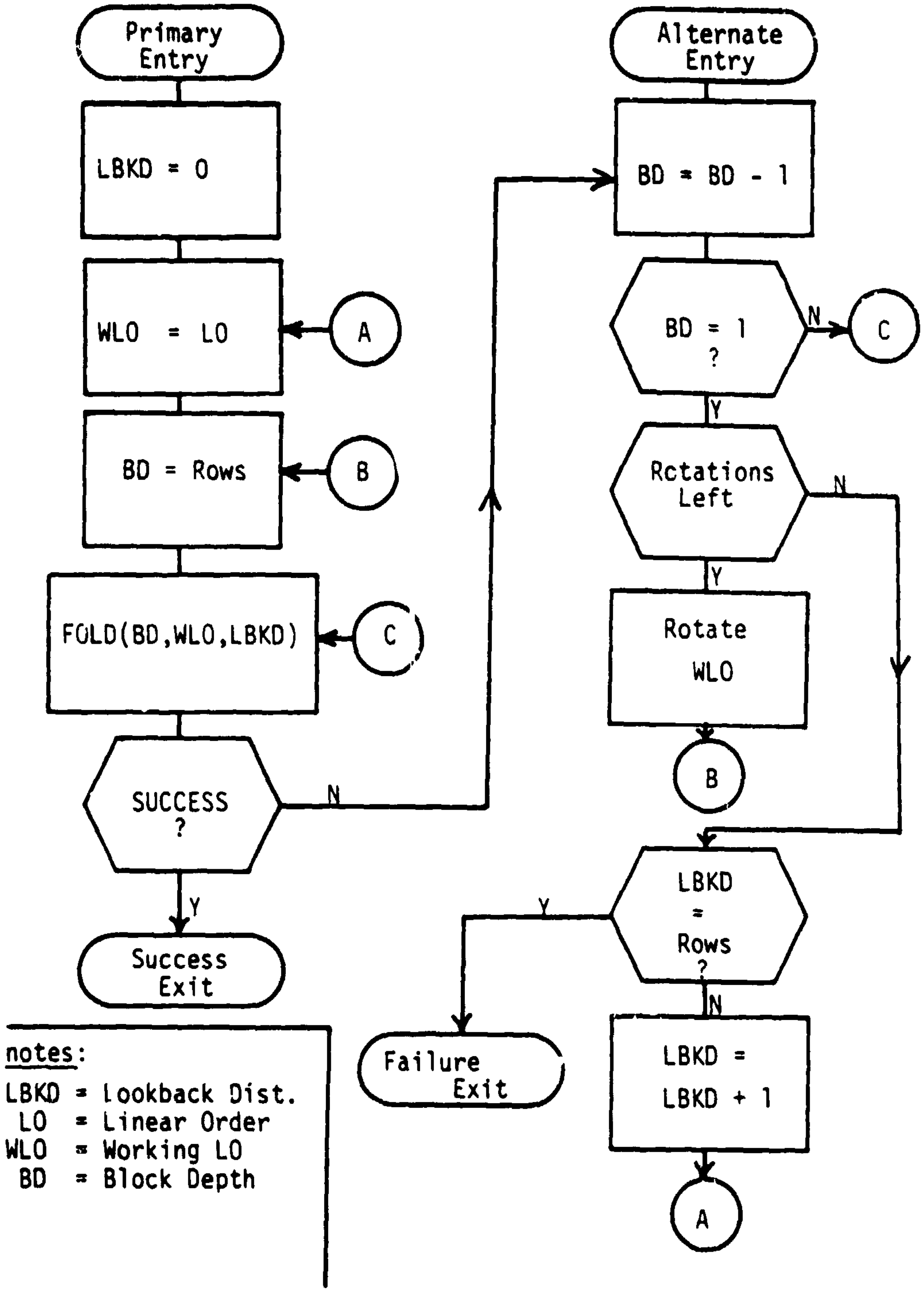


Figure 29. Functional Flow of a Fold Cycle

repeated until either failure is noted or MAXSOL solutions are found.

The result of the folding step is the set of IMPROVE best placements generated. These placements are passed to the succeeding CAPSTAR segment which iteratively improves them with respect to rating. This PI segment is described in the following section.

Placement Improvement

The use of simple placement improvement routines in the CAPSTAR system is based on two considerations:

1. while excellent placements can be obtained by use of the repetitive folding procedure, it is almost always possible to identify some way in which they might be slightly improved, and
2. the use of a PI routine which is driven directly by the CAPSTAR rating procedure can improve the placements in ways not easily obtained by the folding strategies.

The PI techniques utilized in CAPSTAR are based on simple neighborhood PWI concepts. The routine consists of two segments. In the first of these, each row in the placement is trial interchanged with the rows that are within a given row-neighborhood (RN) of it. After each trial interchange, the placement is rated by use of the CAPSTAR placement rating facility. A trial interchange is accepted if the resulting placement has a higher rating.

In the second segment, each cell in the placement is trial interchanged with the cells on its row which are within a given cell-neighborhood (CN) of it. As in the row interchange segment, a trial interchange is accepted only if the rating of the overall placement is improved.

It should be noted that the set of cells occupying a single row in the original placement also occupy a single row in the final placement. The use of general NPWI routines, in which cells can be interchanged between rows, has been avoided in CAPSTAR. For a non-uniform width structure like STAR, it may be impossible to interchange a pair of cells between rows since the space on a row left by the removal of one cell may not be enough for the placement of the other cell. Thus, for true NPWI routines, processing is required before each attempted interchange to determine if cell sizes permit swapping.

In the simplified NPWI procedures used in CAPSTAR, no such processing is required since each row must fit the space occupied by any other and since a re-ordering of the cells within a row does not affect row length. The interchange iterations, then, proceed more quickly than in a true NPWI procedure. As will be shown in a later chapter, overall placement optimality is not sacrificed by use of this simplified procedure.

Placement Rating

As can be seen from the preceding discussion, the CAPSTAR placement rating routine is critical to the derivation of "good" placements. Much effort, then, has been devoted to the development of this segment.

Four factors relating to desired characteristics of CAPSTAR placements are measured by this routine. These are:

1. horizontal channel usage,
2. vertical channel usage,
3. fraction of potentially linear routes, and
4. distance of unused transistors from the horizontal STAR center.

Factors 1 through 3 in this set have been previously discussed and U(H), U(V) and FSN are calculated as described.

The fourth factor relates to the observation that the highest channel densities in a typical STAR placement occur toward the horizontal center of the array. Since transistors which are not included in any cell do not require any internal connections, the paths normally used for internal cell connection can (with care so as to assure electrical isolation) be allocated to global interconnections. If the unused transistors, then, are assigned positions near the STAR center, the number of effective global channel segments is increased in the densest area and increased routing ease should result.

The process for measuring this factor involves summing the horizontal distance to the STAR center from all unused transistors to obtain the number ETD. A normalized distance figure, ETR, is then computed by

$$ETR = \frac{ETD}{ETDWC}$$

where ETDWC is the ETD which would be noted if the unused transistors were evenly distributed over the rows and packed at the row ends (i.e., ETDWC is the worst-case ETD). As shown in Appendix A,

$$ETDWC = \frac{MT}{2} \left[COLS - 1 - \frac{MT}{2ROWS} \right] \qquad (6\text{-}1)$$

where MT is the number of unused transistors in the placement. The normalized distance, ETR lies between 0 and 1 with higher values signifying less-desirable unused transistor placements.

Each of the four factors can thus be easily measured. However, comparison of placements on the basis of four different measurements is not straightforward. To provide the capability of simple placement comparison, the measurements should be combined into one overall placement rating.

After consideration of many techniques for measurement combination, one of the simplest conceivable methods was selected. Each of the measures is translated into a fraction from 0 to 1 with undesired qualities producing higher fractions. For each measure, a weighting factor is calculated which indicates the importance of the measure to placement optimality. The weighting factors are then multiplied by the appropriate fractions and summed together. This sum is normalized (0 to 1) by dividing by the sum of the weighting factors. Subtracting this result from one produces the placement rating.

At first consideration, for the STAR placement rating problem, U(H) and U(V) are equally important to placement optimality and should be assigned the same weighting factors. However, as will be recalled from a previous chapter, the objective is not to minimize the sum of U(H) and U(V), but to assure that both are as small as possible.

The scheme adapted for this measurement combination techn‸que is to define two variables, UW and UB, where

$$UW = MAX(U(H),U(V))$$

and

$$UB = MIN(U(H),U(V)).$$

By as‸‸gning a high weighting factor to UW and a lower one t‸ UB, CAPSTAR can be forced to always strive to minimize the worst measure.

The rating for STAR placements (PR) is thus obtained as

$$PR = 1 - \frac{(UWWF)(UW)+(UBWF)(UB)+(FSNWF)(1-FSN)+(ETRWF)(ETR)}{UWWF+UBWF+FSNWF+ETRWF}$$

where XXWF is the weighting factor associated with the measure XX.

This rating combination procedure lends itself to future addition of other rating criteria and re-evaluation of th‹ relative importance of the factors. For present purposes, the weighting factors are set at (UWWF = 6, UBWF = 2, FSNWF = 1, and ETRWF = 1).

Pad Placement

Once the highest-rated STAR placement has been identified, the pads specified in the circuit description are located on the chip periphery. For each STAR size, the possible pad locations are pre-specified in a disk file. The pad placement problem, then, consists of assigning each of the circuit pads to one of the pad locations.

The assumption made is that no two pads are directly connected so that pad placement can be performed by use of a simple linear assignment procedure. The procedure assigns, to each pad, an optimum pad location based on nearness to cells directly connected to the pad. The most optimum assignment over all pads is then selected and the pad is placed at the specified location.

The procedure continues by selecting the most optimum assignment and placing the pad until all pads are placed. If the assigned location for a pad is occupied, a new optimum location is selected from those not filled and the procedure selects the most optimum from the new set of assignments.

Experimental data indicates that this procedure achieves near-optimum pad placements with respect to the cell layout. Due to the simplicity and standard nature of this technique, pad placement will not be included in the discussion of CAPSTAR performance in the following chapter.

VI. PERFORMANCE OF PROCEDURES

The organization of a placement program for use with the STAR processing technology has been described in the preceding chapter. The performance of this program (CAPSTAR) is discussed in this chapter.

There are two primary objectives to this performance analysis. First, it is desired to indicate that CAPSTAR can form near-optimum cell placements in a computationally feasible amount of time and that the time requirements and placement optimality can be influenced by certain program variables.

Second, the validity of the CAPSTAR approach (and, by inference, the LOF procedure) is to be tested by its comparison to a more common placement technique.

The first portion of this chapter will deal with CAPSTAR performance characteristics, alone. Several test circuits will be identified and results of CAPSTAR execution with various input parameter settings will be shown. In the latter part of the chapter, results of comparisons between CAPSTAR performance and that of the pair-wise interchange technique will be given.

CAPSTAR Performance

The operation of CAPSTAR has been verified by use of 6 test circuits, TC1 through TC6. These circuits represent actual digital logic applications and have been selected as typical of STAR application circuits. Data describing the test circuits is shown in Table 1. The CAPSTAR test runs for these circuits were performed on an IBM 370/158 and program compilation was performed by means of the IBM FORTRAN-IV Level G compiler.

Table 1

Test Circuit Parameters

CIRCUIT	CELLS	NETS	AVG WIDTH	AVG PINS/CELL	AVG CELLS/NET
TC1	61	88	7.6	3.62	2.60
TC2	20	25	8.4	3.15	2.52
TC3	96	104	6.9	3.05	2.97
TC4	24	35	7.4	3.21	2.60
TC5	20	25	8.0	3.15	2.52
TC6	305	440	7.6	3.62	2.60

Six CAPSTAR input variables were modified during testing to study the effects of different values. These variables are MAXSOL, IMPROVE, RN, CN, ROWS and COLS, the functions of which have been described in previous sections.

Unless otherwise noted, the values for these variables are set at MAXSOL=200, RN=CN=2, and IMPROVE=8. The default values for (ROWS, COLS) are (8,24) for circuits TC2, TC4, and TC5, (16,48) for TC3, (20,25) for TC1, and (28,96) for TC6.

The results of the first test series are displayed in Figure 30. In this simple series, CAPSTAR was veıified to be capable of placement of all test circuits in acceptable time. In the figure shown, it is interesting to note the correspondence between required processing time and intuitive circuit complexity. The exact relationship between input circuit complexity and CAPSTAR performance has not been established.

A second type of result from the first test series is also illustrated in Figure 30. In this figure, the relative gain in placement rating that is provided by the placement improvement routine is illustrated. It should be noted at this point that no conclusions can properly be drawn from the comparıson of the ratings of different circuits. The only methods of contrasting the ratings of two circuits must, in some way, include the optimum ratings for placements of each circuit. Since these, in general, are unknown, comparison of ratings of different circuits should not be attempted.

The remainder of the test series to be discussed in this section deal with variance of CAPSTAR input parameters.

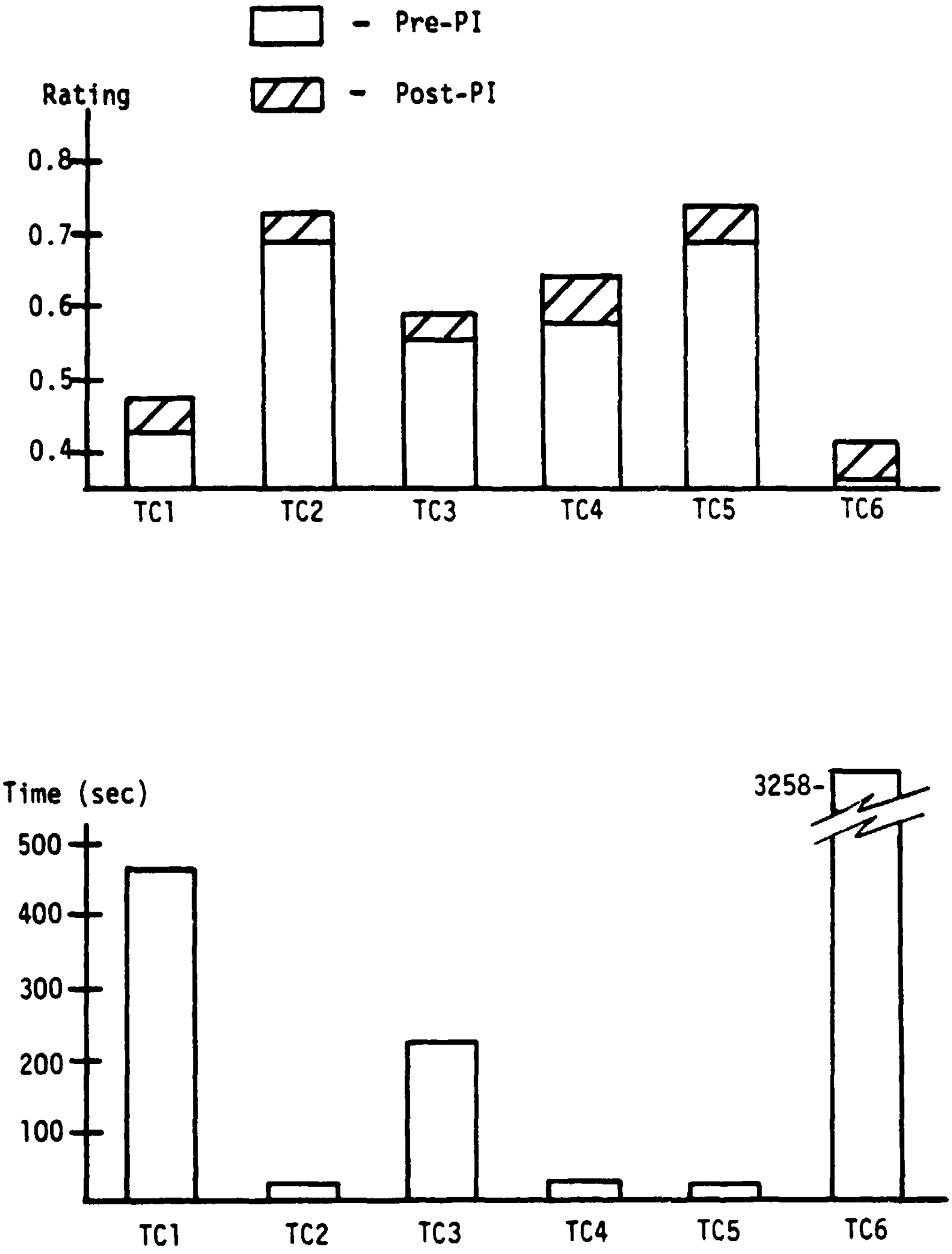


Figure 30. CAPSTAR Performance for Test Circuits

Since similar results have been noted for all six test circuits, the results for these series will be presented for the circuit TC2 only.

The results of test series 2, in which MAXSOL was varied are shown in Figure 31. As implied in this figure, the highest-rated placements that are found are usually those formed early in the folding history. This is in line with the nature of the folding procedure which should produce the best placements first. However, the CAPSTAR operating philosophy has been to make MAXSOL at least 200 in order to allow selection of the IMPROVE best placements from as large a placement sampling as possible. The simplicity of the folding procedure allows this without incurring severe time penalties.

Test series 3, the results of which are summarized in Figure 32, involved a study of the effects of IMPROVE on processing time and final placement rating. As indicated in this figure, the best post-improvement placement is generally obtained from among the best two to three folding solutions. In addition, the relative slowness of the NPWI improvement procedure causes severe execution-time penalties for a large IMPROVE.

The fourth series of tests consisted of a study of CAPSTAR performance with respect to variance of the row neighborhood distance (RN). The curve shown in Figure 33 summarizes the results of this series. As can be seen in

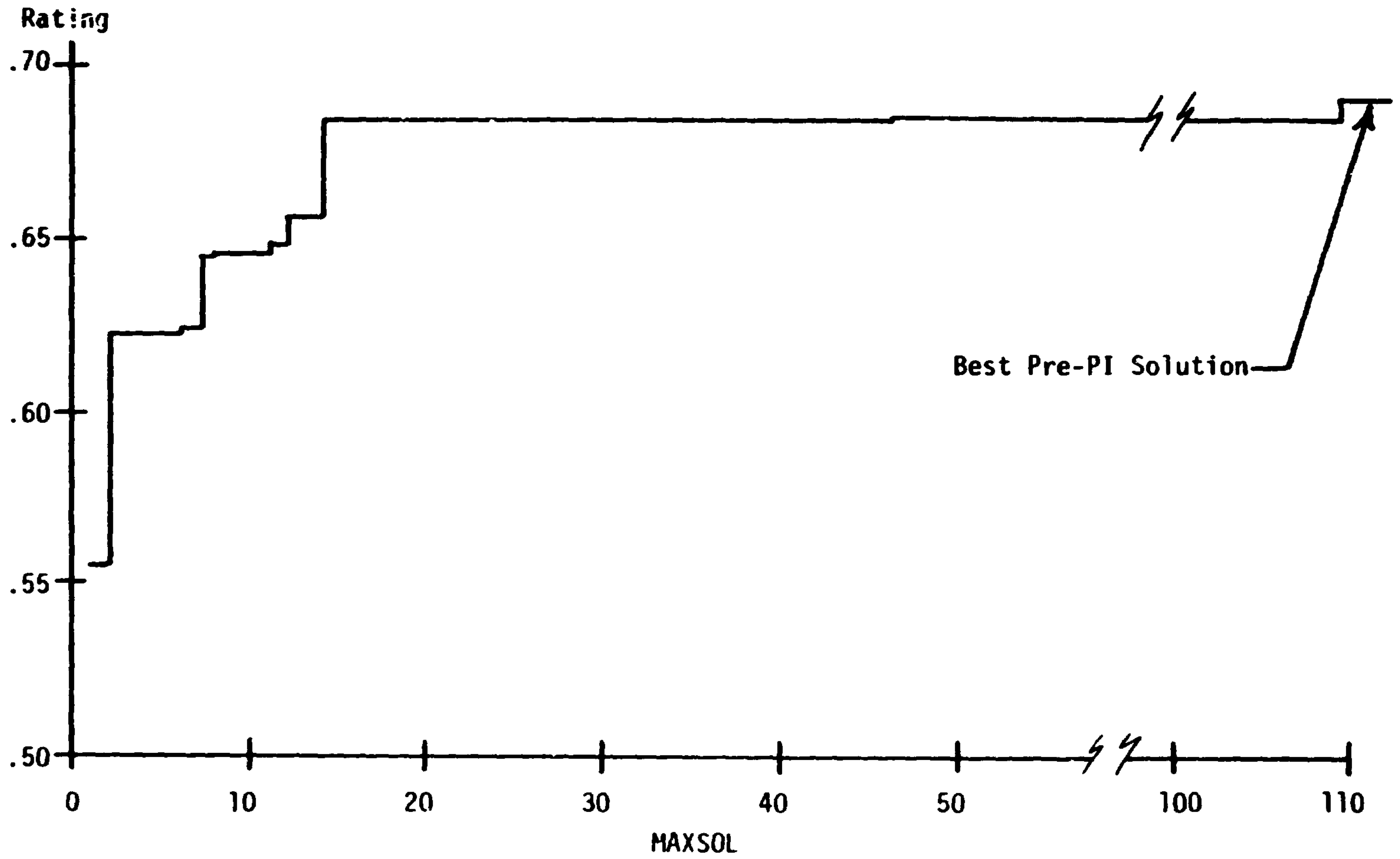


Figure 31. Effect of MAXSOL on Best Pre- ng

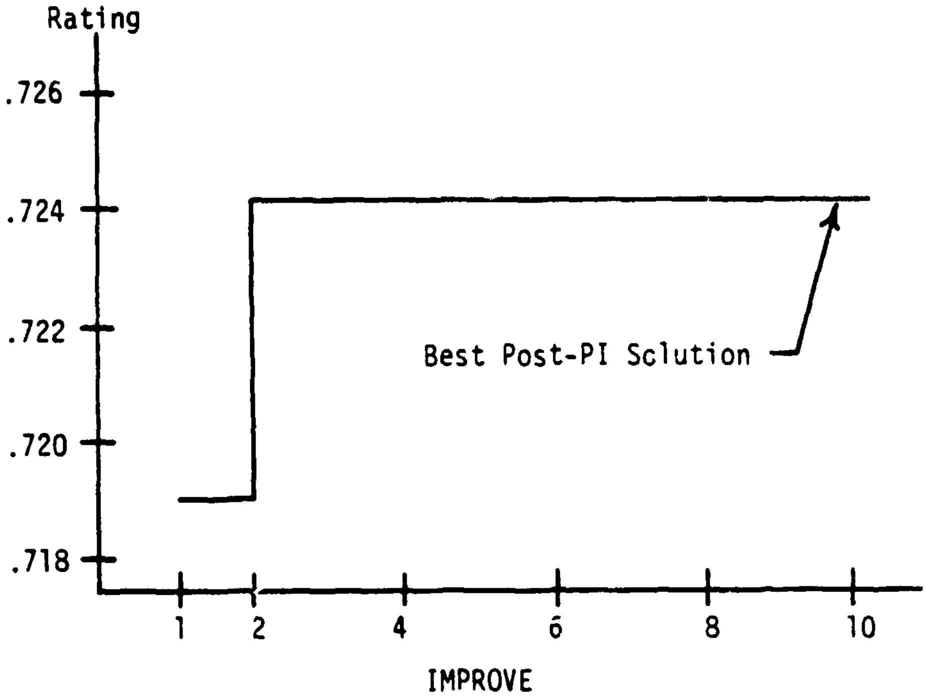


Figure 32. Effect of IMPROVE on Best Post-PI Rating

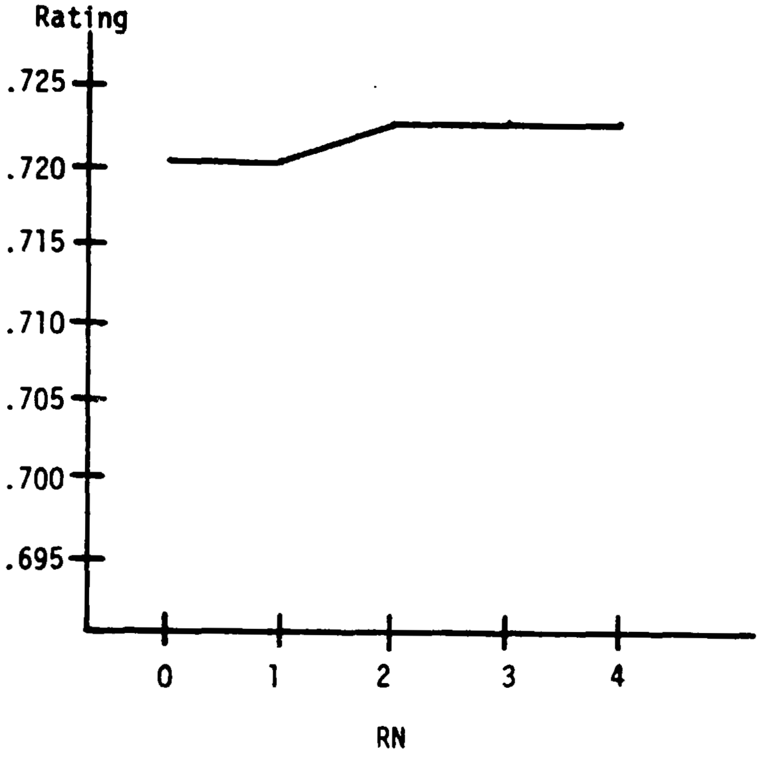


Figure 33. Effect of RN on Best Post-PI Rating

this figure, the value of RN has very little overall effect on placement rating. This indicates that the rows of the placement produced by the folding step are very nearly in an optimal relative placement. This can be intuitively supported by considering that the nature of the folding operation is such that a cell in row J most likely has its neighbors from the linear order on rows (J+1) and (J-1). Thus, row J should be tightly linked to the surrounding rows and interchange with another row would not be apt to provide improvement.

The results of test series 5, in which the cell neighborhood distance (CN) was varied, are shown in Figure 34. It can be seen from this figure that the main power of the placement improvement routine lies in reorganization of the cells within a placement row rather than in row movement. However, only limited improvement is achieved by increasing CN above 2. Based on the results of this and the preceding test series, the nominal settings for RN and CN have been established as 2.

Test series 6 was a study of the effects on placement rating when the size of the STAR is increased. Figure 35 shows a portion of the results obtained by fixing the number of STAR columns and increasing the number of rows. As might be expected, the ratings of the resultant placements show significant improvement as the number of rows increases (i.e., as STAR density decreases). Similar results as shown

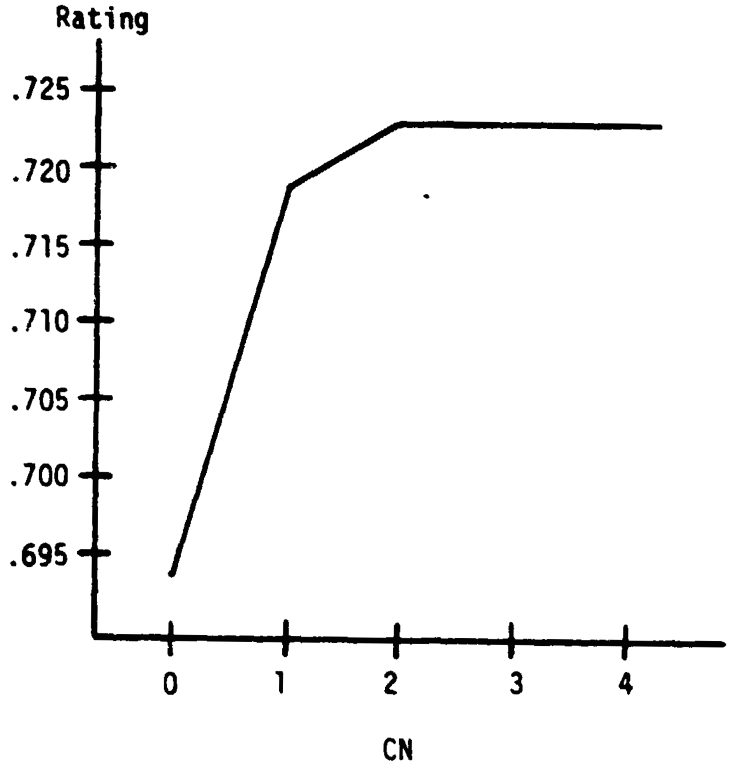


Figure 34. Effect of CN on Best Post-PI Rating

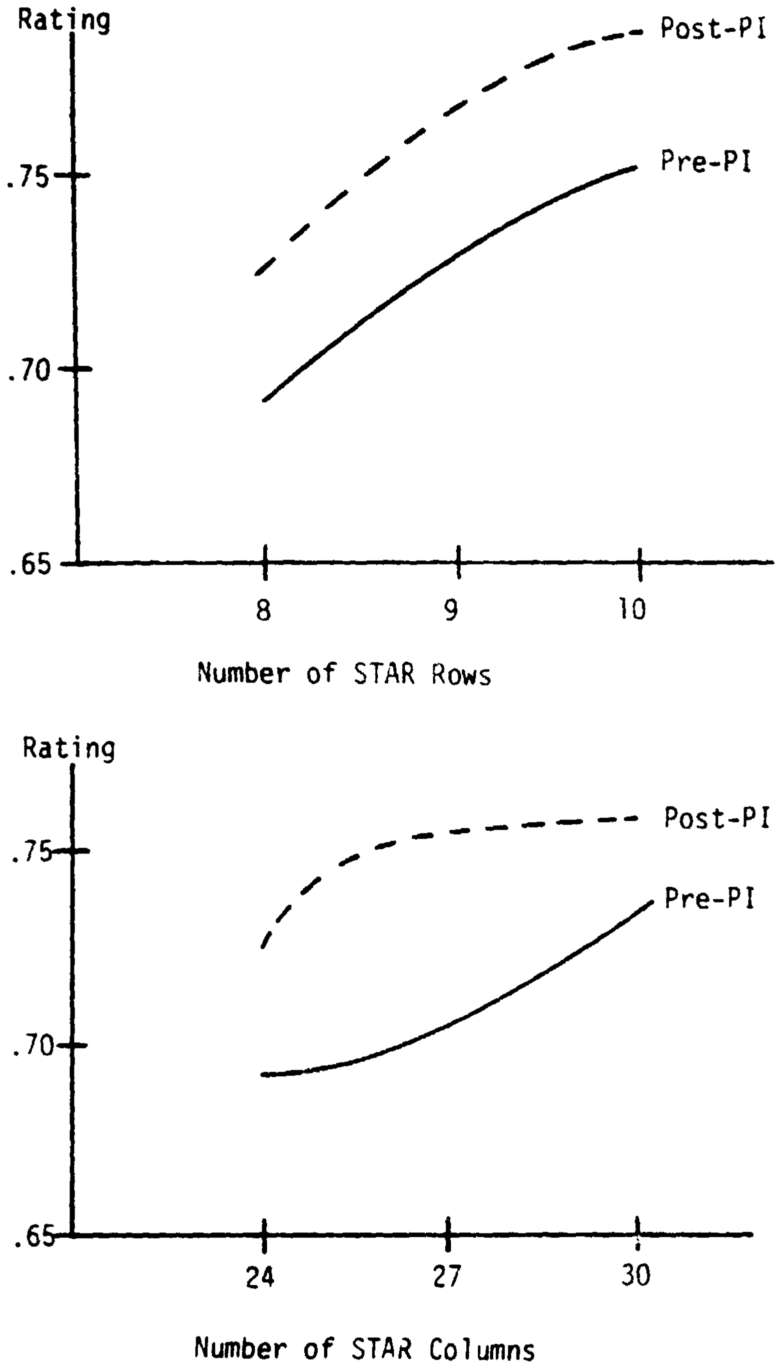


Figure 35. Effects of the Number of STAR Rows and Columns on Placement Rating

in Figure 35 for an equivalent increase in the number of STAR columns with a constant number of rows.

Figure 36 illustrates an interesting phenomenon observed from the results of test series 6. As shown in this curve, decreasing effect is produced by the placement improvement procedures as the STAR area is increased. The implication is that the folding strategy used is most effective (i.e., the folding solutions are more nearly optimum) for sparsely-populated STARs. This would be the expected result, since restrictions on STAR size tend to force more reliance on the optimality-reducing rotation and lookback operations.

Test series 7 considered effects of block depth on the ratings of unimproved placements. The curve shown in Figure 37 displays the average pre-improvement rating of 477 solutions for the TC2 circuit versus the block depths used in obtaining the solutions. The peak at block depth 4 in this figure indicates that, at this depth, the horizontal and vertical channel usages were approximately equal for the majority of the generated placements.

The sharp upward trend at the maximum block depth is unexplained. Similar unpredicted peaks have been noted in the rating-versus-block depth characteristics of several of the test circuits. The existence of these peaks tends to reinforce the usefulness of completely sweeping the block depth range in the folding procedure.

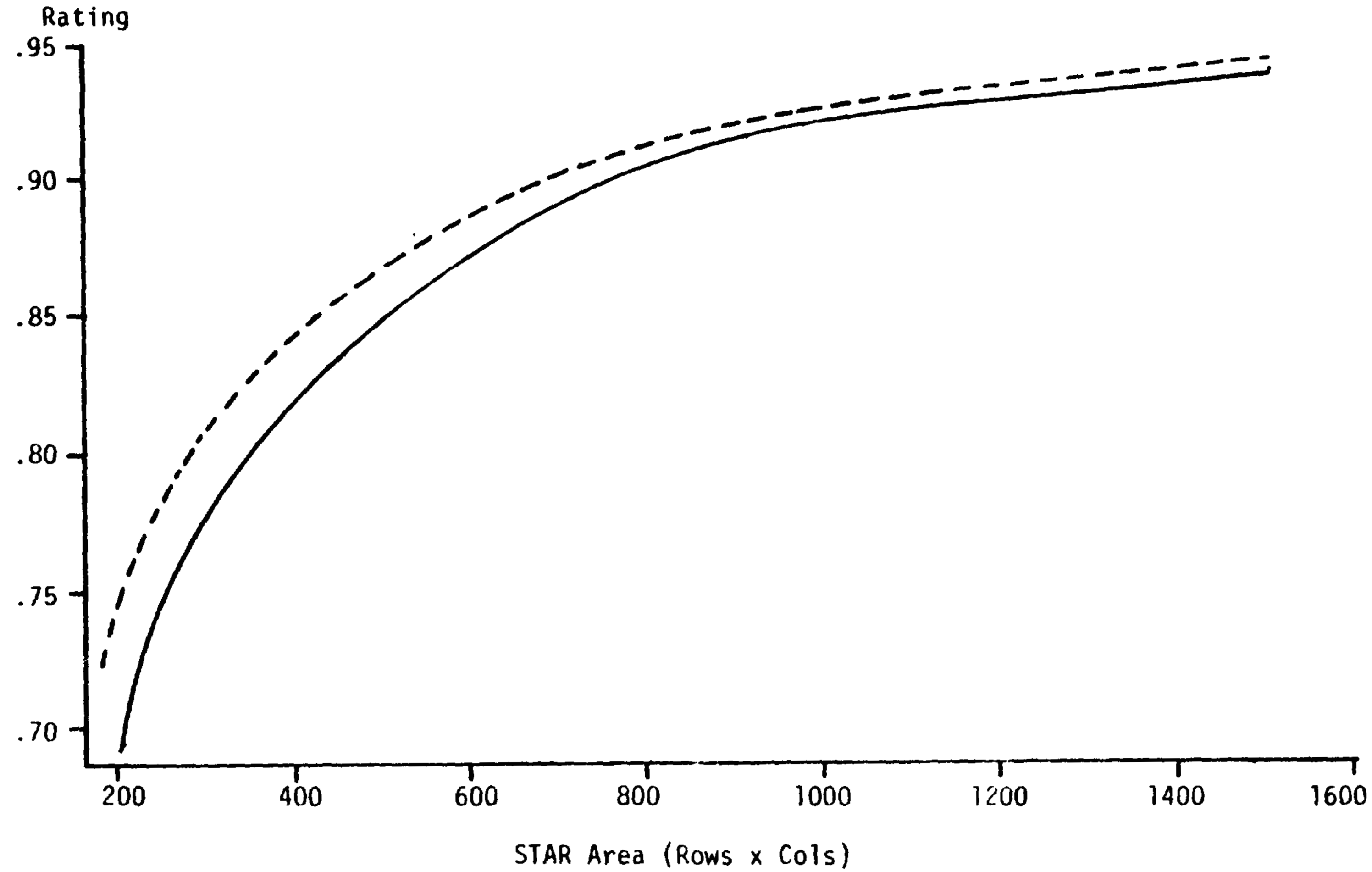


Figure 36. Effect of STAR Area on Placement Rating

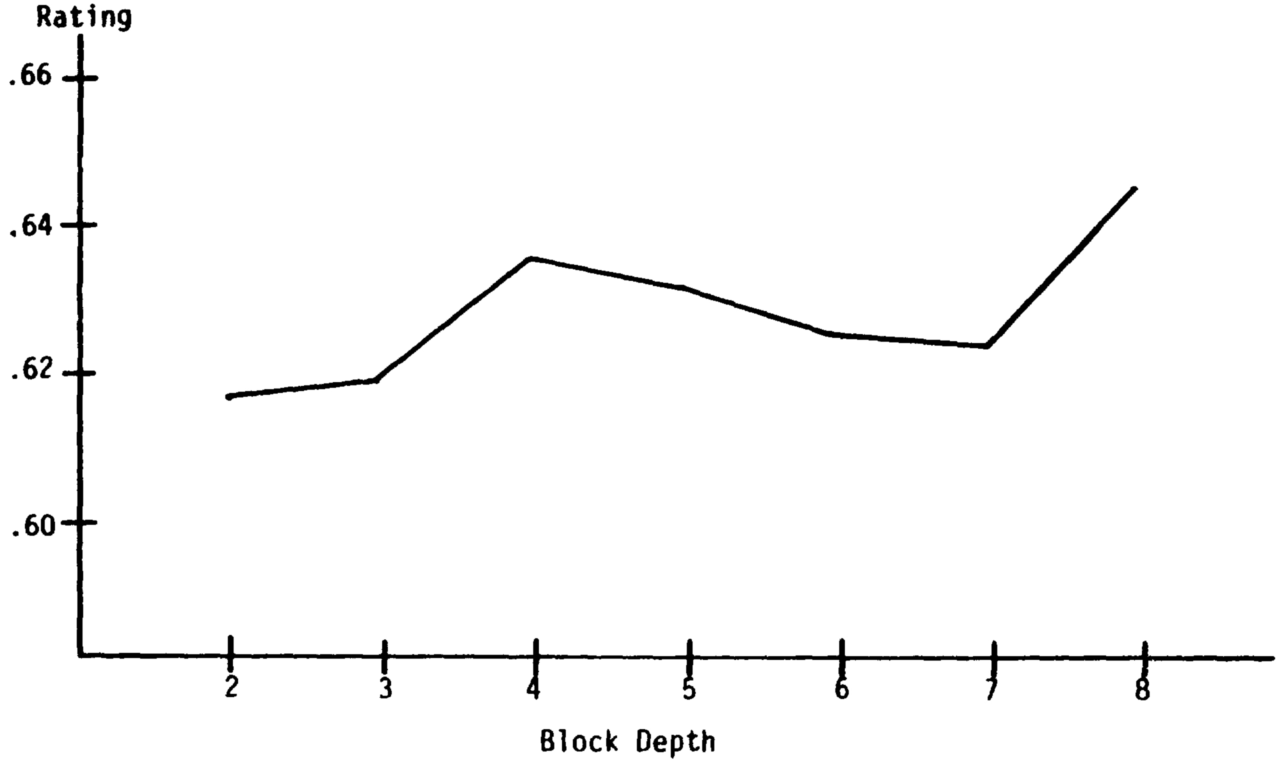


Figure 37. Effect of Block Depth on Average Pre-PI Ratings

An important exercise in the validation of CAPSTAR procedures has been the attempt to identify situations in which it is possible to place a circuit on a given-size STAR, but in which the CAPSTAR folding procedures fail to form a placement. Early CAPSTAR versions, in which the rotation and lookback operations were not used, exhibited numerous folding failures. Later versions, in which only one of the two operations appeared produced folding failures for high chip densities (over 90% of the STAR used).

Since the use of both operations was initiated, roughly 5000 trial executions of CAPSTAR have been made. The test applications have ranged up to 95% chip density. During this time, no folding failures have been detected.

While no validation certainties can be based on this limited testing, the current operating assumption is that no folding failures will occur for placement densities less than 95%. It is anticipated that future use of CAPSTAR in the application environment for which it is designed will allow establishment of this bound with more certainty.

The preceding discussion has not treated the problem of determination of the nearness of CAPSTAR-produced placements to the optimum. As indicated previously, the only known methods of identifying optimum placements involve complete investigation of the solution space which is computationally feasible for trivial cases, only.

A previously-noted method of estimating nearness of a placement to the optimum involves use of Monte Carlo techniques to form the distribution of ratings of a large number of placements of a circuit. The distribution is treated as a normal distribution and the expected fraction of placements with ratings lower than the placement of interest is calculated by standard techniques.

A modified version of this procedure is implemented in the CAPSTAR system as noted in the preceding discussion of placement quality. The sample space is the set of MAXSCL solutions produced by the folding step with the set of IMPROVE improved placements. The qualities calculated for the highest-rated placement of each test circuit range from 0.99707 for TC6 to 0.99918 for TC5. In other words the best solutions to each problem are expected to lie within 0.3% of the optimum.

Due to the non-random methods used for formation of the sample space and to the limited sample space size, the CAPSTAR quality estimation procedure cannot be used for performance validation. The use of random (Monte Carlo) techniques has thus been undertaken to form large numbers of random placements for the TC3 and TC4 circuits. Quality estimation procedures performed using these data bases has shown quality in excess of 0.999 for each circuit (indicating ratings within 0.1% of optimum).

While the use of these methods for estimation of the optimum is suspect for a number of reasons (among them, the unproven Gaussian nature of the rating distribution), it seems a reasonable approach for predicting the fraction of placements with ratings lower than a given rating. Based on experimental data, this fraction for the average CAPSTAR placement is (conservatively) set at 0.98.

A final note regarding quality computation is in order. Since methods of rating placements for routing ease are imperfect, a technique for quality computation which is based on placement ratings does not compute quality on the basis of nearness to the optimum (most easily routed) placement but, rather, on the nearness to the highest expected rating. Properly, then, placement quality should not be interpretted as "nearness to the optimum" but as "nearness to the highest expected rating".

This section has dealt with CAPSTAR performance without regard to other placement techniques. In the following section, the performance of CAPSTAR will be contrasted with that of the PWI placement improvement technique.

Comparison With PWI Techniques

The second phase of the CAPSTAR testing procedure involves comparison of the performance of the CAPSTAR procedures with that of another, more commonly used placement method.

The placement method selected for comparison to CA?STAR is a modified version of the Monte Carlo IP technique in combination with the PWI PI technique. Thesa methods were selected due to their relative simplicity and the good placements which can be obtained if the PWI procedure is allowed to run to completion.

The initial placement is generated by forming a dummy linear order (cells ordered by cell number) and by using the CAPSTAR folding section to form a STAR placement. The normal methodology used for cell numbering prevents this from being a true random initial placement. Cell numbers are usually assigned at the logic-diagram stage and cells (gates) which are drawn in the same area of the diagram are typically assigned numbers in the same range. Since these cells have a higher-than-random probability of being connected, the dummy linear order has a lower total interconnection length than a random linear order. Since the folding strategy preserves the linear order, the performance of the PWI routine might be expected to be slightly better than that obtained from a truly random start.

After construction of the initial placement, the PWI routine is begun and proceeds in the manner outlined in Chapter II. The decision to accept or to reject a trial interchange is made on the basis of a placement rating produced by the CAPSTAR rating facility. Only trial interchanges which result in an increased rating are accepted. The PWI procedure terminates when no trial interchange between any pair of cells results in placement improvement.

Four of the test circuits (TC1 through TC4) were selected for use in the comparison procedure. The use of TC6 was rejected due to the excessive amount of time required to perform the PWI procedure for such a large circuit.

CAPSTAR and PWI performance for the four circuits is shown in Figure 38. As shown in this figure, post-improvement CAPSTAR placement ratings are on the same order as those of the PWI routine.

For very small circuits, such as TC2 and TC4, the time costs associated with the PWI routine are slightly less than that of CAPSTAR. For the handling of large circuits (TC3), the time spent in comparison of all cell pairs quickly forces the execution time for the PWI routine above that of the CAPSTAR procedures.

As can be seen from the results for TC1, CAPSTAR execution time can increase greatly when extremely dense placements are required. In the case of TC1, over 92% of the STAR area is occupied by cells. Because of this high density, and corresponding difficulty of fitting the placement onto the STAR, over 11,000 placement attempts were required by the CAPSTAR folding section in order to obtain

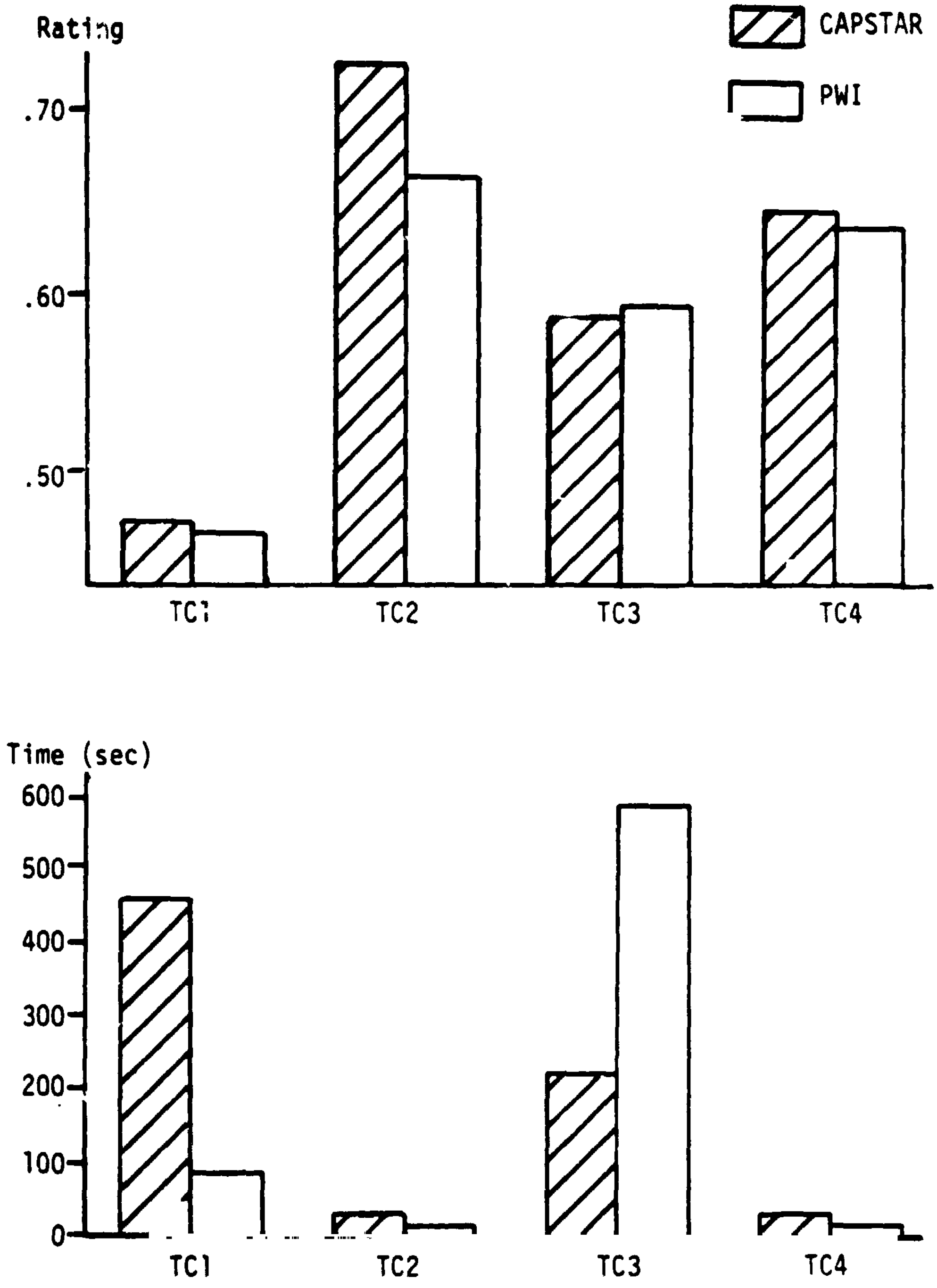


Figure 38. CAPSTAR and PWI Performance

200 (MAXSOL) successful placements. If desired, the CAPSTAR time requirements for these high-density cases can be reduced by use of smaller values for MAXSOL (for TC1, a MAXSOL value of 6 would produce the same final placement rating).

The results of the comparison procedure indicate that the LOF-NPWI techniques used in CAPSTAR produce placements with ratings roughly equal to those of the PWI routines. However, the execution-time required for use of PWI procedures on large circuits can, in general, be reduced by use of the CAPSTAR techniques.

VII. Conclusion

A strategy for the placement of digital logic cells for the Standard Transistor Array (STAR) has been presented in this dissertation. The placement procedures used are based on the linear ordering-folding (LOF) technique which have been utilized in a number of simpler organizations. Modifications to the usual folding methods which provide minimization of interconnection channel crowding and which allow placement of extremely dense layouts have also been given. Methods for measurement of placement optimality have been developed.

The organization of a program which implements the placement procedures has been shown and the results of program performance testing have been indicated. The program has been shown to produce cell placements which are comparable in optimality to those produced by an existing placement procedure. The execution time which can be saved by use of the new procedure for the placement of large circuits has been noted.

A number of areas for further study are indicated. Several of these are outlined in the following paragraphs.

A method for translation of "routing ease" into criteria measurable during placement formation is required.

This method should be applicable for use with any technology and routing strategy. Placement characteristics which lead to ease of routing should be identified and methods for measurement of the characteristics developed.

Improved methods for estimation of the nearness of a given placement to the optimum are needed. As noted previously, the Monte Carlo methods currently in use are unsatisfactory in several respects. If these techniques are to be used in the future, the distribution of ratings of the sample set should either be proven to be normal or should be re-defined. In addition, the effects on the rating distribution of various rating techniques should be analyzed.

Another future study could be devoted to analysis of the characteristics of folding strategies other than the one proposed here. It might be found that folding techniques can be developed which optimize a placement with respect to other criteria, or, which better optimize with respect to the criteria proposed. LOF technique performance could then be increased and applicability broadened.

The LOF procedures have been shown to perform relatively well for STAR-like structures when they are followed by a simple placement improvement routine. Future work in this area might incorporate studies of LOF technique performance as an initial placement procedure when followed by a more powerful PI method.

Achieving the goals listed above would aid development of highly effective placement routines for use with STAR and related technologies. In addition, the identification of other suitable placement improvement routines may reduce execution time with a corresponding reduction in semicustom IC development costs.

REFERENCES

[1] Franson,P. "Dont Overlook Semicustom ICs for your Next Design Project," EDN, Feb.5,1977, pp.72-76.

[2] Edge,T., "Semicustom and Custom Integrated Circuits and the Standard Transistor Array Radix (STAR)," Electronics and Control Lab, George C. Marshall Space Flight Center, RTOP: 506-18-31.

[3] Hanan,M. and J. Kurtzberg, "Placement Techniques," Chapter 5 in Design Automation of Digital Systems, M.A.Breuer, ed., Englewood Cliffs, NJ, Prentice-Hall, vol.1, 1972, pp.213-282.

[4] Deo,N., Graph Theory with Applications to Engineering Computer Science, Englewood Cliffs, NJ., Prentice-Hall, 1974, p.32.

[5] Goldstein, A.J. and D.G. Schweikert, "A Proper Model for Testing the Planarity of Electrical Circuits," Bell System Technical Journal, vol.52, no.1, Jan,1973, pp.135-142.

[6] Vancleemput,W.M. and J.G.Linders, "An Improved Graph-Theoretic Model for the Circuit Layout Problem," Proceedings of the 11th Design Automation Workshop, Denver, CO., June 17-19,1974, IEEE no. 74 CH0865-6C, pp.82-90.

[7] Kurtzberg,J.M., "Algorithms for Backplane Formation," Microelectronics in Large Systems, Spartan Books, 1965, pp.51-76.

[8] Shupe,C.F., "Automatic Component Placement in the NOMAD System," Proceedings of the 12th Design Automation Conference, Boston,MA, June 23-25,1975, IEEE no. 75 CH0980-3C, pp.162-172.

[9] Lawler,E.L. and D.Wood, "Branch and Bound Methods: A Survey," Journal of Operations Research, vol.14, 1966, pp.699-719.

[10] Little,J., et.al., "An Algorithm for the Travelling Salesman Problem," Journal of Operations Research, vol.11, Nov.1963, pp.972-989.

[11] Gilmore,P., "Optimal and Suboptimal Algorithms for the Quadratic Assignment Problem," SIAM Journal, vol.10, no.2, June,1962, pp.305-313.

[12] Hillier,F. and M.Conners, "Quadratic Assignment Problems and the Location of Indivisible Facilities," Management Science, vol.13, no.1, Sept.,1966, pp.42-57.

[13] Weinberger,A., "Large Scale Integration of MOS Complex Logic: A layout Method," IEEE Journal of Solid-State Circuits, vol.SC-2, no.4, Dec.,1967, pp.182-190.

[14] Larsen,R., "Computer-Aided Preliminary Layout Design of Customized MOS Arrays," IEEE Transactions on Computers, vol.C-20, no.5, May,1971, pp.512-523.

[15] Yoshizawa,H., H.Kawanishi, and K.Kani, "A Heuristic Procedure for Ordering MOS Arrays," Proceedings of the 12th Design Automation Conference, Boston,MA, June 23-25,1975, IEEE no. 75 CH0980-3C, pp.384-393.

[16] Mattison,R., "Design Automation of MOS Artwork," Computer, Jan.,1974, pp.21-27.

[17] Feller,A., "Automatic Layout of Low-Cost Quick-Turnaround Random-Logic Custom LSI Devices," Proceedings of the 13th Design Automation Conference, San Francisco, CA, June 28-30,1976, IEEE no. 76 CH1098-3C, pp.79-85.

[18] Hanan,M., P.Wolff, and B.Agule, "A Study of Placement Techniques," Journal of Design Automation and Fault-Tolerant Computing, vol.1, no.1, Oct.,1976, pp.28-61.

[19] Ciampi,P., "A System for Solution of the Placement Problem," Proceedings of the 12th Design Automation Conference, Boston,MA, June 23-25,1975, IEEE no. 75 CH0980-3C, pp.317-323.

[20] Matthews,A., "A Human Engineered PCB Design System," Proceedings of the 14th Design Automation Conference, New Orleans,LA, June 20-22,1977, IEEE no. 77 CH1216-1C, pp.182-186.

[21] Steinberg,L. "The Backboard Wiring Problem: A Placement Algorithm," SIAM Review, vol.3, no.1, Jan.,1961, pp.37-50.

[22] Harvey,J. "Automated Board Layout," Proceedings of the ACM-IEEE Design Automation Workshop, Dallas,TX, June 26-28,1972, IEEE no. 72 CH0706-2-SC, pp.264-271.

[23] Breuer,M.A., "A Class of Min-Cut Placement Algorithms," Proceedings of the 14th Design Automation Conference, New Orleans,LA, June 20-22,1977, IEEE no. 77 CH1216-1C, pp.284-290.

[24] Hanan,M. and J.M.Kurtzberg, "A Review of the Placement Problem and Quadratic Assignment Problems," SIAM Review, vol.14, no.2, April,1972, pp.324-342.

[25] Scanlon,F., "Automated Placement of Multi-Terminal Components," Proceedings of the SHARE-ACM-IEEE Design Automation Workshop, Atlantic City,NJ, June 28-30,1971 , pp.143-154.

[26] Giugliano,A. and F.Bosisio, "Present and Future on P.C.B. Layout Design Automation at SIT-Siemens," Proceedings of the 12th Design Automation Conference, Boston,MA, June 23-25,1975, IEEE no. 75 CH0980-3C, pp.134-143.

[27] Quinn,N., "The Placement Problem as Viewed from the Physics of Classical Mechanics," Proceedings of the 12th Design Automation Conference, Boston,MA, June 23-25,1975, IEEE no. 75 CH0980-3C, pp.173-178.

[28] Crocker,N., R.McGuffin, and A.Micklethwaite, "Automatic ECL LSI Design," Proceedings of the 14th Design Automation Conference, New Orleans,LA, June 20-22,1977, IEEE no. 77 CH1216-1C, pp.158-167.

[29] Ueda,K., Y.Sugiyama, and K.Wada, "An Automatic Layout System for Masterslice LSI: MARC," IEEE Journal of Solid-State Circuits, vol.SC-13, no.5, Oct.,1978, pp.716-721.

[30] Wilson,D. and R.Smith, "An Experimental Comparison of Force Directed Placement Techniques," Proceedings of the 11th Design Automation Workshop, Denver,CO, June 17-19,1974, IEEE no. 74 CH0865-6C, pp.194-199.

[31] Schuler,D. and E.Ulrich, "Clustering and Linear Placement," Proceedings of the ACM-IEEE Design Automation Workshop, Dallas,TX, June 26-28,1972, IEEE no. 72 CH0706-2-SC, pp.50-56.

[32] Cox,G. and B.Carroll, "Placement Techniques for the Standard Transistor Array (STAR)," Technical Report, NASA Contract no. NAS8-31572, Electrical Engineering Dept., Auburn University, Auburn,AL, Sept.,1978.

[33] Graham,R., "The Combinatorial Mathematics of Scheduling," Scientific American, March,1978, pp.124-132.

[34] Cox, G. and B. Carroll, "CAPSTAR Programmer's Guide," NASA Contract no. NAS8-31572, Electrical Engineering Dept., Auburn University, Auburn, AL, July, 1979.

[35] Cox, G. and B. Carroll, "CAPSTAR User's Guide," NASA Contract no. NAS8-31572, Electrical Engineering Dept., Auburn University, Auburn, AL, July, 1979.

APPENDIX A

EQUATIONAL DEVELOPMENT

Development of Equations 3-1 and 3-2

The area of interest is the entire STAR, so

$$U(H) = \frac{L}{4ROWS \cdot COLS}\left[\left(\frac{(n-1)(W+1)}{n}\right) F(1) + \left(\frac{(n-2)(2W+1)+2(W+1)}{n}\right) F(2) \right]$$

$$U(V) = \frac{L}{ROWS \cdot COLS}\left[\left(\frac{2}{n} \right) F(1) + \left(\frac{4}{n} \right) F(2) \right]$$

Complete filling of the STAR is assumed. Then

$$COLS = nW$$

and

$$nROWS = C$$

where C is the number of cells.

Then

$$\frac{L}{ROWS \cdot COLS} = \frac{L}{CW} \quad .$$

Using the same worst-case conditions as in Chapter 2,

$$\frac{L}{CW} \approx 1 \quad .$$

Then, for F(1) = 0.5 and F(2) = 0.1,

$$U(H)_{wc} \approx (0.175)W(1- \frac{1}{n}) - (0.125)(\frac{1}{n}) + 1.5$$

$$U(V)_{wc} \approx \frac{1.4}{n}$$

Development of Equations 3 -3 and 3 -4

Initially,

$$U(H) \approx \frac{L}{4ROWS \cdot COLS} \left[\left(\frac{(n-1)(W+1)}{2n} \right) F(1) + \right.$$

$$\left. \left(\frac{(n-1)(W+1)}{n} \right) F(2) \right]$$

$$U(V) \approx \frac{L}{ROWS \cdot COLS} \left[\left(\frac{n+1}{n} \right) F(1) + \left(\frac{2n+1}{n} \right) F(2) \right] \quad .$$

By the same reasoning as in the previous development,

$$\frac{L}{ROWS \cdot COLS} \approx 1 \quad ,$$

for the worst case.

Then, for $F(1) = 0.5$ and $F(2) = 0.1$,

$$U(H)_{wc} \approx (0.088) \left[\left(1 - \frac{1}{n}\right) W - \frac{1}{n} + 1 \right]$$

$$U(V)_{wc} \approx 0.7 + \frac{0.6}{n} \quad .$$

Development of Equation 6-1

The worst-case ETD occurs if the unused transistors lie as far as possible toward tte ends of each row. ETDWC, then is thf ETD obtained when the unused transistors are evenly divided among the rows, and evenly divided on a row between the two ends.

Then,

$$ETDWC = (2ROWS) \sum_{i=1}^{\frac{MT}{2ROWS}} \left[\frac{COLS}{2} - i \right]$$

$$= (2ROWS)\left(\frac{MT}{2ROWS}\right)\left(\frac{COLS}{2}\right)$$

$$- (2ROWS) \sum_{i=1}^{\frac{MT}{2ROWS}} i \quad .$$

Now,

$$\sum_{j=1}^{x} j = \frac{x}{2}(1+x) \quad .$$

So,

$$ETDWC = \frac{MT \cdot COLS}{2} - (2ROWS)\left(\frac{MT}{4ROWS}\right)\left(1 + \frac{MT}{2ROWS}\right)$$

$$\mathrm{ETDWC} = \frac{\mathrm{MT}\cdot\mathrm{COLS}}{2} - \frac{\mathrm{MT}}{2} - \frac{\mathrm{MT}^2}{4\mathrm{ROWS}}$$

$$= \frac{\mathrm{MT}}{2}\left[\ \mathrm{COLS} - 1 - \frac{\mathrm{MT}}{2\mathrm{ROWS}}\ \right] \quad .$$