Analysis of Strategies for Constructive General Block Placement

SHMUEL WIMER AND ISRAEL KOREN, MEMBER, IEEE

Abstract—The problem of general block placement in VLSI is the topic of this paper. Among the existing approaches to its solution we concentrate on the constructive one, where blocks are selected and located one at a time. Two of the main features of the constructive approach are its high computational efficiency and its ability to support both automatic and interactive placement.

We present some well-known strategies for the selection of the next block to be located, propose new ones and establish a methodology to evaluate them. We then show that the optimization problem arising in constructive placement can be reduced into several, much simpler, subproblems. Next, objective functions for locating the selected block to achieve a "good" layout are presented. We discuss objective functions of three different metrics: the squared Euclidean, rectilinear and Euclidean, obtain appropriate optimization problems and solve them analytically, using efficient computational schemes. These solutions have been implemented and are used in a real VLSI chip design environment. Finally, we show that the squared Euclidean and the rectilinear metrics are preferable to the Euclidean one.

Index Terms—Constructive placement, layout, physical design, optimization.

I. Introduction

To cope with the problem of designing a chip such that some requirements on the electrical performance, total area, design duration and flexibility for changes, are met, an *hierarchical* methodology should be adopted. We call the building blocks at some level of the hierarchy *son blocks*, and the block created by combining son blocks together is called a *father block*. When performing the layout of a father block, we have to know only the total dimensions of its son blocks and how they communicate with each other. The layout procedure is in general very complicated and hence, is partitioned into two separate phases, *placement* and *routing*. An overview of various placement algorithms can be found in [6] and [10].

This paper concentrates on *constructive* placement algorithms, where at each step we first *select* a block and then *locate* it, one at a time. Consequently, a constructive algorithm consists of a strategy for selecting a yet unplaced block and a scheme for locating the selected block in the available area, optimizing some objective function. A major advantage of these algorithms is the considerably lower computational effort required to execute them, compared to that required by other placement methods, since they lead to simpler optimization problems. However, a global minimum is seldom achieved and it is possible to get stuck at a local minimum. Another important advantage of constructive placement algorithms is that they allow human intervention at any step of the algorithm, which suits an environment where the designer may wish to be involved in the layout process.

In the next section we review several well-known strategies for selecting the next block to be located, we then propose new ones and compare them. In Section III we propose a significant reduction in computations for optimal locations. Then, in Section IV we present objective functions for various metrics.

A layout package based on the theory presented in this paper was developed and is used in a real chip design environment [9]. It supports human intervention in the layout process, i.e., the selection and location of a block at any step can be accomplished either automatically or manually.

II. Selection Strategies and Their Evaluation

The placement problem can be stated as follows: given a collection of blocks with logical interconnections between them, find among all the feasible arrangements of blocks the "best" one. In order to find it, we must have some *objective function* that should reflect an estimate of the expected cost of routing which is performed once the placement is completed, and the total chip area used. A constructive placement algorithm attempts at each step to expand a partial configuration such that the increase of the objective function is minimal.

Given the geometry of the father block B_0 and its b son blocks B_i, $1 \leq i \leq b$, and the logical interconnections between their *ports* through several *nets*, the problem of the optimal placement of general blocks is to minimize a nonnegative real function $f(\Phi_1, x_1, y_1, \cdots, \Phi_b, x_b, y_b)$, where Φ_i is the orientation of the ith son block (generally, there are eight possible orientations resulting from two reflections and four rotations), and (x_i, y_i) is the location of its center within the coordinates of B_0. The function f is the cost of a complete *valid* placement, in which all the

Manuscript received November 11, 1985; revised May 1986, February 1987, and May 1987. Part of this research was performed while S. Wimer was with the Design Center of National Semiconductor, Tel-Aviv, Israel. The review of this paper was arranged by Associate Editor R. H. J. M. Otten.

S. Wimer is with the Department of Electrical Engineering, Technion, Israel Institute of Technology, Haifa 32000, Israel, and with IBM Israel Scientific Center, Technion City, Haifa 32000, Israel.

I. Koren is with the Department of Electrical and Computer Engineering, University of Massachusetts, Amherst, MA 01003, on leave from the Technion, Haifa 32000, Israel.

IEEE Log Number 8718402.

son blocks are contained within the area of the father block and no two disjoint son blocks overlap. The algorithms presented in this paper can be employed without a given geometry for B_0 by initially allowing B_0 to be arbitrarily large and then reducing it to the desired size after the placement of all B_i's has been completed.

Even for a very simple objective function, the solution of the placement problem is very complicated. There are b discrete variables Φ_i, $1 \leq i \leq b$, each one of them can assume one out of eight possible values, and $2b$ continuous variables x_i, y_i, $1 \leq i \leq b$. The non-overlapping requirement imposes $O(b^2)$ constraints of mixed type. For a typical number of $b = 16$ it is prohibitively complex to solve the problem analytically or even numerically. In the constructive placement scheme a block configuration is obtained incrementally. At every step a new unplaced block is selected and then located optimally within the available free area. A phase of placement improvement can take place after all B_is have been placed, where local changes like reflection and rotation of blocks, and local shifts, might further reduce the cost.

Several selection strategies were suggested [4]. In what follows we present four strategies and propose a methodological approach to their comparison. Let $\boldsymbol{B}$ denote the set of all the blocks, $\boldsymbol{B}'$ the set of already placed blocks and $\boldsymbol{B}''$ the set of yet unplaced blocks. Initially $\boldsymbol{B}' = \{B_0\}$ and $\boldsymbol{B}'' = \boldsymbol{B} - \{B_0\}$, while finally $\boldsymbol{B}'' = \phi$ and $\boldsymbol{B}' = \boldsymbol{B}$. We define a linkage $l(B_k, B_l)$ between two blocks B_k and B_l as the total sum of *weights* of nets that connect B_k and B_l. Each net is a collection of ports located in various blocks, and weights are assigned to nets according to their relative significance (e.g., number of wires, required signal propagation speed, etc.). The following are strategies for the selection of the next block to be located.

- **Among all $B_k \in \boldsymbol{B}''$ select the one for which $\Sigma_{B_l \in B'} l(B_k, B_l)$ is maximized.** This strategy tends to locate strongly connected blocks as close as possible, but it does not consider the utilization of the father block's area and often leads to a non-feasible configuration. As the placement proceeds, the free area becomes fragmented and placing large son blocks becomes more difficult. To avoid this problem, we may use the following strategy.
- **Among all $B_k \in \boldsymbol{B}''$ that satisfy $\Sigma_{B_l \in B'} l(B_k, B_l) > 0$ select the one with the maximal area.** Although this strategy increases the likelihood for a complete feasible configuration, it often results in an unacceptable routing length.
- A third strategy, which is a compromise between the first two, suggests to select blocks according to their linkage to already placed blocks and according to their area as well. **Among all $B_k \in \boldsymbol{B}''$ select the one for which $A_k \Sigma_{B_l \in B'} l(B_k, B_l)$ is maximized, where A_k is the area of B_k.** The reason for taking the product of the area and the linkage, and not their sum is to avoid normalization.
- A selection strategy should consider not only the linkage to already placed blocks, but also to unplaced blocks. For example, the next block to be located can be selected as follows: **Among all $B_k \in \boldsymbol{B}''$ select the one for which $\Sigma_{B_l \in B'} l(B_k, B_l) > 0$ and $\Sigma_{B_l \in B'} l(B_k, B_l) - \Sigma_{B_l \in B''} l(B_k, B_l)$ is maximized.** This strategy tends to defer the selection of son blocks which are strongly connected to unplaced blocks. Actually, it results in clustering of the blocks [4].

In Section IV we propose another selection strategy which is motivated by a physical interpretation of the objective function.

Figs. 1-3 are three configurations obtained for an example of 16 blocks and 16 nets, employing the first three selection strategies. The cost appearing in the three figures is the weighted sum of the squared Euclidean distances between ports participating in the same net. For the example depicted in Figs. 1-3, the third strategy based on connectivity and area considerations yields the best result, while the second one which is based only on area considerations yields the worst result. The above mentioned cost does not consider the total area directly. It depends only on the length of the interconnections between blocks, which in turn may affect the area. Still, the example shows that blocks are placed tightly for all three strategies. Part of unoccupied inter-block area may be utilized in the routing phase that follows the placement and which demands the establishing of *channels* between blocks. However, if after channels are opened, there is still free area in the periphery, the frame of the father block can be contracted, thus reducing its total area.

Clearly, the order of preference among the various selection strategies may depend on the definition of the placement cost, the method used to locate the selected block and the specific example considered. We would like therefore, to have a more "objective" measure for comparing the effectiveness of various selection strategies, a measure that will be applicable to any definition of the placement cost. Given a problem, each selection strategy determines a sequence of blocks to be placed. We say that a sequence of blocks is *feasible* if its results in a feasible placement, that is, every selected block can be successfully located in its turn within the currently available area, using the given optimization algorithm. Consider now the space of all feasible sequences, each yielding some value of the placement cost. If we select at random a sequence from the above space, then the corresponding cost is a random variable with some probability density function. Fig. 4 is an histogram obtained by drawing a random sample of 2500 feasible sequences from the space imposed by the example of Figs. 1-3.

Let S be a selection strategy, and let C_S denote the cost of a complete placement achieved by using S. Let z denote the cost of a placement which is based on a random selection of blocks, and let $g(z)$ denote its probability density function. Then, the average number of random place-

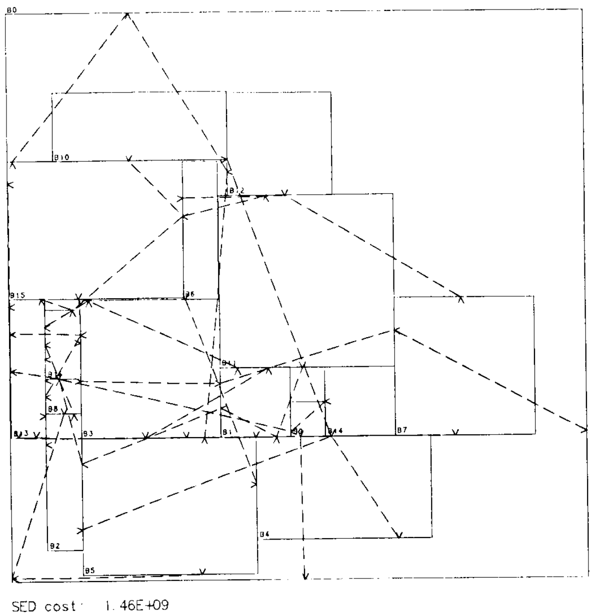


Fig. 1. Placement with strategy based on connectivity.

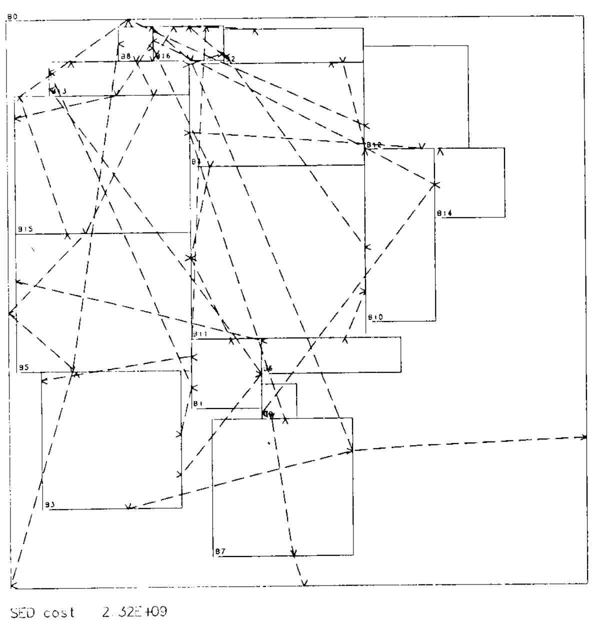


Fig. 2. Placement with strategy based on area.

ments (random feasible sequences) that must be performed to achieve a configuration that yields a cost lower than C_S, is denoted by N_S, and given by

$$N_S = \frac{1}{Pr(z \le C_S)} = \frac{1}{\int_{-\infty}^{C_S} g(z)\, dz}. \tag{2.1}$$

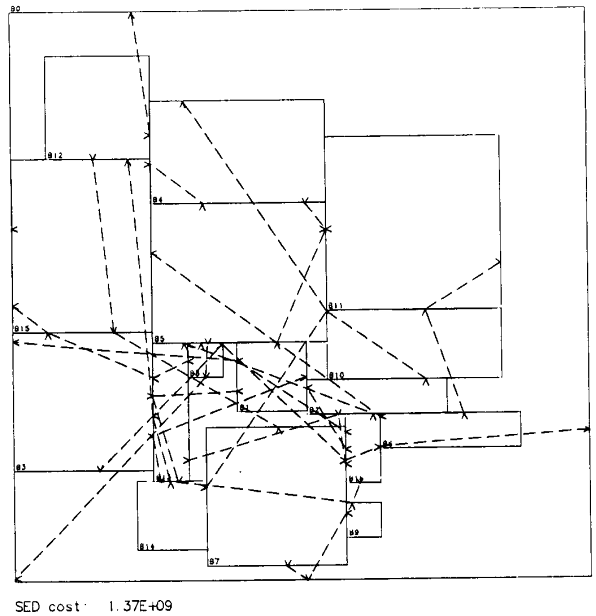


Fig. 3. Placement with strategy based on area and connectivity.

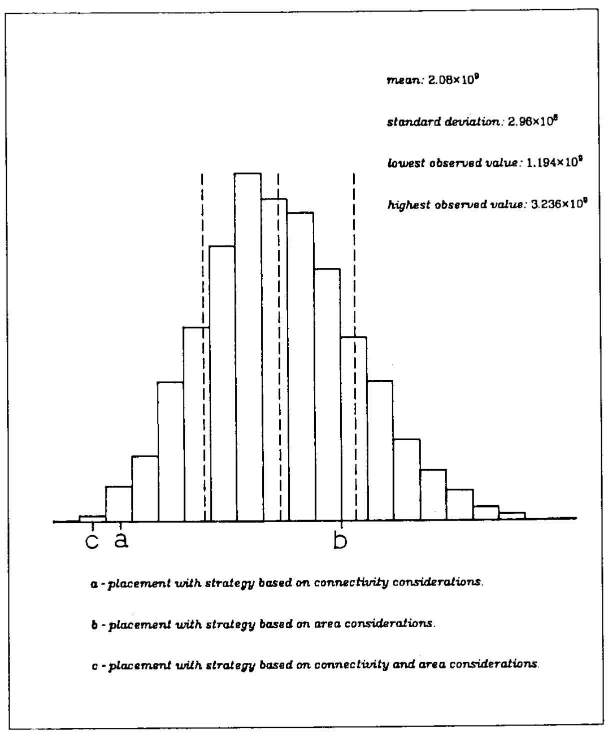


Fig. 4. Distribution of placement cost.

This type of strategy evaluation was proposed in [5] for the quadratic assignment problem. The probability density function $g(z)$ can be obtained from the sampled data using curve fitting. The problem of deriving $g(z)$ analytically is now being investigated. For our example we have estimated N_S numerically obtaining $N_a = 110$, $N_b = 1.2$, and $N_c = 1000$, as shown in Figure 4. This means for example, that under the above cost of placement and the algorithm to optimally locate the currently selected block, 1000 random selection sequences on the average, must be performed to achieve a cost lower than that obtained by the third strategy, compared to the first strategy where only 110 random selection sequences are required. These figures (i.e., 110, 1.2, and 1000) are more informative than the placement costs and they may serve as a better indication which strategy we should employ. An analytical expression for $g(z)$ (when and if found) may even further improve our situation.

III. Simplification of the Optimization Problem

The location of a new block is restricted by the frame of the father block, the frame of already placed blocks and their position within the father block. These restrictions can make the minimization problem very difficult to solve. However, the rectangular shape of the blocks makes the solution feasible. Let us decompose the free area between the already placed blocks into smaller rectangles called *prime free rectangles* (PFR) which are the maximal rectangles contained in the free area. PFR's were defined in [7] for describing rectilinear figures. In Fig. 5 *abdc* and *efgc* are prime, while *hijb* is not. There is a finite number of PFR's. Obviously, for every valid location of a new block there exists at least one PFR containing that block. Therefore, it is suggested to solve the minimization problem for each PFR separately, and then select the solution that yields the lowest value of the objective function. Computationally this is much simpler than solving the optimization problem at once by constraining the currently located block not to overlap any already placed block.

In a similar way we can introduce external constraints on the location of blocks. For example, assume that we want to place the pads of a chip only after the internal placement has been completed. It is known however, that the pads will occupy the periphery. Therefore, this area must be reserved as long as placement of blocks continues. In the layout package that has been developed, the user may define two types of constraints: those that are common to all blocks, like the one mentioned above, and those that are defined only for a certain block. The latter are generally used to prevent a block from being located in certain areas. For instance, when we locate the microcode ROM in a CPU, it be desirable to have the flexibility of increasng the ROM area. This flexibility may be achieved by disallowing the placement of the ROM in the interior area of the CPU, leaving only the area in the periphery available for its location. Invoking such a "local" constraint, the interior area of the CPU is blocked, leaving only the area in the periphery available for the location of the microcode ROM.

When a new block is located, we attempt to find its optimal orientation and position in every PFR. Before

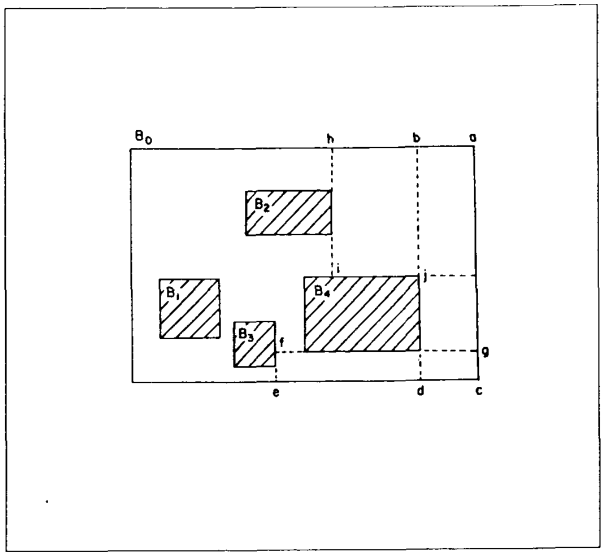


Fig. 5. Prime free rectangles.

solving the optimization problem, it is checked whether locating the block in the current orientation is feasible, otherwise the orientation is changed. Clearly, the total computation time is directly determined by the number of the PFR's, which is given in the following lemma.

Lemma: Assuming that n blocks have already been placed, there are at most $(1/2)n^2 + O(n)$ PFR's.

The proof is a trivial modification of a proof given in [8], where the problem of finding the PFR's in a rectangle enclosing a set of n points is discussed. An $O(n^2)$ algorithm to generate all the PFR's is presented there, and it is also proved that the expected number of PFR's is $O(n \log n)$.

IV. Minimizing the Objective Function

To locate the selected block optimally, an objective function which reflects its contribution to the total placement cost must be determined. It may reflect the expected routing length, the utilization of the area or both, and it requires the representation of nets in some way. Many graph representations of nets are possible. Among them are the complete graph, shortest spanning tree, Steiner tree and others [4], [13].

In the following a net is represented by the *complete graph* connecting its ports, a representation which is sometimes used [2] [12]. The edge length is measured in three different metrics: *Squared Euclidean Distance* (SED), *Rectilinear Distance* (RD) and *Euclidean Distance* (ED).[1] Let B_k denote the block to be located, and assume that the preliminary test for a feasible placement (in one of its eight possible orientations) within some PFR, has been successful. Let R denote the feasible rect-

[1]The notion of a metric is adopted although SED does not satisfy the triangle inequality.

angle for positioning the center of B_k, $R = \{(x, y) | a \leq x \leq b, c \leq y \leq d\}$. The width (height) of R equals the width (height) of the PFR under consideration minus the width (height) of B_k (after orientation), as illustrated in Fig. 6. Assume further that B_k is connected to p ports of the father and the already placed blocks. Let (x_i, y_i), $1 \leq i \leq p$, be the ports' positions in the father block (B_0) coordinate system. Let (u_i, v_i), $1 \leq i \leq p$, denote the positions of the above p ports of B_k in the B_k coordinate system, where $(0, 0)$ is its center position. Since the relative positions of B_k's ports are fixed, their positions within the father block are determined by the position of B_k's center. Therefore, we are looking for $(x, y) \in R$ such that if we locate there the center of B_k, the contribution to the cost of the current partial placement is minimized. Let $f(x, y)$ denote that contribution, then our problem can be stated as: minimize $f(x, y)$, subject to: $a \leq x \leq b$, $c \leq y \leq d$.

4.1. Squared Euclidean Distance (SED) Metric

Our problem is:

$$\text{minimize } f(x, y) = \sum_{i=1}^{p} w_i\left[(x + u_i - x_i)^2 + (y + v_i - y_i)^2\right] \quad (4.1)$$

$$\text{subject to: } a \leq x \leq b, \quad c \leq y \leq d \quad (4.2)$$

where w_i is the weight assigned to the net to whom the ith port belongs. Denoting $\xi_i = x_i - u_i$, $\eta_i = y_i - v_i$ results in

$$f(x, y) = \sum_{i=1}^{p} w_i\left[(x - \xi_i)^2 + (y - \eta_i)^2\right]. \quad (4.3)$$

Expression (4.3) has the following physical interpretation. If we look at every w_i as the elasticity constant of a spring that connects the point (x_i, y_i) with the point $(x + u_i, y + v_i)$, then f is the potential energy of a system whose components are the already placed blocks and B_k. This energy is added to the system, and we search for an (x, y) which minimizes this addition. This interpretation is related in some way to the *directed forces* algorithms for placing IC's on a PCB or standard blocks in a VLSI chip [11]. A feasible PFR is identified with an empty slot on the PCB or an empty space in a row of standard blocks.

Equations (4.1)–(4.2) are a convex program and can be solved analytically [1]. Let (x^*, y^*) solve (4.1)–(4.2), then x^* and y^* can be found separately. For x^* there exist three possibilities:

1) If
$$x^* = \frac{\sum_{i=1}^{p} w_i \xi_i}{\sum_{i=1}^{p} w_i}$$
satisfies $a < x^* < b$, then this x^* solves the problem, else
2) If $\sum_{i=1}^{p} w_i(a - \xi_i) \geq 0$ then $x^* = a$ solves the problem, else

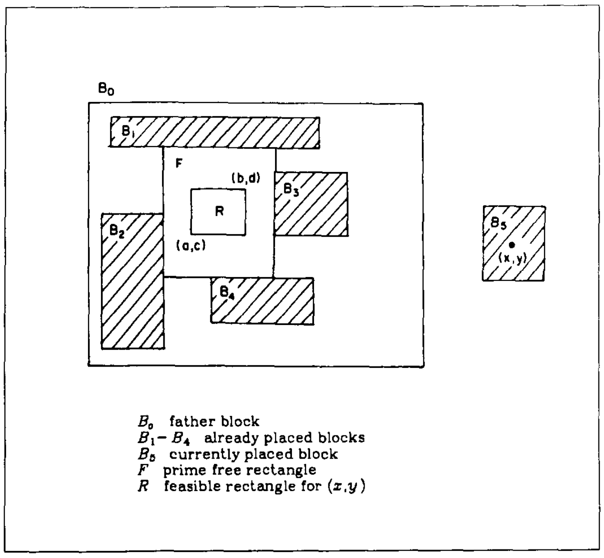


Fig. 6. The feasible rectangle for locating the center of the new block.

3) The sum $\sum_{i=1}^{p} w_i(b - \xi_i) \leq 0$, and $x^* = b$ solves the problem.

The value of y^* is obtained in a similar way.

Before proceeding to the next objective function, let us give another interpretation to the SED cost which naturally leads to a new selection strategy. Assume that we have n nets N_j, $1 \leq j \leq n$, each one has n_j ports. Let (x_k^j, y_k^j) be the position of a port that belongs to N_j within the father block, $1 \leq k \leq n_j$. Then, the cost of a complete placement is given by

$$\text{SED cost} = \sum_{j=1}^{n} w_j \sum_{k,l=1}^{n_j} \left[(x_k^j - x_l^j)^2 + (y_k^j - y_l^j)^2\right]. \quad (4.4)$$

Define the "center of gravity" $(\bar{x}^j, \bar{y}^j)$ of a net N_j to be

$$\bar{x}^j = \frac{1}{n_j} \sum_{k=1}^{n_j} x_k^j, \quad \bar{y}^j = \frac{1}{n_j} \sum_{k=1}^{n_j} y_k^j, \qquad 1 \leq j \leq n.$$

Assigning a unit mass to every port, we can define the "moment of inertia" I_j, of N_j as

$$I_j = \sum_{k=1}^{n_j} \left[(x_k^j - \bar{x}^j)^2 + (y_k^j - \bar{y}^j)^2\right], \qquad 1 \leq j \leq n.$$

It can be shown that the minimization of (4.4) is equivalent to the minimization of the total sum of the moments of inertia of the nets, where every moment is weighted by the factor $n_j w_j$, namely,

$$\text{SED cost} = \sum_{j=1}^{n} n_j w_j I_j. \quad (4.5)$$

From (4.5) we see that in order to minimize the SED cost, the ports must be concentrated around the center of gravity of the nets to whom they belong. This motivates a new strategy for selecting the next block to be placed, namely: **select a block from B'' such that nets with larger product of weight and cardinality (number of participating ports) are completed first.**

4.2. Rectilinear Distance (RD) Metric

The problem is:

$$\text{minimize } f(x, y) = \sum_{i=1}^{p} w_i(|x - \xi_i| + |y - \eta_i|) \quad (4.6)$$

$$\text{subject to: } a \le x \le b, \quad c \le y \le d. \quad (4.7)$$

f is continuous in R, but at the points (ξ_i, y) and (x, η_i) it is not differentiable. Consequently, the technique used in the SED case is not applicable here and another approach is required. Let us define $\xi_0 = a$, $\xi_{p+1} = b$, $\eta_0 = c$ and $\eta_{p+1} = d$. Now sort $\{\xi_i\}_{i=0}^{p+1}$ and $\{\eta_i\}_{i=0}^{p+1}$ in increasing order and let $\{\alpha_i\}_{i=0}^{p+1}$ and $\{\beta_i\}_{i=0}^{p+1}$ be the sorted lists, respectively. Indexes s and t, satisfying $s < t$, exist such that $\alpha_s = a$ and $\alpha_t = b$. In the same manner, indexes q and r, satisfying $q < r$, exist such that $\beta_q = c$ and $\beta_r = d$. Let $R_{k,l}$ be a rectangle defined as follows:

$$R_{k,l} = \{(x, y) | \alpha_k \le x \le \alpha_{k+1}, \beta_l \le y \le \beta_{l+1}\},$$
$$s \le k \le t - 1, \quad q \le l \le r - 1. \quad (4.8)$$

Clearly,

$$R = \bigcup_{k,l} R_{k,l}. \quad (4.9)$$

The functions $x - \xi_i$ and $y - \eta_i$, $1 \le i \le p$, do not change sign on $R_{k,l}$ and hence f is linear there and achieves its minimum at one of the four corner points (α_k, β_l), (α_{k+1}, β_l), (α_k, β_{l+1}) or $(\alpha_{k+1}, \beta_{l+1})$. It follows from (4.9) that f achieves its minimum value on R among the $(t - s + 1) \times (r - q + 1)$ points

$$(\alpha_k, \beta_l), \quad s \le k \le t, \quad q \le l \le r.$$

It is easy to solve (4.6)–(4.7) by observing that the function $f(x, y)$ defined in (4.6) is a positive combination of the convex functions $|x - \xi_i| + |y - \eta_i|$, $1 \le i \le p$. Hence, f is convex too, and it has one minimum in R, which is global. It follows that the minimum of f can be found by searching on the grid points (α_k, β_l), $s \le k \le t$, $q \le l \le r$, requiring $O[(t - s) + (r - q)]$ evaluations of f.

4.3. Euclidean Distance (ED) Metric

The problem is:

$$\text{minimize } f(x, y) = \sum_{i=1}^{p} w_i \sqrt{(x - \xi_i)^2 + (y - \eta_i)^2}$$

$$\text{subject to: } a \le x \le b, \quad c \le y \le d. \quad (4.11)$$

The function $f(x, y)$ is convex in R, but the program (4.10)–(4.11) cannot be solved analytically and some search technique is required. This search should not rely on the derivatives of f since they do not exist at the points (ξ_i, y) and (x, η_i). However, any search technique that necessitates only the calculation of f guarantees the global minimum.

4.4. Comparison of the Objective Functions

An interesting question is which one of the above three metrics is preferable. To answer this question we must consider the quality of the final placement and the computation time required to achieve it. Experimental results show that a "good" placement is "good" in all the three metrics. Table I shows some results of a 16 block and 16 net example. We see that even when optimization is done by using RD or ED metrics, the results obtained by using the SED metric are the best. Table II shows some results for random placements of the above example obtained by random sequence of block selection. We observe that the ratio between the RD and ED costs is almost constant. This ratio is $4/\pi$ and it can be explained as follows. Assume that every block is placed randomly. For every pair of ports which belong to the same net, the ratio between their RD and ED distance is $|\cos \gamma| + |\sin \gamma|$, where $\gamma \in [0, 2\pi]$. Now, if we look at γ as a random variable in $[0, 2\pi]$, then the mean of $|\cos \gamma| + |\sin \gamma|$ is given by

$$\frac{1}{2\pi} \int_0^{2\pi} (|\cos \gamma| + |\sin \gamma|) \, dy = \frac{4}{\pi}$$

which is the observed ratio.

Consider now the ratio between SED and RD costs. Let the net N_j have n_j ports. Assuming a unit weight ($w_i = 1$) for every net, we can interpret a complete placement as a point in a $2 \sum_{j=1}^{n} n_j (n_j - 1)$ dimensional vector space. A component of that vector is $(x_k^j - x_l^j)$ or $(y_k^j - y_l^j)$, where (x_k^j, y_k^j) and (x_l^j, y_l^j) are the coordinates of two ports belonging to the same net N_j. In the above notations, the RD cost and the square root of the SED cost are the l_1 and l_2 norms, respectively. It is known that any two norms on a finite dimensional vector space are equivalent [3]. Hence, there exist two constants which bound the ratio between the RD cost and the square root of the SED cost from above and below. But unlike the case of RD and ED costs, these constants depend on the specific problem.

We have seen that the SED and RD minimization problems have an analytical solution whose evaluation is very fast. In contrast, the ED minimization problem necessitates the use of a search technique which might be time consuming. Since the above analysis shows that all three metrics are somehow "equivalent", we recommend to adopt either the SED or the RD objective function and reject the ED one.

V. Conclusions and Further Research

The constructive approach to the placement of general blocks in VLSI has been discussed in this paper. Five se-

TABLE I
COST OF A 16 BLOCK 16 NET OPTIMAL PLACEMENT

Metric Used for Minimization / Cost in Terms of	SED	RD	ED
SED	1.46×10^9	1.72×10^9	1.86×10^9
RD	2.99×10^6	3.02×10^6	3.29×10^6
ED	2.45×10^6	2.49×10^6	2.67×10^6

TABLE 2
COST OF A 16 BLOCK 16 NET RANDOM PLACEMENT

Random Placement / Cost in Terms of	No. 1	No. 2	No. 3	No. 4
SED	3.70×10^9	3.61×10^9	4.76×10^9	4.06×10^9
RD	5.25×10^6	5.01×10^6	6.04×10^6	5.58×10^6
ED	4.14×10^6	3.99×10^6	4.76×10^6	4.42×10^6
$\frac{\text{RD cost}}{\text{ED cost}} \frac{\pi}{4}$	0.996	0.986	0.997	0.992

lection strategies were presented, three of which have been implemented in a placement package that was developed. In an example we saw that a selection strategy which combines connectivity and area considerations, yields better final results than when either area or connectivity are considered alone. To determine which selection strategy is preferable in general, we established a probabilistic methodology allowing us, for a given objective function, to compare various selection strategies. This methodology can be further improved by deriving a law of distribution for the placement cost. We have also shown that the complexity of optimally placing a block in the fragmented available area can be significantly simplified by decomposing the area into prime free rectangles.

The major part of this paper is devoted to the analysis of objective functions based on various net metrics. The resulting optimization problems are convex programs whose global minima (for the cost contributed by the currently placed block, and not for the whole problem) are guaranteed and can be found by using very efficient computational schemes. It has been shown that the final results in all the metrics are highly correlated. Consequently, the squared Euclidean and the rectilinear metrics, which are computationally much simpler, are preferred.

ACKNOWLEDGMENT

The Design Center support of this research and fruitful discussions with A. Levi are gratefully acknowledged.

REFERENCES

[1] M. Avriel, *Nonlinear Programming: Analysis and Methods.* Englewood Cliffs, NJ: Prentice-Hall, 1976.

[2] J. P. Blanks, "Near-optimal placement using a quadratic objective function," in *Proc. 22nd Design Automation Conf.*, pp. 609-615, 1985.

[3] A. L. Brown and A. Page, *Elements of Functional Analysis.* New York: Van Nostrand, 1970.

[4] M. A. Breuer Ed., *Design Automation of Digital Systems: Theory and Techniques.* Englewood Cliffs, NJ: Prentice-Hall, 1972.

[5] C. H. Heider, "An N-step 2-variable search algorithm for the component placement problem," *Nav. Res. Logist. Q.*, vol. 20, pp. 699-724, 1973.

[6] D. P. LaPotin, "A global floor-planning approach for VLSI design," Ph.D dissertation, Dept. of EE and CE Carnegie-Mellon Univ., 1985.

[7] E. Lodi et al., "On two-dimensional data organization I," *Ann. Soc. Polonae, Series IV: Fundamenta Informaticae II*, 1979, pp. 211-226.

[8] A. Naamad et al., "On the maximum empty rectangle problem," *Discrete Applied Math.*, vol. 8, 1984, pp. 267-277.

[9] National semiconductor Tel-Aviv, CHIPLAN user's guide, 1984.

[10] T. Ohtsuki Ed., *Layout Design and Verification.* New York: North-Holland, 1986.

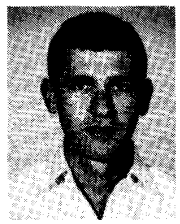
[11] N. R. Quinn, Jr., "The placement problem as viewed from the physics of classical mechanics," in *12th Design Automation Conf. Proc.*, pp. 173-178, 1975.

[12] L. Sha and R. W. Dutton, "An analytical algorithm for placement of arbitrarily sized rectangular blocks," in *Proc. 22nd Design Automation Conf.*, pp. 602-608, 1985.

[13] J. Soukup, "Circuit layout," *Proc. IEEE*, vol. 69, pp. 1281-1304, 1981.

[14] D. J. Ullman, *Computational Aspects of VLSI*, Computer Science Press, 1984.

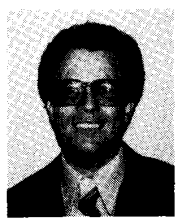
*



Shmuel Wimer received the B.Sc. and M.Sc. degrees from Tel-Aviv University, Tel-Aviv, Israel, in mathematics, in 1977 and 1980, respectively.

From 1978 to 1981 he was with the Israeli Aircraft Industry. From 1981 to 1985 he was with National Semiconductor Design Center in Tel-Aviv. Since 1985 he is with the IBM Scientific Center in Haifa, Israel. He is currently working towards his D.Sc degree in the Department of Electrical Engineering at the Technion - Israel Institute of Technology. His current research interest is algorithms for layout of VLSI circuits and systems.

*



Israel Koren (S'72-M'76) received the B.Sc. (cum laude), M.Sc. and D.Sc. degrees from the Technion - Israel Institute of Technology, Haifa, all in electrical engineering, in 1967, 1970, and 1975, respectively.

Since 1979 he is with the Departments of Electrical Engineering and Computer Science at the Technion - Israel Institute of Technology, where he became the Head of the VLSI Systems Research Center in 1985. Previously he has held positions with the University of California, Santa Barbara and the University of Southern California, Los Angeles. In 1982 he was on sabbatical leave with the University of California, Berkeley. Currently he is a Visiting Professor at the University of Massachusetts, Amherst. He has been a consultant to National Semiconductor, Israel, in architecture of microprocessors and high-speed algorithms for arithmetic operations, in 1984-1986, to Tolerant Systems, San Jose, CA, in architecture of fault-tolerant distributed computer systems in 1983, and to ELTA, Electronics Industries, Israel, in architecture of parallel signal processors in 1981-1982. His current research interests are fault-tolerant VLSI and WSI architectures, models for yield and performance, floor-planning of VLSI chips and computer arithmetic.