A New Approach to the Pin Assignment Problem*

Xianjin Yao[†] *Massaki Yamada*[††] *C. L. Liu*

Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, IL. 61801 U.S.A.

Abstract

In this paper we study a pin assignment problem for macro-cells which is motivated by the goal of integrating the placement and routing steps in the physical design of VLSI circuits. We assume that the macro-cells have already been placed. In the mean time, we assume that the design of the macro-cells is still "soft" in that although the pins in a cell have a fixed relative order, they can be shifted around the boundary of the cell. A new algorithm was developed to determine the optimal shiftings of the pins so that a weighted sum of the lengths of the connecting wires is minimum. Good experimental results were obtained.

1. Introduction

An important step in the general problem of placement and routing of macro-cells is the assignment of pins to positions on the boundaries of the macro-cells. We assume that in the placement and routing phase the design of a macro-cell is still "soft" in that circuits or I/O pads inside of the macro-cell can be rearranged so that positions of connection pins on the boundary can be rearranged accordingly. A major reason for such rearrangement is the potential of obtaining better routing results [1] [2]. Also, rearrangement of pins can also facilitate the routing of special nets such as the power/ground nets [3]. The most general form of our problem is to assume that placement of the macro-cells has not been committed and that pins in the cells can be rearranged in any arbitrary fashion. Thus, we want to place the cells and to position the pins on the boundaries of the cells so that total wire length (or some other measure) will be minimised. There are several restricted versions of the problem: We may assume that the macro-cells have already been placed and can no longer be moved. We may also assume that the relative positions of the pins in each cell are fixed. In other words, they must be arranged according to a fixed order. In this paper, we shall present a model which we believe is suitable to handle the general problem. We shall also present a method of solution for the case with fixed placement for the cells and fixed relative positions for the pins. Our experimental results are, indeed, most promising.

2. The Model

We assume that a fixed placement for a set of rectangular macro-cells was given. We also assume that the relative positions of the pins along the boundary of each macro-cell has been predetermined. We shall represent a macro-cell by an inscribing circle with the diameter of the circle equal to the smaller of the height and the width of the rectangular cell. (Clearly, for a macro-cell with an aspect ratio that is close to 1, an inscribing circle is a good approximate representation. In Sec. 6, we shall discuss how adjustments in the final solution can be made for macro-cells with aspect ratios that are significantly larger than or smaller than 1.) We shall place the pins on the circumference of the circle representing a macro-cell with the order of the pins and the distances between them fixed. Fig. 1(a) shows an example of a placement of cells and Fig. 1(b) shows the corresponding representation. Thus, the pin assignment problem can be modeled as that of rotating the circles (equivalently, rotating the pins on the circumferences of the circles) so that the total length of all connecting wires, or the value of some other cost function, is minimised. Fig. 1(c) shows a new assignment of pin positions.

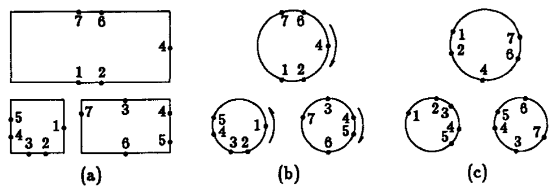


Fig. 1

Using such a model, we can formulate the pin assignment problem as a minimisation problem as follows: Suppose that there are n macro-cells. For macro-cell i, let x_i and y_i denote the coordinates of the center of the circle representing the macro-cell. Suppose that the circle is rotated by an angle θ_i (with respect to an initial reference placement.) It follows that the positions of all the pins in cell i can be expressed in terms of x_i, y_i, and θ_i. We can then compute the distance between every two pins that belong to the same net and are in different cells. The sum of these distances will be the total wire length which can be minimised with respect to the variables θ_1, θ_2, ..., θ_n. In general, instead of the total wire length, the cost function may assume a more general form of a weighted sum of wire lengths. For example, we can adjust the weighting factors so that longer wires will have larger weights. In this case, the lengths of long connecting wires will be reduced in the final solution. Also, we can adjust the weighting factors so that a selected subset of the connecting wires (such as power connection) will have larger weights. In this case, more emphasis will be placed on the reduction of the total length of this subset of connecting wires. We can also adjust the weighting factors so that connecting wires among some of the macro-cells will have larger weights in order to assure that the total length of connecting wires aomng these macro-cells will be reduced.

* This work was partially supported by the National Science Foundation under grant MIP 87-03273, by the Semiconductor Research Corporation under contract 87-DP-109, by a grant from the General Electric Company.

† Xi'an Jiaotong University, People's Republic of China.

†† Toshiba Corporation, Japan.

CH2540-3/88/0000/0566$01.00 © 1988 IEEE

So that we can handle such a general formulation of the problem, we propose a new approach based on a physical analogy of rotational torques acting on the circles.

3. A Physical Analogy

We use a physical analogy in which a connecting wire between two pins in two different circles exerts a rotational torque on each of the two circles. Consider the two macro-cells i and j represented by the two circles shown in Fig. 2. Let p and q denote two pins that belong to the same net. Imagine that there is an attractive force, which will be denoted F_{pq} or F_{qp}, between the two pins along the line joining the pins. Since the centers of the circles are held fixed, the attractive force produces a rotational torque equal to $r_i F_{pq} \sin\gamma_{i_{pq}}$ that tends to rotate circle i in the clockwise direction and a rotational torque equal to $r_j F_{qp} \sin\gamma_{j_{qp}}$ that tends to rotate circle j also in the clockwise direction, where r_i and r_j are the radii of circles i and j, respectively. The total rotational torque acting on each circle is equal to the sum of the torques due to the connecting wires at the circle. According to well-known principles of Physics, at *equilibrium*, the angular positions of the circles will be such that the total rotational torque acting on each of them is zero. Such an analogy is indeed the basis of our general formulation of the pin assignment problem. Intuitively, the total wire length will be minimized at equilibrium. (This is indeed the case as will be shown in Appendix 1.) However, by taking advantage of the flexibility in specifying the magnitude of the force F_{pq}, we can handle a very general class of cost functions.

For the magnitude of the attractive force F_{pq}, we shall depart from the formulae we found in a Physics textbook and assume that the magnitude is given by the formula

$$F_{pq} = c_{pq} * [(\Delta x_{pq})^2 + (\Delta y_{pq})^2]^{h_{pq}} \quad (1)$$

where the proportional constant c_{pq} and the power h_{pq} can, in general, be chosen to assume different values for different pairs of pins. It is indeed such flexibility in choosing the values of c_{pq} and h_{pq} that provides us with the capability to handle cost functions of a very general form. For example, by choosing c_{pq} to be 1 and h_{pq} to be the same value h for all pairs of pins p and q, we can control the relative weights of long connecting wires by changing the value of h. If h is chosen to have a large value (for example, 4), it means that a long connecting wire will exert a large rotational torque on the circles it connects. Consequently, a long connecting wire will contribute a larger weighted sum to the value of the cost function. In other words, choosing a large value for h amounts to accentuate the effect of far away cells. On the other hand, if h is chosen to have a small value (for example, 1), it means that the difference in the magnitudes of the rotational torques due to long and short connecting wires will not be as significant. In fact, if h is chosen to be 0, it means that the effect of all connecting wires, long and short, will be the same. As will be seen in Appendix 1, this indeed leads to the case that at equilibrium, the total (unweighted) connecting wire length is minimized. In Sec. 7, we shall present experimental results to demonstrate the effect of different choices of the value of h.

4. Mathematical Details

We shall fill in some more details in the formulae for computing the overall rotational torque acting on a circle. Note that in Fig. 2,

θ_i is the angular position (with respect to $o_i I_0$).
θ_j is the angular position (with respect to $o_j J_0$).
α_{i_p} is the angle between $o_i I_0$ and $o_i p$, $0 \le \alpha_{i_p} \le 2\pi$.
α_{j_q} is the angle between $o_j J_0$ and $o_j q$, $0 \le \alpha_{j_q} \le 2\pi$.
$\beta_{i_{pq}}$ is the angle between the horizontal line and pq, $-\pi \le \beta_{i_{pq}} \le +\pi$.
$\beta_{j_{qp}}$ is the angle between the horizontal line and qp, $-\pi \le \beta_{j_{pq}} \le +\pi$.
$\gamma_{i_{pq}}$ is the angle between $o_i p$ and pq, $-\pi/2 \le \gamma_{i_{pq}} \le +\pi/2$.
$\gamma_{j_{qp}}$ is the angle between $o_j q$ and qp, $-\pi/2 \le \gamma_{j_{pq}} \le +\pi/2$.

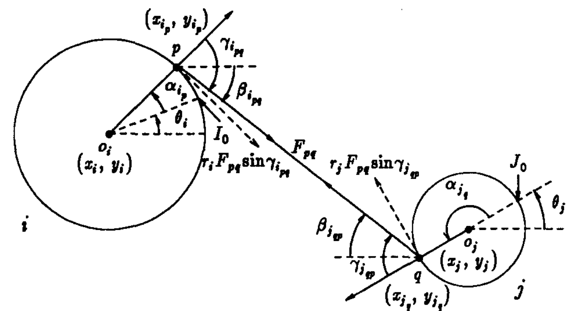


Fig. 2

Clearly,

$$\gamma_{i_{pq}} = \beta_{i_{pq}} - (\theta_i + \alpha_{i_p})$$
$$\gamma_{j_{qp}} = \beta_{j_{qp}} - (\theta_j + \alpha_{j_q})$$

Let x_{i_p}, y_{i_p} be the coordinates of point p, x_{j_q}, y_{j_q} be the coordinates of point q, and Δx_{pq}, Δy_{pq} be the horizontal and vertical distances between p and q, respectively. Since

$$x_{i_p} = r_i * \cos(\theta_i + \alpha_{i_p}) + x_i$$
$$y_{i_p} = r_i * \sin(\theta_i + \alpha_{i_p}) + y_i$$
$$x_{j_q} = r_j * \cos(\theta_j + \alpha_{j_q}) + x_j$$
$$y_{j_q} = r_j * \sin(\theta_j + \alpha_{j_q}) + y_j$$
$$\Delta x_{pq} = x_{j_q} - x_{i_p}$$
$$\Delta y_{pq} = y_{j_q} - y_{i_p}$$

we have:

$$\beta_{i_{pq}} = \begin{cases} \arctan(\Delta y_{pq}/\Delta x_{pq}) & \text{if } \Delta x_{pq} > 0 \\ \arctan(\Delta y_{pq}/\Delta x_{pq}) + \pi & \text{if } \Delta x_{pq} < 0 \text{ and } \Delta y_{pq} > 0 \\ \arctan(\Delta y_{pq}/\Delta x_{pq}) - \pi & \text{if } \Delta x_{pq} < 0 \text{ and } \Delta y_{pq} < 0 \\ \pi/2 & \text{if } \Delta x_{pq} = 0 \text{ and } \Delta y_{pq} > 0 \\ -\pi/2 & \text{if } \Delta x_{pq} = 0 \text{ and } \Delta y_{pq} < 0 \\ 0 & \text{if } \Delta y_{pq} = 0 \text{ and } \Delta x_{pq} > 0 \\ \pi & \text{if } \Delta y_{pq} = 0 \text{ and } \Delta x_{pq} < 0 \end{cases}$$

$$\gamma_{i_{pq}} = \beta_{i_{pq}} - (\theta_i + \alpha_{i_p})$$

It follows that the rotational torque acting on cell i due to the connecting wire between p and q, $T_{i_{pq}}$, can be computed as

$$T_{i_{pq}} = r_i * F_{pq} * \sin\gamma_{i_{pq}}$$

Consequently, the total rotational torque acting on cell i is given by

$$T_i(\theta_1, \theta_2, \ldots, \theta_n) = \sum_{p \in i \; q \notin i} T_{i_{pq}} = r_i * \sum_{p \in i \; q \notin i} F_{pq} * \sin\gamma_{i_{pq}}$$

Thus, at equilibrium, we have the equations:

$$T_i(\theta_1, \theta_2, \ldots, \theta_n) = r_i * \sum_{p \in i \; q \notin i} F_{pq} * \sin\gamma_{i_{pq}} = 0 \quad i = 1, 2, \ldots, n \quad (2)$$

which will be referred as the torque equations.

5. Iterative Solution of the Torque Equations

From the torque equations in (2) we can solve for the values of the angular positions $\theta_1, \theta_2, \ldots, \theta_n$. Although, in principle, we can employ the classical Newton–Jacobi iterative method of solution [4, 5], computational considerations led us to a more efficient iterative procedure which we shall present in this section. In the classical Newton–Jacobi iterative method, we need to evaluate the inverse of the Jacobian matrix in each iteration. For a system with 5,000 macro–cells, this amounts to computing the inverse of of a 5,000 × 5,000 matrix. (The largest example we shall present in Sec. 7 contains 6,400 macro–cells.) For our pin assignment problem, the number of non–zero entries in the Jacobian matrix is approximately equal to the number of connceting wires. (The largest example we tested has over 100,000 connecting wires.) Thus, even though there are more efficient methods for evaluating the inverses of sparse matrices, both the computation time and the accuracy required are still quite formidable for large problems. We employ instead a set of *decoupled* equations in our iterative procedure. Intuitively, such a set of decoupled equations suggests itself naturally when we examine the physical analogy we introduced earlier.

We begin with the initial solutions $\theta_1^{(0)}, \theta_2^{(0)}, ..., \theta_n^{(0)}$. Let $\theta_1^{(m)}, \theta_2^{(m)}, ..., \theta_n^{(m)}$ denote the solutions obtained in the m^{th} iteration. We shall compute $\theta_1^{(m+1)}, \theta_2^{(m+1)}, ..., \theta_n^{(m+1)}$ according to the following equations:

$$\theta_i^{(m+1)} = \theta_i^{(m)} + \frac{T_i(\theta_1^{(m)}, \theta_2^{(m)}, ..., \theta_n^{(m)})}{r_i * \overline{F}_i^{(m)} * c_i^{(m)}} \qquad i = 1, 2, ..., n \qquad (3)$$

Note that the equations in (3) are indeed based on the physical intuition that changes in the value of the angular position θ_i in successive iterations depend both on the sign and the magnitude of the rotational torque acting on cell i, $T_i(\theta_1, \theta_2, ..., \theta_n)$. $\overline{F}_i^{(m)}$ in (3) is the average force acting on circle i which is computed according to the formula

$$\overline{F}_i^{(m)} = (\sum_{p \in i\, q \notin i} F_{pq}^{(m)}) / d$$

where $F_{pq}^{(m)}$ is the value of F_{pq} computed in the m^{th} iteration, and d is the total number of connecting wires at cell i. $c_i^{(m)}$ in (3) is the value of a control parameter c_i in the m^{th} iteration which is computed according to the formula

$$c_i^{(m)} = \begin{cases} 0.85 * c_i^{(m-1)} & \text{if } T_i(\theta^{(m)}) * T_i(\theta^{(m-1)}) > 0^{\dagger} \\ 1.25 * c_i^{(m-1)} & \text{if } [T_i(\theta^{(m)}) * T_i(\theta^{(m-1)}) < 0 \text{ and} \\ & |T_i(\theta^{(m-1)})| - |T_i(\theta^{(m)})| < 0.6 * |T_i(\theta^{(m-1)})|\,] \\ & \text{or } [\; |\Delta\theta_i^{(m)}| > \pi\,] \\ c_i^{(m-1)} & \text{otherwise} \end{cases}$$

The initial value $c_i^{(0)}$ is set to be $(k * d)^{1/2}$ where k is the number of pins in cell i. Note that the value of $c_i^{(m)}$ is decreased if $T_i(\theta^{(m)})$ and $T_i(\theta^{(m-1)})$ are of the same sign, and the value of $c_i^{(m)}$ is increased if $T_i(\theta^{(m)})$ and $T_i(\theta^{(m-1)})$ are of opposite signs. In the former case we want to speed up the convergence, while in the latter case we need to move slower in searching for the cross over point at which $T_i(\theta^{(m)})$ is equal to 0.

Finally, the termination criterion for the iterative procedure is :

$$|\,\theta_i^{(m+1)} - \theta_i^{(m)}\,| < 0.003 \qquad \text{for each cell}$$

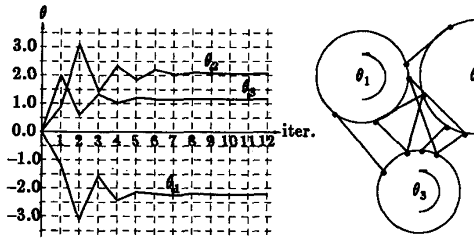
or

$$|\,D^{(m+1)} - D^{(m)}\,| \,/\, D^{(m)} < 0.00005 \qquad \text{for 3 succesive iterations}$$

where $D^{(m+1)}$ and $D^{(m)}$ are the values of the total wire length computed in the $(m+1)^{st}$ and the m^{th} iterations, respectively.

The rate of convergence of our iterative method is very good. For all the examples we tested, including some very large examples with thousands of cells and tens of thousands of connecting wires, the number of iterations is no more than 30. Fig. 3 shows an example in which convergence of the angular positions toward their values at equilibrium is demonstrated. Note that the system reached equilibrium after about 10 iterations.

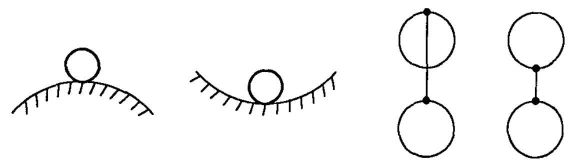
† We use $T_i(\theta^{(m)})$ to denote $T_i(\theta_1^{(m)}, \theta_2^{(m)}, ..., \theta_n^{(m)})$.



(a) (b)

Fig. 3

It is well–known that when a physical system is at equilibrium, it might be in a state of *meta–stable equilibrium*, or it might be in a state of *stable equilibrium*. A physical system is said to be in a state of meta–stable equilibrium if any minor pertubation will lead the system away from equilibrium. On the other hand, a physical system is said to be in a state of stable equilibrium if it always returns to equilibrium after any minor perturbation. A simple illustrative example is shown in Fig. 4(a). For the pin assignment problem, similar phenomena are illustrated in Fig. 4(b): So that we will not accept a meta–stable equilibrium solution as the final solution, our iterative procedure always goes on for one more step of iteration (corresponding to a minor perturbation) when the termination condition is satisfied. If the total wire length continues to decrease, we realise that we were at a state of meta–stable equilibrium and the iterative procedure continues. On the other hand, if the total wire length increases instead, we realise that we were at a state of stable equilibrium and the iterative procedure terminates.



meta–stable stable meta–stable stable

(a) (b)

Fig. 4

6. Other Considerations

In our physical model, two–terminal nets and multi–terminal nets are handled in the same fashion. For a net that has more than one pin on a circle, we shall replace them by a "representative" pin in the computation of rotational torques. Clearly, the position of the representative pin should be the "average" of the positions of the pins that belong to the same net as illustrated in Fig. 5.

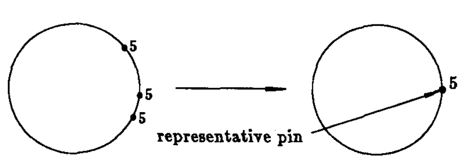
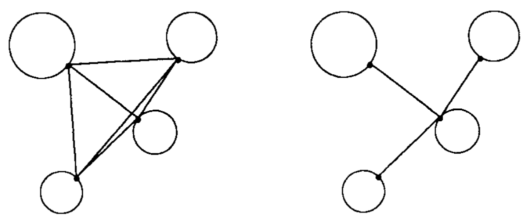


Fig. 5

Note that in computing the total rotational torque acting on a cell according to the torque equations the number of terms is equal to the number of wires connecting the pins in the cell to pins in other cells. Consequently, the total computational complexity is proportional to the total number of connecting wires. For a two–terminal net, there is one connecting wire between the two pins. On the other hand, multi–terminal nets can be handled in two different ways. In the first way, we assume that there is a connecting wire between every two pins in the net as illustrated in Fig. 6(a). Thus, for a k–terminal net, the number of connecting wires is $k(k-1)/2$. In the second way, we determine a minimum spanning tree for the centers of the circles, and assume that there is a connecting wire between two pins in the net only if there is a corresponding edge in the minimum spanning tree as illustrated in Fig. 6(b). Thus, for a k–terminal net, the number of connecting wires is only $k-1$. In both cases, once the connecting wires among the pins are determined, the rotational torques acting on the cells can be computed according to the torque equations. We have experimented with both ways to determine the rotational torques and found that the "spanning tree" representation of multi–terminal nets produces experimental results that are almost as good as that produced by the "complete graph" representation of multi–terminal nets. Experimental results to demonstrate the differences will be presented in Sec. 7.



(a) (b)

Fig. 6

After the angular positions of the circles at equilibrium are determined, we shall map the pin positions from the circumference of the inscribing circle back to the boundary of the rectaugular macro–cell. Such a mapping is most straight–forward when the aspect ratio of the rectangualr macro–cell is very close to 1. On the other hand, when the aspect ratio of a rectangular macro–cell is significantly larger than or smaller than 1, a procedure for modifying the final solution as illustrated in Fig. 7 is employed to reduce the total wire length. The details of such a procedure which can be use to solve a more general pin assignment problem in which the relative positions of the pins are not fixed will be reported in a forthcoming technical paper [6].

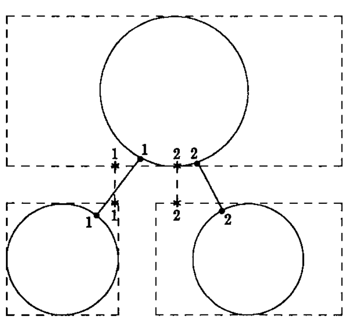
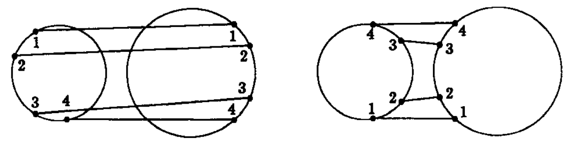


Fig. 7

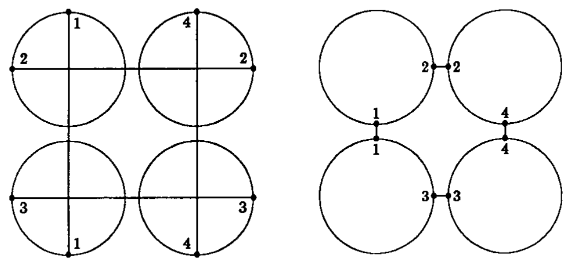
7. Experimental Results

There are four sets of results we want to present to justify our approach. The first set of results are extremely simple situations in which the optimal solutions are evident. Indeed, our algorithm was able to obtain the optimal solutions as expected. These examples are shown in Figs. 8–10.



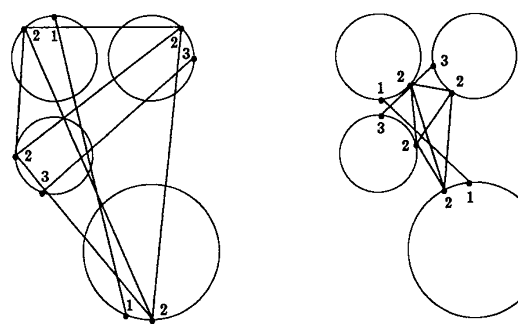
initial solution final solution

Fig. 8



initial solution final solution

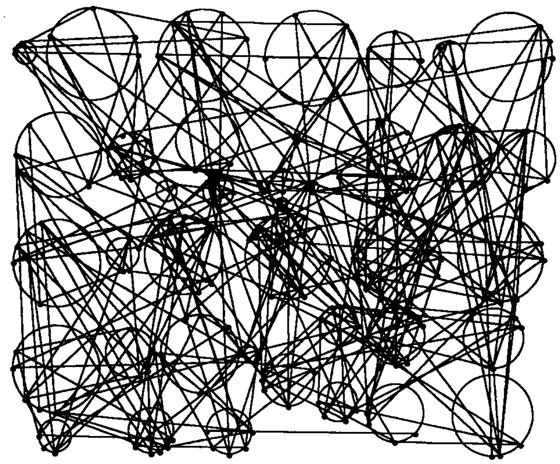
Fig. 9



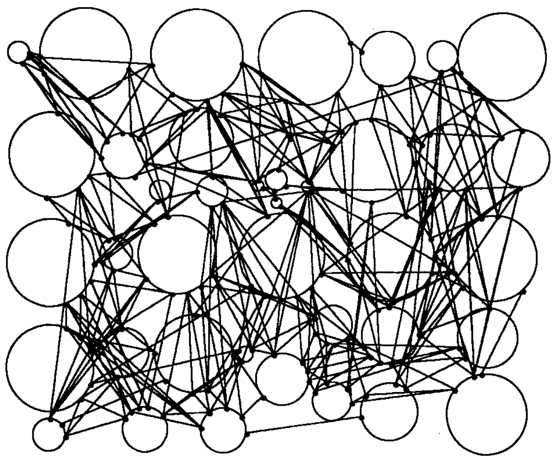
initial solution final solution

Fig. 10

The second set of results are large examples that illustrate the quality of the results produced by our algorithm as well as the speed of computation. Fig. 11 shows an arbitrary initial solution and the final solution obtained by our algorithm for a problem with 40 cells. Table 1 is a summary of results for examples with number of cells ranging from 100 to 6,400. In the column *Wire Length*, *total* is the total wire length, and *max* is the sum of the lengths of the 60 or top 10% longest connecting wires. Note that computation time for the largest example (6,400 cells, 9,745 nets, 60,845 connecting wires, and 38,304 pins) is less than 10 minutes on a Gould PN9050 computer.



initial solution



final solution

Fig. 11

Input					Wire Length				CPU
No	cells	nets	wires	pins	total		max		time
					initial	final	init.	final	(sec.)
1	100	254	254	508	52,780	4,836	362	62	2.95
2	200	289	1,660	1,097	495,924	43,269	552	104	14.07
3	400	580	3,355	2,203	1,422,588	92,128	764	104	28.28
4	800	1,375	6,803	4,815	3,844,748	654,207	1,103	375	62.20
5	1,600	2,710	14,330	9,802	11,611,727	1,366,972	1,574	391	131.07
6	3,200	6,333	15,605	16,737	25,154,880	1,495,870	3,008	429	160.08
7	4,800	8,748	34,675	27,952	57,412,816	3,402,121	3,071	487	328.65
8	6,400	9,745	60,845	38,304	103,269,296	6,092,337	3,150	500	548.82

Table 1

The third set of results we want to present is the effect of the exponent h_{pq} in Eq. (1) on the lengths of connecting wires. Fig. 12 illustrates a situation in which the value of h_{pq} is the same for all connecting wires. Note that when h_{pq} is set to 0.0, the total wire length is the smallest. However, there are some relatively long wires. When h_{pq} is set to larger values, the value of the total wire length increases, while the lengths of the longer wires decrease. Fig. 13 illustrates a situation in which the value of h_{pq} for wires that belong to a particular net (net1) is held fixed while the value of h_{pq} for the other wires varies. Note how the lengths of the wires for net1 increases when the value of h_{pq} for the other wires increases. Table 2 is a summary of results for examples in which the value of h_{pq} is set to be the same, h, for all pairs of pins. Note how the total wire length increases when the value of h increases from 0.0 to 2.0. In the mean time, note how the value *max* decreases with increasing h.

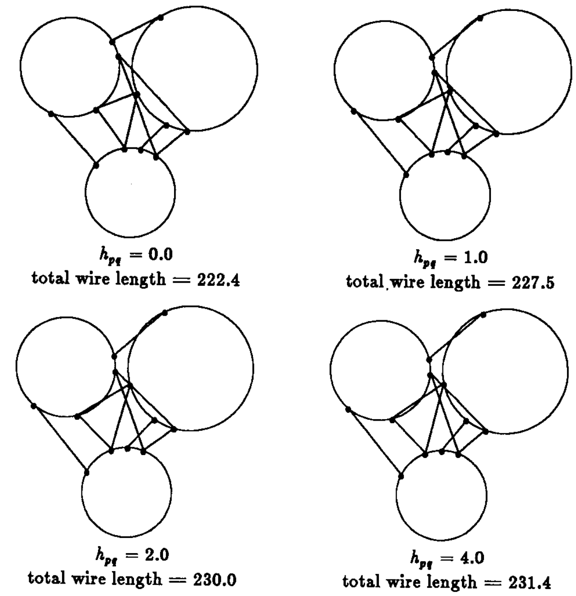


Fig. 12

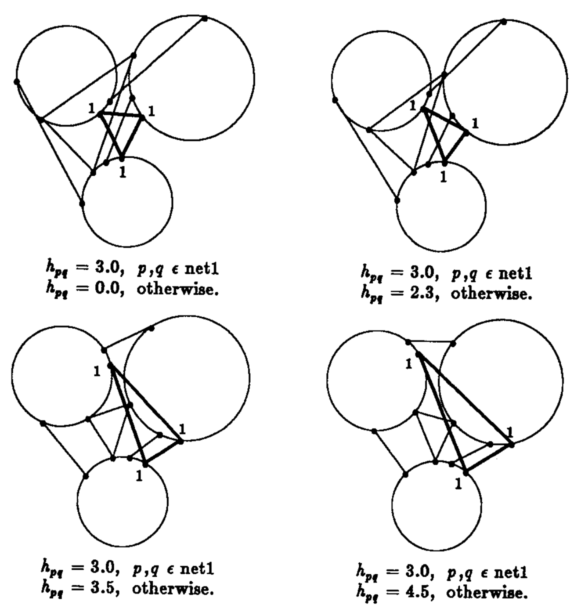


Fig. 13

Examples		*Input*				*Output*		*CPU time (sec.)*
No.	*h*	*cells*	*nets*	*wires*	*pins*	*total*	*max*	
0	0.0	3	5	9	12	222.4	43.1	0.03
	0.5					224.8	37.8	0.07
	1.0					227.6	35.5	0.07
	2.0					230.2	34.3	0.08
1	0.0	100	254	254	508	4,835.5	62.4	2.95
	0.5					4,914.2	59.2	3.07
	1.0					5,039.6	50.8	5.67
	2.0					5,133.8	48.7	4.12
2	0.0	200	289	1,660	1,097	43,268.6	104.0	14.07
	0.5					43,513.6	104.2	18.40
	1.0					43,807.3	104.3	13.30
	2.0					44,235.4	103.9	15.35
3	0.0	400	580	3,355	2,203	92,128.3	103.9	28.28
	0.5					92,711.3	101.7	31.07
	1.0					93,554.6	101.5	38.95
	2.0					94,542.8	101.5	50.07
4	0.0	800	1,375	6,803	4,815	650,207.2	374.9	62.20
	0.5					651,567.8	376.5	68.77
	1.0					653,544.8	375.4	69.03
	2.0					656,552.5	373.6	69.57
5	0.0	1,600	2,710	14,330	9,802	1,366,972.0	390.7	131.07
	0.5					1,369,910.0	384.7	107.02
	1.0					1,373,894.0	383.8	107.48
	2.0					1,379,513.0	382.6	143.22
6	0.0	3,200	6,333	15,605	16,737	1,495,870.0	429.4	160.08
	0.5					1,500,784.0	431.2	217.92
	1.0					1,506,357.0	431.1	173.37
	2.0					1,512,456.0	429.7	130.10
7	0.0	4,800	8,748	34,675	27,952	3,402,121.0	487.3	328.65
	0.5					3,410,676.0	484.2	446.35
	1.0					3,421,779.0	479.9	268.82
	2.0					3,436,168.0	479.4	267.08
8	0.0	6,400	9,745	60,845	38,304	6,092,337.0	499.6	548.82
	0.5					6,104,405.0	497.3	600.43
	1.0					6,121,580.0	490.6	488.72
	2.0					6,146,415.0	490.6	600.00

Table 2

The fourth set of results we want to present is a comparison between the "complete graph" and the "spanning tree" representations of multi–terminal nets. Table 3 is a summary of our results. For each example, we use the complete graph representation of multi–terminal nets to set up the torque equations and to obtain a final pin position assignment. For this final pin position assignment we compute the total wire length for the two cases that a multi–terminal net will be connected by a complete graph and by a spanning tree. We then turn around and use the spanning tree representation of multi–terminal nets to set up the torque equations and to obtain an final pin position assignment. For this final pin position assignment we also compute the total wire length for the two cases that a multi–terminal net will be connected by a complete graph and by a spanning tree. Note that the results obtained in these two ways are quite indistinguishable. In Table 3, in the column *Complete Graph* total wire length is computed for the case that a multi–terminal net is connected by a complete graph. The numbers not in parentheses are obtained when the torque equations are set up using the complete graph representation, and the numbers in parentheses are obtained when the torque equations are set up using the spanning tree representation. In the column *Spanning Tree* total wire length is computed for the case that a multi–terminal net is connected by a minimum spanning tree. The numbers not in parentheses are obtained when the torque equations are set up using the spanning tree representation, and the numbers in parentheses are obtained when the torque equations are set up using the complete graph representation. It is also interesting to observe that computation time for the spanning tree representation is not significantly smaller than computation time for the complete graph representation. The major reason is that computation time depends more on the number of cells (number of torque equations) than on the number of connecting wires (number of terms in the torque equations.)

8. Concluding Remarks

We hope the results presented above have demonstrated the power of a novel approach to the pin assignment problem. There are two extensions that we plan to carry out. The first extension is the case in which we assume the locations of the cells are fixed but pins of a cell may be positioned anywhere along the boundary of the cell. A second extension is to assume that the locations of the cells have not been fixed. In this case, there will be translational forces as well as rotational torques acting on the circles. Finally, we note that in many practical situations some of the pins are not movable, some of the pins must maintain some fixed relative positions with respect to other pins, while some other pins can be moved freely, it is our hope that the methodology developed in this paper will provide a general approach to the many possible variations of the pin assignment problem.

Appendix 1

We shall show that when h_{pq} in Eq. (1) is set to 0.0 for every pair of pins, total wire length is indeed minimized when the rotational torques acting on the circles are all 0. Because both wire lengths and rotational torques are additive in their respective computations, we only need to examine the simple case in which there is one connecting wire between two pins. Note that in Fig. 2,

$$\begin{aligned}\Delta x_{pq} &= x_{j_q} - x_{i_p}\\ &= r_j\cos(\theta_j+\alpha_{j_q}) + x_j - r_i\cos(\theta_i+\alpha_{i_p}) - x_i\\ &= r_j\cos(\theta_j+\alpha_{j_q}) - r_i\cos(\theta_i+\alpha_{i_p}) + (x_j-x_i)\end{aligned}$$

$$\begin{aligned}\Delta y_{pq} &= y_{j_q} - y_{i_p}\\ &= r_j\sin(\theta_j+\alpha_{j_q}) + y_j - r_i\sin(\theta_i+\alpha_{i_p}) - y_i\\ &= r_j\sin(\theta_j+\alpha_{j_q}) - r_i\sin(\theta_i+\alpha_{i_p}) + (y_j-y_i)\end{aligned}$$

Thus,

$$D_{pq} = D_{qp} = (\Delta x_{pq}^2 + \Delta y_{pq}^2)^{1/2}$$

To determine the angular positions of circles i and j at which D_{pq} is minimum, we compute the partial derivatives of D_{pq} with respect to θ_i and θ_j and set them to 0:

$$\frac{\partial D_{pq}}{\partial \theta_i} = \frac{r_i}{(\Delta x_{pq}^2+\Delta y_{pq}^2)^{1/2}} * [\Delta x_{pq}\sin(\theta_i+\alpha_{i_p}) - \Delta y_{pq}\cos(\theta_i+\alpha_{i_p})] = 0$$

$$\frac{\partial D_{qp}}{\partial \theta_j} = \frac{r_j}{(\Delta x_{qp}^2+\Delta y_{qp}^2)^{1/2}} * [\Delta x_{qp}\sin(\theta_j+\alpha_{j_q}) - \Delta y_{qp}\cos(\theta_j+\alpha_{j_q})] = 0$$

On the other hand, according to the torque equations in (2), we have

$$T_{i_{pq}} = r_i * \sin(\arctan\frac{\Delta y_{pq}}{\Delta x_{pq}} - (\theta_i+\alpha_{i_p})) = 0$$

$$T_{j_{qp}} = r_j * \sin(\arctan\frac{\Delta y_{qp}}{\Delta x_{qp}} - (\theta_j+\alpha_{j_q})) = 0$$

That is,

$$T_{i_{pq}} = r_i * [\sin(\arctan\frac{\Delta y_{pq}}{\Delta x_{pq}}) * \cos(\theta_i + \alpha_{i_p}) - \cos(\arctan\frac{\Delta y_{pq}}{\Delta x_{pq}}) * \sin(\theta_i + \alpha_{i_p})]$$

$$= \frac{r_i}{(\Delta x_{pq}^2 + \Delta y_{pq}^2)^{1/2}} * [\Delta y_{pq} \cos(\theta_i + \alpha_{i_p}) - \Delta x_{pq} \sin(\theta_i + \alpha_{i_p})]$$

$$T_{j_{qp}} = r_j * [\sin(\arctan\frac{\Delta y_{qp}}{\Delta x_{qp}}) * \cos(\theta_j + \alpha_{j_q}) - \cos(\arctan\frac{\Delta y_{qp}}{\Delta x_{qp}}) * \sin(\theta_j + \alpha_{j_q})]$$

$$= \frac{r_i}{(\Delta x_{qp}^2 + \Delta y_{qp}^2)^{1/2}} * [\Delta y_{qp} \cos(\theta_j + \alpha_{j_q}) - \Delta x_{qp} \sin(\theta_j + \alpha_{j_q})]$$

It follows that

$$T_{i_{pq}} = -\frac{\partial D_{pq}}{\partial \theta_i} \quad \text{and} \quad T_{j_{qp}} = -\frac{\partial D_{qp}}{\partial \theta_j}$$

Thus, in the general case, setting the rotational torques acting on the circles to 0 is equivalent to setting the partial derivatives of the total wire length with respect to the angular positions θ_1, θ_2,...,θ_n to 0. In other words, total wire length is minimised at equilibrium.

References

[1] P. Widmayer and C. K. Wong, "An Optimal Algorithm for the Maximum Alignment of Terminals", *Information Processing Letters, Vol. 20,* pp. 75–82, 1985.

[2] M. D. F. Schlag, L. S. Woo and C. K. Wong, "Maximising Pin Alignment by Pin Permutations", *INTEGRATION, the VLSI journal, 2,* pp. 279–307, 1984.

[3] Xiao-Ming Xiong and Ernest S. Kuh, "The Scan Line Approach to Power and Ground Routing", *Proc. ICCAD,* pp. 6–9, 1986.

[4] J. E. Dennis, Jr. and Robert B. Schnabel, "Numerical Methods for Unconstrained Optimisation and Nonlinear Equations", *Prentice-Hall, Inc.,* 1983.

[5] J. M. Ortega and W. C. Rheinboldt, "Iterative Solution of Nonlinear Equations in Several Variables", *Academic Press,* 1970.

[6] Xianjin Yao and C. L. Liu, "Pin Position Assignment for Moveable Pins in Macro-cells" *to appear.*

Examples		*Input*				*Output*			
						Complete Graph		*Spanning Tree*	
No.	*h*	*cells*	*nets*	*wires*	*pins*	*total wire length*	*t(sec.)*	*total wire length*	*t(sec.)*
0	0.0	3	5	9	12	222.4 (228.7)	0.03	155.1 (160.2)	0.07
	0.5					224.8 (224.7)	0.07	157.5 (167.9)	0.07
	1.0					227.6 (225.8)	0.07	161.6 (172.8)	0.07
	2.0					230.2 (230.2)	0.08	167.5 (177.0)	0.08
1	0.0	100	254	254	508	4,835.5 (4,835.5)	2.95	4,835.5 (4,835.5)	3.42
	0.5					4,914.2 (4,914.2)	3.07	4,914.2 (4,914.2)	3.02
	1.0					5,039.6 (5,039.6)	5.67	5,039.6 (5,039.6)	5.13
	2.0					5,133.8 (5,133.8)	4.12	5,133.8 (5,133.8)	4.20
2	0.0	200	289	1,660	1,097	43,268.6 (44,143.1)	14.07	16,363.2 (16,728.8)	12.80
	0.5					43,513.6 (44,302.4)	18.40	16,579.2 (16,872.4)	16.10
	1.0					43,807.3 (44,709.9)	13.30	16,828.0 (17,011.6)	11.67
	2.0					44,235.4 (45,340.3)	15.35	17,118.9 (17,174.2)	20.18
3	0.0	400	580	3,355	2,203	92,128.3 (93,996.7)	28.28	34,432.7 (35,172.4)	26.08
	0.5					92,711.3 (94,458.6)	31.07	34,863.1 (35,535.7)	28.20
	1.0					93,554.6 (95,424.9)	38.95	35,437.7 (35,956.9)	38.93
	2.0					94,542.8 (96,498.1)	50.07	36,045.1 (36,402.7)	28.57
4	0.0	800	1,375	6,803	4,815	650,207.2 (653,480.2)	62.20	280,421.4 (282,015.6)	49.30
	0.5					651,567.8 (655,222.2)	68.77	281,582.9 (282,936.5)	52.85
	1.0					653,544.8 (657,494.0)	69.03	282,984.6 (283,817.0)	41.65
	2.0					656,552.5 (659,441.4)	69.57	284,383.7 (285,325.2)	52.92
5	0.0	1,600	2,710	14,330	9,802	1,366,972.0 (1,373,643.0)	131.07	572,441.0 (575,768.6)	81.55
	0.5					1,369,910.0 (1,377,421.0)	107.02	574,956.8 (577,644.5)	86.38
	1.0					1,373,894.0 (1,382,569.0)	107.48	578,084.1 (579,621.3)	108.22
	2.0					1,379,513.0 (1,387,521.0)	143.22	581,098.5 (581,998.1)	105.18
6	0.0	3,200	6,333	15,605	16,737	1,495,870.0 (1,502,510.0)	160.08	943,916.3 (948,265.1)	151.57
	0.5					1,500,784.0 (1,508,961.0)	217.92	948,247.5 (951,415.5)	129.52
	1.5					1,506,357.0 (1,513,061.0)	173.37	952,143.6 (954,507.9)	129.55
	2.0					1,512,456.0 (1,517,848.0)	130.10	955,894.3 (958,154.4)	96.95
7	0.0	4,800	8,748	34,675	27,952	3,402,121.0 (3,419,534.0)	328.65	1,696,029.0 (1,705,628.0)	221.75
	0.5					3,410,676.0 (3,429,439.0)	446.35	1,703,373.0 (1,710,404.0)	237.87
	1.0					3,421,779.0 (3,441,238.0)	268.82	1,711,188.0 (1,715,687.0)	297.92
	2.0					3,436,168.0 (3,452,059.0)	267.08	1,718,440.0 (1.722,400.0)	237.70
8	0.0	6,400	9,745	60,845	38,304	6,092,337.0 (6,122,938.0)	548.82	2,387,331.0 (2,401,430.0)	410.58
	0.5					6,104,405.0 (6,138,488.0)	600.43	2,397,613.0 (2,408,447.0)	350.83
	1.0					6,121,580.0 (6,158,490.0)	488.72	2,409,194.0 (2,415,428.0)	440.18
	2.0					6,146,415.0 (6,178,623.0)	600.00	2,420,580.0 (2,425,162.0)	352.60

Table 3