A Performance-Driven I/O Pin Routing Algorithm

Dongsheng Wang Ping Zhang Chung-Kuan Cheng Arunabha Sen
Department of Computer Sciences and Engineering
University of California at San Diego, La Jolla, CA 92093, USA

Abstract

This paper presents a performance-driven I/O pin routing algorithm with special consideration of wire uniformity. First, a topological routing based on min-cost max-flow algorithm is proposed. In this phase, an exponential weight function is used to guide the flow distribution which is very helpful in distributing wires, globally and uniformly, on the whole routing area. Then a physical routing phase is applied to implement one-to-one connection between chip pads and I/O pins, which focuses on the wire uniformity of the fanout area nearby the periphery of chip pads. Finally, a balanced position based wire polishing approach is proposed to further improve the local wire uniformity which tries to modify each wire into a smooth curve instead of broken line while satisfying the specified design rules such as wire-wire pitch and wire-pin pitch. A routing cost function is adequately defined to guide the whole routing process, which leads to a good trade-off between wire uniformity and wire length. The algorithm has been implemented and tested on up to 10-ring 600-pin PGA and the experimental results are very promising.

1 Introduction

Due to the tremendous increase in the complexity of IC designs, the number of I/O pins on a IC chip becomes larger and larger. It may be reaching up to 2000 in the near future [1]. In addition, designers are trying to extend the excellent package technologies to increasingly higher signal speeds. So a high performance automatic I/O pin routing tool is required. The routing on the typical package structures like Pin Grid Array (PGA) and Ball Grid Array (BGA) have been studied well. However, few researchers have studied the wire uniformity problem, which is very important for high performance PGA or BGA routing since it directly affect the electrical characteristics of the interconnections. Uniform configurations of all the wires provide predictable characteristics and allow accurate RLC modeling of smaller subsections. They also facilitate differential signal routing which is very important for packages that are expected to have higher simultaneous switching noise in the future.

There exist some PGA or BGA routers [2]-[4]. These routers took advantage of the special geometrics and symmetries of their respective problems and the freedom of interchangeable pins. [2] and [3] was the earlier work in this field. The wire uniformity problem for BGA routing was discussed by Yu and Dai in [4]. Their algorithm guaranteed that the difference of the number of wires between the adjacent pins on the same ring is less than or equal to 1. However, since their physical routing was based on the rubber-band approach, the wire distribution between the adjacent pins was still non-uniform, i.e., all the wires crowded to one of the pins. Recently, Yu, Darnauer and Dai proposed a more general pin routing approach in [5]. They used a min-cost max-flow algorithm to solve the interchangeable pin routing problem. However, wire uniformity problem was not addressed in this approach.

In this paper, we present a performance-driven I/O pin routing algorithm with special consideration of wire uniformity. The algorithm contains three phases. First, a topological routing based on min-cost max-flow algorithm tries to achieve a globally uniform wire distribution on the whole routing area. Then, a constructive physical routing phase is proposed with special consideration on wire uniformity of the fanout area nearby the periphery of chip pads. Finally, each wire is adjusted by a balanced position based wire polishing phase to further improve the local wire uniformity by polishing each wire from broken line into smooth curve. In addition, around the periphery area of each pin, the algorithm guarantees the specified wire-pin pitch requirements to be satisfied while the wire goes around the pin.

2 Problem Formulation

We define *pins* to be connectors on the package that are arranged in a grid array. *Pads* are via pads that are arranged on the periphery of IC chip inside the pin array, as shown in Fig. 1. Assume a PGA package has P pins $Pin = \{p_1, p_2, \cdots, p_P\}$ and P pads $Pad = \{q_1, q_2, \cdots, q_P\}$. All the pins are arranged in R rings and each ring contains P^r pins $Ring^r = \{p_1^r, p_2^r, \cdots, p_{P_r}^r\}$. $Ring^1$ and $Ring^R$ are called the inner ring and outer ring respectively. We assume each pad is connected to one and only one pin. So P wires $Wire = \{w_1, w_2, \cdots, w_P\}$ are required to complete the one-to-one routing. Wire w_i connects a pad q_i and a pin p, i.e., $w_i = (q_i, p)$. But we do not care which pad is connected to which pin. In expression $w_i = (q_i, p)$, we do not indicate the subscript of p because we do not know pad q_i is connected to which pin before the wire is routed. Once a wire is routed, a subscript will be assigned to pin p. We will use single layer for routing. The routing area is the minimum bounding box including all the pins but excluding the chip area, i.e., the area between chip periphery and outer ring. Each square area surrounded by four adjacent pins is called the routing cell, as shown in Fig. 1.

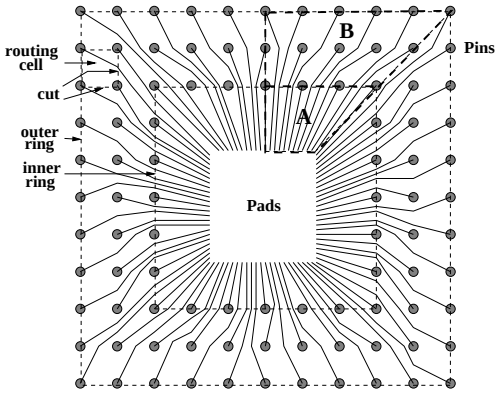


Fig. 1. PGA package

We assume the routing topology is monotonic. A monotonic topological routing is such a topological routing that wire $w_i = (q_i, p)$ connecting pad q_i and pin p intersects exactly one cut (p_k^s, p_{k+1}^s) for some k in each ring $1 \leq s \leq R-1$.

We allow a wire to be a curve instead of broken line. Each wire is approximated by a set of wire segments, i.e. $w_i = \{seg_1, seg_2, \cdots, seg_{K_i}\}$. Let $L_{i,j}$ be the length of segment seg_j of wire w_i, $DL_{i,j}$ and $DR_{i,j}$ be the distances between segment seg_j of wire w_i and its left and right adjacent wire or pin, respectively. Thus, the cost function of the PGA routing can be defined as:

$$cost = \sum_{w_i \in W} \sum_{seg_j \in w_i} L_{i,j} \cdot (\frac{1}{DL_{i,j}^2} + \frac{1}{DR_{i,j}^2} + \alpha) \quad (1)$$

Parameter α is a positive constant which takes a trade-off between wire length and wire uniformity. The goal of the algorithm is to complete the routing within given area with minimum cost function.

We specify a minimum wire-wire pitch $pitch_{w,w}$ and a minimum wire-pin pitch $pitch_{w,p}$. With the above assumptions and denotations, the performance-driven PGA routing problem is formulated as follows.

input:	a given PGA routing area net list $w_i = (q_i, p), i = 1, 2, \cdots, P$ pitch specification $pitch_{w,w}$ and $pitch_{w,p}$
output:	wires $Wire = \{w_1, w_2, \cdots, w_P\}$
minimize:	$cost$ in equation (1)
subject to:	$\forall i, j, pitch_{w_i,w_j} \geq pitch_{w,w}$ and $\forall i, j, pitch_{w_i,p_j} \geq pitch_{w,p}$

3 Algorithm Description

The routing algorithm is composed of three phases: topological routing, physical routing and wire polishing based on balanced position method.

3.1 Topological Routing

We define a flow network G(V,E) where V is a vertex set and E is a edge set. Each vertex $v_i \in V$ presents either a pin node v_p or a routing cell node v_r or a super node. A super node is either a source node v_s into which all the pads are grouped or a target node v_t which collects all the flows injected into all the pin nodes. Each directed edge $e(i,j)$ connects two nodes from v_i to v_j, associated with a capacity $c(i,j)$ which presents the maximum possible flow for edge $e(i,j)$. If the flow of edge $e(i,j)$ is $f(i,j)$ then $\forall i, j, f(i,j) \leq c(i,j)$. Because the routing is monotonic, $\forall i, j, f(i,j) = 0$ if $v_i = v_t$ or $v_i = v_p$ or $v_j = v_s$. Because of the symmetrics of PGA structure of Fig. 1, we only need to consider one-eighth of the whole flow network of PGA which is shown in Fig. 2. According to these denotations, the topological routing for PGA can be formulated as the min-cost max-flow problem.

input:	PGA routing problem
output:	all the flows $f(i,j) > 0$
minimize:	$\sum_{v_i, v_j \in V} g(i,j) \cdot f(i,j)$
subject to:	$\forall i, \sum_{v_j \in V} (f(i,j) - f(j,i)) = b_i$ $\forall i, j, 0 \leq f(i,j) \leq c(i,j)$

Where, $g(i,j)$ is exponential weight function associated with flow $f(i,j)$, which is defined as:

$$g(i,j) = exp(\beta \cdot \frac{f(i,j)}{c(i,j)}) \quad (2)$$

β is a constant determined by experiments, and b_i is defined as:

$$b_i = \begin{cases} P & \text{if } v_i = v_s, \text{(source node)} \\ -P & \text{if } v_i = v_t, \text{(target node)} \\ 0 & \text{others} \end{cases} \quad (3)$$

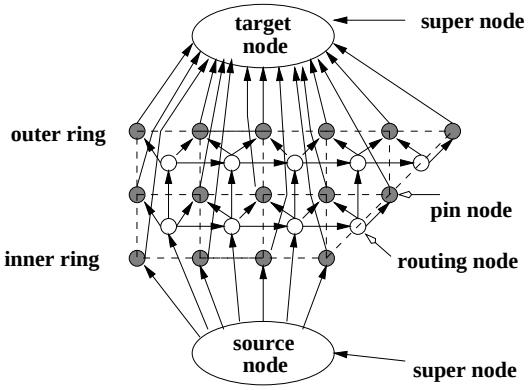


Fig. 2. Flow network of PGA

The min-cost max-flow problem defined above can be easily solved using the lower bound algorithm in [6]. Our experiments show that the weight function (2) does help to improve the wire uniformity.

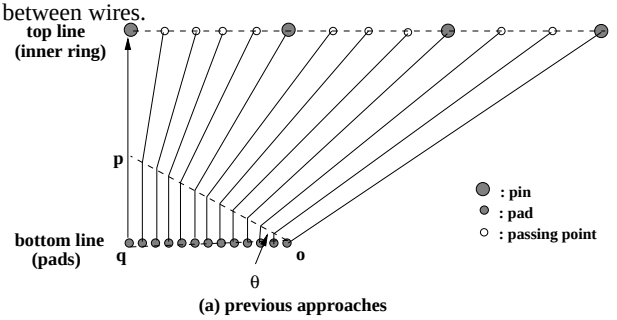
3.2 Physical Routing

Physical routing in the one-eighth routing area is divided into two regions, A and B, as shown in Fig. 1. The routings in region A and B are separately processed.

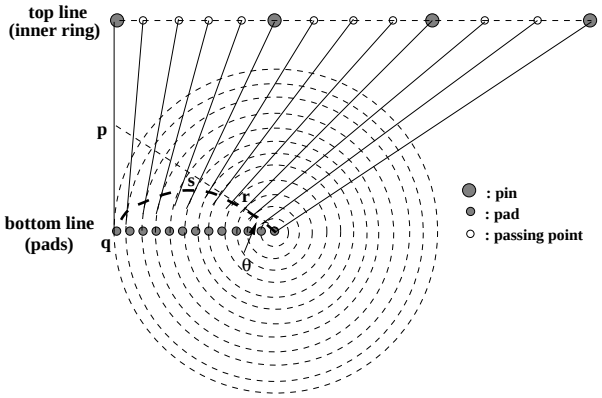
3.2.1 Routing In Region A

Region A contains the most congested area. The goals of the routing in region A are to reduce the most congested area into

a smaller area compared with the area used by other approaches and to generate smooth curves instead of broken lines to make the wire distribution in the most congested area more uniform. Such smooth curves could potentially result in reduced crosstalk between wires.



(a) previous approaches



(b) our approach

Fig. 3. Routing in region A

Previous approaches used broken lines to completed the one-to-one connection. Each line connects one pad to a pin or a passing point using one broken point. Passing points on inner ring will pass the connections to the pins outside inner ring. All the broken points lies on a straight line which keeps a fixed angle θ with the bottom line. Thus, the fanout area is right triangle opq, as shown in Fig. 3(a), where the number of passing points between the adjacent two pins is determined by the flow distribution generated in phase 1. This approach has two disadvantages. One is the fanout area, the most congested area, is too big. The other is that the broken points may lead to increased capacitive discontinuity and crosstalk.

Our approach uses curves instead of broken lines to complete routing in region A. The approach is outlined as follows. Taking point o (the position of the most right pad) as center, the distance between pads and the center as radius, for each pad, draw a circle passing it with the same center. For each circle, draw a tangent passing the corresponding pin or passing point. Thus, a tangential point is obtained for each circle. For example, points r, s, and q are three of these tangential points, as shown in Fig. 3(b). The connection between each pad and each pin or passing point is composed of two parts. One is the tangent between the pin or passing point and its corresponding tangential point. The other is the arc between the tangential point and the corresponding pad. The smooth arc nearby the periphery of chip improves the wire uniformity and the signal continuity, and the fanout area is dramatically reduced. Let S_{orsq} denote fanout area of the region surrounded by curve $orsq$ and the bottom line oq, S_{opq} denote fanout area of the right triangle opq, and a be length of the bottom line oq. The following theorem gives the difference between area S_{orsq} and S_{opq}.

Theorem 1 *The difference between area S_{orsq} of the fanout area orsq and area S_{opq} of the right triangle opq is:*

$$\Delta S = S_{orsq} - S_{opq} = \frac{1}{6}a^2(\theta - 3 \cdot tan(\theta)) \qquad (4)$$

Because of the space limitation, the proof is omitted. It should be noted that in actual layout implementation, the arcs within area $orsq$ might have to be approximated by straight line segments depending on tool limitations.

3.2.2 Routing In Region B

When the routing in region A is finished, routing in region B can be easily carried out. For each edge $e(i, j)$ with non-zero flow in region B, do the division $lc/f(i, j)$. Thus all the turning point on all the edges are obtained. Then according the the flow network of region B, connect the corresponding turning points along with the flow directions. Finally, each pad is connected to only one pin, which is guaranteed by the flow network.

3.3 Wire Polishing

The basic idea of balanced position based wire polishing approach is to split each wire into many short segments, each of which has such short length that the wire can be approximated as a curve by linking all the segments together. For each wire segment, the distances between it and its two adjacent obstacles are measured. Then, we try to move the segment to a "middle" position, called the *balanced position*, between the two adjacent obstacles. The operation of such a wire movement is called the *wire polishing*. The "balanced position" is defined as a physical position with minimum cost function that is defined by equation (1) in section 2. Other performance criteria could be used such as reduced trace capacitance or crosstalk.

This idea can be implemented by the "string balls" technique. Imagine that each wire bunches a string of balls, say, m balls $\{b_1, b_2, \cdots, b_m\}$. The radius of each ball and the distance between two adjacent balls are initially set to be equaling the wire width. In Fig. 4, the thick solid lines indicate the routed wires (obstacles) w_k and w_l respectively. The thin solid line indicates wire w_i to be adjusted. For the sake of clearness, the balls in Fig. 4 are drawn with radius larger than wire width. All the balls on wire w_i are iteratively processed one by one. For each iteration, a ball is pumped to increase its radius. If the ball touches one of its two adjacent obstacles, it is moved by a small distance toward the other adjacent obstacles. The movement is accepted and the ball is pumped again if it has the routing cost function (1) reduced. The ball pumping and movement continues until the cost function can not be further reduced. At this time, the ball has reached the balanced position. The iteration terminates when there is no ball left to be moved to reduce the cost function. Finally, the whole wire w_i has reached the balanced position, which is indicated by the thick dash line in Fig. 4.

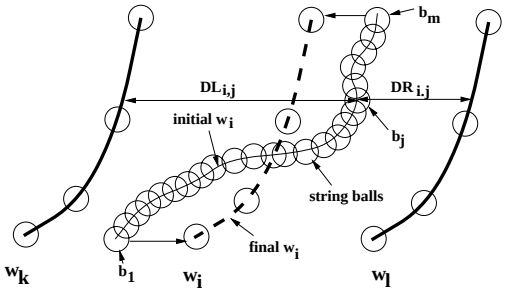


Fig. 4. String balls

4 Experiments

The performance-driven I/O pin routing algorithm been implemented in C programming language, and all experiments are performed on a Sun Sparc20 workstation.

Table 1. PGA Specifications

# rings	4	5	6	7	8	9	10
# pins	144	200	264	336	416	504	600
$pitch_{w,w}$	120			40			
$pitch_{w,p}$	930	830	610		510	410	

Eight PGAs have been tested. Parameter α in equation (1) is selected as 0.00005. Table 1 lists the main characteristics of the tested PGAs. The results are shown in Table 2, where # indicates the number of rings of each PGA. In order to show the effect of exponential weight function (2), we report two medium results of physical routing with $\beta = 0$ and $\beta > 0$, which are indicated as case I and case II, respectively. The final results of wire polishing phase is indicated as case III.

As shown, according to case I and case II tested, by using weight function (2) in min-cost max-flow algorithm, the routing cost (1) can be reduced by up to 10.35% (6-ring PGA), and there is some reduction in the total wire length for most PGAs. By comparing case II and case III, we can see that wire polishing can reduce the routing cost (1) by up to 30.35% (10-ring PGA) with wire length increase by at most 1.14% (6-ring PGA). If case I and case III are directly compared, the routing cost (1) can be reduced by up to 36.58% (10-ring PGA) with wire length increase by at most 0.73% (10-ring PGA). These results show that the wire uniformity has been significantly improved. Finally, Table 2 tells us that the algorithm we proposed runs very fast. It needs only three minutes for 10-ring 600-pin PGA routing.

Table 2. PGA routing results

	routing cost (1)			wire length			cpu
#rings	I	II	III	I	II	III	(s)
4	16.5	16.3	16.0	207563	206880	207652	4
5	28.4	27.9	26.5	309889	309008	311610	13
6	45.4	40.7	37.8	398115	395526	400032	19
7	165	153	143	672148	668192	669660	54
8	224	213	208	857972	858202	860329	110
9	314	309	288	1051849	1052390	1055877	109
10	626	570	397	1341635	1338770	1351410	184

5 Conclusions

A performance-driven I/O pin routing algorithm has been presented in this paper. Experimental results have shown some unique features of the algorithm. First, an exponential weight function is adequately used in the min-cost max-flow algorithm to build the routing topology with low routing cost and globally uniform wire distribution. Second, the physical routing approach can dramatically reduce the fan out area nearby the periphery of chip pads. Third, the balanced position based wire polishing approach can significantly improve the wire uniformity based on the adequately defined routing cost function. Such uniform connection should result in less capacitive discontinuities, line-to-line crosstalk, ease of electrical modeling and differential signal routing for future high performance chip-carriers.

Acknowledgements

We would like to thank Alina Deutsch for her very helpful comments on this paper. This work was supported in part by grants from the NSF project MIP-9529077 and the California MICRO Program.

References

[1] "The National Technology Roadmap for Semiconductors", by Semiconductor Industry Association, 1994.

[2] Chia-Chun Tsai, Sao-Jie Chen, "Planar Routing on A Pin Grid Array Package". *ICCAD/Graphics'93*, Beijing, pp.439-444, 1993.

[3] C.S. Ying, J. Gu, "Automated Pin Grid Array Package Routing on Multilayer Ceramic Substrates", IEEE trans. VLSI System, vol. 1, no. 4, pp571-575, 1993.

[4] M.F. Yu, W.W. Dai, "Single-Layer Fanout Routing and Routability Analysis for Ball Grid Arrays", *ICCAD'95*, pp.581-586, 1995.

[5] M.F. Yu, J. Darnauer, W.W. Dai, "Interchangeable Pin Routing with Application to Package Layout", *ICCAD'96*, pp.668-673, 1996.

[6] Kennington, Helgason, "Algorithms for Network Programming", John Wiley & Sons, 1980.