Ordered Escape Routing Using Network Flow and Optimization Model

Kashif Sattar, Anjum Naveed
School of Electrical Engineering and Computer Science (SEECS),
National University of Sciences and Technology (NUST), Islamabad, Pakistan
kashif.sattar@seecs.edu.pk, anjum.naveed@seecs.edu.pk

Abstract—**With the advancement in technology, BGA based integrated circuits for robotics and other devices are being prepared in small sizes with more pin count. This increase in number requires more number of pins to be escaped from the inner side of the IC towards the escape boundary. Ordered escape routing is very important due to its great impact on area and length routing at later stage. Basic design rules of planarity and capacity along with constraints like length matching make the ordered escape routing problem more difficult. In this paper we formulate flow model on the basis of inter-pin capacity. Using flow model we propose optimization model that solves ordered escape routing problem under design constraints. Evaluation of model using randomly generated examples shows that maximum possible nets are being routed by the model.**

Keywords-PCB; BGA; Optimization Modelling; Flow Model; Ordered Escape Routing;

I. INTRODUCTION

PCB provides mechanical support to integrated circuits (ICs). Recently, majority of ICs are being produced as BGA type with hundreds of pins attached below the body of the IC in form of grid as shown in Fig. 1. The pins from different IC components are connected through conductive pathways called nets. The nets require planar (no net cross other net) routing to complete the proper circuit. Multiple layers of PCB fabric may be used if one PCB layer is not enough to route all nets in a planar fashion.

PCB routing can be divided into two parts of escape routing and length or area routing. Escape routing connects the pins under the component body to the boundary of the component. Length routing connects the routes from boundary of one component to the other. Unordered escape routing leads to complex length routing at stage 2. Therefore, ordered escape routing [1]-[3] or simultaneous escape routing is preferred over unordered escape routing. Given the high complexity of simultaneous escape routing, ordered escape routing has not received significant attention by researchers. The complexity further increase when types of nets are considered. Nets can be signal, power or ground. Separation between nets of different types is desired in general to avoid cross cable interference.

Lot of automated design tools are available for PCB routing; however, no tool is able to provide 100% rout-ability for complex components and the PCBs containing multiple of such components. Consequently, experts perform manual routing, which is a highly time consuming process. In this paper we focus on ordered escape routing and develop an optimization model to get the optimized and ordered escape routing results.

This research work contributes in two ways: First we propose a flow model on the basis of inter-pin capacity and model the PCB as a graph with vertices and edges. This graph provides multiple escape paths for each connecting pin and provides the basis for application of network flow model solutions to PCB routing problem. Second, we formulate a network flow based Integer Linear Program (ILP) for ordered escape routing. This model not only considers the basic constraints of planarity, capacity and net ordering but also considers other important constraints of power and signal integrity. This model provides maximal routing solution that can be achieved in order to maximize the net routing on the PCB for the given set of input and the constraints.

The rest of the document is organized as follows: In Section-II, we highlight some basic techniques used for planar routing and also discuss advancements in this area with latest literature review and the research gap. In Section-III, we formulate the flow model on the basis of inter-pin capacity. Section-IV explains the mathematical model along with the constraints for optimal results. In Section-V, we discuss the experimental setup and validate the model to get some results. Section-VI concludes the paper with a summary of our research contribution and findings.

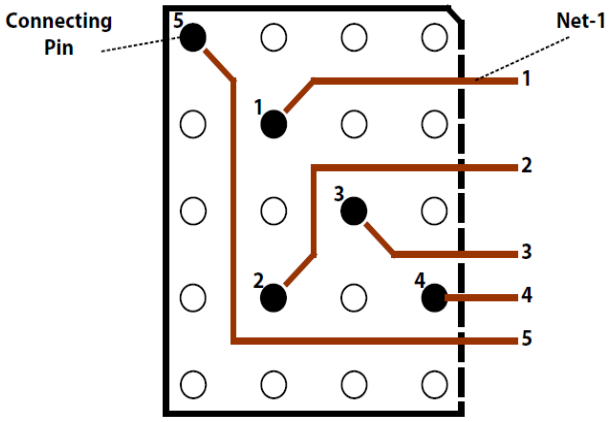

Figure 1. BGA Ordered Net Escaping

978-1-4799-6466-6/15/$31.00 ©2015 IEEE

II. Literature Review

At present, there are two main research areas in PCB routing, one is escape routing and the length or area routing. In escape routing, the objective is to escape all the pins towards the boundary of the component in an order or without order of the nets, on the basis of some constraints. In area routing, objective is to provide routing between the boundaries of components, mostly based on length matching constraints. Escape routing from a single component is of two types, ordered and unordered. In unordered escape routing, pins are routed to boundary without specific order [4]-[10]. Unordered escape routing is relatively easy due to fewer restrictions on paths. On the other hand, ordered escape routing aims at routing the pins to the border in specific order [11]-[14]. The order of escape is decided by the connectivity requirements of the two components under consideration. Predefined order significantly increases the complexity of routing and often presents the problem where escaping is not possible at all.

Bus Escape Routing (also called Max. Disjoint Subset Problem) [11] is a research work in ordered escape routing. The approach considers multiple nets simultaneously, instead of considering the individual nets. The group of nets known as bus is routed towards nearest boundary such that maximum buses can reach the boundary. The technique reduces the complexity of routing but is not feasible for ordered escape routing. For individual nets routing, fixed escape boundary points technique is used [12], [13] to reduce complexity. However, not all possible paths are considered. Ordered escape routing based on Boolean satisfiability has been proposed to map the rout-ability problem on optimization problem [14]. The time complexity of the solution is high, making it unsuitable for large problems.

Although multiple researchers have addressed the problem of unordered escape routing, the literature on ordered escape routing is relatively limited and mostly relies on heuristics. Furthermore, only basic constraints of planarity and capacity are considered without taking into consideration the constraints on power and signal separation and signal integrity. Similarly, length of routes has not been considered in ordered escape routing until now.

III. Proposed Flow Model

In this section, we define the flow model, for escaping of pins through the BGA component. Pins of BGA component usually consist of 2-Dimensional array or grid. Some of these pins need to be connected to the pins in another component for the entire PCB to be operational. Such pins are called the connecting pins. We consider a BGA tile (a part of BGA grid) consisting of '4' adjacent pins as shown in Fig. 2. We propose a flow model for this tile, which is extendable for the entire grid for the complete escaping solution. Within a tile, inter-pin capacities are of three type, i.e. Horizontal (H-cap), Vertical (V-cap) and Diagonal (D-cap), as shown in Fig. 2. In this paper, we consider inter-pin capacities of '1'. However, with little change, the capacity values can be of any generic setup.

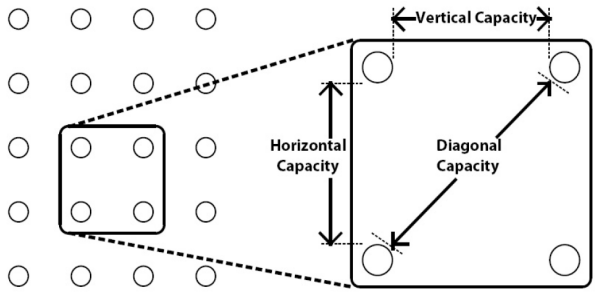

Figure 2. BGA Tile Consisting of 4-Pins

We introduce an intermediate point in the center of the tile, which helps us to establish an escape path using the available set of edges. For higher available capacity, multiple points will be introduced inside each tile. The edges from connecting pin to intermediate point are directional whereas the edges from intermediate point of a tile to another intermediate point of another tile are bidirectional, as shown in Fig. 3(a). In case of a single intermediate point we have *H-cap=V-cap=D-cap=1*. The intermediate point of a tile can be transit for some other connecting pin in the grid or can provide direct connectivity to the adjacent connecting pin, as shown in Fig. 3(b) and (c) respectively.

As discussed earlier, we can increase D-cap by introducing some more intermediate points as long as cable width and cable separation constraints are not violated. Since we have to maintain planarity of nets to avoid net crossings, hence by introducing four intermediate points instead of one, we can increase the D-cap=2 as shown in Fig. 3(d), however the other capacities are still same i.e. *H-cap=V-cap=1*. We can see from the two example instances in Fig. 3(e) and (f), that the number of escaping nets increases with this change and two wires can cross a single tile instead of one.

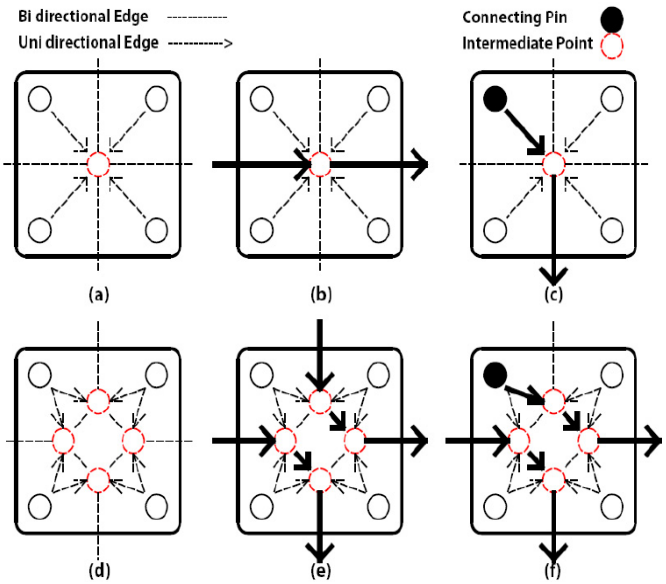

Figure 3. Proposed Flow Model

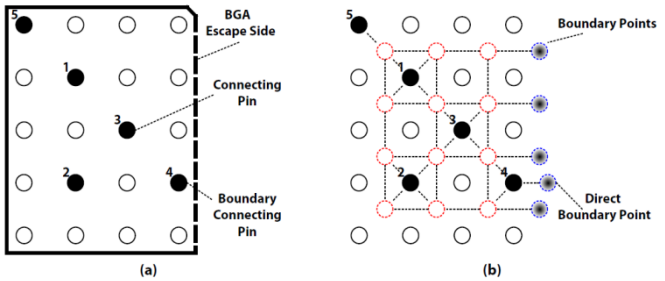

Figure 4. Boundary points of Flow Model

In addition to intermediate points, we are also considering boundary points as shown in Fig. 4(b). These points are the escaping points on the boundary of the BGA component towards escaping side. Usually boundary points are the points outside the component boundary between two adjacent pins, but sometimes if we have a boundary connecting pin just on the last column of pins towards escaping side as shown in Fig. 4(a), then it has an extra direct boundary point for direct connectivity as shown in Fig. 4(b).

IV. Proposed Optimization Model

In this section, we propose the ILP optimization model that takes a graph *G(V,E)* as input, where *V* is a set of vertices (which includes connecting pins, intermediate points and boundary points) and *E* is a set of all possible edges (available for net escaping from the component) and returns a planar graph G*′(V′,E′)* with maximum possible nets that can be escaped under given set of constraints.

The model takes following sets as input:

V = Set of all pins and points of a BGA component.
I = Set of all intermediate points.
E = Set of all possible edges of a component.
BE= Set of all boundary edges of a component.
N= Set of nets (pins) that need to be escape
NE[e_{ij}]= Set of neighboring edges, where $e_{ij} \in BE$.

The model also takes the set P of the pair of nets that cannot escape together (from adjacent boundary points). This maintains the power and signal integrity for error free routing.

We define the decision variable $Y_{(n_a,e_{ij})}$ of the model for all nets $n_a \in N$, all links $e_{ij} \in E$ as a binary variable. The variable acquires the value one if edge e_{ij} is used for the routing of net n_a, otherwise, the value of the variable is zero.

$$Y_{(n_a,e_{ij})} = \begin{cases} 1 & \text{if edge } e_{ij} \text{ is used for routing } n_a \\ 0 & \text{otherwise} \end{cases}$$

The routing is constrained by connectivity constraints, planar graph constraints, net order constraints and system constraints. The constraints of each type are explained below.

A. Connectivity Constraints

This set of constraints defines basic routing of the nets. The constraints ensure that the pin of any net are connected to the boundary escape point, using the edges from set *E* and there is exactly one route to escape.

Single Path Activation Constraint: Multiple edges are incident on the connectivity pins. For every connectivity pin that needs routing, at the most only one incident edge can be used for the routing of a particular net. Since the edges from connectivity pins are unidirectional towards the intermediate points only, therefore we need to apply this constraint only for the edges going outside from pins of component for that net. This constraint allows no net routing for the connectivity pins if the complete escape routing path construction in a planar fashion is not possible.

$$\sum_{e_{ij}\in E \,\&\, i=a} Y_{(n_a,e_{ij})} \le 1 \qquad \forall n_a \in N$$

Route Completion Constraints: This constraint ensures that if one incident edge on an intermediate point connects the point to one of the connectivity pins of a particular net (directly or through a path of edges), then another edge must be active for the same net to allow the connectivity to the boundary point of the same net. This constraint also ensures that if a route cannot initiate from an intermediate point, without being connected to any connectivity pin.

$$\sum_{e_{ij}\in E} Y_{(n_a,e_{ij})} = \sum_{e_{jk}\in E} Y_{(n_a,e_{jk})}$$
$$\forall n_a \in N, \forall j \in I$$

The above constraint does not eliminate the possibility that the same edge is used twice. If the same edge is used twice, the path construction stops at the intermediate point with incomplete routing. Following constraint ensures that different edges are selected for any net at each intermediate point.

$$Y_{(n_a,e_{ij})} + Y_{(n_a,e_{ji})} \le 1$$
$$\forall n_a \in N, \forall e_{ij}, e_{ji} \in E$$

B. Planar Graph Constraint

The above set of constraints results in route-able nets. However, the resultant connectivity graph may not planar. Although all edges are used by at the most one net, the routes for different nets can intersect at intermediate or boundary points. This condition can be eliminated by introducing a simple constraint. The constraint ensures that any point on the PCB is used by at the most one net.

$$\sum_{n_a\in N} \sum_{e_{ij}\in E} Y_{(n_a,e_{ij})} \le 1, \quad \forall j \in I$$

C. Net Order Constraint

All the escaped nets of a component must be in some order to ensure ordered escape routing. Since we are assuming that all nets will escape in sorted ascending order hence ath net number must be less than (a+1)th net number. Similarly due to grid of pins, if ath net is passing through escape boundary point *'j'* and (a+1)th net is passing through escape boundary point *'l'*, then $j < l$.

$$\sum_{e_{ij} \in BE} Y_{(n_a, e_{ij})} . j \le \sum_{e_{kl} \in BE} Y_{(n_{a+1}, e_{kl})} . l$$
$$\forall n_a \in N$$

D. System Constraint

It is possible to add a variety of system constraints into model. For example, it is preferred that the power signal wires (nets) should not be adjacent to the data and control signal wires, specifically if the wire width is less. Presence of such wires adjacent to one another can result in signal errors. In this paper, we add this constraint into the model as an example.

Power Signal Integrity Constraint: This constraint ensures that the specific pair of nets must not escape through adjacent boundary points. Keeping the safe distance between them, allows transferring of error free signals. Following constraint checks that if one boundary edge is active for one net then the second net from the pair must not be active on any of its adjacent neighboring boundary edges.

$$X_{(n_a, e_{ij})} + \sum_{e_{kl} \in NE[e_{ij}]} X_{(n_b, e_{kl})} \le 1$$
$$\forall (n_a, n_b) \in P, \forall e_{ij} \in BE$$

E. Objective Function

The objective of this model is to maximize the number of nets that can be routed and escaped. Following expression ensures the objective:

$$\max . \sum_{n_a \in N} \sum_{e_{aj} \in E} Y_{(n_a, e_{aj})}$$

The complete optimization model is presented in equation set 1 - 7.

$$\max . \sum_{n_a \in N} \sum_{e_{aj} \in E} Y_{(n_a, e_{aj})} \quad (1)$$

Subject to:

$$\sum_{e_{ij} \in E \,\&\, i=a} Y_{(n_a, e_{ij})} \le 1 \qquad \forall n_a \in N \quad (2)$$

$$\sum_{e_{ij} \in E} Y_{(n_a, e_{ij})} = \sum_{e_{jk} \in E} Y_{(n_a, e_{jk})} \quad (3)$$
$$\forall n_a \in N, \forall j \in I$$

$$Y_{(n_a, e_{ij})} + Y_{(n_a, e_{ji})} \le 1 \quad (4)$$
$$\forall n_a \in N, \forall e_{ij}, e_{ji} \in E$$

$$\sum_{n_a \in N} \sum_{e_{ij} \in E} Y_{(n_a, e_{ij})} \le 1, \qquad \forall j \in I \quad (5)$$

$$\sum_{e_{ij} \in BE} Y_{(n_a, e_{ij})} . j \le \sum_{e_{kl} \in BE} Y_{(n_{a+1}, e_{kl})} . l \quad (6)$$
$$\forall n_a \in N$$

$$X_{(n_a, e_{ij})} + \sum_{e_{kl} \in NE[e_{ij}]} X_{(n_b, e_{kl})} \le 1 \quad (7)$$
$$\forall (n_a, n_b) \in P, \forall e_{ij} \in BE$$

Optimization models are generally NP-hard. However, for specific problem instance and the use of available solvers, the models can produce optimal results. In the following section, we solve the proposed optimization model using commercially available solvers for the example instances.

V. Results and Discussion

In this section, we validate the ILP optimization model proposed in section-IV for correctness and evaluate its performance on example instances.

A. Model Validation

In this section we verify, that the route-ability, planar graph construction and net ordering constraints are able to achieve the desired routing for the given instances. For this purpose we test the model using the example component of BGA having grid size of 5x4. This BGA has 20 pins in total as shown in Fig. 5(a). We randomly selected 5 connecting pins for escape routing. For better understanding, we have assumed inter-pin capacity of one and single side escaping is permitted.

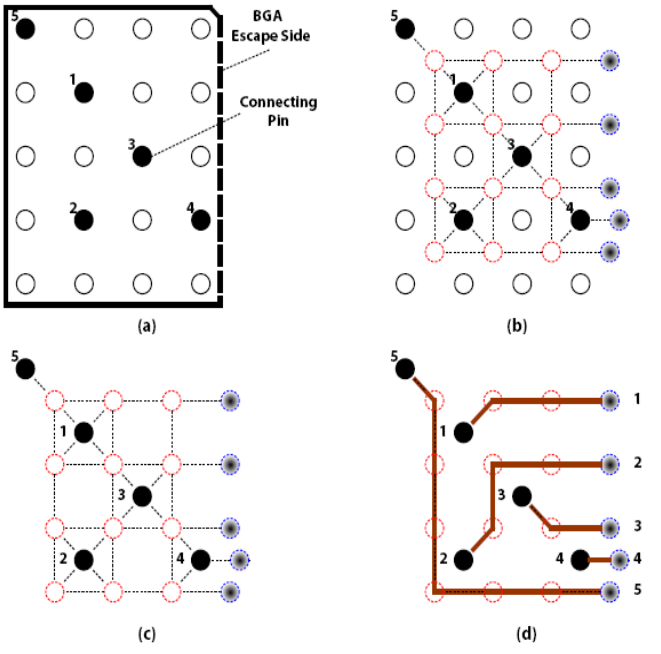

Figure 5. Ordered Escape Routing on BGA 5x4

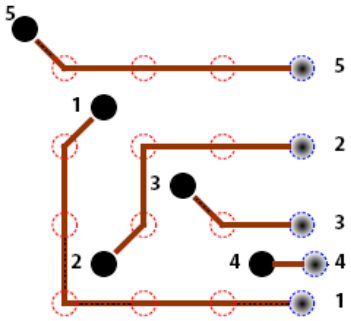

Figure 6. Validation of System Constraint

We draw intermediate points, boundary points and all possible connecting edges as shown in Fig. 5(b). Then we remove unnecessary pins and gets a graph *G(V,E)* as shown in Fig. 5(c). This graph is the input to the proposed optimization model, which returns another graph *G'(V',E')* as an escape routing solution by considering all set of constraints as shown in Fig. 5(d). Example clearly shows that the proposed model is able to perform ordered escape routing for BGA based PCB component. The BGA has maximum 5 boundary points and the model routes all the 5 nets by escaping in the given order. The net order it chooses in a sequence 1,2,3,4 and 5. Also we can see that it is not using any edge or vertex twice, to ensure planarity. This example validates our constraints of connectivity, planar graph and net order.

We consider the same example to validate the system constraints. The model takes the set *P* of the pair of nets (4 and 5) that cannot escape together. We enable the `power and signal integrity' constraint and run the model once again, keeping all other things and constraints be the same. Again it routes all the nets in a planar fashion but with different net order. The order it follows this time is 5,2,3,4 and 1 as shown in Fig. 6. This new order shows that model performs exactly as required, by keeping away the net 4 and 5. Hence our proposed optimization model validates this constraint too.

B. Experimental Results

We implement our flow model in C++ language, which generates data file as an input for the solver. We have used AMPL language to implement the optimization model and AMPL modeler [15] available at NEOS server [16] to get the optimization model results.

We test our model on 6 data sets having different types of BGA components with different net routing requirement. We compare the result with the Proteus auto router by Labcenter Electronics Ltd. The results are reported in Table I. It can be seen that Proteus is unable to route all the nets and number of unrouted nets increases as the complexity of the problem increases, whereas the proposed optimization model routes all the nets for all the data sets.

We discuss one of the test example in detail, as shown in Fig. 7. It is a BGA of size 17x6 and we have 18 nets that need to be routed in sorted order from 1 to 18. Fig. 7(a) is the output of Proteus, whereas Fig. 7(b) is the output of our proposed model. It can be seen that proteus was only able to route 15 nets in sorted order. However the proposed model routed all the 18 nets in the required order.

TABLE I. EXPRIMENTAL RESULTS

	Array *R x C*	Escape Pins	Proteus Routing		Our Model	
			Nets	*%age*	*Nets*	*%age*
Test-1	5x4	5	5	100%	5	100%
Test-2	7x7	7	6	85%	7	100%
Test-3	10x7	12	10	83%	12	100%
Test-4	17x6	18	15	83%	18	100%
Test-5	50x32	56	45	80%	56	100%
Test-6	86x50	100	78	78%	100	100%

VI. CONCLUSION

This research has proposed the use of optimization models for solving the problem of ordered escape routing for BGA type of PCB components. The problem of ordered escape routing has been mapped on the planar graph construction problem from a given graph consisting of nodes and edges. Flow model has been proposed on the basis of inter-pin capacity, which helps in finding the all possible paths for escaping. An optimization model has been proposed for planar graph and has been validated to solve the ordered escape routing problem. To the best of our knowledge, this is the first work that considers the power and signal integrity constraint.

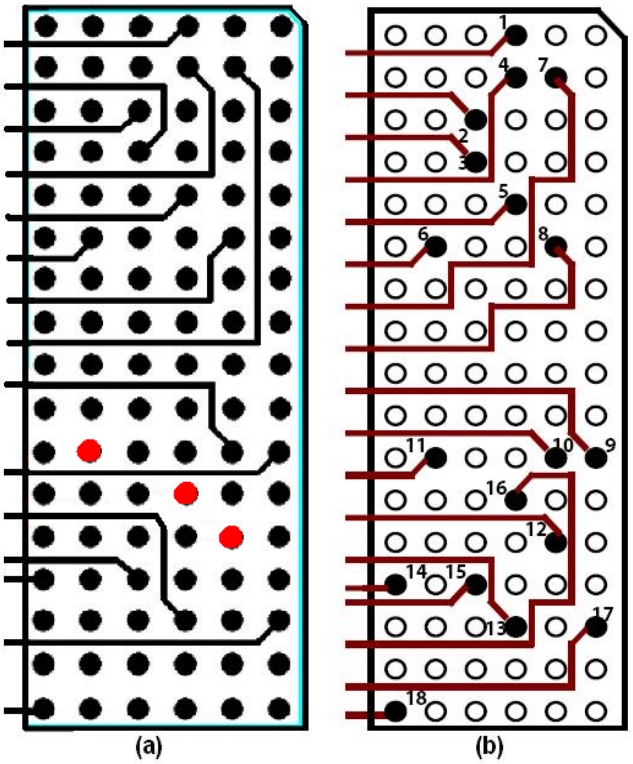

Figure 7. Routing solution of BGA size 17x6

REFERENCES

[1] G. Marcantonio, "Ball grid array (BGA) integrated circuit packages," U.S. Patent 5,796,170, issued August 18, 1998.

[2] D. Y. Chong, B. Lim, K. J. Rebibis, S. Pan, S. Krishnamoorthi, R. Kapoor, A. Y. Sun, and H. Tan. "Development of a new improved high performance flip chip BGA package," In Proc. Electronic Components and Technology Conference, vol. 2, pp. 1174-1180, 2004.

[3] R. Plieninger, M. Dittes, and K. Pressel. "Modern IC packaging trends and their reliability implications," Microelectronics Reliability 46, No. 9, 2006.

[4] Y. K. Ho, H. C. Lee, and Y. W. Chang, "Escape routing for staggered-pinarray PCBs," IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems, vol. 32, no. 9, pp. 1347-1356, 2013.

[5] K. Wang, S. Dong, H. Wang, Q. Chen, and T. Lin, "Mixed-crossing-avoided escape routing of mixed-pattern signals on staggered-pin-array PCBs," IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems, vol. 33, no. 4, pp. 571-584, 2014.

[6] Y. Birk, and J. B. Lotspiech, "On Finding Non-Intersecting Straightline Connections of Grid Points to the Boundary," Journal of Algorithms, 13(4): 636-656, Dec. 1992.

[7] W.-T. Chan, and F. Y. L. Chin, "Efficient Algorithms for Finding Maximum Number of Disjoint Paths in Grids," SODA, pp. 454-463, 1997.

[8] J.W. Fang, I. J. Lin, P. H. Yuh, Y. W. Chang, and J. H.Wang, "A Routing Algorithm for Flip-Chip Design," Proc. of ICCAD, pp. 753-758, 2005.

[9] Tan Yan, Martin DF Wong. "A correct network flow model for escape routing," In Design Automation Conference, 2009. DAC'09. 46th ACM/IEEE, pp. 332-335. IEEE, 2009.

[10] Qiang Ma, and Martin DF Wong. "NP-Completeness and an Approximation Algorithm for Rectangle Escape Problem With Application to PCB Routing," Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on 31, No. 9, 2012.

[11] Kong, Hui, Qiang Ma, Tan Yan, and Martin DF Wong. "An optimal algorithm for finding disjoint rectangles and its application to PCB routing." In Design Automation Conference (DAC), 2010 47th ACM/IEEE, pp. 212-217. IEEE, 2010.

[12] Y. Tomioka, and A. Takahashi, "Monotonic Parallel and Orthogonal Routing for Single-Layer Ball Grid Array Packages," Proc. of ASPDAC, pp.642-647, 2006.

[13] J.W. Fang, C.H. Hsu, and Y.W. Chang, "An Integeer Linear Programming Based Routing Algorithm for Flip-Chip Design," Proc. of DAC, pp. 606-611, 2007.

[14] Lijuan Luo and Martin D.F. Wong, "Ordered Escape Routing Based on Boolean Satisfiability," In proceedings of the 2008 Asia and South Pacific Design Automation Conference, pp. 244-249. IEEE computer society press, 2008.

[15] R. Fourer, D. M. Gay, B. W. Kernighan, AMPL: a modeling language for mathematical programming, Duxbury Press/Brooks/Cole, 2002.

[16] J. Czyzyk, M. Mesnier, and J. More, "The NEOS Server," IEEE Journal on Computational Science and Engineering, vol. 5, issue 3, 1998.