Generalized Via Pattern Awareness Substrate Routing Framework for Fine Pitch Ball Grid Array

Jun-Sheng Wu   Chi-An Pan   Yi-Yu Liu

Department of Computer Science and Information Engineering,
National Taiwan University of Science and Technology, Taipei, Taiwan, R.O.C.
m10615112@mail.ntust.edu.tw   m10615050@mail.ntust.edu.tw   yyliu@mail.ntust.edu.tw

Abstract

Packaging substrate has become one of the most important carriers to enable system-level and heterogeneous design within a small footprint size. Instead of applying advanced semiconductor interposer process technologies, the fine pitch ball grid array (FBGA) package substrates are manufactured by mechanical processes. To tackle stringent design rules owing to the mismatched via dimension and miscellaneous routing obstacles, substrate interconnect designs are usually customized by experienced substrate layout engineers. However, fully net-by-net manual design for hundred-scale FBGA is time consuming and error-prone. In this paper, we model the FBGA substrate routing as an integer linear programming (ILP) problem taking various via patterns and design-dependent constraints into account. Two-stage early exit methodology and ILP constraint reduction techniques are developed to boost the runtime of ILP solver. Experimental results indicate the potential of the proposed framework. We argue that complex FBGA designs could be semi-automated by using via pattern candidates to reduce the substrate layout design cycle time.

I. INTRODUCTION

Fine pitch BGA (FBGA) is one of the most representative BGA variants for high pin count and small form factor. Compared to the advanced 2.5D silicon interposer, 3D through-silicon integration, and wafer-level packaging technologies, the FBGA packaging technology is the only relative low cost choice among these packaging styles when there are more than hundreds of I/O pins in a chip package. Nowadays, FBGA packaging substrate has become one of the most important carriers to enable system-level and heterogeneous design within a small footprint size. The cross-section view and the top view of a 2-layer wire-bonding FBGA package are drawn in Figure 1(a) and Figure 1(b), respectively. There is a die glued on the top layer of package substrate. The bump ball array is on the bottom layer. The die pads are wire-bonded to substrate bonding fingers on the top. Signals are transmitted to the bottom bump balls by C1, via, and C2 layers.

Instead of using expensive semiconductor process technology for the package substrates, FBGA substrates are manufactured by mechanical processes. However, the mechanical processes incur one major challenge – the mismatched via dimension. Typically, the FBGA via diameter is up to 4-6 times of the wire width (trace size) and the via diameter is even as large as 0.7 times of the package bump ball diameter. The mismatched via dimension impacts the substrate routability when there are only few number of routing layers. Figure 1(b) depicts one stringent example, the wire-bonding fingers overlap with package bump balls in the top view. Obviously, the blockages of bonding fingers and vias impact the top metal routing; while the blockages of vias and bump balls impact the bottom metal routing. Therefore, proper via planning is a crucial design challenge in FBGA packages. Empirically, substrate interconnect designs are usually customized by experienced layout engineers to tackle complex and stringent design rules. However, fully net-by-net manual design for hundred-scale FBGA is time consuming and error-prone. In this paper, an integer linear programming (ILP) formulation is proposed taking various via patterns and design-dependent constraints into account. To the best of our knowledge, this is the first work that addresses the mismatched via dimension issue in wire-bonding FBGA routing.

The rest of this paper is organized as follows. The background of BGA substrate routing and our proposed grid-based graph model are discussed in Section II. Section III presents a generalized ILP-based framework for FBGA routing. The experimental results are summarized in Section IV. Section V concludes this paper.

II. PRELIMINARIES

A. Review of BGA Substrate Routing

Given the number of BGA substrate layers, the locations of bonding fingers on the top metal layer, and the locations of bump balls on the bottom metal layer, the BGA substrate routing problem is to interconnect the bonding fingers and their corresponding bump balls. Thanks to the wide wire pitch compared to the interconnects inside chips, typical BGA substrate routing wire could orient toward any angle. Nevertheless, to optimize manufacturability, most BGA substrate layouts still use horizontal/vertical (i.e., rectilinear) and X-shape (i.e., 45° and 135° diagonal) wire segments. There are a variety of research works related to BGA substrate routing [1]–[4]. Yan and Wong classify common PCB routing problem into two major categories – the unordered escape and the ordered escape routings [1].

In ordered escape routing problems, many previous works are based on partitioning substrates into four triangular sectors and routing each sector independently. Kubo and Takahashi propose a 2-layer BGA global router based on novel via assignment and iterative improvement techniques [3].
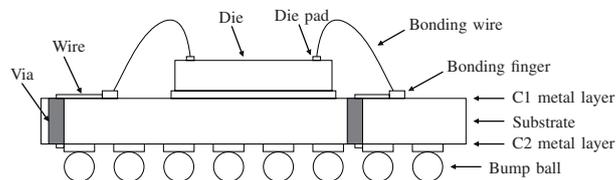
(a) The cross-section view.
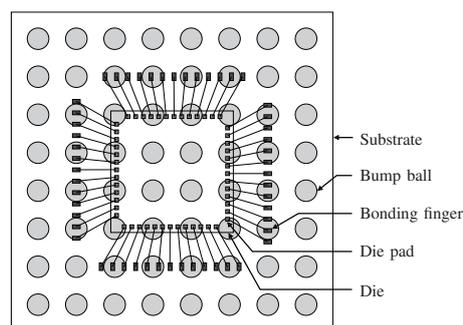
(b) The top view.

Fig. 1. The schematic of a wire-bonding FBGA.

Tomioka and Takahashi further improve the runtime and routability of the global router in the bottom routing layer and also reduce the number of wire congestion violations in the top routing layer [4]. Nevertheless, the presumed one via candidate at the center of four bump balls could incur more wire congestion violations in the top routing layer. These violations would increase the difficulty to map the global routing result during the detailed routing stage. Most of the previous works assume that the vertical projection of bonding fingers and bump balls are disjoined. However, this assumption is too optimistic for area-constrained BGA designs. In fact, wire-bonding fingers, as well as flip-chip C4 bump balls, are usually overlapped with package bump balls in a package top view to minimize package size. Therefore, we have to carefully handle these design-dependent obstacle regions simultaneously in BGA routing.

B. Square Lattice Grid Routing Model for FBGA

In order to maintain sufficient routing flexibility and at the same time to keep the routing graph in a compact manner, we use a uniform square lattice grid model for FBGA routing. The granularity of lattice grid is a trade-off between the runtime efficiency and the routing quality. A model with large grid size reduces the runtime at the cost of routing quality degradation; while a model with small grid size improves the routing quality at the cost of longer runtime. In resource-constrained FBGA substrate routing, the size of lattice grid pitch is set to the minimum wire pitch, which is the sum of minimum wire width and wire spacing.

Since the routing resource is tightly restricted in FBGA substrates, increasing the number of diagonal routing candidates could improve the routability. Therefore, we adopt the same lattice grids for X-shape diagonal wire candidate generation instead of using virtual X-shape wire tracks. As a result, the number of diagonal routing candidates on the same lattice grids is $\sqrt{2}$ times more than those on virtual wire tracks. The X-shape diagonal edges are superimposed on the lattice grids with 45° and 135° orientations. Notice that the improved routing flexibility in diagonal edge could result in design rule violation, and thus the additional avoidance constraints will be addressed in Section III. To interconnect routing planes, vertical via edges are between the lattice grids of adjacent routing planes. Each via edge indicates candidate location for via placement. The via edges could be further stacked for multiple substrate routing planes (layers) with the support for various via technologies, such as through hole via, blind via, and buried via. Finally, based on FBGA design rules, metal wire segments are restricted or even forbidden in bonding finger regions, fiducial mark regions, solder mask regions, and package edge regions. Our routing model maintains these design-dependent constraints by removing corresponding rectilinear or X-shape edges as keep out zones. Since the proposed routing graph is capable of covering many design details of real FBGA substrates,

we are able to further extend the wire-bonding routing graph to the flip-chip C4 bump array routing graph.

III. ILP-BASED MCMCF PROBLEM FORMULATION

In this section, we formulate the FBGA routing as a minimum cost multi-commodity flow problem [5]. We use the aforementioned grid-based graph model in our ILP formulation. For simplicity, the grid size is based on the minimum wire pitch of the target FBGA design.

Assume that N is the set of all 2-pin nets, the source vertices are the bonding finger grids and the target vertices are the bump ball grids, respectively. In the routing graph G, the i-th grid vertex in the vertex set V is denoted as v_i and the j-th edge in the edge set E is denoted as e_j. The edge flow capacity is 1 and the edge cost is equivalent to the corresponding Euclidean wire length. Since the flow capacity is 1, we use binary variable f_{n_i,e_j} to represent the flow of the net n_i on the graph edge e_j. Variable f_{n_i,e_j} is 1 when the net n_i occupies the edge e_j. In this case, variables f_{n_k,e_j} $(k \neq i)$ are 0, so that the flow capacity constraint could be satisfied. Constant W_{e_i} is the cost of edge e_i in terms of actual wire length. The objective of FBGA routing is to find a routing result with minimum total wire length. Hence, the objective function of the ILP-based MCMCF problem can be formulated in Equation 1.

$$\text{Minimize:} \sum_{i=1}^{|N|} \sum_{e_j \in E} W_{e_j} \times f_{n_i,e_j} \tag{1}$$

We define the edge sets $E_{v_i}^{fanout}$ and $E_{v_i}^{fanin}$ as sets of outgoing and incoming edges through vertex v_i, respectively. The connectivity constraints in Equation 2 guarantee continuous flows from source vertices to their corresponding target vertices. Hence, Equation 2 is either 1 or -1 when the vertex v_i is the source or target of net n_l, respectively. Otherwise, the equation is 0 when the vertex v_i is either an internal vertex of net n_l or irrelevant to net n_l.

$$\sum_{e_j \in E_{v_i}^{fanout}} f_{n_l,e_j} - \sum_{e_k \in E_{v_i}^{fanin}} f_{n_l,e_k} = \begin{cases} 1 : \text{source} \\ -1 : \text{target} \\ 0 : \text{others} \end{cases}, \forall v_i \in V, \forall n_l \in N \tag{2}$$

We define the edge set E_{v_i} as the set of all edges incident to vertex v_i. Equation 3 ensures that there are at most 2 wire segments connected to vertex v_i.

$$\sum_{k=1}^{|N|} \sum_{e_j \in E_{v_i}} f_{n_k,e_j} \leq 2, \forall v_i \in V \tag{3}$$

We define the edge set E_x as the set of all X-shape edges. The set X_{e_x} contains only one edge, denoted as $\{e_y\}$, which is diagonally intersects with the X-shape edge e_x. Equation 4 allows at most one diagonal wire segment within a square lattice.

$$\sum_{i=1}^{|N|} (f_{n_i,e_x} + f_{n_i,e_y}) \leq 1, \forall e_x \in E_x, \exists \{e_y\} = X_{e_x} \tag{4}$$

Since the actual pitch between two adjacent X-shape wire segments is only $\frac{\sqrt{2}}{2}$ of the minimum wire pitch, additional avoidance constraints are required to prevent design rule violations. We define the vertex set V_{e_y} as the set of the two vertices of the edge e_y. Equation 5 avoids design rule violations of the two vertices of the edge e_y.

$$\sum_{i=1}^{|N|} (f_{n_i,e_x} + \sum_{e_k \in E_{v_j}} f_{n_i,e_k}) \leq 2, \forall e_x \in E_x, \forall v_j \in V_{e_y} | \{e_y\} = X_{e_x} \tag{5}$$

Since the mismatched via dimension issue is severe in FBGA routing, placing a via requires to block its surrounding edges in order to avoid design rule violations. Figure 2 draws a region of via blockage. All the via blocked edges must be included in the via blockage constraint. We define the edge set $B_{e_{via}}$ as the set of the via blocked edges of the via edge e_{via}. Equation 6 compels all surrounding edges to be 0 when there is a via placed on the edge e_{via}. With this constraint, we are capable of handling the most critical design issue in FBGA routing.

$$f_{n_i,e_{via}} + \sum_{n_j=1}^{|N|} f_{n_j,e_b} \leq 1, \forall n_i \in N, n_j \neq n_i, \forall e_{via} \in E_{via}, \forall e_b \in B_{e_{via}} \tag{6}$$

IV. EXPERIMENTAL RESULTS

We implement our substrate router in C++ and use Gurobi Optimizer 8.0 as our ILP solver. Our experiments are conducted on a Linux-based workstation with 2.6 GHz Intel Xeon processor and 64 GB memory. Three industrial designs are used in our experiment. These designs are imported from a commercial tool, Cadence Allegro Package Designer (APD). Our routing results are then exported back to Cadence APD.

Table I compares the routing results of the original ILP formulation and our proposed two-stage early exit methodology. Columns ILP, Limit,
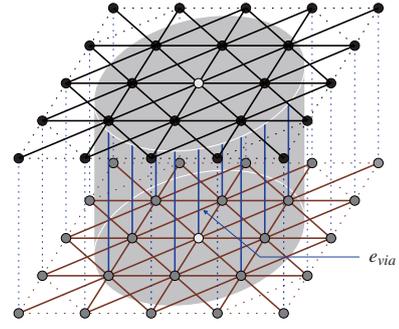
Fig. 2. Example of via blockage constraint.

TABLE I
ROUTING RESULT COMPARISON

Design	ILP	Limit (min)	Route	WL (μm)	Iter	T (min)
Small	Single	60	56	85,222	0	60.1
		360	56	82,598	0	360.1
	2-stage	5	53	76,120	7	5.2
		10	55	81,987	4	10.1
		15	55	81,987	4	15.1
Medium	Single	60	92	156,965	0	60.3
		360	93	160,178	0	360.1
	2-stage	5	94	168,225	6	11.6
		10	94	172,104	4	20.9
		15	94	170,787	3	30.2
Large	Single	60	238	561,122	0	65.7
		360	243	586,876	0	364.0
	2-stage	5	N/A	N/A	N/A	N/A
		10	251	644,549	14	28.3
		15	254	653,312	14	24.0

Route, WL, Iter, and T represent ILP formulation style, runtime limit of each ILP iteration, number of routed nets, total wire length, number of rip-up and re-route iterations, and total runtime, respectively. Since the ILP solver is unable to obtain a preliminary routing result for design *Large* in 5 minutes, we mark "N/A" to the corresponding row. From Table I, the two-stage methodology is capable of obtaining a routing result with competitive completion rates efficiently. In *Medium* and *Large* designs, the completion rates of the two-stage methodology are even better than those of the original ILP formulation owing to the runtime limit. Nevertheless, the solution quality of our early exit heuristic may be slightly inferior to that of the original ILP formulation such as design *Small*.

V. CONCLUSIONS AND FUTURE WORK

To tackle the mismatched via dimension issue in FBGA, we have formulated the wire-bonding package substrate routing as an ILP problem. With the inclusion of FBGA design-dependent details in our grid-based model, the proposed routing framework could be further extended for multi-layer substrate routing and flip-chip design style. To further speed up the framework, we are now working on developing more ILP reduction techniques while maintaining the grid-based routing flexibility for more FBGA variants. Since the routing result can be exported back to commercial tool for follow-up refinement, we argue that complex FBGA designs could be semi-automated by using via pattern candidates to reduce the substrate layout design cycle time.

REFERENCES

[1] T. Yan and M. D. F. Wong, "Recent research development in PCB layout", *in Proceeding of International Conference on Computer-Aided Design*, pp.398-403, 2010.

[2] L. Luo, T. Yan, Q. Ma, M. D. F. Wong, and T. Shibuya "A new strategy forsimultaneous escape based on boundary routing", *in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, pp.205-214, vol.30, i.2, 2011.

[3] Y. Kubo and A. Takahashi, "Global routing by iterative improvements for two-layer ball grid array packages", *in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, pp.725-733, vol.25, i.4, 2006.

[4] Y. Tomioka and A. Takahashi, "Routability driven via assignment method for 2-layer ball grid array packages", *in IEICE Transactions on Fundamentals*, pp.1433-1441, vol.E92-A, i.6, 2009.

[5] X. Jia, Y. Cai, Q. Zhou, and B. Yu, "A multicommodity flow-based detailed router With efficient acceleration techniques", *in IEEE Transaction on Computer-Aided Design of Integrated Circuits and Systems*, pp.217-230, vol.37, i.1, Jan. 2018.