

# Reticle Floorplanning with Guaranteed Yield for Multi-Project Wafers

Andrew B. Kahng  
CSE and ECE Departments  
University of CA, San Diego  
La Jolla, CA, 92093  
abk@ucsd.edu

Sherief Reda  
CSE Department  
University of CA, San Diego  
La Jolla, CA, 92093  
sreda@cs.ucsd.edu

## Abstract

*With the dramatic increase in mask costs, multi-project wafers have become an attractive choice for low-volume chip fabrication. By using the same set of masks to fabricate a number of different chips, the mask-set cost is amortized among different chip providers, leading to significant cost reduction especially for chip prototyping. In this paper we present a new algorithm for reticle floorplanning with wafer yield guarantees. The previous approach of Kahng et al. [4] considers optimizing both the reticle area and the wafer yield, leading to suboptimal solutions with no yield bounds. By contrast, we consider the yield as a constraint and optimize the area accordingly. We characterize yield constraints and provide a mechanism through which yield can be incorporated into an optimal-area packer. The incorporation of yield constraints prunes large parts of the search space of the optimal-area packer, leading to runtime-efficient optimal-area floorplans with guaranteed yields. Empirical results demonstrate that our approach dominates previous results, i.e., we give floorplans that consume less area and have higher die yields. For the 10 benchmarks studied in [4], we achieve a yield improvement of 14% with an area reduction of 2%.*

## 1 Introduction

In recent years, mask (or reticle) costs have dramatically increased. This is attributed to several factors including (i) huge rise in number of features due to design size increases enabled by scaling, and (ii) the use of reticle enhancement techniques such as optical proximity correction (OPC) and phase shifting mask (PSM). These elevated mask costs hinder cost-effective prototyping and low-volume manufacturing.

To reduce and share manufacturing costs, fabrication facilities introduced multi-project wafers so that different chips can share the same set of masks, leading to mask-cost

amortization among the different chips. A crucial step in multi-project mask (or reticle) preparation is *reticle floorplanning*, which arranges the dies within the reticle.

One objective for reticle floorplanning is minimizing the reticle area, hence increasing the raw number of printed chips. On the other hand, diamond sawing technology used for wafer dicing requires any wafer cut to proceed from side to side. Such side-to-side wafer cuts cut through some die copies as they extract other dies, and hence decrease the actual yield of diced chips.

Skillful reticle floorplanning is the key issue in attaining high-yielding multi-project wafers. While classical chip floorplanning is concerned with minimizing the area and wirelength [3], the objective of reticle floorplanning is minimizing the reticle area as well as maximizing the yield. Minimizing the area increases the number of reticles a wafer can accommodate, while maximizing the yield reduces the number of wafers necessary to manufacture the required production volume of each die.

Drawbacks of previous approaches include (i) ignoring yield constraints [5], (ii) assuming expensive wafer dicing equipment [6], and (iii) simultaneously optimizing the area and yield [4] leading to no guarantees on either. In this paper we take a different approach to reticle floorplanning: we consider the yield as a constraint and develop a number of floorplan constraints that guarantee yield bounds. The yield constraints prune large portions of the floorplan search space, and lead to a runtime-efficient optimal-area packer. Our empirical results indicate that we dominate previous approaches [4] with floorplans that occupy less area and provide higher yields.

The organization of this paper is as follows. Section 2 gives the necessary background for understanding this work, and motivates our approach of floorplanning under yield guarantees. Section 3 outlines the proposed methodology by establishing a characterization for yield guarantee. Our empirical results are presented in Section 4, and Section 5 gives concluding remarks.

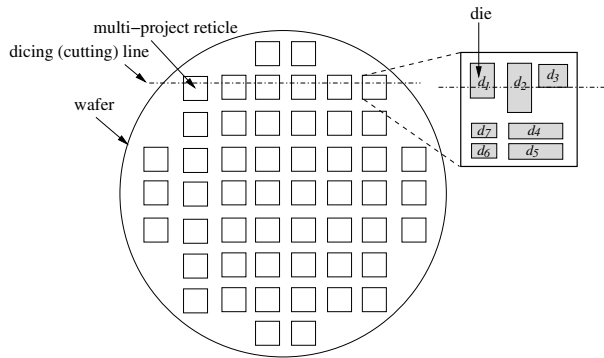


Figure 1. Reticle/Die overview.

## 2 Background and motivation

A wafer consists of a number of reticle “shots” arranged in  $m$  rows and  $n$  columns as shown in Figure 1; we refer to these as *reticle rows* and by *reticle columns*, respectively. Each reticle can have a number of dies from different designs. The *side-to-side dicing model* [4] assumes that the wafer-cutting tool cannot arbitrarily stop during cutting; consequently, all reticles in same reticle row (or reticle column) will be cut (or diced) by the same set of horizontal (or vertical) cuts. A *dicing plan* specifies a set of cutting lines. These cutting lines determine which dies will be usable and which will be destroyed. For example, the cutting line on the first reticle row in Figure 1 successfully extracts die  $d_3$ , but destroys dies  $d_1$  and  $d_2$ . Notice also that a cut line that extracts  $d_1$  does not eliminate the need for cut lines for  $d_2$  and  $d_3$ , since such a cut line will leave excess margin for each of  $d_2$  and  $d_3$ . Given a reticle floorplan with a set of dies  $D = \{d_1, d_2, \dots, d_l\}$ , the *wafer yield* is defined as the minimum, over all dies  $d_k \in D$ , of the number of successfully extracted copies of  $d_k$ . The objective of a dicing plan is to maximize the yield, since this translates to fewer wafers.

In the reticle floorplanning approach of Kahng *et al.* [4], the primary objective is minimizing the total reticle area and the secondary objective is maximizing the yield. We notice that minimizing the reticle area will likely lead to adverse impacts on the yield. From Kahng *et al.* [4] results (duplicated in Table 1 for convenience), we observe that there is strong correlation between area and yield. Comparing results from Parquet - a well-established yield-oblivious area packer - and the “shelf” packer - an area packer with yield as a secondary objective - we find that there is a strong correlation between the area and yield in the results of these floorplanners, i.e., smaller areas lead to lower yields and larger areas lead to higher yields. The low yield associated with smaller areas will eventually lead to large wasted die areas during side-to-side dicing, outweighing any initial area savings.

In this work we suggest a different approach. We con-

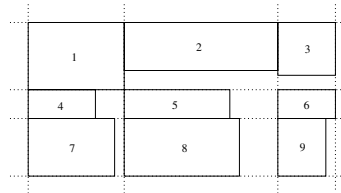


Figure 2. Reticle floorplans as grids.

struct floorplans that consider specified yield bounds as constraints and find a packing that minimizes the reticle area while respecting the constraints. The main challenge with our approach is specifying and satisfying the yield constraints at all times while constructing the reticle floorplan. To simplify the yield calculations, we restrict the floorplans to grids as shown in Figure 2 and use this as a starting point for developing a characterization for yield guarantee. We note that Andersson *et al.* [5] propose a Polynomial-Time Approximation Scheme (PTAS) to pack dies into grids under the sole objective of minimizing the reticle area with no regard to the yield. Our work is different: we take the yield as a constraint and develop a practical methodology for reticle floorplanning. Our approach can be summarized in three steps: (i) restrict floorplans to grids, (ii) develop a characterization for yield of grid floorplans, and (iii) pack the dies in the grid optimally using branch and bound respecting the specified rules for yield guarantee.

## 3 Proposed Methodology

In our methodology, we restrict floorplans to grids. A grid is composed of a number  $q$  of *grid rows* and a number  $p$  of *grid columns* as shown in Figure 2 (compare against reticle rows and columns). A grid representation allows the development of a characterization for yield guarantee which we give in Subsection 3.2. We discuss our optimal area packer in Subsection 3.3. We now formulate the reticle floorplan problem.

### 3.1 Problem Formulation

Given a set of dies  $D = \{d_1, d_2, \dots, d_l\}$ , we define the following:

- $w_i$  gives the width of a given die  $d_i \in D$ .
- $h_i$  gives the height of a given die  $d_i \in D$ .

Let a variable  $x_{ijk} = 1$  if and only if die  $d_k$  is placed at grid row  $i$  and grid column  $j$ ; otherwise,  $x_{ijk} = 0$ . Define the following two sets:

$$\lambda_j = \{w_k \mid \exists i, k \text{ such that } x_{ijk} = 1\}, \quad (1)$$

and

$$\pi_i = \{h_k \mid \exists j, k \text{ such that } x_{ijk} = 1\}. \quad (2)$$

$\lambda_j$  ( $\pi_i$ ) gives the set of width (height) values of cells placed in grid column  $j$  (grid row  $i$ ). Considering that the width (height) of a column (row) is determined by the largest width (height) die residing in that row (column), the objective of minimizing the grid area can be written as

$$\min \sum_{j=1}^p \max_{w \in \lambda_j} w \times \sum_{i=1}^q \max_{h \in \pi_i} h, \quad (3)$$

such that

$$\forall k : \sum_{j=1}^p \sum_{i=1}^q x_{ijk} = 1 \quad (4)$$

$$\forall i, j : \sum_{k=1}^l x_{ijk} \in \{0, 1\} \quad (5)$$

The constraint given by Equation (4) states that each die must be placed, and the constraint given by Equation (5) states that each slot in the grid can be occupied by at most one die.

While optimizing the objective function given by Equation (3) gives an optimal grid packing, these packings offer no yield guarantees. We next develop the necessary constraints for attaining any desired yield, and for obtaining a smooth tradeoff between yield and packing area.

### 3.2 Floorplanning with Yield Guarantee

An *independent row set* is a set of dies within a grid row that have the same height;  $\pi_i$  (as given by Equation 2) gives the number of independent row sets within a grid row  $i$ . Similarly, an *independent column set* is a set of dies within a grid column that have the same width;  $\lambda_j$  (as given by Equation 1) gives the number of independent column sets within a column grid  $j$ . An independent set can be diced at the same time.

For the special case of 100% yield, there must be at most one independent row set per row and at most one independent column set per column, i.e.,  $\forall i : \pi_i = 1$  and  $\forall j : \lambda_j = 1$ . That is, if 100% yield is to be achieved, each row in a grid must accommodate dies of only the same height and each column in the grid must accommodate dies of only the same width. Given 100% yield as a constraint, one has to find the minimum area floorplan that satisfies this yield. To reduce the reticle area at the expense of the yield, we can allow (i) different die heights to reside in the same grid row, and (ii) different die widths to reside in the same grid column. Under the assumed dicing model, if a row (column) contains dies of different heights (widths) then any dicing plan that extracts dies of the one height (width) must damage all dies of different heights (widths).

Assume a wafer has  $m$  reticle rows and  $n$  reticle columns, with a total of  $m \times n$  reticles shots. Further

assume that we are given a reticle floorplan where some die  $d_k$  is placed at grid row  $i$  and grid column  $j$ , i.e.,  $x_{ijk} = 1$ . Then, a constructive lower bound on the number of reticle rows where copies of die  $d_k$  are extracted is  $\lfloor \frac{m}{\pi_i} \rfloor$ . Similarly, a constructive lower bound on the number of reticle columns where copies of die  $d_k$  are extracted is  $\lfloor \frac{n}{\lambda_j} \rfloor$ . Hence the number of copies  $c_k$  extracted of die  $d_k$  is at least  $c_k = \lfloor \frac{m}{\pi_i} \rfloor \times \lfloor \frac{n}{\lambda_j} \rfloor$ . The yield is equal to  $\min_{d_k \in D} c_k$ . The following example illustrates yield calculation.

**Example 1:** Assume a wafer of  $m = 10$  reticle rows and  $n = 10$  reticle columns. Consider some grid row  $i$  that has a set of dies of two different heights  $\pi_i = \{h_1, h_2\}$ , while for all grid columns  $j$ ,  $|\lambda_j| = 1$ , i.e., no grid column accommodates dies of different width. Furthermore, assume a number  $m_1$  of reticle rows will be used to extract all dies of height  $h_1$ , while a number  $m_2$  of reticle rows will be used to extract dies of height  $h_2$ . Since  $m = m_1 + m_2$  maximizing the yield implies maximizing the minimum of  $m_1 n$  and  $m_2 n$ . This occurs when  $m_1 = m_2 = \frac{10}{2} = 5$ , giving a yield of 50.

The general relationship between yield, the number of different heights per row  $\pi_i$ , and the number of different widths per column  $\lambda_j$  is captured by the following theorem. We derive the relation between yield and area using a single parameter  $\gamma \in Z^+$ . As  $\gamma$  increases, the reticle area decreases and the yield decreases. To produce concise expressions, we make the reasonable assumption that  $m = n$ .

**Theorem 1.** Given a wafer of  $m$  rows and  $n$  columns, a yield of at least  $\frac{m^2 - m(\gamma + 1)}{\gamma}$  is attained if the following constraint is met:  $\forall d_k \in D : [x_{ijk} = 1] \rightarrow [|\pi_i| \times |\lambda_j| \leq \gamma]$ .

**Proof:** Assume  $\forall d_k \in D : [x_{ijk} = 1] \rightarrow [|\pi_i| \times |\lambda_j| \leq \gamma]$ . Hence, according to our earlier discussion: for any die  $d_k$ , where  $x_{ijk} = 1$ , a constructive lower bound on the number of die copies of  $d_k$  is  $c_k = \lfloor \frac{m}{\pi_i} \rfloor \times \lfloor \frac{n}{\lambda_j} \rfloor = \frac{m - m \bmod \pi_i}{\pi_i} \times \frac{n - n \bmod \lambda_j}{\lambda_j}$ . Since  $\pi_i \lambda_j \leq \gamma$ ,

$$c_k \geq \frac{(m - m \bmod \pi_i)(n - n \bmod \lambda_j)}{\gamma}. \quad (6)$$

Assuming that  $m = n$  then

$$c_k \geq \frac{(m^2 - m(m \bmod \pi_i + m \bmod \lambda_j) + (m \bmod \pi_i \times m \bmod \lambda_j))}{\gamma}. \quad (7)$$

Since  $m \bmod \pi_i < \pi_i$  and  $m \bmod \lambda_j < \lambda_j$ , we get

$$c_k \geq \frac{(m^2 - m(\pi_i + \lambda_j) + (\pi_i \times \lambda_j))}{\gamma} \geq \frac{m^2 - m(\gamma + 1)}{\gamma}, \quad (8)$$

since  $\pi_i + \lambda_j \leq \pi_i \times \lambda_j + 1 = \gamma + 1$ .  $\square$

**Input:** Die  $d$  to be placed.  $n$  is the total number of dies.  $w$  is the number of grid columns.  $h$  is the number of grid rows. Initialize  $bestcost = \infty$

**Output:** Optimal floorplanning for the specified yield

1. **if**  $d \leq n$  **then**
2.   **for**  $i = 1$  to  $h + 1$  **do**
3.     **for**  $j = 1$  to  $w + 1$  **do**
4.       **if** no dies are placed at  $(i, j)$  **and** placing  $d$  at  $(i, j)$  does not violate yield constraints **then**
5.          Place die  $d$  at  $(i, j)$
6.           $cost =$  evaluate area of partial floorplan
7.          **if**  $cost < bestcost$  **then** search( $d + 1, n, w + 1, h + 1$ )
8.          Undo placing  $d$  at  $(i, j)$
9.          Flip die  $d$  and repeat steps 4 . . . 7
10.   **else**
11.      $cost =$  evaluate area of complete floorplan
12.     **if**  $cost < bestcost$  **then**
13.        $bestcost = cost$
14.     Store the current floorplan as the best floorplan

**Figure 3. search(): a branch and bound procedure with yield guarantee.**

Maintaining the constraint of Theorem 1 while reticle floorplanning ensures packings with guaranteed yields as given by Equation (6). Note that Equation (8) gives a less tight lower bound than that given by Equation (6); this was necessary to arrive to a concise expression. The addition of the yield constraint to the formulation of Subsection 3.1 prunes the search space for finding optimal packings under guaranteed yields. We exploit this in the next subsection.

### 3.3 Floorplanning Under Yield Guarantees

Our reticle floorplanner seeks to construct a grid floorplan with optimal area as given by Equation (3) while respecting the yield rule as given by Theorem 1 and the constraints given by Equations (4) and (5). For this purpose, we employ a branch and bound procedure with lower bounds. Experimental results show that such approach is practically feasible. We can attribute this to three reasons: (i) reticle floorplan instances are small, (ii) yield constraints prune a huge amount of the search space, and (iii) using lower bounds further prunes large portions of the search space.

The complete branch and bound *search()* procedure is given in Figure 3. Given a grid of  $w$  columns and  $h$  rows, the branch and bound procedure places die sequentially. Given a current die  $d$ , the procedure places  $d$  at each grid slot (steps 2 . . . 3) and branches (step 7) to place the remain-

ing dies. For a die to be *legally* placed in a grid slot, none of the yield constraints should be violated (step 4). Lower bounds are utilized (steps 6 . . . 7) to prune solutions that are not promising. The procedure also expands the current grid by an extra column and row (in loop bounds of steps 2 and 3), and evaluates placements in the additional slots. This expansion is required since it is possible that none of current grid slots is legal. For each die, we also consider *flipping*, i.e., rotating the die by 90 degrees as given in Step 9. If all dies are placed then grid area is evaluated and compared to the best grid floorplan area constructed so far (steps 11 . . . 14).

## 4 Experimental Results

We evaluate our technique using the benchmarks supplied by authors of [4]. We also utilize the dicing-plan code of [4] as an independent means to evaluate the yield of our reticle floorplans. We assume (as [4]) a wafer that can accommodate  $m = 10$  reticle rows and  $n = 10$  reticle columns. This means that the wafer has 100 reticle shots. All of our experiments are run using a Linux Intel Xeon 2.4GHz workstation with 2GB memory.

For convenience, we duplicate all the results of [4] in Table 1. #Die gives the number of dies in each test case, and Die area gives the total die area. GTMUCH [2] is a commercial reticle floorplanner. Parquet [1] is a yield-oblivious min-area floorplanner. Shelf+Shift and SA+IASA are the two approaches proposed by [4]. For each floorplanner, we report the yield, reticle area, and the CPU runtime. All of these approaches allow die flipping.

Table 2 gives the results from our method. We give a spectrum of results for yields corresponding to  $\gamma = 1, 2, 3, 4,$  and  $5$ . We observe that our approach allows a graceful tradeoff between the reticle area and yield. The right most column of Table 2 “Matched Yield” gives results when we use the best yield values of [4] in Table 1 as a lower bound on the yield for our floorplanner, and calculate the area using our approach. Our approach outperforms previous method in the following respects:

- We afford the ability to control the yield, in contrast to previous approaches which offer no guarantees about the yield. Furthermore, our results offer a graceful tradeoff between yield and area.
- Our **matched yield** approach dominates the SA+IASA technique of [4] for all but one benchmark, i.e., we give reticle floorplans that are better in both yield and area. For the 10 reticle floorplans studied in [4], we achieve a simultaneous yield *improvement* of 14% and an area *reduction* of 2%.

The results of Table 2 are summarized in the plot of Figure 4, where we plot all Pareto points corresponding to the

| Case No. | # Die | Die area | GTmuch |      | Parquet |      |        | Shelf+shift |      |        | SA+IASA |      |        |
|----------|-------|----------|--------|------|---------|------|--------|-------------|------|--------|---------|------|--------|
|          |       |          | yield  | area | yield   | area | CPU(s) | yield       | area | CPU(s) | yield   | area | CPU(s) |
| 1        | 10    | 231      | 18     | 255  | 14      | 255  | 0.03   | 16          | 276  | 0.00   | 28      | 288  | 80     |
| 2        | 18    | 226      | 16     | 285  | 5       | 252  | 0.07   | 21          | 253  | 0.16   | 25      | 270  | 783    |
| 3        | 11    | 252      | 15     | 280  | 10      | 272  | 0.03   | 16          | 280  | 0.01   | 30      | 294  | 132    |
| 4        | 9     | 203      | 18     | 221  | 14      | 220  | 0.03   | 25          | 224  | 0.01   | 30      | 288  | 118    |
| 5        | 10    | 226      | 12     | 272  | 15      | 247  | 0.02   | 25          | 280  | 0.00   | 25      | 260  | 94     |
| 6        | 15    | 227      | 10     | 285  | 6       | 252  | 0.06   | 18          | 238  | 0.00   | 18      | 238  | 226    |
| 7        | 15    | 234      | 14     | 285  | 9       | 252  | 0.06   | 15          | 288  | 0.01   | 20      | 285  | 782    |
| 8        | 14    | 232      | 20     | 285  | 9       | 255  | 0.03   | 20          | 288  | 0.00   | 20      | 288  | 152    |
| 9        | 10    | 231      | 20     | 285  | 14      | 255  | 0.03   | 16          | 276  | 0.00   | 28      | 288  | 161    |
| 10       | 20    | 215      | 6      | 304  | 6       | 238  | 0.09   | 15          | 255  | 0.01   | 20      | 260  | 1020   |
| Total    |       | 2277     | 149    | 2757 | 102     | 2518 |        | 187         | 2668 |        | 244     | 2757 |        |

**Table 1. Results of [4]. Yield is the minimum percentage of die copies diced from the wafer under the respective dicing plan.**

| Case  | # Die | Die area | $\gamma = 1$ |      |        | $\gamma = 2$ |      |        | $\gamma = 3$ |      |        | $\gamma = 4$ |      |         | $\gamma = 5$ |      |        | Matched Yield |      |         |
|-------|-------|----------|--------------|------|--------|--------------|------|--------|--------------|------|--------|--------------|------|---------|--------------|------|--------|---------------|------|---------|
|       |       |          | yield        | area | CPU(s) | yield        | area | CPU(s) | yield        | area | CPU(s) | yield        | area | CPU(s)  | yield        | area | CPU(s) | yield         | area | CPU(s)  |
| 1     | 10    | 231      | 100          | 494  | 0.02   | 50           | 312  | 0.03   | 30           | 288  | 0.02   | 25           | 270  | 0.07    | 20           | 264  | 0.01   | 30            | 288  | 0.02    |
| 2     | 18    | 226      | 100          | 396  | 25.23  | 50           | 286  | 227.02 | 30           | 275  | 344.09 | 28           | 260  | 4705.30 | 20           | 275  | 417.11 | 28            | 260  | 4705.30 |
| 3     | 11    | 252      | 100          | 456  | 0.04   | 50           | 312  | 0.04   | 30           | 300  | 0.03   | 30           | 290  | 0.15    | 20           | 288  | 0.01   | 30            | 290  | 0.15    |
| 4     | 9     | 203      | 100          | 500  | 0.01   | 50           | 300  | 0.03   | 30           | 264  | 0.01   | 25           | 252  | 0.14    | 20           | 256  | 0.02   | 30            | 264  | 0.01    |
| 5     | 10    | 226      | 100          | 560  | 0.03   | 50           | 336  | 0.05   | 30           | 285  | 0.09   | 25           | 260  | 0.33    | 20           | 280  | 0.04   | 25            | 260  | 0.03    |
| 6     | 15    | 227      | 100          | 494  | 8.61   | 50           | 306  | 10.22  | 30           | 275  | 6.46   | 25           | 276  | 93.55   | 30           | 275  | 6.41   | 30            | 275  | 6.41    |
| 7     | 15    | 234      | 100          | 416  | 0.41   | 50           | 297  | 4.03   | 30           | 280  | 4.00   | 25           | 270  | 49.78   | 30           | 280  | 1.46   | 25            | 270  | 49.78   |
| 8     | 14    | 232      | 100          | 416  | 0.41   | 50           | 297  | 2.58   | 30           | 280  | 3.66   | 25           | 270  | 48.58   | 20           | 273  | 1.55   | 25            | 270  | 48.58   |
| 9     | 10    | 231      | 100          | 494  | 0.02   | 50           | 312  | 0.03   | 30           | 288  | 0.02   | 25           | 270  | 0.07    | 20           | 264  | 0.01   | 30            | 288  | 0.02    |
| 10    | 20    | 215      | 100          | 360  | 36.74  | 50           | 270  | 290.32 | 30           | 240  | 306.33 | 25           | 270  | 6446.10 | 30           | 240  | 278.07 | 30            | 240  | 278.07  |
| Total |       | 2277     | 1000         | 4586 | -      | 500          | 3028 | -      | 300          | 2775 |        | 258          | 2688 |         | 230          | 2443 | -      | 283           | 2705 | -       |

**Table 2. Results of the proposed technique. Yield is the minimum percentage of die copies diced from the wafer under the respective dicing plan.  $\gamma$  is an upper bound on  $\pi_i \times \lambda_j$  for all rows  $i$  and columns  $j$  as defined in the text.**

(yield, area) tradeoff. The results establish a Pareto frontier, allowing a graceful tradeoff between area and yield.

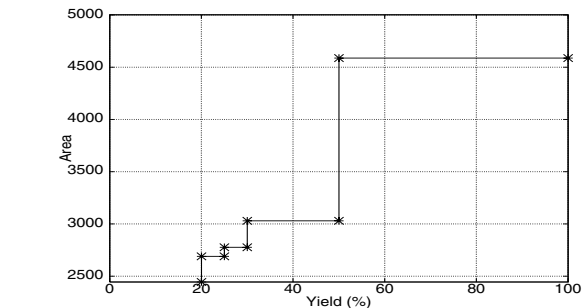
## 5 Discussions and Conclusions

In this paper we have presented a new algorithm for reticle floorplanning in multi-project wafers. The challenging aspect of reticle floorplanning is the need to optimize both the packing area and the yield. Previous approaches lead to suboptimal results and offer no guarantees with respect to either area or yield. We propose a reticle floorplanning algorithm that guarantees a prescribed level of yield, and optimizes the floorplan area as an objective. Our empirical validation shows that our approach delivers solutions along a Pareto frontier that trades yield versus area. Moreover, our solution quality dominates that achieved by previous approaches, i.e., we achieve packings of higher yields in less area. For the 10 benchmarks studied in [4], we achieve a yield improvement of 14% with an area reduction of 2%.

A number of extensions are possible for future work: (i) satisfying different production volume requirements since different dies may have different production volumes, and (ii) respecting reticle aspect ratio restrictions.

## References

[1] “<http://vlsicad.eecs.umich.edu/bk/parquet/>,” *Parquet Floorplanner*.



**Figure 4. Area versus yield tradeoff establishes a Pareto frontier.**

- [2] “<http://www.xyalis.com/>,” *Xyalis GTSuite*.
- [3] H. Murata, K. Fujiyoshi, S. Nakatake, and Y. Kajitani, “VLSI Module Placement Based on Rectangle-Packing by the Sequence Pair,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 15(12), pp. 1518–1524, 1996.
- [4] A. B. Kahng, I. Mandoiu, Q. Wang, X. Xu, and A. Zelikovsky, “Multi-Project Reticle Floorplanning and Wafer Dicing,” in *ACM/IEEE International Symposium on Physical Design*, 2004, pp. 70–77.
- [5] M. Andersson, C. Levcopoulos, and J. Gudmundsson, “Chips on Wafers,” in *Proc. Workshop on Algorithms and Data Structures*, 2003.
- [6] G. Xu, R. Tian, D. F. Wong, and A. Reich, “Shuttle Mask Floorplanning,” in *Proc. SPIE*, 2003, pp. 185–194.