Metal-Density Driven Placement for CMP Variation and Routability

Tung-Chieh Chen[1*]   Minsik Cho[2]   David Z. Pan[2]   Yao-Wen Chang[13]

[1]Graduate Institute of Electronics Engineering, National Taiwan University, Taipei 106, Taiwan
[2]Department of Electrical & Computer Engineering, University of Texas at Austin, TX 78712
[3]Department of Electrical Engineering, National Taiwan University, Taipei 106, Taiwan

donnie@eda.ee.ntu.edu.tw; thyeros@cerc.utexas.edu; dpan@ece.utexas.edu;
ywchang@cc.ee.ntu.edu.tw

ABSTRACT

In this paper, we propose the first metal-density driven placement algorithm to reduce CMP variation and achieve higher routability. Based on an analytical placement framework, we use a probabilistic routing model to estimate the wire density during the placement. Then, the metal density and thickness are predicted by a predictive CMP model. The spreading forces are adjusted according to the metal density map to reduce the metal density variation. Experimental results show that our method reduces the topography variation by 12% and the number of dummy fills by 6% and achieves much better routability, compared with wirelength-driven placement.

Categories and Subject Descriptors: B.7.2 [Integrated Circuits]: Design Aids

General Terms: Algorithms, Design, Performance

Keywords: VLSI, Physical Design, Placement, Manufacturability

1. INTRODUCTION

For 90nm and more advanced process techniques, manufacturability and yield related issues are becoming more and more important. Especially, topography (thickness) variation after chemical-mechanical planarization/polishing (CMP), i.e., CMP variation, is shown to be systematically determined by wire density distribution [20, 24, 32]. Even after CMP, intra-chip topography variation can still be on the order of 20–40% [14, 24]. Such topography variation leads to not only significant performance degradation due to increased wire resistance and capacitance, but also serious manufacturing issues like etching and printability [14,24,28]. If the copper thickness of interconnect is more uniform, the timing analysis in the early stage can be more accurate, and the timing can be optimized more effectively.

Cho et al. proposed a wire-density driven global routing to reduce CMP variation [9]. Their results showed that the

---

---

wire-density driven global router can lead to 7.5% less topography variation, 2.2% less dummy fills, and better routability with 28.6% less routing overflows. However, the wire density is highly related to placement since the wire density optimization in routing is limited by pin locations [8]. Since a router shall complete all the connections among pins, it cannot avoid routing through some highly dense region in which a pin lies. Therefore, effectively distributing pins/cells into a placement region with wire density consideration can provide better flexibility for routing, leading to better wire density topography.

Although there are many recent works on placement optimizing for wirelength [4, 5, 7, 15, 18, 25, 27, 29], cell density [7, 25], congestion [21, 26, 31], and timing [16], none of them considers the CMP variation. Although the cell density has been considered for placement [1,7,25], it cannot address the metal density well. Since the wire density among cells of a functional unit is usually much higher than the wire density between cells of different functional units, simply evenly distributing cells cannot guarantee uniform metal density. Congestion and metal density are related, but the congestion-driven placement still cannot reflect the metal density directly and effectively. Wire density is computed for the wires inside a bin, while congestion is computed for the wires crossing the edges of a bin [9].

In this paper, we propose a metal-density driven placement algorithm for CMP variation and routability. A probabilistic routing model [22,30] and a predictive CMP model [9] are used to evaluate the metal density. Then, the metal density map is used to guide the block (cell/macro) spreading to find a uniform metal-density result. The experimental results based on five adaptec benchmarks [1] show that our metal-density driven placement can effectively reduce the topography variation by 12%, compared with the traditional wirelength-driven placement. In addition, the metal-density driven placement can also lead to higher routability since there are fewer high wire-density regions. We used BoxRouter [8] to perform global routing to evaluate the routability. All circuits placed by the metal-density driven placement do not have any routing overflow, while only one circuit placed by the wirelength-driven placement has no overflow. Higher routability from our metal-density driven placement enables BoxRouter to achieve 33.4X speedup, compared with BoxRouter after wirelength-driven placement. These results show that our metal-density driven placement is effective in improving both CMP variation and routability.

The remainder of this paper is organized as follows. Section 2 gives essential background for the analytical placement, CMP model, and metal density estimation. Our metal-

Table 1: The notations in this paper.

x_v, y_v	center coordinate of block v
w_v, h_v	width and height of block v
w_b, h_b	width and height of bin b
a_b^p	area of preplaced blocks in bin b
a_b^m	area of movable blocks in bin b
a_b^u	the maximum allowable area in bin b
d_b^v, d_b^h	the wire density in the vertical/horizontal routing layer of bin b
m_b^v, m_b^h	the metal (wire + dummy) density in the vertical/horizontal routing layer of bin b
r^h	the average metal width for horizontal routings
r^v	the average metal width for vertical routings
u	target cell density

density driven placement algorithm is explained in Section 3. Section 4 reports the experimental results. Finally, the conclusions are given in Section 5.

2. PRELIMINARIES

We describe the placement model, placement metrics, and the predictive CMP model in the following.

2.1 Placement Model

We use a hypergraph $H = (V, E)$ to model a circuit. Let vertices $V = \{v_1, v_2, ..., v_n\}$ represent blocks (cells and macros), and hyperedges $E = \{e_1, e_2, ..., e_m\}$ represent nets. Let x_{v_i} and y_{v_i} be the x and y coordinates of the center of block v_i, respectively. The circuit may contain some *preplaced blocks* which have fixed x and y coordinates and cannot be moved. Table 1 gives the notation used in this paper.

2.2 Placement Metrics

The main purpose for placement is to find desired positions for all blocks such that some placement metrics are optimized. The following gives important placement metrics for modern circuit designs.

- **Wirelength** is the main objective for placement. Usually shorter wirelength leads to better routability and timing. However, simply considering total wirelength is not enough for modern circuit designs. For example, squeezing blocks can reduce total wirelength but may be harmful for routability.

- **Routability** is an important metric for both placement and routing. A placement with high routability can reduce the routing time and result in fewer detour routes. It is important to have fewer detour routes so that the wirelength estimation in the placement stage can be more accurate.

- **Cell density** is also an important consideration for modern placement. Since buffer insertion and gate sizing are commonly used in modern designs, some whitespace should be reserved for future optimization.

- **Manufacturability** also needs to be considered in the placement stage for deep submicron designs. Cell/macro positions roughly determine the wire density distribution. To effectively reduce the CMP variation, we shall change cell/macro positions.

2.3 Predictive CMP model

We use the predictive CMP model proposed in [8]. The metal thickness variation after CMP is determined by metal
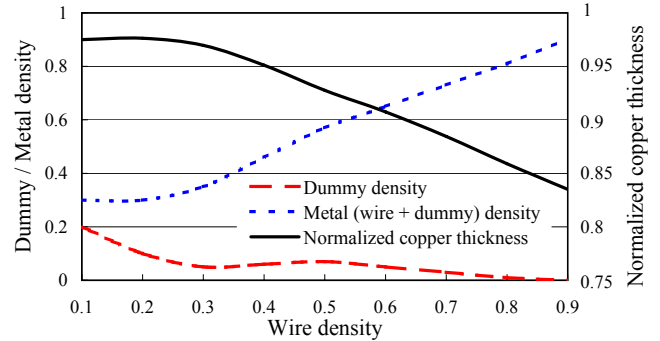

Figure 1: The predictive CMP model.

density that includes both wires and dummies. The number of dummy fills depends on the wire density. Thus, we can predict the resulting normalized copper (Cu) thickness from the wire density. The normalized Cu thickness t_b can be computed by

$$t_b = \alpha \left(1 - \frac{m_b^2}{\beta}\right), \qquad 0.2 \leq m_b \leq 0.8, \qquad (1)$$

where m_b is the metal density of a bin, α and β are technology dependent constants. The metal density m_b includes the wire density and the dummy density in a bin. Figure 1 shows the required dummy density and the predicted Cu thickness with respect to the wire density. Given the wire density d_b, we can look up the number of dummy fills to be inserted to obtain the total metal density m_b. Then, the final Cu thickness can be predicted using Eq. (1). This predictive model is verified with a commercial CMP simulator [3] and industry test cases.

3. METAL-DENSITY DRIVEN PLACEMENT

Our metal-density driven placement is based on the analytical placement framework. We use metal-density-aware spreading forces to guide block spreading to reduce CMP variation. Figure 2 illustrates our placement flow. The wire density is first updated during the placement process based on the predictive CMP model described in Section 2.3. Then, the metal density topography is fed back to the placement database so that the analytical placement can continue with metal-density-aware block spreading to reduce the metal density variation.
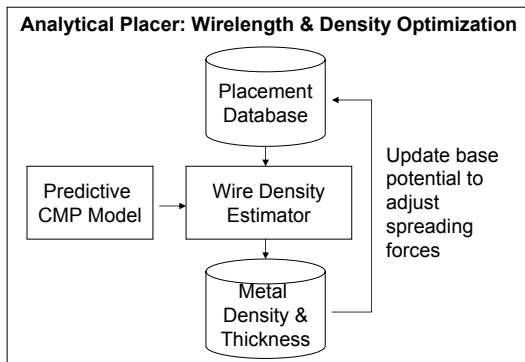
Figure 2: Our metal-density driven placement flow.

3.1 Placement Framework

The analytical placement framework optimizes wirelength and spreads blocks to reduce the overlaps between blocks [7]. To evenly distribute the blocks, we divide the placement region into uniform non-overlapping bin grids. Then, the global placement problem can be formulated as a constrained minimization problem as follows:

$$\begin{array}{ll} \min & W(\mathbf{x}, \mathbf{y}) \\ \text{s.t.} & D_b(\mathbf{x}, \mathbf{y}) \leq a_b^u, \quad \text{for each bin b,} \end{array} \qquad (2)$$

where $W(\mathbf{x}, \mathbf{y})$ is the wirelength function, $D_b(\mathbf{x}, \mathbf{y})$ is the potential function that is the total area of movable blocks in bin b, and a_b^u is the maximum allowable area of movable blocks in bin b. a_b^u can be expressed as

$$a_b^u = u(w_b h_b - a_b^p), \qquad u \leq 1.0, \qquad (3)$$

where u is a user-specified target cell density value for each bin, w_b (h_b) is the width (height) of bin b, and a_b^p is the *base potential*. The base potential equals the preplaced block area to prevent blocks from being overlapped with preplaced blocks.

The wirelength $W(\mathbf{x}, \mathbf{y})$ is defined as the total half-perimeter wirelength (HPWL):

$$W(\mathbf{x}, \mathbf{y}) = \sum_{\text{net } e} (\max_{v_i, v_j \in e} |x_{v_i} - x_{v_j}| + \max_{v_i, v_j \in e} |y_{v_i} - y_{v_j}|). \quad (4)$$

Since $W(\mathbf{x}, \mathbf{y})$ is not smooth and non-convex, it is hard to minimize it directly. Thus, several smooth wirelength approximation functions are proposed, such as quadratic wirelength [12, 19], L_p-norm wirelength [6, 18], and log-sum-exp wirelength [5, 17, 23].

We express the function $D_b(\mathbf{x}, \mathbf{y})$ as

$$D_b(\mathbf{x}, \mathbf{y}) = \sum_{v \in V} P_x(b, v) P_y(b, v), \qquad (5)$$

where P_x and P_y are the overlap functions of bin b and block v along the x and y directions. However, the overlap functions P_x and P_y are neither smooth nor differentiable. We adopt the bell-shaped potential function $\tilde{P}_x$ to smooth P_x. The bell-shaped function was first proposed in [23] and then was extended to handle mixed-size blocks in [17]. We use the later version of the bell-shaped function $\tilde{P}_x$, which is defined by

$$\tilde{P}_x(b, v) = \\ \begin{cases} 1 - p l_x^2, & 0 \leq l_x \leq \frac{w_v}{2} + w_b \\ q(l_x - \frac{w_v}{2} - 2w_b)^2, & \frac{w_v}{2} + w_b \leq l_x \leq \frac{w_v}{2} + 2w_b \\ 0, & \frac{w_v}{2} + 2w_b \leq l_x, \end{cases} \qquad (6)$$

where

$$\begin{aligned} p &= \frac{4}{(w_v + 2w_b)(w_v + 4w_b)} \\ q &= \frac{2}{w_b(w_v + 4w_b)}, \end{aligned} \qquad (7)$$

w_b is the bin width, w_v is the block width, and l_x is the center-to-center distance of the block v and the bin b in the x direction.

The quadratic penalty method is used to solve Eq. (2), implying that we solve a sequence of unconstrained minimization problems of the form

$$\min \quad W(\mathbf{x}, \mathbf{y}) + \lambda \sum_b (D_b(\mathbf{x}, \mathbf{y}) - a_b^u)^2 \qquad (8)$$

with increasing λ's. The solution of the previous problem is used as the initial solution for the next one. We solve the unconstrained problem in Eq. (8) by the conjugate gradient (CG) method.

3.2 Metal Density Estimation

To compute the metal density, we need to know the wire density first. Based on the bin structure, the wire density of a bin can be computed by the numbers of tracks going through four edges of the bin. Let t_b^t, t_b^b, t_b^l, and t_b^r be the numbers of tracks going through the left, top, right, and bottom edge of the bin b, respectively. In each bin, we use two layers, a vertical routing layer and a horizontal routing layer, to compute the wire density. The average wire density d_b^v in the vertical routing layer of bin b can be estimated by

$$d_b^v = r^v \left(\frac{t_b^t + t_b^b}{2w_b} \right) + b_b^v, \qquad (9)$$

and the average wire density d_b^h in the horizontal routing layer of bin b is

$$d_b^h = r^h \left(\frac{t_b^l + t_b^r}{2h_b} \right) + b_b^h, \qquad (10)$$

where r_w^v (r_w^h) is the average metal width for vertical (horizontal) routing, w_b (h_b) is the bin width (height), and b_b^v (b_b^h) is the base density of the vertical (horizontal) routing layer contributed by the internal routing of the cells/macros. After obtaining the wire density, we can use the predictive CMP model to compute the metal density and the resulting Cu thickness.

To predict the expected horizontal/vertical track usage for each bin, we first decompose multi-terminal nets into Steiner trees using FLUTE [10, 11]. Then, the track usage for each two-pin net is estimated by the probabilistic routing model [22, 30]. Figure 3 gives an example. For a two-pin net from S to T, there are two possible L-shaped routes and three possible Z-shaped routes. The expected track usage for each edge is computed by the number of possible routes that go through it divided by the number of the total possible routes.

3.3 Metal-Density-Aware Block Spreading

To reduce the metal density variation, the placement needs to have uniform wire density. We use Figure 4 to illustrate the concept of reducing the wire density variation. Usually the high wire density region is caused by not only local nets but also global nets. In Figure 4(a), there are three nets in the central region, two local nets and one global net. Then, extra forces are applied to the blocks in the central region
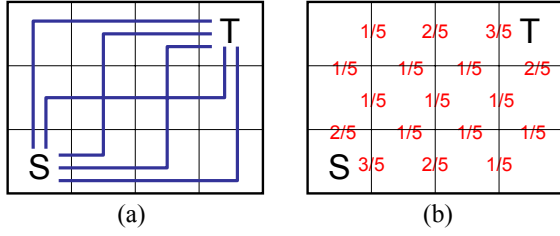
(a)   (b)

Figure 3: The probabilistic routing model for a two-pin net from S to T. (a) Five possible L- and Z-shaped routings. (b) The expected track usage for each edge.
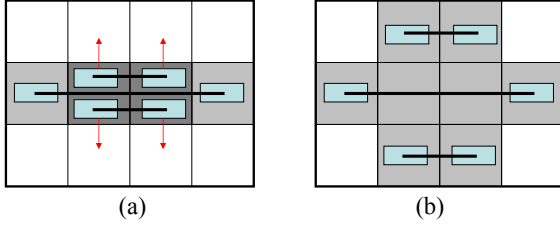

(a)   (b)

Figure 4: The concept of reducing the wire density. (a) The blocks in the high metal-density regions receive forces to leave the region. (b) After moving blocks, a more uniform wire density result is obtained.

to push out blocks from this region to obtain the result in Figure 4(b). As a result, the metal density in the central region is reduced and a more uniform metal density result is obtained.

Simply adding extra forces or directly moving blocks during the analytical placement may cause the numerical instability. Thus, we add extra forces implicitly by modifying the base potential to maintain the stability of the nonlinear solver. The original base potential a_b^p considers only preplaced blocks; the metal-density-aware base potential a_b^{p+m} considers both preplaced blocks and metal density,

$$a_b^m = m_b^v - \min_{\text{bin } b} m_b^v + m_b^h - \min_{\text{bin } b} m_b^h, \qquad (11)$$

$$a_b^{p+m} = \min(a_b^p + \kappa a_b^m, w_b h_b), \qquad (12)$$

where κ is a parameter to control the strength of the metal-density-aware forces. The value of a_b^{p+m} is restricted to be less than the area of bin b to ensure the numerical stability. Then, we use the new base potential to compute the maximum allowable block area $a_b^{u'}$,

$$a_b^{u'} = u(w_b h_b - a_b^{p+m}). \qquad (13)$$

As a result, the placement problem becomes

$$\min \quad W(\mathbf{x}, \mathbf{y}) + \lambda \sum_b (D_b(\mathbf{x}, \mathbf{y}) - a_b^{u'})^2. \qquad (14)$$

We can adjust κ in Eq. (12) according to the total available whitespace of the design. The total maximum allowable block area in a placement region must be large enough to contain all movable blocks:

$$\sum_{\text{bin } b} a_b^{u'} \geq \text{area\_of\_all\_movable\_blocks}. \qquad (15)$$

If we set κ larger, we may have a more uniform metal-density

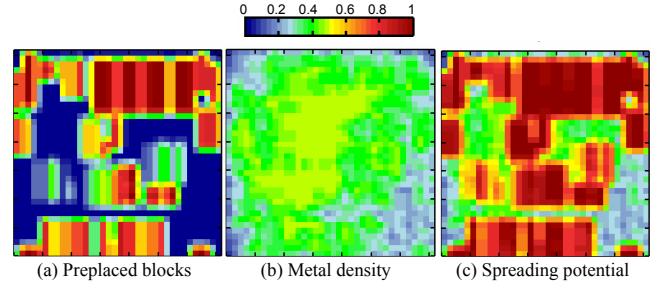(a) Preplaced blocks   (b) Metal density   (c) Spreading potential

Figure 5: The metal-density-aware base potential generation. (a) The preplaced block density. (b) The predicted metal density. (c) The base potential considering both (a) and (b).
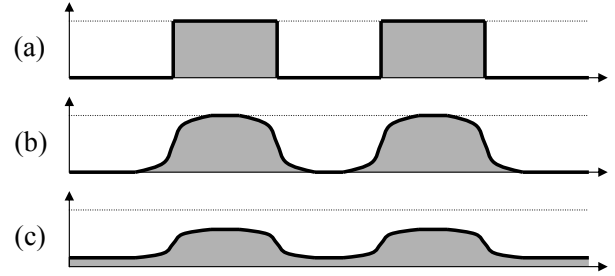

Figure 6: A two-step smoothing example. (a) Original density. (b) After Gaussian smoothing. (c) After level smoothing.

result, but the resulting wirelength may be worse. This parameter controls the tradeoff between the wirelength and the metal density variation. The effects of adjusting κ will be reported in Section 4.

Figure 5 gives an example of computing the metal-density-aware base potential. Figures 5(a) and (b) show the preplaced block density and the predicted metal density, respectively. Summing up the two density values using Eq. (12), we can obtain the resulting base potential as shown in Figure 5(c).

3.4 Base Potential Smoothing

A smooth objective function helps the gradient method to find a desired solution. Thus, we need to smooth the base potential to achieve better solutions. We apply the two-stage smoothing technique [7], *Gaussian smoothing* followed by *level smoothing* to smooth metal-density-aware base potential.

We use Figure 6 to explain our smoothing method. First, Gaussian smoothing is applied to the configuration of Figure 6(a) and avoid the dramatic change in the landscape to obtain the configuration of Figure 6(b). Then, level smoothing is applied to reduce the potential levels to obtain the configuration of Figure 6(c)

Gaussian smoothing works as a low-pass filter, which can smooth the local potential change. The two-dimensional Gaussian has the form

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}, \qquad (16)$$

where σ is the standard deviation of the distribution. Applying convolution $(*)$ to the Gaussian function G with the

base potential P,

$$S(x, y) = G(x, y) * P(x, y), \qquad (17)$$

we can obtain a smoother base potential S. The value σ defines the smoothing range. A larger σ leads to a more smooth potential. In global placement, the smoothing range gradually decreases so that the smoothed potential approaches the exact density gradually.

After the Gaussian smoothing, we apply another landscape smoothing function [13] to reduce the potential levels. As a result, the final smoothed based potential $\tilde{P}$ is given by

$$\tilde{P}(x, y) = \begin{cases} \overline{S} + (S(x,y) - \overline{S})^\delta & \text{if } S(x,y) \geq \overline{S} \\ \overline{S} - (\overline{S} - S(x,y))^\delta & \text{if } S(x,y) < \overline{S}, \end{cases} \qquad (18)$$

where $\overline{S}$ is the average value of $S(x,y)$, and $\delta \geq 1$. We normalize S so that every S is between 0 and 1 to ensure $|S(x,y) - \overline{S}| < 1.0$. Smoothing potential levels reduces the height of high potential regions so that movable blocks can spread to the whole placement region smoothly.

In summary, there are three parameters for generating smoothing metal-density-aware base potential. These three parameters are controlled empirically as follows.

- κ controls the strength of the metal-density-aware forces by allocating whitespace for the metal-density-aware base potential. The amount of allocated whitespace gradually increases to the user-specified whitespace ratio during the placement.

- σ controls the range of the Gaussian smoothing. The smoothing range starts from 15% of the chip width and gradually decreases to around 1% of the chip width.

- δ controls the degree of level smoothing. It decrease from 5 to 1. When $\delta = 1$, there is no level smoothing.

3.5 Placement Flow

Figure 7 shows our placement flow. Our metal-density driven placement is based on a multilevel analytical placement framework. First, we iteratively cluster blocks to obtain the hierarchy. The cluster positions are initialized by solving the minimum quadratic wirelength, which is widely used in quadratic placement.

There are three loops in the placement flow.

- The 1st loop is the multilevel loop (in lines 3–17). The bin dimension is initialized so that the number of bins is proportional to the number of clusters in the current level. After finding the optimal placement of the current level, the circuit hierarchy is declustered once.

- The 2nd loop is the block spreading loop (in lines 5–15). The smoothing parameters, κ, σ, and δ, are updated in this loop. Then, the value of λ_m in Eq. (8) gradually increases inside the loop to spread blocks to the desired positions.

- The 3rd loop is to find the minimal value of Eq. (8) by the conjugate gradient method (in lines 8–13). Since all clusters/blocks are moving inside this loop, we need to update the metal density map (line 10) and corresponding base potential (line 11) inside this loop.

The placement progress continues until all clusters are declustered and the value of Eq. (8) cannot be further reduced. Then, we legalize the placement by removing all overlaps and report the final result.

Multilevel Metal-Density Driven Placement

Input: a circuit hypergraph
Output: desired block positions

```
01.   create the clustering hierarchy;
02.   initialize block positions;
03.   do
04.       initialize the bin structure and λ;
05.       do
06.           update smoothing parameters;
07.           // find min W(x, y) + λ ∑(D_b − a_b^{u'})²;
08.           do
09.               compute the conjugate gradient direction;
10.               update current block positions;
11.               estimate wire and metal density;
12.               update base potential;
13.           until (the minimal value is found);
14.           increase λ by 2;
15.       until (spreading enough);
16.       decluster blocks;
17.   until (all clusters are declustered);
18.   legalize the placement;
19.   return block positions;
```

Figure 7: Our placement flow.

3.6 Extension of Handling Multiple Metal Layers

In modern VLSI designs, there are usually multiple metal layers. Our method can be extended to optimize multiple metal layers directly. First, a fast global router with layer assignment is needed to find the number of tracks used for each edge of the bin in each layer. Based on the track usage, we can use using Eq. (9) and Eq. (10) to compute the wire density for each layer. Then, Eq. (11) can be modified as

$$a_b^m = \sum_{\text{Layer } l} w^l \left(m_b^l - \min_{\text{bin } b} m_b^l \right), \qquad (19)$$

where m_b^l is the metal density of bin b in layer l, and w^l is the weight of the layer l. Because it is important to have smaller metal density variation for lower layers, we can set higher weights for lower layers than those of upper layers. After computing the new a_b^m, we can use Eq. (12) to compute the new potential and Eq. (14) to solve the multi-layer metal-density driven placement problem.

4. EXPERIMENTAL RESULTS

We implemented the proposed algorithm in C++ by augmenting the placer NTUplace3 [7]. All the experiments were performed on a 2.2GHz AMD Opteron machine. The adaptec benchmarks are taken from the ISPD'06 placement contest [1]. The circuit information is shown in Table 2. The number of movable blocks ranges from 211K to 842K, and the number of nets ranges from 221K to 868K. The design utilization rate ranges from 27% to 57%. We follow the routing configurations used in the ISPD 2007 routing contest [2]. Six metal layers are assumed, and only 20% wire tracks are available for routing in metal layers 1 and 2. The size of the global routing tile in terms of track numbers (Track#) and macro block porosity (Blk.) is shown in the last two columns of Table 2. The block porosity defines the remaining amount of routing resource above macro blocks in metal layers 3 and 4. Two experiments were conducted. In the first experiment, we studied the tradeoff between HPWL and metal density standard deviation by adjusting the parameter κ described in Section 3.3. In the second experiment,

Table 2: Benchmark statistics.

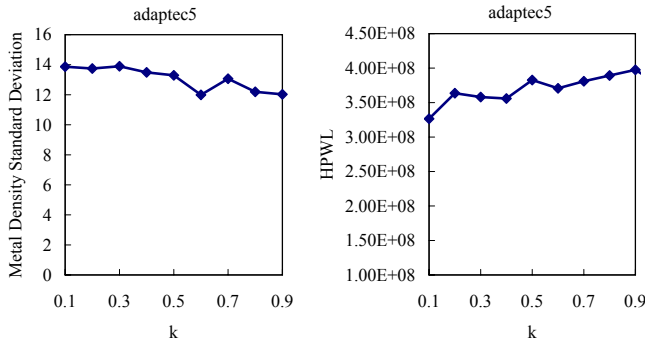| Circuit	Placement				Routing	
	Mov #	Fixed #	Net #	Util. (%)	Track #	B.P.[†] (%)
adaptec1	211k	543	221k	57	35	50
adaptec2	254k	566	266k	44	35	20
adaptec3	451k	723	467k	33	30	50
adaptec4	495k	1329	516k	27	30	50
adaptec5	842k	646	868k	49	50	20

[†] macro block porosity

Figure 8: The metal density standard deviation and HPWL versus κ for adaptec5.

we compared our metal-density driven placement with two other placement methods. For this experiment, BoxRouter was used to perform global routing,[1] and Cu thickness variation was evaluated by the predictive CMP model [8].

4.1 Tradeoff Between HPWL and Metal Density Variation

Since κ controls the force strength to even the metal density implicitly in Eq. (12), we can adjust the force strength by controlling κ. Figure 8 gives the resulting HPWL and the metal density standard deviation using different κ's for the circuit adaptec5. We only show the results of adaptec5 since all other circuits have the same trend. The x-axis is normalized by the amount of whitespace used. For example, the number 0.9 means that κ is adjusted to use 90% whitespace for metal-density-aware base potential. From the figure, when κ is larger, the resulting metal density standard deviation is smaller but the HPWL is longer. The resulting metal density standard deviation can be reduced by 15% while HPWL is increased by 18% when 90% whitespace is used for metal-density-aware spreading forces.

4.2 Comparison Between Different Placement Algorithms

Since there is no previous work on CMP-aware placement, we compared three different placement modes based on our placer, *wirelength-driven (WLD)*, *cell-density driven (CDD)*, and *metal-density driven (MDD)* modes. The WLD placement optimizes wirelength alone with the target cell density $u = 1.0$. For CDD placement, we set the target cell density to its design utilization rate to evenly spread blocks to the whole chip region. The MDD placement algorithm is described in Section 3. The parameter κ is set to use 90%

---

[1]We used the default parameters for BoxRouter, which may spend more runtime to achieve fewer overflows. Note that BoxRouter completed the routing for most circuits with zero overflows in the ISPD'07 routing contest [2].

---

whitespace for the metal-density-aware base potential. Table 3 gives the placement, routing, and CMP results on the adaptec benchmarks. "HPWL" is the half-perimeter wirelength after placement, while "WL" is the total wirelength after global routing. "WL/HPWL" is the ratio of WL and HPWL, which stands for the wirelength increase after global routing. "Overflow" is the number of total routing overflows after global routing. The overflow is defined as the sum of the number of tracks that exceeds the routing capacity for each edge of the global routing tiles. "Dummy" is the number of dummy fills based on the predictive CMP model. "Cu-Avg" and "Cu-Std" are the average and standard deviation of the normalized copper thickness, respectively.

CMP. The average Cu thicknesses are similar for the three placement modes. Compared with WLD's variation, CDD averagely reduces 13% variations of the Cu thickness, and MDD can further reduce 3% more, 11–12% in total, variations. In addition to Cu thickness variation reduction, the total number of dummy fills for MDD is 2% less than CDD's and 6% less than WLD's. It is important to have fewer dummy fills because it not only can increase circuit performance by reducing coupling capacitance, but also can save manufacturing cost by decreasing its mask data volume.

Wirelength. The MDD's HPWL is 8% longer than CDD's and 19% longer than WLD's. After global routing, The MDD's WL is 4% longer than CDD's and 11% longer than WLD's. However, it should be noted that since there are still some overflows in CDD's and WLD's routing results, it is not fair to simply compare the WL, as each overflow can cause significant overheads in final (detailed) wirelength. For the ratio of WL/HPWL, MDD incurs only a 17% increase while CDD and WLD incur 22% and 29% increases in wirelength, respectively. The difference in wirelength increases is mainly because of detour routes, implying that there are more detours in CDD's and WLD's routing results than in MDD's.

CPU time. The CDD placement time is 14% more than the WLD one since it takes more time to spread block to the whole chip. The MDD placement time is the longest, 24% more than WLD's; the major placement time penalty comes from the computation for the metal density. However, the routing time of MDD is the smallest, 4.67X faster than CDD, and 33.42X faster than WLD. If we consider both placement and routing, MDD used the least runtime since routing time dominates the total runtime.

Routability. A placement with better routability usually has fewer routing overflows, less routing time, and less wirelength increase. All placements obtained by MDD do not have any overflow, while only 3 (1) placements obtained by CDD (WLD) have no overflow. MDD also has the smallest routing time and wirelength increase, implying that MDD's placement results have higher routability. We also found that pure wirelength-driven placement usually generates non-routable results for modern circuit designs. Controlling the cell density can result in better routability, but metal-density driven placement has the best routability among the three modes because there are fewer high wire-density regions in its resulting placement.

The aforementioned results all show that metal-density driven (MDD) placement leads to not only more uniform metal density distribution, but also better routability. Figures 9 and 10 give the respective resulting normalized Cu thickness maps in the vertical and horizontal routing layers for adaptec5, using the three placement modes. The standard deviation of the topography variations is shown below the figures.

Table 3: The comparisons of our placer using different modes for the adaptec benchmarks.

	Placement		Routing				CMP (Hori. Layer)			CMP (Vert. Layer)		
	HPWL	CPU	WL	WL/	Over-	CPU	Dummy	Cu-Avg	Cu-Std	Dummy	Cu-Avg	Cu-Std
Circuit	($\times$ e7)	(min)	($\times$ e7)	HPWL	flow	(min)						
Wirelength-Driven (WLD)												
adaptec1	8.15	14	11.87	1.46	0	711	5984	0.95	3.16	6034	0.95	3.09
adaptec2	9.02	15	12.38	1.37	2758	2279	13567	0.96	3.84	13644	0.96	4.03
adaptec3	22.35	32	26.27	1.18	938	140	45432	0.96	7.48	43843	0.96	7.78
adaptec4	20.02	29	22.38	1.12	31	1839	51258	0.96	7.81	50322	0.96	7.73
adaptec5	36.09	76	47.36	1.33	26672	2386	16766	0.96	5.73	16462	0.96	5.70
Comp.	0.81	0.76	0.89	1.29	—	33.42	1.06	1.00	1.12	1.06	1.00	1.11
Cell-Density Driven (CDD)												
adaptec1	9.20	15	12.12	1.32	0	130	5929	0.95	3.09	5986	0.95	3.14
adaptec2	10.32	17	13.08	1.27	0	208	13166	0.96	3.50	13248	0.96	3.83
adaptec3	27.42	28	31.19	1.14	28	1119	43575	0.96	7.24	42096	0.96	7.44
adaptec4	22.61	52	24.67	1.09	0	30	49078	0.96	7.36	48174	0.96	7.18
adaptec5	38.90	67	49.62	1.29	875	1788	15972	0.96	4.58	15560	0.96	4.54
Comp.	0.92	0.90	0.96	1.22	—	4.67	1.02	1.00	1.03	1.02	1.00	1.03
Metal-Density Driven (MDD)												
adaptec1	9.41	17	11.54	1.23	0	42	5914	0.95	3.02	5927	0.95	3.03
adaptec2	11.63	23	13.63	1.17	0	28	12913	0.96	3.36	12952	0.96	3.69
adaptec3	30.35	38	34.10	1.12	0	121	42253	0.96	7.06	40525	0.96	7.24
adaptec4	26.40	35	28.69	1.09	0	28	46481	0.96	7.05	45741	0.96	7.03
adaptec5	39.82	110	48.73	1.22	0	707	16198	0.96	4.52	15687	0.96	4.45
Comp.	1.00	1.00	1.00	1.17	—	1.00	1.00	1.00	1.00	1.00	1.00	1.00

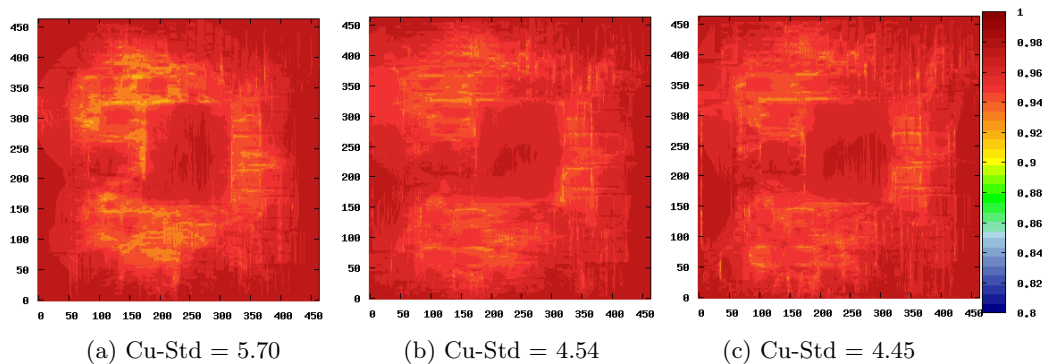(a) Cu-Std = 5.70   (b) Cu-Std = 4.54   (c) Cu-Std = 4.45

Figure 9: The normalized Cu thickness map in the vertical routing layer for adaptec5 using (a) wirelength-driven placement, (b) cell-density driven placement, and (c) metal-density driven placement.

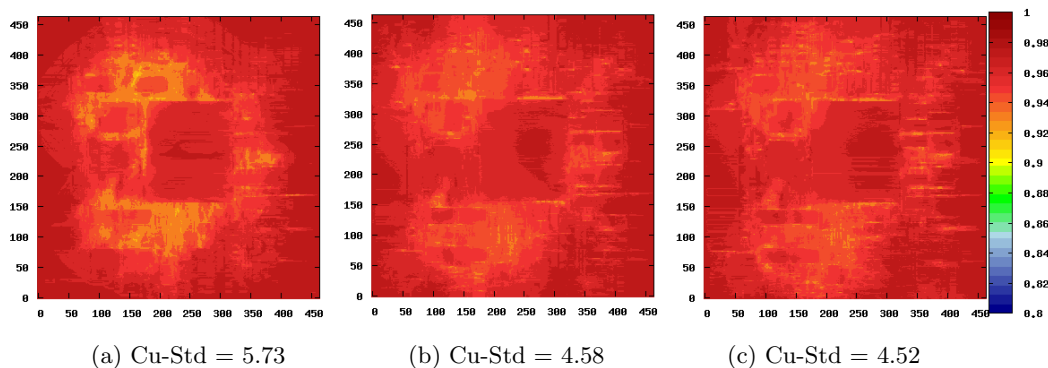(a) Cu-Std = 5.73   (b) Cu-Std = 4.58   (c) Cu-Std = 4.52

Figure 10: The normalized Cu thickness map in the horizontal routing layer for adaptec5 using (a) wirelength-driven placement, (b) cell-density driven placement, and (c) metal-density driven placement.

5. CONCLUSIONS

Metal density is an important issue for manufacturability of nanometer circuit designs. We have presented the first metal-density driven placement to reduce CMP variation and improve routability. Experimental results have shown that the proposed metal-density driven placement algorithm reduces the copper thickness variation by 12% and dummy fills by 6%, compared with the wirelength-driven placement. In addition, the results generated by our metal-density-driven placement algorithm lead to higher routability.

6. ACKNOWLEDGMENTS

7. REFERENCES

[1] *ISPD 2006 Placement Contest.* http://www.sigda.org/ispd2006/contest.html.

[2] *ISPD 2007 Global Routing Contest.* http://www.sigda.org/ispd2007/ispd07_contest.html.

[3] *Praesagus, Inc.* http://www.praesagus.com/.

[4] A. R. Agnihotri, S. Ono, C. Li, M. C. Yildiz, A. Khatkhate, C.-K. Koh, and P. H. Madden. Mixed block placement via fractional cut recursive bisection. *IEEE Trans. Computer-Aided Design*, 24(5):748–761, May 2005.

[5] T. Chan, J. Cong, J. Shinnerl, K. Sze, and M. Xie. mPL6: Enhanced multilevel mixed-size placement. In *Proc. ACM Int. Symp. on Phys. Des.*, pages 212–214, San Jose, CA, Apr. 2006.

[6] T. Chan, J. Cong, and K. Sze. Multilevel generalized force-directed method for circuit placement. In *Proc. ACM Int. Symp. on Phys. Des.*, pages 185–192, San Francisco, CA, Apr. 2005.

[7] T.-C. Chen, Z.-W. Jiang, T.-C. Hsu, and Y.-W. Chang. A high-quality mixed-size analytical placer considering preplaced blocks and density constraints. In *Proc. IEEE/ACM Int. Conf. on Comput.-Aided Des.*, San Jose, CA, Nov. 2006.

[8] M. Cho and D. Z. Pan. BoxRouter: a new global router based on box expansion and progressive ilp. In *Proc. ACM/IEEE Des. Autom. Conf.* , pages 373–378, San Francisco, CA, July 2006.

[9] M. Cho, H. Xiang, R. Puri, and D. Z. Pan. Wire density driven global routing for cmp variation and timing. In *Proc. IEEE/ACM Int. Conf. on Comput.-Aided Des.*, pages 487–492, San Jose, CA, Nov. 2006.

[10] C. Chu and Y.-C. Wong. Fast and accurate rectilinear steiner minimal tree algorithm for vlsi design. In *Proc. ACM Int. Symp. on Phys. Des.*, pages 28–35, 2005.

[11] C. C. N. Chu. FLUTE: Fast lookup table based wirelength estimation technique. In *Proc. IEEE/ACM Int. Conf. on Comput.-Aided Des.*, pages 696–701, 2004.

[12] H. Eisenmann and F. M. Johannes. Generic global placement and floorplanning. In *Proc. ACM/IEEE Des. Autom. Conf.* , pages 269–274, June 1998.

[13] J. Gu and X. Huang. Efficient local search with search space smoothing: A case study of the traveling salesman problem (TSP). *IEEE Trans. Syst., Man, Cybern.*, 24(5):728–735, 1994.

[14] L. He, A. B. Kahng, K. Tam, and J. Xiong. Design of IC interconnnects with accurate modeling of CMP. In *Int. Society for Optical Engineering (SPIE) Symp. on Microlithopraghy*, Mar. 2005.

[15] B. Hu, Y. Zeng, and M. Marek-Sadowska. mFAR: fixed-points-addition-based VLSI placement algorithm. In *Proc. ACM Int. Symp. on Phys. Des.*, pages 239–241, San Francisco, CA, Apr. 2005.

[16] A. B. Kahng and Q. Wang. An analytic placer for mixed-size placement and timing-driven placement. In *Proc. IEEE/ACM Int. Conf. on Comput.-Aided Des.*, pages 565–572, San Jose, CA, Nov. 2004.

[17] A. B. Kahng and Q. Wang. Implementation and extensibility of an analytic placer. *IEEE Trans. Computer-Aided Design*, 24(5), May 2005.

[18] A. B. Kahng and Q. Wang. A faster implementation of APlace. In *Proc. ACM Int. Symp. on Phys. Des.*, pages 218–220, San Jose, CA, Apr. 2006.

[19] M. Kleinhans, G. Sigl, F. M. Johannes, and K. J. Antreich. Gordian: VLSI placement by quadratic programming and slicing optimization. *IEEE Trans. Computer-Aided Design*, 10(3):356–365, 1991.

[20] S. Lakshminarayanan, P. J. Wright, and J. Pallinti. Electrical characterization of the copper CMP process and derivation of metal layout rules. *IEEE Trans. Semiconduct. Manufact.*, 16(11):668–676, 2003.

[21] C. Li, M. Xie, C.-K. Koh, J. Cong, and P. H. Madden. Routability-driven placement and white space allocation. In *Proc. IEEE/ACM Int. Conf. on Comput.-Aided Des.*, pages 394–401, San Jose, CA, Nov. 2004.

[22] J. Lou, S. Thakur, S. Krishnamoorthy, and H. S. Sheng. Estimating routing congestion using probabilistic analysis. *IEEE Trans. Computer-Aided Design*, 21(1):32–41, Jan. 2002.

[23] W. C. Naylor, R. Donelly, and L. Sha. US patent 6,301,693: Non-linear optimization system and method for wire length and dealy optimization for an automatic electric circuit placer. 2001.

[24] X. Qi, A. Gyure, Y. Luo, S. C. Lo, M. Shahram, and K. Singhal. Measurement and characterization of pattern dependent process variations of interconnect resistance, capacitance and inductance in nanometer technologiesn. In *Proc. ACM Great Lakes Symp. on VLSI*, pages 14–18, New York, NY, 2006.

[25] J. Roy, D. Papa, A. Ng, and I. Markov. Satisfying whitespace requirements in top-down placement. In *Proc. ACM Int. Symp. on Phys. Des.*, pages 206–208, San Jose, CA, Apr. 2006.

[26] J. A. Roy, J. F. Lu, and I. L. Markov. Seeing the forest and the trees: Steiner wirelength optimization in placement. In *Proc. ACM Int. Symp. on Phys. Des.*, pages 78–85, San Jose, CA, Apr. 2006.

[27] P. Spindler and F. M. Johannes. Fast and robust quadratic placement combined with an exact linear net model. In *Proc. IEEE/ACM Int. Conf. on Comput.-Aided Des.*, pages 179–186, San Jose, CA, Nov. 2006.

[28] R. Tian, D. F. Wong, and R. Boone. Model-based dummy feature placement for oxide chemical-mechanical polishing manufacturability. *IEEE J. Technol. Computer Aided Design*, 20(7):902–910, 2001.

[29] N. Viswanathan, M. Pan, and C. Chu. FastPlace 2.0: An efficient analytical placer for mixed-mode designs. In *Proc. IEEE/ACM Asia South Pacific Des. Autom. Conf.*, pages 195–200, Yokomaha, Japan, Jan. 2006.

[30] J. Westra, C. Bartels, and P. Groeneveld. Probabilistic congestion prediction. In *Proc. ACM Int. Symp. on Phys. Des.*, pages 204–209, Phoenix, AZ, 2004.

[31] X. Yang, B.-K. Choi, and M. Sarrafzadeh. Routability-driven white space allocation for fixed-die standard-cell placement. In *Proc. ACM Int. Symp. on Phys. Des.*, pages 42–47, Del Mar, CA, Apr. 2002.

[32] P. Zarkesh-Ha, S. Lakshminarayanan, K. Doniger, W. Loh, and P. Wright. Impact of interconnect pattern density information on a 90nm technology ASIC design flow. In *Proc. IEEE/ACM Int. Symp. on Quality of Electronic Des.*, Nov. 2003.