Timing-Aware Fill Insertion

Bo Yang, Sriraaman Sridharan
Synopsys Inc.

Revised: April 28, 2018

I.   Introduction

It is a mandatory step for modern manufacture processes to fill the empty conductor layers with fills, and it is commonly conducted after physical design stage is finished. These fills can reduce the dielectric thickness variation, increase planarity and provide better pattern density, all of which are important to mitigate the process variability thereby achieving better yield.

II.   Problem Description

When a fill is inserted, it improves the metal density and increases planarity. But on the other hand it inevitably couples to the signal tracks (Figure 1). If the coupling capacitance to a critical net is prominent, the original timing-closure may not be achieved anymore. Therefore it's important to reduce the capacitive impact during metal fill insertion.

Figure 1: Signal track-metal fill coupling

The current main-stream solution is to set up a buffer region(minimal space of a metal track to its nearest fill, see Figure. 2) for the tracks of critical signals to keep the metal fills out of the buffer region so that the coupling capacitance introduced by the metal fills are safe to be ignored. However it's hard to achieve the best metal fill insertion with this method. It could easily result in pessimistic fill patterns around the critical nets if a large buffer region is specified (which also creates less-even density boxes that negatively impacts the yield), or a dense layout that may break the timing if a smaller buffer region is given. What's worse is that no matter how small or large a buffer region we specify, the impact to timing is unknown without involving a parasitics extraction tool. Therefore it's necessary to develop an efficient algorithm to inset metal fill wisely.
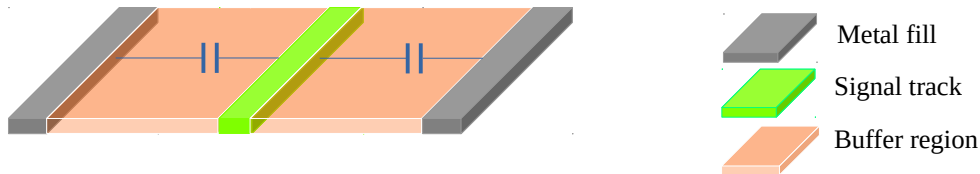

Figure 2: buffer regions

Capacitance Calculation

Three kinds of capacitance will be considered when evaluating a metal fill insertion algorithm.

1) Area Capacitance

Any two conductor pieces will form an area capacitance when 1) they are on different metal layers, 2) their projections on the ground plane overlap and 3) no other metal that appears on the layer between the two conductors crosses the overlapped region (See Figure 3 for example).
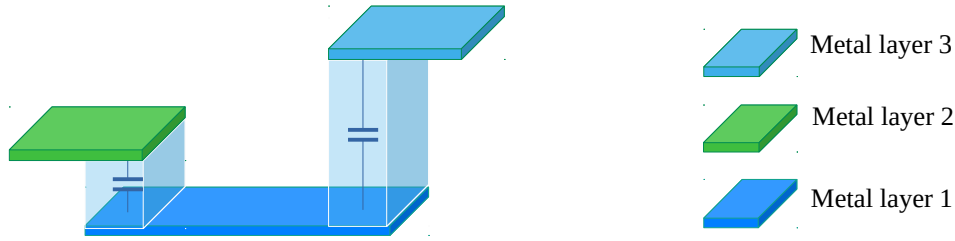
Figure 3: Area Capacitance

For an overlapped area s formed by conductors on layer l_1 and l_2, the area capacitance Ca is calculated by

$$C_a = P_{l_1,l_2}(s) \times s, \quad \text{where} \quad P_{l_1,l_2}(s) \text{ is the area capacitance per unit area and it's a function of overlapped area } s.$$

2) Lateral Capacitance

The lateral capacitance is formed by the conductors on the same layer. Any two conductors that horizontally overlap will form lateral capacitance as shown in Figure 4. The capacitance is calculated by $C_l = P_l(d) \times l$, where l is the length of the parallel edges of the conductors and $P_l(d)$ is the lateral capacitance per unit length, which is a function of distance of the parallel edges d.
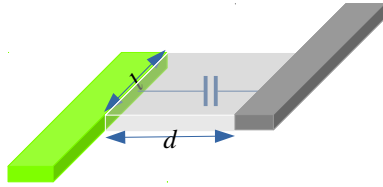

Figure 4: Lateral Capacitance

3) Fringe Capacitance

The fringe capacitance is formed by two conductors on different layers and its calculation is based on the following formula, where these two conductors don't overlap with each other.

$$C_f = \begin{cases} P_{l_1,l_2}(d) \times l + P_{l_2,l_1}(d) \times l, & d \geq 0 \\ 0, & d < 0 \end{cases}$$

where l is the length of parallel edges and d is the horizontal distance of two conductors (See Figure 5). d will be negative if any overlap of the two conductors exists (i.e., $C_a > 0$). $P_{l_1,l_2}(d)$ and $P_{l_2,l_1}(d)$ are the fringe capacitance per unit length.
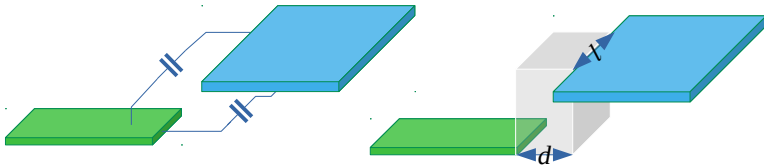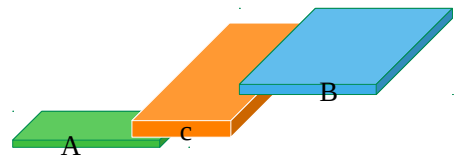

Figure 5: Fringe Capacitance

Figure 6: Shielding effect

3.1) Shielding Effect

If two conductors A, B are on the different layers *L1* and *L2* respectively, the fringe capacitance between A and B is considered to be shielded (i.e., capacitance is 0) when there is conductor existing on the intermediate layer horizontally between A and B. Please note the coupling capacitance to the shielding conductor will still be calculated. Please see Figure 6 for example. In Figure 6, the fringe capacitance between A and B is shielded by C, but the coupling capacitance (either fringe capacitance or area capacitance) between A and C will be calculated.

4) Total Capacitance Calculation
After metal fill insertion, the total capacitance of a critical net will be calculated as the equivalent capacitance to ground. Please note resistance has been ignored. All non-critical nets, except for the power/ground nets, are treated as floating (i.e., not being tied to any specific potentials). Coupling to the power/ground nets is calculated as direct-coupling to ground. Please see Figure 7 for example.
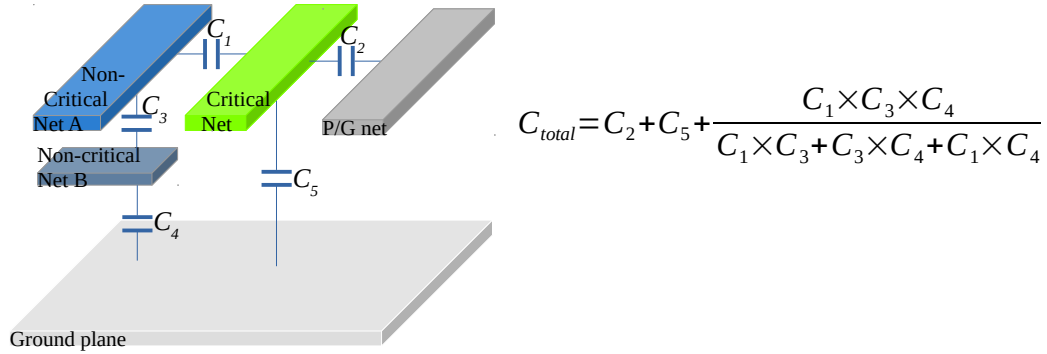

$$C_{total} = C_2 + C_5 + \frac{C_1 \times C_3 \times C_4}{C_1 \times C_3 + C_3 \times C_4 + C_1 \times C_4}$$

Figure 7: Total capacitance calculation

Metal Density

The layout with metal fill must meet the density criteria. The density is calculated in a window based manner (Figure

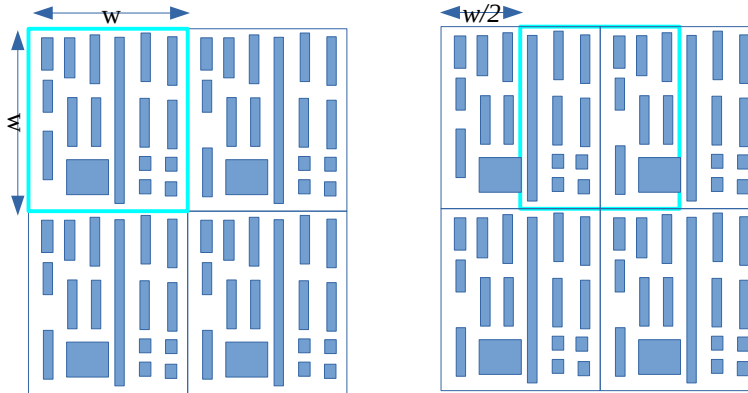
Figure 8: Density Calculation Window

8): w is the window size for density calculation, and the step length is set to w/2. The density in a window W_i is calculated as

$$D_{l,W_i} = \sum_{S \in W_i} S / w^2,$$

where l is the metal layer, and S is the metal area that is enclosed inside W_i. The density in each window must be larger than a given minimal layer metal density.

III. Algorithm Evaluation

The optimization algorithm should insert metal-fills subject to the density criteria and design rules and meanwhile the total capacitance of a given critical net should be minimized. All algorithms must not introduce any design rule

violation. Results with design rule violation will not be accepted for further evaluation. The results also need pass window-based density check. Results with density violation will be rejected as well.

The accepted result with smaller total capacitance will get higher quality score with the highest of 15 points. For example, the best result (smallest total cap) gets 15 points and the second place gets 14 points. The accepted result with shorter runtime gets higher performance score with the highest of 5 points. For example, the result with shortest runtime gets 5 points and the second place gets 4 points. Any results take longer runtime than the fifth place get 0 performance point.

All teams will be ranked with sum of quality points and performance points.

IV. Benchmark suites

Five benchmark cases will be provided. They are all text files with different number of polygons (rectangular). All lines start from a semicolon (;) are comments.

Layout File Format:
; the first non-comment line of layout file is the chip boundary: bottom_left_x bottom_left_y top_right_x top_right_y
Chip_boundary_BL_x Chip_boundary_BL_y Chip_boundary_TR_x Chip_boundary_TR_y
; Polygon_id is a positive integer. The polygon type will be one of the four types: Drv_Pin that marks this polygon is a
; drive pin of the net, Normal marks this polygon is a normal conductor, Load_Pin indicates the polygon is a load pin
; and Fill indicates this is a fill polygon. For a fill polygon, its net id can be set to any integer.
Polygon_id bl_pt tr_pt net_id layer_id <polygon type: Drv_Pin|Normal|Load_Pin|Fill>

One rule file and one process file will be provided which describe design rules and process-related information:
Rule file:
Layer_id <conductor|via> min_width min_space max_fill_width min_density max_density
Please note the density criteria does not apply to via layer, so if the second field of the rule file is "via", min_density will be set to "0" and max_density will be "1".

Process file:
;window size for density calculation (unit: nano-meter)
window: 5000
;Unit Capacitance
; when a table does not exist, it's marked by '*'
; lateral_table_* can be thought of as a special case of fringe_table_*
;layer id (Please layer id 0 means ground plane)

	1	2	... n
0	(area_table_1_0,*)	(area_table_2_0, *)	(area_table_n_0, *)
1	(*, lateral_table_1)	(area_table_1_2, fringe_table_1_2)	... (area_table_1_n, fringe_table_1_n)
2	(area_table_2_1, fringe_table_2_1)	(*, lateral_table_1)	... (area_table_2_n, *)
...	...	...	... ...
n	(area_table_n_1, fringe_table_n_1)	(area_table_n_2, fringe_table_n_2)	... (*, lateral_table_n)

TableName: area_table_1_0
; p(s)= a*s + b
; the unit cap is a piece-wise linear function of s
; s1=<s <s2, s2=<s<= s3 …
s1, s2, s3, ...
(a1, b1), (a2, b2), (a3, b3) ….

TableName: area_table_1_2
; p(s)= a*s + b
; the unit cap is a piece-wise linear function of s

; s1=<s <s2, s2=<s< s3 …
s1, s2, s3, ...
(a1, b1), (a2, b2), (a3, b3) ….

TableName: area_table_1_3
; p(s)= a*s + b
;
s1, s2, s3, ...
(a1, b1), (a2, b2), (a3, b3) ….
….
;lateral cap table
TableName: lateral_table _1
;p(d) = a*d + b
d1, d2, d3, …
(a1, b1), (a2, b2), (a3, b3)…
…
;fringe cap table
TableName: fringe_table_1_0
;p(d) = a*d +b
d1, d2, d3, …
(a1, b1), (a2, b2), (a3, b3)…

The contest program should read in an config file, for example, input.conf, to parse all necessary files and parameters:
;Test1
design: <benchmark_file_name>
output: <output file>
rule_file: <design rule file>
process_file: <process file>
critical_net: net_id1, net_id2,...
power_nets: power_net1, power_net2, …
ground_nets: ground_net1, ground_net2, …
The program should output a design with metal-fills. The output file format should be same as input benchmark file.

Cap Extraction Example
example1.conf:

```
design: example1.layout
output: example1.fill
rule_file: rule.dat
process_file: process.dat
critical_net: 1
power_nets: 2
ground_nets: 0
```


Figure 9: Example layout with one metal-fill (dark green rectangle)

example1.layout:

```
0 0 100 80 ; chip boundary
1 60 0 100 10 2 1 Normal; A normal polygon on layer 1 (net 2: power)
2 0 40 100 50 1 1 Normal; A normal polygon on layer 1 (net 1: critical net)
3 0 40 10 80 1 2 Normal; A normal polygon on layer 2 (net 1: critical net)
4 60 0 70 80 2 2 Normal; A normal polygon on layer 2 (net 2: power)
```
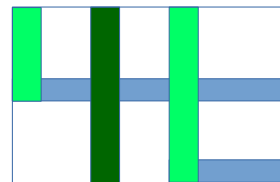
rule.dat:

```
;Layer_id <conductor|via> min_width min_space max_fill_width min_density max_density
1 conductor 10 10 30 0.3 1
2 conductor 10 10 30 0.3 1
```

process.dat:

```
;table matrix, layer id 0 means ground plane
         1                        2
0 (area_1_0, *)         (area_2_0, *)
1 (*, lateral_1)        (area_2_1, fringe_2_1)
2 (area_2_1, fringe_1_2) (*, lateral_2)
; process tables
TableName: area_1_0
  100                 200              300    400
  (0.01, 0.017) (0.0102, -0.02) (0.0101, 0.015)

TableName: lateral_1
  10                  50              100      200
  (0.01, 0.017) (0.0102, 0.001) (0.0101, 0.015)

TableName: fringe_1_2
  0                   50              100      150
  (0.007, 0.012) (0.0102, 0.001) (0.0101, 0.015)


TableName: area_2_0
  100                 150              200      300
  (0.01, 0.017) (0.0102, -0.01) (0.0101, 0.015)

TableName: area_2_1
  100                 300              400      500
  (0.01, 0.017) (0.0102, -0.02) (0.00101, 0.015)

TableName: lateral_2
  10                  50              100    200
  (0.01, 0.011) (0.0102, 0.001) (0.0101, 0.015)

TableName: fringe_2_1
  10                  50              100      150
  (0.011, 0.01) (0.0102, 0.001) (0.0101, 0.015
```

Now assuming the metal-fill output file example1.fill:

```
1 30 0 40 80 0 2 Fill; A metal-fill on layer 2
```

The coupling capacitance between the critical net (net id 1) and the metal-fill consists of area-capacitance between polygon 2 and the fill and the lateral-capacitance between polygon 3 and the fill. For the area capacitance, the overlapped area is 100 nm^2 between layer 1 and 2. So the area cap table should be "area_2_1". Because the area is between the index 100 and 300, then coefficients 0.01, 0.017 is selected to calculate the unit area capacitance: C_{unit} =0.01x overlapped area + 0.017 = 1.017 ff/nm^2, and the area-capacitance is C_{unit} x overlapped area = 101.7ff.

Please note if the overlapped area is larger than the largest area sampling point of the area cap table, then the largest sampling point is used to calculate an initial capacitance and then scaled by a factor of overlapped-area / largest area-sampling point. For example, assuming an overlapped area of 800 nm^2, formed by two polygons on layer 1 and 2, because the largest sampling point in table area_2_1 is 400nm^2, we use 400 to calculate an initial cap:

C_{init} = (0.0101 x 400 + 0.015)x400 = 1622 fF

The final area capacitance is calculated by

C_{area} = C_{init} x (800/400) = 1622 x 2 = 3244 fF

The similar calculation procedure can also apply to the overlapped area smaller than the smallest area sampling point:

C_{area} = C_{init} * $S_{overlap}$/$S_{min\_sampling\_area}$, where C_{init} is the initial capacitance calculated with the smallest area sampling point, $S_{overlap}$ is the overlapped area and $S_{min\_sampling\_area}$ is the smallest sampling point in the table.

For the lateral-capacitance between polygon 3 and the fill, the distance of the two polygons is 20nm, and the table is "lateral_2". Since 20nm is between 10nm and 50nm, we choose the first coefficient pair: 0.01, 0.011. Then the unit capacitance will be:

C_l = 0.01 * 20 + 0.011 = 0.211 ff/nm.

Because the parallel edges between the polygons are 40nm long, the lateral-capacitance will be

C_l x 40nm = 8.44 ff

Similar, we can calculate the coupling cap between the fill and net 2 (power net). Please note there is a fringe cap between polygon 1 and the fill. The distance between polygon 1 and the fill is 20nm. There are two tables we need consider: fringe_2_1 and fringe_1_2 (because different metal layer has different thickness, the two table may not be the same). We can get two unit capacitance:

C_{f12} = 0.007 X 20 + 0.012 = 0.152/nm

C_{f21} = 0.011 X 20 + 0.01 = 0.23/nm

The parallel edge length is 10nm, so the fringe capacitance between the file and polygon 1 will be

C_{f12} * 10nm + C_{f12}*10nm = 3.82 ff.

Please note for any lateral-capacitance or fringe-capacitance, if it is formed by two polygons with horizontal distance larger than the largest sampling point in the corresponding unit-cap table, the capacitance can be simply assumed 0 because the value is too small.

V.   Reference

[1] VS Shilimkar, SG Gaskill, A Weisshaar, "Impact of metal fill on on-chip interconnect performance", 2009 The 42nd International Symposium on Microelectronics, 983-990

[2] Vikas S Shilimkar, Andreas Weisshaar, "Modeling of Metal-Fill Parasitic Capacitance and Application to On-Chip Slow-Wave Structures," IEEE Transactions on Microwave Theory and Techniques 65 (5), 1456-1464

[3] Luis Charre, Bruno Gravano, Rémi Pôssas, Chen Zheng, "   Timing Aware Dummy Metal Fill Methodology", arXiv:1711.01407v2, https://arxiv.org/abs/1711.01407v2