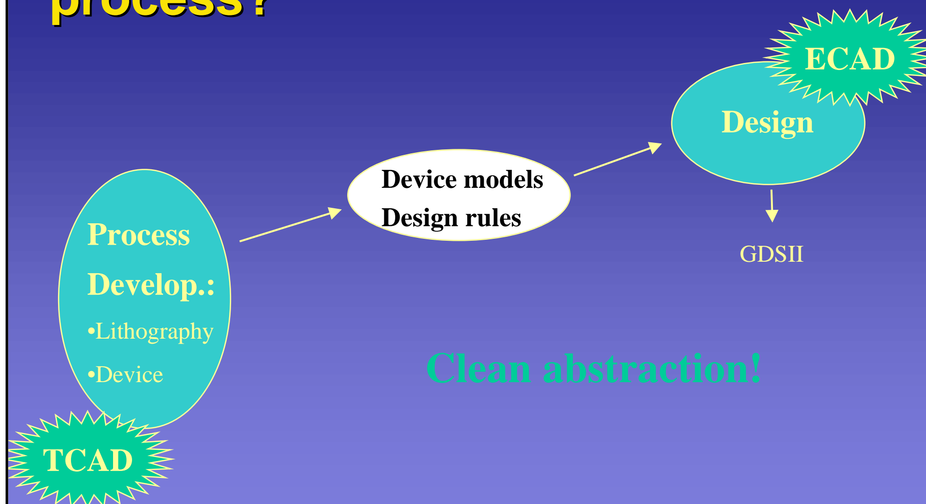
Modern Physical Design:

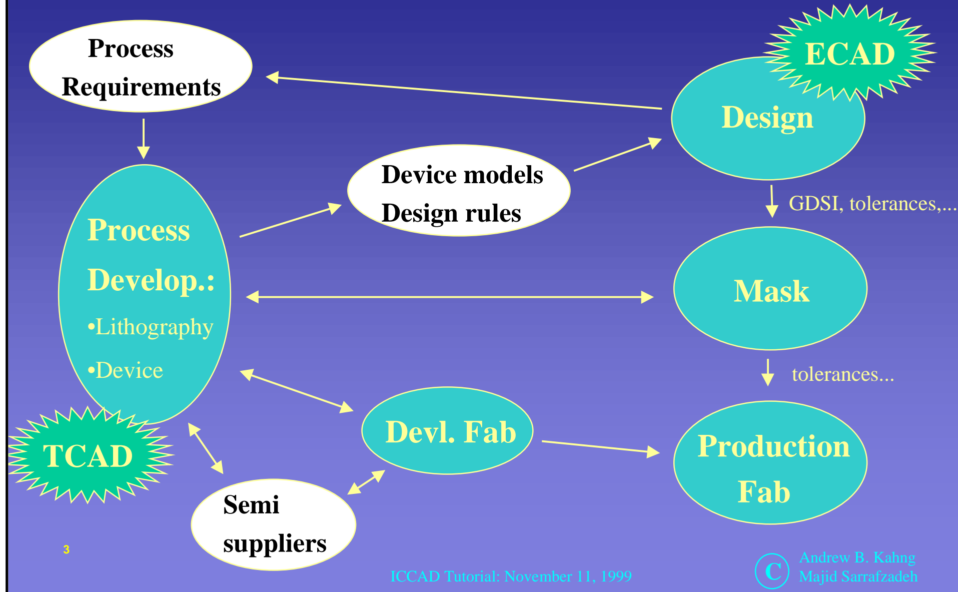
Algorithm
Technology
Methodology
(Part V)

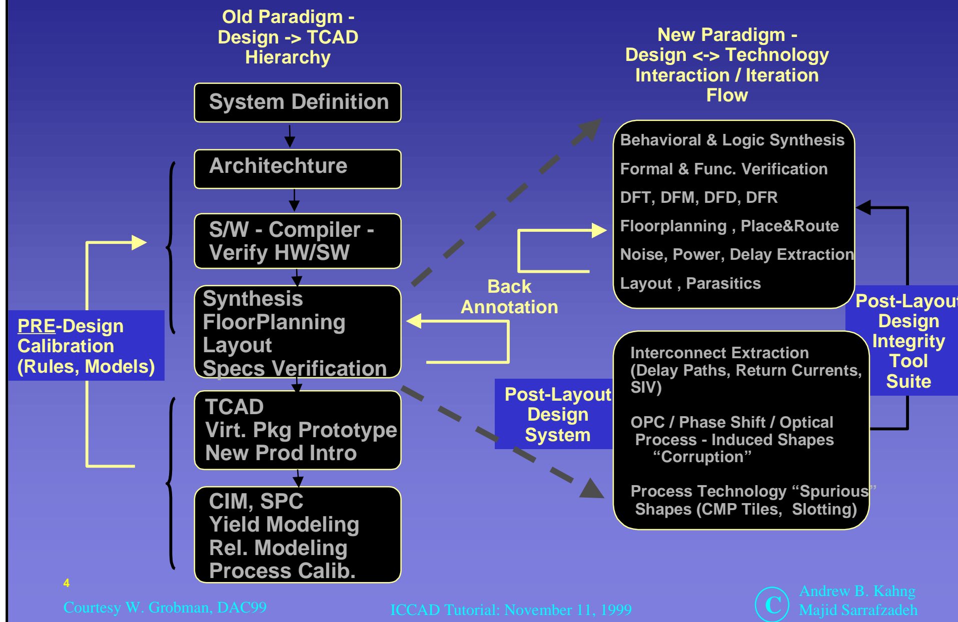
Andrew B. Kahng UCLA

Majid Sarrafzadeh Northwestern

C Andrew B. Kahng
Majid Sarrafzadeh




Developmental fab in tight loop
Process Requirements
ECAD
Design
Device models
Design rules
GDSI, tolerances,...
Process Develop.:
•Lithography
•Device
Mask
tolerances...
TCAD
Devl. Fab
Production Fab
Semi suppliers
3
ICCAD Tutorial: November 11, 1999
Andrew B. Kahng
Majid Sarrafzadeh

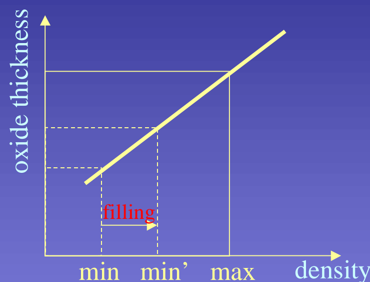

New ECAD-TCAD Paradigm
Old Paradigm - Design -> TCAD Hierarchy
New Paradigm - Design <-> Technology Interaction / Iteration Flow
System Definition
Architechture
S/W - Compiler - Verify HW/SW
Synthesis
FloorPlanning
Layout
Specs Verification
TCAD
Virt. Pkg Prototype
New Prod Intro
CIM, SPC
Yield Modeling
Rel. Modeling
Process Calib.
PRE-Design Calibration (Rules, Models)
Behavioral & Logic Synthesis
Formal & Func. Verification
DFT, DFM, DFD, DFR
Floorplanning , Place&Route
Noise, Power, Delay Extraction
Layout , Parasitics
Back Annotation
Post-Layout Design Integrity Tool Suite
Interconnect Extraction (Delay Paths, Return Currents, SIV)
OPC / Phase Shift / Optical Process - Induced Shapes "Corruption"
Process Technology "Spurious" Shapes (CMP Tiles, Slotting)
Post-Layout Design System
4
Courtesy W. Grobman, DAC99
ICCAD Tutorial: November 11, 1999
Andrew B. Kahng
Majid Sarrafzadeh

Density Control for CMP

- Chemical-mechanical polishing (CMP)
  - applied to interlayer dielectrics (ILD) and inlaid metals
  - polishing pad wear, slurry composition, pad elasticity make this a very difficult process step
- Cause of CMP variability
  - pad deforms over metal feature
  - greater ILD thickness over dense regions of layout
  - "dishing" in sparse regions of layout
  - huge part of chip variability budget used up (e.g., 4000Å ILD variation across-die)

Min-Variation Objective

- Relationship between oxide thickness and local feature density



- Minimizing variation in window density over layout preferable to satisfying lower and upper density bounds

Density Control for CMP

- Layout density control
  - density rules minimize yield impact
  - uniform density achieved by post-processing, insertion of dummy features
- Performance verification (PV) flow implications
  - accurate estimation of filling is needed in PD, PV tools (else broken performance analysis flow)
  - filling geometries affect capacitance extraction by > 50%
  - is a multilayer problem (coupling to critical nets, contacting restrictions, active layers, other interlayer dependencies)

© Andrew B. Kahng
Majid Sarrafzadeh

Density Rules

- Modern foundry rules specify layout density bounds to minimize impact of CMP on yield
- Density rules control local feature density for $w \times w$ windows
  - e.g., on each metal layer every 2000um × 2000um window must be between 35% and 70% filled
- Filling = insertion of "dummy" features to improve layout density
  - typically via layout post-processing in PV / TCAD tools
    - boolean operations on layout data
  - affects vital design characteristics (e.g., RC extraction)
  - accurate knowledge of filling is required during physical design and verification

© Andrew B. Kahng
Majid Sarrafzadeh

Need for Density Awareness in Layout

- Performance verification flow:

- Filling/slotting geometries affect RC extraction

VICTIM LAYER TOTAL CAPACITANCE (10^{-15} F)			
Same layer-i neighbors?	Fill layers i-1, i+1?	ε = 3.9	ε = 2.7
N	N	2.43 (1.0)	1.68 (1.0)
N	Y	3.73 (1.54)	2.58 (1.54)
Y	N	4.47 (1.84)	3.09 (1.84)
Y	Y	5.29 (2.18)	3.66 (2.18)

Up to 1% error in extracted capacitance
Reliability also affected (e.g. slotting of power stripes)

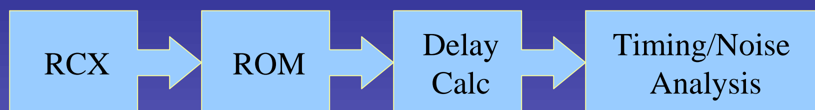
9

ICCAD Tutorial: November 11, 1999

C Andrew B. Kahng Majid Sarrafzadeh

Need for Density Awareness in Layout

- Performance verification flow:



- Can be considered as ``single-layer'' problem

Middle Victim Conductor Total Capacitance (10^{-15} F)			
Fill layer offset	Fill geometry	ε = 3.9	ε = 2.7
N	10 × 1	3.776 (1.0)	2.614 (1.0)
N	1 × 1	3.750 (0.99)	2.596 (0.99)
Y	10 × 1	3.777 (1.00)	2.615 (1.00)
Y	1 × 1	3.745 (0.99)	2.593 (0.99)

- Caveat: contacting, active+gate layers, other layer interactions

10

ICCAD Tutorial: November 11, 1999

C Andrew B. Kahng Majid Sarrafzadeh

Limitations of Current Techniques

- Current techniques for density control have three key weaknesses:
  - (1) only the average ***overall*** feature density is constrained, while local variation in feature density is ignored
  - (2) density analysis does not find ***true*** extremal window densities - instead, it finds extremal window densities only over fixed set of window positions
  - (3) fill insertion into layout does not minimize the maximum variation in window density

11

ICCAD Tutorial: November 11, 1999

© Andrew B. Kahng
Majid Sarrafzadeh

Layout Density Control Flow

Density Analysis
- find total feature area in each window
- find maximum/minimum total feature area over all $w \times w$ windows

- find slack (available area for filling) in each window

Fill synthesis
- compute amounts, locations of dummy fill
- generate fill geometries

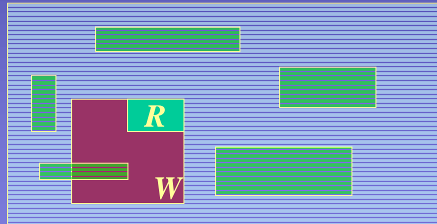
12

ICCAD Tutorial: November 11, 1999

© Andrew B. Kahng
Majid Sarrafzadeh

Exact Max-Density Window Analysis

- For each pivot rectangle ***R*** do
  - find density of ***w***×***w*** window ***W*** that abuts ***R*** on top and right
  - while ***W*** intersects ***R*** do
    - slide ***W*** right till intersection with other rectangle edge
    - record changes in density

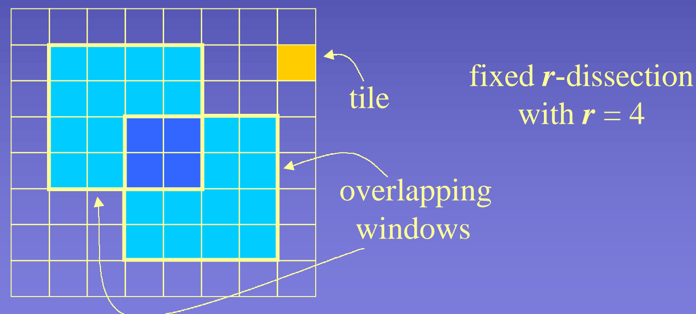


- $O(k^2)$ algorithm for ***k*** rectangles

C Andrew B. Kahng
Majid Sarrafzadeh

Fixed *r*-Dissection Regime

- Feature area density bounds enforced only for ***fixed*** set of ***w*** × ***w*** windows
- Layout partitioned by r^2 distinct fixed dissections
- Each ***w*** × ***w*** window is partitioned in r^2 ***tiles***



C Andrew B. Kahng
Majid Sarrafzadeh

Drawbacks of Fixed *r*-Dissection Analysis

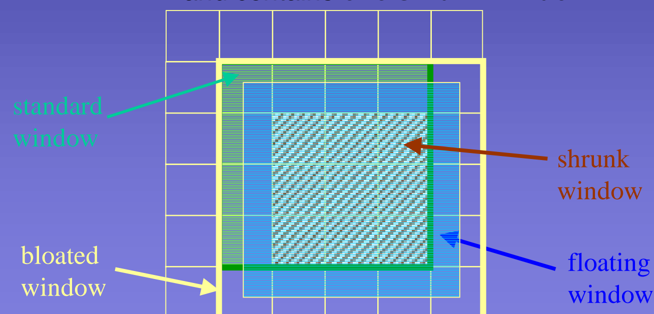
- If all $w \times w$ windows of fixed *r*-dissection have density $\leq U$, there may be ***floating*** $w \times w$ window with density $\min\{1, U + 1/r - 1/(4r^2)\}$
- Fixed-dissection algorithm is ***inaccurate***
- Exact algorithm is ***slow*** $= O(k^2)$

15

(C) Andrew B. Kahng
Majid Sarrafzadeh

Shrunk and Bloated Windows

- ***Standard window*** = fixed *r*-dissection $w \times w$ window
- ***Floating window*** = arbitrary $w \times w$ window
- ***Bloated window*** = standard window bloated by one tile
- ***Shrunk window*** = standard window shrunk by one tile
- Any ***floating window*** is contained in ***one bloated window*** and contains ***one shrunk window***



16

(C) Andrew B. Kahng
Majid Sarrafzadeh

Multilevel Approach

- ***Estimation***:
  - max floating window density ≤ max bloated window density
  - min floating window density ≥ min shrunk window density
- ***Zooming***:
  - remove standard windows in underfilled bloated windows
  - subdivide remaining tiles and find area of new bloated windows
- Terminate subdivision when either:
  - # of rectangles is small (run exact density analysis), or
  - (max bloated density)/(max standard density) ≤ ε ***(say, ε=1%)***

© Andrew B. Kahng
Majid Sarrafzadeh

Multilevel Algorithm

Tiles = list of all windows (r =1)
Accuracy = ∞
While ***Accuracy*** > 1+ ε
    find are in each bloated and standard window
        MAX = max area of standard window
        BMAX = max area of bloated window
    refine ***Tiles*** = list of tiles from bloated windows of area ≥ MAX
    subdivide each tile in ***Tiles*** into 4 subtiles
    Accuracy = BMAX / MAX
Output max standard window density = MAX/ w^2

© Andrew B. Kahng
Majid Sarrafzadeh

Runtime of Multilevel Algorithm

- Each iteration decreases difference in area between bloated and standard window by half
- Original difference is $3w^2$
- Main loop terminates after t iterations:

$$3w^2/2^t \leq 2\varepsilon$$

- Maximum t is $O(\log(w/\varepsilon))$
- Runtime is $O((n/w * \log(w/\varepsilon))^2)$

C Andrew B. Kahng
Majid Sarrafzadeh

Filling Problem

- **Given** design rule-correct layout
  of k disjoint rectilinear features in $n \times n$ region
- **Find** design rule-correct **filled** layout
  - no fill geometry is added within distance B of any layout feature
  - no fill is added into any window that has density $\geq U$
  - minimum window density in the filled layout is maximized (or has density $\geq$ lower bound L)

C Andrew B. Kahng
Majid Sarrafzadeh

Filling Problem in Fixed-Dissection Regime

- **Given**
  - fixed ***r***-dissection of layout
  - feature ***area***[T] in each tile T
  - ***slack***[T] = area available for filling in T
  - maximum window density ***U***
- **Find** total fill area p[T] to add in each T s.t.

  any $w \times w$ window W has density $\leq U$ and

  $\min_W \sum_{T \in W}$ (area[T] + p[T]) is maximized

© Andrew B. Kahng Majid Sarrafzadeh

Fixed-Dissection LP Formulation

- Maximize M (= lower bound on window density)
- Subject to:
  - For any tile T: $0 \leq p[T] \leq$ ***pattern*** $\times$ slack[T]
  - For any window W:

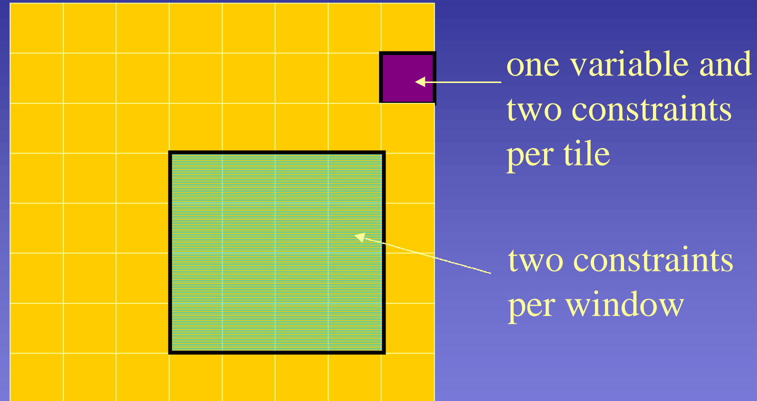
    $\sum_{T \in W} p[T] + area[T] \leq U \times w^2$

    $M \leq \sum_{T \in W} (p[T] + area[T])$

  (***pattern*** = max achievable pattern area density)

© Andrew B. Kahng Majid Sarrafzadeh

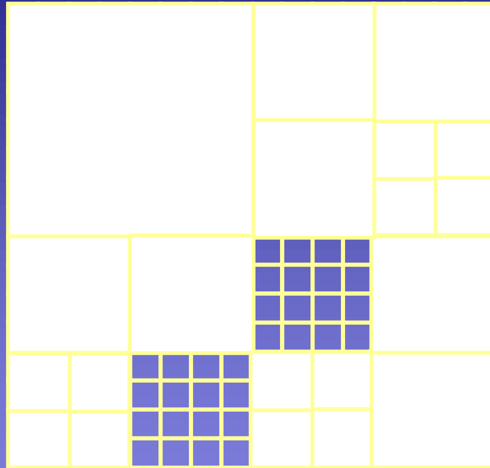
Fixed-Dissection LP Formulation



Multilevel LP Formulation

- Use multilevel density analysis in LP
- ***Tiles[r]*** = list of fixed r-dissection tiles from bloated windows of area ≥ MAX
- ***Saved tiles*** = subdivided ***Tiles[r] minus Tiles[r+1]***
- ***Saved windows*** = all standard windows W for which area is found
- Multilevel LP uses only constraints for ***saved tiles*** and ***saved windows***

Multilevel LP Formulation



Saved tiles have different sizes: tiles with more feature area are more subdivided

ML LP has one variable and two constraints per tile and two constraints per window

© Andrew B. Kahng
Majid Sarrafzadeh

Floating Deviation LP Formulation

- ***Floating deviation*** = the difference between max and min floating window density
- Floating deviation ≤
  max bloated window density - min shrunk window density
- ***Floating deviation LP:***
  - for any ***bloated*** window W:

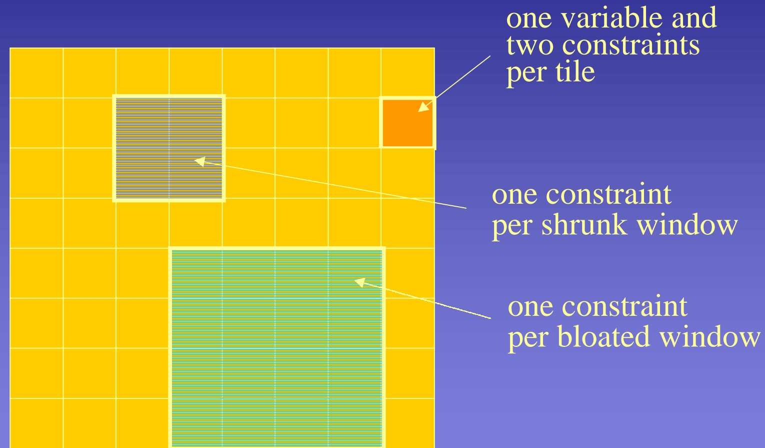
    $$\Sigma_{T \in W} (p[T] + area[T]) \leq U \times w^2$$

  - for any ***shrunk*** window W:

    $$M \leq \Sigma_{T \in W} (p[T] + area[T])$$

© Andrew B. Kahng
Majid Sarrafzadeh

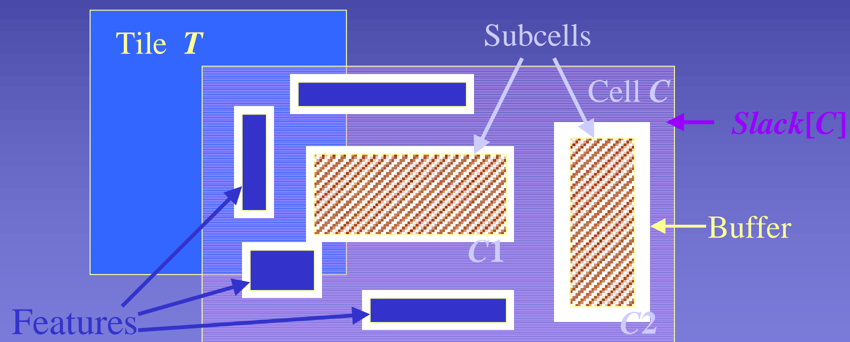
Floating Deviation LP Formulation



C Andrew B. Kahng
Majid Sarrafzadeh

Hierarchical Density Control

- Hierarchical filling = master cell filling



C Andrew B. Kahng
Majid Sarrafzadeh

Hierarchical LP Formulation

- For any cell instance *C* of master cell **C** and tile T, $\gamma[C,T]$ is portion of slack[**C**] in intersection of *C* with T:

  $\gamma[C,T] = \text{slack}(C \cap T)/\text{slack}[\mathbf{C}]$
- New variable d[**C**] per each master cell **C:**

  d[**C**] = ***filling per master cell*** **C**
- New constraints:
  - for total amount of filling added into tile T:

    $p[T] = \sum_{C \cap T} d[\mathbf{C}] \cdot \gamma[C,T]$
  - for amount of filling added into each master cell C:

    $0 \leq d[\mathbf{C}] \leq$ ***pattern*** $\times$ slack[**C**]

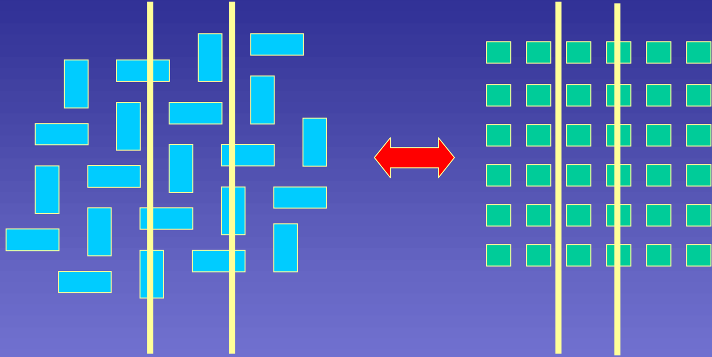
© Andrew B. Kahng
Majid Sarrafzadeh

Synthesis of Filling Patterns

- Given area of filling pattern p[i,j], insert filling pattern into tile T[i,j] ***uniformly*** over available area
- Desirable properties of filling pattern
  - uniform coupling to long conductors
  - either grounded or floating

© Andrew B. Kahng
Majid Sarrafzadeh

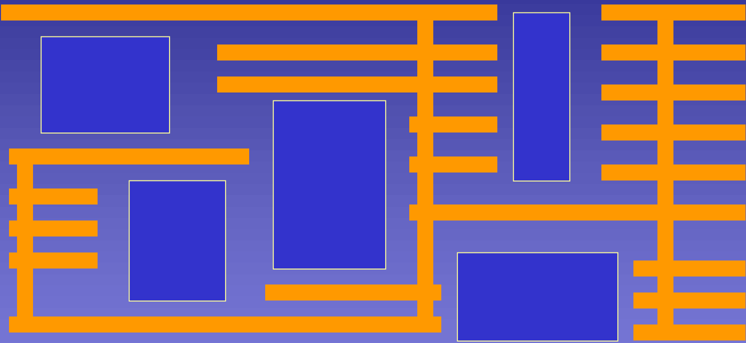
Basket-Weave Pattern



Each vertical/horizontal crossover line has same overlap capacitance to fill

© Andrew B. Kahng
Majid Sarrafzadeh

Grounded Pattern



Fill with horizontal stripes,
then span with vertical lines

© Andrew B. Kahng
Majid Sarrafzadeh

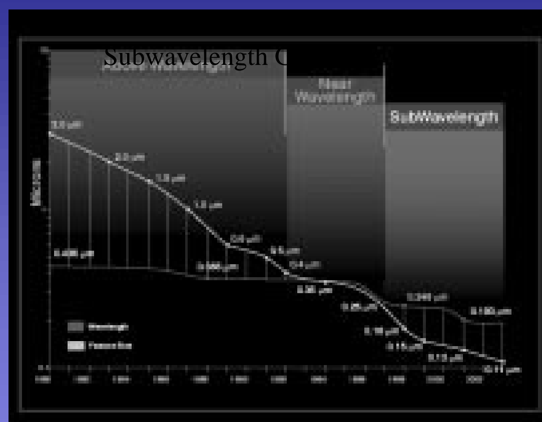
Subwavelength Optical Lithography — Technology Limits

- Implications of Moore's Law for feature sizes
- Steppers not available; WYSIWYG (layout = mask = wafer) fails after .35µm generation
- Optical lithography
  - circuit patterns optically projected onto wafer
  - feature size limited by diffraction effects
  - Rayleigh limits
    - resolution *R* proportional to λ / *NA*
    - depth of focus *DOF* proportional to λ / NA^2
- Available knobs
  - amplitude (aperture): OPC
  - phase: PSM

© Andrew B. Kahng
Majid Sarrafzadeh

Next-Generation Lithography and the Subwavelength Gap

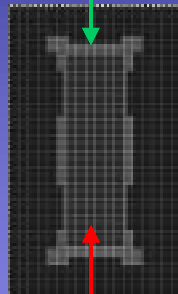
- EUV
- X-rays
- E-beams
- All at least 10 years away; require significant R&D, major infrastructure changes
- > 30 years of infrastructure and experience supporting optical lithography

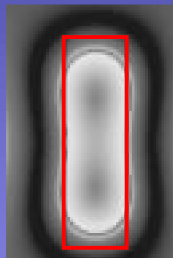


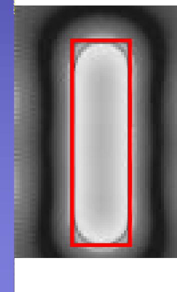
© Andrew B. Kahng
Majid Sarrafzadeh

Optical Proximity Correction (OPC)

- Corrective modifications to improve process control
  - improve yield (process latitude)
  - improve device performance







© Andrew B. Kahng Majid Sarrafzadeh

Optical Proximity Correction (OPC)

- Mostly cosmetic corrections; complicates mask manufacturing and dramatically increases cost (with little benefit?)
- Post-design verification is essential

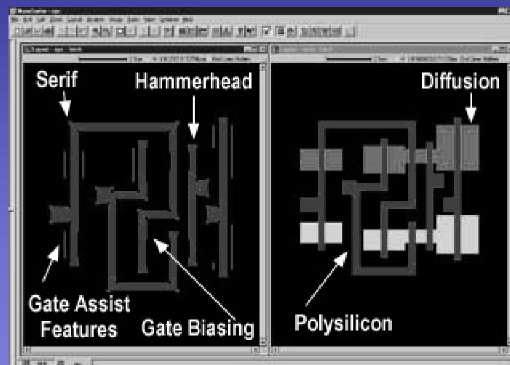
Rule-based OPC
- apply corrections based on a set of predetermined rules
- fast design time, lower mask complexity
- suitable for less aggressive designs

Model-based OPC
- use process simulation to determine corrections on-line
- longer design time, increased mask complexity
- suitable for aggressive designs

© Andrew B. Kahng Majid Sarrafzadeh

OPC Features

- Serifs - for corner rounding
- Hammerheads - for line-end shortening
- Gate assists (subresolution scattering bars) - for CD control
- Gate biasing - for CD control
- Issues for custom, hierarchical and reuse-based layout methodologies



© Andrew B. Kahng Majid Sarrafzadeh

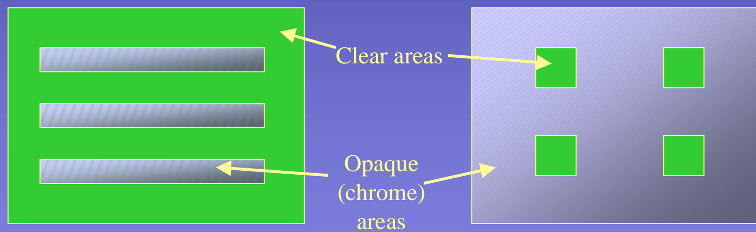
OPC Issues

- WYSIWYG broken → (mask) verification bottleneck
- Pass functional intent down to OPC insertion
  - make corrections that win $$$, reduce performance variation
  - OPC insertion is for predictable circuit performance, function
- Pass limits of manufacturing up to layout
  - don't make corrections that can't be manufactured or verified
  - Mask Error Enhancement Factor, etc.
- Layout needs models of OPC insertion process
  - geometry effects on cost of required OPC to yield function
  - costs of breaking hierarchy (beyond known verification, characterization costs)

© Andrew B. Kahng Majid Sarrafzadeh

Mask Types

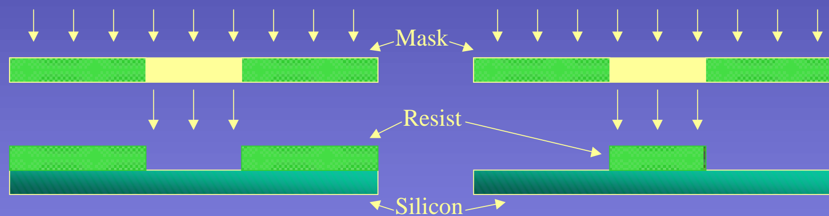
- Bright field masks
  - opaque features defined by chrome
  - background is transparent
  - used, e.g., for poly and metal
- Dark field masks
  - transparent features defined
  - background is opaque (chrome)
  - used, e.g., for contacts
  - used also for damascene metals



C Andrew B. Kahng
Majid Sarrafzadeh

Photoresist Types

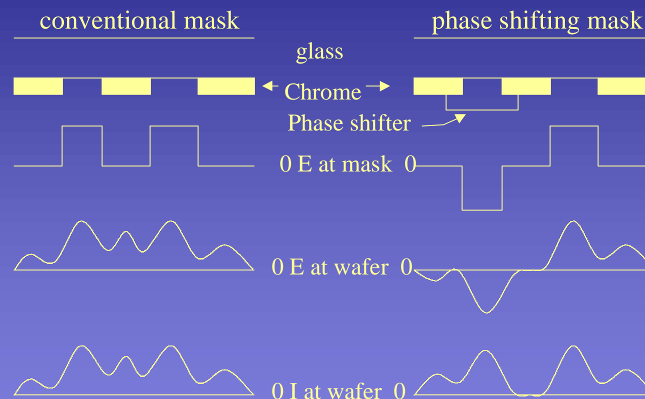
- Positive resists
  - material is removed from *exposed* areas during development
  - most widely used
- Negative resists
  - material is removed from *unexposed* areas during development
  - less mature



Post development profile for positive and negative photoresists

C Andrew B. Kahng
Majid Sarrafzadeh

Phase Shifting Masks



41

ICCAD Tutorial: November 11, 1999

© Andrew B. Kahng
Majid Sarrafzadeh

Phase Shifting Masks

- no phase shifting: poor contrast due to diffraction
- phase shifting by 180°: reverse electric field on mask, destructive interference yields zero-intensity on wafer (high contrast)
- Background
  - invented in 1982 by Levenson at IBM
  - interest in early 1990s, but near wavelength → no pressing need
- Many forms of phase-shifting proposed
- Key issues: manufacturability, design tools
- Today: subwavelength gap forces PSM into every process (example: Motorola 90nm gates using 248nm stepper, announced in early 1999)

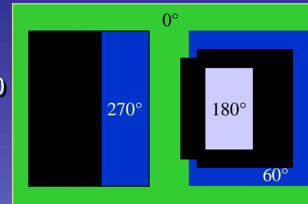
42

ICCAD Tutorial: November 11, 1999

© Andrew B. Kahng
Majid Sarrafzadeh

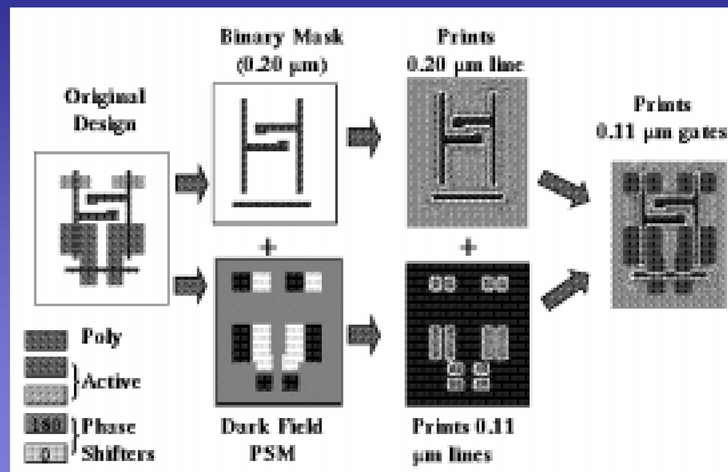
Forms of PSM

- Bright Field Phase-Shifting
  - single exposure
    - phase transitions required, e.g., 0-60-120-180 or 90-0-270 to avoid printing phase edges
    - throughput unaffected
    - limited improvement in process latitude
    - mask manufacturing difficult, mask cost very high
  - double exposure
    - PSM with 0 and 180 degree phase shifters
    - define only critical features ("locally bright-field"), rest of mask is chrome
    - second exposure with clear-field binary mask protects critical features, defines non-critical features as well
    - excellent process latitude
    - decrease in throughput (double exposure)



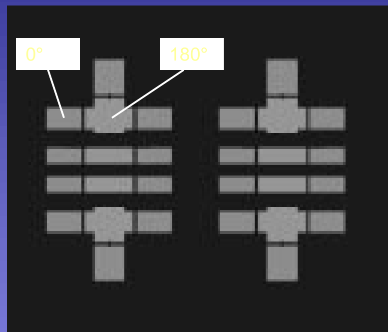
C Andrew B. Kahng
Majid Sarrafzadeh

Gate Shrinking and CD Control Using Phase Shifting



C Andrew B. Kahng
Majid Sarrafzadeh

Double-Exposure Alternating PSM

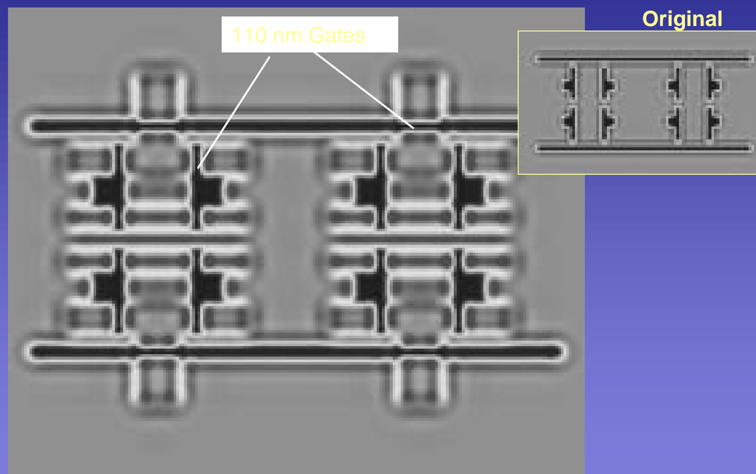


1. Alternate PSM Mask

2. Trim Mask (COG)

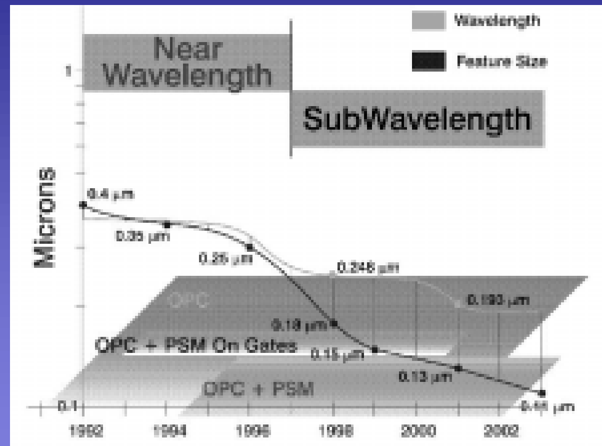
C Andrew B. Kahng
Majid Sarrafzadeh

Benefits of PSM



C Andrew B. Kahng
Majid Sarrafzadeh

Applicability of OPC and PSM

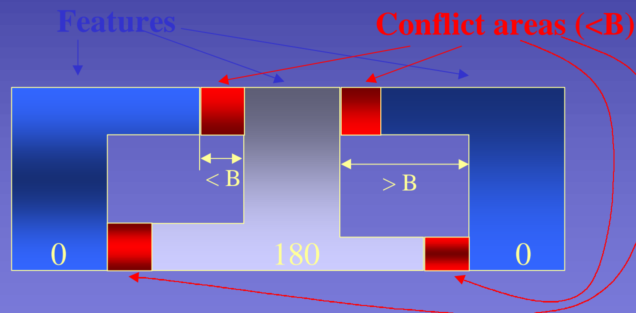


© Andrew B. Kahng
Majid Sarrafzadeh

The Phase Assignment Problem in PSM

Assign 0, 180 phase regions such that

- (dark field) feature pairs with separation < B have opposite phases
- (bright field) features with width < B are induced by adjacent phase regions with opposite phases



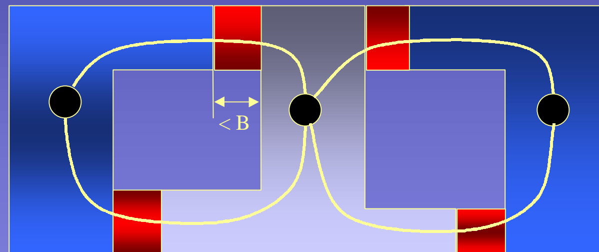
b ≡ minimum separation or width, with phase shifting

B ≡ minimum separation or width, without phase shifting

© Andrew B. Kahng
Majid Sarrafzadeh

Phase Conflict and the Conflict Graph

- Vertices: features (or phase regions)
- Edges: “conflicts” (necessary phase contrasts) (feature pairs with separation < B)



49

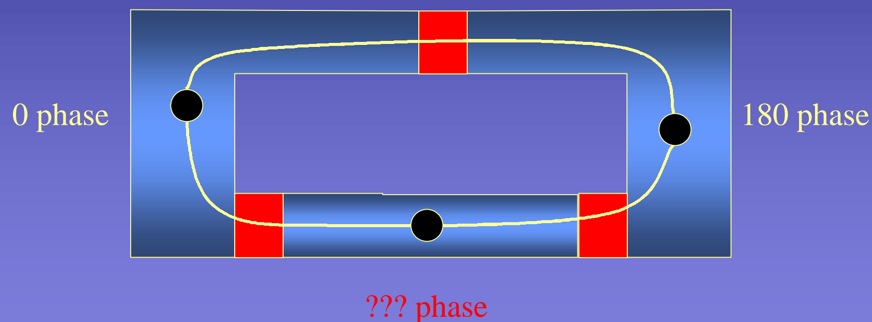
ICCAD Tutorial: November 11, 1999

C Andrew B. Kahng Majid Sarrafzadeh

Odd Cycles in Conflict Graph

- Self-consistent phase assignment is not possible if there is an odd cycle in the conflict graph
- Phase-assignable ≡ bipartite ≡ no odd cycles

0 phase



180 phase

??? phase

50

ICCAD Tutorial: November 11, 1999

C Andrew B. Kahng Majid Sarrafzadeh

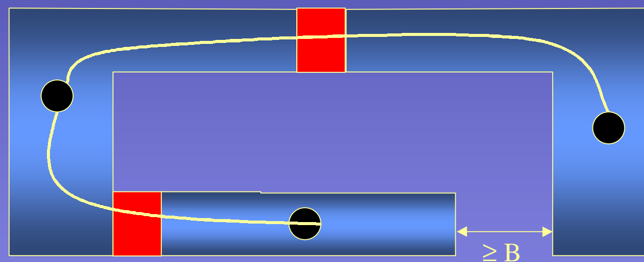
Phase Conflict and the Conflict Graph

- Self-consistent phase assignment is not possible if there is an odd cycle in the conflict graph
- Phase-assignable = bipartite = no odd cycles
  - this is a global issue!
  - features on one side of chip can affect features on the other side
- Breaking odd cycles: must change the layout!
  - change feature dimensions, and/or change spacings
  - degrees of freedom include layer reassignment for interconnects

C Andrew B. Kahng
Majid Sarrafzadeh

Breaking Odd Cycles

- Must change the layout:
  - change feature dimensions, and/or
  - change spacings
  - <u>PSM phase-assignability is a layout, not verification, issue</u>



C Andrew B. Kahng
Majid Sarrafzadeh

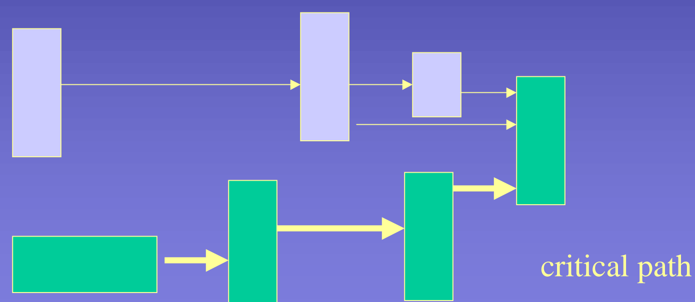
Phase Assignment: Key Technologies

- Key technologies: incremental gridless routing, incremental compaction
- Issues for custom, hierarchical and reuse-based layout methodologies

© Andrew B. Kahng
Majid Sarrafzadeh

Conflict Edge Weight

- Which conflict edges are cheapest to break?
- Critical paths (e.g., in compactor) in x- and y-directions define layout area
- Conflict edges not on critical path: break for free



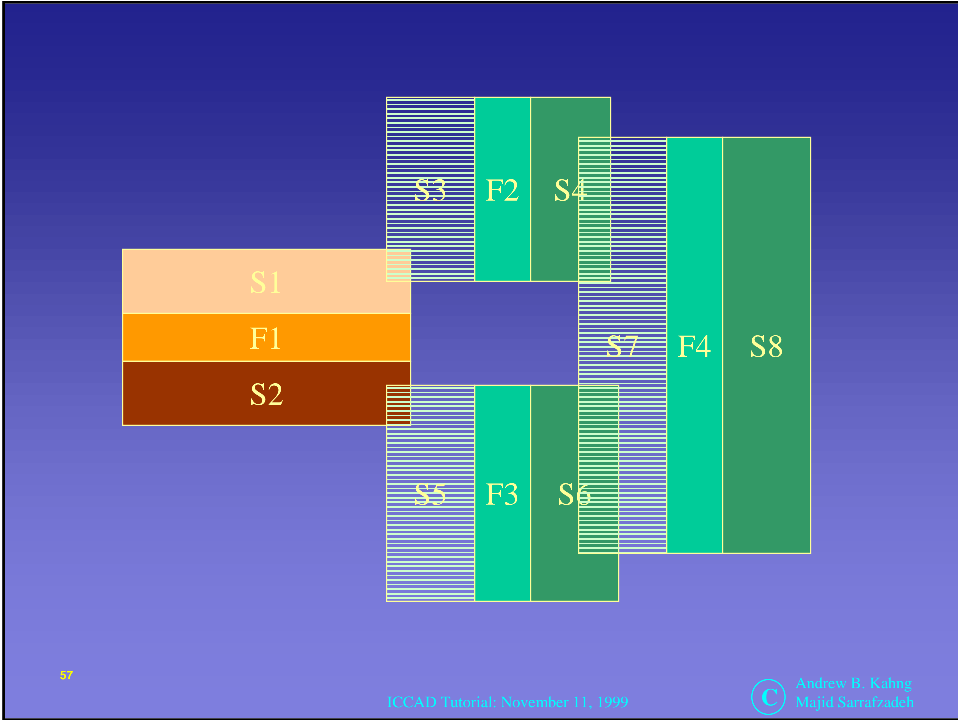
© Andrew B. Kahng
Majid Sarrafzadeh

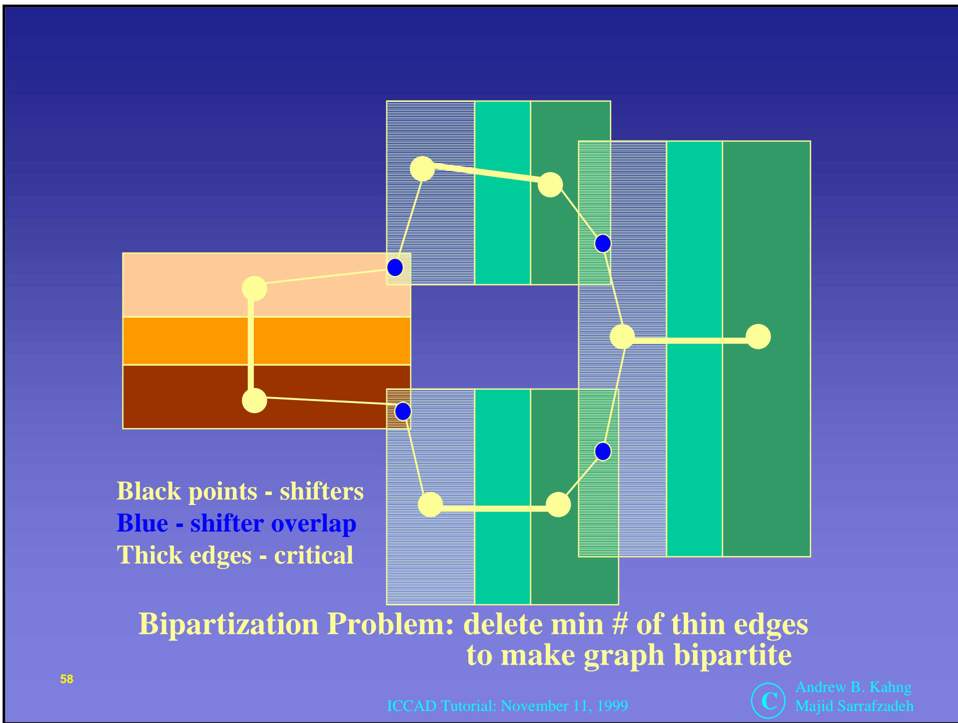
Minimum-Perturbation PSM Layout

- Input: layout in, e.g., .25um design rules
- Goal: adjust feature sizes and spacings to new PSM design rules, e.g., .15um
  - keep topology as close to original as possible
- Application to new design and to migration
  - assumes existence of a starting layout
  - hope to attain fewer violations in verification, require less manual cleanup of output layout

Compaction-Oriented Approach

- Analyze input layout
- Determine constraints for output layout
  - new PSM-induced (shape, spacing) constraints
- Compact (e.g., solve LP) with min perturbation objective
  - e.g., minimize sum of differences between old and new positions of each edge
- Key: Minimize the set of new constraints, i.e., break all odd cycles in conflict graph by deleting a minimum number of edges.


S3
F2
S4
S1
F1
S2
S7
F4
S8
S5
F3
S6
57
ICCAD Tutorial: November 11, 1999
C
Andrew B. Kahng
Majid Sarrafzadeh


Black points - shifters
Blue - shifter overlap
Thick edges - critical
Bipartization Problem: delete min # of thin edges
to make graph bipartite
58
ICCAD Tutorial: November 11, 1999
C
Andrew B. Kahng
Majid Sarrafzadeh

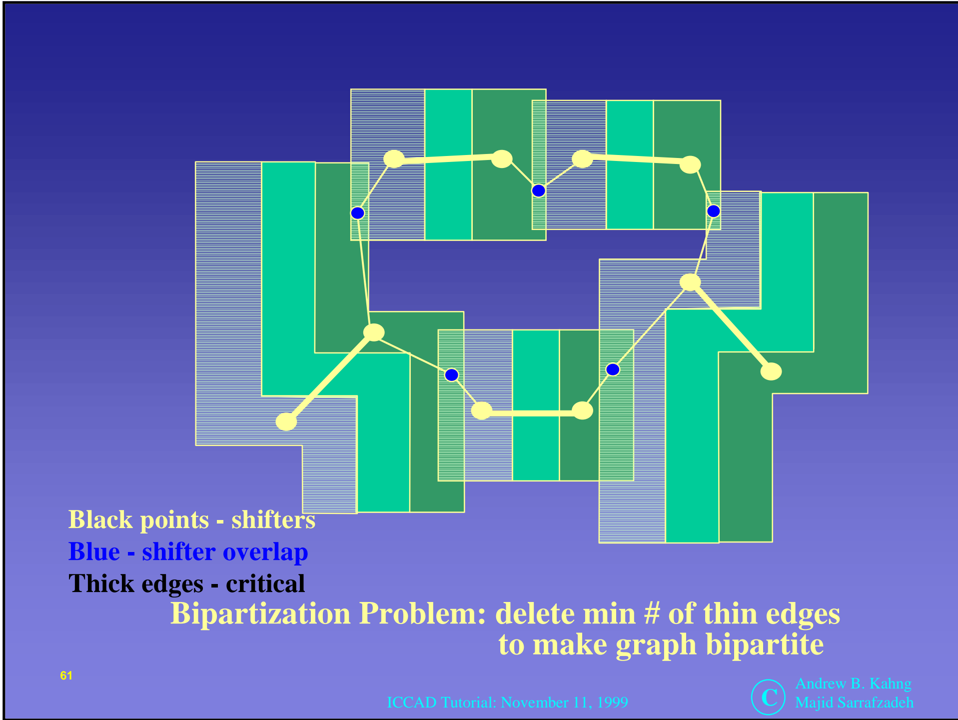
Black points - features
Blue - shifter overlap
Pink - extra nodes to distinguish opposite shifters

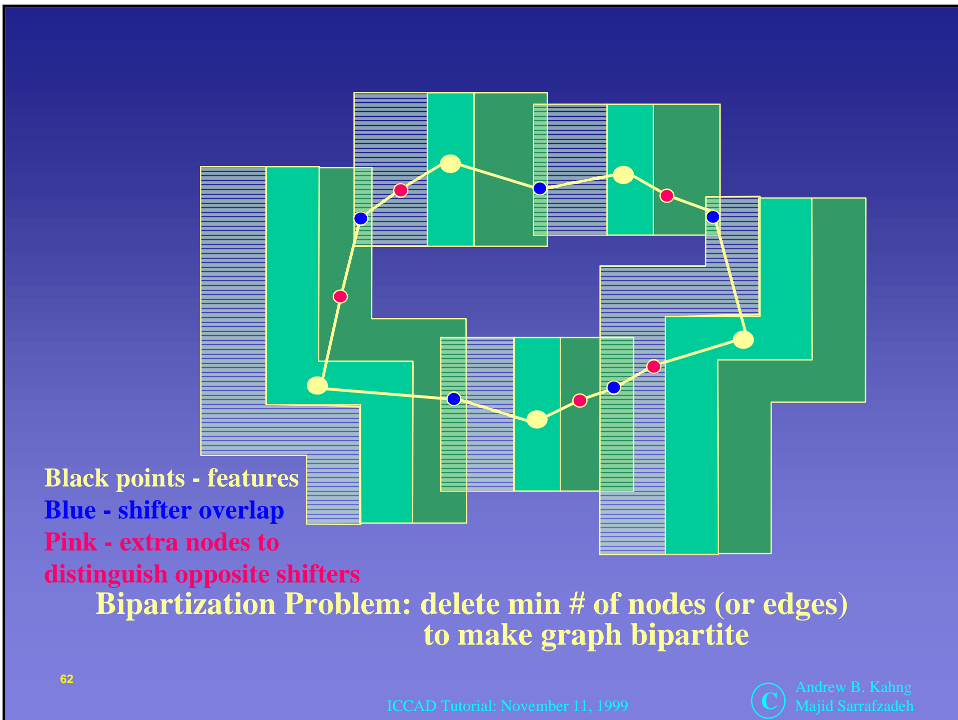
Bipartization Problem: delete min # of nodes (or edges) to make graph bipartite

C Andrew B. Kahng
Majid Sarrafzadeh

ICCAD Tutorial: November 11, 1999

C Andrew B. Kahng
Majid Sarrafzadeh


Black points - shifters
Blue - shifter overlap
Thick edges - critical
Bipartization Problem: delete min # of thin edges
to make graph bipartite
61
ICCAD Tutorial: November 11, 1999
C
Andrew B. Kahng
Majid Sarrafzadeh


Black points - features
Blue - shifter overlap
Pink - extra nodes to
distinguish opposite shifters
Bipartization Problem: delete min # of nodes (or edges)
to make graph bipartite
62
ICCAD Tutorial: November 11, 1999
C
Andrew B. Kahng
Majid Sarrafzadeh

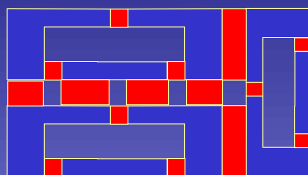
The T-join Problem

- How to delete **minimum-cost** set of edges from conflict graph G to eliminate odd cycles?
- Construct geometric dual graph D=dual(G)
- Find odd-degree vertices T in D
- Solve the **T-join problem** in D:
  - find min-weight edge set J in D such that
    - all T-vertices has odd degree
    - all other vertices have even degree
- Solution J corresponds to desired min-cost edge set in conflict graph G

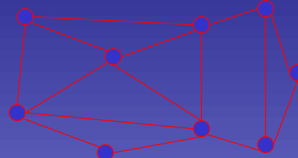
C Andrew B. Kahng
Majid Sarrafzadeh

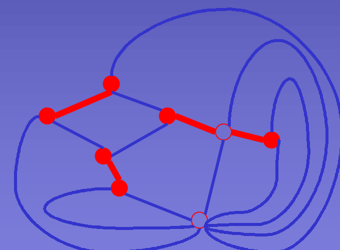
Optimal Odd Cycle Elimination

blue features/red conflicts

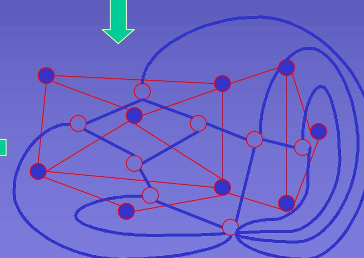


conflict graph G





T-join odd degree nodes in D



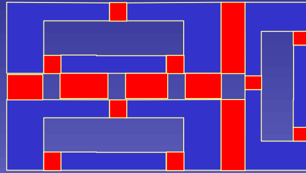
dual graph D

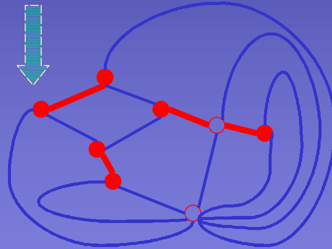
ICCAD Tutorial: November 11, 1999

C Andrew B. Kahng
Majid Sarrafzadeh

Optimal Odd Cycle Elimination

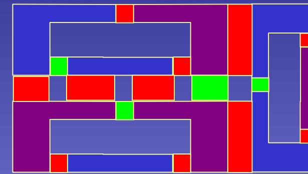
blue features/red conflicts

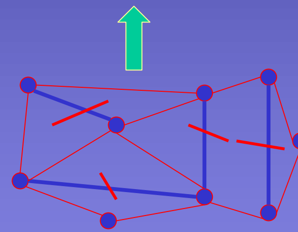




T-join odd degree nodes in D

Assign phases:
only green conflicts left





conflict graph

65

© Andrew B. Kahng
Majid Sarrafzadeh

T-join Problem in Sparse Graphs

- Reduction to matching
  - construct a complete graph T(G)
    - vertices = T-vertices
    - edge costs = shortest-path cost
  - find minimum-cost perfect matching
- Typical example = sparse (not always planar) graph
  - note that conflict graphs are sparse
  - #vertices = 1,000,000
  - #edges ≈ 5 × #vertices
  - # T-vertices ≈ 10% of #vertices = 100,000
- Drawback: finding all shortest paths too slow and memory-consuming
  - #vertices = 100,000 → #edges = 5,000,000,000

66

© Andrew B. Kahng
Majid Sarrafzadeh

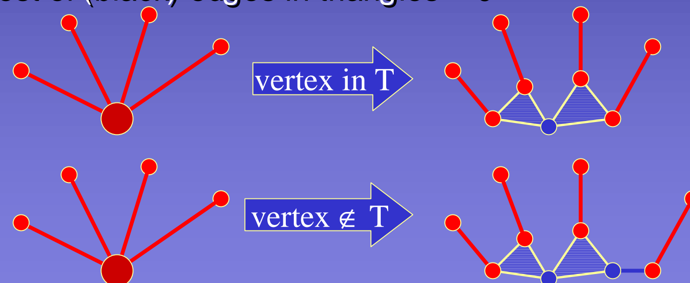
T-join Problem: Reduction to Matching

- **Desirable properties of reduction to matching:**
  - exact (i.e., optimal)
  - not much memory (say 2-3Xmore)
  - results in a very fast solution
- **Solution: gadgets!**
  - replace each edge/vertex with gadgets s.t.
    matching all vertices in gadgeted graph
    ⇔ T-join in original graph

© Andrew B. Kahng
Majid Sarrafzadeh

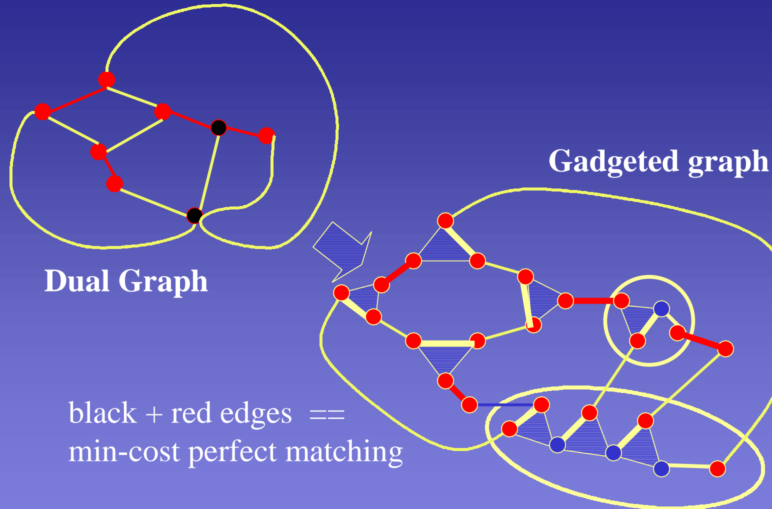
T-join Problem: Reduction to Matching

- replace each vertex with a chain of triangles
- one more edge for T-vertices
- in graph D: m = #edges, n = #vertices, t = #T
- in gadgeted graph: 4m-2n-t vertices, 7m-5n-t edges
- cost of red edges = original dual edge costs
  cost of (black) edges in triangles = 0



© Andrew B. Kahng
Majid Sarrafzadeh

Example of Gadgeted Graph



C Andrew B. Kahng
Majid Sarrafzadeh

Results

Testcase	Layout1		Layout2		Layout3	
	polygons	edges	polygons	edges	polygons	edges
	3769	12442	9775	26520	18249	51402
Algorithm	edges	runtime	edges	runtime	edges	runtime
Greedy	2650	0.56	2722	3.66	6180	5.38
GW	1612	3.33	1488	5.77	3280	14.47
Exact	1468	19.88	1346	16.67	2958	74.33

- Runtimes in CPU seconds on Sun Ultra-10
- Greedy = breadth-first-search bicoloring (similar to Ooi et al.)
- GW = Goemans/Williamson95 heuristic
- Cook/Rohe98 for perfect matching

Latest improved gadgets: runtimes decrease by factor of 6

C Andrew B. Kahng
Majid Sarrafzadeh

Manufacturing Process Issues

- Clean abstractions of process
  - OPC: a “rectangle” has 26 edges in 3 polygons
  - PSM: cell layouts and legal routes not freely composable

Manufacturing Process Issues

- Mask inspection / validation bottleneck
  - if Layout ≠ Silicon but the circuit is still functional… is it really an error ?
  - must the Design ↔ Mask boundary change ?

Manufacturing Process Issues

- Tool flow
  - line end extensions in the router
  - hammerheads in the library generator
  - OPC insertion in physical verification

  - RC extraction for timing signoff in P&R
  - post-P&R density control (filling and cheesing)

Manufacturing Process Issues

- “Life changes” for circuit and layout designers
  - only particular geometries and pitches ?
  - self-adapting circuits ?
  - huge verification guardbands ?
  - reuse-based design at risk -- at all levels ?

Manufacturing Process Issues

- Partial ownership of the problem = no ownership ?
  - when designs fail, who owns the scrap ?

Implications of Technology

- A hard look at hard IP reuse
  - divergent foundry processes
  - design- and context-specific variants of cells, macros, cores
    - filling densities
    - thermal, noise sensitivity contexts
    - layer usage and local region porosity constraints, physical access
    - incompatibility of separate phase solutions, or phase solutions + local routing
    - tool-specific variants (e.g., for different auto-routers)
    - diffusion sharing, continuous device sizing, tuning (dual Vt, multiple supply voltages (thermal, IR drop contexts), different input arrival times/slews,…)
- Hard-reuse: An ideal that must be tempered by hard realities

Many Worries

- Implications of UDSM technology trends
  - subwavelength lithography
    - WYSIWYG fails completely!
    - OPC, PSM, CMP-driven filling essential for printability, process windows
    - huge worries about broken hierarchy, reusability of layout
  - other yield-driven objectives
    - critical area, antennas
  - signal integrity
    - delay uncertainty, crosstalk noise, IR drop
  - reliability
    - electromigration, AC self-heat, hot electrons
- All of these are first-class objectives
- Can ASIC methodology handle all these effectively?