Subwavelength Optical Lithography: Challenges and Impact on Physical Design

Part II: Problem Formulations and Tool Integration

Andrew B. Kahng, UCLA CS Department

ISPD-99 TUTORIAL

April 13, 1999

Forcing Trends in EDA

- Silicon complexity and design complexity
  - many opportunities to leave major $$$ on the table
  - issues: physical effects of process, migratability
  - design rules more conservative, design waivers ↑
  - device-level layout opts in cell-based methodologies
- Verification cost increases dramatically
- Prevention a necessary complement to checking
- Successive approximation = design convergence
  - upstream activities pass intentions, assumptions downstream
  - downstream activities must be predictable
  - models of analysis/verification == objectives for synthesis

EDA Awareness of Process

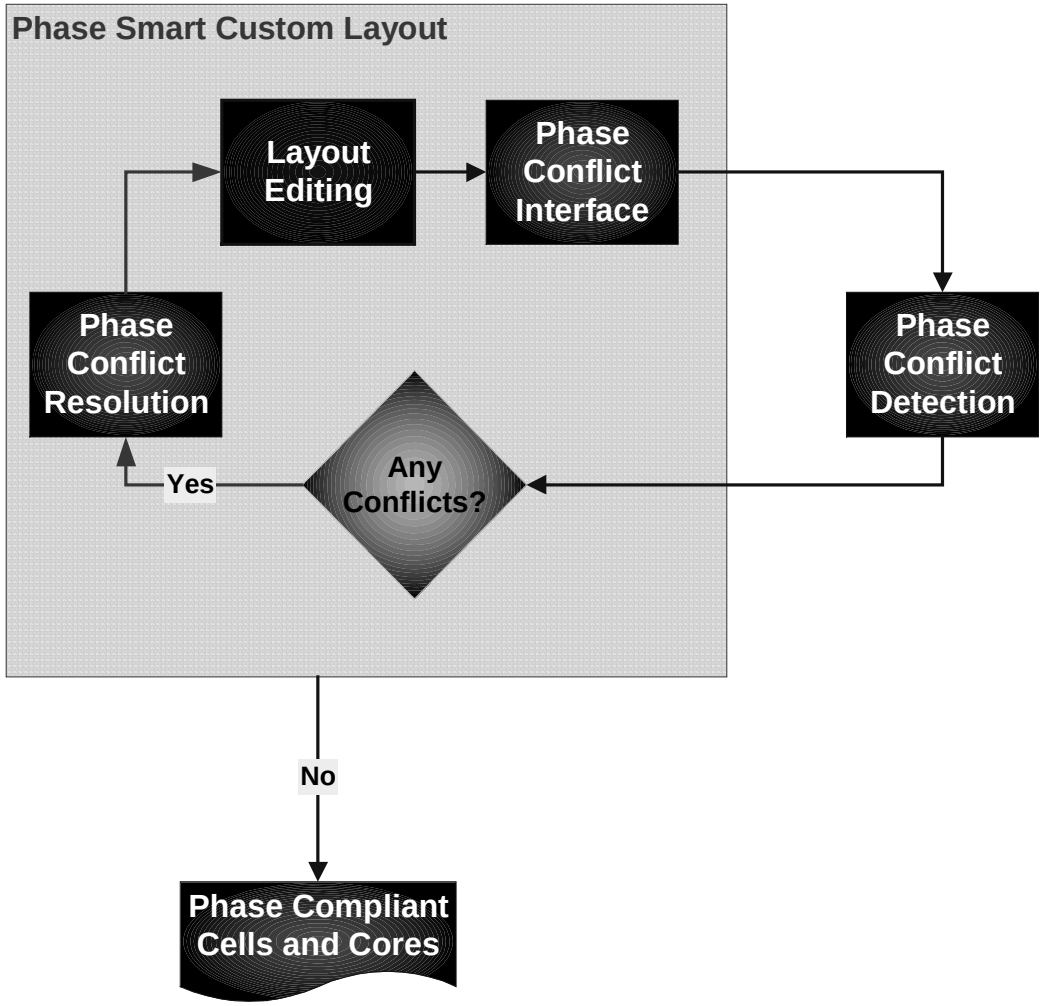
EDA wants to know as little as possible

This talk: The problems that can't be avoided

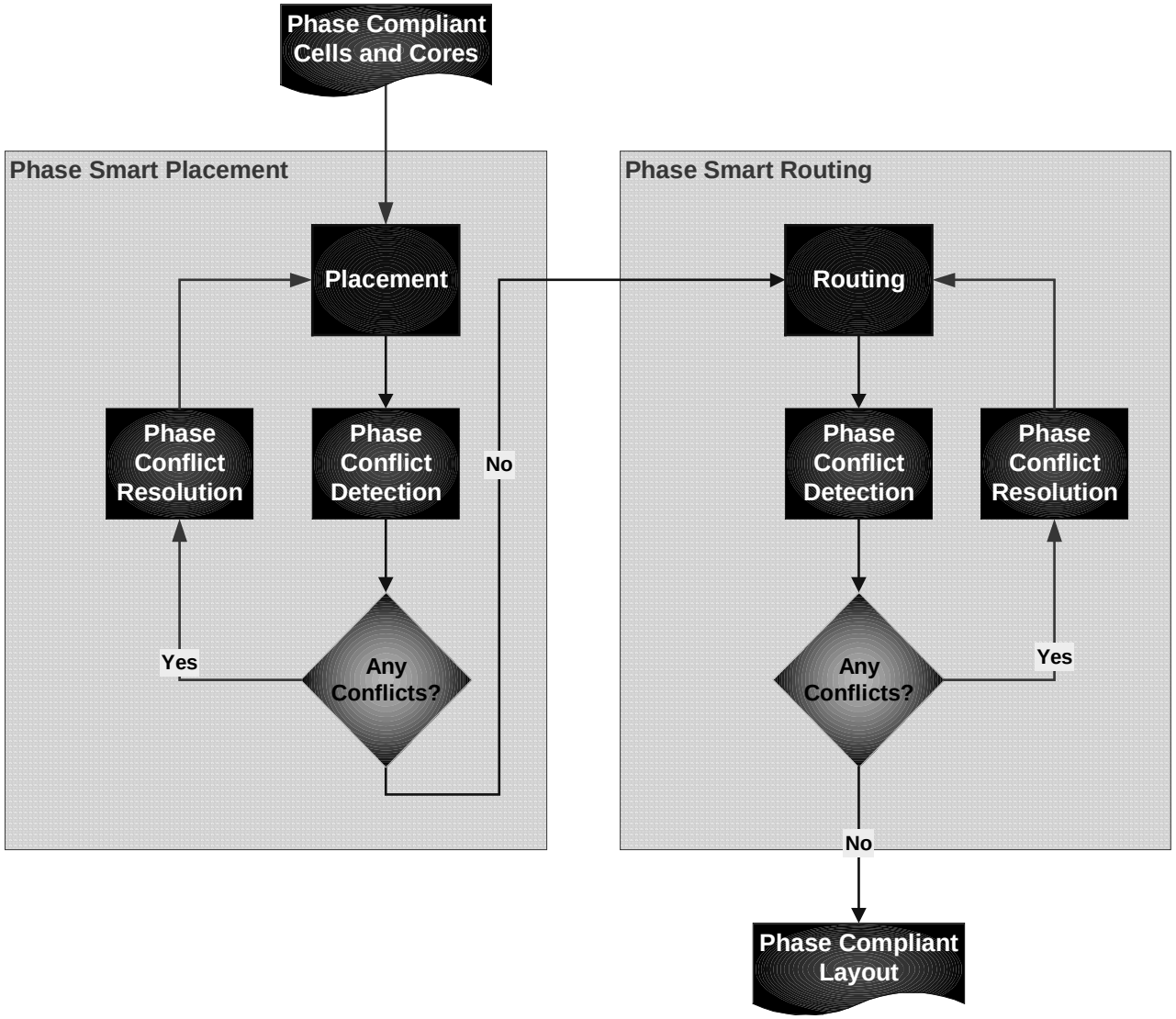
Necessary Formulations, Flows

- PD objectives want to capture downstream layout operations “transparently”
- New problem formulations
  - PSM: more global phenomena, scalability issues
  - OPC: mostly local phenomena
  - function-driven corrections
  - hierarchical and reuse-centric regimes
- New tool integrations

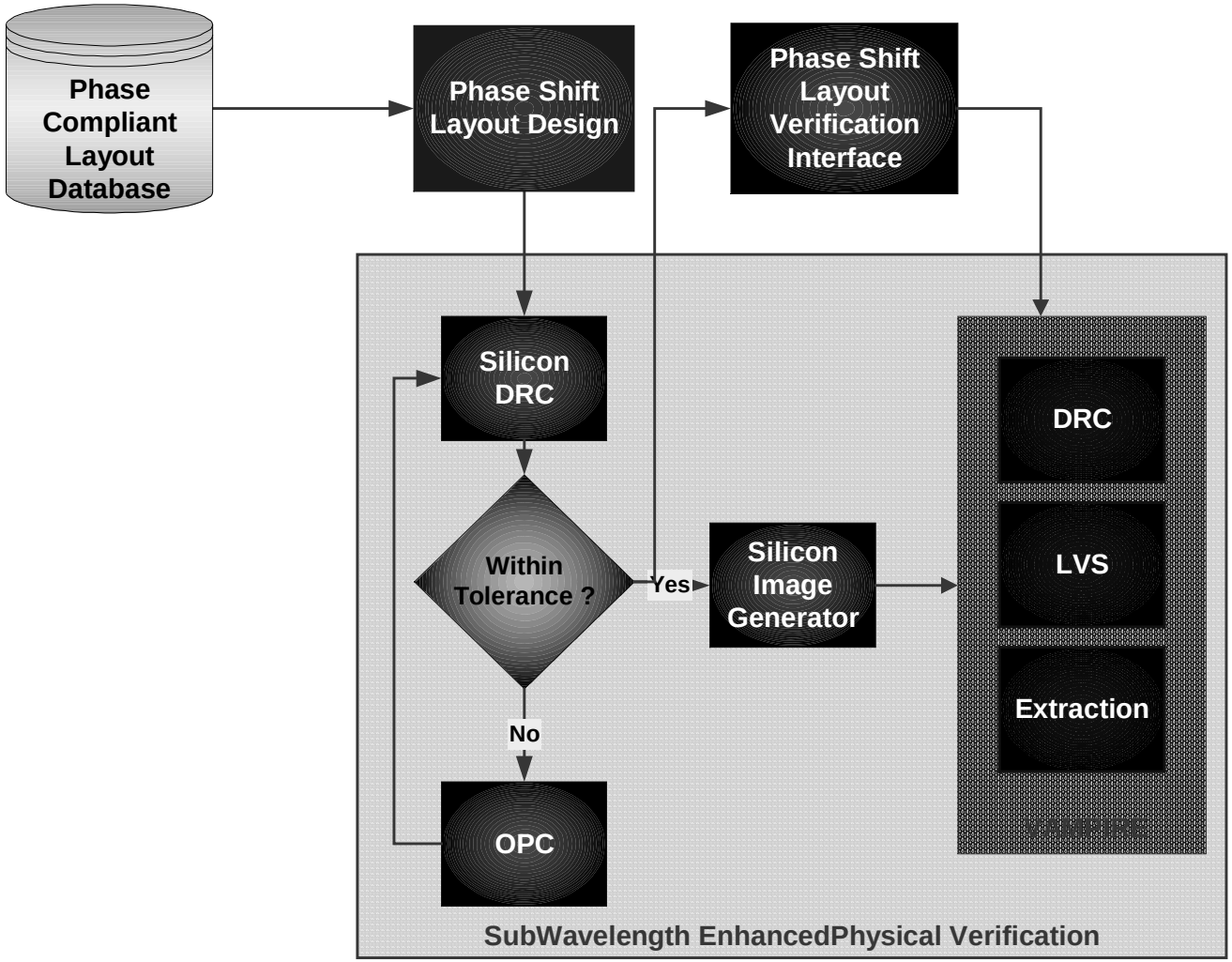
Phase Smart Custom Layout



Phase Smart Place and Route



Phase Smart Verification

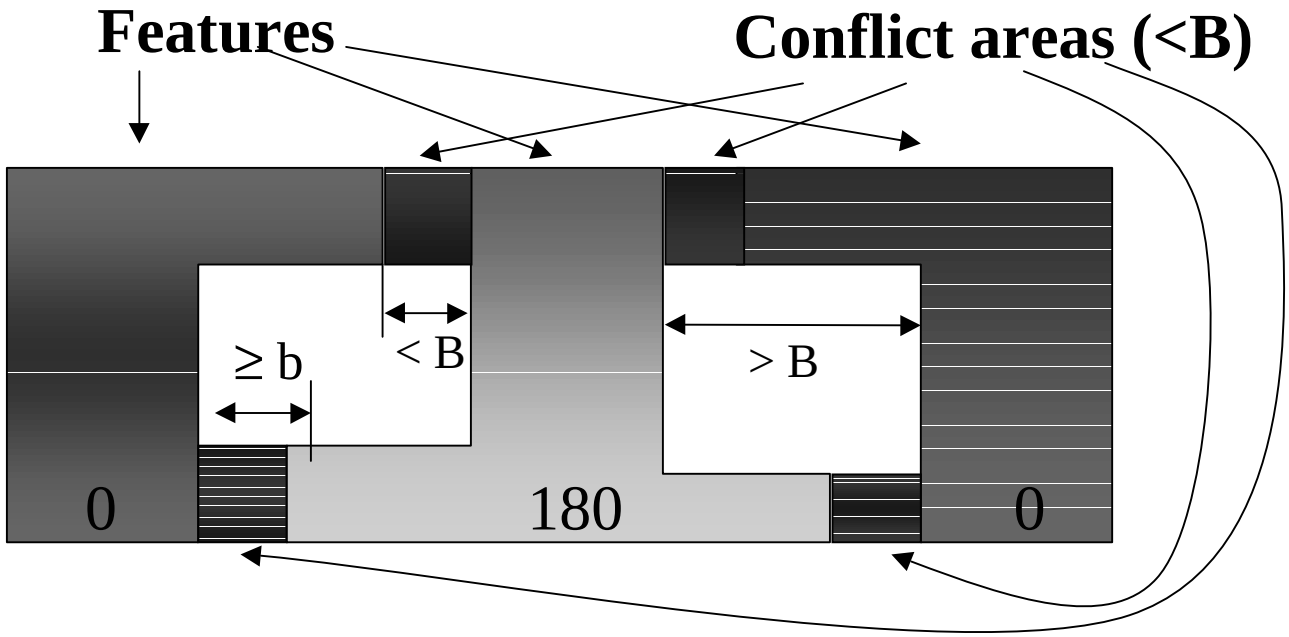


Global phenomena in PSM phase layout

Phase Assignment in PSM

Assign 0, 180 phase regions such that:

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



(Dark field, neg resist)

b ≡ minimum separation or width, with phase shifting

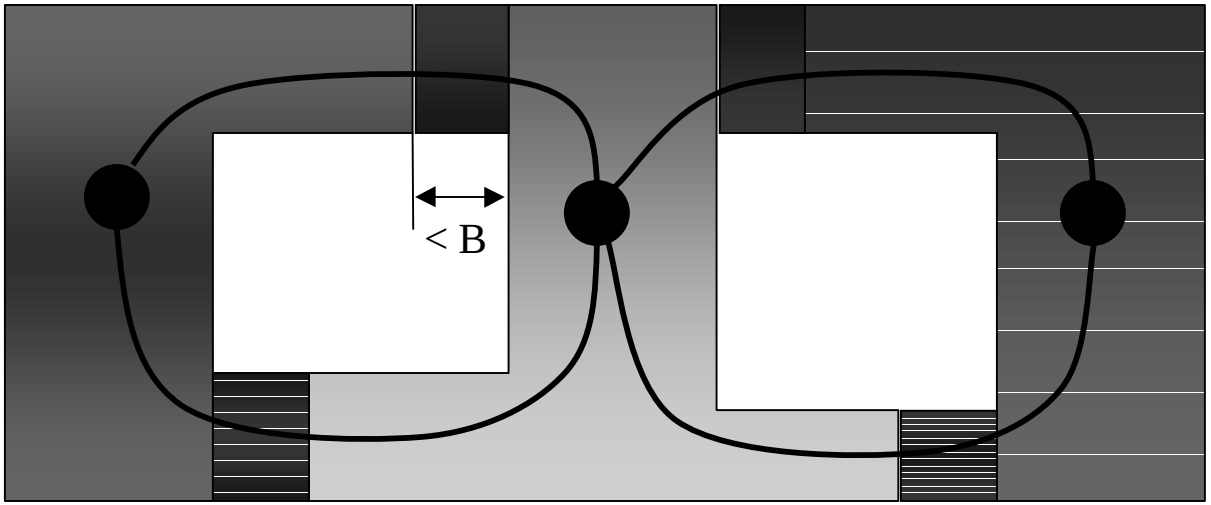
B ≡ minimum separation or width, without phase shifting

Conflict Graph

Vertices: features (or phase regions)

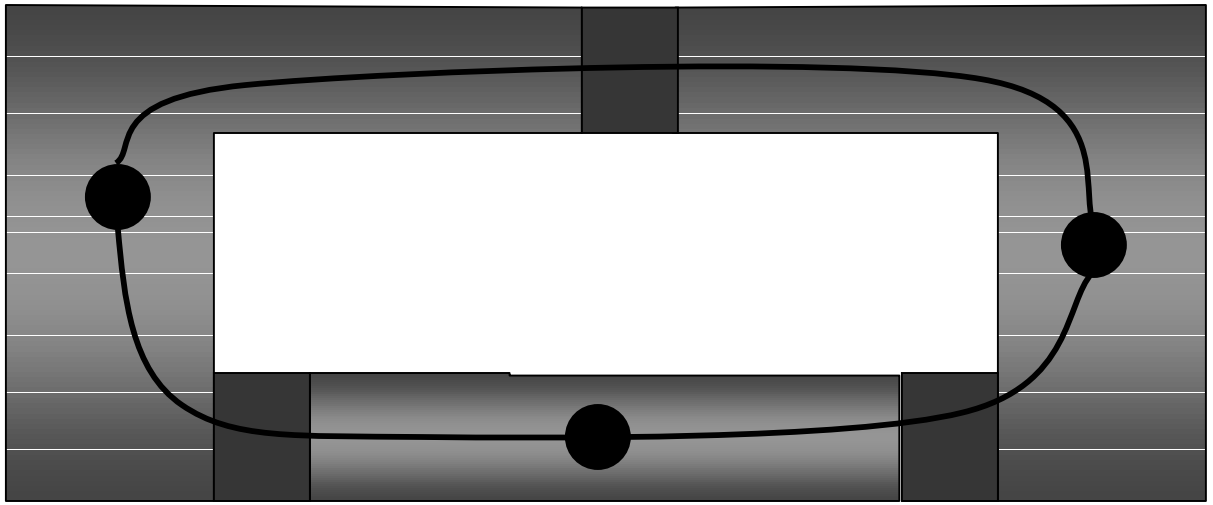
Edges: “conflicts” (necessary phase contrasts)

(feature pairs with separation < B)



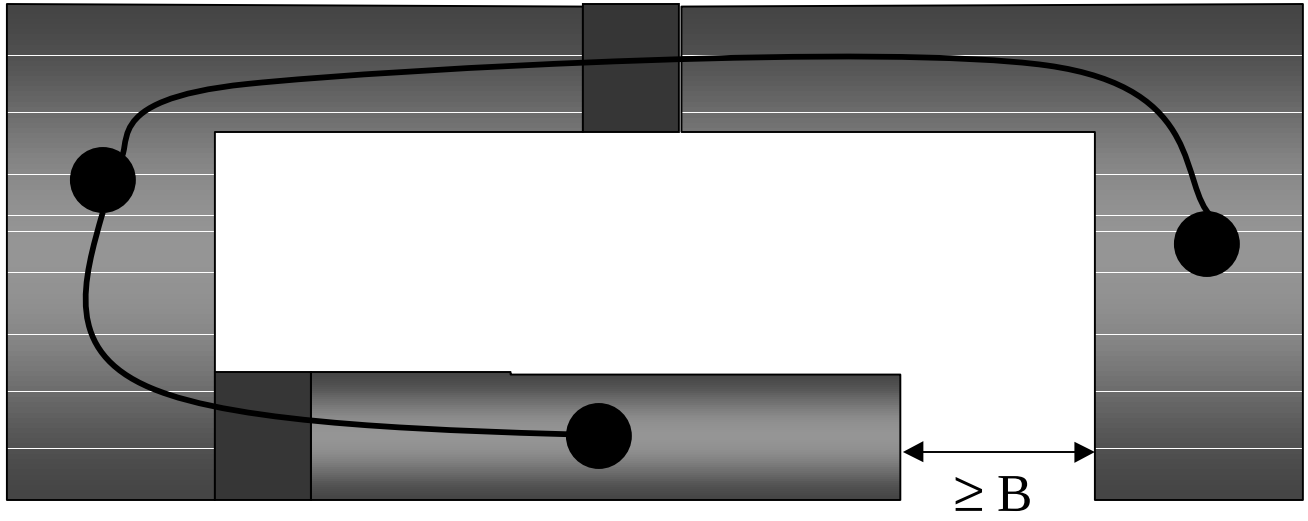
Odd Cycles in Conflict Graph

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



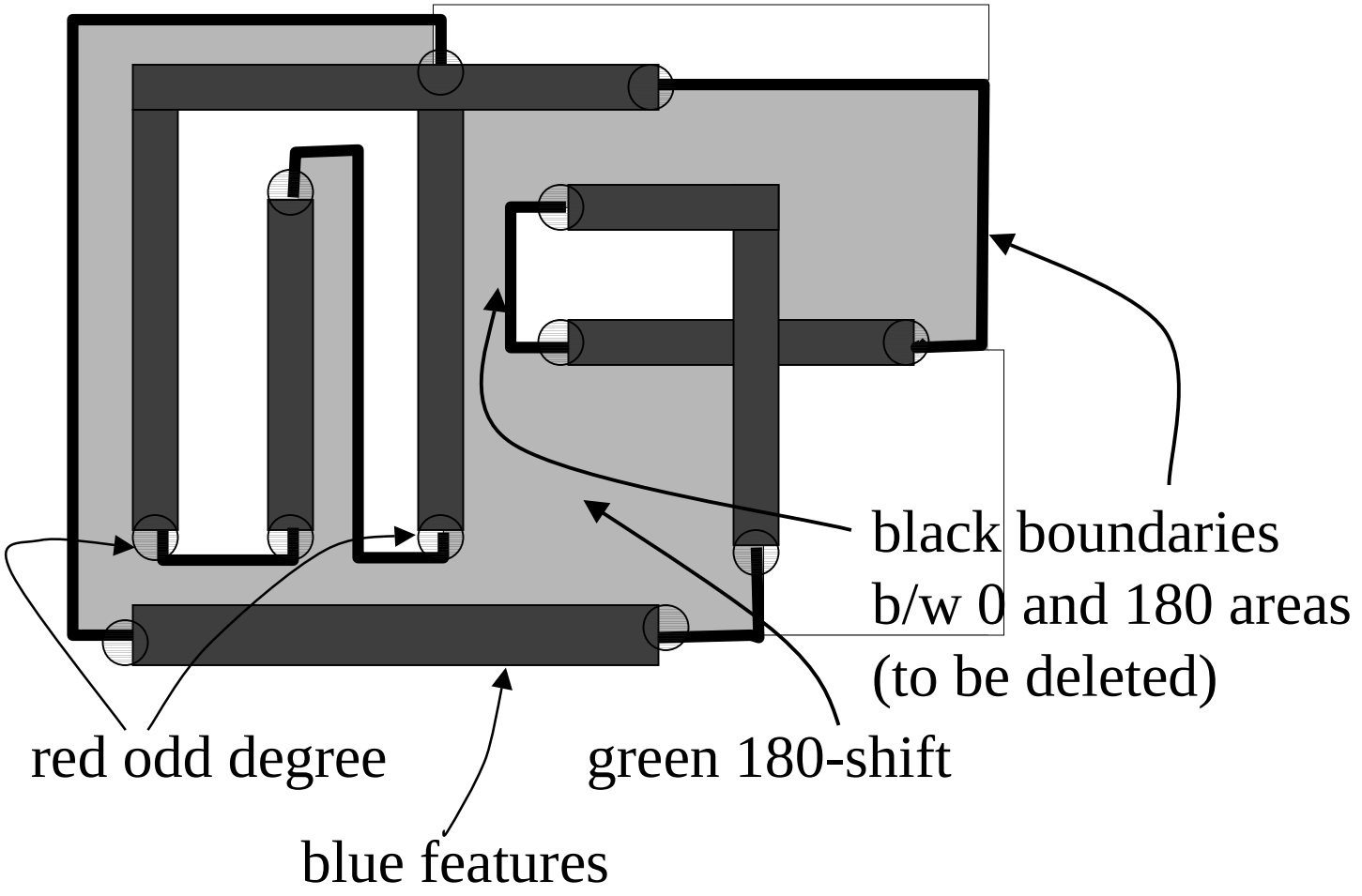
Breaking Odd Cycles

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



Bright-Field (Positive-Resist) Context

- **Every critical-width feature defined by opposite-phase regions**
- **Regions not defined a priori**



Value Proposition to Designers

- 0.10μm feature sizes in production in 1999
  - ≥2x performance
  - Higher yield
  - *"Transparent" to designer*

Benefit	***Gate-PSM***	***Full PSM***
Speed	++++	++++
Yield	++++	++++
Power	++	++++
Die Size	N/A	++++
Initial Generation	0.35 μm - 0.25 μm	0.15 μm

Problem Statements I

- Develop efficient algorithms for minimum-cost phase region definition and phase assignment in bright-field context
  - open: definition of cost (mfg difficulty, area, …)
- Continuum between sparse, dense criticality
  - DF Alt PSM + BF binary trim mask approach simple and elegant for sparse critical features
  - what about when all features are critical?

    (full-chip area opt, in addition to gate shrink)
  - can be treated as a routing problem (of phase edges)

Problem Statements II

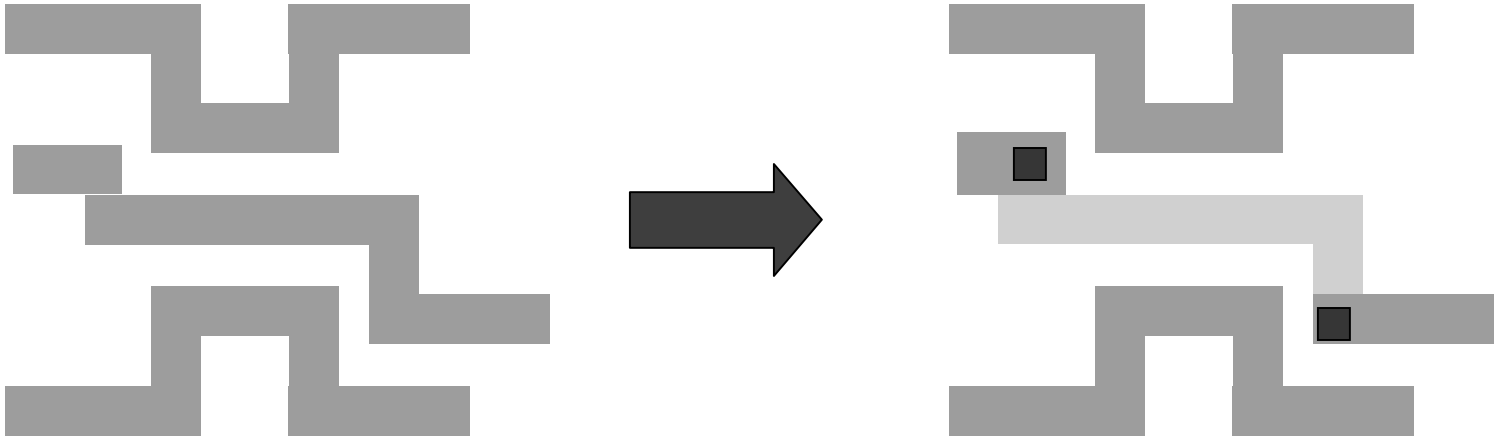
- New logic (mapping) and performance optimization formulations
  - with phase shifting, gate lengths and wire widths continuously variable between b and B
  - without phase shifting, gate lengths and wire widths must be at least B
  - not all features can be phase-shifted: function-driven

    What is optimal choice of phase-shifted features, and their sizes?

Problem Statements III

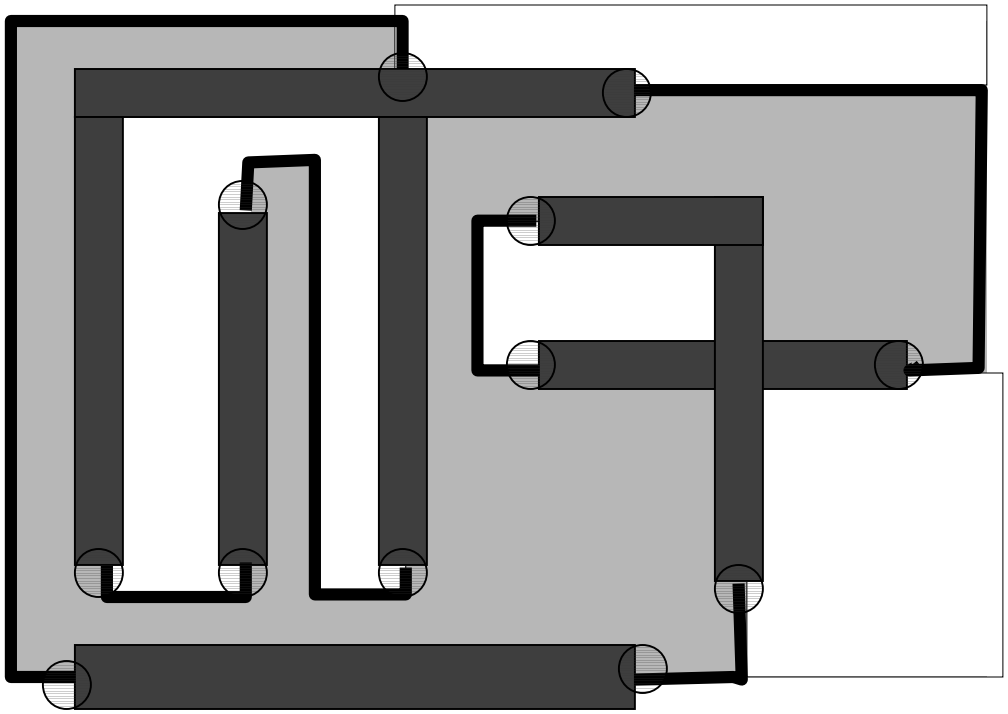
- Understand PSM implications for custom layout
  - define a taxonomy of phase conflict
  - no set of traditional design rules can handle all phase conflicts $\rightarrow$ what are "good layout practices"?
    - "no T's on poly"
    - "fingered transistors should have even-length fingers"
    - etc.
- Address PSM as a multi-layer problem
  - e.g., conflict can be solved by re-routing a connection to another layer

Layer Assignment

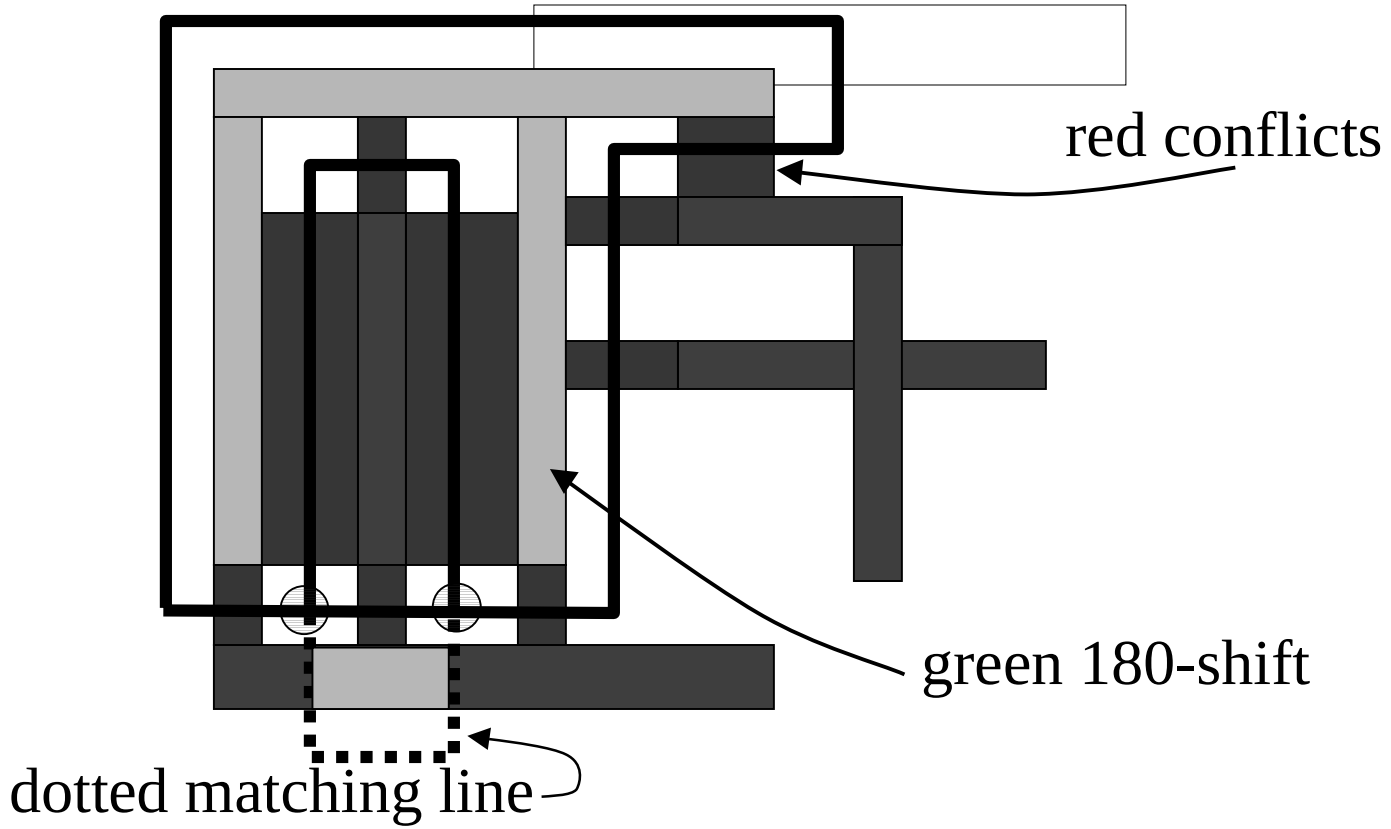


Problem Statements IV

- **<u>Unified theory of PSM design</u>**: Can bright- and dark-field, positive and negative resist contexts all be addressed by a single graph-algorithmic framework?



Near-Duality for Dark Field



any path matching odd nodes of dual graph should go through features - split into different phases

Local phenomena in OPC

Problem Statements V

- Pass functional intent down to OPC insertion
  - OPC insertion is for predictable circuit performance, function
  - Problem: make only corrections that win $$$, reduce perf variation (i.e., link to performance analysis, optimization) ?
- Pass limits of mask verification up to layout
  - Problem: avoid making corrections that can't be manufactured or verified

Problem Statements VI

- Minimize data volume
  - Problem: make corrections that win $$$, reduce perf variation up to some limit of data volume for resulting layout (== mask complexity, cost)
- Layout needs models of OPC insertion process
  - Problem: taxonomize implications of layout geometry on cost of the OPC that is required to yield function or “faithfully” print the geometry
  - find a realistic cost model for breaking hierarchy (including verification, characterization costs)

Hierarchical and Reuse-Centric Contexts

Issues Raised by Hierarchy, Reuse

- Large data volume and verification costs when hierarchy is broken -- but PSM and OPC are both context-dependent!
- Standard-cell approach requires absolute composability of cells -- this must somehow be guaranteed as we move into PSM regime

Problem Statements VII

- Given a cell library, what is its flexibility (i.e., composability with respect to PSM) ?
- Given a standard-cell layout and allowed increase in hierarchical layout data volume, what is the maximum reduction in area obtainable by creating new cell masters with different phase layout solutions?
- Given a standard-cell layout with phase-solution instantiations that induce conflicts, what is minimum-cost removal of phase conflicts?
  - DOF's: change instance, shift, space, mirror, ...

Integrated Layout Flow, 1

- Gate-level netlist, performance constraint budgeting, early context (mask/litho technology, area density...)
- Standard-cell placement with integrated compatibility awareness (composable PSM layouts)
- Global and detailed routing, cell resynthesis on fly
  - delay, noise, reliability assumptions = constraints
  - OPC- and PSM-aware min-cost layout synthesis subject to constraints (e.g., minimize costs of breaking hierarchy, follow “good practices”, etc.)
  - fill abstractions (for parasitic extraction) in constraint-driven routing

Integrated Layout Flow, 2

- Density analysis, CMP-fill estimation based on detailed routing
- Post-detailed routing performance analysis
- PSM phase assignability check for all layers
  - new compaction constraints as necessary
  - layout compaction or incremental detailed routing
  - until pass phase assignability, performance analysis
  - note: integration with full-chip geometric compaction!
- Actual dummy fill insertion
  - issues: data volume

Integrated Layout Flow, 3

- Detailed physical verification (geom, conn, perf)
- Full-chip OPC insertion
  - issues: min-cost OPC that achieves required function
  - issues: data volumes, metrics, intermediate formats
  - issues: tools stepping on each other (line extensions in DSM router rules are “zeroth-order OPC”, for example)
- Full-chip printability check
- Silicon-level DRC/LVS/performance analysis

Conclusions

- New problem formulations
  - PSM: layout practices, automated full-chip and standard-cell compatible solutions
  - OPC: taxonomy of local phenomena, data reduction
  - function-driven corrections (can filter complexity)
  - hierarchy, data volume, reuse concerns
- New tool integrations
  - compaction, on-the-fly cell synthesis, incremental detailed routing
  - graph-based (verification-type) layout analyses
  - new performance opts, even logic opts