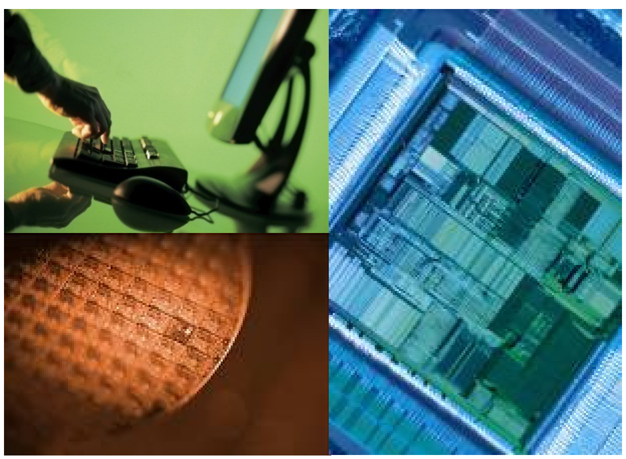
On Improving Optimization Effectiveness in Interconnect-driven Physical Synthesis

Prashant Saxena
Vishal Khandelwal
Changge Qiao
Pei-Hsin Ho
J.-C. Lin
Mahesh Iyer



March 29, 2009

ACM International Symposium on Physical Design

San Diego, CA

SYNOPSYS®
Predictable Success

Overview

- Evolution of physical synthesis net models
  - Net delay prediction during pre-route optimization
- Guaranteed routes for predictable delays
  - *Persistence*: pros and cons
  - Focus on system architecture and flow, rather than on implementation details
- Experimental Validation

Optimization Effectiveness

- Uncertainty due to :
    - Subsequent design transforms
    - Estimated routes and parasitics
    - Process variation
- Bad estimates can mislead optimization trajectory
- Pre-route vs. post-route optimization
    - Tradeoff between data accuracy and flexibility

The Evolution of Net Models - I

- Poor wire scaling ➔better net delay prediction needed

- Fanout-based wireload models from logic synthesis
  - Statistical, ignores actual wirelength
- Placement-based HRPM models
  - Underestimates wirelength for multi-pin nets
  - Best possible length even for 2,3-pin nets
- Empirical fanout compensation factors (eg RISA)
  - Inaccurate source-to-sink estimates

The Evolution of Net Models - II

- Rectilinear Spanning/Steiner trees
  - Simple greedy schemes
  - Modeling of blockages
  - Timing-driven trees
  - Layer assignment
  - Congestion-aware topologies
  - Variable accuracy
    - Increase sophistication as design converges

- Use the global router for virtual routes
  - …rather than merely using background congestion map
  - Same engine for estimation and actual routing

So What's the Problem?

- Routers are unstable
  - Small congestion map error can change a route significantly
  - Ditto for timing
  - Additional ripup-and-reroute iterations can also change routes
  - Netlist evolves during estimation and actual routing
- Runtime overhead

- Estimation will *always* have the instability problem

Guaranteeing the Routes

- Fix the topology, layer assignment, and detours
  - Accurate net delays and parasitics
    - Custom designers often “preroute” some nets by hand
- If it’s so obvious, why don’t we always do it?
  - Netlist is still evolving during pre-route estimations
    - Frozen nets inhibit optimization
  - **Routability impact**
    - Restricts flexibility of ripup-and-reroute
- Persistence:
  - Select and route a ***small*** number of ***carefully chosen*** nets
  - Update nets/routes as needed during subsequent optimization

Net Selection - I

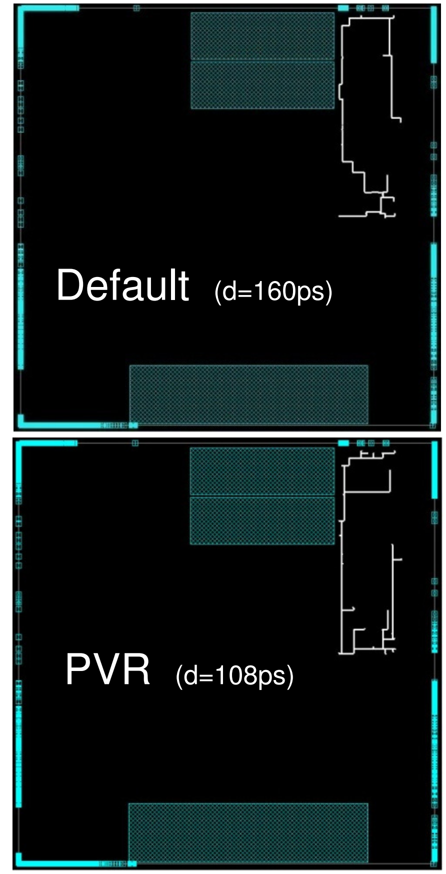
- *Only a few nets are critical and unpredictable*
  - Very critical nets: usually not unpredictable
    - Get preferential resource access from timing-driven router
    - Side-loads isolated as expected
  - Non-critical nets: Unpredictability doesn't matter
  - Small nets: Net delay is insignificant
- Focus on the "almost-critical" nets
  - Delay estimator doesn't identify them as critical (ideal routing assumed)
    - So, they are left unoptimized
  - If routed badly, expensive post-routing fixes are needed
- High fanout nets : increased unpredictability due to topology choices

Net Selection - II

- Empirically profiled unpredictable nets in several real-world designs
  - Correlated pre-routing vs. post-routing delay discrepancy with parameters like fanout, delay, criticality, wirelength etc.
  - Net sensitivity: Normalized net length increase before timing or electrical DRC violation occurs
- Net selection engine
  - First filter out nets with one or more of:
    - Small physical span
    - Large timing slack
    - Low sensitivity and single sink
  - Heuristically cost the remaining nets
    - Cost function based on parameters that correlate strongly with unpredictability

Routing the Selected Nets

- Use high-quality interconnect synthesis
  - Few enough nets that runtime impact is limited
    - Proportional to number of selected nets
  - Must be congestion-aware
    - Leave enough tracks for remaining nets
- Alternatively, extract routes from a global routing pass
  - Individual routes may not be as great
  - Runtime overhead independent of number of selected nets
- Good route quality desirable, but improved predictability can make up for degraded quality

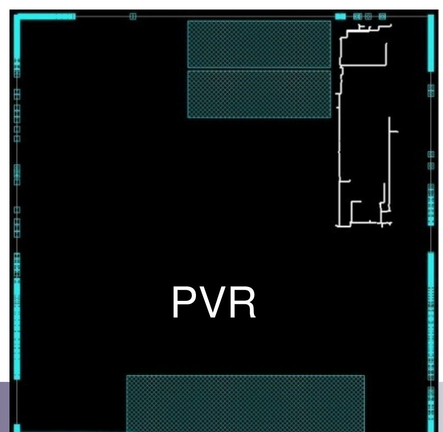


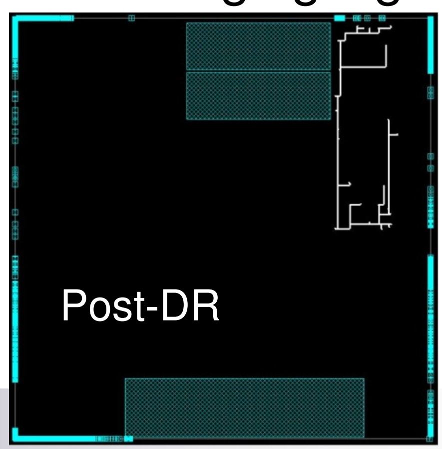
Interconnect-aware Optimization

- Use actual, route-based parasitics for persistent nets
- Don't "freeze" a net's routing (as with preroutes)
  - Else, optimization is restricted
- Some transforms require ECO route updates
  - Sizing
  - Buffering: derive routes of newly-created nets from original persistent route
- A few (e.g., remapping) may invalidate routes
  - Revert to default virtual router
- Synthesis changes set of nets desired for persistence
  - In practice, large fraction (>70%) of nets survives

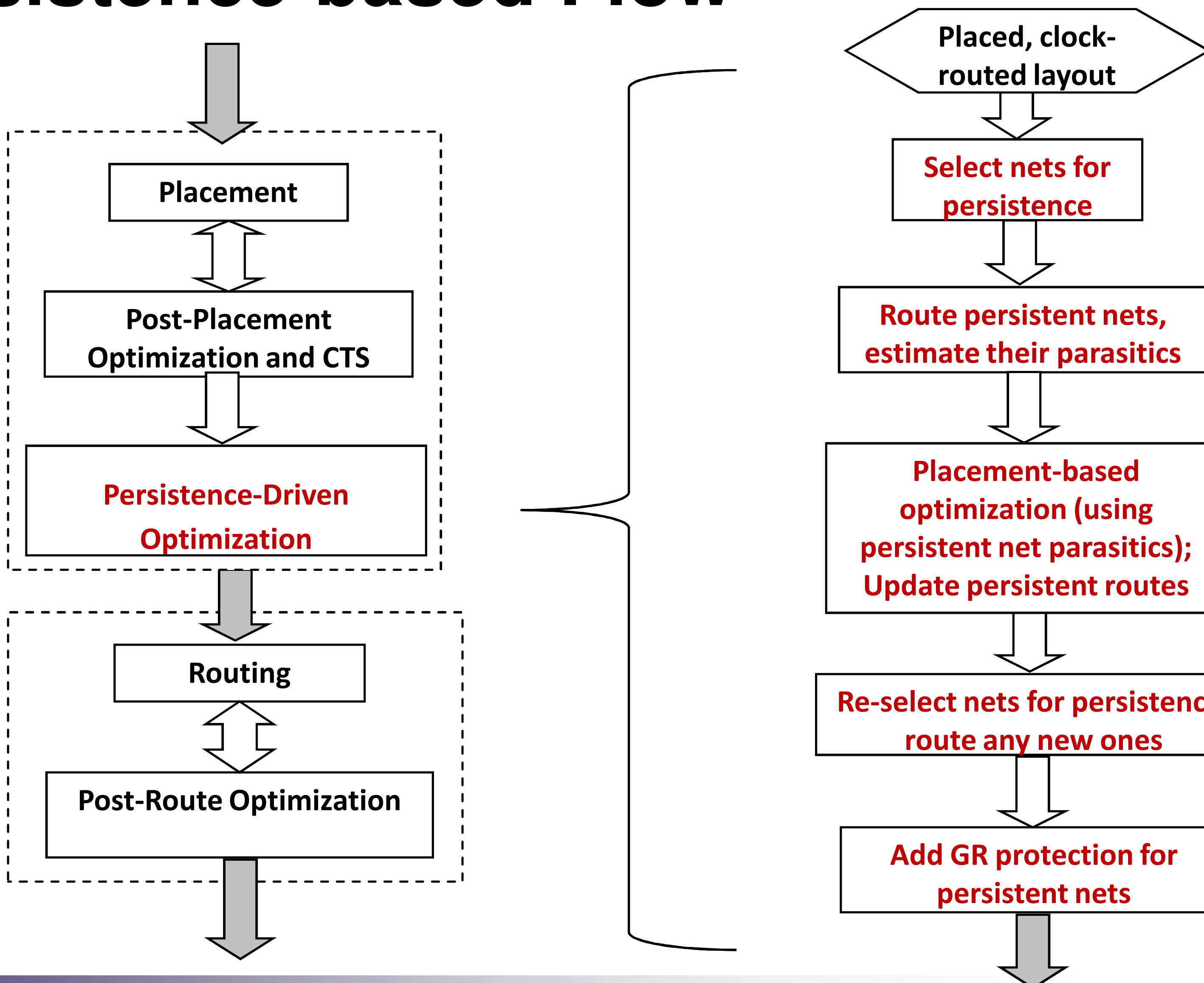
Controlling the Router

- Preserve persistent routes in first GR pass
  - Translate from lightweight pre-route representation to actual global routes
  - Timing-driven GR for remaining nets
  - Protecting >4-5% of nets starts impacting routability
- No protection during TA, DR, or subsequent optimization/ECO-routing
  - “Soft” constraints that can be overridden
  - Unusual to find persistent routes changing significantly





Persistence-based Flow

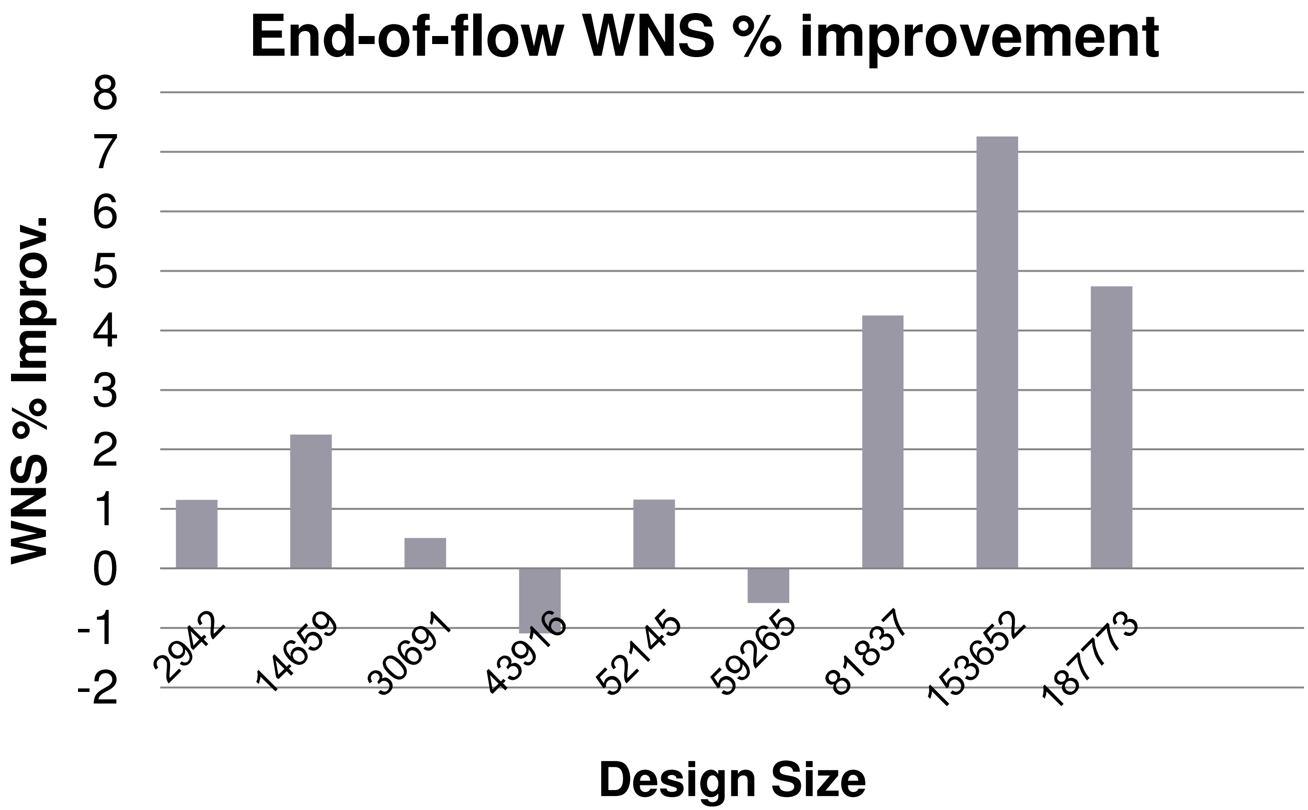


Validating the Automated Flow

- Implemented in industrial physical synthesis flow
- Tested on 65nm real-world benchmarks
- Results reported at end of post-routing optimization
  - To investigate whether QoR benefits percolate to end-of-flow
- Apples-to-apples: control flow gets extra optimization pass
- Route persistent nets using global router
  - Avoids contamination of results due to improved interconnect synthesis

- WNS change normalized to clock periods

Automated Flow: Experimental Data

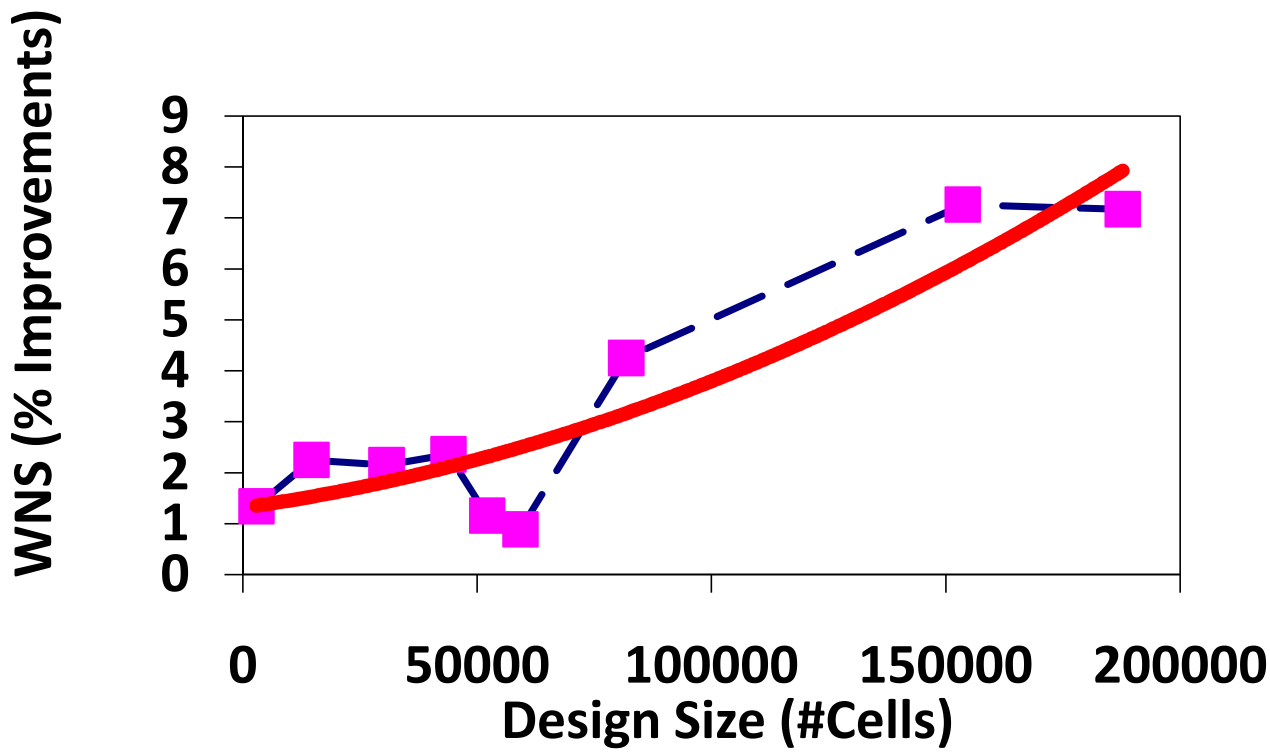


- Avg. WNS Improvement: 2.19%
- Avg. CPU Overhead: 3.4%
- No discernible routability impact (Post-DR DRViolations)

Maximizing QoR

- High end designs: often rerun with multiple settings to maximize QoR
- To model this niche, we tried limited design-specific parameter tuning
- Tried four versions of our flow
  - %age of selected nets: 2% or 3%
  - Cost function for net selection: two variants

Maximizing QoR: Experimental Data



- Avg. WNS Improvement: 3.21%

Concluding Remarks

- Proposed persistence as a way to make pre-route optimization more effective
- Discussed how to address its biggest risk (viz., routability impact)
  - Smart selection of a small number of key nets is crucial
- Implemented in a leading real-world physical synthesis system
  - Significant QoR benefits that percolate to final handoff
  - Benefits scale well with design size
- Natural next step in evolution of net models for physical synthesis