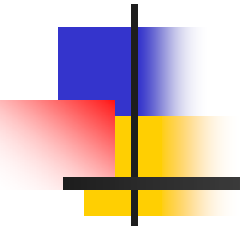
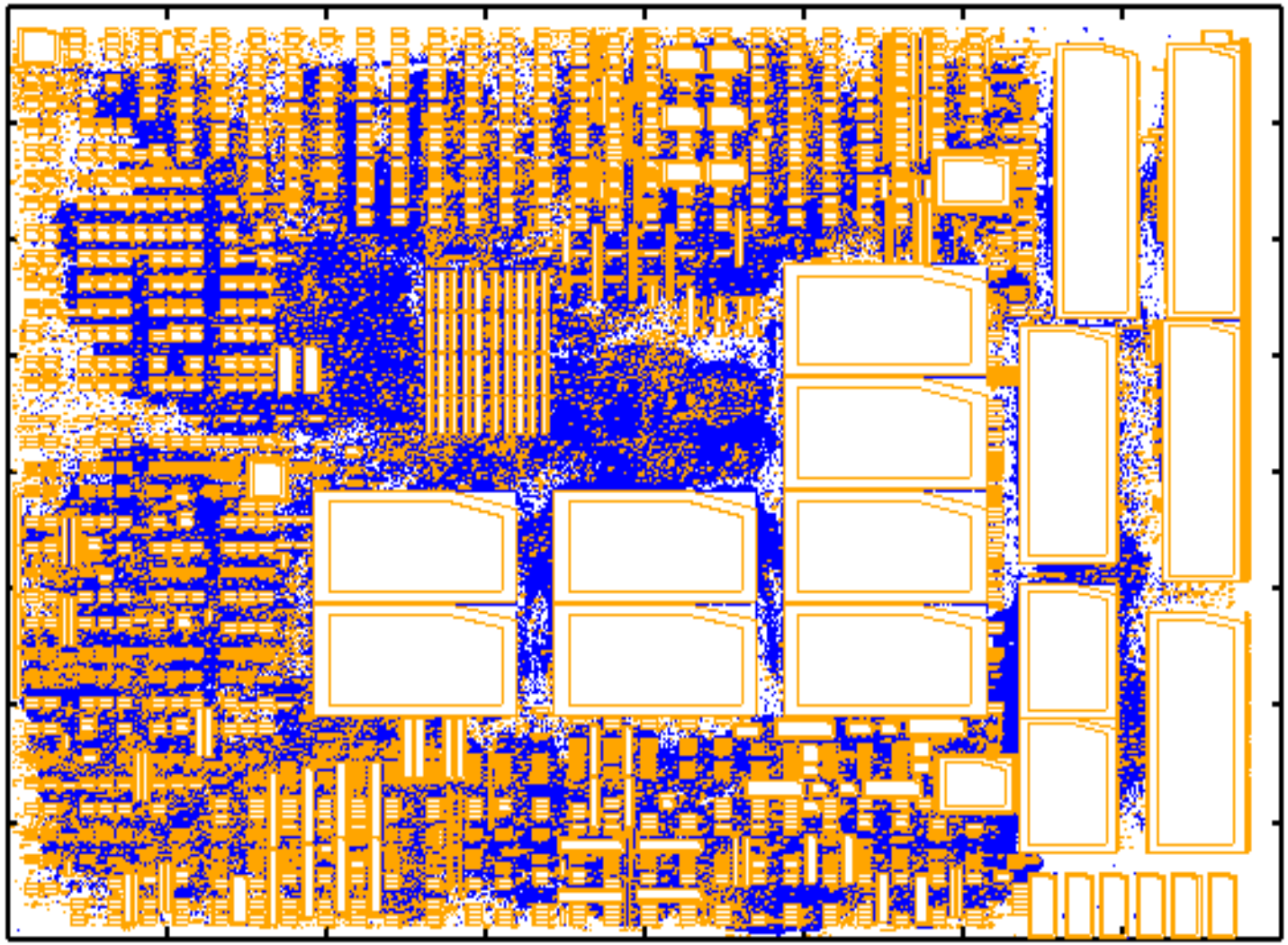


Fast Electrical Correction Using Resizing and Buffering

Shrirang Karandikar, Charles J Alpert, Mehmet Yildiz, Paul Villarrubia, Steve Quay, Tuhin Mahmud

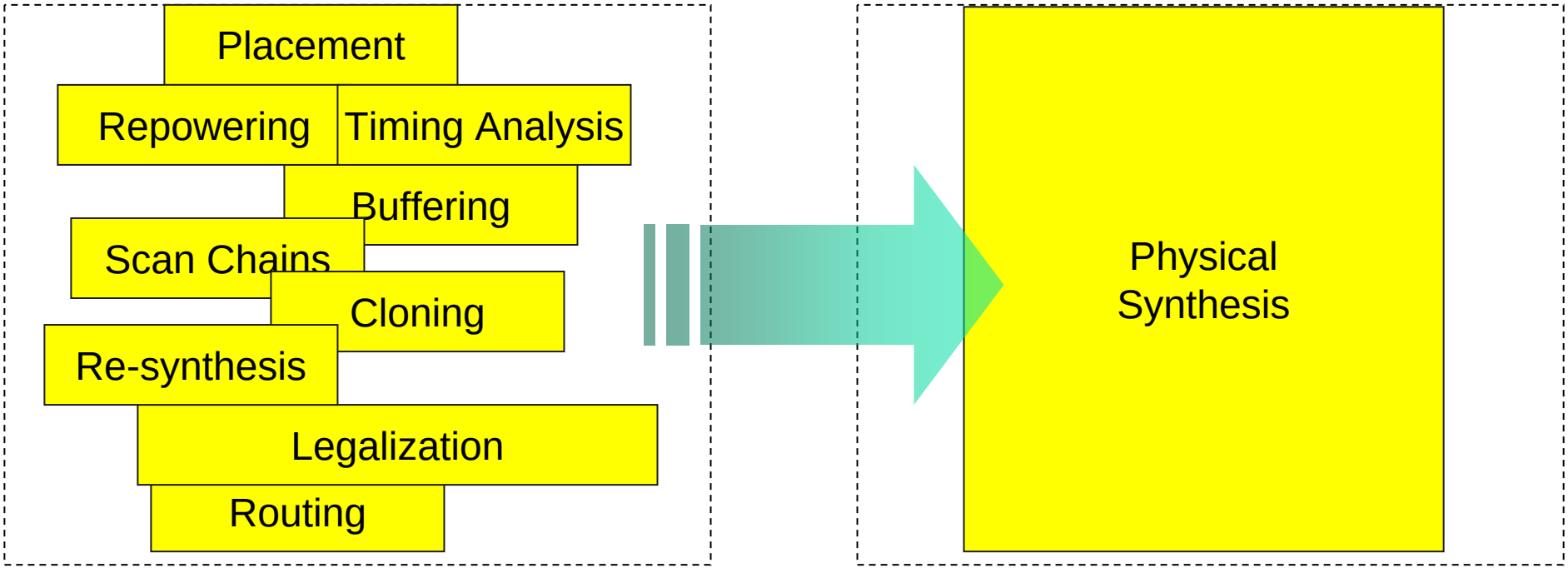
Constructive versus Polishing

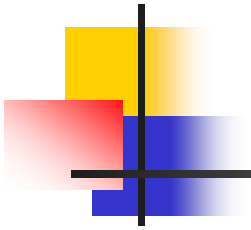
Example Large ASIC



What is Physical Synthesis?

Combines multiple separate steps into one to perform timing (design) closure

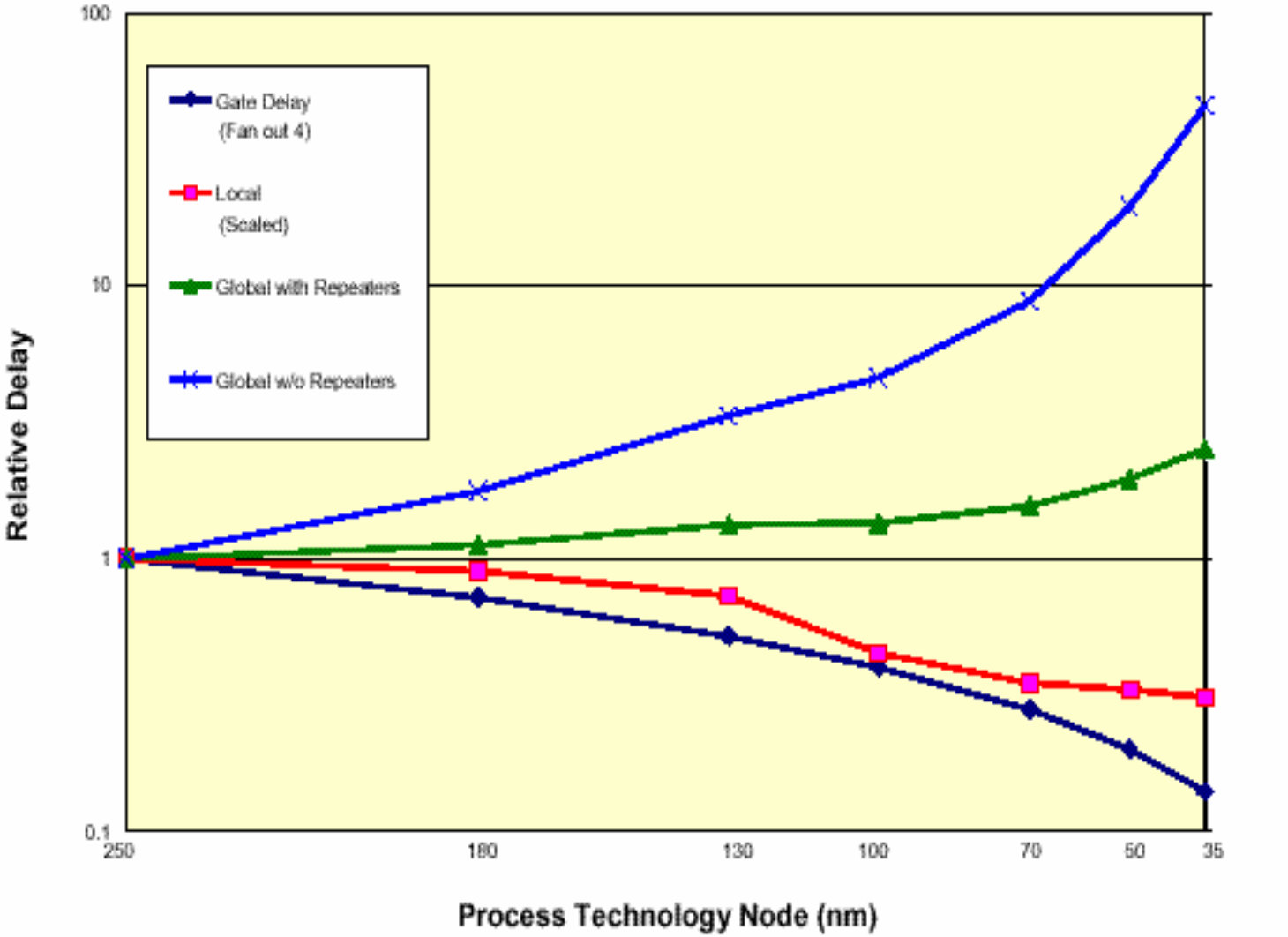




Big Challenges in Physical Synthesis

- Designs are enormous
  - 4 to 8 million objects now
  - Must be fast
- Interconnect resistance increasing
  - Density / white space management
  - Buffer explosion
  - Multiple cycles to cross chip
- Support for full design closure
  - Noise, power, routability, yield

Global Interconnect Dominance



Building Blocks of Physical Synthesis

Placement
(find non-overlapping locations to minimize wirelength)

Critical Path Opts
(incremental synthesis to optimize most critical regions)

Electrical Correction
(buffer and repower to fix slew and cap constraints)

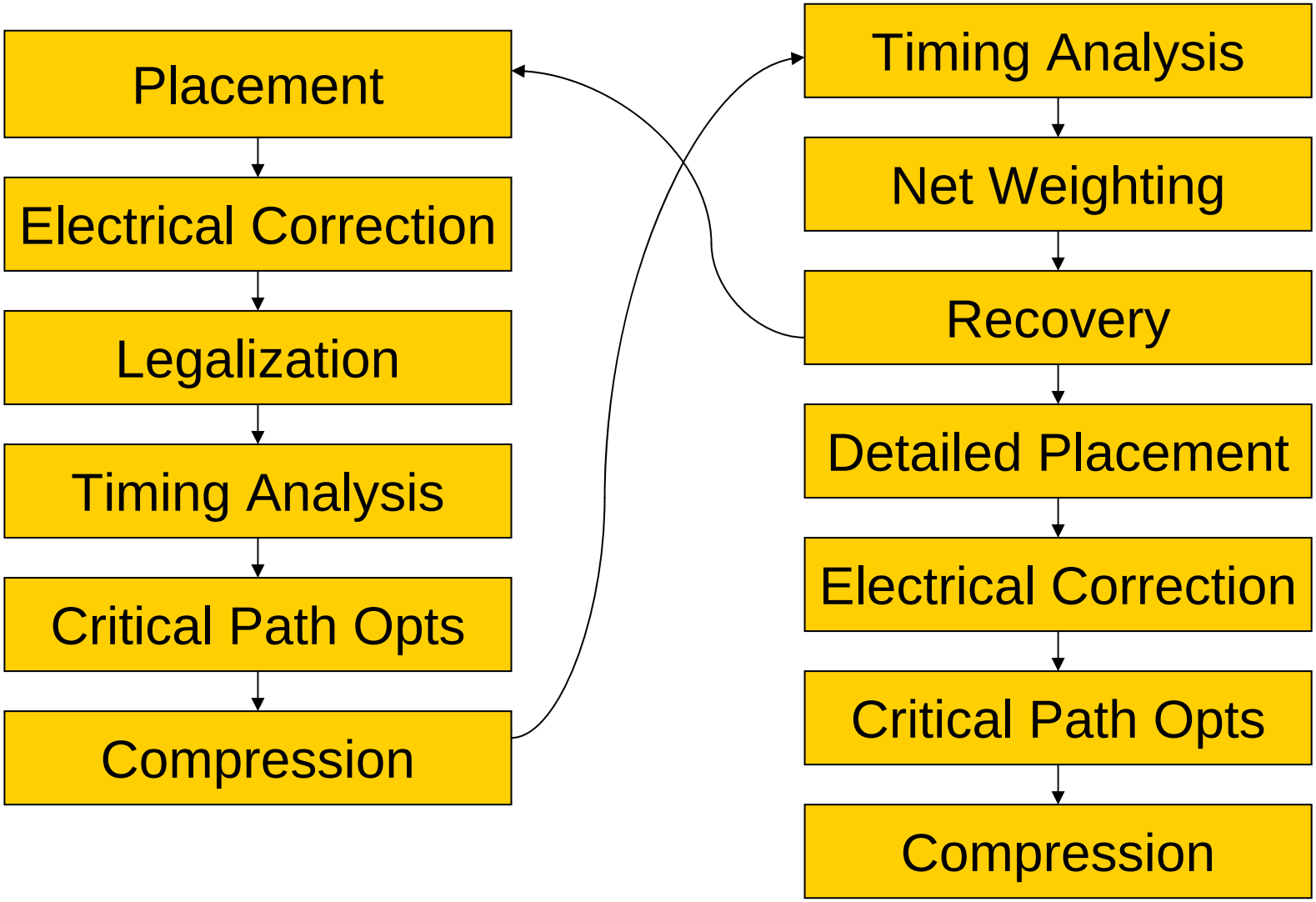
Compression
(Optimize rest of critical regions)

Legalization
(place those buffers and logic in legal locations)

Area Recovery
(reduce area without hurting timing)

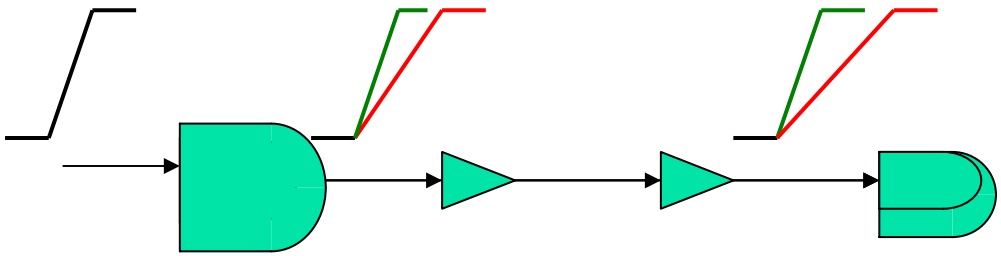
Timing Analysis
(identify critical logic)

Example Physical Synthesis Flow



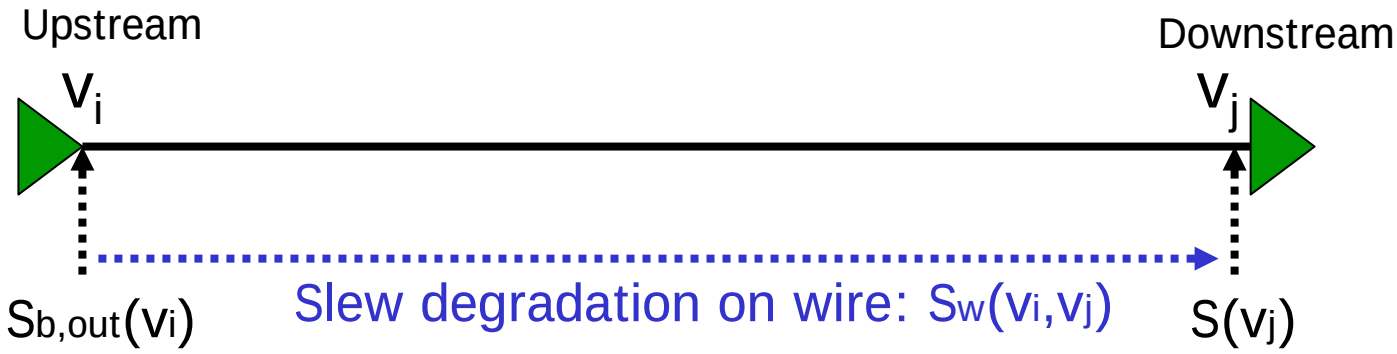
Electrical Correction Necessary to Time Design

- Slew violations
  - Size down sink
  - Resize source
  - Buffer net



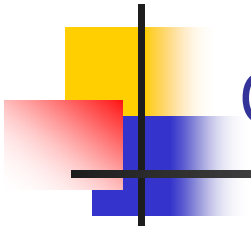
- Capacitance violations
- Fanout violations

Slew Model



$$s(v_j) = \sqrt{s_{b,out}(v_i)^2 + s_w(v_i, v_j)^2}$$

$$s_w(v_i, v_j) = ElmoreDelay(v_i, v_j) \cdot \ln 9$$



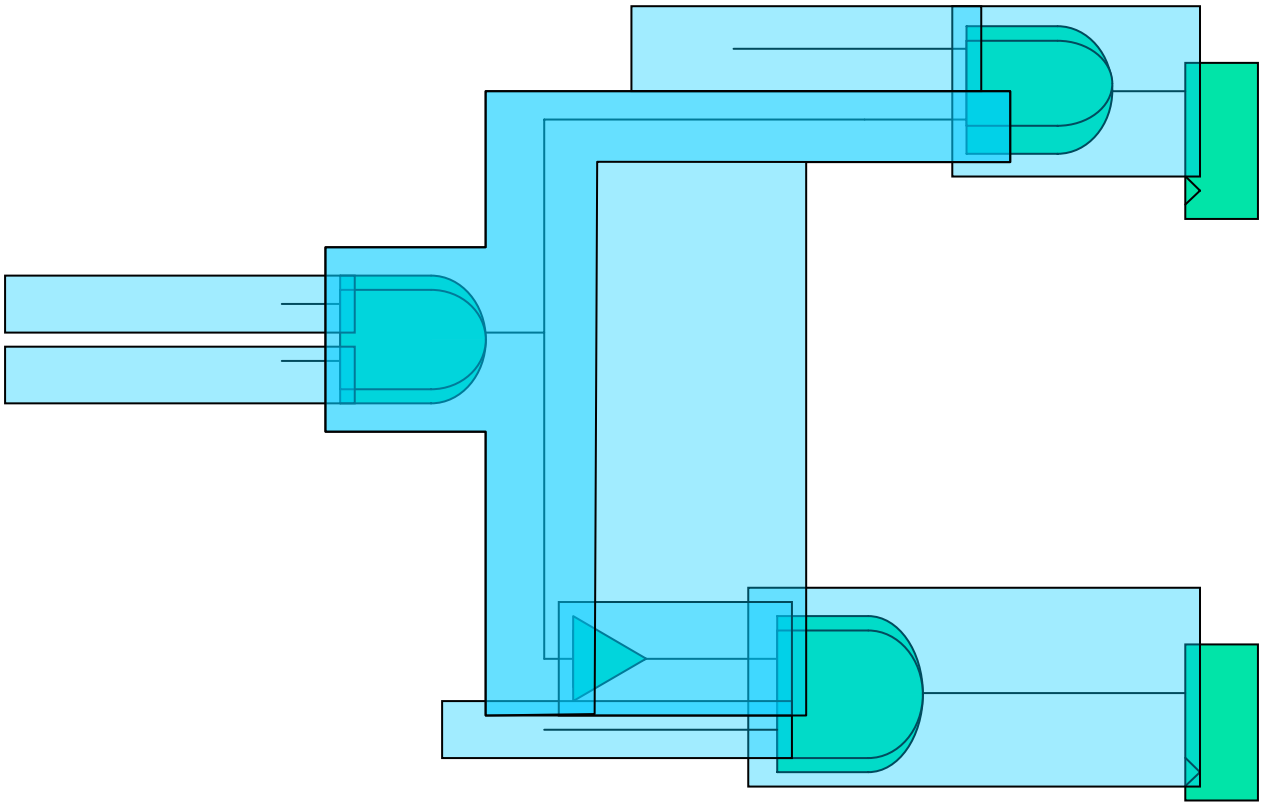
Get the Design Into Good Shape

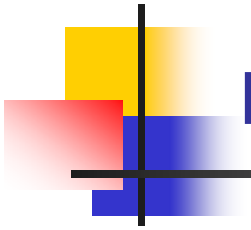
- Pick aggressive target slew
- Buffer / repower to meet target
- Power down non-critical gates
- Power up critical ones
- Choice of slew target has large impact on final area and timing

Basic Approach to Electrical Correction

- Process gates right to left
- If net has no violations, size source gate down
  - saves area
  - reduce load on inputs
- If net has violations, size gate up
- If violations still exist, buffer the net
- Linear time algorithm

EVE in Action





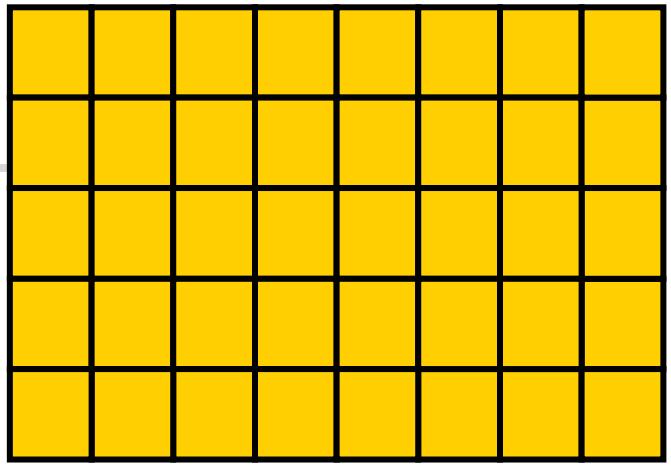
Legalization

- Electrical correction just repowered and performed massive buffering. How do you legalize?
- No bin model
- Bin-based model
- Incremental legalization

No Bin Model

- Place every cell, buffer at ideal (x,y) coordinate
- Like having one giant bin
- Permits “pile-ups”
- Let legalization handle it afterwards
- Advantage: faster for optimization
- Disadvantage: things move far away

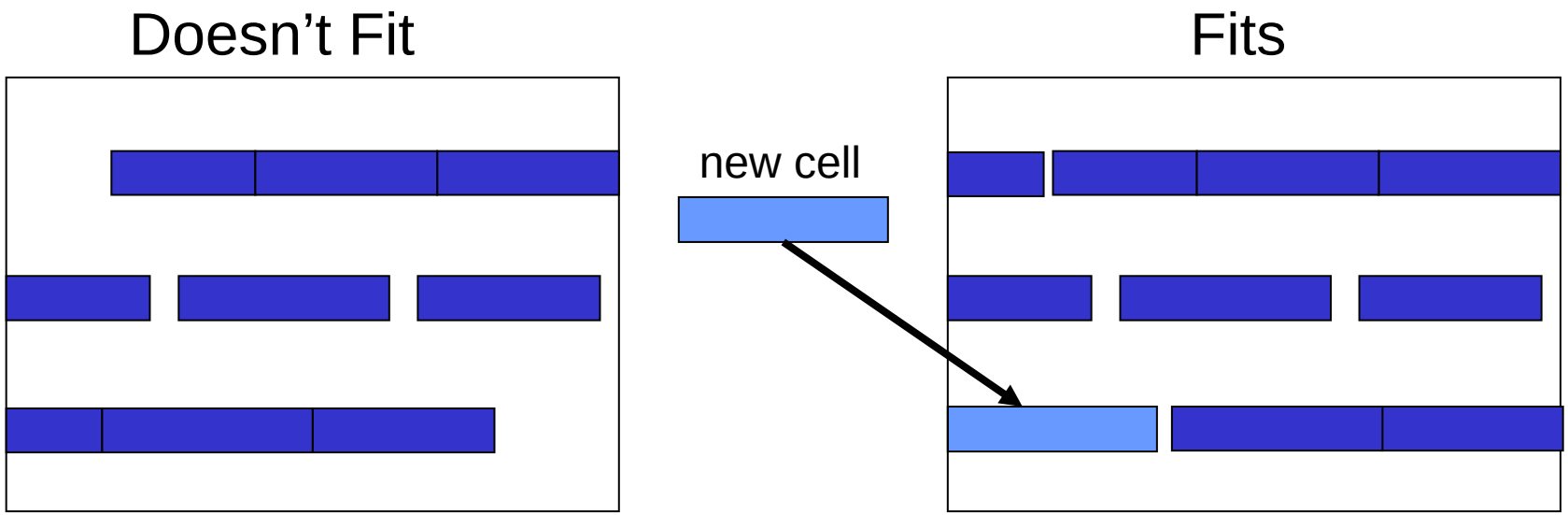
Bin Model



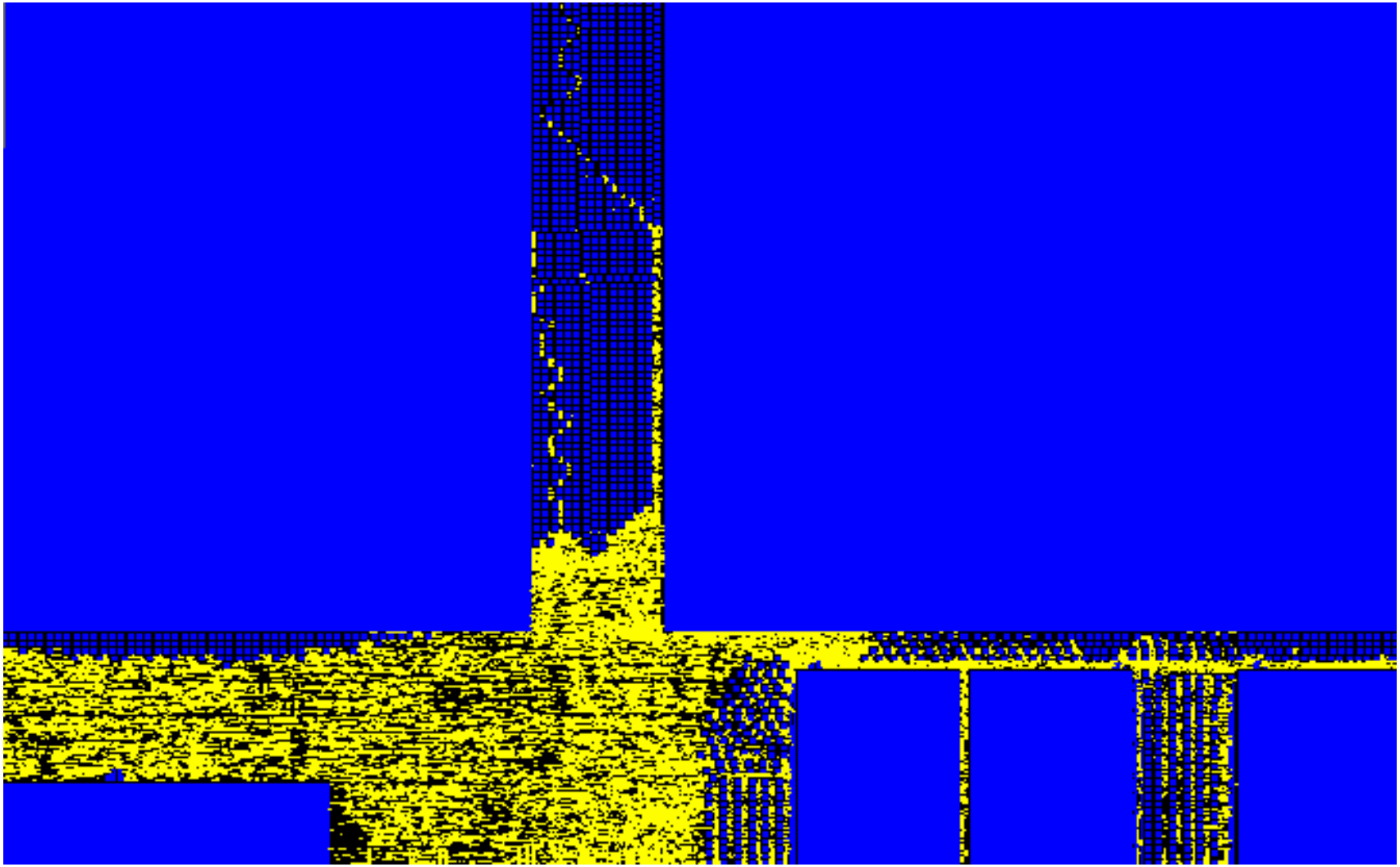
- Divide region into bins
- Each cell assigned to bin
- If inserting into bin, check area
- Full bins prevent repowering /buffering
- Could try finding adjacent bin with free space
- Advantage: less likely that legalization will move things far away
- Disadvantage: may prevent fixing violations

Puzzle Fit Problem

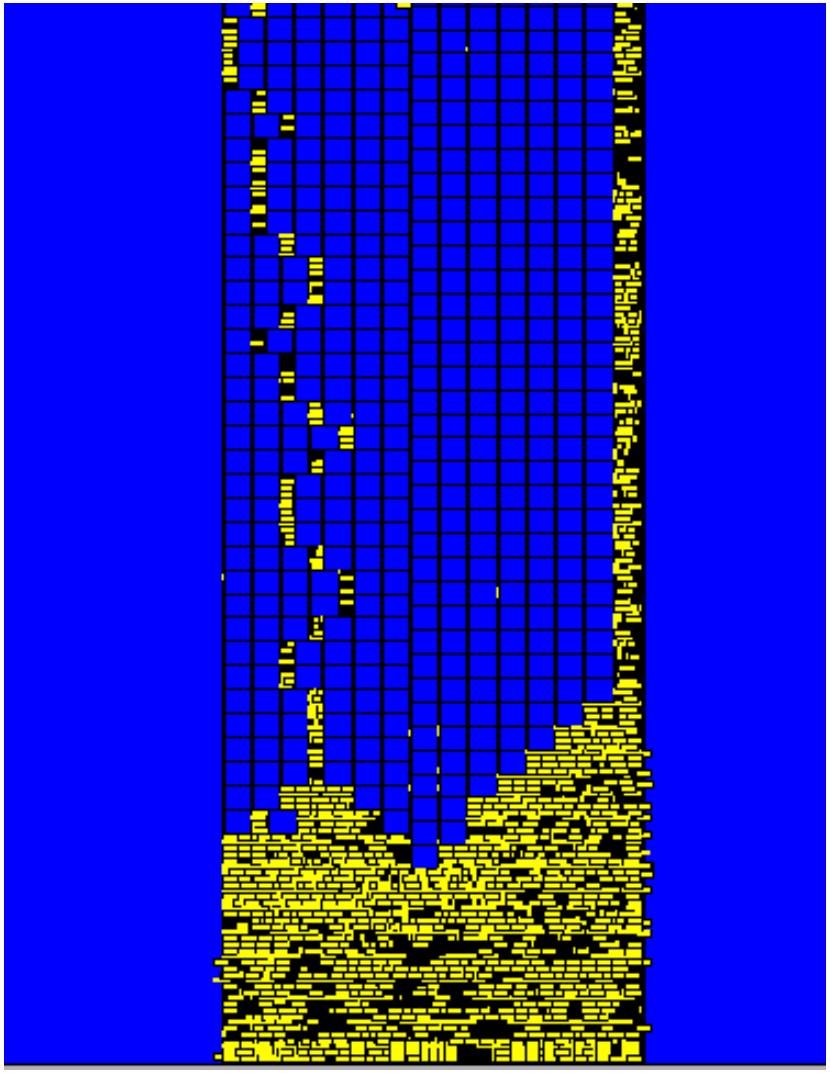
Two bins with same set of cells and same total area

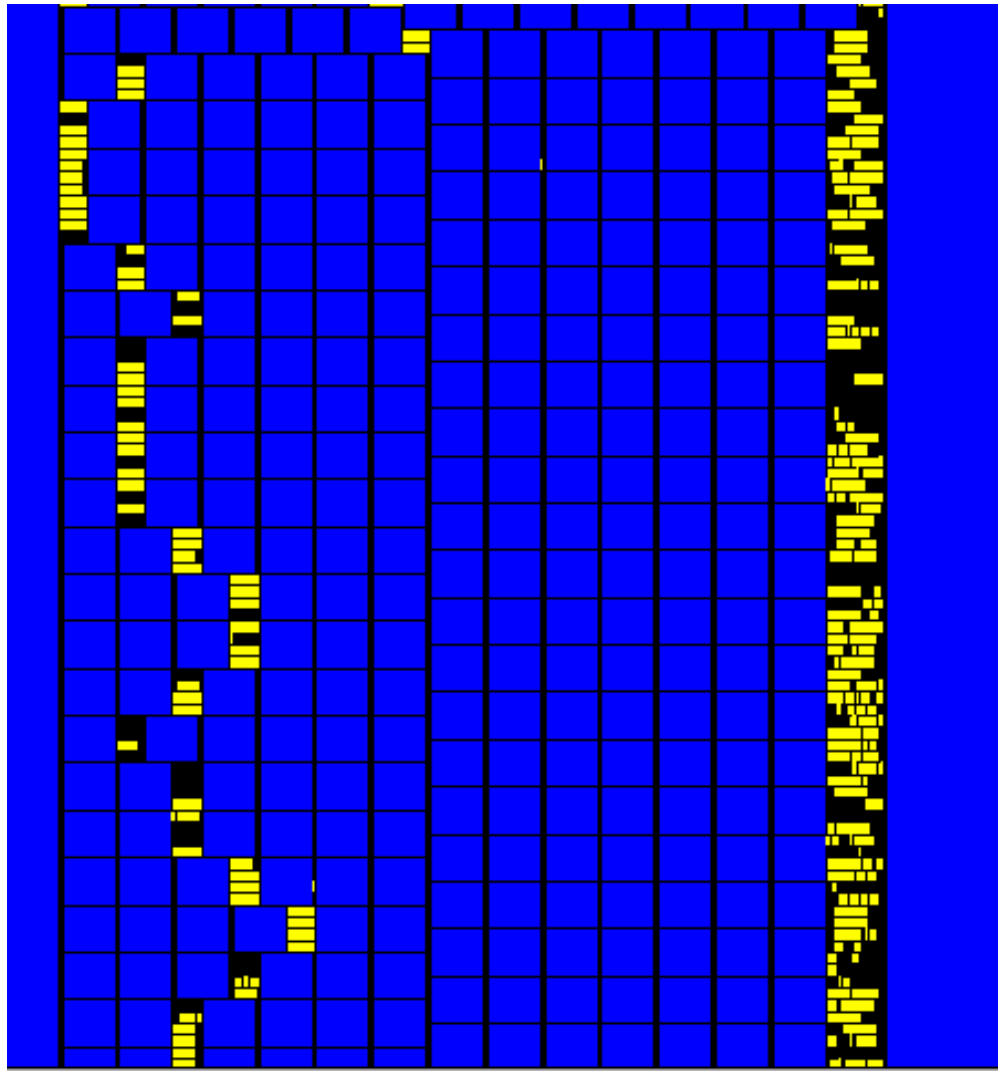


Puzzle Fit Example



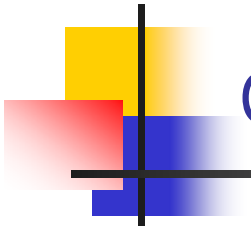
A Closer Look





Buffering for Electrical Correction

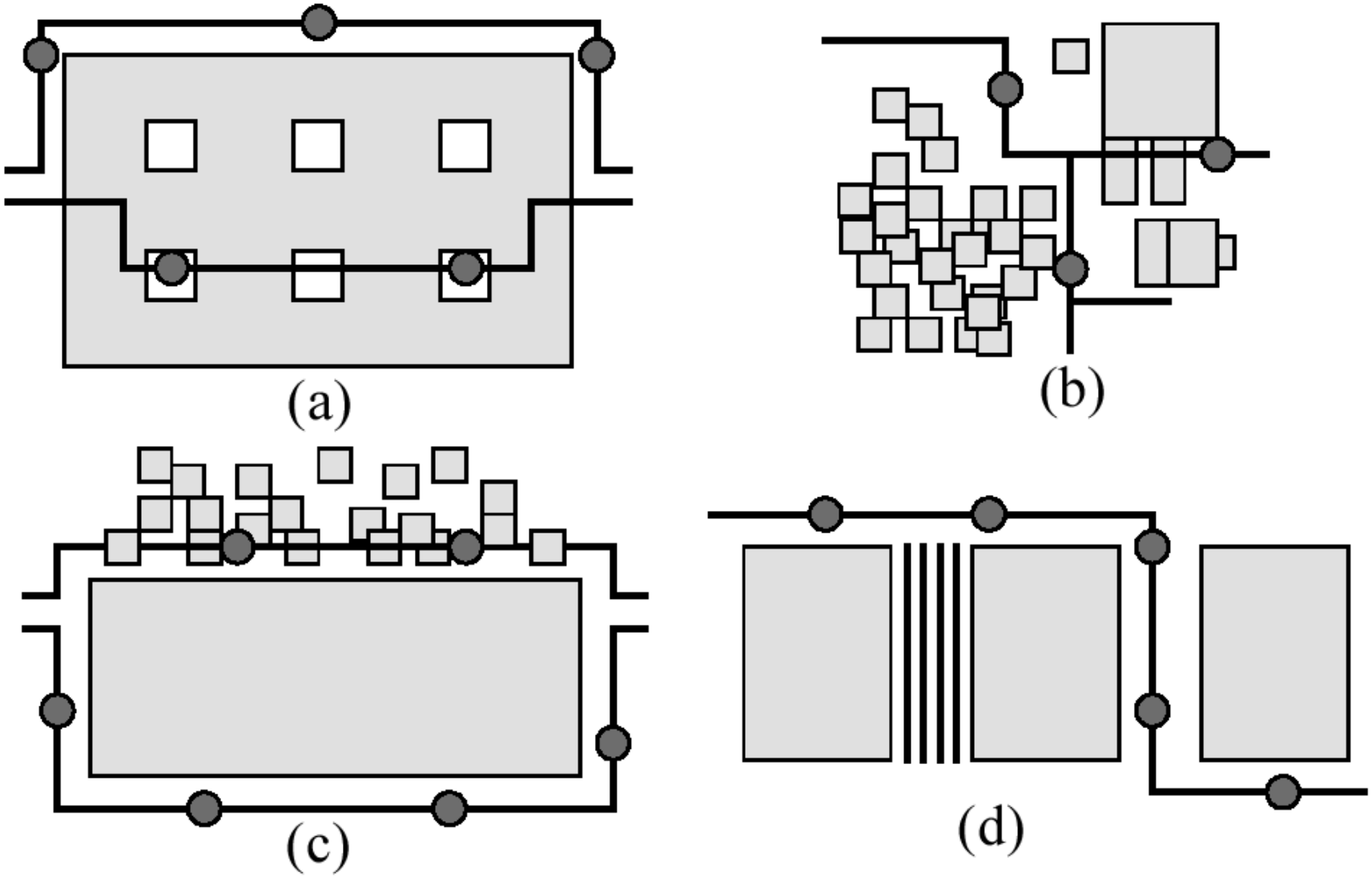
- Van Ginneken: prune on three variables
  - Delay/slack
  - Capacitance
  - Power/Area
- What if you removed delay
  - Obtain minimum area solution such that slew constraints are satisfied
  - Order of magnitude faster than Van Ginneken
  - Linear “length based” in case of 1 buffer



Chuck's "Buffer Tree Philosophy"

- Buffers completely inhibit routing
- Millions of buffer trees needed, so build them fast
- Required for speed, simultaneous stuff too time consuming and not necessarily any better
- Clean up messes afterward

Environmentally Friendly "Green" Buffering



Congestion Mitigation High-level Illustration

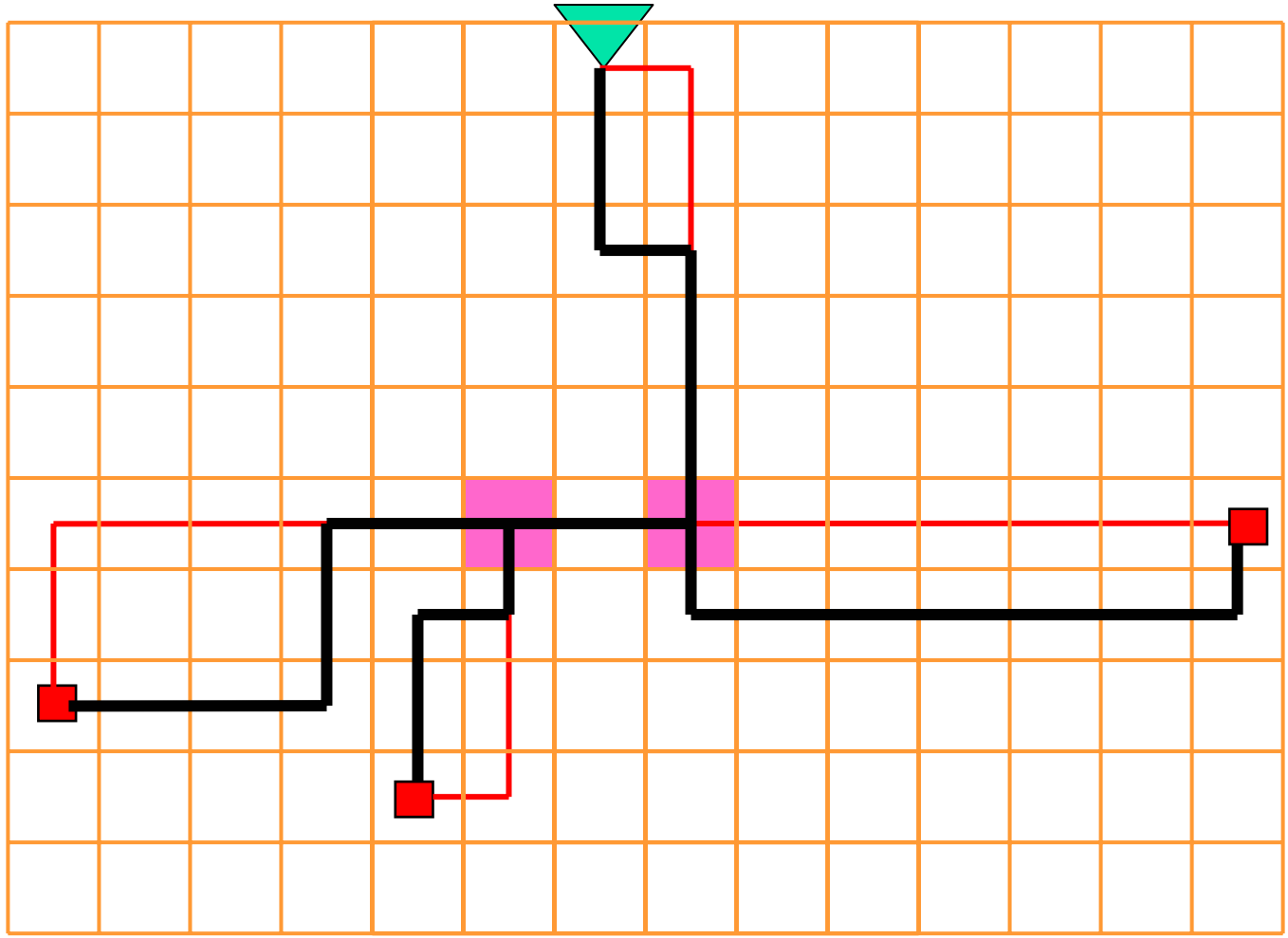
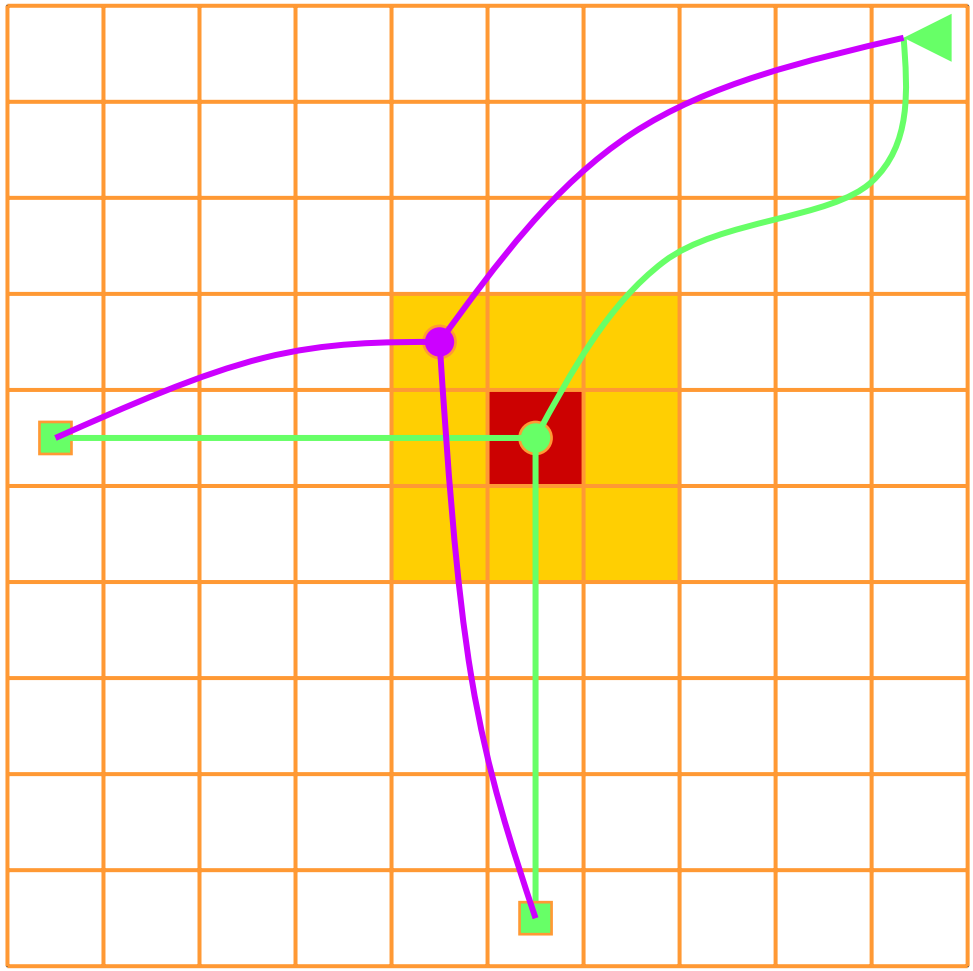
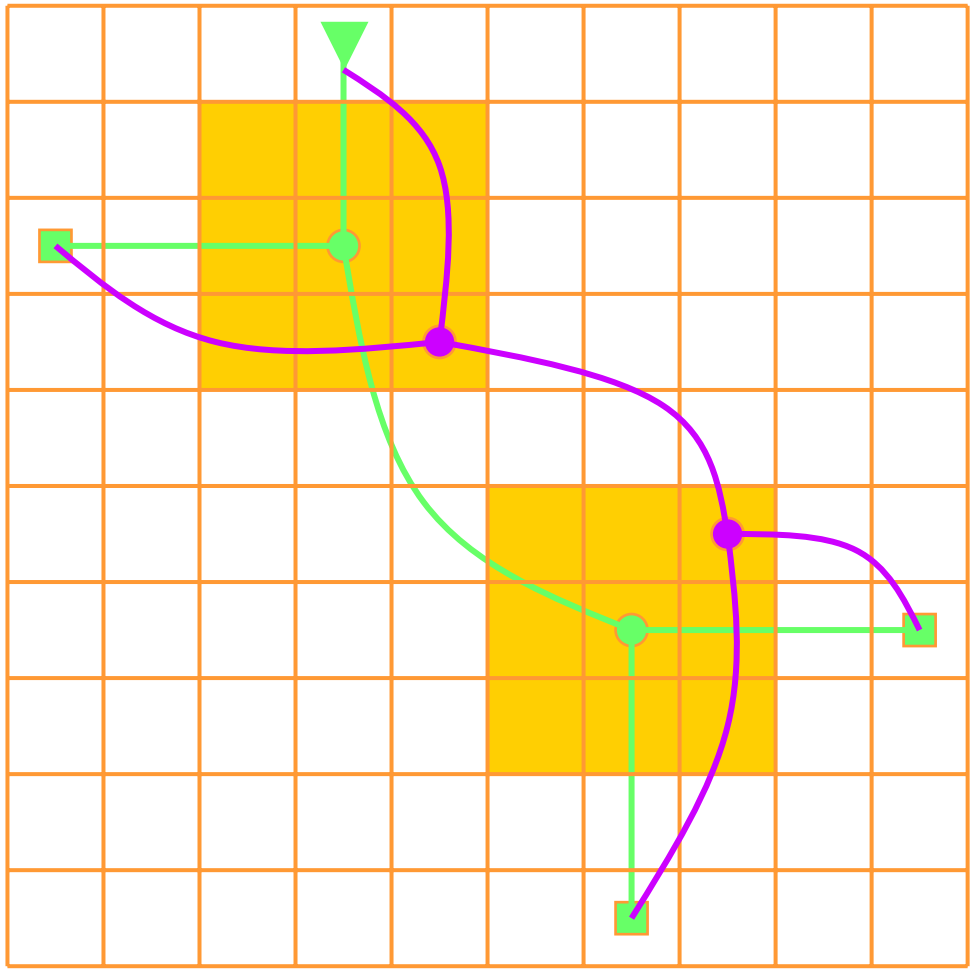


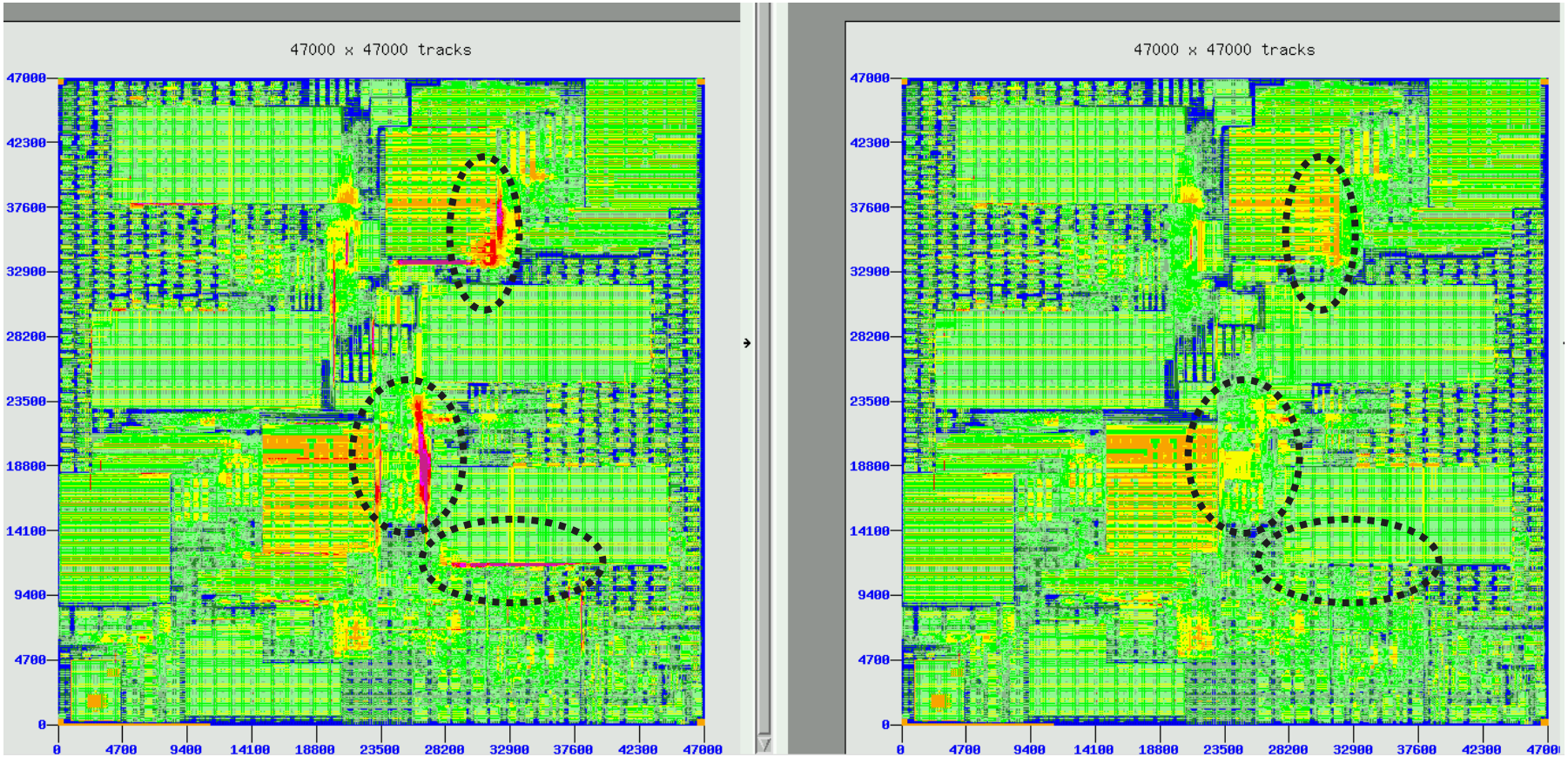
Plate Example



Dual Plate Example



Congestion Reduction Example



Final Thoughts

- On the surface, electrical correction simple
  - Right to left
  - First try repowering
  - Then buffering
- Flow decisions affect solution quality
  - Which legalization strategy
  - How tight a slew constraint
  - Saving area versus getting design in good shape