IEEE/ACM
2026 INTERNATIONAL CONFERENCE ON COMPUTER-AIDED DESIGN
ICCAD
45th Edition

CU3D: A Comprehensive Timing-driven Design Flow for Face-to-Face-Bonded 3D ICs

(Invited)



Haoyang Xu, Leilei Jin, Tsung-Yi Ho

The Chinese University of Hong Kong

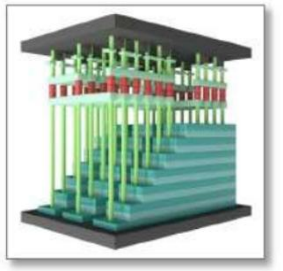
Commercial 3D Chips

Data center

Memory



SK Hynix



Yangtze



FUJITSU-MONAKA

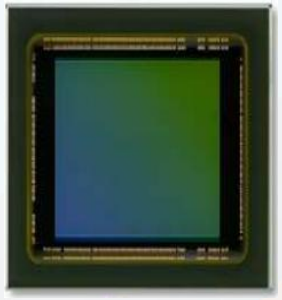
Logic



Intel



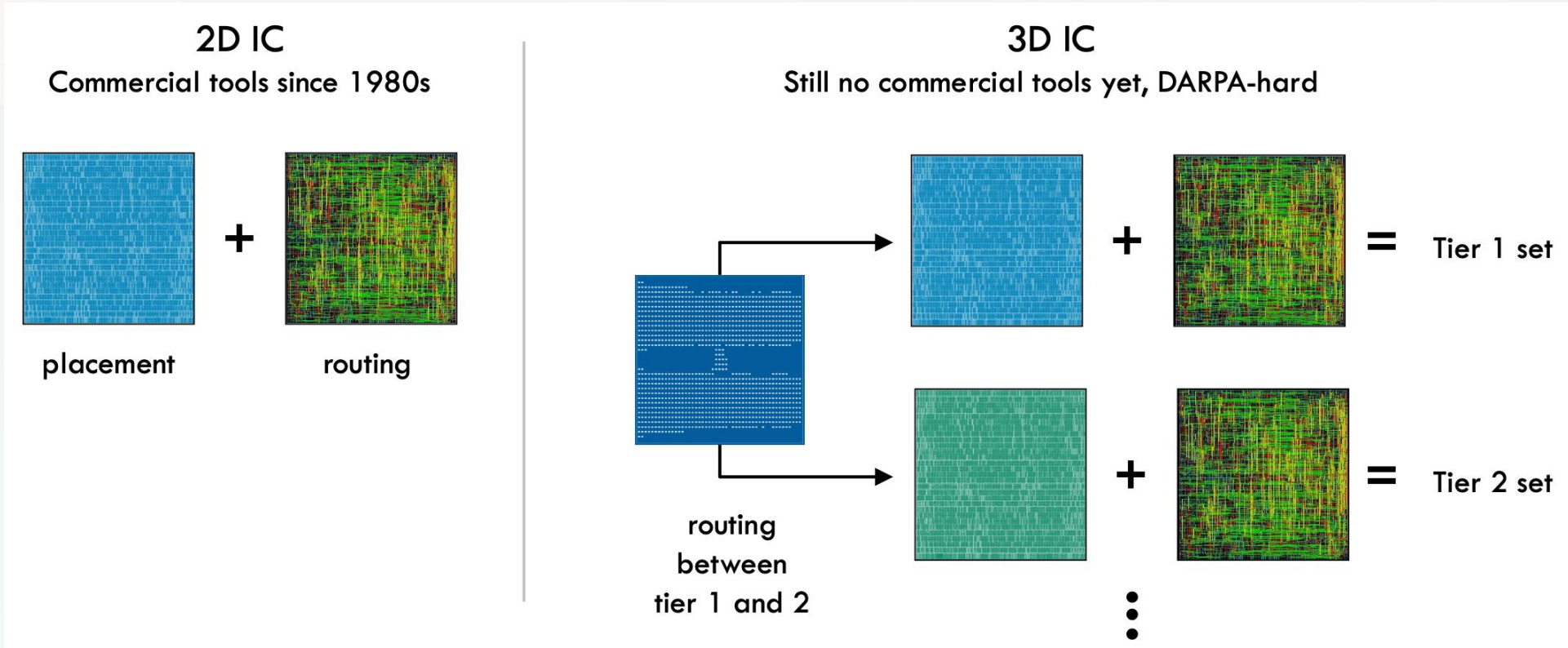
AMD



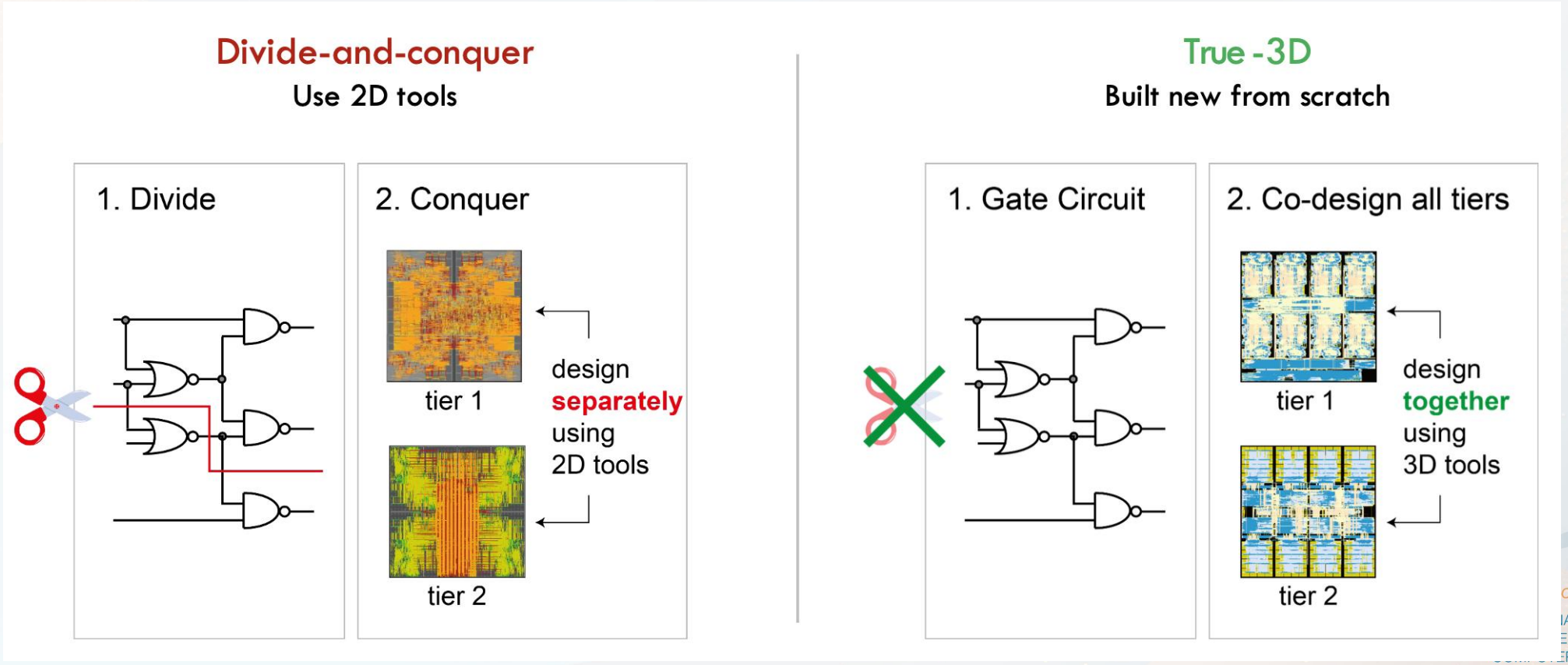
Sony

3D IC Physical Design Harder Than 2D IC

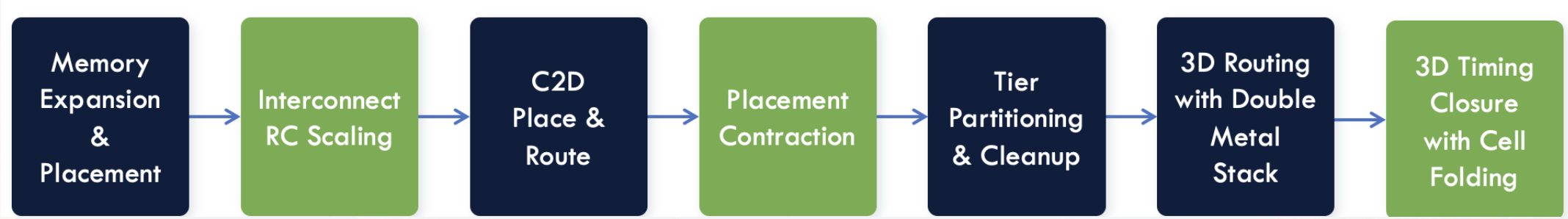
ALL Tiers Must Be Co-Optimized

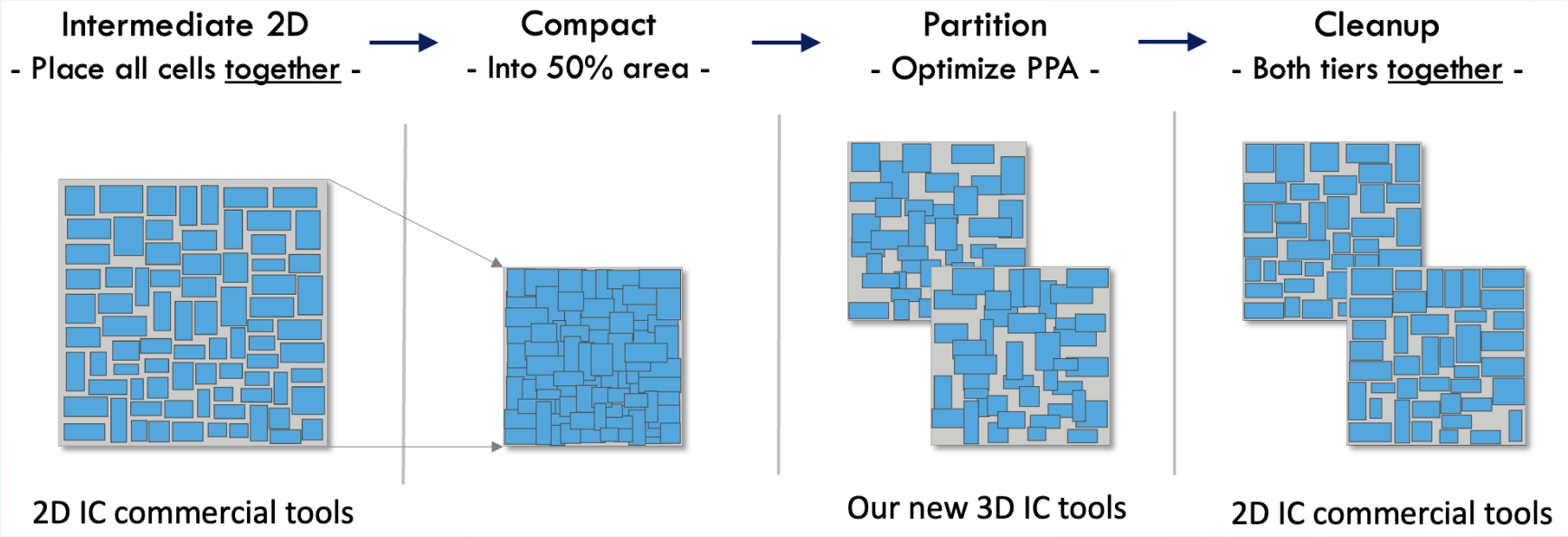


ALL Tiers Must Be Co-Optimized
Two approaches:



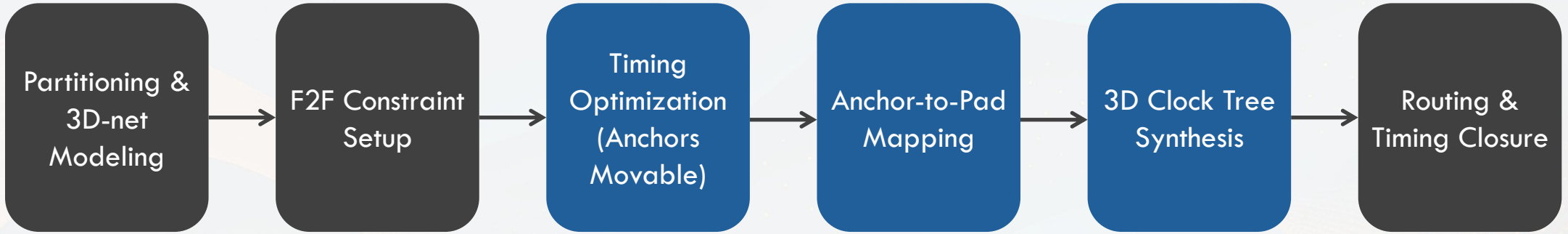
Prior Works — Compact2D [1]

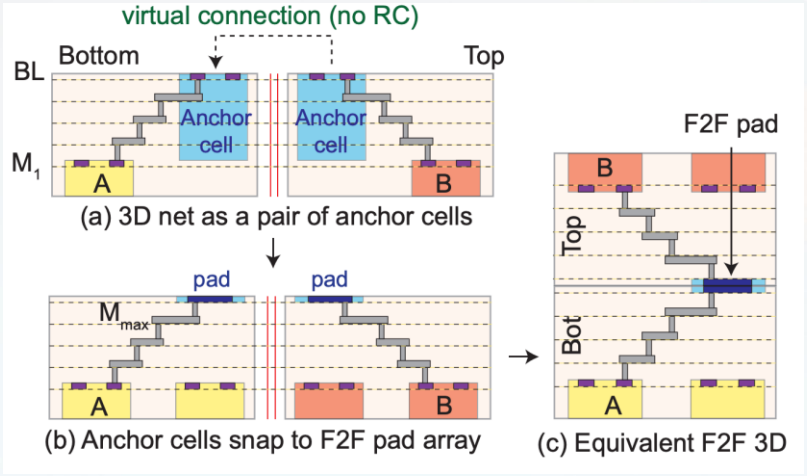




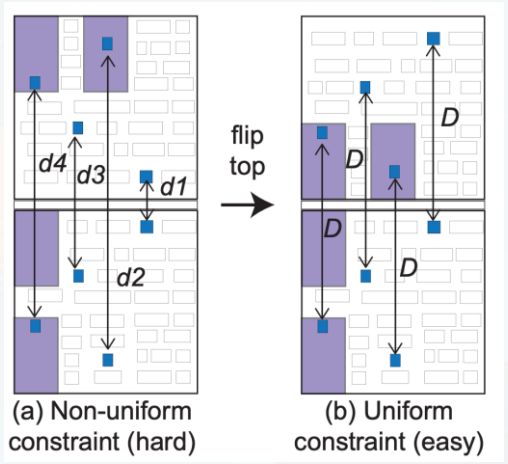
[1] B. Ku, et al., "Compact-2D: A Physical Design Methodology to Build Two-Tier Gate-level 3D ICs," TCAD, 2019.

Prior Works — AnchorGrid [2]



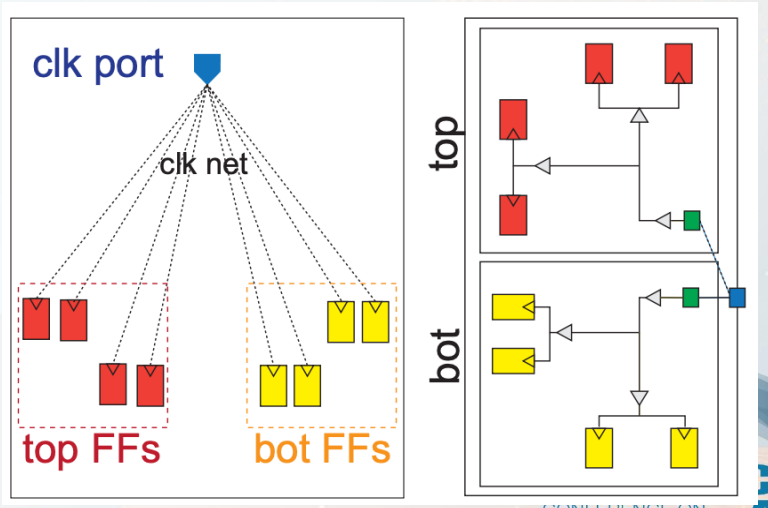


Movable Anchors → Fixed Pads



Uniform Constraint

- Fixed offset D: $\Delta y_t = \Delta y_b + D$
- Easy for placement engine

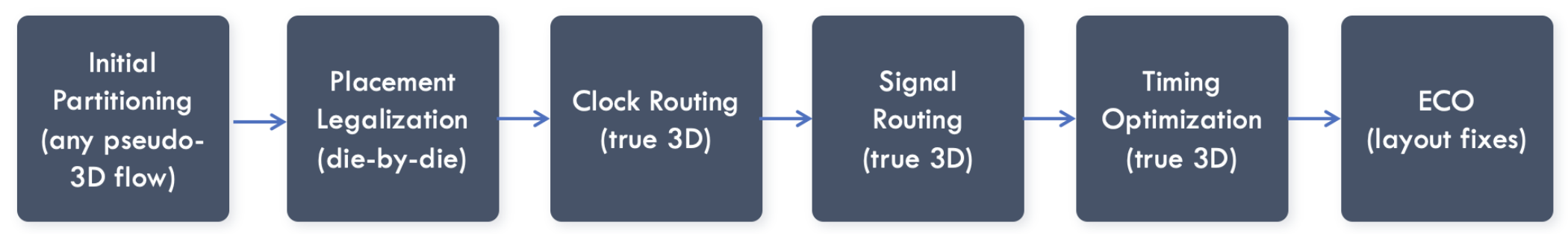


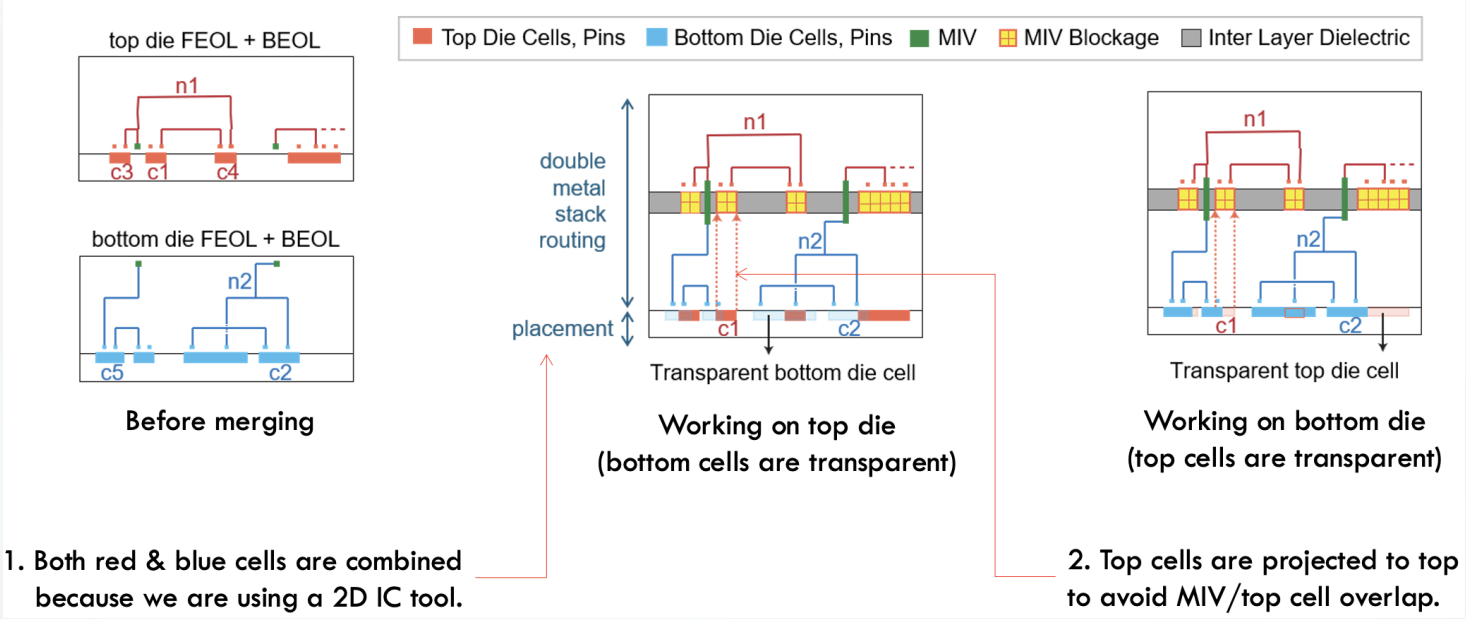
3D Clock Tree

- Balanced F2F pad insertion

[2] Vanna-Iampikul P, Yoon J, Park C, et al. Placement-aware 3D net-to-pad assignment for array-style hybrid bonding 3D ICs,ISPD2025

Prior Works — Pin-3D [3]





- Pin-3D keeps BOTH dies in the physical design database at all times, but only one is "active" at a time — the other is a Transparent Die:
- Transparent Cells (The "COVER" Trick) + Pin Projection.

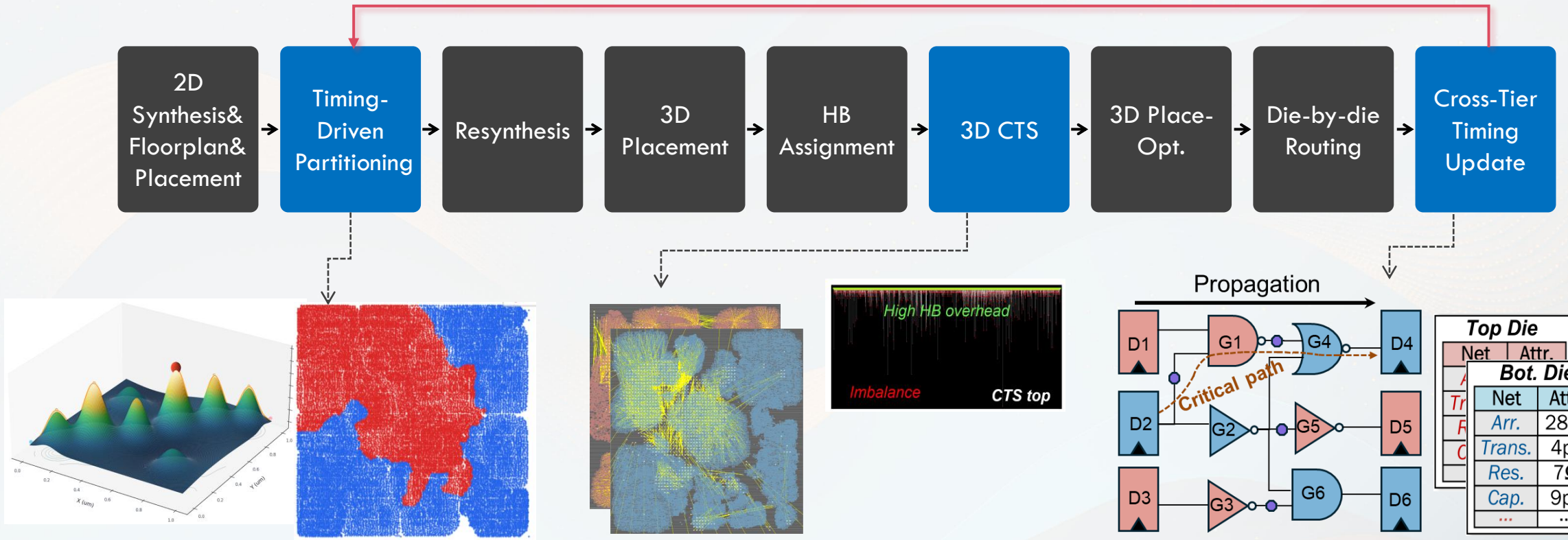
[3] Pin-3D: A physical synthesis and post-layout optimization flow for heterogeneous monolithic 3D ICs[C], ICCAD 2020.

Pseudo-3D FLOW Summary

Method	Compact2D [1]	AnchorGrid [2]	Pin3D [3]	CU3D
Key Idea	Row halving	Array-style HB assignment	Pin projection	Cross-tier abstraction
3D Stack	Two separate dies	Two separate dies	Double metal stack	Two dies with awareness
Strength	Shrinking unnecessary	Optimizing with anchor constrains	3D ECO & Timing closure	PPA-driven
Weakness	DRVs due to row halving	Poor timing closure	Ignoring HB constraints	-
Placement	2D + Tier partitioning	Multi-die co-placement	2D + Tier partitioning	Cross-tier aware placement
Clock Tree Design	2D + Tier partitioning	Die-by-die	Optimized for 3D	3D CTS co-optimized with signal
Post Route Opt.	Both dies at once	Not supported	Die-by-die	PPA abstraction
HB Constraints	Not supported	Supported	Not supported	Supported
Techfile	Hack-necessary	Homogeneous-only	Hack-necessary	Independent without hacking

[1] B. Ku, et al., "Compact-2D: A Physical Design Methodology to Build Two-Tier Gate-level 3D ICs," TCAD, 2019.
[2] Vanna-Iampikul P, Yoon J, Park C, et al. Placement-aware 3D net-to-pad assignment for array-style hybrid bonding 3D ICs,ISPD2025
[3] Pin-3D: A physical synthesis and post-layout optimization flow for heterogeneous monolithic 3D ICs[C], ICCAD 2020.

CU3D: 3D IC Physical Design Flow



IEEE/ACM
2026 INTERNATIONAL CONFERENCE ON COMPUTER-AIDED DESIGN
ICCAD
45th Edition

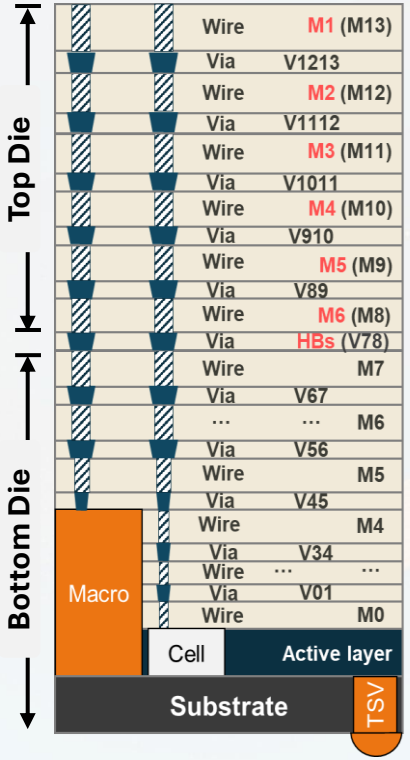
Native Support for Heterogeneous Integration

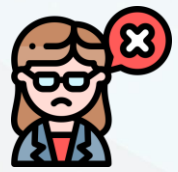
Single Die Solution

Technology adaptation for single die design flow
Single 3D LEF
Top Die LEF + Bot Die LEF

Predictive PDK only
Single 3D Lib
Top Die Lib + Bot Die Lib

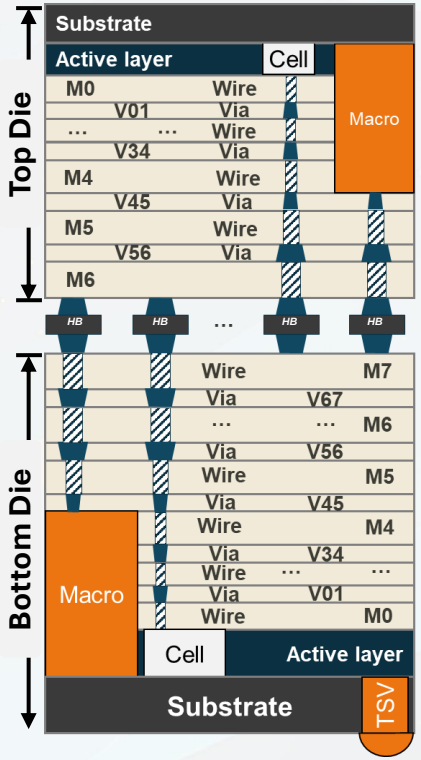
Hard for heterogeneity
Single 3D QRC
Top Die QRC + Bot Die QRC





- QRC: Encrypted Access
- Timing: RC Scaling Mismatch
- Heter. PDK: Hard to access

Multi-Die Solution



No Technology Adaptation
Top Die LEF
Bot Die LEF

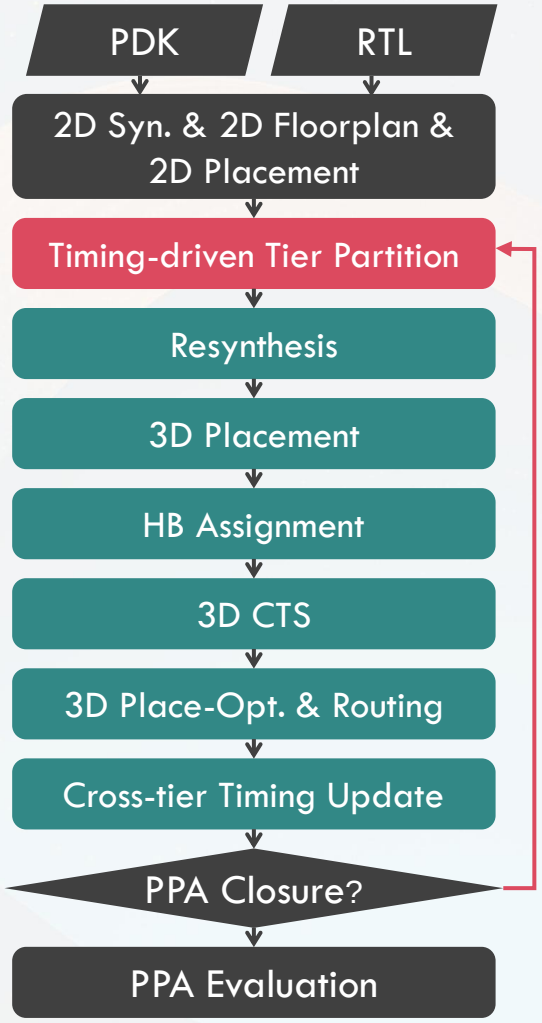
Support Commercial PDK
Top Die Lib
Bot Die Lib

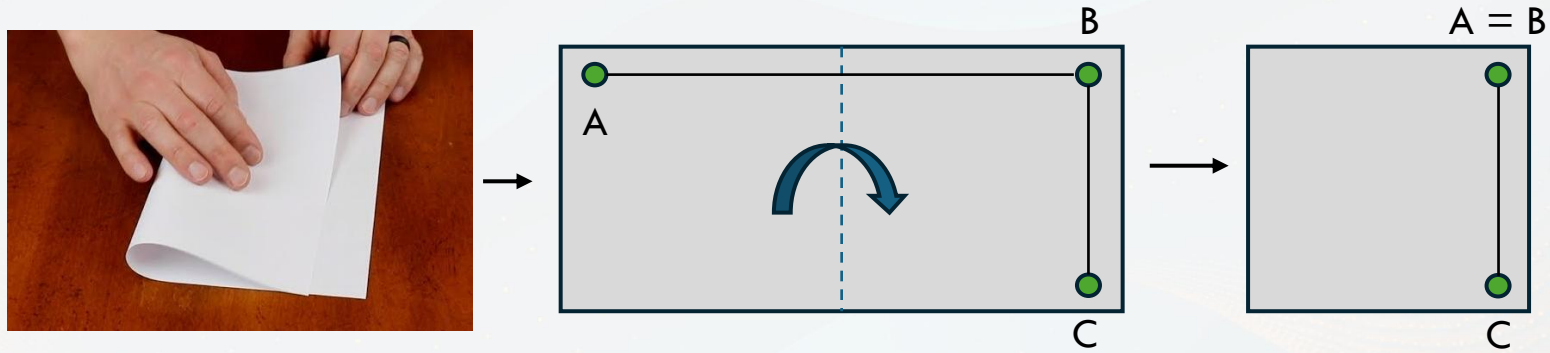
Heterogeneity-Friendly
Top Die QRC
Bot Die QRC



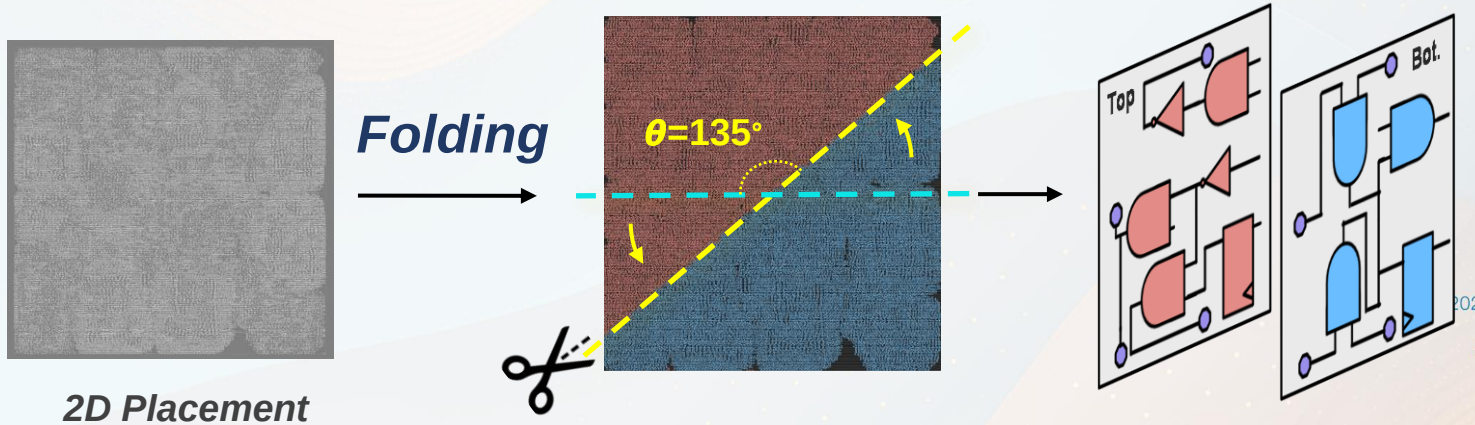
- QRC: No Change or Rebuild
- Timing: SDC Calibration
- Heter. PDK: No PDK Change

Advantage I: Timing-Driven 3D IC Partitioning





Leverage the optimization results from 2D placement



2D Placement

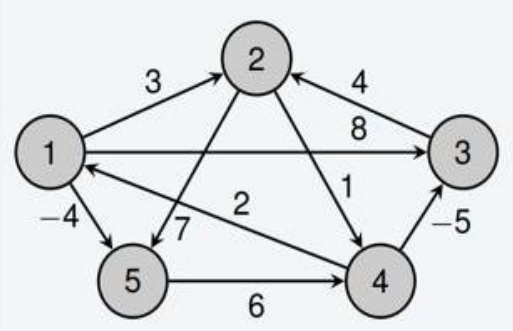
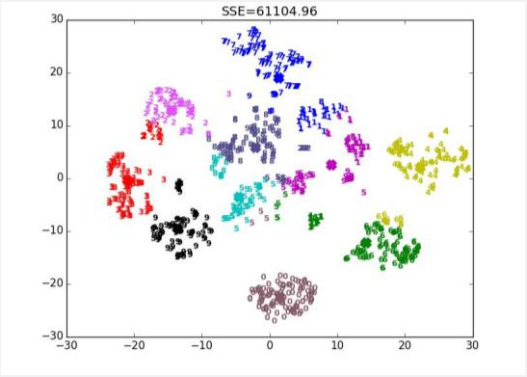
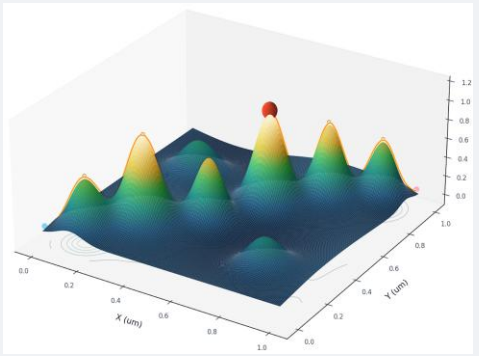
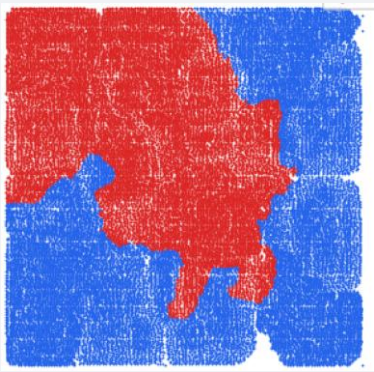
Why Timing-Driven 3D IC Partitioning Works

01 / Register Delay Graph Construction: Model timing relationships between registers using a delay graph, with criticality-weighted delays guiding the potential field.

02 / Combinational Logic Clustering: Assign each combinational cell to its nearest register in the XY plane, forming register-anchored clusters while preserving placement locality.

03 / One-Dimensional Potential Equilibration: Compute an auxiliary scalar potential for each register using delay-driven forces, soft repulsion, and confinement, while keeping XY coordinates fixed.

04 / Spatially Connected Tier Partitioning: Smooth the potential field over a spatial k-nearest-neighbor graph, grow two connected regions from its extrema, and refine the boundary to reduce HB usage while maintaining area balance.

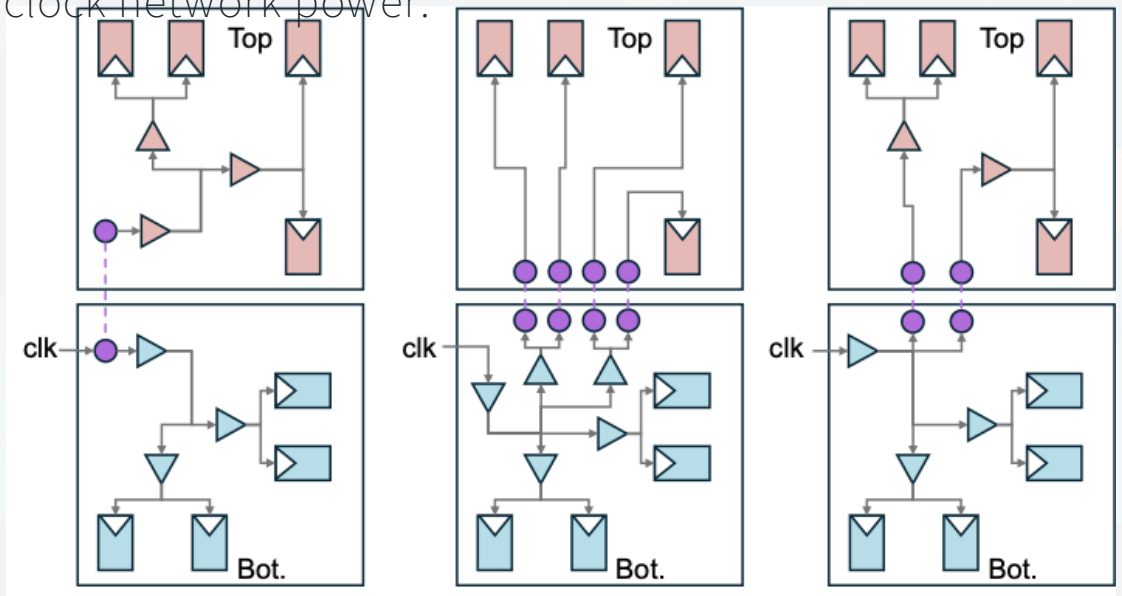
Graph Visualization Insight	*Spatial Clustering Visualization*	*Potential Energy Landscape*	*Smooth Boundary Visualization*
			

Advantage II: Flexible Clock Tree Synthesis (CTS)

Problem: Pin3D forces all clock buffers to a single tier, which results in a unbalanced clock tree structure and introduces significant clock skew.

CU3D Solution: Latency-Aware 3D CTS — Decouple clock trees across tiers while co-optimizing them with precise latency constraints for global balance.

Method & Result: Use a tier-balanced, distributed approach: cluster top-tier sinks to build a balanced subtree, then use its latency as a hard constraint for bottom-tier CTS, achieving drastically lower clock skew and reduced clock network power.

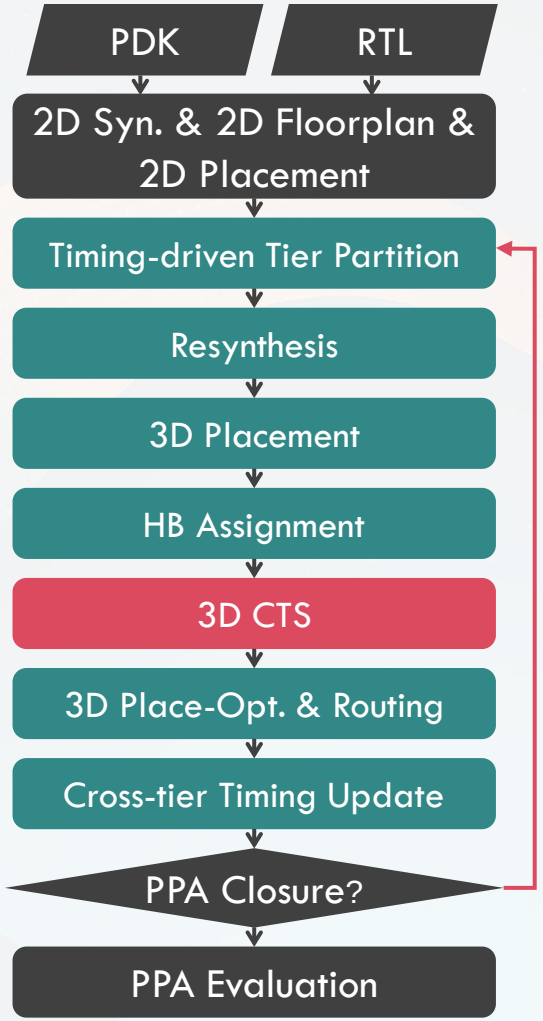


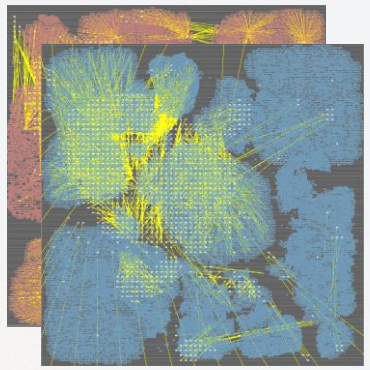
(a) AnchorGrid [2] (b) Pin3D [3] (c) CU3D CTS

CTS Strategy Comparison:

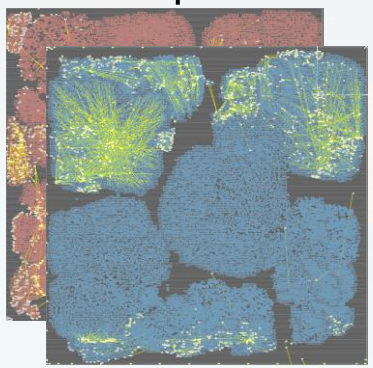
The figure illustrates three 3D CTS strategies. CU3D's 3D CTS (c) achieves a more balanced tree than Cascade-2D (a) and Pin3D (b) by distributing buffers across tiers and respecting inter-tier latency, eliminating bottlenecks that cause skew.

How Latency-Aware 3D CTS Works



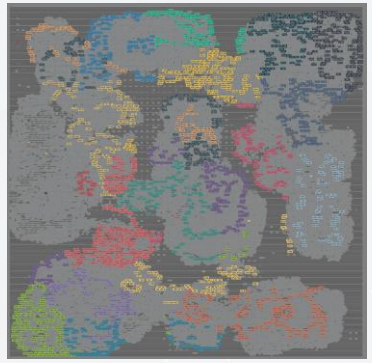


Assigned HBs



Bridge cells

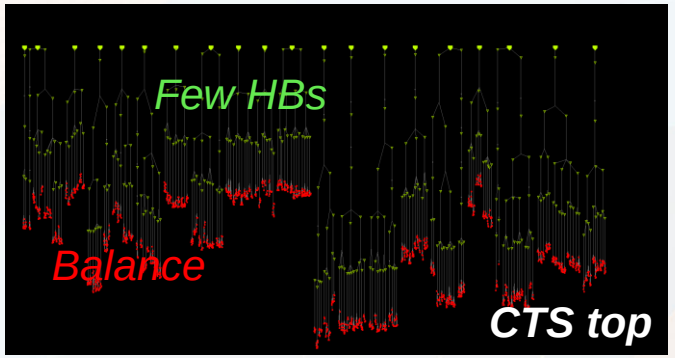
HB Assignment

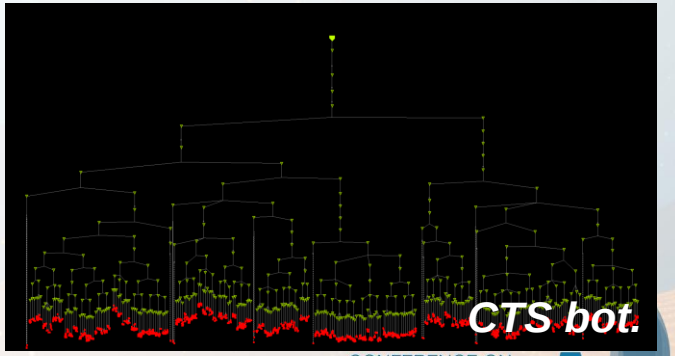


Top clock group clustering

- 01. Top-Tier Clustering
- 02. Top-Tier CTS
- 03. Latency Constraint Extraction
- 04. Bottom-Tier CTS

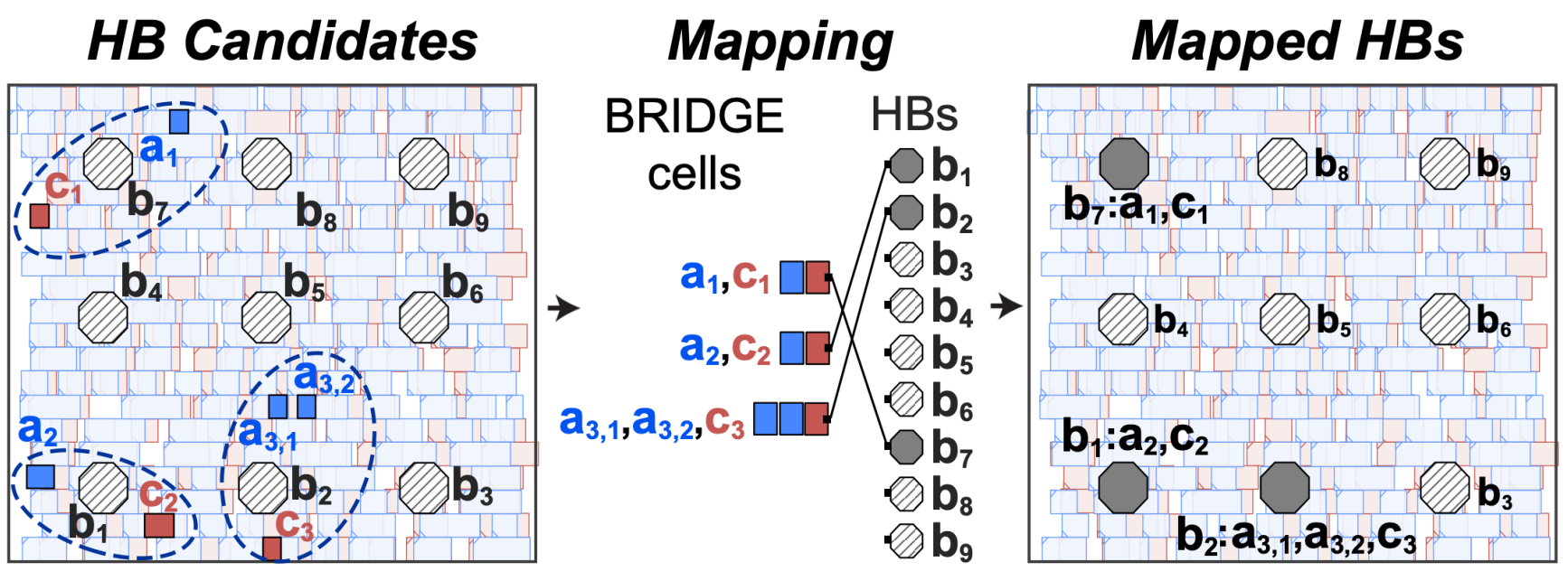
Latency-aware 3D CTS



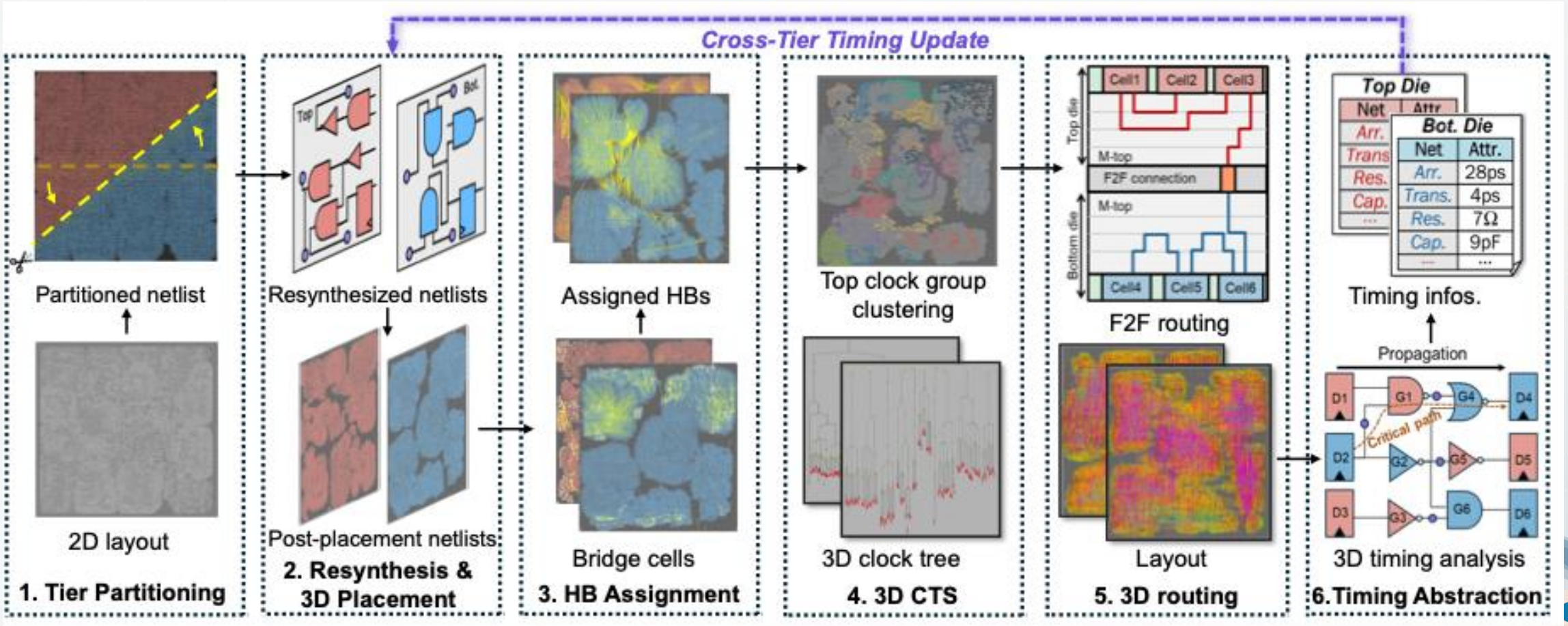


Hybird Bond Allocation Strategy

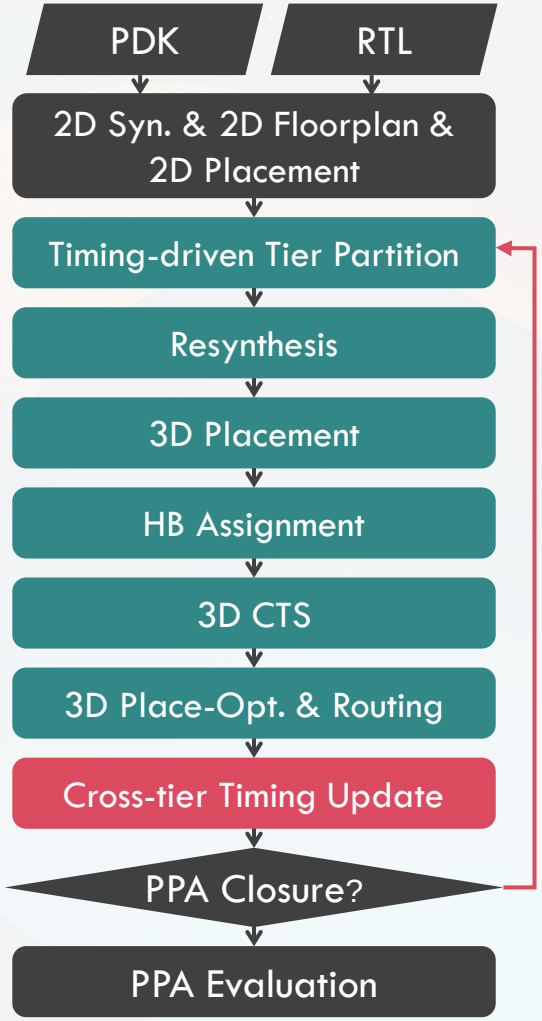
- Signal-first HB Allocation
- Signal nets are assigned via bipartite matching.
- Clocks consume leftover HBs.



Advantage III: Cross-Die Timing Propagation



How Cross-Tier Timing Propagation Works



01. BRIDGE Cell Modeling: Insert **BRIDGE_OUT** and **BRIDGE_IN** cells at inter-tier net boundaries.

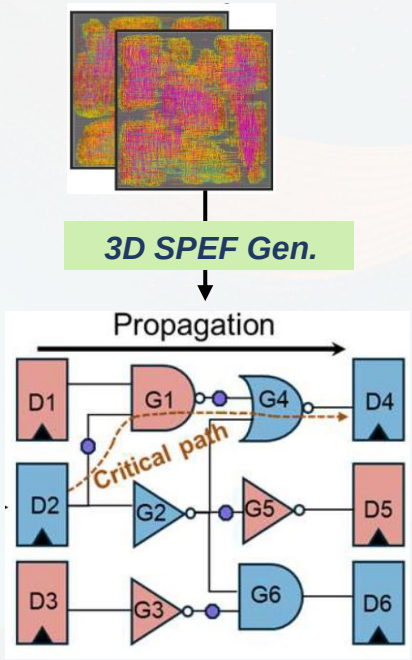
02. Unified Parasitic Merge: Merge **SPEF files** from all tiers into a single 3D parasitic database.

03. SDC Constraint Generation: Run STA on the **merged 3D spef** to compute accurate timing.

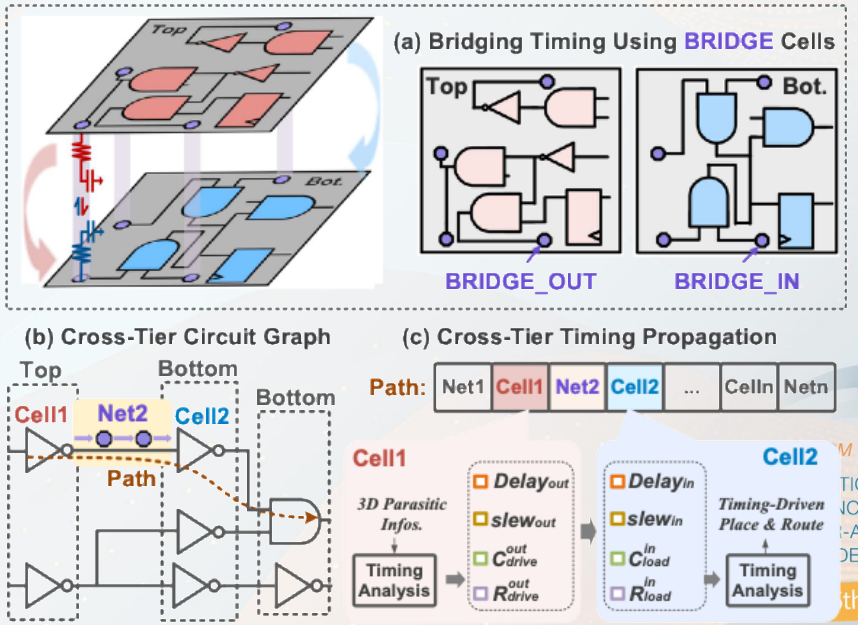
04. SDC Write-back: Convert calculated timing into **standard SDC constraints** and apply to each die.

05. Iterative Optimization: Refine each die under updated cross-tier timing constraints and repeat the analysis–optimization cycle.

3D Routing & Timing Analysis



Cross-die Timing Constraints



Why Cross-Tier Timing Propagation Works

01. Merged Parasitic Extraction:

- Combine post-route SPEF files from both dies with explicit HB RC models at interfaces to create a unified parasitic database.

02. Cross-Tier Timing Analysis:

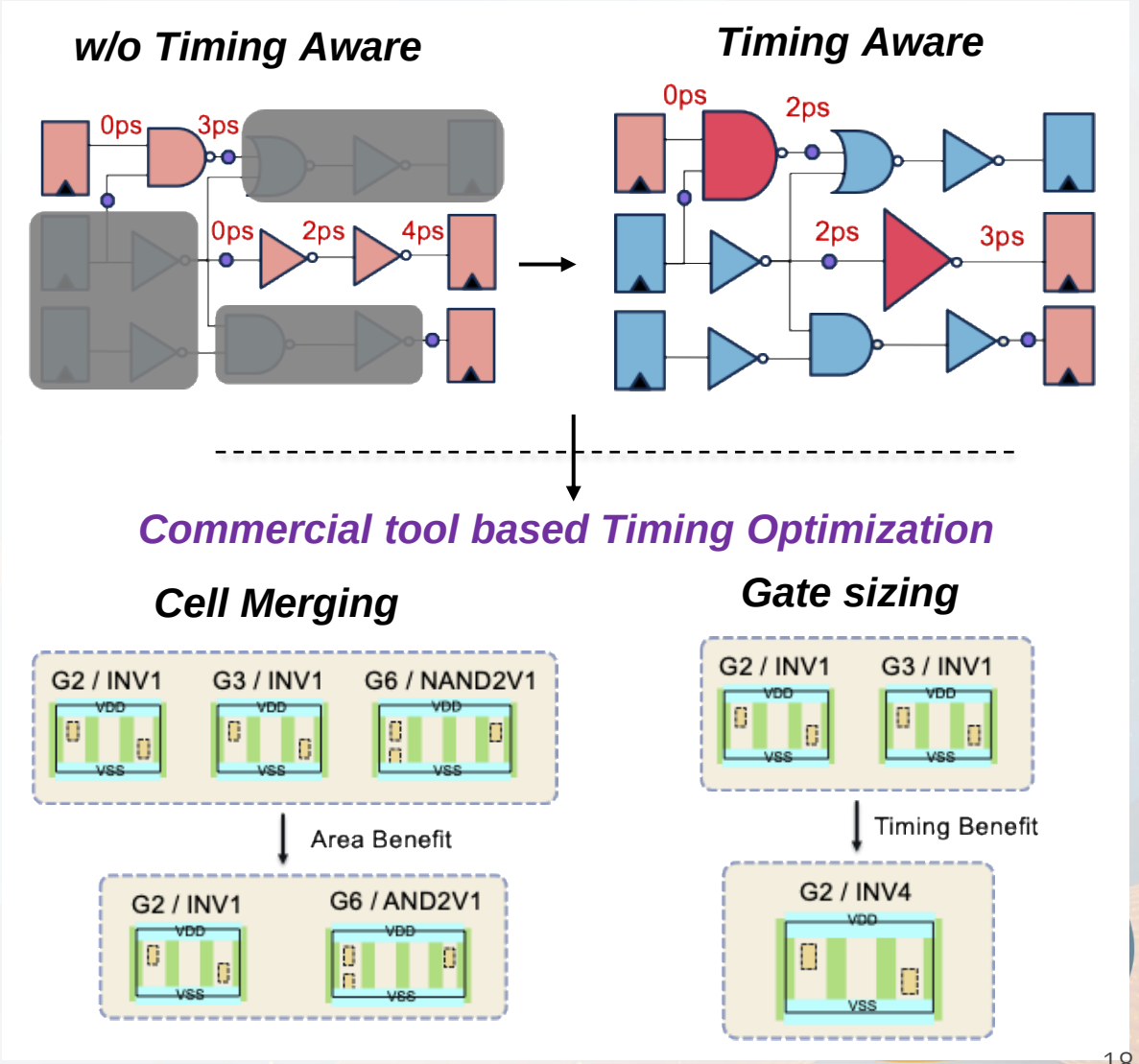
- Perform STA on the merged 3D design to extract accurate arrival times, required times, slew rates, and parasitic loads at all BRIDGE interface pins.

03. SDC Constraint Generation:

- Convert extracted timing quantities to standard SDC constraints for each die's BRIDGE cells.

04. Iterative Timing-Driven Optimization:

- Apply updated SDC constraints to each die, perform timing-driven placement and routing optimization, and feed the results into the next round.



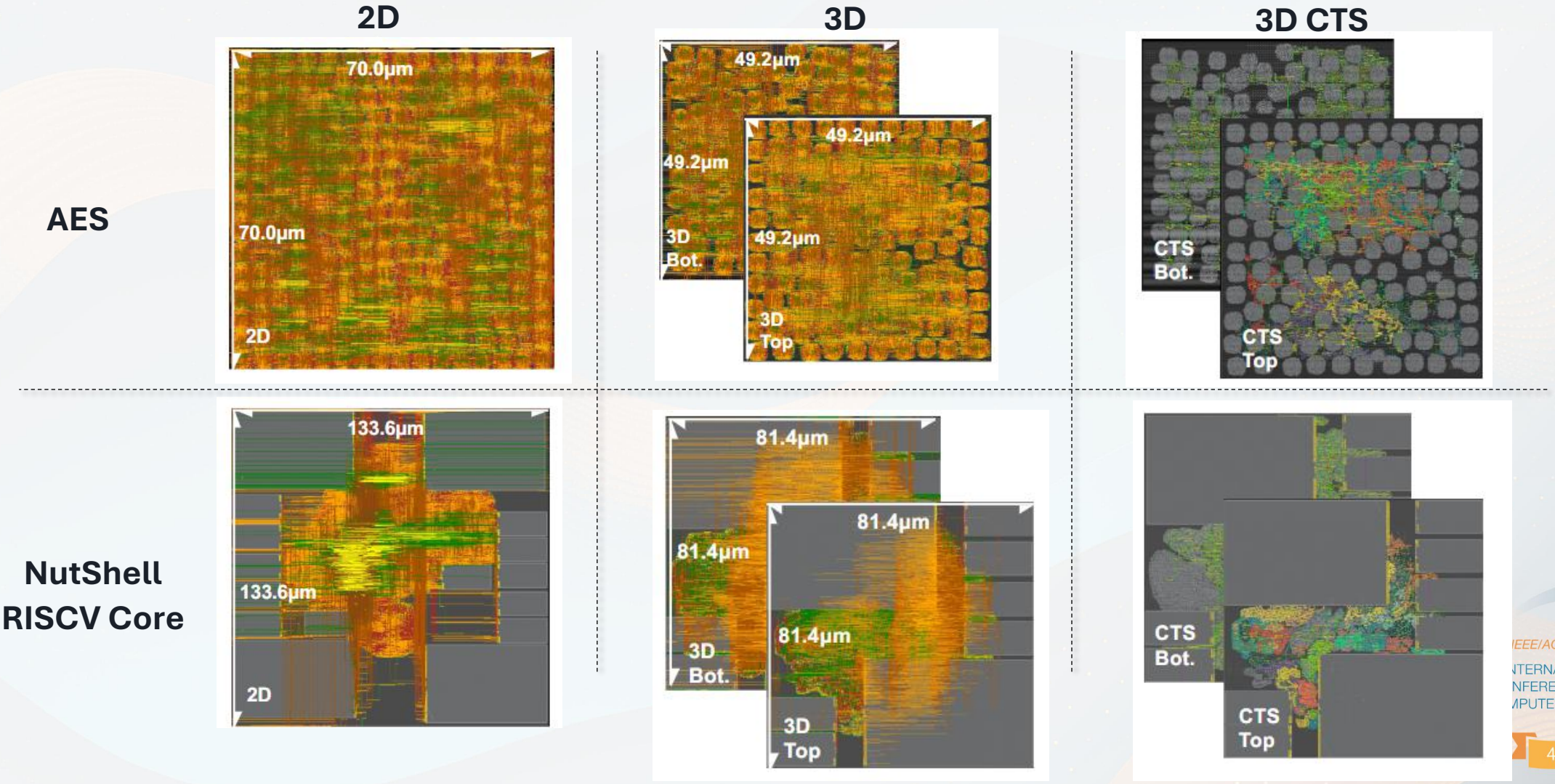
Benchmark Results: Timing Improvement

Circuit	Scale	WNS Improvement	TNS Improvement
AES	141k cells, 10k FFs	+10%	+5%
wb_conmax	18k cells, 770 FFs	+49%	+42%
JPEG	259k cells, 37k FFs	+42%	+69%
b18	47k cells, 4.8k FFs	+19%	+28%
NutShell	80k cells, 20k FFs	+47.1%	+75.4%

*Key Findings: Cross-tier timing optimization improves **WNS by 33.4%** and **TNS by 43.9%** on average across all five benchmarks, proving CU3D's effectiveness in handling complex timing paths across 3D tiers.*

LAYOUT COMPARISON

GDS Layouts: 2D vs Logic Folding CU3D



PPA ANALYSIS · COMPARISON TABLE

Overall PPA Comparison: CU3D vs. Pin3D vs. 2D

Based on AES (3.3GHz) design metrics

The framework supports iterative refinement. In this work, we demonstrate the timing improvements obtained over two optimization rounds, rather than convergence to a violation-free design.

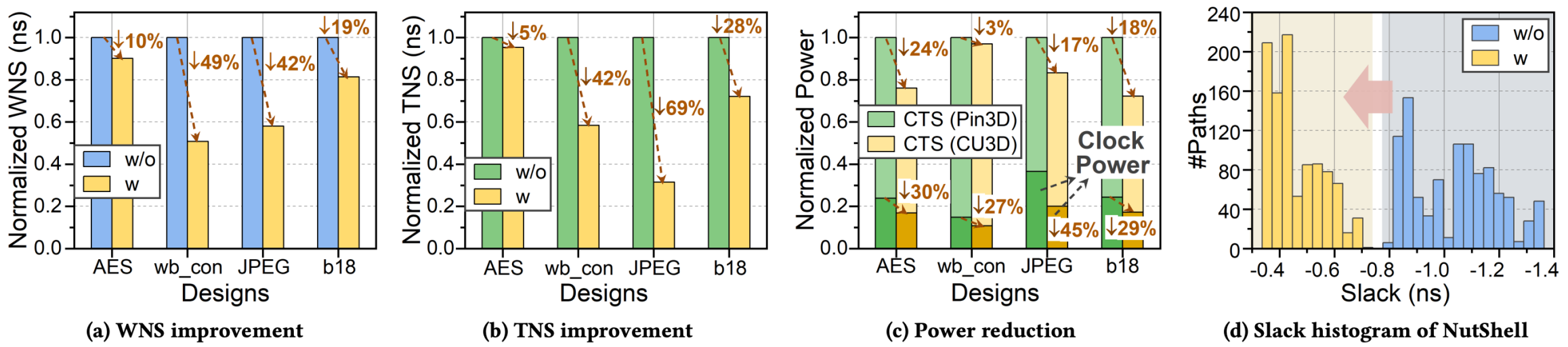
Metric	2D	Pin3D	CU3D	Improvement (vs Pin3D)
Effective Freq. (GHz)	2.114	2.045	**2.146**	**+4.9%**
# Hybrid Bonds	-	8742	**2348**	**-73.1%**
TNS (ns)	-176.72	-190.66	**-106.57**	**+44.1%**
Total Power (mW)	144.9	171.8	**130.6**	**-24.0%**
Footprint (μm^2)	4900	2500	**2401**	**-4.0%**

KEY TAKEAWAY: CU3D delivers a comprehensive PPA win over Pin3D, with a 73.1% reduction in hybrid bonds, a 44.1% TNS improvement, and better results across frequency, power, and area.

Detail PPA Comparison

Based on Nutshell (target=1GHz) design metrics

The framework supports iterative refinement. In this work, we demonstrate the timing improvements obtained over two optimization rounds, rather than convergence to a violation-free design.



KEY TAKEAWAY: CU3D delivers a comprehensive PPA win over the traditional 2D flow on the NutShell RISCV CPU, achieving a +14.3% frequency boost, -43.5% power reduction, and -62.9% footprint shrinkage.

Conclusion

01 / A Comprehensive Timing-Driven 3D Flow

- Integrate partitioning, placement, clock tree synthesis, and routing using commercial 2D EDA tools without modifying technology files.

02 / Timing-Aware Partitioning and Latency-Aware CTS

- Preserve placement locality through timing-driven tier partitioning and coordinate per-die clock trees through latency constraints.

03 / Effective Cross-Tier Timing Optimization

- Enable coordinated optimization through iterative timing constraint exchange, improving WNS and TNS by **33.4% and 43.9% on average** across five benchmarks compared with the flow without cross-tier timing updates.

04 / Improved Power and Timing Quality

- Reduce total power and TNS on all four logic benchmarks compared with Pin-3D, demonstrating the benefits of cross-tier-aware physical design.

IEEE/ACM

2026 INTERNATIONAL CONFERENCE ON COMPUTER-AIDED DESIGN

ICCAD

45th Edition

Thanks



Leilei Jin

Doctor