

香港中文大學
The Chinese University of Hong Kong

3D-IC Physical Design

Bei Yu
Department of Computer Science & Engineering
Chinese University of Hong Kong
byu@cse.cuhk.edu.hk

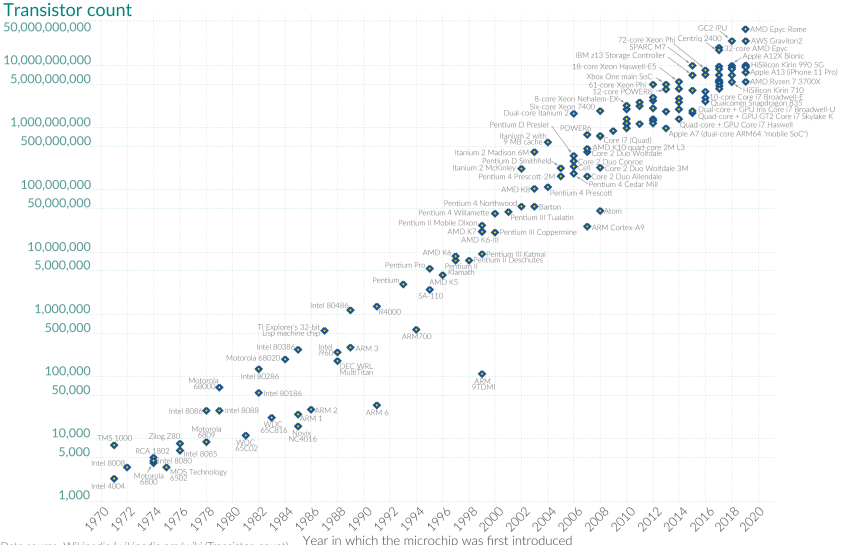
September 1, 2026

Cramming More Components onto Integrated Circuits

Moore's Law: The number of transistors on microchips doubles every two years

Our World in Data

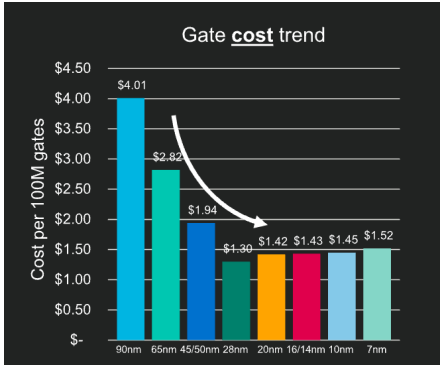
Moore's law describes the empirical regularity that the number of transistors on integrated circuits doubles approximately every two years. This advancement is important for other aspects of technological progress in computing – such as processing speed or the price of computers.



Data source: Wikipedia (wikipedia.org/wiki/Transistor_count)
OurWorldinData.org – Research and data to make progress against the world's largest problems.

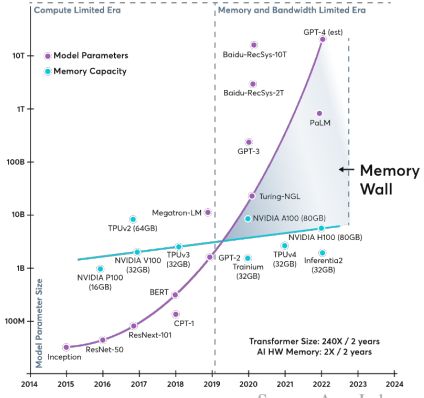
Licensed under CC-BY by the authors Hannah Ritchie and Max Roser.

No Longer the Best Way to Scale



Source: Marvell

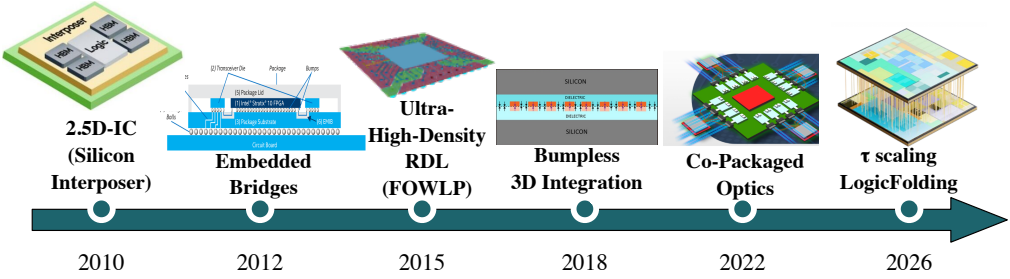
Diminishing Economic Advantages



Source: Ayar Labs

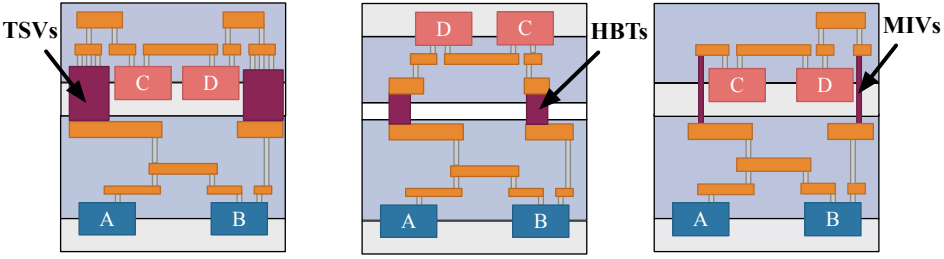
Memory Wall

More-than-Moore Heterogeneous Integration



- Heterogeneous integration (HI) is a promising solution for better PPA, cost, and memory bandwidth.
- 3D integration is a key technology for HI to achieve Moore style gains.

3D Integrated Circuits



TSV based

- Commercially mature
- High area overhead
- High RC parasitics
- Pitch: > 10 um

Hybrid bonding

- Low fabrication cost
- No area overhead
- Coupling issues
- Pitch: 1 ~ 10 um

Monolithic

- High fabrication cost
- High integration density
- Small RC parasitics
- Pitch: ~ 0.1 um

In-House 3D Engine vs Commercial Tool Plus

- **In-house 3D Engine (自主 3D 引擎)**: Native 3D-aware algorithms designed by researchers [TCAD'13][1][TCAD'13][2][ISPD'16][3][ISPD'24][4], enabling exploration of a larger 3D solution space with full control over implementation; currently realized mainly as point-tools.
- **Commercial Tool Plus (复用商业工具)**: Uses mature 2D commercial P&R/STA/sign-off engines augmented by customized scripts to support 3D flow [TCAD'17][5][ISPD'18][6][ICCAD'20][7]; inserts 3D-specific stages (e.g., tier assignment) atop 2D-centric kernels, which may constrain the solution space.

---

[1] **Guojie Luo, Yiyu Shi, and Jason Cong (2013). "An analytical placement framework for 3-D ICs and its extension on thermal awareness".** In: *IEEE TCAD* 32.4, pp. 510–523.

[2] **Meng-Kai Hsu, Valeriy Balabanov, and Yao-Wen Chang (2013). "TSV-aware analytical placement for 3-D IC designs based on a novel weighted-average wirelength model".** In: *IEEE TCAD* 32.4, pp. 497–509.

[3] **Jingwei Lu et al. (2016). "ePlace-3D: Electrostatics based placement for 3D-ICs".** In: *Proc. ISPD*, pp. 11–18.

[4] **Siting Liu, Jiaxi Jiang, et al. (2024). "Routing-aware legal hybrid bonding terminal assignment for 3D face-to-face stacked ics".** In: *Proc. ISPD*, pp. 75–82.

[5] **Shreepad Panth et al. (2017). "Shrunk-2-D: A physical design methodology to build commercial-quality monolithic 3-D ICs".** In: *IEEE TCAD* 36.10, pp. 1716–1724.

[6] **Bon Woong Ku, Kyungwook Chang, and Sung Kyu Lim (2018). "Compact-2D: A physical design methodology to build commercial-quality face-to-face-bonded 3D ICs".** In: *Proc. ISPD*, pp. 90–97.

[7] **Sai Surya Kiran Pentapati et al. (2020). "Pin-3D: A physical synthesis and post-layout optimization flow for heterogeneous monolithic 3D ICs".** In: *Proc. ICCAD*, pp. 1–9.

In-House 3D Engine vs Commercial Tool Plus

In-house 3D Engine (自主 3D 引擎)

- ☺ Full algorithmic control; easy to encode 3D objectives.
- ☺ Rapid research prototyping; transparent internals.
- ☹ Point-tool maturity; lacks end-to-end closure and sign-off.
- ☹ Runtime/robustness and capacity gaps vs. industrial tools.

In-House 3D Engine vs Commercial Tool Plus

In-house 3D Engine (自主 3D 引擎)

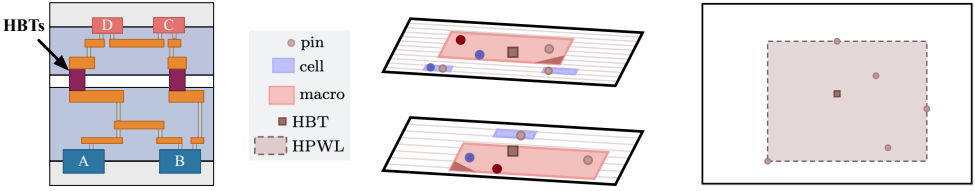
- ☺ Full algorithmic control; easy to encode 3D objectives.
- ☺ Rapid research prototyping; transparent internals.
- ☹ Point-tool maturity; lacks end-to-end closure and sign-off.
- ☹ Runtime/robustness and capacity gaps vs. industrial tools.

Commercial Tool Plus (复用商业工具)

- ☺ Mature optimization stack with sign-off correlation and scalability.
- ☺ Faster path to production-like flows using existing CAD tools.
- ☹ 2D-centric optimizations; added 3D stages may constrain solution space.
- ☹ Limited access to internal knobs; API/scripting constraints.

Our In-House 3D Engine

3D Mixed-Size Placement Problem [ICCAD'23][8]

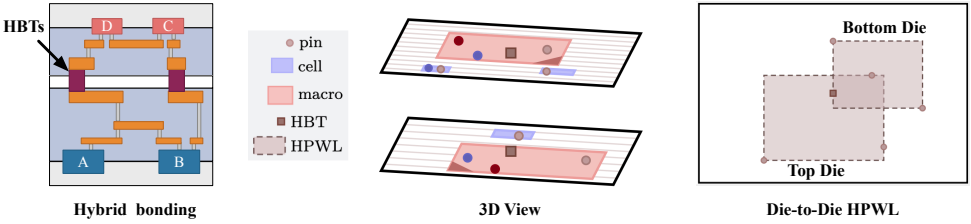


- All cells and macros on both dies should be placed at legal positions. Macros can be rotated with 0°, 90°, 180°, and 270°.
- For any crossing-die net, one hybrid bonding terminal (HBT) is inserted.
- Conventionally, the wirelength objective of a net e is the projected HPWL:

$$W_e(\boldsymbol{x}, \boldsymbol{y}) = p_e(\boldsymbol{x}) + p_e(\boldsymbol{y}), \qquad p_e(\boldsymbol{x}) = \max_{i \in e} x_i - \min_{i \in e} x_i, \tag{1}$$

Kai-Shun Hu et al. (2023). "2023 ICCAD CAD Contest Problem B: 3D placement with macros". In: *Proc. ICCAD*. IEEE, pp. 1–6.

3D Mixed-Size Placement Problem [ICCAD'23][8]



Hybrid bonding | 3D View | Die-to-Die HPWL

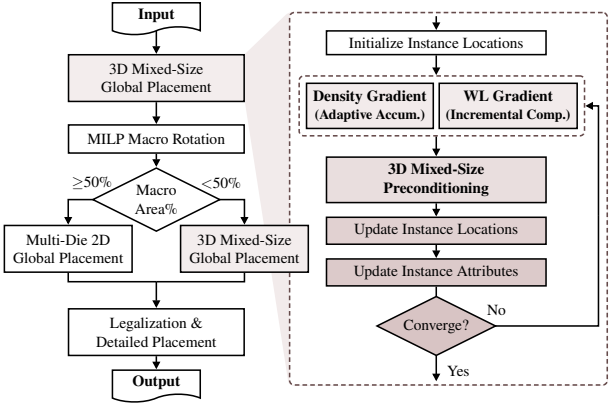
- **Die-to-Die** HPWL: the wirelength of a net e is the sum of the top net e^+ and bottom net e^- including the HBT,

$$W_e = W_{e^+} + W_{e^-}, \quad W_{e^+} = p_{e^+}(\boldsymbol{x}) + p_{e^+}(\boldsymbol{y}), \quad W_{e^-} = p_{e^-}(\boldsymbol{x}) + p_{e^-}(\boldsymbol{y}). \tag{1}$$

- **Heterogeneous** technology integration: instance layouts are different on different dies.

Kai-Shun Hu et al. (2023). "2023 ICCAD CAD Contest Problem B: 3D placement with macros". In: *Proc. ICCAD*. IEEE, pp. 1–6.

Our 3D Mixed-Size Placement Framework [TCAD'25][9]



- 3D analytical formulation:

Constrained optimization

$$\min_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{z}} \ \sum_{e\in E} W_e(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z}) + \beta \sum_{e\in E} C_e,$$
$$\text{s.t.} \quad D(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z}) \le t_d$$

Nonlinear analytical placement

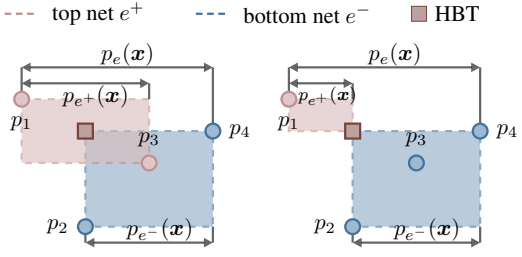
$$\min \underbrace{\sum_{e\in E} \hat{W}_e(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})}_{\text{Wirelength model}} + \lambda \underbrace{\hat{D}(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})}_{\text{Density model}}$$

- GPU-accelerated wirelength and density algorithms.

- Differentiable approximation of HPWL.
- The de facto choice: weighted-average model [TCAD'13],

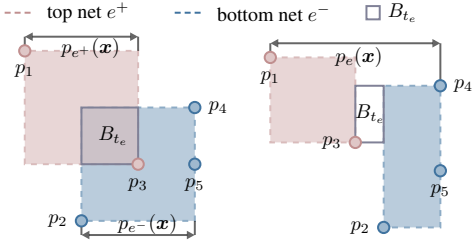
$$W_e(\boldsymbol{x}) \approx \hat{W}_{e,\mathrm{WA}}(\boldsymbol{x}) = \sum_{i=1}^{n} x_i \frac{\mathrm{e}^{\frac{x_i}{\gamma}}}{\sum_{i=1}^{n} \mathrm{e}^{\frac{x_i}{\gamma}}} - \sum_{i=1}^{n} x_i \frac{\mathrm{e}^{-\frac{x_i}{\gamma}}}{\sum_{i=1}^{n} \mathrm{e}^{-\frac{x_i}{\gamma}}}. \tag{2}$$

- The cut C_e can also be optimized by $\hat{W}_{e,\mathrm{WA}}(\boldsymbol{z})$.



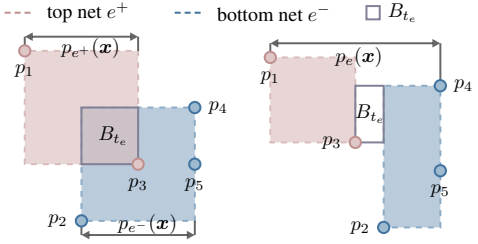
Applying WA to $p_e(\boldsymbol{x})$ is not always equivalent to $p_{e^+}(\boldsymbol{x}) + p_{e^-}(\boldsymbol{x})$.

Bistratal Wirelength Model [TCAD'23][10]



- Where to place the HBT t_e for the crossing-die net e?
- The optimal region $B_{t_e} = [x'_{\min}, x'_{\max}] \times [y'_{\min}, y'_{\max}]$ for the HBT t_e is defined as,

$$
\begin{aligned}
x'_{\min} &= \min\left\{\max\left\{x^+_{\min}, x^-_{\min}\right\}, \min\left\{x^+_{\max}, x^-_{\max}\right\}\right\}, \\
x'_{\max} &= \max\left\{\max\left\{x^+_{\min}, x^-_{\min}\right\}, \min\left\{x^+_{\max}, x^-_{\max}\right\}\right\},
\end{aligned}
\tag{3}
$$



- Where to place the HBT t_e for the crossing-die net e?
- The optimal region $B_{t_e} = [x'_{\min}, x'_{\max}] \times [y'_{\min}, y'_{\max}]$ for the HBT t_e is defined as,

$$x'_{\min} = \min\left\{\max\left\{x^+_{\min}, x^-_{\min}\right\}, \min\left\{x^+_{\max}, x^-_{\max}\right\}\right\},$$
$$x'_{\max} = \max\left\{\max\left\{x^+_{\min}, x^-_{\min}\right\}, \min\left\{x^+_{\max}, x^-_{\max}\right\}\right\}, \quad (3)$$

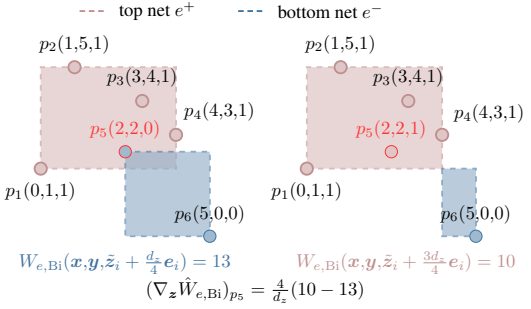
Bistratal Wirelength

The partial nets e^+ and e^- are determined by the tentative partition $\boldsymbol{\delta} = P(\boldsymbol{z})$.

$$W_{e,\mathrm{Bi}}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}) = \max\left\{p_e(\boldsymbol{x}), p_{e^+}(\boldsymbol{x}) + p_{e^-}(\boldsymbol{x})\right\} + \max\left\{p_e(\boldsymbol{y}), p_{e^+}(\boldsymbol{y}) + p_{e^-}(\boldsymbol{y})\right\}. \quad (4)$$

- Apply WA to $W_{e,\mathrm{Bi}} \to \hat{W}_{e,\mathrm{Bi}}$.

- How to optimize z by $\nabla_{\boldsymbol{z}} \hat{W}_{e,\mathrm{Bi}}$? (non-smooth)



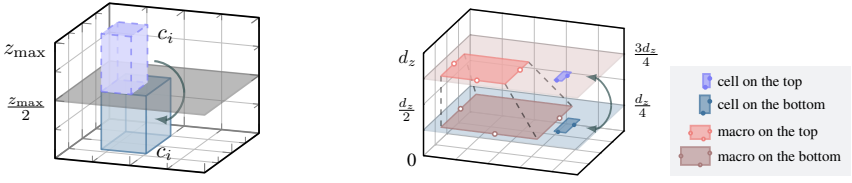
Finite Difference Approximation

The finite difference approximation (FDA) $(\nabla_{\boldsymbol{z}} \hat{W}_{e,\mathrm{Bi}})_i$ is defined by

$$(\nabla_{\boldsymbol{z}} \hat{W}_{e,\mathrm{Bi}})_i = \frac{4}{d_z}\Big(W_{e,\mathrm{Bi}}(\boldsymbol{x}, \boldsymbol{y}, \tilde{\boldsymbol{z}}_i + \tfrac{3d_z}{4}\boldsymbol{e}_i) - W_{e,\mathrm{Bi}}(\boldsymbol{x}, \boldsymbol{y}, \tilde{\boldsymbol{z}}_i + \tfrac{d_z}{4}\boldsymbol{e}_i)\Big), \tag{5}$$

where $\boldsymbol{x}$ and $\boldsymbol{y}$ are fixed, and $\tilde{\boldsymbol{z}}_i = \boldsymbol{z} \odot (\boldsymbol{1} - \boldsymbol{e}_i)$ and $\boldsymbol{e}_i \in \mathbb{R}^{|V|}$ is the unit vector with the i-th entry being 1 and others being 0.

- **Electrostatics-Based Density Model**. We adopt the eDensity3D [ISPD'16] with dynamically updating instance size for different technologies.



(a) (b)

- Let w_i^+ denote the instance width on the top die, and w_i^- on the bottom die. The dynamic width w_i can be derived as

$$w_i = \delta_i w_i^+ + (1 - \delta_i) w_i^- . \tag{6}$$

- For any macro $v_i \in V_M$, we propose to linearly transform macro width as

$$w_i = (\tfrac{2z_i}{d_z} - \tfrac{1}{2}) w_i^+ + (\tfrac{3}{2} - \tfrac{2z_i}{d_z}) w_i^- . \tag{7}$$

- eDensity3D computes the density gradient by sovling the 3D Poisson's equation using spectral methods.

$$\begin{aligned} \Delta\phi(\boldsymbol{v}) &= -\rho(\boldsymbol{v}), \quad \boldsymbol{v} \in \Omega \\ \hat{\boldsymbol{n}} \cdot \nabla\phi(\boldsymbol{v}) &= 0, \quad \boldsymbol{v} \in \partial\Omega. \end{aligned} \tag{8}$$

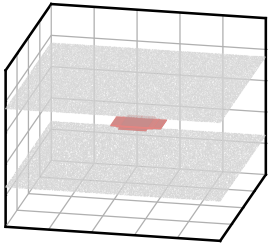
- Let $(\omega_j, \omega_k, \omega_l) = (\frac{j\pi}{d_x}, \frac{k\pi}{d_y}, \frac{l\pi}{d_z})$ denote the frequency indices. The density frequency coefficients a_{jkl} are computed as

$$a_{jkl} = \frac{1}{N} \sum_{x,y,z} \rho(x, y, z) \cos(\omega_j x) \cos(\omega_k y) \cos(\omega_l z), \tag{9}$$

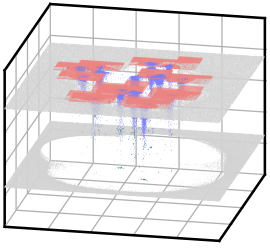
- The electric field $\boldsymbol{E}(x, y, z) = (E_x, E_y, E_z)$ is computed as below

$$\begin{aligned} E_x &= \sum_{j,k,l} \frac{a_{jkl}\omega_j}{\omega_j^2 + \omega_k^2 + \omega_l^2} \sin(\omega_j x) \cos(\omega_k y) \cos(\omega_l z), \\ E_y &= \sum_{j,k,l} \frac{a_{jkl}\omega_k}{\omega_j^2 + \omega_k^2 + \omega_l^2} \cos(\omega_j x) \sin(\omega_k y) \cos(\omega_l z), \\ E_z &= \sum_{j,k,l} \frac{a_{jkl}\omega_l}{\omega_j^2 + \omega_k^2 + \omega_l^2} \cos(\omega_j x) \cos(\omega_k y) \sin(\omega_l z). \end{aligned} \tag{10}$$

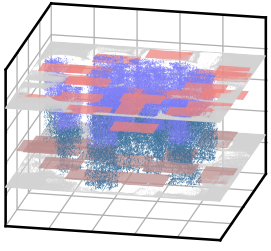
Placement Visualization



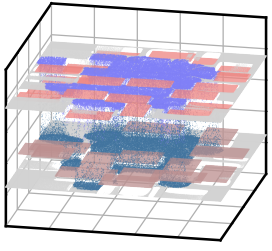
Iter.	WL (10^7)	#HBTs	OVFL
0	3.60	88507	1.00



Iter.	WL (10^7)	#HBTs	OVFL
1000	4.49	1254	0.89

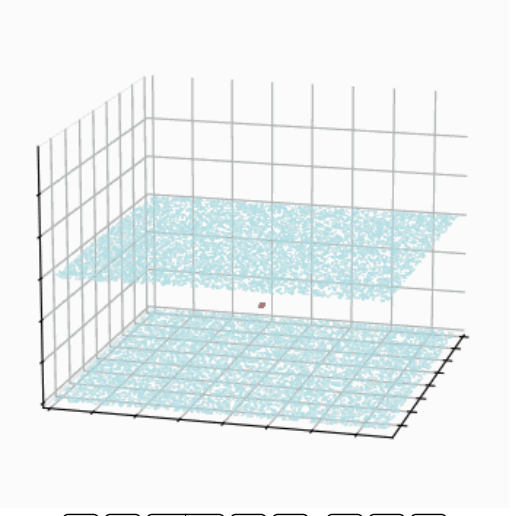


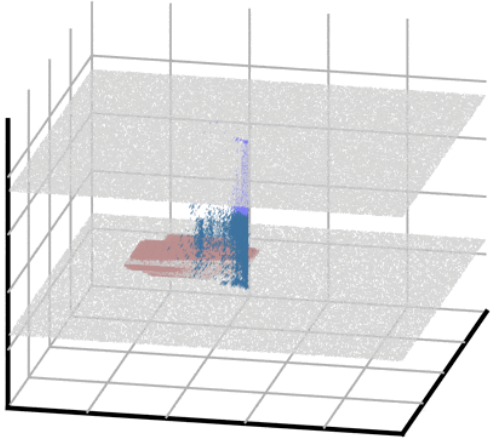
Iter.	WL (10^7)	#HBTs	OVFL
1800	10.30	30961	0.34



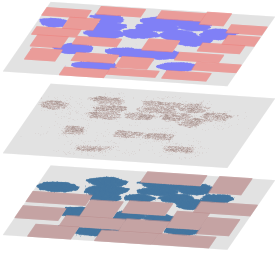
Iter.	WL (10^7)	#HBTs	OVFL
2142	10.51	13798	0.07

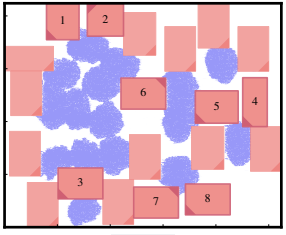
Placement Visualization



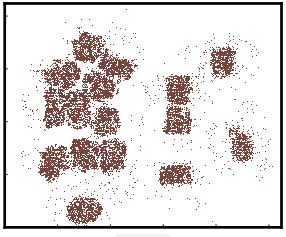


Placement Visualization

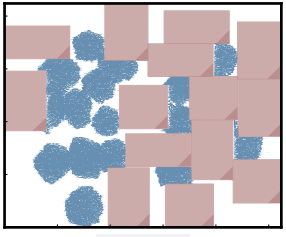




top die

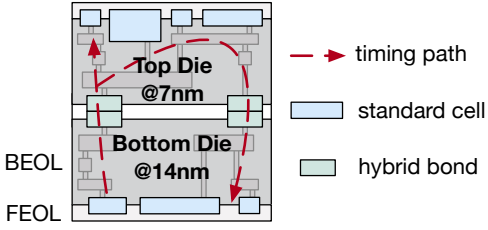


HBTs



bottom die

Motivation: Native Multi-Corner 3D STA



A 3D timing path.

Independent die corners

Each die has its own process, voltage, and temperature (PVT) corners.

All timing views

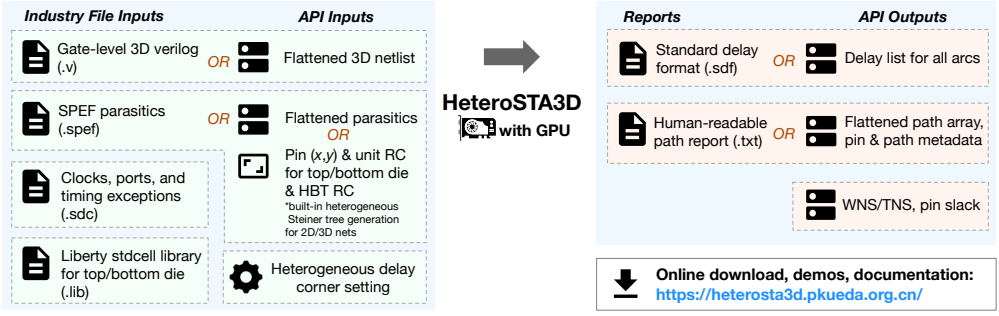
One view pairs a top-die corner with a bottom-die corner.

N top corners $\times$ M bottom corners $= N \times M$ views.

Key insight

All timing views share one graph spanning both dies and the HBT; only their timing parameters change.

HeteroSTA3D Architecture [CSTIC'26]

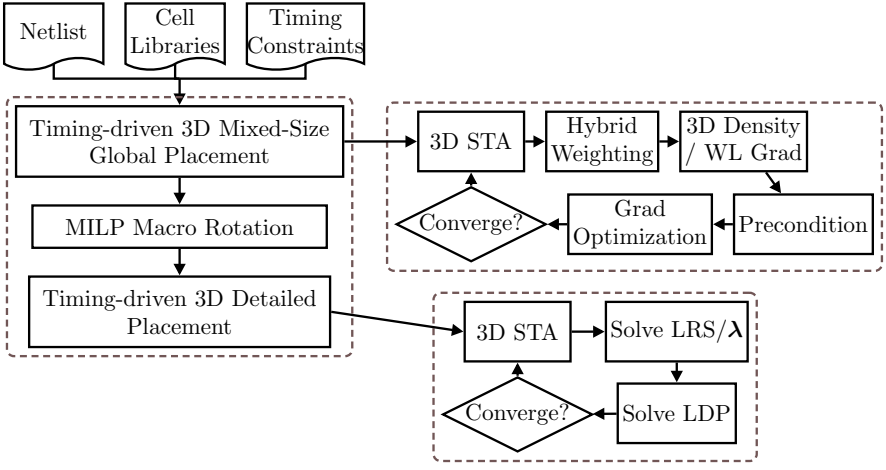


Per-die data	Native 3D model	Reusable results
Independent Liberty views	Explicit HBT R/C	WNS/TNS and top-k paths
Independent wire R/C	Unified timing graph	CPU/GPU result arrays

Reusable timing engine

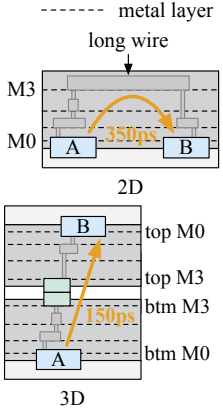
One reusable 3D STA library serves many corners and downstream optimizers in memory.

Timing-Driven 3D Placement Framework [DAC'26]



In-loop 3D STA

Both global and detailed placement repeatedly call the native 3D STA engine and use its timing results for the next placement update.



A far, critical sink can use the other die as a shortcut.

Net weight: shorten the critical net

Negative slack $\Longrightarrow$ larger net weight

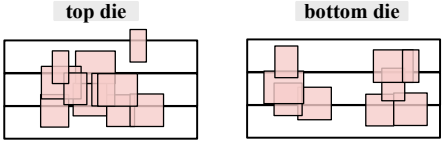
The placer pulls timing-critical connections closer.

Pin weight: help the critical sink cross dies

Distant + timing-critical $\Longrightarrow$ smaller penalty for placing the source and sink on different dies.

This allows global placement to explore a shorter xy connection through the other die.

Native 3D Legalization [DAC'25][11]



(a) Global Placement

(b) 2D Legalizer

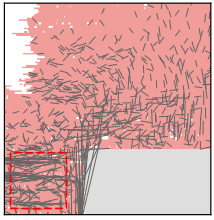
(c) 3D-Flow Legalizer

Yuxuan Zhao, Peiyu Liao, and Bei Yu (2025). "3D-Flow: Flow-based standard cell legalization for 3D ICs". In: *Proc. DAC*.

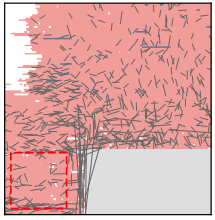
Ablation Study on Die-to-Die Cell Movement

Table: Comparison on the legalization results for our 3D-Flow legalizer with and without die-to-die (D2D) cell movement.

ICCAD 2023	w/o. D2D		Ours		
	Avg. Disp.	Max. Disp.	Avg. Disp.	Max. Disp.	#Move
case2	2.197	11.33	2.109	9.09	35
case2h1	2.506	35.28	2.433	27.03	46
case2h2	2.711	32.10	2.518	27.73	54
case3	2.582	98.98	2.442	70.82	493
case3h	2.229	107.84	2.204	99.05	373
case4	0.931	15.32	0.920	13.77	2479
case4h	1.358	54.33	1.130	54.31	2081
Average	1.068	1.19	**1.000**	**1.00**	-



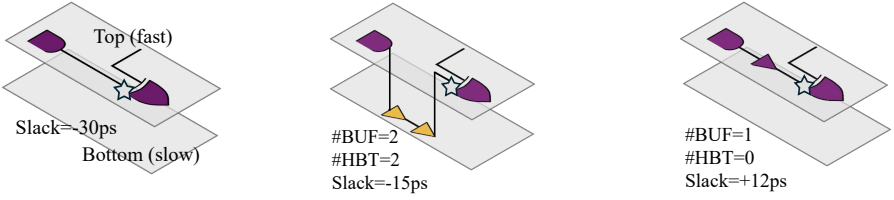
(c) w/o. D2D cell movement



(d) Our 3D-Flow Legalizer

- By moving only a few cells vertically, our legalizer finds white space for other cells to minimize displacement in a global view.

Timing Optimization with Die Reassignment [ICCAD'26][12]



Initial: -30 ps Fixed die assignment: -15 ps Die reassignment: $+12$ ps

Cell library

The target die determines the available cells.

HBT parasitics

Die reassignment may add or remove HBT crossings.

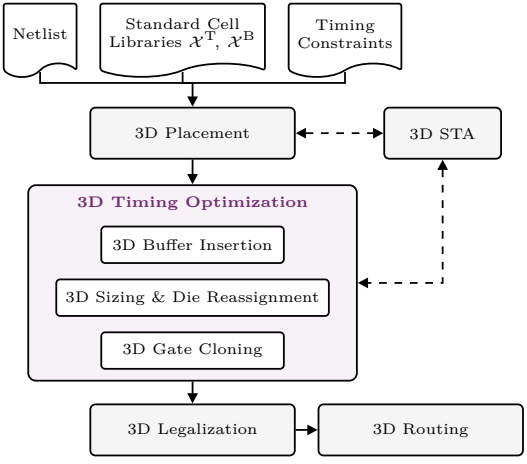
Timing operators

Buffering, sizing, and cloning must consider both dies.

Key insight

Timing optimization and die assignment must be considered together.

Post-Placement Timing Optimization Flow



Core idea

Keep die assignment editable during timing closure. 3D STA stays in the optimization loop and evaluates all three operators with both libraries and HBT parasitics.

Results Across Technology Configurations

Compared with OpenROAD timing optimization before 3D placement

Heterogeneous Scenario

ASAP7 on the top die and NanGate45 on the bottom die

Average WNS: 20% lower

Average TNS: 37% lower

Homogeneous Scenario

ASAP7 on both dies

WNS: 8% lower; TNS: 17% lower

NanGate45 on both dies

WNS: 3% lower; TNS: 25% lower

Takeaway

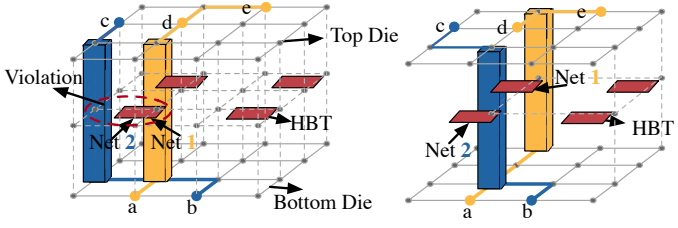
Our method improves timing in both scenarios. Gains are larger in the heterogeneous case.

How to Handle the HBT in Routing?

Connection Type	C4 Bump	TSV & μbump	Hybrid Bond	Metal Via*
Typical Pitch	250 μm	50 μm	<10 μm	0.1 μm
# of Connection†	16	400	$> 10^4$	10^8

*: *Via below top-most metal in 28 nm node.* †: *In mm^2.*

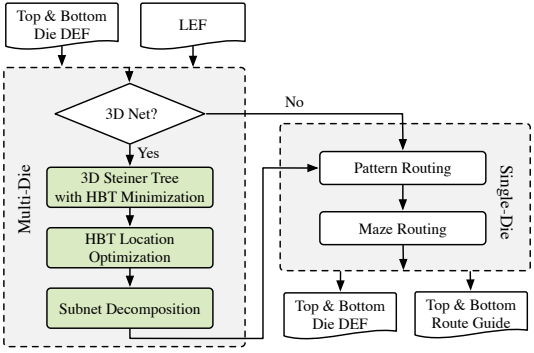
- Existing commercial pseudo-3D flows [TCAD'19] require post-routing HBT legalization [TCAD'24][13] to avoid spacing violations.
- Existing 3D pattern routers face the same issue.
- We model routing and HBT resources jointly with a heterogeneous grid graph.



(a) Uniform Grid Graph **(b) Heterogeneous Grid Graph**

[13] **Yen-Hsiang Huang et al. (2024). "On Legalization of Die Bonding Bumps and Pads for 3D ICs".** In: *IEEE TCAD*.

Our 3D Global Router Framework [ICCAD'25][14]



- H3D is a sequential global router that handles both 3D and 2D nets.
- H3D processes nets in ascending order of their HPWL.
- H3D efficiently minimizes the HBT counts and optimizes HBT locations to achieve minimal wirelength without spacing violations.

Yuxuan Zhao, Feng Gu, et al. (2025). "H3D: Heterogeneous Resources Aware Global Router for Face-to-Face Bonded 3D ICs". In: *Proc. ICCAD*, pp. 1–9.

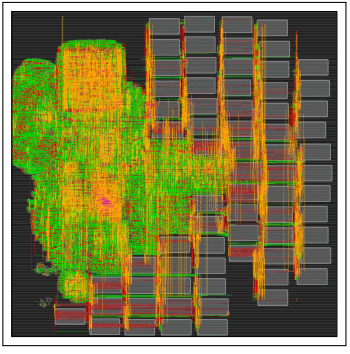
Table: Comparison of experimental results between our router and SOTA global routers.

Bench.	FastRoute [ASPDAC'09][15] + Legal [TCAD'24]				CUGR 2.0 [DAC'23] + Legal [TCAD'24]				Ours			
	WL	#Vias	#HBTs	RT	WL	#Vias	#HBTs	RT	WL	#Vias	#HBTs	RT
rocket	374 421	200 585	2800	9.163	369 069	210 891	2581	4.370	**336830**	205 913	**2179**	**2.441**
aes	156 045	119 885	703	4.996	157 048	125 413	831	2.466	**151657**	123 480	**526**	**1.335**
ethmac	557 884	221 480	5912	13.436	545 503	238 256	5738	6.338	**439322**	223 486	**4954**	**2.975**
jpeg	3 056 841	1 402 528	25 552	64.726	2 679 224	1 441 034	22 064	33.901	**2514129**	1 413 384	**19507**	**18.648**
mor1kx	1 516 476	559 389	10 443	26.012	1 380 750	577 274	9413	14.769	**1330973**	562 741	**7664**	**7.684**
ariane	3 684 778	1 377 210	25 837	72.921	3 506 201	1 459 846	25 163	38.048	**3142161**	1 396 919	**19504**	**18.736**
BP	6 401 377	2 018 460	34 430	113.263	5 634 172	2 094 227	28 428	72.662	**4700736**	1 986 244	**24315**	**35.960**
Average	1.186	**0.989**	1.318	3.701	1.116	1.036	1.249	1.937	**1.000**	1.000	**1.000**	**1.000**

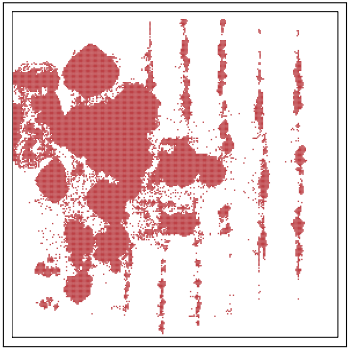
- **12%** shorter wirelength, **25%** fewer HBTs, and **1.9**× speedup compared to the baseline.

[15] **Yue Xu, Yanheng Zhang, and Chris Chu (2009). "FastRoute 4.0: Global router with efficient via minimization".** In: *Proc. ASPDAC*. IEEE, pp. 576–581.

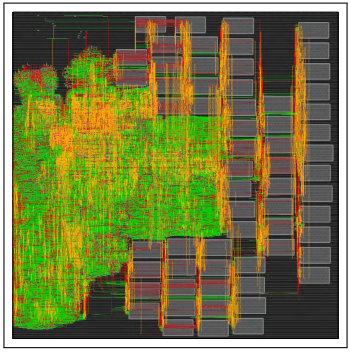
Routing Results Visualization



(e) Top Die



(f) Hybrid Bonding Terminals



(g) Bottom Die

The layouts for top, HBT, and bottom dies of `ariane`.

Impact of HBT Pitch Size

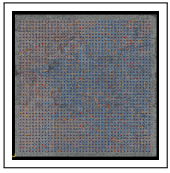
Table: Comparison of different HBT pitch sizes.

Bench.	Ours-2 µm				Ours-3 µm			
	WL	#Vias	#HBTs	RT	WL	#Vias	#HBTs	RT
rocket	**317907**	204 640	2275	2.427	336 830	205 913	**2179**	2.441
aes	**150940**	123 640	628	1.398	151 657	123 480	**526**	1.335
ethmac	**382196**	221 734	5563	2.666	439 322	223 486	**4954**	2.975
jpeg	**2442417**	1 413 078	20 297	18.067	2 514 129	1 413 384	**19507**	18.648
mor1kx	**1315771**	561 775	8030	7.708	1 330 973	562 741	**7664**	7.684
ariane	**3037041**	1 394 232	20 739	18.645	3 142 161	1 396 919	**19504**	18.736
BP	**4300681**	1 974 723	27 324	35.543	4 700 736	1 986 244	**24315**	35.960
Average	**0.950**	**0.997**	1.091	**0.985**	1.000	1.000	**1.000**	1.000

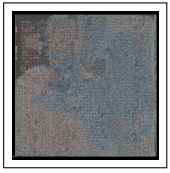
- With the 2 µm pitch, H3D achieved a 5% improvement in wirelength while using 9% more HBTs.

Our Commercial Tool Plus

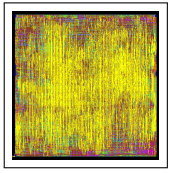
Industry-aligned implementation flow: Integrity 3DIC



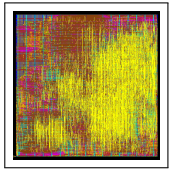
(a) Bottom



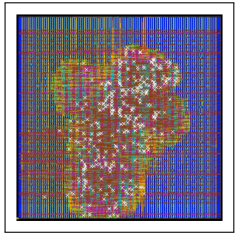
(b) Top



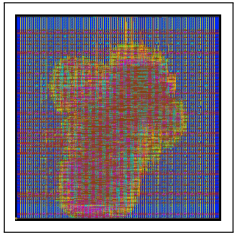
(c) Bottom



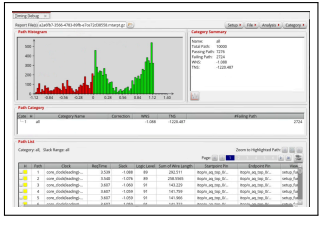
(d) Top



(e) Before fixing



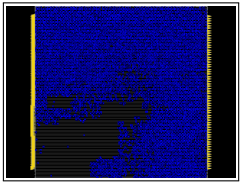
(f) After fixing



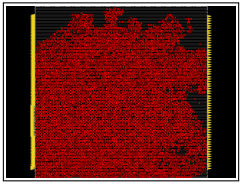
(g) Timing result

- Optimized for an advanced 3 nm GAA PDK
- Achieve DRC-clean routing.

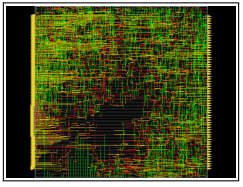
2D placement → **partition** → tier-coordinate scaling → legalization and refinement



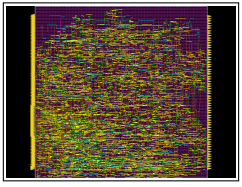
Bottom placement



Top placement



Bottom routing



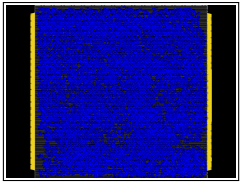
Top routing

Partition-last implementation

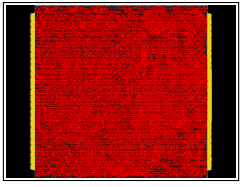
The monolithic 2D placement supplies the physical seed before cells are partitioned and mapped to the two tiers.

2D commercial-based 3D: Rocket_BIGRTL

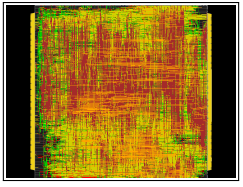
Partition → top-die placement → HB-layer insertion → bottom-die placement



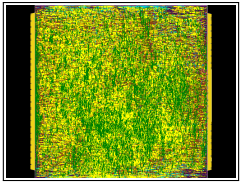
Bottom placement



Top placement



Bottom routing

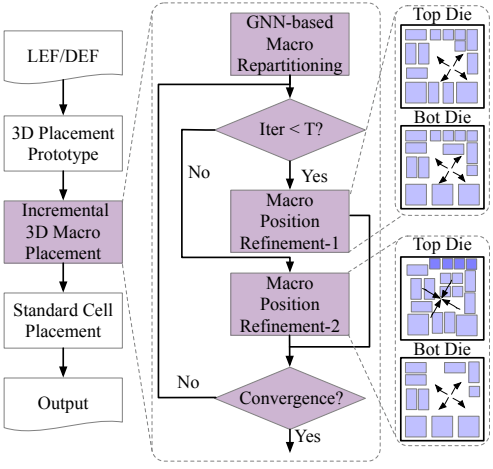


Top routing

Partition-first implementation

The design is partitioned before placement; the bottom tier is then placed around the fixed inter-tier interface.

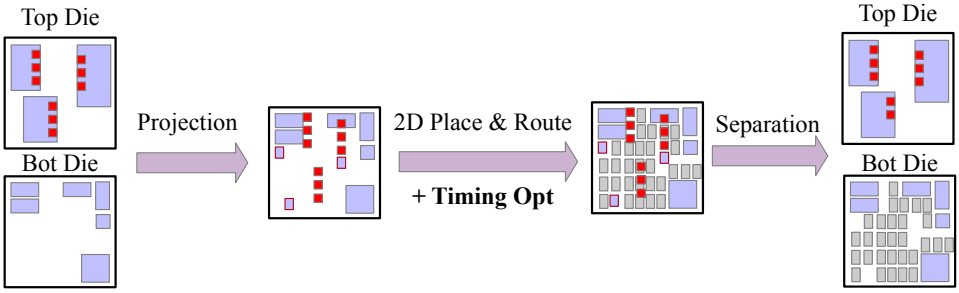
IncreMacro-3D: Incremental Macro Placement [DATE'26]



- Objective:[a] Given a 3D placement prototype, refine the tier assignment and positions of macros.
- 3D placement prototype is generated by analytical 3D MoL placer.

[a] **Lancheng Zou et al. (2026). "IncreMacro-3D: Incremental Macro Placement for Face-to-Face Stacked Memory-on-Logic 3D ICs". In:** *Proc. DATE.*

Pseudo 3D Physical Design Flow



- Step 1: Obtain the 3D macro placement from IncreMacro-3D.
- Step 2: Project macros from the top die to the bottom die, reducing their footprint while preserving pin locations and layers.
- Step 3: Use the commercial 2D P&R engine for standard-cell placement and routing. The doubled metal stack and matched extraction model enable **buffer insertion and gate sizing**.
- Step 4: Separate into two dies.

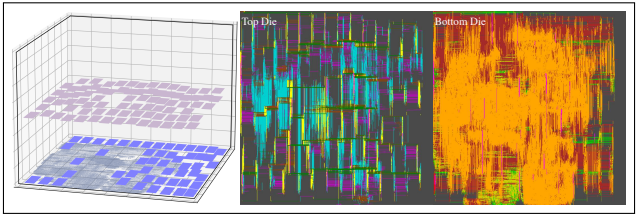
Experimental Results

Table: Comparison of post-route results among Open3D-DMP[16], IncreMacro-3D w.o. GNN based Macro Repartitioning(GMRP), and IncreMacro-3D.

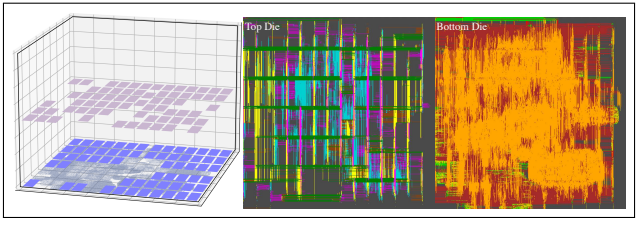
Benchmark	Open3D-DMP					IncreMacro-3D w.o. GMRP					IncreMacro-3D				
	rWL (μm)	WNS (ns)	TNS (ns)	Power (mW)	Runtime (s)	rWL (μm)	WNS (ns)	TNS (ns)	Power (mW)	Runtime (s)	rWL (μm)	WNS (ns)	TNS (ns)	Power (mW)	Runtime (s)
ariane133	4394142	-0.070	-33.120	840.96	80	**4074873**	-0.063	-4.553	**836.54**	416	4134944	**-0.038**	**-4.186**	846.03	437
ariane136	4694011	-0.056	-20.710	866.86	85	**4383761**	-0.044	-12.693	**853.16**	432	5051825	**-0.036**	**-5.510**	858.79	455
bp_quad	30365842	-0.687	-11649.900	4511.72	325	27993175	-0.516	-8499.100	4490.81	742	**25876519**	**-0.465**	**-6595.600**	**4455.67**	795
mempool_tile	2538213	-0.021	-1.361	79.66	110	2578753	-0.093	-27.690	80.82	340	**2467694**	**-0.019**	**-1.225**	**79.55**	378
NVDLA	11106336	-0.143	-6.136	2368.23	70	9810449	-0.009	-0.010	2345.79	149	**9500393**	**-0.003**	**-0.011**	**2337.83**	159
Average	1.000	1.000	1.000	1.000	**1.000**	**0.936**	1.386	4.365	0.996	3.103	0.939	**0.558**	**0.372**	**0.994**	3.319

IncreMacro-3D outperforms the state-of-the-art analytical placer for MoL 3D ICs by reducing routed wirelength, worst negative slack (WNS), total negative slack (TNS), and total power consumption by **6.1**%, **44.2**%, **62.8**%, and **0.6**%, respectively.

[16] **Yunqi Shi et al. (2025). "Open3DBench: Open-Source Benchmark for 3D-IC Backend Implementation and PPA Evaluation".** In: *arXiv:2503.12946*.

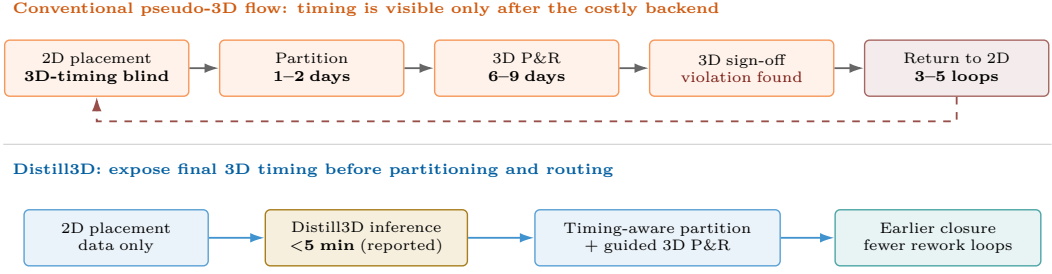


(h) Open3D-DMP on ariane133



(i) IncreMacro-3D on ariane133

The 2D-to-3D Timing Gap [ICCAD'26][17]

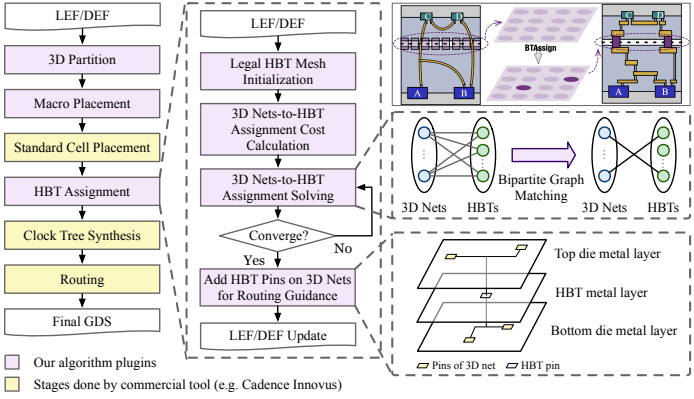


Shift-left objective

Shift 3D sign-off timing visibility to the 2D placement stage.

Mingjun Wang et al. (2026). "Distill3D: Cross-Stage Knowledge Distillation for Early-Stage 3D IC Timing Prediction". In: *Proc. ICCAD*.

- We extended our HBT assignment algorithm to support commercial tools (e.g., Cadence Innovus).

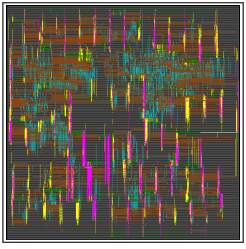


Pseudo 3D physical design flow with HBT assignment

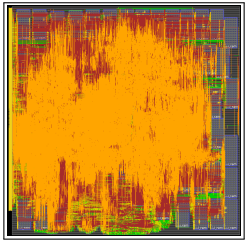
- 3D partition and macro placement stages can be done by our 3D incremental macro placement framework.
- We double the metal stacks and add a special metal layer (blocked planar routing) for HBTs.
- **Timing-aware optimization** can be conducted with STA information from commercial tools.

Commercial HBT Assignment: Routing Visualization

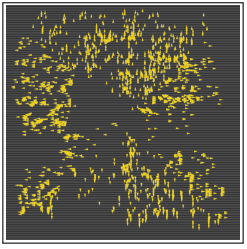
`ariane133` implemented in Cadence Innovus after legal HBT assignment



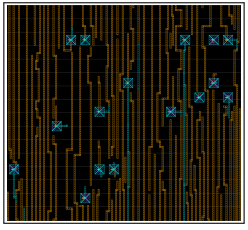
Top die



Bottom die



HBT layer



Routed HBT connections

Die-level evidence

Top- and bottom-die routing are completed after HBT assignment.

Interface evidence

The shared HBT layer and selected HBT-to-wire connections are visible in the routed database.

Conclusion

Conclusion

Summary

- Native 3D engines in physical design for 3D ICs.
- With a larger solution space, more optimization opportunities.
- Leveraging 3D point-tools w. 2D commercial tools

Future Work

- Cross-stage Optimization: Cover more native 3D physical design stages, including power delivery network, clock tree synthesis, and enable cross-stage optimization.
- Routing Optimization: With the increased density of 3D interconnects, metal layer sharing becomes feasible, which can be tuned in the commercial tools.