RUPlace: Optimizing Routability via Unified Placement and Routing Formulation

Yifan Chen, Jing Mai, Zuodong Zhang, Yibo Lin
Peking University, Beijing, China
chenyifan2019@pku.edu.cn

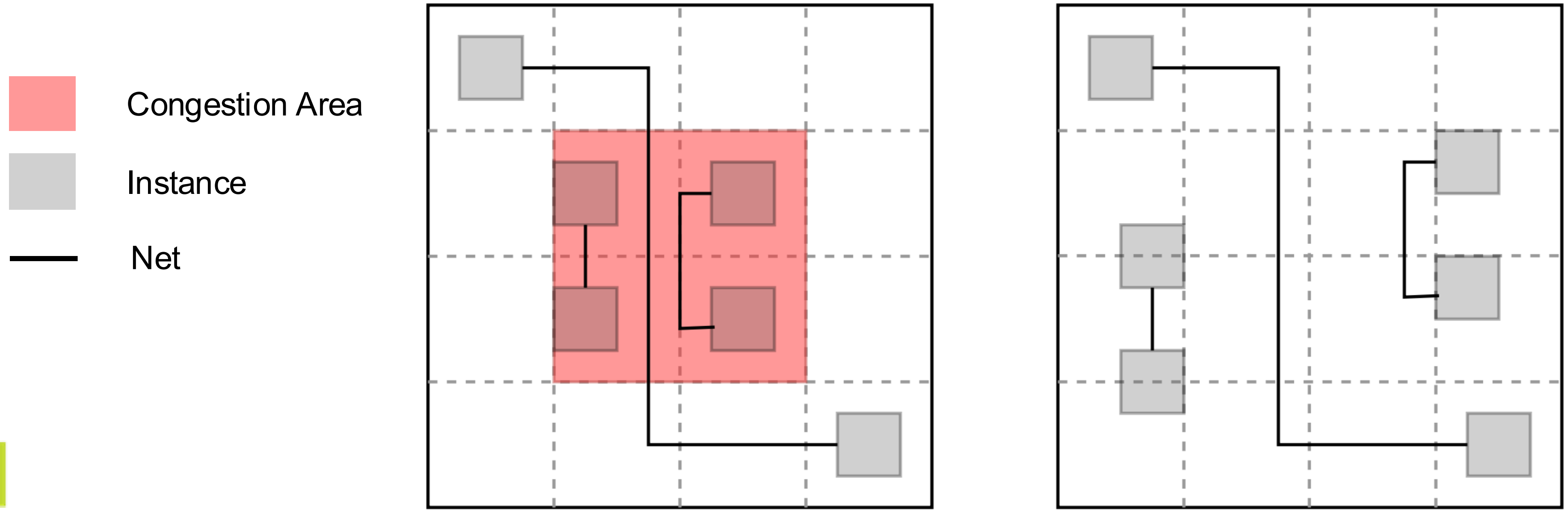
THE CHIPS TO SYSTEMS CONFERENCE

SPONSORED BY

Background

Routability Optimization in Placement

- Placement is critical to VLSI physical design, especially routability
- Increasing chip complexity → congestion challenges

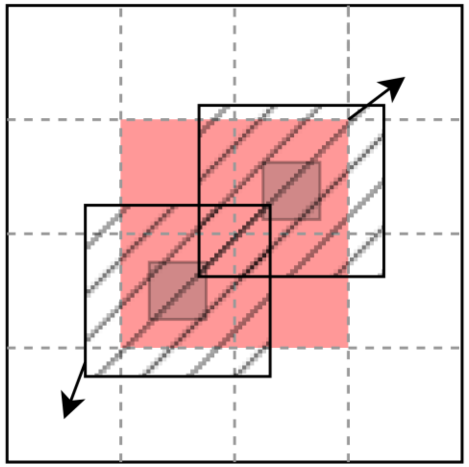


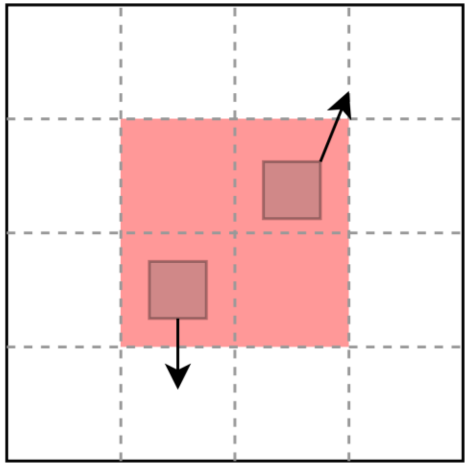
A typical example of routability optimization in placement.

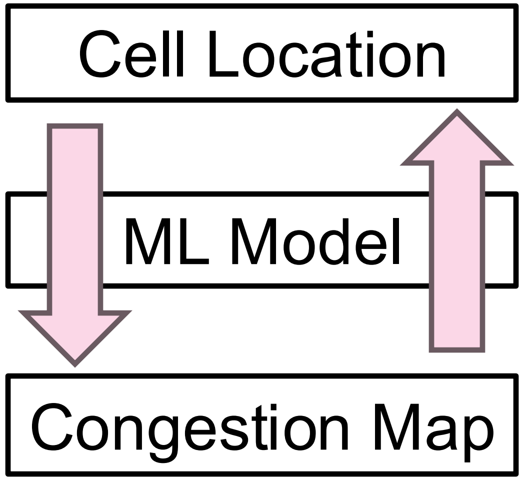
Limitations of Existing Approaches

- Heuristic-based cell inflation
  - [Lin+, DAC2014][He+, TODAES2016] [Liu+, TCAD2023]
- Force-based methods
  - [Huang+, TCAD2018][Lin+, ICCAD2021]
- ML methods
  - [Liu+, DATE2021][Park+, ICCAD2023]

Lack theoretical foundations
Treat routing as black-box







Contributions

- Unified Formulation of placement and routing optimization
- ADMM-based algorithm with Wasserstein distance and bilevel optimization
- Convex optimization-based cell inflation with modularity-based clustering
- Achieves substantial congestion reduction, better wirelength

Methodologies

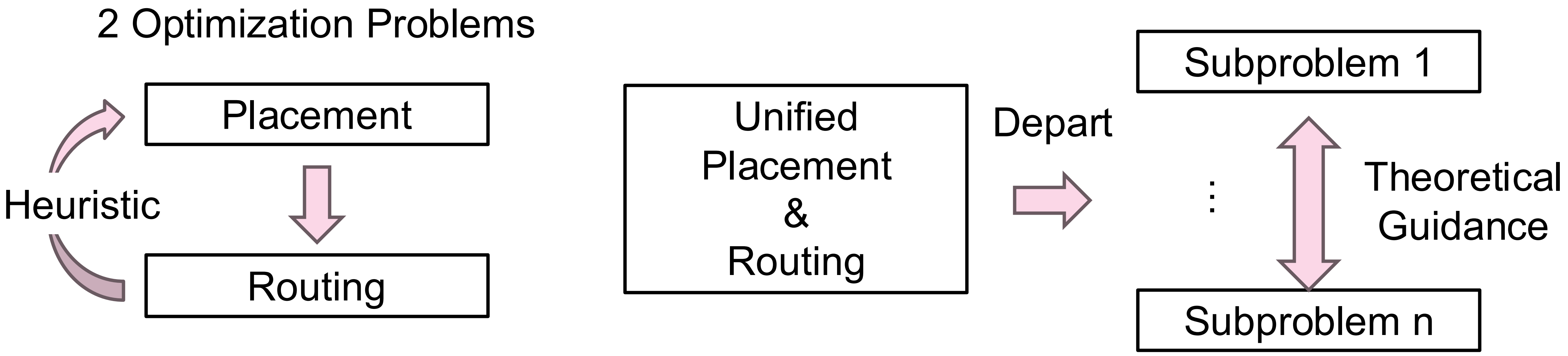
THE CHIPS TO SYSTEMS CONFERENCE

SPONSORED BY

Overview - Unified Placement & Routing

Traditional Flow

Unified Placement & Routing



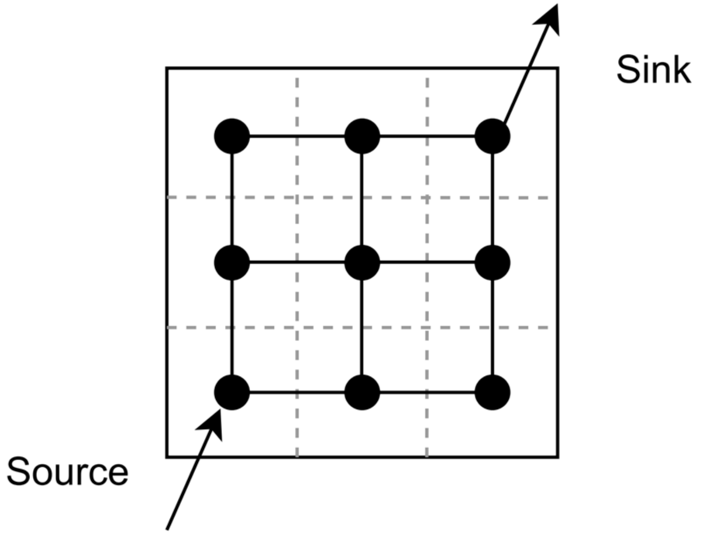
Global Routing Formulation

- ILP based Global Routing (Network Flow)

$$\mathcal{R} : \min_f \quad R(f) = \sum_{e \in E} c_e \sum_{n \in N} f_{n,e} + \mu CONG(f),$$

$$\text{s.t.} \quad Af = h(x),$$

$$f_{n,e} \in \{0,1\}, \forall n \in N, e \in E,$$



$$CONG(f) = \|\max(\sum_n f_{n,e} - cap_e, 0)\|.$$

$$h(x)_{n,u} = \begin{cases} +1, & \text{if the source pin of net } n \text{ in gcell } u, \\ -1, & \text{if the sink pin of net } n \text{ in gcell } u, \\ 0, & \text{otherwise,} \end{cases}$$

Unified Placement & Routing Formulation

- Analytical Placement

$$\begin{aligned} \min_x \quad & WL(x), \\ \text{s.t.} \quad & d_i(x) \leq d_t, \forall i \in B, \end{aligned}$$

+

- Global Routing

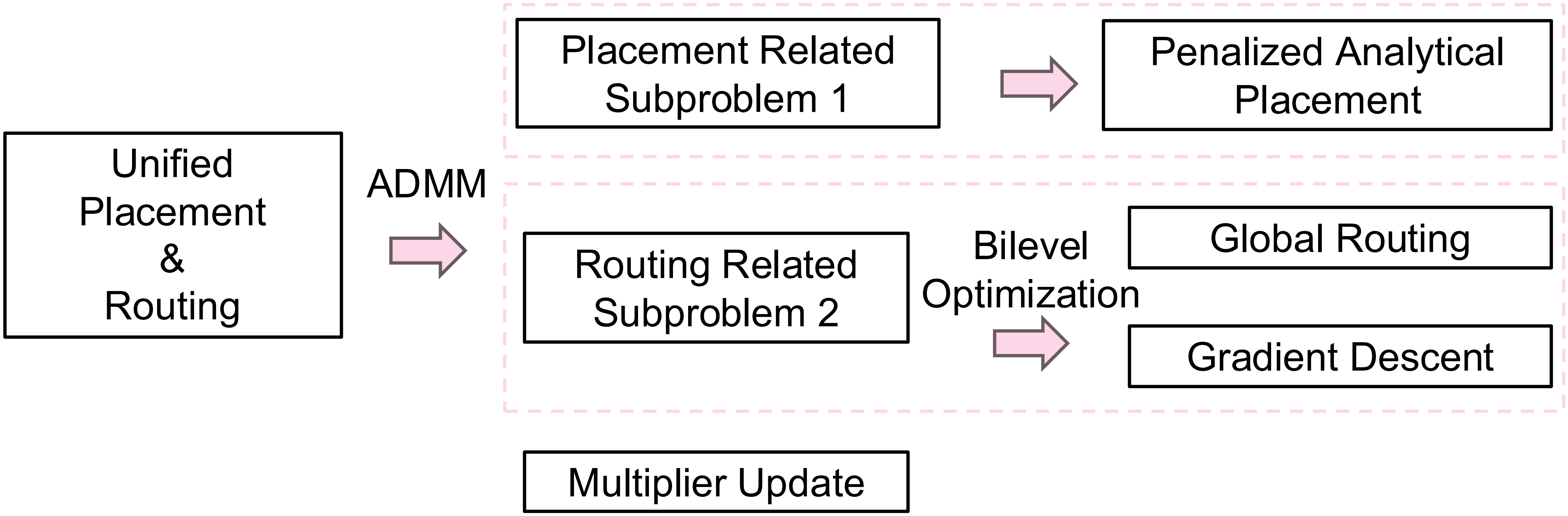
$$\begin{aligned} \mathcal{R}: \min_f \quad & R(f) \\ \text{s.t.} \quad & Af = h(x), \\ & f_{n,e} \in \{0,1\}, \forall n \in N, e \in E, \end{aligned}$$

⇒

Routability-driven Placement

$$\begin{aligned} \min_{x,f} \quad & R(f), \\ \text{s.t.} \quad & d_i(x) \leq d_t, \forall i \in B, \\ & Af = h(x), \\ & f_{n,e} \in \{0,1\}, \forall n \in N, e \in E, \end{aligned}$$

Overview – Solve Unified P&R



Solve by ADMM

- ADMM framework

$$t^{k+1}, f^{k+1} = \text{argmin}_{f,t} \quad L(x^k, t, f, \lambda^k, \sigma)$$
$$x^{k+1} = \text{argmin}_{x} \quad L(x, t^{k+1}, f^{k+1}, \lambda^k, \sigma)$$
$$\lambda^{k+1} = \lambda^k + \sigma(x^{k+1} - t^{k+1})$$

- Sub1: $t^{k+1}, f^{k+1} = \text{argmin}_{f,t} \quad L(x^k, t, f, \lambda^k, \sigma)$

$$t^{k+1} = \text{argmin}_t \quad \mathcal{R}(t) + \lambda^{k\dagger}(x^k - t) + \frac{\sigma}{2}(x^k - t)^2$$

Routing Problem

- Sub2: $x^{k+1} = \text{argmin}_x \quad L(x, t^{k+1}, f^{k+1}, \lambda^k, \sigma)$

$$\min_x \quad WL(x) + \lambda^{k\dagger}(x - t^{k+1}) + \frac{\sigma}{2}(x - t^{k+1})^2,$$
$$\text{s.t.} \quad d_i(x) \leq d_t, \forall i \in B.$$

Penalized Analytical Placement

Solve Routing-Related Subproblem

- Bilevel Optimization $t^{k+1} = \text{argmin}_t \quad \mathcal{R}(t) + \lambda^{k\dagger}(x^k - t) + \frac{\sigma}{2}(x^k - t)^2$

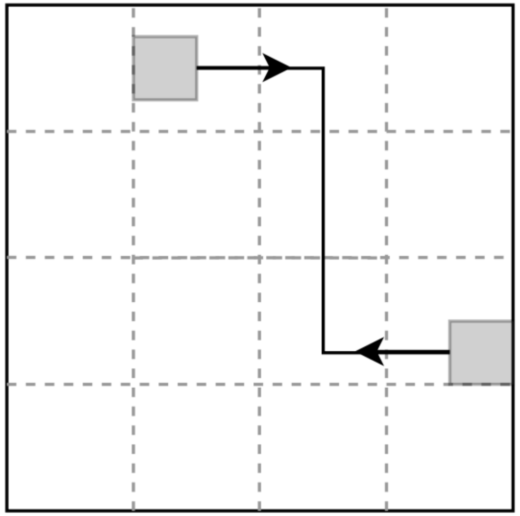
$$f^{k+1} = \mathcal{R}(t^k),$$

Solve by Global Router

$$t^{k+1} = \text{argmin}_t \quad \hat{\mathcal{R}}(f^{k+1}, t) + \lambda^{k\dagger}(x^k - t) + \frac{\sigma}{2}(x^k - t)^2$$

Approximation of routing
in the neighborhood of f^{k+1}
Only move t along the routed wires

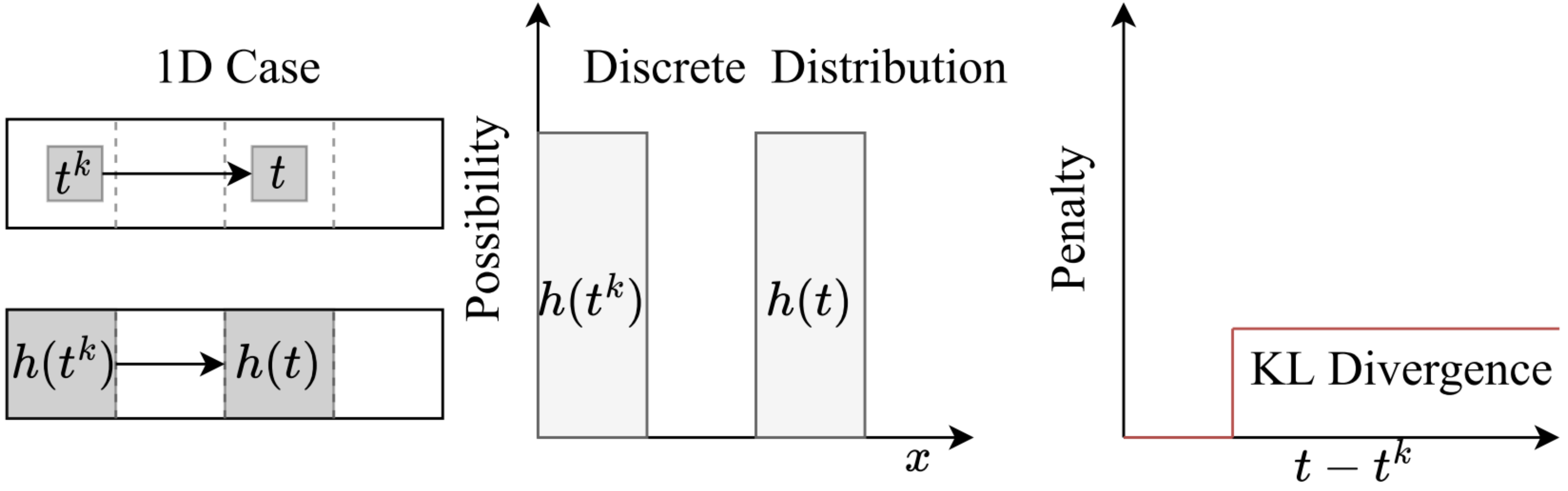
How to ensure this?



Neighborhood Constraint Issue

- $h(t)$: a discrete distribution
- Metrics like KL divergence: No overlap → constant distance, zero gradient

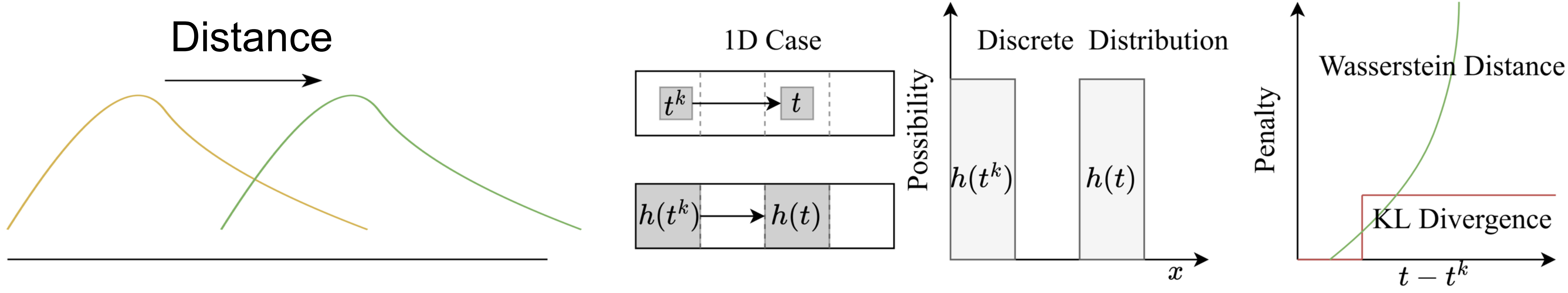
$$h(x)_{n,u} = \begin{cases} +1, & \text{if the source pin of net } n \text{ in gcell } u, \\ -1, & \text{if the sink pin of net } n \text{ in gcell } u, \\ 0, & \text{otherwise,} \end{cases}$$



Wasserstein Distance

- The Wasserstein distance is used to measure the distance between two probability distributions.

$$W_p(a,b) = \left\{ \inf_{\pi \in \Pi(a,b)} \sum_{(x_a, x_b) \in M \times N} \|x_a - x_b\|^p \, d\pi(x_a, x_b) \right\}^{1/p},$$



Approximation of Routing

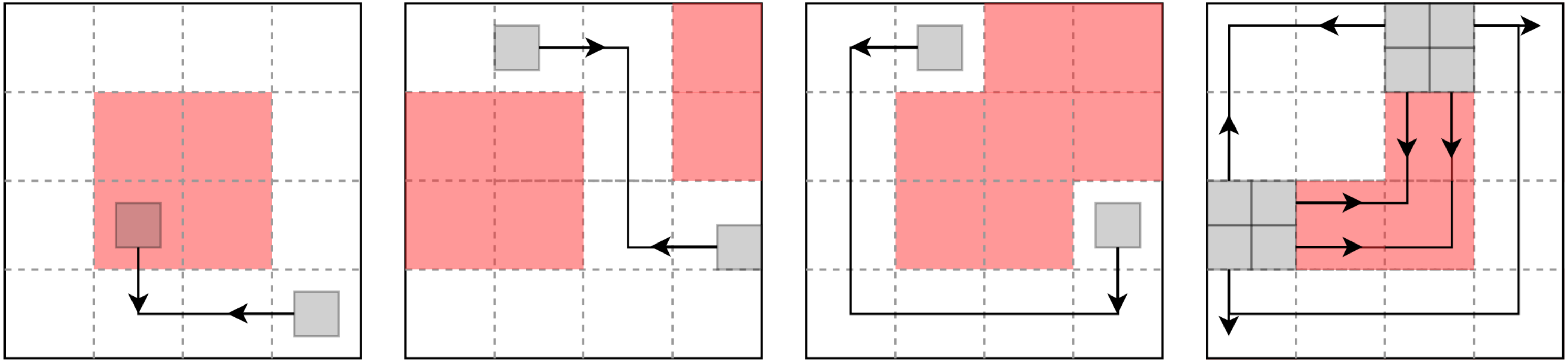
- Ensure t in the neighborhood, add Wasserstein distance penalty

$$t^{k+1} = \operatorname{argmin}_t \quad \hat{\mathcal{R}}(f^{k+1}, t) + \eta W_2(h^+(t), h^+(t^k))^2$$
$$+ \eta W_2(h^-(t), h^-(t^k))^2 + \lambda^{k\dagger}(x^k - t) + \frac{\sigma}{2}(x^k - t)^2$$

- Only move t along the routed wires
- Keep topological structure of wires unchanged
- Unconstrained quadratic programming, solve by single step of gradient descent

How unified placement and routing works

- Move cells along the routed wires



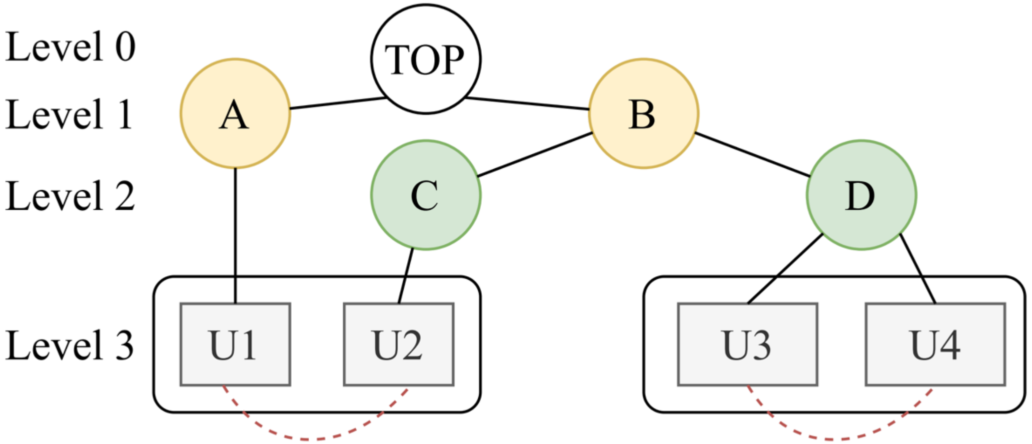
Examples of how unified placement and routing works

Modularity based Clustering

- Hypergraph modularity quantifies the quality of clustering in hypergraphs

$$Q = \frac{1}{|E|}\left(EC - \sum_{d=2}^{D} E_d \sum_{A_i \in A} \frac{|\mathrm{Vol}(A_i)|^d}{|\mathrm{Vol}(V)|}\right),$$

- Merge nodes in logical hierarchy tree
- From leaf to root
- Merge nodes when modularity gain > 0



Convex Global Inflation & Local Inflation

- Assuming uniform inflation across the cluster $\mathcal{F}: \quad \min_{v} \quad \sum_{g} \frac{WL_g}{v_g},$

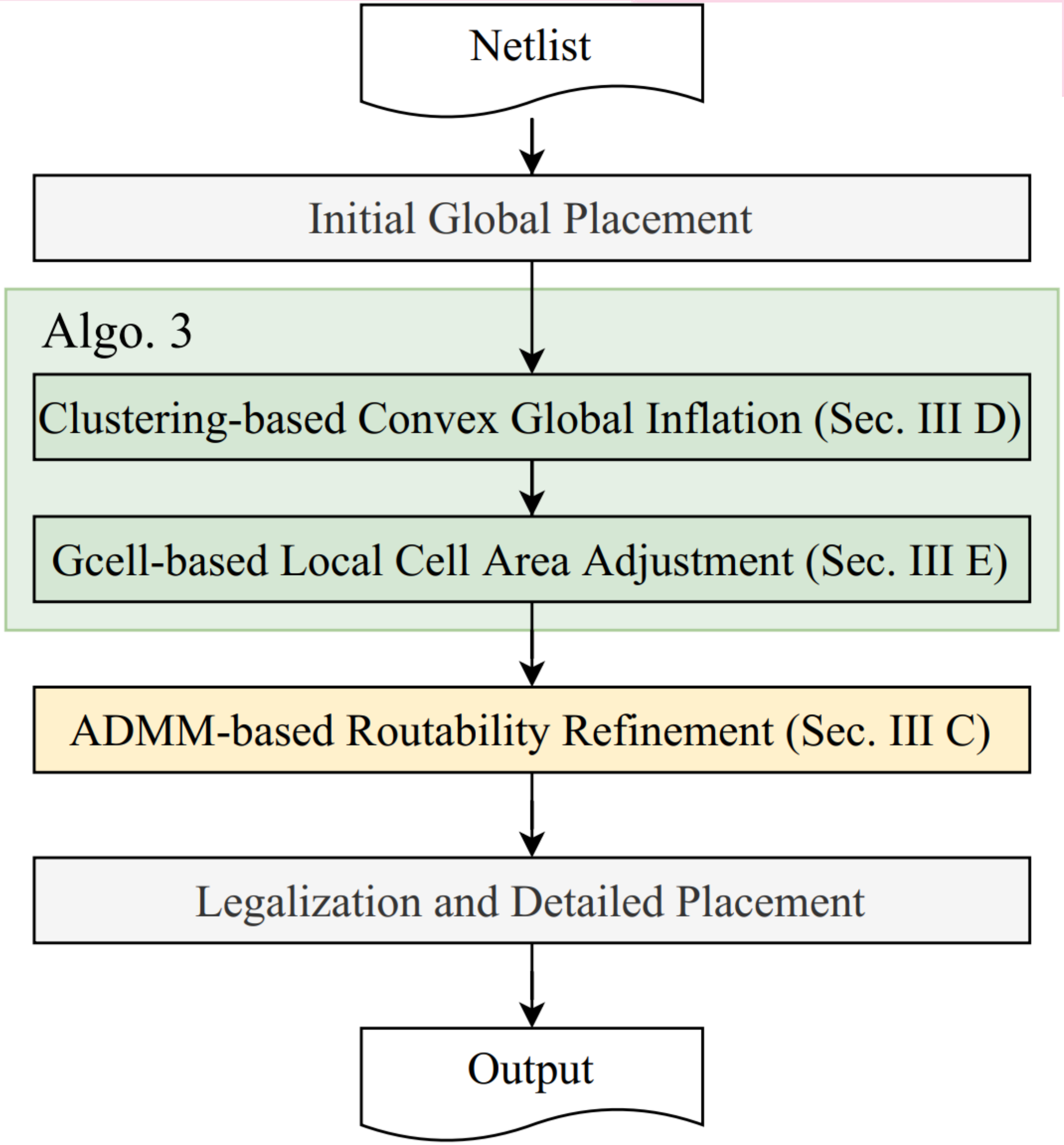
$$\text{s.t.} \quad \sum_{g} D_g(b) \cdot v_g \leq \text{cap}(b),$$

- $\text{cap}(b)$:capacity at bin b
- WL_g , v_g , $D_g(b)$:wirelength, inflation rate, routing demand at bin b of cluster g
- Inflate nodes by local routing demand

$$l'_c(b) = \max(1, \frac{ldmd(b)}{cap(b) - gdmd(b)}),$$

$$\hat{l}_c(b) = (1 - \gamma) l_c(b) + \gamma \min(\text{max\_inflate}, l'_c(b)),$$

Overall Flow



Experimental Results

THE CHIPS TO SYSTEMS CONFERENCE

SPONSORED BY

Experiment Setting

- Implementation
  - Based on DREAMPlace[Chen+, ICCAD2023] for placement
  - HeLEM-GR [Zhao+, ICCAD2024] for global routing (full GPU acceleration).
- Hardware:
  - Linux workstation, Intel Xeon Platinum 8358 CPU (32 cores) and NVIDIA A800 GPU (80GB memory).
- Benchmarks:
  - Open-source designs from CircuitNet[Chai+, TACD2023] and Chipyard[Amid+, Micro2020], 14nm PDK
- Compared Tools:
  - OpenROAD[Ajayi+, DAC19] (CPU), Xplace 2.0[Liu+, TCAD2023] (GPU), DREAMPlace 4.1[Chen+, ICCAD2023] (GPU).
- Metrics:
  - Congestion (horizontal/vertical overflow %), routed wirelength (μm), runtime (minutes).
  - Evaluated by Earlyglobalroute command in Innovus

Comparison with State-of-Art Placers

- Congestion much better
- Competitive wirelength
- Global router brings 1.32× runtime overhead

Design	OpenROAD				Xplace 2.0				DREAMPlace 4.1				RUPlace			
	rWL	C_H	C_V	RT	rWL	C_H	C_V	RT	rWL	C_H	C_V	RT	rWL	C_H	C_V	RT
OPENC910	1.34e7	7.17	4.18	20.4	1.47e7	7.27	2.68	2.8	1.22e7	10.56	5.47	1.6	1.56e7	2.02	0.72	4.3
NVDLA_S	4.98e6	0.90	0.26	4.1	4.67e6	1.01	0.37	0.7	4.43e6	1.54	0.49	0.8	4.95e6	0.09	0.09	1.8
NVDLA_L	3.92e7	3.67	0.55	28.0	3.80e7	3.78	0.69	4.3	3.58e7	4.78	1.36	3.3	4.43e7	1.36	0.23	7.1
VORTEX_S	2.63e6	2.42	0.94	5.8	1.64e6	0.85	0.34	0.5	1.59e6	1.22	0.59	0.3	1.71e6	0.28	0.16	0.8
VORTEX_L	1.17e7	0.17	0.08	12.6	1.12e7	0.24	0.14	1.6	1.10e7	0.60	0.29	2.2	1.09e7	0.13	0.10	4.9
GEMMINI	1.68e7	2.56	1.78	10.7	9.38e6	0.10	0.21	1.1	9.04e6	0.08	0.10	2.0	1.04e7	0.01	0.01	4.6
LARGEBOOM	1.20e7	0.06	0.02	10.5	1.00e7	0.97	0.51	1.4	9.78e6	1.55	0.93	1.7	1.17e7	0.31	0.11	4.0
Geo. Mean	1.07	4.74	3.47	3.67	0.93	4.11	3.88	0.45	0.88	5.91	5.80	0.43	1.00	1.00	1.00	1.00

Conclusion

- RUPlace presents an unified Formulation of placement and routing optimization
- ADMM-based algorithm with Wasserstein distance and bilevel optimization
- Convex optimization for cell inflation significantly enhances routability
- Proven performance improvements in congestion, wirelength, runtime

Future Works:
- Routing Acceleration
- Calibration using commercial tools

Thanks!
Questions are welcome!

THE CHIPS TO SYSTEMS CONFERENCE

SPONSORED BY

CEDA
IEEE Council on Electronic Design Automation

sigda acm