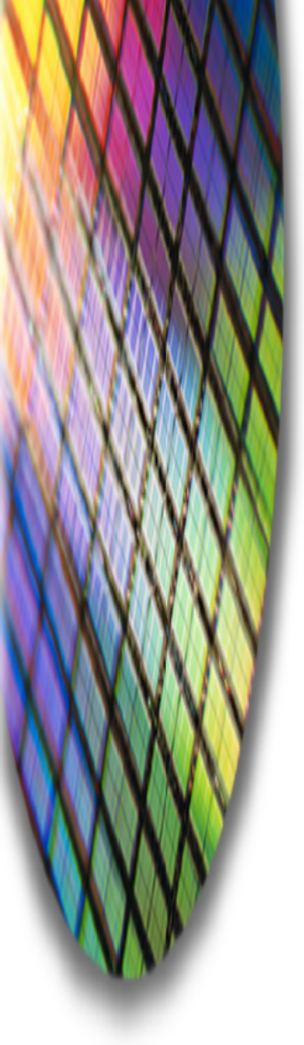


The University of Texas at Austin
Electrical and Computer Engineering
Cockrell School of Engineering

Device Layer-Aware Analytical Placement for Analog Circuits

Biying Xu[1], Shaolan Li[1], Chak-Wa Pui[2], Derong Liu[3], Linxiao Shen[1], Yibo Lin[1], Nan Sun[1], David Z. Pan[1]

[1] ECE Dept., the University of Texas at Austin

[2] CSE Dept., the Chinese University of Hong Kong

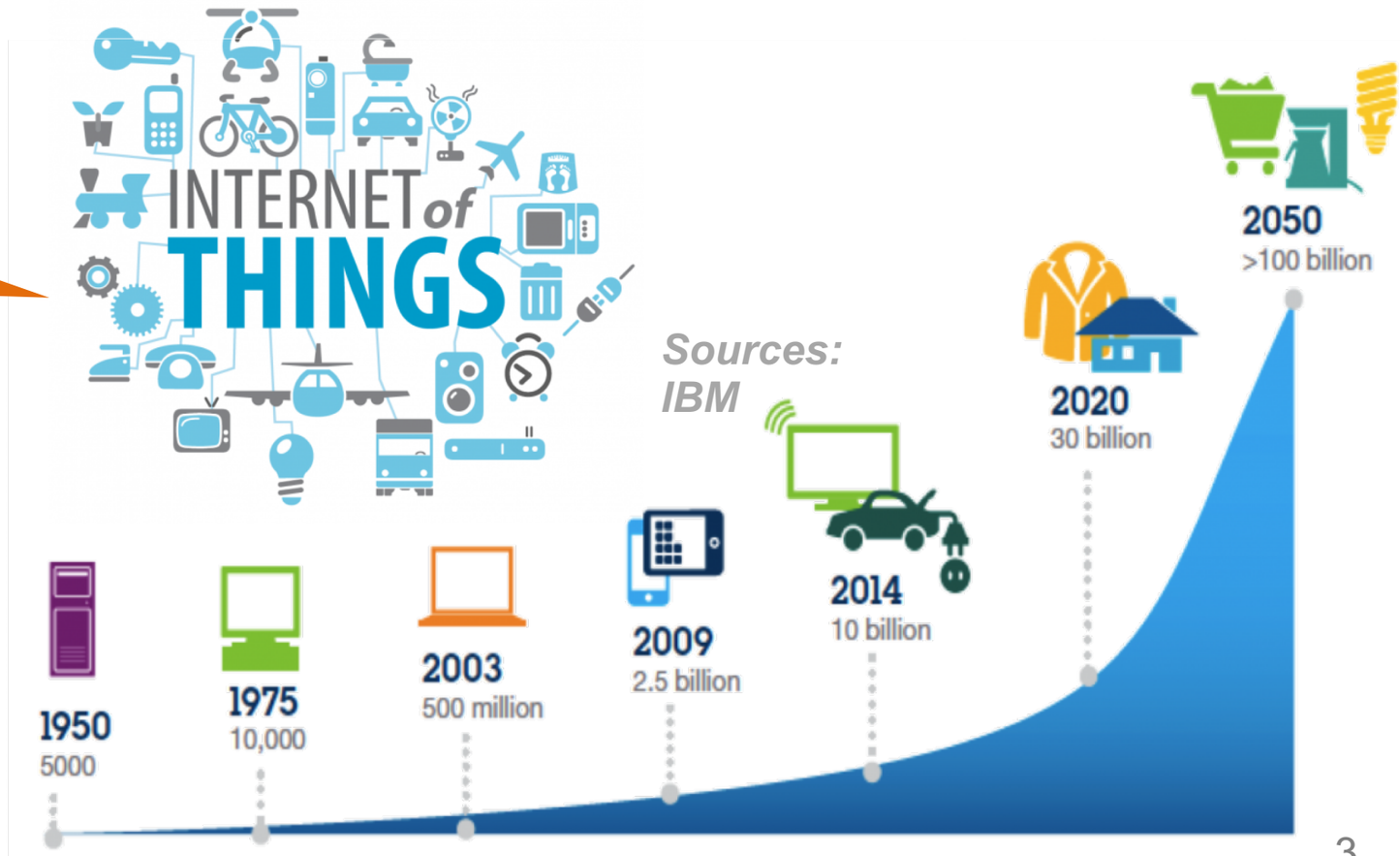
[3] Cadence Design Systems, Inc.

Outline

- Introduction of Device Layer-Aware Analog Placement
- Device Layer-Aware Analog Placement
  - ✓ Non-linear optimization based global placement
  - ✓ Linear programming based legalization and detailed placement
- Experimental Results
- Summary

Analog IC Trend

- High demand in emerging applications: Internet of Things (IOT), autonomous and electric vehicles, communication and 5G networks

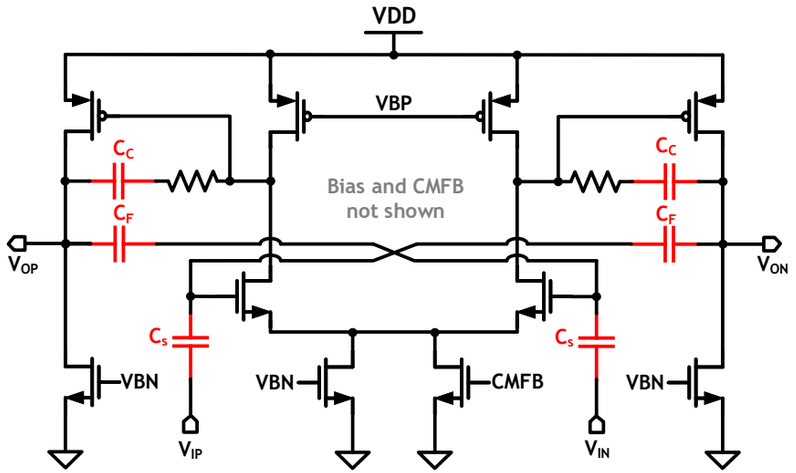


Analog Layout Automation Challenges

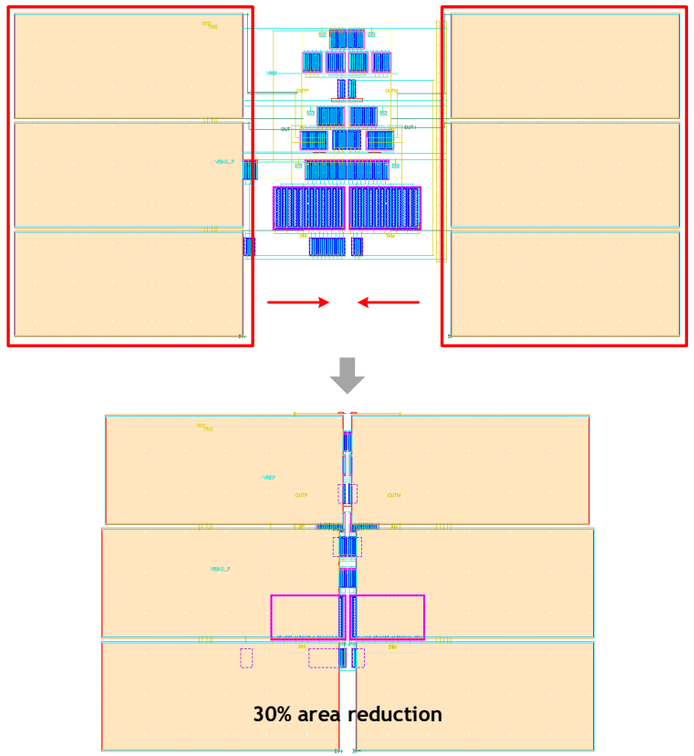
- Modern SoCs: 20% or less analog, but maybe over 80% design time
- Analog IC layout design still heavily manual
  - ✓ Cf. digital IC layout automation
  - ✓ Very time-consuming, tedious, and error-prone
- Some prior work on analog placement
  - ✓ [Lampaert+, JSSC'95], [Strasser+, ICCAD'08], [Ma+, TCAD'11], [Wu+, ICCAD'12], [Lin+, TCAD'16], [Ou+, TCAD'16]
- Limitations of previous approaches
  - ✓ Efficiency and scalability issues for stochastic or enumerative approaches
  - ✓ Still limited to consider complex scenarios and characteristics unique to analog designs, which can contribute to better layout quality

Introduction

- Analog circuits often contain different types of devices
- Sometimes overlaps are allowed and beneficial
- Circuit example: capacitive-coupled OTA
  - ✓ 30% and 4% area and wirelength reductions, respectively



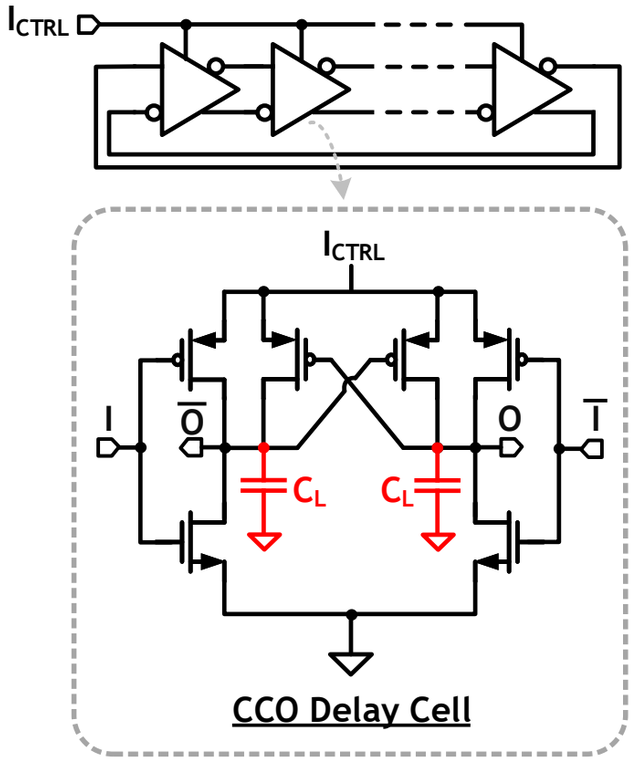
Layout	Phase Margin (deg.)	Unity Gain Bandwidth (MHz)	Loop Gain (dB)
Non-overlap	71.9	103.7	36.3
overlap	71.5	105.4	36.3

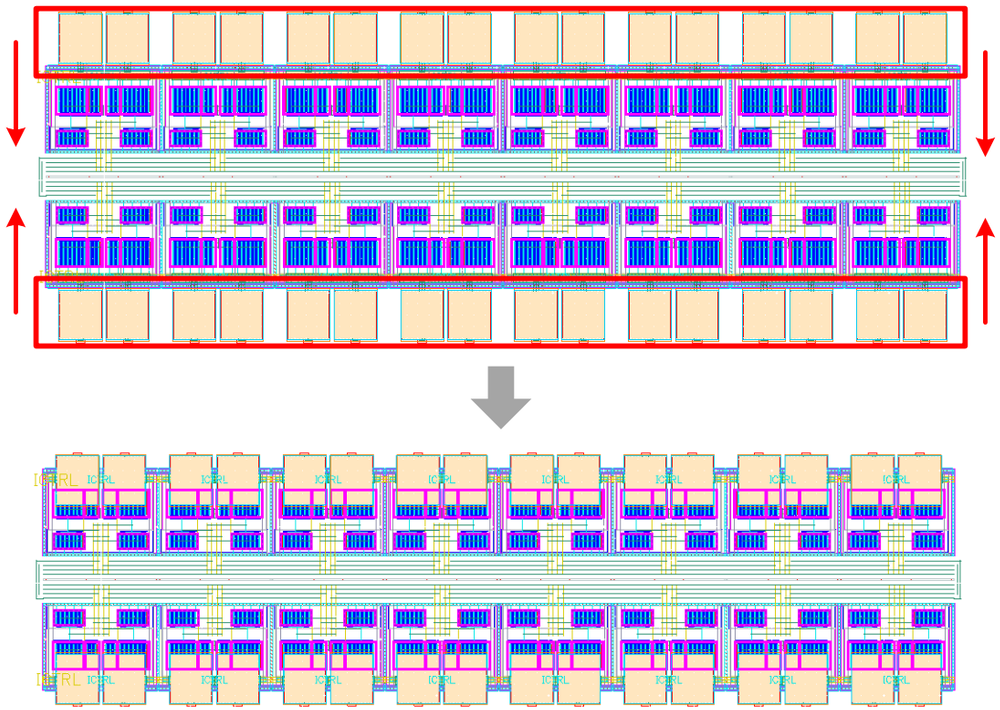


Introduction

- Circuit example: current-controlled ring oscillator (CCO)
  - ✓ 30% and 20% area and wirelength reductions, respectively

Layout	Center Frequency (f_{CCO}) (kHz)	Tuning Gain (k_{CCO}) (THz/A)
Non-overlap	609	0.89
overlap	610	0.90





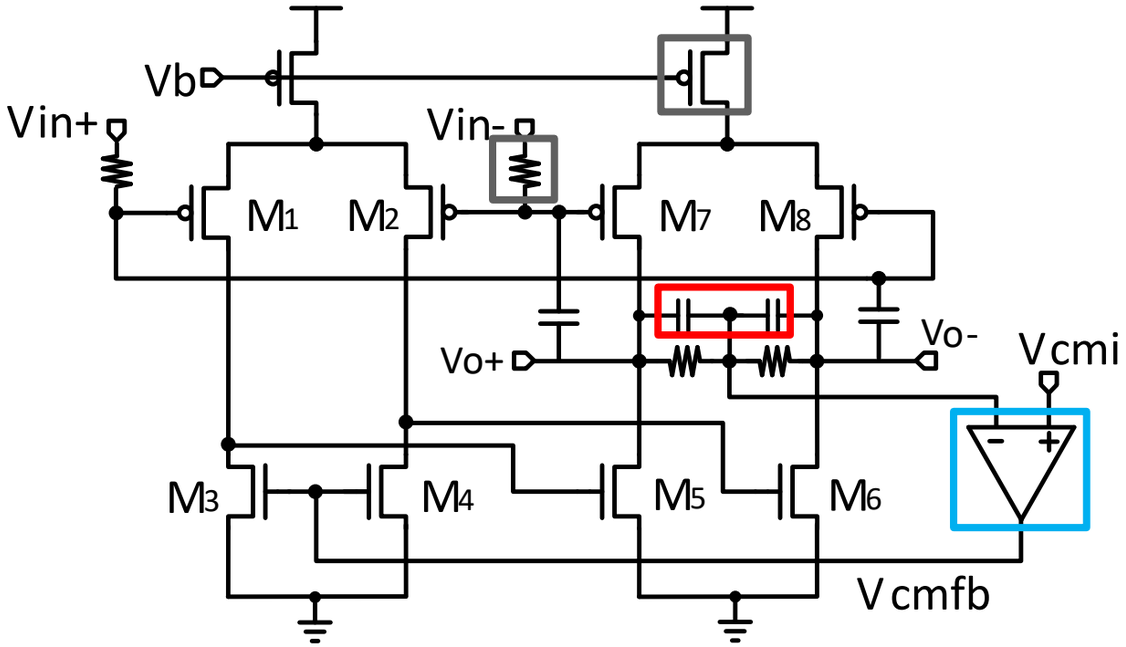
30% area reduction

Our Contributions

- Consider device-specific overlapping in analog circuits during placement, which offers high flexibility for layout optimization
  - ✓ Devices that are insensitive to coupling and built on mutually exclusive layers are allowed to overlap
- A holistic analytical framework to solve the device layer-aware analog placement problem
- An analog global router is developed to verify the routability of our device layer-aware placement results

Preliminary

- Type I devices can overlap Type II devices without degrading circuit performance
- Device types are specified by circuit designers



An analog circuit example

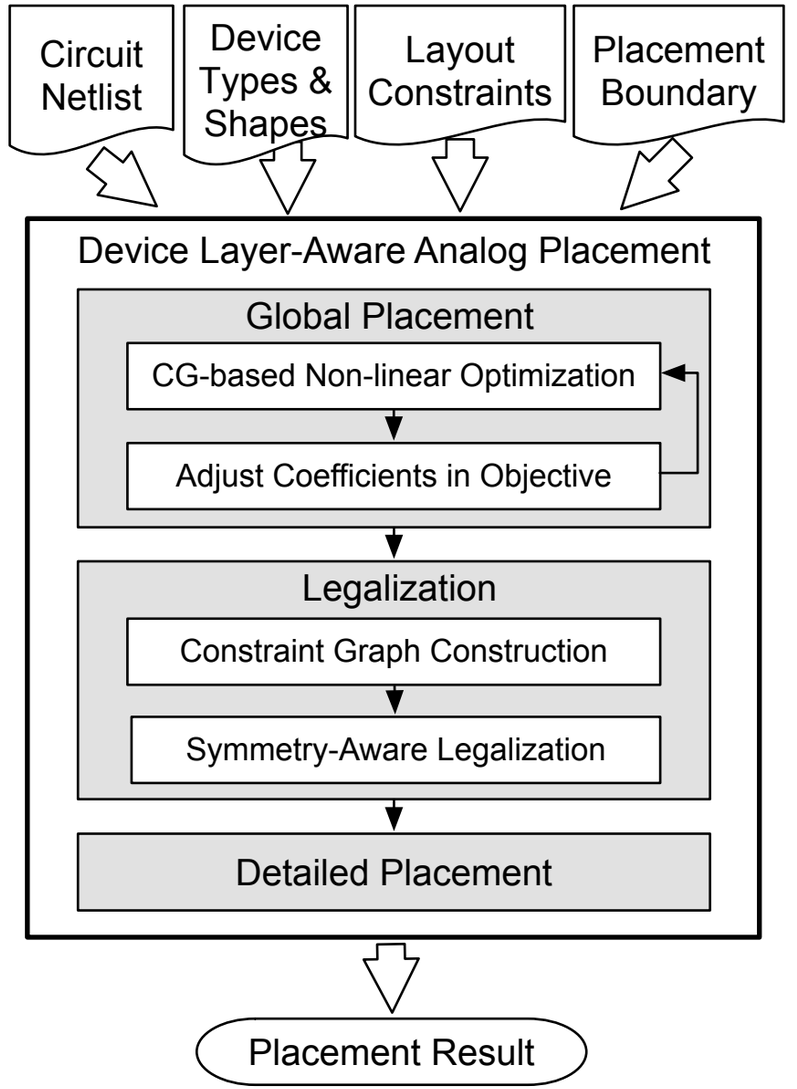
Type I devices: built without metal or via layers, and not sensitive to coupling, e.g. some transistors and resistors

Type II devices: built only with metal and via layers, and not sensitive to coupling, e.g. some metal-oxide-metal capacitors

Type III devices: occupying not only the metal and via layers but also substrate and polysilicon layers, or the device that is critical and sensitive to coupling, e.g. some pre-laid-out sub-circuits or sensitive devices

Device Layer-Aware Analog Placement

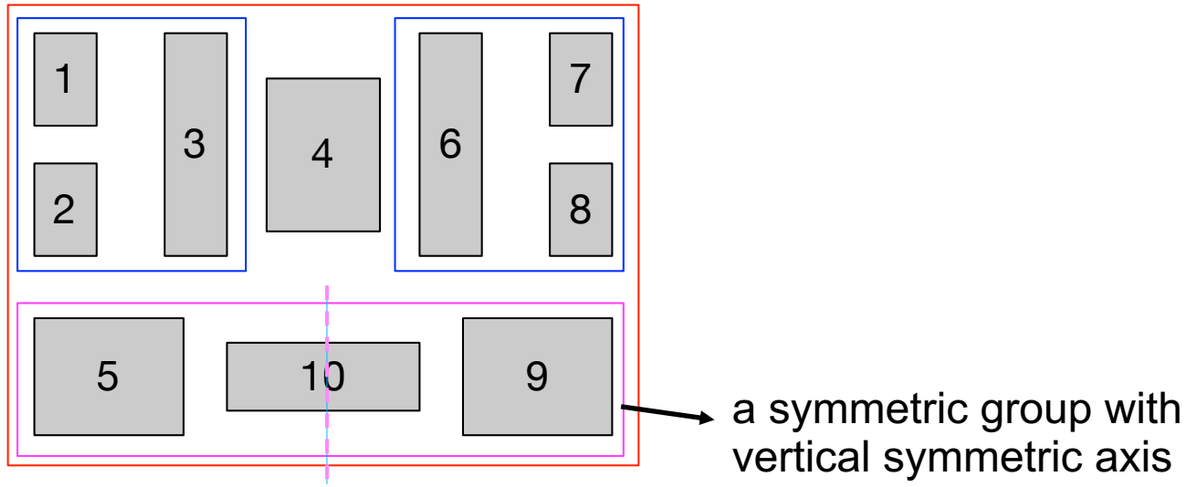
- Inputs:
  - ✓ Circuit netlist
  - ✓ Device sizes and designer specified device types
  - ✓ Analog layout constraints (e.g., symmetry)
  - ✓ Placement boundary (as generated from desired utilization rate and aspect ratio)
- Output: a legal placement solution
- Objectives:
  - ✓ Total area
  - ✓ Total wirelength



Overall flow

Device Layer-Aware Analog Placement

- Constraints:
  - ✓ Symmetric device group shares a common symmetric axis in the placement



An analog placement example

  - ✓ Device-specific overlapping constraints:
    - Devices built by mutually exclusive manufacturing layers and insensitive to coupling are allowed to overlap each other; while others are not
  - ✓ Placement boundary constraint

Global Placement

- We relax the constraints into penalties in the objective, and transform the problem into an unconstrained nonlinear optimization problem
- Objective:

$$Objective = f_{WL} + a \cdot f_{OL} + b \cdot f_{BND} + c \cdot \left(f_{SYM}^{x} + f_{SYM}^{y}\right)$$

- Wirelength term (half-perimeter wirelength):

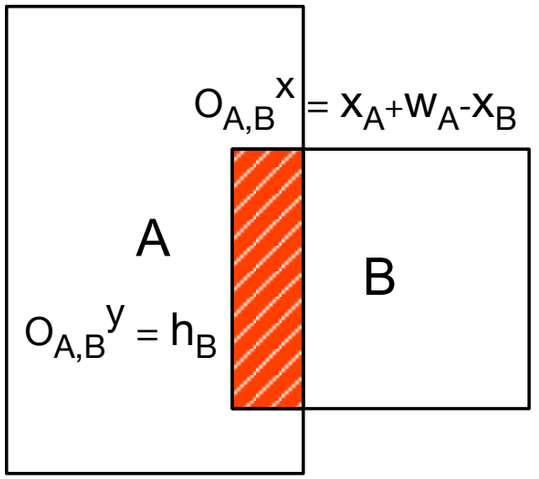
$$f_{WL} = \Sigma_{n_k}\left(\max_{i \in n_k} x_i - \min_{i \in n_k} x_i + \max_{i \in n_k} y_i - \min_{i \in n_k} y_i\right)$$

- Device-specific overlap penalty:

$$f_{OL} = \sum_{(i,j) \in L} O_{i,j}^{x} \cdot O_{i,j}^{y}$$

$$O_{i,j}^{x} = \max\left(\min\left(x_i + w_i - x_j, x_j + w_j - x_i, w_i, w_j\right), 0\right)$$

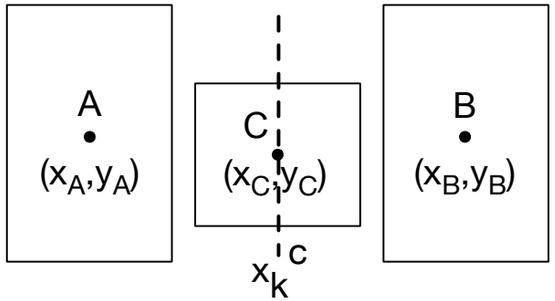
$$O_{i,j}^{y} = \max\left(\min\left(y_i + h_i - y_j, y_j + h_j - y_i, h_i, h_j\right), 0\right)$$



Global Placement

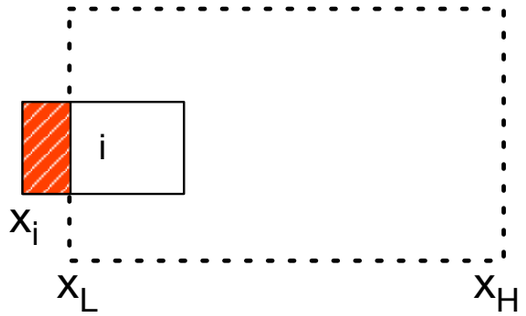
- Asymmetry penalty:

$$f_{SYM}^{x} = \sum_{g_k \in G} \left(\sum_{(i,j) \in g_k^p} \left(\left(x_i + x_j - 2 \cdot x_k^c \right)^2 + \left(y_i - y_j \right)^2 \right) + \sum_{i \in g_k^s} \left(x_i - x_k^c \right)^2 \right)$$



- Out of boundary penalty:

$$f_{BND} = \sum_{i \in D} \left(\max(x_L - x_i, 0) + \max(x_i + w_i - x_H, 0) + \max(y_L - y_i, 0) + \max(y_i + h_i - y_H, 0) \right)$$



Global Placement

- Log-sum-exp (LSE) to smooth max and min functions

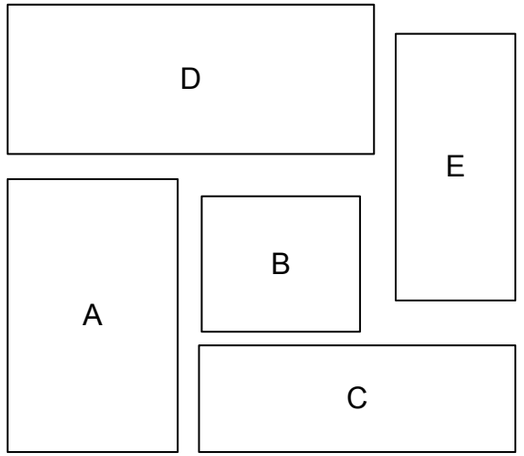
$$\text{Smoothed } \max_i x_i : \gamma \log \sum_i e^{x_i/\gamma} \qquad \text{Smoothed } \min_i x_i : -\gamma \log \sum_i e^{-x_i/\gamma}$$

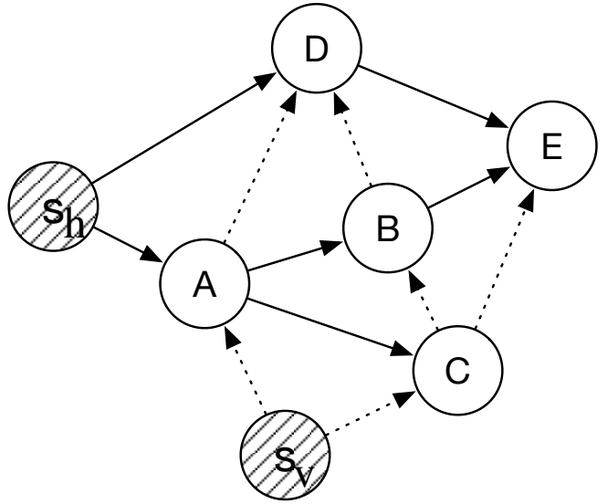
- The unconstrained nonlinear optimization problem is solved with nonlinear conjugate gradient method provided by WNLIB
- Iteratively update the weights of different terms in the objectives
  - ✓ The weight of the wirelength term is larger than other weights at the beginning
  - ✓ Weights of other terms are increased gradually, until the penalties are below certain thresholds
  - ✓ The algorithm stops when all the penalties are below the preset thresholds, or after it reaches the preset max. #iterations

WNLIB: W. Naylor and B. Chapman, http://www.willnaylor.com/wnlib.html

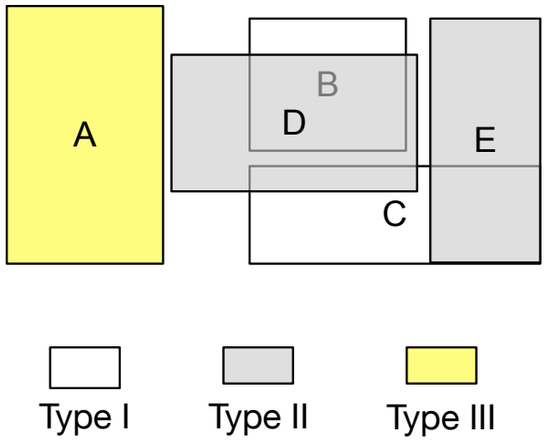
Legalization

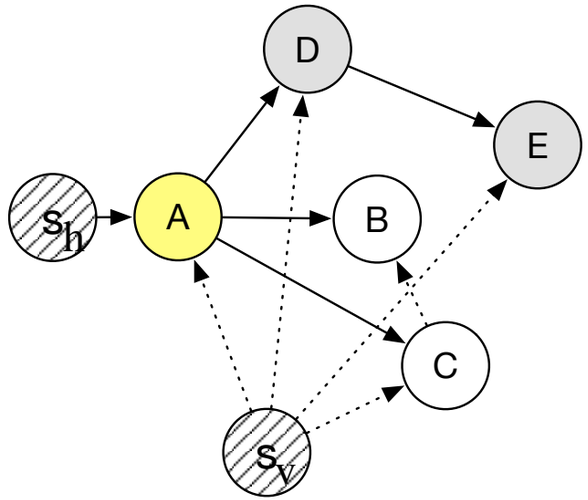
- Constraint graph construction
  - ✓ Plane sweep algorithm [Doenhardt+, TCAD'87]
  - ✓ Solid edges: horizontal; dashed edges: vertical



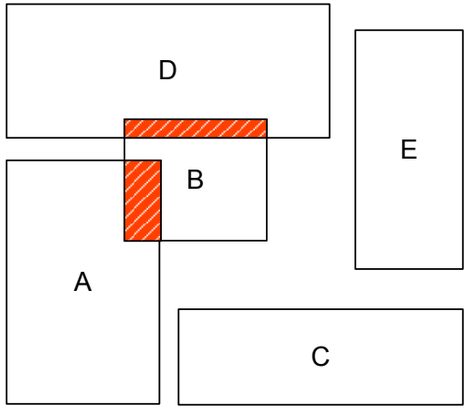


- Device layer-aware constraint graph construction

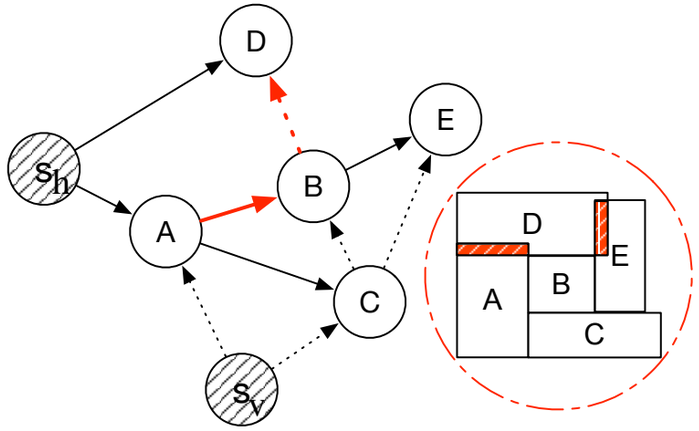




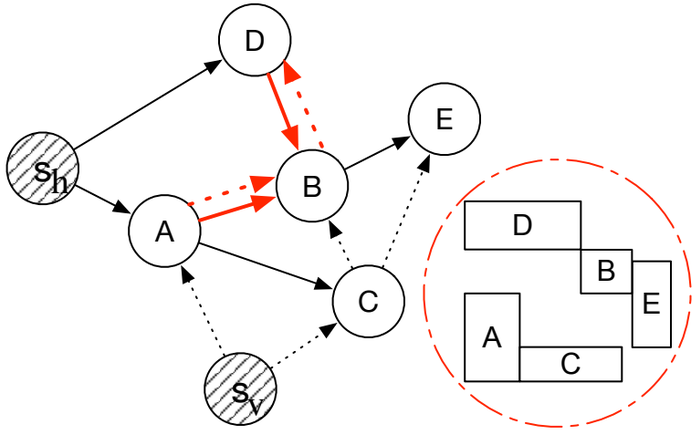
Legalization



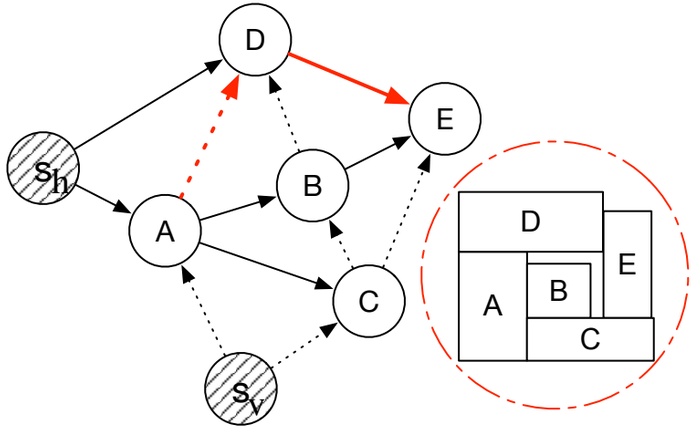
(a) Global placement result example w/ illegal device overlaps



(c) Constraints graphs after greedily determining overlap edges



(b) Constraint graphs after applying plane sweep algorithm



(d) Constraints graphs after missing positional relationship detection

Legalization

- Linear programming (LP)-based legalization to minimize area
- Decomposed into x- and y- direction sub-problems and solved independently

$$\text{Minimize} \quad W$$

$$\text{Subject to} \quad 0 \le x_i \le W - w_i, \forall i \in D,$$

$$x_i + w_i \le x_j, \forall e_{i,j} \in G_h,$$

$$x_i + x_j + w_j = 2 \cdot x_k^c, \forall (i,j) \in g_k^p,$$

$$2 \cdot x_i + w_i = 2 \cdot x_k^c, \forall i \in g_k^s,$$

$$\forall g_k \in G,$$

Boundary constraints

Topology order constraints

Symmetry constraints

$$\text{Minimize} \quad H$$

$$\text{Subject to} \quad 0 \le y_i \le H - h_i, \forall i \in D,$$

$$y_i + h_i \le y_j, \forall e_{i,j} \in G_v,$$

$$y_i = y_j, \forall (i,j) \in g_k^p, \forall g_k \in G,$$

Boundary constraints

Topology order constraints

Symmetry constraints

Detailed Placement

- LP-based wirelength refinement

$$
\begin{aligned}
&\text{Minimize} && Wirelength \\
&\text{Subject to} && 0 \le x_i \le W^* - w_i, \forall i \in D, \\
& && x_i + w_i \le x_j, \forall e_{i,j} \in G_h, \\
& && 0 \le y_i \le H^* - h_i, \forall i \in D, \\
& && y_i + h_i \le y_j, \forall e_{i,j} \in G_v, \\
& && x_i + x_j + w_j = 2 \cdot x_k^c, \forall (i,j) \in g_k^p, \\
& && 2 \cdot x_i + w_i = 2 \cdot x_k^c, \forall i \in g_k^s, \qquad \forall g_k \in G, \\
& && y_i = y_j, \forall (i,j) \in g_k^p,
\end{aligned}
$$

Fixed boundary and topology order constraints

Symmetry constraints

Experimental Results

- All algorithms are implemented in C/C++
- All experiments are performed on a Linux machine with 3.4GHz Intel(R) core and 32GB memory.
- Benchmark information

Design	# Devices	# Type I Devices	# Type II Devices	# Type III Devices	# Nets
opamp	46	42	4	0	29
g_m-C integrator	15	13	2	0	9
CTDSM	21	6	2	13	27

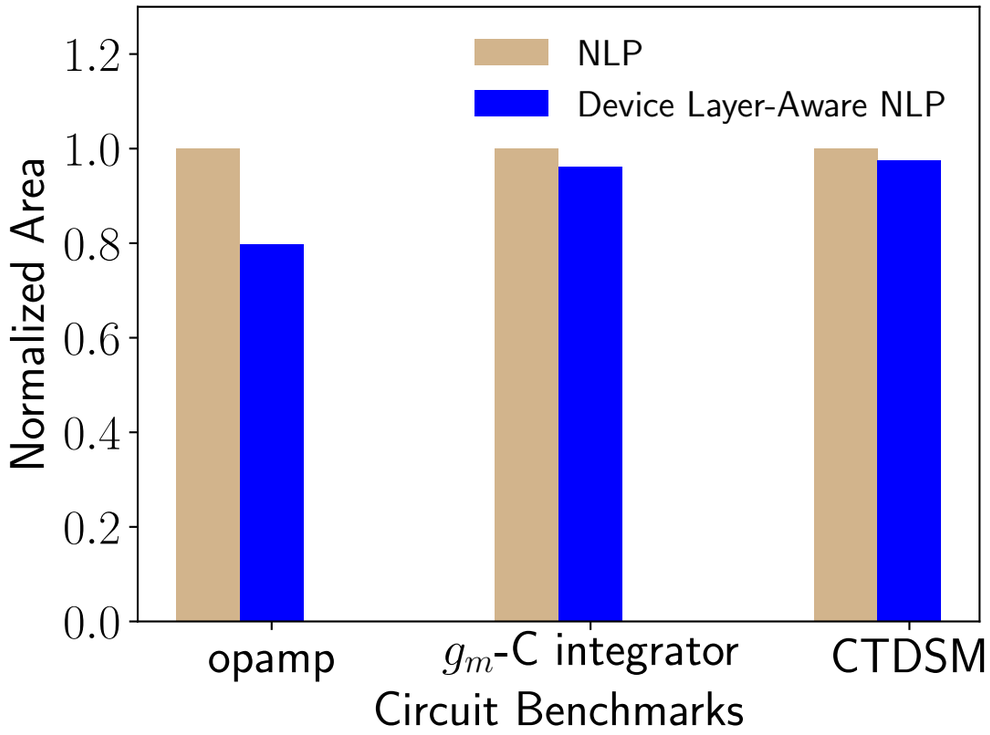
Experimental Results

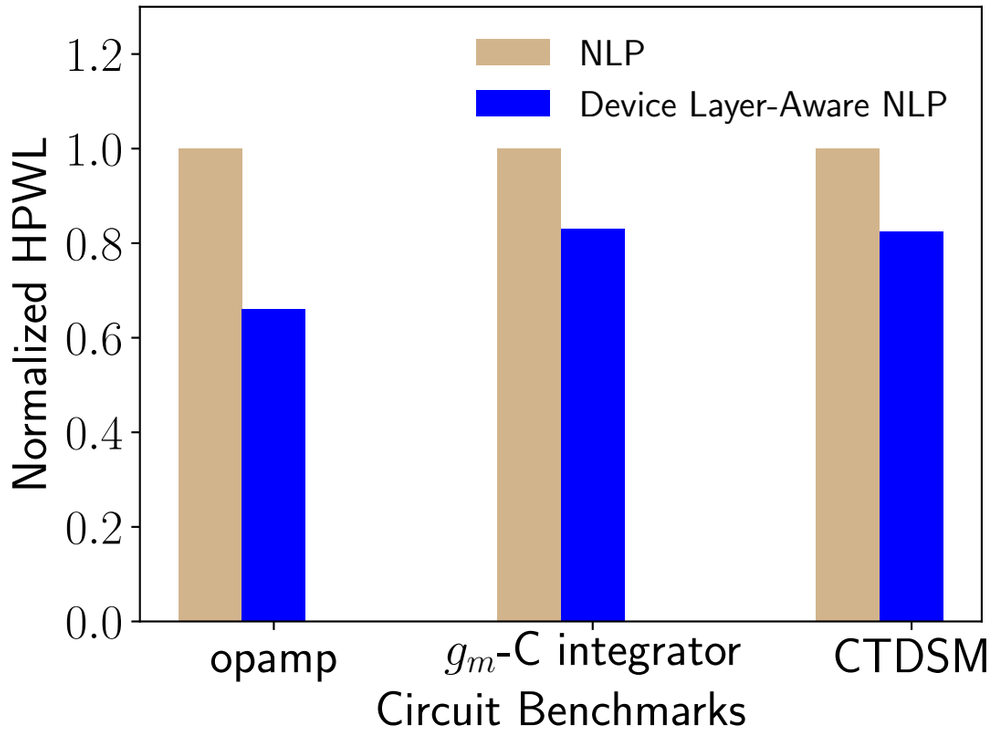
- Compare effects of device layer awareness

Design	NLP Without Device Layer Awareness						Device Layer-Aware NLP					
	Area (μm^2)		HPWL (μm)		Run-time (s)		Area (μm^2)		HPWL (μm)		Run-time (s)	
	Actual	Norm.	Actual	Norm.	Actual	Norm.	Actual	Norm.	Actual	Norm.	Actual	Norm.
opamp	2972.7	1	753.2	1	17.1	1	2369.9	**0.797**	497.7	**0.661**	10.9	0.637
g_m-C integrator	182.0	1	72.8	1	1.2	1	175.1	**0.962**	60.5	**0.831**	1.2	1.000
CTDSM	57454.5	1	3129.4	1	6.5	1	56059.8	**0.976**	2580.0	**0.824**	6.5	0.997
Average		1		1		1		**0.912**		**0.772**		0.878

Experimental Results

- 9% and 23% area and HPWL reductions, respectively, when considering device-specific overlapping

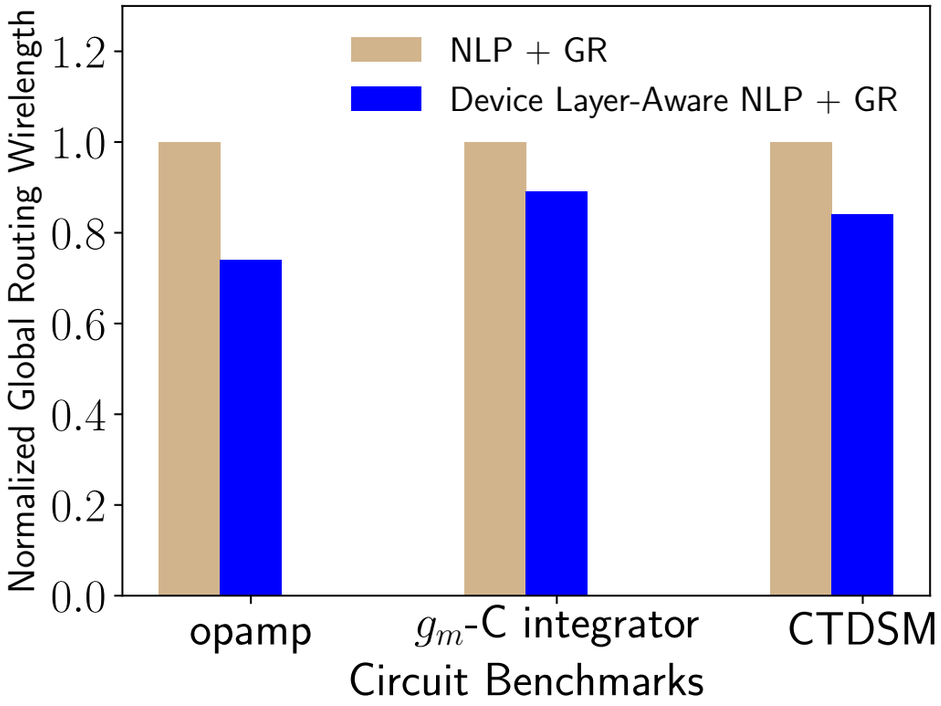




Experimental Results

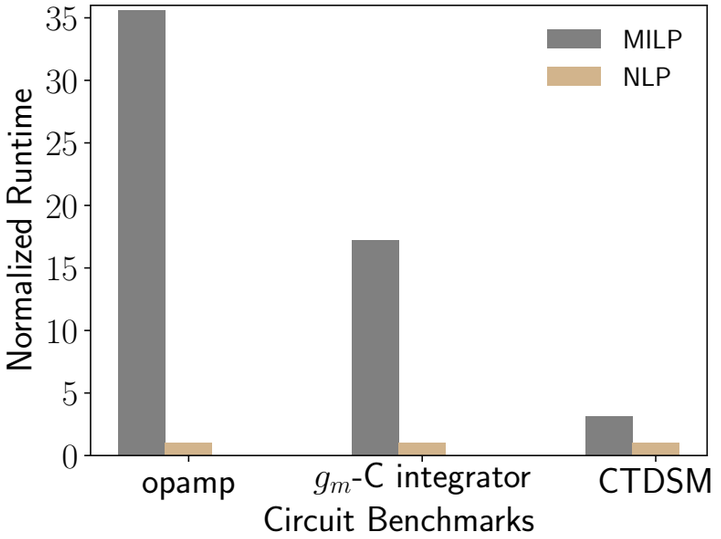
- To evaluate routing congestion, we develop a maze routing based analog global router
- Compare global routing results with and without device layer-awareness
- 18% global routing wirelength reduction

Wirelength	NLP+GR		Device layer-aware NLP+GR	
	Actual	Norm.	Actual	Norm.
opamp	839	1	617	0.74
g_m-C integrator	89	1	79	0.89
CTDSM	3591	1	3034	0.84
Average		**1**		**0.82**



Experimental Results

- Compare with previous MILP-based analog placement [Xu+, ISPD'17]
- 18X speedup



Design	NLP-based Placement						MILP-based Placement [Xu+, ISPD'17]					
	Area (μm²)		HPWL (μm)		Run-time (s)		Area (μm²)		HPWL (μm)		Run-time (s)	
	Actual	Norm.	Actual	Norm.	Actual	Norm.	Actual	Norm.	Actual	Norm.	Actual	Norm.
opamp	2972.7	1	753.2	1	17.1	1	3295.5	1.11	714.0	0.95	607.2	**35.57**
g_m-C integrator	182.0	1	72.8	1	1.2	1	186.8	1.03	69.0	0.95	20.7	**17.21**
CTDSM	57454.5	1	3129.4	1	6.5	1	57959.9	1.01	3611.5	1.15	20.5	**3.16**
Average		1		1		1		1.05		1.02		**18.65**

Summary

- We have introduced the device layer-aware analog placement problem
- A holistic analytical analog placement framework is proposed for the device layer-aware analog placement
  - ✓ Non-linear optimization based global placement
  - ✓ LP-based legalization and detailed placement
  - ✓ Symmetry constraints in analog layout are honored
- Experimental results show that our framework is both efficient and effective

Thank you!

- Questions?

Backup slides

Legalization - Constraint Graph Construction

Algorithm 1 Missing Positional Relationships Detection

Input: n devices and their vertical and horizontal constraint graphs G_v, G_h

Output: Find all the missing relationships of the devices

```
1:  let M_h, M_v be two (n + 1) × (n + 1) Boolean matrices;
2:  let s_h, s_v be the source of G_h, G_v respectively;
3:  DFS(G_h, s_h, M_h)
4:  DFS(G_v, s_v, M_v)
5:  for i = 0 to n − 1 do
6:      for j = i + 1 to n − 1 do
7:          if ¬(M_h[i][j]∨M_h[j][i]) ∧ ¬()M_v[i][j]∨M_v[j][i]) then
8:              no positional relationship between devices i, j;
9:          end if
10:     end for
11: end for

12: function DFS(G, v, M)
13:     label v as discovered;
14:     M[v][v] = 1;
15:     for all edges from v to w in G.adjacentEdges(v) do
16:         if vertex w is not labeled as discovered then
17:             DFS(G, w, M)
18:         end if
19:         if M[v] is not all 1 then
20:             for i = 0 to n − 1 do
21:                 M[v][i] = M[v][i] ∨ M[w][i]
22:             end for
23:         end if
24:     end for
25: end function
```