

# Interconnect Delay and Area Estimation for Multiple-Pin Nets

Jason Cong and David Z. Pan

UCLA Computer Science Department  
Los Angeles, CA 90095



Sponsored by SRC and Avant! under CA-MICRO

# Presentation Outline

- Introduction
- Problem Formulation
- Interconnect Delay and Area Estimation  
Models for Multiple-Pin Nets
- Application and Conclusion

# Interconnect Optimization

- **UCLA TRIO (Tree-Repeater-Interconnect-Optimization)** package [Cong et al, ICCAD'97] (as an example)
  - ◆ Topology construction
  - ◆ Optimal buffer insertion
  - ◆ Cell (driver/buffer/repeater) sizing
  - ◆ Wire sizing and spacing
  - ◆ ...
- Timing can be improved significantly (e.g., a factor of 10x)!
- The earlier, the better => **timing convergence**

# Complexity of Existing Interconnect Optimization Algorithms

## ■ Mainly iterative based

### ◆ Dynamic Programming (DP):

[van Ginneken, ISCAS'90], [Lillis et al., JSSC'96] ...

### ◆ Local Refinement (LR)

[Cong-Leung, TCAD'94], [Cong-He, ICCAD'96] ...

### ◆ Mathematical Programming (MP):

[Fishburn-Dunlop, ICCAD'85], [Sapatnekar et al, TCAD'93],  
[Menezes et al., ICCAD'95] ...

## ■ Although in polynomial time complexity, they are not suitable for high-level planning/synthesis:

### ◆ too expensive

### ◆ lack of details at high levels

### ◆ .....

# CPU Matters

- Interconnect optimization for **one** net takes about 0.1 to 10+ seconds [Cong et al., ICCAD'97]
- [Keutzer, TAU'99]
  - ◆ 80,000 to 200,000 global nets
  - ◆ 100 to 100,000 iterations between synthesis and PD => hopefully timing convergence
- Take a typical scenario:
  - ◆ 100,000 global nets
  - ◆ 1,000 iterations
  - ◆ 1 second to optimize each net
- => 100,000,000 second = 3 years !

# Needs for Efficient Interconnect Estimation Models

- **Efficiency**
- **Abstraction** to hide detailed design information
  - ◆ granularity of wire segmentation
  - ◆ number of wire widths, buffer sizes, ...
- **Explicit relation** (such as closed-form formula) to enable optimal design decision at high levels
- **Ease of interaction** with high level tools

# Previous Work on Interconnect Delay Estimation

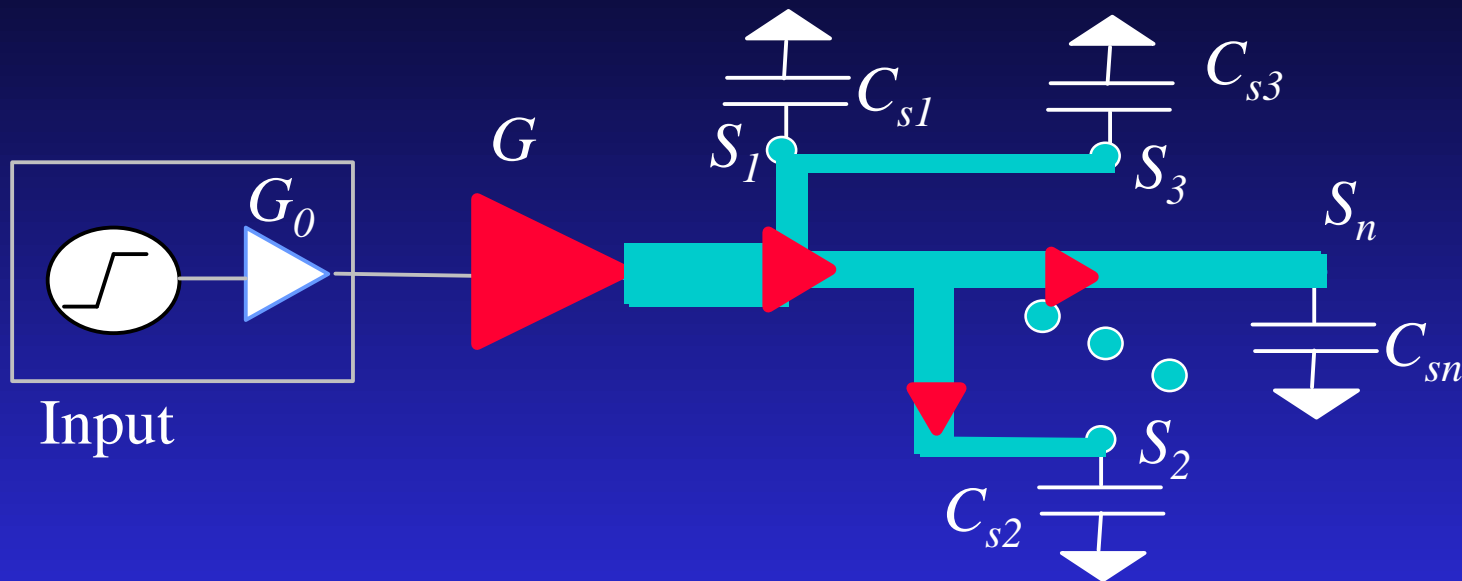
- Simple RC model **with uniform (min.) wire width:**
  - ◆ Wire delay  $\mu l^2$  : [Ramachandran et al., ICCAD-92]
  - ◆ With buffer insertion: [Bakoglu-90] [Alpert-Devgan DAC'97] [Brayton-Otten, DAC'98]
- Distributed RC model **with optimal wire sizing:**  
[Cong-Pan, IWLS'98, ASP-DAC'99] => a set of delay estimation models (**DEM**) with interconnect optimization
  - ◆ Optimal Wire Sizing (**OWS**): sub-quadratic function of length
  - ◆ Simultaneous Driver and Wire Sizing (**SDWS**)
  - ◆ Simultaneous Buffer Insertion and Wire Sizing (**BIWS**)
  - ◆ Simultaneous Buffer Insertion/Sizing and Wire Sizing (**BISWS**)
- Limitations: 2-pin nets only; no area estimation

# Main Contribution of This Work

- Develop **delay and area** estimation models for **multiple-pin nets** with consideration of various interconnect optimizations
- Consider different optimization objectives
  - ◆ Single critical sink (SCS)
  - ◆ Multiple critical sinks (MCS)
- Apply various optimization alternatives:
  - ◆ OWS
  - ◆ BISWS
  - ◆ ...



# Problem Formulation



## ■ Different targets:

1. Minimize the delay to a single critical sink (SCS)
2. Minimize the maximum delay (defined as the tree delay) for multiple critical sinks (MCS)

➔ **What is the optimized delay/area?**

**Do not run TRIO or other optimization tools !**

# Parameters and Notations

■ Based on 1997 National Technology Roadmap for Semiconductors (NTRS'97)

## ■ Interconnect

- ◆  $c_a$  area capacitance coefficient
- ◆  $c_f$  fringing capacitance coefficient
- ◆  $r$  sheet resistance

## ■ Device

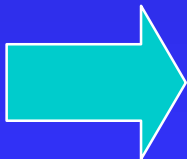
- ◆  $t_g$  intrinsic gate delay
- ◆  $c_g$  input capacitance of the minimum gate
- ◆  $r_g$  output resistance of the minimum gate

# Presentation Outline

- Introduction
- Problem Formulation
- **Interconnect Delay and Area Estimation Models for Multiple-Pin Nets**
  1. Single Critical Sink (SCS)
    - ◆ SCS/OWS
    - ◆ SCS/BISWS
  2. Multiple Critical Sink (MCS)
    - ◆ MCS/OWS
    - ◆ MCS/BISWS
- Application and Conclusion

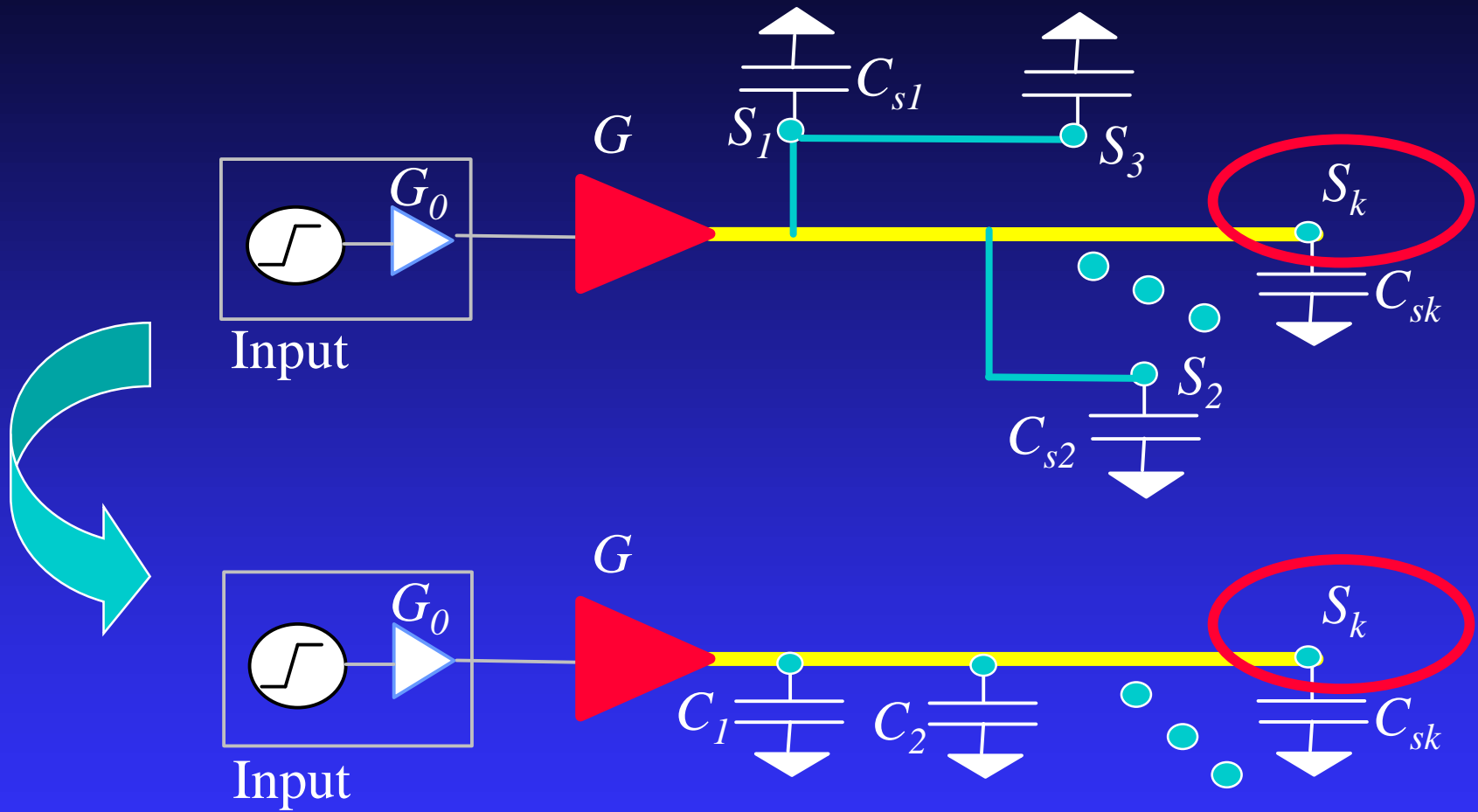
# Challenges for Multiple-Pin Net Estimation

- No closed-form wire shaping function available
- Current optimization algorithms
  - ◆ Iterative based method
    - ✦ Local refinement
    - ✦ Dynamic Programming
    - ✦ Lagrangian relaxation
    - ✦ Mathematical programming
  - ◆ Not suitable for estimation



**Key idea: transform to 2-pin net !**

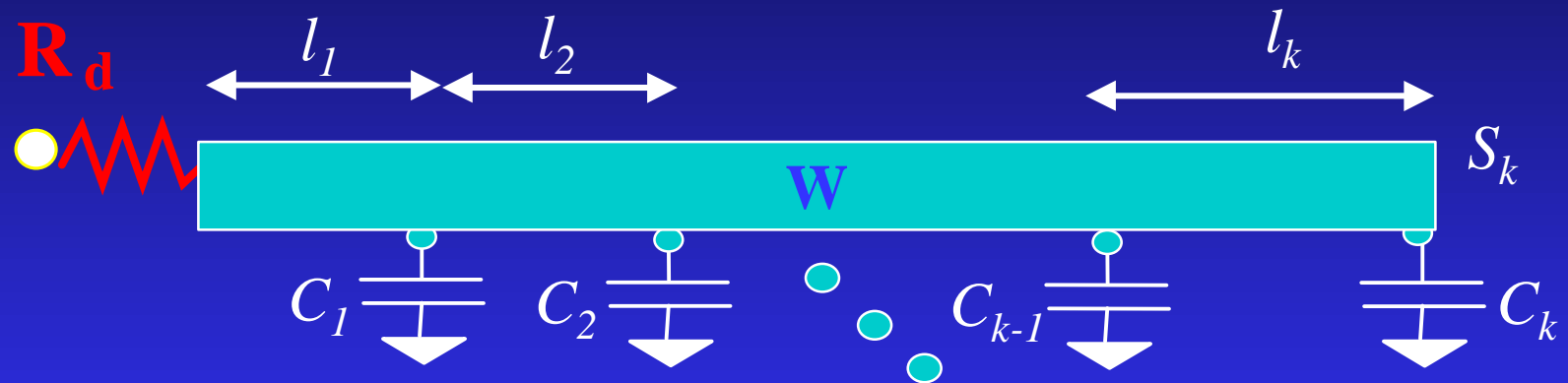
# Single Critical Sink (SCS)



**Single-Line-Multiple-Load**

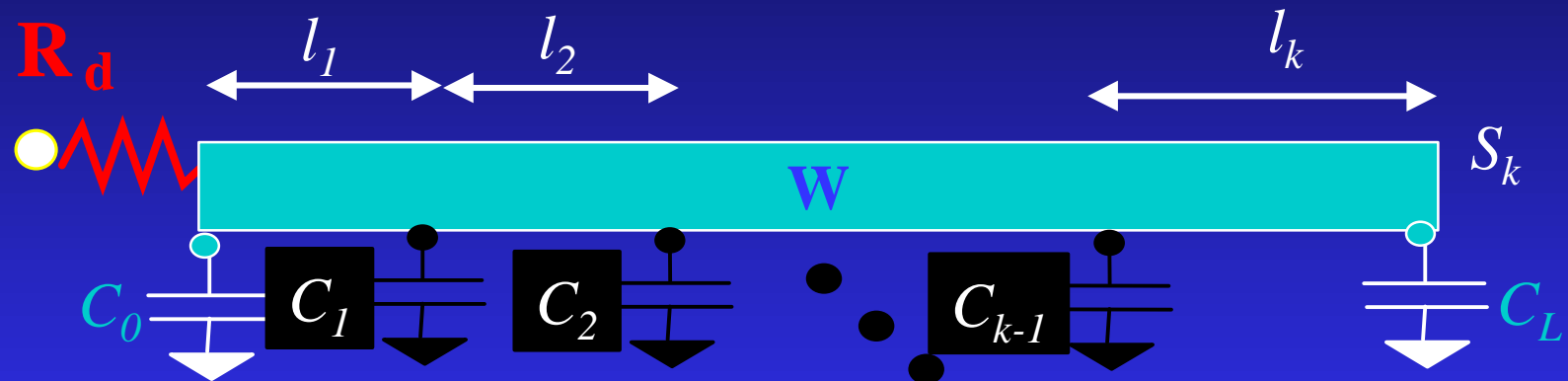
# OVS for SCS

- Transform SLML to SLSL (i.e., 2-pin net)



# OWS for SCS

- Transform SLML to SLSL (i.e., 2-pin net)



$$C_L = \sum_{j=1}^k \frac{\sum_{i=1}^j l_i}{l} \cdot C_j \quad C_0 = \sum_{j=1}^k C_j - C_L$$

# Delay/Area Estimation for SCS/OWS

- Closed-form delay estimation for the critical sink

$$T_{ows}(R_d, l, C_L) = R_d C_0 + \left[ \frac{a_1 l}{W^2(a_2 l)} + \frac{2a_1 l}{W(a_2 l)} + R_d c_f + \sqrt{R_d r c_a c_f l} \right] \cdot l$$

where

$$a_1 = \frac{1}{4} r C_a, \quad a_2 = \frac{1}{2} \sqrt{\frac{r C_a}{R_d C_L}}$$

$W(x)$  is Lambert's  $W$  function defined as  $w e^w = x$

- Closed-form area estimation for the critical path

$$A_{ows}(R_d, l, C_L) = \sqrt{\frac{r(c_f l + 2C_L)}{2R_d C_a}} \cdot l$$

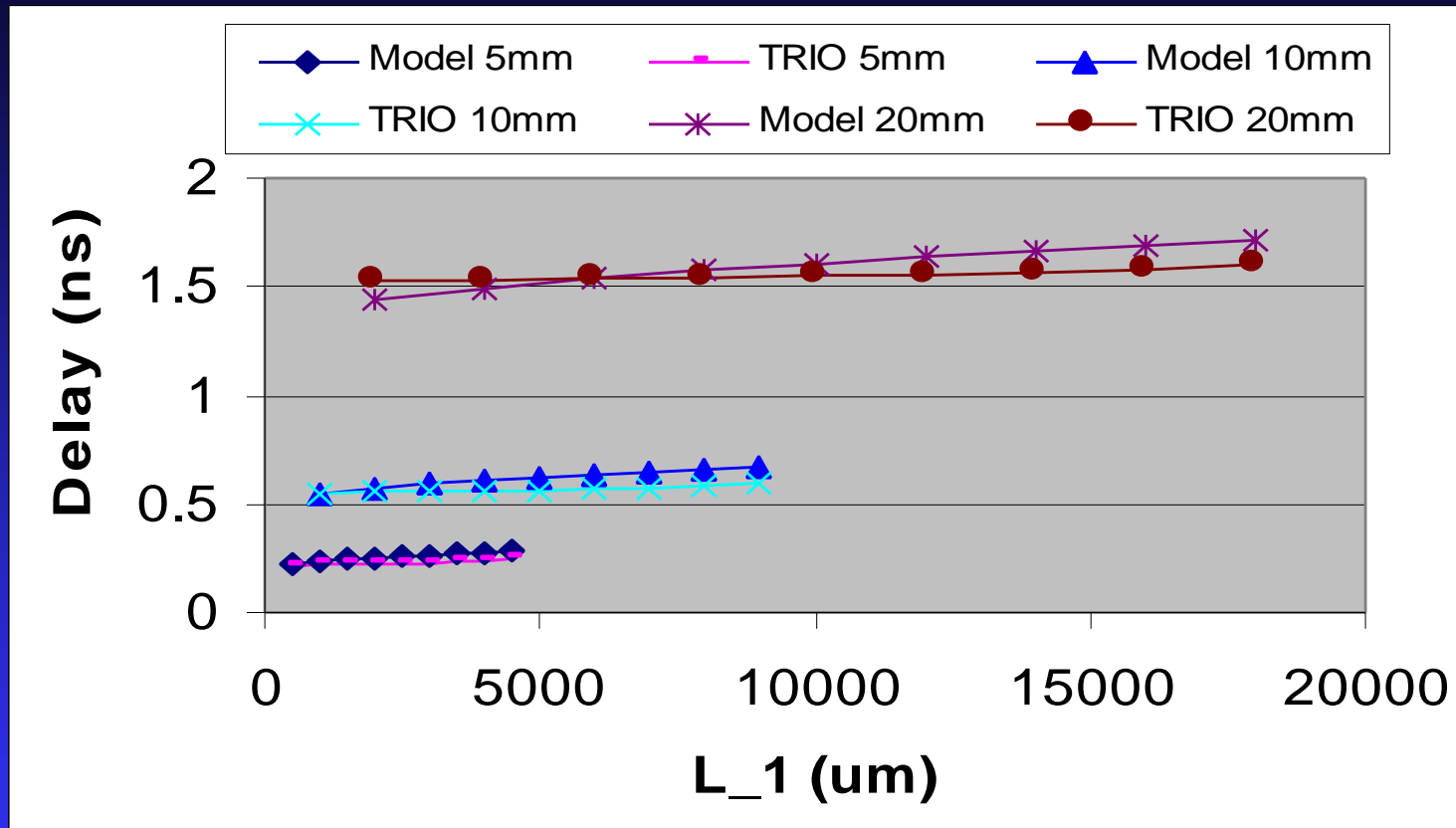


# Experimental Setting for OWS/SCS



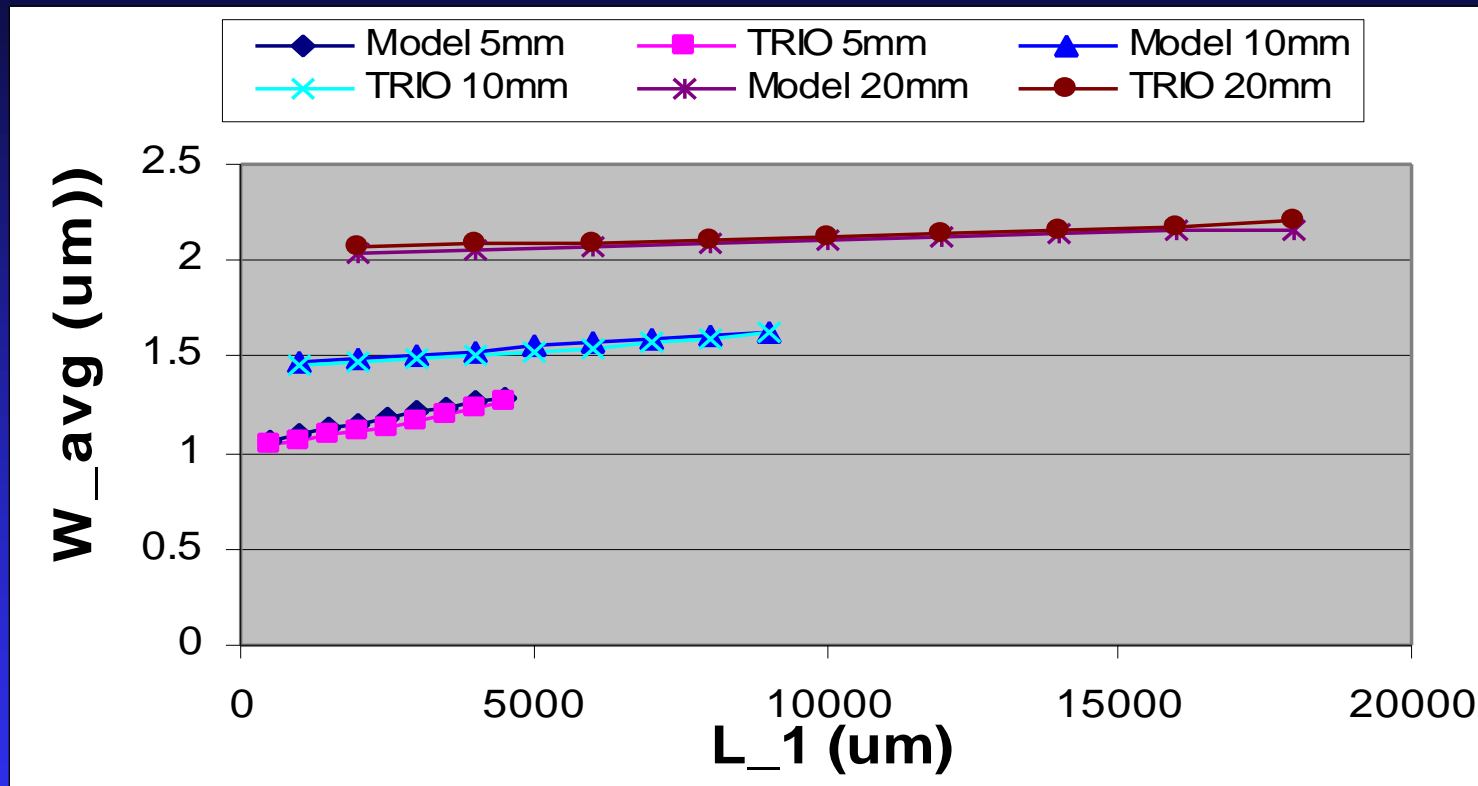
- One internal load  $C_1$
- Change  $l_1 = 0.1$  to  $0.9 \times l$
- $R_d = 180\text{ohm}$ ,  $C_1 = 100\text{ fF}$ ,  $C_2 = 10\text{ fF}$

# Delay Comparison with TRIO



- $R_d = 180\text{ohm}$ ,  $C_1 = 100\text{fF}$ ,  $C_2 = 10\text{fF}$
- Max. allowable wire width is 20x min. width; wire is segmented in every 10um.

# Avg. Width Comparison with TRIO



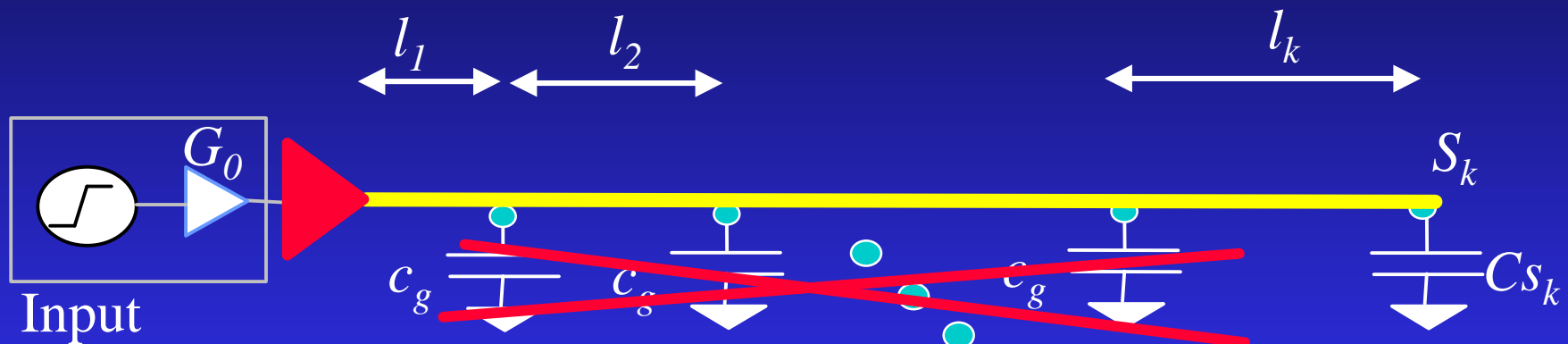
- $R_d = 180\text{ohm}$ ,  $C_1 = 100\text{fF}$ ,  $C_2 = 10\text{fF}$
- Max. allowable wire width is 20x min. width; wire is segmented in every 10 $\mu m$ .

# Run Time Comparison with TRIO

- SUN, Ultra-SPARC 1, with 256M memory
- **TRIO**: one net takes about 0.9 second, using 20 discrete wire widths, and wire segmentation of 10um (total wire length 1cm-2cm)
- **Our model**: 10,000 nets take 0.8 second
- Therefore, our model is an order of **>10,000 times faster!**
- 3 years => 3 hours

# Single Critical Sink-BISWS

- Insert min. buffer to shield non-critical sinks
- Transform into a simple SLML problem



- SLML  $\Rightarrow$  SLSL
- Use previous 2-pin net results to estimate delay and area on the critical path

# BISWS for SCS

- Linear delay model for the critical path

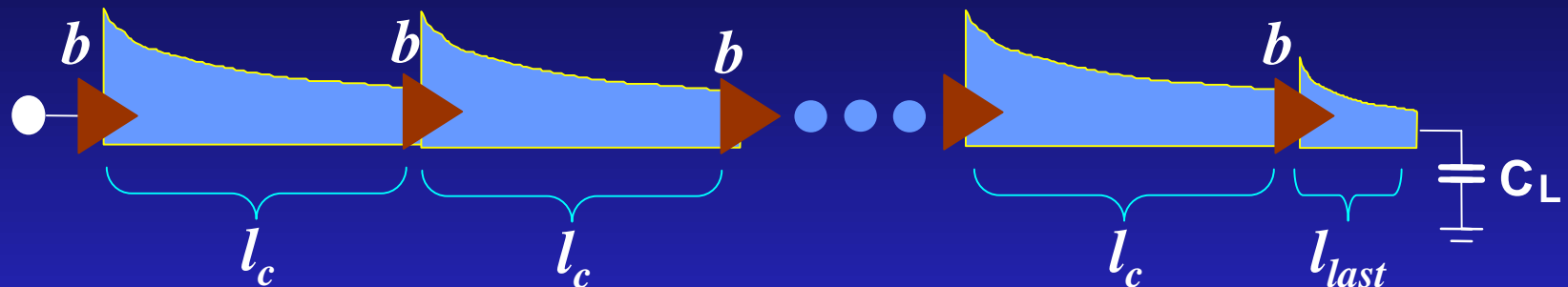
$$T_{bisws} = t_{bisws} \cdot l + t_g$$

where  $t_{bisws} = \min_{b \in B} t_{biws}$ ,  $B$  is the buffer set

- Essentially the best BIWS from available buffer types
- Complexity  $O(|B|)$ . Since the set  $B$  is normally less than 20, constant time in practice.

# BIWS for 2-Pin Nets

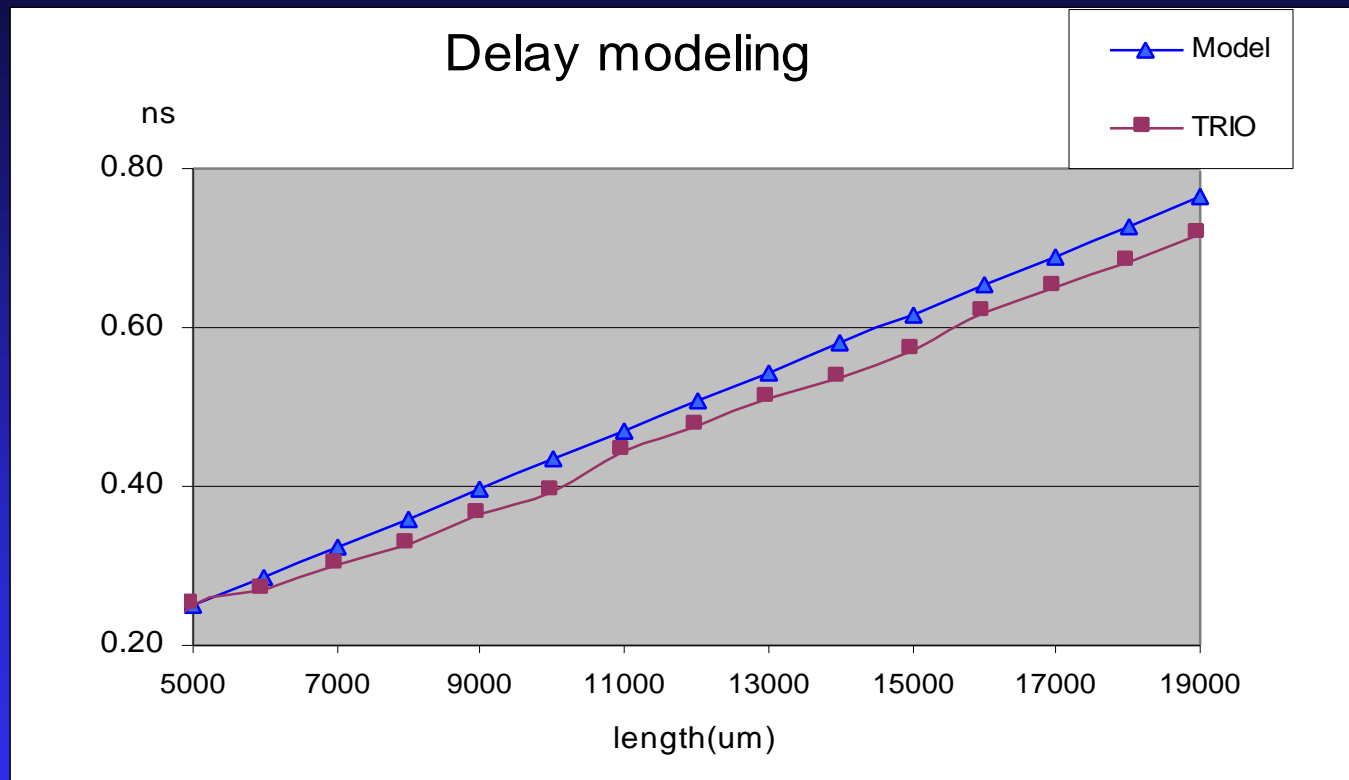
[Cong-Pan, ASP-DAC'99]



$$T_{biws} = t_{biws} \cdot l + t_g$$

$t_{biws}$  is the slope, and can be obtained from  
 $T_{ows}(R_b, l_c, C_b)$

# SCS/BISWS: Comparison with TRIO



- $0.18\mu\text{m}$ ,  $R_{d0} = r_g/10$ ,  $C_L = c_g \times 10$ ,
- TRIO uses max. buffer size of  $400 \times \text{min}$ , wire width of  $20 \times \text{min}$ . width; wire is segmented in every  $500\mu\text{m}$ .



# Presentation Outline

- Introduction
- Problem Formulation
- Interconnect Delay and Area Estimation  
Models for Multiple-Pin Nets
  1. Single Critical Sink (SCS)
    - ✦ SCS/OWS
    - ✦ SCS/BISWS
  2. Multiple Critical Sink (MCS)
    - ✦ MCS/OWS
    - ✦ MCS/BISWS
- Application and Conclusion

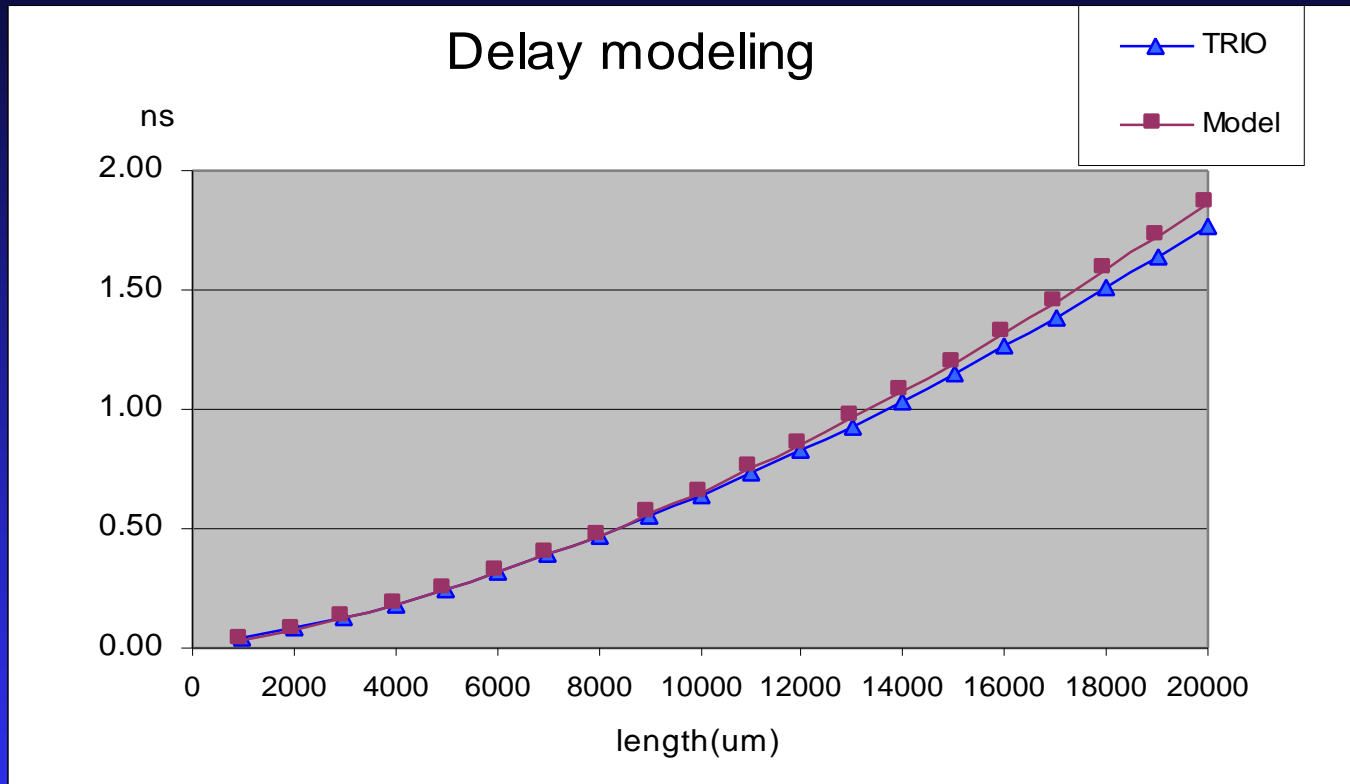
# Multiple Critical Sinks (MCS)

- **Optimization objective:** the maximum delay to all critical sinks, i.e. the tree delay
- **Key idea:** transform MCS to a sequence of SCS
- **Theorem:** The most critical sink with max delay must be a leaf critical sink.
- **Theorem:** The optimal delay to any critical sink under SCS formulation is a lower bound for the optimal tree delay.

# Multiple Critical Sinks/OWS

- **Key observation:** take the **maximum delay** of all **leaf** critical sinks under SCS formulation  
=> accurately estimate the optimal tree delay
- **Justification:** we shall keep wire load from less critical sinks as small as possible. To the most critical sink, the main difference is
  - ◆ (A) 'minimum width' under SCS formulation
  - ◆ (B) 'as small as possible width' under MCS formulation
  - ◆ In DSM, area capacitance is relatively small (cf. fringing + coupling cap.) => Two wire loads (A) and (B) differ not much.

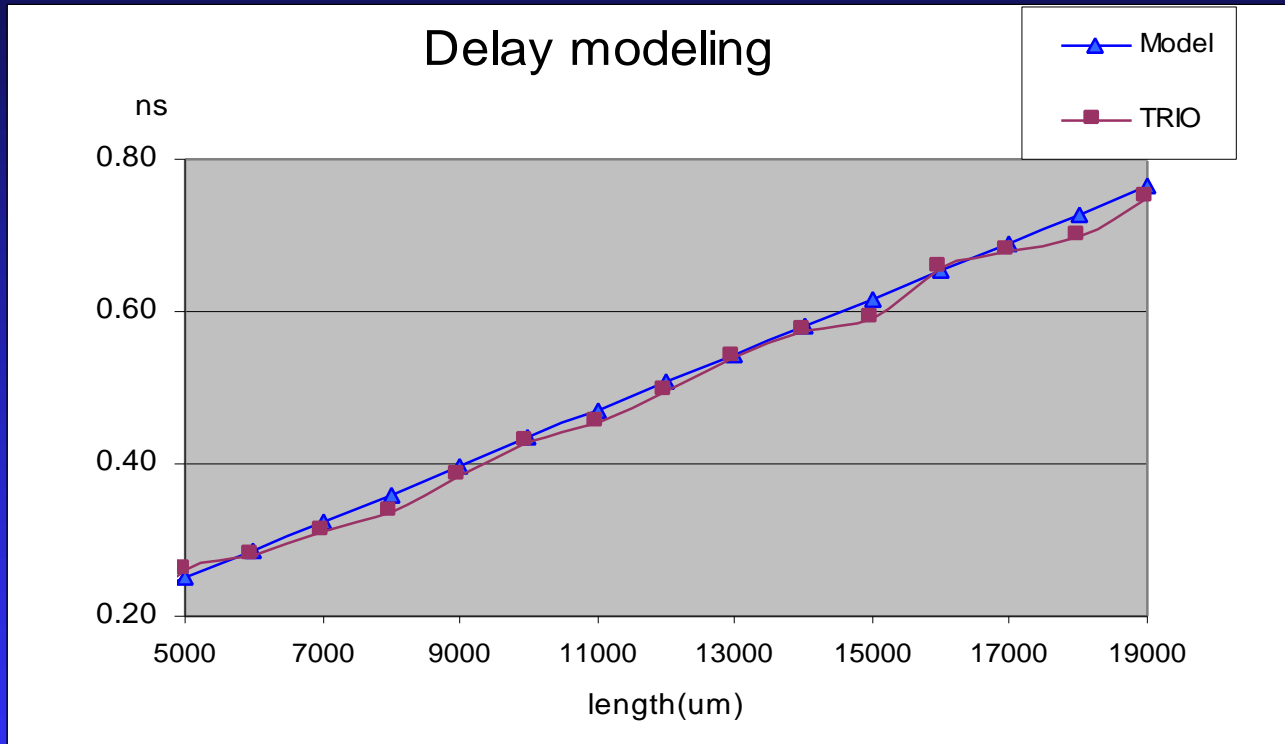
# Multiple Critical Sinks/OWS



- Random 4-pin nets,  $0.18\mu\text{m}$  tech,  $R_d = 180\text{ohm}$ ,  $C_s = 10\text{fF}$
- TRIO uses max. allowable wire width of  $20\times$  min; wire is segmented in every  $500\mu\text{m}$ .
- Length is the distance from source to ‘most critical’ sink

# MCS/BISWS

- Similar to OWS, take the max of SCS/BISWS



- Random 4-pin nets,  $0.18\mu\text{m}$ ,  $R_{d0} = r_g/10$ ,  $C_s = c_g \times 10$
- TRIO uses max. buffer size of 400x min, wire width of 20x min. width; wire is segmented in every 500um.

# Some Applications of Our Models

- **Layout-driven physical and RTL level floorplanning**
  - ◆ **Predict accurate interconnect delay and routing resource without really going into layout details;**
  - ◆ **Use accurate interconnect delay/area to guide floorplanning/placement**
- **Interconnect Architecture Planning**  
[Cong-Pan, DAC'99]
- **Note: TRIO or other interconnect optimization engines will still be needed to generate the final layout!**

# Conclusion

- **Interconnect delay and area estimation model with closed-form formula or simple characteristic equations for multiple-pin nets under various interconnect optimizations**
  - ◆ **Very accurate**
  - ◆ **Extremely fast**
  - ◆ **High level abstraction**
  - ◆ **Very easy to interact with synthesis/planning tools**
- **Future work:**
  - ◆ **Crosstalk noise estimation**
  - ◆ **Buffer planning**
  - ◆ **Interconnect-driven floorplanning**