

# Timing Driven Global Placement

Dimitrios Gerostathis

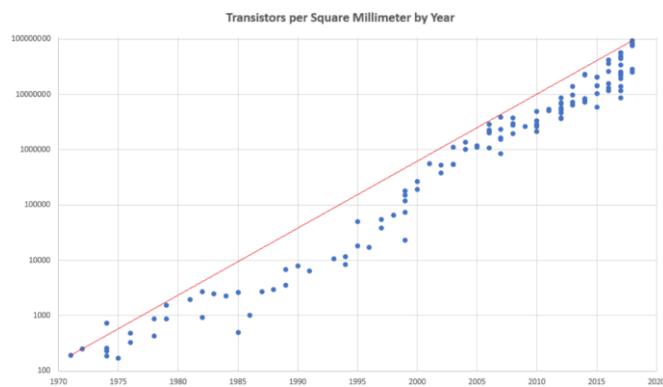
Supervisors: Christos Sotiriou, Fotis Plessas, Giorgos Stamoulis

1

## Introduction to EDA

### ► Moore's Law

- The transistor number on a chip doubles every two years



► 2

Timing Driven Global Placement

7/6/2025

2

## Introduction to EDA

- ▶ Chips nowadays consist of millions to billions of transistors
  - ▶ The task of laying out gates and interconnect wires by hand is impossible
- ▶ Electronic Design Automation (EDA) solves the problem of scaling
  - ▶ Participate at every stage of the ASIC design flow

▶ 3 Timing Driven Global Placement 7/6/2025

3

## Introduction to EDA

- ▶ Placement:
  - ▶ Determines the locations of components into the chip layout
  - ▶ These locations meet certain layout constraints
  - ▶ Consists of
    - ▶ Global Placement
    - ▶ Legalization
    - ▶ Detailed Placement

▶ 4 Timing Driven Global Placement 7/6/2025

4

## Introduction to EDA

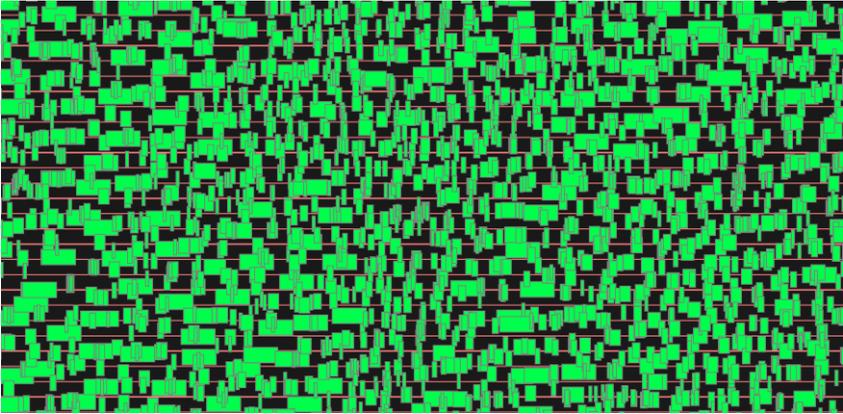
- ▶ Global Placement
  - ▶ Finds an initial placement of the logic cells
  - ▶ Allows cells to be unaligned and to overlap at a certain extent with each other

▶ 5 Timing Driven Global Placement 7/6/2025

5

## Introduction to EDA

- ▶ After Global Placement



▶ 6 Timing Driven Global Placement 7/6/2025

6

## Introduction to EDA

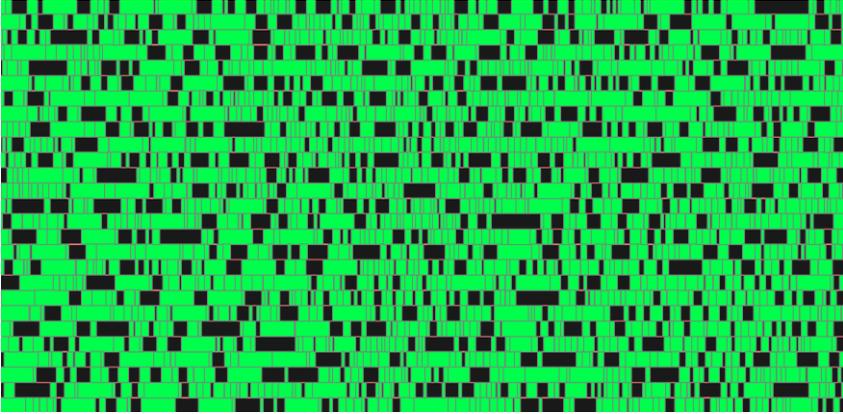
- ▶ Legalization
  - ▶ Removes overlap and properly aligns cells
    - ▶ Alignment is based on the power grid, the rows and columns of the chip layout
  - ▶ Seeks to minimize displacement from Global Placement
    - ▶ Have the minimum impact possible on wire length increase

▶ 7 Timing Driven Global Placement 7/6/2025

7

## Introduction to EDA

- ▶ After Legalization



▶ 8 Timing Driven Global Placement 7/6/2025

8

## Introduction to EDA

### ▶ Detailed Placement

- ▶ Improves the legalized result further concerning given objectives
  - ▶ Reduce wire length
  - ▶ Improve timing
  - ▶ Create room for another object when space is available

▶ 9

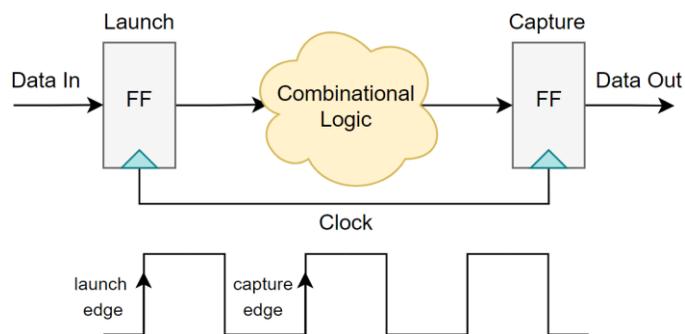
Timing Driven Global Placement

7/6/2025

9

## Introduction to Timing

### ▶ Synchronous Digital Circuit



▶ 10

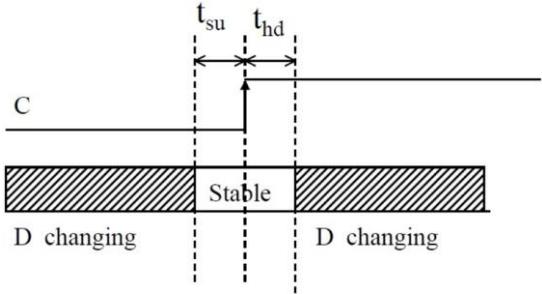
Timing Driven Global Placement

7/6/2025

10

## Introduction to Timing

- ▶ Ensure proper capture of the launch data
  - ▶ Setup Constraint
    - ▶ Longest paths
  - ▶ ~~Hold Constraint~~
    - ▶ ~~Shortest paths~~



▶ 11

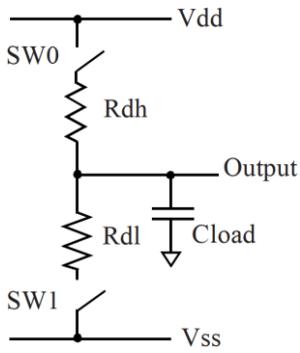
Timing Driven Global Placement

7/6/2025

11

## Introduction to Timing

- ▶ Propagation Delay
  - ▶ Exists due to parasitics
  - ▶ Depends on the charging/discharging speed of the Cload
- ▶ Clload consists of
  - ▶ Output capacitance of driver
  - ▶ **Interconnect** → *Dominating Factor!*
  - ▶ Input capacitance of sinks



▶ 12

Timing Driven Global Placement

7/6/2025

12

## Introduction to Timing

- ▶ Important Metrics
  - ▶ Actual Arrival Time (AAT)
  - ▶ Required Arrival Time (RAT)

The diagram illustrates a timing path through a circuit. It starts with 'Data In' entering a flip-flop (FF). A dashed vertical line indicates the 'Input Arrival Time (2ns)'. The signal then passes through a cloud labeled 'Combinational Logic'. A second dashed vertical line indicates the 'AAT (5ns)'. The signal then enters another flip-flop (FF). A third dashed vertical line indicates the 'RAT (7ns)'. The 'Data Out' is shown exiting this second flip-flop. A 'Clock' signal is shown at the bottom, with a double-headed arrow indicating a 'slack = 2ns' between the AAT and RAT.

▶ 13 Timing Driven Global Placement 7/6/2025

13

## Introduction to Timing

- ▶ Static Timing Analysis (STA)
  - ▶ Identifies longest and shortest paths
  - ▶ Does not depend on the current input
  - ▶ Propagates the AAT and RAT values

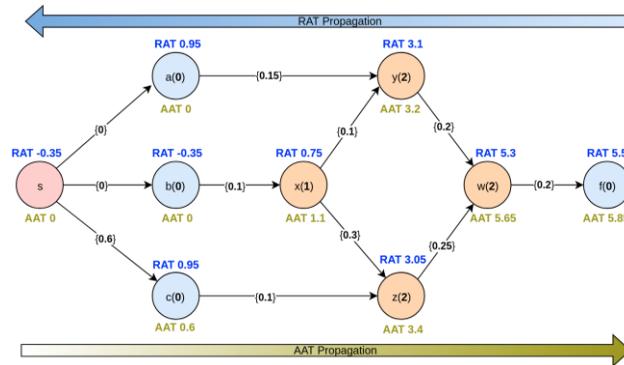
The diagram shows a logic circuit with three inputs:  $a<0>$ ,  $b<0>$ , and  $c<0.6>$ .  $a<0>$  is connected to a 2-input AND gate  $y(2)$ .  $b<0>$  is connected to a 1-input NOT gate  $x(1)$ . The output of  $x(1)$  is connected to a 2-input AND gate  $z(2)$ .  $c<0.6>$  is also connected to  $z(2)$ . The output of  $y(2)$  is connected to a 2-input AND gate  $w(2)$ . The output of  $z(2)$  is also connected to  $w(2)$ . The final output is  $f$ . Delay values are shown in brackets:  $\{0.1\}$  for  $x(1)$ ,  $\{0.3\}$  for  $z(2)$ ,  $\{0.2\}$  for  $y(2)$ , and  $\{0.25\}$  for  $w(2)$ .

▶ 14 Timing Driven Global Placement 7/6/2025

14

## Introduction to Timing

### ▶ AAT, RAT Propagation



▶ 15

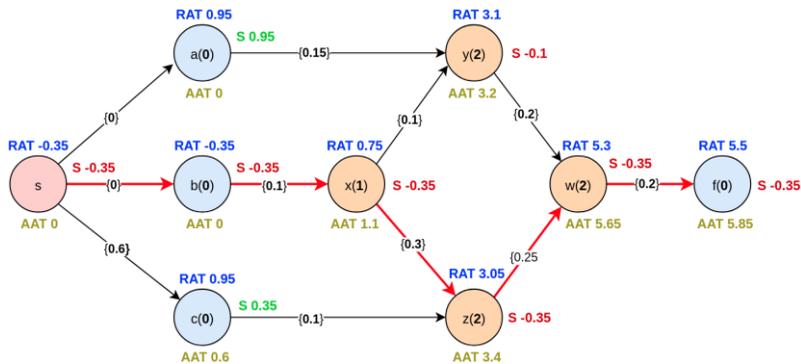
Timing Driven Global Placement

7/6/2025

15

## Introduction to Timing

### ▶ Slack calculation based on longest path (setup check)



▶ 16

Timing Driven Global Placement

7/6/2025

16

## Introduction to Timing

- ▶ Important Timing Metrics
  - ▶ Total Negative Slack (TNS)
  - ▶ Worst Negative Slack (WNS)

▶ 17 Timing Driven Global Placement 7/6/2025

17

## Introduction to Timing Driven Global Placement

- ▶ Timing Driven Global Placement encapsulates timing constraints
  - ▶ Aids the placement algorithm into minimizing critical connections

```
graph LR; A[Timing Analysis STA] --> B[Interface to Placer weights]; B --> C[Global Placer]; C -- Obtain timing information --> A;
```

▶ 18 Timing Driven Global Placement 7/6/2025

18

## Global Placement

- ▶ A variety of methods exist in the literature
  - ▶ Partitioning Based
  - ▶ **Analytic Techniques**
  - ▶ Stochastic
- ▶ **We implement an Analytic Force-Directed algorithm**

▶ 19

Timing Driven Global Placement

7/6/2025

19

## Global Placement

- ▶ Analytic Problem Formulation
  - ▶ Minimizes the following cost function

$$\Gamma(\mathbf{x}, \mathbf{y}) = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n c(i, j) \left( \underbrace{(x_i - x_j)^2}_{\text{Quadratic Distance}} + \underbrace{(y_i - y_j)^2}_{\text{Quadratic Distance}} \right)$$

↑  
**Wire Length**
↑  
**Connection Cost**
↓  
**Quadratic Distance**

▶ 20

Timing Driven Global Placement

7/6/2025

20

## Global Placement

- ▶ Vector Notation and Connectivity Matrix
  - ▶ Minimization can be separate in the x and y direction
  - ▶ For the x-direction we have:

$$\Gamma_x(\mathbf{x}) = \frac{1}{2} \sum_i \sum_j c_{ij} (x_i - x_j)^2 = \frac{1}{2} \mathbf{x}^T \mathbf{L} \mathbf{x}$$

$\mathbf{x} = (x_1, x_2, \dots, x_n)^T$       **Laplacian Matrix**

▶ 21

Timing Driven Global Placement

7/6/2025

21

## Global Placement

- ▶ Laplacian Matrix

$$\mathbf{L} = \mathbf{D} - \mathbf{C}$$

**Diagonal Matrix**      **Connectivity Matrix**

- ▶ Diagonal Matrix
  - ▶ Entry  $\mathbf{D}(i, i)$  stores the summation of every connection cost generating from component  $i$ .
- ▶ Connectivity Matrix (Symmetric)
  - ▶ For a connection between components  $i, j$ , it stores the connection costs  $c_{ij}$ , at matrix positions  $\mathbf{C}(i, j)$  and  $\mathbf{C}(j, i)$  and has zero in the diagonal positions

▶ 22

Timing Driven Global Placement

7/6/2025

22

## Global Placement

The circuit diagram shows a network of gates:
 

- Gate **a**: inputs  $i1, i2$ , output  $\bar{c}_{ad}$
- Gate **b**: inputs  $i2, i3$ , output  $c_{bd}$
- Gate **c**: inputs  $i3, i4$ , output  $c_{bc}$
- Gate **d**: inputs  $\bar{c}_{ad}, c_{bd}$ , output  $c_{do1}$
- Gate **e**: inputs  $c_{bc}, c_{bd}$ , output  $c_{ef}$
- Gate **f**: inputs  $c_{do1}, c_{ef}$ , output  $c_{fo2}$
- Gate **g**: inputs  $c_{do1}, c_{ef}, c_{cg}$ , output  $c_{go3}$

The matrix  $\mathbf{L}$  is defined as:

$$\mathbf{L} = \begin{bmatrix} d_a & 0 & 0 & -c_{ad} & 0 & 0 & 0 \\ 0 & d_b & 0 & -c_{bd} & -c_{bc} & 0 & 0 \\ 0 & 0 & d_c & 0 & 0 & 0 & -c_{cg} \\ -c_{da} & -c_{db} & 0 & d_d & 0 & -c_{df} & 0 \\ 0 & -c_{eb} & 0 & 0 & d_e & -c_{ef} & -c_{eg} \\ 0 & 0 & 0 & -c_{fd} & -c_{fe} & d_f & 0 \\ 0 & 0 & -c_{gc} & 0 & -c_{ge} & 0 & d_g \end{bmatrix}$$

The diagonal elements are defined as:

$$d_a = \bar{c}_{11a} + \bar{c}_{12a} + \bar{c}_{ad}$$

$$d_b = c_{12b} + c_{13b} + c_{bd} + c_{be}$$

$$d_c = c_{13c} + c_{14c} + c_{cg}$$

$$d_d = c_{da} + c_{db} + c_{do1} + c_{df}$$

$$d_e = c_{da} + c_{db} + c_{do1} + c_{df}$$

$$d_f = c_{fd} + c_{fe} + c_{fo2} + c_{be} + c_{fo1}$$

$$d_g = c_{ge} + c_{gc} + c_{go3}$$

▶ 23

Timing Driven Global Placement

7/6/2025

23

## Global Placement

▶ If we also include fixed cell locations, the Matrix-Vector notation is given as

$$\Gamma_x(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \mathbf{L} \mathbf{x} + \mathbf{b}^T \mathbf{x} + const$$

$\downarrow$   
 $b_i = - \sum_f c_{if} x_f$

▶ 24

Timing Driven Global Placement

7/6/2025

24

## Global Placement

$$\mathbf{b} = \begin{bmatrix} b_a \\ b_b \\ b_c \\ b_d \\ b_e \\ b_f \\ b_g \end{bmatrix} = \begin{bmatrix} -\sum_f c_{af} x_f \\ -\sum_f c_{bf} x_f \\ -\sum_f c_{cf} x_f \\ -\sum_f c_{df} x_f \\ -\sum_f c_{ef} x_f \\ -\sum_f c_{ff} x_f \\ -\sum_f c_{gf} x_f \end{bmatrix} = \begin{bmatrix} c_{a1}x_{i1} + c_{a2}x_{i2} \\ c_{b2}x_{i2} + c_{b3}x_{i3} \\ c_{c3}x_{i3} + c_{c4}x_{i4} \\ c_{d1}x_{o1} \\ 0 \\ c_{f2}x_{o2} \\ c_{g3}x_{o3} \end{bmatrix}$$

▶ 25 Timing Driven Global Placement 7/6/2025

25

## Global Placement

- ▶ **Cost function minimization**
  - ▶  $\Gamma_{\mathbf{x}}$  is convex and its minimum is obtained by setting its gradient to zero
 
$$\nabla \Gamma_{\mathbf{x}} = \nabla \left( \frac{1}{2} \mathbf{x}^T \cdot \mathbf{L} \mathbf{x} + \mathbf{b}^T \mathbf{x} + \text{const} \right)$$

$$\nabla \Gamma_{\mathbf{x}} = \mathbf{L} \mathbf{x} + \mathbf{b} = \vec{0}$$

$$\mathbf{L} \mathbf{x} = -\mathbf{b}$$
  - ▶ Produces a  $n \times n$  system of linear equations which can be solved by computer software, like the intel MKL.

▶ 26 Timing Driven Global Placement 7/6/2025

26

## Global Placement

- ▶ Cost function minimization
 
$$\nabla\Gamma_x(\mathbf{x}) = \mathbf{L}_x\mathbf{x} + \mathbf{b}_x = \vec{0}$$

$\swarrow$  constants       $\searrow$  positions
- ▶ Hooke's Law
 
$$\vec{F} = k \cdot x$$

$\swarrow$  constant       $\searrow$  position

▶ 27      Timing Driven Global Placement      7/6/2025

27

## Global Placement

- ▶ Each two-pin connection is resembled as an elastic spring
 
$$\mathbf{F}_x^{\text{net}}(\mathbf{x}) = \nabla\Gamma_x(\mathbf{x}) = \mathbf{L}_x\mathbf{x} + \mathbf{b}_x$$

$\swarrow$   
 stores the summation  
 of the net forces that  
 act on each component
- ▶ The cost function represents the energy of the spring system

▶ 28      Timing Driven Global Placement      7/6/2025

28

## Global Placement

- ▶ *Net force* solely minimizes wirelength
  - ▶ Creates overlap
- ▶ Global Placement utilizes *move force*
  - ▶ Confronts overlap
- ▶ Both forces work in conjunction

▶ 29 Timing Driven Global Placement 7/6/2025

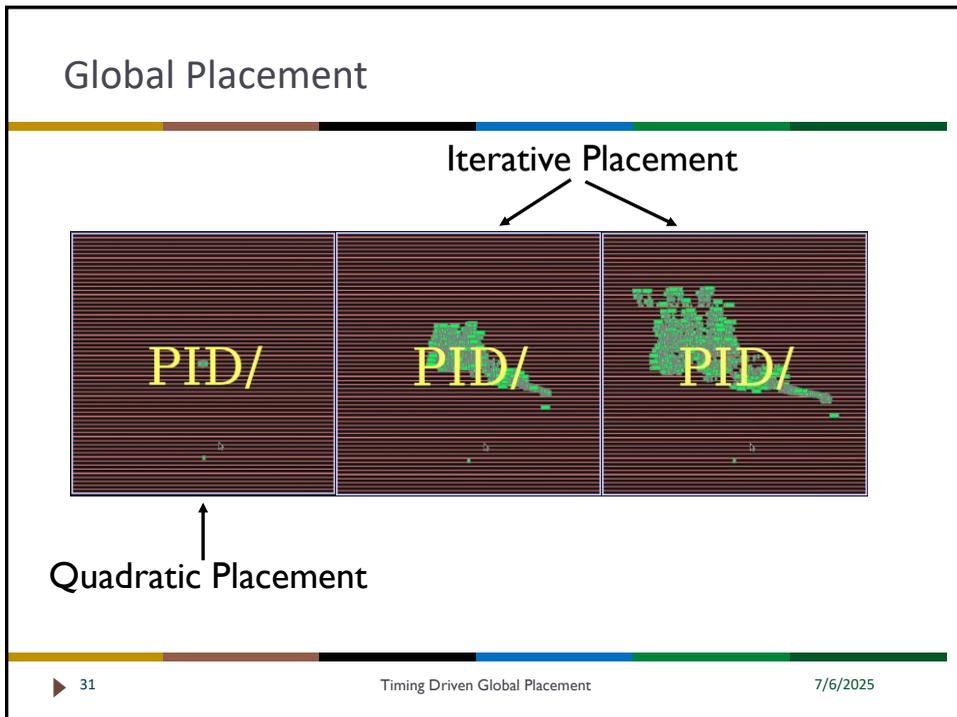
29

## Global Placement

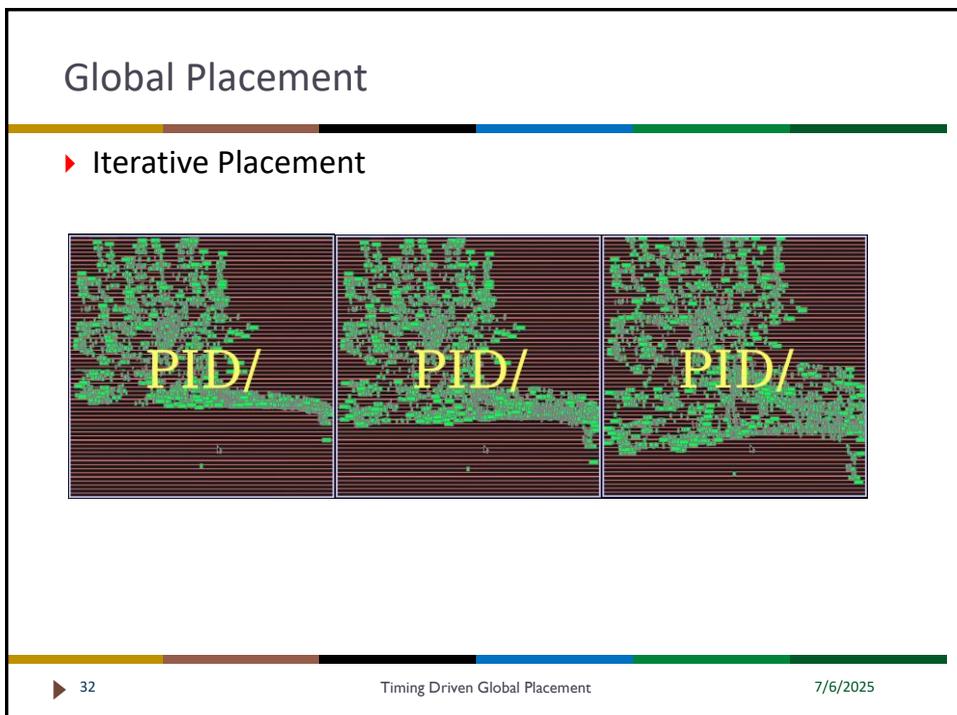
- ▶ Global Placement consists of two stages
  1. Quadratic Placement
    - ▶ Exclusively Minimizes Cost Function
    - ▶ Contracts all components to the center
  2. Iterative Placement
    - ▶ Spreads components using move force
    - ▶ Stops when overlap reaches a certain threshold

▶ 30 Timing Driven Global Placement 7/6/2025

30



31



32

## Global Placement

- ▶ Hold Force
  - ▶ Compensates Net Force at the beginning of each iteration
    - ▶ Instead, the modules would collapse back to high overlap positions
  - ▶ Inverse of net force
- ▶ Solution Convergence
  - ▶ Force-Equilibrium Solution

$$\mathbf{F}_x^{net} + \mathbf{F}_x^{move} + \mathbf{F}_x^{hold} = \vec{0}$$

▶ 33

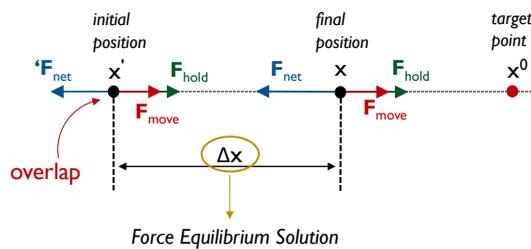
Timing Driven Global Placement

7/6/2025

33

## Global Placement

- ▶ Forces acting in each iteration



▶ 34

Timing Driven Global Placement

7/6/2025

34

## Global Placement

### ► Solution

$$\mathbf{F}_x^{net} + \mathbf{F}_x^{move} + \mathbf{F}_x^{hold} = \vec{0}$$

$$\underbrace{(\mathbf{L}_x + \overset{\circ}{\mathbf{C}}_x)}_{\substack{\text{Influence} \\ \text{Net-force}}} \Delta \mathbf{x} = \underbrace{-\overset{\circ}{\mathbf{C}}_x \Phi_x}_{\text{constant}} \quad \mathbf{x} = \mathbf{x}' + \Delta \mathbf{x}$$

sparse
definite
Influence

Move-force

► 35

Timing Driven Global Placement

7/6/2025

35

## Net Models

- Placement is before routing stage
  - Wire information not available
- We use net model estimations
  - HPWL
  - P2P
  - Clique
  - B2B

► 36

Timing Driven Global Placement

7/6/2025

36

## Net Models

- ▶ Net definition

output gate-pin

input gate-pin

net a

net b

net c

edge

edge

edge

edge

edge

edge

net can have multiple edges

▶ 37

Timing Driven Global Placement

7/6/2025

37

## Net Models

- ▶ In quadratic placement, components are considered dimensionless
  - ▶ Represented by a single point (pin) which lies in their geometric center

▶ 38

Timing Driven Global Placement

7/6/2025

38

## Net Models

- ▶ The quadratic cost function is re-written as:

$$\Gamma(\mathbf{x}, \mathbf{y}) = \frac{1}{2} \sum_{n \in \mathcal{N}} \sum_{e \in \mathcal{E}_n} c(p, q) \left( (x_p - x_q)^2 + (y_p - y_q)^2 \right)$$

↙
↘

For every net      For every edge

- ▶ Net models specify the set  $\mathcal{E}_n$

▶ 39

Timing Driven Global Placement

7/6/2025

39

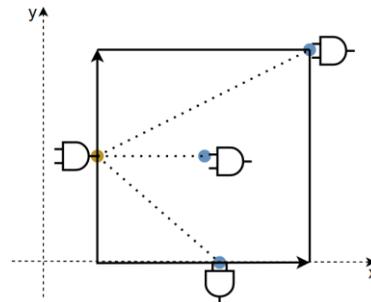
## Net Models

- ▶ Half Perimeter Wirelength (HPWL)
  - ▶ Create a bounding box and take its half perimeter

- ▶ Cost function becomes:

$$\Gamma(\mathbf{x}, \mathbf{y}) = \frac{1}{2} \sum_{n \in \mathcal{N}} (x_n^{\max} - x_n^{\min}) + (y_n^{\max} - y_n^{\min})$$

- ▶ Fast Computation
- ▶ Good Estimation
- ▶ Not quadratic
- ▶ Cannot create symmetric matrices



▶ 40

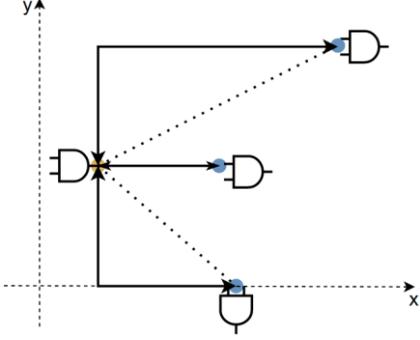
Timing Driven Global Placement

7/6/2025

40

## Net Models

- ▶ Point 2 Point (P2P)
  - ▶ Follows the net definition logic
  - ▶ Agrees with the quadratic mathematical formulation



▶ 41

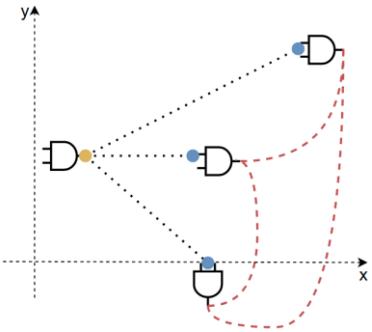
Timing Driven Global Placement

7/6/2025

41

## Net Models

- ▶ Clique
  - ▶ Includes all the possible edges of a net
  - ▶ Creates denser matrices and facilitates more on the minimization of the cost function



▶ 42

Timing Driven Global Placement

7/6/2025

42

## Net Models

- ▶ By utilizing the clique net model the cost function becomes:

$$\Gamma(\mathbf{x}, \mathbf{y}) = \frac{1}{2} \sum_{p=1}^{|\mathcal{n}|} \sum_{q=p+1}^{|\mathcal{n}|} c(p, q) \left( (x_p - x_q)^2 + (y_p - y_q)^2 \right)$$

- ▶ Clique net model creates complete graphs and it hinders the solving efficiency of the matrix solvers

▶ 43

Timing Driven Global Placement

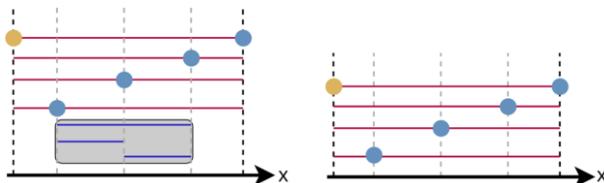
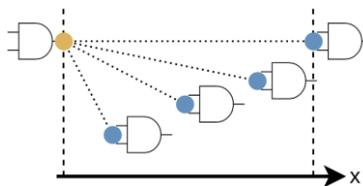
7/6/2025

43

## Net Models

- ▶ Bound2Bound net model
  - ▶ Utilizes only edges to the boundary pins
  - ▶ All inner edges are removed

$$c_x^{B2B} = \begin{cases} 0 & p, q \text{ inner pins} \\ \frac{2}{P-1} \frac{1}{|x_p - x_q|} & \text{else} \end{cases}$$



▶ 44

Timing Driven Global Placement

7/6/2025

44

## Net Models

- ▶ **Bound2Bound net model**
  - ▶ Exactly represents the HPWL metric
  - ▶ Lower number of connections at each net
    - ▶ Sparse Matrices
    - ▶ Better run times
  - ▶ Linearizes the cost function
    - ▶ No need for gradient Calculation
  - ▶ Achieves the best cost function minimization

▶ 45 Timing Driven Global Placement 7/6/2025

45

## Existing Work

- ▶ **Net Based Approaches**
  - ▶ Prioritizes the minimization of critical nets
    - ▶ Net weights → *We focus here!*
    - ▶ Net constraints
- ▶ **Path Based Approaches**
  - ▶ Attempts to meet timing closure on a subset or all the paths directly
    - ▶ Linear Programming optimization techniques

▶ 46 Timing Driven Global Placement 7/6/2025

46

## Existing Work

- ▶ Net Based Approaches
  - ▶ Good runtime
  - ▶ Good scalability
  - ▶ Easy implementation
  - ▶ Unpredictable
  - ▶ Can increase total wirelength a lot
- ▶ Path Based Approaches
  - ▶ Direct problem handling
  - ▶ High complexity
  - ▶ Very bad scalability

▶ 47 Timing Driven Global Placement 7/6/2025

47

## Existing Work

- ▶ Net Weighting techniques
  - ▶ Assign weights to nets considering their criticality
  - ▶ Placer minimizes the weighted total wirelength
- ▶ Empirical Net Weighting
  - ▶ Maps certain critical factors to weights
    - Easy implementation
    - Low runtime
    - High unpredictability
- ▶ Sensitivity Net Weighting
  - ▶ Look-ahead mechanism
    - Tries to predict the impact of weights to design metrics
    - More complex and higher runtime

▶ 48 Timing Driven Global Placement 7/6/2025

48

## Existing Work

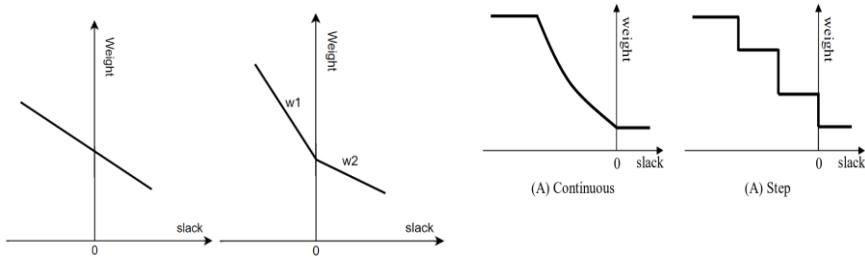
### ▶ Empirical Net Weighting

#### ▶ *Slack Based*

- ▶ Uses functions that map slack values into timing weights

$$W(\text{slack}) = a * \text{slack} + b$$

- ▶ A variety of heuristic functions can be used



▶ 49

Timing Driven Global Placement

7/6/2025

49

## Existing Work

### ▶ Empirical Net Weighting

#### ▶ *Fanout and delay bound*

- ▶ Fanout is used to estimate wirelength and wire delay
- ▶ Delay bound is the maximum allowable wire delay

$$w \approx \frac{\text{fanout}}{\text{net delay bound}}$$

#### ▶ Does not require timing information

- ▶ Can run pre-placement
- ▶ Negligible increase in runtime

▶ 50

Timing Driven Global Placement

7/6/2025

50

## Existing Work

- ▶ Delay bound
  - ▶ Maximum possible delay of a net, before it turns negative
  - ▶ Can be used to generate wirelength constraints
    - ▶ Permits the placement program on increasing these nets
  - ▶ Can be generated with Zero Slack Algorithm(ZSA)

▶ 51

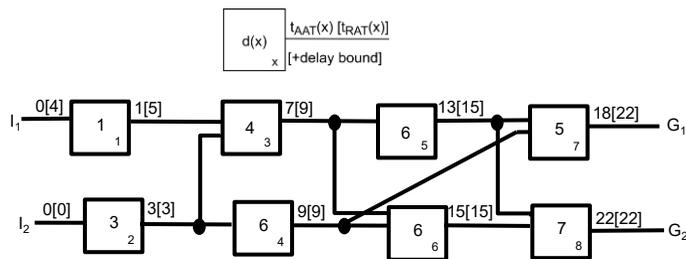
Timing Driven Global Placement

7/6/2025

51

## Existing Work

### Zero Slack Algorithm



▶ 52

Timing Driven Global Placement

7/6/2025

52

## Existing Work

### Zero Slack Algorithm

$$d(x) = \frac{t_{AAT}(x) - t_{RAT}(x)}{x} + \text{delay bound}$$

I<sub>1</sub> 0[4] → 1<sub>1</sub> 1[5] [+2] → 4<sub>3</sub> 7[9] [+1] → 5<sub>7</sub> 13[15] [+1] → 7<sub>7</sub> 22[22] [+1] → G<sub>1</sub>  
 I<sub>2</sub> 0[0] → 3<sub>2</sub> 3[3] → 6<sub>4</sub> 9[9] → 6<sub>6</sub> 15[15] → 7<sub>8</sub> 22[22] → G<sub>2</sub>

▶ 53      Timing Driven Global Placement      7/6/2025

53

## Existing Work

- ▶ Delay bound generation
  - ▶ Zero Slack Algorithm (ZSA)
    - ▶ Generates upper bound delays on each net of the circuit
      - Corresponds to the maximum possible delay a net can have before it turns negative
    - ▶ Delay bounds are used to generate wirelength constraints
      - Permits the placement program on increasing these nets

▶ 54      Timing Driven Global Placement      7/6/2025

54

## Existing Work

### ▶ Empirical Net Weighting

#### ▶ *Path depth & driver strength*

- ▶ A deep path is more likely to have longer wire length
- ▶ A weak driver imposes a larger delay than a strong driver for the same wirelength

$$w \approx D_l \times R_d$$

longest path      driver resistance

▶ 55

Timing Driven Global Placement

7/6/2025

55

## Existing Work

### ▶ Empirical Methods

- ▶ Hard to predict the impact of weights on design metrics
- ▶ Extensive parameter tuning is needed to make it work on specific design styles

▶ 56

Timing Driven Global Placement

7/6/2025

56

## Existing Work

### ▶ Sensitivity Net Weighting

- ▶ Identifies how sensitive design metrics are on a weight change
  - ▶ Sensitive nets should receive higher weights
  - ▶ Insensitive nets should not receive high weights
  
- ▶ Defines sensitivity factors for different design metrics
  - ▶ Wirelength
  - ▶ TNS
  - ▶ WNS

▶ 57

Timing Driven Global Placement

7/6/2025

57

## Existing Work

### ▶ Sensitivity Net Weighting

- ▶ Weight for net  $i$  is:

$$W(i) = \begin{cases} W_{init}(i) & Slk(i) \geq Slk_t \\ W_{init}(i) + \Delta W^*(i) & Slk(i) \leq Slk_t \end{cases}$$

→ slack threshold

$W_{init}(i)$ : The initial weight of net  $i$

$$\Delta W^*(i) = a[Slk_t - Slk(i)] \underbrace{S_W^{Slk}(i)}_{\text{slack to weight sensitivity}} + \beta \underbrace{S_W^{TNS}(i)}_{\text{TNS to weight sensitivity}}$$

▶ 58

Timing Driven Global Placement

7/6/2025

58

## Existing Work

- ▶ Incremental net weighting
  - ▶ In net weighting a net can alternate between been critical and noncritical
  - ▶ Mitigates this problem by keeping a history of the criticality of a net, on previous iterations

59

## Existing Work

- ▶ Incremental net weighting
  - ▶ Uses criticality information from two previous steps
  - ▶ Critical nets get 1, non-critical get 0

$$w_i^k = \begin{cases} w^{k-1} + W & \text{if } c_i^k = 1 \\ 1 & \text{if } c_i^k = 0 \wedge c_i^{k-1} = 0 \wedge c_i^{k-2} = 0 \\ \lceil w_i^{k-1} / 2 \rceil & \text{if } c_i^k = 0 \wedge c_i^{k-1} = 0 \wedge c_i^{k-2} = 1 \\ w^{k-1} & \text{if } c_i^k = 0 \wedge c_i^{k-1} = 1 \end{cases}$$

- ▶  $W$  is equal to the number of pins of the net.
  - ▶ *High fanout nets are more likely to be critical*

60

## Our Work

- ▶ Map slack to weight
  - ▶ The net force increases/decreases accordingly

*best slack* ————— **weight lower bound**

*average slack*  
 $w = 1$  —————

*WNS* ————— **weight upper bound**

$w < 1$

$w > 1$

$w$  increases

---

▶ 61      Timing Driven Global Placement      7/6/2025

61

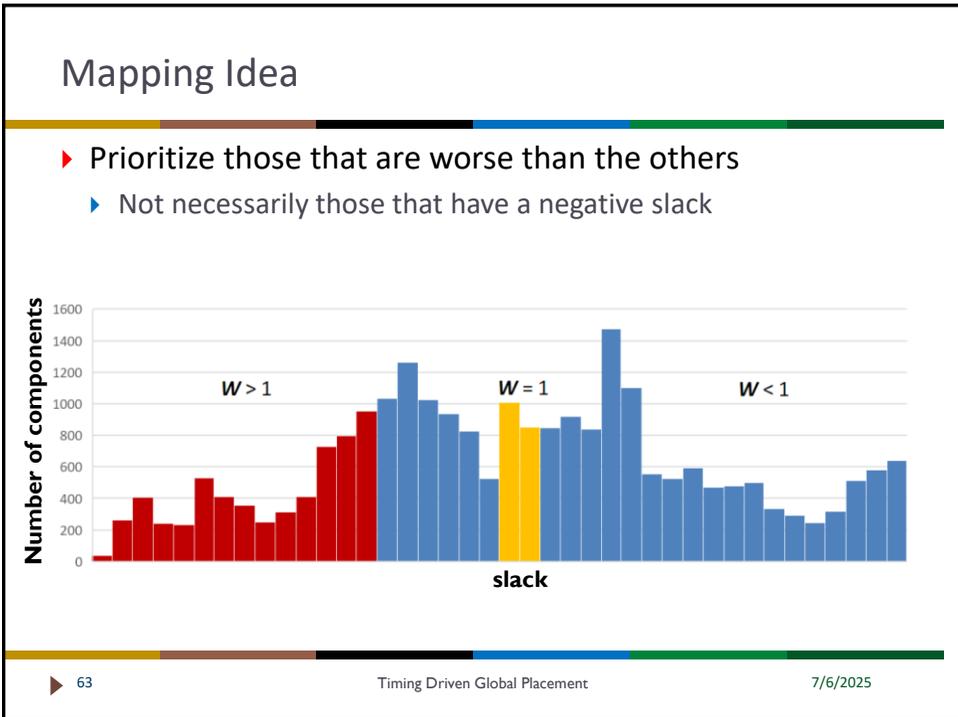
## Mapping Idea

- ▶ Relaxation of net force has a positive impact
  1. Separates noncritical components from critical
  2. Reduces overlap in critical areas
  3. Critical components stay closer due to less move force

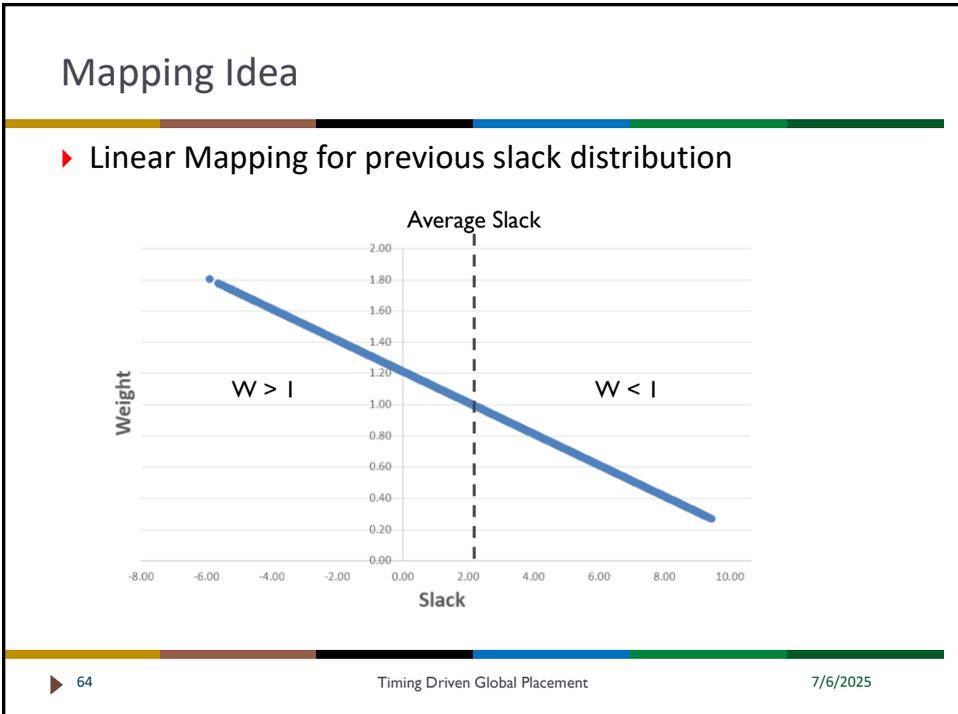
---

▶ 62      Timing Driven Global Placement      7/6/2025

62



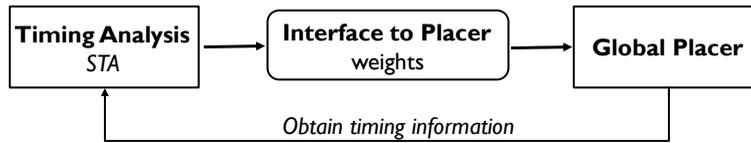
63



64

## Timing Information

- ▶ Provide Timing Feedback

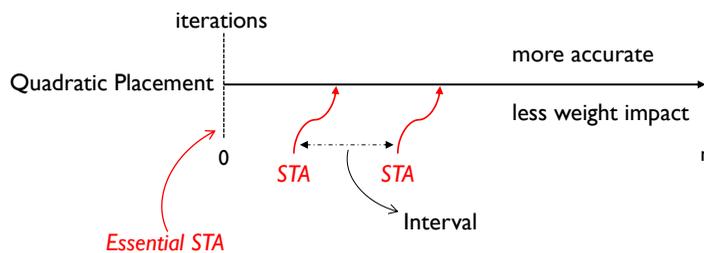


- ▶ How often should we obtain it?

65

## Our Work

- ▶ STA
  - ▶ Placement advances -> More accurate timing information
  - ▶ Placement advances -> Less weight impact
    - ▶ Hold force does not allow component retraction

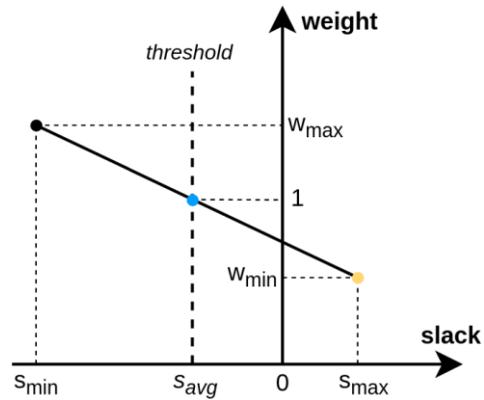


66

## Mapping Functions

### ▶ Linear Mapping Function

- ▶ Specified by two points
  - ▶ threshold point
  - ▶ weight lower bound



▶ 67

Timing Driven Global Placement

7/6/2025

67

## Mapping Functions

### ▶ Formula extraction

$$y - y_0 = a(x - x_0)$$

$$W(s) - 1 = a(s - s_{avg})$$

$$W(s) = 1 + \frac{1 - w_{min}}{s_{avg} - s_{max}}(s - s_{avg})$$

▶ 68

Timing Driven Global Placement

7/6/2025

68

## Mapping Functions

- ▶ Linear Function
  - ▶ Very low expense in HPWL
  - ▶ Inability to control weight upper bound values
  - ▶ Relatively small decrease in timing metrics

▶ 69

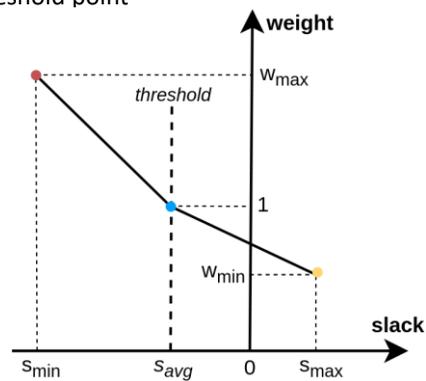
Timing Driven Global Placement

7/6/2025

69

## Mapping Functions

- ▶ Piece Wise Linear Function
  - ▶ Consists of two independent linear segments
    - ▶ Separated by the threshold point



▶ 70

Timing Driven Global Placement

7/6/2025

70

## Mapping Functions

- ▶ Formula extraction

- ▶ Slope of left segment

$$a = \frac{w_{max} - 1}{s_{min} - s_{avg}}$$

- ▶ Final formula

$$W_{PWL}(s) = \begin{cases} 1 - \frac{1-w_{min}}{s_{max}-s_{avg}}(s - s_{avg}) & s_{avg} \leq s \leq s_{max} \\ 1 - \frac{w_{max}-1}{s_{avg}-s_{min}}(s - s_{avg}) & s_{min} \leq s < s_{avg} \end{cases}$$

▶ 71

Timing Driven Global Placement

7/6/2025

71

## Mapping Functions

- ▶ Piece Wise Linear Function

- ▶ Substantial decrease in timing metrics
  - ▶ Might lead to high expense in HPWL for large  $w_{max}$  values

- ▶ Piece Wise Exponential Function

- ▶ Attempts to mitigate the previous problem
  - ▶ Follows a softer approach till it reaches the  $w_{max}$  value

▶ 72

Timing Driven Global Placement

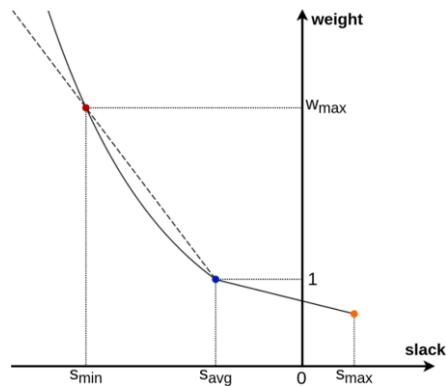
7/6/2025

72

## Mapping Functions

### ► Piece Wise Exponential Function

- Stays below the linear counterpart for the  $\{s_{min}, s_{avg}\}$  window



► 73

Timing Driven Global Placement

7/6/2025

73

## Mapping Functions

### ► PWE Formula extraction

- General formula

$$\mathbf{W}(s) = e^{-\alpha(s-s_{avg})}$$

- Satisfies the following condition

$$\mathbf{W}(s_{min}) = w_{max}$$

- Solving for  $\alpha$

$$e^{-(\alpha \cdot w_{min} + s_{avg})} = w_{max}$$

$$-(\alpha \cdot w_{min} + s_{avg}) = \ln(w_{max})$$

$$-\alpha \cdot w_{min} = \ln(w_{max}) - s_{avg}$$

$$\alpha = -\frac{\ln(w_{max})}{w_{min} - s_{avg}}$$

► 74

Timing Driven Global Placement

7/6/2025

74

## Mapping Functions

### ▶ PWE Final

$$W_{PWE}^{slow}(s) = \begin{cases} 1 - \frac{1-w_{min}}{s_{max}-s_{avg}}(s - s_{avg}) & s_{avg} \leq s \leq s_{max} \\ e^{-\frac{\ln(w_{max})}{w_{min}-s_{avg}}(s-s_{avg})} & s_{min} \leq s < s_{avg} \end{cases}$$

- ▶ Substantially less expense in HPWL
- ▶ Proved to be less aggressive than required
  
- ▶ The best outcome is to achieve high reduction and low HPWL expense at the same time

▶ 75

Timing Driven Global Placement

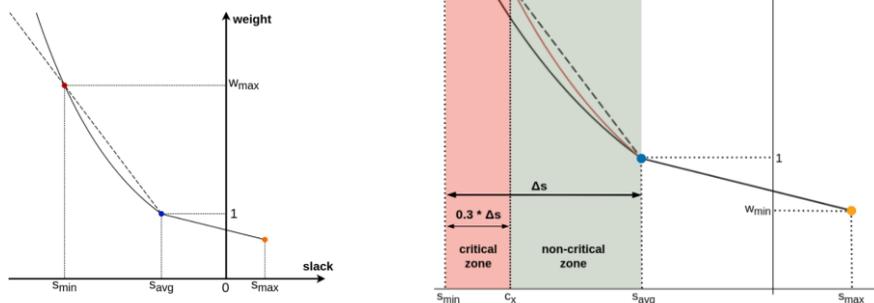
7/6/2025

75

## Mapping Functions

### ▶ Piece Wise Exponential Fast

- ▶ Stays below the PWL in the noncritical zone
- ▶ Stays above the PWE for the critical zone



▶ 76

Timing Driven Global Placement

7/6/2025

76

## Mapping Functions

### ▶ Formula extraction algorithm

```

// Calculate the left segment domain
1  $\Delta s = s_{min} - s_{avg}$ ;
// Define critical zone as 30% of the domain
2  $critical\_zone = 0.3 \cdot \Delta s$ ;
// Find the point  $(c_x, c_y)$  which separates the critical and
// non-critical zone
3  $c_x = s_{min} - critical\_zone$ ;
4  $c_y = W_{PWL}(c_x)$ ;
// Use points  $(c_x, c_y)$  and  $(s_{avg}, 1)$  to define the fast PWE
// function.
5  $a = \frac{\ln(c_y)}{c_x - threshold}$ ;
6  $W_{PWE}^{fast}(s) = e^{-a(s - s_{avg})}$ ;

```

▶ 77

Timing Driven Global Placement

7/6/2025

77

## Weight Distribution

- ▶ A weight is generated for an *output* gatepin of a component
  - ▶ It is then spread to the whole net defined by the gatepin
  - ▶ Follows the construction of the Laplacian matrix
  - ▶ Pessimistic in nature
  
- ▶ What happens to multiple output components?

▶ 78

Timing Driven Global Placement

7/6/2025

78

## Weight Distribution

► Weight Distribution multiple output component

connection cost

$c_{13}$

3

$c_{14}$

4

$s_1 = 2$

1

2

$s_2 = -4$

$c_{25}$

5

$c_{26}$

6

$c_{27}$

7

► 79

Timing Driven Global Placement

7/6/2025

79

## Weight Distribution

► Weight Distribution multiple output component

$c_{13}$

3

$c_{14}$

4

$s_1 = 2$

1

2

$s_2 = -4$

$c_{25}$

5

$c_{26}$

6

$c_{27}$

7

► 80

Timing Driven Global Placement

7/6/2025

80

## Weight Distribution

▶ Weight Distribution multiple output component

The diagram shows a vertical bar with two points, 1 and 2. Point 1 is connected to outputs 3 and 4, and point 2 is connected to outputs 5, 6, and 7. Weights are assigned to each point:  $W_1 = 0.6$  for point 1 and  $W_2 = 5$  for point 2. The diagram also shows the calculation of weighted components:  $W_2 * C_{13}$ ,  $W_2 * C_{14}$ ,  $W_2 * C_{25}$ ,  $W_2 * C_{26}$ , and  $W_2 * C_{27}$ . The diagram includes a green oval for  $s_1 = 2$  and a red oval for  $s_2 = -4$ .

▶ 81 Timing Driven Global Placement 7/6/2025

81

## Weight Distribution

▶ What happens in case of unconstrained pins?

- ▶ Two different cases
  - ▶ At least one constrained pin exists
  - ▶ Constrained pin does not exist

▶ 82 Timing Driven Global Placement 7/6/2025

82

## Weight Distribution

► At least one constrained pin exists

Diagram illustrating weight distribution for a constrained pin. A vertical bar has pins 1 and 2. Pin 1 is connected to nodes 3 and 4, with  $s_1 = \infty$ . Pin 2 is connected to nodes 5, 6, and 7, with  $s_2 = -4$ . Nodes are labeled  $c_{13}$ ,  $c_{14}$ ,  $c_{25}$ ,  $c_{26}$ , and  $c_{27}$ .

► 83 Timing Driven Global Placement 7/6/2025

83

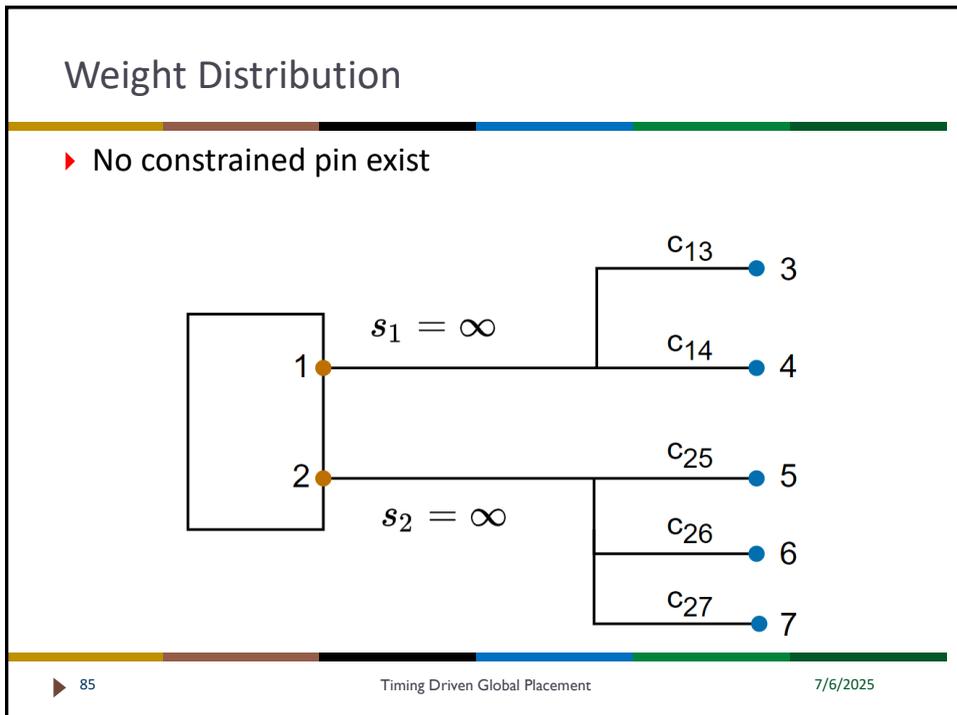
## Weight Distribution

► At least one constrained pin exists

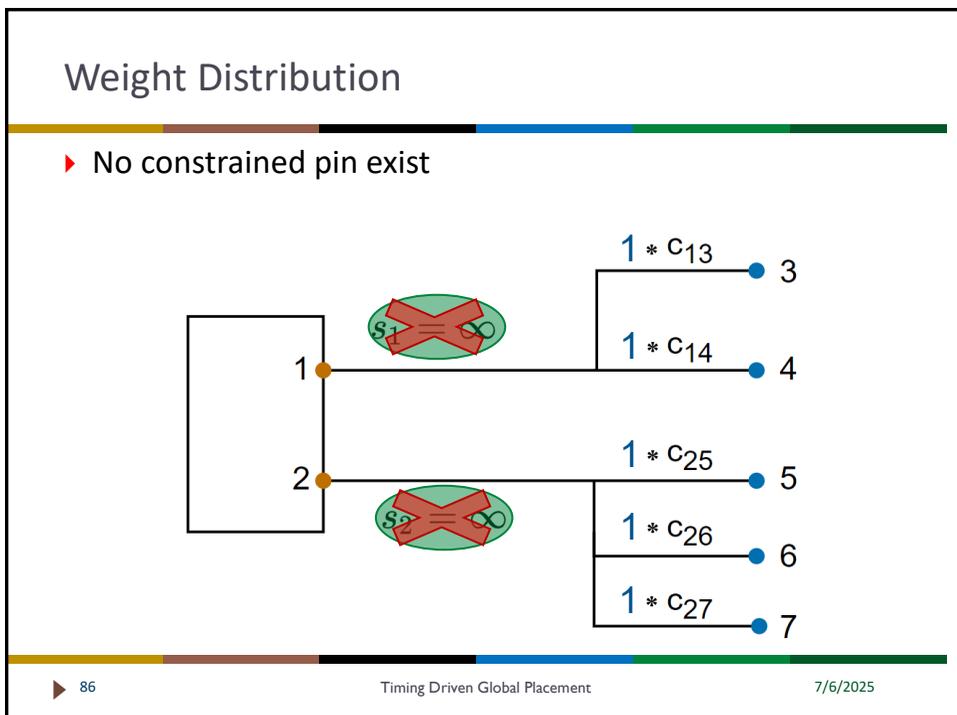
Diagram illustrating weight distribution for a constrained pin. A vertical bar has pins 1 and 2. Pin 1 is crossed out with a red X and labeled  $s_1 = \infty$ . Pin 2 is labeled  $s_2 = -4$ . A weight calculation shows  $W(s_2)$  leading to  $W_2 = 5$ . Nodes are labeled  $W_2 * c_{13}$ ,  $W_2 * c_{14}$ ,  $W_2 * c_{25}$ ,  $W_2 * c_{26}$ , and  $W_2 * c_{27}$ .

► 84 Timing Driven Global Placement 7/6/2025

84



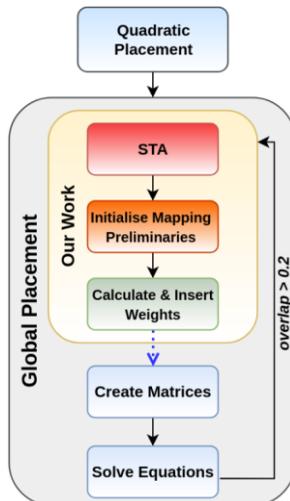
85



86

## Timing Driven Placement

### ► Final Algorithm



► 87

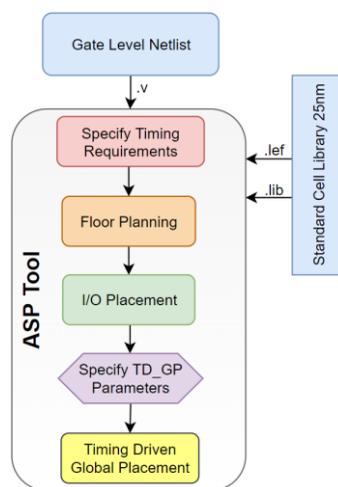
Timing Driven Global Placement

7/6/2025

87

## Experimental Results

### ► Flow



► 88

Timing Driven Global Placement

7/6/2025

88

## Experimental Results

- ▶ Specify Timing Driven Parameters
  - ▶ Function type
  - ▶ sta iteration interval
  - ▶ sta iterations
  - ▶  $W_{\min}$
  - ▶  $W_{\max}$

▶ 89      Timing Driven Global Placement      7/6/2025

89

## Experimental Results

- ▶ Specify Timing Driven Parameters
  - ▶ **Function type**
    - ▶ Linear
    - ▶ Piece Wise Linear (PWL)
    - ▶ Piece Wise Exponential Slow ( $PWE_{\text{slow}}$ )
    - ▶ Piece Wise Exponential Fast ( $PWE_{\text{fast}}$ )
  - ▶ sta iteration interval
  - ▶ sta iterations
  - ▶  $W_{\min}$
  - ▶  $W_{\max}$

▶ 90      Timing Driven Global Placement      7/6/2025

90

## Experimental Results

- ▶ Specify Timing Driven Parameters
  - ▶ Function type
  - ▶ sta iteration interval
  - ▶ sta iterations
  - ▶  $W_{\min}$
  - ▶  $W_{\max}$

Constant

91 Timing Driven Global Placement 7/6/2025

91

## Experimental Results

- ▶ Specify Timing Driven Parameters
  - ▶ Function type
  - ▶ sta iteration interval
  - ▶ sta iterations
  - ▶  $W_{\min}$
  - ▶  $W_{\max}$

Use a variety of combinations

92 Timing Driven Global Placement 7/6/2025

92

## Experimental Results

- ▶ Specify weight combinations  $\{w_{min}, w_{max}\}$ 
  1. Test *linear function* for different  $w_{min}$  values
    - ▶ Start at 0.1, end at 0.9 at increments of 0.1
  2. Keep the  $w_{min}$  that produces the biggest *WNS & TNS* reduction
  3. Try different  $w_{max}$  values and test the other functions

▶ 93

Timing Driven Global Placement

7/6/2025

93

## Experimental Results

- ▶ AES
  - ▶ Linear

Weight Bounds	HPWL	WNS	TNS
{0.9,x}	+0.3%	+1.5%	-1.0%
{0.8,x}	+0.4%	-1.3%	-4.7%
{0.7,x}	+0.8%	-4.9%	-6.8%
{0.6,x}	+0.7%	-5.7%	-8.5%
{0.5,x}	+1.3%	-1.8%	-10.7%
{0.4,x}	+1.7%	0.8%	-12.1%
{0.3,x}	+2.5%	-3.9%	-14.8%
{0.2,x}	+3.8%	-11.9%	-19.0%
{0.1,x}	+6.5%	-12.1%	-21.5%

Use  $w_{min} = 0.2$ 

▶ 94

Timing Driven Global Placement

7/6/2025

94

## Experimental Results

- ▶ Low  $w_{min}$  has positive impact on timing
  - ▶ Low  $w_{min}$  increases the gradient of the linear function
  - ▶ Low  $w_{min}$  relaxes net force of non-critical components
    - ▶ Do not interfere with the density calculation of critical areas
    - ▶ Critical components have less move force

▶ 95

Timing Driven Global Placement

7/6/2025

95

## Experimental Results

- ▶ AES
  - ▶ PWL &  $PWE_{slow}$

Weight Bounds	PWL			$PWE_{slow}$		
	HPWL	WNS	TNS	HPWL	WNS	TNS
{0.2,7}	4.6%	-22.2%	-31.3%	3.5%	-16.0%	-25.1%
{0.2,8}	4.9%	-21.3%	-32.4%	3.5%	-16.5%	-25.5%
{0.2,9}	5.1%	-25.0%	-34.4%	3.5%	-17.5%	-25.9%
{0.2,10}	5.4%	-22.7%	-33.5%	3.4%	-18.0%	-26.4%
{0.2,15}	6.7%	-19.3%	-37.8%	3.6%	-15.1%	-23.1%
{0.2,20}	8.3%	-19.3%	-42.0%	3.8%	-22.7%	-29.9%
{0.2,30}	11.3%	-23.2%	-48.3%	4.3%	-21.2%	-32.3%
{0.2,50}	13.9%	-27.4%	-58.7%	4.8%	-24.0%	-34.8%
{0.2,100}	21.3%	-38.8%	-71.5%	6.2%	-28.2%	-38.2%
{0.2,200}	-	-	-	7.4%	-29.9%	-41.2%

▶ 96

Timing Driven Global Placement

7/6/2025

96

## Experimental Results

- ▶ AES
  - ▶  $PWE_{fast}$

Weight Bounds	Critical Zone = 40%			Critical Zone = 70%		
	HPWL	WNS	TNS	HPWL	WNS	TNS
{0.2,20}	5.3%	-24.0%	-35.2%	9.0%	-37.3%	-49.0%
{0.2,30}	6.6%	-29.7%	-38.9%	11.4%	-43.9%	-58.0%
{0.2,50}	7.9%	-32.3%	-44.2%	16.2%	-50.1%	-67.2%
{0.2,100}	10.1%	-41.2%	-52.7%	27.7%	-52.1%	-73.9%
{0.2,150}	11.3%	-43.4%	-57.2%	34.1%	-61.0%	-78.8%
{0.2,170}	11.9%	-45.2%	-58.5%	35.7%	-61.8%	-79.2%
{0.2,200}	12.8%	-45.0%	-60.1%	37.7%	-58.7%	-80%

▶ 97
Timing Driven Global Placement
7/6/2025

97

## Experimental Results

- ▶ Trade off
  - ▶ Increase in Wirelength vs Decrease of TNS

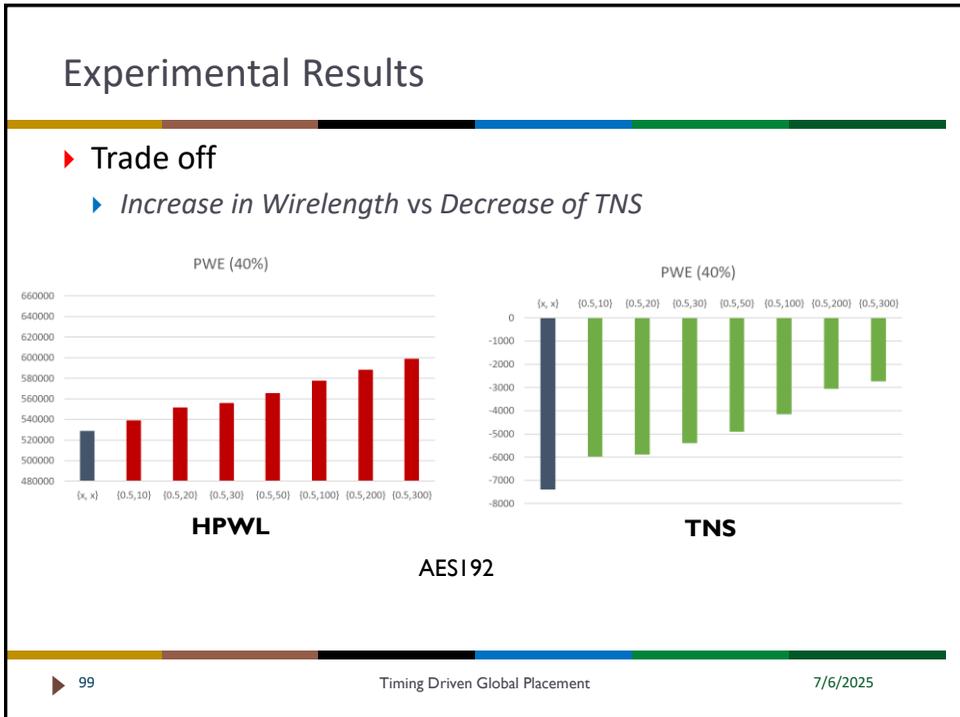
**HPWL**

**TNS**

AES

▶ 98
Timing Driven Global Placement
7/6/2025

98



99

### Experimental Results

- AES
  - Best results per mapping function

Mapping Function	HPWL	WNS	TNS
<i>Linear</i>	+6.5%	-12.1%	-21.5%
<i>PWL</i>	+21.3%	-38.8%	-71.5%
<i>PWE<sub>slow</sub></i>	+7.4%	-29.9%	-41.2%
<i>PWE<sub>fast</sub></i>	+37.7%	-58.7%	-80%

100 Timing Driven Global Placement 7/6/2025

100

## Experimental Results

- ▶ PID
  - ▶ Best Results per mapping function

Mapping Function	HPWL	WNS	TNS
<i>Linear</i>	+3.3%	-17%	-13.3%
<i>PWL</i>	+12%	-33%	-32%
<i>PWE<sub>slow</sub></i>	+11%	-34%	-32%
<i>PWE<sub>fast</sub></i>	+19.4%	-42%	-40%

▶ 101
Timing Driven Global Placement
7/6/2025

101

## Experimental Results

- ▶ Trade off
  - ▶ Increase in Wirelength vs Decrease of TNS

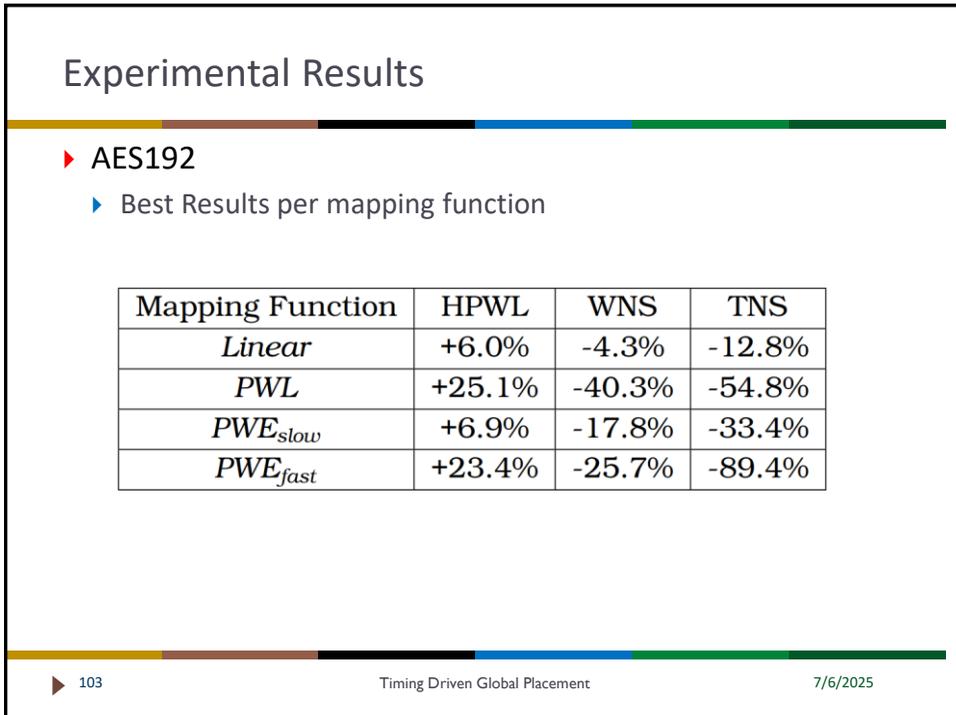
**HPWL**

**TNS**

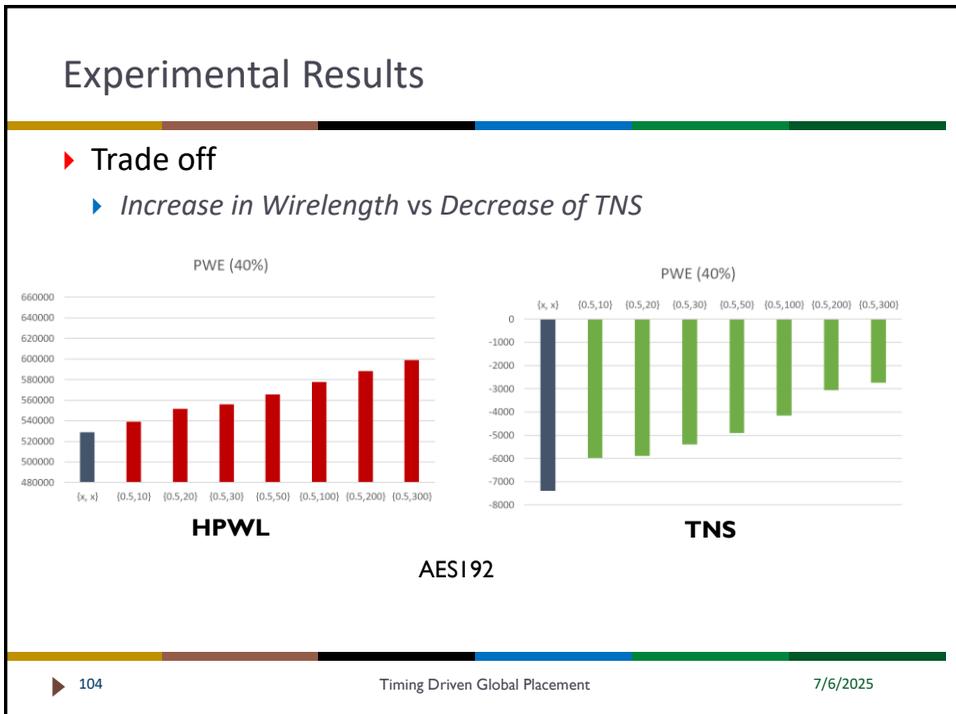
PID

▶ 102
Timing Driven Global Placement
7/6/2025

102



103



104

## Experimental Results

### ► B19

#### ► Best Results per mapping function

Mapping Function	HPWL	WNS	TNS
<i>Linear</i>	+1.9%	-21.2%	-19.8%
<i>PWL</i>	+30.3%	-35.8%	-33.5%
<i>PWE<sub>slow</sub></i>	+17%	-26.6%	-16.9%
<i>PWE<sub>fast</sub></i>	+19.8%	-38.2%	-25.4%

► 105

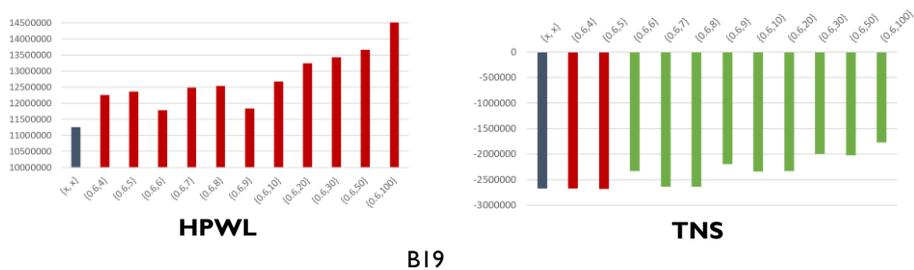
Timing Driven Global Placement

7/6/2025

105

## Experimental Results

### ► Low $w_{\max}$ might lead to increase in TNS

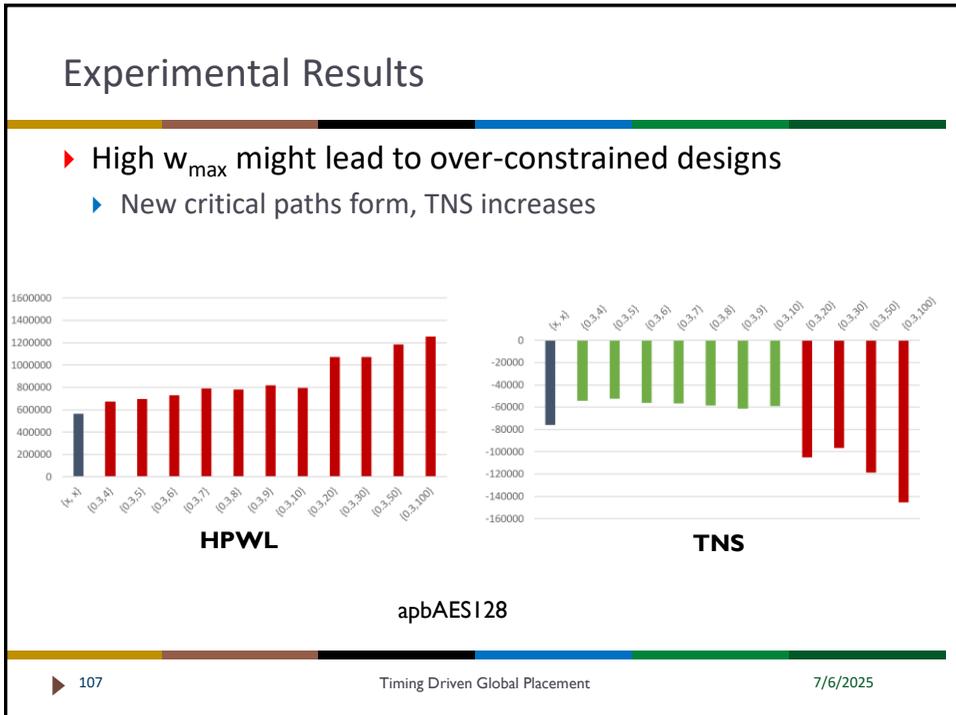


► 106

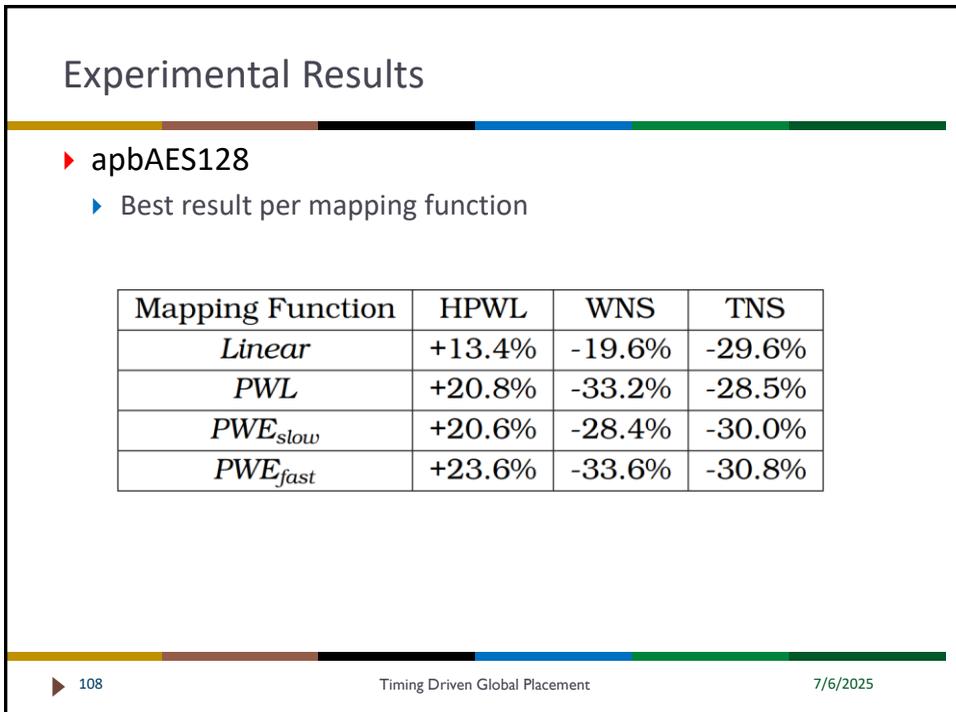
Timing Driven Global Placement

7/6/2025

106



107



108

## Experimental Results

### ▶ Best results per design

Design	HPWL	WNS	TNS
<i>PID</i>	+19.4%	-42.0%	-40.0%
<i>AES</i>	+37.7%	-58.7%	-80%
<i>AES192</i>	+23.4%	-25.7%	-89.4%
<i>apbAES128</i>	+23.6%	-33.6%	-30.8%
<i>LDPC</i>	+6.9%	-38.2%	-43.3%
<i>B19</i>	+30.3%	-35.8%	-33.5%
Average	+23.55%	-39.0%	-52.8%

▶ 109

Timing Driven Global Placement

7/6/2025

109

## Experimental Results

### ▶ After Legalization

Increase after Legalization			
Designs	HPWL	WNS	TNS
PID	3%	3%	3%
AES	2%	12%	33%
AES192	3%	4%	277%
apbAES182	3%	1%	2%
LDPC	1%	0%	2%
b19	2%	3%	4%

Still **72.8%** better than non-TD!

Still **60%** better than non-TD!

▶ 110

Timing Driven Global Placement

7/6/2025

110



Thanks for your time!



▶ 111

Timing Driven Global Placement

7/6/2025

111

## Outline



- ▶ Introduction to EDA
  - ▶ Global Placement
  - ▶ Legalization
- ▶ Introduction to Timing
  - ▶ Net Based Approaches
  - ▶ Path Based Approaches
- ▶ Timing Driven Global Placement
  - ▶ Kraftwerk2 Algorithm
  - ▶ Mapping Idea
  - ▶ Weight Functions
  - ▶ Timing Driven Kraftwerk2
- ▶ Experimental Results
  - ▶ Flow
  - ▶ Results
- ▶ Conclusions and Future Work
- ▶ References



▶ 112

Timing Driven Global Placement

7/6/2025

112

## Kraftwerk2 Algorithm

---

- ▶ Demand and Supply System

$$D(x, y) = D_{dem}(x, y) - D_{sup}(x, y)$$

↓

quantifies overlap  
at point (x,y)

↓

measures the number  
of components overlapping  
at point (x,y)

↓

1 -> if free space exist at point (x,y)  
0 -> otherwise

- ▶ System  $D(x,y)$  is interpreted as a charge distribution which creates an electrostatic potential  $\Phi(x,y)$

---

▶ 113
Timing Driven Global Placement
7/6/2025

113

## Kraftwerk2 Algorithm

---

- ▶ Calculate  $\Phi(x,y)$  by Poisson equation

$$\left( \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right) \Phi(x, y) = -D(x, y)$$

- ▶ The electric field produced by the electrostatic potential

$$\mathbf{E}(x, y) = \nabla \Phi(x, y)$$

- ▶ Force in the x-direction is

$$\mathbf{F}_x^{move}(x) = q_{charge,x} \cdot \mathbf{E}(x)$$

↓

user-defined

↓

calculated

---

▶ 114
Timing Driven Global Placement
7/6/2025

114

## Kraftwerk2 Algorithm

- ▶ Represent move force as a spring force

$$\mathbf{F}_x^{move}(x) = q_{charge,x} \cdot \mathbf{E}(x)$$

$$\mathbf{F}_x^{move}(x_i) = \dot{\omega}_i \cdot (x_i - \dot{x}_i)$$

target point

use above equation

- ▶ Calculate the target point position

$$\dot{x}_i = x'_i - \mathbf{E}(x'_i) = x'_i - \left. \frac{\partial}{\partial x} \Phi(x, y) \right|_{(x'_i, y'_i)}$$

position at the beginning  
of placement iteration

▶ 115

Timing Driven Global Placement

7/6/2025

115

## Kraftwerk2 Algorithm

- ▶ Move force to vector-matrix notation

$$\mathbf{F}_x^{move} = \dot{\mathbf{C}}_x (\mathbf{x} - \dot{\mathbf{x}})$$

$$\dot{\mathbf{C}}_x = \text{diag}(\dot{\omega}_i)$$

$$\dot{\mathbf{x}} = \mathbf{x}' - \Phi_x$$

$$\Phi_x = [(\partial/\partial x)\Phi(x, y)|_{(x'_1, y'_1)}, (\partial/\partial x)\Phi(x, y)|_{(x'_2, y'_2)}, \dots, (\partial/\partial x)\Phi(x, y)|_{(x'_n, y'_n)}]^T$$

▶ 116

Timing Driven Global Placement

7/6/2025

116