EE582
Physical Design Automation of VLSI Circuits and Systems

Prof. Dae Hyun Kim
School of Electrical Engineering and Computer Science
Washington State University

Interconnect Optimization

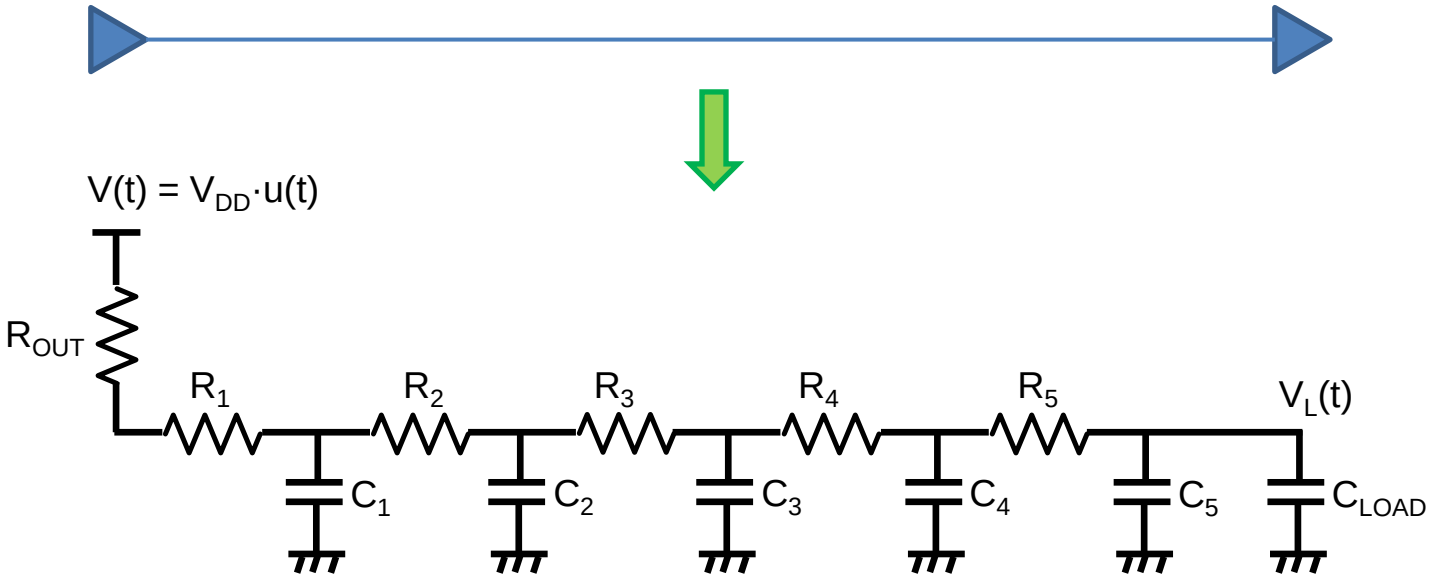
WASHINGTON STATE UNIVERSITY

What We Will Study

- Interconnect analysis
  - Elmore delay
  - Simple timing analysis
- Interconnect Optimization
  - Dynamic-programming-based buffer insertion
    - Two-pin nets
    - Multi-pin nets
    - Single buffer type
    - Multiple buffer types

Interconnect Analysis

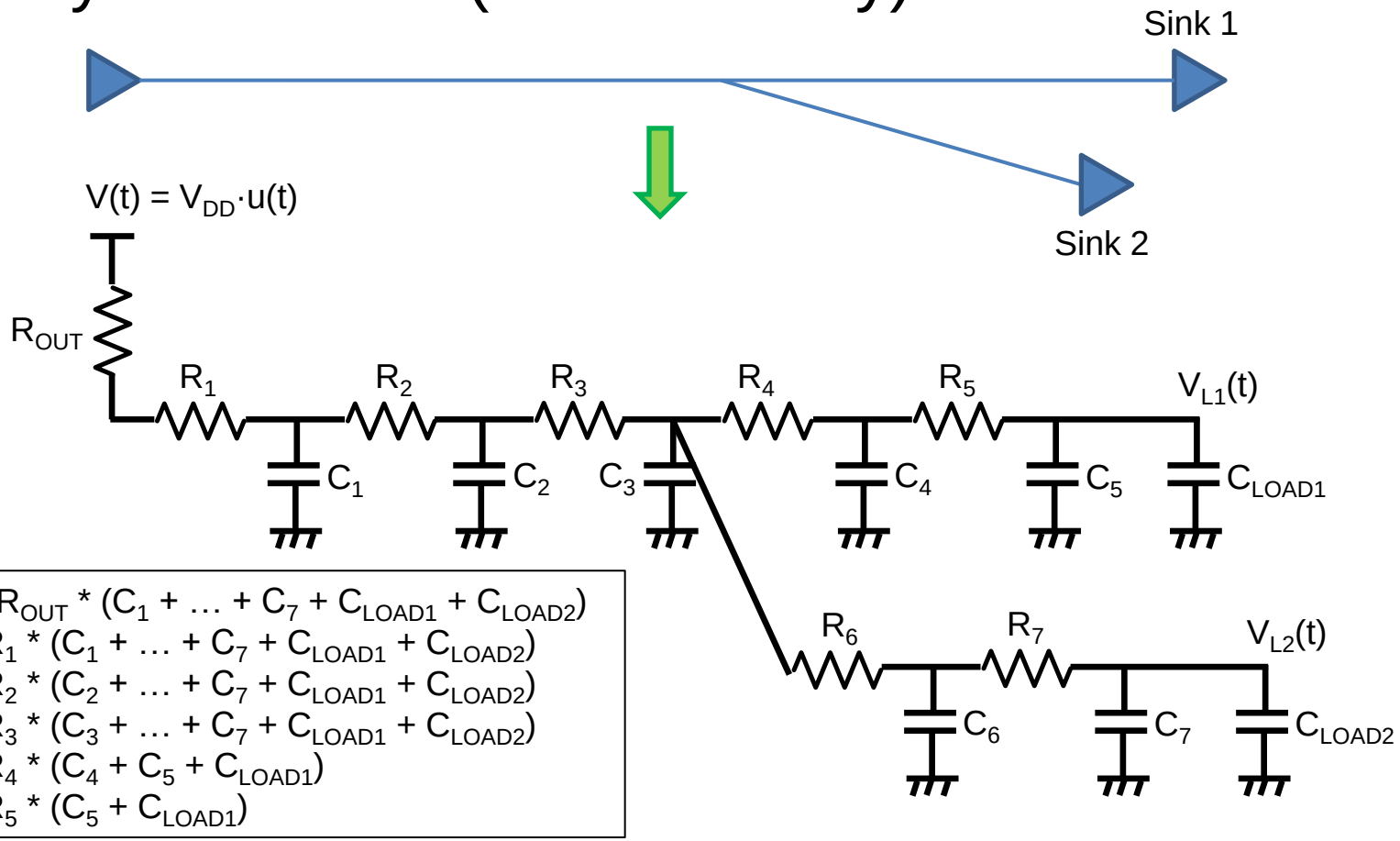
- Delay calculation (Elmore delay)



$$\tau = R_{OUT} * (C_1 + \ldots + C_5 + C_{LOAD}) + R_1 * (C_1 + \ldots + C_5 + C_{LOAD}) + R_2 * (C_2 + \ldots + C_5 + C_{LOAD}) + R_3 * (C_3 + \ldots + C_5 + C_{LOAD}) + R_4 * (C_4 + C_5 + C_{LOAD}) + R_5 * (C_5 + C_{LOAD})$$

Interconnect Analysis

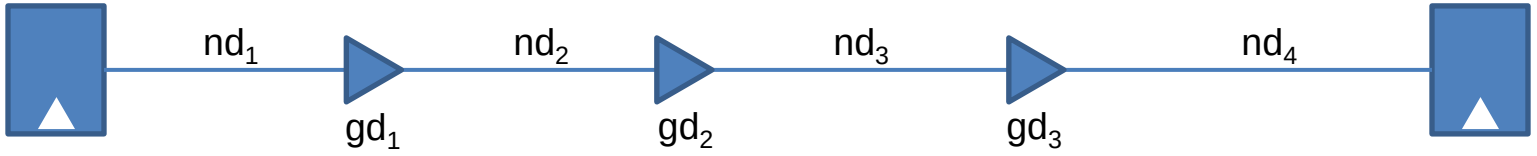
- Delay calculation (Elmore delay)



$$\tau_1 = R_{OUT} * (C_1 + \ldots + C_7 + C_{LOAD1} + C_{LOAD2}) + R_1 * (C_1 + \ldots + C_7 + C_{LOAD1} + C_{LOAD2}) + R_2 * (C_2 + \ldots + C_7 + C_{LOAD1} + C_{LOAD2}) + R_3 * (C_3 + \ldots + C_7 + C_{LOAD1} + C_{LOAD2}) + R_4 * (C_4 + C_5 + C_{LOAD1}) + R_5 * (C_5 + C_{LOAD1})$$

Interconnect Analysis

- Timing analysis

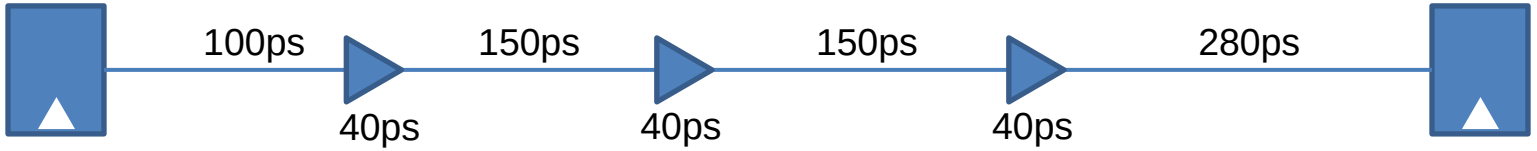


  - CLK period: T_{CLK}
  - Path delay: $d_p = nd_1 + gd_1 + nd_2 + gd_2 + nd_3 + gd_3 + nd_4$
  - Setup time: t_s

  - Required Arrival Time (RAT) = $T_{CLK} – t_s$
  - Arrival Time (AT) = Signal delay = Path delay = d_p
  - Slack = RAT – AT = $T_{CLK} – t_s - d_p$

Interconnect Analysis

- Timing analysis



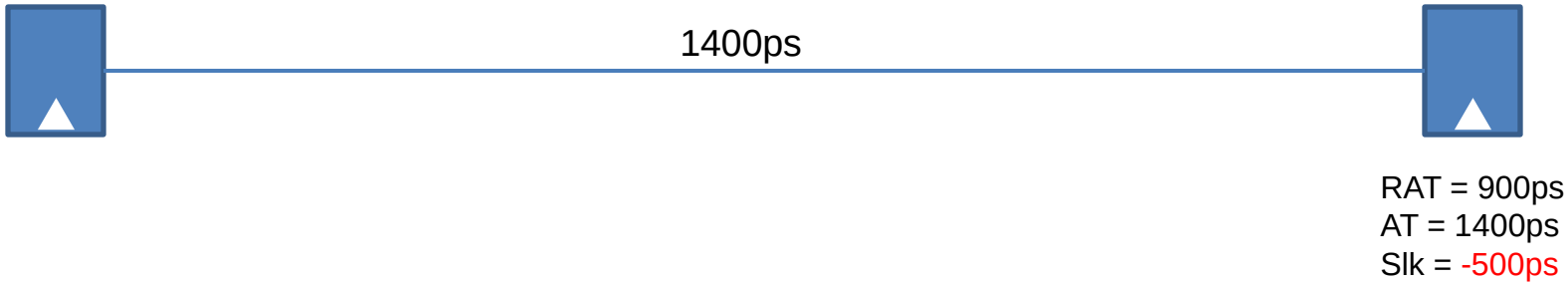
RAT = 900ps
AT = 800ps
Slk = +100ps

  - Required Arrival Time (RAT) = $T_{CLK} – t_s$
  - Arrival Time (AT) = Signal delay = Path delay = d_p
  - Slack = RAT – AT = $T_{CLK} – t_s - d_p$

  - Example
    - T_{CLK}: 1ns
    - d_p: 800ps
    - t_s: 100ps
    - Slack: 1ns – (800ps) – 100ps = +100ps

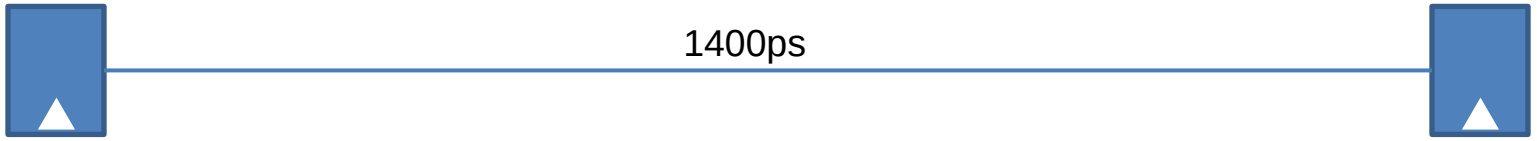
Interconnect Analysis

- Timing analysis
  - Positive slack: We have some margin for optimization.
  - Negative slack: We are violating the timing constraint.



Interconnect Optimization

- How to optimize interconnects
  - Gate sizing
  - Wire sizing
  - Buffer insertion
  - Placement adjustment
  - Gate decomposition
  - New routing topology generation
  - …



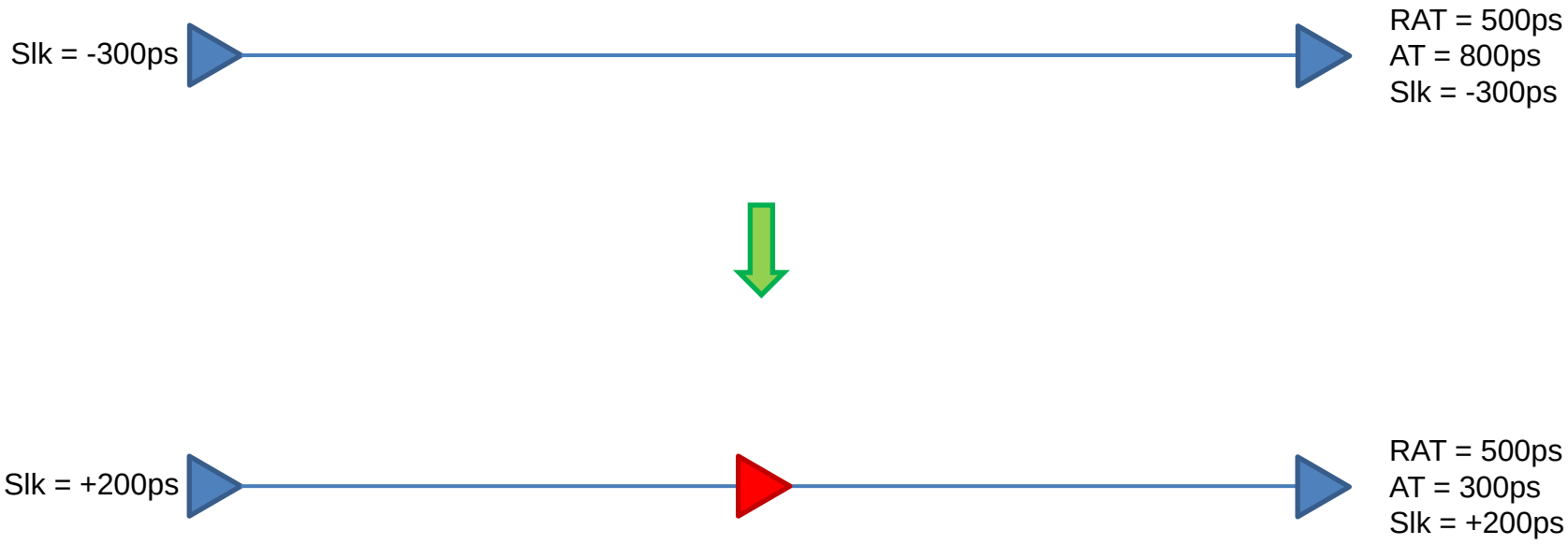
RAT = 900ps
AT = 1400ps
Slk = -500ps

Interconnect Optimization

- Problem definition for buffer insertion
  - Given
    - An RC tree or a routing topology
    - Characteristics (R_{OUT}) of the driver node
    - Characteristics (C_{LOAD}) of the sink nodes
    - Characteristics (R_{OUT} and C_{IN}) of a buffer type
    - Characteristics (r_{wire} and c_{wire}) of the wires
    - Timing constraints (RAT at each sink node)
    - Bufferable locations
  - Maximize
    - Slack at the driver node

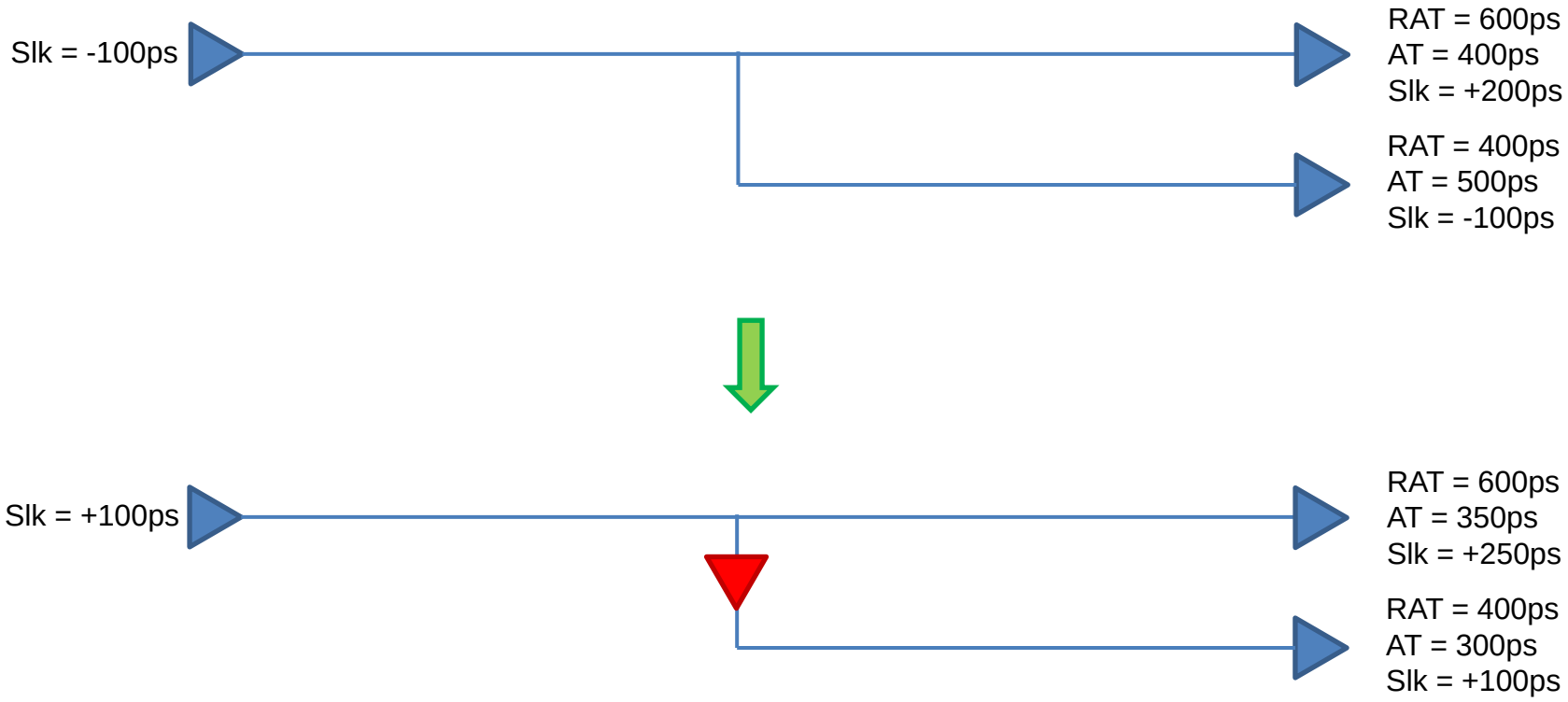
Interconnect Optimization

- Example



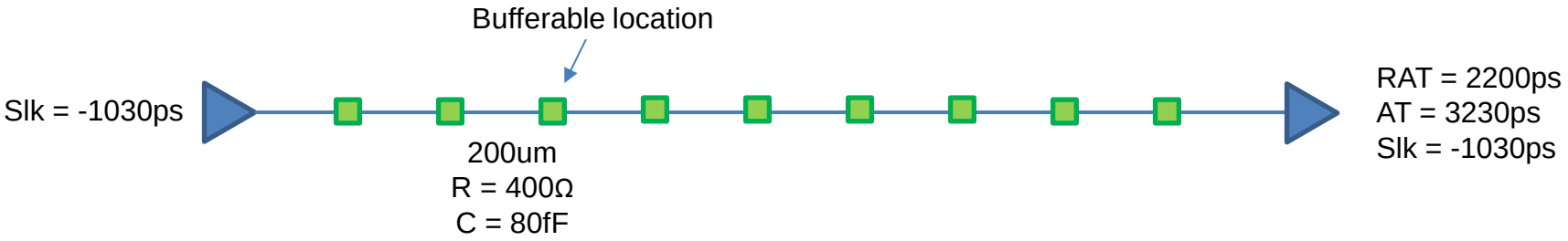
Interconnect Optimization

- Example



Interconnect Optimization

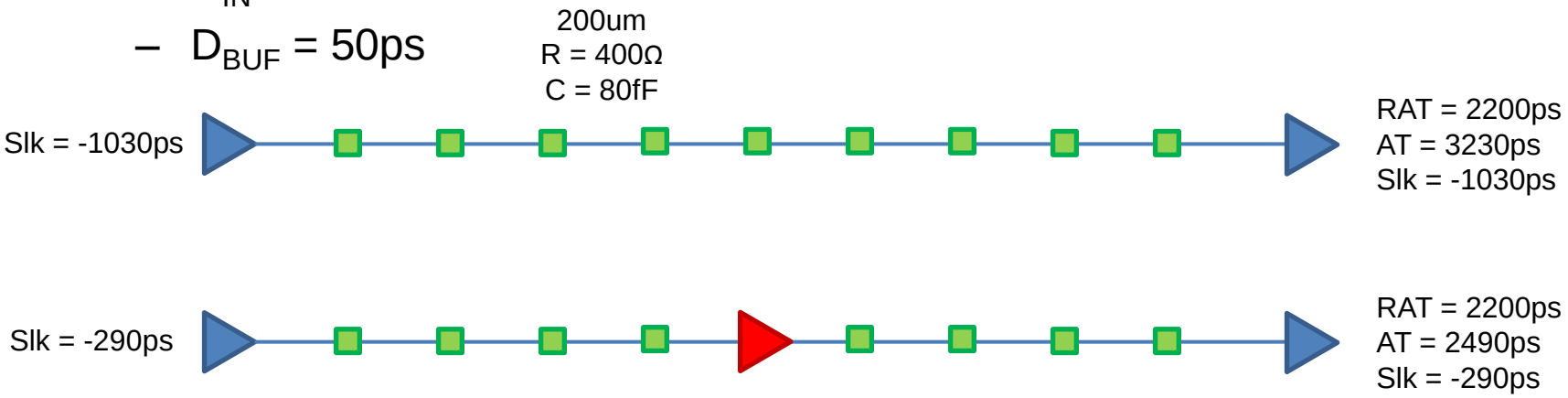
- Example
  - r_{wire} = 2Ω/um
  - c_{wire} = 0.4fF/um
  - R_{OUT} = 2kΩ
  - C_{IN} = 5fF
  - D_{BUF} = 50ps



Delay = 2k * (805f) + 0.4k * (765f) + 0.4k * (685f) + 0.4k * (605f) + … + 0.4k * (125f) + 0.4k * (45f) = 3230ps

Interconnect Optimization

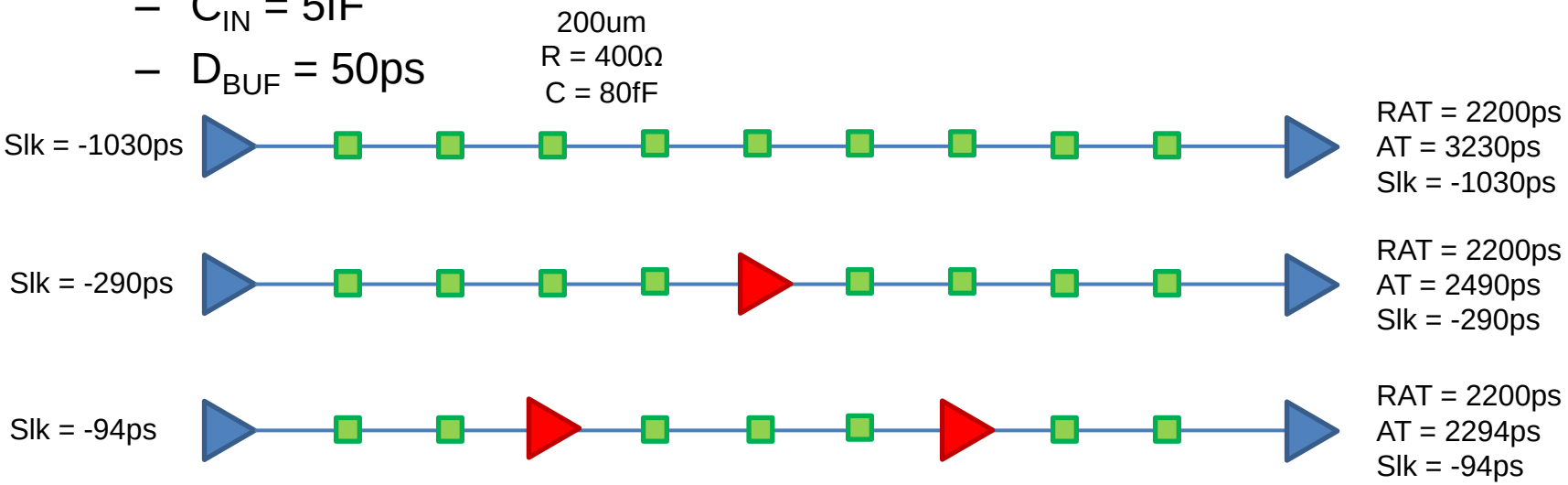
- Example
  - r_{wire} = 2Ω/um
  - c_{wire} = 0.4fF/um
  - R_{OUT} = 2kΩ
  - C_{IN} = 5fF
  - D_{BUF} = 50ps



Delay = 2k * (405f) + 0.4k * (365f) + 0.4k * (285f) + 0.4k * (205f) + 0.4k * (125f) + 0.4k * (45f)
+ 50ps (buffer delay)
+ 2k * (405f) + 0.4k * (365f) + 0.4k * (285f) + 0.4k * (205f) + 0.4k * (125f) + 0.4k * (45f)
= 1220ps + 50ps + 1220ps = 2490ps

Interconnect Optimization

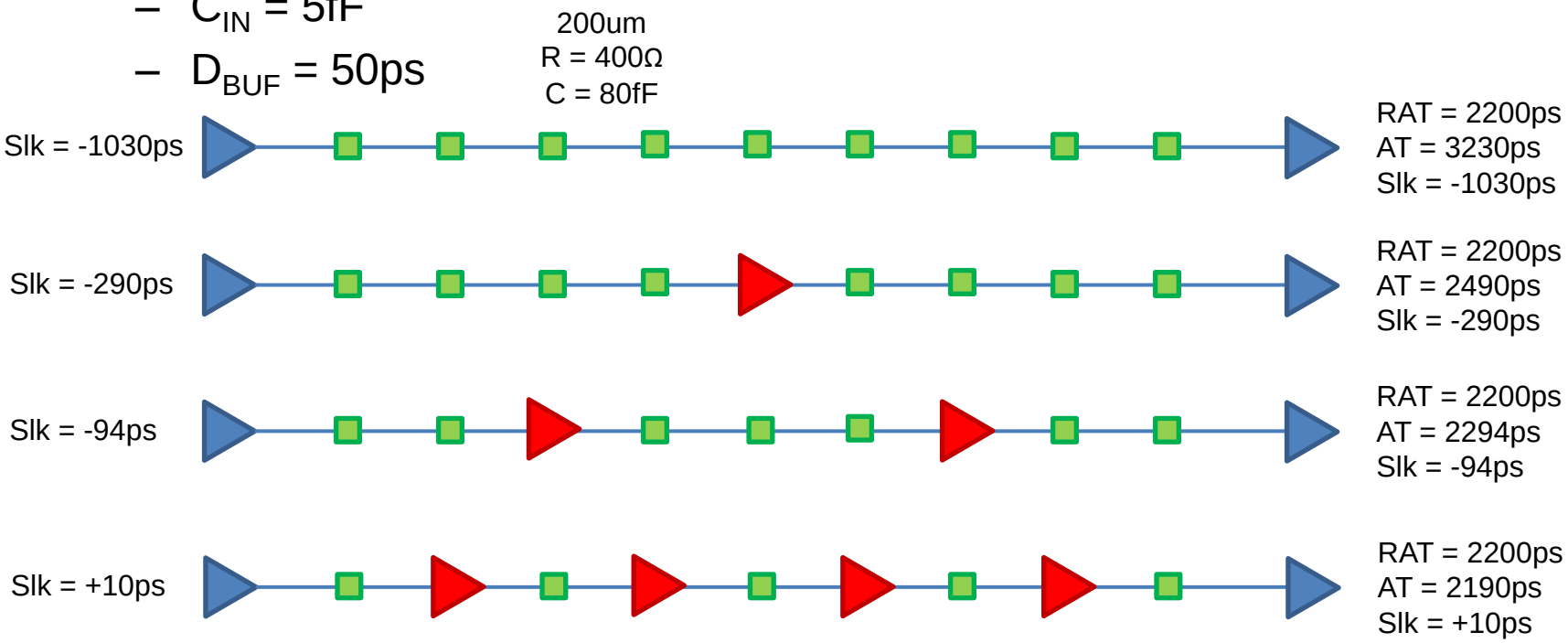
- Example
  - R_{OUT} = 2kΩ
  - C_{IN} = 5fF
  - D_{BUF} = 50ps



Delay = 2k * (245f) + 0.4k * (205f) + 0.4k * (125f) + 0.4k * (45f)
+ 50ps
+ 2k * (325f) + 0.4k * (285f) + 0.4k * (205f) + 0.4k * (125f) + 0.4k * (45f)
+ 50ps
+ 2k * (245f) + 0.4k * (205f) + 0.4k * (125f) + 0.4k * (45f)
= 640ps + 50ps + 914ps + 50ps + 640ps = 2294ps

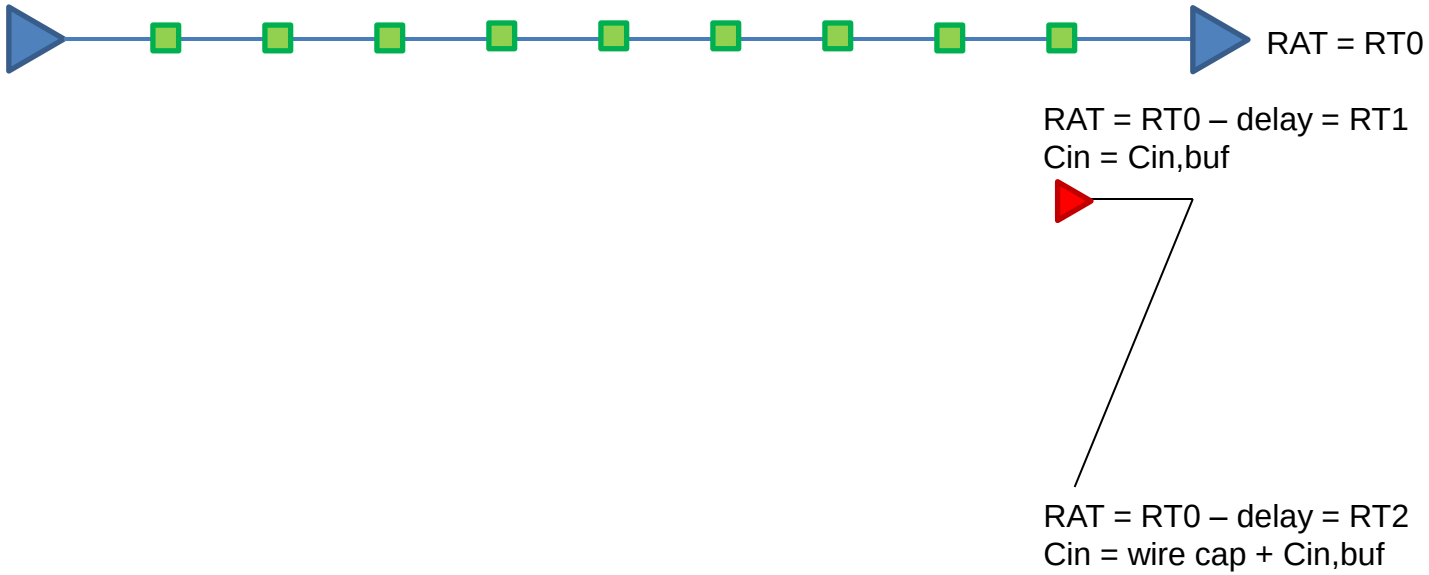
Interconnect Optimization

- Example
  - R_{OUT} = 2kΩ
  - C_{IN} = 5fF
  - D_{BUF} = 50ps



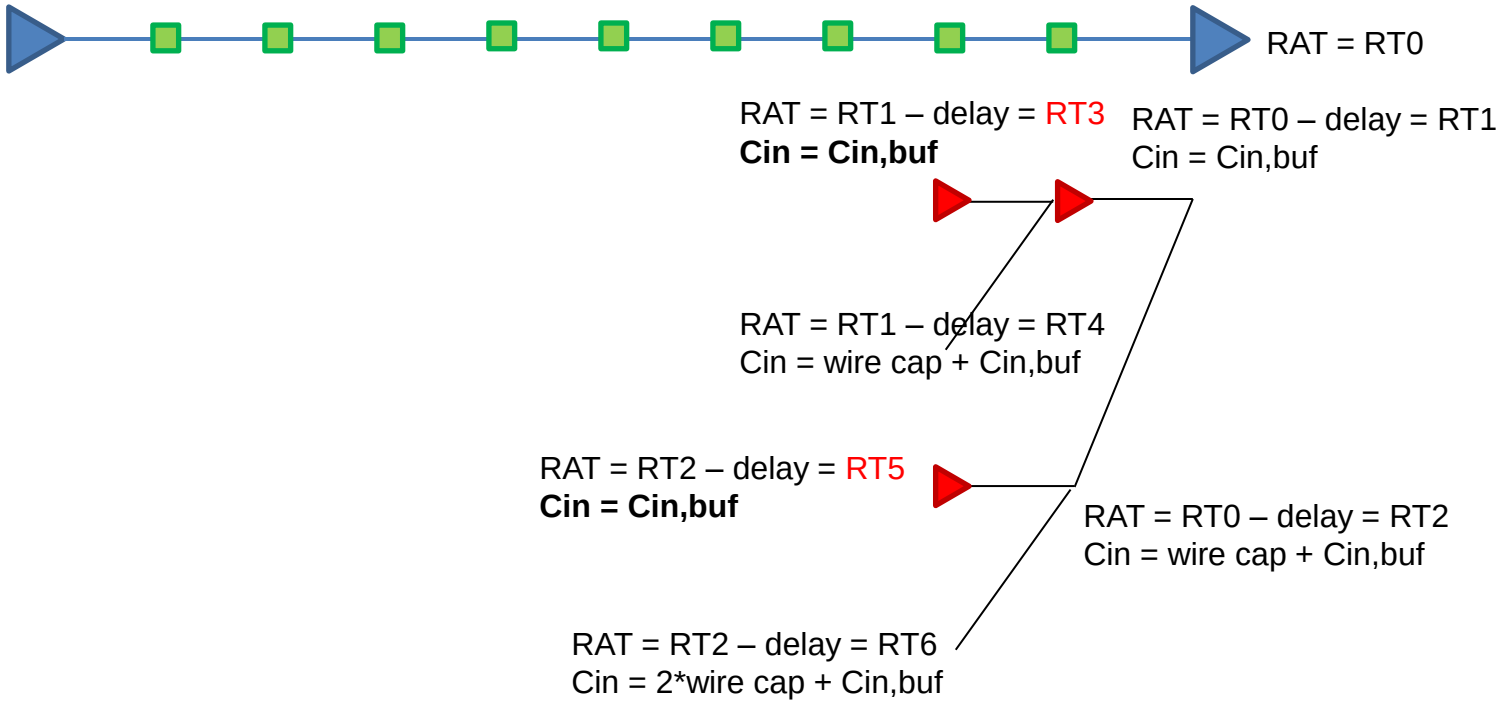
Interconnect Optimization

- Dynamic programming-based buffer insertion
  - van Ginneken algorithm



Interconnect Optimization

- Dynamic programming-based buffer insertion
  - van Ginneken algorithm



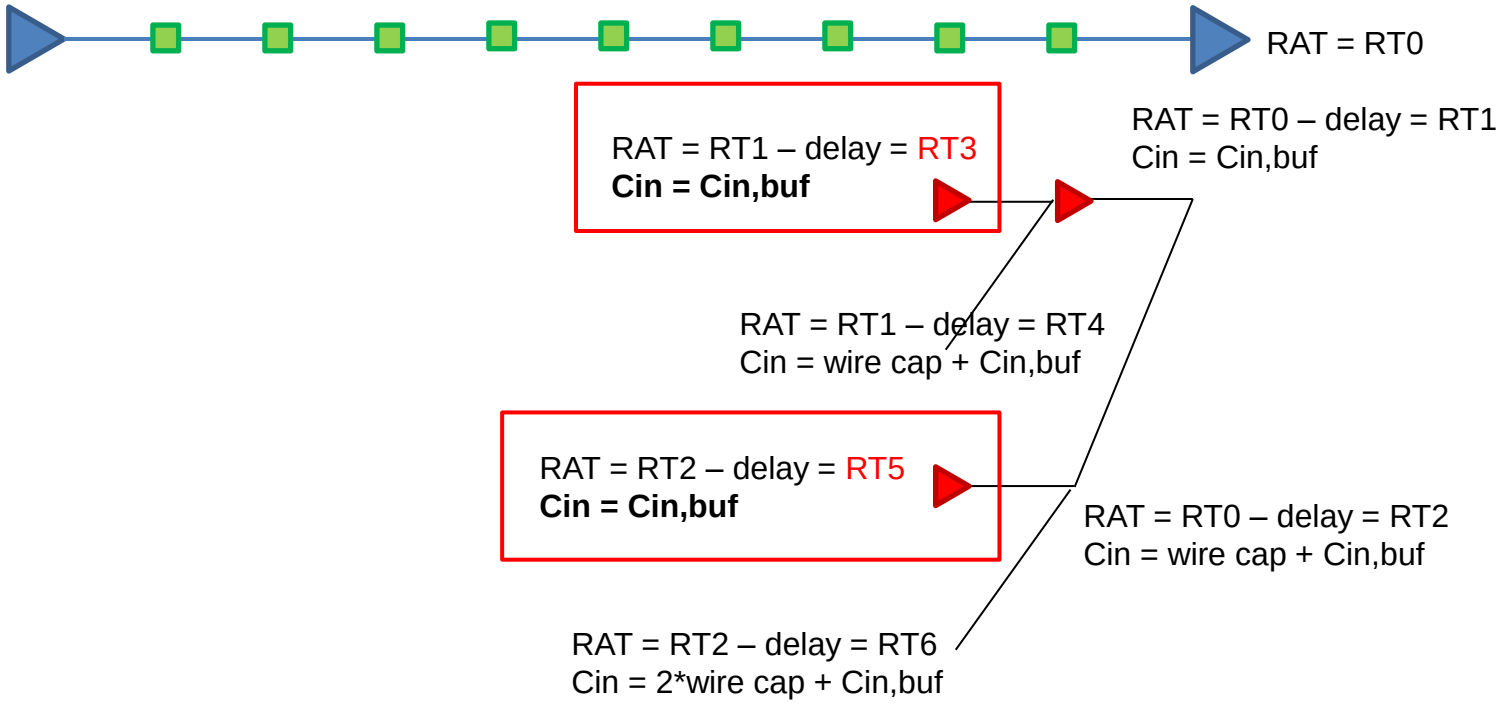
Interconnect Optimization

- Comparison of two candidates

RAT	Cap	Action
RAT (C_1) > RAT (C_2)	Cap (C_1) > Cap (C_2)	Keep both
RAT (C_1) > RAT (C_2)	Cap (C_1) < Cap (C_2)	Keep C_1 only
RAT (C_1) < RAT (C_2)	Cap (C_1) > Cap (C_2)	Keep C_2 only
RAT (C_1) < RAT (C_2)	Cap (C_1) < Cap (C_2)	Keep both

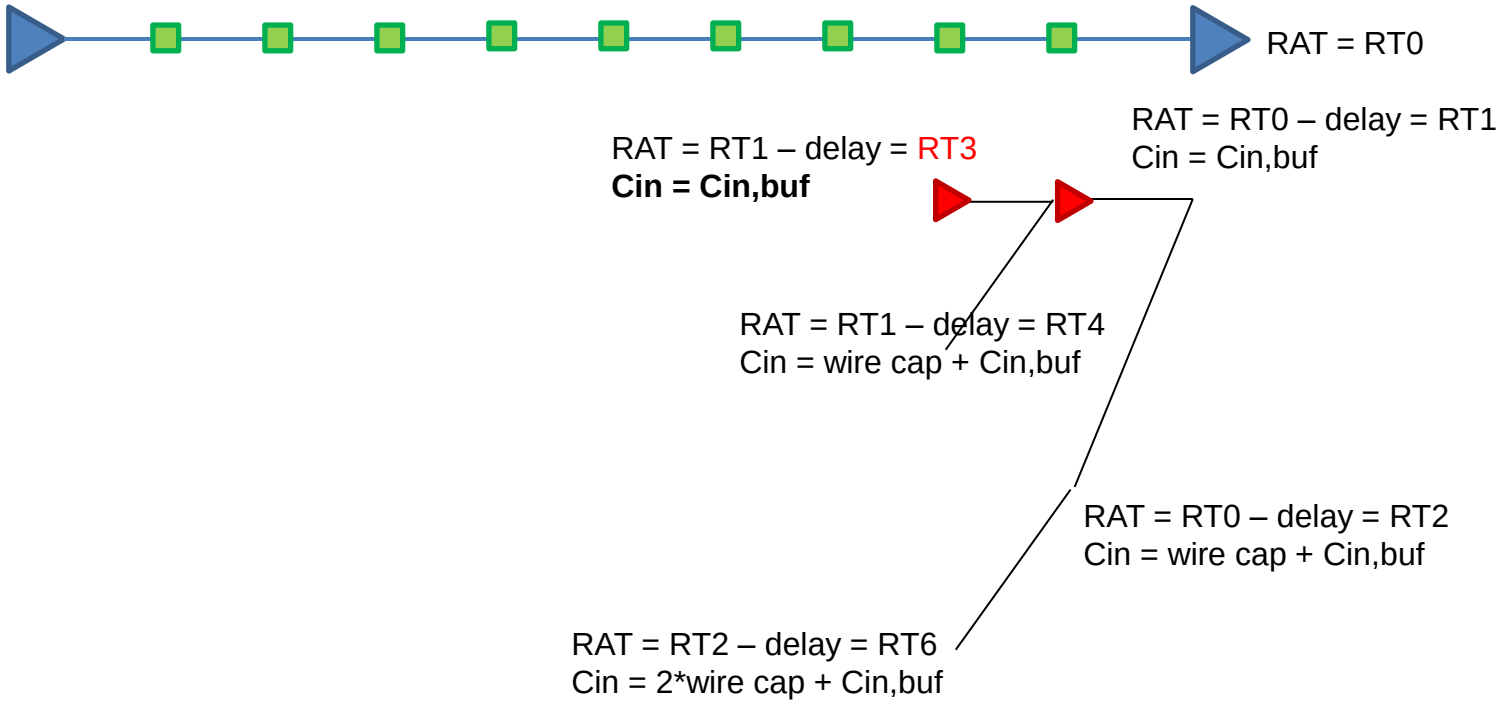
Interconnect Optimization

- Dynamic programming-based buffer insertion
  - van Ginneken algorithm



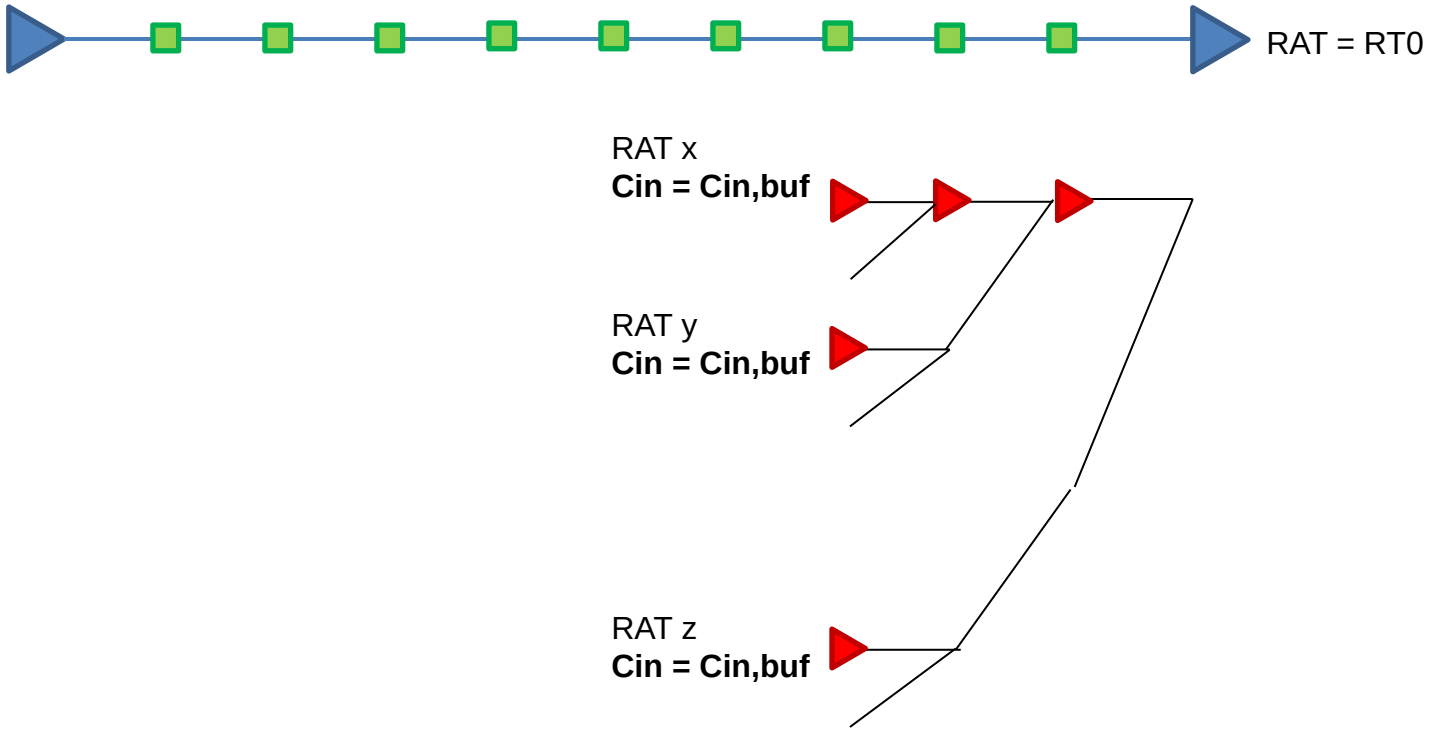
Interconnect Optimization

- Dynamic programming-based buffer insertion
  - van Ginneken algorithm



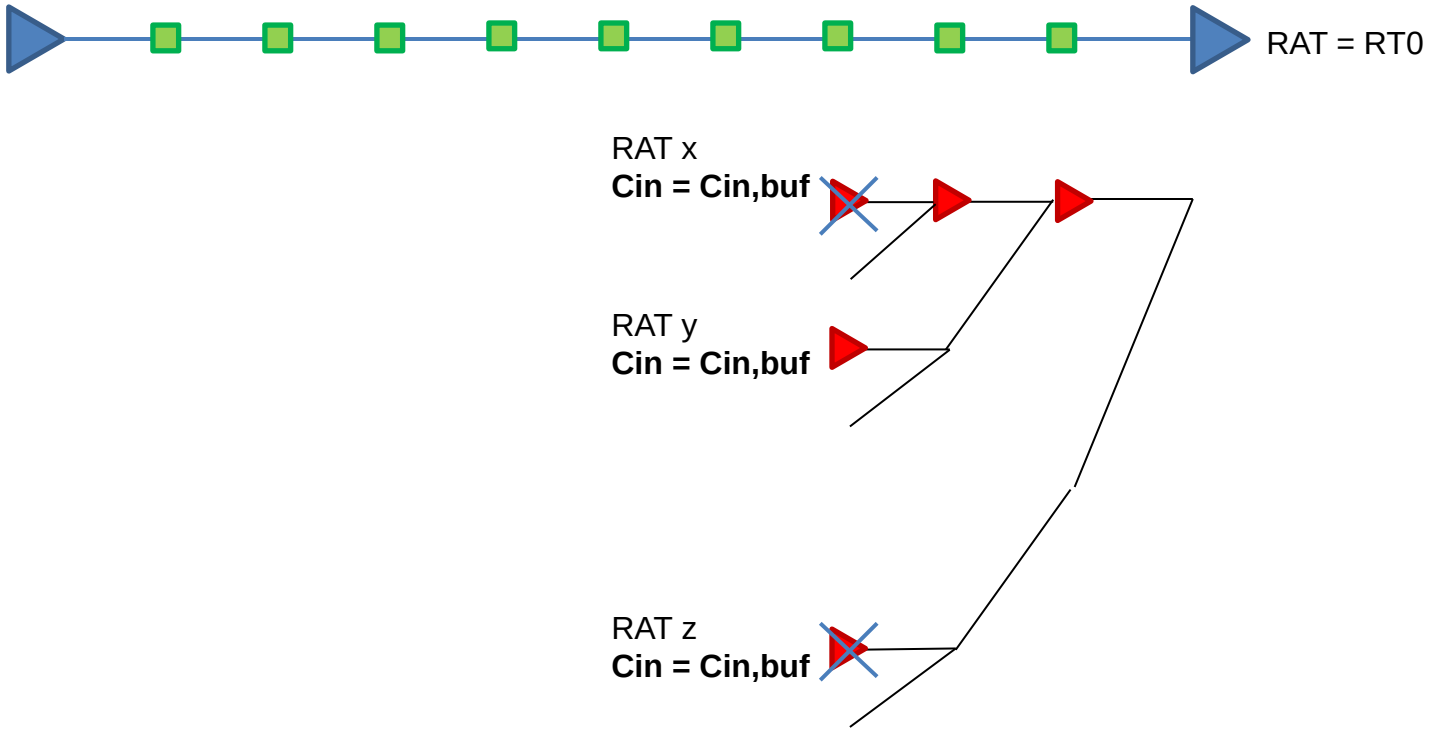
Interconnect Optimization

- Dynamic programming-based buffer insertion
  - van Ginneken algorithm



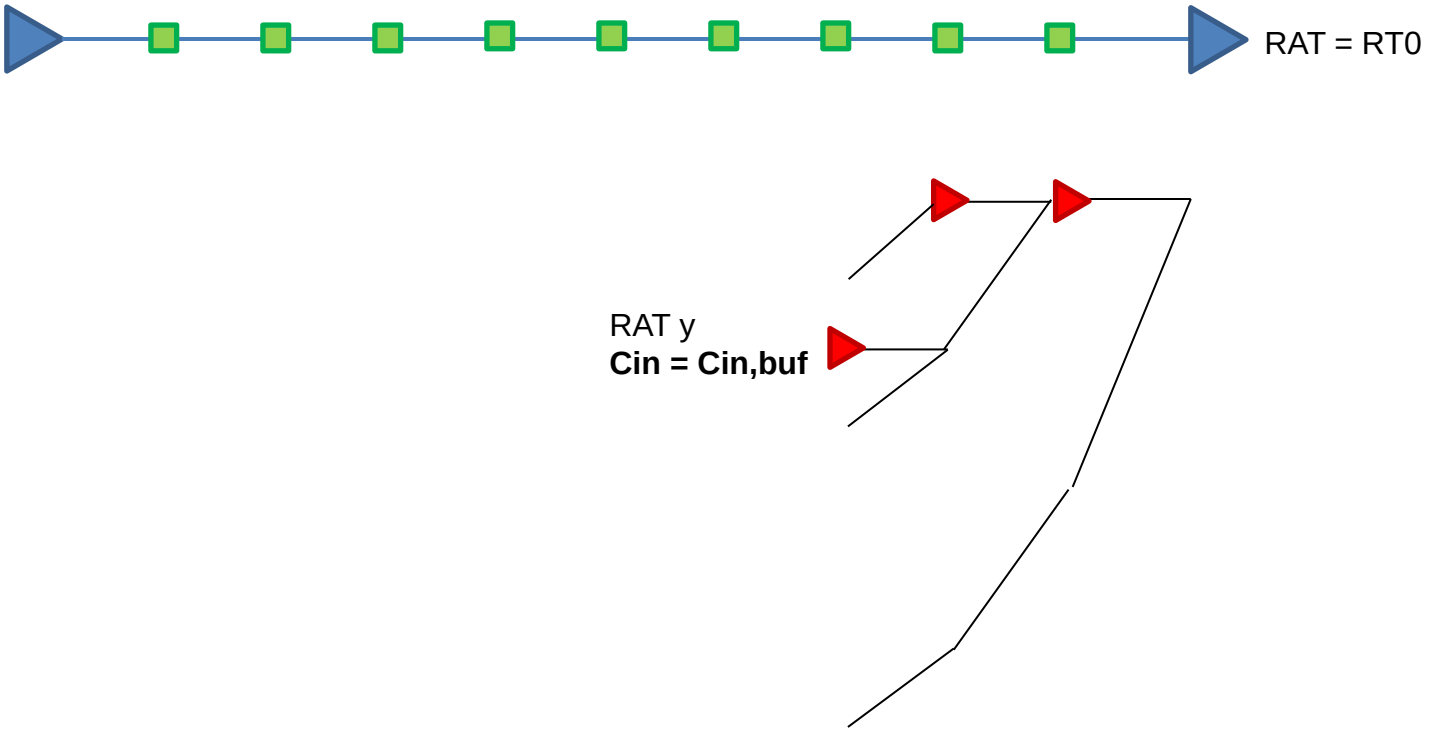
Interconnect Optimization

- Dynamic programming-based buffer insertion
  - van Ginneken algorithm



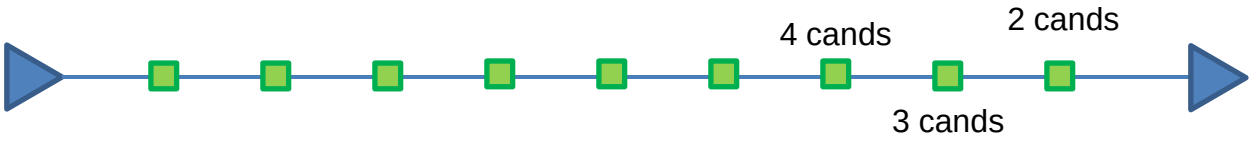
Interconnect Optimization

- Dynamic programming-based buffer insertion
  - van Ginneken algorithm

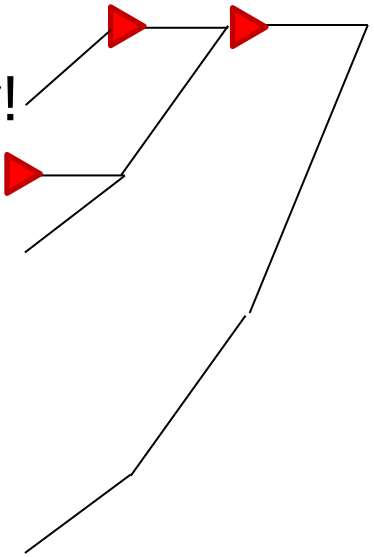


Interconnect Optimization

- How many candidates do we have at each location?

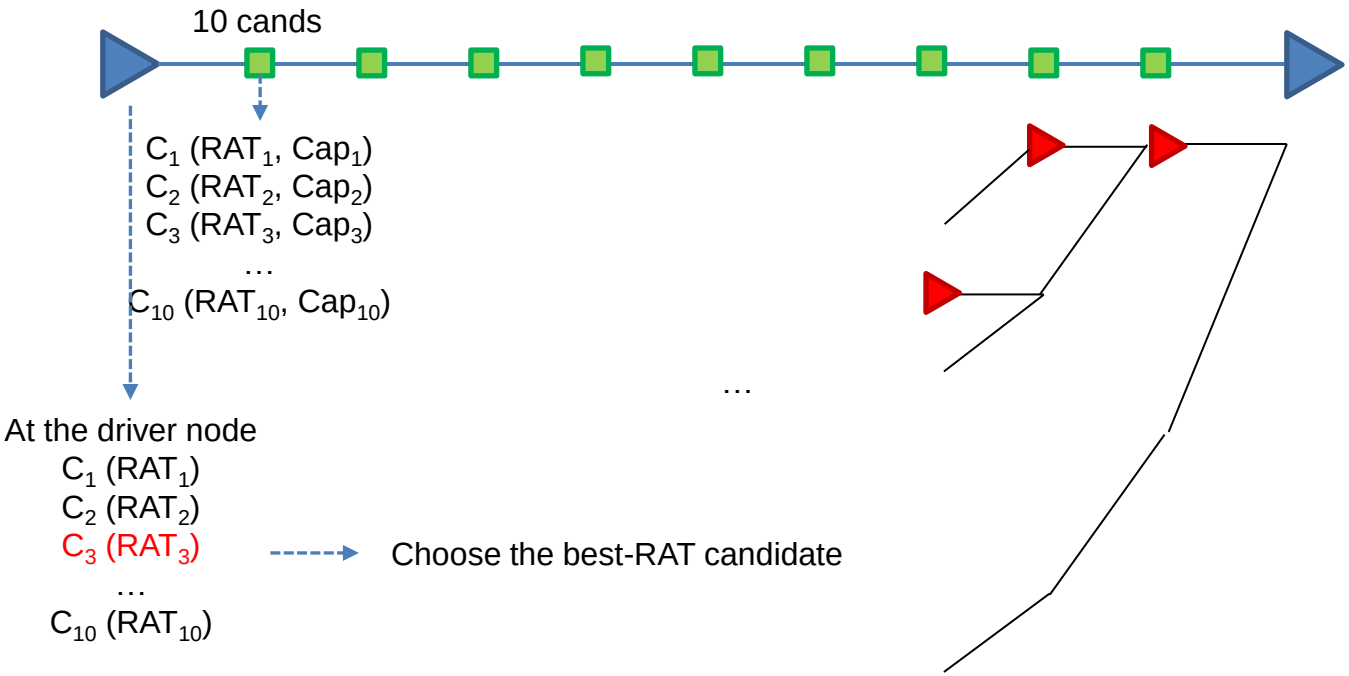


  - # candidates increases linearly!



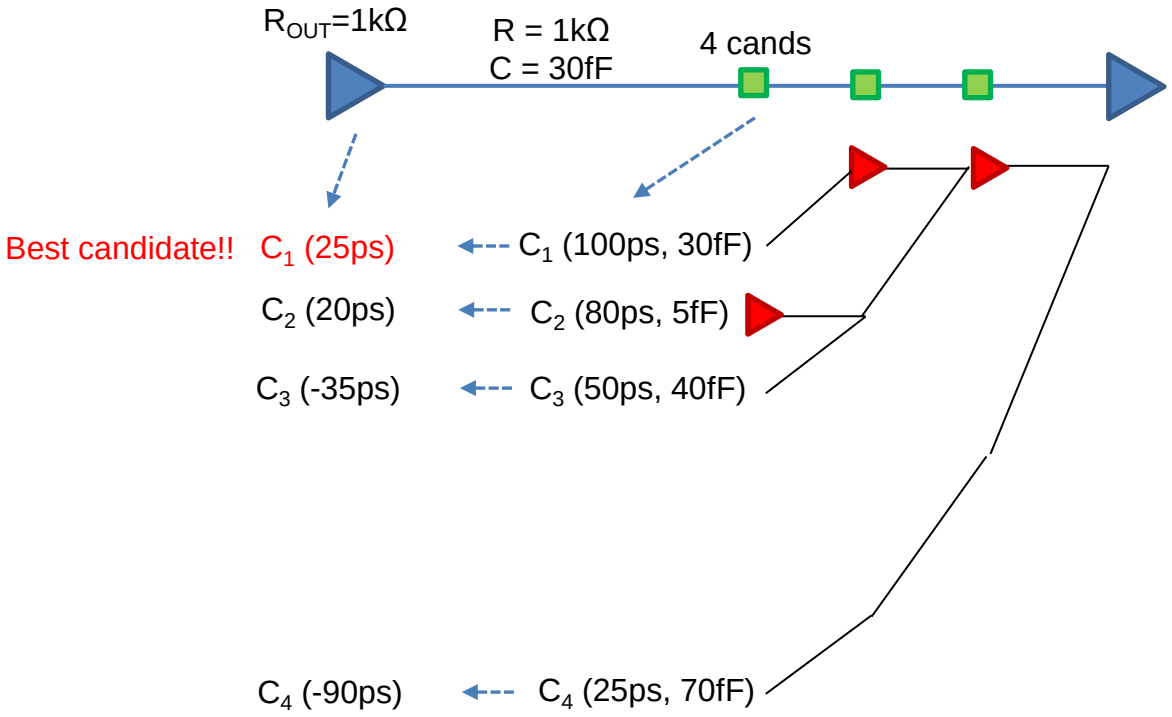
Interconnect Optimization

- After candidate construction



Interconnect Optimization

- Example and back-traversal for the construction of the final solution

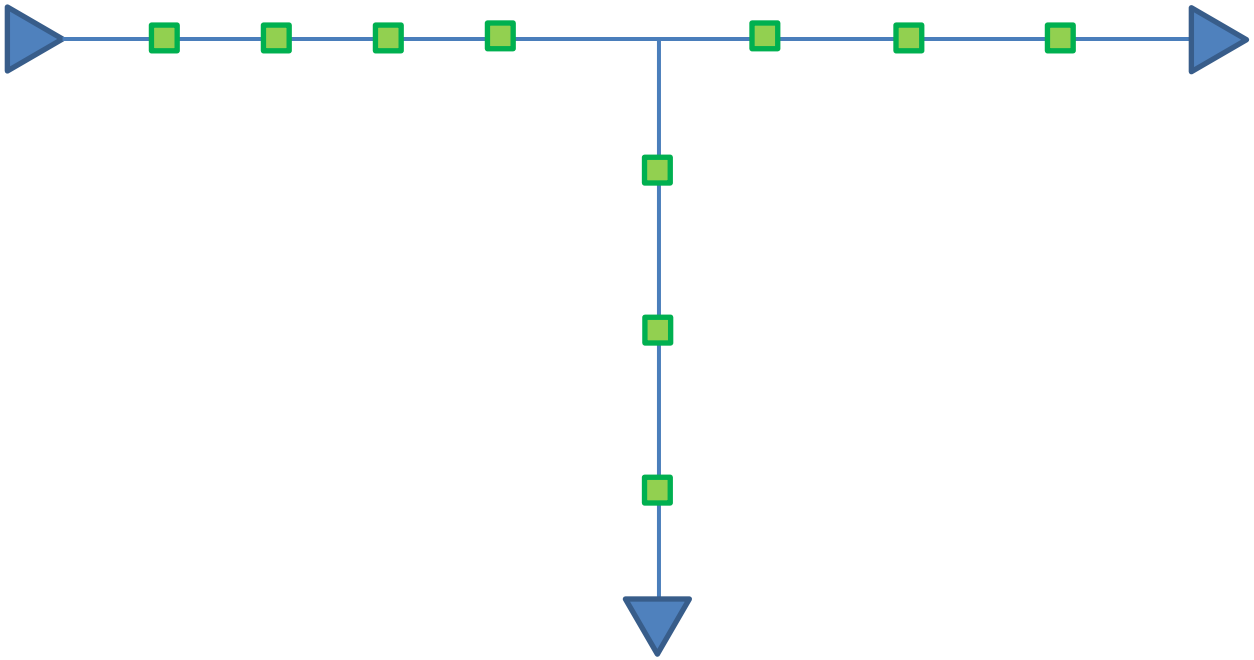


Interconnect Optimization

- Example

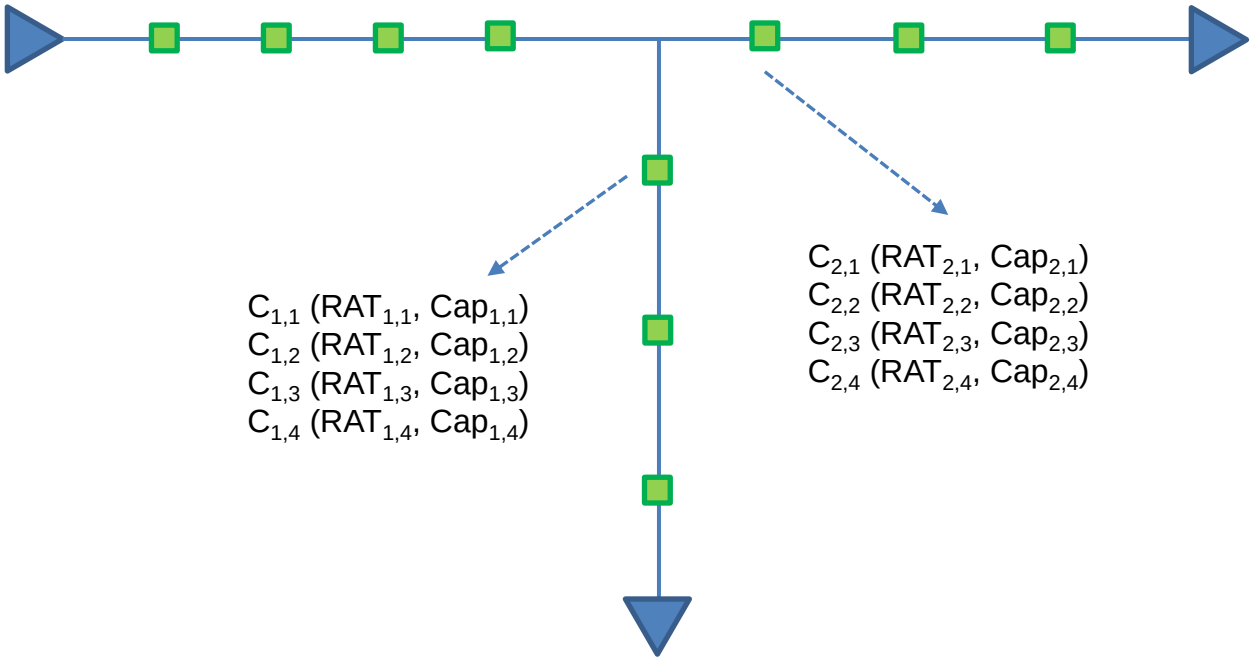
Interconnect Optimization

- Multi-fanout nets



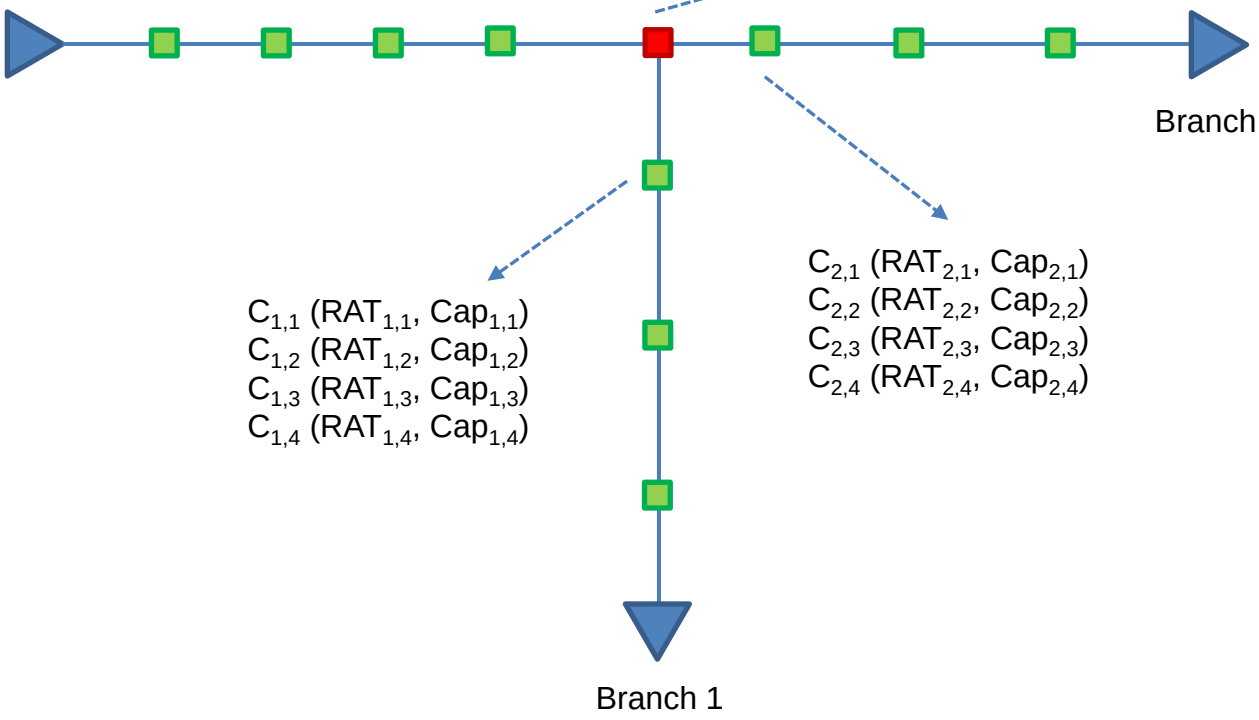
Interconnect Optimization

- Multi-fanout nets



Interconnect Optimization

- Multi-fanout nets



16 combinations

$C_{1,1} + C_{2,1}$
$C_{1,1} + C_{2,2}$
$C_{1,1} + C_{2,3}$
$C_{1,1} + C_{2,4}$

$C_{1,2} + C_{2,1}$
$C_{1,2} + C_{2,2}$
$C_{1,2} + C_{2,3}$
$C_{1,2} + C_{2,4}$

$C_{1,3} + C_{2,1}$
$C_{1,3} + C_{2,2}$
$C_{1,3} + C_{2,3}$
$C_{1,3} + C_{2,4}$

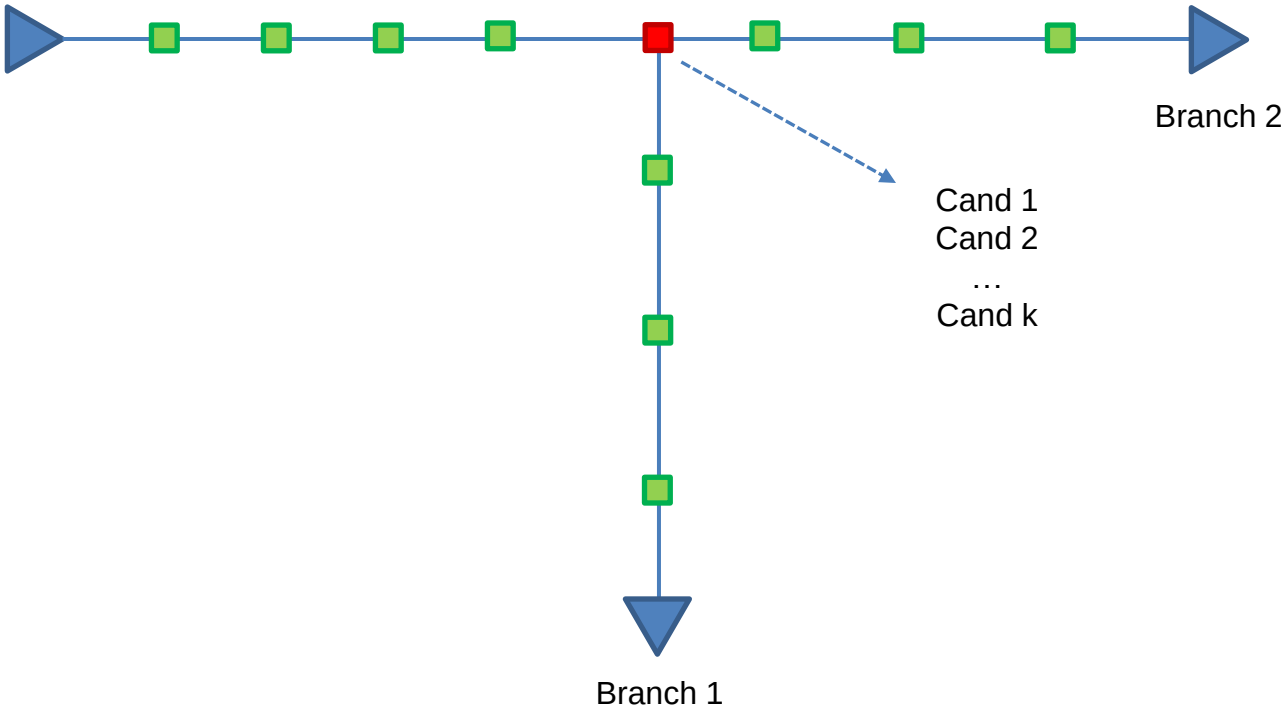
$C_{1,4} + C_{2,1}$
$C_{1,4} + C_{2,2}$
$C_{1,4} + C_{2,3}$
$C_{1,4} + C_{2,4}$

Interconnect Optimization

- Multi-fanout nets
  - (RAT, Cap) of ($C_{1,1}$ + $C_{2,1}$) = (MIN($RAT_{1,1}$, $RAT_{2,1}$), $Cap_{1,1}$ + $Cap_{2,1}$)
  - (RAT, Cap) of ($C_{1,1}$ + $C_{2,2}$) = (MIN($RAT_{1,1}$, $RAT_{2,2}$), $Cap_{1,1}$ + $Cap_{2,2}$)
  - …

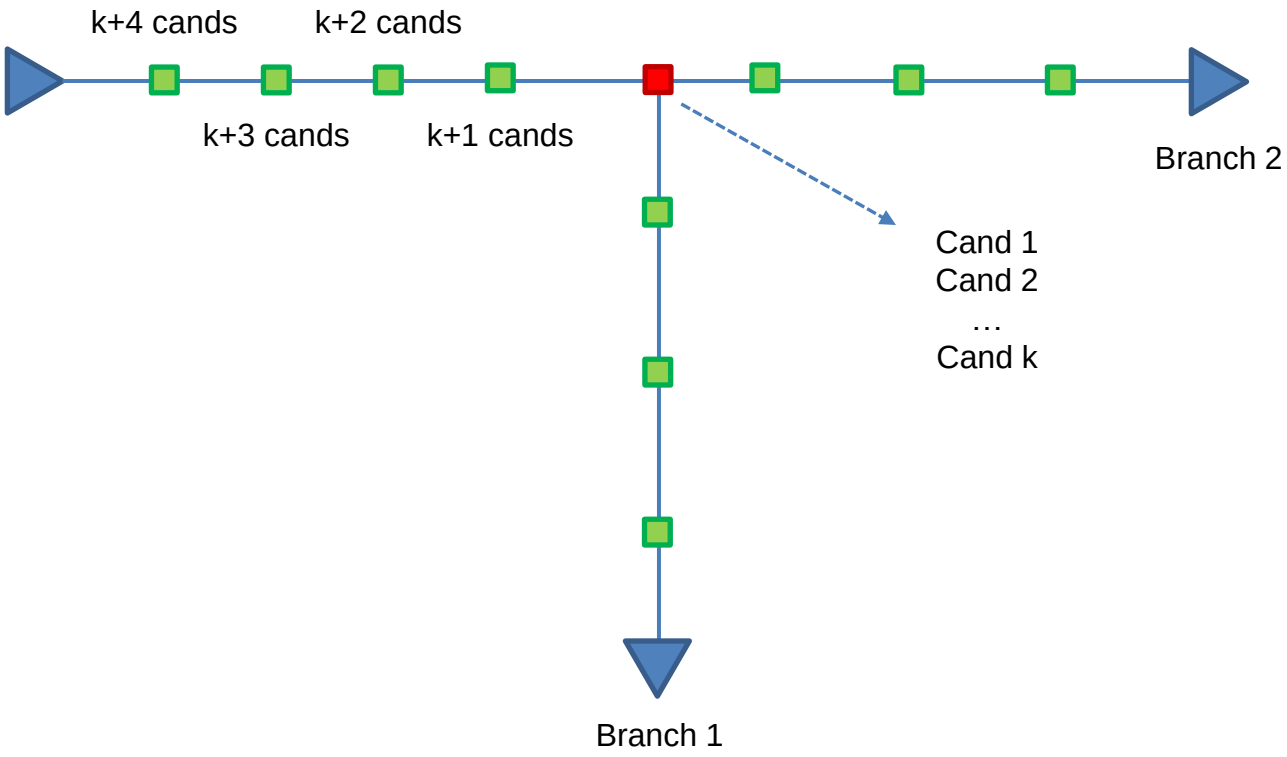
Interconnect Optimization

- Multi-fanout nets



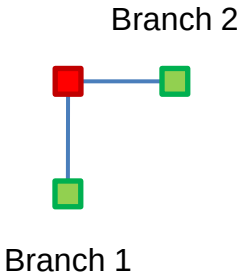
Interconnect Optimization

- Multi-fanout nets



Interconnect Optimization

- # candidates at a branch point
  - a candidates from branch 1
  - b candidates from branch 2



Branch 1

$C_{1,1}$ ($RAT_{1,1}$, $Cap_{1,1}$)
$C_{1,2}$ ($RAT_{1,2}$, $Cap_{1,2}$)
…
$C_{1,a}$ ($RAT_{1,a}$, $Cap_{1,a}$)

Branch 2

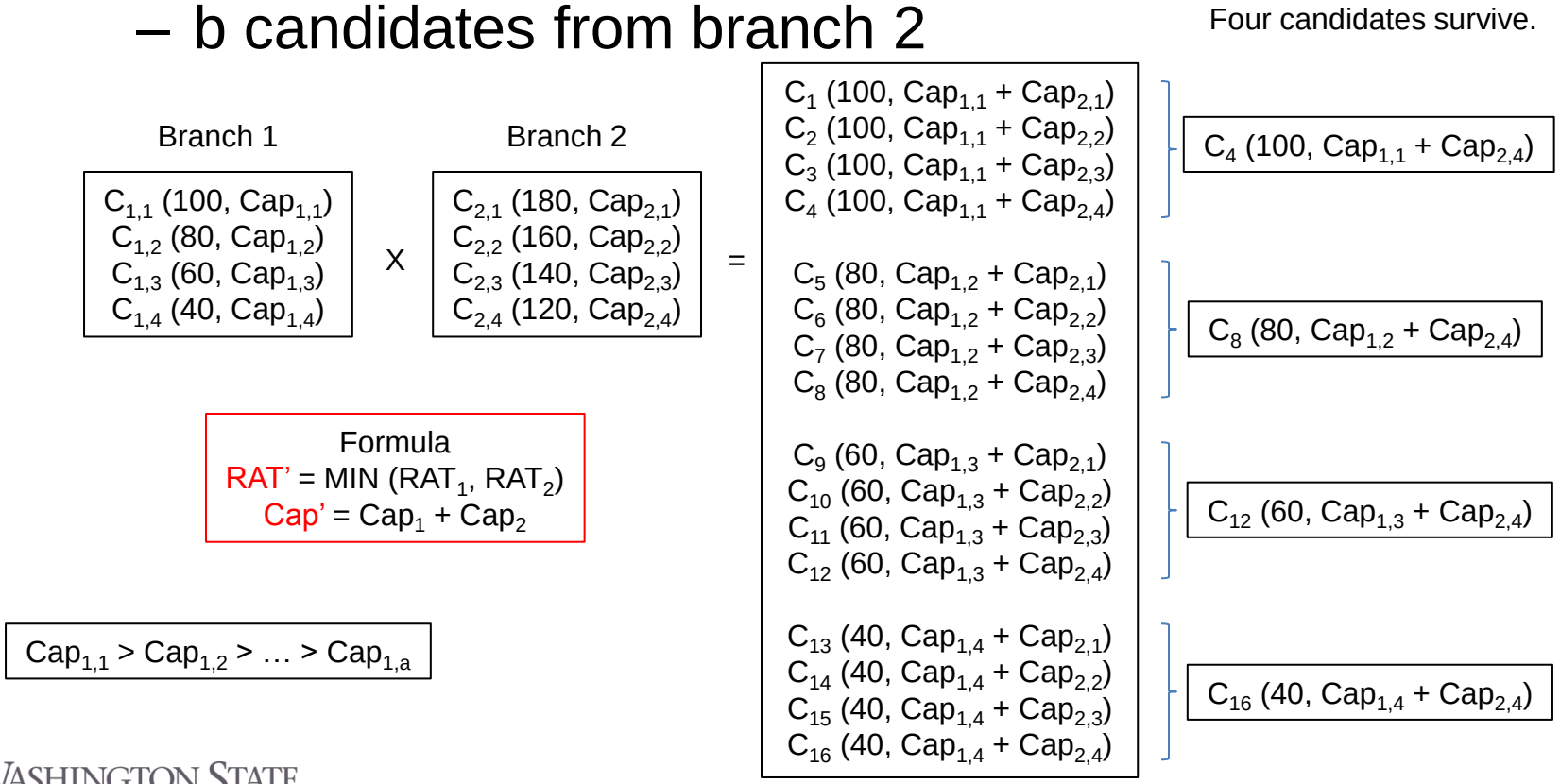
$C_{2,1}$ ($RAT_{2,1}$, $Cap_{2,1}$)
$C_{2,2}$ ($RAT_{2,2}$, $Cap_{2,2}$)
…
$C_{2,b}$ ($RAT_{2,b}$, $Cap_{2,b}$)

$RAT_{1,1} > RAT_{1,2} > \ldots > RAT_{1,a}$
$Cap_{1,1} > Cap_{1,2} > \ldots > Cap_{1,a}$

$RAT_{2,1} > RAT_{2,2} > \ldots > RAT_{2,b}$
$Cap_{2,1} > Cap_{2,2} > \ldots > Cap_{2,b}$

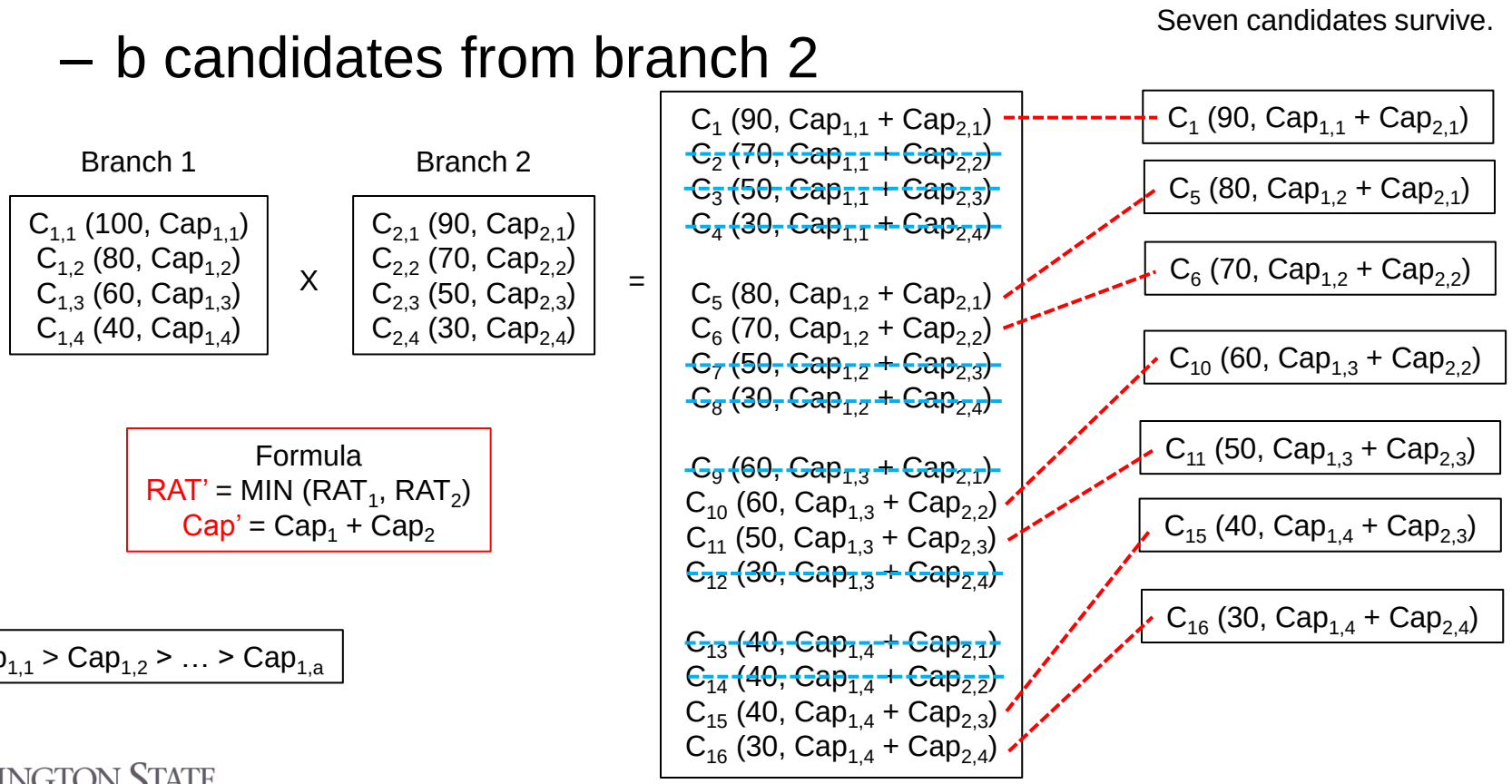
Interconnect Optimization

- # candidates at a branch point
  - a candidates from branch 1
  - b candidates from branch 2



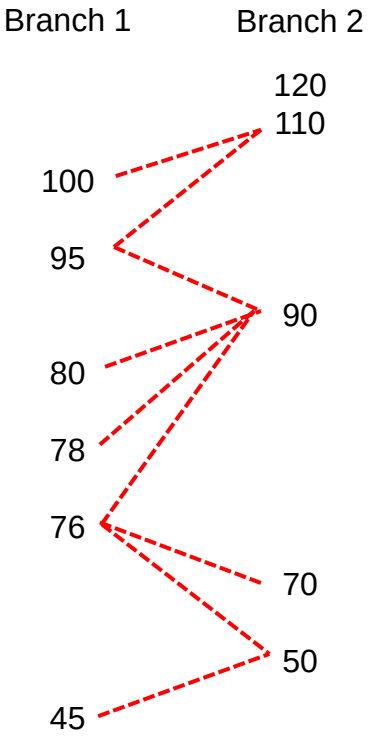
Interconnect Optimization

- # candidates at a branch point
  - a candidates from branch 1
  - b candidates from branch 2



Interconnect Optimization

- # candidates at a branch point (max. a + b – 1)
  - RAT

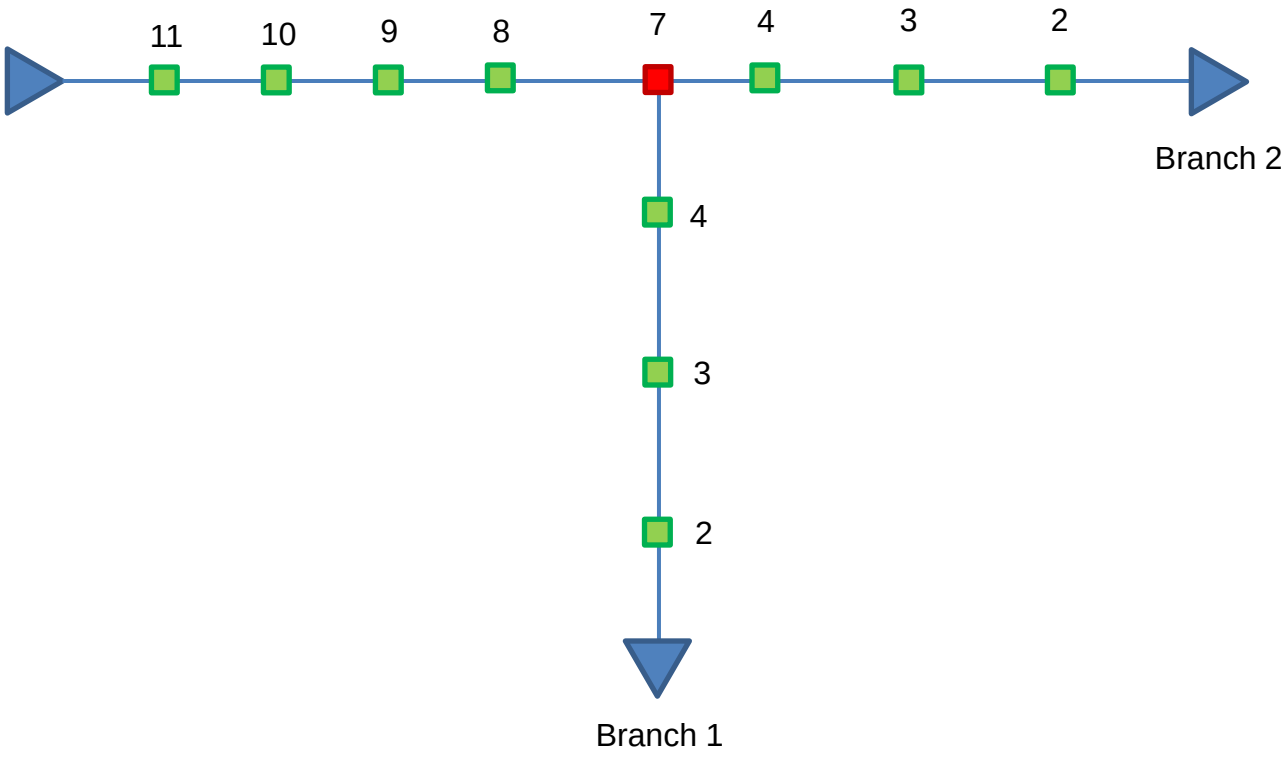


candidates from Branch 1: 6
candidates from Branch 2: 5

candidates survived: 9

Interconnect Optimization

- # candidates: O(n) where n is # bufferable locations



Interconnect Optimization

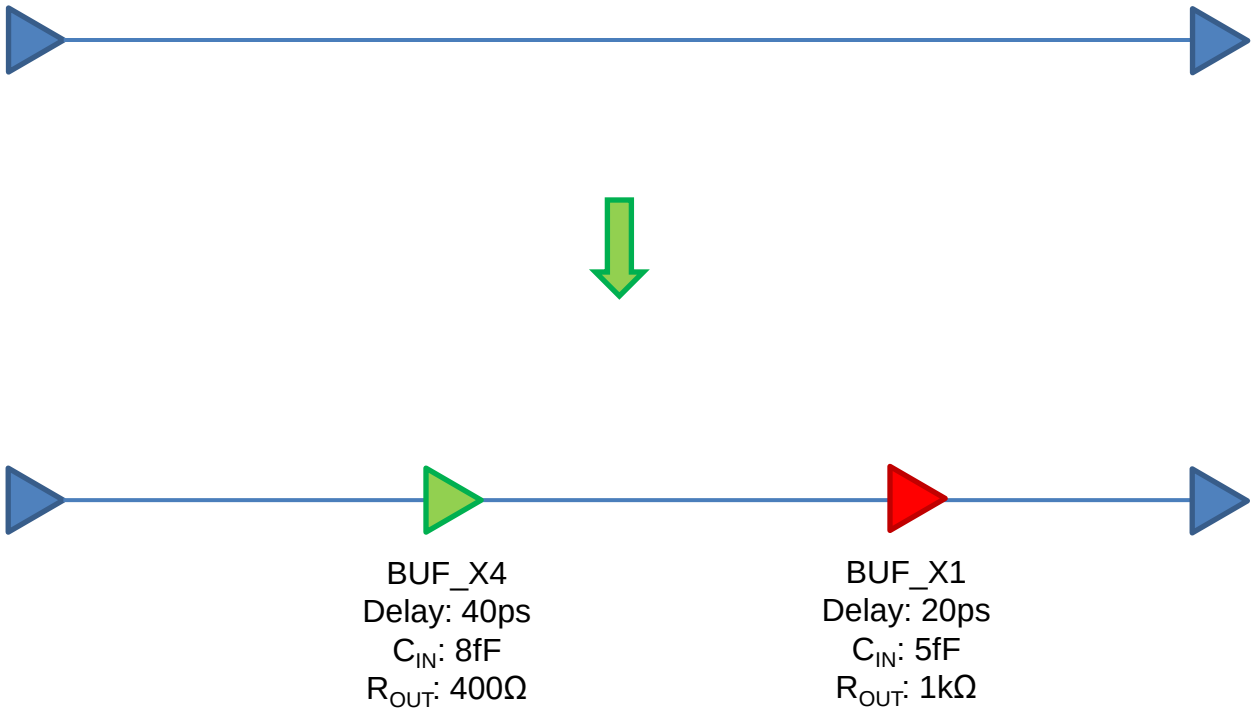
- This page is left blank intentionally.

Interconnect Optimization

- Problem definition for buffer insertion with different buffer types
  - Given
    - An RC tree or a routing topology
    - Characteristics (R_{OUT}) of the driver node
    - Characteristics (C_{LOAD}) of the sink nodes
    - Characteristics (R_{OUT} and C_{IN}) of each buffer type
    - Characteristics (r_{wire} and c_{wire}) of the wires
    - Timing constraints (RAT at each sink node)
    - Bufferable locations
  - Maximize
    - Slack at the driver node

Interconnect Optimization

- Example

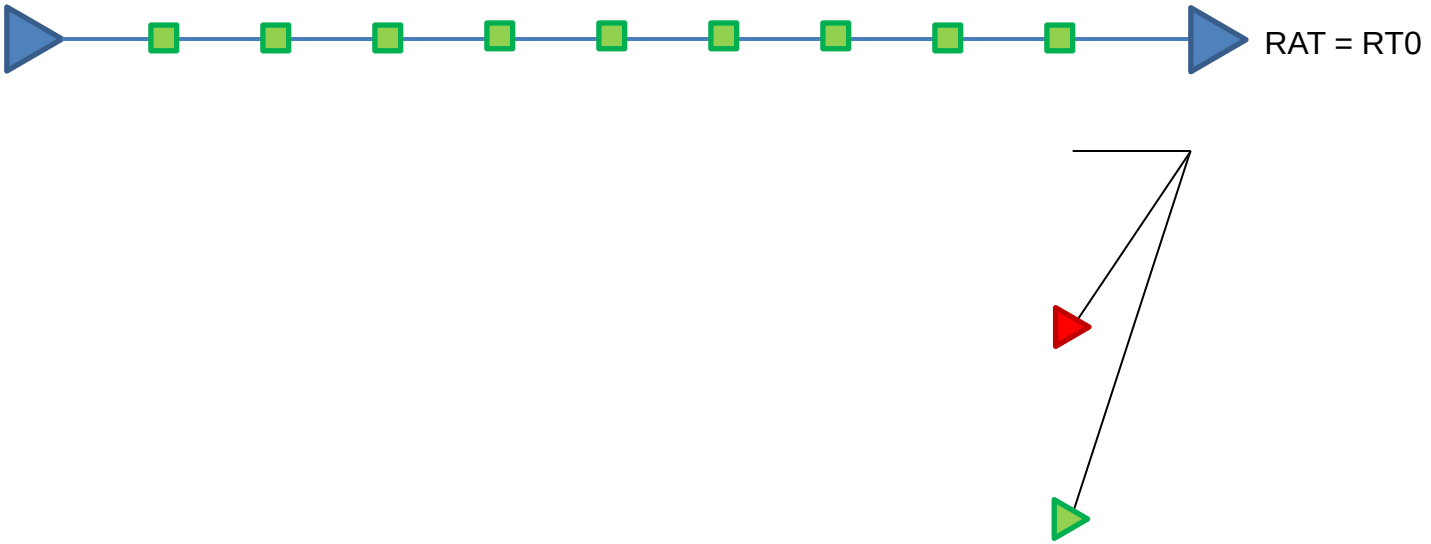


Interconnect Optimization

- Handling multiple buffer types can be done in the same way.
- The complexity goes up.

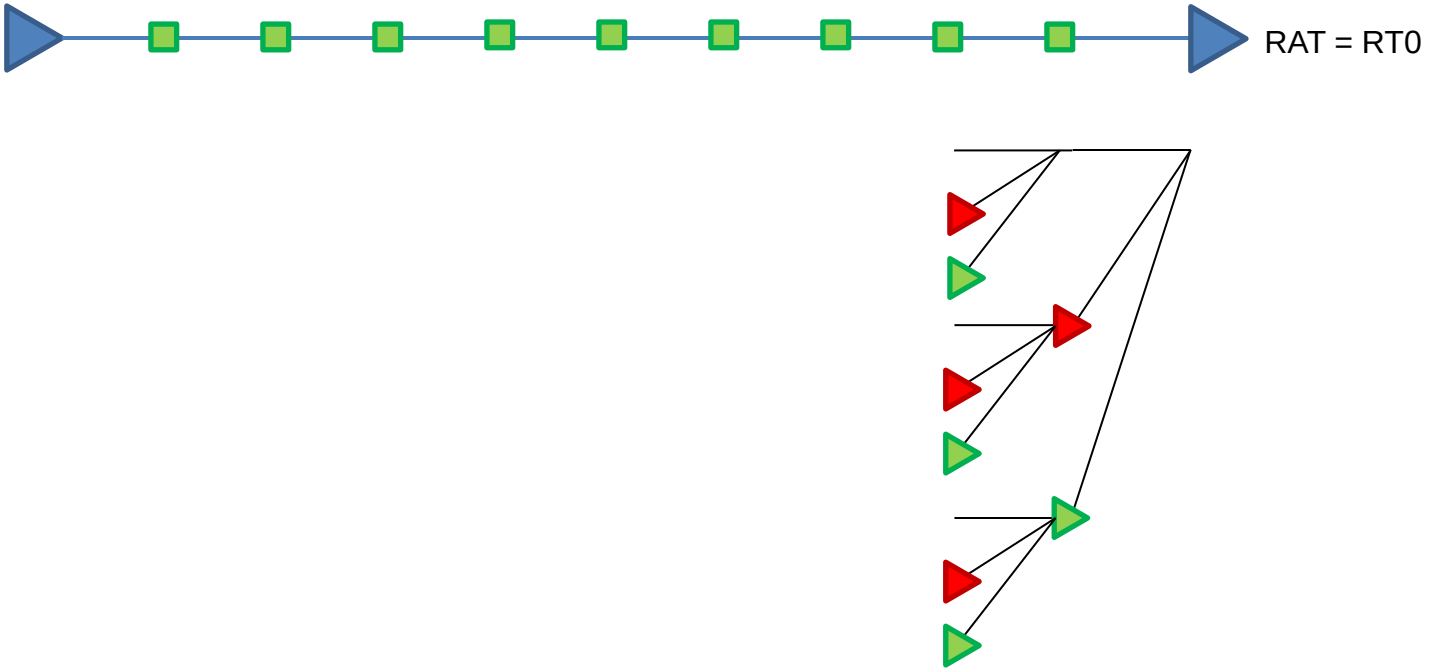
Interconnect Optimization

- Dynamic programming-based buffer insertion
  - van Ginneken algorithm



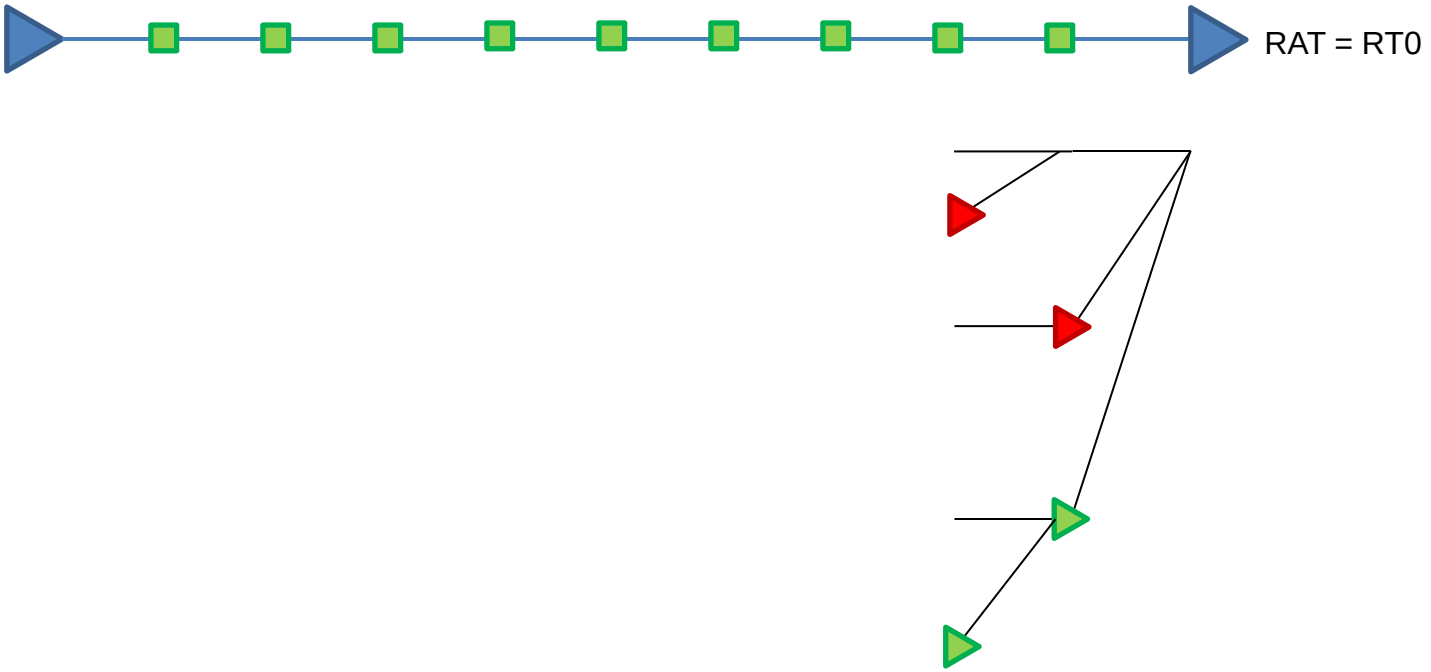
Interconnect Optimization

- Dynamic programming-based buffer insertion
  - van Ginneken algorithm



Interconnect Optimization

- Dynamic programming-based buffer insertion
  - van Ginneken algorithm



Interconnect Optimization

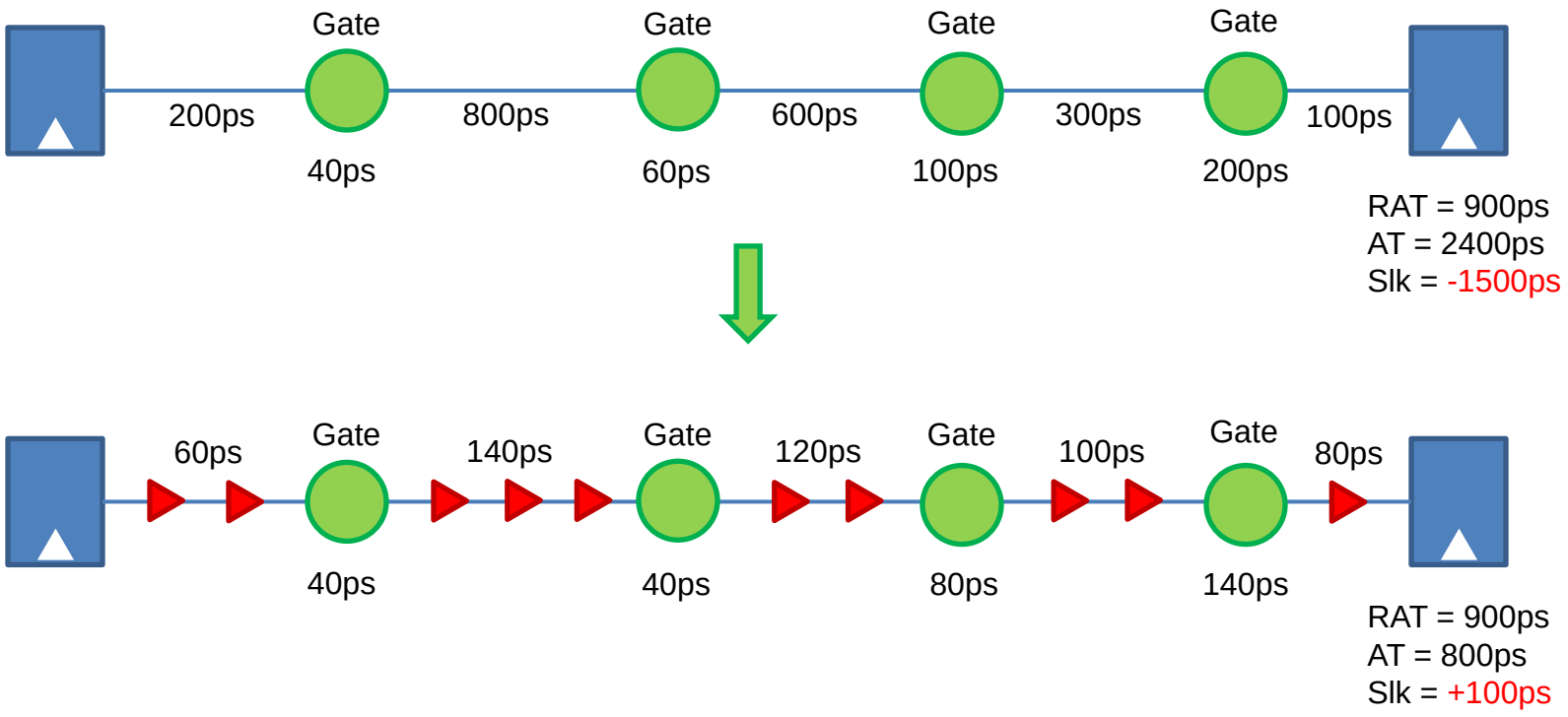
- # candidates (B is # buffer types)
  - 1st position
    - B+1 (un-buffered, b_1, …, b_B)
  - 2nd position
    - (B+1)*(B+1) – B*B = 2B+1
  - 3rd position
    - (2B+1)*(B+1) – (2B)*B = 3B+1
  - 4th position
    - (3B+1)*(B+1) – (3B)*B = 4B+1
  - …
- # candidates: O(B)

Interconnect Optimization

- This page is left blank intentionally.

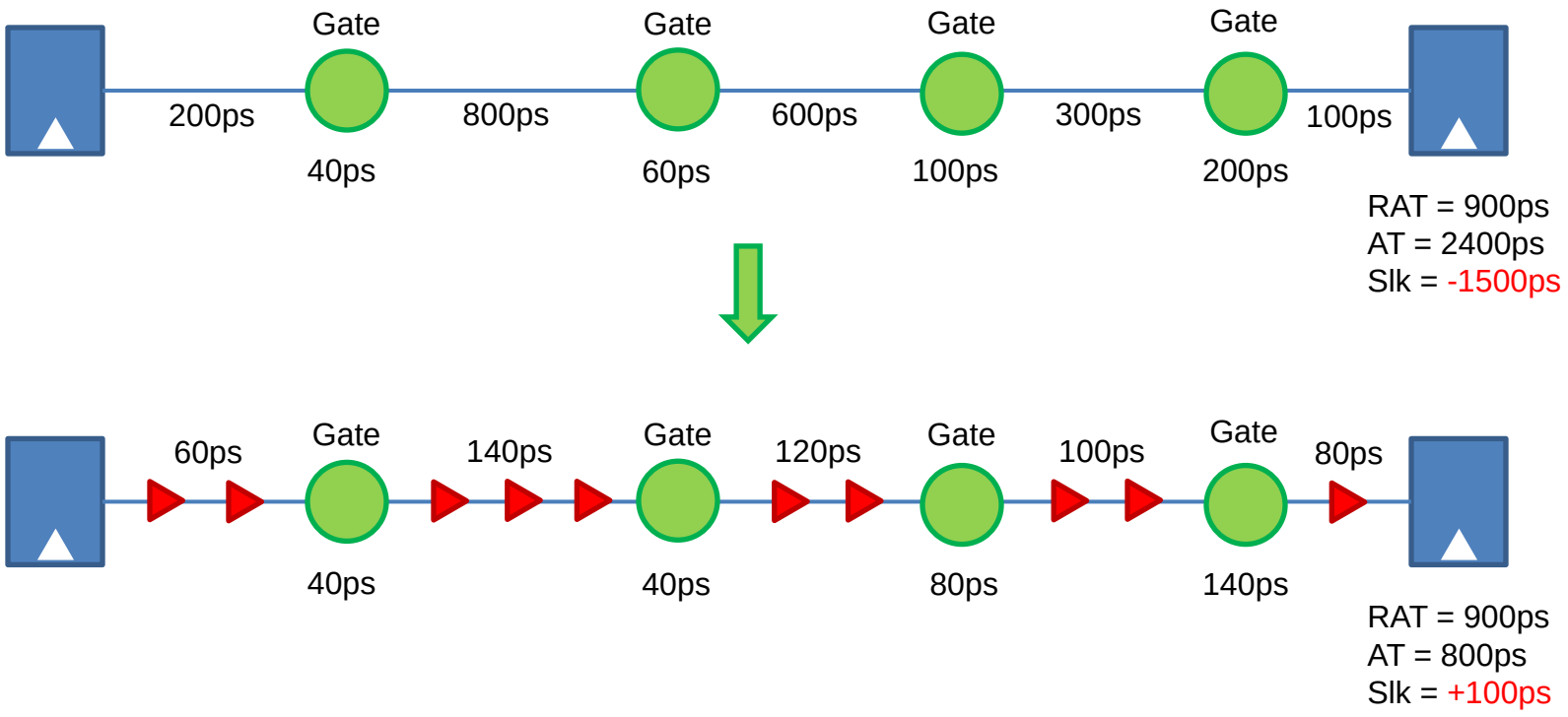
Interconnect Optimization

- More issues – Over-optimization
  - DP is for each net.



Interconnect Optimization

- More issues – Over-optimization
  - DP is for each net.

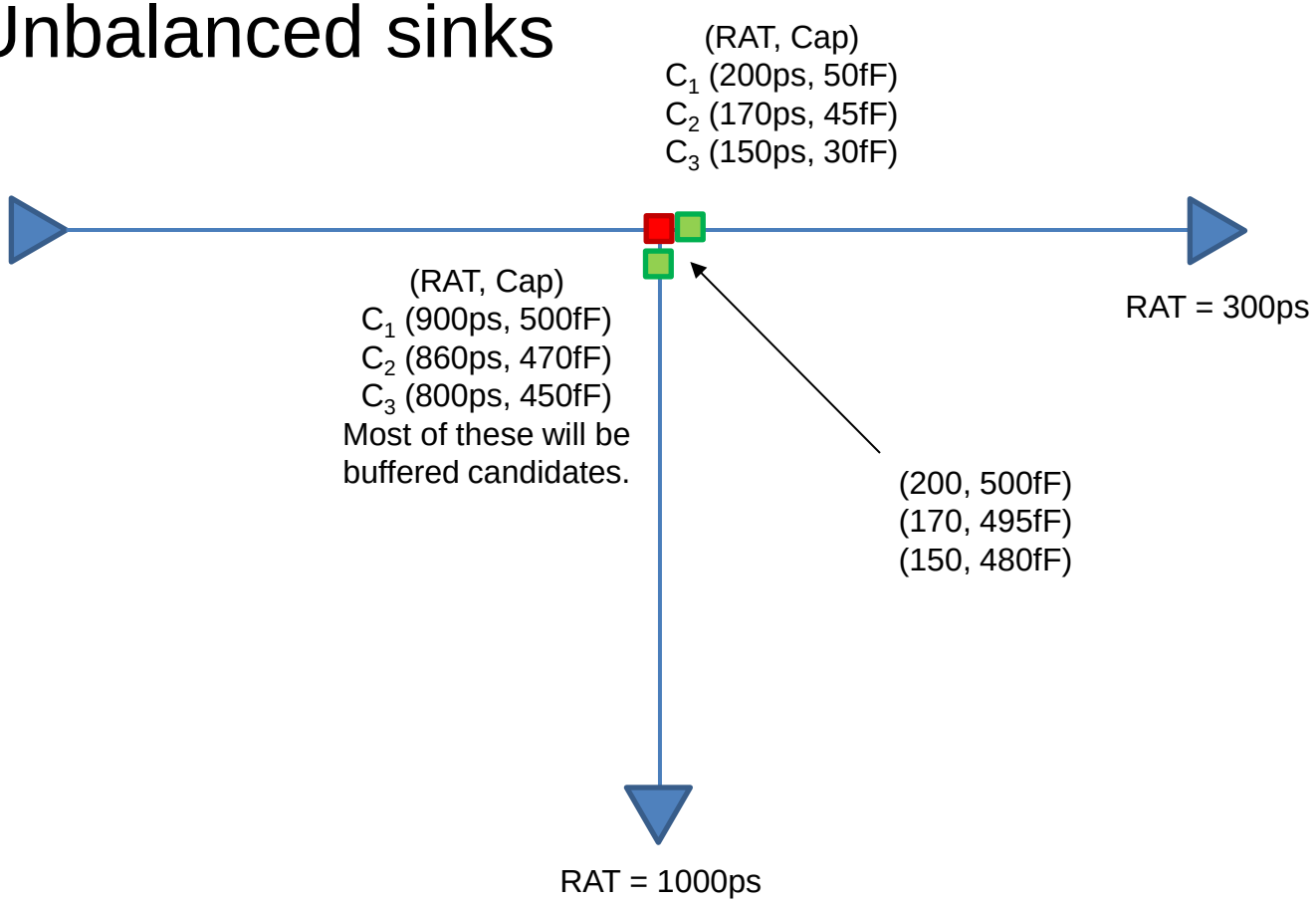


Interconnect Optimization

- More issues – Over-optimization
  - DP maximizes the RAT at the source node.
  - i.e., DP minimizes the delay from the source to the worst sink.
  - Problems when the timing conditions of the sinks are unbalanced.

Interconnect Optimization

- Unbalanced sinks



Interconnect Optimization

- Unbalanced sinks