EE 201A
(Starting 2005, called EE 201B)

Modeling and Optimization for VLSI Layout

Instructor: Lei He

Email: LHE@ee.ucla.edu

Chapter 7 Interconnect Delay

7.1 Elmore Delay
7.2 High-order model and moment matching
7.3 Stage delay calculation

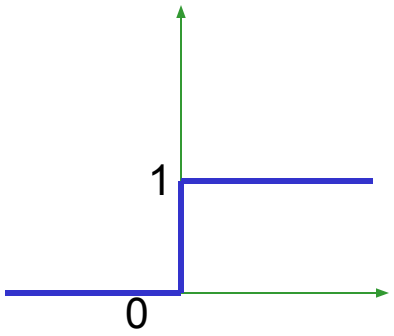
Basic Circuit Analysis Techniques

- Output response
  - Network structures & state → Natural response $v_N(t)$ (zero-input response)
  - Input waveform & zero-states → Forced response $v_F(t)$ (zero-state response)

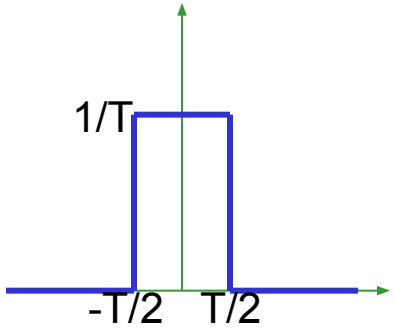
For linear circuits: $v(t) = v_N(t) + v_F(t)$

- Basic waveforms
  - Step input
  - Pulse input
  - Impulse Input
- Use simple input waveforms to understand the impact of network design

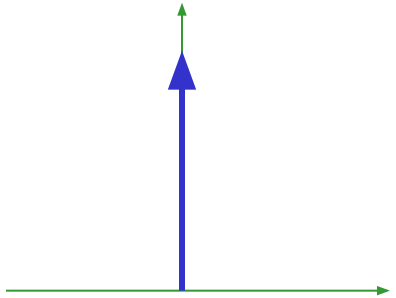
Basic Input Waveforms



unit step function



pulse function of width *T*



unit impulse function

$$u(t)=\begin{cases} 0 & t<0 \\ 1 & t\geq 0 \end{cases}$$

$$P_T(t)=\frac{1}{T}\left[u(t+\frac{T}{2})-u(t-\frac{T}{2})\right]$$

$$\delta(t):P_T(t) \quad \text{when } T\to 0$$

$$\delta(t)=0 \quad \text{for } t\neq 0$$

$$\text{singular} \quad \text{for } t=0$$

s.t. for any $\zeta>0$

$$\int_{-\zeta}^{\zeta}\delta(t)dt=1$$

By definition

$$u(t)=\int_{-\infty}^{t}\delta(x)dx$$

$$\text{or } \delta(t)=\frac{du(t)}{dt}$$

Step Response vs. Impulse Response

- Definitions:
  - (unit) step input u(t) ⟶ (unit) step response g(t)
  - (unit) impulse input δ(t) ⟶ (unit) impulse response h(t)

    (Input Waveform) (Output Waveform)
- Relationship

$$\delta(t) = \frac{du(t)}{dt} \quad \rightarrow \quad h(t) = \frac{dg(t)}{dt}$$

$$u(t) = \int_{-\infty}^{t} \delta(x)dx \quad \rightarrow \quad g(t) = \int_{0}^{t} h(x)dx$$

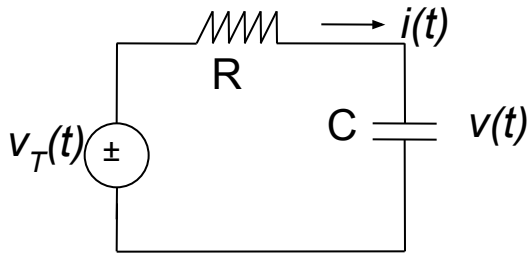
- Elmore delay

$$T_D = \int_0^\infty g'(t)t \cdot dt = \int_0^\infty h(t)t \cdot dt$$

Analysis of Simple RC Circuit

$$R \cdot i(t) + v(t) = v_T(t)$$

$$i(t) = \frac{d(Cv(t))}{dt} = C\frac{dv(t)}{dt}$$



$$\Rightarrow RC\frac{dv(t)}{dt} + v(t) = v_T(t)$$

first-order linear differential equation with constant coefficients

state variable

Input waveform

Analysis of Simple RC Circuit

zero-input response: $$RC\frac{dv(t)}{dt}+v(t)=0$$

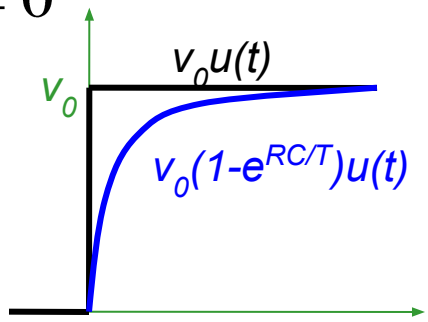
(natural response) $$\frac{1}{v(t)}\frac{dv(t)}{dt}=-\frac{1}{RC}\Rightarrow v_N(t)=Ke^{-t/RC}$$

step-input response: $$RC\frac{dv(t)}{dt}+v(t)=v_0u(t)$$

$$v_F(t)=v_0u(t)\Rightarrow v(t)=Ke^{-t/RC}+v_0u(t)$$

match initial state: $$v(0)=0 \quad\Rightarrow\quad K+v_0u(t)=0$$

output response for step-input: $$v(t)=v_0(1-e^{-t/RC})u(t)$$

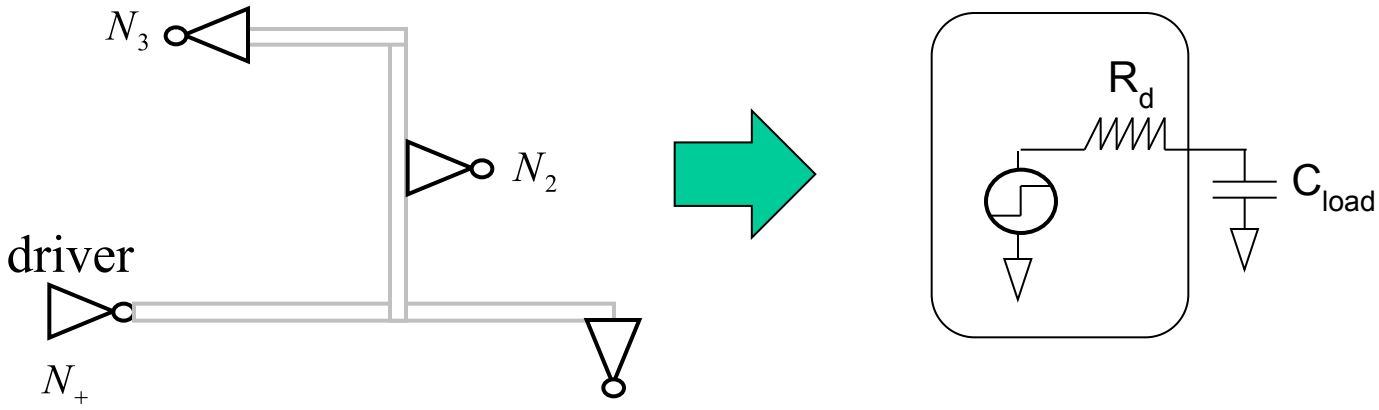


Delays of Simple RC Circuit

- $v(t) = v_0(1 - e^{-t/RC})$ under step input $v_0u(t)$
- $v(t)=0.9v_0 \Rightarrow t = 2.3RC$ $v(t)=0.5v_0 \Rightarrow t = 0.7RC$
- Commonly used metric $T_D = RC$
  (Elmore delay to be defined later)

Lumped Capacitance Delay Model

- R = driver resistance
- C = total interconnect capacitance + loading capacitance
- Sink Delay: $t_d = R \cdot C$

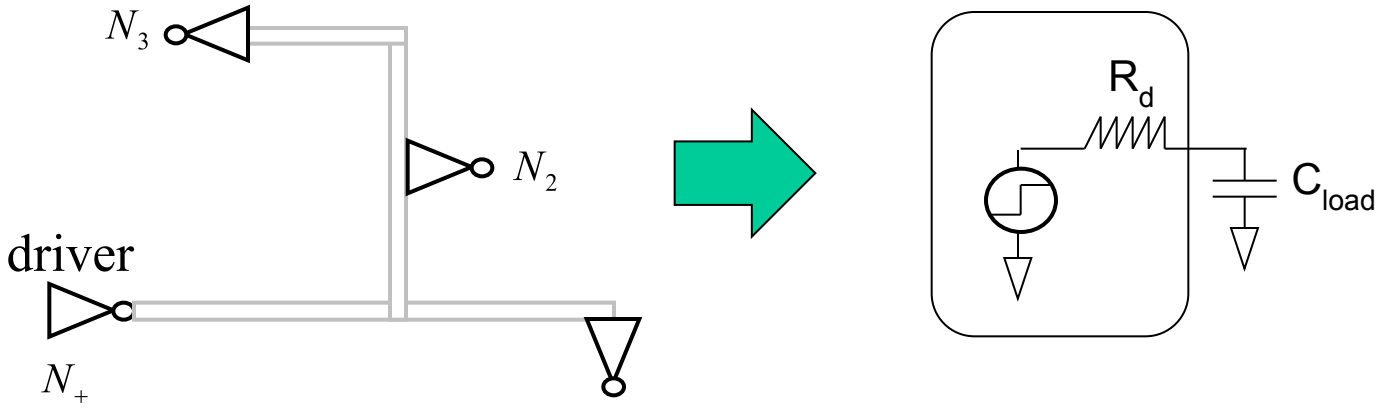


- 50% delay under step input = 0.7RC
- Valid when driver resistance >> interconnect resistance
- All sinks have equal delay

Lumped RC Delay Model

$$t_D = R_d \cdot C_{load} = R_d \cdot (C_{int} + C_g)$$
$$= R_d \cdot (C_0 \cdot L + C_g)$$

- Minimize delay ⇔ minimize wire length



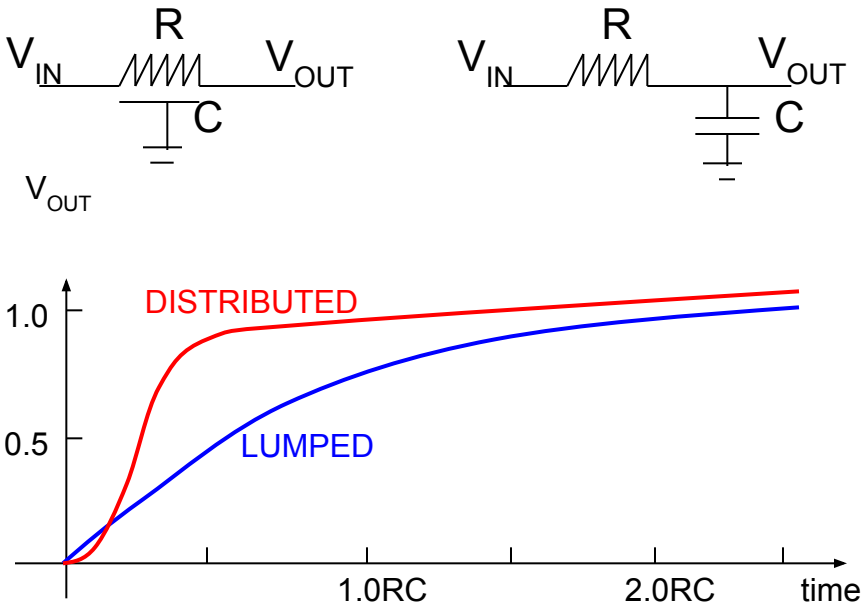
Delay of Distributed RC Lines

$$V_{out}(t) \xrightarrow[\text{Transform}]{\text{Laplace}} V_{out}(s)$$

$$V_{out}(s) = \frac{1}{s\cosh\sqrt{sRC}}$$

$$\cosh(x) = \frac{e^{x} + e^{-x}}{2}$$

$$= 1 + \frac{x^{2}}{2!} + \frac{x^{4}}{4!} + \ldots\ldots$$



Step response of distributed and lumped RC networks. A potential step is applied at V_{IN}, and the resulting V_{OUT} is plotted. The time delays between commonly used reference points in the output potential is also tabulated.

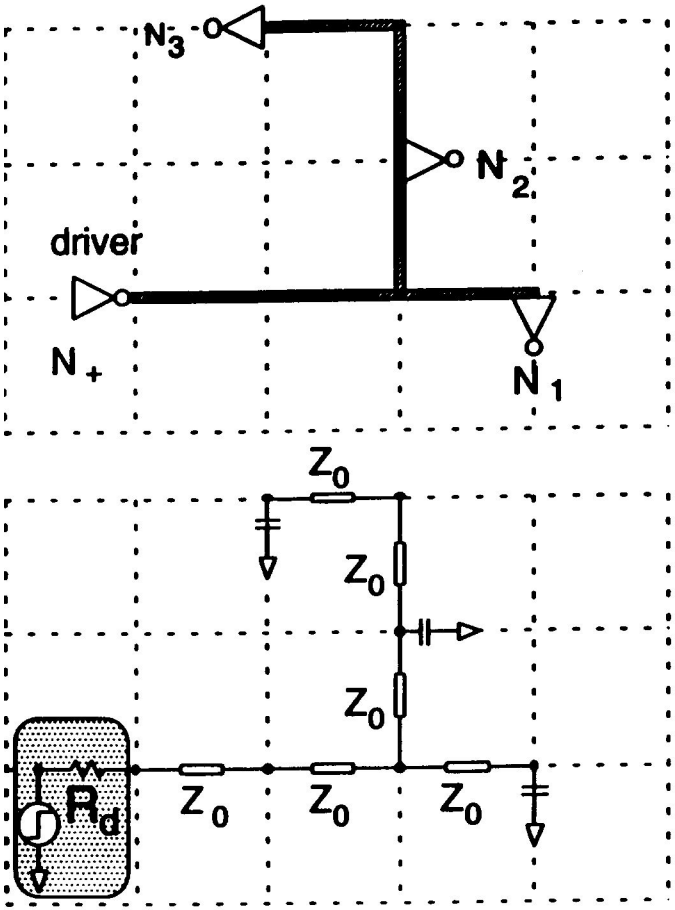
Delay of Distributed RC Lines (cont'd)

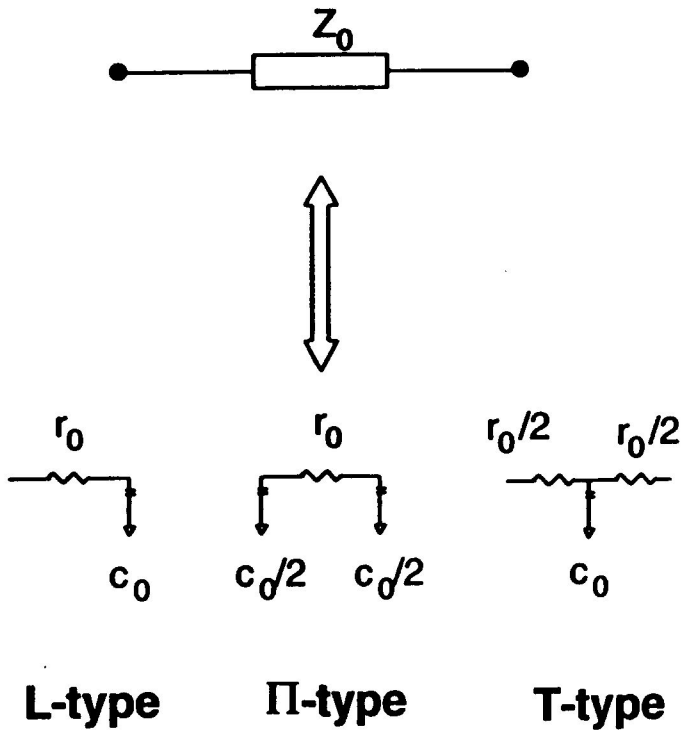
Output potential range	Time elapsed (Distributed RC Network)	Time elapsed (Lumped RC Network)
0 to 90%	1.0 *RC*	2.3 *RC*
10% to 90% (rise time)	0.9 *RC*	2.2 *RC*
0 to 63%	0.5 *RC*	1.0 *RC*
0 to 50% (delay)	0.4 *RC*	0.7 *RC*
0 to 10%	0.1 *RC*	0.1 *RC*

Distributed Interconnect Models

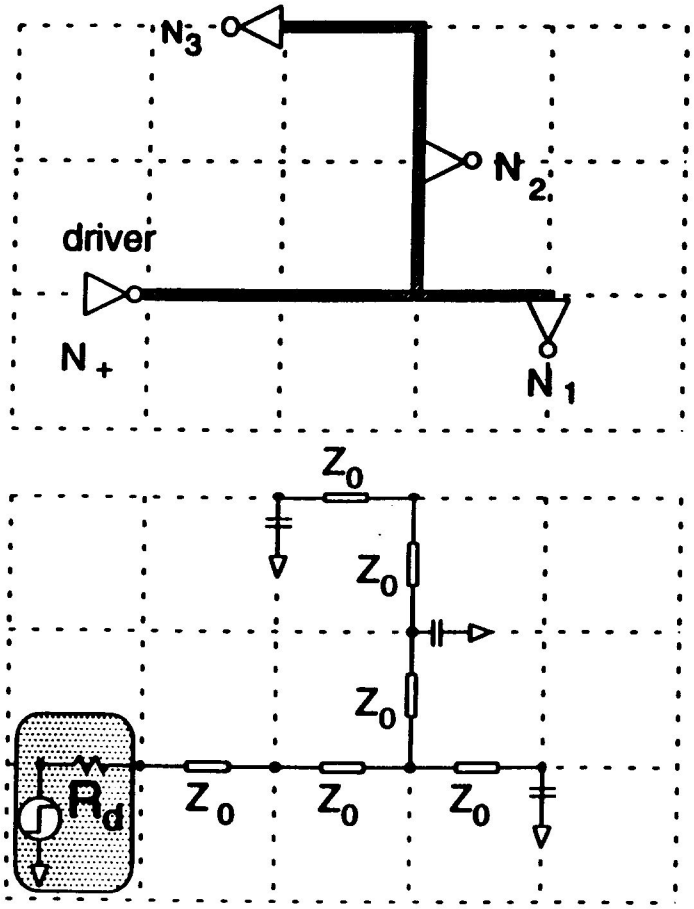
- Distributed RC circuit model
  - L,T or Π circuits
- Distributed RCL circuit model
- Tree of transmission lines

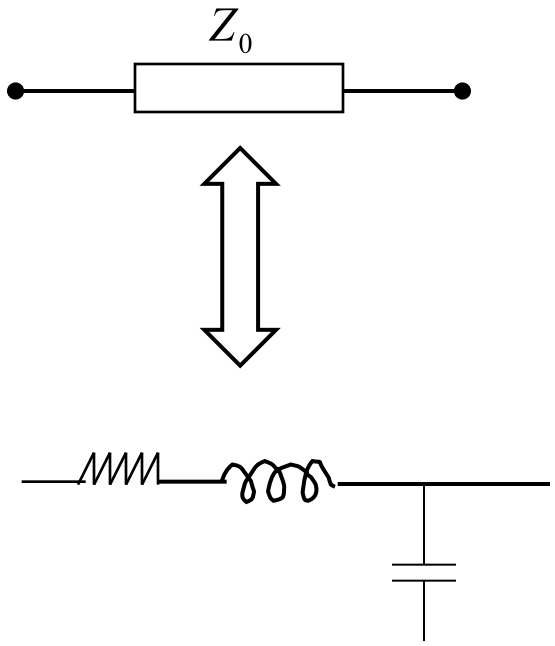
Distributed RC Circuit Models





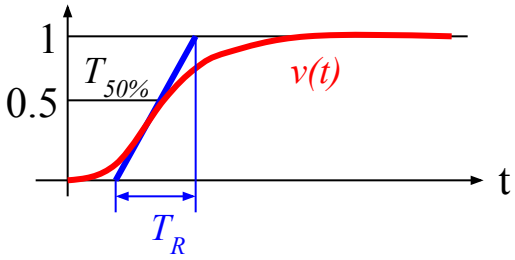
Distributed RLC Circuit Model (without mutual inductance)





Delays of Complex Circuits under Unit Step Input

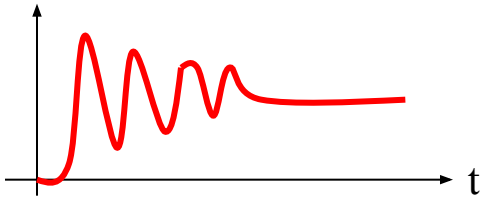
- Circuits with monotonic response



- Easy to define delay & rise/fall time
- Commonly used definitions
  - Delay $T_{50\%}$ = time to reach half-value, $v(T_{50\%}) = 0.5V_{dd}$
  - Rise/fall time $T_R = 1/v'(T_{50\%})$ where $v'(t)$: rate of change of $v(t)$ w.r.t. t
  - Or rise time = time from 10% to 90% of final value
- Problem: lack of ***general*** analytical formula for $T_{50\%}$ & T_R!

Delays of Complex Circuits under Unit Step Input (cont'd)

- Circuits with non-monotonic response



- Much more difficult to define delay & rise/fall time

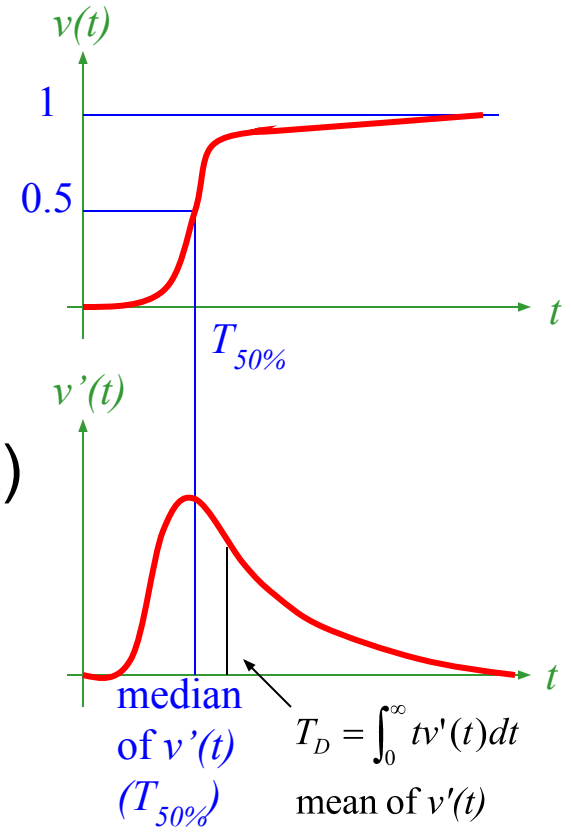
Elmore Delay for Monotonic Responses

- Assumptions:
  - Unit step input
  - Monotone output response

- Basic idea: use of mean of v'(t) to approximate median of v'(t)

$v(t)$: output response (monotone)

$v'(t)$: rate of change of $v(t)$



Elmore Delay for Monotonic Responses

- $T_{50\%}$: median of $v'(t)$, since

$$\int_0^{T_{50\%}} v'(t)dt = \int_{T_{50\%}}^{+\infty} v'(t)dt$$
$$= \text{half of final value of } v(t) \quad \text{(by def.)}$$

- Elmore delay T_D = mean of $v'(t)$

$$T_D = \int_0^{\infty} v'(t)t\,dt$$

Why Elmore Delay?

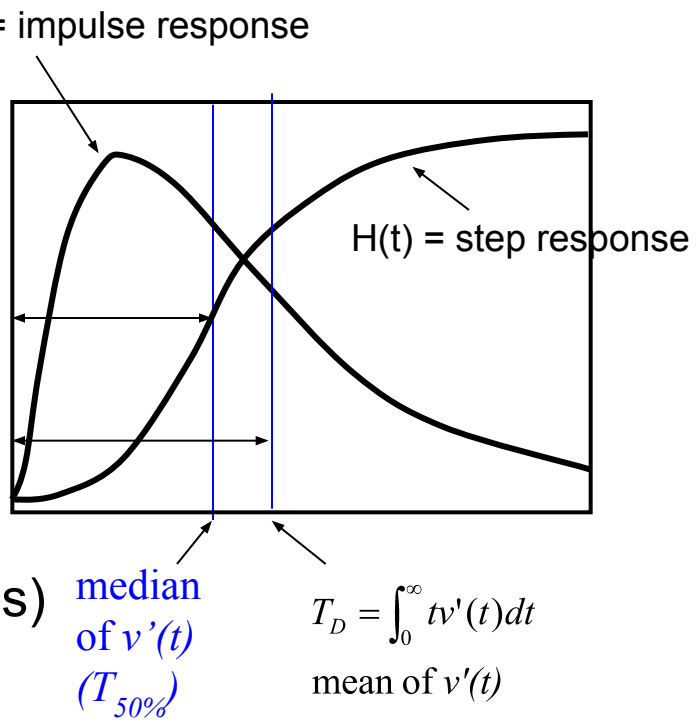
- Elmore delay is easier to compute analytically in most cases
  - Elmore's insight [Elmore, J. App. Phy 1948]
  - Verified later on by many other researchers, e.g.
    - Elmore delay for RC trees [Penfield-Rubinstein, DAC'81]
    - Elmore delay for RC networks with ramp input [Chan, T-CAS'86]
    - .....
- For RC trees: [Krauter-Tatuianu-Willis-Pileggi, DAC'95]
  $T_{50\%} \leq T_D$

- Note: Elmore delay is not 50% value delay in general!

Elmore Delay for RC Trees

- Definition
  - h(t) = impulse response
  - T_D = mean of h(t)

$$= \int_0^\infty h(t) \cdot t \, dt$$



- Interpretation
  - H(t) = output response (step process)
  - h(t) = rate of change of H(t)
  - $T_{50\%}$ = median of h(t)
  - Elmore delay approximates the median of h(t) by the mean of h(t)

Elmore Delay of a RC Tree

[Rubinstein-Penfield-Horowitz, T-CAD'83]

Lemma: when a step input is applied to a RC tree

$v_i(t)$ is monotonic in t for every node i in tree

Proof: $\Leftrightarrow v'_i(t) \geq 0$ at every node i $(v'i(t) = h_i(t))$

$\Leftrightarrow$ impulse response $h_i(t) \geq 0$ at every node i

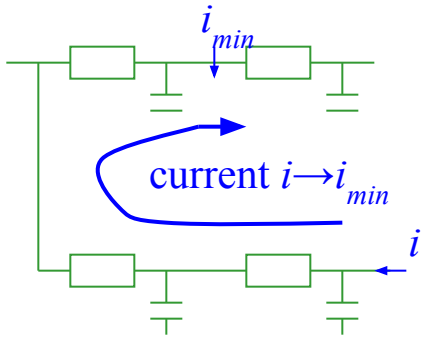
Let $h_{\min}(t)$ be the min. voltage of any node at t

Apply impulse func. at t=0:

$h_{\min}(0+) \geq 0$

Assume that $h_{\min}(t_0) < 0$

Then, $\exists t_1 < t_0$ s.t. $h'_{\min}(t_1) < 0$



Let node $i_{\min}$ achieve $h_{\min}(t_1)$ at t_1

Then, the current from any node i to $i_{\min}$ is ≥ 0 at t_1

Since $h_i(t_1) \geq h_{\min}(t_1)$ & i connects $i_{\min}$ via resistors

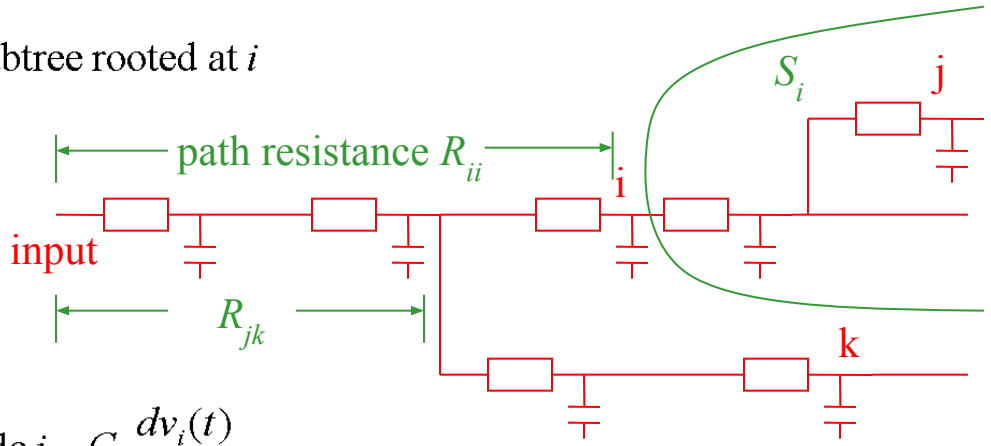
Since all currents $i \rightarrow i_{\min}$ charge the capacitor at $i_{\min}$

$h'_{\min}(t_1) \geq 0 \Rightarrow$ contradict ion!

Elmore Delay in a RC Tree (cont'd)

P_i : path from input to node i ; s_i subtree rooted at i

R_{jk} : resistance of common path $P_j \cap P_k$ from input to j & k



Theorem : Elmore delay to node i

$$T_{D_i} = \sum_k R_{ki} C_k$$

Proof : The current to cap. of node $i = C_i \dfrac{dv_i(t)}{dt}$

$$1 - v_i(t) = \text{The voltage drop on } P_i = \sum_{k \in P_i} R_k \cdot (\text{current to all cap's in } S_i)$$

$$= \sum_k (\text{current to cap } k) \cdot (\text{common path res. between } P_i \text{ and } P_k)$$

$$= \sum_k C_k \frac{dv_k(t)}{dt} \cdot R_{ki} = \sum_k R_{ki} C_k \frac{dv_k(t)}{dt}$$

$$T_{D_i} = \int_0^\infty v'_i(t) t \cdot dt = v_i(t) \cdot t \,|_0^\infty - \int_0^\infty v_i(t) dt$$

$$= \lim_{T \to \infty} [v_i(T) \cdot T - \int_0^T v_i(t) dt] = \lim_{T \to \infty} (v_i(T) - 1) \cdot T + \int_0^\infty (1 - v_i(t)) dt$$

Elmore Delay in a RC Tree (cont'd)

- We shall show later on that $\lim_{T\to\infty}(1-v_i(T))\cdot T = 0$ i.e. $1-v_i(T)$ goes to 0 at a much faster rate than $1/T$ when $T\to\infty$

- Let $f_i(t) = \int_0^t [1-v_i(x)]dx$

$$f_i(t) = \int_0^t \sum_k R_{ki}C_k \frac{dv_k(x)}{dx} dx$$

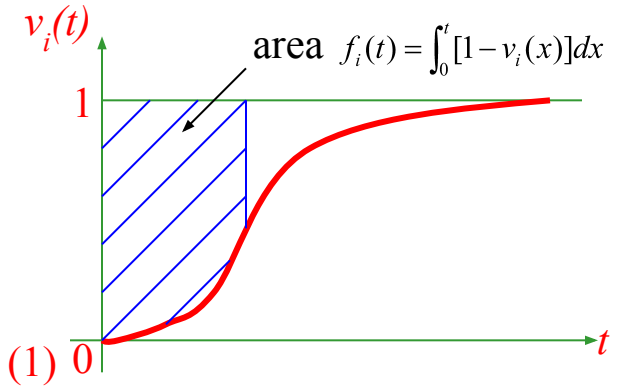
$$= \sum_k R_{ki}C_k v_k(t)$$

$$= \sum_k R_{ki}C_k - \sum_k R_{ki}C_k[1-v_k(t)]$$

$$f_i(\infty) = \sum_k R_{ki}C_k$$

$$\therefore\ T_{D_i} = \lim_{T\to\infty}(1-v_i(T))T + \int_0^\infty [1-v_i(t)]dt$$

$$= f_i(\infty) = \sum_k R_{ki}C_k$$



Some Definitions For Signal Bound Computation

Let $T_p = \sum_k R_{kk} C_k$

$T_{R_i} = \left(\sum_k R_{ki}^2 C_k\right) \Big/ R_{ii}$

Then, $T_{R_i} \le T_{D_i} \le T_p$

Recall $T_{D_i} = \sum_k R_{ki} C_k$

(since $R_{kk} \ge R_{ki}$ & $R_{ii} \ge R_{ki}$)

Signal Bounds in RC Trees

- Theorem

Lower bounds

$$v_i(t) \geq \begin{cases} 0 & t \geq 0 \\ 1 - \dfrac{T_{D_i}}{t + T_{Ri}} & t \geq 0 \quad (\text{non-trivial when } t \geq T_{D_i} - T_{R_i}) \\ 1 - \dfrac{T_{D_i}}{T_p} e^{(T_p - T_{Ri})/T_p} \cdot e^{-t/T_p} & t \geq T_p - T_{R_i} \end{cases}$$

Upper bounds

$$v_i(t) \leq \begin{cases} 1 - \dfrac{T_{D_i} - t}{T_p} & t \geq 0 \\ 1 - \dfrac{T_{R_i}}{T_p} e^{(T_{D_i} - T_{R_i})/T_{R_i}} \cdot e^{-t/T_{R_i}} & t \geq T_{D_i} - T_{R_i} \end{cases}$$

Delay Bounds in RC Trees

Lower bounds :

$$t \geq T_{\mathrm{D_i}} - T_p\left[1 - v_i(t)\right]$$

$$t \geq T_{D_i} - T_{R_i} + T_{R_i} \ln \frac{T_{R_i}}{T_p\left[1 - v_i(t)\right]} \qquad \text{when } v_i(t) \geq 1 - \frac{T_{R_i}}{T_p}$$

Upper bounds :

$$t \leq \frac{T_{D_i}}{1 - v_i(t)} - T_{R_i}$$

$$t \leq T_p - T_{R_i} + T_p \ln \frac{T_{Di}}{T_p\left[1 - v_i(t)\right]} \qquad \text{when } v_i(t) \geq 1 - \frac{T_{D_i}}{T_p}$$

Computation of Elmore Delay & Delay Bounds in RC Trees

- Let $C(T_k)$ be total capacitance of subtree rooted at *k*
- *Elmore delay*

$$T_{D_i} = \sum_{k \in p_i} R_k \cdot C(T_k)$$

upper bound :

$$T_p = \sum_{k} R_k \cdot C(T_k)$$

lower bound :

$$T_{R_i} = \sum_{k} R_{ki}^2 \cdot \frac{C_k}{R_{ii}}$$

* all three formula can be computed in linear time recursively in a bottom - up fashion

Comments on Elmore Delay Model

- Advantages
  - Simple closed-form expression
    - Useful for interconnect optimization
  - Upper bound of 50% delay [Gupta et al., DAC'95, TCAD'97]
    - Actual delay asymptotically approaches Elmore delay as input signal rise time increases
  - High fidelity [Boese et al., ICCD'93],[Cong-He, TODAES'96]
    - Good solutions under Elmore delay are good solutions under actual (SPICE) delay

Comments on Elmore Delay Model

- Disadvantages
  - Low accuracy, especially poor for slope computation
  - Inherently cannot handle inductance effect
    - Elmore delay is first moment of impulse response
    - Need higher order moments

Chapter 7.2
Higher-order Delay Model

Time Moments of Impulse Response *h*(*t*)

- Definition of moments

$$h(t) \xrightarrow{\text{L}} H(s)$$

$$H(s) = \int_0^\infty h(t)e^{-st}dt = \int_0^\infty h(t)\left(\sum_{i=0}^{\infty}\frac{1}{i!}(-st)^i\right)dt$$

$$= \sum_{i=0}^{\infty}\frac{1}{i!}(-s)^i\int_0^\infty h(t)\cdot t^i dt$$

i-th moment $$m_i = \frac{1}{i!}(-1)^i\int_0^\infty h(t)\cdot t^i dt$$

- Note that m_1 = Elmore delay when *h*(*t*) is monotone voltage response of impulse input

Pade Approximation

- H(s) can be modeled by Pade approximation of type (*p/q*):

$$\hat{H}_{p,q}(s) = \frac{b_p s^p + \cdots + b_1 s + b_0}{a_q s^q + \cdots + a_1 s + 1}$$

  - where q < p << N

- Or modeled by *q*-th Pade approximation (*q* << *N*):

$$\hat{H}_q(s) \equiv \hat{H}_{q-1,q}(s) = \sum_{j=1}^{q} \frac{k_j}{s - p_j}$$

- Formulate 2*q* constraints by matching 2*q* moments to compute k_i's & p_i's

General Moment Matching Technique

- Basic idea: match the moments $m_{-(2q-r)}, \ldots, m_{-1}, m_0, m_1, \ldots, m_{r-1}$

$$\hat{H}(s) = \frac{k_1}{s - p_1} + \frac{k_2}{s - p_2} + \cdots + \frac{k_q}{s - p_q}$$

$$= \hat{m}_0 + \hat{m}_1 s + \cdots + \hat{m}_{r-1} s^{r-1} + \mathrm{O}(s^r)$$

- When $r = 2q-1$:

(i) initial condition matches, i.e.

$$\hat{h}(0^+) = h(0^+), \text{ or } \lim_{s \to \infty} s\hat{H}(s) = \lim_{s \to \infty} sH(s)$$

$$\text{or } (-\hat{m}_{-1} = -m_{-1})$$

(ii) $\hat{m}_k = m_k$ for $k = 0, 1, \ldots, 2q-2$

$\hat{m}_k$: moments of $\hat{H}(s)$

Compute Residues & Poles

$$\frac{k_i}{s-p_i} = -\frac{k_i/p_i}{1-s/p_i} = -\frac{k_i}{p_i}\sum_{j=0}^{\infty}\left(\frac{s}{p_i}\right)^j$$

EQ1

$$k_1 + k_2 + \cdots + k_q = h(0) = -m_{-1} \quad \left(= \lim_{s\to\infty} s\hat{H}(s)\right)$$

initial condition

$$-\left(\frac{k_1}{p_1} + \frac{k_2}{p_2} + \cdots + \frac{k_q}{p_q}\right) = m_0$$

$$-\left(\frac{k_1}{p_1^2} + \frac{k_2}{p_2^2} + \cdots + \frac{k_q}{p_q^2}\right) = m_1$$

$$\vdots$$

$$-\left(\frac{k_1}{p_1^{2q-1}} + \frac{k_2}{p_2^{2q-1}} + \cdots + \frac{k_q}{p_q^{2q-1}}\right) = m_{2q-2}$$

match first 2q-1 moments

Basic Steps for Moment Matching

Step 1: Compute $2q$ moments $m_{-1}, m_0, m_1, \ldots, m_{(2q-2)}$ of $H(s)$

Step 2: Solve $2q$ non-linear equations of EQ1 to get

$$p_1, p_2, \ldots, p_q : \text{poles}$$

$$k_1, k_2, \ldots, k_q : \text{residues}$$

Step 3: Get approximate waveform

$$\hat{h}(t) = k_1 e^{p_1 t} + k_2 e^{p_2 t} + \ldots + k_q e^{p_q t}$$

Step 4: Increase q and repeat 1-4, if necessary, for better accuracy

Components of Moment Matching Model

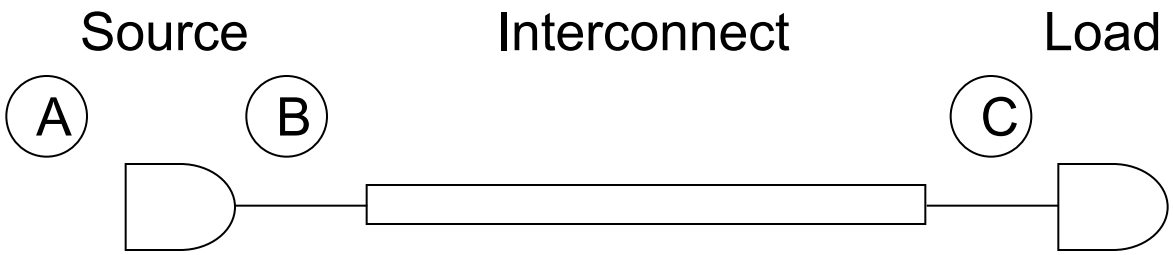
- Moment computation
  - Iterative DC analysis on transformed equivalent DC circuit
  - Recursive computation based on tree traversal
  - Incremental moment computation
- Moment matching methods
  - Asymptotic Waveform Evaluation (AWE) [Pillage-Rohrer, TCAD'90]
  - 2-pole method [Horowitz, 1984] [Gao-Zhou, ISCAS'93]...

- Moment calculation will be provided as an OPTIONAL reading

Chapter 7 Interconnect Delay

7.1 Elmore Delay
7.2 High-order model and moment matching
7.3 Stage delay calculation

Stage Delay



$$T_{AC} = T_{AB}(I, L) + T_{BC}(L)$$

$T_{AB}(0, L)$: Gate delay without interconnect

$T_{AB}(I, L)$: Gate delay with interconnect

$T_{BC}(L)$: Propagation delay + Transition delay

$$T_{Interconnect} = T_{AB}(I, L) - T_{AB}(0, L) + T_{BC}(L)$$

Modeling of Capacitive Load

- First-order approximation: the driver sees the total capacitance of wires and sinks
- Problem: Ignore shielding effect of resistance ⇒ pessimistic approximation as driving point admittance
- Transform interconnect circuit into a π-model [O'Brian-Savarino, ICCAD'89]
  - Problem: cannot be easily used with most device models
- Compute effective capacitance from π-model [Qian-Pullela-Pileggi, TCAD'94]

Π-Model
[O'Brian-Savarino, ICCAD'89]

- Moment matching again!
- Consider the first three moments of driving point admittance (moments of response current caused by an applied unit impulse)
- Current in the downstream of node k

$$I_k(s) = \sum_{j \in T_k} C_j s V_j(s)$$

$$Y_k(s) = I_k(s) / V_{in}(s)$$

$$Y_k(s) = \sum_{j \in T_k} C_j s H_j(s)$$

$$= \sum_{j \in T_k} C_j s \cdot (1 + m_j^{(1)} s + m_j^{(2)} s^2 + \square\ ) = \sum_{i=1}^{\infty} \sum_{j \in T_k} C_j m_j^{(i)} s^{i+1}$$

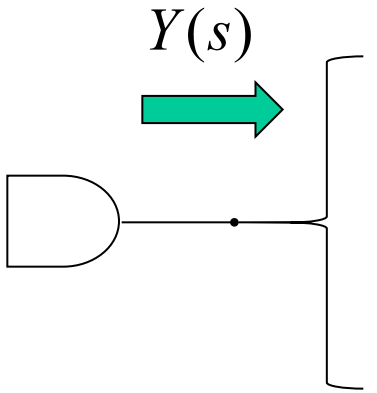
Driving-Point Admittance Approximations

- Driving-point admittance = Sum of voltage moment-weighted subtree capacitance

$$y_k^{(i)} = \sum_{j \in T_k} C_j m_j^{(i-1)}$$

$$Y_k(s) = \sum_{i=1}^{\infty} y_k^{(i)} s^i$$

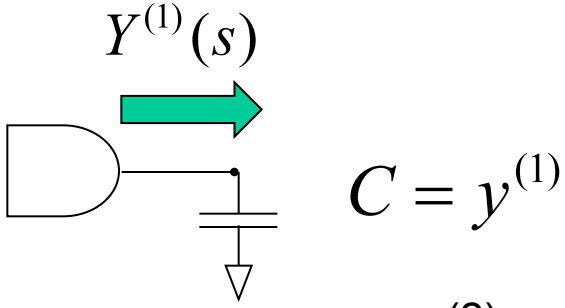
- Approximation of the driving point admittance at the driver



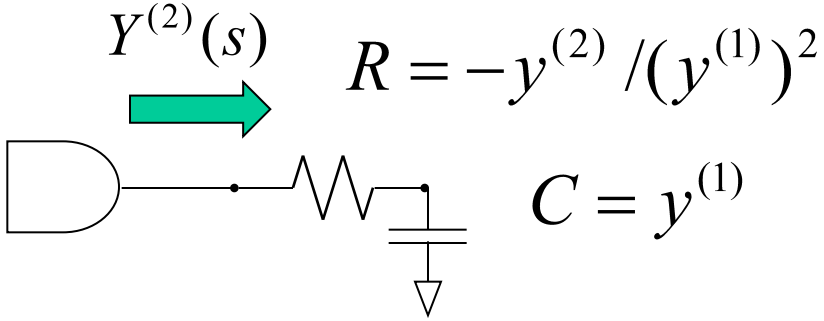
General RC Tree:
lumped RC elements,
distributed RC lines

Driving-Point Admittance Approximations

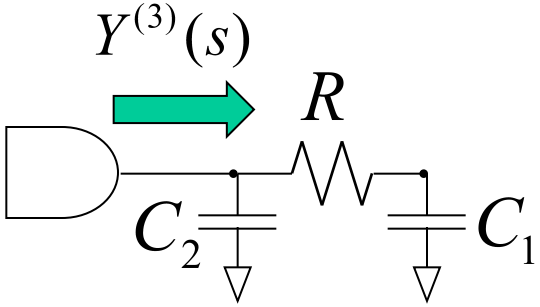
- First order approximation: $y^{(1)}$ = sum of subtree capacitance



- Second order approximation: $y_k^{(2)}$ = sum of subtree capacitance weighted by Elmore delay



Third Order Approximation: Π Model



$$y^{(1)} = C_1 + C_2$$

$$y^{(2)} = -RC_1^2$$

$$y^{(3)} = R^2 C_1^3$$

$$C_1 = \frac{(y^{(2)})^2}{y^{(3)}}$$

$$C_2 = y^{(1)} - C_1$$

$$R = \frac{y^{(2)}}{C_1^2}$$

Current Moment Computation

- Similar to the voltage moment computation
- Iterative tree traversal:
  - O(n) run-time, O(n) storage
- Bottom-up tree traversal:
  - O(n) run-time
  - Can achieve O(k) storage if we impose order of traversal, k = max degree of a node
  - O'Brian and Savarino used bottom-up tree traversal

Bottom-Up Moment Computation

- Maintain transfer function $H_{v\sim w}(s)$ for sink w in subtree T_v, and moment-weighted capacitance of subtree:

$$m_w^j \text{ for } j = 0 \ldots p, \quad C_{T_v}^j \text{ for } j = 0 \ldots p-1$$

- As we merge subtrees, compute new transfer function $\overline{H}_{u\sim v}(s)$ and weighted capacitance recursively:

$$\overline{C}_{T_v}^{j-1} = \overline{m}_v^{j-1} C_v + \sum_{q=0}^{j-1} \overline{m}_v^{j-1-q} C_{T_v}^q,$$

$$\overline{m}_v^j = -(R_v \overline{C}_{T_v}^{j-1} + L_v \overline{C}_{T_v}^{j-2}) \text{ for } j = 1 \ldots p$$

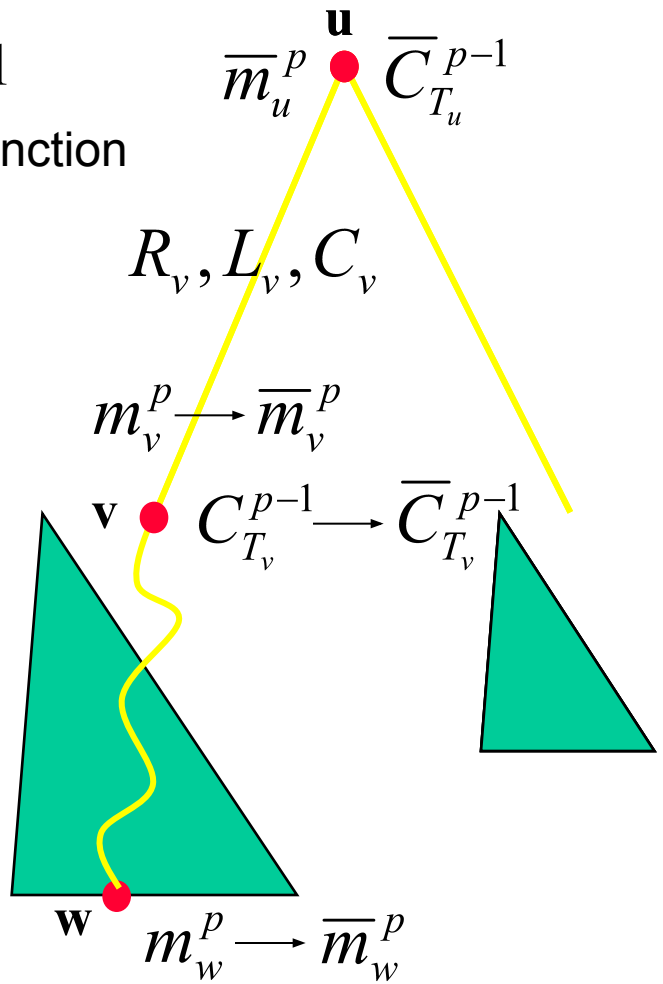
- New transfer function for node w

$$\overline{H}_{u\sim w}(s) = \overline{H}_{u\sim v}(s) \times H_{v\sim w}(s)$$

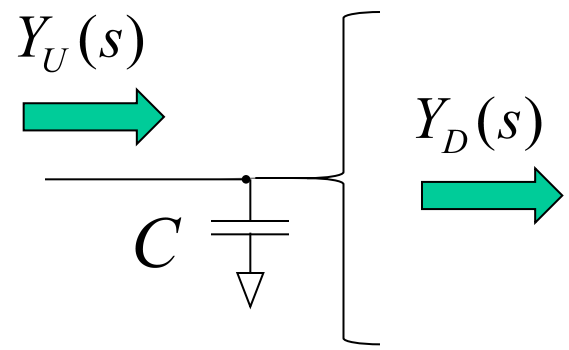
$$\overline{m}_w^j = \sum_{q=0}^{j} \overline{m}_v^{j-q} m_w^q \text{ for } j = 0 \ldots p$$

- New moment-weighted capacitance of T_u:

$$\overline{C}_{T_u}^j = \sum_{v \in child(u)} \overline{C}_{T_v}^j \text{ for } j = 0 \ldots p-1$$



Current Moment Computation Rule #1

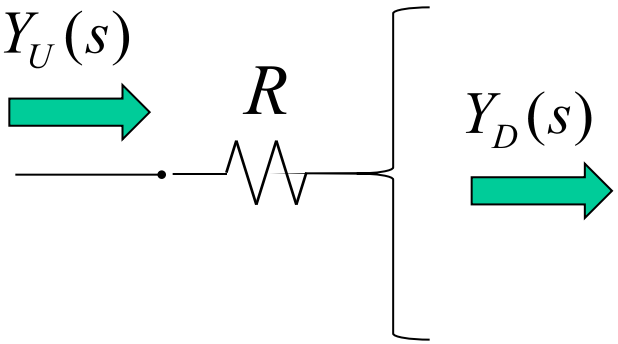


$$y_U^{(1)} = y_D^{(1)} + C$$

$$y_U^{(2)} = y_D^{(2)}$$

$$y_U^{(3)} = y_D^{(3)}$$

Current Moment Computation Rule #2

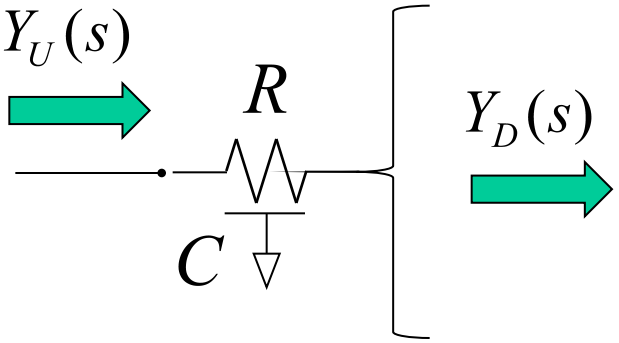


$$y_U^{(1)} = y_D^{(1)}$$

$$y_U^{(2)} = y_D^{(2)} - R(y_D^{(1)})^2$$

$$y_U^{(3)} = y_D^{(3)} - 2R(y_D^{(1)})(y_D^{(2)}) + R^2(y_D^{(1)})^3$$

Current Moment Computation Rule #3



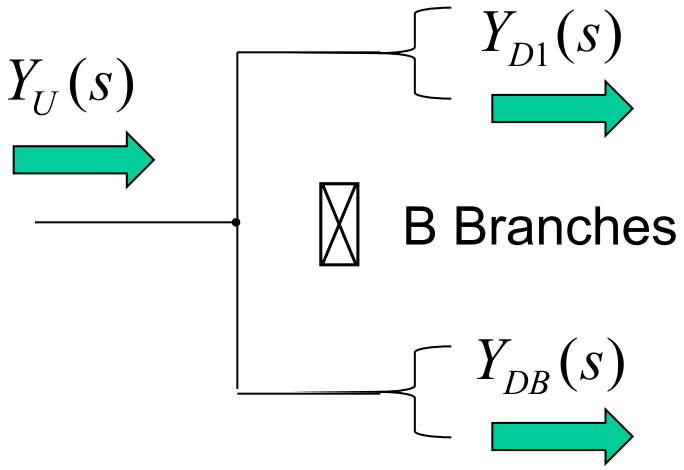
$$y_U^{(1)} = y_D^{(1)} + C$$

$$y_U^{(2)} = y_D^{(2)} - R\left[(y_D^{(1)})^2 + C(y_D^{(1)}) + \frac{1}{3}C^2 \right]$$

$$y_U^{(3)} = y_D^{(3)} - R\left[2(y_D^{(1)})(y_D^{(2)}) + C(y_D^{(2)})\right] +$$

$$R^2\left[(y_D^{(1)})^3 + \frac{4}{3}C(y_D^{(1)})^2 + \frac{2}{3}C^2(y_D^{(1)}) + \frac{2}{15}C^3 \right]$$

Current Moment Computation Rule #4 (Merging of Sub-trees)

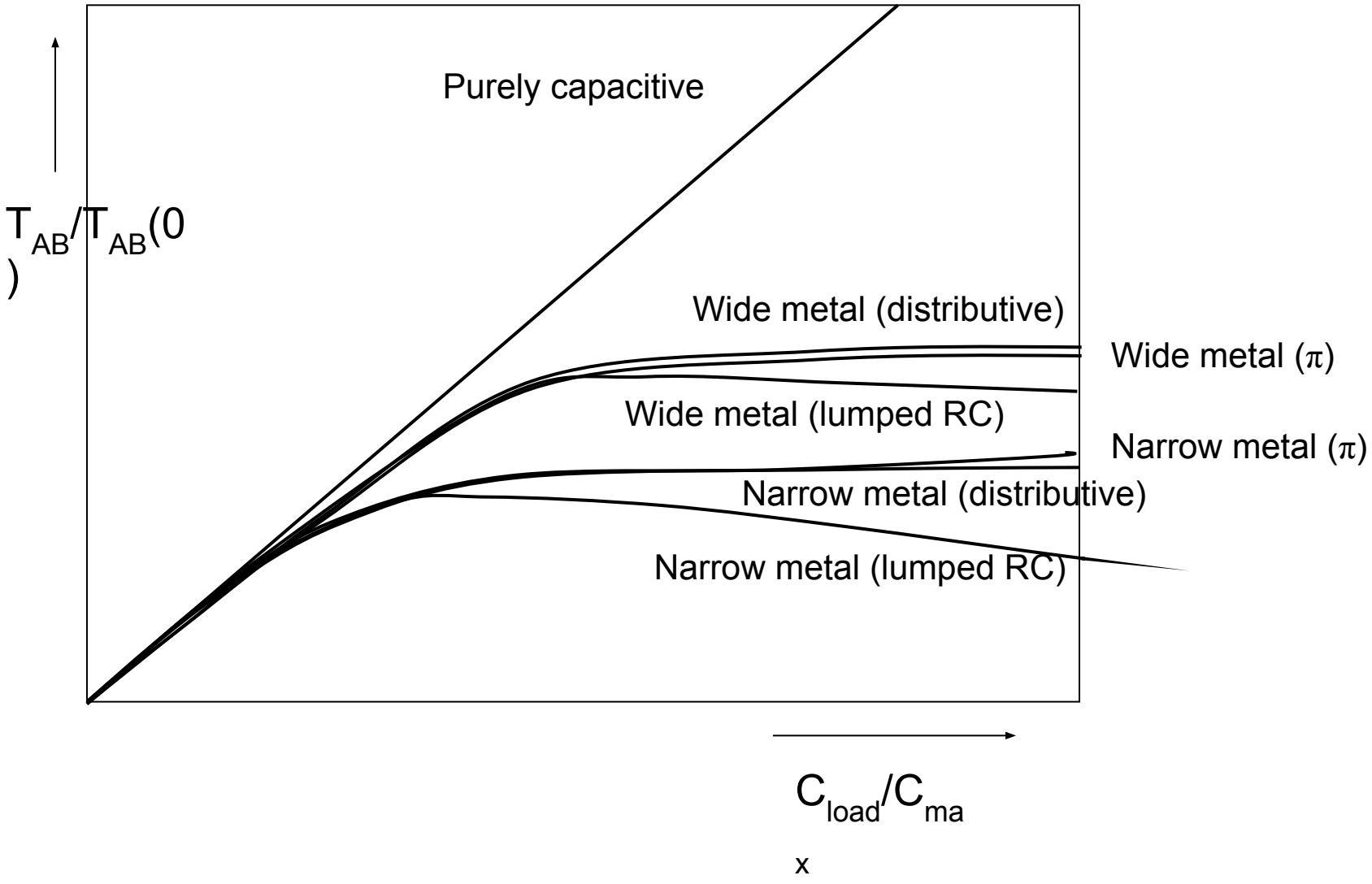


$$y_U^{(1)} = \sum_{i=1}^{B} y_{Di}^{(1)}$$

$$y_U^{(2)} = \sum_{i=1}^{B} y_{Di}^{(2)}$$

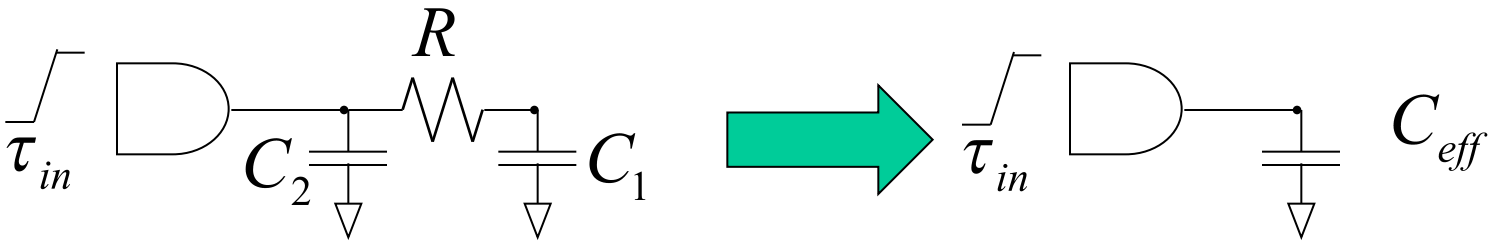
$$y_U^{(3)} = \sum_{i=1}^{B} y_{Di}^{(3)}$$

Example: Uniform Distributed RC Segment

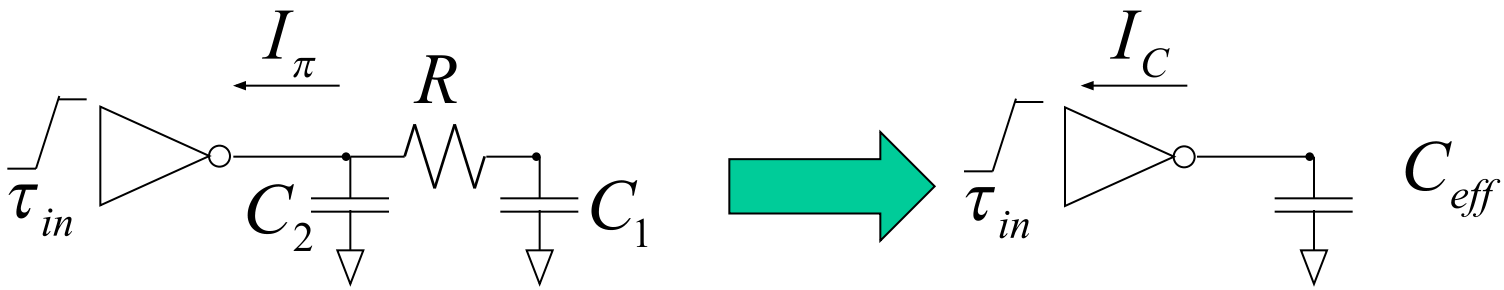


Why Effective Capacitance Model?

- The π-model is incompatible with existing empirical device models
  - Mapping of 4D empirical data is not practical from a storage or run-time point of view
- Convert from a π-model to an effective capacitance model for compatibility
- Equate the average current in the π-load and the C_{eff} load



Equating Average Currents



$$\frac{1}{t_D}\int_0^{t_D} I_\pi(t)dt = \frac{1}{t_D}\int_0^{t_D} I_C(t)dt$$

$$I_\pi(s) = Y_\pi(s)V_{out}(s)$$

$$I_C(s) = sC_{eff}V_{out}(s)$$

t_D = time taken to reach 50% point,
not 50% point of input to 50% point of output

Waveform Approximation for $V_{out}(t)$

- Quadratic from initial voltage (Vi = VDD for falling waveform) to 20% point, linear to the 50% point

$$V_{out}(t) = \begin{cases} V_i - ct^2 & 0 \le t \le t_x \\ a + b(t - t_x) & t_x \le t \le t_D \end{cases}$$

- Voltage waveform and first derivative are continuous at t_x

$$a = V_i - ct_x^2$$

$$b = -2ct_x$$

Average Currents in Capacitors

$$\bar{I}_C(t) = \frac{1}{t_D}\left[\int_0^{t_x} C_{eff}\cdot(-2ct)dt + \int_{t_x}^{t_D} C_{eff}\cdot(-2ct_x)dt\right]$$

$$= \frac{-2C_{eff}\cdot c\cdot t_x}{t_D}[t_D - \frac{t_x}{2}]$$

$$\bar{I}_{C_2}(t) = \frac{-2C_2\cdot c\cdot t_x}{t_D}[t_D - \frac{t_x}{2}]$$

- Average current of C_1 is not quite as simple:
  - Current due to quadratic current in C_2
  - Current due to linear current in C_2

Average Currents in C_1

- Average current for $(0,t_x)$ in C_2

$$\bar{I}_{C_1}(t) = \frac{-2C_1 \cdot c}{t_x}\left[\frac{t_x^2}{2} - RC_1 t_x + (RC_1)^2\left(1 - e^{\frac{-t_x}{RC_1}}\right)\right]$$

- Average current for (t_x,t_D) in C_2

$$\bar{I}_{C_1}(t) = \frac{-C_1 \cdot c}{t_D - t_x}\left[2RC_1 t_x - 2(RC_1)^2\left(1 - e^{\frac{-t_x}{RC_1}}\right)\right]\left(1 - e^{\frac{-(t_D - t_x)}{RC_1}}\right)$$

$$-2ct_xC_1\left[1 - \frac{RC_1}{t_D - t_x}\left(1 - e^{\frac{-(t_D - t_x)}{RC_1}}\right)\right]$$

- Average current for $(0,t_D)$ in C_2

$$\bar{I}_{C_1}(t) = \frac{-2C_1 \cdot c}{t_D}\left[\frac{t_x^2}{2} - RC_1 t_x + (RC_1)^2\left(e^{\frac{-(t_D - t_x)}{RC_1}} - e^{\frac{-t_D}{RC_1}}\right)\right]$$

Computation of Effective Capacitance

- Equating average currents

$$C_{eff} = C_2 + C_1\left[1 - \frac{RC_1}{t_D - \frac{t_x}{2}} + \frac{(RC_1)^2}{t_x\left(t_D - \frac{t_x}{2}\right)}\left(e^{\frac{-(t_D - t_x)}{RC_1}} - e^{\frac{-t_D}{RC_1}}\right)\right]$$

- Problem: t_D and t_x are not known a priori
- Solution: iterative computation
  - Set the load capacitance equal to total capacitance
  - Use table-lookup or K-factor equations to obtain t_D and t_x

$$t_D = t_d + \tau_{in}/2$$
$$t_x = t_D - \tau/2$$

  - Equate average currents and calculate effective capacitance
  - Set load capacitance equal to effective capacitance and iterate

Stage Delay Computation

- Calculate output waveform at gate
  - Using Ceff model to model interconnect
- Use the output waveform at gate as the input waveform for interconnect tree load
- Apply interconnect reduced-order modeling technique to obtain output waveform at receiver pins