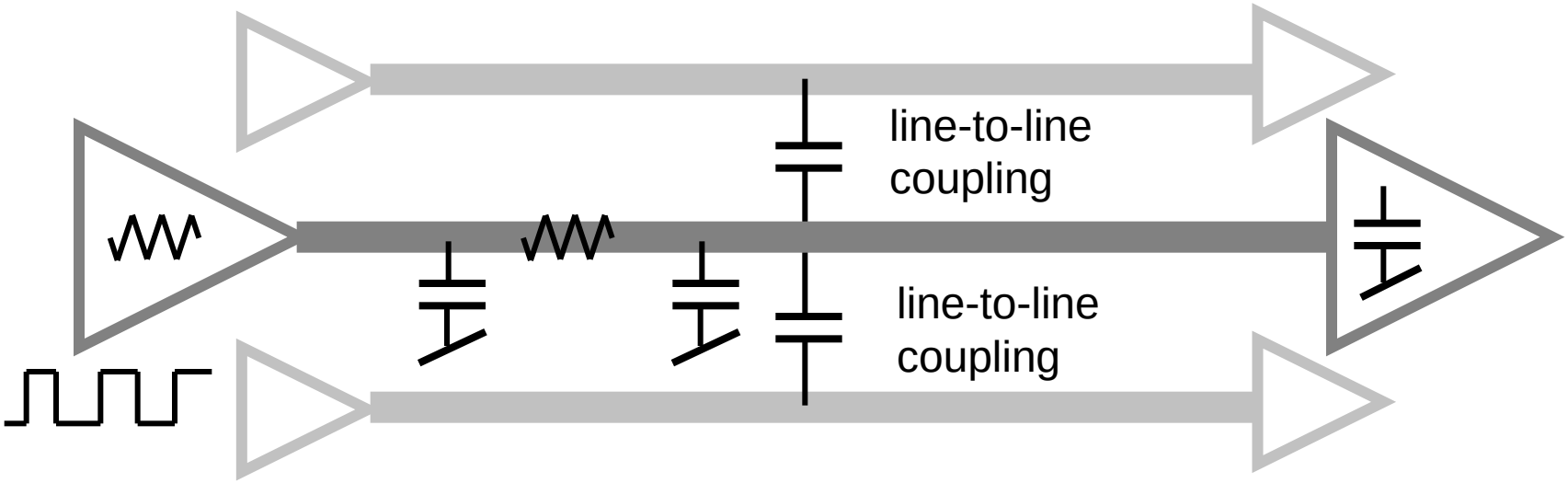
Interconnect Power and Delay Optimization
by
Dynamic Programming
in
Gridded Design Rules

Konstantin Moiseev, Avinoam Kolodny
EE Dept. Technion, Israel Institute of Technology
Shmuel Wimer
Eng. School, Bar-Ilan University

Agenda

- What is the problem?
- Results for 32nm design
- It is NP complete
- Optimal solution by dynamic programming
- How it works in practice
- Further research problems

Interconnect Power and Delay

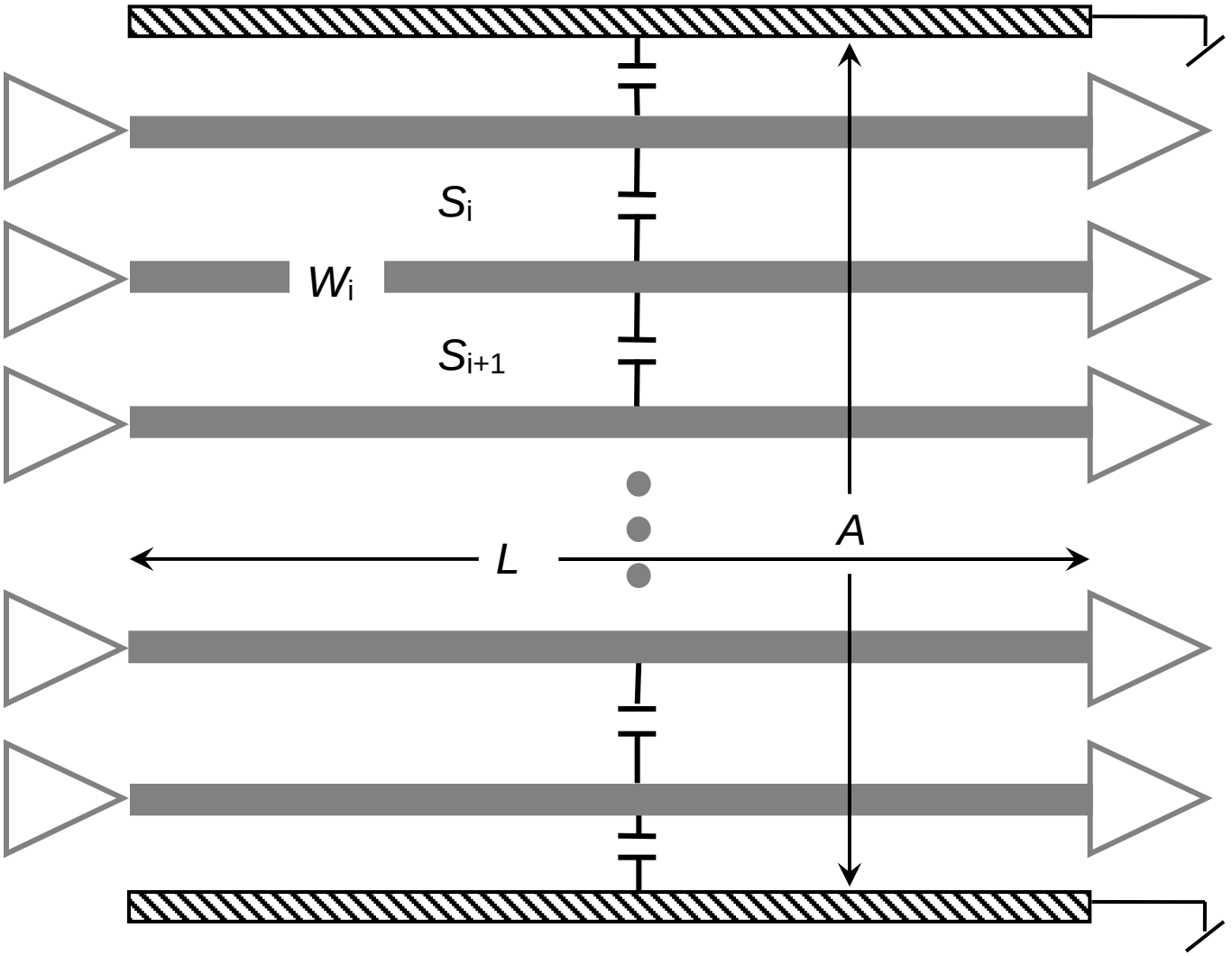


signal's activity, 0<= *AF* <=1

Line-to-line coupling is dynamic power killer

Using Elmore delay model, simple, inaccurate but with high fidelity

Interconnect Bus Model



Delay and Dynamic Power Minimization

Delay:

$$D_i\left(s_{i-1}, w_i, s_i\right) = \alpha_i + \beta_i w_i + \frac{\gamma_i}{w_i} + \left(\delta_i + \frac{\varepsilon_i}{w_i}\right)\left(\frac{1}{s_{i-1}^{\mu}} + \frac{1}{s_i^{\mu}}\right)$$

$\alpha_i,\ \beta_i,\ \gamma_i,\ \delta_i,\ \varepsilon_i$ - technology parameters, driver's resistance, capacitive load and bus length L.

Dynamic power:

$$P_i\left(s_{i-1}, w_i, s_i\right) = \kappa_i w_i + \eta_i\left(\frac{1}{s_{i-1}^{\mu}} + \frac{1}{s_i^{\mu}}\right)$$

κ_i, η_i - technology parameters, signal's activity, and bus length L.

Formulation of the Problem

Minimize delay: $D^{sum} = \sum_{i=1}^{n} D_i\left(s_{i-1}, w_i, s_i\right)$ o $D^{\max} = \max_{1 \le i \le n} D_i\left(s_{i-1}, w_i, s_i\right)$

Minimize power: $P = \sum_{i=1}^{n} P_i\left(s_{i-1}, w_i, s_i\right)$

Subject to

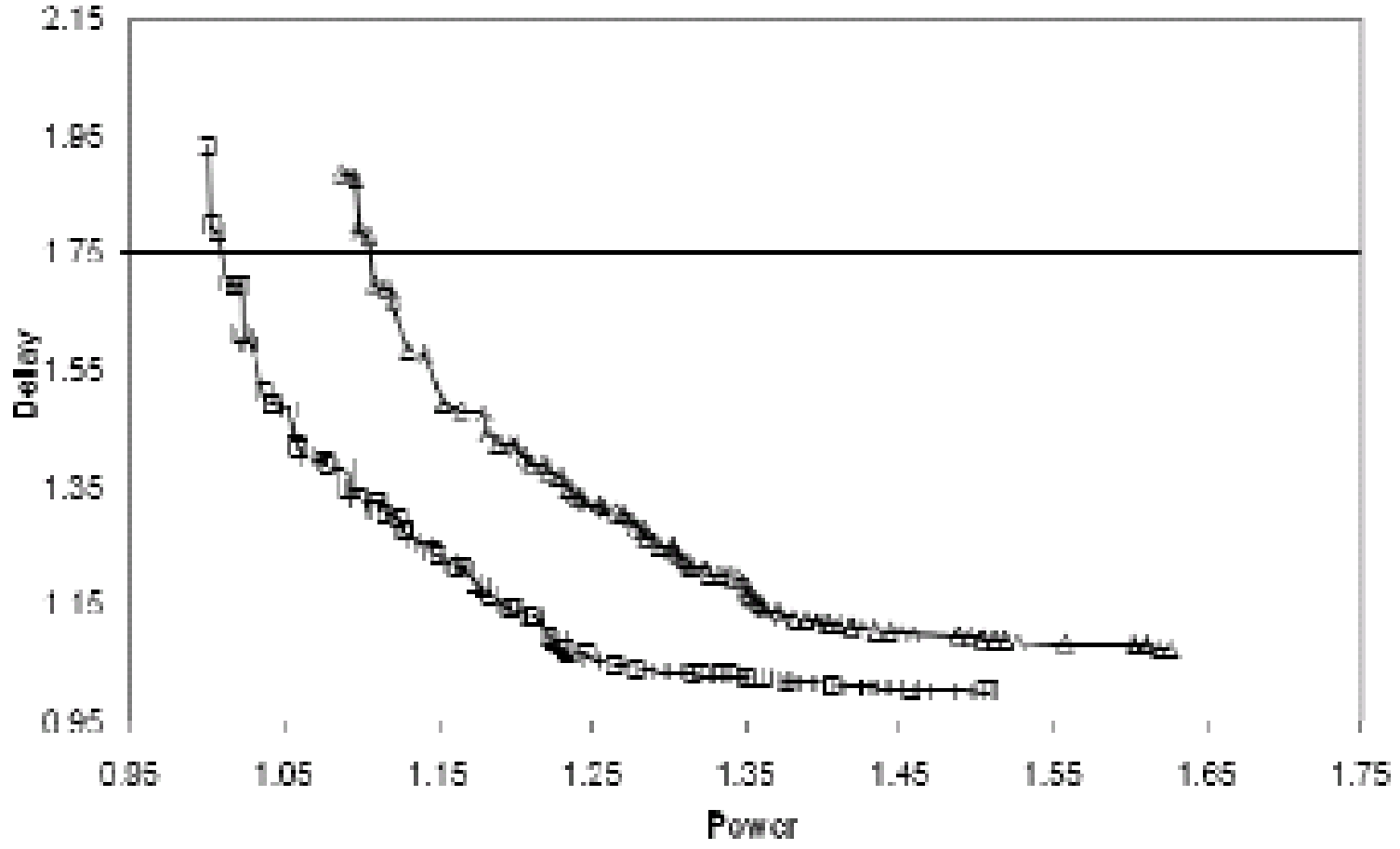
Constrained area: $\sum_{i=1}^{n} w_i + \sum_{i=0}^{n} s_i = A$

In 32nm and 22nm: $s_i \in \left\{S_1, \ldots, S_p\right\},\ w_i \in \left\{W_1, \ldots, W_q\right\}$

Discrete optimization: ☹ problem is NP-complete ☹☹

Dynamic programming works ☺☺☺

Power-Delay “Shape Function”



Results Obtained for 32nm

- Implemented in C++ / OpenAccess
- Ran on 32nm control blocks of Intel mobile processor
  - Routed by Synopsys tool
  - Width and space re-allocated in metal 2, 3 and 4
  - Used effective drivers and loads from netlist
  - Typical block size was 250u X 250u
- Both dynamic power and delays are reduced
- 10%-15% dynamic power reduction
  - Per optimized layer
- 2% - 5% delay reduction

MIN_DLYPWR Problem

Question **:** Is there a setting of widths and spaces such that delay reduction from base is δD at least, while power increase from base is δP at most?

MIN_DLYPWR is NPC by polynomial reduction of PARTITION, which attemps to answer whether for a given set B whose elements have size $s(b) \in \mathbb{Z}^+$, $\forall b \in B$, there's a subset satisfying $\sum_{b \in B'} s(b) = \sum_{b \in B - B'} s(b)$.

MIN_MAX_DLYPWR Problem

Question : Is there a setting of widths and spaces such that power decrease from base power is at least δP while maximal delay is increasing by δD at most?

MIN_MAX_PWRDLY is NPC by polynomial reduction of SUBSET_SUM which answers whether for B whose elements have size $s(b) \in \mathbb{Z}^+$, and diven a number $N \in \mathbb{Z}^+$, there is $B' \subseteq B$ satisfying $\sum_{b \in B'} s(b) = N$.

Dynamic Programming Solution

$$\text{Delay is additive: } D(1,n) = \begin{cases} D(1,j) + D(j+1,n) \\ \max\{D(1,j), D(j+1,n)\} \end{cases}$$

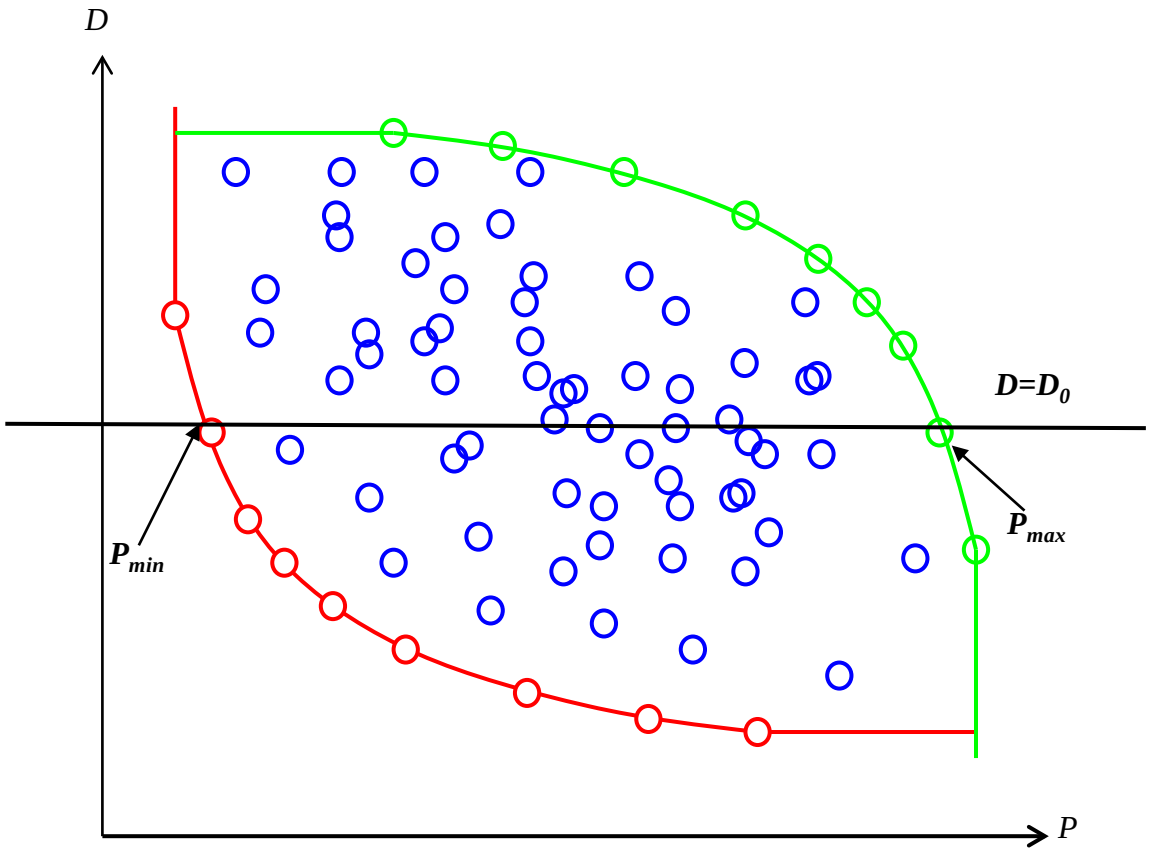
$$\text{Power is additive } P(1,n) = P(1,j) + P(j+1,n)$$

$$\text{Area is additive } A(j,n) = A - \left(\sum_{i=0}^{j} w_i + \sum_{i=0}^{j} s_i \right)$$

Minimization of power and delay from *j+1* to *n* is independent of power and delay from *1* to j.

This suggests dynamic programming. Algorithm generates only essential *(P,D)* pairs in progression from wire to wire.

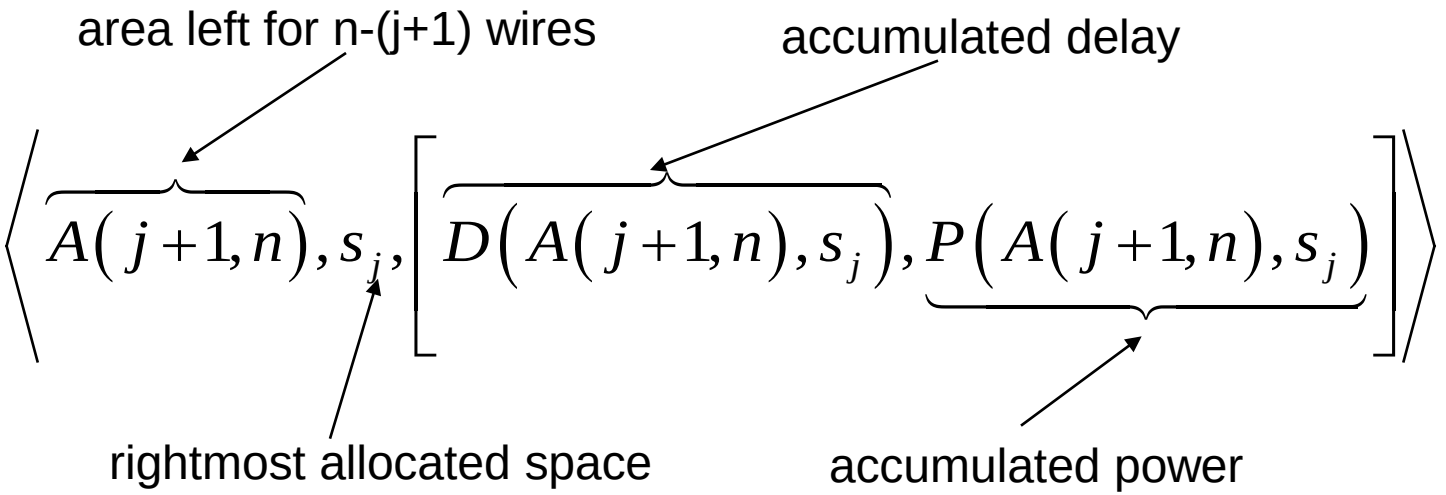
Power-Delay Solution Space



Dynamic Programming finds the red curve progressively.

Optimal solution is derived from solution space of last wire.

State Definition for Dynamic Programming



State Dominancy and Redundancy

Allocation $\omega' : \left(w'_0, s'_0, \ldots, w'_j, s'_j \right)$

is ***dominating*** allocation $\omega'' : \left(w''_0, s''_0, \ldots, w''_j, s''_j \right)$ if

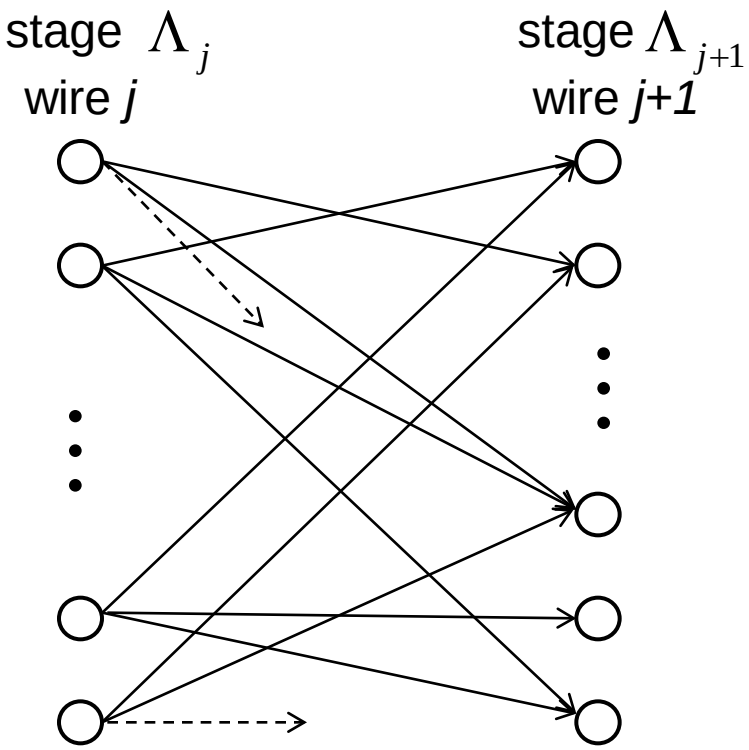
$$A - \left(\sum_{i=0}^{j} s'_i + \sum_{i=0}^{j} w'_i \right) \geq A - \left(\sum_{i=0}^{j} s''_i + \sum_{i=0}^{j} w''_i \right)$$

$s'_j \geq s''_j$ and

$$D(\omega') \leq D(\omega'') \wedge P(\omega') \leq P(\omega'')$$

Stage Progression and State Augmentation

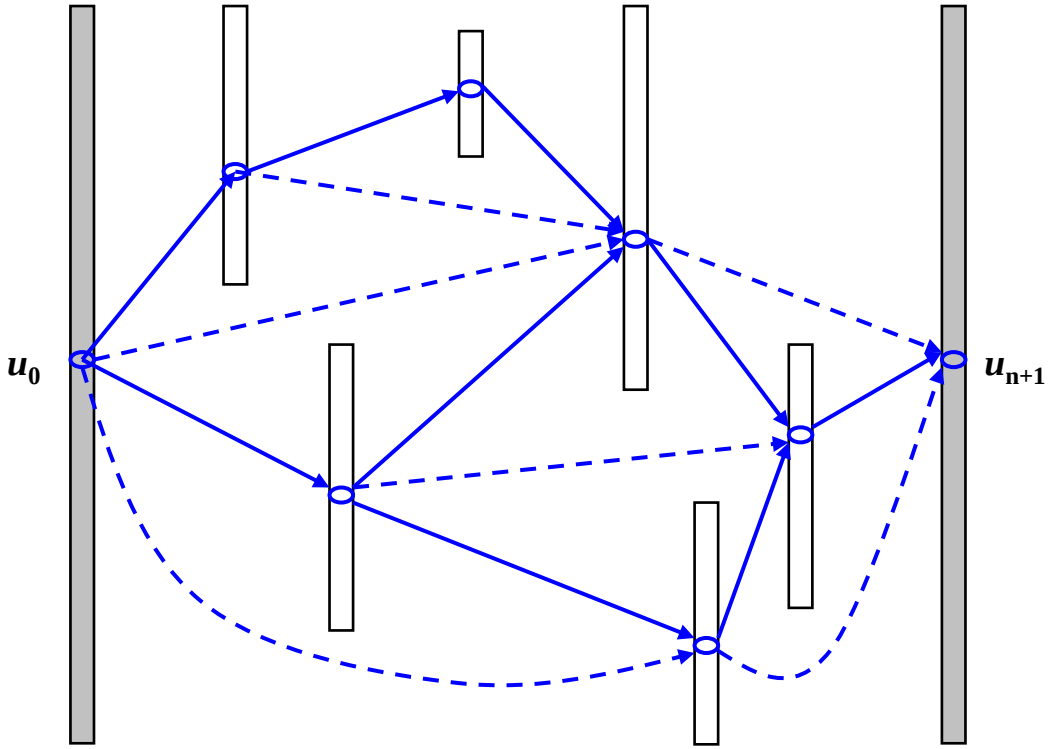
Width and space allocations of next wire



Theorem *(optimality):* Stage Λ_n of the DP algorithm contains all the feasible non-redundant, and hence optimal, power-delay pairs that can be obtained by any width and space allocation to *n* wires

Theorem: Any power-delay function $f(P, D)$ monotonically increasing in P and D achieves minimum on the boundary of the power-delay feasible region.

Modeling Real Layout



Use transitive reduction of wire visibility graph

Design rules are transitively closed

Process wires from left to right in topological order with appropriate enhancement to power-delay calculations

Time and Storage

Time complexity:

$$O\left(pq^{\alpha+\beta} n^3 \log n/\varepsilon \right)$$

α and β are max in-degree and out-degree, respectively, of wire adjacency graph vertex

Storage complexity:

$$O\left(q^{\beta} n^3 / \varepsilon \right)$$

Time and storage in practice are manageable due to power grid which decomposes the problem into many independent smaller problems.

Further Research Directions

- Filling aware optimization
  - Dynamic programming can generate filling patterns!
  - Line-to-line capacitance can be measured on the spot
- Current algorithm works on P&R style only
  - Enhancement for full-custom design style
  - Cross-hierarchy dynamic programming
- Is “bang-bang” sizing possible?
  - Using two values only is tremendous for litho!
- Simultaneous cell and interconnect resizing
  - Use cell families with same footprint

Thank You!

Backup

NP Completeness Proof of Delay Sum

It is NP since any substitution of valid guess into delay and power equations can be checked for YES or NO answer.

Reduction of PARTITION into MIN_DLYPWR:

1. A wire is allocated for every element of PARTITION.
2. Resistance of drivers and wires are set to 0 and 1, resp., hence wire resistance is not affecting delays.

Coeficients in delay and power equations are set to zero or one, except load and activity, yielding:

$D_b = C_b / w_b$ and $P_b = F_b w_b$.

3. Only one spacing is allowed,

   hence not affecting the problem.

4. Only two width values are allowed $\{W_1, W_2\}$.

5. Bus area is set sufficiently large,

   hence not affecting the problem.

6. Activity factors are set to $s(b)/(W_2 - W_1)$.

   Capacitive loads are set $s(b)W_1W_2/(W_2 - W_1)$.

   Delay turns into: $D^{sum} = \sum_{b \in B} (1/w_b)\, s(b) W_1 W_2 / (W_2 - W_1)$,

   Power turns into: $P = \sum_{b \in B} w_b s(b) / (W_2 - W_1)$.

7. Bounds of power increase and delay reduction

   are set to $\delta P = \delta D = \sum_{b \in B} s(b)/2$.

Transformation consumes polynomial time.

Let the answer to $f(I)$ of MIN_DLYPWR be YES.

Instance $f(I)$ is set such that P increases and D decreases in $\overline{w}$. There's single P and D where:

$$\sum_{b\in B'}\delta D_b = \sum_{b\in B'}\delta P_b = \sum_{b\in B} s(b)/2.$$

We obtained $\left(\sum_{b\in B} s(b)\right)/2 = \sum_{b\in B'}\delta D_b =$

$$\sum_{b\in B'}\left(\frac{1}{W_1}-\frac{1}{W_2}\right)s(b)\frac{W_1W_2}{W_2-W_1} = \sum_{b\in B'} s(b),$$

implying that $(B', B-B')$ solves PARTITION.

Conversely, let $B' \subseteq B$ be a YES answer to PARTITION.

Set width of w_b, $b \in B'$, to W_2, and rest wires stay W_1.

D reduction is $\delta D_b = C_b \left(1/W_1 - 1/W_2\right) = s\left(b\right)$, and P increase is $\delta P_b = F_b \left(W_2 - W_1\right) = s\left(b\right)$, thus yielding a YES answer to the MIN_DLYPWR problem.

Q.E.D

NP Completeness Proof of Max Delay

Base delay and power are obtained by setting widths to W_2, resulting maximum power and minimum delay for each signal.

Settings 1 to 5 of MIN_MAX_DLYPWR instance are similar to those of MIN_DLYPWR proof.

Base delays are increasing upon wire narrowing. Setting 6 in MIN_DLYPWR proof is modified such that load is set to $C_b = N W_1 W_2 / (W_2 - W)$, yielding:

$$D^{max} = \max_{b \in B} \left\{ N (1/w_b) \left(W_1 W_2 / (W_2 - W) \right) \right\}.$$

For any $b \in B,$ wire narrowing from W_2 to W_1 reduces power by $s(b)$. Delay grows by $\delta D_b = N,$ so maximum delay increase by N always.

Setting 7 in MIN_DLYPWR proof turns to $\delta P = \delta D = N.$

Consequently, $\delta P = \delta D = N$ iff $\sum_{b \in B'} \delta P_b = \max_{b \in B} \{\delta D_b\} = N.$

This holds iff the answer to SUBSET _ SUM is YES.

Q.E.D