Performance Driven Multi-level and Multiway Partitioning with Retiming

Jason Cong and Sung Kyu Lim
UCLA VLSI CAD Lab

Chang Wu
Aplus Design Technologies, Inc.

Support : MARCO/DARPA GSRC
and NSF Young Investigator Award MIP-9357582
Outline

• Introduction
• Problem Formulation
• HPM Algorithm
• Experimental Results
• Conclusions & Ongoing Work
Motivation

- DSM VLSI technology
  - Interconnect delay far exceeds gate delay
  - Post-layout optimization is still not enough [NTRS97, Cong97]

<table>
<thead>
<tr>
<th>Tech (um)</th>
<th>0.25</th>
<th>0.18</th>
<th>0.15</th>
<th>0.13</th>
<th>0.10</th>
<th>0.07</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gate delay (ns)</td>
<td>0.071</td>
<td>0.051</td>
<td>0.049</td>
<td>0.046</td>
<td>0.039</td>
<td>0.022</td>
</tr>
<tr>
<td>1mm (ns)</td>
<td>0.059</td>
<td>0.049</td>
<td>0.051</td>
<td>0.044</td>
<td>0.052</td>
<td>0.042</td>
</tr>
<tr>
<td>2cm un-opt (ns)</td>
<td>2.08</td>
<td>1.97</td>
<td>2.06</td>
<td>2.07</td>
<td>2.89</td>
<td>3.52</td>
</tr>
<tr>
<td>2cm opt (ns)</td>
<td>0.89</td>
<td>0.79</td>
<td>0.77</td>
<td>0.70</td>
<td>0.77</td>
<td>0.67</td>
</tr>
<tr>
<td>Exp. clk (GHz)</td>
<td>0.8</td>
<td>1.2</td>
<td>1.4</td>
<td>1.6</td>
<td>2.0</td>
<td>2.5</td>
</tr>
<tr>
<td>Req. delay (ns)</td>
<td>1.07</td>
<td>0.67</td>
<td>0.57</td>
<td>0.50</td>
<td>0.40</td>
<td>0.32</td>
</tr>
</tbody>
</table>

- Interconnect-centric design methodology is essential
Role of Partitioning

• **New perspective**
  – Conventional view: enables divide-and-conquer
  – DSM view: defines global and local interconnects
  – Small blocks (50K-100K gates) can be synthesized reliably
  – Key is *simultaneous* optimization of delay and cutsizes

**D >> d !!!**

![Diagram](image-url)
Impact of Simultaneous Retiming

- Wider solution space [Pan et al, TCAD98]
  - allows retiming to hide some global interconnect delays

\[ \phi(A) = 8 \]

\[ \phi(B) = 8 \]

\[ \phi(A) = 8 \]

\[ \phi(B) = 6 \]
Related Works

• Top-down approaches (cutsize only)
  – Iterative improvement [KL70, FM82, Kr84, San89]
  – Spectral based [HK92, AZ95]
  – Clustering method [SU72, NOP87, WC92, SS93, CS93, HK95]
  – Network flow based [YW94, YW97]
  – Analytical based [RDJ94, LLC95]
  – Multi-level [CS93, HB95, AHK97, KA+97, KK99]

• Bottom-up approaches (delay only)
  – Unit delay model [LLT69, CD93]
  – General delay model [MBV91, RW93, YW95]
  – Sequential circuits with retiming [PKL98, CLW99]
Our Contributions

• New perspective
  – Partitioning defines interconnects
  – Combined objective: cutsize and delay
  – Retiming to allow multiple clock cycles across the chip

• HPM algorithm
  – Multi-level approach with simultaneous retiming
  – Efficient, effective, and scalable
  – Cutsize of hMetis [Karypis et al, DAC97]
  – Delay of PRIME [Cong et al, DAC99]
Problem Definition

- **Performance Driven Partitioning**

  **Instance:** netlist $NL = (C, N)$, area $a(c)$ and delay $d(c)$ for each cell $c$, and area constraints $A = (a_i, b_i)$ for each partition.

  **Question:** is there a partition $P$ of $C$ into non-empty disjoint sets $B_1, B_2, \ldots, B_K$ such that $a_i \leq |B_i| \leq b_i$, and such that $\alpha \cdot \text{cutsise} + \beta \cdot \text{delay}$ is minimized?
Cutsize Objective

- Cutsize objective $\theta(P)$
  - Based on hypergraph model $H = (V, E)$
  - Cost 1 metric: $c(e) = 1$ if $e$ spans more than 1 block
  - Efficient gain computation and update

$\theta(P) = 3$
Delay Objective

- Delay objective $\phi(P)$
  - Based on directed graph model $G = (V, E)$
  - Gate delay model [Murgai et al, ICCAD91]: $d(v), d(e) = D$
  - $\phi(P) = \max\{d(p) | p: a \rightarrow b, a \in PI \text{ or FF} \& \ b \in PO \text{ or FF}\}$
  - Complete path analysis ($=O(n)$) required

\[ \phi(P) = 4d + D \]
Retiming

- **Node labeling [LS91]**
  - Edge weight: number of FFs
  - \( r(v) = \# \) of FFs moved from fan-out to fan-in
  - \( w^r \geq 0 \) & \( w^r(p) \geq 1 \) for each path \( p \) such that \( d(p) > \varphi \)

\[ \varphi(P) = 4d + D \]

\[ \varphi(P) = d + D \]
Sequential Arrival Time (SAT)

- **Definition [Pan et al, TCAD98]**
  - $l(v) = \text{delay from PIs to } v \text{ after retiming under given } \phi$
  - $l(v) = \max\{l(u) - \phi \cdot w(u,v) + d(u,v) + d(v)\}$

\[
\begin{align*}
  l(u) &\quad w(u,v) &\quad d(v) \\
  u &\quad &\quad v
\end{align*}
\]

- **Relation to retiming:** $r(v) = \left\lceil \frac{l(v)}{\phi} \right\rceil - 1$
- **Theorem:** $P$ can be retimed to $\phi + D$ iff $l(POs) \leq \phi$

\[
\begin{align*}
  l(u) &= 7 &\quad u &\quad v \\
  l(w) &= 3 &\quad w
\end{align*}
\]

\[
\begin{align*}
  d(v) &= 1, \quad d(e) = 2, \quad \phi = 5 \\
  l(v) &= \max\{7-5\cdot1+2+1, 3+2+1\} = 6
\end{align*}
\]
Computation of SAT

- Single source longest path algorithm
  - Edge lengths may be negative due to FFs
  - Converge fast: $O(n)$ in practice

initialize $l(v) = -\infty$, $l(PI) = 0$;
for ($i = 1$ to $|V|$)
  DONE = false;
  visit each vertex $v$
    for each fan-in $u$ of $v$
      $l'(v) = \max\{l(u) - \phi \cdot w(u,v) + d(u,v) + d(v)\}$;
      if ($l(v) < l'(v)$)
        $l(v) = l'(v)$; DONE = true;
      if ($v = PO$ and $l'(v) > \phi$)
        return(FAILURE);
    if (DONE = false)
      return(SUCCESS);
Multi-level Method

- Recursive coarsening and refinement
  - From coarse-grain into finer-grain optimization
  - Successfully used in partial differential equations, image processing, etc, and circuit partitioning
Overview of HPM Algorithm

- Multi-level approach

- Cutsize oriented ESC clustering

- Delay oriented PRIME clustering

- Retiming

- Both cutsize and delay oriented xLR partitioning
Phase 1: PRIME Clustering

• Quasi-optimal clustering algorithm with retiming
  – Node label = sequential arrival time
  – If node label of every PO $\leq \phi$, then $\phi + D$ is feasible

• Limitations of existing work [Pan et al, TCAD98]
  – High space requirement: $O(n^2)$
  – High time complexity: $O(n(n+m)\log^2 n)$

• Contributions of PRIME [Cong et al, DAC99]
  – Revealed monotone property of node labels
  – Significant reduction of candidate set for label update
  – Efficient longest path computation
  – Limitation: large cutsize
Phase 2: ESC Clustering

• Edge separability [Cong & Lim, ASPDAC00]
  – Min # of edges to separate $x$ and $y$: $x$-$y$ mincut

  \[ w(e) \leq q(e) \leq \lambda(e) \]

• ESC clustering algorithm
  – Can compute a tight lower-bound $q(e)$ of $\lambda(e)$ for all edges in $O(n\log n)$ time [Nagamochi & Ibaraki, Algorithmica92]
  – Use $q(e)$ for bottom-up multi-level clustering
  – Produce very good cutsize, comparable to hMetis [KA+97]
Phase 3: xLR Partitioning

- Acyclic partitioning [Cong & Lim, ASPDAC00]
  - Maintain acyclicity among partitions
  - New gain function that discourages violation of acyclicity
  - Effective in improving delay while maintaining cutsize
Declustering and Refinement

- Declustering increases search space at each level
  
  Cutsize can be reduced from 4 to 3

- MRP (Multiple Rollback Point) scheme
  
  FM : keep \( m3 \) only
  MRP : keep \( m1, m2, m3 \) and pick min-delay sol.
HPM: Experimental Results

- Comparison among existing algorithms
  - FM [FM82], hMetis [DAC97], PRIME [DAC99], and HPM [DAC00]
HPM: Experimental Results

- D/d increases as technology further scales
  - Delay advantage increases as technology advances
Conclusions & Ongoing Work

• New perspective
  – Partitioning defines interconnects
  – Retiming to allow multiple clock cycles across the chip
  – Combined objective: cutsize and delay

• HPM algorithm
  – Multi-level approach with simultaneous retiming
  – Efficient, effective, and scalable
  – Tradeoff: cutsize of hMetis + delay of PRIME

• Ongoing Work
  – Consider geometric info: GEO [Cong & Lim, ICCAD 2000]
## Benchmark Characteristics

- MCNC and Quickturn circuits

<table>
<thead>
<tr>
<th>Name</th>
<th>GA</th>
<th>PI</th>
<th>PO</th>
<th>FF</th>
<th>Net</th>
</tr>
</thead>
<tbody>
<tr>
<td>s9234</td>
<td>1290</td>
<td>28</td>
<td>39</td>
<td>135</td>
<td>1492</td>
</tr>
<tr>
<td>s5378</td>
<td>1443</td>
<td>35</td>
<td>49</td>
<td>163</td>
<td>1690</td>
</tr>
<tr>
<td>s13207</td>
<td>3146</td>
<td>59</td>
<td>152</td>
<td>486</td>
<td>3843</td>
</tr>
<tr>
<td>s15850</td>
<td>3784</td>
<td>76</td>
<td>150</td>
<td>515</td>
<td>4525</td>
</tr>
<tr>
<td>bigkey</td>
<td>8599</td>
<td>228</td>
<td>197</td>
<td>224</td>
<td>9248</td>
</tr>
<tr>
<td>s38584</td>
<td>13209</td>
<td>38</td>
<td>304</td>
<td>1423</td>
<td>14974</td>
</tr>
<tr>
<td>clma</td>
<td>30552</td>
<td>61</td>
<td>82</td>
<td>33</td>
<td>30728</td>
</tr>
<tr>
<td>ind1</td>
<td>29780</td>
<td>2630</td>
<td>3242</td>
<td>603</td>
<td>36255</td>
</tr>
<tr>
<td>ind2</td>
<td>26060</td>
<td>2772</td>
<td>6242</td>
<td>1755</td>
<td>36829</td>
</tr>
<tr>
<td>ind3</td>
<td>52197</td>
<td>2801</td>
<td>3070</td>
<td>2001</td>
<td>60069</td>
</tr>
<tr>
<td>ind4</td>
<td>101531</td>
<td>4155</td>
<td>4575</td>
<td>8333</td>
<td>118566</td>
</tr>
</tbody>
</table>
Delay Ratio Computation

- Computation of $D$ based on NTRS’97
  - $R_d$ and $C_L$ are x10 (local) and x100 (global) min device

<table>
<thead>
<tr>
<th>Tech ($\mu m$)</th>
<th>Local Interconnect</th>
<th>Global Interconnect</th>
</tr>
</thead>
<tbody>
<tr>
<td>$l$ (mm)</td>
<td>0.25 0.18 0.13 0.10 0.07</td>
<td>0.25 0.18 0.13 0.10 0.07</td>
</tr>
<tr>
<td>$w$ ($\mu m$)</td>
<td>1.25 0.9 0.65 0.5 0.35</td>
<td>1.20 18.4 20.7 22.8 24.9</td>
</tr>
<tr>
<td>$R_d$ ($\Omega$)</td>
<td>1620 1710 2210 2340 2210</td>
<td>162 171 221 234 221</td>
</tr>
<tr>
<td>$C_L$ ($fF$)</td>
<td>2.82 2.34 1.35 0.72 0.66</td>
<td>28.2 23.4 13.5 7.2 6.6</td>
</tr>
<tr>
<td>$R_w$ ($\Omega$)</td>
<td>365 340 405 460 475</td>
<td>187 182 101 110 95</td>
</tr>
<tr>
<td>$C_w$ ($fF$)</td>
<td>121 67.3 31.8 25.2 15.4</td>
<td>2924 2348 1779 1962 2423</td>
</tr>
<tr>
<td>Delay (ps)</td>
<td>224 131 80 67 40</td>
<td>757 623 487 569 653</td>
</tr>
<tr>
<td>$D$</td>
<td>1 1 1 1 1</td>
<td>3.4 4.8 6.1 8.5 16.3</td>
</tr>
</tbody>
</table>