A-Priori Wirelength and Interconnect Estimation Based on Circuit Characteristics

Shankar Balachandran    Dinesh Bhatia

Center for Integrated Circuits and Systems

University of Texas at Dallas

shankars@utdallas.edu
dinesh@utdallas.edu
Outline

- Introduction
- Motivation and Prior Work
- Definitions
- Wirelength Prediction
- Routing Demand Prediction
- Conclusion
**Introduction**

- **Interconnect Prediction**
  - breaks repetitive design convergence loop
  - helps in performing early feasibility studies

- **A-Priori Prediction**
  - is done before placement stage
  - is used to provide congestion map, wirelength metrics
  - can be used for architecture evaluation
Scope of This Work

- A-Priori wirelength and interconnect prediction for island-style FPGAs
- Bounding box prediction for all wires
  - Identifying important circuit characteristics which constrain placement
  - Assumes that wirelength is minimized during placement
- Routing demand estimation
  - Channel Width calculation
- No prior characterization of placement/router
Prior Work

- Rent’s Rule [TVLSI 2000, Bakoglu]
  - Interconnect prediction builds models for architecture, circuit and placement
  - Can calculate average/total wirelength, congestion etc.

- Sechen [ICCAD 87]
  - Average wirelength of optimized placements
  - For all possible bounding boxes, enumerate all possible positions for sources and sinks to calculate average wirelength of the whole netlist

- Hamada et al. [DAC 92]
  - Break down nets into cliques and perform neighborhood analysis on them
  - Placement is considered a stochastic process
  - Wirelength distribution is calculated

- Bodapati et al. [SLIP 00]
  - Bounding box estimates using structural analysis
  - Needs calibration of placement/router
Motivation

- **Average wirelength is not sufficient**
  - Rent’s rule, Sechen and Hamada et al. report only these figures

- **Individual wirelength is useful**
  - Logic synthesis, floorplanners

- **Congestion metrics should be quantifiable**
  - Very important for FPGAs
  - Channel Width requirements for routers
    - To avoid the chicken and egg problem
Island Style FPGA

Tracks

Connection Block

Switch Block

CLBs

$W = \text{Track Width}$
Methodology

- FPGA Architecture
- Circuit (Techmapped)

Wirelength Estimation

VPR Placement

Routing Demand Estimation

Compare Results

VPR Detailed Router

Estimation Quality
### Definitions [CIRC - TCAD 98, TVLSI 02]

<table>
<thead>
<tr>
<th><strong>Combinational level</strong> $c(x)$ of a node is:</th>
</tr>
</thead>
<tbody>
<tr>
<td>$c(x) = \max(c(x')</td>
</tr>
</tbody>
</table>

- $c_{\text{max}} = \max(c(x))$

<table>
<thead>
<tr>
<th><strong>Sequential level</strong> $s(x)$ of a node is:</th>
</tr>
</thead>
<tbody>
<tr>
<td>$s(x) = \begin{cases} 0 &amp; \text{if } x \text{ is a PI-node} \ s(x') + 1 &amp; \text{if } x \text{ is a FF-node with input } x' \ \min(s(x')</td>
</tr>
</tbody>
</table>

- $s_{\text{max}} = \max(s(x))$

#### Shape
- **Shape**: A vector
- $Shape[i] = c_0 \prod_{i=0}^{c_{\text{max}}} c_i$; $c_i = \text{number of nodes in level } i$
- For sequential circuits, the combinational shape vector in each level are concatenated back to back

**Diagram**

- **Combinational Level**
  - Level 1: Nodes A, B, C, D, E, F, G
  - $c_{\text{max}} = 5$
  - Shape = \{1, 2, 2, 1, 1\}

- **Sequential Level**
  - Level 1: Nodes A, B, C, D, E, F, G
  - $s_{\text{max}} = 1$
  - Shape = \{1, 2, 1\} \{2, 1\}
# Definitions (2)

- **Reconvergence** $R_{xy}$
  - Has multiple paths from $x$ to $y$
  - $x$ is the origin of reconvergence
  - $y$ is the destination of reconvergence
  - Always contained within one sequential level

- **Number of reconvergences**
  - $RN_{xy} = \text{Number of paths from } x \text{ to } y$

- **Length of reconvergence**
  - $RO_{xy}(x) = RI_{xy}(y) = \frac{1}{RN_{xy}} \sum_{p \in P_{xy}} l(p)$

<table>
<thead>
<tr>
<th>Path</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>$R_{BE}$</td>
<td>$RN = 2$</td>
</tr>
<tr>
<td>$l(p_1 = B \rightarrow C \rightarrow E) = 2$</td>
<td></td>
</tr>
<tr>
<td>$l(p_2 = B \rightarrow D \rightarrow E) = 2$</td>
<td></td>
</tr>
<tr>
<td>$RO_{BE}(B) = RI_{BE}(E) = 2$</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Path</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>$R_{AG}$</td>
<td>$RN = 2$</td>
</tr>
<tr>
<td>$l(p_1 = A \rightarrow (BE) \rightarrow G) = 3$</td>
<td></td>
</tr>
<tr>
<td>$l(p_2 = A \rightarrow F \rightarrow G) = 2$</td>
<td></td>
</tr>
<tr>
<td>$RO_{AG}(A) = RI_{AG}(G) = 2.5$</td>
<td></td>
</tr>
</tbody>
</table>

- $P_{xy}$ is the path-set from $x$ to $y$
- $l(p)$ is the length of path $p$
Overview of Our Methodology

- Reconvergence Analysis
  - Reconvergence Weights
- Bounding Box Estimation
  - Bounding Box Estimates
- Track Width Estimation
  - Track Width Estimates
Wirelength Estimation

- Wirelength of a circuit depends on
  - Structural properties of the circuit
  - Placement of the circuit

- Nets from **Input** pads usually feed more nodes than the other nodes
  - Hence, classify the nets as *logic nets* and *IO-nets* and treat them separately

- Wirelength of an individual net $N$ will depend on
  - Number of terminals – $t_N$
  - Interaction with other nets
Phase 1 - Minimum Span for Logic Nets

- Assume a net $N$ is tightly placed. Wirelength is optimal when
  - Source is placed in the center
  - All sinks are tightly packed around the source
- Minimum span $L$ is entirely dependent only on $t_N$

$$L = \frac{\sqrt{1 + 2 \cdot t_N + 1}}{2}$$

$$4 + 8 + 12 + L + 4 \cdot L \geq t_N$$
Phase 1 - Minimum Span for IO-Nets

- Input pads are placed along the periphery
- IO-Nets have many sinks usually, and the location of the Input pads cannot be guessed
- Assume the Input pad is in the corner of the FPGA
  - Worst-case calculation of wirelength
- Tight placement of sinks around the pad

\[ 2 + 3 + 4 + L + L \geq t_N \]

\[ L = \sqrt{2 \cdot t_N + 9/4} - 1/2 \]

\[
\begin{align*}
L &= 1 & N &= 2 \\
L &= 2 & N &= 5 \\
L &= 3 & N &= 9
\end{align*}
\]
Phase 2 – Dilation of Nets

- Tight placement is always **not** possible
  - **Push** and **Pull** from other nets are ignored
- Net \( N \) has no incident reconvergences \( \Rightarrow \) no dilation
- \( N \) has incident reconvergences \( \Rightarrow \) other nets pull the cells away \( \Rightarrow \) dilation
- Net dilation is based on reconvergences on its immediate neighborhood: source, sinks, fanin

- For any node that is an origin of any reconvergence \( R_{x^*}, \) let the **out-weight** be
  \[
  RO(x) = \frac{\sum RO_{x^*}}{\sum RN_{y^*}}
  \]
  the average length of all out-bound reconvergences
- For all node that is a destination of any reconvergence \( R_{y^*}, \) let the **in-weight** be
  \[
  RI(y) = \frac{\sum RI_{y^*}}{\sum RN_{y^*}}
  \]
  the average length of all in-bound reconvergences
Dilation Factor

- **Raw weight of a node** $x$ is $RW'(x) = RI(x) + RO(x)$

- **Flip-Flops have many incident reconvergences**, hence an adjustment w.r.t. LUT-size $k$

\[
RW(x) = \begin{cases} 
\log_k RW'(x); & \text{if } x \text{ is a FF-node} \\
RW'(x); & \text{otherwise}
\end{cases}
\]

- **Dilation on a net** $N$ **with** $v_N$ **as its source is**

\[
R'(N) = \left( \sum_{x \in \text{fanin}(v_N)} RW(x) + \sum_{y \in \text{fanout}(v_N)} RW(y) + RW(v_N) \right) / t_N
\]

- **Similar to flip-flops, IO-Nets have large weights**, hence an empirical value for IO-Nets

\[
R(N) = \begin{cases} 
3 / \sqrt{2}; & \text{if } N \text{ is a IO-Net} \\
R'(N); & \text{otherwise}
\end{cases}
\]
Phase 3 – Uniform Distribution

- Let \( p \) be the position in which \( Shape \) vector has the maximum value.

- The nodes in this level are so many in number that they are expected to be uniformly distributed in the layout.
  - The nodes in this level are not connected to each other.
  - They are however strongly connected to the other nodes.

- If a node has more than one such cell, the net will dilate more.
Spread Due To Uniform Distribution

- \( SP \) = Set of nodes in the peak level
- \( N \) = Minimum required FPGA size

- Construct a hypothetical grid which has only one cell from \( SP \)

- The hypothetical grid size is related to the FPGA dimension as

\[
G = \frac{N}{\sqrt{\sqrt{SP}}}
\]

- If a net \( N \) has some nodes in \( SP \) then the span must respect the uniform distribution assumption

- The uniformity factor is calculated as

\[
U = \sqrt{|SP \cap \text{fanout}(N)|} \cdot G
\]
Bounding Box Span of a Net

- The bounding box span of the net $N$ depends on
  - $L$ – the minimum span of the net
  - $R(N)$ – the dilation of the net due to reconvergences
  - $U$ – the uniformity factor

- The horizontal span of the net $N$ is
  \[ HSpan(N) = \begin{cases} 
  \max(L, U) & \text{if } R(N) < 1 \\
  \max(L \times R(N), U) & \text{if } R(N) > 1
  \end{cases} \]

- The vertical span of the net is same as $HSpan$

- The total span of the net is
  \[ Span(N) = HSpan(N) + VSpan(N) = 2 \cdot HSpan(N) \]
<table>
<thead>
<tr>
<th>Circuit</th>
<th>Total Error (%)</th>
<th>I/O Error (%)</th>
<th>#Nets</th>
<th>Nets</th>
<th>Total Error w/o $R_N$ (%)</th>
<th>I/O Error w/o $R_N$ (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>alu4</td>
<td>2.95</td>
<td>-2.24</td>
<td>1299</td>
<td>223</td>
<td>48.45</td>
<td>48.48</td>
</tr>
<tr>
<td>apex2</td>
<td>-19.68</td>
<td>29.10</td>
<td>1616</td>
<td>262</td>
<td>53.15</td>
<td>66.58</td>
</tr>
<tr>
<td>bigkey</td>
<td>-21.45</td>
<td>34.37</td>
<td>1699</td>
<td>8</td>
<td>-21.42</td>
<td>47.48</td>
</tr>
<tr>
<td>dsip</td>
<td>11.08</td>
<td>-4.07</td>
<td>1367</td>
<td>3</td>
<td>10.94</td>
<td>-3.46</td>
</tr>
<tr>
<td>misex3</td>
<td>4.49</td>
<td>6.77</td>
<td>1170</td>
<td>227</td>
<td>54.11</td>
<td>55.69</td>
</tr>
<tr>
<td>pdc</td>
<td>13.44</td>
<td>-14.7</td>
<td>4051</td>
<td>524</td>
<td>65.01</td>
<td>45.93</td>
</tr>
<tr>
<td>s298</td>
<td>-5.15</td>
<td>0</td>
<td>1837</td>
<td>94</td>
<td>20.86</td>
<td>38.16</td>
</tr>
<tr>
<td>s38417</td>
<td>-32.42</td>
<td>-4.84</td>
<td>5955</td>
<td>451</td>
<td>37.82</td>
<td>50.57</td>
</tr>
<tr>
<td>seq</td>
<td>4.66</td>
<td>13.24</td>
<td>1522</td>
<td>228</td>
<td>56.11</td>
<td>58.8</td>
</tr>
<tr>
<td>spla</td>
<td>-0.68</td>
<td>-14.34</td>
<td>3329</td>
<td>361</td>
<td>58.29</td>
<td>46.09</td>
</tr>
<tr>
<td>Totals</td>
<td></td>
<td></td>
<td>23845</td>
<td>2194</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Avg.</td>
<td>11.6%</td>
<td>12.4%</td>
<td></td>
<td></td>
<td>42.61</td>
<td>46.13</td>
</tr>
</tbody>
</table>
Routing Demand Estimation

- We use RISA to calculate number of routing elements needed
  - An empirical technique based on wirelength of nets with various terminal sizes
  - The routing demand is based on two factors
    - $q$ - an empirical factor dependent on $t_N$
    - Bounding Box Sizes

- The actual routing demand for a net $N$ is calculated as

\[
D_h^N = q \times \frac{1}{H\text{Span}(N)}; D_v^N = q \times \frac{1}{V\text{Span}(N)}
\]
Definitions

- \( C \) = Number of Logic Blocks
- \( nIO \) = Number of I/O blocks
- If the circuit is placed in the smallest possible device, its width (also height) is given as
  \[
  N = \max\left(\frac{nIO}{4}, \sqrt{C}\right)
  \]
- \( TD \) = Total Number of Routing Elements Needed
  \[
  TD = \sum_{N} D_{h}^{N} + D_{v}^{N}
  \]
Channel Width Estimation for Pad Unconstrained Circuits

- Pad-Unconstrained Circuits
  - $N = \sqrt{C}$
  - $TD$ routing elements are uniformly distributed across the device
  - Channel width $W$ is calculated as $W = \frac{TD}{C} = \frac{TD}{N \times N}$
Pad-Constrained Circuits

- $N = nIO/4$

- Assume that all the logic blocks are placed in the center – consistent with modern placers

- However, $TD$ routing elements should be distributed across the whole device

- Channel width $W$ is calculated as

$$W = \frac{TD}{C} \times \frac{\sqrt{C}}{N} = \frac{TD}{\sqrt{C \times N}}$$
Experimentation - Other Methods Compared

- RISA [ICCAD 94, DAC 2002]
  - Post-placement technique
  - Add up demands for different sites in the layout and find the maximum channel width

- Yang et al. [ISPD 2001]
  - Rentian Method
  - Extended for FPGAs in [8]
  - Recursive partitioning of circuit and layout
  - Worst-case congestion analysis on the boundaries
## Results for Channel Width Estimation

<table>
<thead>
<tr>
<th>Circuit</th>
<th>( W_{\text{VPR}} )</th>
<th>( W )</th>
<th>( T )</th>
<th>( W_{\text{RISA}} )</th>
<th>( T_{\text{RISA}} )</th>
<th>( W_{\text{RENT}} )</th>
<th>( T_{\text{RENT}} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>alu4</td>
<td>11</td>
<td>11.322</td>
<td>0.139</td>
<td>13.506</td>
<td>0.012</td>
<td>10.717</td>
<td>1.54</td>
</tr>
<tr>
<td>apex2</td>
<td>12</td>
<td>12.981</td>
<td>0.234</td>
<td>14.911</td>
<td>0.022</td>
<td>21.322</td>
<td>2.49</td>
</tr>
<tr>
<td>bigkey</td>
<td>9</td>
<td>5.7</td>
<td>0.385</td>
<td>12.105</td>
<td>0.027</td>
<td>4.761</td>
<td>2.6</td>
</tr>
<tr>
<td>dsip</td>
<td>7</td>
<td>6.452</td>
<td>0.298</td>
<td>9.699</td>
<td>0.019</td>
<td>4.176</td>
<td>1.97</td>
</tr>
<tr>
<td>misex3</td>
<td>11</td>
<td>11.252</td>
<td>0.156</td>
<td>13.649</td>
<td>0.015</td>
<td>11.682</td>
<td>1.37</td>
</tr>
<tr>
<td>pdc</td>
<td>16</td>
<td>11.991</td>
<td>5.034</td>
<td>20.418</td>
<td>0.103</td>
<td>19.067</td>
<td>14.61</td>
</tr>
<tr>
<td>s298</td>
<td>8</td>
<td>8.27</td>
<td>1.06</td>
<td>9.963</td>
<td>0.010</td>
<td>14.15</td>
<td>2.33</td>
</tr>
<tr>
<td>s38417</td>
<td>8</td>
<td>10.544</td>
<td>8.712</td>
<td>13.192</td>
<td>0.035</td>
<td>14.963</td>
<td>21.43</td>
</tr>
<tr>
<td>seq</td>
<td>12</td>
<td>11.826</td>
<td>0.208</td>
<td>14.53</td>
<td>0.019</td>
<td>13.468</td>
<td>2.07</td>
</tr>
<tr>
<td>spla</td>
<td>15</td>
<td>11.593</td>
<td>3.148</td>
<td>17.663</td>
<td>0.06</td>
<td>35.149</td>
<td>9.42</td>
</tr>
<tr>
<td>Total</td>
<td>109</td>
<td>102.29</td>
<td>19.38</td>
<td>139.63</td>
<td>0.324</td>
<td>149.455</td>
<td>59.86</td>
</tr>
<tr>
<td>Error</td>
<td>-</td>
<td></td>
<td>6.1%</td>
<td></td>
<td>28.1%</td>
<td></td>
<td>37.1%</td>
</tr>
</tbody>
</table>
Summary

- Identified some important circuit characteristics which dictate placement
  - **Push** and **Pull** from reconvergences stretch wires
  - Reconvergences capture more than the local neighborhood of cells
  - 30% more accuracy with reconvergences factored in

- Bounding box prediction is accurate within 11.6% of post-placement lengths
- Channel widths are predicted within 6% of post-route results
Illustration of Bounding Box Calculation

**Phase 1**  
Sinks are tightly placed around the source node

**Phase 2**  
Node A has high reconvergence weight. Pulled away from the net

**Phase 3**  
Node D and Node C are in peak level and hence should spread out

Node A has high reconvergence weight. Pulled away from the net
Overview of Our Methodology

Reconvergence Analysis
- Perform reconvergence analysis within different sequential levels
- Assign weights to nodes based on reconvergences

Bounding Box Estimation
- For every net
  - Calculate the minimum possible bounding box
  - Find dilation factor using reconvergence weights
  - Uniformly distribute *peak nodes*
  - Calculate the actual span using these 3 factors

Track Width Estimation
- Calculate the number of routing elements required using RISA for every net
- Calculate the total number of routing elements
- Distribute this routing demand evenly in the layout to obtain maximum channel width