Modeling Flip Flop Delay Dependencies in Timing Analysis

Amit M Jain, David Blaauw
University of Michigan, Ann Arbor
{amjain,blaauw}@umich.edu
Motivation

Static Timing Analyzers
- Used to verify behavior of large digital circuits
- Core engine is circuit optimization tools

Current Designs
- Flip-flop delay is getting increasingly significant.
- Flip-Flop delay is a function of data arrival time

Our Work
- Formulation and solution method for finding the optimal clock period which incorporates these dependencies.
Outline

- Traditional Approach
  - Delay Model
  - Problem Formulation and Properties
  - Proposed Solution Approaches
  - Short Path Problem Formulation
- Results
- Conclusion
Objective
- Find the clock period of the circuit.

Problem Formulation
\[ T_{c,ij} = T_{\text{setup}} + T_{\text{ff},i} + T_{\text{Logic,ij,\text{long}}} + T_{\text{skew}} \]
\[ T_c = \max\{T_{c,ij}\} \quad \forall (i, j \leq n) \]

Assumptions
- Flip-Flop delay is constant.
- Setup Time of a Flip-Flop is fixed.

Implications
- No interdependence amongst stages.
- Pessimism involved.
Pessimism in Traditional Approach

\[ T_{	ext{logic}1} \rightarrow T_{	ext{setup}1} \rightarrow T_{	ext{ff1}} \rightarrow \text{CLK} \]

\[ T_{	ext{logic}2} \rightarrow T_{	ext{setup}2} \rightarrow T_{	ext{ff2}} \rightarrow \text{CLK} \]

\[ T_{c1} > T_{c2}, \quad T_c = T_{c1} \]

\[ T_{c1} = T_{c2}, \quad T_c = T_{c1} \]
Outline

- Traditional Approach
- Delay Model
- Problem Formulation and Properties
- Proposed Solution Approaches
- Short Path Problem Formulation
- Results
- Conclusion
Delay Model.

Flip-Flop Delay Vs Data Arrival Time

\[ y = A \cdot \exp(B \cdot x) + C \]

Quality of Fit

<table>
<thead>
<tr>
<th>Flip-Flops in typical library</th>
<th>Output Loads</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>R-Square</td>
</tr>
<tr>
<td>DFF1</td>
<td>.1pf</td>
<td>.99476</td>
</tr>
<tr>
<td></td>
<td>.05pf</td>
<td>.99483</td>
</tr>
<tr>
<td>DFF2</td>
<td>.1pf</td>
<td>.99913</td>
</tr>
<tr>
<td></td>
<td>.05pf</td>
<td>.99919</td>
</tr>
<tr>
<td>DFF4</td>
<td>.1pf</td>
<td>.98554</td>
</tr>
<tr>
<td></td>
<td>.05pf</td>
<td>.98858</td>
</tr>
<tr>
<td>DFF4</td>
<td>.1pf</td>
<td>.99435</td>
</tr>
<tr>
<td></td>
<td>.05pf</td>
<td>.99619</td>
</tr>
</tbody>
</table>
Outline

- Traditional Approach
- Delay Model
- **Problem Formulation and Properties**
- Proposed Solution Approaches
- Short Path Problem Formulation
- Results
- Conclusion
Problem Formulation and Properties

\[ T_{c,ij} = T_{ff,i} + T_{Logic,ij,\text{long}} + T_{skew} + T_{Diff,ij} \quad \forall (i, j \leq n) \]

\[ T_{Diff,j} = \min\{T_{Diff,ij}\} \quad \forall (i \leq n) \]

\[ T_{ff,j} = A \cdot \exp\{B \cdot T_{Diff,j}\} + C \]

\[ T_c = \max\{T_{c,ij}\} \quad \forall (i, j \leq n) \]

Objectives: \( \min\{T_c\} \)

Non-Linear Optimization Problem
- Convex
- Exponential Functions retain Convexity
- Maximum Function retains Convexity
- Minimum Function retains Convexity
Properties

Balanced Delay

\[ T_{c} = T_{\text{Diff},1} + A \cdot \exp\{B \cdot T_{\text{Diff},1}\} + C + T_{\text{Logic1}} + T_{\text{skew}} \]

\[ \frac{\partial T_{c}}{\partial T_{\text{Diff},1}} = 0 \quad \text{and} \quad \frac{\partial T_{\text{ff},1}}{\partial T_{\text{Diff},1}} = -1 \]

New Criteria for defining \( T_{\text{setup}} \)

\[ T_{\text{logic1}} = T_{\text{logic2}} \]
Properties

Un-Balanced Delay

\[ \prod_{i} \left| \frac{\partial T_{ff,i}}{\partial T_{Diff,i}} \right| = 1 \]

\[ T_{\text{logic1}} < T_{\text{logic2}} \]
Properties

Balanced Delay

Unbalanced Delay
Outline

- Traditional Approach
- Delay Model
- Problem Formulation and Properties
- **Proposed Solution Methods**
  - Sequential Quadratic Programming
  - Rectilinear Manhattan Decent
  - Satisfiability Approach
- Short Path Problem Formulation
- Results
- Conclusion
Proposed Solution Methods

- **Sequential Quadratic Programming**
  - General MinMax Optimization Problem.
  - Uses a function from Commercial Non-linear Optimizer.
  - Integration with current static analyzers not that easy.
  - Run Time very high.
  - Used to check the correctness of results from other two approaches.
Proposed Solution Methods

**Rectilinear Manhattan Decent**

- Optimization function is a Convex Function.
- Repeatedly decreases the $T_{\text{diff}}$ variable to which $T_c$ is most sensitive. (The Critical Path)

**Algorithm**

- Calculate the critical path.
- Decrease the corresponding $T_{\text{Diff}}$ by a small step and update the timing.
- If there ever is an increase in $T_c$ then the previous $T_c$ was optimal else iterate.
Proposed Solution Methods

Satisfiability Approach

- Performs a bounded binary search on the optimization space.
- Ease in integration with current static timing analyzers.
- Very Acceptable runtime.

Assign $T_{\text{upper}}$ & $T_{\text{lower}}$

$T_c = \text{mean } \{T_{\text{upper}}, T_{\text{lower}}\}$

$T_{\text{Diff}} = T_c - (T_{\text{ff}} + T_{\text{logic}})$

$T_{\text{upper}} = T_c$

$T_{\text{lower}} = T_c$

Propagate

Check $T_{\text{Diff}}$

If $T_{\text{Diff}}$ is +ve:

$T_{\text{upper}} = T_{\text{lower}}$

Result = $T_c$

If $T_{\text{Diff}}$ is -ve:

$T_{\text{upper}} = T_{\text{lower}}$

Result = $T_c$
Satisfiability Approach

Optimization Function $T_c$

$T_{ff,1} + T_{logic,1}$

$T_{Diff,1}$

$T_{ff,2} + T_{logic,2}$

$T_{Diff,2}$

$T_c$
Outline

- Traditional Approach
- Delay Model
- Problem Formulation and Properties
- Proposed Solution Methods
- **Short Path Problem Formulation**
- Results
- Conclusion
Short Path Problem Formulation.

Traditional a short path violation is reported under the following circumstances.

\[ T_{ff,i} + T_{Logic,ij,short} < T_{skew} + T_{hold,i} \quad \forall (i, j \leq n) \]

As data arrival time gets closer to the clock edge the delay of the flip-flop increases.

This effect can in turn be used to remove some the short time constraints.
Short Path Problem Formulation.

\[ T_{c,ij} = T_{\text{Diff,ij}} + T_{ff,i} + T_{\text{Logic,ij,short}} + T_{\text{skew}} \quad \forall (i, j \leq n) \]

\[ T_{\text{Diff},j} = \max \{ T_{\text{Diff,ij}} \} \quad \forall (i, j \leq n) \]

\[ T_{ff,j} = A \cdot \exp \{ B \cdot T_{\text{Diff},j} \} + C \quad \forall (i, j \leq n) \]

\[ T_{ff,i} + T_{\text{Logic,ij,short}} < T_{\text{skew}} + T_{\text{hold},i} \quad \exists (i, j \leq n) \]

\[ T_c = \max \{ T_{c,ij} \} \quad \forall (i, j \leq n) \]

Objective: \( \min \{ T_c \} \)

\[ T_c - \delta \quad \text{Minimum Frequency of operation} \]

- Valid Operation
- \( f_{\text{short}} \)
- \( f_{\text{long}} \)
- Frequency axis
Outline

- Traditional Approach
- Delay Model
- Problem Formulation and Properties
- Proposed Solution Methods
- Short Path Problem Formulation
- Results
- Conclusion
## Results

### Comparing runtimes and Iterations of three Approaches

<table>
<thead>
<tr>
<th>Benchmarks</th>
<th># FF</th>
<th>Runtime in seconds.</th>
<th>Iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>SQP</td>
<td>RMD</td>
</tr>
<tr>
<td>s953</td>
<td>29</td>
<td>3.72</td>
<td>.411</td>
</tr>
<tr>
<td>s9234</td>
<td>228</td>
<td>207</td>
<td>4.39</td>
</tr>
<tr>
<td>s838</td>
<td>32</td>
<td>4.87</td>
<td>.14</td>
</tr>
<tr>
<td>s713</td>
<td>19</td>
<td>1.12</td>
<td>.10</td>
</tr>
<tr>
<td>s641</td>
<td>19</td>
<td>1.21</td>
<td>.091</td>
</tr>
<tr>
<td>s526</td>
<td>21</td>
<td>1.49</td>
<td>.1</td>
</tr>
<tr>
<td>s400</td>
<td>21</td>
<td>1.45</td>
<td>.16</td>
</tr>
<tr>
<td>s349</td>
<td>15</td>
<td>.75</td>
<td>.02</td>
</tr>
<tr>
<td>s382</td>
<td>21</td>
<td>1.25</td>
<td>.11</td>
</tr>
<tr>
<td>s27</td>
<td>3</td>
<td>.13</td>
<td>.11</td>
</tr>
<tr>
<td>s13207</td>
<td>669</td>
<td>1222</td>
<td>24.6</td>
</tr>
<tr>
<td>S15850_1</td>
<td>534</td>
<td>1021</td>
<td>19.9</td>
</tr>
</tbody>
</table>

- **Satisfiability Approach has the least runtime of all methods**
- **All the methods approach the same answer**
## Results

Clock period calculated by modeling flip-flop delay dependency and Traditional method using various Tsetup times.

<table>
<thead>
<tr>
<th>Benchmark</th>
<th>New Approach ps</th>
<th>5% T_{setup} Time ps</th>
<th>d/dx = -1 T_{setup} Time ps</th>
<th>Constant T_{ff} T_{setup} Time ps</th>
</tr>
</thead>
<tbody>
<tr>
<td>s953</td>
<td>2274.2</td>
<td>2828</td>
<td>2825</td>
<td>2854</td>
</tr>
<tr>
<td>s9234</td>
<td>4484.2</td>
<td>4538</td>
<td>4525</td>
<td>4564</td>
</tr>
<tr>
<td>s838</td>
<td>6531.2</td>
<td>6588</td>
<td>6585</td>
<td>6614</td>
</tr>
<tr>
<td>s713</td>
<td>5404.2</td>
<td>5358</td>
<td>5355</td>
<td>6614</td>
</tr>
<tr>
<td>s641</td>
<td>5404.2</td>
<td>5358</td>
<td>5355</td>
<td>5384</td>
</tr>
<tr>
<td>s526</td>
<td>2635.2</td>
<td>2688</td>
<td>2685</td>
<td>2734</td>
</tr>
<tr>
<td>s400</td>
<td>2254.2</td>
<td>2308</td>
<td>2305</td>
<td>2334</td>
</tr>
<tr>
<td>s349</td>
<td>3264.2</td>
<td>3318</td>
<td>3315</td>
<td>3344</td>
</tr>
<tr>
<td>s382</td>
<td>2594.2</td>
<td>2648</td>
<td>2645</td>
<td>2674</td>
</tr>
<tr>
<td>s27</td>
<td>1192.2</td>
<td>1198</td>
<td>1195</td>
<td>1224</td>
</tr>
<tr>
<td>s13207</td>
<td>4994.2</td>
<td>5048</td>
<td>5045</td>
<td>5074</td>
</tr>
<tr>
<td>S15850_1</td>
<td>6344.2</td>
<td>6448</td>
<td>6445</td>
<td>6474</td>
</tr>
</tbody>
</table>

Consistent Improvement of 50 – 60 ps
## Results

**Comparison with HSPICE simulation.**

<table>
<thead>
<tr>
<th>Circuit</th>
<th>Load Type</th>
<th>HSPICE model</th>
<th>Proposed Approach</th>
<th>Mismatch</th>
</tr>
</thead>
<tbody>
<tr>
<td>Circuit_A</td>
<td>Balanced</td>
<td>2270ps</td>
<td>2265ps</td>
<td>-5ps</td>
</tr>
<tr>
<td>Circuit_B</td>
<td>Unbalanced</td>
<td>2310ps</td>
<td>2313ps</td>
<td>3ps</td>
</tr>
<tr>
<td>Circuit_C</td>
<td>Balanced</td>
<td>692ps</td>
<td>681ps</td>
<td>-11ps</td>
</tr>
<tr>
<td>Circuit_D</td>
<td>Unbalanced</td>
<td>2290ps</td>
<td>2291ps</td>
<td>1ps</td>
</tr>
</tbody>
</table>

**Average Deviation 5ps**

**Maximum Deviation 11ps**
Conclusion

**Proposed Approach**

- Integration with current STAs
- Acceptable runtimes.

**Tested on ISCAS ’89 Benchmarks**

**Traditional methods overestimates the clock period as much as 50-60ps**
Questions?