CSCI 4120 Computer-Aided Design of VLSI Circuits

Project

Proposal Presentation: April 10, 2024 2:30-4:30pm
Report and Program: May 5, 2024 11:59pm
Final Presentation: May 7, 2024 10:30am-12:30pm

Assume that we have a very regular top-level clock tree that delivers a clock signal from the clock source to a set of "clock taps" distributed across the floorplan with exact zero skew and very low latency, we want to assign each clock pin to one of the taps. Each tap with a set of pins assigned to it forms a clock tree synthesis problem. The problem here is about assigning and routing pins to taps. The routing region is just a simple grid graph with integral coordinates (x, y) where $x = 0 \ldots p$ and $y = 0 \ldots p$ and each edge in the grid graph has a constant capacity *C*, ie., can accommodate *C* wires at maximum.

Input: a set of clock pins and a set of clock source taps

Output: assignment and routing of each pin to one of the taps

Objective: Minimize the total wirelength of all the clock trees and satisfy the constraint

Constraint:

A loading limit is given which specifies a maximum number of pins that can be driven by a tap

For most of the cases, there will be 4 - 32 taps and 5K - 10K pins.

Objective Function:

The cost function *c* will be calculated as:

$$\left(\max_i d_i - \min_j d_j\right) \times numTaps + \sum_{i=1}^{numTaps} TreeLength(TAP_i)$$

Where d_i is the delay (path length) for pin p_i, $TreeLength(TAP_i)$ is the total length of the rectilinear tree connecting all the pins assigned to TAP_i, $numTAP$ is the total number of taps. Note that the cost is the lower the better.

Evaluation:

Proposal Presenation : 10%
Report: 10%
Final Presentation: 10%
Program: 70%

Five benchmarks will be released. There will be two hidden benchmarks. This 70% scores will be given as follows:

(i) Success to connect each pin to a tap without open or short: 30% (for undergraduate teams only)

(ii) Achieve (i) and satisfy the constraint: 60%

The remaining 10% will be given to the top five teams (getting 10%, 8%, 6%, 4% and 2%), minimizing the objective function, and there can at most be one postgraduate team in these top five.
An undergraduate team can be formed by one, two or three students, while a postgraduate team can only have one student.

***Input Format**:*

```
MAX_RUNTIME time
MAX_LOAD load
GRID_SIZE p
CAPACITY C

PINS numPins
  PIN index0 x0 y0
  PIN index1 x1 y1
  …
END PINS

TAPS numTaps
  TAP index0 x0 y0
  TAP index1 x1 y1
  …
END TAPS
```

Input Formal Description:

```
MAX_RUNTIME time          % Maximum runtime
MAX_LOAD load             % Maximum number of pins driven by a tap
GRID_SIZE p               % Size of grid graph
CAPACITY C                % Capacity of each edge in the grid graph

PINS num                  % Number of clock pins
                            followed by the definitions of clock pins
PIN index x y             % index specifies the pin index
                            x and y specify the pin position
TAPS num                  % Number of clock taps
                            followed by the definitions of clock taps
TAP index x y             % index specifies the tap index
                            x and y specify the tap position
```

***Output Format**:*

```
TAP index0
  PINS num
    PIN index0
    PIN index1
    …

ROUTING numEdges
  EDGE x0 y0 x0’ y0’
  EDGE x0 y0 x0’ y0’
  …

TAP index1
  PINS num
    PIN index0
    PIN index1
    …

ROUTING numEdges
  EDGE x0 y0 x0’ y0’
  EDGE x0 y0 x0’ y0’
  …
```

Output Formal Description:

```
TAP index                  % The tap index
PINS num                   % Number of pins connected by the tap
PIN index                  % The pin index
ROUTING numEdges           % Number of edges in the routing solution
EDGE x y x’ y’             % (x, y) and (x’, y’) specify the two ends of an edge.
                             Overlapped edges will be merged automatically.
```

Command:

The binary should be named “cts”, and should support the following command.

(C/C++) ./cts --input test.in --output test.out
(Python) python3 cts.py --input test.in --output test.out

Evaluation:

python3 eval.py --input test.in --output test.out --plot true

Sample Testcase:

A sample testcase (test0.in) and an example output (test0.out) have been provided. The example output gives the solution shown below. It has a total tree length of 25. The max delay and min delay are 8 and 4 respectively. The final cost is 33.