Chip Design
Summer term 2009

Prof. Dr. Jens Vygen
Dr. Stephan Held

Exercises 11

1. *Dynamic programming buffer insertion.*

   (a) *Combined buffer insertion and layer assignment.* Given a repeater tree topology A rooted in r, the task is to compute two functions $f : V(A) \to L \cup \{0\}$ and $g : E(A) \to \{1, \ldots, z_{\max}\}$, where L is the repeater library and $\{1, \ldots, z_{\max}\}$ the set of available routing layers (we neglect preference directions of routing layers).

   The wire capacitance and resistance of an edge $e = (v, w) \in E(A)$ assigned to layer $g(e)$ are given by $c_{g(e)}||v-w||_1$ and $r_{g(e)}||v-w||_1$, where $c_z, r_z \in \mathbb{R}_+$ ($z \in \{1, \ldots, z_{\max}\}$) are layer-dependent constants. For books $l \in L$ there is an input pin capacitance C_l and a monotonically increasing function $d_l : \mathbb{R} \to \mathbb{R}$, mapping a load capacitance $C \in \mathbb{R}$ to a delay $d_l(C)$. Wire delays are computed according to the Elmore formula. Show that, if all segments of a net must be assigned to a common layer, a solution maximizing the worst slack at r can be computed in $O(|L| \cdot z_{\max} \cdot |V(A)|^2)$ time. [5 points]

   (b) *Area efficient buffer insertion.* In addition to the formulation in the lecture notes (i.e. $z_{\max} = 1$), an area consumption $area(l) \in \mathbb{R}_{\geq 0}$ is given for each buffer $l \in L \cup \{0\}$, where $area(0) = 0$. Now, instead of maximizing the worst slack at the root, the area consumption is to be minimized preserving timing constraints, i.e. given a target $\underline{\text{slack}}$ at the root, find an assignment $f : V(A) \to L \cup \{0\}$ that minimizes the total area consumption

   $$\sum_{v \in V(G)} area(f(v))$$

   such that $\text{slack}(r) \geq \underline{\text{slack}}$. What is the running time of your algorithm? Determine a pruning criterion to reduce the number of solutions at a vertex $v \in V(G)$ similar to the one from the lecture notes. [5 points]

2. Pruning solution candidates in BonnClock:

   - Prove the following theorem for the number of solution candidates of the BonnClock algorithm:

Considering only solutions candidates $sol_v \in SOL_v$ with maximal arrival time windows among those with equal output slew, the number of candidates per cluster v is bounded by:

$$|\mathcal{S}||L| \sum_{\text{v}' \in \text{succ(v)}} |SOL_{v'}|,$$

where $\mathcal{S}$ is the finite set of slew values,
L is the set of available inverters, and
$SOL_{v'}$ is the set of solution condidates in $v' \in V(T)$. [5 points]

- Show that the set of all solution candidates in v with output slew s and maximal arrival time windows can be computed in $O(S \log S)$ time, where S is the number of all successor solution candidates with input slew s. [5 points]

The deadline for submitting solutions is July 14 before the lecture (12:15).