

Advances in Global Routing

Jens Vygen

Research Institute for Discrete Mathematics
University of Bonn

VLSI Routing: Task

Given a netlist, i.e. a family of sets of pins (terminals), routing and timing constraints, we look for

- a feasible routing
- such that all timing constraints are met
- and the (estimated) power consumption is minimized.

Simplified view: find vertex-disjoint Steiner trees connecting given terminal sets in a 3-dimensional grid graph.

Order of magnitude: 4 million Steiner trees in a graph with 20 billion vertices
(50000 · 50000 · 8)

⇒ Even linear-time algorithms are too slow!

Overall Routing Flow

Efficient detailed routing:

- route nets sequentially, mainly by shortest path algorithms
- goal-oriented shortest path algorithms
- label intervals rather than single points
- restrict path search to small areas

First step: Global Routing (compute areas)

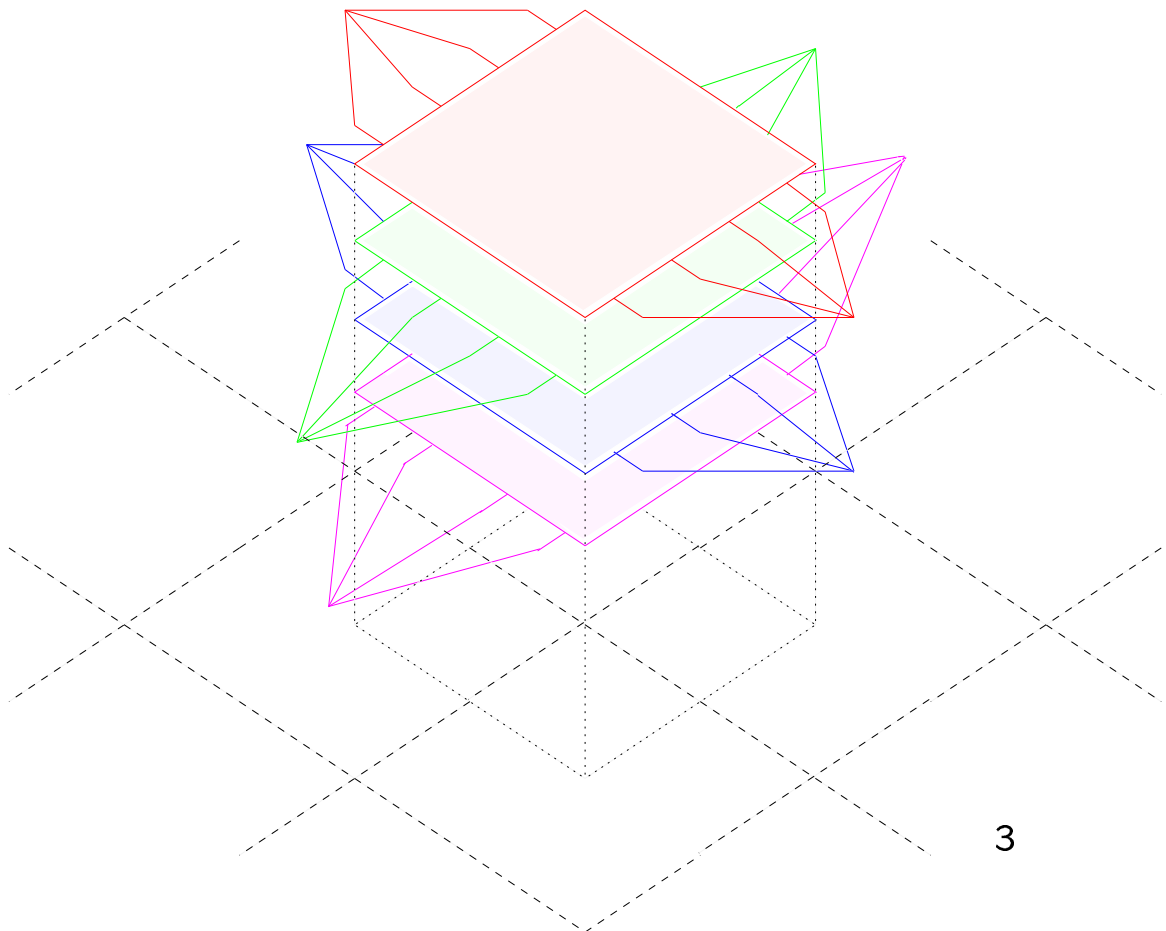
- contract regions of approx. 50x50 points to a single vertex
- compute capacities of edges between adjacent regions
- pack Steiner trees with respect to these edge capacities
- define a detailed routing area for each net according to its Steiner tree

Capacity Estimation

First route very short nets (within one region or two adjacent regions).

Then consider each pair of adjacent regions. Assume that planes are mainly used in preferred wiring direction, alternatingly horizontal and vertical.

Consider the following multicommodity flow problem:



Capacity Estimation (2)

Apply a very fast multicommodity flow heuristic, exploiting the structure of the instances.
(Müller [2002])

Each augmenting path requires only $O(k)$ constant-time bit pattern operations, where k is the number of edges orthogonal to the preferred wiring direction.

Heuristic finds a feasible integral multicommodity flow solution whose value is approx. 90% of the (weak) max-flow upper bound.

Complete chip with 300 million paths in 15 minutes (Goldberg-Tarjan runs 1 month)

Global Routing is Hard

Restriction: Edge-Disjoint Paths Problem.

Given a pair of graphs (G, H) , find a family $(P_h)_{h \in E(H)}$ of edge-disjoint paths in G such that $P_h + h$ is a circuit for each $h \in E(H)$.

NP-complete even if

- G is a rectangle (Raghavan [1986])
- G is a rectangle, and we allow shortest paths only (Vygen [1994])
- G is a rectangle, and $G+H$ is Eulerian (Marx [2002])
- G is series-parallel (Nishizeki, Vygen, Zhou [2001])
- G is directed and planar, H consists of two sets of parallel edges (Müller [2002])

Positive Results

- There is a combinatorial fully polynomial approximation scheme for the Multicommodity Flow Problem (Sharokhi, Matula [1990], Leighton, Makedon, Plotkin, Stein, Tardos, Tragoudas [1991], Plotkin, Shmoys, Tardos [1991], Radzik [1995], Young [1995], Grigoriadis, Khachiyan [1996], Garg, Könemann [1998], Fleischer [2000])
- If edges have sufficient capacity, randomized rounding can be applied to get an integral solution violating capacity constraints only slightly (Raghavan, Thompson [1987,1991], Raghavan [1988])
- This can be applied to Steiner trees instead of paths and works efficiently for large global routing instances (Albrecht [2001])

Problem: Does not take timing constraints and power consumption into account.

Main Design Objectives in Routing (1)

meet all timing constraints

The delay on each path must not exceed its bound. A path can be viewed as a sequence of nets. The delay of a net depends on its electrical capacitance.

- first assume delay-optimal Steiner trees for all nets
- distribute slack optimally (Albrecht, Korte, Schietke, Vygen [2000], Held [2001]) to all nets for which sufficient slack is available. For these nets the slack defines a maximum tolerable capacitance
- call the remaining nets (with no or insufficient slack assigned) critical
- compute weights and a bound on the weighted sum of capacitances for each path containing a critical net

Main Design Objectives in Routing (2)

minimize power consumption

- active power consumption roughly proportional to the electrical capacitance, weighted by switching activity
- leakage power and capacitance of active components not influenced by routing.
- capacitance of nets depends on length, width, plane, and existence of neighbour wires

minimize cost

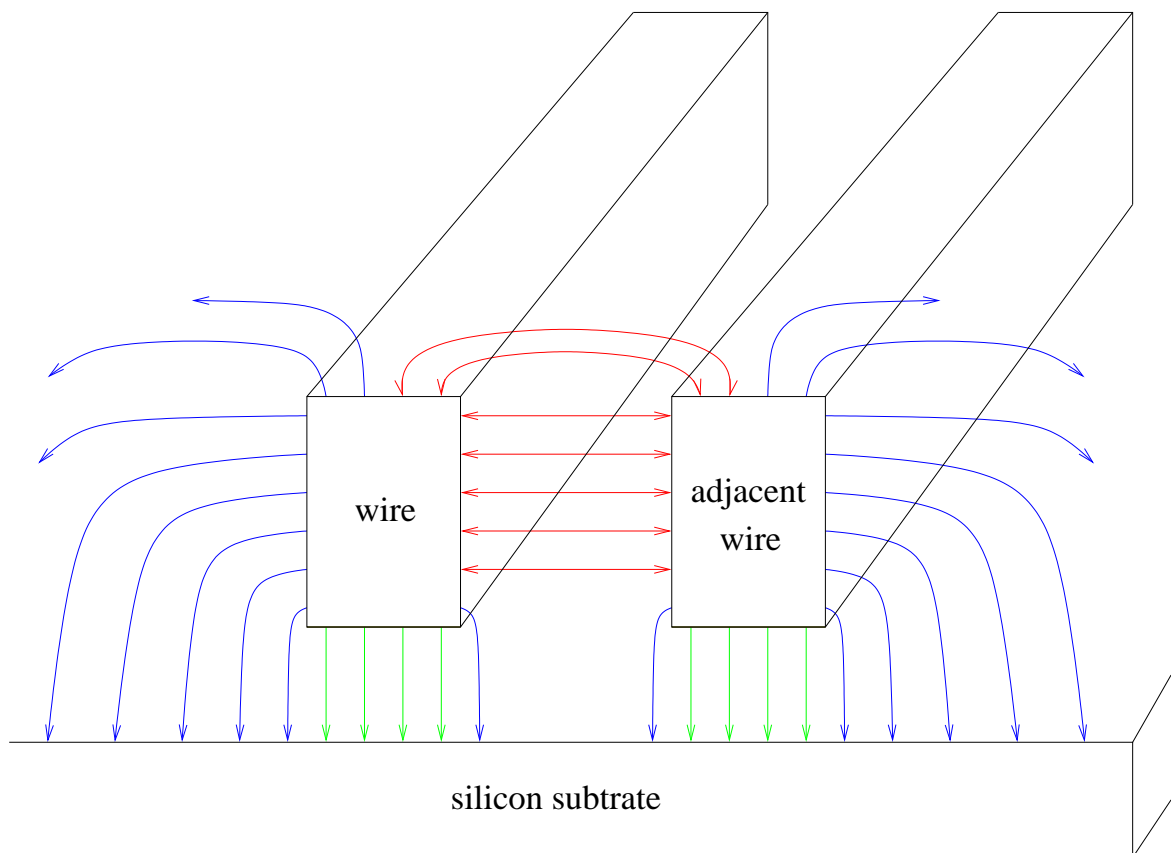
- minimize number of masks (number of routing planes), maximize yield (spreading), minimize design effort

Capacitance Estimation

area capacitance (parallel plate capacitor) – proportional to length times width

fringing capacitance – proportional to length

coupling capacitance – proportional to length if adjacent wire exists

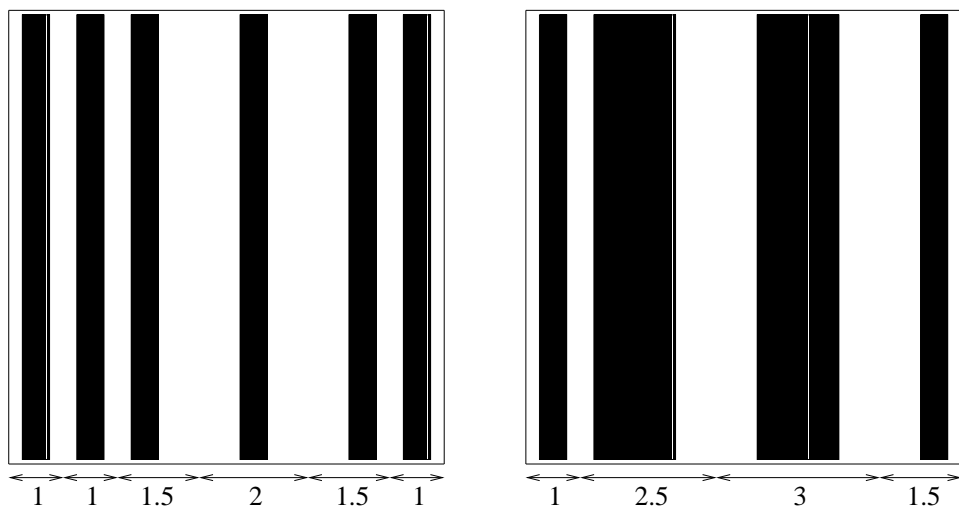


Modeling Coupling Capacitance

Assume linear dependence on distance to adjacent wire between the following bounds:

- minimum distance \Rightarrow coupling cap. $\frac{1}{2}v(e)$
- minimum distance plus 1 \Rightarrow coupling cap. 0

Example: global routing edge e of capacity $u(e) = 8$, with two global routing solutions:



- **Left:** six unit width wires use 6–12 channels. coupling capacitance $v(e)$ times $1, 1, \frac{1}{2}, 0, \frac{1}{2}, 1$
- **Right:** two unit width wires and two double width wires use 6–10 channels. coupling capacitance $v(e)$ times $1, \frac{1}{2}, 0, \frac{1}{2}$

Global Routing Problem: Instance

- an undirected graph G with capacities $u: E(G) \rightarrow \mathbb{Z}_+$ and lengths $l: E(G) \rightarrow \mathbb{R}$
- a family \mathcal{N} of nets and a set \mathcal{Y}_N of Steiner trees for each $N \in \mathcal{N}$
(Assumption: we can find a $Y \in \mathcal{Y}_N$ with $\sum_{e \in E(Y)} \psi(e)$ (almost) minimum fast for arbitrary $\psi: E(G) \rightarrow \mathbb{R}_+$)
- a width $w_{N,e} \geq 1$ for $N \in \mathcal{N}, e \in E(G)$
- maximum coupling capacitance $v: E(G) \rightarrow \mathbb{R}_+$ (per wire along edge)
- a family \mathcal{M} of subsets of \mathcal{N} , where $\mathcal{N} \in \mathcal{M}$, with bounds $U: \mathcal{M} \rightarrow \mathbb{R}_+$ and weights $c_{M,N} \in \mathbb{R}_+$ for $N \in M \in \mathcal{M}$

Global Routing Problem: Task

Find a Steiner tree $Y_N \in \mathcal{Y}_N$ and numbers $0 \leq y(e, N) \leq 1$, $e \in Y_N$, for each $N \in \mathcal{N}$, such that

$$\sum_{N \in \mathcal{N}: e \in E(Y_N)} (w_{N,e} + 1 - y(e, N)) \leq u(e) \quad \text{for } e \in E(G),$$

$$\sum_{N \in M} c_{M,N} \sum_{e \in E(Y_N)} (l(e) + v(e)y(e, N)) \leq U(M) \quad \text{for } M \in \mathcal{M},$$

and

$$\sum_{N \in \mathcal{N}} c_{N,N} \sum_{e \in E(Y_N)} (l(e) + v(e)y(e, N))$$

is minimum.

LP-Relaxation

min λ subject to:

$$\sum_{N \in \mathcal{N}} \left(\sum_{Y \in \mathcal{Y}_N: e \in E(Y)} (w_{N,e} + 1)x_{N,Y} - y_{e,N} \right) \leq \lambda u(e) \quad \text{for } e \in E(G)$$

$$\sum_{M \in \mathcal{M}} c_{M,N} \left(\sum_{Y \in \mathcal{Y}_N} \sum_{e \in E(Y)} l(e)x_{N,Y} + \sum_{e \in E(G)} v(e)y_{e,N} \right) \leq \lambda U(M) \quad \text{for } M \in \mathcal{M}$$

$$\sum_{Y \in \mathcal{Y}_N} x_{N,Y} = 1 \quad \text{for } N \in \mathcal{N}$$

$$y_{e,N} \leq \sum_{Y \in \mathcal{Y}_N: e \in E(Y)} x_{N,Y} \quad \text{for } e \in E(G), N \in \mathcal{N}$$

$$y_{e,N} \geq 0 \quad \text{for } e \in E(G), N \in \mathcal{N}$$

$$x_{N,Y} \geq 0 \quad \text{for } N \in \mathcal{N}, Y \in \mathcal{Y}_N$$

Dual LP

max $\sum_{N \in \mathcal{N}} z_N$ subject to:

$$\sum_{e \in E(G)} u(e)\omega_e + \sum_{M \in \mathcal{M}} U(M)\mu_M = 1$$

$$\sum_{e \in E(Y)} \left(l(e) \sum_{M \in \mathcal{M}: N \in M} c_{M,N}\mu_M + (w_{N,e} + 1)\omega_e - \chi_{N,e} \right) \geq z_N$$

for $N \in \mathcal{N}, Y \in \mathcal{Y}_N$

$$\omega_e \leq v(e) \sum_{M \in \mathcal{M}: N \in M} c_{M,N}\mu_M + \chi_{e,N}$$

for $e \in E(G), N \in \mathcal{N}$

$$\omega_e \geq 0$$

for $e \in E(G)$

$$\chi_{e,N} \geq 0$$

for $e \in E(G), N \in \mathcal{N}$

$$\mu_M \geq 0$$

for $M \in \mathcal{M}$

Proposition: Let $\omega \in \mathbb{R}_+^{E(G)}$ and $\mu \in \mathbb{R}_+^{\mathcal{M}}$. Let

$$\psi_{N,e} := \min_{\delta \in \{0,1\}} ((l(e) + \delta v(e)) \sum_{M \in \mathcal{M}: N \in M} c_{M,N}\mu_M + (w_{N,e} + 1 - \delta)\omega_e)$$

Then $\frac{\sum_{N \in \mathcal{N}} \min_{Y \in \mathcal{Y}_N} \psi_{N,e}}{\sum_{e \in E(G)} u(e)\omega_e + \sum_{M \in \mathcal{M}} U(M)\mu_M}$

is a lower bound for the optimum LP value.

Algorithm

Set $\omega_e := \frac{1}{u(e)} \forall e$ and $\mu_M := \frac{1}{U(M)} \forall M$

Set $x := 0, y := 0$ and $Y_N := \emptyset \forall N$

For $p := 1$ to t **do**: **For** $N \in \mathcal{N}$ **do**:

Let $Y_N \in \mathcal{Y}_N$ with $z_N := \sum_{e \in E(Y_N)} \psi_{N,e}$ minimum

Set $x_{N,Y_N} := x_{N,Y_N} + 1$

For $e \in E(Y_N)$ **do**:

If $v(e) \sum_{M \in \mathcal{M}: N \in M} c_{M,N} \mu_M < \omega_e$ **then** $\delta_e := 1$
else $\delta_e := 0$

Set $y_{e,N} := y_{e,N} + \delta_e$

Set $\omega_e := \omega_e e^{\frac{w_{N,e} + 1 - \delta_e}{u(e)}}$

For $M \in \mathcal{M}$ with $N \in M$ **do**:

$\mu_M := \mu_M e^{\epsilon c_{M,N} \frac{l(e) + \delta_e v(e)}{U(M)}}$

Set $x_{N,Y} := \frac{1}{t} x_{N,Y}$ for $N \in \mathcal{N}, Y \in \mathcal{Y}_N$

Set $y_{e,N} := \frac{1}{t} y_{e,N}$ for $e \in E(G), N \in \mathcal{N}$

Results and Extensions

This is a **fully polynomial approximation scheme** for the primal-dual pair of LPs

Enhancements:

- Compute new Steiner tree for net N only if previous one is longer than $(1 + \epsilon_1)z_N$, where z_N is a continuously updated lower bound.
- If a new Steiner tree has to be computed, a $(1 + \epsilon_2)$ -optimal one suffices.

Theorem: Let λ^* be the optimum LP value and $t\epsilon\lambda^* > \log(m + |\mathcal{M}|)$. Then the algorithm computes feasible primal and dual solutions, whose values differ by at most a factor

$$\frac{\epsilon(1 + \epsilon)(1 + \epsilon_1)(1 + \epsilon_2)}{\epsilon(1 - \epsilon(1 + \epsilon)(1 + \epsilon_1)(1 + \epsilon_2)\lambda^*) \left(1 - \frac{\log(m + |\mathcal{M}|)}{t\epsilon\lambda^*}\right)}$$

By choosing $\epsilon, \epsilon_1, \epsilon_2, t$ appropriately, we get a $(1 + \epsilon_0)$ -optimal solution in $\frac{2 \ln(m + |\mathcal{M}|)}{\epsilon_0^2}$ iterations, for any $\epsilon_0 > 0$.

Randomized Rounding

Let (x, y, λ) be a fractional solution to the primal LP. Compute a rounded solution $(\hat{x}, \hat{y}, \hat{\lambda})$ as follows:

- choose $Y \in \mathcal{Y}_N$ as Y_N with probability $x_{N,Y}$ (independently for all $N \in \mathcal{N}$); then set $\hat{x}_{N,Y_N} := 1$ and $\hat{x}_{N,Y} := 0$ for $Y \in \mathcal{Y}_N \setminus \{Y_N\}$.
- Set $\hat{y}_{N,e} := \frac{y_{N,e}}{\sum_{Y \in \mathcal{Y}_N: e \in E(Y)} x_{N,Y}}$ if $e \in E(Y_N)$ and $\hat{y}_{N,e} := 0$ otherwise.
- Choose $\hat{\lambda}$ minimum possible such that $(\hat{x}, \hat{y}, \hat{\lambda})$ is a feasible solution to the primal LP.

Let $\Lambda \leq \frac{U(M)}{c(M,N) \sum_{e \in E(Y)} (l(e) + v(e))}$ for $N \in M \in \mathcal{M}$ and $Y \in \mathcal{Y}_N$, and $\Lambda \leq \frac{u(e)}{w_{N,e} + 1}$ for $N \in \mathcal{N}$. Moreover, suppose that $|\mathcal{M}| + |E(G)| < e^{\lambda \Lambda}$.

Then $\hat{\lambda} \leq \lambda \left(1 + (e - 1) \sqrt{\frac{\ln(|\mathcal{M}| + |E(G)|)}{\lambda \Lambda}} \right)$.

Remarks and Conclusions

In practice, results are much better than theoretical performance guarantees. Usually 10–20 iterations suffice.

Only few upper bounds are violated; these are corrected easily by *ripup-and-reroute*.

Detailed Routing can realize the solution well, due to excellent capacity estimations.

Small integrality gap and approximate dual solution implies that an infeasibility proof can be found for most infeasible instances.

First global routing algorithm to take into account coupling, timing, and power consumption directly. Provably near-optimal.

Connection to Traffic Flows

The global routing problem is equivalent to routing traffic flow

- with hard capacity bounds on edges (streets)
- without capacity bounds on vertices
- in a static setting (flow continuously repeated over time)
- with bounds on weighted sums of travel times
- and with the following transit time model: the transit time along an edge (latency) is constant up to $x\%$ congestion and grows linearly between $x\%$ and 100% congestion

Algorithm is equivalent to selfish routing (under fair conditions) but with different edge costs (exponential dependence on congestion)

Open question: For what latency functions does fair selfish routing work?