THE INTERCONNECTION TOPOLOGY LAYOUT PROBLEM I.E. THE ORTHOGONAL GRID GRAPH DRAWING PROBLEM

Prof. Tiziana Calamoneri
Network Algorithms
A.y. 2018/19

THE THOMPSON'S MODEL

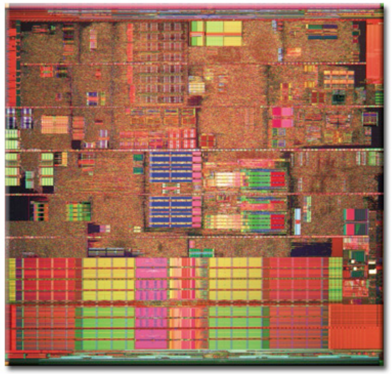
THOMPSON'S MODEL (1)

- The interconnection topology *layout problem* arises from the problem of producing efficient VLSI (Very Large Scale Integration) layouts on a silicon board.
- It was born in the '40s, but it got a significative interest only relatively recently, when the technology has allowed to layout circuits in two and three dimensions at reasonably low price.

THOMPSON'S MODEL (2)

Two exemples of VLSI circuits:

Intel 2004



Intel 2013

THOMPSON'S MODEL (3)

Model the circuit as a graph (nodes = ports, switches, etc. and edges = wires).

There is a tight relation between the VLSI layout and the graph drawing.

Drawing Γ of a graph G: it is a function mapping each node v in a distinct point $\Gamma(v)$, and each edge (u,v) in an open Jordan curve $\Gamma(u,v)$ not crossing any point that is the mapping of a node, starting in $\Gamma(u)$ and arriving in $\Gamma(v)$.

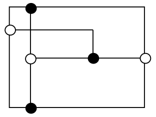
The VLSI technology production imposes many constraints; in particular, we have to keep into account the following:

- …

THOMPSON'S MODEL (4)

- … the device pressing the connections can only approximate slanting lines by tiny horizontal and vertical segments (⇒orthogonal drawing);

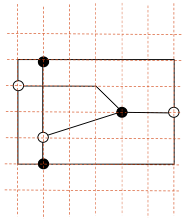
Orthogonal drawing: drawing of a graph where edges are represented as broken lines whose segments are horizontal or vertical (parallel to the coordinate axes)



THOMPSON'S MODEL (5)

- In order to avoid interference, it is necessary to keep wires far enough (⇒grid drawing);

Grid drawing: drawing of a graph so that all nodes, crosses and bends of the edges are put on grid points
(scaling property - resolution)



THOMPSON'S MODEL (6)

- Wires cannot cross; in order to avoid crossings, it is possible to route the crossing wires on the two separate sides of the board, introducing small "holes" trepassing the board from a side to the other one; the number of such holes must be small, as their realization is rather expensive (⇒crossing number minimization)
- …

THOMPSON'S MODEL (7)

- The silicon is very expensive; so the layout must have small area (⇒area minimization).
- Wires should not be too long, as the propagation delay is proportional to their length; in case of layered topology, wires in the same layer should have (approximately) the same length, so to avoid synchronization problems

(⇒edge length minimization).

THOMPSON'S MODEL (8)

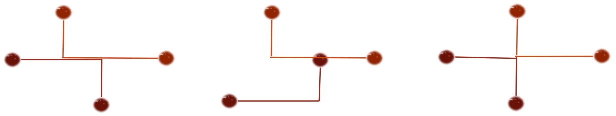
In 1980 **Thompson** introduced a **model** that is consistent with all the mentioned constraints:

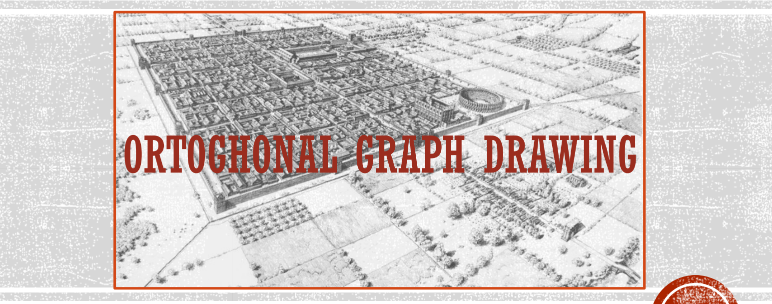
the layout of a topology G is a plane representation on a bunch of unit distance horizontal and vertical traces that maps:

…

THOMPSON'S MODEL (9)

- nodes of G in the intersecion points of the traces,
- edges of G in disjoint paths constituted by horizontal and vertical segments on traces; such paths cannot cross nodes that are not their extremes and they can cross each other only in corrispondence of trace intersection points;
- Overlappings (edge-edge) are not allowed
- Node-edge crosses are not allowed
- "knock-knees" are not allowed





ORTOGHONAL GRAPH DRAWING

ORTHOGONAL GRID DRAWING (1)

- Def. An *orthogonal grid drawing* of a graph *G=(V,E)* is a bijection mapping:
  - nodes $v \in V$ on plane points $\Gamma(v)$ at integer coordinates
  - edges $(v,w) \in E$ on not overlapping paths so that the images of their extremes $\Gamma(v)$ and $\Gamma(w)$ are connected by the corresponding paths.
  - These paths are constituted by horizontal and vertical segments; the possible bends have integer coordinates
- Obs. only graphs with degree ≤ 4 can be correctly drawn.

ORTHOGONAL GRID DRAWING (2)

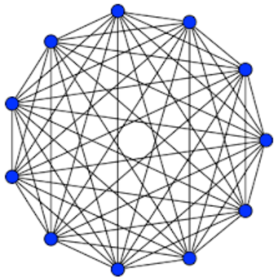
- So, the interconnection topology layout is an orthogonal grid drawing of the corresponding graph with the aim of minimizing the area, the number of crossings and the wire length.
- There is a huge literature in the GRAPH DRAWING area…
- Shall we use the known algorithms for orthogonal grid drawing in order to solve the layout problem?

ORTHOGONAL GRID DRAWING (3)

- No: these algorithms guarantee some bounds on the optimization functions that hold FOR EACH input graph having the required input hypotheses
- Interconnection topologies are very structured graphs (usually regular, symmetric, recursively built, …) and, exploiting these properties, it is possible to get better results.

ORTHOGONAL GRID DRAWING (4)

- Graph drawing algorithms get a graph in input and draw it on the plane.
- Layout algorithms are designed for a single special interconnection topology and so they get only its dimension in input.
- Obs. Improving an optimization function by “only” a constant factor is an important issue (especially the area): if a layout occupies ½ of the area of another one, it will cost the half!



COMPLETE GRAPH LAYOUT

COLLINEAR LAYOUT (1)

- The Thompson model [Thompson '79] requires that the wires coming out of each processing element are at most 4 (6 in 3D)

What if the degree is higher? (end of the '90s)

non-constant node degree model:

- a node of degree *d* occupies a square of side *Θ(d)* (here deg is *n-1*)
- the wires can run either horizontally or vertically along grid lines.

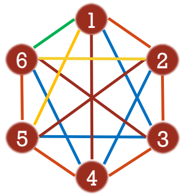
COLLINEAR LAYOUT (2)

- Layout proposed by Yeh and Parami [98]
- Collinear layout with area $n^4/4$ - optimal
- In a collinear layout all nodes are placed on the same line. Instead of computing its area, it is usual to count the number of necessary tracks.

COLLINEAR LAYOUT (3)

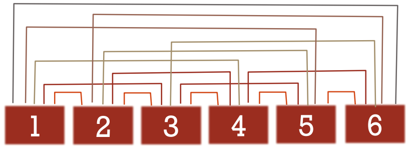
To obtain the collinear layout of the complete graph:

- let a link be *type-i* if it connects two nodes whose labels differ by *i;* so, the $n(n-1)/2$ links can be classified into types 1, 2, …, *n-1,* and there are *n-i* type-*i* links.
- …



COLLINEAR LAYOUT (4)

- …
- place the n nodes, labeled 1 through n, along a row;
- place the type-1 links in one track,
- place the type-2 links in two tracks, where links connecting odd nodes are put in one track and links connecting even nodes are put in the other one
- place the type-i links in min(i, n-i) tracks



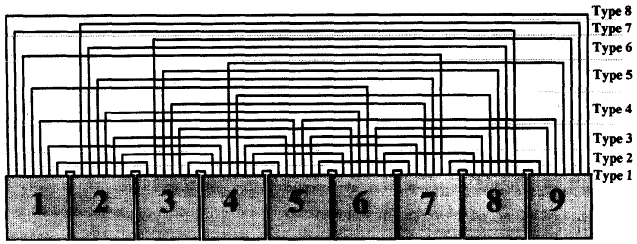
COLLINEAR LAYOUT (5)

- Total number of tracks in this layout:

substituting $j=n-i$

$$\sum_{i=1}^{n-1} \min(i, n-i) = \sum_{i=1}^{n/2} i + \sum_{i=\frac{n}{2}+1}^{n-1} (n-i) =$$

$$= \sum_{i=1}^{n/2} i + \sum_{j=1}^{\frac{n}{2}-1} j = \frac{1}{2}[\frac{n}{2}\left(\frac{n}{2}+1\right) + \frac{n}{2}\left(\frac{n}{2}-1\right)] = \frac{n^2}{4} + o(n^2)$$



COLLINEAR LAYOUT (6)

Def. : The bisection width of a network is the **minimum** number of edges one has to cut to disconnect the network into two **equally** sized sub-networks.

Property. The bisection width of the complete graph is $n^2/4+o(n^2)$.

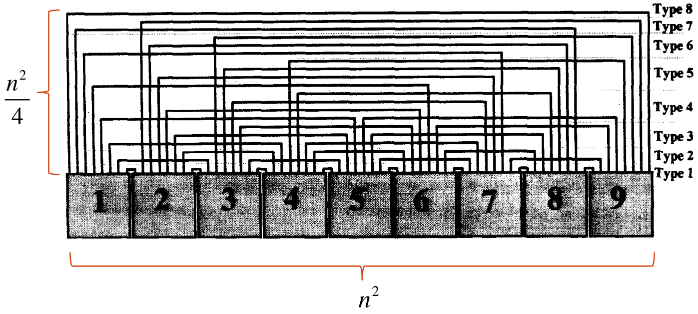
Th. A lower bound on the number of tracks in the collinear layout of a network is its bisection width *(to be proved later)*.

Cor. A lower bound on the number of tracks in the collinear layout of the complete graph is $n^2/4+o(n^2)$.

ORTHOGONAL LAYOUT (1)

- **Note.** The area of the collinear layout is:

$$\frac{n^4}{4}$$

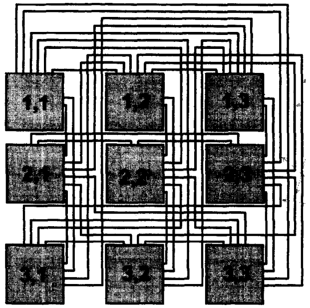


ORTHOGONAL LAYOUT (2)

- Although the collinear layout leads to the smallest possible number of tracks, layouts with smaller area can be obtained.
- An area efficient layout for complete graphs is based on the previous collinear layout.
- W.l.o.g. $n=m_1 \times m_2$, where m_1 and m_2 are $\Theta(\sqrt{n})$
- Each node can be labeled (i,j) with $i=1,\dots,m_1$ and $j=1,\dots,m_2$.

ORTHOGONAL LAYOUT (3)

- Put node (i,j) at coordinates (i,j) on an $m_1 \times m_2$ grid.
- Without entering into details:



$$Area = \frac{n^4}{16} + o(n^4)$$

ORTHOGONAL LAYOUT (4)

Th. A lower bound on the layout area of a network is the square of its bisection width.

Reminder. The bisection width of the complete graph is $n^2/4+o(n^2)$.

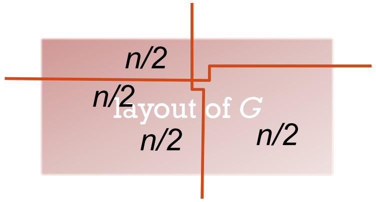
Cor. A lower bound on the layout area of the complete graph is $n^4/16+o(n^4)$.

ORTHOGONAL LAYOUT (5)

Let us prove the theorem:

Th. [Thompson '79] A lower bound on the layout area of a network is the square of its bisection width.

Proof. Suppose that the bisection width of a network G can be counted when partitioning its nodes in two sets of k and n-k nodes, respectively.



width at least as large as the bisection width…
the same holds for the height… ■

BUTTERFLY LAYOUT

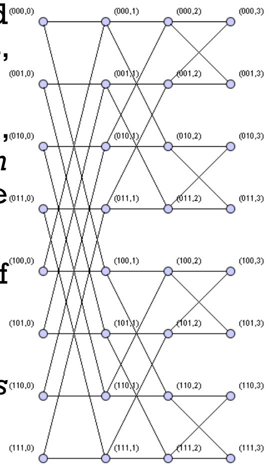
BUTTERFLY NETWORK (MEMORANDUM)

Def. (reminder) Let $N=2^n$ (and $n=\log N$); an *n-dimensional Butterfly* is a layered graph having $N\ (n+1)$ nodes ($n+1$ layers, with 2^n nodes each) and $2Nn$ edges.

The nodes are labeled with a pair (w, i), where i is the *layer* of the node and w is an n bit binary number indicating the row of the node.

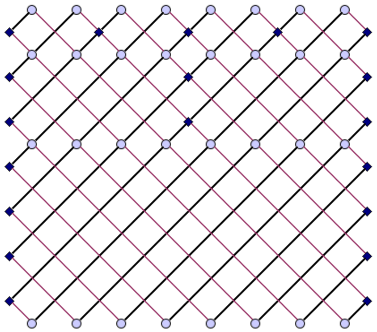
Two nodes (w, i) and (w', i') are adjacent iff $i'=i+1$ and:

- $w=w'$ (*straight edge*) or
- w e w' differ in exactly the i-th bit (*cross edge*).



WISE LAYOUT (1)

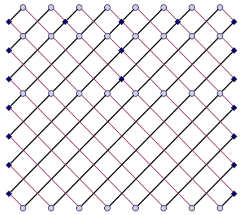
Layout proposed by D.S.Wise ['81]



He writes:

"This paper offers a result that can be described as a picture. […] The perceptive reader may stop here, since the remainder of this paper only describes it."

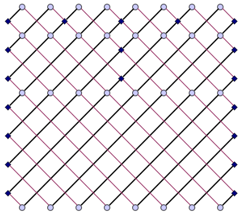
WISE LAYOUT (2)



This layout has a property that is very important in a layered topology:

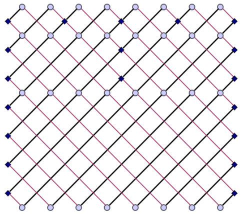
- All the wires in the same layer are of equal length.
- Nevertheless, this length grows exponentially up with the layer.

WISE LAYOUT (3)



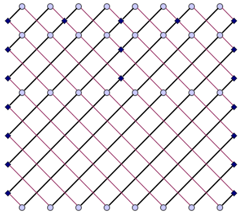
- The **longest** path length from any input to any output is linear in N (namely, $2(N-1)$).
- Indeed:
  - All the paths have the same length.
  - For the sake of simplicity, consider the path from the upper-left node to the lower-right node.
  - The length of this path coincides with the diagonal of the square having side $\sqrt{2}\,(N-1)$, so it is $2(N-1)$.

WISE LAYOUT (4)



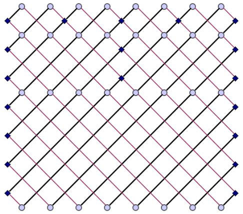
- The layout is performed on the two sides of the silicon board, so it can be considered a 2-layer layout; one layer is composed of all diagonal wires running "north-east" (from lower-left to upper-right) -red lines- and the other layer is composed of "north-west" wires (from lower-right to upper-left) -black lines.

WISE LAYOUT (5)



- PROs:
  - Good area : $\sqrt{2}\,(N-1) \times \sqrt{2}\,(N-1) = 2N^2+o(N^2)$
  - Same wire length on each layer; this is not true in every layout: in the classical drawing of the butterfly network, for example, the straight-edges on the last layer have unit length while the cross-edges on the same layer have linear length in the input size N; this is extremely bad, because synchronization of the information flow goes lost;
  - The input and output nodes lie on the boundary of the layout, and this can be required by some applications.

WISE LAYOUT (6)



- CONs:
  - "slanted" lines, so that the area of the layout is measured by a rectangle whose sides are not parallel to coordinate axes but lie at 45°; if we follow the standard definition of layout area, it becomes $2(N-1) \times 2(N-1) = 4N^2+o(N^2)$; indeed, the circumscribed square with sides parallel to the coordinate axes has side equal to to the length of the path from the upper-left node to the lower-right node, that is $2(N-1)$;
  - …

WISE LAYOUT (7)

- CONNs (cntd):
  - … it is a ‘cheating’ layout, indeed the “knock-knees” are not avoided but arranged in the layout thanks to some devices that have no null area and so enlarge the layout area.
  - The Wise layout “looks like” the usual representation. Nevertheless, in order to get the Wise layout from the usual representation, nodes must be permuted: