Principles of Distributed Computing
Roger Wattenhofer

8 Basic Network Topologies

(With thanks to Christian Scheideler, Johns Hopkins University, for a preliminary version of this text.)

In this chapter we will introduce some popular families of network topologies. Many of these topologies are used in classic parallel computers or telecommunication networks, or more recently in the emerging area of peer-to-peer computing.

The most basic network topologies used in practice are trees, cycles, grids and tori. Many other suggested networks are simply combinations or derivatives of these. The advantage of trees is that the path selection problem is very easy: for every source-destination pair there is only one possible simple path. However, since the root of a tree is usually a severe bottleneck, so-called *fat trees* have been used. These trees have the property that every edge connecting a node v to its parent u has a capacity that is equal to all leaves of the subtree routed at v. See Figure 1 for an example.

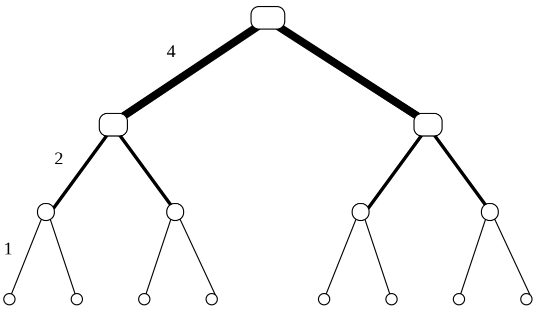


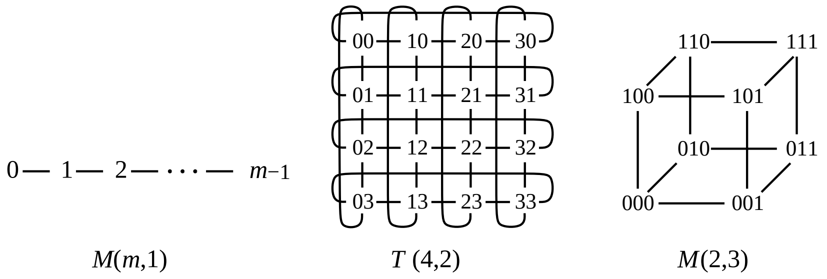
Figure 1: The structure of a fat tree.

Fat trees belong to a family of networks that require edges of non-uniform capacity to be efficient. Easier to build are networks with edges of uniform capacity. This is usually the case for grids and tori. Unless explicitly mentioned, we will treat all edges in the following to be of capacity 1. In the following, $[x]$ means the set $\{0, 1, \ldots, x-1\}$.

Definition 8.1 (Torus, Mesh) *Let $m, d \in \mathbb{N}$. The (m,d)-*mesh *$M(m,d)$ is a graph with node set $V = [m]^d$ and edge set*

$$E = \left\{ \{(a_{d-1} \ldots a_0), (b_{d-1} \ldots b_0)\} \mid a_i, b_i \in [m], \sum_{i=0}^{d-1} |a_i - b_i| = 1 \right\} .$$

*The (m,d)-*torus *$T(m,d)$ is a graph that consists of an (m,d)-mesh and additionally wrap-around edges from $(a_{d-1} \ldots a_{i+1}(m-1)\ a_{i-1} \ldots a_0)$ to $(a_{d-1} \ldots a_{i+1}\ 0\ a_{i-1} \ldots a_0)$ for all $i \in [d]$ and all $a_j \in [m]$ with $j \neq i$. $M(m,1)$ is also called a* line*, $T(m,1)$ a* cycle*, and $M(2,d) = T(2,d)$ a* d-dimensional hypercube.



$M(m,1)$ $T(4,2)$ $M(2,3)$

Figure 2: The structure of $M(m,1)$, $T(4,2)$, and $M(2,3)$.

Figure 2 presents a linear array, a torus, and a hypercube.

The hypercube is a very important class of networks, and many derivatives, the so-called *hypercubic networks*, have been suggested for it. Among these are the butterfly, cube-connected-cycles, shuffle-exchange, and de Bruijn graph. We start with the butterfly, which is basically a rolled out hypercube.

Definition 8.2 (Butterfly) *Let* $d \in \mathbb{N}$. *The* d-dimensional butterfly $BF(d)$ *is a graph with node set* $V = [d+1] \times [2]^d$ *and an edge set* $E = E_1 \cup E_2$ *with*

$$E_1 = \{\{(i,\alpha),(i+1,\alpha)\} \mid i \in [d],\ \alpha \in [2]^d\}$$

and

$$E_2 = \{\{(i,\alpha),(i+1,\beta)\} \mid i \in [d],\ \alpha,\beta \in [2]^d,\ \alpha \textit{ and } \beta \textit{ differ only at the } i\textit{th position}\}\ .$$

A node set $\{(i,\alpha) \mid \alpha \in [2]^d\}$ *is said to form* level i *of the butterfly. The* d-dimensional wrap-around butterfly *W-BF*(d) *is defined by taking the* $BF(d)$ *and identifying level* d *with level 0.*

Figure 3 shows the 3-dimensional butterfly $BF(3)$. The $BF(d)$ has $(d+1)2^d$ nodes, $2d \cdot 2^d$ edges and degree 4. It is not difficult to check that combining the node sets $\{(i,\alpha) \mid i \in [d]\}$ into a single node results in the hypercube.

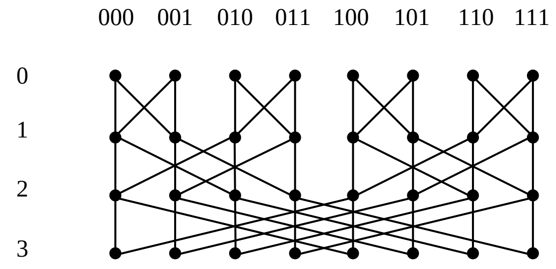


Figure 3: The structure of BF(3).

Next we define the cube-connected-cycles network. It only has a degree of 3 and it results from the hypercube by replacing the corners by cycles.

Definition 8.3 (Cube-Connected-Cycles) *Let* $d \in \mathbb{N}$. *The* cube-connected-cycles *network CCC(d) is a graph with node set* $V = \{(a,p) \mid a \in [2]^d, p \in [d]\}$ *and edge set*

$$\begin{aligned} E &= \Big\{\{(a,p),(a,(p+1) \text{ mod } d)\} \mid a \in [2]^d, p \in [d]\Big\} \\ &\cup \Big\{\{(a,p),(b,p)\} \mid a,b \in [2]^d, p \in [d], a = b \text{ except for } a_p\Big\} \end{aligned}$$

Two possible representations of a CCC can be found in Figure 4.

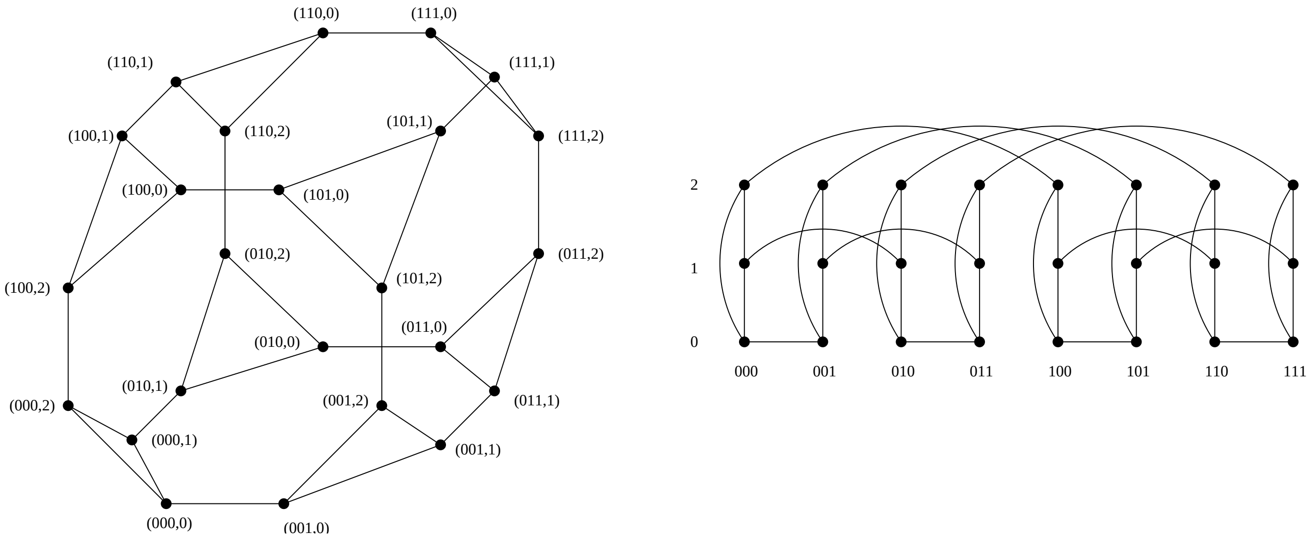


Figure 4: The structure of CCC(3).

The shuffle-exchange is yet another way of transforming the hypercubic interconnection structure into a constant degree network.

Definition 8.4 (Shuffle-Exchange) *Let* $d \in \mathbb{N}$. *The* d-dimensional shuffle-exchange $SE(d)$ *is defined as an undirected graph with node set* $V = [2]^d$ *and an edge set* $E = E_1 \cup E_2$ *with*

$$E_1 = \{\{(a_{d-1} \dots a_0), (a_{d-1} \dots \bar{a}_0)\} \mid (a_{d-1} \dots a_0) \in [2]^d,\ \bar{a}_0 = 1 - a_0\}$$

and

$$E_2 = \{\{(a_{d-1} \dots a_0), (a_0 a_{d-1} \dots a_1)\} \mid (a_{d-1} \dots a_0) \in [2]^d\} \ .$$

Figure 5 shows the 3- and 4-dimensional shuffle-exchange graph.

Definition 8.5 (DeBruijn) *The* b-ary DeBruijn graph of dimension d $DB(b,d)$ *is an undirected graph* $G = (V,E)$ *with node set* $V = \{v \in [b]^d\}$ *and edge set* E *that contains all edges* $\{v,w\}$ *with the property that* $w \in \{(x, v_{d-1}, \dots, v_1) :\ x \in [b]\}$, *where* $v = (v_{d-1}, \dots, v_0)$.

Two examples of a DeBruijn graph can be found in Figure 6.

One important goal in choosing a topology for a network is that it has a small diameter. The following theorem presents a lower bound for this.

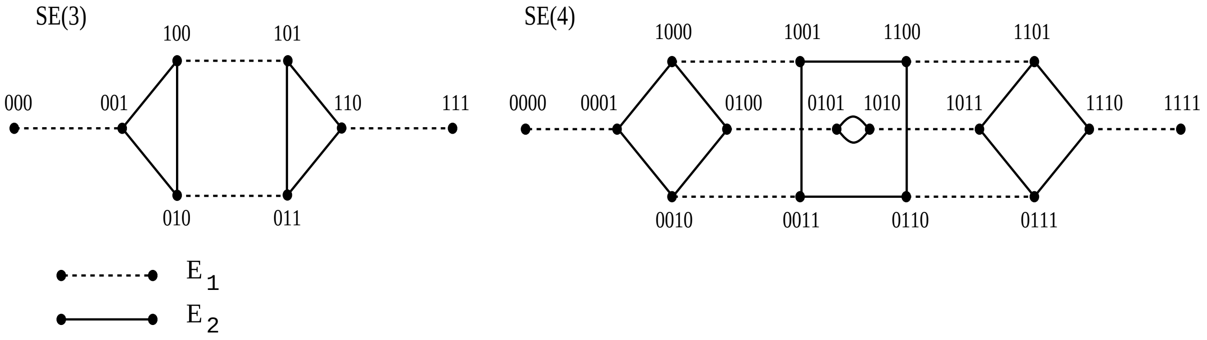


Figure 5: The structure of SE(3) and SE(4).

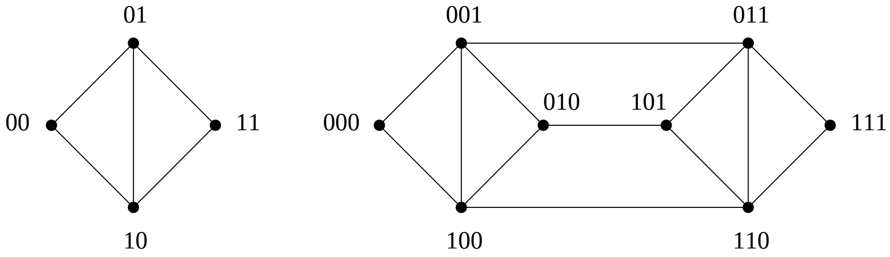


Figure 6: The structure of $DB(2,2)$ and $DB(2,3)$.

Theorem 8.6 *Every graph of maximum degree $d > 2$ and size n must have a diameter of at least $\lfloor(\log n)/(\log(d-1))\rfloor - 1$.*

Proof. Suppose we have a graph $G = (V, E)$ of maximum degree d and size n. Start from any node $v \in V$. In a first step at most d other nodes can be reached. In two steps at most $d \cdot (d-1)$ additional nodes can be reached. Thus, in general, in at most k steps at most

$$1 + \sum_{i=0}^{k-1} d \cdot (d-1)^i = 1 + d \cdot \frac{(d-1)^k - 1}{(d-1) - 1} \leq \frac{d \cdot (d-1)^k}{d-2}$$

nodes (including v) can be reached. This has to be at least n to ensure that v can reach all other nodes in V within k steps. Hence,

$$(d-1)^k \geq \frac{(d-2) \cdot n}{d} \quad \Leftrightarrow \quad k \geq \log_{d-1}((d-2) \cdot n/d)\ .$$

Since $\log_{d-1}((d-2)/d) > -2$ for all $d > 2$, this is true only if $k \geq \lfloor \log_{d-1} n \rfloor - 1$. ⊓