Assignments Week 43

Parallel Computing, DM818 (Fall 2017)

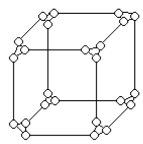
Department of Mathematics and Computer Science
University of Southern Denmark
Daniel Merkle

For preparing solutions for the following exercises you need to read the complete chapter 2 of the course book.

Exercise 1

Static Network Topology

We discussed the static network topology d-dimensional hypercube. A disadvantage of a hypercube-based architecture is that the degree grows with growing values for d. A modification of the hypercube is the so called cube-connected cycle graph (CCC-graph), that doesn't have that property.



Cube-connected cycle graph for $d = 3$

For a CCC-graph each node of a hypercube is replaced by a ring of d nodes. The set of nodes is defined by $V_{CCC} = V_H \times \{0, \ldots, d-1\}$. In a CCC-graph there is a edge between nodes (v, i) and (v', i') if

- $i = i'$ and v and v' differ only in the $i-$th bit, or
- $|i - i'| = 1$ and $v = v'$, or
- $i - i' = d$ and $v = v'$.

a) Determine the number of shortest paths between two nodes v_1 and v_2 in a standard hypercube. Hint: use the binary node labeling.

b) Determine the number of nodes and edges in a CCC-graph and of a hypercube with dimensionality d.

c) Determine the diameter of a CCC-graph for $d = 3$.

d) Determine the diameter of a CCC-graph for $d \geq 4$. Explain how you derived the formula and give an example for the case of $d = 4$. (A proof would be very complicated and is not necessary).

e) * Explain an optimal routing algorithm in a CCC-graph. (The solution for this question is quite complicated, we won't discuss the solution for this part in detail.)

Exercise 2 — Routing

Consider the routing of messages in a parallel computer that uses store-and-forward routing. In such a network, the cost of sending a single message of size m from P_{source} to $P_{\text{destination}}$ via a path of length d is $t_s + t_w \times d \times m$. An alternate way of sending a message of size m is as follows. The user breaks the message into k parts each of size m/k, and then sends these k distinct messages one by one from P_{source} to $P_{\text{destination}}$. For this new method, derive the expression for the time to transfer a message of size m to a node d hops away under the following two cases:

a) Assume that another message can be sent from P_{source} as soon as the previous message has reached the next node in the path.

b) Assume that another message can be sent from P_{source} only after the previous message has reached $P_{\text{destination}}$.

For each case, comment on the value of this expression as the value of k varies between 1 and m. Also, what is the optimal value of k if t_s is very large, or if $t_s = 0$?

Exercise 3 — Mesh of Trees

A mesh of trees is a network that imposes a tree interconnection on a grid of processing nodes. A $\sqrt{p} \times \sqrt{p}$ mesh of trees is constructed as follows. Starting with a $\sqrt{p} \times \sqrt{p}$ grid, a complete binary tree is imposed on each row of the grid. Then a complete binary tree is imposed on each column of the grid. Figure 1 illustrates the construction of a 4×4 mesh of trees. Assume that the nodes at intermediate levels are switching nodes. Determine the bisection width, diameter, and total number of switching nodes in a mesh of trees.

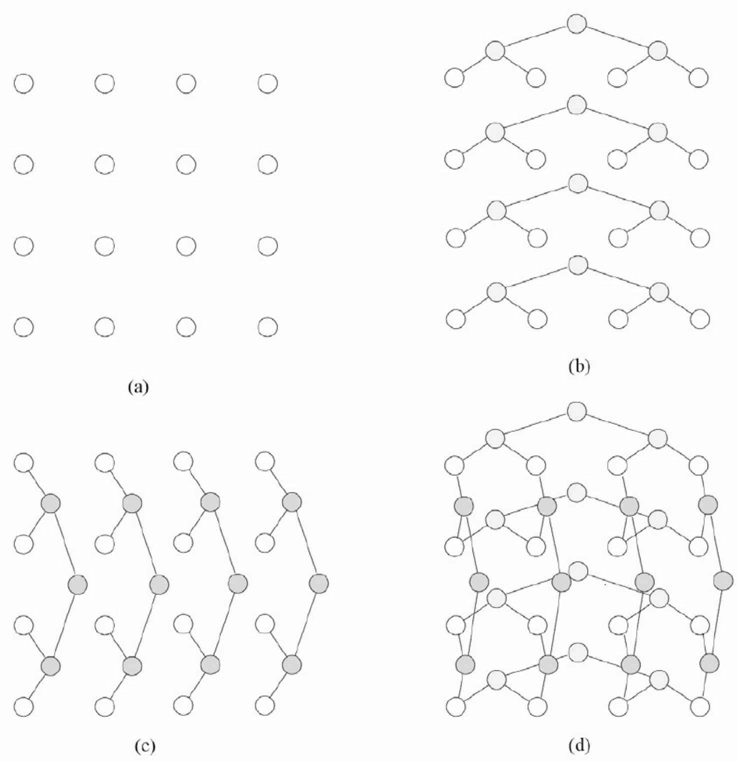


Figure 1: Mesh of Trees (Exercise 3)

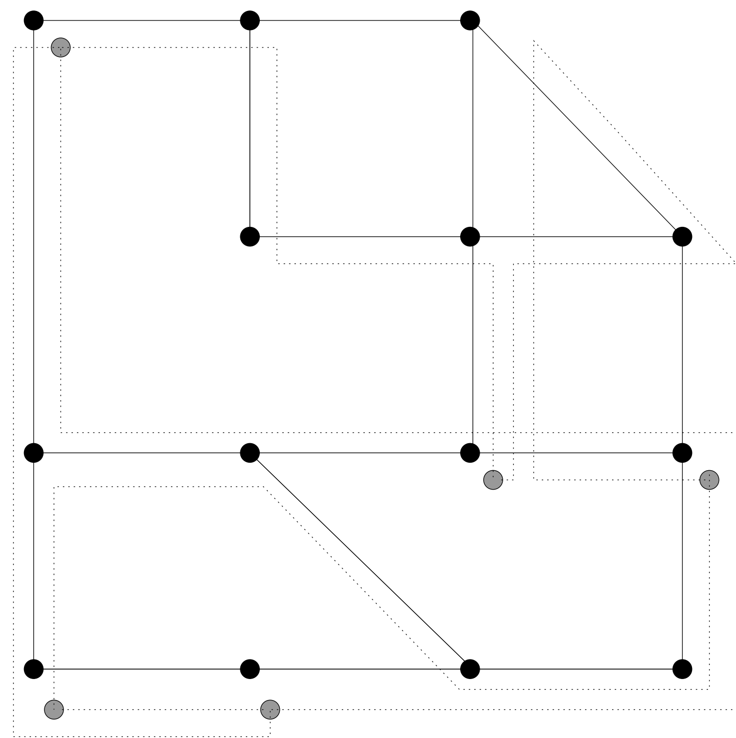


Figure 2: Embedding (Exercise 4)

Exercise 4 Embedding

Let $G = (V_G, E_G)$ and $H = (V_H, E_H)$ be two graphs. An embedding of G into H is a pair of mappings (f_V, f_E), where $f_V : V_G \to V_H$ is a mapping of node sets and $f_E : E_G \to E_H^*$ maps edges in G to paths in H. For an embedding it holds that for every edge $e = (u, v) \in E_G$ the path $f_E(e)$ is a path from $f_V(u)$ to $f_V(v)$ in H.

a) Formalize congestion and dilation using the definition above.

b) Determine congestion and dilation of the embedding of graph G (dotted edges, gray nodes) into graph H (solid edges, black nodes) in Figure 2.

c) Proof the following: There is no injective embedding with dilation 1 of a complete binary tree with depth $d > 2$ (and having therefore $2^d - 1$ nodes) into an hypercube of dimension d.