Dr. Christian Scheideler
Johns Hopkins University

Baltimore, Nov. 5 2003

Theory of Network Communication

Fall 2003

Solutions to Assignment 7

Problem 15 (4 points)**:**
The caching strategy for the d-dimensional hypercube is $O(d)$-competitive, with high probability.

Proof. In order to prove the above result we investigate two caching simulations. First, we consider the problem of simulating an optimal caching strategy in the hypercube H by the tree $T(H)$, and then we consider the problem of simulating a caching strategy in $T(H)$ by the hypercube H.
In order to simulate an optimal caching strategy in the hypercube by $T(H)$, we use the simple strategy that whenever a request is sent from node v to w in H, we send it along the unique path from the leaf representing v to the leaf representing w in $T(H)$.
In order to simulate a caching strategy in $T(H)$ by H, we use the following strategy: For every object x we embed $T_x(H)$ randomly into H (with the help of a pseudo-random hash function), and whenever a request is sent along an edge $\{v, w\}$ in $T_x(H)$, we send it along the path from $\pi(v)$ to $\pi(w)$ in H that results from adjusting the bits one-by-one from $\pi(v)$ to $\pi(w)$.
Let $C_{\mathrm{OPT}}(H)$ be the best possible congestion achievable in the hypercube H for the given application and let $C_{\mathrm{OPT}}(T(H))$ be the congestion caused in $T(H)$ when simulating this best possible strategy by $T(H)$. Then we get the following result.

Lemma 0.1 $C_{\mathrm{OPT}}(T(H)) \leq C_{\mathrm{OPT}}(H)$.

Proof. Let e denote an edge of $T(H)$ with bandwidth $b(e)$, and let $C(e)$ be the congestion caused by messages traversing e when using the simulation strategy above (i.e. the total number of messages traversing e is $b(e) \cdot C(e)$). Any message that crosses $e = \{v, w\}$ in $T(H)$ corresponds to a message that either has to leave or enter the sub-hypercube $H(w)$ in H. Since there are only $b(e)$ edges leaving $H(w)$, there must be an edge in $H(w)$ with congestion at least $(C(e) \cdot b(e))/b(e) = C(e)$. Thus, the maximum congestion over all edges in $T(H)$ is at most the maximum congestion over all edges in H, which proves the lemma. □

The next lemma gives an upper bound on the expected congestion caused by simulating the access tree strategy on H. In the following, let $C_T(e)$ denote the congestion caused at e for simulating $T(H)$.

Lemma 0.2 *For any edge e of H,* $\mathrm{E}[C_T(e)] = O(\log n \cdot C_{\mathrm{OPT}}(T(H)))$.

Proof. Let h denote the height of $T(H)$ and let $C_\ell(e)$ denote the congestion caused at e due to the simulation of edges between level $\ell+1$ and ℓ of $T(H)$, $0 \le \ell \le h-1$. We show that $\mathrm{E}[C_\ell(e)] = O(C_{\mathrm{OPT}}(T(H)))$ for all $\ell \in \{0,\ldots,h-1\}$, which yields the lemma as $h = O(\log n)$.
Consider some fixed level ℓ. Let v be a node of $T(H)$ on level ℓ such that $M(v)$ includes edge e. (If such a node does not exist then $\mathrm{E}[C_\ell(e)] = 0$.) Let v' be one of the two children of v. We bound the expected congestion on e due to the simulation of the tree edge $e_T = \{v, v'\}$ on level ℓ of $T(H)$.
Notice that any movement down a level in $T(H)$ the dimension of the sub-hypercube reduces by one. Hence, $H(v)$ is an $(d-\ell)$-dimensional hypercube. Suppose that the congestion at e_T is $C(e_T)$. Since e_T has a bandwidth of $\Theta(2^{d-\ell-1})$, this means that $\Theta(C(e_T)\cdot 2^{d-\ell-1})$ requests pass through e_T. Since every object x chooses a random place for v and v' in $H(v)$, the bit adjustment strategy moves through $e = \{u, w\}$ only if for the dimension i it represents, $(u_{i-1}\ldots u_0) = (v_{i-1}\ldots v_0)$ and $(u_{d-\ell}\ldots u_i) = (v'_{d-\ell}\ldots v'_i)$ when moving from v' to v, or $(u_{i-1}\ldots u_0) = (v'_{i-1}\ldots v'_0)$ and $(u_{d-\ell}\ldots u_i) = (v_{d-\ell}\ldots v_i)$ when moving from v to v'. This can only happen with probability $\Theta(1/2^{d-\ell})$. So for $\Theta(C(e_T)\cdot 2^{d-\ell-1})$ requests using e_T the expected congestion at e is

$$\Theta(C(e_T)\cdot 2^{d-\ell-1}) \cdot \Theta(1/2^{d-\ell}) = \Theta(C(e_T)) \ .$$

Thus, $\mathrm{E}[C_\ell(e)] = O(C(e_T)) = O(C_{\mathrm{OPT}}(T(H))$, which proves the lemma. □

Combining the two lemmata, we get $\mathrm{E}[C_T(e)] = O(d \cdot C_{\mathrm{OPT}}(H))$ for all edges e in H. One can also show that this bound holds with high probability, using the fact that for an object x with k write accesses the congestion in OPT caused for object x must be at least $k/4$ because all of these requests must update x at the same, static location. □