Supplementary file for: Signature Searching in a Networked Collection of Files

Zhongwen Ying, Thomas G. Robertazzi, *Fellow, IEEE*

1 Proof

This part is the derivation of the maximum of $T_{n,m}$ in page 5.

$$\begin{aligned} T_{n,m} &= \textstyle\sum_{i=m}^{n} t_i \\ &= \sum_{i=m}^{n} \frac{\binom{i-1}{m-1}}{\binom{n}{m}}(2i-m)\overline{X} && (1) \\ &= 2\sum_{i=m}^{n} i\frac{\binom{i-1}{m-1}}{\binom{n}{m}}\overline{X} - m\sum_{i=m}^{n} \frac{\binom{i-1}{m-1}}{\binom{n}{m}}\overline{X} && (2) \\ &= 2\sum_{i=m}^{n} m\frac{\binom{i}{m}}{\binom{n}{m}}\overline{X} - m\overline{X} && (3) \\ &= 2m\frac{n+1}{m+1}\overline{X} - m\overline{X} && (4) \\ &= \frac{2nm-m^2+m}{m+1}\overline{X} && (5) \\ & && (6) \end{aligned}$$

$$\frac{\mathrm{d}}{\mathrm{d}m}T_{m,n} = \frac{2n-m^2-2m+1}{(m+1)^2}\cdot\overline{X} \tag{7}$$

Then get $m=\sqrt{2n+2}-1$. We demonstrate $\sqrt{2n+2}-1$ is between 1 and n.

$$\begin{aligned} \sqrt{2n+2}-1 &< n && (8) \\ 2n+2 &< (n+1)^2 && (9) \\ 1 &< n^2 && (10) \\ & && (11) \end{aligned}$$

2 Mesh Networks: Circuit Switched and Wormhole Routing

We discuss a type of mesh network as shown in figure 1. It is proposed for circuit-switched and wormhole routing. [1] In circuit switching and wormhole routing the communication delay does not depend on the distance between originator node and lower layer nodes that are searched by the originator node. As a result it is possible to search the signature of those nodes very far from the originator. Each scattering consists of two phases: chess queen moves and then cross moves using the torus wrap-around connections. The searching process pattern recursively repeats itself in sub-meshes of five times smaller side size. Let p denote the number of processors used in each phase of the searching process. Here the communication path patterns are repeated in sub-meshes with $p+1$ times shorter side size, and all p ports of the active processors are busy in each searching phase. In the first searching phase the originator searches p processors. In the next phase each searched processor search their p new processors. Thus, $p(p+1)^{i-1}$ processors are being searched in phase i. At the end of phase i the number of processors that have been searched is $(p+1)^i$. The searching process can be viewed as a tree which can be called a $(p+1)$ - nominal tree. Let us call a layer the set of processors activated in the same phase, and hence on the same level of the $(p+1)$ -

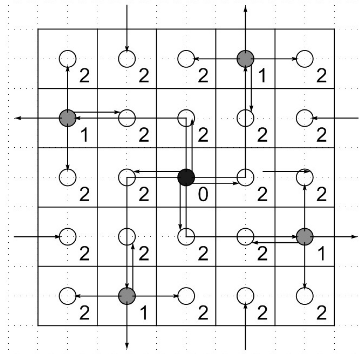

Fig. 1. Circuit-switched and wormhole routing mesh network

nominal tree. However, it is different from the searching signature process in the tree network. For the tree network, the nodes which have searched their children nodes will not search any other nodes, but in this mesh structure, each node will be searched once, but later each node will search other nodes during the search process. Assume that there are m processors(nodes) should be searched and the expected time to search one node is $\overline{X}$, so the total search time is $T = \log_{p+1} m \cdot \overline{X}$.

3 Hypercubes

In this section we assume that the system is homogeneous [2]. A hypercube is an n-dimensional generalization of a square (n = 2) and a cube (n = 3). [wikipedia] The search process in a hypercube is commanded by activating processor along consecutive hypercube dimensions. For example, P_0 search its file and then commands P_1 to search its own file along dimension 0, then P_0, P_1 command P_2, P_3, respectively, along dimension 1, etc. This scattering method uses binomial trees embedded in the log m - dimensional mesh of side size 2. Note that this is a special case of $(p+1)$ - nominal tree. Let $d = \log m$, which m is the total number of processors here. Each processor of the hypercube is labeled with d - bit binary number such that neighboring nodes differ in exactly one bit. Thus, there are $\log m$ ports in each node. Suppose the originator node P_0 has label 0. Then, its direct neighbors have exactly one bit equal to 1 in their labels. If a processors is in a distance of two hops from the originator, it will have two bits equal to 1 in their labels. So, processors with i number of 1_s in their labels are i number of hops away from the originator. Let us call a layer the set of processors in equal hop from the originator. The number of processors in layer i is $\binom{d}{i}$. A processor in layer i may command layer $i+1$ through $d-i$ ports. So in the hypercubes network, any node can be the original node to command other nodes to search their own files, assume the time to search one node is $\overline{X}$, so the total searching time should be $\log m \cdot \overline{X}$.

References

[1] M. Drozdowski, *Scheduling for Parallel Processing*. Springer-Verlag New York Inc, 2009.
[2] J. Blazewicz and M. Drozdowski, "Scheduling divisible jobs on hypercubes+," *Parallel computing*, vol. 21, no. 12, pp. 1945–1956, 1995.