Choosing the Best $\log_k(N, m, P)$ Strictly Nonblocking Networks

F. K. Hwang

***Abstract*—We extend the $\log_2(N, m, P)$ network proposed by Shyy and Lea to base k. We give a unifying proof (instead of three separate cases as done by Shyy and Lea) for the condition of being strictly nonblocking, and a simpler expression of the result. We compare the number of crosspoints for $\log_k(N, m, p)$ over various k.**

***Index Terms*— Banyan network, Clos network, strictly nonblocking network.**

I. Introduction

THE NOTION of a $\log_2(N, m, p)$ network for designing photonic switching systems was introduced by Lea in [1] and by Shyy and Lea in [2]. Following [2], we use the *Banyan network* as representative. An *m-extra-stage Banyan network* is a cascade of the Banyan network with m extra stages which are the mirror image of the first m stages of the Banyan network. A $\log_2(N, m, p)$ network can be treated as a symmetrical three-stage Clos network with $N = 2^s$ inlets, while the middle stage consists of p copies of an m-extra-stage Banyan network, and the first stage consists of 2^{s-1} copies of a $2 \times p$ crossbar.

While a Banyan network usually uses 2×2 crossbars as components, it can be easily extended to a k-nary Banyan network using $k \times k$ crossbars. Fig. 1 shows a ternary Banyan network with $N = 3^3$ inlets.

By using the k-nary Banyan networks in the middle stage, we can define a $\log_k(N, m, p)$ network with $N = k^s$ inlets, where the input stage consists of N $1 \times p$ crossbars and the output stage N $p \times 1$ crossbars.

The value of p to guarantee strict nonblockingness of the $\log_2(N, m, p)$ network was given in [1] and [2]. Their arguments, divided into three cases $m = 0, 1, \geq 2$ also hold for the $\log_k(N, m, p)$ network by simply replacing base 2 with base k. We give a simpler proof by unifying the three cases (also a simpler expression for p).

Theorem 1: A $\log_k(N, m, p)$ network is strictly nonblocking if

$$p = \left\lceil \frac{2m(k-1)+1}{k} \right\rceil + (k+1)k^{(s-m)/2-1} - 2, \qquad \text{for } s+m \text{ even}$$
$$= \left\lceil \frac{2m(k-1)+1}{k} \right\rceil + 2k^{(s-m-1)/2} - 2, \qquad \text{for } s+m \text{ odd.}$$

Proof: Suppose $N = k$. Define $\delta_x = 1$ if x is even and $= 0$ otherwise. We use an argument analogous to the one given in [1] for the $k = 2$ case.

Consider the channel graph $G(i, o)$ between an input i and an output o. From the structure of BY^{-1} and the pattern F^{-1}, it is easily verified that $G(i, o)$ is a symmetric series-parallel channel graph with k branching at the m outer shells. Let NP_j denote the number of paths at shell j. Then

$$NP_j = \begin{cases} k^j, & \text{for } j \leq m \\ k^m, & \text{for } j \geq m. \end{cases}$$

A stage-j link may also be seized by a connection (i', o') where $i' \neq i$ and $o' \neq o$. We call such a connection an *intersecting connection*. To avoid counting twice, we must assign such an intersecting connection either to i' or to o'. We assign it to the input side of inputs (outputs) which can generate an intersecting connection seizing a shell-j link. Then

$$NI_j = NO_j = (k-1)k^{j-1},$$
$$\text{for } 1 \leq j \leq (n+m)/2 \, (\text{except } NO_{(n+m)/2} = 0.$$

Assuming the worst case that the NI_j and NO_j intersecting connections are all disjoint, then a portion $(NI_j + NO_j)/NP_j$ of the paths in $G(i, o)$ is unavailable to (i, o). Therefore, the condition of SNB is

$$p > \sum_{i=1}^{m} \frac{2(k-1)k^{(i-1)}}{k^i} + \sum_{i=m+1}^{\lfloor (s+m-1)/2 \rfloor} \frac{2(k-1)k^{i-1}}{k^m} + \delta_{s+m} \frac{(k-1)k^{(s+m)/2-1}}{k^m}$$
$$= \frac{2m(k-1)}{k} + \frac{2(k-1)}{k^m} \frac{k^{\lfloor (s+m-1)/2 \rfloor} - k^m}{k-1} + \delta_{s+m}(k-1)k^{(s-m)/2-1}$$
$$= \frac{2m(k-1)}{k} + 2\left(k^{\lfloor (s-m-1)/2 \rfloor} - 1\right) + \delta_{s+m}(k-1)k^{(s-m)/2-1}.$$

Theorem 1 follows immediately. ■

II. Minimizing the Number of Crosspoints

Let # denote the number of crosspoints. We first compare strictly nonblocking $\log_k(N, m, p)$ over various k for # by keeping N invariant. It is tacidly assumed that N can be approximated by a power of k so that the formula in Theorem 1 applies. We will also ignore the integrality of p.

An m-extra-stage Banyan network has $s + m$ stages each consisting of k^{s-1} $k \times k$ crossbars; therefore, it has $(s+m)Nk$ crosspoints. Thus, $\log_k(N, m, p)$ has $(s+m)Nkp$ crosspoints in the middle and N $1 \times p$ crossbars in each of its input and output stages. Thus

$$\# = 2Np + (s+m)Nkp = Np[2 + (s+m)k].$$

Paper approved by N. McKeown, the Editor for Switching and Routing of the IEEE Communications Society. Manuscript received October 16, 1996; revised September 12, 1997.

The author is with the Department of Applied Mathematics, Chiao-Tung University, Hsin-Chu, Taiwan 30050, R.O.C.

Publisher Item Identifier S 0090-6778(98)03136-5.

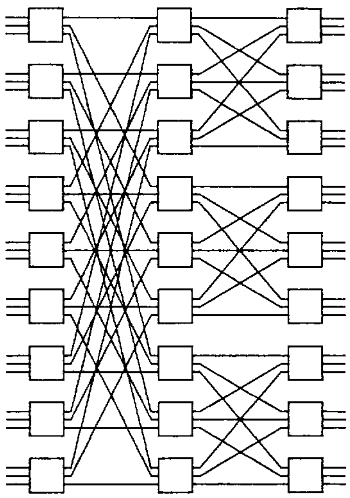

Fig. 1. A ternary Banyan network with 27 inlets.

Case 1: $s+m$ is even. Then

$$\# = N\left[\frac{2m(k-1)+1}{k} + (k+1)k^{(s-m)/2-1} - 2\right] \cdot [2+(s+m)k].$$

Let c denote a positive integer much smaller than s. If $m=c$, then

$$\begin{aligned} \# &\sim N(k+1)k^{(s-c)/2-1}(s+c)k \\ &\sim N^{3/2}(k+1)k^{-(c/2)} \log_k N \end{aligned}$$

which is essentially increasing in k for $c=0$ or 1 and decreasing in k for $c \geq 2$, where "essentially increasing" means exceptions are allowed for very small k. If $m=s-c$ then

$$\begin{aligned} \# &\sim N\frac{2(s-c)(k-1)+1}{k}(2s-c)k \\ &= N[2(\log_k N - c)(k-1)+1][2 \log_k N - c] \end{aligned}$$

which is essentially increasing in k for all c.

Case 2: $s+m$ is odd. Then

$$\# = N\left[\frac{2m(k-1)+1}{k} + 2k^{(s-m-1)/2} - 2\right][2+(s+m)k].$$

If $m=c$, then

$$\begin{aligned} \# &\sim N2k^{(s-c-1)/2}(s+c)k \\ &= 2N^{3/2}k^{-[(c-1)/2]}(\log_k N + c) \end{aligned}$$

which is essentially increasing in k for $c=0$ but decreasing in k for $c \geq 1$. If $m=s-c$, then

$$\# \sim N\frac{2(\log_k N - c)(k-1)+1}{k}(2 \log_k N - c).$$

The analysis is the same as in case 1.

When # is essentially increasing in k, the optimal k is a small k which can be determined by standard method. When # is decreasing in k, the optimal k should be as large as practicality allows. Note that the optimal k is independent of N. Also note that $\log_k(N, m, p)$ requires $O(N^{3/2})$ crosspoints for $m=c$ but only $O(N \log^2 N)$ crosspoints for $m=s-c$ ($m=s-1$ yields a variation of the Cantor network).

Sometimes, for a technology or performance reason, it is necessary to keep s constant. Then $N=k^s$ varies with k and will be denoted by N_k. In this case one should compare $\#/N_k$, the number of crosspoints per input (or output). We have previously shown

$$\frac{\#}{N_k} = p[2+(s+m)k].$$

It is easily verified that p as given in Theorem 1 is increasing in k. Hence, $\#/N_k$ is increasing in k, and the optimal choice of k is $k=2$.

Since there is no *a priori* reason to argue for $k=2$ being the optimal choice, one expects $\#/N_k$ to consist of two factors, one increasing in k and the other decreasing, and the optimal k is determined by balancing these two factors. It is surprising to find both factors in $\#/N_k$ increasing in k.

References

[1] C.-T. Lea, "Multi-$\mathrm{Log}_2 N$ networks and their applications in high-speed electronic and photonic switching systems," *IEEE Trans. Commun.*, vol. 38, pp. 1740–1749, Oct. 1990.

[2] D.-J. Shyy and C.-T. Lea, "$\mathrm{Log}_2(N, m, p)$ strictly nonblocking networks," *IEEE Trans. Commun.*, vol. 39, pp. 1502–1510, Oct. 1991.