1990 International Conference on Parallel Processing

STRUCTURE OF DIGIT PERMUTATION NETWORKS

Abdou Youssef

Department of Elect. Eng. & Comput. Sci.
The George Washington University
Washington, DC 20052

Bruce Arden

College of Engineering and Applied Science
University of Rochester
Rochester, NY 14627

Abstract – This paper examines the class of digit permutation networks (DPN) which are banyan multistage networks where the interconnections are operations that permute bits or digits in a specified manner. It will be shown that all DPN's are efficiently controllable and functionally equivalent to the baseline. The paper will also present an efficient, parallel algorithm that relabels the terminals of the baseline to simulate an arbitrary DPN.

Introduction

Banyan multistage interconnection networks (MIN's) are increasingly important in parallel computing systems. Several MIN's have been proposed and studied, such as omega and its inverse [2], the indirect binary n-cube [3], the baseline [5], and the generalized cube network [4].

As MIN's have the unique path property, they can be self-routed via control tags, also called path descriptors. The control efficiency depends then on the speed of control tag computation. The class of networks where the control tags from all sources to a certain destination are equal and depend only on the destination address are then of special importance. Such networks are called FD-controllable networks in [6] and delta networks in [1]. The topological structure of these networks was studied by the authors of these two references independently and was shown to be recursive. Furthermore, the subclass of doubly FD-controllable networks (also called bidelta networks in [1]) was also studied in [1] and [6]. A network is doubly FD-controllable if the network and its inverse are both FD-controllable. It was shown in [6] and [1] that all FD-controllable MIN's are functionally equivalent to the baseline. That is, given any doubly FD-controllable network W, the terminals of the baseline can be relabeled so that the latter network realizes the same permutations as W.

This paper examines the sub-class of $r^k \times r^k$ k-column MIN's with $r \times r$ crossbar switches and the interconnections between columns are digit permutations to be defined later. These networks are called digit permutation networks. The reason for studying this subclass is that it includes all existing banyan multistage networks (for $r = 2$) and that digit permutation interconnections have a rich structure and yield to mathematical analysis.

It will be shown that all digit permutation networks are doubly FD-controllable and therefore functionally equivalent to the baseline network. An efficient, parallel algorithm that relabels the terminals of the baseline to simulate another digit permutation network will be given.

Digit Permutation Networks

An algebraic approach will be used to show that digit permutation networks are doubly FD-controllable. A relation will be derived relating the input terminal, the output terminal and the control tag that establishes the path in between. This relation will be used to find necessary and sufficient conditions for $k+1$ digit permutations to construct a MIN that has the unique path property. Later the control tags in digit permutation networks are shown to be functions of the destination tags only. This makes them FD-controllable. As the inverse of a DPN is a DPN, it will be concluded that digit permutation networks are doubly FD-controllable and hence functionally equivalent to the baseline network.

Definition 1. A permutation f of $S_N = \{0, 1, ..., N-1\}$, where $N = r^k$, is a digit permutation in the system of base r if there exists a permutation π of $S_k = \{0, 1, ..., k-1\}$ such that $f(x_{k-1}...x_1x_0) = x_{\pi(k-1)}...x_{\pi(1)}x_{\pi(0)}$, where $x_{k-1}...x_1x_0$ is an arbitrary k-digit r-ary label. In this case, f is denoted f_π and π is called the kernel of f_π.

Definition 2. A digit permutation network, denoted DPN$(f_0, f_1, ... f_k)$, is a k-column MIN with $r \times r$ switches where the leftmost and rightmost interconnections are f_0 and f_k, the interconnection from column $i-1$ to column i is f_i, for $i = 1, ..., k-1$, and all the f_j's are digit permutations of S_{r^k} in the system of base r. Note that the k columns of a MIN are labeled $0, 1, ..., k-1$ from left to right.

Denote by E_a^i, where a is an r-ary digit and $i = 0, 1, ..., k-1$, the following mapping from S_N to S_N: $E_a^i(x_{k-1}...x_0) = x_{k-1}...x_{i+1}ax_{i-1}...x_0$. E_a^i replaces the i-th digit of its argument by a.

Next, the relation between an input terminal s, an output terminal d and the control tag $c = c_{k-1}c_{k-2}...c_0$ for the path $s \to d$ in a DPN$(f_{\pi_0}, f_{\pi_1}, ..., f_{\pi_k})$ will be drived. The digit c_{k-1-i} controls column i for $i = 0, 1, ..., k-1$. If the path $s \to d$ enters column i through input port $x_{k-1}...x_1x_0$, it exits that column through the output port $x_{k-1}...x_1c_{k-1-i}$ which is equal to $E^0_{c_{k-1-i}}(x_{k-1}...x_1x_0)$. Note that if the path exits column $i-1$ through some output port y, it then enters the next column, that is, column i, through input port $f_{\pi_i}(y)$ because the interconnection between column $i-1$ and column i is f_{π_i}. We thus have:

Lemma 1. In a DPN$(f_{\pi_0}, f_{\pi_1}, ..., f_{\pi_k})$ an output terminal d is related to an input terminal s and the control tag $c = c_{k-1}c_{k-2}...c_0$ for the path $s \to d$ by: $d = (s)f_{\pi_0}E^0_{c_{k-1}}f_{\pi_1}E^0_{c_{k-2}}f_{\pi_2}E^0_{c_{k-3}}...E^0_{c_0}f_{\pi_k}$.

The E's will be "filtered" out to the right of the f's in the relation above.

Lemma 2. $f_\alpha f_\beta = f_{\beta\alpha}$ and $E_a^i f_\pi = f_\pi E_a^{\pi^{-1}(i)}$.

Lemma 3. Under the assumptions of Lemma 1 we have $d = (s)f_{\beta_0}E_{c_{k-1}}^{\beta_1^{-1}(0)}E_{c_{k-2}}^{\beta_2^{-1}(0)}...E_{c_0}^{\beta_k^{-1}(0)}$ where $\beta_i = \pi_k\pi_{k-1}...\pi_i$.

Proof. Let $g = f_{\pi_0}E^0_{c_{k-1}}f_{\pi_1}E^0_{c_{k-2}}...E^0_{c_1}f_{\pi_{k-1}}E^0_{c_0}f_{\pi_k}$.

By making repeated use of Lemma 2 on the expression of g (from right to left) we conclude that $g = f_{\beta_0} E_{c_{k-1}}^{\beta_1^{-1}(0)} E_{c_{k-2}}^{\beta_2^{-1}(0)} \ldots E_{c_0}^{\beta_k^{-1}(0)}$. As $d = g(s)$ (from Lemma 1), the lemma follows. ∎

The necessary and sufficient conditions as well as the relation between the control tag and the output terminal can now be easily derived as follows.

Theorem 1. Let f_{π_0}, f_{π_1}, ... , and f_{π_k} be $k+1$ digit permutations, and $\beta_i = \pi_k \pi_{k-1} \ldots \pi_i$. Then:
(a) DPN($f_{\pi_0}, f_{\pi_1}, \ldots, f_{\pi_k}$) has the unique path property if and only if $\beta_1^{-1}(0)$, $\beta_2^{-1}(0)$, ... , $\beta_k^{-1}(0)$ are pairwise distinct .
(b) The control tag $c = c_{k-1}c_{k-2}\ldots c_0$ for a path $s \to d$ in DPN($f_{\pi_0}, f_{\pi_1}, \ldots, f_{\pi_k}$) that has the unique path property is $c = f_\gamma(d)$, where $\gamma(i) = \beta_{k-i}^{-1}(0)$.

Proof. (a) Let s be an input terminal, d an output terminal and $c = c_0c_1\ldots c_{k-1}$ the control tag that establishes the path $s \to d$. Let $s' = f_{\beta_0}(s)$. By Lemma 3, $d = (s')E_{c_{k-1}}^{\beta_1^{-1}(0)} E_{c_{k-2}}^{\beta_2^{-1}(0)} \ldots E_{c_0}^{\beta_k^{-1}(0)}$. The effect of each $E_{c_{k-i}}^{\beta_i^{-1}(0)}$ is to replace the digit in position $\beta_i^{-1}(0)$ of s' by c_{k-i}.

Assume first that the network has the unique path property. If $\beta_1^{-1}(0)$, $\beta_2^{-1}(0)$, ... , $\beta_k^{-1}(0)$ are not pairwise distinct, then $\{\beta_1^{-1}(0), \beta_2^{-1}(0), \ldots, \beta_k^{-1}(0)\}$ is a proper subset of $\{0, 1, \ldots, k-1\}$, and therefore, there exists some j in $\{0, 1, \ldots, k-1\}$ such that $j \neq \beta_i^{-1}(0)$ for all i. Consequently, the digits in the j-th digit position of s' and d must always agree. It follows that for a fixed s, and thus fixed s', no matter what control tag we use, we can never reach any output terminal d whose j-th digit differs from that of s'. This contradicts the unique path property.

Conversely, if $\beta_1^{-1}(0)$, $\beta_2^{-1}(0)$, ... , $\beta_k^{-1}(0)$ are pairwise distinct, then $\{\beta_1^{-1}(0), \ldots, \beta_k^{-1}(0)\} = \{0, 1, \ldots, k-1\}$, and therefore the mapping γ where $\gamma(i) = \beta_{k-i}^{-1}(0)$ is a permutation of $\{0, 1, \ldots, k-1\}$. Furthermore,

$$d = (s')E_{c_{k-1}}^{\gamma(k-1)} E_{c_{k-2}}^{\gamma(k-2)} \ldots E_{c_0}^{\gamma(0)}$$

implying that the digit in position $\gamma(i)$ of d is c_i, that is, $d_{\gamma(k-1)}d_{\gamma(k-2)}\ldots d_{\gamma(0)} = c_{k-1}c_{k-2}\ldots c_0$. Therefore, $c = f_\gamma(d)$ and $d = f_{\gamma^{-1}}(c)$. As $f_{\gamma^{-1}}$ is a permutation of S_N, it follows that for a fixed input terminal s there corresponds to every control tag c one and only one output terminal. Therefore, the network has the unique path property.
(b) The relation $c = f_\gamma(d)$ has just been proved in (b). ∎

Theorem 2. Every DPN network is functionally equivalent to the baseline network.

Proof. Since the inverse of a DPN is a DPN and every DPN is FD-controllable, it follows that every DPN is doubly FD-controllable, and hence functionally equivalent to the baseline network. ∎

Theorem 2 supersedes the results in [4] and [5] about the functional equivalence among existings MIN's and shows that this equivalence is no coincidence but follows from the fact that the existing networks are DPN's.

Baseline Simulation of Digit Permutation Networks

To simulate a network $W = \text{DPN}(f_{\pi_0}, \ldots, f_{\pi_k})$ by the baseline, the terminals of the latter need to be relabeled. Note that the control tag to a destination d in W is $f_\gamma(d)$, where γ is as given in theorem 1. Thus, if every output terminal d of the baseline is relabeled $f_{\gamma^{-1}}(d)$, the control tags in both W and the baseline become identical. To relabel the inputs of the baseline, we compute the γ of W^{-1} (call it τ to avoid confusion). Every input s of the baseline should be relabeled $f_{\tau^{-1}}(s)$. These steps are summarized below.

Procedure Simulate (W)
begin
(1) Compute γ and then γ^{-1};
(2) Compute τ and then τ^{-1};
(3) Broadcast τ^{-1} to all inputs and γ^{-1} to all outputs;
(4) **for** $i = 0$ **to** $N-1$ **do in parallel**
(5) relabel input i of the baseline by $f_{\tau^{-1}}(i)$;
(6) relabel output i of the baseline by $f_{\gamma^{-1}}(i)$;
end

Time Complexity: Steps 1-2 can be shown to take $O(k^2)$ each. Step 3 takes $O(k)$. Steps 5 and 6 take $O(k)$ each as the relabeling of a node consists of permutating its k digits. Thus, the procedure takes $O(\log_r^2 N)$.

Conclusions

This paper has examined the class of digit permutation networks which includes all existing banyan multistage networks. We have shown that every DPN is doubly FD-controllable and hence functionally equivalent to the baseline. An immediate consequence is that every digit permutation network can be simulated by the baseline by appropriately relabeling the terminals of the baseline. An efficient algorithm for such relabeling was given.

References

[1] C. P. Kruskal and M. Snir, "A unified Theory of Interconnection Network Structure, *Theoretical Computing Science,* Vol. 48, pp. 75–94, 1986.

[2] D. K. Lawrie, "Access and Alignment of Data in an Arrary Processor," *IEEE Trans. Comput.,* C-24, pp. 1145–1155, Dec. 1975.

[3] M. C. Pease, "The Indirect Binary n-Cube Multiprocessor Array," *IEEE Trans. Comput.,* C-26, pp. 458–473, May 1976.

[4] H. J. Siegel, "A model of SIMD Machines and a Comparison of Various Interconnection Networks," *IEEE Trans. Comput.,* Vol C-28, pp. 907–917, Dec. 1979.

[5] C.L. Wu and T.Y. Feng, "On a Class of Multistage Interconnection Networks," *IEEE Trans. Comput.,* vol. C-29, No. 8, pp. 694–702, August 1980.

[6] A. Youssef, *Properties of Multistage Interconnection Networks,* Ph.D. dissertation, Princeton University, Feb. 1988.