Codes for Network Switches

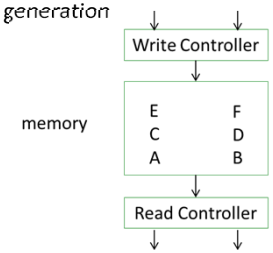
Zhiying Wang, Omer Shaked†, Yuval Cassuto†, and Jehoshua Bruck **

* California Institute of Technology, † Technion - Israel Institute of Technology

Introduction

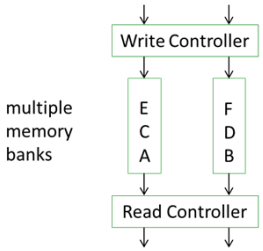
Network Switches

- Multiple (R) input and output ports
- Packets stored in memory upon arrival
- R write and read bandwidth
- Packets in the same time slot are called a *generation*



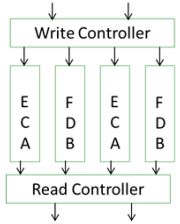
Affordable Memories

- Multiple (R) memory banks
- Unit write and read bandwidth
- Contention when requests several reads from the same memory bank



Switch Codes

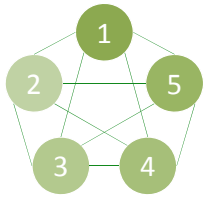
- n memory banks
- R input bits and R request bits
- Solve any output request by reading at most one bit from each memory



Pair Parities

Complete Graph Construction

- Includes all pairs of information bits as parities
- $n = R(R+1)/2$

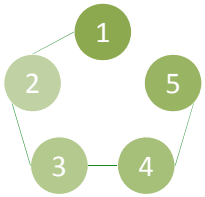


Request (1,1,2,2,3) from 5 different generations can be solved by (1),(4,4+1),(2),(5,5+2),(3).

Theorem: The above construction solves any request and is optimal for pair parities.

Connected Line Construction

- Includes adjacent pairs as parities
- $n = 2R-1$



Request (1,2,3) from the first generation, and (1,2) from the second generation can be solved by: (1),(2),(3) from the first generation, and (4),(4+3),(3+2),(2+1) from the second generation.

Theorem: The above solves all requests from two consecutive generations and is optimal for pair parities.

Triple Parities

Theorem (Lower bound): A systematic switch code with constant degree d parities satisfies

$$n - R \geq R(R-1)/d.$$

Proof: Every information element is contained in at least R memory banks. Lower bound can be shown by counting argument.

Balanced Incomplete Block Design (BIBD)

- *k=3:* size of blocks
- *R:* number of elements
- *r=R-1:* repeats for each element
- Every *pair* appears exactly *twice*
- *b=R(R-1)/3:* number of blocks
- *n=b+R=R(R+2)/3*

Example: Let information bits be 0,1,…,5. Each block of size 3 corresponds to the XOR of three bits. Let parities be {0, 1, 2}, {0, 2, 3}, {0, 1, 4}, {1, 2, 5}, {0, 3, 5}, {2, 3, 4}, {0, 4, 5}, {1, 4, 3}, {1, 5, 3}, {2, 5, 4}. Then a request (0, 0, 0, 1, 4, 5) can be solved by:

(0)
(2, {0, 4, 5}, {2, 5, 4})
(3, {0, 1, 4}, {1, 4, 3})
(1)
(4)
(5)

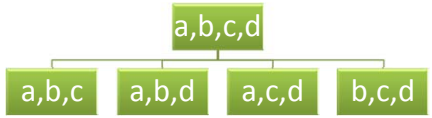
If only one bit is requested several times, it is called a *one-burst* request.

Top-down Construction

- BIBD with *k=4, R, r=(R-1)/3*, every *pair* appears exactly *once*
- Break each block into 4 triples
- *n= R(R+2)/3*

Theorem: The above solves any one-burst request with burst length $(R-1)/3+1$. And it exists for all $R \equiv 1, 4 \bmod 12$.

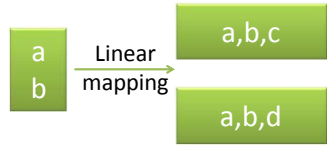
Pair Parities



Take any two triples have two bits in common, thus can be used to solve one of the two remaining bits, e.g., (a,{a,b,c},{a,b,d}) can solve for d.

Linear Mapping Construction

- Map every two bits to two triples, which are used to solve one of the two bits.
- $n = R(R+2)/3$



(a,{a,b,c},{a,b,d}) can solve for d.

Theorem: The linear construction serves any one-burst request, when $R > 3$ is a prime and -3 is a perfect square mod R.

Bibliography

[1] C. J. Colbourne and J. H. Dinitz, *Handbook of Combinatorial Designs (Second Edition)*. CRC Press, 2007.
[2] R. Fisher and F. Yates, *Statistical tables for biological, agricultural, and medical research*. Longman Group United Kingdom, June 1995.
[3] P. Gopalan, C. Huang, H. Simitci, and S. Yekhanin, "On the locality of codeword symbols," *Information Theory, IEEE Transactions on*, vol. 58, no. 11, pp. 6925–6934, nov. 2012.
[4] G. H. Hardy and E. M. Wright, *An Introduction to the Theory of Numbers (Fifth Edition)*. Oxford University Press, 1980.
[5] F. Oggier and A. Datta, "Self-repairing homomorphic codes for distributed storage systems," in *INFOCOM, 2011 Proceedings IEEE*. IEEE, 2011, pp. 1215–1223.
[6] L. Pamies-Juarez, H. D. L. Hollmann, and F. E. Oggier, "Locally repairable codes with multiple repair alternatives," *CoRR*, vol. abs/1302.5518, 2013.
[7] S. Yekhanin, "Locally decodable codes," *Computer Science–Theory and Applications*, pp. 289–290, 2011.