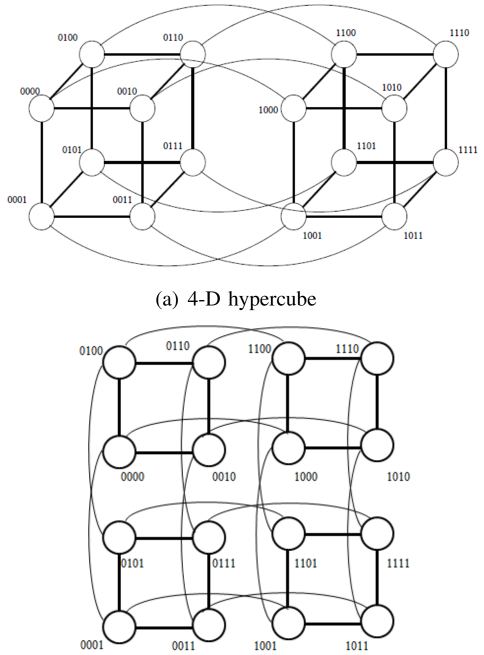
Traffic dynamics and vulnerability in hypercube communication networks

Mario di Bernardo*, Elisa Maini*, Antonio Manzalini† and Nicola Mazzocca*
*Department of Electrical Engineering and Information Technology
University of Naples Federico II, Via Claudio 21, 80125 Naples, Italy
Email:{mario.dibernardo, elisa.maini, nicola.mazzocca}@unina.it
†Telecom Italia Strategy - Future Centre
Via Reiss Romoli 274, 10148, Turin, Italy
Email:antonio.manzalini@telecomitalia.it

***Abstract*—In this paper we introduce the hypercube topology and show effects over the performance of communication networks in term of throughput and average lifetime. Hypercube network structures have been proposed as effective topologies to increase the network performance. We test this hypothesis by considering appropriate nonlinear traffic generation models and comparing the performance of hypercube networks with that of other network topologies which were already studied in the literature. We show that the hypercube network structure presents better features combining some of the advantages of regular lattices with those of other more complex network structures.**

I. Introduction

The rapid growth of on-line services such as Cloud Computing is placing tremendous demands on the performance of the underlying network infrastructure. To address these challenges, network operators design routing protocols and tune their parameters so as to control how traffic is routed across the network optimize performance and use network resources effectively. The use of specialized topologies to improve network performance is an open problem [1], [2]. Indeed, any improvement of the performance can help reduce energy used in Data Centers. Within this context, hypercube topologies have been proposed as a possible solution and have been heavily used to implement parallel algorithms that require all-to-all communication [3]. In the Nonlinear Circuits and Systems community, much research effort has been spent to investigate the link between the network structure and its performance. For example in [4], [5], [6], [7], the effects of different network topologies on average throughput and delivery time in packet data networks have been analyzed. By using an innovative model of traffic generation based on the use of chaotic maps, it was shown that under certain conditions scale-free networks can perform better than other topologies such as random or regular lattices. Also, it was shown that the traffic behavior on the network is influenced by several factors such as transmission rate and queue lengths at the vertices. The aim of this paper is to extend the analysis presented therein to encompass the more recent case of hypercube network structures. The goal is to assess what advantages/disadvantages this type of networks present when subject to varying traffic loads under different circumstances. Also, we shall seek to analyze and compare the vulnerability properties of the network to evaluate the variation in its throughput when one or more nodes fail due to random or intentional attacks. We show that hypercube networks do perform well under different circumstances showing some of the advantages both regular planar lattices and scale-free networks combined that were discussed in [5].



(a) 4-D hypercube

(b) 4-D hypercube into a 2-D mesh

Figure 1. 4-hypercube mapped into a 2-D mesh using Gray Code

II. Hypercube network topology

Hypercube networks play an increasingly important role in global communication operations, interconnecting networks of microcomputers in parallel and distributed environments [1], [2]. Generally speaking, an n-dimensional p-ary hypercube consists of p^n nodes. Each node has n nearest neighbors. Nodes are labeled using base-p numbers, and it is assumed

978-1-4799-3432-4/14/$31.00 ©2014 IEEE

that any two nodes *i* and *j* are connected via a bidirectional link if their labels differ in exactly one coordinate position, i.e. Node $(x_{n-1}, x_{n-2}, ..., x_i, ..., x_0)_{base-p}$ is connected to Node $(x_{n-1}, x_{n-2}, ..., \underline{x_i}, ..., x_0)_{base-p}$ if $(x_i \neq \underline{x_i})$, $0 \leq i < n$ [13]. Here we will focus on a binary hypercube topology where *p=2*. Such a structure can be associated to a graph $G(V, E)$ in which [12]: a) V has 2^n vertices; b) every vertex has degree *n*; c) *G* is connected; d) any two adjacent nodes A and B are such that the nodes adjacent to A and those adjacent to B are linked one-to-one. Note that it is possible to map the *n*-D hypercube topology onto a 2-D mesh topology using Gray Code, modelling this kind of problem in graph-theoretical terms as that of a graph embedding [12]. In the nominal case a binary hypercube topology with $n = 4$, in Figure 1(a), is mapped onto the $L \times L$ mesh with $L = 4$ shown in Figure 1(b).

III. Network model

Our network model consists of two types of nodes: *routers* (that store and forward packets) and *hosts* (that are also sources of traffic). Assuming the network has N nodes and a density $\rho \in [0, 1]$ of hosts then ρN is the number of total hosts and $(1 - \rho)N$ is the total number of routers. We suppose for the sake of simplicity that host nodes are randomly distributed in the network.

The network model considered in this paper consists of the following key ingredients:

1) *Traffic generation model*: a packet is generated at a host using either a uniform random distribution (Poisson like) or a Long Range Dependence (*LRD*) distribution defined by a chaotic map. Each source generates its traffic independently of the other sources; the traffic load is increased or decreased by varying the probability of packet generation at each node [8], [9]. A random destination is assigned to each newly generated packet. The destination node is selected with uniform probability among all other hosts in the network.
2) *Buffer size*: each node keeps a queue of unlimited or limited length where the newly generated packets or those waiting to be routed are stored. Any packets that is generated is put at the end of the host' s queue. If a packet arrives at a router is stacked at the end of the router's queue. Packets at the head of each queue, exceeding its maximum capability, are dropped. The packets are removed when they arrive at their destination site.
3) *Routing algorithm at every time-steps*: each node picks a packet at the head of its queue and forwards the packet to the next node. The information that each packet carries about its source and destination is used by the following routing algorithm as follow. a) A neighbor closest to the destination node is selected. b) If more than one neighbor is at the minimum distance from the destination, the link through which the smallest number of packets has been forwarded is selected. c) If more than one of these links shares the same minimum number of packets forwarded, then a random selection is made.

The process of packet generation, hop movement, queue updating and updating of the routing table occurs at each time step.

To generate packet data traffic, a chaotic map has been used following the approach presented in [10]. We used the family of maps defined over the unit interval as

$$x_{n+1} = \begin{cases} x_n + (1-\lambda)(\frac{x_n}{\lambda})^{m_1}, & \text{if } x_n \in [0, \lambda] \\ x_n - \lambda(\frac{1-x_n}{1-\lambda})^{m_2}, & \text{if } x_n \in (\lambda, 1] \end{cases} \tag{1}$$

where $\lambda \in (0, 1)$ and the parameters $m_1, m_2 \in (\frac{3}{2}, 2)$ induce intermittency.

The map produces a sequence of real numbers $x_n \in [0, 1]$ which is converted into a binary Off-On sequence given by

$$y_n = \begin{cases} 1, & \text{if } x_n \in [0, \lambda] \\ 0, & \text{if } x_n \in (\lambda, 1] \end{cases} \tag{2}$$

where λ is used to tune the "load" on the network (a new packet is generated only if $y_n = 1$).

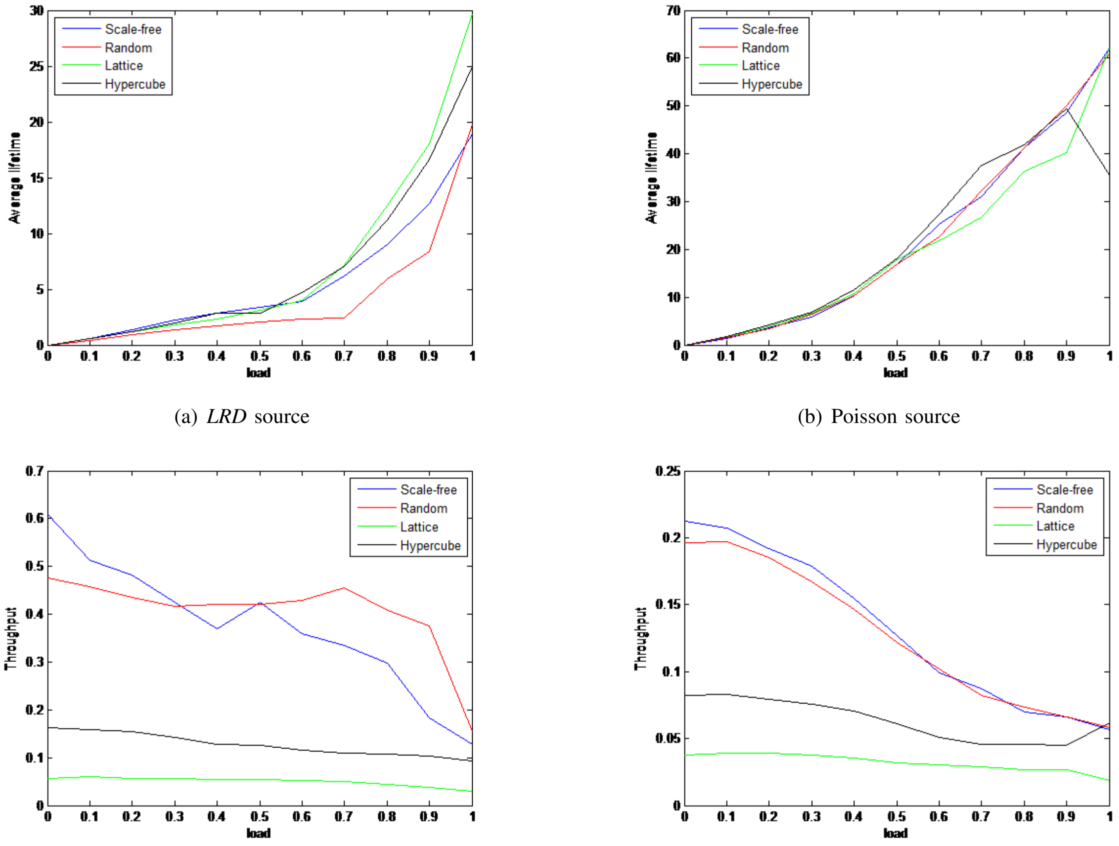
Furthermore, the Hurst parameter, *H* associated to this map is given by:

$$H = 1 - \frac{\beta}{2} = \frac{3m - 4}{2(m - 1)} \tag{3}$$

where $\beta = \frac{2-m}{m-1} \in (0, 1)$ and $m = max\{m_1, m_2\}$ with $m_1, m_2 \in (\frac{3}{2}, 2)$. Thus $m_1, m_2 = 1.5$ corresponds to Poisson-like behavior and as m_1 and m_2 increase towards 2, the behavior is increasingly LRD as proved in [8] (e.g. in our model $m_1 = 1.95$ and $m_2 = 1.95$).

IV. Network performance

We consider networks with different topologies generated using the most appropriate algorithms. For example, we use Erdős-Rényi (ER) algorithm to generate a random network and the static model introduced in [6] to generate scale-free topologies. Furthermore, square lattices with periodic boundary conditions are also considered. Using the network model and traffic generator detailed above, simulations were carried out to analyze various aspects of the end-to-end performance for each different type of network. In all cases, the network size was set to $N = 256$ nodes and the host density to $\rho = 1$. In Figure 2(a) average lifetimes are plotted versus the load for the four cases with LRD traffic sources at each host. Here the average lifetime is simply the average time spent by packets in the network. The load is computed as the average number of packets produced by each traffic source per time step of the simulation. Figure 2(a) confirms that, as pointed out in [6], [8] the network topology is indeed a very important factor. In fact, a regular lattice network has longer lifetimes than a scale-free network. In Figure 2(b) the same measurements are made with Poisson traffic sources substituted for LRD sources. When the traffic sources act following a Poisson distribution we notice a less pronounced



(a) *LRD* source

(b) Poisson source

(c) *LRD* source

(d) Poisson source

Figure 2. Average lifetime and throughput versus the generation rate λ. A LRD traffic source is used to generate panels (a) and (c) while a Poisson traffic source for panels (b) and (c). The number of nodes is $N = 256$ and host density $\rho = 1$. Network considered are: square lattice with periodic boundary $L \times L$ with $L = 16$; Erdős-Rényi (ER) random networks with $p = 0.1$; scale-free network with $\gamma = 3$; hypercube network with $N = 2^{16} = 256$ nodes.

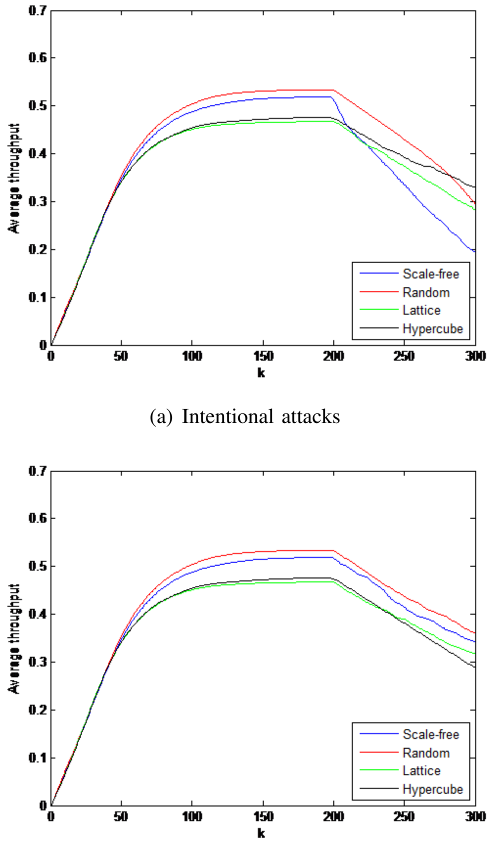
influence of the network structure, confirming the importance of selecting the traffic generation model. Note that the hypercube network presents shorter lifetimes than regular lattices and even scale-free networks. Figure 2(c) shows throughput plotted as a function of the traffic load. The throughput is defined as the number of packets reaching their destination per unit time per host. Results are consistent with those for the average lifetime. The scale-free network performs more efficiently for both types of algorithm. We observe that the hypercube topology presents a higher throughput than that of a square regular lattice. Similar behavior is observed for Poisson sources (Figure 2(d)). It is worth mentioning here that in both cases, the hypercube network structure was found to present a performance which is better than a regular square lattice but also combines the beneficial effects of other topologies such as scale-free and random networks when the average lifetime is considered. This effects is even more pronounced when a more realistic LRD traffic generation model is considered.

V. Vulnerability analysis

Network vulnerability has been studied in a number of papers as an important feature of complex networks. For example in [11] a methodology is presented based on a convex optimization problem to measure maximum end-to-end throughput as a performance indicator considering networks with a fixed number of flows defined through a static routing matrix. We use the dynamic model described above to assess the vulnerability of hypercube networks and compare it with that of different structures. In so doing, we consider the throughput as a performance indicator measuring its change in the presence of different types of attack strategies. In particular, two types of attacks are considered:

- *intentional attacks* where nodes with the highest degrees are removed from the network;
- *random attacks* where nodes selected at random are removed.

Figure 3(a) and Figure 3(b) show the throughput measured for different networks (such as square lattice, random, scale-free and hypercube topologies) in presence of intentional and random attacks as a function of the time k. We next look at the effects of attacks on the same networks assuming that nodes are removed intentionally or at random when $k = 200$. As expected we note that intentional attacks have a much higher effect on the overall network performance. Further, we observe that hypercube networks show much higher resilience to



(a) Intentional attacks

(b) Random attacks

Figure 3. Effects on throughput of intentional (a) and random attacks (b) for different types of networks .

intentional attacks than scale free networks, again combining some of the beneficial, features of different network structures.

VI. Conclusion

We investigated performance and vulnerability of some network topologies. In particular, we extended the analysis previously presented in [6] to hypercube networks. We found that hypercube structures show better performance and resilience than square lattices while retaining some beneficial properties of scale-free networks. We also noted the importance of considering realistic LRD traffic generation models. Future activities will be aimed at generalizing the model by introducing a new type of node, i.e. a computing node (that is a node able to process packets). This is important in order to address the growing interest on new network architectures (e.g. those based on Software Defined Network and Network Function Virtualization [14]) where processing, storage and networking are going to be integrated. In particular, the new node will also allow to assess the effects on performance of but also on the average response time of processing and the dynamic evolution of the memory usage of the network.

References

[1] C. Raiciu, S. Barre, C.Pluntke, A. Greenhalgh, D. Wischik and M. Handley, *Improving Data Center Performance and Robustness with Multipath TCP*, Proceedings of ACM SIGCOMM, pages 266-277, 2011

[2] S. Radhakrishnan, M.Tewari, R. Kapoor, G. Porter and A. Vahdat, *Dahu: Commodity Switches for Direct Connect Data Center Networks*, Proceedings of the 9th ACM/IEEE Symposium on Architectures for Networking and Communications Systems, 2013

[3] D.S. Scott, *Efficient All-to-All Communication Patterns in Hypercube and Mesh Topologies*, Proceedings Sixth Distributed Memory Computing Conference, pages 398-403, 1991

[4] Y. Xia, C.K. Tse, F.C.M. Lau, W. M. Tam and X. Shan *Traffic Congestion Analysis in Complex Networks*, Proceedings IEEE Symposium on Circuits and Systems, pages 2625-2628, 2006

[5] S. Manfredi, F. Garofalo and M. di Bernardo, *Analysis and effects of retransmission mechanisms on data network performance*, Proceedings IEEE Symposium on Circuits and Systems, pages 625-628, 2004

[6] D.K. Arrowsmith, M. di Bernardo and F. Sorrentino, *Communication models with distributed transmission rates and buffer sizes*, Proceedings IEEE Symposium on Circuits and Systems, pages 5047-5050, 2006

[7] D.K. Arrowsmith, M. di Bernardo and F. Sorrentino, *Effects of variations of load distribution on network performance*, Proceedings IEEE Symposium on Circuits and Systems, pages 3773-3776, 2005

[8] D.K. Arrowsmith, R. J. Mondrcigon-C, J.M. Pitts and M. Woolf, *Internet Packet Traffic Congestion*, Proceedings IEEE Symposium on Circuits and Systems, pages 746-749, 2003

[9] D.K. Arrowsmith, R.J. Mondragon and M. Woolf, *Data Traffic, Topology and Networks*, eds. Vattay and Kocarev Springer Verlag, pages 127-158, 2005

[10] A. Erramilli, R.P. Singh and P. Pruthi, *Chaotic maps as models of packet traffic*, Proceedings of14th International Teletraffic Congress, pages 329-338, 1994

[11] I. Mishkovski, R. Kojchev, D. Trajanov and L. Kocarev, *Vulnerability Assessment of Complex Networks based on Optimal Flow Measurements under Intentional node and Edge Attacks*, ICT Innovations, pages 167-176, 2009

[12] Y. Saad and M.H. Schultz, *Topological properties of hypercube*, IEEE Transactions on Computers, vol. 37, pages 867-872, 1988

[13] S.Banerjee and D. Sarkar, *Hypercube Connected Rings: A Scalable and Fault-Tolerant Logical Topology for Optical Networks*, Computer Communications, vol. 24, pages 1060-1079, 2001

[14] A. Manzalini, R. Minerva, E. Dekel, Y. Tock, E. Kaempfer, W.Tavernier, K. Casier, S. Verbrugge, D. Colle, F. Callegati, A. Campi, W. Cerroni, R. Vilalta, R. Muñoz, R. Casellas, R. Martínez, N. Crespi, N. Mazzocca, E. Maini, *Manifesto of Edge ICT Fabric*, Proceedings of 17th International Conference on Intelligence in Next Generation Networks, pages 9-15, 2013