The Chip Is the Network:
Toward a Science of
Network-on-Chip Design
The Chip Is the Network: Toward a Science of Network-on-Chip Design

Radu Marculescu
Carnegie Mellon University
Pittsburgh, PA 15213
USA
radum@ece.cmu.edu

Paul Bogdan
Carnegie Mellon University
Pittsburgh, PA 15213
USA
pbogdan@ece.cmu.edu

now
the essence of knowledge

Boston – Delft
Editorial Scope

Foundations and Trends® in Electronic Design Automation will publish survey and tutorial articles in the following topics:

- System Level Design
- Behavioral Synthesis
- Logic Design
- Verification
- Test
- Physical Design
- Circuit Level Design
- Reconfigurable Systems
- Analog Design

Information for Librarians
Foundations and Trends® in Electronic Design Automation, 2007, Volume 2, 4 issues. ISSN paper version 1551-3939. ISSN online version 1551-3947. Also available as a combined paper and online subscription.
The Chip Is the Network: Toward a Science of Network-on-Chip Design

Radu Marculescu\(^1\) and Paul Bogdan\(^2\)

\(^1\) Department of Electrical and Computer Engineering, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213, USA, radum@ece.cmu.edu
\(^2\) Department of Electrical and Computer Engineering, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213, USA, pbogdan@ece.cmu.edu

Abstract

In this survey, we address the concept of network in three different contexts representing the deterministic, probabilistic, and statistical physics-inspired design paradigms. More precisely, we start by considering the natural representation of networks as graphs and discuss the main deterministic approaches to Network-on-Chip (NoC) design. Next, we introduce a probabilistic framework for network representation and optimization and present a few major approaches for NoC design proposed to date. Last but not least, we model the network as a thermodynamic system and discuss a statistical physics-based approach to characterize the network traffic. This formalism allows us to address the network concept in the most general context, point out the main limitations of the proposed solutions, and suggest a few open-ended problems.
Living in a world where the concept of network is ubiquitous, makes the quest for a science of network design inevitable. Informally speaking, the science of network design seeks to discover and explain the main properties affecting the network structure and behavior, as well as the mathematical models for predicting their dynamics and guiding their implementation.

The science of networks is primarily based on the graph theory, one of the most successful developments in mathematics ever. Indeed, the graph theory has led to an incredible number of real life applications that continue to grow even today. The graph theory originates in 1736 when Leonhard Euler offered the first rigorous solution to the “Seven Bridges of Konigsberg” problem. More precisely, by abstracting each landmass with a node (or vertex) and each bridge with an edge (link) (see Figure 1.1), Euler was able to show that a path that crosses all seven bridges in the city of Konigsberg only once cannot exist [18]. The newly proposed modeling paradigm based on sets of points linked by edges and represented via adjacency matrices (i.e., a matrix of zeros and ones, where each row in the matrix represents the connectivity of a node) has found many real-world applications. For instance, Cayley
graphs have been applied in the field of chemistry for studying various compound molecules [14]. Other applications of static graphs include the graph coloring with applications to register allocation, scheduling and compiler design; construction of trees with applications in designing the railway system; the problem of finding the optimal cycle in a graph (i.e., the traveling salesman problem) with application to VLSI design; computation of voltages and currents in electric circuits via Kirchhoff’s law [38].

Almost two centuries after the “Seven Bridges of Konigsberg” problem was solved, a new discovery in the graph theory, namely the random graphs proposed by Erdos and Renyi [68, 69], revolutionized the way we perceive real systems (e.g., rail, road, airplane, electronic networks, etc.). Random graphs are similar to regular graphs with the distinction that any edge between two arbitrary nodes is established with a certain probability. Consequently, the corresponding entry in the adjacency matrix is the probability $p \in [0, 1]$ that two nodes are connected.
(instead of a fixed 1 value which would correspond to the classical graph model). This way, a new bridge is created between graph and probability theories \[35\]; this introduces the possibility of naturally modeling social communication, collaboration or biological networks \[62, 140\].

Understanding the network behavior requires a deep analysis of the topology and pattern of communication among network components. Thus, a breakthrough in the field of graph theory took place in the 1990s, when physicists questioned the evolution and structure of real networks \[8, 71, 78\]. The fundamental contribution of this new body of work is the replacement of the static view of random graphs with a dynamic view based on probability distributions for node connectivity in the hope of finding the optimal design of any communication network.

In recent years, the level of understanding of networking concepts needed to design and control complex systems, has reached unprecedented peaks. As such, a holistic approach to the network paradigm is essential. Such an approach involves understanding the theoretical basis (e.g., graph theory, stochastic modeling and analysis), the essential properties (e.g., structure, dynamics, communication paradigm), and the metrics (e.g., energy, fault-tolerance, robustness) which are relevant to designing and characterizing different networks in either engineered or biological systems.

In order to put the network concept into the proper perspective, we need to step back and consider Milner’s fundamental work on calculus of communicating systems (also referred to as process algebra) \[132\], which enables the compact description of communication actions between any two agents. Inspired by the synergy between concurrency and communication, the design of electronic systems has moved recently from computation-based design to communication-based design. Indeed, nowadays, Systems-on-Chips (SoCs) represent true distributed systems at nanoscale where communication aspects dominate. From a technology point of view, this paradigm shift is intended to mitigate the problem of interconnects, keep the design complexity under control, and reduce costs. Since none of the classical architectures based on point-to-point or bus-based communication scales
nicely in terms of power and performance figures, a networked architecture based on packet switching has been suggested for future multicore systems \[58\].

Starting from these overarching ideas, in this survey we address the concept of network in Multiprocessor Systems-on-Chip (MPSoCs) and identify specific design principles and optimization techniques that are relevant to the design automation research community. Understanding the structure and behavior of seemingly different networks is crucial for our ability to master complex behaviors that characterize the emergent application domains. For instance, for SoCs, it has been suggested to replace the global interconnects with packet switching communication via the Network-on-Chip (NoC) paradigm \[22, 30, 87, 88, 102\] and then avoid the interconnect performance issues \[91, 115\], while allowing for higher level of fault-tolerance in Deep Submicron (DSM) technologies \[24, 167\]. Several concrete NoC architectures have been investigated in the literature \[61, 116, 198, 201\], as well as energy-efficient Globally Asynchronous Locally Synchronous (GALS) designs \[20, 21, 42, 181, 212\].

The emergent NoC communication paradigm consists of exchanging packets of information among various nodes in the network. As shown in Figure 1.2, each node consists of a core (e.g., DSP or CPU modules, video processors, embedded memory blocks or application specific Processing Elements (PEs)) and a router meant to forward the incoming packets toward the appropriate destination according to the header information. To support the inter-tile communication, each core has embedded input and output buffers to temporarily store the incoming packets from the neighboring nodes in the network. For instance, a packet generated at source (1,1) that needs to be delivered to destination (2,3) via a static XY routing, is first sent from the local PE to its associated router at tile (1,1); then, at each intermediate node, a routing decision is made based on the header information as shown with the dotted arrows in Figure 1.2 for the shortest source–destination path. To avoid the stalling of information flow due to the dependencies on network resources (i.e., shared routing paths), the concept of virtual channels (VCs) has been introduced as well \[57, 114, 133, 131\]. VCs share dedicated links and provide multiple buffers for each channel.
Fig. 1.2 A 3×3 mesh Network-on-Chip (NoC) architecture consisting of Processing Elements (PEs), routers, and interconnection links. Flows $F_1$ and $F_2$ are used to depict a stalling problem along the path from node (2,1) to (2,3) which can be solved by utilizing virtual channels.

For instance, the problem of stalling the flow $F_1$ from (1,1) to (2,3) due to a flow $F_2$ from (3,1) to (2,3), is solved by reserving the VCs between nodes (2,1), (2,2), and (2,3) for $F_1$.

A natural question now is what are the fundamental mathematical techniques that can be used to design, control, and optimize such networks in a rigorous manner. One such powerful technique is the linear programming (LP) approach used to solve maximum flow problems [54]. The goal of the linear program is to find a legal flow assignment of the edges of a given graph satisfying the flow conservation constraints. Another mathematical programming approach, namely the quadratic programming (QP) was proposed for solving the max-cut
problem \[5,4,86\]. Due to the requirement of discrete values, many real world optimization problems need an integer analysis done either via integer or mixed integer linear programming \[156\]. For small problems, efficient algorithms such as branch-and-bound or LP-relaxation have been proposed. However, for large problems, the solution is still based on heuristics.

Early studies of the network dynamics use queueing theory first developed by Markov \[128\] and Erlang \[70\], formalized by Kolmogorov \[112\] and Kendall \[104\], and extensively used in the context of packet switching networks by Kleinrock \[110\]. Nonetheless, in order to ease the mathematical tractability, most queueing approaches rely on exponential type distributions for the event/job arrival and/or service time. As shown in several studies, this may not be an accurate model for the real network traffic. Consequently, one of our objectives in this survey is to also highlight alternatives to the conventional paradigm of network design. This new vision is based on rigorous developments in the field of statistical physics and information theory that allow us to model the network as a thermodynamical system. The hope is that this new modeling paradigm enables not only capturing the intrinsic interactions among various network components, but also helps proposing more powerful techniques for predicting and optimizing the network.

These ideas are detailed in the remainder of this survey. More precisely, Section 2 presents a graph-based formalism for the network design and the major approaches proposed in a deterministic setup. Section 3 takes a probabilistic view on designing NoC architectures under the Markovian assumption. Finally, Section 4 introduces new problems in a statistical physics-based context for network design and enumerates some preliminary steps taken toward solving these problems.
References


References


Full text available at: http://dx.doi.org/10.1561/1500000011


References


References

References


References


References


References


References


Full text available at: http://dx.doi.org/10.1561/1500000011
References


References


