J Supercomput (2012) 59:779–810
DOI 10.1007/s11227-010-0472-5

Stochastic communication for application-specific Networks-on-Chip

Nitin · Durg Singh Chauhan

Published online: 4 September 2010

Abstract In this paper, we have developed analytical stochastic communication technique for inter and intra-Networks-on-Chip (NoC) communication. It not only separates the computation and communication in Networks-in-Package (NiP) but also predicts the communication performance. Moreover, it will help in tracking of the lost data packets and their exact location during the communication. Further, the proposed technique helps in building the Closed Donor Controlled Based Compartmental Model, which helps in building Stochastic Model of NoC and NiP. This model helps in computing the transition probabilities, latency, and data flow from one IP to other IP in a NoC and among NoCs in NiP. From the simulation results, it is observed that the transient and steady state response of transition probabilities give state of data flow latencies among the different IPs in NoC and among the compartments of NoCs in NiP. Furthermore, the proposed technique produces low latency as compared to the latencies being produced by the existing topologies.

Keywords Networks-on-Chip · Stochastic communication and model · Closed donor controlled compartmental modeling · Transition probabilities · Latency

Nitin (✉)
Visiting Faculty, College of Information Science and Technology, The Peter Kiewit Institute, University of Nebraska at Omaha, 6001 Dodge Street, Omaha, NE 68182, USA
e-mail: fnunitin@mail.unomaha.edu

Nitin
e-mail: delnitin@ufl.edu

D.S. Chauhan
Uttarakhand Technical University, Dehradun 248001, Uttarakhand, India
e-mail: pdschauhan@gmail.com

1 Introduction and Motivation

Systems-on-chip (SoC) design provides an integrated solution for challenging design problems appearing in the telecommunications, multimedia, and consumer electronic domains. With Deep Sub-Micron (DSM) technology, chip designers are expected to create SoC solutions by connecting different Intellectual Property (IP) using efficient and reliable interconnection schemes known as Networks-on-Chip (NoC). This technique makes a clear distinction between computation (the tasks performed by the IPs) and communication (the interconnect between the IPs).

A NoC is formed by connecting either homogeneous or heterogeneous IPs on a single chip. Since modern NoCs are becoming extremely complex, therefore, there are many challenges in this new area of research. One of the major challenges, which we are concentrating on in this paper, is On-chip wire delays or latency which has become more critical than gate delays, and recently the synchronization problem between the IPs are more apparent. This trend worsens more as the clock frequency increases and the feature size decreases [1]. However, the problem of high latency remains an open question and important factor in real time applications [2, 3]. This problem worsens more during parallel communication among NoCs using Interconnects-on-Chip (IoC) as suggested in [4–7] and initiated new types of malfunctions and failures such as loss of data packets and its information (location).

Latency in NoC can be measured by calculating the latency in switches and propagation delay in IoC [8]; however, it depends on the NoC topology considered. The different NoC topologies already in used are discussed in [9] and they provide different communication structure in NoC as discussed in [10]. The issue of latency, lost data packets, and their exact location during parallel communication cannot be characterized using deterministic measurements and, therefore, we have used the probabilistic technique called Stochastic Communication.

Since 2003, various researchers have already developed stochastic communication technique for on-chip [11–13] or inter-NoC (local or communication within the IPs) communication and reported in [3, 14, 15] with its drawbacks comprehensively discussed in [14]. However, the stochastic communication technique for intra- (global or among different NoCs in NiP) NoC communication was ignored and remained out of the lime light. Therefore, in this paper, we have developed a stochastic communication technique for intra-NoC communication. It not only separates the computation and communication in NiP but also predicts the communication performance.

The communication structure in NiP is divided into two zones; one is On-chip (inter or local) communication and other is On-package (intra or global) communication. In order to set up stochastic communication in a local and global zone, a Closed Donor Controlled Based Compartmental Model is used. This model treats IPs as compartmental IPs, which represents the flow of data from source IP to Destination IP and helps in deriving the compartmental matrix. This matrix is converted into the transition probability matrix in order to obtain Absorbing Markov Chain for Stochastic Modeling. This modeling is very useful to calculate the latency, transition probabilities, and expected time of data flow from one IP to other IP in a NoC and among NoCs in NiP.

The rest of the work is as follows: Sect. 2 introduces the various terms and technologies used throughout this paper. Section 3 discusses the data flow networks in

NoC followed by its compartmental and stochastic modeling, respectively, in Sect. 4 and Sect. 5. The simulation results of stochastic modeling of NoC are discussed in Sect. 6. From Sect. 7 to Sect. 9, we have discussed the stochastic modeling of NiP as a case study followed by the conclusion.

2 Preliminaries and background

2.1 Systems-on-chip

SoC refers to integrate all components of a computer or other electronic system into a single integrated circuit [16]. It may contain digital, analog, mixed-signal, and often radio-frequency functions all on one chip. A typical application is in the area of embedded systems. SoC is believed to be more cost effective since it increases the yield of the fabrication and because its packaging is simpler.

2.2 Networks-on-chip

NoC is a new approach to SoC design [1, 17, 18]. NoC-based systems can accommodate multiple asynchronous clocking that many of today's complex SoC designs use. The NoC solution brings a networking method to on-chip communication and brings notable improvements over conventional bus systems. It is an emerging paradigm for communications within the large VLSI systems implemented on a single silicon chip [16].

In a NoC system, components such as processor, memories, and specialized IP blocks exchanges data using a network. It is constructed using multiple point-to-point data links interconnected by switches, such that messages can be relayed from any source module to any destination module over several links, by making routing decisions at the switches. Moreover, it is similar to a modern telecommunications network, using digital bit-packet switching over multiplexed links. Although packet switching is sometimes claimed as necessity for a NoC, however, there are several NoC proposals utilizing circuit-switching techniques. In this paper, we are concentrating on Application-Specific NoCs [18], which have been designed for providing specific functionality.

Further, a NoC may consist of different or similar IPs and these communicate through a local communication network. First, data is processed by the first IP and then it is communicated to other IP. Finally, the processed data in a NoC is communicated to the other NoC through global interconnect on the (same) NiP. A NoC separates the local communication from the IP computation and NiP separates the global communication from NoC computation (local communication).

2.3 Networks-in-package

NiP is a solution for intra-NoC communication, where power dissipation, performance, and size of processing package are the major concerns. It is a group of NoCs mounted on a single package. Its design provides integrated solutions to challenging design problems in the field of multimedia and real time embedded applications. Moreover, NiP is a platform that interlinks different applications, which are running on different NoCs to execute a real time task. It links the processes executed under different NoCs through a Multi-stage Interconnection Network (MIN). Specifically, modern NiP structure composed of different Application-Specific NoCs; and these are allowed to communicate with each other through a multistage interconnection network (MIN) [5] in order to execute the mission critical.

2.3.1 Multi-stage Interconnection Network

MINs are originated from the design of high performance parallel computers. A major factor that differentiates the modern multiprocessor architectures is topology, routing strategy, and switching technique. Moreover, MINs are building up of switching elements; topology is the pattern in which the individual switches are connected to other elements, like processors, memories and other switches. In this paper, MIN has been used as a IoC [5–7].

2.4 Stochastic

Generally, the word stochastic describes an approach to anything that is based on probability. In mathematics, a stochastic approach is one in which values are obtained from a corresponding sequence of jointly distributed random variables. Classical example: guessing the amount of water in a reservoir based on the random distribution of rainfall and water usage [18, 19].

2.4.1 Stochastic process

A stochastic process [19, 20], or sometimes random process, is the counterpart to a deterministic process in probability theory. Instead of dealing only with one possible "reality" of how the process might evolve over time, in a stochastic or random process there is some indeterminacy in its future evolution described by probability distributions. This means that even if the initial condition is known, there are many possibilities the process might go to, but some paths are more probable and others less [20].

2.4.2 Stochastic communication

The Stochastic Communication [3, 14, 15, 21] is a novel communication paradigm for SoC. This scheme separates communication from computation by allowing a IoC to be designed as a reusable IP, which provides a built-in tolerance to Deep Submicron (DSM) failures without a significant performance penalty. By using this communication scheme, a large percentage of data upsets, packet losses due to buffer overflows, and severe levels of synchronization failures can be tolerated, while providing high levels of performance.

2.5 Application-specific NiP architecture

The general architecture of Application-Specific NiP resembles with the Open Systems Interconnection (OSI) Model. The Physical Layer refers to the electric details of wires, the circuits and techniques to drive information, while the Data Link Layer ensures a reliable transfer and deals with Medium Access Sublayer. At the Network Layer, there are issues related to the topology and the consequent routing scheme, while the Transport Layer manages the end-to-end services and the packet segmentation/reassembly. Upper Layers can be viewed (merged up to the Application) as a sort of Adaptation Layer, which implements services in hardware. Further, it provides the common interface through that all NoCs communicate together more efficiently and robustly. It has three different types of building blocks [10] and they are as follows:

1. The on-chip network interface (N/W IF) connects the IP core (present inside NoC-1) with on-chip network router. Further, off-chip N/W IF connects the entire NoC-1 with other NoC (which is present on the same package) using IoC.
2. The router is responsible for the flow of data traffic across the network and for the Quality of Service (QoS) offered by the network.
3. The physical link is responsible for the actual propagation of the signals across the network and to/from the external IP and the subsystems.

Figures 1, 2 show a general NiP architecture where four NoC chips are mounted on a single package. These NoCs communicate with each other through an intermediate chip, called as IoC [11–14]. Figure 3, describes the internal architecture of NoC-1 used in NiP.

2.5.1 Interconnects-on-Chip architecture

The architectural design of IoC is similar to a small MIN, widely used for broadband switching technology and for multiprocessor systems. Besides this, it offers an enthusiastic way of implementing switches/routers used in data communication networks. With the performance requirement of the switches/routers exceeding several terabits/sec and teraflops/sec, it becomes imperative to make them dynamic and fault-tolerant. The typical modern day application of the MINs includes fault-tolerant packet switches, designing unicast, and multicast router fabrics [22–33].

Figures 4, 5 show the different types of IoC i.e. one belongs to the class of irregular fault-tolerant MIN and other belongs to the class of regular fault-tolerant MIN. This first IoC, shown in Fig. 4 consists of five routers; working as Switching Elements (SE) is known as PNN and Fig. 5 shows the architecture of HXN with 6 SE [4–7]. These routers are connected together through a links. Some of the links are known as chaining or express links, which makes the IoC highly, fault tolerant.

3 Data flow network in NoC for stochastic communication

In this section, the compartmental-based probabilistic data broadcasting, among the IPs is suggested. This broadcasting is a random process, which is similar to the randomized gossip protocols as discussed in [3]. When the data (in the form of packets) transmitted from source IP to destination IP through its neighbors in the grid based square NoC as shown in Fig. 6, then IP communicates with this data using proposed probabilistic broadcasting scheme. It is a well-known fact that in a NoC, any IP can be used as the source IP or intermediate IP or destination IP and the data can flow in many possible ways (depending upon the requirements).

Figure 7, shows one of the data flow network for Application-Specific NoC, which consist of few IPs and routers. The Network Adapter (or Interface) implements the interface, which help to connect one IP to other IP on the same NoC. Its function is to

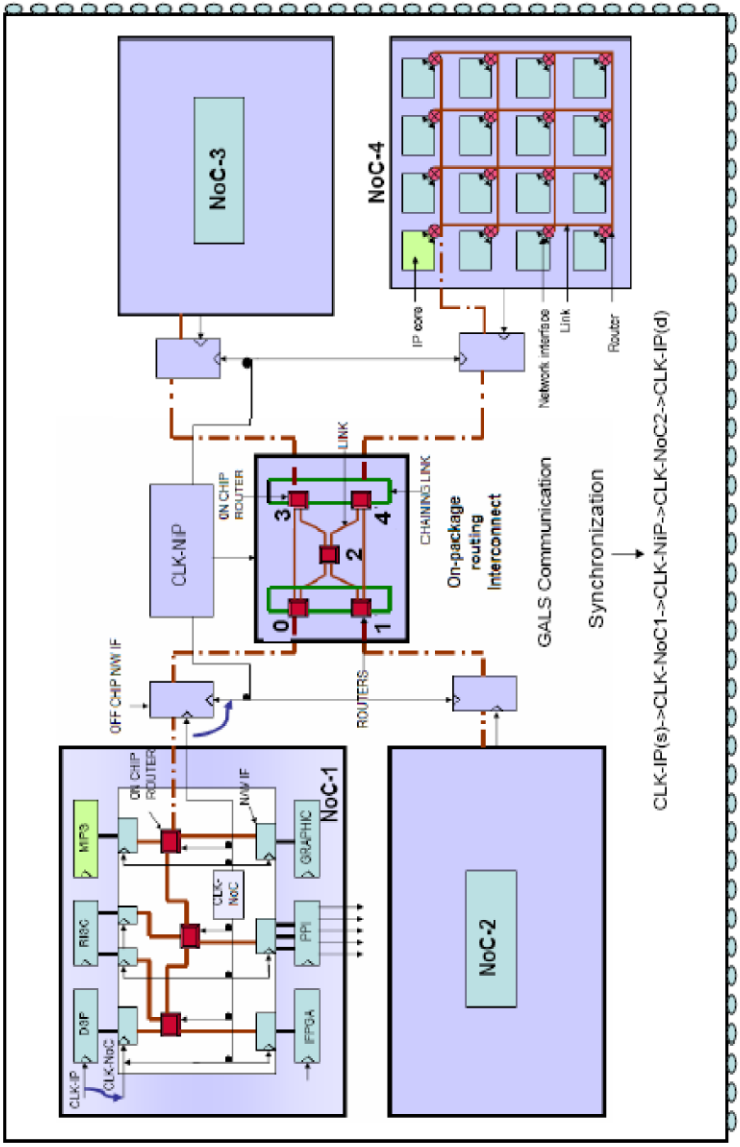


Fig. 1 Parallel communication among various NoCs using PNN

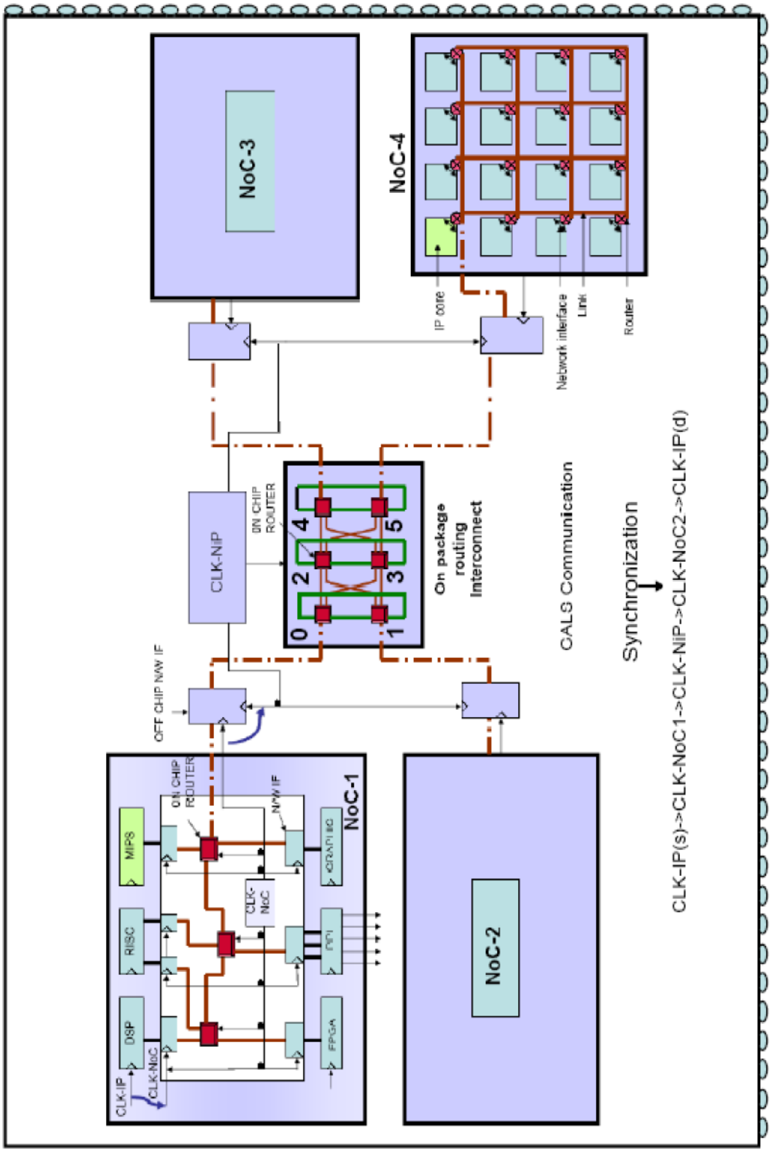


Fig. 2 Parallel communication among various NoCs using HXN

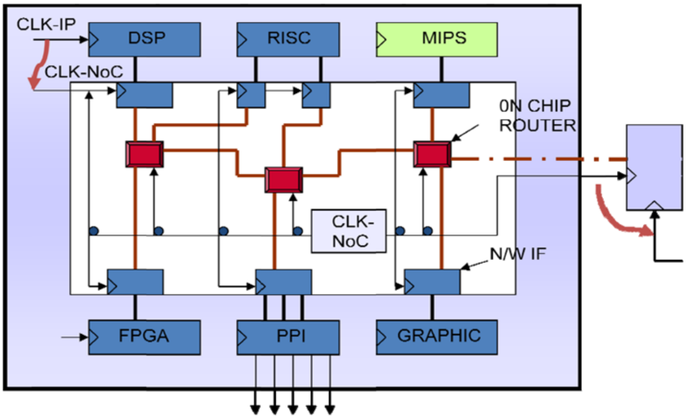


Fig. 3 Internal architecture of NoC-1 used in NiP

Fig. 4 PNN as Interconnect-on-Chip

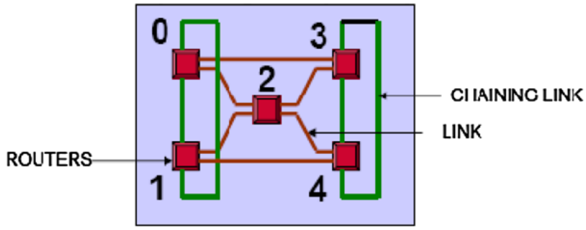
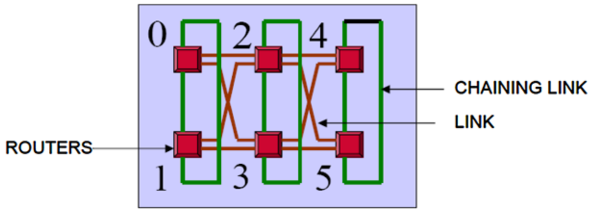


Fig. 5 HXN as Interconnect-on-Chip



decouple computation (the IPs) from communication (the network). Router routes the data according to chosen protocol and implements the routing strategy. Links connect the nodes, providing the raw bandwidth and may consist of one or more logical or physical channels [10]. Here, Fig. 7 is used again to build the scenario for inter-NoC stochastic communication (more specifically IP to IP communication): If the data has to be sent from Digital Signal Processor (DSP) to Field-programmable Gate Array (FPGA) and Processing Unit (PU) then one of the data flow network from NoC can be extract and can be classified in to five compartments. These five compartments are as follows:

1. Source IP is represented using (X_1),
2. Intermediate IPs are represented using $(X_2$ and $X_3)$ and
3. Destination IPs are represented using $(X_4$ and $X_5)$.

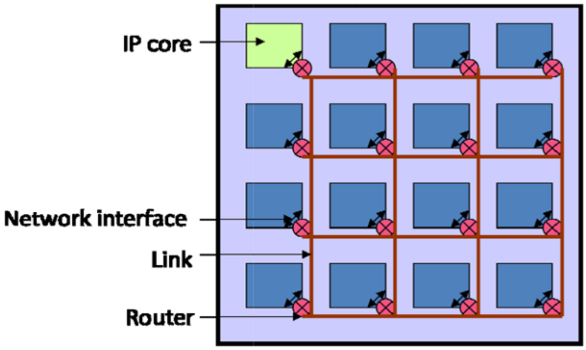


Fig. 6 Topological illustration of a 4-by-4 grid structured homogeneous NoC

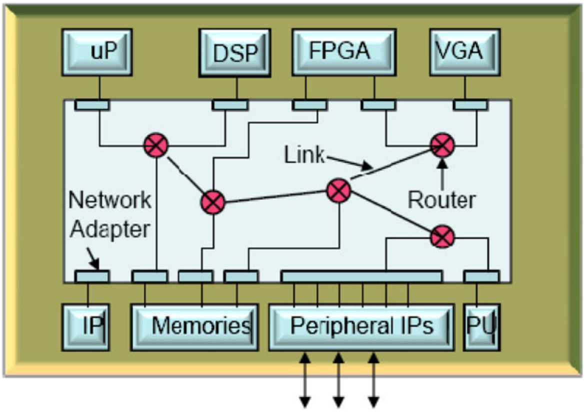


Fig. 7 Application-Specific NoC

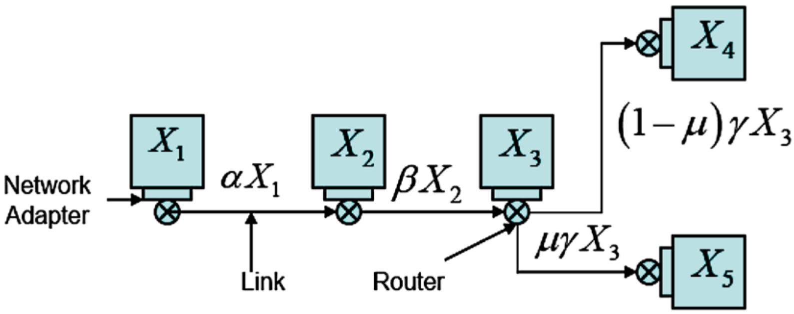


Fig. 8 Data flow network for stochastic communication

The data flow network shown in Fig. 8 is also known as stochastic network and can be used for stochastic modeling by following certain assumptions:

1. The total number of data packets is constant.
2. The model is closed donor controlled based model.
3. The model is mass conservative.

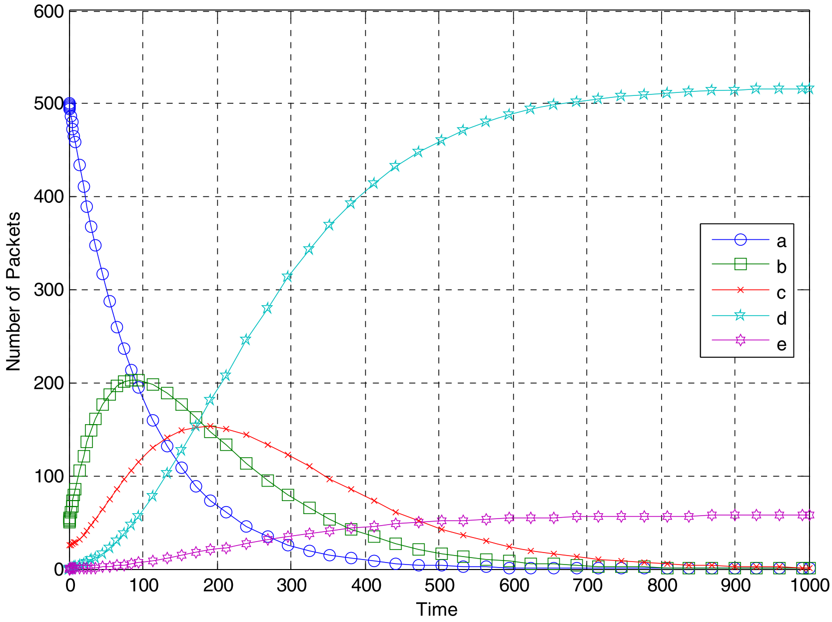


Fig. 9 Dynamic behavior of data flow network in NoC. Here, the graph legends are coded as follows: a = X_1, b = X_2, c = X_3, d = X_4 and e = X_5

The behavior of data flow network for stochastic communication, shown in Fig. 8 can be further described by the following set of differential equations:

$$\frac{dX_1}{dt} = -\alpha X_1 \tag{1}$$

$$\frac{dX_2}{dt} = \alpha X_1 - \beta X_2 \tag{2}$$

$$\frac{dX_3}{dt} = \beta X_2 - \gamma X_3 \tag{3}$$

$$\frac{dX_4}{dt} = (1 - \mu)\gamma X_3 \tag{4}$$

$$\frac{dX_5}{dt} = \mu\gamma X_3 \tag{5}$$

$$N = X_1 + X_2 + X_3 + X_4 + X_5 \tag{6}$$

Where α, β and γ are the different data flow rates from respective compartment.

Figure 9 shows the dynamic behavior (number of packets transferred per unit of time) of basic data flow network in NoC for $N = 575$, $X_1(0) = 500$, $X_2(0) = 50$, $X_3(0) = 25$ and $X_4(0) = X_5(0) = 0$, where N is total number of data packets to be transmitted. For on-chip synchronization, all the flow rates are taken equal, i.e., $\alpha = \beta = \gamma = 0.01$ and the value of separation constant μ is 0.1. Equations (1)–(6) describe the behavior of stochastic network and the five compartments $(X_1 - X_5)$

appearing in these equations are treated as physical state space variables. Since these equations are linear differential equations, thus the homogeneous solution for these can be derived further.

4 Compartmental modeling of data flow network in NoC

In this section, the compartmental matrix from the state space equations (1)–(6) is derived. These equations define the dynamic behavior of data flow networks as shown in Fig. 9 and can be further expressed in the form of matrix as given below.

$$\dot{X}(t) = AX(t) \tag{7}$$

$$\begin{pmatrix} \dot{X}_1 \\ \dot{X}_2 \\ \dot{X}_3 \\ \dot{X}_4 \\ \dot{X}_5 \end{pmatrix} = \begin{pmatrix} -\alpha & 0 & 0 & 0 & 0 \\ \alpha & -\beta & 0 & 0 & 0 \\ 0 & \beta & -\gamma & 0 & 0 \\ 0 & 0 & (1-\mu)\gamma & 0 & 0 \\ 0 & 0 & \mu\gamma & 0 & 0 \end{pmatrix} \begin{pmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \\ X_5 \end{pmatrix} \tag{8}$$

$$A = \begin{pmatrix} -\alpha & 0 & 0 & 0 & 0 \\ \alpha & -\beta & 0 & 0 & 0 \\ 0 & \beta & -\gamma & 0 & 0 \\ 0 & 0 & (1-\mu)\gamma & 0 & 0 \\ 0 & 0 & \mu\gamma & 0 & 0 \end{pmatrix} \tag{9}$$

where A, represents a compartmental matrix. The solution of homogeneous state (7) is as follows:

$$X(t) = e^{At} X(0) \tag{10}$$

$$X(t) = \begin{pmatrix} X_1(t) \\ X_2(t) \\ X_3(t) \\ X_4(t) \\ X_5(t) \end{pmatrix} \tag{11}$$

$$e^{At} = L^{-1}\big[(sI - A)^{-1}\big] \tag{12}$$

or

$$e^{At} = I + At + \frac{1}{2!}A^2t^2 + \cdots + \frac{1}{i!}A^i t^i \tag{13}$$

where e^{At} represents the state transition matrix of data flow network, $X(0)$ is the column matrix (which shows the initial conditions of the stochastic model) and I is the identity matrix.

4.1 Properties of compartmental matrix

The certain important properties of compartmental matrix hold by the matrix A are given below:

1. The diagonal elements of compartmental matrix are zero or negative elements.
2. The nondiagonal elements of compartmental matrix are zero or positive.
3. The first Eigen value of compartmental matrix is zero.
4. The sum of elements in each column of compartmental matrix is equal to zero.
5. Compartmental matrix is a Metzler matrix.
6. It obeys the law of mass conservation.

5 Stochastic modeling of data flow network in NoC

In this section, the compartmental matrix A is converted into the transition probability matrix P in order to obtain Absorbing Markov Chain for stochastic modeling. This modeling is very useful to calculate the latency, transition probabilities, and expected time of data flow from one IP to other IP in a NoC and among NoCs in NiP.

Before we proceed to obtain transition probability matrix P, we need the help of derivation of (14) is already explained and provided by the authors of [34]. The same proof (in brief) is provided here to provide more clarity and simplifying the calculations clear and understandable.

Definition 1 The transition probability matrix can be derived from compartmental matrix using following relation:

$$P = (I + hA)^T \tag{14}$$

Proof The probability $p_i(n)$ that the random variable is in state i at any time n may be found from the level of numbers or quantity of random variables $x_i(n)$ in that state (now called compartment) at time n. Indeed, $p_i(n) = x_i(n)/\sum_{j=1}^{k} x_j(n)$, where k denotes the number of states and the levels at time $n+1$ are given in terms of those at time n by the same equation,

$$X_{n+1}^T = X_n^T P, \quad n = 0, 1, 2, \ldots, \tag{15}$$

as the probabilities. Here, X_n is a column vector of material levels. Then

$$X_{n+1}^T \begin{bmatrix} 1 \\ 1 \\ . \\ 1 \end{bmatrix} = \left[x_{n+1,1}, x_{n+1,2}, \ldots, x_{n+1,k}\right] . \begin{bmatrix} 1 \\ 1 \\ . \\ 1 \end{bmatrix} = X_n^T P \begin{bmatrix} 1 \\ 1 \\ . \\ 1 \end{bmatrix} = X_n^T \begin{bmatrix} 1 \\ 1 \\ . \\ 1 \end{bmatrix} \tag{16}$$

since $[1, 1, \ldots, 1]^T$ is always a right eigenvector corresponding to the steady state Eigen value of 1 of P. If started with a quantity $q = \sum_{j=1}^{n} x_j(0)$ of materials in the system, then the total quantity in the system remains at q for all time by (16). Thus,

$p_i(0) = \frac{x_i(0)}{q.}$ and (15) is one form of equation of a compartmental system, but a more common format is as a difference equation

$$X_{n+1}^T - X_n^T = X_n^T (P - I)$$

Alternatively, by taking transpose, it becomes

$$\Delta X_n = (P^T - I)X_n \tag{17}$$

If the time step, i.e., the time between trials, is h rather then 1, then $X_n = X(nh)$ and the left side of (17) is replaced by the difference quotient, i.e.,

$$\frac{X(nh+h) - X(nh)}{h} = \frac{1}{h}(P^T - I)X(nh) = AX(nh) \tag{18}$$

Let

$$\begin{aligned} t &= nh \\ \Rightarrow \frac{X(t+h) - X(t)}{h} &= AX(t) \end{aligned} \tag{19}$$

This left side is approximately the derivative, therefore, $X' = AX$. This is the differential equation for the compartmental matrix and hence $A = \frac{1}{h}(P^T - I)$, where P the transition probability matrix and h is the time between events or trials or more specifically $P = (I + hA)^T$. □

Now put the values of (9) in (14) and it will result in the following matrix P as follows:

$$P = \begin{pmatrix} 1-h\alpha & 0 & 0 & 0 & 0 \\ h\alpha & 1-h\beta & 0 & 0 & 0 \\ 0 & h\beta & 1-h\gamma & 0 & 0 \\ 0 & 0 & h(1-\mu)\gamma & 1 & 0 \\ 0 & 0 & h\mu\gamma & 0 & 1 \end{pmatrix}^T \tag{20}$$

$$P = \begin{pmatrix} p_{x_1x_1} & p_{x_1x_2} & p_{x_1x_3} & p_{x_1x_4} & p_{x_1x_5} \\ p_{x_2x_1} & p_{x_2x_2} & p_{x_2x_3} & p_{x_2x_4} & p_{x_2x_5} \\ p_{x_3x_1} & p_{x_3x_2} & p_{x_3x_3} & p_{x_3x_4} & p_{x_3x_5} \\ p_{x_4x_1} & p_{x_4x_2} & p_{x_4x_3} & p_{x_4x_4} & p_{x_4x_5} \\ p_{x_5x_1} & p_{x_5x_2} & p_{x_5x_3} & p_{x_5x_4} & p_{x_5x_5} \end{pmatrix} \tag{21}$$

$$P = \begin{pmatrix} 1-h\alpha & h\alpha & 0 & 0 & 0 \\ 0 & 1-h\beta & h\beta & 0 & 0 \\ 0 & 0 & 1-h\gamma & h(1-\mu)\gamma & h\mu\gamma \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{22}$$

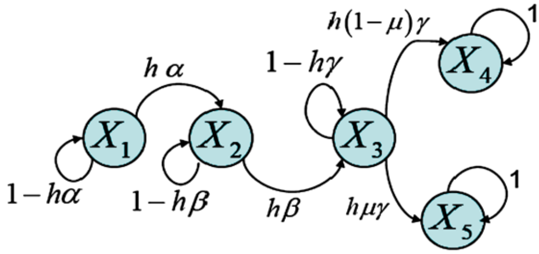


Fig. 10 Stochastic diagraph (Absorbing Markov Chain) of data flow network in NoC

From (22), it is understood that sum of all the elements in each row of transition probability matrix P is equal to 1. Hence,

$$\sum_{j=1}^{5} p_{ij} = 1 \quad \text{where } i, j = 1, \ldots, 5 \tag{23}$$

5.1 Properties of transition probability matrix

The certain important properties of transition probability matrix P are given below:

1. The first eigen value of transition probability matrix is equal to 1.
2. The sum of all elements in each row of transition probability matrix is equal to 1.
3. This matrix is also known as Markov matrix.

5.2 Deriving Markov chain from transition probability matrix

In this section, we have derived the absorbing Markov chain for stochastic modeling from the transition probability matrix P. Figure 10 represents the stochastic diagraph of transition probability matrix P.

Before we proceed to obtain the absorbing *Markov* chain, we need the help of explanation of (24) is already explained and provided by the authors of [34] and some portions are further discussed by the authors of [35–37]. The same explanation (in brief) is provided here to provide more clarity and simplifying the calculations clear and understandable.

Definition 2 In an Absorbing Markov Chain with states ordered such that the transition probability matrix P has the form:

$$P = \begin{bmatrix} Q & R \\ 0 & I \end{bmatrix} \tag{24}$$

In addition, the following holds:

1. $Q^t \to 0$ as $t \to \infty$.
2. $R_\infty = (I - Q)^{-1} R$.
3. The expected number of times a chain is in the non-absorbing state k_j given that it started in k_i is given by the corresponding element of $(I - Q)^{-1}$.

The matrix $(I - Q)^{-1}$ is often referred as Markov Chain Fundamental Matrix for each nonabsorbing state, there is an absorbing state with a path of minimum length. Let

r be the maximum length of all such paths. Therefore, in r steps, there is a positive probability p of entering one of the absorbing states regardless of where you started. The probability of not reaching an absorbing state in r steps is $(I-p)$. After the next r steps, it is $(I-p)^2$ and after kr steps, it is $(I-p)^k$. Since this approaches 0 as $k \to \infty$, the probability of being in any non absorbing state approaches 0 as $t \to \infty$. However, the elements of Q^t are just these probabilities. Here, $(I-Q)^{-1}$ will give us the expected time of data flow from one IP to other IP. Moreover, $R_\infty = (I-Q)^{-1}R$ will give us the probability of data transmission to the destination IP.

5.3 Stochastic analysis of NoC communication

In this section, the compartmental-based stochastic communication scheme is verified using (22) and (24). Hence, the required matrix P is as follows:

$$P = \begin{bmatrix} Q & R \\ 0 & I \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} 1-h\alpha & h\alpha & 0 \\ 0 & 1-h\beta & h\beta \\ 0 & 0 & 1-h\gamma \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ h(1-\mu)\gamma & h\mu\gamma \end{bmatrix} \\ \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} & \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \end{bmatrix} \tag{25}$$

The last state of this Markov chain I is the absorbing state, which consists of destination IP in NoC. For $\alpha = \beta = \gamma = 0.01$ and μ is 0.1. The time for each event or transition h is 0.1. This implies

$$P = \begin{bmatrix} 0.999 & 0.001 & 0 & 0 & 0 \\ 0 & 0.999 & 0.001 & 0 & 0 \\ 0 & 0 & 0.999 & 0.0009 & 0.0001 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{26}$$

$$Q = \begin{bmatrix} 0.999 & 0.001 & 0 \\ 0 & 0.999 & 0.001 \\ 0 & 0 & 0.999 \end{bmatrix}, \quad R = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0.0009 & 0.0001 \end{bmatrix} \text{ and } I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{27}$$

For transient response,

$$(I-Q)^{-1} = \begin{bmatrix} 1000 & 1000 & 1000 \\ 0 & 1000 & 1000 \\ 0 & 0 & 1000 \end{bmatrix} \tag{28}$$

The $(I-Q)^{-1}$ helps to calculate the following transition probabilities:

1. Expected time during which the data is available with source IP(X_1) = 1000 nanoseconds.

2. Expected delay to reach the intermediate IP(X_3) $= 1000 + 1000 = 2000$ nanoseconds.
3. Expected time during which the data is live on intermediate IP(X_3) $= 1000$ nanoseconds.
4. Expected delay to reach from intermediate IP(X_2) to the destination IP(X_4) $= 1000 + 1000 = 2000$ nanoseconds.
5. Expected delay to reach from source IP(X_1) to the destination IP(X_4) $= 1000 + 1000 + 1000 = 3000$ nanoseconds.

For steady state response

$$R_\infty = (1 - Q)^{-1} R = \begin{bmatrix} 0.9 & 0.1 \\ 0.9 & 0.1 \\ 0.9 & 0.1 \end{bmatrix} \tag{29}$$

From $R_\infty = (1 - Q)^{-1} R$ matrix, the following can be calculated:

1. Probability of data reception by IP $X_4 = 0.9$
2. Probability of data reception by IP $X_5 = 0.1$

For the steady state, complete transition probability matrix is

$$P_\infty = \begin{bmatrix} Q_\infty & R_\infty \\ 0 & I \end{bmatrix} \tag{30}$$

$$P_\infty = \begin{bmatrix} 0 & 0 & 0 & 0.9 & 0.1 \\ 0 & 0 & 0 & 0.9 & 0.1 \\ 0 & 0 & 0 & 0.9 & 0.1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{31}$$

The regular matrix P at ∞ will be a null matrix and it will have the certain probabilities to receive the data depending upon the value of the separation constant h.

6 Simulation and results

Figures 11, 12, 13, Tables 1–3 and $P(X_i X_j)$ shows the transition probabilities of data flow from one (X_i) IP to (X_j) IP where $i = 1, \ldots, 3$ and $j = 1, \ldots, 5$. From this stochastic model, one can calculate the total transition probabilities between any two IPs and further, it will help in calculating the latency. In addition, the proposed technique makes separation between communication and computation in NoC. Furthermore:

1. It gives the information about the data packets propagation.
2. It provides the directional movement of data packet, i.e., when the column entries of the transition probabilities table are 0 that implies data is no more available at that node.
3. It gives the information about the time during which the data is live across a load, i.e., the data will remain with the node until the transition probabilities of concern node becomes to zero.

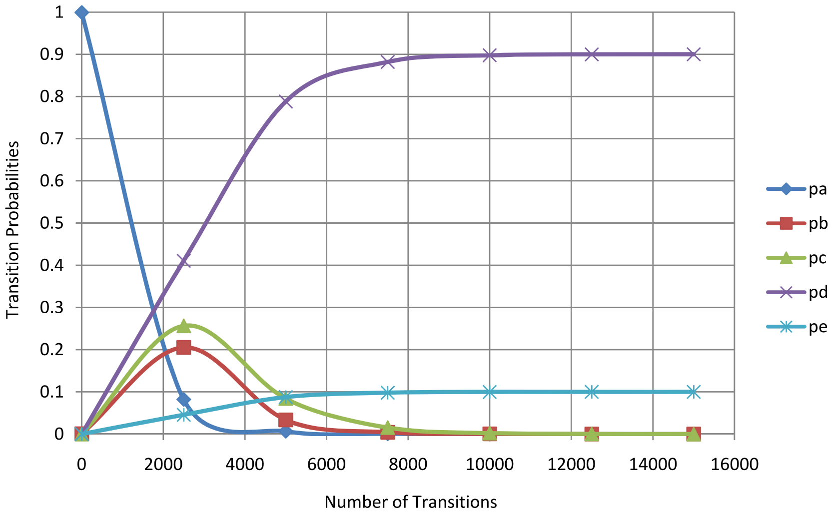


Fig. 11 Transition probabilities of data flow for IP(X_1). Here, the graph legends are coded as follows: pa = $p(X_1X_1)$, pb = $p(X_1X_2)$, pc = $p(X_1X_3)$, pd = $p(X_1X_4)$, and pe = $p(X_1X_5)$

4. It gives the status of data propagation along with their transition probabilities, i.e., the data can be traced or dragged across any intermediate node, while the transition probability is at maximum peak. It helps to use the data from intermediate SEs for real time interfacing applications.
5. These results (the combine output of step 1 to step 4) helps to predict the latency from node to node and from Input to Output using the formula: Latency = Number of transitions at which maximum transition probability appear × Transition interval.
6. This technique also fits in the different architecture of NoC where link length varies, consequently, the value of h varies accordingly.

7 Case study

7.1 Data flow network in NiP for stochastic communication

In this section, the compartmental-based probabilistic data broadcasting among the NoCs through HXN [5–7] is proposed. Again, this process of communication is a random process, which is similar to the randomized gossip protocols or epidemic process already explained in [3]. The data is transmitted (in the form of packets) from source IP to destination IP (should be mounted on different NoCs) using a probabilistic broadcast scheme. Specifically, the source IP sends the data packets to the destination IP through the off-chip N/W IF and HXN as shown in Fig. 14. Here, one of the data flow network in NiP is used and it consists of both heterogeneous and homogeneous NoCs. Here, Fig. 14 is used again to build the scenario for the intra

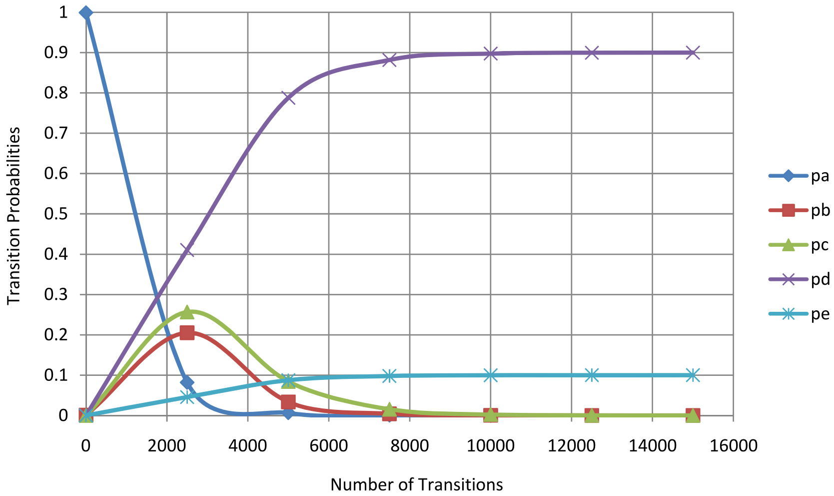


Fig. 12 Transition probabilities of data flow for IP(X_2). Here, the graph legends are coded as follows: pa = $p(X_2X_1)$, pb = $p(X_2X_2)$, pc = $p(X_2X_3)$, pd = $p(X_2X_4)$, and pe = $p(X_2X_5)$

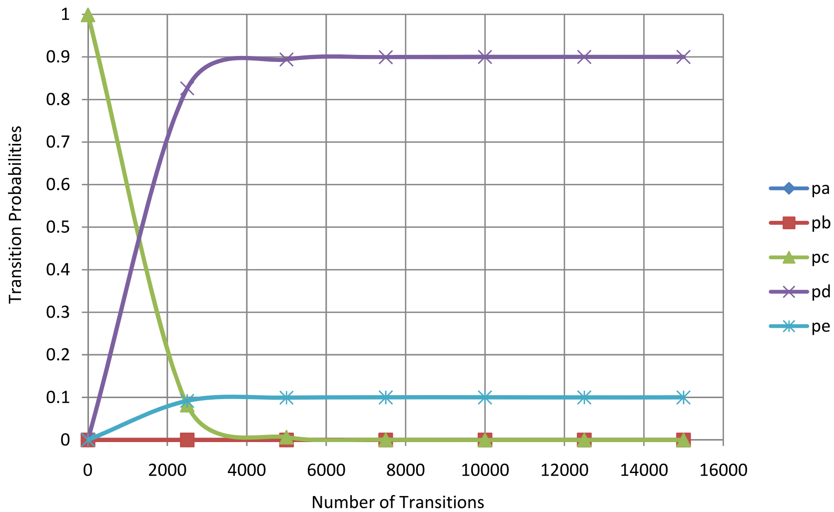


Fig. 13 Transition probabilities of data flow for IP(X_3). Here, the graph legends are coded as follows: pa = $p(X_3X_1)$, pb = $p(X_3X_2)$, pc = $p(X_3X_3)$, pd = $p(X_3X_4)$, and pe = $p(X_3X_5)$

NoC stochastic communication (more specifically NoC to NoC communication). Example: If the data has to be sent from Multiple Instructions Per Second (MIPS) IP, residing in NoC-1 (has a heterogeneous architecture) to IP in NoC-4 (has a homogeneous architecture then one of the data flow network from NiP can be extract and can be classified in to seven compartments. These seven compartments are as follows:

1. $X_1 \rightarrow$ MIPS IP in NoC-1

Table 1 Transition probabilities of data flow for IP(X_1)

Number of transitions	Transition probabilities				
	$p(X_1X_1)$	$p(X_1X_2)$	$p(X_1X_3)$	$p(X_1X_4)$	$p(X_1X_5)$
1	0.999	0.001	0	0	0
2500	0.082	0.2052	0.2566	0.4106	0.0456
5000	0.0067	0.0336	0.0842	0.7879	0.0875
7500	0.0006	0.0041	0.0155	0.8818	0.098
10,000	0	0.0005	0.0023	0.8975	0.0997
12,500	0	0	0.0003	0.8997	0.1
15,000	0	0	0	0.9	0.1

Table 2 Transition probabilities of data flow for IP(X_2)

Number of transitions	Transition probabilities				
	$p(X_2X_1)$	$p(X_2X_2)$	$p(X_2X_3)$	$p(X_2X_4)$	$p(X_2X_5)$
1	0	0.999	0.001	0	0
2500	0	0.082	0.2052	0.6416	0.0713
5000	0	0.0067	0.0336	0.8637	0.096
7500	0	0.0006	0.0041	0.8958	0.0995
10,000	0	0	0.0005	0.8996	0.1
12,500	0	0	0	0.9	0.1
15,000	0	0	0	0.9	0.1

Table 3 Transition probabilities of data flow for IP(X_3)

Number of transitions	Transition probabilities				
	$p(X_3X_1)$	$p(X_3X_2)$	$p(X_3X_3)$	$p(X_3X_4)$	$p(X_3X_5)$
1	0	0	0.999	0.0009	0.0001
2500	0	0	0.082	0.8262	0.0918
5000	0	0	0.0067	0.894	0.0993
7500	0	0	0.0006	0.8995	0.0999
10,000	0	0	0	0.9	0.1
12,500	0	0	0	0.9	0.1
15,000	0	0	0	0.9	0.1

2. $Y_2 \rightarrow$ Off-chip N/W IF
3. $Z_3 \rightarrow$ HXN on-chip router
4. $Z_4 \rightarrow$ HXN on-chip router
5. $Z_5 \rightarrow$ HXN on-chip router
6. $Y_6 \rightarrow$ Off-chip N/W IF
7. $X_7 \rightarrow$ IP in NoC-4

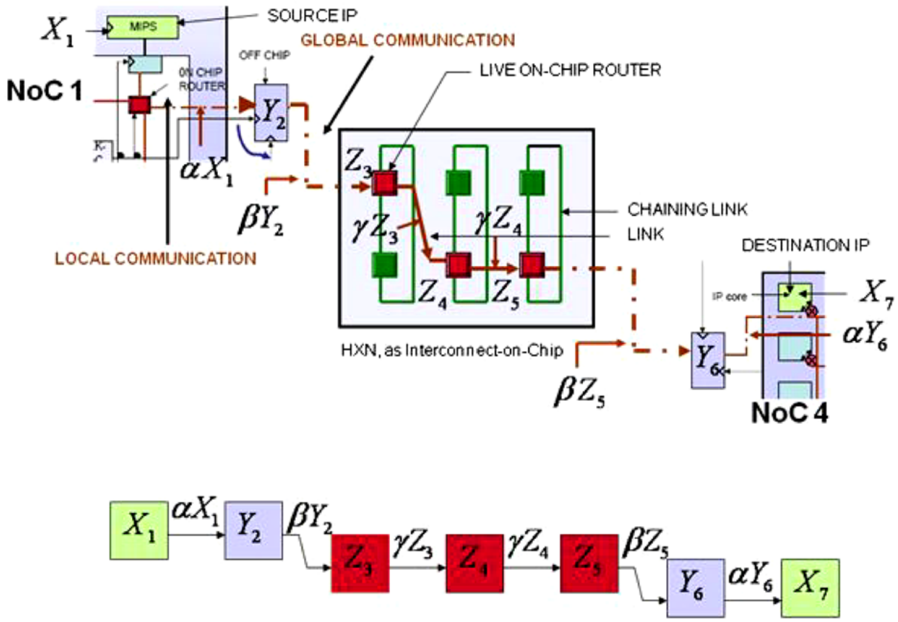


Fig. 14 Data flow network for stochastic communication

The behavior of data flow model shown in Fig. 14 can be described by the following set of differential equations:

$$\frac{dX_1}{dt} = -\alpha X_1 \tag{32}$$

$$\frac{dY_2}{dt} = \alpha X_1 - \beta Y_2 \tag{33}$$

$$\frac{dZ_3}{dt} = \beta Y_2 - \gamma Z_3 \tag{34}$$

$$\frac{dZ_4}{dt} = \gamma Z_3 - \gamma Z_4 \tag{35}$$

$$\frac{dZ_5}{dt} = \gamma Z_4 - \beta Z_5 \tag{36}$$

$$\frac{dY_6}{dt} = \beta Z_5 - \alpha Y_6 \tag{37}$$

$$\frac{dX_7}{dt} = \alpha Y_6 \tag{38}$$

$$N = X_1 + Y_2 + Z_3 + Z_4 + Z_5 + Y_6 + X_7 \tag{39}$$

where α, β and γ are the different data flow rates for respective compartment.

Figure 15 shows the dynamic behavior (number of packets transferred per unit of time) of basic data flow network in NiP for $N = 525$, $X_1(0) = 500$, $Y_2(0) = 25$,

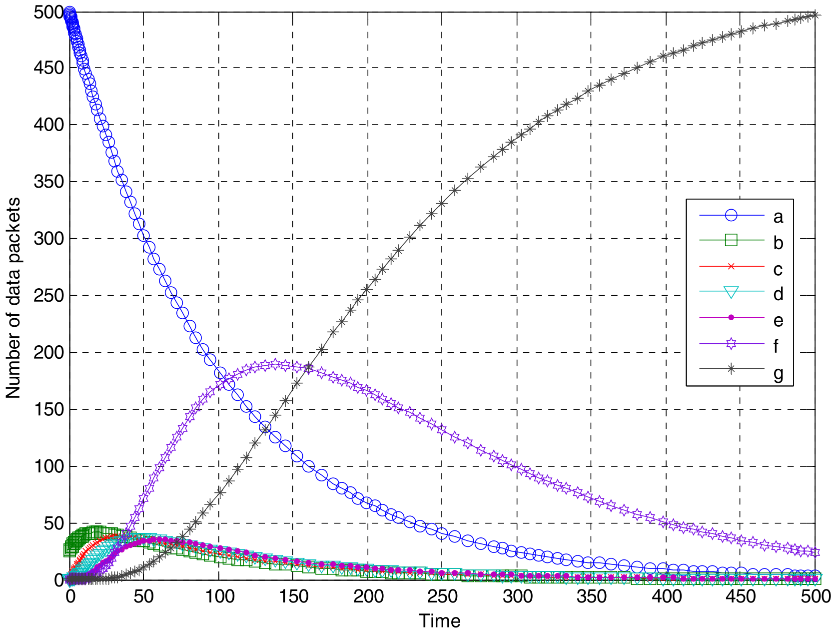


Fig. 15 Dynamic behavior of data flow network in NiP. Here, the graph legends are coded as follows: a = X_1, b = X_2, c = X_3, d = X_4, e = X_5, f = X_6 and g = X_7

$Z_3(0) = Z_4(0) = Z_5(0) = 0$, and $Y_6(0) = X_7(0) = 0$, where N is total number of data packets to be transmitted. For on-chip synchronization, the flow rates in HXN are considered as $\alpha = 0.01$ and $\beta = \gamma = 0.1$. Equations (32)–(39) describe the behavior of stochastic network and the seven compartments i.e. $(X_1 - X_7)$ appearing in these equations are treated as physical state space variables. Since these equations are linear differential equations, thus the homogeneous solution for these can be derived further.

7.2 Compartmental modeling of data flow network in NiP

In this section, we have used the technique as discussed in Sect. 4 to derive the compartmental matrix from the state space equations (32)–(39), which defines the dynamic behavior of data flow networks. These state space equations can be further expressed in the form of matrix as given below.

$$\dot{X}(t) = AX(t) \tag{40}$$

$$\begin{pmatrix} \dot{X}_1 \\ \dot{Y}_2 \\ \dot{Z}_3 \\ \dot{Z}_4 \\ \dot{Z}_5 \\ \dot{Y}_6 \\ \dot{X}_7 \end{pmatrix} = \begin{pmatrix} -\alpha & 0 & 0 & 0 & 0 & 0 & 0 \\ \alpha & -\beta & 0 & 0 & 0 & 0 & 0 \\ 0 & \beta & -\gamma & 0 & 0 & 0 & 0 \\ 0 & 0 & \gamma & -\gamma & 0 & 0 & 0 \\ 0 & 0 & 0 & \gamma & -\beta & 0 & 0 \\ 0 & 0 & 0 & 0 & \beta & -\alpha & 0 \\ 0 & 0 & 0 & 0 & 0 & \alpha & 0 \end{pmatrix} \begin{pmatrix} X_1 \\ Y_2 \\ Z_3 \\ Z_4 \\ Z_5 \\ Y_6 \\ X_7 \end{pmatrix} \tag{41}$$

$$A = \begin{pmatrix} -\alpha & 0 & 0 & 0 & 0 & 0 & 0 \\ \alpha & -\beta & 0 & 0 & 0 & 0 & 0 \\ 0 & \beta & -\gamma & 0 & 0 & 0 & 0 \\ 0 & 0 & \gamma & -\gamma & 0 & 0 & 0 \\ 0 & 0 & 0 & \gamma & -\beta & 0 & 0 \\ 0 & 0 & 0 & 0 & \beta & -\alpha & 0 \\ 0 & 0 & 0 & 0 & 0 & \alpha & 0 \end{pmatrix} \tag{42}$$

where A is the compartmental matrix. The solution of homogeneous state (40) is as follows:

$$X(t) = e^{At} X(0) \tag{43}$$

$$X(t) = \begin{pmatrix} X_1(t) \\ Y_2(t) \\ Z_3(t) \\ Z_4(t) \\ Z_5(t) \\ Y_6(t) \\ X_7(t) \end{pmatrix} \tag{44}$$

$$e^{At} = L^{-1}\left[(sI - A)^{-1}\right] \tag{45}$$

or

$$e^{At} = I + At + \frac{1}{2!}A^2t^2 + \cdots + \frac{1}{i!}A^i t^i \tag{46}$$

where e^{At} represents the state transition matrix of data flow network and $X(0)$ is the column matrix that shows initial conditions of the model.

7.3 Stochastic modeling of data flow network in NiP

In this section, we have used the technique as discussed in Sect. 5 to evaluate the matrix P as follows:

$$P = \begin{pmatrix} 1-h\alpha & 0 & 0 & 0 & 0 & 0 & 0 \\ h\alpha & 1-h\beta & 0 & 0 & 0 & 0 & 0 \\ 0 & h\beta & 1-h\gamma & 0 & 0 & 0 & 0 \\ 0 & 0 & h\gamma & 1-h\gamma & 0 & 0 & 0 \\ 0 & 0 & 0 & h\gamma & 1-h\beta & 0 & 0 \\ 0 & 0 & 0 & 0 & h\beta & 1-h\alpha & 0 \\ 0 & 0 & 0 & 0 & 0 & h\alpha & 1 \end{pmatrix}^T \tag{47}$$

$$P = \begin{pmatrix} p_{x_1x_1} & p_{x_1y_2} & p_{x_1z_3} & p_{x_1z_4} & p_{x_1z_5} & p_{x_1y_6} & p_{x_1x_7} \\ p_{y_2x_1} & p_{y_2y_2} & p_{y_2z_3} & p_{y_2z_4} & p_{y_2z_5} & p_{y_2y_6} & p_{y_2x_7} \\ p_{z_3x_1} & p_{z_3y_2} & p_{z_3z_3} & p_{z_3z_4} & p_{z_3z_5} & p_{z_3y_6} & p_{z_3x_7} \\ p_{z_4x_1} & p_{z_4y_2} & p_{z_4z_3} & p_{z_4z_4} & p_{z_4z_5} & p_{z_4y_6} & p_{z_4x_7} \\ p_{z_5x_1} & p_{z_5y_2} & p_{z_5z_3} & p_{z_5z_4} & p_{z_5z_5} & p_{z_5y_6} & p_{z_5x_7} \\ p_{y_6x_1} & p_{y_6y_2} & p_{y_6z_3} & p_{y_6z_4} & p_{y_6z_5} & p_{y_6y_6} & p_{y_6x_7} \\ p_{x_7x_1} & p_{x_7y_2} & p_{x_7z_3} & p_{x_7z_4} & p_{x_7z_5} & p_{x_7y_6} & p_{x_7x_7} \end{pmatrix} \tag{48}$$

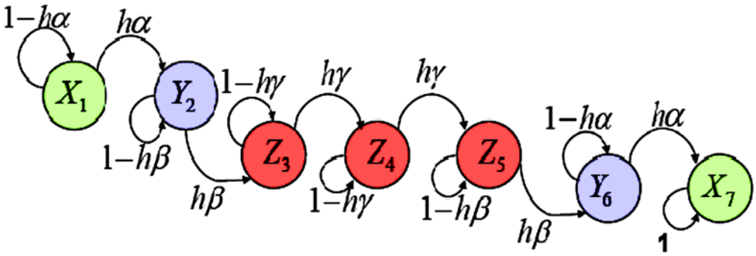


Fig. 16 Stochastic diagraph (Absorbing Markov Chain) of data flow network in NiP

$$P = \begin{pmatrix} 1-h\alpha & h\alpha & 0 & 0 & 0 & 0 & 0 \\ 0 & 1-h\beta & h\beta & 0 & 0 & 0 & 0 \\ 0 & 0 & 1-h\gamma & h\gamma & 0 & 0 & 0 \\ 0 & 0 & 0 & 1-h\gamma & h\gamma & 0 & 0 \\ 0 & 0 & 0 & 0 & 1-h\beta & h\beta & 0 \\ 0 & 0 & 0 & 0 & 0 & 1-h\alpha & h\alpha \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{49}$$

From (49), it is understood that sum of all the elements in each row of transition probability matrix P is equal to 1. Hence,

$$\sum_{j=1}^{7} p_{ij} = 1 \quad \text{where } i, j = 1, \ldots, 7 \tag{50}$$

7.3.1 Deriving Markov chain from transition probability matrix

In this section, the stochastic diagraph of transition probability matrix P is proposed (refer Fig. 16). Use the Definition 2, with following changes i.e. $(I - Q)^{-1}$, results in the expected time of data flow from IP in NoC-1 to IP in NoC-4 and $R_\infty = (I - Q)^{-1}R$, results in the probability of data transmission to the destination IP.

8 Stochastic analysis of on-package communication

In this section, the compartmental-based stochastic communication scheme for NiP is verified using (24) and (49). Hence, the required matrix P is as follows:

$$P = \begin{bmatrix} Q & R \\ 0 & I \end{bmatrix} = \left(\begin{array}{cc} \begin{pmatrix} 1-h\alpha & h\alpha & 0 & 0 & 0 & 0 \\ 0 & 1-h\beta & h\beta & 0 & 0 & 0 \\ 0 & 0 & 1-h\gamma & h\gamma & 0 & 0 \\ 0 & 0 & 0 & 1-h\gamma & h\gamma & 0 \\ 0 & 0 & 0 & 0 & 1-h\beta & h\beta \\ 0 & 0 & 0 & 0 & 0 & 1-h\alpha \end{pmatrix} & \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ h\alpha \end{pmatrix} \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} & (I) \end{array} \right) \tag{51}$$

The last state of this Markov chain I is the absorbing state, which consists of destination IP in NoC-4 in NiP. For $\alpha = 0.01$, and $\beta = \gamma = 0.1$ the time for each event or transition h is 0.1. This implies

$$P = \begin{pmatrix} 0.999 & 0.001 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.99 & 0.01 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.99 & 0.01 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.99 & 0.01 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.99 & 0.01 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.999 & 0.001 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{52}$$

$$Q = \begin{pmatrix} 0.999 & 0.001 & 0 & 0 & 0 & 0 \\ 0 & 0.99 & 0.01 & 0 & 0 & 0 \\ 0 & 0 & 0.99 & 0.01 & 0 & 0 \\ 0 & 0 & 0 & 0.99 & 0.01 & 0 \\ 0 & 0 & 0 & 0 & 0.99 & 0.01 \\ 0 & 0 & 0 & 0 & 0 & 0.999 \end{pmatrix} \tag{53}$$

$$R = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0.001 \end{pmatrix}, \quad I = [I] \tag{54}$$

For transient response,

$$\text{The matrix } (I - Q)^{-1} = \begin{pmatrix} 1000 & 100 & 100 & 100 & 100 & 1000 \\ 0 & 100 & 100 & 100 & 100 & 1000 \\ 0 & 0 & 100 & 100 & 100 & 1000 \\ 0 & 0 & 0 & 100 & 100 & 1000 \\ 0 & 0 & 0 & 0 & 100 & 1000 \\ 0 & 0 & 0 & 0 & 0 & 1000 \end{pmatrix} \tag{55}$$

can be further divided into small matrices such as

$$(I - Q)^{-1} = \begin{pmatrix} a_{11} = (1000) & a_{12} = (100\ 100\ 100\ 100) & a_{13} = (1000) \\ a_{21} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} & a_{22} = \begin{pmatrix} 100 & 100 & 100 & 100 \\ 0 & 100 & 100 & 100 \\ 0 & 0 & 100 & 100 \\ 0 & 0 & 0 & 100 \end{pmatrix} & a_{23} = \begin{pmatrix} 1000 \\ 1000 \\ 1000 \\ 1000 \end{pmatrix} \\ a_{31} = (0) & a_{32} = (0\ 0\ 0\ 0) & a_{33} = (1000) \end{pmatrix}$$

$$\text{where, } a_{22} = \begin{pmatrix} 100 & 100 & 100 & 100 \\ 0 & 100 & 100 & 100 \\ 0 & 0 & 100 & 100 \\ 0 & 0 & 0 & 100 \end{pmatrix}$$

represents a Jordan Block, which has two important properties are as follows:

Property 1: If the entries of the Jordan Block are same, then the data flow is smooth otherwise,

Property 2: If any entry of Jordan Block mismatches, then data flow will not be regular and the node that differs in value needs a buffer.

From (55), the following can be calculated:

1. Expected time during which the data available with source compartment IP(X_1)in NoC-1 = 1000 nanoseconds.
2. Expected delay to reach the router (Z_3) in HXN = 1000 + 100 = 1100 nanoseconds.
3. Expected time during which the data live in HXN = 100 + 100 + 100 = 300 nanoseconds.
4. Expected delay to reach from HXN IoC to the destination compartment IP(X_7) in NoC-4 = 300 + 1000 = 1300 nanoseconds.
5. Expected delay to reach from source compartment IP(X_1)in NoC-1 to the destination compartment IP(X_7) in NoC-4 using HXN IoC = 1000 + 300 + 100 + 1000 = 2400 nanoseconds.

For steady state response,

$$R_\infty = (1 - Q)^{-1} R = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} \tag{56}$$

From (56), it is predicted that the probability of data reception by destination compartment IP(X_7) on NoC-4 = 1.

For the steady state, complete transition probability matrix is given by

$$P_\infty = \begin{bmatrix} Q_\infty & R_\infty \\ 0 & I \end{bmatrix} \tag{57}$$

$$P_\infty = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{58}$$

The regular matrix P at ∞ will be a null matrix and it will have the certain probabilities to receive the data depending upon the value of the separation constant h.

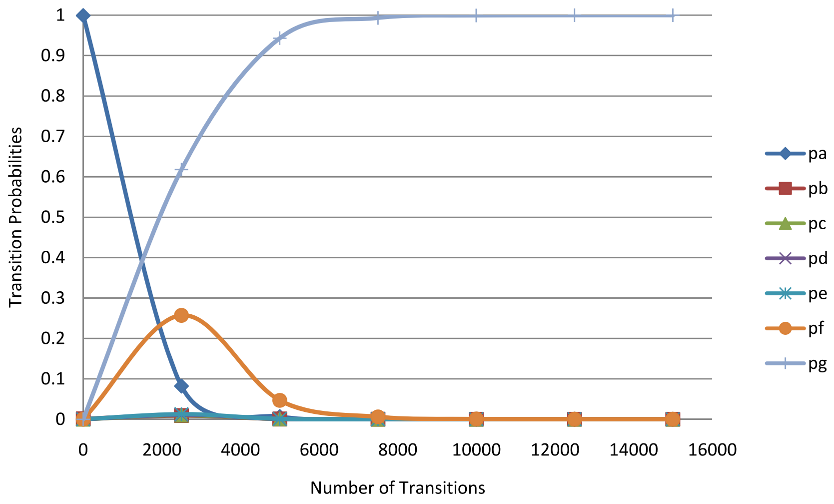


Fig. 17 Transition probabilities of data flow for compartment (X_1). Here, the graph legends are coded as follows: pa $= p(X_1X_1)$, pb $= p(X_1Y_2)$, pc $= p(X_1Z_3)$, pd $= p(X_1Z_4)$, pe $= p(X_1Z_5)$, pf $= p(X_1Y_6)$, and pg $= p(X_1X_7)$

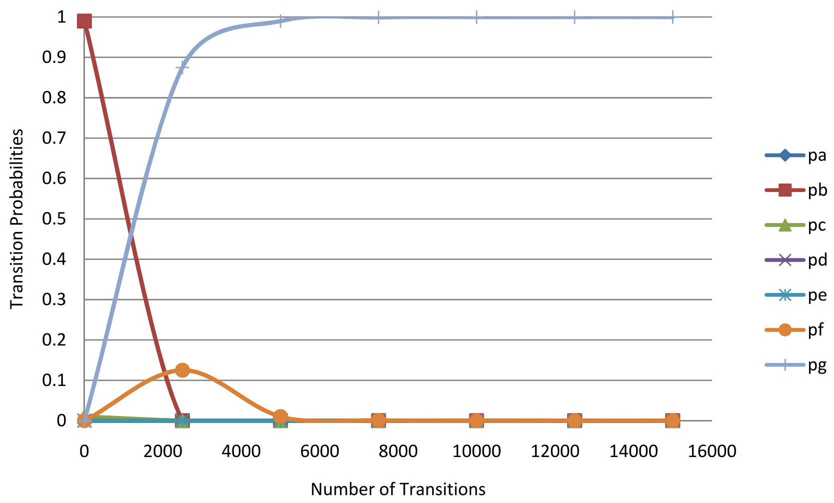


Fig. 18 Transition probabilities of data flow for compartment (Y_2). Here, the graph legends are coded as follows: pa $= p(Y_2X_1)$, pb $= p(Y_2Y_2)$, pc $= p(Y_2Z_3)$, pd $= p(Y_2Z_4)$, pe $= p(Y_2Z_5)$, pf $= p(Y_2Y_6)$, and pg $= p(Y_2X_7)$

9 Simulation and results

Figures 17–20, Tables 4–7 and P(X_iX_j) shows that the transition probabilities of data flow from (X_i) IP (of any NoC) to (X_j) IP (must be on other NoC), where $i = 1, \ldots, 6$ and $j = 1, \ldots, 7$. Specifically, this will help to calculate the total transition probabilities between any two NoCs (including IoC) and further, it will help in

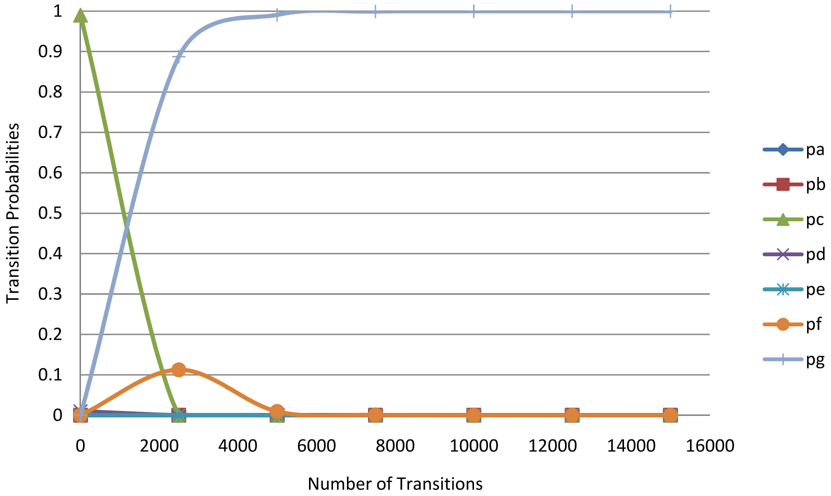


Fig. 19 Transition probabilities of data flow for compartment (Z_3). Here, the graph legends are coded as follows: pa = $p(Z_3X_1)$, pb = $p(Z_3Y_2)$, pc = $p(Z_3Z_3)$, pd = $p(Z_3Z_4)$, pe = $p(Z_3Z_5)$, pf = $p(Z_3Y_6)$, and pg = $p(Z_3X_7)$

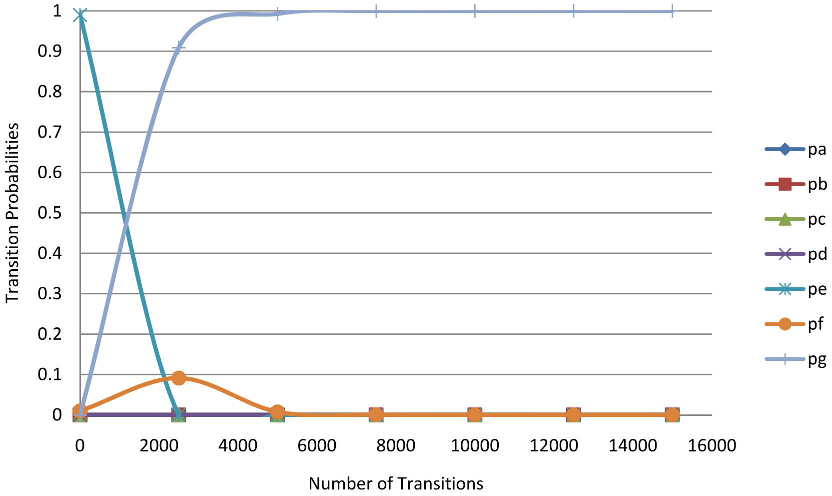


Fig. 20 Transition probabilities of data flow for compartment (Z_5). Here, the graph legends are coded as follows: pa = $p(Z_5X_1)$, pb = $p(Z_5Y_2)$, pc = $p(Z_5Z_3)$, pd = $p(Z_5Z_4)$, pe = $p(Z_5Z_5)$, pf = $p(Z_5Y_6)$ and pg = $p(Z_5X_7)$

calculating the latency. In addition, the proposed technique makes separation between communication and computation in NiP. Furthermore:

1. It gives the information about the data packets propagation.

Table 4 Transition probabilities of data flow for compartment (X_1)

Number of transitions	Transition probabilities						
	$p(X_1X_1)$	$p(X_1Y_2)$	$p(X_1Z_3)$	$p(X_1Z_4)$	$p(X_1Z_5)$	$p(X_1Y_6)$	$p(X_1X_7)$
1	0.999	0.001	0	0	0	0	0
2500	0.082	0.0091	0.0101	0.0112	0.0125	0.2572	0.6179
5000	0.0067	0.0007	0.0008	0.0009	0.001	0.0467	0.943
7500	0.0006	0.0001	0.0001	0.0001	0.0001	0.0059	0.9932
10,000	0	0	0	0	0	0.0007	0.9993
12,500	0	0	0	0	0	0	0.9999
15,000	0	0	0	0	0	0	1

Table 5 Transition probabilities of data flow for compartment (Y_2)

Number of transitions	Transition probabilities						
	$p(Y_2X_1)$	$p(Y_2Y_2)$	$p(Y_2Z_3)$	$p(Y_2Z_4)$	$p(Y_2Z_5)$	$p(Y_2Y_6)$	$p(Y_2X_7)$
1	0	0.99	0.01	0	0	0	0
2500	0	0	0	0	0	0.125	0.875
5000	0	0	0	0	0	0.0102	0.9898
7500	0	0	0	0	0	0.0008	0.9992
10,000	0	0	0	0	0	0.0001	0.9999
12,500	0	0	0	0	0	0	1
15,000	0	0	0	0	0	0	1

Table 6 Transition probabilities of data flow for compartment (Z_3)

Number of transitions	Transition probabilities						
	$p(Z_3X_1)$	$p(Z_3Y_2)$	$p(Z_3Z_3)$	$p(Z_3Z_4)$	$p(Z_3Z_5)$	$p(Z_3Y_6)$	$p(Z_3X_7)$
1	0	0	0.99	0.01	0	0	0
2500	0	0	0	0	0	0.1125	0.8875
5000	0	0	0	0	0	0.0092	0.9908
7500	0	0	0	0	0	0.0008	0.9992
10,000	0	0	0	0	0	0.0001	0.9999
12,500	0	0	0	0	0	0	1
15,000	0	0	0	0	0	0	1

2. It provides the directional movement of data packet i.e., when the column entries of the transition probabilities table are 0 that implies data is no more available at that node.
3. It gives the information about the time during which the data is live across a load, i.e., the data will remain with the node until the transition probabilities of concern node becomes to zero.

Table 7 Transition probabilities of data flow for compartment (Z_5)

Number of transitions	Transition probabilities						
	$p(Z_5X_1)$	$p(Z_5Y_2)$	$p(Z_5Z_3)$	$p(Z_5Z_4)$	$p(Z_5Z_5)$	$p(Z_5Y_6)$	$p(Z_5X_7)$
1	0	0	0	0	0.99	0.01	0
2500	0	0	0	0	0	0.0911	0.9089
5000	0	0	0	0	0	0.0075	0.9925
7500	0	0	0	0	0	0.0006	0.9994
10,000	0	0	0	0	0	0.0001	0.9999
12,500	0	0	0	0	0	0	1
15,000	0	0	0	0	0	0	1

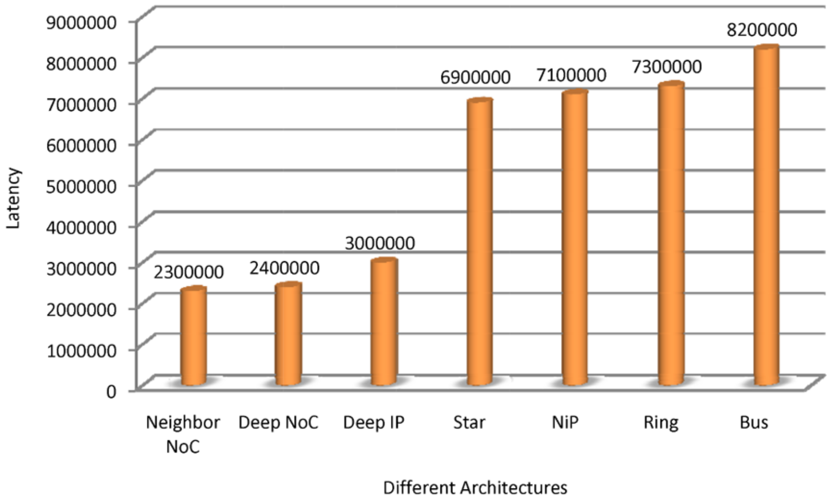


Fig. 21 NoC diversity: comparing different architectures

4. It gives the status of data propagation along with their transition probabilities, i.e., the data can be traced or dragged across any intermediate node, while the transition probability is at maximum peak. It helps to use the data from intermediate SEs for real time interfacing applications.
5. These results (the combine output of step 1 to step 4) helps to predict the latency from node to node and from Input to Output using the formula: Latency = Number of transitions at which maximum transition probability appear × Transition interval.
6. This technique also fits in the different architecture of NiP where link length varies consequently the value of h varies accordingly.

Table 8 The comparison of the existing and proposed stochastic communication model

Features	Existing systems	Proposed model
1. Communication	Exist only for Broadcasting scheme.	It not only exists for the broadcasting scheme but also for parallel and dedicated communication for real time application.
2. Prediction of Latency	Calculated within the group of NoCs.	Calculated within the group of NoCs group as well as for individual NoC.
3. Probabilistic Data Tracking	Available for Individual NoC only.	Available for the group of NoCs as well as for Individual NoC.
4. Prediction Based QoS	Exists.	Exists. However, design prospects of the proposed system and Stochastic Communication technique are the best.
5. Stochastic Communication	Present for Inter-NoC (local) as well as Intra (global) NoCs Communication. The architecture design of NoCs was homogeneous only.	The proposed system merges the inter-NoC and intra NoCs communication in NiP. The architectural design of a NoC can be homogeneous or heterogeneous.
6. Presence of Livelock	Yes.	No.
7. Presence of Deadlock	Yes.	No.
8. Loss of Data Packets	Yes.	The lost data packets (during the parallel communication among NoCs in NiP) and their exact locations can be tracked and predicted using Stochastic Communication Technique. Specifically, this communication scheme, a large percentage of data upsets, packet losses due to buffer overflows, and severe levels of synchronization failures can be tolerated, while providing high levels of performance.

10 Conclusion

We have proposed a new technique of stochastic communication between the different IPs and NoCs. In addition, it helps in building the compartmental model, calculating the latency and the transition probabilities of data flow between IPs and NoCs. Figures 11–13 and Tables 1–3 show the transient and steady state response of transition probabilities and provide the state of data flow latencies among the different IPs in a NoC. Moreover, Figures 17–20 and Tables 4–7 show the transient and steady state response of transition probabilities and provide the state of data flow latencies among the compartments in NiP. Further, from Fig. 21, it is predicted that the latency of NiP is lower in comparison to the ring and bus topology. However, it is higher than the star topology. Furthermore, the latency of inter-IP communication in NoC and communication among NoCs in NiP is very low as compared to the latency produced by the existing topologies. Therefore, the stochastic communication technique proposed in this paper have produced low latency over existing topologies, which have been used for setting up inter- and intra-NoC communication in the past. Various features of the old system and the proposed system are tabulated in Table 8. From this data, we can easily figure out that the proposed system proves to be superior in comparison to the existing system.

Acknowledgements This research work is dedicated to my Shri Shri 1008 Swami Shri Paramhans Dayal Sacchidanand Maharaja Ji and the loving memories of my departed Maternal Grandfather and Grandmother and Father-in-law, who continue to guide me in spirit.

References

1. Kangmin L, Se-Joong L, Donghyun K, Kwanho K, Gawon K, Joungho K, Hoi-Jun Y (2005) Networks-on-chip and networks-in-package for high-performance SoC platforms. In: Proceedings of international conference on Asian solid-state circuits, pp 485–488
2. Kangmin L, Se-Joong L, Hoi-Jun Y (2006) Low-power network-on-chip for high-performance SoC design. IEEE Trans Very Large Scale Integr Syst 14(2):148–160
3. Dumitras TA (2003) On-chip stochastic communication. Masters Thesis, Department of Electrical and Computer Engineering Carnegie Mellon University, Pittsburgh, Pennsylvania, 2003
4. Nitin, Sehgal VK, (2007) Stochastic communication on application specific networks on chip. Springer, Berlin, pp 11–16, ISBN 978-1-4020-6265-0
5. Nitin, Chauhan DS, Sehgal VK (2008) Two $O(n^2)$ time fault-tolerant parallel algorithm for inter NoC communication in NiP. Springer, Berlin, pp 267–282, ISBN 978-3-540-79186-7, Invited
6. Nitin, Sehgal VK, Chauhan DS (2008) A new approach for inter networks-on-chip communication in networks-in-package. In: Proceedings of the 6th international conference on embedded systems and applications, pp 16–23
7. Nitin, Sharma R, Sehgal VK, Mehta R, Sethi P., Gupta E. (2009) Asymptotic analysis of dynamic algorithms designed to provide parallel communication among NoC in NiP using MIN. In: Proc 11th international conference on computer modeling and simulation, pp 443–448
8. Kim K, Lee S-J, Lee K, Yoo H-J ? An arbitration look-ahead scheme for reducing end-to-end latency in networks on chip. In: Proc IEEE international symposium on circuits and systems, 2005, pp 2357–2360
9. Murali S, Micheli G (2004) SUNMAP: a tool for automatic topology selection and generation for NoCs. In: Proceedings of design automation conference, pp 914–919
10. Bjerregaard T, Mahadevan S (2006) A survey of research and practices of network-on-chip. ACM Comput Surv 38(1):1–51
11. Madl G, Pasricha S, Dutt ND, Abdelwahed S (2009) Cross-abstraction functional verification and performance analysis of chip multiprocessor designs. IEEE Trans Indust Inform 5(3):241–256
12. Gajski D, Dutt ND, Wu A, Lin S (1992) High level synthesis: introduction to chip and system design. Kluwer Academic, Norwell, ISBN: 978-0-7923-9194-4
13. Pasricha S, Dutt ND (2008) On-chip communication architectures. Kauffman, Los Altos, ISBN: 978-0-12-373892-9
14. Bogdan P, Marculescu R (2006) A theoretical framework for on-chip stochastic communication analysis. In: Proceedings of 1st international conference on nano-networks, pp 1–5
15. Bogdan P, Dumitras TA, Marculescu R (2007) Stochastic communication: a new paradigm for fault-tolerant networks-on-chip, VLSI Des, pp 1–17. doi:10.1155/2007/95348
16. Sgroi M, Sheets M, Mihal A, Keutzer K, Malik S, Rabaey J, Vincentelli AS (2001) Addressing the system-on-a-chip interconnect woes through communication-based design. In: Proceedings of the 38th annual ACM IEEE conference on design automation conference, pp 667–672
17. Benini L, De Micheli G (2002) Networks-on-chips: a new SoC paradigm. IEEE Comput 35(1):70–78
18. Benini L, De Micheli G (2006) Networks-on-chips: technology and tools. Kaufmann, San Francisco, ISBN: 978-0-12-370521-1
19. Jayasimha DN, Zafar B, Hoskote Y (2006) On-chip interconnection networks: why they are different and how to compare them, Intel Corporation, pp 1–11
20. Inwood B (1986) Goal and target in stoicism. J Philos 83(10):547–556
21. Asmussen S, Glynn PW (2007) Stochastic simulation: algorithms and analysis. Springer, Berlin, ISBN: 978-0-387-30679-7
22. Duato J, Yalamanchili S, Ni LM (2003) Interconnection networks: an engineering approach. Kaufmann, San Francisco, ISBN: 1-55860-852-4

23. Dally WJ, Towles B (2004) Principles and practices of interconnection networks. Kaufmann, San Francisco, ISBN: 978-0-12-200751-4
24. Nitin, Subramanian A (2008) Efficient algorithms to solve dynamic MINs stability problems using stable matching with complete ties. J Discrete Algorithms 6(3):353–380
25. Nitin, Sehgal VK, Sharma N, Krishna K, Bhatia A (2007) Path-length and routing-tag algorithm for hybrid irregular multi-stage interconnection networks. In: Proceedings of the 8th ACIS international conference on software engineering, artificial intelligence, networking, and parallel/distributed computing, pp 652–657
26. Nitin, Sehgal VK, Bansal PK (2007) On MTTF analysis of a fault-tolerant hybrid MINs. WSEAS Trans Comput Res 2(2):130–138 ISSN: 1991-8755
27. Nitin (2006) Component level reliability analysis of fault-tolerant hybrid MINs. WSEAS Trans Comput 5(9):1851–1859 ISSN: 1109-2750
28. Nitin (2006) Reliability analysis of multi-path multi-stage interconnection network. In: Proceedings of the 10th WSEAS international conference on circuits, systems, communication and computers, pp 1018–1023
29. Nitin (2006) On analytic bounds of regular and irregular fault-tolerant multi-stage interconnection networks. In: Proceedings of the international conference on parallel and distributed processing techniques and applications, pp 221–226
30. Nitin, Subramanian A (2006) On reliability analysis of cost-effective hybrid zeta network: a fault-tolerant multi-stage interconnection network. In: Proceedings of the international conference on parallel and distributed processing techniques and applications, pp 260–265
31. Subramanian A, Nitin (2004) On a performance of multistage interconnection network. In: Proceedings of the 12th international conference on advanced computing and communication, pp 73–79
32. Nitin, Garhwal S, Srivastava N (2009) Designing a fault-tolerant fully-chained combining switches multi-stage interconnection network with disjoint paths, J. Supercomput, doi:10.1007/s11227-009-0336-z, pp 1–32
33. Nitin, Chauhan DS (2010) Comparative analysis of traffic patterns on *k-ary n-tree* using adaptive algorithms based on burton normal form, J Supercomput, doi:10.1007/s11227-010-0454-7, pp 1–20
34. Walter GG, Contreras M (1999) Compartmental modeling with networks. Birkhauser, Boston, ISBN: 0-8176-4019-3
35. Ladde GS (1976) Cellular systems-II. stability of compartmental systems. Math Biosci 30:1–21
36. Jacquez JA, Simon CP (2002) Qualitative theory of compartmental systems with lags. Math Biosci 180:329–362
37. Sandberg IW (1978) On the mathematical foundations of compartmental analysis in biology, medicine, and ecology. IEEE Trans Circuits Syst 25(5):273–279