

Performance Modeling and Throughput Analysis of Perfect Difference Network for Parallel and Distributed Systems

Rakesh Kumar Katare

Professor, Department of Computer Science, Awadhesh Pratap Singh University, Rewa, Madhya Pradesh
Email: katare1962@gmail.com

Manish Bhardwaj

Senior Scientist, CSIR Human Resource Development Group, CSIR Complex, New Delhi
Email: manish@csirhrdg.res.in

Abstract

Numerous challenges are faced by researchers in the efficient design of future interconnection networks. We are in a phase of strengthening, improving the architectural design and hardware of new interconnection networks importantly Perfect Difference Network (PDN). This paper focuses on the study of various performance metrics and modeling techniques of interconnection networks. Bandwidth, functionality, latency, scalability, density and throughput are the decisive metrics for the cost effective optimal design of interconnection networks. Interconnection systems can be modeled using performance modeling techniques such as analytical, simulation and empirical. PDN interconnection network communication arc analysis has been done in this paper. Routing tables have been derived for PDN with $\delta=2,3,4$ with their linear, communication and diagonal connectivity. Nodes are represented using a binomial and polynomial code. A PDN ($\delta=2$) is simulated on Network Simulator ver3 (NS3) to measure the throughput for the User Datagram Protocol (UDP) and Transmission Control Protocol (TCP) for limited and unlimited data packet transfer.

Keywords—Interconnection network (ICN), Perfect Difference Network (PDN), Perfect Difference Set (PDS), Network Simulator (NS)

1. Introduction

The performance of interconnection network is implicitly defined by its network topology. This paper focuses on the performance evaluation of PDN interconnection network with $\delta=2$. Simulated model predicts the behavior of the PDN network by measuring the inter-node communication metrics of the model. Moreover, the most common metrics for the performance evaluation of interconnection network is defined as a diamond model with each vertex represents the well defined parameter such as bandwidth, functionality, latency, scalability, hardware complexity and network throughput. The behavioral analysis of communication pattern of interconnection network can be modeled using analytical, simulation and empirical techniques. Our interest revolves around the simulation model. We consider Network Simulator (ver3) NS3 to study the behavior pattern of PDN with $\delta=2$ value. PDN is based on perfect difference set $\{0, 1, S_2, \dots, S_\delta\}$ having nodes $n=\delta^2+\delta+1$, with labels from 0 to $n-1$. There is a communication link between any PEs, if and only if, $i \pm 1$ and $i \pm s_j \pmod n$, for $2 \leq j \leq \delta$ [1]. We have assigned the binary value to the nodes for polynomial representation of the connected links of PDN. A routing table of the communication arcs of the PDN with linear nodes, communication nodes and diagonal nodes is also derived.

The remaining sections of this paper are organized as follows. In section 2, proposed metrics and models have been discussed and section 3, describe the interconnection network PDN from PDS, routing table and link study lemmas. Section 4 examines the performance and throughput analysis of results of simulation of PDN ($\delta=2$) on NS3. The concluding remarks are cited in section 5.

2. Proposed Metrics and Models

PDN interconnection network is represented using graph theoretic model with the node as processors and edges as communication links. We briefly discussed PDN interconnection network communication links used for exchanging data between two processors. To a large extent delay in data exchange and the routing algorithm between a pair of processors nodes hinders the performance of interconnection network (ICN). Nodes have a limited number of input and output communication links which defines the density of PDN interconnection networks.

2.1 Performance Metrics

The performance of ICN is significantly measured by the graph theoretic properties such as topology, diameter and bisection width. An ICN with low diameter and high bisection width is highly preferred for optimal performance. We rethink the performance metrics of the ICN and propose the Diamond Model in fig.1. Performance metrics play a decisive role in the design and modeling of interconnection networks cost feasibility study of parallel and distributed systems [2].

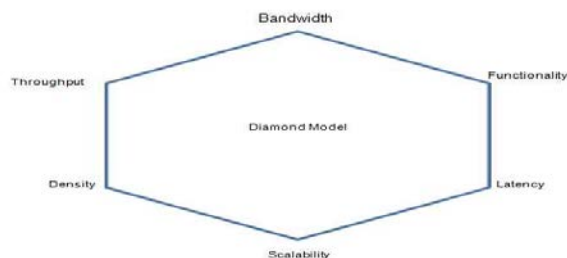


Figure1. Diamond Model of performance metrics

Bandwidth: Data transfer rate supported by a communication channel between two nodes is measured by bandwidth.

Functionality: Data routing techniques or mapping function to predict the behavior of interconnection network.

Latency: Data transfer delay to ship message between two extreme end to end nodes.

Scalability: Ability of the interconnection network to grow dynamically with increasing hardware complexity.

Density: Ratio of total number of existing communication links to the communication links potentially possible.

Throughput: Ratio of total number of actually transferred messages to the total capacity of the communication link.

The discussed diamond model metrics are implemented in section 4 to measure the performance and throughput analysis of the PDN $\delta=2$, simulated on the NS 3.

2.2 Performance Modeling Techniques

Cost effective approach to model an efficient interconnection network requires the bug free architectural design. Topological properties and routing algorithms of interconnection network plays a important role to fine tune the modeled interconnection network. Architectural prototype of the real interconnection network can be modeled using performance modeling techniques.

A proposed pyramid model in fig.2 of modeling techniques. Prototype design of the interconnection network can be modeled with mathematics probability using a analytical technique, architectural design can be verified and validated through simulation and empirical method can evaluate the prototype [4].

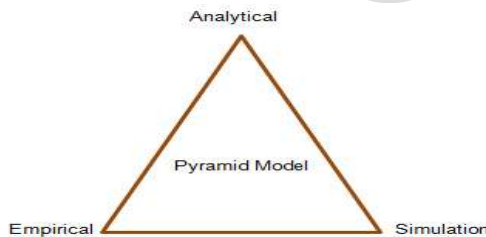


Figure 2. Pyramid Model of Performance Modeling Techniques

Analytical Modeling: Design prototype of advance interconnection networks is based on analytical modeling. Analytical modeling technique is a tool that uses basic mathematical technique of probability concepts and queuing theory.

Simulation Modeling: Prototype of interconnection networks with design limitations is based on simulation modeling. Simulation helps in visualizing and analyzes the interconnection networks. Performance of interconnection network can be measured with performance metrics in real simulated environment. Simulation model development life cycle (SmDLC) has three phases: (i) model building, (ii) verification and (iii) validation. Previously developed and required specification plays a major role in simulating a

interconnection network. Simulation helps in developing and testing the cost effective prototype of the actual interconnection model with performance metrics as parameters.

In this paper, we have simulated Perfect Difference Network ($\delta=2$) using NS3 simulator under factual implementation of discussed performance metrics in section 2.1.

Empirical Modeling: Developed prototype can be verified using empirical techniques. Two popular empirical techniques are dynamic and static evaluation. The resulting values of the prototype model can be measured using empirical techniques.

3. Derivation of PDN from PDS

Blend of mathematical properties of a finite field, projective geometry and Perfect Difference Set (PDS) defines a recently developed and coveted interconnection network known as the Perfect Difference Network. With advancement in on-chip and inter communication technology between processor cores for fast performance, PDN marks its presence. PDN is an undirected ICN of diameter 2 with equivalent in and out node degree. PDN is derived from perfect difference set $\{0, 1, S_2, \dots, S_\delta\}$ for n number nodes where $n=\delta^2+\delta+1$ and nodes are numbered from 0 to n-1. δ is prime or a power of prime. Table 1 represents a perfect difference sets [5] for values of $\delta=2, 3, 4$.

δ	n	PDS
2	7	{0,1,3}
3	12	{0,1,3,,9}
4	21	{0,,1,4,14,16}

3.1 PDN Communication Link Analysis

Without improved interconnection networks, progress in all spheres of distributed systems is likely to come to a halt. It is important to adapt the emerging trends like Perfect Difference Network as shown in fig.3 to design the future interconnection networks. In this section, we study the communication links of PDN ($\delta=2$) as in fig.3.

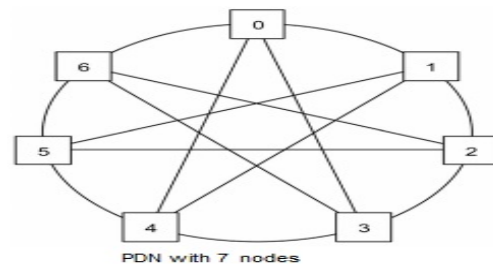


Figure 3. Perfect Difference Network

Table 2 represents the relationship between δ , binary value and polynomial codes for PDN ($\delta= 2$)

PDN $\delta= 2$		
$\delta-2$	0	0
$\delta-1$	1	1
δ	10	x
$\delta+1$	11	x+1
$\delta+2$	100	x^2
$\delta+3$	101	x^2+1
$\delta+4$	110	x^2+x

In Table 2.1, the nodes of PDN are assigned a binary value for their polynomial code representation.

P_E	Binary Code	Polynomial Code	P_E	Binary Code	Polynomial Code
	n_0	00000		0	n_{11}
n_1	00001	1	n_{12}	01100	$x^3 + x^2$
n_2	00010	x	n_{13}	01101	$x^3 + x^2 + 1$
n_3	00011	x+1	n_{14}	01110	$x^3 + x^2 + x$
n_4	00100	x^2	n_{15}	01111	$x^3 + x^2 + x + 1$
n_5	00101	x^2+1	n_{16}	10000	x^4
n_6	00110	x^2+x	n_{17}	10001	x^4+1
n_7	00111	x^2+x+1	n_{18}	10010	x^4+x
n_8	01000	x^3	n_{19}	10011	x^4+x+1
n_9	01001	x^3+1	n_{20}	10100	$x^4 + x^2$
n_{10}	01010	x^3+x			

Undirected wired PDN routing table shows the point to point node connectivity for linear links, communication links and diagonal links. [8]

Node i is connected to adjacent nodes via linear links. Coupled nodes of i are bind together with communication links. Reliability and robustness topological properties of PDN are supported via diagonal links. Using the polynomial $\delta^2+\delta+1, i \pm 1 \& i \pm s_i \pmod n$ where $2 \leq j \leq \delta$ and graph theoretic model [3]. We drive the routing table for PDN $\delta=2, 3, 4$.

Table 3 signifies the routing table for PDN with $\delta=2$ and $n=7$ nodes. Table 4 signifies the routing table for PDN with $\delta=3$ and $n=13$ nodes. Table 5 signifies the routing table for PDN with $\delta=4$ and $n=21$ nodes.

Two lemmas are proposed with a mathematical relationship between the linear and diagonal communication links of PDN.

Lemma 1: A total communication arcs for a given node under observation of a PDN is defined by a total number of linear nodes and diagonal nodes. Equation 1 formulation is $2+2(\delta-1) = 2\delta$.

Proof: We assume a PDN with $\delta= 2, 3, 4$ value. PDN is based on perfect difference set $\{0, 1, S_2, \dots, S_n\}$ having nodes $n=\delta^2+\delta+1$, with labels from 0 to $n-1$.

Case i: - Consider Table 3 for $\delta=2$

Using equation 1 of lemma 1

$$2+2(\delta-1) = 2\delta$$

$$2+2(2-1) = 2*2$$

$$4=4$$

Case ii:- Consider Table 4 for $\delta=3$

Using equation 1 of lemma 1

$$2+2(\delta-1) = 2\delta$$

$$2+2(3-1) = 2*3$$

$$6=6$$

Case iii:- Consider Table 5 for $\delta=4$

Using equation 1 of lemma 1

$$2+2(\delta-1) = 2\delta$$

$$2+2(4-1) = 2*4$$

$$8=8$$

Therefore, it is proved as in case i, ii and iii, the equation $2+2(\delta-1) = 2\delta$ is true for PDN of all the values of δ as prime or power of prime.

Lemma 2: A polynomial $n=\delta^2+\delta+1$ defines the total number of nodes of PDN. The number of distinct communication arcs (DCA) in an undirected PDN is $\delta (\delta^2+\delta+1)$ designated as equation 2.

Proof: We assume a PDN with $\delta= 2, 3, 4$ value.

Case i: - Consider Table 3 for $\delta=2$

Using equation 2 of lemma 2

$$DCA= \delta (\delta^2+\delta+1).$$

$$=2*7$$

=14

Case ii:- Consider Table 4 for $\delta = 3$

Using equation 2 of lemma 2

$$DCA = \delta (\delta^2 + \delta + 1).$$

$$= 3 * 13$$

$$= 39$$

Case iii:- Consider Table 5 for $\delta = 4$

Using equation 2 of lemma 2

$$DCA = \delta (\delta^2 + \delta + 1).$$

$$= 4 * 21$$

$$= 84$$

Therefore, it is proved as in case i, ii and iii, the equation $\delta (\delta^2 + \delta + 1)$ is true for PDN of all the values of δ as prime or power of prime.

4. Network Simulators

Cost effective and efficient architectural design of interconnection networks is confirmed by implementing the desired prototype on the simulator. Graph theoretical models implemented on network simulators allow the researcher to visualize the working prototype of real complex interconnection networks. Two types of network simulators are available commercial & open source. We have reviewed three network simulators of interest: OPNET, NS2 and NS3. Famous commercial network simulator is OPNET and popular open source network simulators are NS2 and NS3 [7].

OPNET: OPNET is a popular commercial simulator. OPNET flexible design can study interconnection, devices protocols and applications. It has the powerful visual support for researches. Object Oriented Programming language can be used to translate the graphical design to the real systems. OPNET is discrete event simulator that can implement the tasks prioritize by the researcher. It supports the functions of modeling, simulation and analysis.

NS2: Network Simulator ver2 (NS2) is one of the most popular among the researchers as an open source simulator because of it highly rich library of network support components. Discrete events are supported by NS2. It is an object oriented simulator. C++ & OTcl are the two programming languages supported by NS2. Both the control and data support paths are different in NS2 simulator. Object oriented language C++ is used to support the network protocol library and OTcl is to simulate the scenario and schedule the events.

NS3: Network Simulator Ver 3 (NS3) is also an open source discrete event modular simulator. NS3 core is written in C++ with python scripting interface. It resembles close to real systems with possible integration of software and virtualization.

4.1 Simulation Environment

We have used Network Simulator ver.3 (NS3) which is much faster as compared to NS2 due to the removal of overheads present in NS2. It has an emulation mode, which give it privileges for the integration with real networks [7]. NS3 has two animation software PyViz and NETANIM, for visual analysis of mobility node, packet routing, statistics etc [15].

PyViz is a live simulator to visualize networks that are most often useful for debugging network simulation. It can detect where packets are being lost and the dynamic nodes are working properly or not. PyViz uses Python based Interactive console. It can also work with C++ [13].

NETANIM is an integrated NS3 module that process as the .xml files for graphical output which is generated through NS3. NetAnim uses a QT build tool called qmake. Qmake should create an executable name NetAnim. It can parse flow-monitor XML files with flowmon extension similar to xml file extension. This flowmon file display statistics for each flow [14].

4.2 Network Simulator NS3 Installation

i. To install NS3 in Ubuntu machine, system needs to have dependencies and libraries by which NS3 package will be installed on the system. Prerequisite is to update Ubuntu system repository. Next step is to download NS3 all-in-one tar.bz2 package through internet browser or from a command prompt with *wget* command.

ii. Next step is to unzip the complete downloaded package.

iii. Change directory command (*cd*) into the NS3 all-in-one folder. *Build.py* file will used to enable the tools necessary for network simulation.

iv. *./build.py --enable-examples --enable-test* the command will run *build.py* file. It will also enable examples and test files too.

v. Now to start, use command *./waf -d debug --enable-examples --enable-tests* configure command *./waf* to build and run scripts.

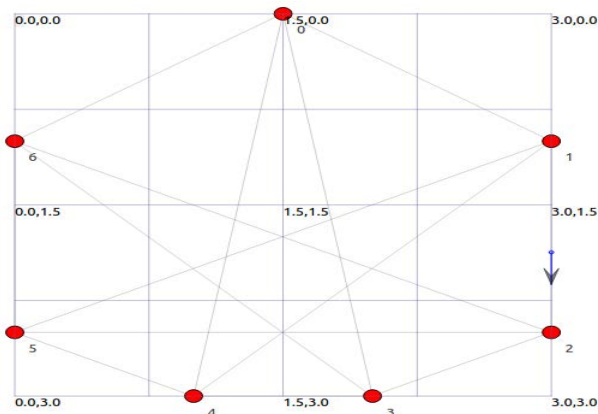
vi. Use command *./test.py* to test the complete working installation of the simulator.

4.3 Network Simulation Variables

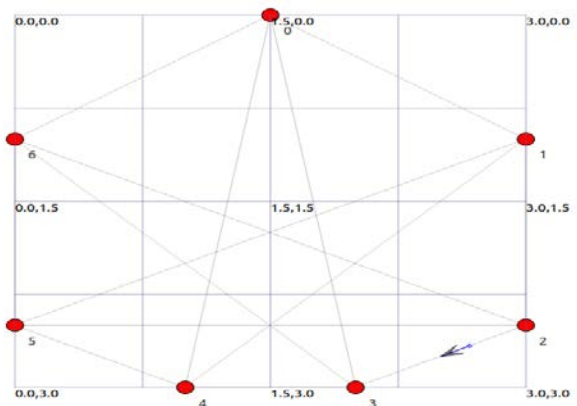
In this section, we list the important network simulation variables for simulating the PDN ($\delta=2$) on NS3. These variables are used to compare User Datagram Protocol (UDP) and Transmission Control Protocol (TCP) network protocols implemented on simulated interconnection network PDN [9]. Table 6 lists the performance variables to analyze the network throughput.

Table 6	
No. of Processor Nodes	7
Network Topology	Perfect Difference Network
Network Channel	Point-to-Point Channel
Network Protocol	UDP and TCP
Bandwidth	5 MBPS
Latency	10 Milliseconds
Data Packet Size	Limited Data Packet Size 1024 Bytes and Unlimited Data Packet Size 12358004 Bytes
Server Processor Node	N ₆
Client Processor Node	N ₁

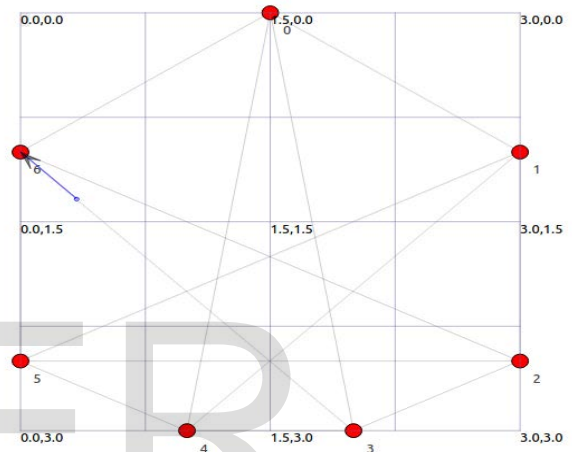
Seven processor nodes of PDN ($\delta=2$) is simulated on NS3 and NETANIM module is used for its graphical visualization. Simulation of PDN is shown in the screen shots 1 to 3. Processor nodes communicate with their neighboring nodes as per the routing table 3. It is assumed N1 as the source node and N6 as destination node. Simulated PDN topology is supported by point-to point channel with UDP and TCP as network protocol. A congestion free communication arc of 5MBPS bandwidth capable of transporting limited data packet size of 1024 Bytes and unlimited data packet size of 12358004 Bytes with 10 Milliseconds latency.



Screen Shot 1: Data packet transmission from Node 1 to Node 6 via Hop 2 and Hop 3. Current data packet flow is between node1 to node2.



Screen Shot 2: Data packet transmission from Node1 to Node6 via Hop 2 and Hop 3. Currently data packet flow is between node2 to node3.



Screen Shot 3: Data packet transmission from Node 1 to Node 6 via Hop 2 and Hop 3. Current data packet flow is between node3 to node6.

4.4 NS3 Simulation Flow Graph

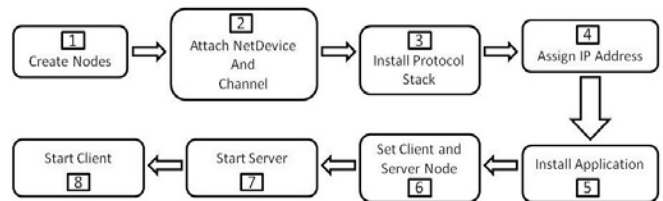


Figure.3: Simulation steps of PDN on NS3

Flow graph in fig.3 lists the steps involved in the simulation of interconnection network on NS3 [16].

1. Create Node: Processor node creation is the initial step for the simulation of PDN ($\delta=2$) interconnection network.
2. NetDevice & Network Channel: NS3 simulation software gives the choice to select NetDevice and Channel. A researcher can choose the network communication channel type like CSMA or Point-to-Point. Selection of NetDevice gives the option to set

attributes like bandwidth and latency in terms of data rate and delay in milliseconds respectively.

3. Network Protocol: Protocol is a set of rules of communication between two communicating nodes of the interconnection network. We have implemented two protocols UDP (User Datagram Protocol) and TCP (Transmission Control Protocol) in the simulation of PDN ($\delta=2$) on NS3.
4. IP Address: Each processor node of a simulated network is identified by an Internet Protocol Id. We have assigned the node address from IPv4 family. IP Address Helper of NS3 sets node the address. It is composed of two parts base IP address assigning and subnet mask which is the number of devices assign with the base IP address.
5. Application: Implemented network protocol UDP and TCP are supported via their traffic sources known as communication agents. CBR (Constant Bit Rate) is a communication agent for UDP and FTP (File Transfer Protocol) is a communication agent for TCP (Transmission Control Protocol).
6. Client and Server Node: Simulation of implementing PDN ($\delta=2$) is completed by defining of client and server processor nodes.
7. Start Server: N6 processor node is designated as the server will be started first to act as a ready listener to client processor node for receiving packet to avoid any traffic delay.
8. Start Client: N1 processor node is designated as a client node to send data to N6.

After simulating the complete PDN ($\delta=2$) on NS3, source code file is build and run using the command `./waf --run pdnsimulation.cc`.

5. Simulation Analysis of PDN ($\delta=2$)

Case i

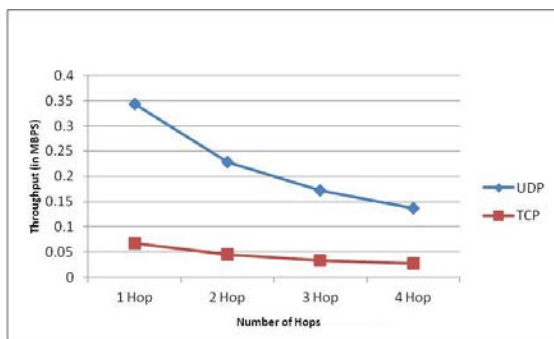


Figure 4. Throughput (Limited Data Packet Size)

Throughput of an interconnection network is measured as the rate of data transfer from client to sever processor node. The

simulated PDN is analyzed for throughput with preset environment variables given in table 6. As analyzed (fig.4) in case i the throughput for network protocol UDP for limited data packet size is more than TCP as number of hops varied. TCP works on the methodology of three way handshake technique where as UDP directly exchanges the data packets between the client-server processor nodes [8].

Case ii

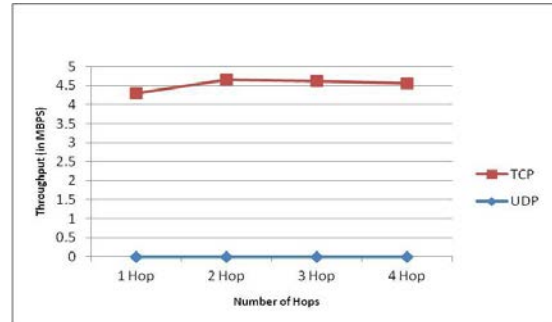


Figure 5. Throughput (Unlimited Data Packet Size)

In the second case, UDP does not support for unlimited data packet size whereas TCP has marginal change in the throughput as the number of hops increases.

6. Conclusion

The performance analysis of an interconnection network is based on performance metrics and simulation model. A polynomial routing table for PDN ($\delta=2, 3, 4$) lists the inter node communication arcs. We define a communication link relationship between linear and diagonal nodes as $2+2(\delta-1)=2\delta$ and the total number of distinct communication arcs for a given PDN is $\delta(\delta^2+\delta+1)$. The perfect difference network ($\delta=2$) has been modeled on open source simulator NS3 with preset performance metrics as simulated environment variables. The interconnection communication arcs of the simulated PDN are based on the routing table. The simulated PDN is analyzed with UDP and TCP network protocol for throughput. With limited data packet size, it is found that with increase in number of hops on the routing path the throughput decreases gradually in case of TCP whereas there is a sharp fall in case of UDP. For unlimited data packet size, UDP has no throughput where as for TCP shows negligible change in throughput with increase in number of hops. We can conclude that for large data packet size, UDP is not suitable, but TCP has the capability to carry forward the data packet with reliability. UDP is a connectionless protocol for fast data transmission and TCP is a connection oriented protocol for secure data transmission.

Acknowledgement

We thank Mr Akshay Kheral, Project Assistant-II, CSIR HRDG, New Delhi for assistance with NS3 simulator.

Table 3

Perfect Difference Network					
$\delta = 2, n = \delta^2 + \delta + 1$					
P_E	Linear Nodes	Communication Nodes	Diagonal Nodes	Communication Arcs $2+2(\delta-1)=2\delta$	Distinct Communication Arcs (DCA) $\delta(\delta^2 + \delta + 1)$
n_0	$1, x^2+x$	$1, x+1, x^2, x^2+x$	$x+1, x^2$	$1, x+1, x^2, x^2+x$	4
n_1	$0, x$	$0, x, x^2, x^2+1$	x^2, x^2+1	$1, x, x^2+1, x^2+2$	3
n_2	$1, x+1$	$1, x+1, x^2+1, x^2+x$	x^2+1, x^2+x	$x+1, 2x+1, x^2+x+1, x^2+2x$	3
n_3	$1, x^2$	$0, x, x^2, x^2+x$	$0, x^2+x$	$x+1, 2x+1, x^2+x+1, x^2+2x+1$	2
n_4	$x+1, x^2+1$	$0, 1, x+1, x^2+1$	$0, 1$	$x^2, x^2+1, x^2+x+1, 2x^2+1$	1
n_5	x^2, x^2+x	$1, x, x^2, x^2+x$	$1, x$	$x^2+2, x^2+x+1, 2x^2+1, 2x^2+x+1$	1
n_6	$0, x^2+1$	$0, x, x+1, x^2+1$	$x, x+1$	$x^2+x, x^2+2x, x^2+2x+1, 2x^2+x+1$	0

IJSER

Table 4

Perfect Difference Network					
$\delta = 3, n = \delta^2 + \delta + 1$					
P_E	Linear Nodes	Communication Nodes	Diagonal Nodes	Communication Arcs	Distinct Communication Arcs (DCA)
				$2+2(\delta-1)=2\delta$	$\delta(\delta^2 + \delta + 1)$
n_0	$1, x^3 + x^2$	$1, x+1, x^2, x^3+1, x^3+x, x^3+x^2$	$x+1, x^2, x^3+1, x^3+x$	$1, x+1, x^2, x^3+1, x^3+x, x^3+x^2$	6
n_1	$0, x$	$0, x, x^2, x^2+1, x^3+x, x^3+x+1$	$x^2, x^2+1, x^3+x, x^3+x+1$	$1, x+1, x^2+1, x^2+2, x^3+x+1, x^3+x+2$	5
n_2	$1, x+1$	$1, x+1, x^2+1, x^2+x, x^3+x+1, x^3+x^2$	$x^2+1, x^2+x, x^3+x+1, x^3+x^2$	$x+1, 2x+1, x^2+x+1, x^2+2x, x^3+2x+1, x^3+x^2+x$	5
n_3	x, x^2	$0, x, x^2, x^2+x, x^2+x+1, x^3+x^2$	$0, x^2+x, x^2+x+1, x^3+x^2$	$x+1, 2x+1, x^2+x+1, x^2+2x+1, x^3+2x+2, x^3+x^2+x+1$	4
n_4	$x+1, x^2+1$	$0, 1, x+1, x^2+1, x^2+x+1, x^3$	$0, 1, x^2+x+1, x^3$	$x^2+2, x^2+x+1, 2x^2+1, 2x^2+x+1, x^3+x^2+1, x^3+x^2+2$	3
n_5	x^2, x^2+x	$1, x, x^2, x^2+x, x^3, x^3+1$	$1, x, x^3, x^3+1$	$2x^2+2x, x^2+2x+1, 2x^2+2, 2x^2+x+2, x^3+x^2+2, x^3+x^2+x+1$	3
n_6	x^2+1, x^2+x+1	$x, x+1, x^2+1, x^2+x+1, x^3+1, x^3+x$	$x, x+1, x^3+1, x^3+x$	$x^2+2x+1, x^3+x^2+x+1, x^3+x^2+2x+1, 2x^3+x+1, 2x^3+2x+2$	3
n_7	x^2+x, x^3	$x+1, x^2, x^2+x, x^3, x^3+x, x^3+x+1$	$x+1, x^2, x^2+x, x^3+x+1$	$x^3+x^2, x^3+x^2+1, x^3+x^2+x+1, 2x^3+x+1, 2x^3+2x+1, x^3+2x+2$	3
n_8	x^2+x+1, x^3+1	$x^2, x^2+1, x^2+x+1, x^3+1, x^3+x+1, x^3+x^2$	$x^2, x^2+1, x^3+x+1, x^3+x^2$	$x^3+x^2, x^3+x^2+1, x^3+x^2+x+1, 2x^3+1, 2x^3+x+1, 2x^3+x^2$	3
n_9	x^3, x^3+x	$0, x^2+1, x^2+x, x^3, x^3+x, x^3+x^2$	$0, x^2+1, x^2+x, x^3+x^2$	$x^3+1, x^3+x^2+2, x^3+x^2+x+1, 2x^3+1, 2x^3+x+1, 2x^3+x^2+1$	2
n_{10}	x^3+1, x^3+x+1	$0, 1, x^2+x, x^2+x+1, x^3+1, x^3+x+1$	$0, 1, x^4, x^2+x+1$	$x^3+x, x^3+x+1, x^3+x^2+2x, x^3+x^2+2x+1, 2x^3+x+1, 2x^3+2x+1$	1
n_{11}	x^3+x, x^3+x^2	$1, x, x^2+x+1, x^3, x^3+x, x^3+x^2$	$1, x, x^2+x+1, x^3$	$x^3+x^2+2, x^3+2x+1, x^3+x^2+2x+1, 2x^3+x+1, x^3+2x+1, 2x^3+x^2+x+1$	1
n_{12}	$0, x^3+x+1$	$0, x, x+1, x^3, x^3+1, x^3+x+1$	$x, x+1, x^3, x^3+1$	$x^3+x^2, x^3+x^2+x, x^3+x^2+x+1, 2x^3+x^2, 2x^3+x^2+1, 2x^3+x^2+x+1$	0

Table5

Perfect Difference Network					
$\delta = 4, n = \delta^2 + \delta + 1$					
P_E	Linear Nodes	Communication Nodes	Diagonal Nodes	Communication Arcs $2+2(\delta-1)=2\delta$	Distinct Communication Arcs (DCA) $\delta(\delta^2 + \delta + 1)$
n_0	$1, x^4 + x^2$	$1, x^2, x^2+1, x^2+x+1, x^3 + x^2 + x, x^4, x^4 + x^2$	$x^2, x^2+1, x^2+x+1, x^3 + x^2 + x, x^4, x^4+1$	$1, x^2, x^2+1, x^2+x+1, x^3 + x^2 + x, x^4, x^4+1, x^4 + x^2$	8
n_1	$0, x$	$0, x, x^2+1, x^2+x, x^3, x^3 + x^2 + x+1, x^4+1, x^4+x$	$x^2+1, x^2+x, x^3, x^3 + x^2 + x+1, x^4+1, x^4+x$	$1, x+1, x^2+2, x^2+x+1, x^3+1, x^3 + x^2 + x+2, x^4+2, x^4+x+1$	7
n_2	$1, x+1$	$1, x+1, x^2+x, x^2+x+1, x^3+1, x^4, x^4+x, x^4+x+1$	$x^2+x, x^2+x+1, x^3+1, x^4, x^4+x, x^4+x+1$	$x+1, 2x+1, x^2+2x, x^2+2x+1, x^3+x+1, x^4+x, x^4+2x, x^4+2x+1$	7
n_3	x, x^2	$x, x^2, x^2+x+1, x^3, x^3+x, x^4+1, x^4+x+1, x^4 + x^2$	$x^2+x+1, x^3, x^3+x, x^4+1, x^4+x+1, x^4 + x^2$	$2x+1, x^3+x+1, x^3+2x+2, x^3+x+1, x^3+2x+1, x^4+x+2, x^4+2x+2, x^4 + x^2 + x+1$	7
n_4	$x+1, x^2+1$	$0, x+1, x^2+1, x^3, x^3+1, x^3+x+1, x^4+x, x^4 + x^2$	$0, x^3, x^3+1, x^3+x+1, x^4+x, x^4 + x^2$	$x^2, x^2+x+1, 2x^2+1, x^3 + x^2, x^3 + x^2 + 1, x^3+x^2+x+1, x^4 + x^2+x, x^4 + 2x^2$	6
n_5	x^2+1, x^2+x	$1, x^2, x^2+x, x^3+1, x^3+x, x^3 + x^2, x^4+x+1$	$0, 1, x^3+1, x^3+x, x^3 + x^2, x^4+x+1$	$x^2+1, x^2+2, 2x^2+1, 2x^2+x+1, x^3 + x^2 + 1, x^3 + x^2 + x+1, x^4 + x^2+x, x^4 + 2x^2$	5
n_6	x^2+1, x^2+x+1	$x, x^2+1, x^2+x+1, x^3+x, x^3+x+1, x^3 + x^2 + 1, x^4 + x^2$	$1, x, x^3+x, x^3+x+1, x^3 + x^2 + 1, x^4 + x^2$	$x^2+x+1, x^2+2x, 2x^2+x+1, 2x^2+2x+1, x^3 + x^2 + 2x, x^3 + x^2 + 2x+1, x^3 + 2x^2+x+1, x^4 + 2x^2+x$	5
n_7	x^2+x, x^3	$x+1, x^2+x, x^3, x^3+x+1, x^3 + x^2, x^3 + x^2 + x$	$0, x, x+1, x^3+x+1, x^3 + x^2, x^3 + x^2 + x$	$x^2+x+1, x^2+2x+1, 2x^2+2x+2, x^2+2x+2, 2x^2+2x+1, x^3 + x^2 + x+1, x^3 + x^2 + 2x+2, x^3 + 2x^2 + x+1, x^3 + 2x^2 + 2x+1$	4
n_8	x^2+x+1, x^3+1	$x^2, x^2+x+1, x^3+1, x^3 + x^2, x^3 + x^2 + 1, x^3 + x^2 + x+1$	$1, x+1, x^2, x^3 + x^2, x^3 + x^2 + 1, x^3 + x^2 + x+1$	$x^3+1, x^3+x, x^3 + x^2, x^3 + x^2 + x+1, 2x^3+1, 2x^3 + x^2, 2x^3 + x^2 + 1, 2x^3 + x^2 + x+1$	4
n_9	x^3, x^3+x	$x^2+1, x^3, x^3+x, x^3 + x^2 + 1, x^3 + x^2 + x, x^4$	$x, x^2, x^2+1, x^3 + x^2 + 1, x^3 + x^2 + x, x^4$	$x^3 + x + 1, x^3 + x^2 + 1, x^3 + x^2 + 2, 2x^3 + 1, 2x^3 + x + 1, 2x^3 + x^2 + 2, 2x^3 + x^2 + x+1, x^4 + x^3 + 1$	4
n_{10}	x^3+1, x^3+x+1	$x^2+x, x^3+1, x^3+x+1, x^3 + x^2 + x, x^3 + x^2 + x+1, x^4+1$	$x+1, x^2+1, x^2+x, x^3 + x^2 + x, x^3 + x^2 + x+1, x^4+1$	$x^3+2x+1, x^3 + x^2 + x+1, x^3 + x^2 + 2x, 2x^3 + x + 1, 2x^3 + 2x + 1, 2x^3 + x^2 + 2x, 2x^3 + x^2 + 2x+1, x^4 + x^3 + x+1$	4

n ₁₁	x^3+x, x^3+x^2	$x^2+x+1, x^3+x, x^3+x^2, x^3+x^2+x+1, x^4, x^4+x$	$x^2, x^2+x, x^2+x+1, x^3+x^2+x+1, x^4, x^4+x$	$x^3+x^2+x+1, x^3+x^2+2x+1, x^3+x^2+2x+2, 2x^3+2x+1, 2x^3+x^2+2x+1, x^4+x^3+2x+1$	4
n ₁₂	x^3+x+1, x^3+x^2+1	$x^3, x^3+x+1, x^3+x^2+1, x^3+x^2+x+1, x^4, x^4+1$	$x^2+1, x^2+x+1, x^3, x^3+x^2+x+1, x^4, x^4+1$	$x^3+2x^2+1, x^3+2x^2+x+1, 2x^3+x^2, 2x^3+x^2+x+1, 2x^3+2x^2+1, 2x^3+2x^2+x+1, x^4+x^3+x^2+1$	4
n ₁₃	x^3+x^2, x^3+x^2+x	$x^3+1, x^3+x^2, x^3+x^2+x, x^4+1, x^4+x, x^4+x^2$	$x^2+x, x^3, x^3+1, x^4+1, x^4+x, x^4+x^2$	$x^3+2x^2+x+1, 2x^3+x^2+1, 2x^3+x^2+2, 2x^3+2x^2+2, 2x^3+2x^2+1, 2x^3+2x^2+x+1, x^4+x^3+x+1, x^4+x^3+2x^2+1$	4
n ₁₄	x^3+x^2+1, x^3+x^2+x+1	$0, x^3+x, x^3+x^2+1, x^3+x^2+x+1, x^4+x, x^4+x+1$	$0, x^2+x+1, x^3+1, x^3+x, x^4+x, x^4+x+1$	$x^3+2x^2+2x+1, 2x^3+x^2+x+1, x^3+x^2+x, 2x^3+x^2+2x, 2x^3+2x^2+x+1, 2x^3+2x^2+2x+1, x^4+x^3+x^2+2x+1$	3
n ₁₅	x^3+x^2+x, x^4	$1, x^3+x+1, x^3+x^2+x, x^4, x^4+x+1, x^4+x^2$	$1, x^3, x^3+x, x^3+x+1, x^4+x+1, x^4+x^2$	$2x^3+x^2+x+1, 2x^3+x^2+2x+1, x^3+x^2+x+2, 2x^3+x^2+2x+2, 2x^3+2x^2+2x+1, x^4+x^3+x^2+x+1, x^4+x^3+x^2+2x+2, x^4+x^3+2x^2+x+1$	3
n ₁₆	x^3+x^2+x+1, x^4+1	$0, x, x^3+x^2, x^3+x^2+x+1, x^4+1, x^4+x^2$	$0, x, x^3+1, x^3+x+1, x^3+x^2, x^4+x^2$	$x^4, x^4+x, x^4+x^3+1, x^4+x^3+x+1, x^4+x^3+x^2, x^4+x^3+x^2+1, 2x^4+1, 2x^4+x^2$	2
n ₁₇	x^4, x^4+x	$1, x+1, x^3+x^2+1, x^4, x^4+x$	$0, 1, x+1, x^3+x, x^3+x^2, x^3+x^2+1$	$x^4+2, x^4+x+2, x^4+x^3+x^2+2, 2x^4+1, 2x^4+x+1, x^4+1, x^4+x^3+x+1, x^4+x^3+x^2+1$	1
n ₁₈	x^4+1, x^4+x+1	$x, x^2, x^3+x^2+x, x^4+1, x^4+x+1$	$1, x, x^2, x^3+x+1, x^3+x^2+1, x^3+x^2+x$	$x^4+x+1, x^4+2x, x^4+x+x, x^4+x^3+2x+1, x^4+x^3+x^2+1, x^4+x^3+x^2+2x, 2x^4+x+1, 2x^4+2x+1$	1
n ₁₉	x^4+x, x^4+x^2	$x+1, x^2+1, x^3+x^2+x+1, x^4+x, x^4+x^2$	$x, x+1, x^2+1, x^3+x^2, x^3+x^2+x, x^3+x^2+x+1$	$x^4+2x+1, x^4+x^3+x^2+x+1, x^4+x^3+x^2+x+2, x^4+2x+1, x^4+x^2+x+1, x^4+x^3+x^2+2x+2, 2x^4+2x+1, 2x^4+x^2+x+1$	1
n ₂₀	$0, x^4+x+1$	$0, x^2, x^2+x, x^4, x^4+x+1$	$x+1, x+1, x^2+x, x^3+x^2+1, x^3+x^2+x+1, x^4$	$x^4+x^2, x^4+2x^2, x^4+2x^2+x, 2x^4+x^2, 2x^4+x^2+x+1, x^4+x^2+x+1, x^4+x^3+2x^2+1, x^4+x^3+2x^2+x+1$	0

References

- [1] Parhami, Behrooz and Rakaov, Mikhail."Perfect Difference Networks and Related Interconnection structures for Parallel and Distributed Systems". IEEE Journal on Parallel and Distributed Systems 16(8),pp 725-736, 2005.
- [2] Elements of Parallel Computing and Architecture Book, Interconnection Network, Unit-3, IGNOU.
- [3] Parhami, Behrooz and Rakaov, Mikhail."Perfect Difference Networks and Graphs and their applications", 2005. Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA), 614-615, Athens, WorldComp, (2013).
- [4] Wu, Xingfu and Sun, Xian-He, "Performance Modeling for Interconnection Networks", IEEE, 2000.
- [5] Katare, R.K and Chaudhari, N.S. "A Comparative study of hypercube and perfect difference network for parallel and distributed systems and its application to sparse linear system". Vladimir Journal of Computer Information Sciences, Vol2, Sandipani Academy, Ujjain(M.P.), India pp13-30, 2007.
- [6] Katare, R.K., Chaudhari, N.S., Mugal, S.A., Verma, S.K., Imran, S., Raina, R.R. "Study of Link Utilization of Perfect Difference Network and Hypercube." Conference on "FECS", The 2013 World Congress in Computer Science, Computer Engineering and Applied Computing, Las Vegas, Nevada, USA, July 22-25, 2013.
- [7] Pan, Jianli, A Survey of Network Simulation Tools: Current Status and Future Developments. Report, Nov 2008. <http://www.cse.wustl.edu/~jain/cse567-08/ftp/simtools/>
- [8] Priyanka Wankar, Anand Sharma, and Ruchika Sinhal, "Performance Analysis and Implementation of Perfect Difference Network Using NS-2", Vol. 2 Issue 1 January 2013, ISSN: 2278-621X.
- [9] Jyotsana Nandagaoli, Jagdish W. Bakal, "Performance Analysis for Perfect Difference Network and N Complete Network using NS-2," International Journal of Computer Applications (0975 – 8887) Volume 128 – No.15, October 2015.
- [10] Ms. Jyotsana Nandagaoli, Dr. Jagdish W. Bakal, "Study of Perfect Difference Network: Review," International Journal Of Engineering And Computer Science ISSN:2319-7242 Volume 3 Issue 6 June, 2014 Page No. 6465-6469.
- [11] Yogesh Golhar, R.K. Krishna and Mahendra A. Gaikwad, "Implementation & Throughput Analysis of Perfect Difference Network (PDN) in Wired Environment," IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 1, No 1, January 2012.
- [12] Mrs. Meenakshi, "Impact of Network Size & Link Bandwidth in Wired TCP & UDP Network Topologies", International Journal of Engineering Research and General Science Volume 2, Issue 5, August – September 2014 ISSN 2091-2730.
- [13] <https://www.nsnam.org/wiki/PyViz>
- [14] https://www.nsnam.org/wiki/NetAnim_3.105
- [15] <http://www.researchgate.net/file.PostFileLoader.html/ns2vsns3.pdf>
- [16] <http://www.slideshare.net/raulhada/ns3-overview>