DESIGN AND MODELING OF DIFFERENT ELEMENTS ON A DISTINCT TYPE OF MULTICOMPUTER NETWORKS, DUAL-RADIO WIRELESS HYPERMESH BASED ON SYSTEM C METHODOLOGY

Syed Sajid Quammaruddin Javid [1], Dr. Syed Abdul Sattar[2], M V. Ramana Murthy[3]

[1]Research Scholar, Department of Computer Science, Rayalaseema University, Kurnool, A.P. India.

[2]Principal, Nawab Shah Alam Khan College of Engineering & Technology, Hyderabad-024.

[3]Department of Mathematics & Computer Science, Osmania University, Hyderabad.(R), M.G.I.T, Hyderabad.(P)

Received: February 18, 2019 Accepted: March 30, 2019

ABSTRACT: *This paper presents an original development methodology for the use of System C to compare two different networks at the system level. A single cluster of a single channel network based on CSMA and multi-channel network cluster based on non-overlapping channels available for each node have been developed. Recently, multi-radio mesh technology in wireless networks has been put under extensive research. This is because of its potential to overcome the inherent wireless multi-hop throughput, scalability and latency problems caused by the half-duplex nature of the IEEE 802.11. This paper also introduces a design and modeling of different elements on a distinct type of multicomputer networks, the dual-radio wireless hypermesh, based on System C methodology.*

Key Words: *dual-radio wireless hypermesh, multicomputer networks, wsn*

I. INTRODUCTION

The goal of the topological design of a computer communication network is to achieve a specified performance at a minimal cost. A reasonable approach is to start with a potential network topology and see if it satisfies the connectivity and delay constraints. If not, the starting network topology is subjected to a small modification yielding a slightly different network, which is now checked to see if it is better. If a better network is found, it is used as the base for more network configurations. If the network resulting from the modification is not better, the original network is modified or reconfigured in some other way. The traditional ways of network designing mainly depend on experience. However, the ways that simply depend on experience to design networks cannot keep up with the development step of new systems. Moreover, traditional design methodologies increasingly fail to handle such reasoning in a cost- and time-effective manner, when product time-to-market is the key to success.

One of the most promising solutions to this problem are system-level modeling and simulation for instance System C methodology which is covered in this work. My methodology reveals that based on the popular System C design methodology fast simulation and an excellent path to implementation with potential reuse for different implementation blocks can be easily done. Another important application of System C modeling techniques is to perform meaningful comparative studies of different protocols, or new implementations to determine which communication scenario performs better and the ability to modify models to test system sensitivity and tune performance. The drawback is that models of well-known protocols have to be completely rewritten in the system description language together with models of nodes outside the design scope.

CHANNEL MODULE REFINEMENT

The channel module developed is initially refined in section to support contention and non-contention based wireless communication. Moreover, it will support different noise and corrupting methods. The initial results conducted in section shows that the packet transfer delay requires additional clock cycles to cross the communication medium. To reduce the number of communication cycles a packet can take to cross the communication medium, the communication medium needs further refinement. This refinement can be done using one of distinct capabilities and features of System C. System C has custom hierarchical channels mechanism. Hierarchical Channels are intended to model quite complex behaviors such as PC WE Express, Hyper Transport, or Advanced Microcontroller Bus Architecture (AMBA). Hierarchical channels and the interface concept forward a very powerful refinement strategy. It is easy to replace a channel with another one; they only have to share the same interfaces.

Primitive channels, which were used in my initial implementation phase, on the other hand are intended to provide very simple and fast communications. (e.g., sc_signal provide a piece of wire behavior). To build complex system level models, System C defines hierarchical channels as modules that implement one or more interfaces, and serves as a container for communication functionality. An interface provides a declaration of methods for accessing a given channel. No implementations or data are provided in a System C interface. In sc_signal, for example, the interfaces are defined by two classes sc_signal_in_if and sc_signal_out_if, and these define methods (e.g., read () and write ()).

Separating the definition of an interface from its methods implementation, System C has a unique coding style in which communication is separated from behavior, a key feature to facilitate refinement from one level of abstraction to another. Channels in System C create connections between module ports allowing modules to communicate. Figure 1 shows a System C hierarchical channel representation. A port acts as an agent that forwards method calls up to the channel on behalf of the calling module.

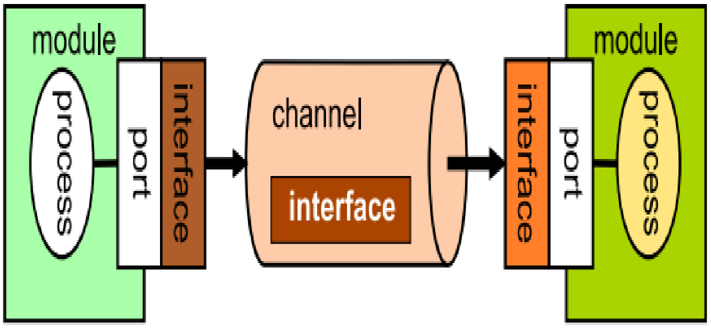

Figure 1: Hierarchical channels representation

Hierarchical channels are implemented as modules in System C: in fact, they are derived from sc_module. Primitive channels have their own base class, sc_prim_channel. This section will show only the implementation of the hierarchical channel model used in this work. Figure 2 shows the communication medium refinement process done in System C.

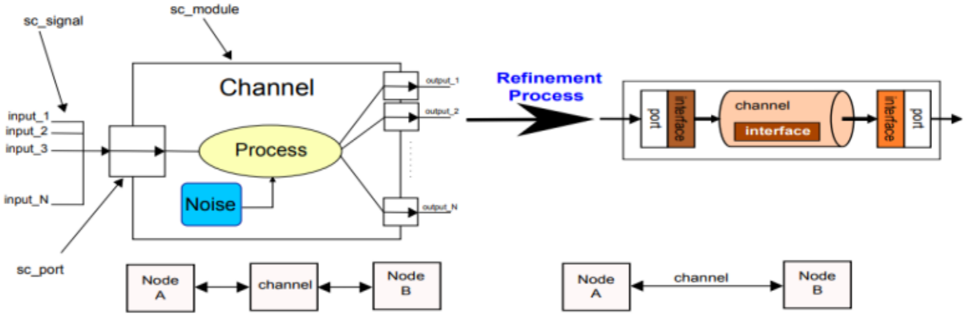

Figure 2: Channel refinement process

II. SHARED CHANNEL NETWORK MODEL ANALYTICAL MODEL

WE consider the shared channel network model developed is to provide an analytical model of the results obtained from the developed System C model. It has been developed and the performance properties are investigated for a number of configurations of the system parameters (e.g., packet sizes, number of nodes). The node structure for this configuration is further shown in figure 3.

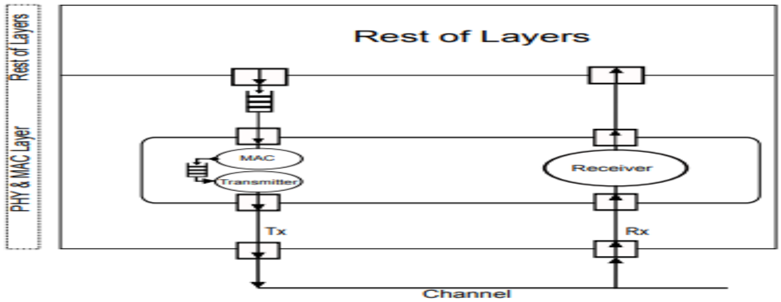

Figure 3: Node structure in a one-dimensional single shared channel cluster

For modeling-based approaches to be effective, it is important to understand how well the analytical models are able to capture the performance as seen in simulation model, which will provide a solid model for a future development. With this in mind, in this section we provide some preliminary results on a comparison of analytical model predictions with System C model performance results on simple shared communication medium network. In my model we do not consider channel errors and packet retransmissions. However, we focused on contention model in single-hop wireless network model.

Delay Analysis for a Noiseless Channel

The latency or the packet delay T using results from M/M/1 queuing theory can be computed. This system has a Poisson input (with an average arrival rate λ) and the average service time is **$\bar{T} = 1/\mu$.**

The expected number of entries in the queue is given by

$$\bar{N} = \frac{\rho}{1-\rho} = \frac{\lambda}{\mu - \lambda}.$$

With variance

$$\sigma_N^2 = \frac{\rho}{(1-\rho)^2}$$

Using Little's result, which relates the average number in the system to the average arrival rate and the average time spent in the system, namely $\bar{N} = \lambda T_w$, we can obtain Tw, the expected waiting time as follows:

$$T_w = \frac{\bar{N}}{\lambda}$$

$$T_w = \left(\frac{\rho}{1-\rho}\right)\left(\frac{1}{\lambda}\right)$$

When channels are error free, each nodal delay becomes similar to the system delay of an M/M/1 queuing system. So we have

$$T_w = \frac{1/\mu}{1-\rho}$$

Using $\rho = \lambda/\mu$, we get the latency for a shared communication medium as:

$$L = \frac{T}{1 - n\gamma T}$$

III. Results of shared channel network

The SystemC model has been compared against the M/M/1 queuing model that mimics the behaviour of a shared communication channel as a single queue with multiple inputs at the phit level. The x-axis in the figures represents the rate at which a node injects packets into the network in packets per cycle. The y-axis gives the mean packet latency to cross the network. The statistics gathering was inhibited for the first 3000 packets to avoid distortions due to the initial start-up conditions. The simulation stops when it reaches the accuracy required, which is set to be 0.05 relative half width, i.e., a confidence limit of 95%.

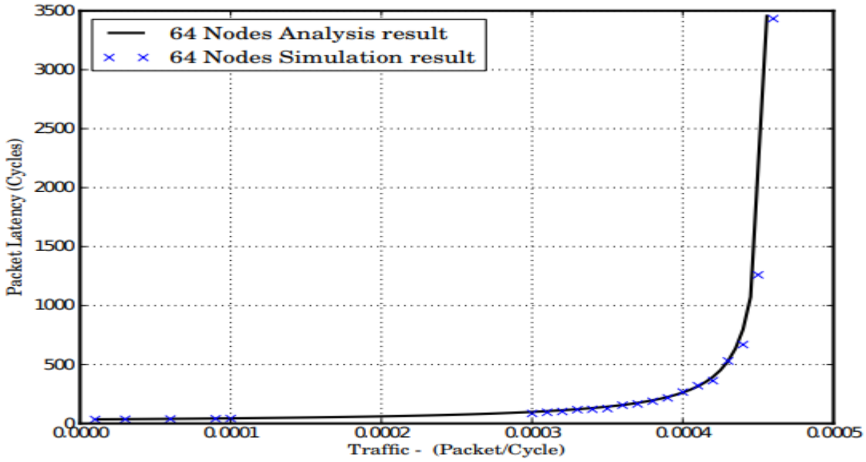

Figure 4: Latency vs. throughput of 64 nodes with packet size of 34 phits

Numerous experiments have been performed for several combinations of network sizes, packet lengths to get a good agreement between the models. However, for the sake of specific illustration Figure 4 and Figure 5 depict latency results for 64 nodes and 144 nodes network model. The figures reveal that in all cases, the analytical model predicts the mean packet latency with a good degree of accuracy in the all network regions, i.e., in the steady state region, heavy traffic region and in the saturation region. In addition, as shown in Figure 6 that the network saturates earlier as the number of nodes increase. This is due to the increase in channel acquisition time due to the contention on accessing the medium.

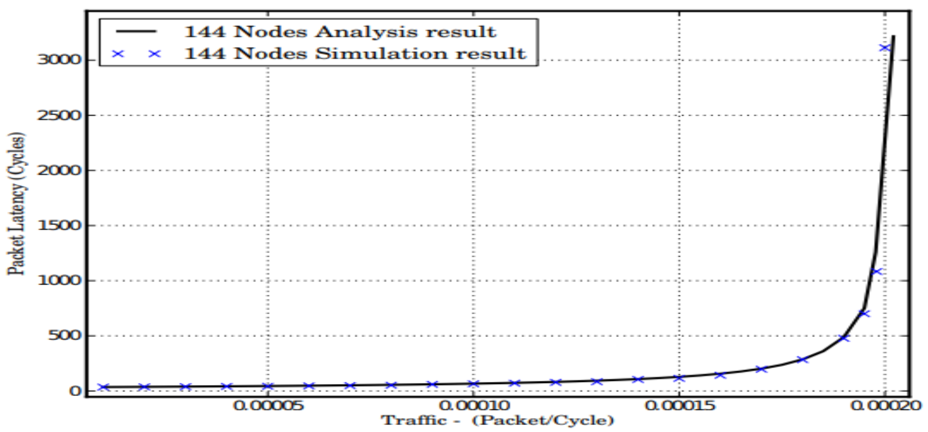

Figure 5 : Latency vs. throughput of 144 nodes with packet size of 34 phits

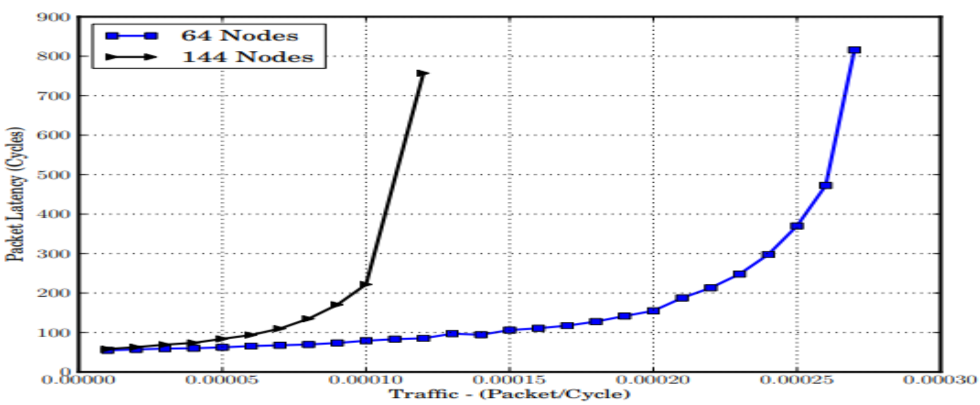

Figure 6: Latency vs. throughput for 64 and 144 nodes with a packet size of 54 phits

IV. ONE-DIMENSIONAL MULTIPLE CHANNELS NETWORK

The shared medium is no more a single channel but a group of channels. Full connection of the nodes to all the channels is performed, both for transmission and for reception. The one-dimensional multiple channels network cluster has been proposed is depicted in Figure 7. Their proposed configuration is known as the Distributed Crossbar Switch Hypermesh (DCSH). This configuration is from the hypergraph type of networks. The basic structure consisting of N nodes connected by unidirectional and not shared channels. Every node has its own channel that connects it to the other (N-1) nodes in the cluster. These multiple channels do not suffer the performance penalties associated with contention based single channel structure. In addition, this structure will contribute to maximise throughput and to minimise latency. According to Old-Khaoua et al. their architecture proposal also offers scalability. More network capacity is easily obtained by increasing the number of channels; however the receiver realisation should be modified to admit more channels.

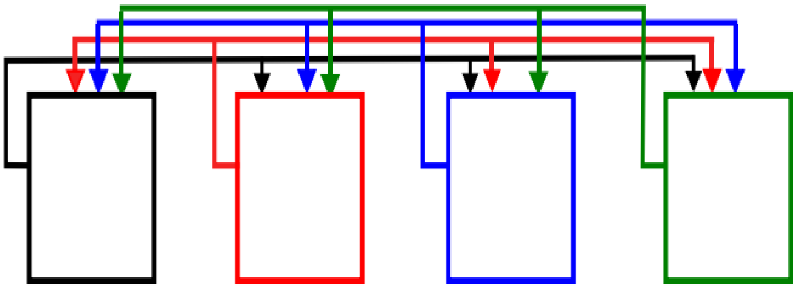

Figure 7: One-dimensional multiple channels network cluster

Figure 8 shows the node structure in a one-dimensional multiple channels cluster. The node structure is also similar to that of the shared communication channel. Except that there are (k-1) inputs in a one-dimensional multiple channels and one output channel. For simplicity only the difference from the shared channel node structure that has been presented, is shown here.

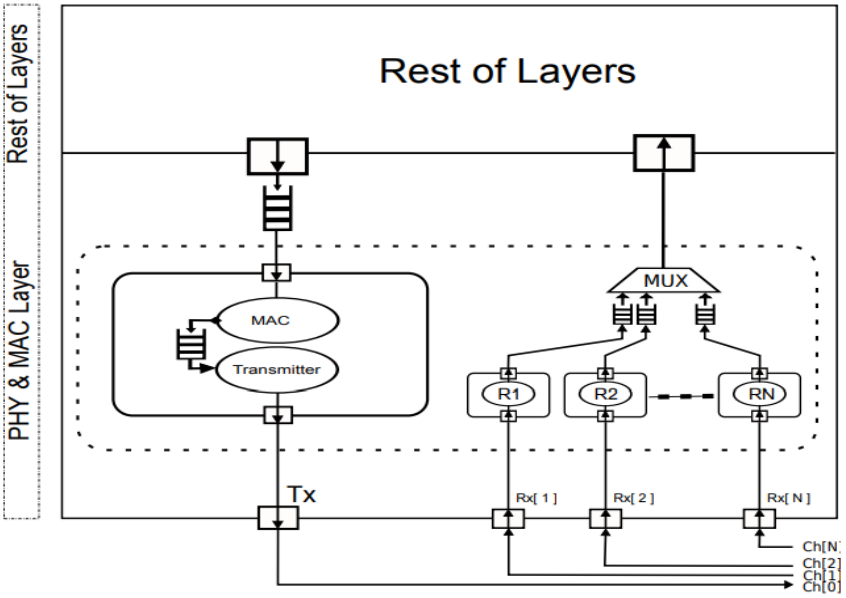

Figure 8: Node structure in a one-dimensional multiple channels cluster

Buffers are provided at both inputs and outputs of the PHY layer. There is one (N-1)-to-1 multiplexer per cluster. An input queue consists of a packet buffer, each of which has enough storage to temporarily hold a few numbers of packets. This multiplexer is to prevent a contention, if multiple packets arrive at the same time to the multiplexer and find it busy

Results of one-dimensional multiple channels network

Figure 9 shows the relationship between the offered traffic load and the latency. The horizontal axis in the figures shows the traffic in Packets/Cycls while the vertical axis shows the mean packet latency to cross the network from source to destination. Packet length is 34 symbols or phits. This packet sizes were chosen to represent a short size packet, and have been used in similar studies. The network comparisons take in to account the bisection width of the network. As described the bisection width across a cluster in DCSH with is givin by

$$B_{DCSH} = NW_{DCSH}$$

To compare the results of the two networks the bisection width is held constant for both networks, the channel width of the first and second DCSH networks are given by

$$B_{DCSH1} = N_1 W_{DCSH1}$$

$$B_{DCSH2} = N_2 W_{DCSH2}$$

If the bisection width is held constant for both networks then we have

$$W_{DCSH2} = \frac{N_1}{N_2} W_{DCSH1}$$

The figure reveals that in all cases for small and large networks the network saturates at different points. Because the network under test has different network sizes. Moreover, the figure shows that as the network size increase the network latency is becoming higher at low traffic rates. This is due to the small channel widths as the network become large.

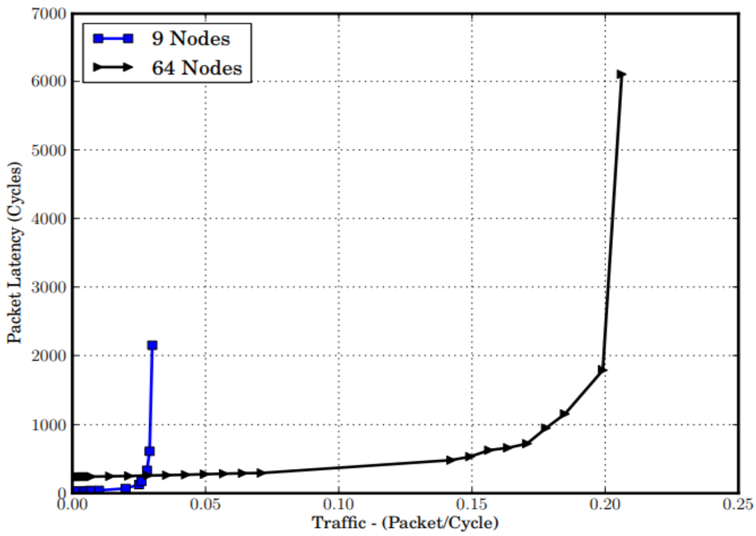

Figure 9: Latency versus Traffic rate for 9and 64 nodes 1D multichannel network model.

V. PERFORMANCE COMPARISON

Performance comparison of any interconnection network should take into account implementation costs to give a meaningful evaluation. Several criteria have been proposed in the literature to make fair comparison between networks of fixed cost including chip pin-out and wiring density. Bisection width as a measure of network cost has been used and other researchers thereafter to account for wiring density. However, we use in my discussions the term channel to refer to a link as a wireless channel. The bisection width of a network is the minimum number of channels cut when the network is divided into two equal halves. One way to approximate the channel density of a network is the calculation of the bisection width. The bisection width across a cluster in Bus and DCSH with a channel width of WBus, WDCSH, respectively, are given by

$$B_{Bus} = W_{Bus}$$

$$B_{DCSH} = NW_{DCSH}$$

If the bisection width is held constant for both networks, the channel width of the DCSH in terms of shared media (Bus) channel is given by

$$W_{DCSH} = \frac{1}{N} W_{Bus}$$

For illustration we will use network sizes of 64 and a packet size of 34 phits, typical figures which has been used but the results shown in figure 10 are applicable across other network sizes and packet lengths.

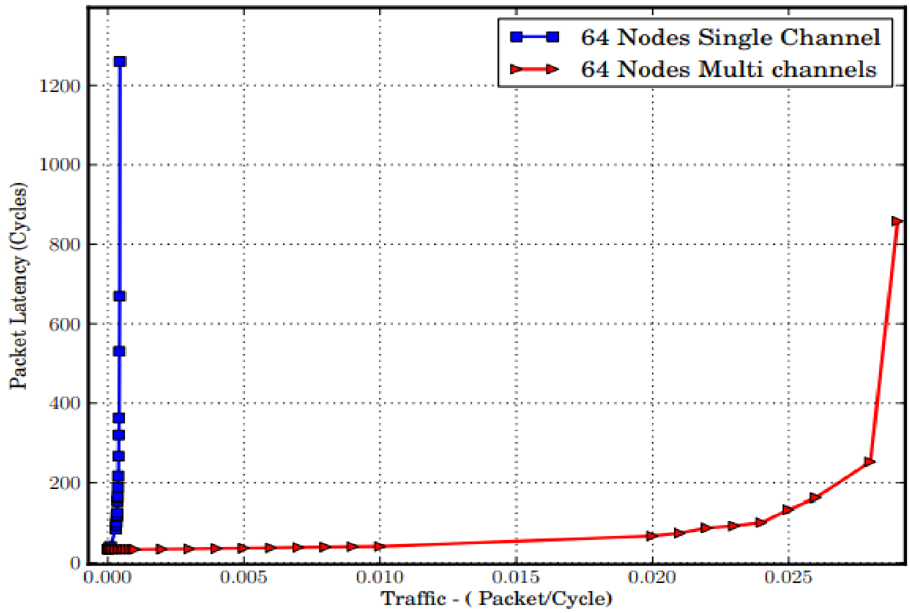

Figure 10: Latency comparison for 64 nodes network model

VI. DUAL-RADIOS HYPERMESH NETWORK BASED ON CSMA PROTOCOL

The hepermesh is a well-known topology that belongs to the hypergraph family of networks. Hypermeshes have been proposed as potential alternatives to the graph networks for the future System Area Networks. In this work, we consider a two dimensional dual-radio wireless hypermesh network, where each router node is equipped with two radio interfaces and two non-overlapping channels are available for each node. We address the problem of assigning channels to communication links in the network with the objective of keeping overall network latency low and provide a relatively high throughput.

The simulations and analysis have shown that my design achieves a significant increase in network throughput with less average network latency for large number of communication nodes, compared with the CSMA shared channel model, which is currently the de facto MAC protocol for most wireless networks. My simulations have been validated analytically to show the accuracy of the developed model. In addition, simulation results have shown that the wireless hypermesh outperforms shared medium wireless networks under the constant total bandwidth argument, especially in large networks.

The increasing demand for high performance computing from the military and research centers in industry and academia for the simulation of complex problems makes SANs a very interesting research area. SANs are Hardware/Software systems designed to perform specific applications in which network communications are essential. For this reason, their processing functionality is strictly connected with communication functionality. Typically, the processing part is implemented through CPU, memory, and application-specific components; the communication part is implemented through HW components (e.g., network interface and wired or wireless links) and SW components (e.g., the protocol stack). The design of SANs is the context of this thesis. It requires the capability of modeling and simulating both their behaviour/architecture and the complex communication environment in which SANs operate.

The success of Large SANs for multicomputer networks is highly dependent on the design and the efficiency of the interconnection network used. Interconnection network is constructed normally from routers, switches and channels. It provides the means by which the information is transferred between nodes. The design and implementation of a communication network for applications such as scientific computing research applications, for instance, fluid dynamics or finite element methods, modeling of nuclear explosions, climate modeling, and others has a direct impact on the cost and performance of the whole system.

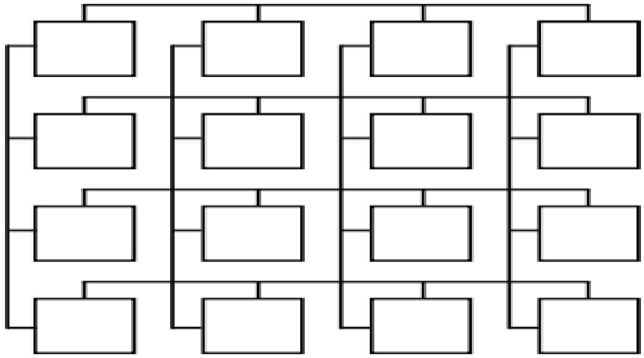

Figure 11: A two-dimensional hypermesh

There are several network architectures and topologies for implementing interconnection networks have been introduced in literature to use an interconnection networks with mixed level of success. Some of the most important networks are: Fully connected or all-to-all, a Circular Ring, a Star, a Binary tree, Mesh (Torus), Hypermeshes, or even Random networks. Most of the above mentioned interconnections are designed to implement wired networks, which can have more applications in a wireless sensors network. Recently hypermeshes have been shown to be a good promising candidate for interconnection networks of parallel systems. Hypermeshes have desirable features over other interconnection networks such as a low diameter, high bandwidth, and low latency network. Those advantages make the hypermeshes to embed naturally a wide range of communication patterns. A number of different hypermesh implementations have been suggested in the literature including shared buses, crossbar switches, and distributed crossbar switch with different implementation technologies, costs and constraints. Figure 11 shows a typical example of a hypermesh implementation, which is the spanning bus hypercube (SBH) proposed. And it has been further studied. In their introduced model nodes in a given dimension are connected together with a shared bus. A bus in the SBH is time multiplexed among the cluster nodes, which is not suitable for a large number of wirelessly communicating nodes.

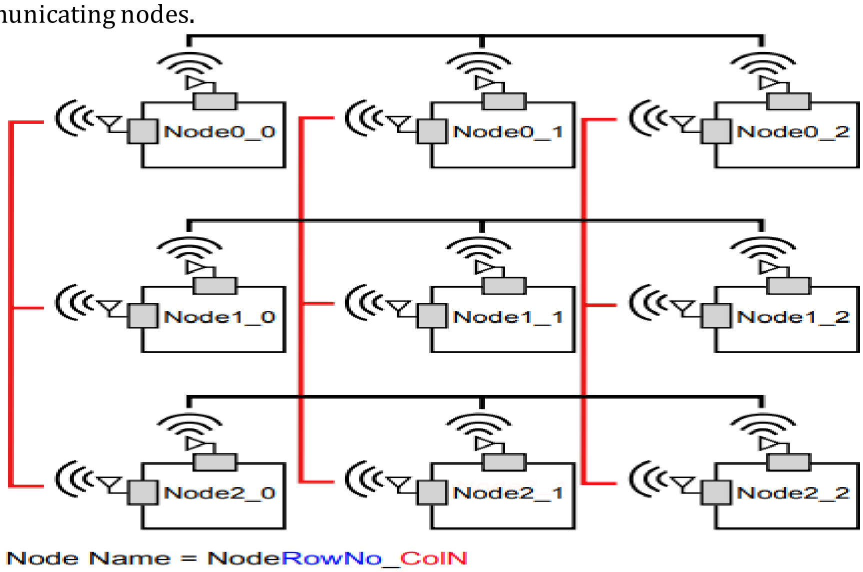

Figure 12: Dual-Radio Hypermesh Topology

VII. DESIGN AND IMPLEMENTATION

When two or more radio interfaces are placed on a node, we are faced with a number of choices as to the way they are interconnected. Two main alternatives are bridging at the link layer and connecting the interfaces at the network layer. While bridging at the link-layer may be acceptable in a wired network, in its wireless counterpart traffic unnecessarily relayed can significantly degrade performance. In this work, we adopted a more generic solution by connecting the interfaces at the routing level. The individual PHYs act as separate entities with different addresses for routing purposes. In this section, we further expand on this design and describe my implementation of a dual-interface node in System C. The communication nodes are the main construction units of the network system, which will communicate with each other through the underlying interconnection network. The nodes of the network are modeled as modules with different methods, which will reflect the application, network and DLC and PHY of the standard reference model of an OSI/ISO. Each node interacts with the network to send or receive data packets. Each node is assumed to have two baseband transmitters and receivers for communication with other network nodes as shown in Figure 13.

The two-dimensional node structures have been modeled by adding the necessary components to the one-dimensional model. By adding a network layer for routing and switching data packets and another PHY layer for transmission of data packets to the second dimension the hypermesh node has been constructed. Throughout the analysis and simulation, the following assumptions have been made.

Traffic generated by nodes independently of each other, and follows a Poisson process of a mean rate λ (packets/cycle). Furthermore destinations are chosen randomly and independently. This assumption is widely used in the literature as it greatly simplifies the analysis.

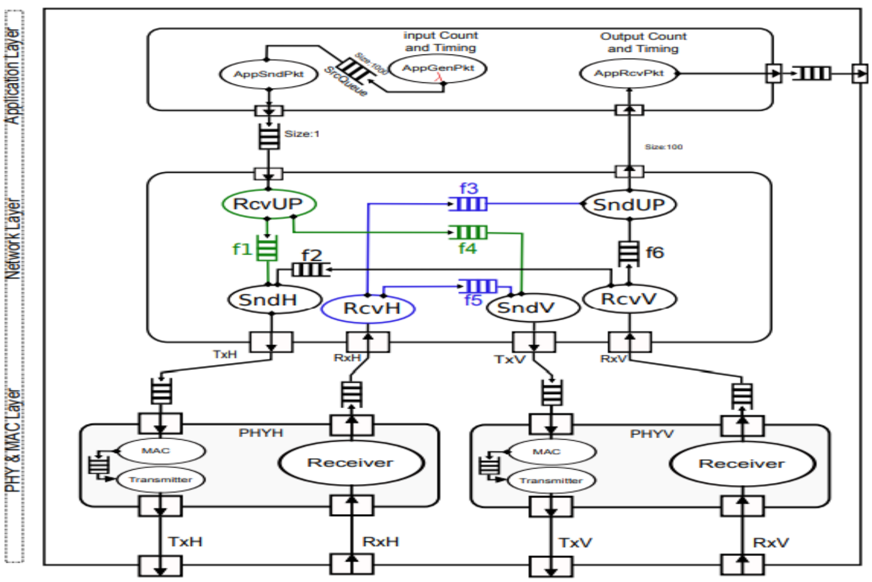

Figure 13: Two-Dimensional model of the communication node

- All packets generated are of equal length, M phits, each of which requires M-cycle transmission time.
- Packet destinations are uniformly distributed across the network unless specified otherwise. Although many network evaluation studies make this simplifying assumption, it is rarely true in practice.
- Routing is restricted; packets are transmitted between routers using packet switching (store-and-forward). Dimension-ordered routing, where packets visit network dimensions in a strict order (it is assumed here that dimensions are visited in a decreasing order), is sufficient to avoid packet deadlock.
- Balanced network to avoid complexity in my analysis and in my proposed topology this assumption is easily realizable. A network is said to be balanced if the utilization factor of all of its channels is the same.
- Packet queues are assumed to have infinite capacity. This assumption has been shown to be realistic under uniform traffic.
- Negligible channel propagation delay

VIII. RESULTS AND PERFORMANCE COMPARISON

A performance comparison has been carried out between the two-dimensional wireless hypermesh to the shared communication medium for a fixed size N and equal implementation cost. Each channel entering a node has a bandwidth of W bits/Cycle. All packets generated are of equal length, M phits, each of which requires one-cycle to be transmitted. A packet length M phits is broken into B = M × W phits, each of which contains W bits. Implementation cost should be taken into account to compare different network configuration to give a meaningful evaluation of the network performance. There are several measures have been proposed in the literature to make fair comparison between networks of fixed cost including pin-out and wiring density.

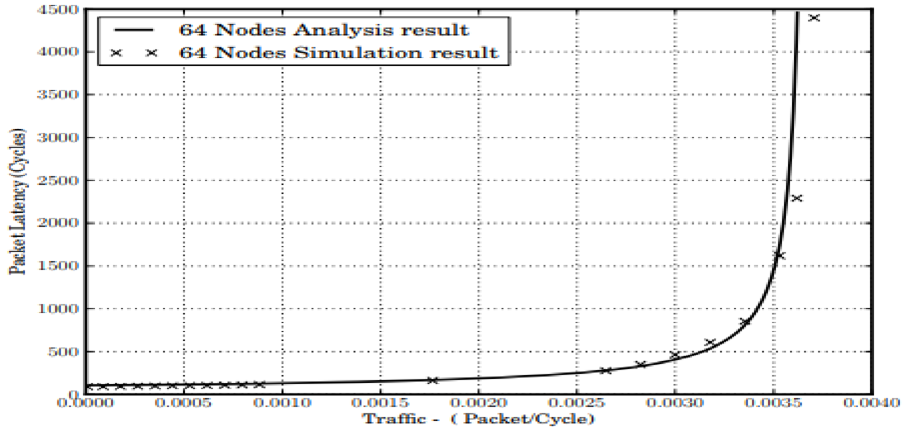

Figure 14: Latency vs traffic for 2D Hypermesh for 64 nodes

Under a constant pin-out argument the total bandwidth entering a node is the same for both networks under comparison. As described in section, to approximate the channels density of a network is the calculation of the bisection width. The bisection width across a cluster in Bus and SBH with a channel width of WBus, WSBH, respectively, are given by

$$B_{Bus} = W_{Bus}$$

$$B_{SBH} = \frac{N}{k} W_{SBH}$$

If the bisection width is held constant for both networks, the channel width of the SBH in terms of shared media (Bus) channel is given by

$$W_{SBH} = \frac{k}{N} W_{Bus}$$

Figure 14 and Figure 15 respectively compare the network latency predicted by our model, and through simulation for 64 and 144 network nodes.

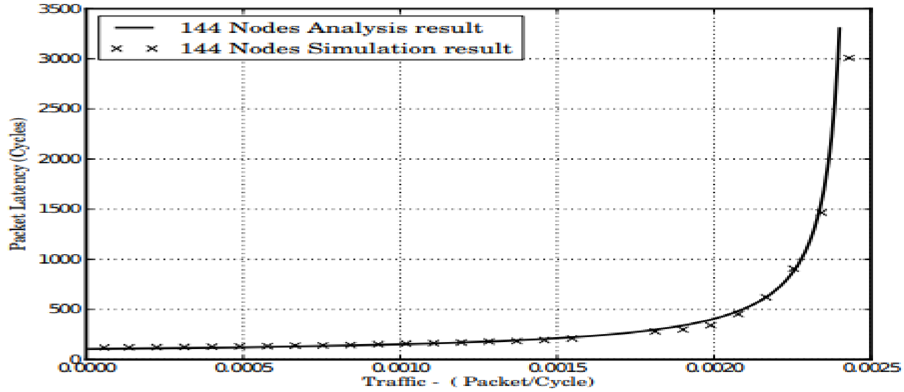

Figure 15: Latency vs traffic for 2D Hypermesh for 144 nodes

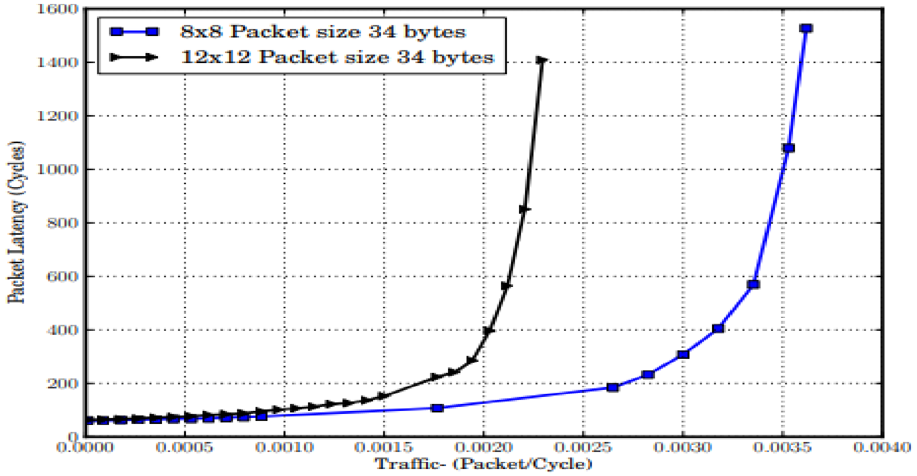

Figure 16: Latency vs traffic for 2D Hypermesh for 64 and 144 nodes

To mimic the Hypermesh and make the comparison with the shared medium network the following simulation sitting have been used:
Number of nodes N = 144 node.
Packet size for both networks is (34 Symbols each of which is 8 bits)
Number of channels in the hypermesh is C = we + j = 12 + 12 = 24 channel, where we and j is the number of channels in each dimension we = j = 12

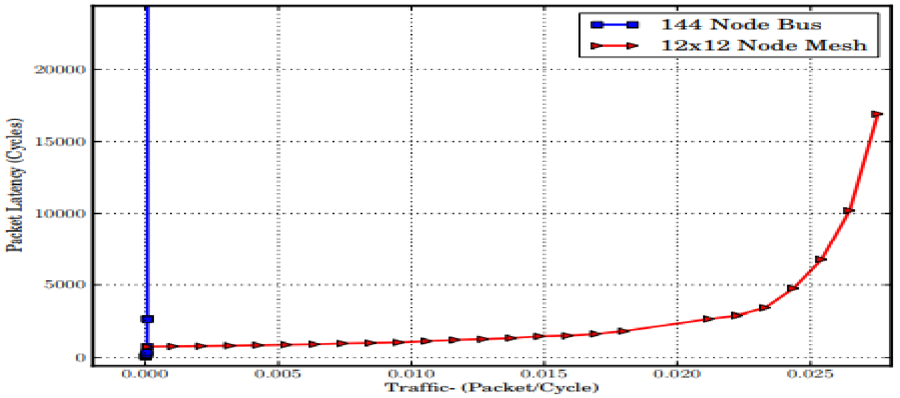

Figure 17: Latency vs traffic for 2D and 1D Hypermesh

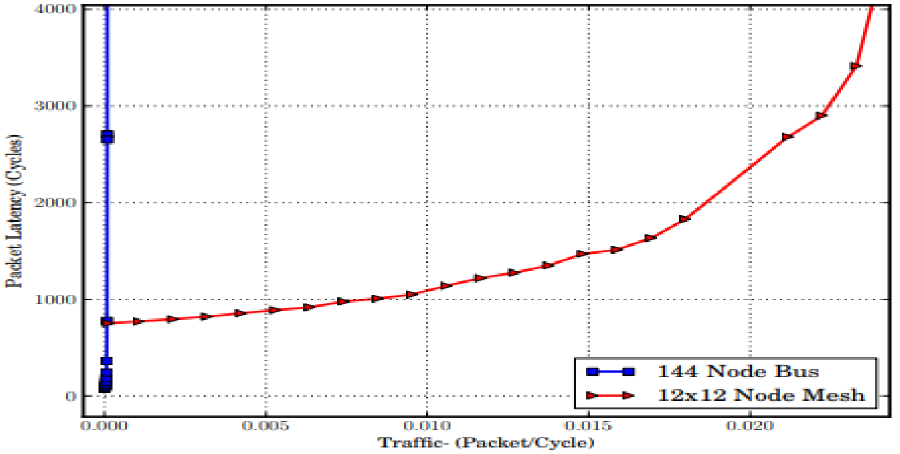

Figure 18: For clarity the same graph but with different scale shows the latency vs traffic for 2D and 1D Hypermesh

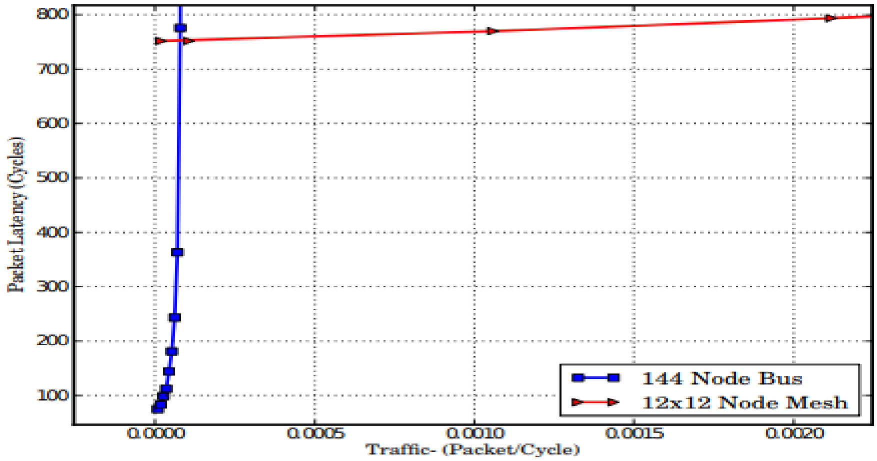

Figure 19: For clarity the same graph but with extra zoom on the crossing point of the curves.

We again note that these results were obtained without introducing noise into the communication channels of the system. Although the SystemC model is designed so that random noise and signal fading can be easily introduced and we showed results of the introduction of such noise provides a detailed noise analysis of multichannel and DRWH systems is outside the scope of this thesis. However it is important to have some indication of the type and magnitude of the effects of noise upon such systems. The prime effect of such noise is to increase both the traffic in a system, and the latency, due to the need for data re-transmission events. The effect of noise due to multipath transmission was estimated at between 20-30dB. As a worked example, considering equations 2.8-2.11, if the communication links were set to a bit error rate of 1/100, signal fading of 20dB in the noisy environment would raise the bit error rate to ¼. Assuming hardware error correction led to an equivalent change in packet loss from 1% to 25% and each packet re-transmission took twice the latency of the original transmission (assuming a message returned to the transmitter, but minimal computational cost) then the overall packet latency would increase by 50%, with a commensurate increase in network congestion. These noise induced increases in latency and network congestion are not trivial, and emphasize the need for networks, like the DRWH, which are robust in the face of high congestion, to be used.

IX. CONCLUSION

In this paper we addressed the performance comparison of contention based wireless networks, and Multi-channel network, using System C design methodology. For this purpose, we developed a conceptual model of the communication node of the multichannel network based on my previous node architecture and the communication channel which was introduced and compared my findings analytically and by appropriate simulations. This paper has investigated the relative performance merit of the two different implementations. The results have revealed that the multi-channel network has superior performance characteristics over the shared communication network.

Simulation results have shown that the wireless hypermesh outperform shared medium wireless networks under the constant total bandwidth argument, especially in large networks. In addition to the simulation results, the performance improvements have been compared against theory.

REFERENCES

1. Heo, N., & Varshney, P. K. Energy-efficient deployment of intelligent mobile sensor networks. IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, Vol.35, pp 78-92, 2005.
2. Karaboga, D., Okdem, S., & Ozturk, C. Cluster based wireless sensor network routing using artificial bee colony algorithm. Wireless Networks, Vol.18, pp847-860,2011.
3. Zheng, W., &Luo, D. Routing in wireless sensor network using Artificial Bee Colony algorithm. In Wireless Communication and Sensor Network (WCSN), 2014 International Conference, pp 280-284,2014.
4. Khoshkangini, R., Zaboli, S., & Conti, M. Efficient Routing Protocol via Ant Colony Optimization (ACO) and Breadth First Search (BFS). In Proceedings of the IEEE International Conference on Cyber, Physical and Social Computing (IEEE CPSCom), in press, Taipei, Taiwan, Vol. 9,pp 4799- 5967,2014.
5. Chandirasekaran, D., & Jayabarathi, T. A Case Study of Bio-Optimization Techniques for Wireless Sensor Network in Node Location Awareness. Indian Journal of Science and Technology, Vol.8,2015
6. Brooks, R. R., Pillai, B., Weigle, M. C., & Pirretti, M. 2007. Optimal layout of multicast groups using network

embedded multicast security in ad hoc sensor networks. International Journal of Distributed Sensor Networks, Vol.3(3), pp273-287,2007.

7. Soe, K. T. Increasing lifetime of target tracking wireless sensor networks. In Proceedings of World Academy of Science, Engineering and TechnologyVol.32, pp 440-445,2008.
8. Chalak, A, R., Misra, S., and obaidat, M. S. " A Cluster Head Selection Algorithm for Wireless Sensor Networks", In IEEE International Conference on Electronics, Circuits and Systems, pp.130-133, 2010.
9. AthanasiosV ,Yanjun Yao, Qing Cao &. Vasilakos, "EDAL. An Energy Efficient, Delay – Aware and Lifetime-Balancing Data Collection Protocol for Heterogeneous Wireless Sensor Networks", in IEEE/ACM Transactions on Networking, Vol. 23, pp 810-823, 2014.
10. Medagliani, P., Ferrari, G., Gay, V., & Leguay, J. Cross-layer design and analysis of WSN-based mobile target detection systems. Ad Hoc Networks, Vol.11, pp712-732,2013.
11. Azad, A. K. M., & Kamruzzaman, J. Energy-balanced transmission policies for wireless sensor networks. IEEE Transactions on Mobile Computing, Vol.10(7), pp 927-940,2011.
12. Cheng, J., Ye, Q., Jiang, H., Wang, D., & Wang, C. STCDG: an efficient data gathering algorithm based on matrix completion for wireless sensor networks. IEEE Transactions on Wireless Communications, Vol.12, pp850- 861, 2013
13. Gupta, I, Riordan, D & Sampalli, S 2005, 'Cluster-head election using fuzzy logic for wireless sensor networks,' in Proc. Annu. Conf. Commun. Netw. Services Res, pp. 255-260.
14. Wang, MW, Qing, L & Zhu, QX 2006, 'Design of a distributed energy-efficient clustering algorithm for heterogeneous wireless sensor networks', Computer Communication (Elsevier), vol. 29, no. 12, pp. 2230-2237.
15. Ming Ma & Yang, Y 2006, 'Clustering and Load Balancing in Hybrid Sensor Network with Mobile Cluster Heads', Proceedings of the 3rd international conference on quality of service in heterogeneous wired / wireless Networks (Qshine).
16. Low, CP, Fang, C, Mee J Ng & Hang, YH 2007, 'Load Balanced Clustering Algorithm for Wireless Sensor Networks', IEEE Communications Society, pp. 3485 - 3490 .
17. Dipak Wajgi & Nileshsingh V Thakur 2012, 'Load Balancing Based Approach to Improve Lifetime of Wireless Sensor Network', International Journal of Wireless & Mobile Networks (IJWMN) vol. 4, no. 4, pp. 155-167.
18. Jau-Yang Chang & Pei-Hao Ju 2012, 'An efficient cluster-based power saving scheme for wireless sensor networks', Springer transaction on Wireless Communications and Networking.
19. Tripathi, RK & Verma, NK 2013, 'Clustering algorithm for nonuniformly distributed nodes in wireless sensor network', Electronics Letters, vol. 49, no. 4, pp. 299-300.
20. Young Sang Yun & Ye Xia 2010, 'Maximizing the lifetime of wireless sensor networks with mobile sink in delay-tolerant applications', IEEE Transactions on Mobile Computing, vol. 9, no. 9, pp. 1308-1318.