IOSR Journal of Electrical and Electronics Engineering (IOSR-JEEE)
e-ISSN: 2278-1676,p-ISSN: 2320-3331, Volume 11, Issue 2 Ver. I (Mar. – Apr. 2016), PP 74-79
www.iosrjournals.org

Implementation of Ant Colony Optimization Adaptive Network-On-Chip Routing Framework Using Network Information Region

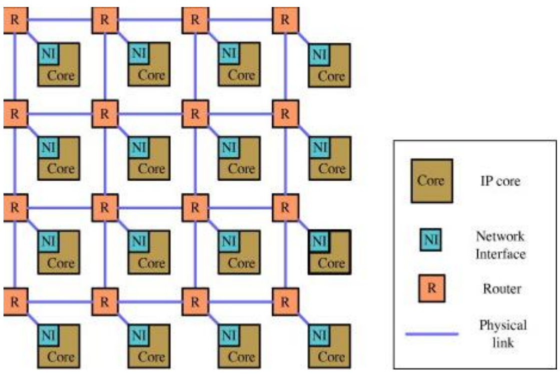
Nandhini . N[1], Mrs.A.Paulene Lourdu Mary[2],
[1]*Pg Student Department Of Eee Gkm College Of Engineering & Technology*
[2]*Assistant Professor Department Of Eee Gkm College Of Engineering & Technology*

Abstract: *Network-on-Chip (NOC) provides regular and scalar design architecture for the system. The efficiency of the routing because of more complex and network scaling. The main work of the proposed implementing Backward Ant Mechanism which provide extra feedback congestion information and also provide network information region framework. This contains the combination of network information with the routing algorithm. By diffusing the pheromone outward, spatial and temporal network information can be exchanged among adjacent routers. The results shows the network performance, area efficiency is improve and congestion is reduced due to updating of routing table is faster.*
Index: *Network-on-Chip, Ant Colony Optimization, Network Performance and Area Efficiency*

I. Introduction

As the number of IP modules in Systems-on-Chip (SoC) increases, bus-based interconnection architectures may prevent these systems to meet the performance required by many applications. For systems with intensive parallel communication requirements buses may not provide the required bandwidth, latency, and power consumption. A solution for such a communication bottleneck is the use of an embedded switching network, called Network-on-Chip (NoC), to interconnect the IP modules in SoCs. NoCs design space is considerably larger when compared to a bus-based solution, as different routing and arbitration strategies can be implemented as well as different organizations of the communication infrastructure. In addition, NoC's have an inherent redundancy that helps tolerate faults and deal with communication bottlenecks. This enables the SoC designer to find suitable solutions for different system characteristics and constraints.

Network on chip (NOC) and System on chip (SOC) both are sub system based integrated circuit that integrates even component of a particular system. But, just as the name suggests NOC is designed for an organized network but SOC is meant for an organized device like computer. This why NOC acts as an integral link between the IP code and the System on chip of the computer, whereas the system on chip connects the computer or any other electronic device in one sole system. So, SOC is more like an embedded system that ties every unit of a particular device together.



When the NOC chip is linked with the SOC of a system then it is going to simplify the hardware applications by reducing the routing functions making the SOC interconnect with the NOC so that they can function promptly at higher operation frequencies. Similarly, a SOC is always embedded in a long sensitive path which need precise placement of such chips so that the IP of the system is not affected because of the integration. NOC is acclaimed to be globally asynchronous locally SOC (GALS) that allows the electronic module to operate in a synchronous manner maintaining a clockwise connection in-between them. Routing a particular data with the SOC requires a lot of wiring; this increases the wiring congestions causing a lot of complications for the system. This is where Network on chips can act as a lifesaver, because when a Network on chip is connected with the SOC then they significantly reduce the wiring connections that are required by the system to function. A system quite often faces semiconductors Intellectual property block, so when you have integrated the circuit with the NOS then you can swap the IP blocks without any trouble. This allows the chips to respond in a timely manner and even makes sure that the embedded system.

II. Related Works

The Networks-on-Chip (NoC) provides regular and scalable design architecture for the chip multiprocessor (CMP) systems. The routing efficiency dominates the overall system performance because of more complex applications and network scaling. The Ant Colony Optimization (ACO) is a distributed collective-intelligence algorithm. The ACO-based selection scheme with Backward-Ant mechanism (ACO-BANT) can provide extra feedback congestion information compared with forward-ant mechanism. However, the storing and computation cost of BANT is too high for the NoC systems. In this work, we implement the ACO-BANT selection scheme with feasible cost on NoC. The simulation results show that the proposed scheme yields improvements in saturation throughput.

Hsien-Kai Hsin, En-Jui Chang, and An-Yeu (Andy) Wu proposed the multi-processor (CMP) systems. The Ant Colony Optimization (ACO) is a distributed algorithm. Applying ACO to selection models of adaptive routing can improve NoC performance. Currently, ACO-based selection only uses the historical traffic information. While additional temporal and spatial information provides better approximation of network status for global load-balancing. In this paper, we first consider the temporal enhancement of congestion information. We propose the Multi-Pheromone ACO-based (MP-ACO) selection scheme which adopts the concept of Exponential Moving Average (EMA) from stock market. We implement a novel ACO system where ants lay two kinds of pheromones with different evaporation rates. The temporal pheromone variation can help to capture hidden-state dependencies of upcoming congestion status. Secondly, to acquire the spatial range of congestion information, we propose Regional-Aware ACO-based (RA-ACO) selection to record historical buffer information from routers within two-hop of distances, which helps to extend spatial pheromone coverage. Information provided by the proposed two schemes improves the system performance.

Chen Wu, Chenchen Deng proposed an efficient application mapping approach is proposed for the co-optimization of reliability, communication energy, and performance (CoREP) in networkon-chip (NoC)-based reconfigurable architectures. A cost model for the CoREP is developed to evaluate the overall cost of a mapping. In this model, communication energy and latency (as a measure of performance) are first considered in energy latency product (ELP), and then ELP is co-optimized with reliability by a weight parameter that defines the optimization priority. Both transient and intermittent errors in NoC are modeled in CoREP. Based on CoREP, a mapping approach, referred to as priority and ratio oriented branch and bound (PRBB), is proposed to derive the best mapping by enumerating all the candidate mappings organized in a search tree. Two techniques, branch node priority recognition and partial cost ratio utilization, are adopted to improve the search efficiency. Experimental results show that the proposed approach achieves significant improvements in reliability, energy, and performance. Compared with the state-of-the-art methods in the same scope, the proposed approach has the following distinctive advantages: 1) CoREP is highly flexible to address various NoC topologies and routing algorithms while others are limited to some specific topologies and/or routing algorithms; 2) general quantitative evaluation for reliability, energy, and performance are made, respectively, before being integrated into unified cost model in general context while other similar models only touch upon two of them; and 3) CoREP-based PRBB attains a competitive processing speed, which is faster than other mapping approaches.

III. System Description

In the natural world, ants (initially) wander randomly and upon finding food return to their colony while laying down pheromone trails. If other ants find such a path, they are likely not to keep travelling at random, but to instead follow the trail, returning and reinforcing it if they eventually find food.

Over time, however, the pheromone trail starts to evaporate, thus reducing its attractive strength. The more time it takes for an ant to travel down the path and back again, the more time the pheromones have to evaporate. A short path, by comparison, gets marched over more frequently, and thus the pheromone density becomes higher on shorter paths than longer ones. Pheromone evaporation also has the advantage of avoiding the convergence to a locally optimal solution. If there were no evaporation at all, the paths chosen by the first ants would tend to be excessively attractive to the following ones. In that case, the exploration of the solution space would be constrained.

Thus, when one ant finds a good (i.e., short) path from the colony to a food source, other ants are more likely to follow that path, and positive feedback eventually leads to all the ants following a single path. The idea of the ant colony algorithm is to mimic this behavior with "simulated ants" walking around the graph representing the problem to solve.

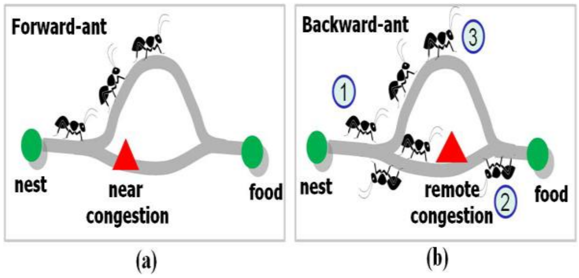


Fig 3: Backward Ant Mechanism

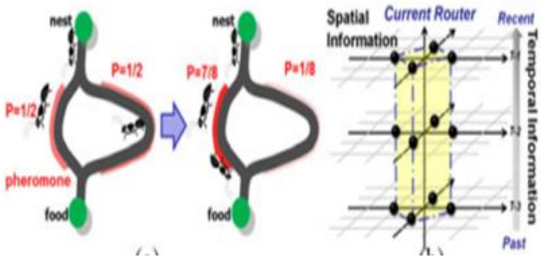


Fig 3.2: Pheromone Diffusion

3.1 Network Information Region and Linking NIR with Existing Selection Functions

In the proposed system, the Backward Ant Mechanism and Network Information Region Framework is combined to improve the network lifetime, throughput and efficiency. The routers under consideration are named information nodes (INs) and are represented with dots in the figures. We identify the router nodes as Nðx; y; tÞ, where x; y denote the location of the node and t denotes the time slot.

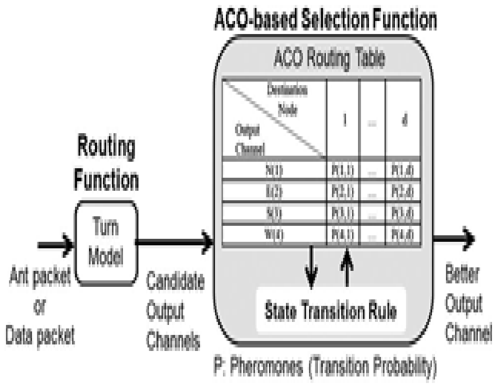


Fig 3.3 : State Transition

3.2 Formulation of Diffusive Pheromone PhD to Link with Spatial Information

The pheromone accumulated in each router represents the observed historical information. By making use of this information and spreading it outward hop by hop, neighbouring routers can acquire not only the present network status but also the status of the past. Therefore, the NIR of ACO-PhD can expand on both temporal domain and spatial domain. Before introducing the formation of PhD, we want to explain the notation of the directions in our NoC system. For each router, the entire NoC is separated into four regions, which are NE, NW, SE, and SW. For example, for packets heading to NE region, selection function may choose North or East channel for transmission. To make a proper choice, network status of the two directions are required. Since there are four regions and two directions for each region, the network status is divided into a total number of eight directions. Note that NE and EN are both required for packets heading to NE region, while NE stands for the network information of North channel and EN stands for that of the East channel. To receive the information from the eight directions, each router has to provide corresponding information for others.

3.3 Design Parameters of ACO-PhD

There are two design parameters in ACO-PhD. One is the ACO weighting a. The other is the pheromone diffusion PhD weighting b. We further describe the physical meaning and the proper value of each weighting.

3.3.1 ACO Weighting

The ACO weighting a determines the composition of PhA .It determines the ratio between the historical information stored in the local router and the incoming information sent from the neighboring routers. A higher a leads to a higher ratio of PhD, indicating that the routing algorithm relies more on the incoming information over the historical information stored. A lower a, on the other hand, indicates more trust on the historical information that the local router observed.

3.3.2 PhD Weighting

PhD is the pheromone value that propagates the network information outward. The ratio between PhACC and PhD is controlled by PhD weighting b. Higher b indicates higher proportion of PhAcc in PhD , which increases the significance of the temporal information. Lower b, on the other hand, indicates higher proportion of PhD in PhD , which increases the significance of the spatial information.

3.3.3 ACO-PhD Weighting Settings

The value of a and b ranges from zero to one. We show the latency of different settings of a and b under saturation throughput. For network performance and hardware friendliness, we choose the optimized value of (a, b) in uniform traffic to be (0:5; 0:375) in our simulations. In addition, we find this setting is also a suitable choice in different kinds of synthetic and real traffic patterns.

IV. Proposed Algorithm

4.1 Network Information Region and Linking NIR with Existing Selection Functions

In the proposed system, the Backward Ant Mechanism and Network Information Region Framework is combined to improve the network lifetime, throughput and efficiency. The routers under consideration are named information nodes (INs) and are represented with dots in the figures. We identify the router nodes as Nðx; y; tÞ, where x; y denote the location of the node and t denotes the time slot.

```
1  ACO-PhD Selection (in : AOC, out: sc) {
2    for ch ∈ AOC {
3      R←ComputeRegion (n_c, n_d, ch)
4      Ph_Acc[R]←(1-α)Ph_Acc[R]+ α Ph_Dif[R]      //Eq. (6)
5    }
6        sr ←R s.t. Ph_Acc[R]= max(Ph_Acc[])
7        sc ←ComputeChannel(sr)
8  }
9  ACO-PhD Diffusion () {
10   for d ∈ Direction {
11       Ph_1[d] ←β Ph_Acc1[d]+ (1-β) Ph_Dif1[d]
12       Ph_2[d] ←β Ph_Acc2[d]+ (1-β) Ph_Dif2[d]
13       Ph'_Dif[d]← 1/2 Ph_1[d]+ 1/2 Ph_2[d]      //Eq. (5)
14   }
15 }
```

The input parameter ACO stands for the set of admissible output channels, which is computed by the routing function RF. The output parameter is the selected channel ACO:

1) Region computation. For each channel in AOC (line 2), we first compute the region for reaching certain columns in the pheromone tables (line 3).
2) Pheromone accumulation. We use (6) to update the accumulation pheromone Ph_{Acc} (line 4).
3) Path selection. We select the route from the highest value of pheromone and compute the corresponding channel from region (lines 6 and 7).

V. Simulation Environment

5.1 Hardware Requirements

System	Intel CORE i5 4200(4th Generation)
Hard Disk	500 GB.
Monitor	15.6 inch 1GB VGA.
RAM	4 GB

5.2 Software Requirements

Operating System	Ubuntu
Tool	Network Simulator Version 2
Front End	TCL (Object Oriented Tool)

VI. Performance Analysis

Simulation Results

The simulation results shows that the Path relocation, Congestion overhead and the Throughput performance of the Ant Colony Optimization system.

Fig 6.1 Path Selection

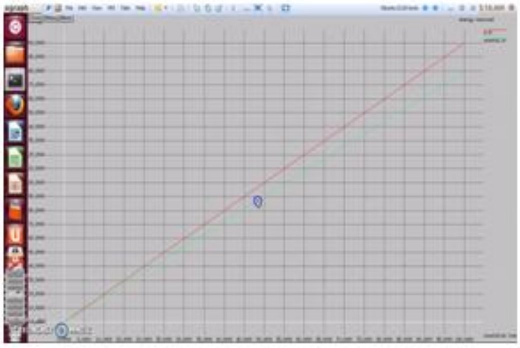


Fig 6.2 Throughput Performance

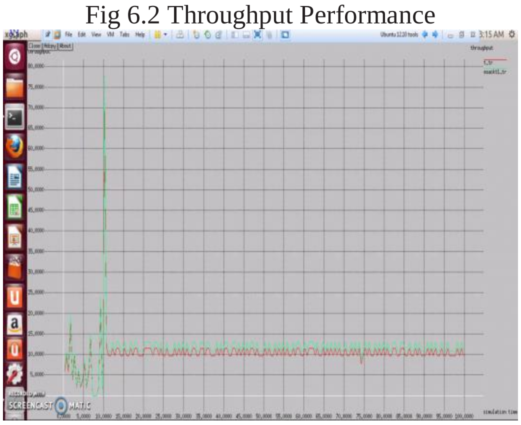
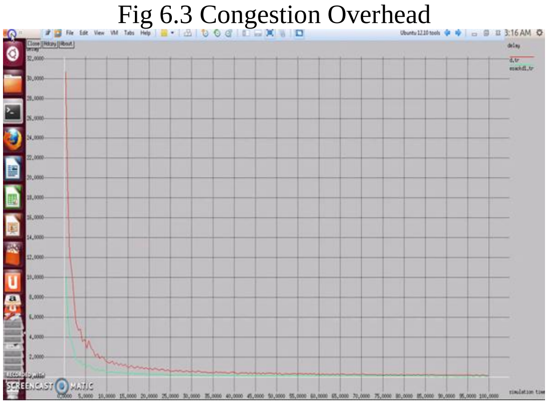


Fig 6.3 Congestion Overhead



VII. Conclusions And Future Work

We bring out the concept of network information region, which indicates the network information utilized by each routing algorithms and also combining the backward ant mechanism. We use the NIR to analyze each related work and propose the ACO-based adaptive routing with pheromone diffusion (ACO-PhD) algorithm. We show that we can reconfigure the ACO-PhD algorithm to each routing algorithm in its NIR subsets by adjusting the parameter settings. This concept can help in developing more advanced reconfigurable schemes for different types of traffic or design constraints. The overall performance is evaluated and compared with other related works, showing that ACO-PhD can achieve the highest performance among all schemes on both saturation throughput and area efficiency. While combining these two mechanisms, there will be occurring some drawbacks like power consumption and processing time will be more. The future work is to reduce these factors and achieve the highest parameters.

References

[1]. L. Benini and G. D. Micheli (2002), "Network on chip: A new SoC paradigm for systems on chip design," Proc. IEEE Des., Autom. Test Conf., pp. 418–419.

[2]. L. Bianchi, L.M. Gambardella, M. Dorigo (2002) An ant colony optimization approach to the probabilistic traveling salesman problem. In Proceedings of PPSN-VII, Seventh International Conference on Parallel Problem Solving from Nature, Lecture Notes in Computer Science. Springer Verlag, Berlin, Germany, pp 883-892.

[3]. G. -M. Chiu, (2000), "The odd-even turn model for adaptive routing," IEEE Trans. Parallel Distrib. Syst., vol. 11, no. 7, pp. 729–738,

[4]. Cloud Computing Intel Labs, (2009.). Single Chip Cloud Computer: Project. [Online]. Available: http://www.intel.com/content/www/us/en/research/intel-labs-single-chip-cloudcomputer.html.

[5]. W. J. Dally and B. Towles(2001), "Route packets, not wires: On-chip interconnection networks," Proc. ACM/IEEE Des. Autom. Conf, pp. 684–689.

[6]. W. J. Dally and B. Towles,(2004) Principles and Practices of Interconnection Networks. San Mateo, CA, USA: Morgan Kaufmann, pp. 159–244.

[7]. C. J. Glass and L. M. Ni,(1994) "The turn model for adaptive routing," J. ACM, vol. 40, no. 1, pp. 874–902, .

[8]. D. Hodges, H. Jackson, and R. Saleh,(2004.) Analysis and Design of Digital Integrated Circuits: In Deep Submicron Technology. New York, NY, USA: McGraw-Hill,

[9]. J. C. Martnez, F. Silla, P. Lopez, and J. Duato,(2000) "On the influence of the selection function on the performance of networks of workstations," in Proc. IEEE High Performance Comput. Conf., , pp. 292–299.

[10]. R. Marculescu, U. Y. Ogras, L.-S. Peh, N. E. Jerger, and Y. Hoskote,(2009) "Outstanding research problems in NoC design: System, microarchitecture," IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst., vol. 28, no. 1, pp. 3–21.

[11]. P. Magarshack and P. Paulin, "System-on-chip beyond the nanometer wall(2003)," Proc. Des. Autom. Conf.., pp. 419–424.

[12]. R. Marculescu, U. Y. Ogras, L.-S. Peh, N. E. Jerger, and Y. Hoskote,(2009.) "Outstanding research problems in NoC design: System, microarchitecture," IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst., vol. 28, no. 1, pp. 3–21.

[13]. U. Y. Ogras, J. Hu, and R. Marculescu,(2005) "Key research problems in NoC design: A holistic perspective," in Proc. IEEE/ACM/IFIP Int. Conf. Hardware/Softw. Codesign Syst. Synthpp. 69–74.

[14]. T. M. Pinkston and S. Warnakulasuriya, (1997)"On deadlocks in interconnection networks," in Proc. Int. Symp. Comput. Archit., , pp. 38–49.

[15]. R. S. Ramanujam and B. Lin,(2010) "Destination-based adaptive routing on 2D mesh networks," in Proc. ACM/IEEE Symp. Comput. Archit.,pp. 1–12.