Evaluating Generalized Semi Markov Process Model of SoC Bus Architectures using HCFG

Ulhas Deshmukh
Govt. Polytechnic College
Dhule, India
Email: deshmukhur@gmail.com

Vineet Sahula
Dept. of ECE
NIT Jaipur, INDIA
Email: sahula@ieee.org

Abstract—This paper presents an efficient approach based on Hierarchical Concurrent Flow Graph (HCFG) for performance evaluation of single shared bus architecture and hierarchical bus bridge architecture. The formulation is based on generalized semi Markov process model of these architectures. In particular, we focus on building model for a single shared bus architecture and extend the approach to model architecture consisting of hierarchical buses connected through bus bridge. Our modeling approach provides early estimation of performance parameters viz. memory bandwidth, processor utilization, average queue length and average waiting time.

We validate the proposed modeling and evaluation approach by comparing the results of evaluation against those that are obtained by SystemC simulation of the same communication architectures under consideration. The HCFG approach is not only time efficient but also provides much detailed evaluation of stochastic properties of performance parameters as compared to SystemC simulation. To illustrate the efficacy of the approach, we compare the results with the results available in the literature for some more examples.

I. INTRODUCTION

A System-on-Chip (SoC) performs two orthogonal aspects of system functionality: computation and communication [1]. System computation task is mapped to Processing Elements (PEs) or Intellectual Property (IP) blocks which are heterogeneous and concurrent, while communication architecture performs communication between these PEs/IPS. Generalized SoC architecture comprising PEs/IPS and communication architecture is depicted in Fig. 1(a). Ever increasing demand for more functions and higher performance of the systems in diverse and emerging applications like telecommunications and multimedia makes SoC design quite complex. Further, stringent time requirement imposed upon them impedes challenges to the designers. In order to achieve higher performance of such systems, designer integrates a number of PEs/IPS in a single SoC. The International Technology Roadmap for Semiconductors (ITRS) [2] emphasizes on maximum reuse of pre-designed and pre-verified hardware/software IPs in order to manage increased complexity and time deadlines of SoCs. Now-a-days fully optimized IPs are available. In SoCs, integration of optimized IPs through suitable communication architecture in short time is desirable. But, it requires comprehensive understanding of functionality and interfaces of each of the IPs. Moreover, no single tool supports seamless modeling and simulation of heterogeneous PEs/IPS along with their communication. Mere integration of pre-verified IPs may not meet communication requirements of entire system if designer underestimates communication architecture exploration. It includes selection of topology, mapping of communication requirement to selected topology and selection of appropriate protocol. Careful design space exploration of communication architecture is necessary not only to meet performance constraints but also to conceive optimum design within design time deadlines. Moreover, plethora of choices available for communication architecture result in larger design space. Thus, quick performance estimation of communication architecture is imperative at all abstraction levels. This has been motivation for our efforts for estimating performance of communication architecture at system level.

In this paper, we propose an efficient approach based on Hierarchical Concurrent Flow Graph (HCFG) for performance evaluation of Single Shared Bus (SSB) architecture and Hierarchical Bus Bridge (HBB) architecture. The formulation is based on generalized semi Markov process (GSMP) model of these communication architectures. In next section, we present brief discussion of GSMP and HCFG, previous work and our contribution. We develop HCFG based performance evaluation framework for SSB architecture in Section III. Section IV gives modeling extension to deal with HBB architecture. We present results in Section V, and conclude in Section VI.
II. BACKGROUND

Communication architecture allows exchange of data and control signals between various system components. In single-chip systems, different types of communication architectures have been used for integrating system components viz. bus-based architectures, Network-on-Chip (NoC) architectures, hybrid bus-NoC architectures, crossbar architectures and multiple-bus architectures. Bus-based architectures can be further classified as dedicated buses, single shared bus and network of shared buses. In SoCs and embedded applications, bus-based architectures are popular because these are simple, consume less power and area. Moreover, performance of bus-based architectures not only suffices for low-end and high-volume applications but also results in a cheaper design. So, we have chosen bus-based architectures for modeling.

A. Generalized Semi Markov Process

Generalized semi Markov process is a stochastic process, which makes transitions from one state to other state in accordance with the Markov chain, and spends a random amount of time in each state before making transition. This time is called sojourn time. Mean value of sojourn time in state \(i\) is denoted by \(\eta_i\).

Another form of concurrency prevalent in NoC architectures is OR concurrency, where communication architecture attempts to transfer data from a source PE to a destination PE through alternate communication paths. Different possible paths depend on routing algorithms used. Communication graph for OR concurrency is shown in Fig 3. Two special nodes, called OR-operation nodes, are included to describe OR-concurrency in the communication flow graph. The nodes are labeled \(\oplus\) and are referred to as “fork” and “join” nodes. The operation nodes are not associated with node weights. We will transform GSMP model of a processing element (PE) connected to the communication architecture under consideration into an HCFG equivalent. Nodes in HCFG correspond to various states of GSMP model of the PE viz. computing, accessing, full waiting or residual waiting. The weight of the nodes corresponds to mean sojourn time of respective state. Directed edges represent communication scheduling information of a PE and influence of other PEs.

C. Related Work and Our Contribution

Many approaches for performance evaluation of on-chip communication architecture have been proposed in literature. Simulation based approach uses communication models at various levels of abstraction. Work in [4] presents simulation by abstracting the system at cycle count accurate at transaction boundaries. Authors in [5] propose formal concurrent modeling approach based on operation state machine for entire system comprising of computation and communication. Analytical approach in [6] estimates communication overhead on data transfer by considering impact of various protocol parameters in the pipelined communication path. S. Dey and S. Bommu [7] report worst case static performance analysis of the system comprising concurrent communicating processes. Their estimate reveals significant underestimate if synchronization overhead is not accounted. Two phase hybrid approach encompasses both simulation and analytical approaches [8] [9] to exploit benefits of both. Work in [9] performs initial analysis tool has been used for evaluation and improvement of process completion time of VLSI design processes [3], where the various stochastic metrics related to process completion time are derived from graph-transmittance. Our work utilizes the same approach for evaluating performance metrics of communication architectures. An SoC communication is hierarchical and concurrent e.g. communication on \(BUS_1\) and \(BUS_2\) of Fig. 6 (Section IV). Hierarchy provides efficient communication between PEs/IPs with varying bandwidth. Besides, two inherent concurrents are present in HBB architecture. Since both communications over \(BUS_1\) and \(BUS_2\) (Fig.6) must be completed before initiating communication across bus bridge, this type of concurrency is called AND concurrency. Two special “operation” nodes are included in the communication graph for AND-concurrency as shown in Fig. 2. These nodes are labeled \(\odot\) and are special operation nodes with no weights associated with them. The operation node with in-degree one is called as “fork” node whereas the operation node with out-degree one is referred as “join” node.
co-simulation with the abstracted communication in the first phase. Time inaccurate communication analysis graph is analytically analyzed in the second phase by specifying communication architecture. Queuing analysis in [8] uses analytical approach to prune the design space and then entire system simulation of the selected architectures.

Main contribution of this paper is system level analytical framework for performance estimation of communication architecture based on GSMP model. The HCFG based evaluation approach is used as an alternative to an approach called Analytical Formulation Of Model Equations (AFOME) reported in [10] [11] for evaluation of GSMP model of SSB and HBB communication architectures. Our modeling framework provides estimation of performance parameters viz. memory bandwidth (BW), processor utilization (PU), average queue length (L), and memory in waiting state labeled as state 1 as modeled as four states. When a PE performs computation, the situation is for memory; while in residual waiting state labeled as state 2 or 3, a PE waits for residual connection time of another PE accessing memory. In each respective state a PE spends average time α1, α2, and α3 called mean sojourn time before making transition.

The process leaves state 0, whenever a PE generates a request and enters either in state 1 or 2 or 3, depending upon status of memory (idle or busy) and arbitration policy employed. A PE transits from state 0 to state 1, when memory is idle and a PE wins arbitration. Transition from state 0 to state 2 takes place, when memory is idle, but a PE does not win the bus arbitration. From state 2, after α2 time, if pending request wins arbitration, the GSMP enters to state 1, otherwise it remains in state 2. The process enters the state 3, if bus is busy, at the time of request. From state 3, after α3 time, process either enters the state 1 if access is granted or it enters the state 2 if fails to get access. From state 1 the process always returns to state 0.

III. HCFG APPROACH FOR EVALUATING GSMP MODEL OF SSB ARCHITECTURE

A. Model formulation for SSB architecture

Synchronous SSB architecture consisting of N processing elements, PE1, PE2,...,PEN competing for the use of a bus is depicted in Fig. 1(b). Arbiter of N-user one-server type resolves the bus access conflict. Characteristic behavior of each PE is assumed to be independent and statistically identical and thus modeling of one PE suffices for performance estimation of the entire system.

![Fig. 4. Generalized semi Markov model of a PE in SSB architecture](image)

The GSMP model of a PE is shown in Fig. 4 and has four states. When a PE performs computation, the situation is modeled as state 0 (computing state). Accessing state labeled as state 1, corresponds to the situation when a PE is accessing memory. In full waiting state labeled as state 2, a PE waits for memory for full connection time of another PE accessing memory; while in residual waiting state labeled as state 3, a PE waits for residual connection time of that another PE accessing memory. In each respective state a PE spends average time η0, η1, η2, and η3 called mean sojourn time before making transition.

In this section, we propose an efficient technique based on HCFG to compute steady state probabilities of being in various states of GSMP model of SSB architecture. These probabilities are used to deduce performance parameters of SSB architecture. Thus, in order to evaluate steady state probabilities of being in various states of GSMP model, we construct HCFG from GSMP model of the SSB architecture as per the transformation technique discussed in [3]. We add one extra node I that represents the initial task, with zero communication time. We draw directed edge (I, x), in HCFG, where x is a state of GSMP with zero in-degree [3]. In present case, we choose node x having minimum in-degree. Hence, residual waiting state (state 3) is taken as x state for GSMP model of SSB. Further, we map average sojourn times ηj’s of the states of GSMP model to the task execution times of corresponding nodes in HCFG. The sojourn time of initial state I is taken as zero. In order to compute steady state probability of being in a particular state, we consider that particular state as a final state. Hence, we change the final state from state 0 through state 3. Thus, we construct one HCFG for every situation while number of PEs vary from one to eight. The structure and sojourn times of all the states in these HCFGs are same, except transition probabilities differ from one HCFG to other HCFG. For illustration, we discuss computation of steady state probability of accessing state 1 when two PEs are mapped to a SSB architecture. In order to compute limiting probability of this state, we treat it as a final state denoted by F and assign Tj = ηj, ∀j ∈ {0,1,2,3} where T0 = 0. The HCFG under this scenario is as depicted in Fig. 5. The HCFG is evaluated to obtain limiting probability of a state. The methodology is repeated for all other states of an HCFG and also for all the states of other HCFGs. Thus, we compute steady state probabilities, P0 through P3 for all the HCFGs, each of which corresponds to a situation having a number of PEs varying from one to eight. We compare efficacy of HCFG approach against AFOME as well as SystemC simulation results.

We compute performance metrics in term of steady state
probabilities as given in equation (1) [10], [11].
\[
BW = N P_1 \\
PU = P_0 + P_1 \\
L = N (P_2 + P_3) \\
\bar{W} = (\eta_2 \alpha_2 + \eta_3 \alpha_3)/\alpha_1
\]

IV. HCFG APPROACH FOR EVALUATING HIERARCHICAL BUS BRIDGE ARCHITECTURE

A. Model formulation for HBB architecture

Hierarchical bus bridge architecture composed of two shared buses BUS_1 and BUS_2 connected by bus bridge is as shown in Fig. 6. Each bus has N processing elements PE_1, PE_2, ..., PE_N competing for the use of shared buses to access shared memories MEM_1 and MEM_2. For brevity, let us consider a scenario when a PE mapped to BUS_1 generates a request to access either MEM_1 or MEM_2. With reference to this PE connected to BUS_1, parameters of MEM_1 and MEM_2 are referred to as local and global parameters, respectively.

In an HBB architecture, each PE can generate two requests. Let X_l be the probability of local request, implying only BUS_1 would be used to access MEM_1 and arbitration of BUS_1 is sufficient. Whereas, let X_g be the probability of global request where both BUS_1 and BUS_2 would be used to access MEM_2 and two stage arbitration of BUS_1 and BUS_2 is essential. GSMP model of a PE in aforementioned scenario is depicted in Fig. 7. Local accessing state labeled as state 1, local full waiting state labeled as state 2 and local residual waiting state labeled as state 3 correspond to MEM_1 and similar to the states of a PE in SSB architecture. State 4, state 5 and state 6 are analogous states when a PE attempts to access MEM_2. These are respectively referred as global accessing state, global full waiting state and global residual waiting state.

State 0 is the computing state.

B. HBB model evaluation using HCFG

As discussed in Section III, we construct HCFG for HBB architecture from GSMP model. Figure 8 shows a HCFG graph for computing steady state probability of local accessing state I for local requesting probability X_l = 0.1. Computing state (state 0) is now taken as x state. We map average sojourn time of the states and transition probabilities of GSMP model to HCFG. We consider one state as a final state F at a time and compute its steady state probability. For this we assign T_j = \eta_j, j = 1, 2, 3, 4, 5, 6, T_I = 0 and execute textual description of HCFG. Thus, steady state probabilities, P_0 - P_6 of all the states of the GSMP model with varying X_l from 0.1 to 0.9 are obtained. Performance metrics for local memory and global memory are computed from steady state probabilities as given in equation (2) [10], [11].

\[
BW_l = N P_1 \\
BW_g = N P_2 \\
L_l = N (P_2 + P_3) \\
L_g = N (P_0 + P_4) \\
\bar{W}_l = \eta_2(\alpha_2 + \alpha_3\beta_1)/\beta_1 + \eta_3\alpha_3 \\
\bar{W}_g = \eta_5(\alpha_5 + \alpha_6\beta_4)/\beta_4 + \eta_6\alpha_6
\]

V. RESULTS

We have employed proposed HCFG based evaluation approach to estimate performance parameters of GSMP based models of SSB and HBB architectures. In this section, we
present performance evaluation results of one example each for SSB and HBB architecture. We validate the proposed evaluation approach by comparing the results of evaluation against those that are obtained by other two approaches- (i) AFOME and (ii) SystemC simulation of the same communication architectures under consideration. The simulation was performed on P-IV, 1 GB Linux-WS.

As first example, a multiprocessor (PEs) system having a SSB architecture is evaluated and various performance parameters are obtained using the approaches discussed in Section III. Various model input parameters are assigned values as follows- \( C = 2 \) cycles, \( T = 0 \) cycles, coefficient of variation of \( C, C_v = 0 \) (where \( C_v = (C^2/(C)^2-1)^{1/2} \)). We varied number of PEs from one to eight. In Fig. 9(a), effect of variation of number of PEs on \( BW \) is shown with proposed approaches as well as simulation approach. Reduction in bandwidth from 1 to 0.82 is observed from the Fig. 9(a) as number of PEs mapped to SSB architecture are increased from one to eight. If single PE is mapped to the bus, we obtain parameter values as: \( BW = 1, \) \( PU = 1, \) \( L = 0 \) and \( W = 0 \). This case refers to dedicated bus and signifies that communication always takes place if sender and receiver are ready without waiting time or contention.

In Fig. 9(b), a plot of variation of processor utilization against number of PEs is shown. As observed from the Fig. 9(b), processor utilization becomes poorer from 1 to 0.073 if number of PEs mapped to SSB architecture are varied from one to eight. This is intuitive as mapping a large number of processors to the bus implies that they spend more time in waiting states 2 and 3 for the bus to become free. Although, degradation is not significant in computational intensive applications. To justify this hypothesis, we incorporate additional input parameter to the GSMP model of SSB architecture, probability of request \( r \) generated by a PE. We take equal number clock cycles for computation and communication i.e. \( C = T = 2 \) cycles, remaining parameters are same as that of previous example. Typical value of bus request rate in multimedia applications is 11.5% [8]. So, we chose \( r = 0.1, \) signifies that 10% of the total execution time is used for communication. Processor utilization of this experiment is also shown in Fig. 9(b). It indicates degradation in processor utilization is from 1 to 0.87 i.e. only 13% as compared with 93% of previous case.

In addition, we consider two example systems comprising SSB architecture [9]. We compare the results obtained by proposed approach with those obtained by using Communication Analysis Graph (CAG) based approach [9]. We chose similar parameters as used by authors in [9]. For both systems, bus protocol parameters were arbitrary chosen. First system is TCP/IP network interface card subsystem with three PEs (components) mapped to the SSB. Authors in [9] state that communication payload ofTCP/IP system was 512 bytes/packets which was simulated by them for 100 packets before CAG could be derived. We have performed estimations choosing similar bus parameters as in [9]- MAX_DMA_SIZE being 16 bus words, PROTOCOL_OVERHEAD being 1 cycle, \( BUS\_WIDTH \) being 8 bytes and speed being 166 MHz. Second example system (SYS1) is composed of two PEs and the borrowed parameters were- average bus access at a time \( (\bar{C}) = 10 \) words, computation time \( (T) = 10 \) cycles, DMA block size= 5 and execution period = 2000 memory accesses from each component. CAG based approach does not report bandwidth directly. We have simulated functionality of these systems in “SystemC” to obtain communication and computation time from which we compute bandwidth. Table I shows comparison of results.

<table>
<thead>
<tr>
<th>System</th>
<th>CAG appro. [9]</th>
<th>HCFG appro.</th>
<th>% Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>TCP/IP</td>
<td>0.83</td>
<td>0.74</td>
<td>13.00</td>
</tr>
<tr>
<td>SYS1</td>
<td>0.88</td>
<td>0.84</td>
<td>4.5</td>
</tr>
</tbody>
</table>

In second example, we consider evaluation of performance metrics of a PE mapped to the HBB architecture by varying probabilities of local request \( X_L \) and global request \( X_g \) using proposed approaches. Input parameters chosen are \( C'_g = 2 \) cycles, \( C'_L = 2 \) cycles and \( N = 2 \). Remaining parameters are same as that of previous example.

Results showing variation of \( BW_g \) with \( X_L = 0.1 \) to 0.9 is reported in Fig. 9(c). It shows, \( BW_g \) increases with \( X_L \), with small boost in the beginning because, probability as well as priority of global request are higher than those of local request. Rate of increase is more beyond \( X_L = 0.4 \). In Fig. 10(b), variation of \( BW_g \) with \( X_L = 0.1 \) to 0.9 are illustrated. It also indicates increase in global bandwidth with global requesting probability. But, maximum local bandwidth is 0.97 while maximum global bandwidth is 0.31 as observed from figures 9(c) and 10(b). This is because two stage arbitration for \( BUS_1 \) and \( BUS_2 \) is essential for accessing global memory, so a PE requesting for global memory has to wait in local as well as global waiting states and wins arbitration of both buses when both would become free. This infers that PE having higher communicating payload with memory should be mapped to the same bus as that of memory to achieve better performance similar to the explanation given in [9]. Thus designer could find optimum mapping early in design cycle.

Fig. 10(a) indicates that local waiting time increases upto \( X_L = 0.4 \) as global request are dominating and then decline rapidly. Similar observation can be noted for global waiting time with global requesting probability \( X_g \). But, global request has to wait more time as compared to local request even if requesting probabilities are same. For example local wait time is 2.01 units for \( X_L=0.9 \), while global wait time is 10.85 units for \( X_g=0.9 \). Fig. 10(c) indicates increase in global queue length with \( X_g \).

Effectiveness of HCFG approach in terms of execution time and average error is compared with simulation in the Table II.

**VI. CONCLUSIONS**

This paper presents HCFG based analytical approach for evaluation of GSMP model of SSB and HBB architectures. We have evaluated performance metric viz. bandwidth, processor
utilization, queue length and waiting time with number of processing elements for SSB architecture. For HBB architecture performance parameters for local and global memories are evaluated with local or global requesting probabilities. Results obtained with HCFG evaluation approach are validated using SystemC simulation of the same communication architectures under consideration. Proposed approach took much less time as compared with simulation as well as equally accurate as that of simulation. HCFG technique provides not only faster evaluation but also gives probability distribution of performance metrics.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>BW</th>
<th>PU</th>
<th>L</th>
<th>W</th>
</tr>
</thead>
<tbody>
<tr>
<td>% Average error (Abs.)</td>
<td>4.08</td>
<td>2.73</td>
<td>21.88</td>
<td>12.78</td>
</tr>
<tr>
<td>Execution Time (second)</td>
<td>HCFG</td>
<td>Simulation</td>
<td>0.0065</td>
<td>884.16</td>
</tr>
</tbody>
</table>

Fig. 9. Variation of (a) BW and (b) PU with N for SSB architecture, and (c) effect of X_t on BW_t for HBB architecture

Fig. 10. Variation of (a) W_f, (b) BW_g and (c) L_g for HBB architecture

**REFERENCES**


