Reducing Null Messages Using Grouping and Status Retrieval for a Conservative Discrete-Event Simulation System

Bevin Thomas, Syed S. Rizvi, Khaled M. Elleithy
Department of Computer Science, University of Bridgeport, Bridgeport, CT 06604, USA
{srizvi, bthomas, elleithy}@bridgeport.edu
Tel: 92 (111) 002-004
Fax: 92 (021) 509-0968

Keywords: Discrete event simulation, conservative algorithms, null message algorithm, parallel systems.

Abstract
In this paper we investigate Chandy-Misra-Bryant Null message algorithm and propose a grouping technique to improve the performance. This technique along with status retrieval which will be explained in detail can improve the performance when compared to the traditional conservative algorithm by Chandy-Misra-Bryant. Null message algorithm is an efficient conservative algorithm that uses null messages to provide synchronization between logical processes in a parallel discrete event simulation (PDES) system. The performance can be decreased if a large number of null messages are generated by LPs to avoid deadlock. The main objective of this research work is to propose a new grouping technique that can be used to reduce the Null messages between the logical processes. Since the performance of Null Message algorithm mainly depends on the Lookahead (L) values, our proposed technique can be used to determine an optimum value of the Lookahead.

1. INTRODUCTION
Parallel and distributed simulation refers to technologies that help a simulation program to execute on multiple processors, interconnected networks. Parallel simulations execute on multiple processors or multiple computers confined to single machine room while distributed simulations execute on computers that are distributed geographically. A Time management is required to ensure the execution of the distributed simulation is properly synchronized. Time management ensures that events are processed in a correct order. Time management algorithms assume that logical processes (LPs) communicate by exchanging time stamped messages or events. The criterion is to make sure that LP process events are processed in timestamp order.

There are two different kinds of parallel simulation: - optimistic and conservative. Optimistic simulation allows processors to independently simulate events assuming they are temporally correct. When it is discovered that there is a temporal discrepancy, the simulation is “rolled back” to the time of the discrepancy and then proceeds again. Conservative simulation never allows discrepancies – event processing is only allowed when it can be guaranteed that the event will not be altered. The principal task of conservative simulation is to determine when it is “safe” to process an event. An event is said to be “safe” if the event containing the time stamp is less than the Lower Bound on the Time Stamp (LBTS).

The algorithms developed by Chandy, Misra and Bryant were the first synchronizations algorithms that were developed. Each LP sends messages with non-decreasing timestamps and it’s received in the same order it was received. Each process will execute an event with non-decreasing timestamps and it’s received in the same order it was received. The Lookahead (L) values, our proposed technique can be used to determine an optimum value of the Lookahead.

1.1 Problem Identification
Null message algorithm (NMA) has indeed resolved the problem of deadlocks which by sending null messages between neighboring LPs. The drawback of NMA is that the performance can degrade drastically if a lot of null messages are sent across the network and between LPs. It depends upon the Lookahead value (L). If the Lookahead value is very small then a lot of null messages are sent across. The main objective is to calculate an optimum Lookahead value so that the null
messages can reduce and therefore improve the performance.

2. RELATED WORK

There are a few researches done in NMA in terms of reducing the no. of null messages and calculating the optimum Lookahead value. Ronald C. De Vries in [4] reduced the number of NULL messages through of prediction of channel lines. A framework is presented on which the distributed discrete event simulation can be built for applications which can be decomposed into feed-forward and feedback networks.

Another notable work done mentioned in [2] was the research done by Syed S. Rizvi, K. M. Elleithy, and Aasia Riasat in which they proposed a mathematical model which can be used to approximate the optimal values of some critical parameters such as frequency of transmission, Lookahead (L) values, and the variance of null message elimination.

According to B. R. Preiss, W. M. Loucks, J. D. MacIntyre, J. A. Field referred in [3], a null message cancellation can improve performance by a great factor. Null Message cancellation is an algorithmic modification to the basic conservative synchronization scheme wherein a null message is discarded before receipt when overcome by a message with a larger timestamp.

3. METHOD OF GROUPING AND STATUS RETRIEVAL

In this paper, the approach to reduce the number of null messages is to group logical processes where each group may consist of $n$ number of LPs. The number of LPs for each group depends upon their similarities or with an optimum value. The optimum number of LPs should be calculated. Each group is controlled by a controller. The role of the controller is to synchronize LPs within a group as well as send synchronization messages across groups to different other controllers. Synchronization messages are sent after fixed interval of time. Whenever a controller receives a message from the neighboring linked controller, it broadcasts the message within the group.

The controller has to be directly connected to the LPs. All LPs are connected to other neighboring LPs using mesh topology. However, an LP can send synchronization messages only to controller. The synchronization message from an LP to its corresponding controller indicates that the sending LP has finished with its assigned tasks. Upon receiving the synchronization message from the one of the LPs, controller broadcast it inside the group. Controllers send synchronization messages after a fixed interval of time.

Fig. 1 represents the implementation of the proposed algorithm.
algorithm. The upper dotted-box of Fig.1 represents the behavior of a controller when it receives a message from a grouped-LP. On the other hand, the lower dotted-box of Fig. 1 represents that how controller deals with the message originating from an ungrouped-LP.

In Fig. 2, there are 9 LPs and are grouped into 3 where each group consist of 3 LPs in this figure. Each group also consists of a controller which is directly connected to each LP.

3.1 Proposed Algorithm

1. While Loop(Simulation is not over)
2. Controller receives from LP inside the group
3. Record the LP and the time stamp
4. Broadcast message to other LPs
5. If Message is received from controller of other group
   6. Broadcast message to the LPs within the group
   7. Send null messages to neighboring controllers
   8. With smallest timestamp indicating a lower bound on future messages sent from that group
   9. Approximately as: (T + L)
10. End if
11. END-LOOP

Fig. 3 shows the implementation of the proposed algorithm with eight LPs. if all LPs were connected interconnected then the total null messages sent for a single deadlock would be totally 32. If the Lookahead is a small value, then the total number of null messages transmitted between the LPs will be approximated as follows:

---

Fig. 2. Topological Map of 3 Groups with nine LPs and three Master Controllers. Within the group, LP are connected via Mesh Topology. LPs can communicate with each other in a group but they can only send synch-message to their respective controller.
Total Null Messages = (32 * n) where \( n \) is iteration number.

It should be noted in Fig. 3 that the total number of null messages are reduced with respect to the implementation of the proposed algorithm. Node A sends a sync message to controller A. On receiving the controller sends a message inside the group except the link it received and also keeps a track of the latest timestamp of each LP. Thus in our example it sends 9 messages within the group. Total number of messages exchanged across all the LP’s are

Total number of null messages = 9 X 2 (inside the group) + 2 (Between controllers) + 3 X 2 (broadcast inside group) = 26.

The main thing to note here is that outside the group only 2 messages were sent. This brief analysis emphasizes the significance of the proposed methodology for efficiently grouping the LPs with their respective controllers and connecting them using the mesh topology.

4. CONCLUSION

In this paper, we presented a new technique of grouping the LPs with the status retrieval method. In order to support the proposed technique, we presented an algorithm for the modified NMA. In addition to the proposed technique and the algorithm, we also provided a discussion on the implementation of the proposed technique with the help of the schematic. We believe that the proposed method of grouping LPs can reduce the null messages transmission to a reasonable extent. Though this technique is expensive since a controller has to be setup, it serves the purpose of reducing the null messages.

REFERENCES