CHAPTER 7

CONCLUSION AND FUTURE WORK

7.1 CONCLUSIONS

A distributed system is much larger and more powerful than typical centralized systems due to the combined capabilities of distributed components. Examples of distributed systems include computer networks, distributed databases, distributed information processing systems and real time process control systems. Many challenging areas that span the middleware and language constructs, down to the implementation of the supporting application-level communication protocols and operating system mechanisms have drawn the attention of researchers in the area of distributed systems. The control tasks of operating systems and database systems like mutual exclusion, deadlock and concurrency control, are much more difficult to solve in distributed systems than in a centralized systems. This thesis addresses various issues associated with one of the fundamental resource management problems, the deadlock.

In distributed systems, the processes have used the resources located at multiple sites to improve the throughput. Due to the absence of centralized resource management, a process might wait for the resources that are held by some other processes. Hence, distributed systems are vulnerable to the occurrence of deadlock. A Deadlock is a canonical, yet an overlooked problem in distributed systems. DD&R is the cheapest method to handle distributed deadlocks. However, deadlock detection is
cumbersome in distributed systems since no site has complete knowledge about the resource requirements of all processes. The distributed deadlock detection algorithms are classified based on the underlying deadlock models such as AND model, OR model, AND-OR model and P out-of Q model. Among the models, P out-of Q model, also called as generalized request model, has the modeling power of all other models and much more concise expressive power than other models. However, it is difficult to detect deadlock in the generalized request model. It is observed that only very few algorithms have been proposed to detect deadlock in the literature. Though the algorithms have reduced the deadlock duration through the years, they have paid very little attention on other key measures such as number of messages and message size.

This thesis proposes three different algorithms for detecting deadlocks in generalized request model. The proposed algorithms have used fewer and short messages to detect generalized deadlocks in contrast to the existing algorithms. Furthermore, the proposed algorithms have simplified the deadlock resolution by minimizing additional message overhead as in the distributed algorithms and additional computational overhead incurred in the centralized algorithms.

This thesis proposes a new distributed algorithm, called FREDDA, which detects a generalized deadlock using less than $2e$ messages in $2d$ time units; where $e$ is the number of edges, $n$ is the number of nodes and $d$ is the diameter in the WFG. Furthermore, it uses messages with the length of $O(1)$ (i.e. fixed sized) and identifies a victim without any additional round of message transmissions.

This thesis presents a new centralized algorithm, called MODDDA, which detects a deadlock by using less than $2e$ messages in $d+2$ time units in the worst case. Since the unblocking conditions of several processes are not
cascaded and carried by the deadlock detection messages, it reduces the message size into $O(n)$ as compared to the existing algorithms. Also, when the system is heavily loaded, it considerably reduces the number of messages that are required to determine the deadlock.

Finally, this thesis proposes another centralized algorithm, called MSODDA, which determines a deadlock by constructing LWFG incrementally at the initiator. It detects a generalized deadlock by using less than $2e$ messages in $d+2$ time units in the worst case. The notable improvement of MSODDA over the existing algorithm is that it uses fixed-sized messages to attain the performance of the best centralized algorithms. Moreover, it does not require additional data structures and more messages to resolve deadlocks as compared to the existing algorithms.

The performance of all proposed algorithms in this thesis is assessed using the discrete event simulator. The numerical results demonstrate that the proposed algorithms have considerably reduced the computational and communicational overhead as with the earlier algorithms. Hence, all proposed algorithms can be used to detect deadlocks in different domains of distributed systems such as resource management in distributed operating systems, store and forward communication networks, communicating processes and replicated databases.

7.2 FUTURE RESEARCH DIRECTIONS

It would be worthwhile to compare the performance of the generalized deadlock detection algorithms with the existing algorithms for detecting deadlocks in other resource request models such as AND model, OR model and AND-OR model. Perhaps all the existing generalized deadlock detection algorithms are based on the assumption that the underlying distributed system is fault-free. However, the real distributed system is subject
to several types of failures including Byzantine failure. Hence, further investigation is required to devise new fault tolerant generalized deadlock detection algorithms. This thesis presents theoretical and simulation results concerning the efficiency of proposed algorithms. To affirm the simulation results, the performance of all proposed algorithms in this thesis would be assessed through an analytic model and compared with the simulation measurements. Finally, the proposed algorithms can be incorporated into the real distributed environment.