Abstract

Performance estimation of a distributed software is a challenging problem. A distributed software runs on multiple processing nodes interconnected in some fashion. In such a situation computational load of a software is distributed onto the processing nodes of the given system. Such a system makes use of an appropriate task scheduling algorithm for obtaining a good performance. The program used in this work emulates a distributed system. An emulator gives the result like an actual system. The emulator is of a fully connected distributed system in which any two processors can directly communicate. The objective of this experiment is to identify the task scheduling algorithm that also performs well in the presence of communication fault delay occurred because of network failure or computation fault delay occurred because of no response from processors in a distributed system.

References

Task Scheduling of a Distributed Computing Software in the Presence of Faults

- Cyril Briquet, Reproducible testing of Distributed software with middleware virtualization and simulation, ACM (2008).
- Kequin Li, Scheduling parallel tasks on multiprocessor computers with efficient power management, IEEE transactions (2010) 978-1-4244-6534, New York, USA.
- V. S. Tondre, V. M. Thakare, S. S. Sherekar, R. V. Dharaskar, Technical computation and communication delay in distributed system, NCICT (2011) IJCA.

Index Terms

Computer Science

Distributed Systems
Keywords
Clustering  distributed computing  homogeneous systems  scheduling  task allocation