Chapter 6

Conclusion and Future Directions

6.1 Conclusion

Despite the popularity of high-performance computing-based transaction processing systems, not everything will result positively. The real-time, heterogeneous, complex and dynamic nature of a multi-core environment makes it challenging to provide unforeseen failures, load balancing, unexpected operating conditions, Resource allocation and memory congestions. Therefore, dependability with a need for memory congestion reduction, long-term resource dependency prediction and load balancing and scheduling becomes important for multi-core cluster computing systems.

Memory congestion, resource allocation, and scheduling in multi-core computing environments are NP-hard problems. To solve such issues, a single method with a fits-for-all solution is impossible, so meta-heuristic approaches are the best options. So, we proposed an Expectation-Maximization (EM) based clustering algorithm to form a suitable multi-core cluster. We chose the concept and scenario of cluster computing as a case study for applying this algorithm. We have also considered different benchmarks provided by NPB, SPEC and SKUs. The algorithm improves the shared memory utilization, data partitioning, and local computation to improve the utilization of parallel processing capabilities in commonly available desktop devices, specifically on multi-core systems. It maximized the node utilization, minimized the make-span, increased the throughput, minimized the miss ratio, and maximized the multi-core speedup. The next step would be to model and analyze the availability, reliability, and dependability of the multi-core cluster by Heterogeneous algorithmic approaches and scheduling algorithms.

The system design faces scalability challenges, particularly in weak scaling scenarios, where memory congestion becomes critical during data transaction processing. The data transaction happens simultaneously between two or more nodes through memory controllers and interconnecting links. The growth of such transactions in multi-core cluster computing environments has become tremendous. So, there is a need to maximize resource availability through the memory congestion reduction technique. In multi-core clusters, the challenge of memory congestion poses a significant obstacle to achieving optimal performance and scalability. Memory congestion arises due to the simultaneous and potentially conflicting demands for data access from multiple cores sharing a common memory subsystem. This issue becomes particularly pronounced in parallel processing scenarios, where efficient data sharing and communication are essential. The irregular data distribution among multi-core systems requires additional parallelization, which leads to varying communication directions and data transfer rates. Thus, we formulated the problem with multiple objectives: maximizing availability and minimizing the make-span. We used two algorithms: first, for gathering the Node Information and Communication pattern (NIC), and second, for Reducing congestion with load balancing (RCLB). We have tasted our proposed algorithm over CG, FT, LU and MG kernels. The experimental results showed that our proposed algorithm performed better than other selected algorithms (Treematch, Packed, and Socket-span).

In a cluster computing environment, the dynamic nature of workloads and resource utilization patterns pose a substantial challenge in anticipating long-term dependencies among resources. The lack of a comprehensive and accurate predictive framework hampers the efficient allocation and management of resources in clusters, leading to suboptimal performance and resource wastage. So, scheduling the workflow and predicting the sequence of execution of tasks is required so that the necessary long-term resources can be easily identified. We proposed a feature-based compatibility prediction algorithm (FBCA) and an accuracy and relative runtime error prediction algorithm (ARRE) to maximize performance. We formulated the resource availability and performability by considering load and make-span. The result showed that the algorithm increased performability and reduced the miss ratio. We simulated the algorithm on two real-time multi-core and virtual cluster scenarios. We compared the proposed algorithm with four modified algorithms. The result showed that resource availability and reliability were maximized.

Our research contributes significantly to the field by presenting novel approaches to resource prediction and load balancing. The FBCA and ARRE algorithms are distinguished by their advanced predictive capabilities, addressing limitations found in recent models such as the Regressive Ensemble Approach for Predicting Resources (REAP) and Bayesian Networks (BN). While REAP and BN offer competitive results, they lack the dynamic adaptability and accuracy of our methods. The FBCA improves prediction accuracy by leveraging feature-based approaches, and ARRE refines ensemble model selection, resulting in superior performance metrics like lower Root Mean Square Error (RMSE) and higher F1-scores.

Compared to recent advancements such as the cost-aware auto-scaling approach and fuzzy neural networks, our research introduces a more adaptive and precise framework. By integrating System of Systems (SoS) and Phold-based approaches, our methods provide a novel perspective on resource prediction and load balancing. This innovative approach addresses the limitations of contemporary models, offering promising directions for future research.

6.2 Future Research Directions

The expansion and evolution of business transactions, data volumes, and the emergence of new applications underscore the ongoing challenges in high-performance computing (HPC). These challenges present diverse avenues for future research, ranging from incremental extensions of current methodologies to more revolutionary approaches. As demonstrated by this study, there are several limitations that need to be addressed in future research: The models and algorithms in this thesis are built upon specific assumptions regarding job and resource behaviors. Real-world deviations from these assumptions could potentially affect the accuracy and robustness of predictions. Implementing the proposed dependency models and predictions in practical HPC systems poses significant challenges. Exploring these challenges further is crucial for refining and validating the algorithms in real-world environments. The validation of the proposed algorithms is limited to specific workflow scheduling scenarios presented in this thesis. Other scheduling algorithms and scenarios remain unexplored, highlighting opportunities for broader benchmarking and comparison.

Future research directions should address the following areas to advance and substantiate this research: Developing automated techniques to optimize the hyperparameters of algorithms is essential. Leveraging machine learning or optimization methods can enhance algorithm performance across diverse HPC environments. Despite utilizing a large-scale supercomputer (Param Shivay with 1.20 Lacs processors and 833 TF speed), not all computing nodes were fully utilized in this study. Future research should conduct comprehensive scalability studies to evaluate algorithm performance as system sizes and workflow complexities increase. Expanding and refining the resource prediction framework by incorporating additional factors such as network variability is also crucial. Developing strategies to improve the accuracy of resource predictions will enhance the practical applicability of the algorithms.

These avenues for future research aim to address current limitations and pave the way for more robust and applicable solutions in the field of high-performance computing.