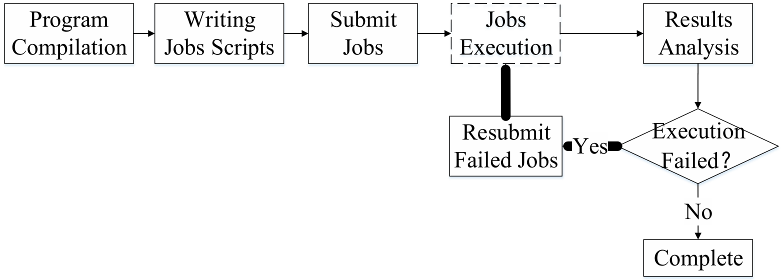
User-level failure detection and auto-recovery of parallel programs in HPC systems

Guozhen ZHANG, Yi LIU, Hailong YANG, Jun XU, Depei QIAN

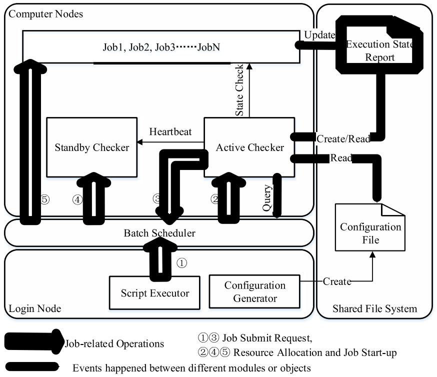
Frontiers of Computer Science, DOI: 10.1007/s11704-020-0190-y

Problems & Ideas

- Problems of the traditional failure detection and execution recovery mechanisms
  - Currently, automated tools for supporting user-level failure detection and auto-recovery of parallel programs in HPC systems are missing. Traditional strategies of failure detection and recovery often require privileges for environment installation or redeployment, the two parts are separate.
- Ideas: A dedicated server is responsible for failure detection and failed jobs resubmission without modification on JMS.
  - The handler is essentially a normal job .
  - Workflow is executed automatically without user involvement

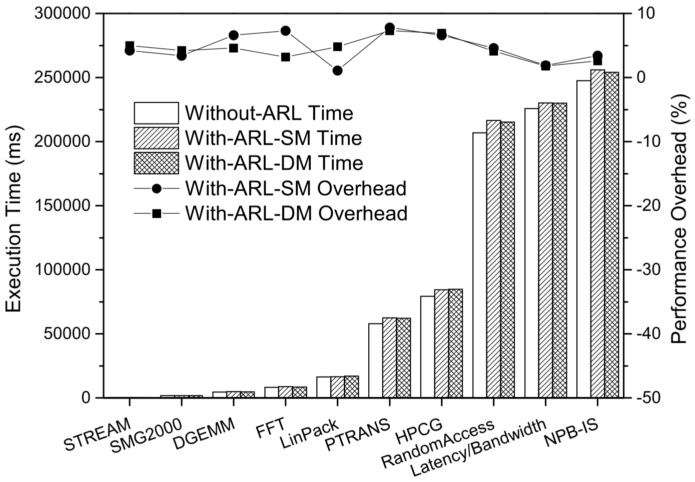


The procedure of user-level failure detection and auto-recovery of parallel programs in HPC systems.

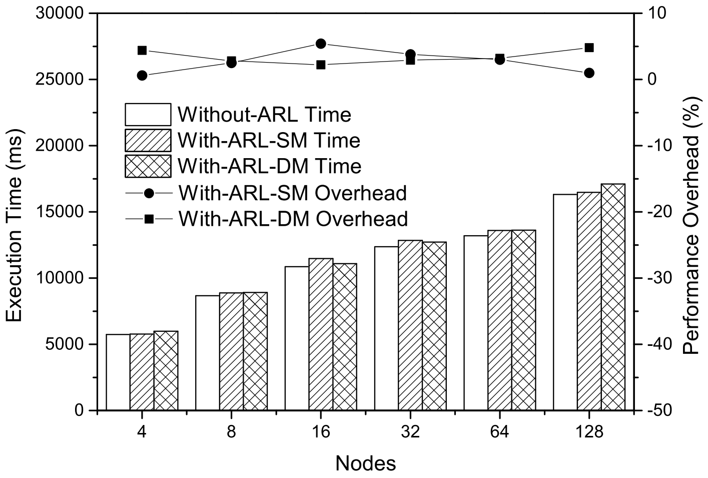


The implementation overview of ARL.

Main Contributions



Comparison of execution time and performance overhead before and after applying ARL on different benchmarks.



Comparison of execution time before and after applying ARL on Linpack with different node scales.

- The execution time with ARL applied does not increases significantly compared to the raw execution. The performance overhead caused by ARL on different benchmarks is negligible.
- The good scalability of ARL indicates that it remains efficient when applied in large-scale HPC systems..