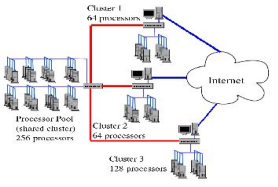
INTELLIGENT PARALLEL JOB SCHEDULERS FOR COMPUTATIONAL MULTI-CLUSTERS

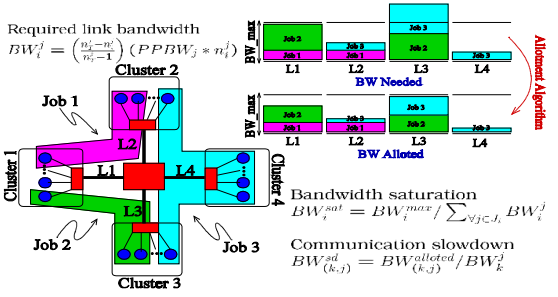
William M. Jones, Louis W. Pang, Nishant Shrivastava, Walter B. Ligon III
ECE DEPARTMENT
CLEMSON UNIVERSITY

Multi-cluster Paradigm

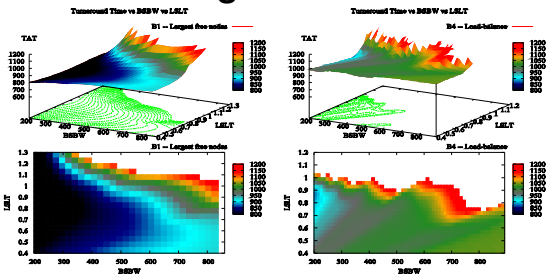
- Computational resource
- Geographic co-location
- Dedicated network
- Multiple participants
- Shared resources
- Distributed processing
- Job scheduling



Co-allocation Model

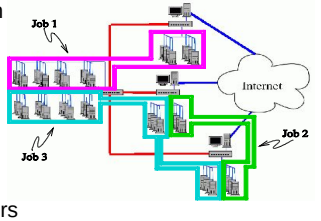


Scheduling Effectiveness



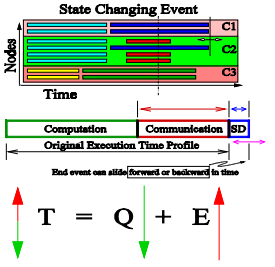
Parallel Job Scheduling

- Resource allocation
  - Where to run job?
  - When to run job?
- Local execution
- Job migration
  - Move entire job
- Job co-allocation
  - Map across clusters
  - Share resources
  - Network contention



Execution-time Dynamics

- State changing events
- New co-allocated job
- Co-allocated job terminates
- Variation w.r.t. time



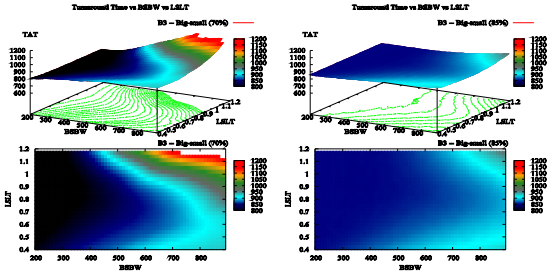
$$T_P^R = \overbrace{T_C^R + T_P^R}^{\text{Residual Execution Time}}$$

$$T_C^R = \overbrace{(T_C^R - T_C^\Delta)(BW_{(k,j)}^{sd})(BW_{(k,j)}^{sd})^{-1}}^{\text{Residual Communication Time}}$$

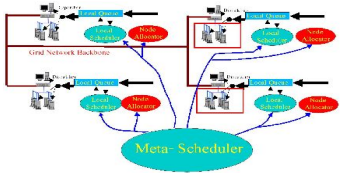
$$T_P^R = \overbrace{(T_P^R - T_P^\Delta)}^{\text{Residual Computation Time}}$$

$$T_P^\Delta = \frac{\Delta_L}{T_P^R} T_P^R \quad \text{and} \quad T_C^\Delta = \frac{\Delta_L}{T_C^R} T_C^R$$

Improved Performance

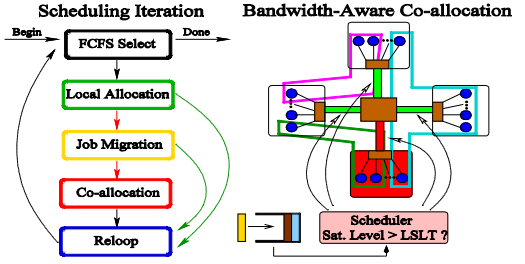


Intelligent Scheduling



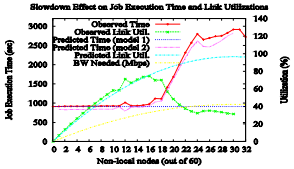
- Global optimizations that improve job response time
- Efficiently manages both node and network resources
- Controls network contention during job co-allocation

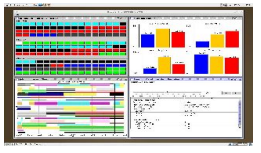
Scheduler Model



Future Work

- Verify existing models and update if necessary
- Add additional parallel job types and topologies
- Enhance workload generation with trace-files
- Integrate configuration GUI into visualizer





Acknowledgements: *This work was supported by the ERC program of the National Science Foundation under Award Number EEC-9731680.*

C A E F F