Modeling Coordinated Checkpointing for Large-Scale Supercomputers

Long Wang, Karthik Pattabiraman,
Zbigniew Kalbarczyk, Ravishankar K. Iyer
Center for Reliable and High-Performance Computing
Coordinated Science Laboratory
University of Illinois at Urbana-Champaign

Lawrence Votta,
Christopher Vick, Alan Wood
Sun Microsystems

Motivation

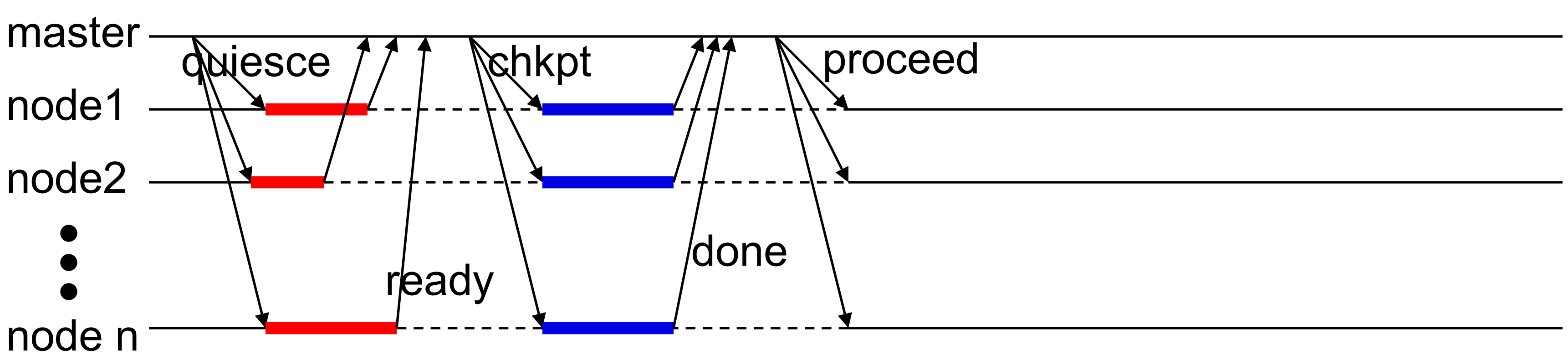
- New generation of supercomputers emerge to meet computational demands of high-performance scientific applications
  - IBM BlueGene/L scales up to 64K dual-processor nodes
- Large number of nodes makes system more vulnerable to errors
- Synchronous checkpointing (and rollback) widely used in supercomputers to recover from failures
  - How does checkpointing scale to several hundred thousand processors?
  - Some usual assumptions no longer hold for large-scale supercomputers
    - Computation interval and checkpoint overhead much smaller than MTBF
    - Failure independence
    - Negligible overhead of checkpointing coordination

Contribution

- Model (SAN) of a coordinated checkpointing for a large-scale (hundreds of thousands of nodes) supercomputer
- Study system scalability, reliability, and performance
  - Analyze impacts of: (i) transient failures during computation and checkpointing/recovery, (ii) correlated failures, (iii) coordination overhead
- Major findings
  - There exist an optimum number of processors for which useful work is maximized
    - e.g., 128K processors (for MTTF per node of 1 year and MTTR of 10 minutes)
  - Useful work fraction is relatively low due to the effect of failures
    - e.g., over 50% of the time is spent on handling failures (128K processors and MTTF per node of 1 year)
  - Correlated failures degrade the performance and limit system scalability

Target System (1)

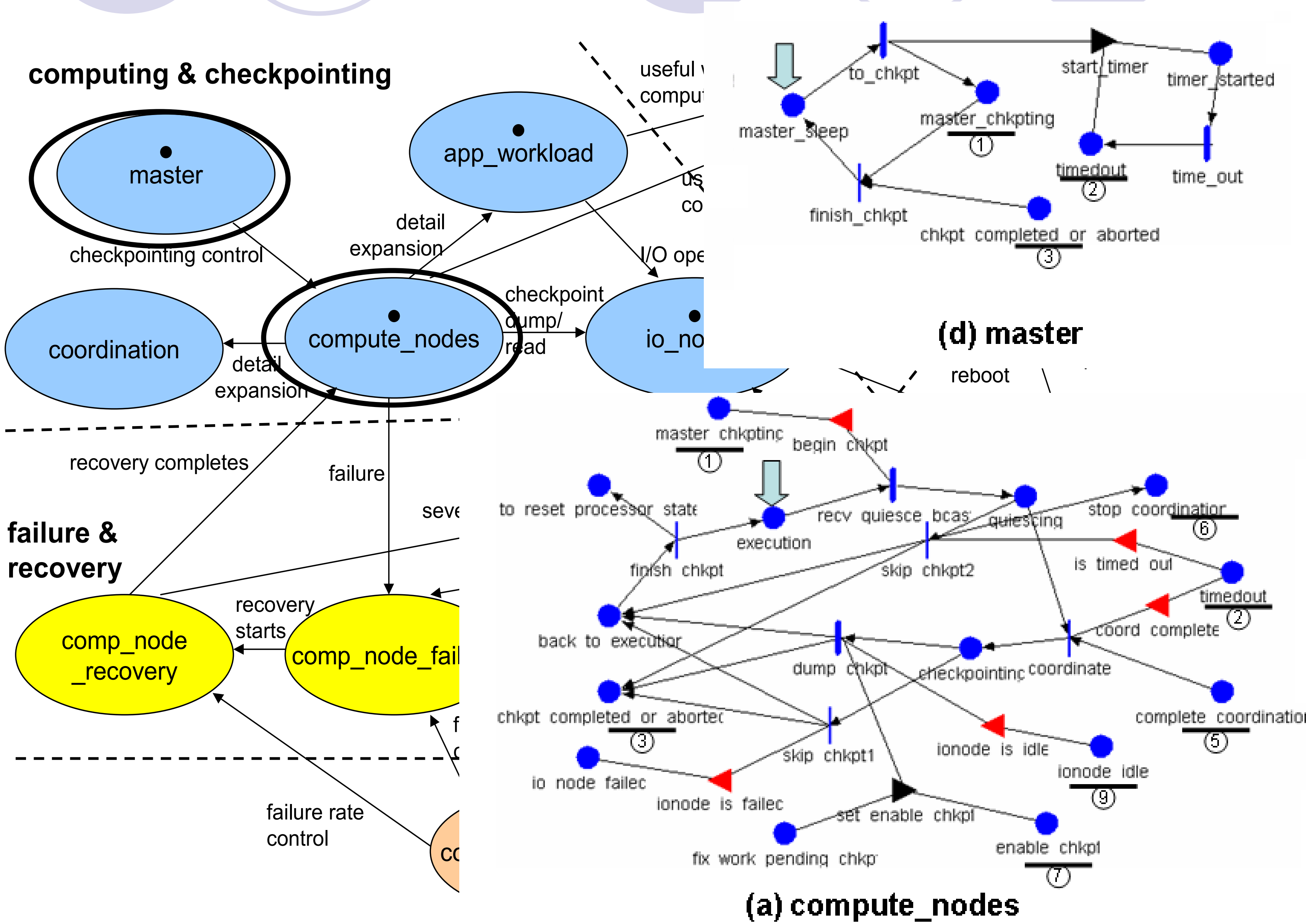
- Architecture
  - Multi-processor nodes
  - Compute nodes and I/O nodes
  - Two-step data transfers: *file system <-> I/O nodes <-> compute nodes*
- Checkpoint protocol
  - System-driven, synchronous, globally coordinated
  - Checkpoint data: memory image of application and OS (files not preserved)
  - Timeout-abort
  - No overwrite of the previous checkpoint unless current checkpoint completes successfully



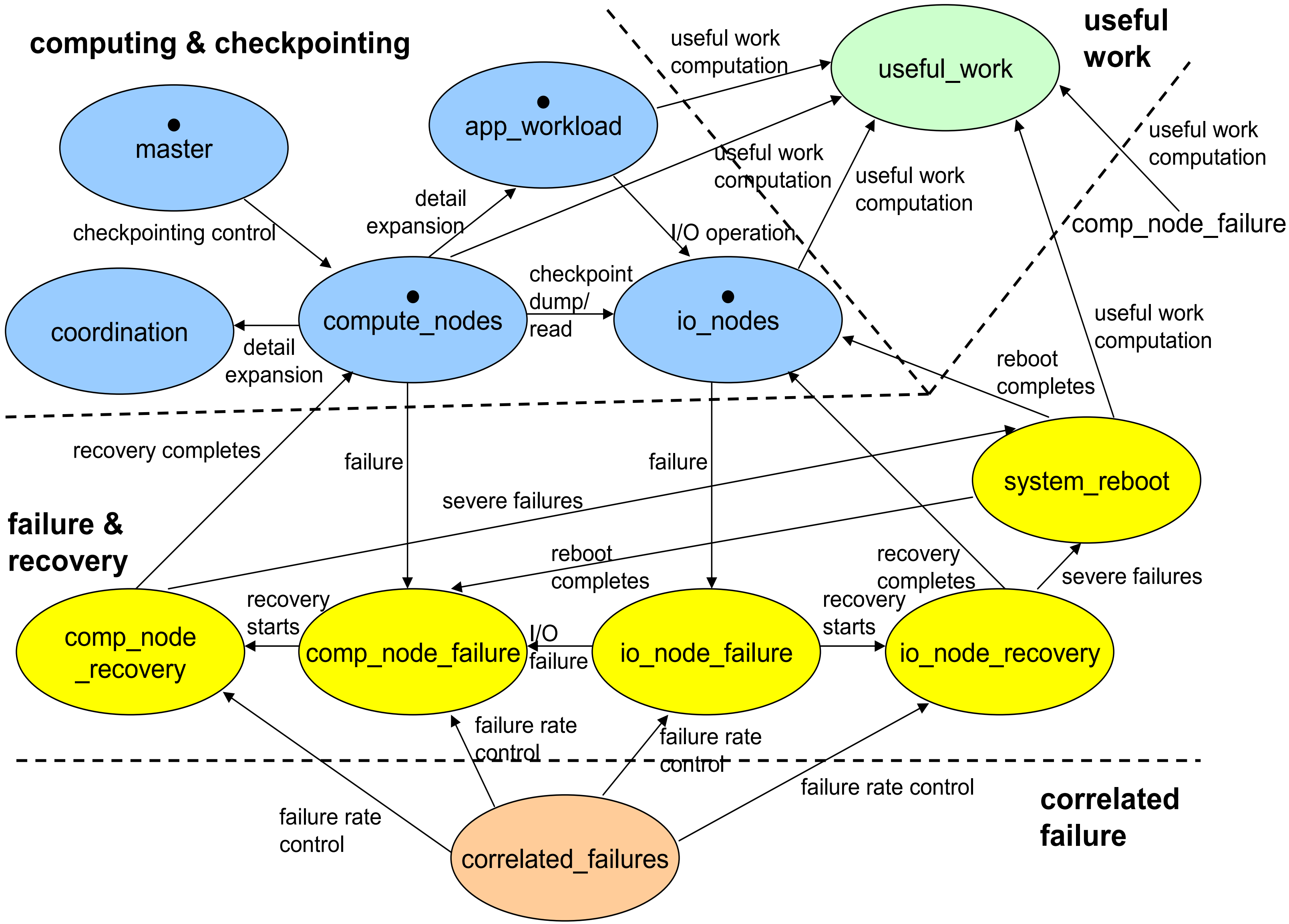
Target System (2)

- Application
  - Each processor runs one task of a parallel application
  - Bulk Synchronous Parallel model: multiple tasks behave as a single unit
  - I/O write cannot be quiesced until it completes
- Failure model and assumptions
  - Transient failures of compute and/or I/O nodes recoverable from a checkpoint
  - On a processor failure the whole system rollbacks to the last checkpoint and resumes the computation
  - Checkpointing coordinated by a *maser* node
    - On master failure, checkpoint protocol is aborted (if it was in progress) and the master resumes from the initial state
- Correlated failures
  - Due to error propagation (only)
  - Due to common cause, e.g., increase of environment temperature

Model Composition



Model Composition

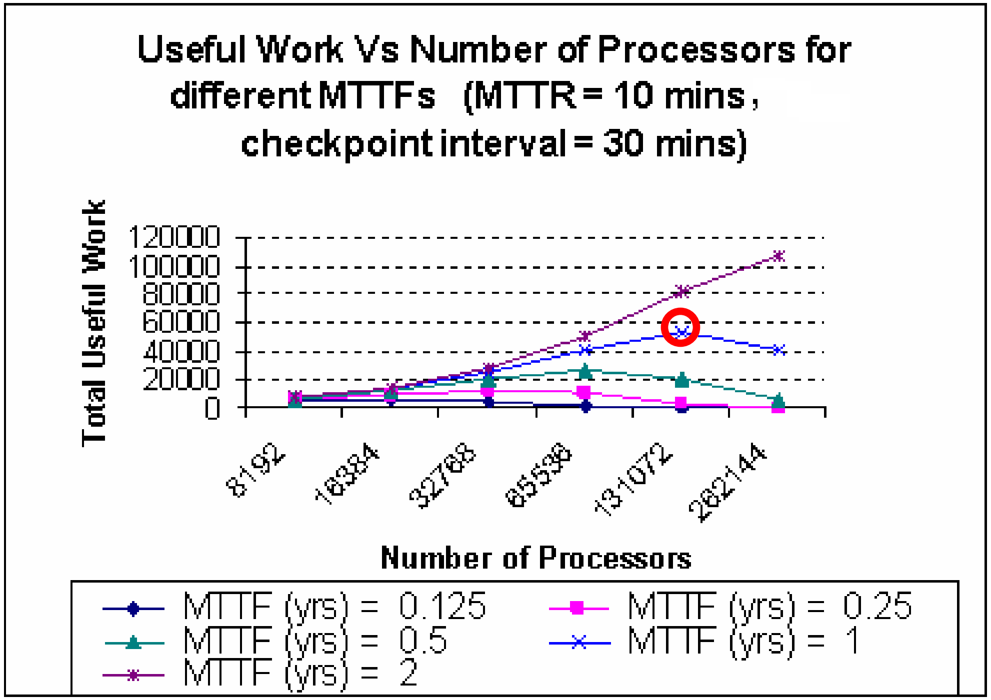


Simulation Experiment Setup

- Modeling and simulation environment: *Mobius*
  - Steady-state simulation (transient period of 1000 hours)
- Simulation experiments
  - Base model (without considering coordination or correlated failures)
  - Effect of checkpoint coordination
  - Impact of correlated failures
- Performance metrics
  - • **Useful work fraction:**
    - Fraction of time the system makes progress towards job completion
    - Work repeated due to failures is excluded
  - • **Total useful work:**
    - *(useful work fraction) x (number of compute processors)*
    - Indicates how many processors are required to achieve the same performance assuming failure-free computation

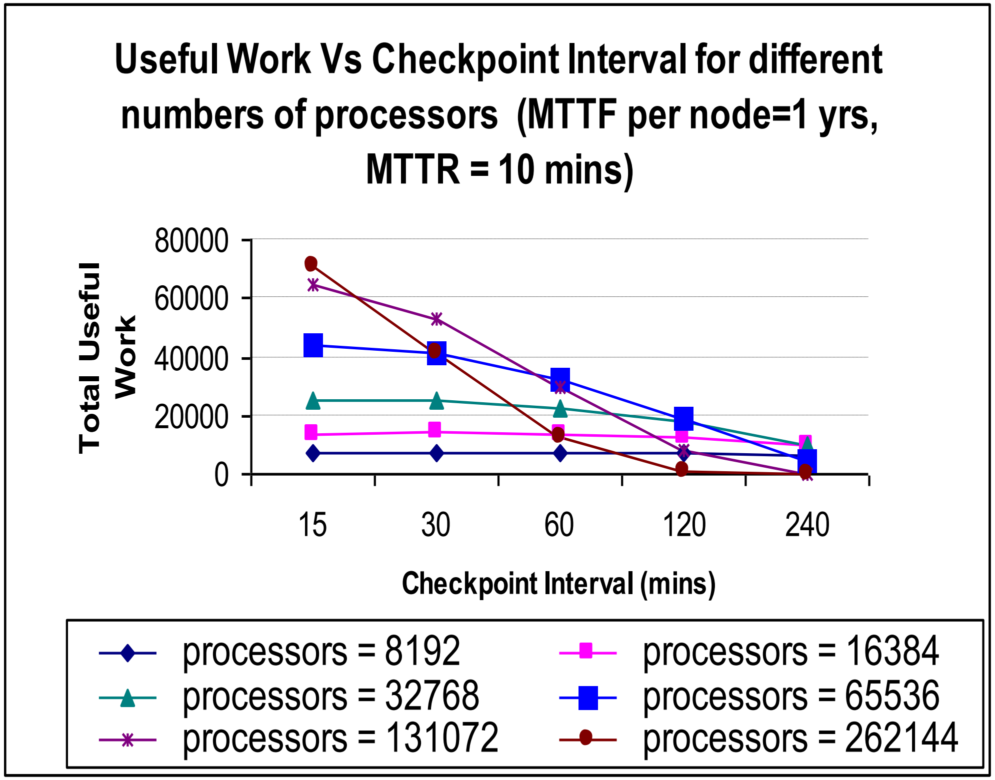
Results – Base Model (1)

- There exist an optimum number of processors for which total useful work is maximized
  - e.g., 128 K processors for *Chkpt interval* 30 min, *MTTR* 10 min, and *MTTF* 1 yr per node
  - adding more processors hurts system performance due to failure effects .
- The useful work fraction is relatively small
  - Less than 50%, for MTTF per node of 1 year
    - i.e., more than 50% of system resources used in checkpointing and recovering from failures



Results – Base Model (2)

- For any practical range there is no optimal checkpoint interval for which total useful work is maximized
  - the theoretical optimum is too short for practical purposes
- A better approach is to partition the system (if possible) and checkpoint each partition

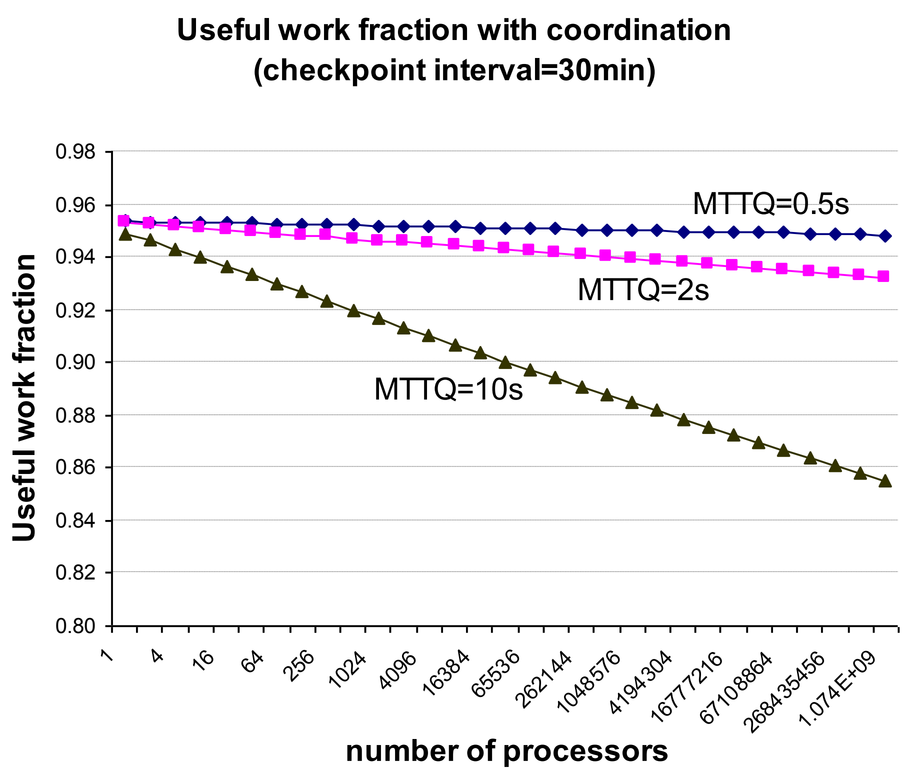


Results – Base Model (3)

- Useful work increases as number of processors per node increases
  - Number of nodes and the per-node failure rate remain the same
  - Use of advanced design and error handling techniques (multiple cores on a chip) may maintain low per-node failure rate with more processors per node
- Failures during checkpointing/recovery do not have a significant effect
  - Duration of checkpointing/recovery is much smaller than computation interval
  - Effects of failures during computation/recomputation dominate in large-scale systems

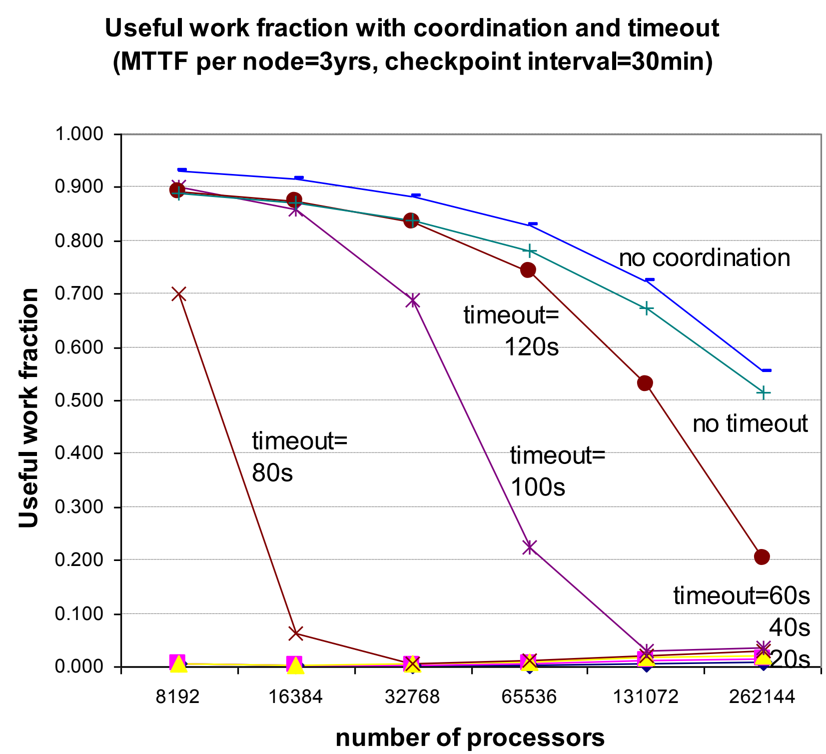
Results – Coordination Effect (1)

- Coordination does not affect system performance significantly
  - Identical exponentially distributed quiesce times assumed for all processors
  - Impact of coordination is logarithmic in the number of processors and scales well



Results – Coordination Effect (2)

- Combination of timeout and coordination behaves like a probabilistic checkpoint-abort
  - Small timeouts hurt the useful work fraction
  - Large timeouts do not significantly degrade performance
  - System performance insensitive to timeout value, when timeout is not less than a threshold value (120s in our experiment)

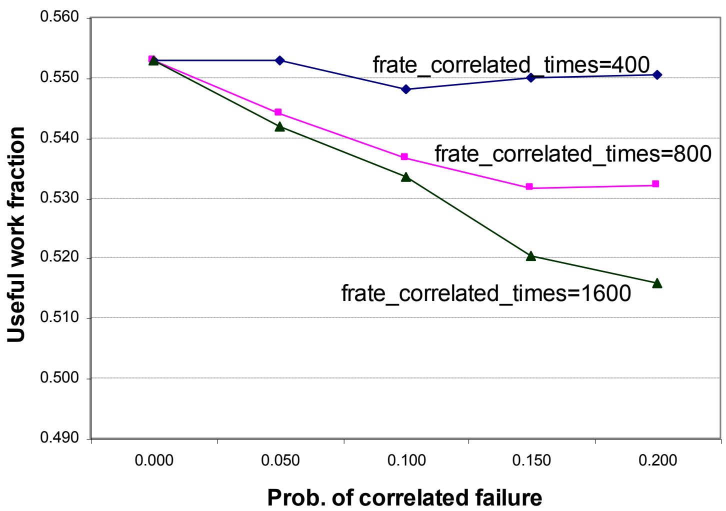


Results – Correlated Failures

Due to Error Propagation

- No significant performance degradation
  - Correlated failures occur during recovery
  - Recovery time much shorter than computation interval

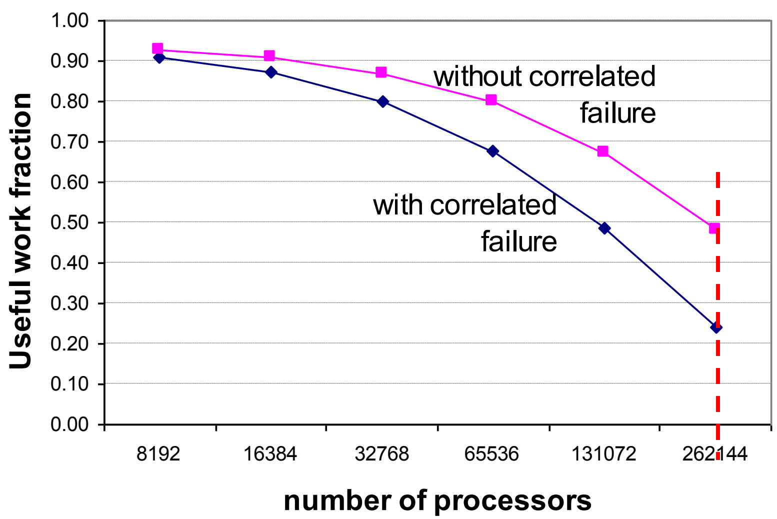
Useful work fraction (MTTF per node=3yrs, number of processors=256K, correlated failure window=3min)



Due to common cause

- Large performance degradation
  - e.g., ~51% reduction in useful work fraction for system with 256K processors and MTTF of 3 years per node

Useful work fraction (MTTF per node=3yrs, correlated failure coefficient=0.0025,correlated failure factor=400, checkpoint interval=30min)



Conclusions

- A model of coordinated checkpointing for supercomputers
- There exist an optimum number of processors for which total useful work is maximized
- Useful work fraction is relatively small due to failure effects
- Failures during checkpointing/recovery do not have a significant effect
- Correlated failures degrade the performance and limit system scalability
- Coordination effect
  - System performance insensitive to the timeout value unless timeout is less than a threshold value

Related Work

- Checkpointing Models
  - [Young74]: assumes MTBF is very large compared to the checkpoint and recovery time
  - [Daly03]: does not model the coordination overhead
  - [Kavanagh97]: does not consider failures during checkpointing and recovery
  - [Plank99]: considers permanent failures
  - [Elnozahy04]: does not consider coordination failure or correlated failure
  - [Vaidya95]: does not consider scalability of checkpointing protocol
- Checkpointing in Large-Scale Systems
  - [Bronevetsky03]: compiler-based technique for coordinated checkpointing
  - [Agarwal04]: adaptive incremental checkpointing for scientific applications
- Failure Study in Large-Scale Systems
  - [Zhang04]: shows existence of temporal and spatial failure correlation