Investigating the Benefits of Redundancy Plus Checkpointing for Hard-Fault and Soft Error Protection in HPC

David Fiala, James Elliot, Kishor Kharbas Advisor: Frank Mueller (NCSU)
Collaborators: Christian Engelmann (ORNL), Rolf Riesen, Kurt Ferreira (SNL)

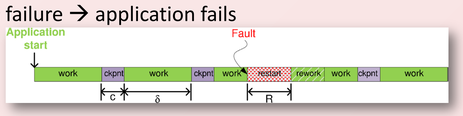
NC STATE UNIVERSITY

Sandia National Laboratories

OAK RIDGE National Laboratory

Motivation

- Component failures require support of checkpoint/restart (C/R)
- Node or switch failure → application fails

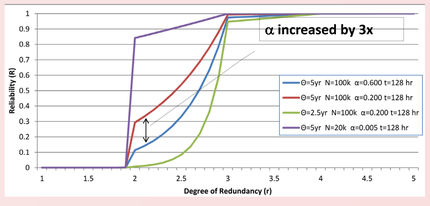


- Adding hardware increases the likelihood of faults
  - The probability of component failure combinatorially explodes
  - The mean-time-between-failure (MTBF) shortens
  - Overhead due to C/R increases exponentially
    - Computation vs. overhead ratio can be between 85%-55%

168-hour Job, 5 year MTBF [Sandia]				
# nodes	Work	Checkpoint	Re-computation	Restart
100	96%	1%	3%	0%
1,000	92%	7%	1%	0%
10,000	75%	15%	6%	4%
100,000	35%	20%	10%	35%

- **C/R does not increase reliability**, but redundancy can reverse this trend
  - Each redundant process decreases the probability of failure of replica processes
  - Less interruptions produces greater utilization
  - 100% redundancy provides 5x job throughput [Sandia]

- In the face of fail-stop failures, **redundancy increases reliability**



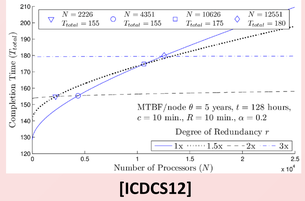
Θ = Mean time between failure (MTBF)

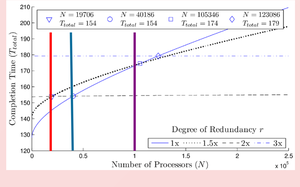
α = Communication to computation ratio

Time becomes function of α

[ICDCS12]

- 2x redundancy may be beneficial now: **at 78,536 processes, two dual redundant jobs of 128 hours can be run in the time of just one job without redundancy**





•Jaguar: node MTBF ~ 50 years (on 18,688 nodes)
•K-Computer: has 2.3X more components (equiv. 44,064)
•Exascale lane 1: ~100k nodes

[ICDCS12]

- Another class of fault: **Silent Data Corruption (SDC)**
- SDC faults manifest themselves as bit-flips in storage or even within processing cores
  - In some cases bit-flips are not correctable or even detected
  - Exacerbating this situation, when SDC goes undetected invalid results are reported
    - Memory becomes corrupt, but applications continue to run
    - This is a severe problem for today's large-scale simulations

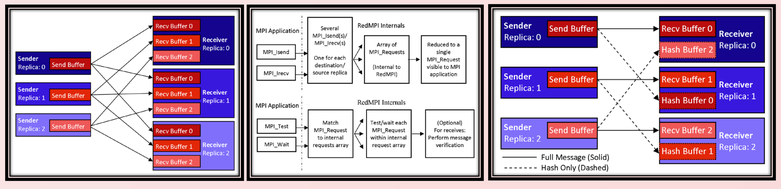
RedMPI: Protecting against SDC Faults

- Redundancy for HPC applications provides more than just fail-stop protection
  - **The framework for providing redundancy also provides a platform to both protect applications and investigate the effects of faults**

- Provide transparency by linking unmodified MPI applications with our RedMPI
- RedMPI provides redundancy to MPI applications by instrumenting the MPI profiling layer
  - Adjusted MPI rank and size provide illusion of normal rank numbers
  - SDC protection is afforded by augmenting MPI functionality to communicate with replicas

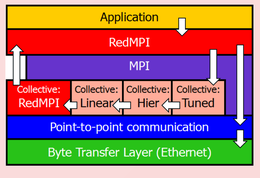
	No Redundancy	Dual Redundancy	Triple Redundancy or higher
Live SDC Detection	No	**Yes**	**Yes**
Live SDC Correction	No	No	**Yes** (via voting algorithm)

- Naïve SDC protection may be achieved by transmitting and comparing *r*r* messages amongst *r* total replicas.
  - Induces high interconnect contention / bandwidth degradation
  - Compare received buffers, discard a mismatch



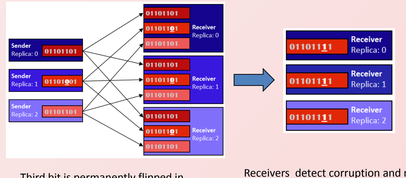
- Enhance performance by sending the original message plus a small hash to a separate replica
  - No longer dependent on *r*r* communication
  - Comparison still performed on receiver-side
  - Hash mismatch triggers secondary voting protocol amongst receiving replicas

- **Collectives require protection too** – and must operate in the face of faults
  - **Reuse existing topology-aware tuned algorithms**, route messages via RedMPI to ensure success



Fault Injection

- Sender side: 1/x messages randomly receive single bit-flip in application buffer
  - This corrupts the sender's memory and spreads to peers as incorrect MPI messages

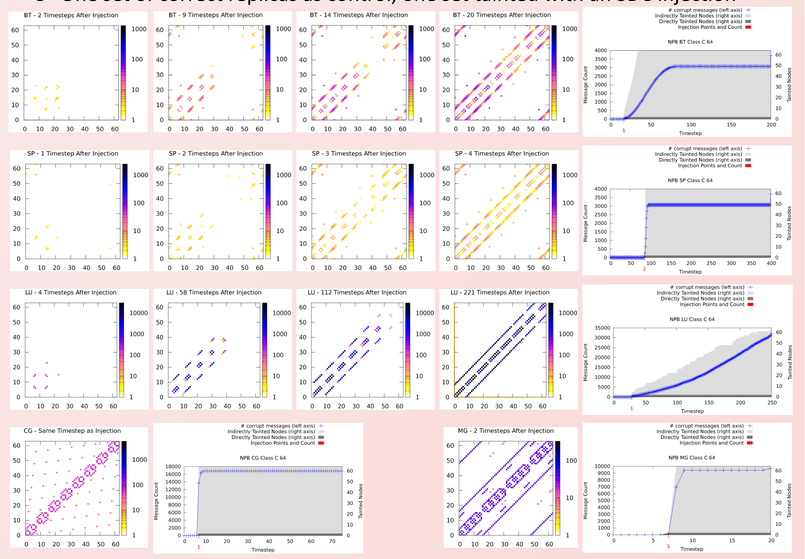


Third bit is permanently flipped in sender's buffer --- passed on to receivers

Receivers detect corruption and may retain only the correct message
(Protection may be disabled to track application progress in the face of faults)

Observation: SDC Propagation

- What is the impact of a just one bit-flip?
  - Run RedMPI with dual redundancy, but **disable protection** & **track** live execution
  - One set of correct replicas as control, one set tainted with an SDC injection



- Profound effects may spread quickly, depending on communication patterns

Observed Redundancy Costs

- Experiments performed on 96 cluster nodes
  - AMD Opteron 6128 (Magny Core) – 16 cores per node – 32GB RAM per node
  - 40Gbit/s Infiniband for MPI Communication

1x: Uninstrumented Open MPI (No Redundancy)
2x: RedMPI with Dual Redundancy
3x: RedMPI with Triple Redundancy

LAMMPS – CHUTE.SCALED					
Size	1x [sec]	2x [sec]	3x [sec]	2x OV	3x OV
128	137.5	138.4	139.0	0.6%	1.1%
256	138.3	140.4	140.0	1.6%	1.3%
512	139.2	140.2	141.0	0.7%	1.1%

SWEEP3D					
Size	1x [sec]	2x [sec]	3x [sec]	2x OV	3x OV
128	390.3	389.5	393.1	-0.2%	0.7%
256	428.2	427.5	431.2	-0.1%	0.7%
512	488.1	488.9	494.1	0.2%	1.2%

HPCCG – wildcard receives present					
Size	1x [sec]	2x [sec]	3x [sec]	2x OV	3x OV
128	99.8	99.8	125.8	0.0%	26.0%
256	99.6	128.8	131.0	29.3%	31.5%
512	126.4	146.2	152.3	15.7%	20.5%

NPB - CG					
Size	1x [sec]	2x [sec]	3x [sec]	2x OV	3x OV
128-D	201.42	205.87	215.51	2.2%	7.0%
256-D	127.21	132.61	136.64	4.2%	7.4%
512-D	70.10	77.54	83.67	10.6%	19.4%

NPB - EP					
Size	1x [sec]	2x [sec]	3x [sec]	2x OV	3x OV
128-D	72.31	72.63	72.74	0.4%	0.6%
256-E	579.94	581.02	581.27	0.2%	0.2%
512-E	289.80	290.83	291.30	0.4%	0.5%

NPB - FT					
Size	1x [sec]	2x [sec]	3x [sec]	2x OV	3x OV
128-C	117.45	117.95	118.68	0.43%	1.05%
256-C	68.82	68.62	71.77	-0.29%	4.29%
512-D	222.75	228.76	234.97	2.70%	5.49%

Conclusions

- For large systems, C/R + redundancy increases job throughout
- Redundancy is cheap in terms of software overhead
- Application sensitivity to soft errors may be very high
- SDC protection comes free as redundancy is used to increase system resilience
  - RedMPI can successfully protect applications from SDC faults and continue execution to a successful, correct completion

This work was supported in part by NSF grants CNS-1058779, CNS-0958311, DOE grant DE-FG02-08ER25837, a subcontract from Sandia National Laboratory. and by the Laboratory Directed Research and Development Program of Oak Ridge National Laboratory (ORNL), managed by UT-Battelle, LLC for the U. S. Department of Energy under Contract No. De-AC05-00OR22725.