NV-PROCESS: A FAULT-TOLERANCE PROCESS MODEL BASED ON NON-VOLATILE MEMORY

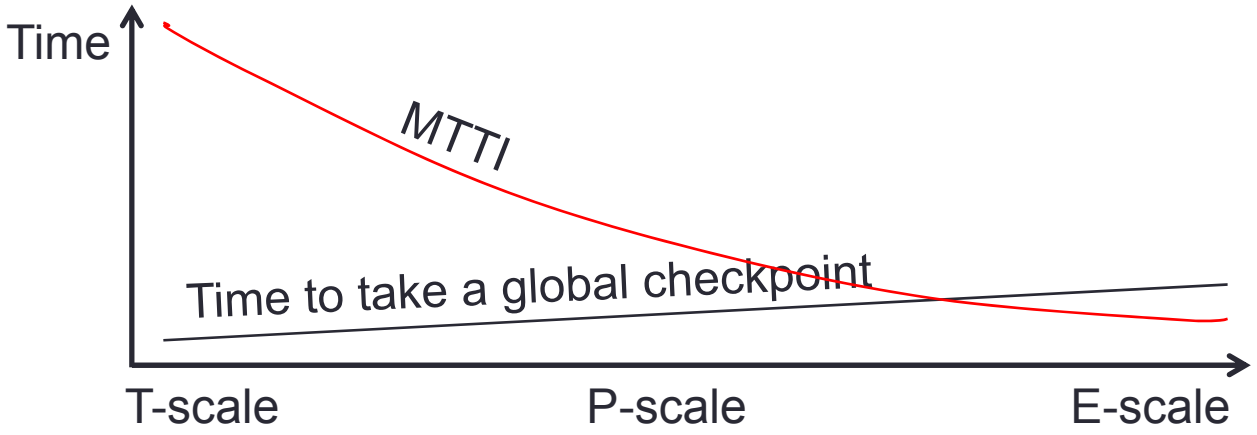
Xu Li, Kai Lu, Xiaoping Wang, and Xu Zhou
National University of Defense Technology, China

Fault tolerance & HPC

- Reliability challenge of HPC systems
  - Inherent reliability of the whole system decreases with system scale increasing
  - The transient error rate of chips increases with increasing circuits on a single chip
  - The software failure rate increases with software complexity increasing
  - The application scale increases dramatically, and an application needs to run for a long time
- HPC systems must rely on fault tolerance technique to preserve system usability

Challenge of Fault Tolerance

- Checkpoint technique is commonly used in HPC systems
- With system scale increasing
  - Checkpoint cost increases
  - MTTI decreases, thus requires more frequent Checkpoint
- When checkpoint time is close to MTTI, the utilization ratio of HPC will be zero



Our approach

- We propose a new process model, called NV-process, based on the emerging NVRAM
- NVRAM vs. DRAM

NVRAM	DRAM
Non-volatile	Volatile
Byte-addressable	Byte-addressable
Fast	Fast

- NVRAM is a candidate of the main memory in future
  - Phase-change memory(PCM)
  - Memristors
  - Spin-transfer Torque MRAM

NV-process

- NV-process instances are independent from the OS
  - NV-process instances persist their states on the NVRAM while running
  - NV-process instances could survive across OS reboot
- NV-process enables native fault-tolerant
  - Just reboots the OS kernel in the presence of OS crash or power failure
  - After OS reboot, continues the execution of a process where it is left off

Design

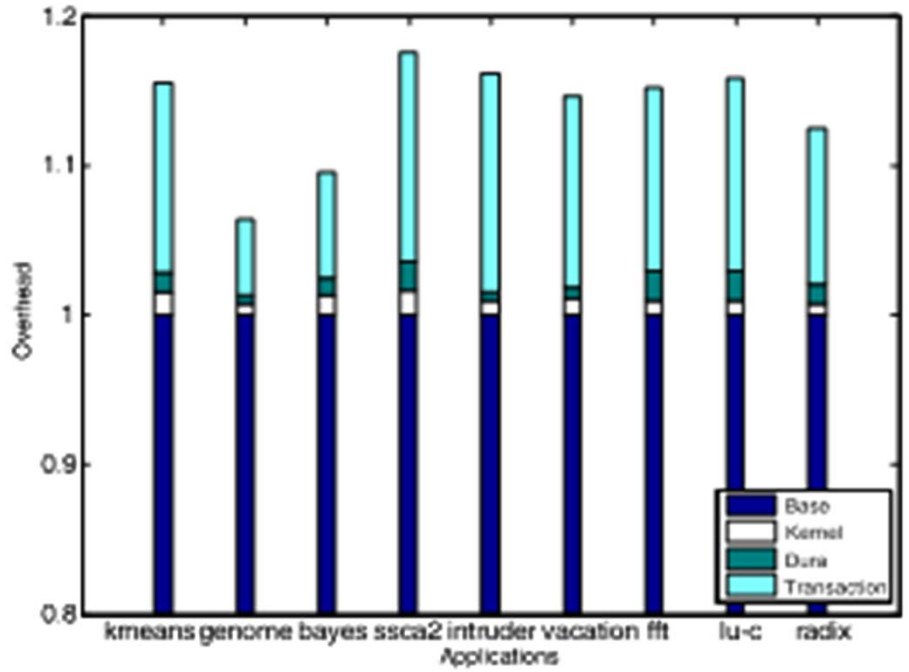
- Decouples a process from OS
  - Manages processes in a self-contained way
  - Provides a process independent running space
- Consistency-preserved execution
  - Maintains process consistent state during execution
- In-place restart
  - Restarts a process in-place and continue the execution where it is left off

Implementation

- Independent virtual and physical(IVP) memory space
  - Adds a new zone named ZONE_NV as independent physical memory space for processes
  - Maintains the PCB and user space of a process in ZONE_NV
- Non-volatile process system(NVPS)
  - Manages NV-process instances in a queue
  - Maintains the queue in ZONE_NV
- Transactional execution
  - Runs an application in transactional mode
- Resumption
  - Recognizes the NV-process instance and puts it in new process scheduling queue
  - Builds a temporary user stack to restart the process in-place
  - Enables the process to execute from the unfinished transaction

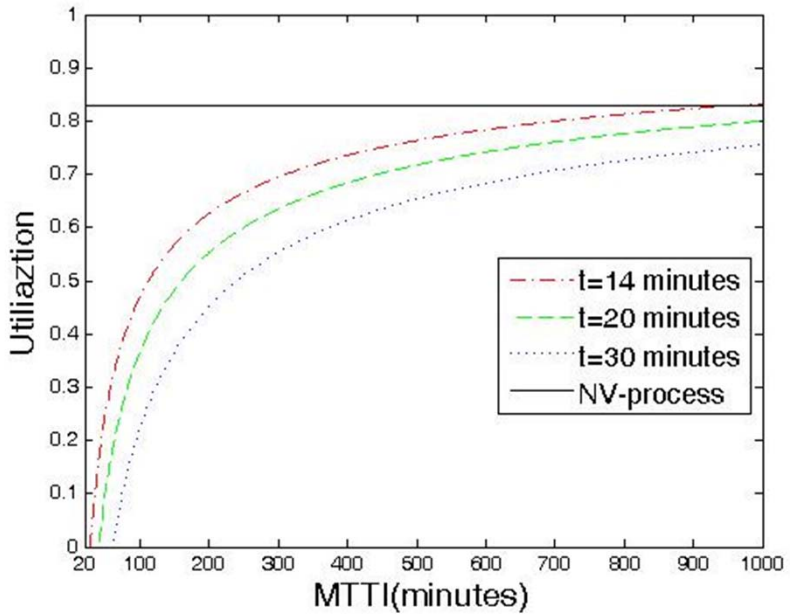
Experiment

- Runtime performance
  - NV-process introduces a time overhead no more than 18%
  - The overhead mainly comes from transactional execution



Experiment

- Fault-tolerance performance
  - NV-process could keep the utilization ratio of system above 80%, while CR decreases the system utilization ratio to zero when MTTI is less than 20 minutes



Conclusions

- We propose a fault tolerance process model based on NVRAM, called NV-process
- NV-process enables an application to be fault tolerant natively, which could continue the execution of a process where it left off in the presence of OS crashes.
- NV-process keeps the HPC system utilization above 80%, and could preserve the utilization ratio of future HPC systems


THANK YOU!