Kento Sato

Computer Scientist

Lawrence Livermore National Laboratory
7000 East Ave., Livermore, CA 94550
☎ *925(422)-6918*
✉ *kento@llnl.gov*
http://people.llnl.gov/sato5

June 4, 2018

Research Statement

Summary

My research interest is High Performance Computing (HPC). My major research areas in HPC include (Area 1) Big data processing, filesystem and I/O optimization; (Area 2) Resilience and Fault tolerance; (Area 3) HPC tools; (Area 4) Cloud computing.

Area 1: User-level filesystem and I/O Optimization

2014-present **HuronFS (Hierarchical, User-level and On-demand File System):** When running data-intensive HPC applications which issue a huge amount of concurrent or parallel I/Os to shared storage, current public clouds cannot provide desirable execution environments for such I/O workloads with respect to performance and data consistency. In order to resolve these problems, our research group proposed a novel fast, scalable and fault tolerant filesystem called CloudBB (Cloud-based Burst Buffer). Unlike conventional filesystems, CloudBB creates an on-demand two-level hierarchical storage system and caches popular files to accelerate I/O performance. CloudBB enables scalable I/O with multiple metadata servers. CloudBB is also resilient to failures by using file replication, failure detection and recovery techniques. We eventually released the CloudBB filesystem as HuronFS.

2014-present **CDC (Clock Delta Compression** The ability to record and replay program execution helps significantly in debugging non-deterministic MPI applications by reproducing message-receive orders. However, the large amount of data that traditional record-and-reply techniques record precludes its practical applicability to massively parallel applications. To reduce record dize, our research group proposed a new compression algorithm, Clock Delta Compression (CDC), for scalable record and replay of non-deterministic MPI applications.

2014 **gmfs (User-level GPU-accelerated I/O Interface):** To exploit accelerators and improve I/O-bound applications, our research group developed gmfs (GPU-accelerated I/O interface) that utilizes GPU device memory as buffer cache.

miscellaneous **Other filesystem works:** Also, I have been partly working on an ephemeral burst-buffer file system for CORAL systems.

Research area 2: Resilience and Fault tolerance

2013-2015 **Lossy compression for checkpoint/restart:** The I/O performance of parallel filesystems will be far behind the increase in computational performance. As such, there have been various attempts to decrease the checkpoint overhead, one of which is to employ compression techniques to the checkpoint files. While most of the existing techniques focus on lossless compression, their compression rates and thus effectiveness remain rather limited. Instead, Our research group proposed a lossy compression technique based on wavelet transformation for checkpoints, and explored its impact to application results.

2013-2014 **Reliable Storage Architecture:** To propose resilient HPC systems, our research group explored how burst buffers can improve efficiency compared to using only traditional node-local storage. To fully exploit the bandwidth of burst buffers, we developed a user-level InfiniBand-based file system (IBIO). We also developed performance models for coordinated and uncoordinated checkpoint/restart strategies, and we applied those models to investigate the best checkpoint strategy using burst buffers on future large-scale systems, and validated effectiveness of use of burst buffers.

2012-2014 **FMI (Fault Tolerant Messaging Interface):** We presented the Fault Tolerant Messaging Interface (FMI), which enables extremely low-latency recovery. FMI accomplishes this using a survivable communication runtime coupled with fast, in-memory C/R, and dynamic node allocation. FMI provides message-passing semantics similar to MPI, but applications written using FMI can run through failures.

2012-2013 **Energy-aware Checkpointing:** Both energy efficiency and system reliability are significant concerns towards exascale high-performance computing. However, checkpoint/restart can use a large portion of runtime, and consumes enormous energy by even non-I/O subsystems, such as CPU and memory. our research group presented an energy-aware I/O optimization technique for NAND flash memory devices based on a Markov model for checkpoint/restart.

2011-2012 **Design and modeling of asynchronous checkpointing:** In this work, our research group designed an asynchronous checkpointing system and combined the benefits of asynchronous checkpointing and multi-level checkpointing. We also proposed the asynchronous and multi-level checkpoint/restart model. Our experiments show that our system can improve efficiency by 1.1 to 2.0 $\times$ on future exascale machines. Additionally, applications using our checkpointing system can achieve high efficiency even when using a PFS with lower bandwidth.

Research area 3: HPC tools

2016-present **NINJA (Noise injection agent tool):** Noise injection tool to expose non-deterministic message-race bugs.

Research area 4: Cloud Computing

2007-2011 **VM migration for efficient I/O:** I/O optimization for data-intensive applications running on virtual machines.