VeloC/SZ

Long-running large-scale simulations and high-resolution, high-frequency instrument detectors are generating extremely large volumes of data at a high rate. While reliable scientific computing is routinely achieved at small scale, it becomes remarkably difficult at exascale due to both an increased number of disruptions as the machines become larger and more complex from, for example, component failures and the big data challenge. The VeloC/SZ project addresses these challenges by focusing on ensuring high reliability for long-running exascale simulations and reducing the data while keeping important scientific outcomes intact.

Big data challenges need to be addressed at exascale for applications to achieve their performance and science goals. The VeloC/SZ project addresses two of these big data challenges. First, the ability to run scientific simulations until completion despite disruptions along the way is critical. VeloC provides a highly reliable environment for exascale applications at a minimal cost, enabling them to fully benefit from the extreme data volume and velocity they produce with a low overhead. Second, data reduction is necessary to reduce the size of the data output to the storage system due to bandwidth and storage space limitations. SZ addresses this challenge by enabling application scientists to reduce their scientific data while keeping scientific outcomes intact.

Most large-scale scientific applications use execution state recording techniques to make sure the execution finishes, despite disruptions. If a disruption occurs, the execution state can be restored and the application can be restarted from this state. This technique is known as checkpoint/restart. At exascale, this technique is difficult to implement at low cost for the applications due to an extremely large volume/velocity of data, complex disruption modes, and limited bandwidth to the storage system. Moreover, the diversity and complexity of the storage hierarchy in exascale systems make it very difficult for application developers to implement checkpoint/restart at low cost. VeloC leverages application developer knowledge about state preservation to provide a solution optimizing the performance of checkpoint/restart while masking the complexity and diversity

of the storage hierarchies. An existing application can be adapted for VeloC in minimal time. Once adapted, the application can run in a highly reliable way on pre-exascale and exascale machines.

As data sizes increase with exascale systems and updated scientific instruments, lossy compression of scientific data becomes a necessity. Lossy compression reduces the data by removing non-useful information. Lossy compression for scientific data needs to satisfy three main requirements: it should remove only information that does not impact scientific discovery; compression and decompression need to be very fast to avoid raising a performance issue; and it needs to be effective at providing data reduction much higher than lossless compression. The SZ software provides lossy compression for scientific datasets satisfying these three requirements. To keep information relevant for scientific discovery, SZ users set constraints in terms of compression quality. To control the information loss for each data point, SZ provides point-wise error bound controls that the user supplies. To reach extremely high performance, the SZ software has a parallel implementation that benefits from GPU acceleration. The advanced compression pipelines used in SZ provide very high compression ratios compared with lossless compression, enabling SZ to overcome the big data challenges at the exascale.

Progress to date

- The VeloC/SZ team released version 1.0 of the VeloC software. The team closely collaborated with several exascale application teams to refine the VeloC API and make sure it addresses their needs. The client library and backend were designed and implemented. The erasure-coding module and the data transfer module were integrated with the backend into a flexible engine that allows VeloC the capability of running in synchronous mode directly in the application processes or in asynchronous mode in a separate process. Results show that the impact of checkpointing (measured as increase in runtime vs. the case when no checkpointing is used) was reduced by up to 10× when using VeloC.

- The team drastically improved the performance of SZ using innovative algorithms and node-level parallelization and GPU accelerators to reduce compression and decompression time. SZ can be integrated directly in the application, or it can be used transparently through the ADIOS, HDF5, and PnetCDF I/O libraries. Compression results are outstanding in terms of performance and compression ratios, and SZ is currently being used by six ECP applications. **Typically, conventional compression will reach compression ratios between 1 and 2 on scientific data sets. ECP application users of SZ typically reached compression factors of 10.**

PI: Franck Cappello, Argonne National Laboratory

Collaborators: Argonne National Laboratory