Towards the Adaptation of the Graph500 benchmark to the SHARed data-structure-centric Programming (SharP) paradigm

Achievement: Designed, implemented, and preliminarily evaluated the adaptation of the Graph500 benchmark to SharP with a data-centric focus.

Significance and Impact: The adaptation of the Graph500 benchmark shows the significance of data-locality with respect to data placement and the capabilities of SharP.

Research Details:

- Designed the necessary modifications to the Graph500 benchmark to properly leverage SharP and become data-centric with respect to data locality near the NIC while also maintaining the same application semantics of the reference implementation.
- Implemented the required modifications to the Graph500 benchmark.
- Performed a preliminary evaluation of the adaptation of Graph500 with SharP on ORNL’s Titan system and compared the performance of this adaptation with the MPI-2 one-sided reference implementation and a Cray SHMEM implementation.

Sponsor/Facility: Work was performed with support from ORNL

PI and affiliation: Manjunath Gorentla Venkata from CSMD – Oak Ridge National Laboratory

Team: Ferrol Aderholdt, Manjunath Gorentla Venkata, and Zachary W. Parchman – CSMD, Oak Ridge National Laboratory.

Overview:
The Graph500 benchmark is a data-intensive benchmark meant to be a compliment to the compute-intensive benchmark commonly used for ranking supercomputers within the Top500 list (i.e, the High Performance Linpack (HPL) benchmark). The Graph500 benchmark constructs a large undirected graph and performs multiple breadth-first search (BFS) operations on this graph. After the BFS operations are completed, the benchmark proceeds to validate the spanning tree that was created to ensure proper traversals. With respect to data usage, the benchmark provides low spatial and temporal locality and views memory as flat with no support for the heterogeneous and hierarchical memory that will be present in future extreme-scale systems. Additionally, the current design and implementation of the MPI-2 one-sided reference implementation does not determine the locality of the data to the nearest high performance NIC, which will provide higher latency for operations on data items far from the NIC when compared to those close to the NIC. This is important as the benchmark itself is extremely latency sensitive. This work adapts the MPI-2 one-sided reference implementation to leverage the SharP
library, which allows for memory to be allocated near the NIC and structures data on behalf of the user. The implementation of this approach was preliminarily evaluated on the Titan supercomputer located at the OLCF comparing the benchmark with its MPI-2 one-sided reference implementation and an implementation in Cray SHMEM. We found the usage of SharP and focusing on a data-centric approach increased the performance of the benchmark with respect to traversed edges per second (TEPS) by as much as 11x with respect to Cray SHMEM and roughly 18x when compared to MPI.