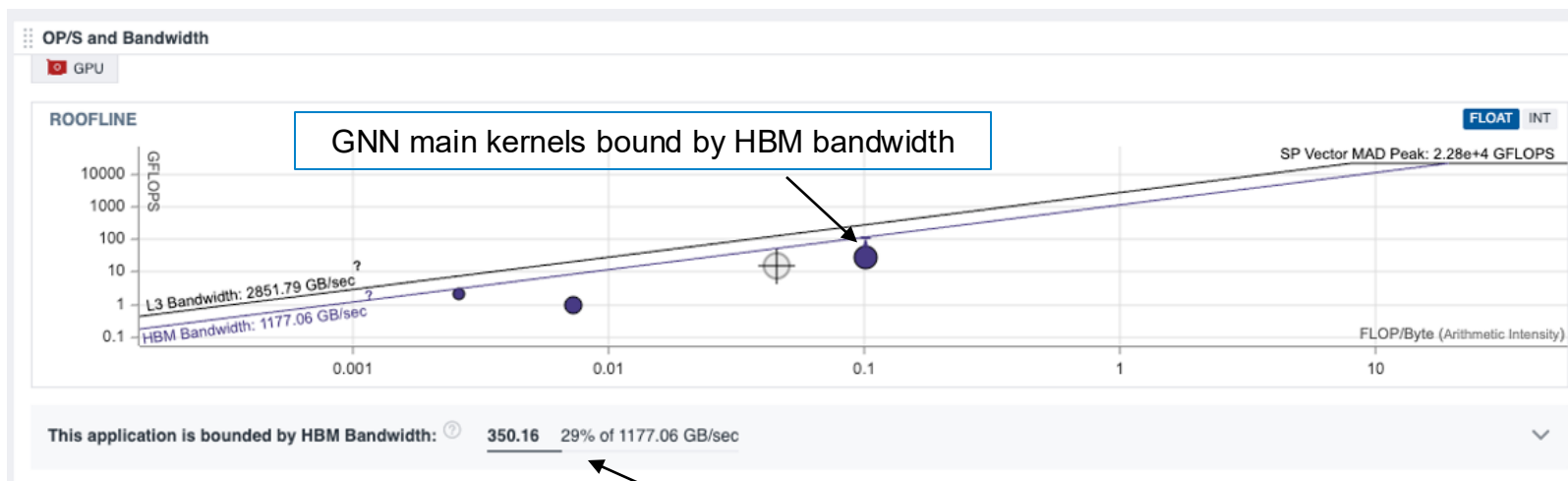


# DIST-GNN: OVERVIEW

- Dist-GNN model for mesh-based modeling of dynamical systems
  - Operates on partitions of the entire graph, called sub-graphs, created directly from the CFD mesh used by nekRS
  - Size of sub-graph determined by nekRS mesh size and by polynomial order used to create GNN graph (this is  $\leq$  p-order used by nekRS)
  - Implements a halo exchange in the neural message passing layers to ensure node aggregation steps span across sub-graphs, thereby enabling training and inference on extremely large graphs
- Implemented in PyTorch and PyTorch Geometric
- Run with FP32 or BF16 (FP16 or lower not suitable for science needs)

# ROOFLINE (FP32)

- Dist-GNN training roofline collected with Intel Advisor on single Aurora PVC tile
  - GNN main kernels are **HMB bandwidth bound** (even GEMM)
  - Dist-GNN application as a whole is also **HBM bandwidth bound**



GNN application bound by HBM bandwidth

# GPU UTILIZATION ON AURORA

- GNN training GPU utilization metrics
  - Collected with xpu-smi on single PVC tile of Aurora
  - Note high (~85%) GPU memory utilization
  - Problem and model size for GNN **bound by GPU HBM memory size**

