

Triton Intel XPU update

jian.hui.li@intel.com

eikan.wang@intel.com

Oct 25, 2023

Status update

- Rebased to latest Triton, Triton LLVM upgrade caused building failure and we fixed
- Performance analysis using Torchinductor-based microbench (targeting Triton)
- GEMM on progress – also try to generalize the MMA layout
- CPU enabling

<https://github.com/intel/intel-xpu-backend-for-triton>

GEMM lowering

GEMM performance recipe on Xe GPU	Challenges to optimize Triton	Potential solutions
Need to use 2d block load	Doesn't match "Tensor of pointer" programming model	Map "block pointer" to 2d block load Map "tensor of pointer" to regular load and feed to DPAS.
Cooperative prefetch to cache vs. slm	Existing Triton-GPU passes doesn't help Xe GPU	Requires Xe-specific passes on Triton GPU dialect
Vector Computation (VC) mode at subgroup/warp level	Triton-GPU lowering directly to thread level for each work item	Gradually lower Triton GPU dialect to warp level and map to XeGPU dialect for both SIMT and VC mode.
more opti.: Maximize 2d load efficiency, named barrier	Further bloat Triton code base	Reuse XeTile dialect (larger tile, hide hardware-specific optimization/limitations)

Questions

- Triton v3.0 release – need extra time period for intel backend to stabilize. Timeline?
- Triton CPU backend – performance requirement and timeline?
 - – explored SPIRV through OpenCL runtime on CPU
- Timeline for TritonGPU generalization?
- Microbenchmark efforts on Triton?
- Adding gradual lowering: add subgroup/warp level before WI/thread level?