Background: Task-based and loop-based

Recently, task-based execution has been attracting attention, because it can reduce the waiting time from synchronization. 
- Loop-based: 
  - Exploit data parallelism, and execute the same computation on different data. 
  - Use barrier between jobs to make sure the loop execution is already finished. 
- Task-based: 
  - Exploit task parallelism, and execute different tasks in parallel. 
  - Use tasks and their dependencies to control the order of task execution. 
A directed acyclic graph (DAG) is used to express the dependency.

High Performance ParalleX (HPX)

- A runtime system for parallel computing based on the partitioned global address space (PGAS) model. 
- Provides a C++ class library to describe tasks and their dependencies.

Proposed Approach

- **STEP 1: Choose the critical task.** 
  - Find the critical path in the DAG. 
  - The longest path in the DAG, which has more tasks than the other paths. 
  - Find the critical task in the critical path. 
  - The task(s) that have the highest number of occurrences in the critical path. 

- **STEP 2: Decoupled task queues and thread pools.** 
  - A higher priority is given to critical tasks in the critical task queue. 
  - Tasks from the critical task queue go to the critical thread pool, and the other tasks go to the default thread pool. 
  - Whenever a worker thread becomes idle, an HPX thread is retrieved from the thread pool, and assigned to the worker thread.

- **STEP 3: Using a NUMA-balanced method for mapping tasks.** 
  - Different from the default thread mapping method, the NUMA-balanced method will assign threads evenly to cores. 
  - More efficient by making better use of the resources.

Task Priority Control

- All tasks are managed using a single default task queue. 
- Tasks will be assigned to HPX threads in a thread pool in FIFO, every task has the same execution priority.

Problem & Motivation

- The newest version of OpenMP supports task priority, resulting in higher performance.
- For multi-node parallel processing systems such as HPX runtime system, there is no task priority control until now.

References and Acknowledgement


This work is partially supported by MEXT Next Generation High-Performance Computing Infrastructures and Applications R&D Program "R&D of A Quantum-Annealing-Assisted Next Generation HPC Infrastructure and its Applications," and Grant-in-Aid for Scientific Research(B) #17H01706.