TER Project - StarPU schedulers

Atte Torri - atte.torri@universite-paris-saclay.fr

Important notes about StarPU-MPI

- StarPU manages data transfers automatically, thus there is no need to `MPI_Send`
- Make sure to register MPI data handles in StarPU in all nodes that will need the data handle. (Nodes need to know where to find the data they need)
- Tasks must be submitted by all nodes that own some data handle used in a task (each node unrolls the task graph)
- Make sure the StarPU MPI cache stays consistent if pruning tasks manually

Using StarPU-MPI

- Initialize StarPU with `starpu_mpi_init_conf`
- Shutdown StarPU with `starpu_mpi_shutdown`
- Registering MPI data handles with `starpu_mpi_data_register`
- Inserting tasks with `starpu_mpi_task_insert`

Using StarPU-MPI

- Manage task execution node manually with `STARPU_EXECUTE_ON_NODE` or `STARPU_EXECUTE_ON_DATA`
- Debugging information about MPI with `STARPU_MPI_STATS` and `STARPU_MPI_CACHE_STATS`

2D Block-cyclic distribution

0	1	0	1	0	1	0	1
2	3	2	3	2	3	2	3
0	1	0	1	0	1	0	1
2	3	2	3	2	3	2	3
0	1	0	1	0	1	0	1
2	3	2	3	2	3	2	3
0	1	0	1	0	1	0	1
2	3	2	3	2	3	2	3

Figure: 2D Block-cyclic distribution

Using cluster with multiple nodes and MPI

- Add to `.ssh/config` the line
  `StrictHostKeyChecking false`
- Include module (ONLY WHEN COMPILING)
  `module load openmpi/4.1.5/gcc-12.3.0`
- Execute with MPI
  `mpirun -H node1,node2,... path/to/executable`
- You can check that MPI works by running
  `mpirun -H node1,node2,... hostname`

Slurm batch

Use Slurm batch to run the code on the cluster. This will allocate multiple machines for the job as requested. A batch file looks something like this

```
#!/usr/bin/env bash
# Sbatch settings
#SBATCH --partition cpu_tp
#SBATCH --exclusive
#SBATCH --qos 8nodespu
# Standard output
#SBATCH -o %x.out
# Standard error
#SBATCH -e %x.err

echo "========== Job Information =========="
echo "Node List: " $SLURM_NODELIST
echo "my jobID: " $SLURM_JOB_ID
echo "Partition: " $SLURM_JOB_PARTITION
echo "submit directory:" $SLURM_SUBMIT_DIR
echo "submit host:" $SLURM_SUBMIT_HOST
echo "In the directory:" $PWD
echo "As the user:" $USER
echo "========== Job Information =========="

nodelist=$(scontrol show hostname $SLURM_NODELIST)
printf "%s\n" "${nodelist[@]}" > nodefile
mpirun --hostfile nodefile -N 1 mpiBench/mpiBench
rm nodefile
```

Next

Continue working on the TER project. For next week read chapter 42 (MPI Support).

```
# Login to cluster
ssh qdcster_XX@chome.metz.supelec.fr
# Allocate a machine to work on
salloc --partition cpu_tp_resa --time 4:00:00
       --reservation M1QDCS_TERSTARPU16 --exclusive
# Allocate multiple machines to run code interactively
salloc --partition cpu_tp_resa --qos 8nodespu
       --reservation M1QDCS_TERSTARPU16 --nodes 4
       --exclusive --time 4:00:00
# Run code with sbatch non-interactively
sbatch --partition cpu_tp_resa --qos 8nodespu
       --reservation M1QDCS_TERSTARPU16 --nodes 4
       --exclusive --time 4:00:00
       --export=ALL batch.sl
```