TER Project - CUDA support

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

Adding CUDA support

- Initialising cublas
- Add StarPU kernels to support CUDA
- Using CUDA streams and asynchronous kernels to overlap communication and computation
- Note that the code should compile without CUDA when `ENABLE_CUDA=OFF` and with CUDA when `ENABLE_CUDA=ON`
- Enabling and disabling CUDA should be possible using a command line argument

Environment variables to control CUDA

- `STARPU_NWORKER_PER_CUDA` - How many workers are attributed to each device (i.e. CUDA streams per device)
- `STARPU_CUDA_THREAD_PER_WORKER` - If set to 1, StarPU will use one thread per CUDA worker as driver
- `STARPU_CUDA_PIPELINE` - Specify how many asynchronous tasks are submitted in advance on CUDA devices

Environment variables to control task limits

Use these environment variables if you have too many tasks in the task graph

- `STARPU_LIMIT_MAX_SUBMITTED_TASKS` - How many tasks can be at maximum in the task graph
- `STARPU_LIMIT_MIN_SUBMITTED_TASKS` - When to restart adding tasks to task graph

Next

Continue working on the TER project.

```
ssh qdcster_XX@chome.metz.supelec.fr
salloc --partition gpu_tp_resa --reservation M1QDCS_TERSTARPU14
       --exclusive --time 4:00:00
```

cuBLAS documentation
https://docs.nvidia.com/cuda/cublas/index.html

For next week read at least chapters 15, 34 of the StarPU documentation. Strong recommendation to read chapters 20, 21, 22 for an overview of some profiling methods.
https://github.com/TER-StarPU/ter-starpu-gemm/blob/main/docs/starpu-documentation.pdf