

Intel Thread Building Blocks Lab Session

SPD course 2012-13
Massimo Coppola
11/12/2012

Preparation

- Download and install TBB
- Download TBB sources to get full documentation and examples
- Set up your environment for compilation
 - e.g. on Mac OSX, with TBB installed in /opt/intel:
 - TBBROOT=/opt/intel/tbb
 - LIBRARY_PATH=/opt/intel/tbb/lib
 - CPATH=/opt/intel/tbb/include:/sw/include
 - DYLD_LIBRARY_PATH=/opt/intel/tbb/lib
- Compile the sub_string_finder example
- Copy and modify that makefile for your first program

Using Parallel for

- Write a trivial fibonacci function $f(n)$
- Sequential work: compute 100 times $f(n)$, with n rather large (e.g. 1000?)
- Parallel work: exploit threads for the computations on the $[1, 100)$ range
 - CPU bound: no cache/memory overhead
 - perfectly balanced
- Measure the speedup with respect to serial
- Modify code for unbalanced computation
 - compute $f(10*i)$, with $i = 0..100$
 - compute $f(\text{rand}()*1000)$
 - in previous cases, store the computed ranges in an array, print them after the execution and analyze them

- Check that your compiler properly supports lambdas
- Installing TBB from sources and binary do not result in the same configuration
 - environment variables (paths, options) affect compilation
 - identify proper switches in compilation
- `tbbvars.sh` can set proper variables for you (but it is buggy in some TBB versions)
- Makefiles to compile examples tend to work reliably, but they rebuild their configuration each time