

Lua functionality and threading/vectorization

Advanced Elmer/Ice workshop at @ CSC

Lua

- ▶ How to enable lua?
 - ▶ cmake variable WITH_LUA
 - ▶ Compiles bundled Lua (5.1.5) under contrib/lua-5.1.5
- ▶ Use custom lua installation by setting following cmake variables. If 2 and 3 are not set, the FindLua51 script will find something.
 1. USE_SYSTEM_LUA=TRUE
 2. LUA_LIBRARIES=<lua_library.a>
 3. LUA_INCLUDE_DIR=<lua_include_dir>
- ▶ Works with LuaJIT too!
- ▶ Walk through testcase Lua:
 - ▶ Try switching field source lua ↔ matc
- ▶ Walk through testcase KeywordUnitTestLua

Lua contd

- ▶ Why not enabled by default?
 - ▶ Doesn't compile on old compilers (eg. gcc 4.8.5)
 - ▶ There might still be some bugs which are not visible with tested compilers
- ▶ Tested on
 - ▶ Intel 17.0.4 compiler stack at least
 - ▶ gcc 5 (works probably with ≥ 5)

ModelPDEthreaded

- ▶ Look at talk.pdf
 - ▶ Threading-coloring picture (page 17)
 - ▶ Vectorized assembly (page 9-14)
- ▶ Walk through `./ModelPDEthreaded/case.sif` (Especially kws related to threading and coloring)
- ▶ Walk through `./ModelPDEthreaded/ModelPDEThreaded.F90` (next slide)

ModelPDEthreaded.F90 (AdvDiffSolver)

- ▶ Inside AdvDiffSolver subroutine (solver main level):
 - ▶ Get number of colors (45)
 - ▶ Deduce VecAsm parameter to be passed to DefaultUpdateEquations (46)
 - ▶ Create OpenMP parallel region, be careful with shared/private data (50)
 - ▶ Loop over colors (51)
 - ▶ Remember to get number of active elements using single thread. This will change active color too. (59)
 - ▶ There is an implicit barrier at `omp end single` (60)
 - ▶ Loop over elements, use OpenmP worksharing constructs (here `!$OMP DO`), call LocalMatrixVec (63)

ModelPDEthreaded.F90 (LocalMatrixVec)

- ▶ Inside LocalMatrixVec:
 - ▶ Remember to declare SAVED variables to be !\$OMP THREADPRIVATE (152)
 - ▶ Use GetElementNodesVec instead of GetElementNodes. It will set Nodes & xyz field as well (179)
 - ▶ Collect all material and RHS data to one contiguous and aligned memory area in Coeffs and set up pointers load, diff_coeff, ... for easier access. (182)
 - ▶ Material/rhs data in integration points is saved in single contiguous and aligned CoeffsInBasis array and pointers are set up for easier access (LoadAtIP, ...). (189)
 - ▶ Evaluate all basis functions in all integration points using ElementInfoVec instead of ElementInfo (236)
 - ▶ Project nodal data to integration points (241)

More on next slide

ModelPDEthreaded.F90 (LocalMatrixVec) (contd.)

- ▶ Inside LocalMatrixVec:
 - ▶ Collect PDE operators associated to this given ModelPDE (261)
 - ▶ Example: Laplacian with isotropic non-homogeneous diffusion (262)
 - ▶ Example. Force term (275)
 - ▶ VecAsm is passed to DefaultUpdateEquations here. If VecAsm is true, then use optimized CRS glueing process which is slightly faster on ordinary processors and much faster on xeon phi. Does not support normal-tangential BCs (313)