Performance optimization for modern many-core architectures using PSyclone embedded-DSL.

Sergi Siso (sergi.siso@stfc.ac.uk)
High-Performance Software Engineer
IPCC @ Hartree, STFC Daresbury Laboratory, United Kingdom
Work done with: Rupert Ford, Andrew Porter, Christopher Maynard
LFRic project

- Met Office (UK) project to develop a replacement for the Unified Model.
- Developed in a collaboration between MetOffice and STFC (Hartree Centre)
- Named in honour of Lewis Fry Richardson (first numerical weather ‘prediction’)

Gungho project

- MetOffice, NERC and STFC initial investigation and design for next-generation model. They produced a set or recommendations for LFRic.
- New dynamical core more scalable and without the singularity on the poles:
  - Mixed finite-element scheme with a cubed-sphere mesh.
- Good software abstraction and separation of concerns.
- Achieve good performance on current and future supercomputers.
Separation of concerns

**Natural Science Knowledge**
- Finite element/volume/difference-specific
- Time-stepping
- Operations over a mesh
- Typically same operation at each element/volume/point

**Computational Science Knowledge**
- Parallelism
- Data decomposition
- Data locality
- Architecture/platform dependent optimizations

---

Domain Scientists

HPC Specialists
Given domain-specific knowledge and information about the Algorithm and Kernels, the Parallel System layer can be generated by ...
• Explore the performance characteristics of LFRic / PSyclone on many-core architectures (Intel Xeon Phi x200)
• Feedback the obtained knowledge to PSyclone and LFRic projects
Matrix Vector Multiplication Kernel
(hotspot and representative kernel)

- LFRic uses multiple FEM fields:

- Operations are encoded in LMA (local matrix assembly) and are designed to transform from one field to another.

\[
A \mathbf{x} = \begin{bmatrix}
    a_{11} & a_{12} & \cdots & a_{1n} \\
    a_{21} & a_{22} & \cdots & a_{2n} \\
    \vdots & \vdots & \ddots & \vdots \\
    a_{m1} & a_{m2} & \cdots & a_{mn}
\end{bmatrix}
\begin{bmatrix}
    x_1 \\
    x_2 \\
    \vdots \\
    x_n
\end{bmatrix} =
\begin{bmatrix}
    a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n \\
    a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n \\
    \vdots \\
    a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n
\end{bmatrix}
\]

- Very Small matrix-vector multiplication – difficult to optimise...
- but we have many to do (horizontal and extruded dims) and they share data
1. **Original**: LFRic current implementation. VTune pinpoints L2 Misses and SIMD-to-L2 ratio issues.

2. **Fused & Interleaved**: Fuse gathers, computation and scatters, do each FMA interleaved with next mv.

3. **Loop K Inner**: Refactor loops to go through vertical domain first.

4. **DL K Contiguous**: Refactor data layout.

5. **Merge Inner Dofs**: Update top dof of i layer at the same time as the bottom dofs i+1 layer.

*(blas and libxsmm couldn’t improve the performance of manual implementation)*
OpenMP Implementations

gungho-mv ‘DL K Contiguous’
implementation

Problem size: 32x32 (x16)
OMP Schedule: Static
OMP Affinity: Scattered
KNL: cache-quadrant

Thread oversubscribing helps across all implementations.
Colouring beats Locking for more than 10 threads.
Skylake vs KNL vs KNL with MCDRAM

Best Performance
Vector Efficiency is ~ x2
Skylake and KNL Rooflines

- Full node mv-kernel for:
  - Skylake is at the DDR STREAM Triad boundary.
  - KNL is at 76% of MCDRAM STREAM Triad boundary.
- Next step could be kernel-fusing:
Future work

- OpenMP strategies, loop fusions, halo exchange optimisations, ... can be extended to the whole code with PSyclone.

- However, changing the data layout and refactoring the loops is a manual process at the moment.
  - IPCC recommends that PSyclone explores mechanisms to abstract the data layout (and/or loop structure).
  - This is essential for achieving optimal performance.

- LFRic is still in development: column matrix assembly, multi-grid and other features are coming.
Thanks for your attention

Sergi Siso: sergi.siso@stfc.ac.uk

http://www.hartree.stfc.ac.uk