DE LA RECHERCHE À L'INDUSTRIE

First experiments with CEA Arm cluster

2019/06/20 Arm HPC Users Group

Guillaume Colin de Verdière guillaume.colin-de-verdiere@cea.fr

- **Atos BullSequana X1310**
  - *A Mont-Blanc 3 design*

MONT-BLANC

- 292 blades
  - 2 sockets
  - Marvell ThunderX2
  - 32 cores, 2.2 GHz
  - 8 channels, DDR4 2666, 256 GB
  - Infiniband EDR

- Peak performance: 329 TFlops

Atos

MARVELL

A WELL BALANCED MACHINE

- HPL
  - 256 nodes, 60% of RAM
  - 242.4 Tflops (peak = 288.4 Tflops)
  - 84% efficiency
- HPCG
  - 256 nodes
  - 8.422 Tflops
  - **3.47 % of HPL**
- Graph500

11/2018 HPCG List

#	Name	HPL	HPCG	%HPL
1	Summit	143500	2925	2,04
2	Sierra	94640	1795	1,90
3	K computer	10510	602	5,73
4	Trinity	20158	546	2,71
5	ABCI	19880	508	2,55

	ThunderX2 32 cores 2.0 Ghz			Skylake Gold 6148 20 cores 2.4 Ghz			Thx2 / Skl
NODE	SCALE	EDGE	score	SCALE	EDGE	SCORE	performance
1	24	16	1.05E+10	24	16	7.91E+09	+33%
2	26	16	1.89E+10	26	16	1.42E+10	+33%
4	27	16	3.77E+10	27	16	2.84E+10	+33%

Porting « real codes » to the machine

OUTLINE

- Porting applications on Arm systems
  - Compilation
  - Status

- First experiments on real applications

COMPILATION ON ARM SYSTEMS

- Compilers
  - Arm 19.0.0 based on CLANG/LLVM
  - GCC 7.3 and 8.2
  - CLANG/LLVM 5.0.1
  - Not available: PGI compilers
- “Just” modify compilation scripts
  - No big modifications
  - Equivalent to a new x86 architecture
- Prototype new methodology for application distribution
  - Using **Spack** for Open Source applications on Arm
    - Linear algebra
    - Mesh partitioning
    - FFTW
  - https://spack.io/

CURRENT STATUS (1/2)

- Proprietary software
  - NAG
  - BLAS
    - Arm PL: few issues at compilation time (PETSc)
    - OpenBLAS: OK
  - Debugger
    - DDT: OK
    - Totalview: not tested yet
  - Profiler
    - Arm Forge: not tested yet
    - Score-P and TAU: not tested yet
  - Optimization tools
    - Arm tools: not tested yet
- Missing tools
  - IMSL: not available on Arm architectures
  - Intel Thread Inspector equivalent
  - CUDA/OpenACC compiler

CURRENT STATUS (2/2)

- OpenSource software
  - Intel TBB
  - FFTW (Spack)
  - Parmetis/Metis (Spack), Scotch (Spack)
  - Hypre (Spack), Trilinos (Spack)
  - PETSc (Spack)

- CEA software
  - Alien (Spack) linear algebra
  - Arcane code dev framework
  - Hercule scientific I/O lib
  - …
  - Issues
    - Floating Point optimizations (runtime failure)
    - OpenMP Tasks (compile time + runtime failure)
  - Still Not-ported tools (pre- and post- processing)

PORTING APPLICATIONS WITH ASSEMBLY CODE

- MPC
  - Thread-based MPI and OpenMPI runtime
    - Official MPI implementation listed in MPI Forum
    - Open source available at http://mpc.hpcframework.com
  - Current status
    - MPI and OpenMP runtimes OK
    - No specific optimizations
  - Compiler support
    - Compiler automatic privatization
    - Linker TLS optimization
- WI4MPI
  - MPI abstraction layer
    - Open source available at http://github.com/cea-hpc/wi4mpi/
  - Assembly code ported (with Arm help for TLS specification)
  - To be tested on real applications

FIRST EXPERIMENTS

- Arm Speed-up on multiple test-cases
  - Architectures
    - HSW : 2-socket 16-core Intel Xeon Haswell
    - KNL : 1-socket 64-core Intel Xeon Phi Knight's Landing
    - SKL : 2-socket 24-core Intel Xeon Skylake
    - Baseline : 2-socket 32-core ThunderX2
  - Compilers: Intel 18.0.3 (Intel architectures) & GCC 7.3 (Arm architectures)
  - Hybrid MPI/OpenMP evaluation
    - 8 MPI ranks on each node
    - OpenMP threads to fill nodes

Test Case	THX2	HSW	KNL	SKL
Hydro (1 node)	1	1.7	1.7	1
Hydro (4 nodes)	1	1.3	2.3	0.8
CG (1 node)	1	1.3	0.9	0.6
	Reference	× THX2 time		

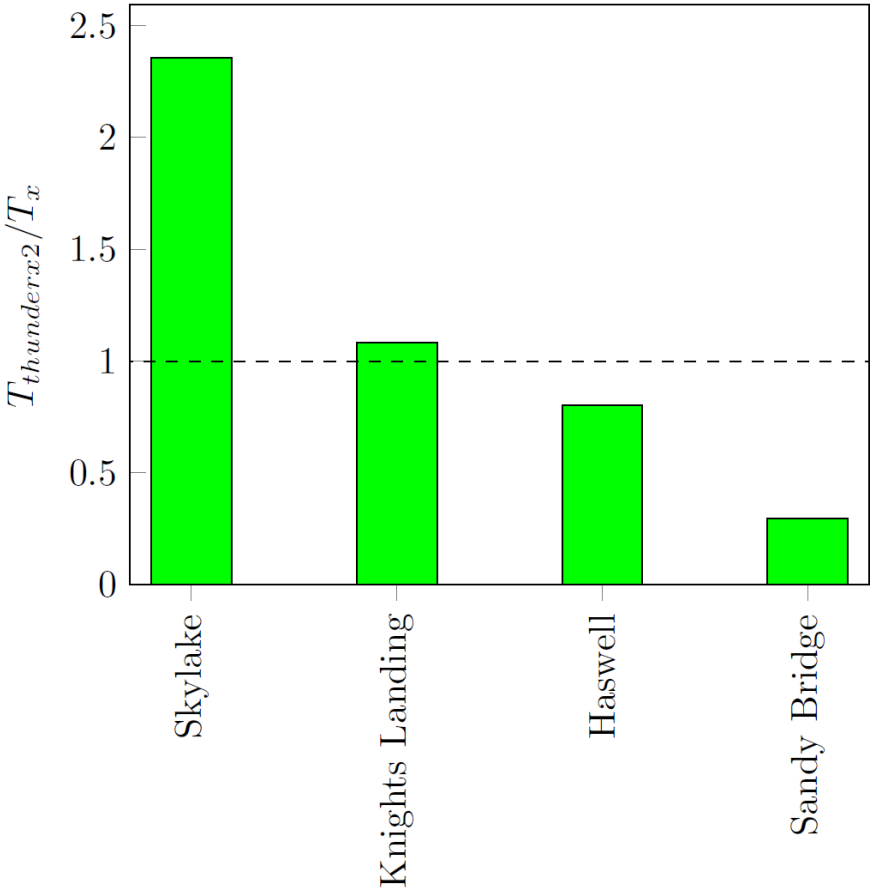
- Arm execution time hydro benchmark
  - Architectures
    - HSW : 2-socket 16-core Intel Xeon Haswell
    - Baseline : 2-socket 32-core ThunderX2
  - Compilers: GCC 4.9 (Intel architectures) & GCC 7.3 (Arm architectures)
  - Test case
    - 125,000 cells weak scaling
    - Simplified Hydro
    - 150 steps
    - No vectorization

Test Case	SEQ	TBB 8	TBB 16	TBB 32	MPI 8	MPI 16	MPI 32
THX2	9.22	10.56	12.10	14.49	11.73	12.01	13.93
HSW	6.41	11.60	11.92	12.67	11.28	12.27	13.10
	Execution time. Lower is better						

- Arm execution benchmark
  - Architectures
    - HSW : 2-socket 16-core Intel Xeon Haswell
    - Baseline : 2-socket 32-core ThunderX2
  - Compilers: GCC 4.9 (Intel architectures) & GCC 7.3 (Arm architectures)
  - Multiple test cases
  - 2 times more MPI ranks on Arm system (high pressure on application scalability)

Test Case	Case 1: 2 nodes	Case 1: 4 nodes	Case 2: 2 nodes	Case 2: 4 nodes	Case 3: 2 nodes	Case 3: 4 nodes
THX2	78,895	63,294	83,407	74,947	25,667	33,617
HSW	87,913	58,579	85,592	57,259	25,146	21,880
	Execution time. Lower is better					

- Execution time comparison
- Highly optimized code
  - High vectorization %
  - Multithreaded
- NEON << AVX512



CONCLUSION/FUTURE WORK

- Conclusion
  - Easy to port from x86 world
  - Promising early results for a first HPC silicon from Marvell
  - Some issues
    - FP optimization/management
    - OpenMP Tasks
  - Some tools are missing
- Future work
  - Build an ecosystem to have the same experience on Arm & x86
  - More in-depth evaluations

THANKS