OpenFOAM – Knights Landing 장비 병렬테스트

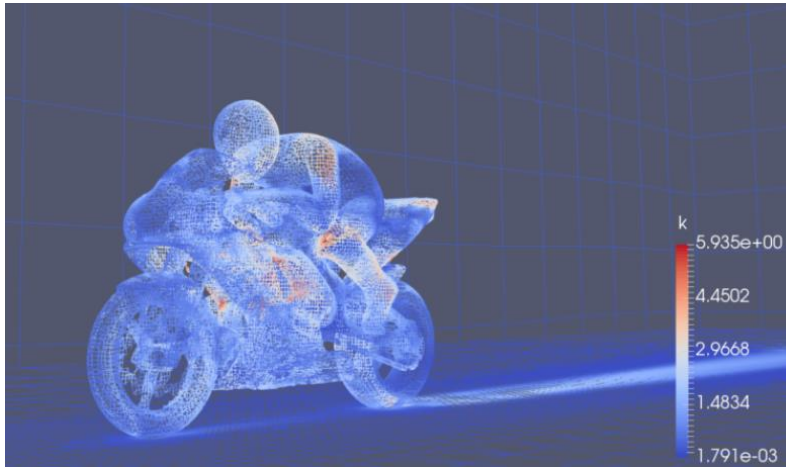
2018. 11. 01

NEXTfoam
open source CFD consulting

OpenFOAM® 성능최적화

- OpenFOAM® 설치 및 성능 최적화
  - Knights Landing architecture 프로세서
    - 컴파일러 옵션을 적용하여 OpenFOAM® 설치
  - KNL하에서 개선된 성능
    - Intel-optimized smoother을 적용한 매트릭스 solver 적용
  - 해석 예제를 통한 병렬 성능 검증 수행

- 검증 결과
  - KNL architecture에 특화된 smoother을 적용한 매트릭스 solver 실행 옵션
  - 선정된 해석 예제의 병렬 성능 결과 곡선

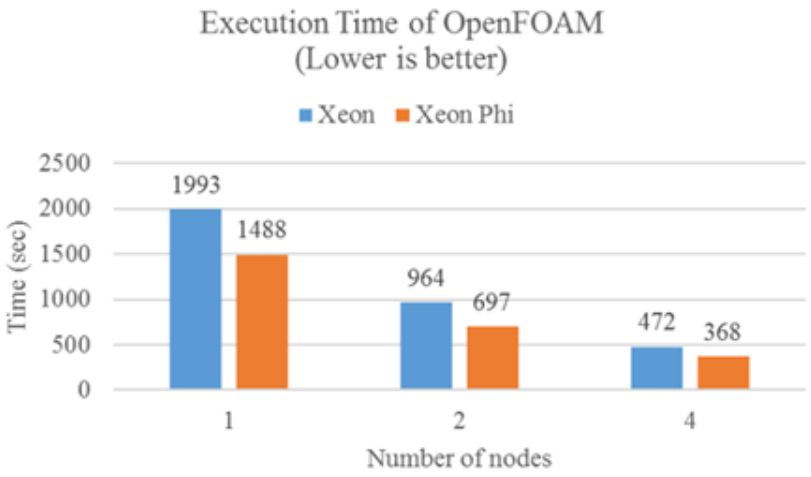


OpenFOAM® 성능최적화

Intel-optimized smoother를 적용한 매트릭스 solver 개발

– OpenFOAM® 매트릭스 solver에 intel-optimized smoother 적용

- 기존의 OpenFOAM® 매트릭스 solver의 소스 코드는 기존 Xeon 프로세서에 맞추어 개발되어 있으므로 high-bandwidth 메모리를 사용할 수 있도록 소스 코드 변경 필요
- 기존의 OpenFOAM® 내의 매트릭스 solver에서 사용되는 GaussSeidel이나 symGaussSeidel preconditioner smoother의 vectorization을 KNL의 high-bandwidth 메모리를 사용할 수 있도록 수정하여 KNL 구조의 프로세서 하에서 OpenFOAM® 의 성능 상승을 유도



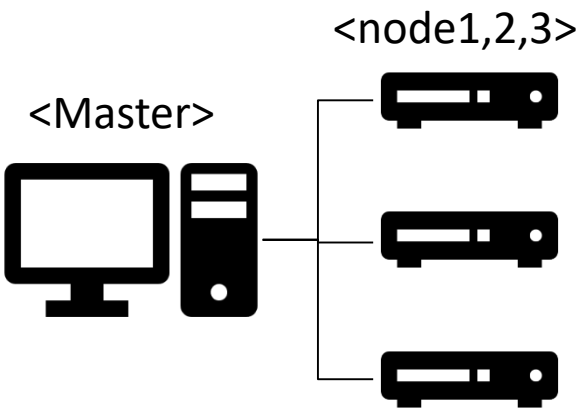
[intel-optimized smoother로 인한 계산 성능 향상 예상]

– 해석 예제를 통한 병렬 성능 검증

- 각 솔버에 특화되어 개발된 예제를 통해 병렬 성능 검증
- 컴파일러에 따른 build 상황과 MPI 사용방법에 따른 병렬 성능 검증
- 매트릭스 수정 및 솔버 개선에 따른 병렬 성능 검증

테스트 하드웨어 사양

- KNL 장비 사양
  - 4 node (one is master node, three nodes are computing nodes)
  - CPU
    - Intel(R) Xeon Phi(TM) CPU 7250 @ 1.4GHz
    - 68 cores / node
  - Memory
    - 115GB / node
  - Switch
    - Infiniband: OmniPath
  - OS
    - CentOS Linux release 7.3.1611 (core)



테스트 프로그램 사양

- C Compiler
  - gcc (GCC) 6.3.1 20170216 (Red Hat 6.3.1-3)
  - parallel_studio_xe_2018_update3_professional_edition
- MPI
  - OpenMPI 2.1.1 (Thrid party-5.x)
  - Intel(R) MPI Library 2018 update 3.222
- Building Version
  - Available OpenFOAM ver.
    - Greater equal 4.1
    - OpenFOAM 5.x
  - OpenFOAM-5.x/wmake/rules/
  - linux64Gcc, linux64GccKNL, linux64Icc, linux64IccKNL
  - MPI
    - OpenMPI
    - IntelMPI

OpenFOAM Build

- GNU C Compile Options

Compiler	Suggested flags
gcc compiler	-march=knl -O3 -mavx512f -mavx512pf -mavx512er - mavx512cd -mfma -malign-data=cacheline -finline-functions
g++ compiler	-std=c11 -march=knl -O3 -mavx512f -mavx512pf -mavx512er -mavx512cd -mfma -malign-data=cacheline -finline-functions
gfortran compiler	-O3 -march=knl -mavx512f -mavx512pf -mavx512er -mavx512cd -mfma -malign-data=cacheline -finline-functions

-O3	Optimize even more. The compiler will also do loop unrolling and function inlining. **RECOMMENDED**.
-mfma, -mavx512f, -mavx512pf, -mavx512er, -mavx512cd	These switches enable the use of instructions in the MMX, SSE, SSE2, SSE3, SSSE3, SSE4.1, AVX, AVX2, AVX512F, AVX512PF, AVX512ER, AVX512CD, SHA, AES, PCLMUL, FSGSBASE, RDRND, F16C, FMA, SSE4A, FMA4, XOP, LWP, ABM, AVX512VL, AVX512BW, AVX512DQ, AVX512IFMA, AVX512VBMI, BMI, BMI2, VAES, WAITPKG, FXSR, XSAVE, XSAVEOPT, LZCNT, RTM, MWAITX, PKU, IBT, SHSTK, AVX512VBMI2, GFNI, VPCLMULQDQ, AVX512BITALG, MOVDIRI, MOVDIR64B, AVX512VPOPCNTDQ, CLDEMOTE, 3DNow! or enhanced 3DNow! extended instruction sets. Each has a corresponding -mno- option to disable use of these instructions. GCC depresses SSEx instructions when **-mavx** is used. Instead, it generates new AVX instructions or AVX equivalence for all SSEx instructions when needed.
-malign-data=type	Control how GCC aligns variables. Supported values for *type* are 'compat' uses increased alignment value compatible uses GCC 4.8 and earlier, 'abi' uses alignment value as specified by the psABI, and 'cacheline' uses increased alignment value to match the cache line size. 'compat' is the default.
-f[no-]inline-functions	Consider all functions for inlining, even if they are not declared inline. The compiler heuristically decides which functions are worth integrating in this way. Enabled at levels -O2, -O3, -Os.

OpenFOAM Build

- Intel C Compile Options

Compiler	Suggested flags
Intel C compiler	-O3 -xMIC-AVX512 -fma -align -finline-functions
Intel C++ compiler	-std=c11 -O3 -xMIC-AVX512 -fma -align -finline-functions
Intel Fortran compiler	-O3 -xMIC-AVX512 -fma -align array64byte -finline-functions

-O3	Enable –O2 plus more aggressive optimizations that may or may not improve performance for all programs.
-xMIC-AVX512	Enable advanced vector instructions set 512 (for future **KNL/SkyLake based systems**)
-fma	Determines whether the compiler generates fused multiply-add (FMA) instructions if such instructions exist on the target processor. This option determines whether the compiler generates fused multiply-add (FMA) instructions if such instructions exist on the target processor. When the [Q]fma option is specified, the compiler may generate FMA instructions for combining multiply and add operations. When the negative form of the [Q]fma option is specified, the compiler must generate separate multiply and add instructions with intermediate rounding. This option has no effect unless setting CORE-AVX2 or higher is specified for option [Q]x, -march (Linux and macOS*), or /arch (Windows).
-align	Analyze and reorder memory layout for variables and Arrays.
-f[no-]inline-functions	Inline [do not inline] certain interprocedural optimizations for single file compilation. These optimizations are a subset of full intra-file interprocedural optimizations. Enables [does not enable] the compiler to perform inline function expansion for calls to functions defined within the current source file

OpenFOAM-Intel

- **https://github.com/OpenFOAM/OpenFOAM-Intel**

- **libhbm**: A wrapper library for handling high bandwidth memory when running in flat memory mode on the KNL.
  - mpirun -np 68 -env LD_PRELOAD /path/to/libhbm.so -env HBM_SIZE 100 -env HBM_THRESHOLD 16 -env MPI_BUFFER_SIZE 1000000 ./myapplication

This library has been written specifically for running OpenFOAM in "Flat" memory mode. The "Flat" memory mode provides a separate NUMA domain for the MCDRAM allowing the user to decide where to allocate the memory. This library provides similar functionality to autohbw in memkind (https://github.com/memkind/memkind). The main difference with libhbm is that the MCDRAM is used to create a fixed size heap for a process. This is allocated at the start and never freed until the application terminates. A threshold decides whether to try and allocate to the heap in MCDRAM (and reverts back to DDR memory if this fails). Both the heap size and threshold are configurable though environment variables.

OpenFOAM-Intel

- **Optimized smoothers**: Updated versions that are optimized for the KNL.
  - The additional library will need to be included in the system/controlDict. Here, libKNL.so needs to be appended to the libs list. Next, look in system/fvSolution and replace any occurances of *GaussSeidel* with *GaussSeidelKNL*, and *symGaussSeidel* with *symGaussSeidelKNL*. Finally, run as usual.

This contains two new versions of the GaussSeidel and symGaussSeidel smoother which have a better performance on the KNL. The optimisations include swapping a divide in the inner loop for a multication by a pre-calculated reciprocal and creating a separate code path for cells with a different number of faces. These new smoothers can be used instead of the existing ones by compiling and then post-fixing the smoother with “KNL” in the configuration files.

Parallel 사용 설정

- Intel MPI
  - 환경변수설정
    - export PSM2_IDENTIFY=1
    - export I_MPI_FABRICS=shm:tmi
    - export I_MPI_TMI_PROVIDER=psm2
    - export I_MPI_FALLBACK=0
    - export I_MPI_SHM_LMT=shm
    - export HFI_NO_CPUAFFINITY=1
  - 실행명령

    # mpirun -machinefile hosts.txt -psm2 -np 64 simpleFoam –parallel
  - libhbm 설정

    # mpirun -machinefile hosts.txt -psm2 -np 64 -env LD_PRELOAD /nfs/OpenFOAM-Intel/libhbm/libhbm.so -env HBM_SIZE 100 -env HBM_THRESHOLD 16 -env MPI_BUFFER_SIZE 1000000 simpleFoam -parallel

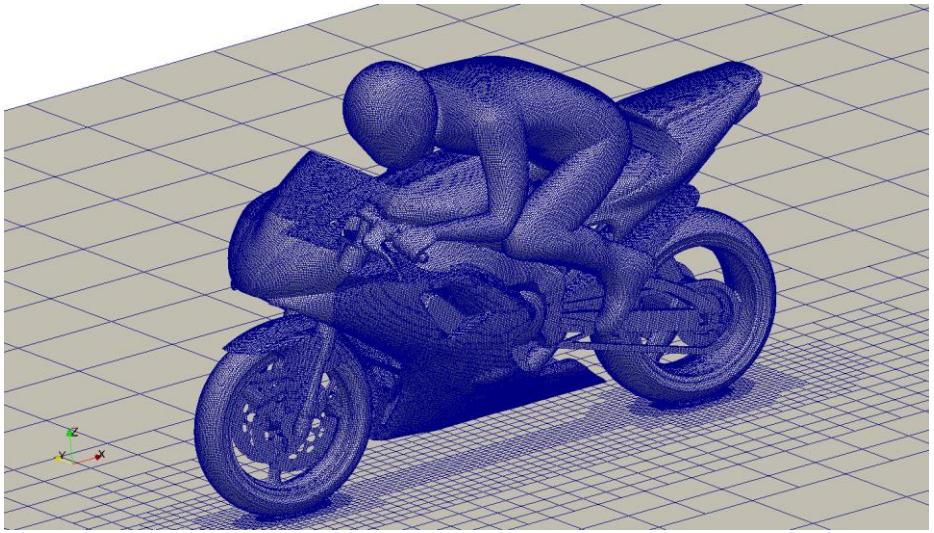
Parallel 사용 설정

- OpenMPI
  - OpenMPI를 사용 할 경우에는 '-mca pml cm -mca mtl psm2' 명령행 옵션을 추가로 사용
  - 실행명령

```
# mpirun --mca pml cm --mca mtl psm2 -np 64 -hostfile hosts simpleFoam -parallel
```

Parallel Performance

- OpenFOAM Tutorial
  - MotorBike
  - Mesh #
    - Medium case: 35만개
    - Fine case: 280만개
  - Test core #
    - Medium case
      - 1, 4, 8, 16, 32, 64, 128
    - Fine case
      - 1, 4, 8, 16, 32, 64, 128
  - OpenFOAM
    - Version: OpenFOAM-5.x
    - Solver: simpleFoam
    - C Compiler
      - GNU
      - Intel Compiler
    - MPI
      - OpenMPI-2.1.1
      - IntelMPI(20180411)



Parallel Performance

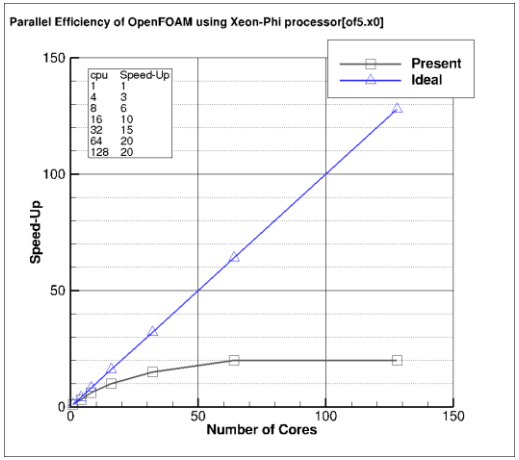
- MotorBike-Medium Case

Execution Time (sec.)

Number of Cores	OF5.x0	OF5.x0-smoothersKNL
1	55.57	52.50
4	16.59	15.82
8	9.50	9.09
16	5.62	5.42
32	3.64	3.58
64	2.82	2.79
128	2.73	2.73

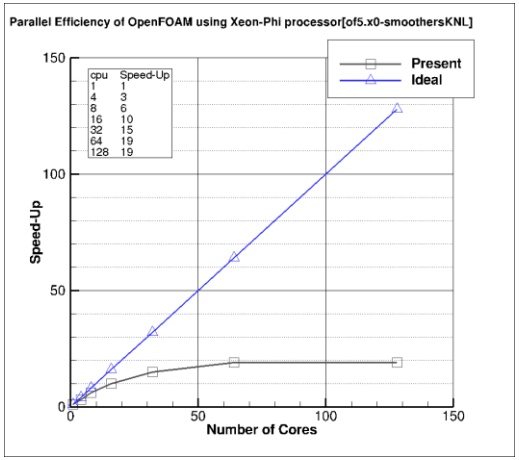
OF5.x0

Compiler	Gcc-6.3.1
Options	-
MPI	OPENMPI
Etc.	-



OF5.x0-smoothersKNL

Compiler	Gcc-6.3.1
Options	-
MPI	OPENMPI
Etc.	smoothersKNL



Parallel Performance

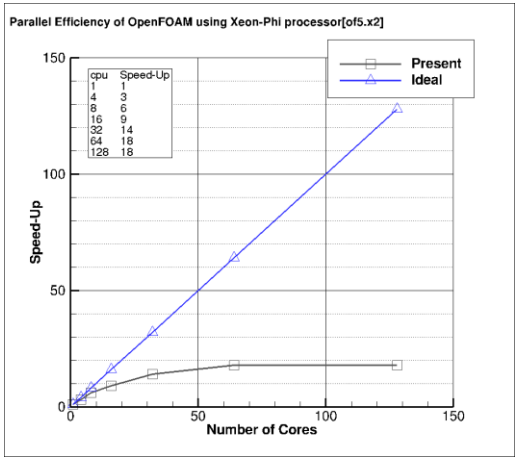
- MotorBike-Medium Case

Execution Time (sec.)

Number of Cores	OF5.x2	OF5.x2-smoothersKNL
1	49.12	46.48
4	14.88	14.17
8	8.61	8.16
16	5.20	5.01
32	3.43	3.33
64	2.72	2.68
128	2.67	2.67

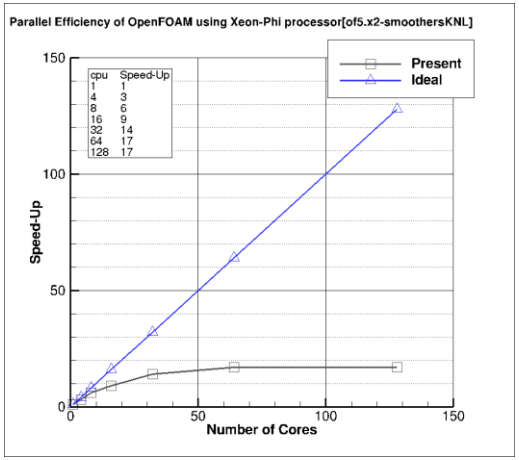
OF5.x2

Compiler	Gcc-6.3.1
Options	GccKNL
MPI	OPENMPI
Etc.	-



OF5.x2-smoothersKNL

Compiler	Gcc-6.3.1
Options	GccKNL
MPI	OPENMPI
Etc.	smoothersKNL



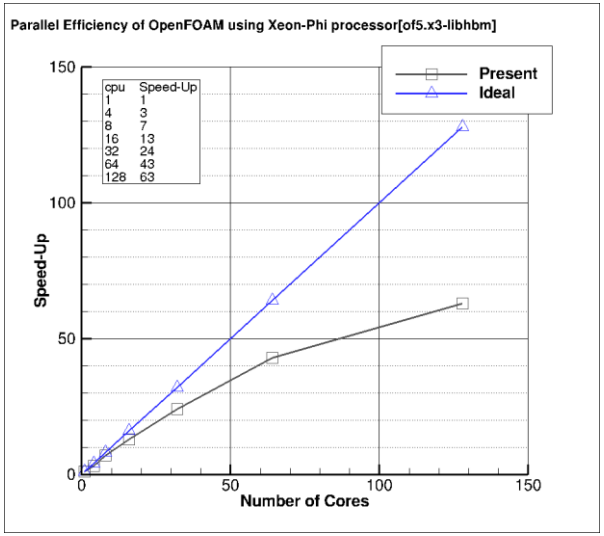
Parallel Performance

- MotorBike-Fine Case

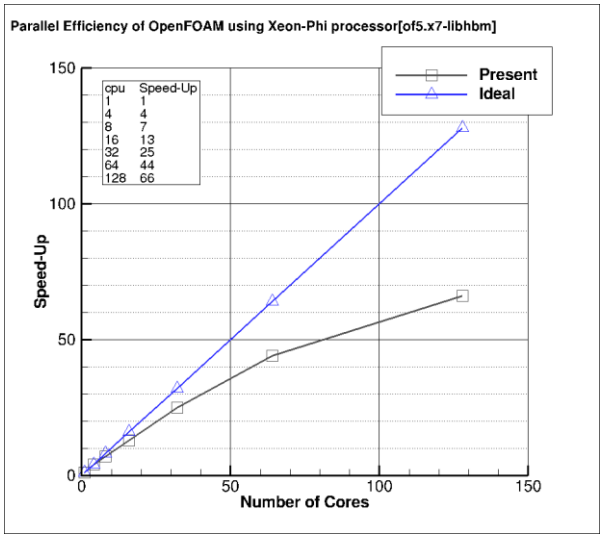
Execution Time (sec.)

Number of Cores	OF5.x3-libhbm	OF5.x7-libhbm
1	419.06	452.64
4	119.95	128.86
8	61.24	66.46
16	31.82	34.52
32	17.25	18.30
64	9.83	10.25
128	6.62	6.84

OF5.x3-libhbm	
Compiler	Gcc-6.3.1
Options	GccKNL
MPI	INTELMPI
Etc.	libhbm



OF5.x7-libhbm	
Compiler	Icc
Options	IccKNL
MPI	INTELMPI
Etc.	libhbm



Parallel Performance

- MotorBike-Fine Case

Execution Time (sec.)

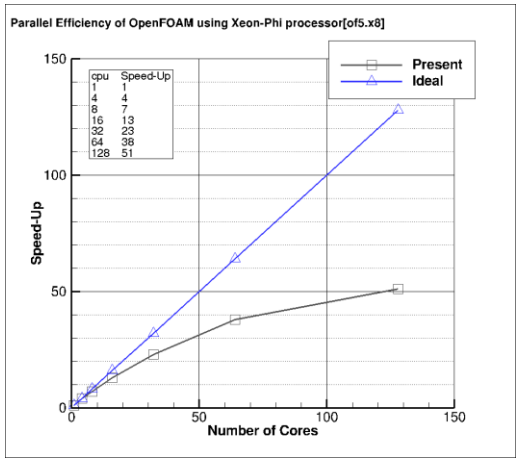
Number of Cores	OF5.x8	OF5.x8-smoothersKNL
1	420.44	391.90
4	117.54	109.94
8	59.80	57.87
16	33.34	31.20
32	18.28	17.16
64	11.06	10.76
128	8.28	8.13

(1)	gcc compiler	-march=knl -O3 -mavx512f -mavx512pf -mavx512er -mavx512cd -mfma –malign -ata=cacheline -finline-functions
	g++ compiler	-std=c11 -march=knl -O3 -mavx512f -mavx512pf -mavx512er -mavx512cd -mfma -malign-data=cacheline -finline-functions

(2)	Intel C compiler	-O3 -xMIC-AVX512 -fma -align -finline-functions
	Intel C++ compiler	-std=c11 -O3 -xMIC-AVX512 -fma -align -finline-functions

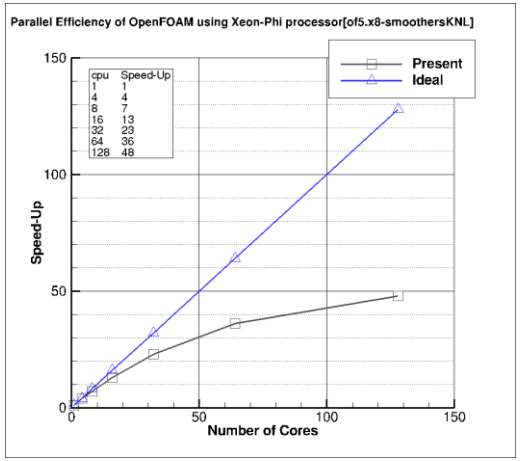
OF5.x8

Compiler	Gcc-6.3.1
Options	GccKNL/ $flags^{(1)}$
MPI	OPENMPI
Etc.	-



OF5.x8-smoothersKNL

Compiler	Gcc-6.3.1
Options	GccKNL/ $flags^{(1)}$
MPI	OPENMPI
Etc.	smoothersKNL



Parallel Performance

- MotorBike-Fine Case

Execution Time (sec.)

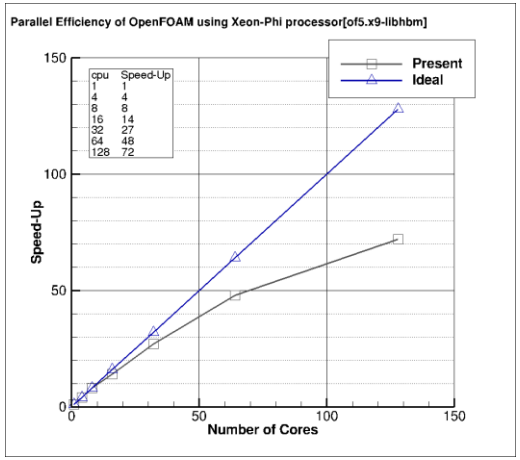
Number of Cores	OF5.x9-libhbm	OF5.x10-libhbm
1	497.58	420.27
4	129.17	119.02
8	66.04	62.01
16	34.49	31.76
32	18.32	17.26
64	10.28	9.82
128	6.90	6.61

(1)	gcc compiler	-march=knl -O3 -mavx512f -mavx512pf -mavx512er -mavx512cd -mfma –malign -ata=cacheline -finline-functions
	g++ compiler	-std=c11 -march=knl -O3 -mavx512f -mavx512pf -mavx512er -mavx512cd -mfma -malign-data=cacheline -finline-functions

(2)	Intel C compiler	-O3 -xMIC-AVX512 -fma -align -finline-functions
	Intel C++ compiler	-std=c11 -O3 -xMIC-AVX512 -fma -align -finline-functions

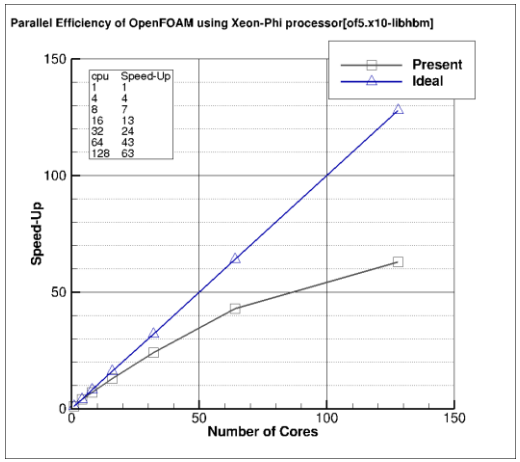
OF5.x9-libhbm

Compiler	Icc
Options	IccKNL/ $flags^{(2)}$
MPI	INTELMPI
Etc.	libhbm



OF5.x10-libhbm

Compiler	Gcc-6.3.1
Options	GccKNL/ $flags^{(1)}$
MPI	INTELMPI
Etc.	libhbm



Parallel Performance

- Best Core Execution Time

	OF5.x8-smoothersKNL	OF5.x9-libhbm
Compiler	Gcc-6.3.1/KNL	Intel-18.0.3(20180410)/ KNL
Compiler Options	Gcc: -march=knl -O3 -mavx512f -mavx512pf -mavx512er -mavx512cd -mfma -malign-data=cacheline -finline-functions	Intel C compiler: -O3 -xMIC-AVX512 -fma -align -finline-functions
	G++: -std=c11 -march=knl -O3 -mavx512f -mavx512pf -mavx512er -mavx512cd -mfma -malign-data=cacheline -finline-functions	Intel C++ compiler: -std=c11 -O3 -xMIC-AVX512 -fma -align -finline-functions
Number of core	1	1
OpenFOAM-version	OpenFOAM-5.x	OpenFOAM-5.x
Execution time	**391.90**	**497.58**
Etc.	smoothersKNL	libhbm

Parallel Performance

- Best Parallel Speed-Up

	OF5.x9-libhbm		OF5.x8-smoothersKNL	
Compiler	Intel-18.0.3(20180410)/KNL		Gcc-6.3.1/KNL	
Compiler Options	Intel C compiler: -O3 -xMIC-AVX512 -fma -align -finline-functions		Gcc: -march=knl -O3 -mavx512f -mavx512pf -mavx512er -mavx512cd -mfma -malign-data=cacheline -finline-functions	
	Intel C++ compiler: -std=c11 -O3 -xMIC-AVX512 -fma -align -finline-functions		G++: -std=c11 -march=knl -O3 -mavx512f -mavx512pf -mavx512er -mavx512cd -mfma -malign-data=cacheline -finline-functions	
MPI	INTELMPI		OPENMPI	
OpenFOAM-version	OpenFOAM-5.x		OpenFOAM-5.x	
Decompose Method	scotch		scotch	
Number of core	1	128	1	128
Execution time	**497.58**	**6.90**	**391.90**	**8.13**
Speed-Up	72		48	

Parallel Performance

- Best Parallel Execution Time

	OF5.x10-libhbm		OF5.x8	
Compiler	Gcc-6.3.1/KNL		Gcc-6.3.1/KNL	
Compiler Options	Gcc: -march=knl -O3 -mavx512f -mavx512pf -mavx512er -mavx512cd -mfma -malign-data=cacheline -finline-functions		Gcc: -march=knl -O3 -mavx512f -mavx512pf -mavx512er -mavx512cd -mfma -malign-data=cacheline -finline-functions	
	G++: -std=c11 -march=knl -O3 -mavx512f -mavx512pf -mavx512er -mavx512cd -mfma -malign-data=cacheline -finline-functions		G++: -std=c11 -march=knl -O3 -mavx512f -mavx512pf -mavx512er -mavx512cd -mfma -malign-data=cacheline -finline-functions	
MPI	INTELMPI		OPENMPI	
OpenFOAM-version	OpenFOAM-5.x		OpenFOAM-5.x	
Decompose Method	scotch		scotch	
Number of core	1	128	1	128
Execution time	419.21	**6.64**	420.44	**8.28**
Speed-Up	63		51	
Etc.	libhbm		-	

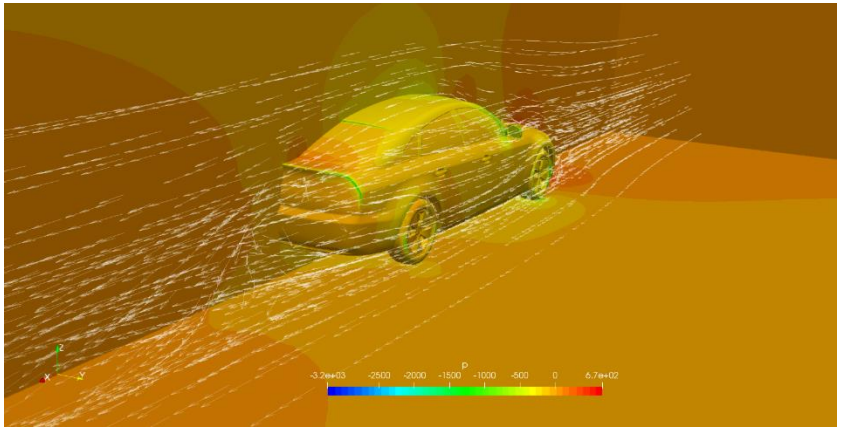
Parallel Performance

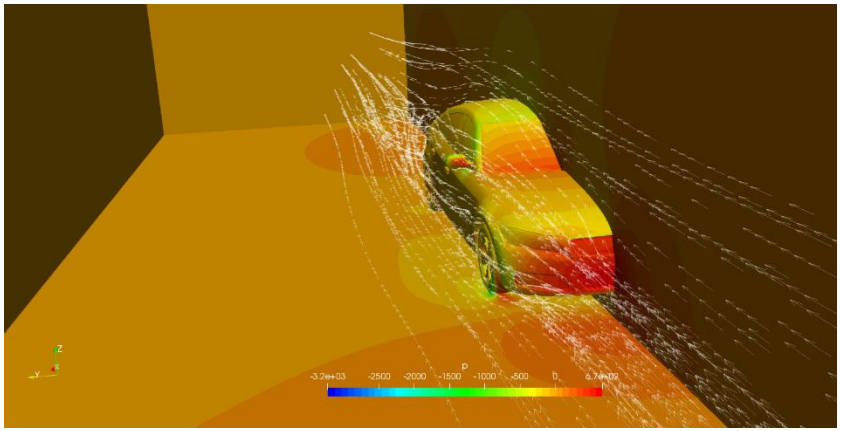
- Parallel Build Summary
  - 계산시간이 가장 빠른 설정: OF5.x10-libhbm
  - Speed-up이 가장 좋은 설정: OF5.x9-libhbm
  - OF5.x10-libhbm
    - 다양한 경우에서 병렬계산시간이 가장 빠름

	Execution time		Speed-up	
	OF5.x10-libhbm		OF5.x9-libhbm	
Compiler	Gcc-6.3.1/KNL		Intel-18.0.3(20180410)/ KNL	
MPI	INTELMPI		INTELMPI	
Number of core	1	128	1	128
Execution time	419.21	6.64	497.58	6.90
Speed-Up	63		72	

Computational Example

- DrivAer
  - Mesh Generator: snappyHexMesh utility
    - Number of cells: 약 210만 개
  - Simulation Conditions
    - U: 34m/s
    - Re: 10,426,666
    - Boundary condition
      - bottom: Inlet-Velocity
      - Tire & wheel: Rotating
  - Turbulence Model
    - K-OmegaSST model
  - Spatial Discretiztion
    - 1st order upwind
  - Machine
    - Intel Xeon cpu E5-2620 V3 @ 2.4GHz 12cores
    - Intel(R) Xeon Phi(TM) CPU 7250 @ 1.4GHz 12&128 cores

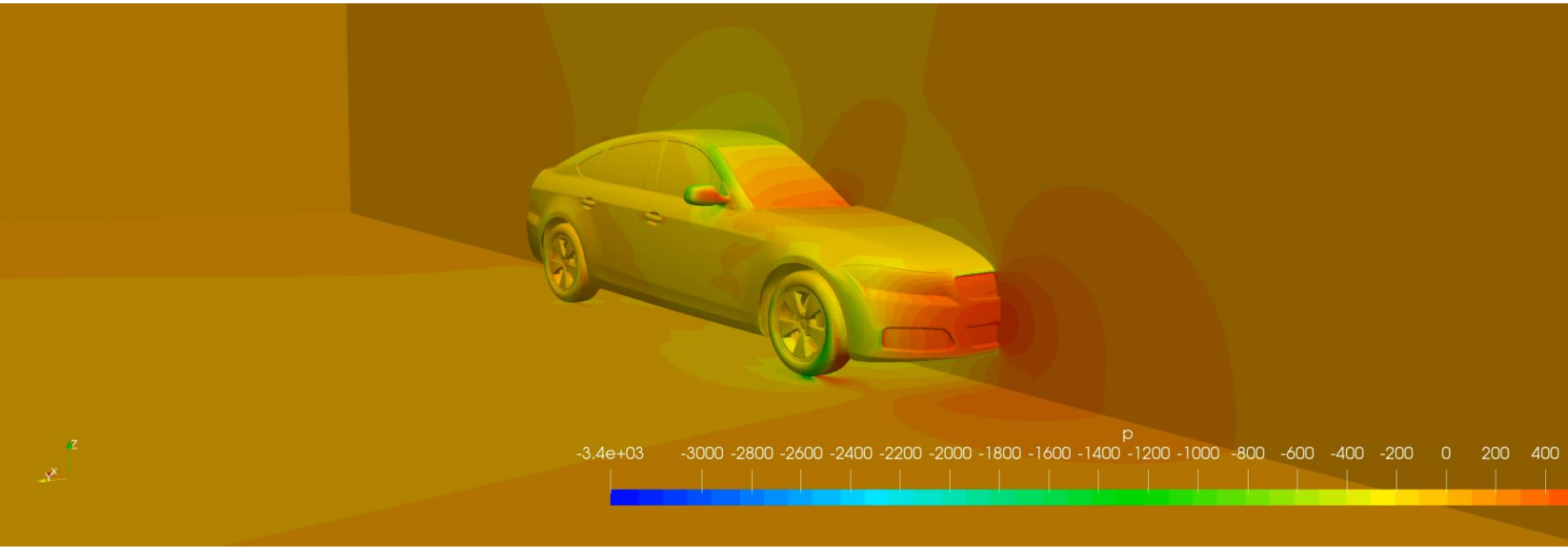




Computational Example

- Computing Time

	Intel Xeon cpu E5-2620 V3	Xeon PHI	
Number of core	12	12	128
Execution Time	7748s	4003.27s	485.73s



Computational Example

- Propeller
  - Mesh Generator: snappyHexMesh utility
    - Number of cells: 약 210만 개
  - Simulation Conditions
    - Solver: pimpleDyMFoam
    - U: 5m/s
    - Omega: 158rad/s
    - Boundary condition
      - Propeller: movingWall
      - Inner-Cylinder: cyclicAMI
  - Turbulence Model
    - kEpsilon model
  - Machine
    - Intel(R) Xeon Phi(TM) CPU 7250 @ 1.4GHz
    - Execution Time
      - 5479.15s