

Evaluation of the Intel Xeon Phi coprocessor for PET reconstruction



T. Dey*, P. Rodrigues, Oncology Solutions, Philips Research, Eindhoven, The Netherlands

Contact: Thomas.dey@philips.com

* Supported by the EU FP7-PEOPLE-2012-ITN project nr 317446, INFIERI, "Intelligent Fast Interconnected and Efficient Devices for Frontier Exploitation in Research and Industry"

Purpose:

- We aim to evaluate the Intel Xeon Phi coprocessor to accelerate 3D-PET reconstruction.
- We addressed sensitivity map generation, attenuation correction in back-projection and scatter correction as three hot spots of PET reconstruction.
- The important sub-module of radiological path calculation was evaluated to assess the benefit of portable and hardware specific vectorization.
- The focus is on total runtime reducing total, by spreading the workload between host and one or two coprocessors (Fig. 1).

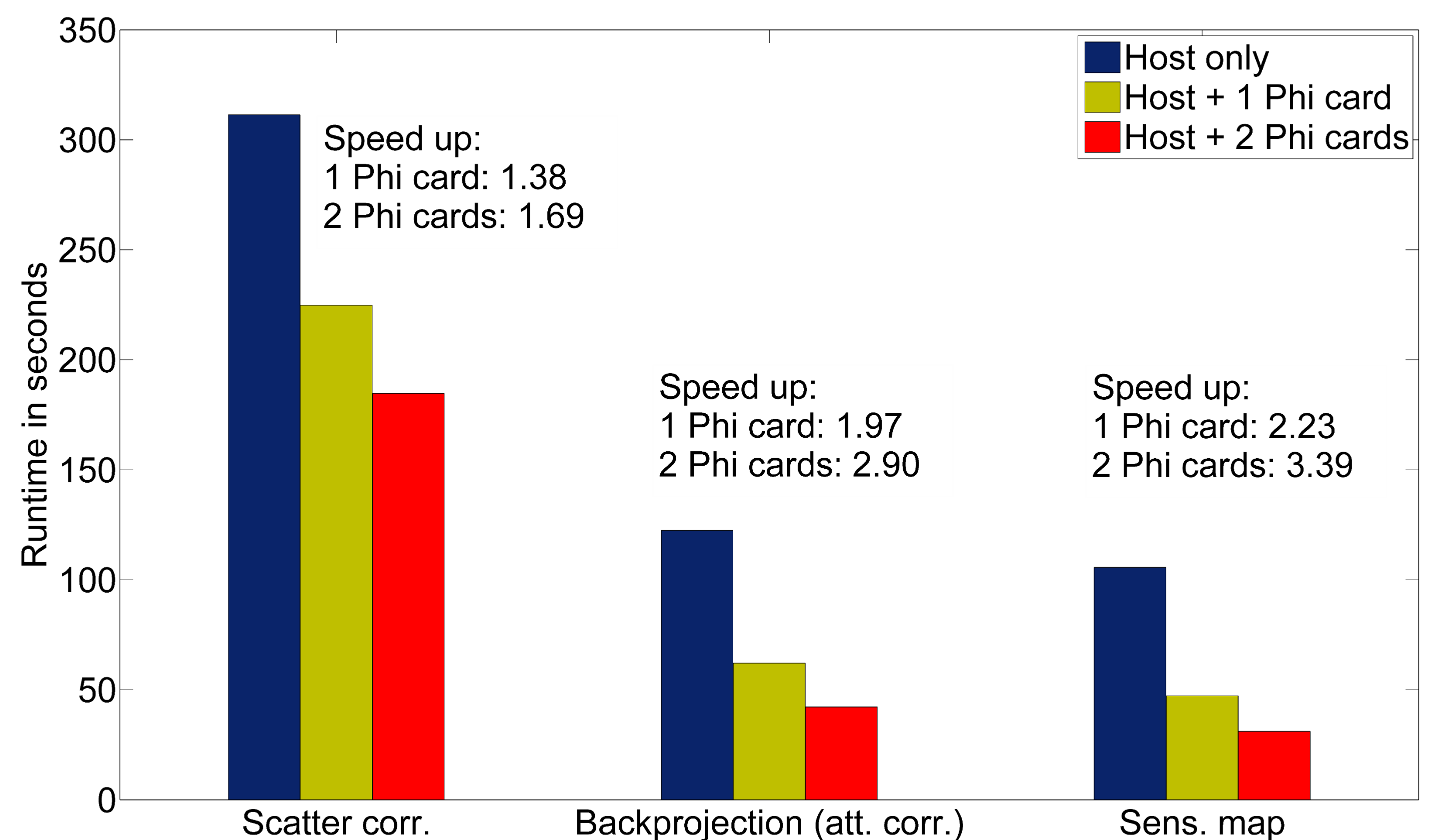


Fig 1: Runtime of PET reconstruction modules using Host and Xeon Phi.

Sensitivity map generation

The algorithm for sensitivity map generation is shown in Fig. 2. We use a voxel grid of $144 \times 144 \times 44$ and 4096 ray samples per voxel. The radiological path is calculated by an algorithm of Jacobs et al¹ and three implementations: C++, Intel Single Program Multiple Data (SPMD) compiler (ispc³) and Xeon Phi intrinsics. See the related section for more details. We used Embree² ray tracing kernels, optimized for host and Xeon Phi system to get the detector hit points.

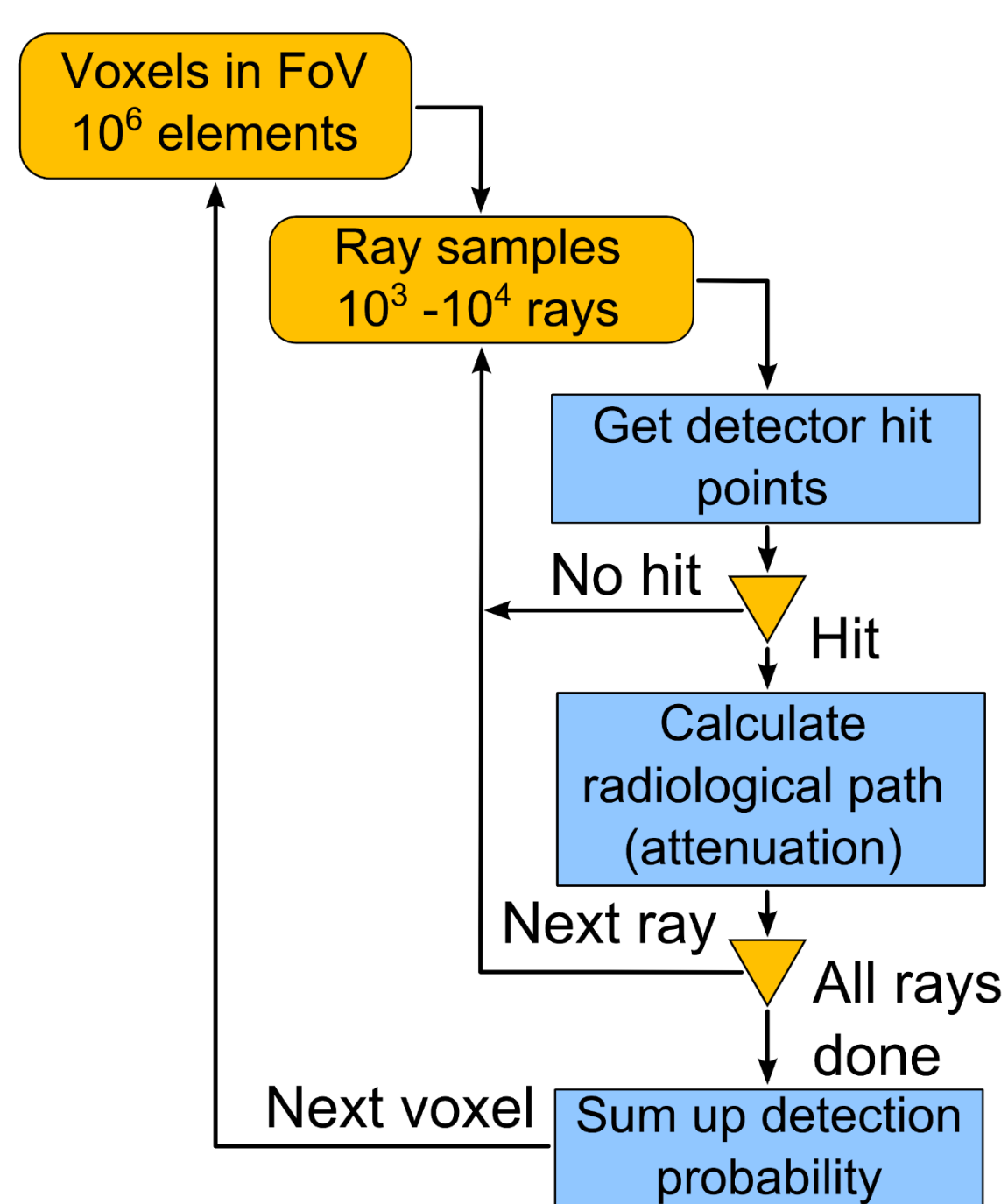


Fig. 2: Flowchart of sensitivity map algorithm.

Results

- Scalability factor of 0.9 up to 16 threads and 0.17 and afterwards (hyper-threading) on host system.
- Xeon Phi showed scalability factor: 0.9 up to 236 threads. Remaining 4 threads are reserved for memory mgt. (Fig. 3).
- Xeon Phi outperformed host by a factor of 1.25 with ispc and 1.43 using Xeon Phi intrinsics.

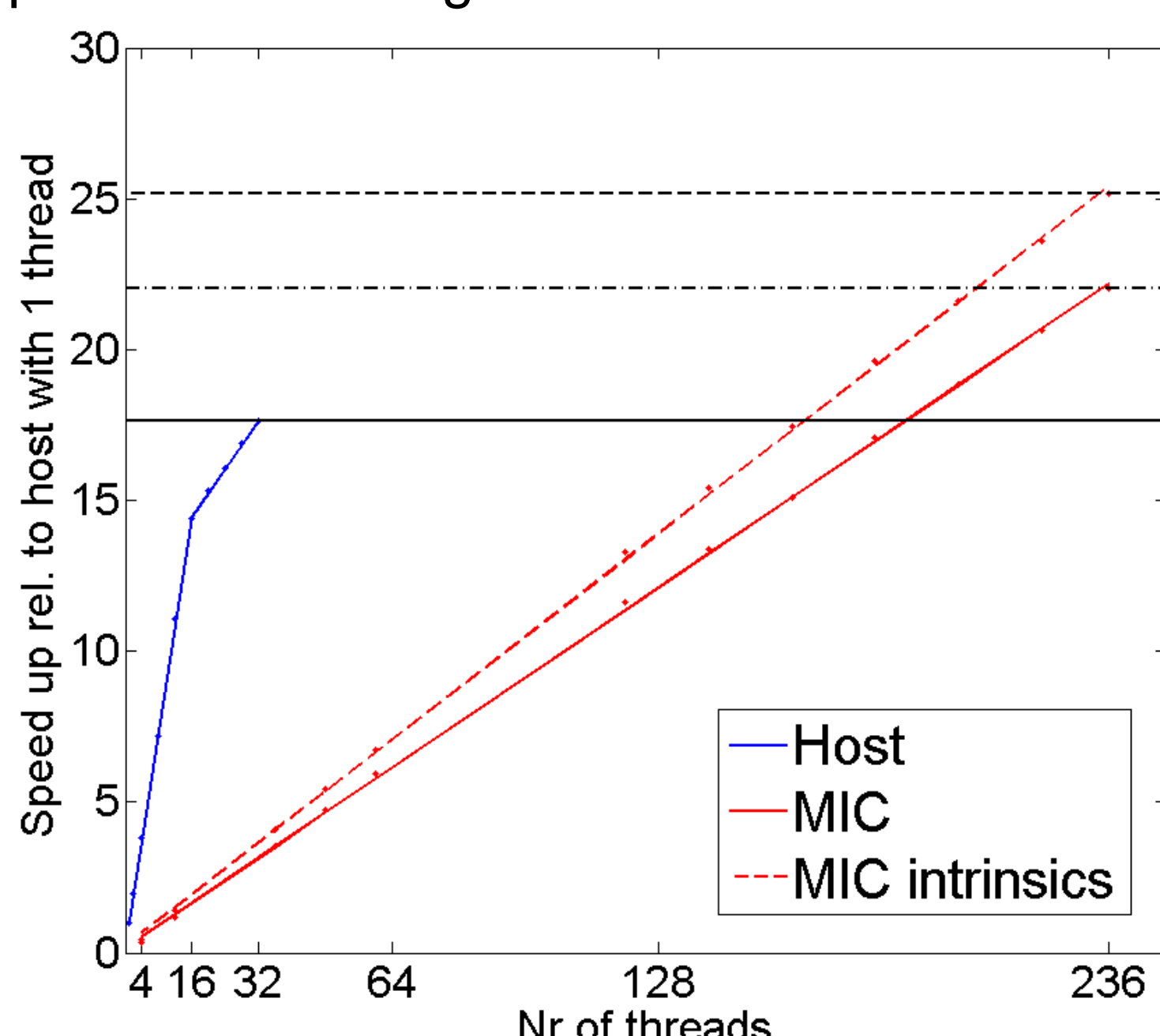


Fig. 3: Scalability graph, normalized to one host thread.

Attenuation corr. in back projection

This module is very similar to sensitivity map generation, but takes also into account the current image estimate during radiological path calculation. Results are comparable to sensitivity map generation, but showed a smaller speed up of 1.2 on Xeon Phi using intrinsics. This is probably caused by the additional gathering of image estimate values from memory.

Radiological path calculation:

Tracking rays through the voxel grid is an important sub-module of reconstruction. Using the vector processing units (VPUs) of CPU and Xeon Phi, promises speed up, by calculating multiple sample rays simultaneously. We compare a standard C++ implementation with portable ispc and hardware specific Xeon Phi programming using vector intrinsics on a Xeon Phi coprocessor in native mode. The results are depicted in Fig. 4. Intrinsics offer an extra speed up compared to ispc, but need to be adapted to the platform specific instruction set and programming is more laborious.

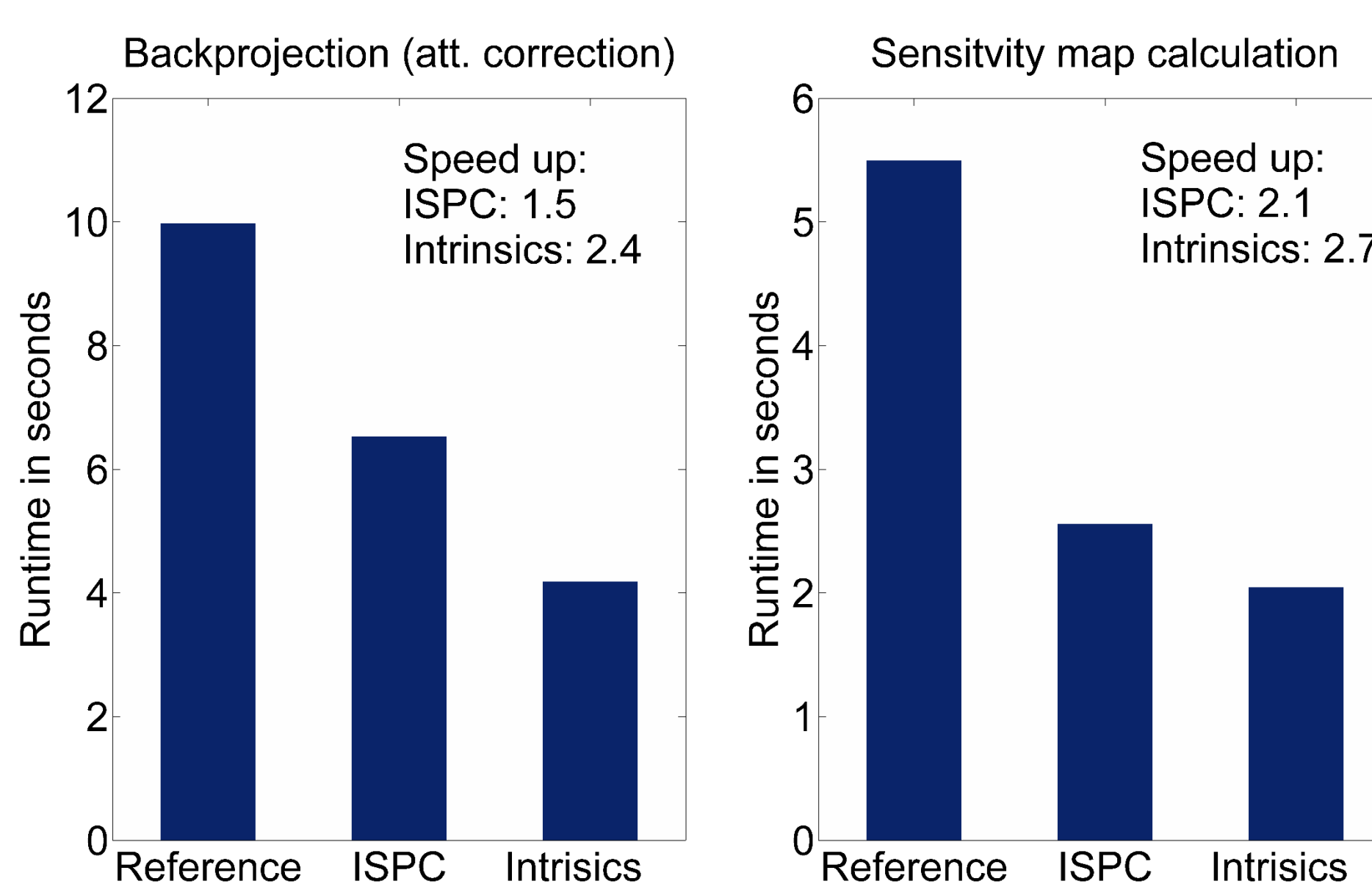


Fig. 4: Runtime on host and single Xeon Phi card.

Scatter correction:

Scatter was estimated by tracking a large number of ($\sim 10^7$) ray samples taking into account Compton and Rayleigh scattering as well as photoelectric interaction. Current implementation showed higher performance on the host CPU (speed up of 0.4 on Xeon Phi). However, still a runtime reduction of 1.7 is feasible by using two Xeon Phi cards.

Computational test platform:

- Host system:** HP SL250s Gen8, 64 GB RAM, 2x Intel Xeon (E5-2670) CPUs @ 2.6-3.3 GHz, 16 threads each (Hyper-threading), 256-bit VPUs (AVX).
- Coprocessors:** 2 Intel Xeon Phi cards (5110P) with 60 cores @ 1GHz, 240 threads and 8GB RAM each. 512-bit VPUs,

Key points and conclusion:

- Reasonable runtime reduction found in all three evaluated modules.
- In scatter estimation the performance of Xeon Phi is inferior to host system, probably due to strong branching. However a considerable reduction in total runtime was feasible.
- Sensitivity map generation and attenuation correction could benefit from vector processing capabilities and showed good scalability and speed up on the Xeon Phi.
- ISPC compiler offers portable code, taking advantage of the vectorization capabilities of host and Xeon Phi vector processing units.
- In radiological path calculation, Xeon Phi vector intrinsics gave an extra speed up, but lacks portability to host platform.
- Future Xeon Phi version (Knight's Landing) is expected to improve out-of order execution and memory integration, probably broadening the scope of suitable applications.

References:

- F. Jacobs et al., "A fast algorithm to calculate the exact radiological path through a pixel or voxel space," Journal of computing and information technology, vol. 6, no. 1, pp. 89-94, 1998.
- S. Woop, L. Feng, I. Wald, and C. Benthin, "Embree ray tracing kernels for CPUs and the xeon phi architecture," in ACM SIGGRAPH 2013 Talks. ACM, 2013, p. 44.
- M. Pharr and W. R. Mark, "ispc: A SPMD compiler for high-performance CPU programming," in Innovative Parallel Computing (InPar), 2012. IEEE, 2012, pp. 1-13.



The research leading to these results has received funding from the People Programme (Marie Curie Actions) of the European Union's Seventh Framework Programme FP7/2007-2013/ under REA grant agreement n° [317446] INFIERI "INtelligent Fast Interconnected and Efficient Devices for Frontier Exploitation in Research and Industry"

PHILIPS