GPU Technology Applied to Seismic Imaging via OpenACC

Ahmad Qawasmeh, Abid Malik, Barbara Chapman
{arqawasm,ammalik3,chapman}@cs.uh.edu, Dept. of Computer Science, University of Houston

Background

- GPU computing offers tremendous potential to accelerate complex scientific applications.
- Developing portable code has been and remains a challenge.
- Can emerging high-level GPU directive-based programming model such as OpenACC extract max potentials of GPUs?
- Two Nvidia GPU cards: 1) M2090 and 2) K40, residing in IBM and CRAY XC30 clusters respectively were used.
- We developed OpenACC implementations for both seismic modeling and Reverse Time Migration (RTM) algorithms that solve the isotropic, acoustic, and elastic wave equations.

Motivation

- Observe the feasibility of using OpenACC to improve the performance of seismic imaging applications.
- Asses OpenACC with respect to its ease of use, performance portability, and performance gain.

Seismic Imaging

- RTM is a seismic migration method for creating images in complex wave propagation fields.
- Seismic modeling forms the forwarding phase of RTM in which a seismic wave is propagated from source to the subsurface.

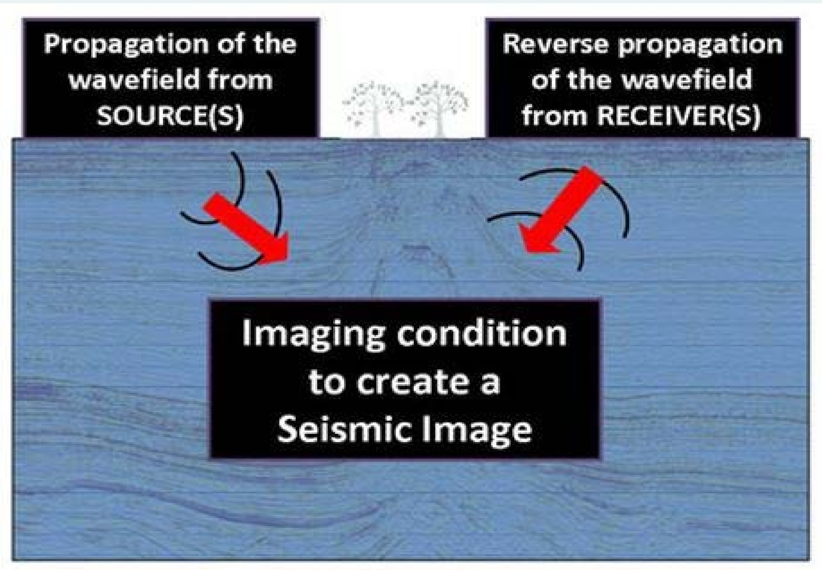


Fig. 1: Seismic imaging

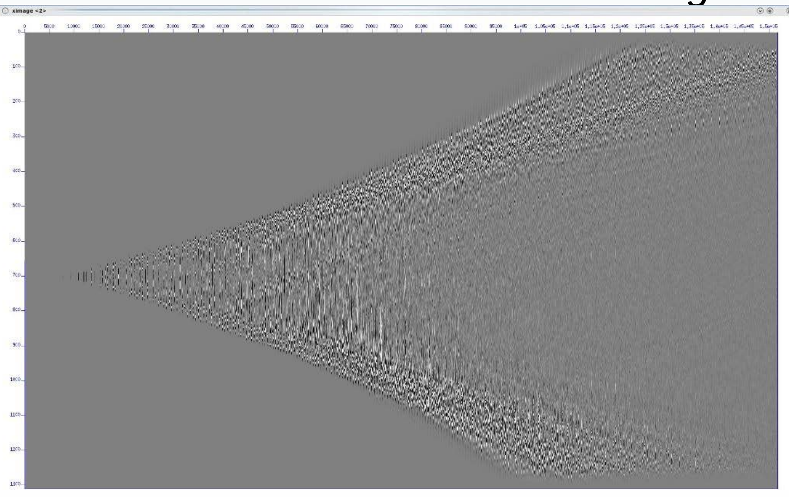


Fig. 2: A 2D seismic modeling snapshot in acoustic media

Fig. 3: A 2D seismic image of RTM in acoustic media

Formulations to represent the earth models

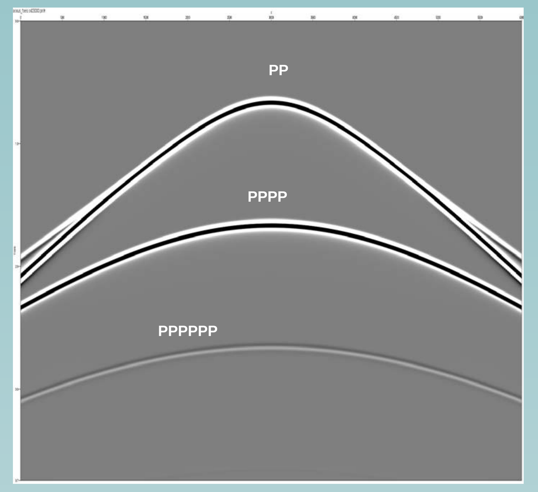


Fig. 4: Acoustic Modeling

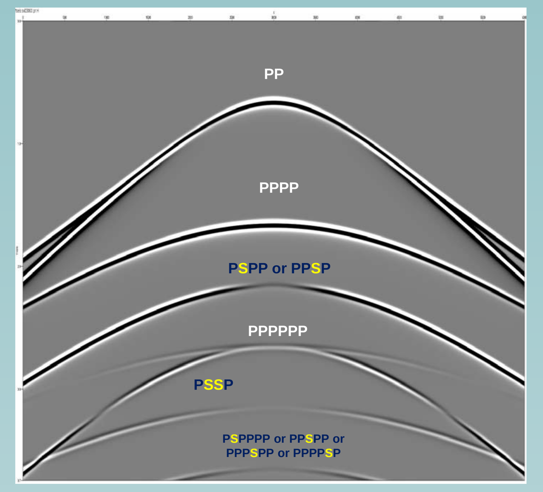


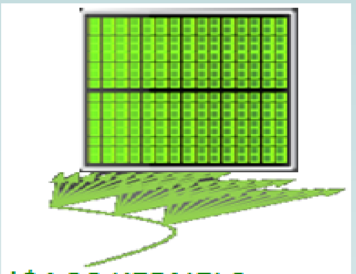
Fig. 5: Elastic Modeling

OpenACC programming model

- Host-directed execution with attached GPU
- Distinct memory spaces on the host and device

```
// C/C++ example
#pragma acc *
{structured block}

! Fortran example
!$acc *
<structured block>
!$acc end *
```



```
!$ACC KERNELS
  do i=0, N
     c(i) = a(i) + b(i)
!$ACC END KERNELS
```

Fig. 6: Basic example

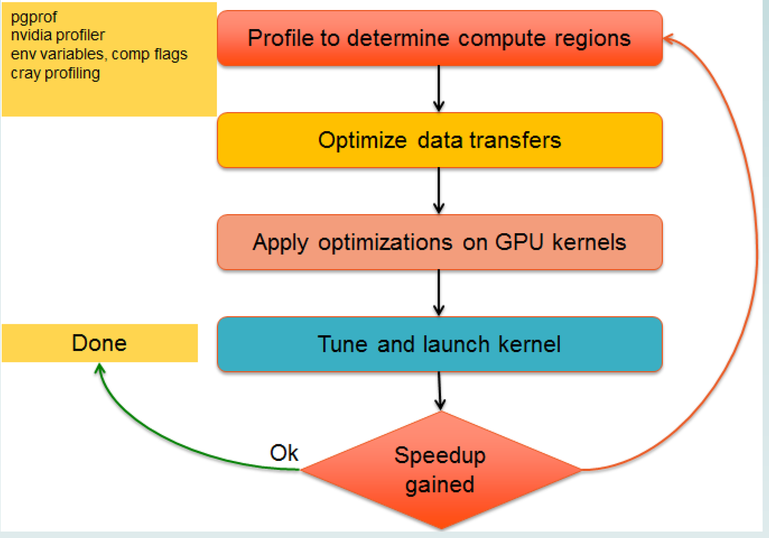
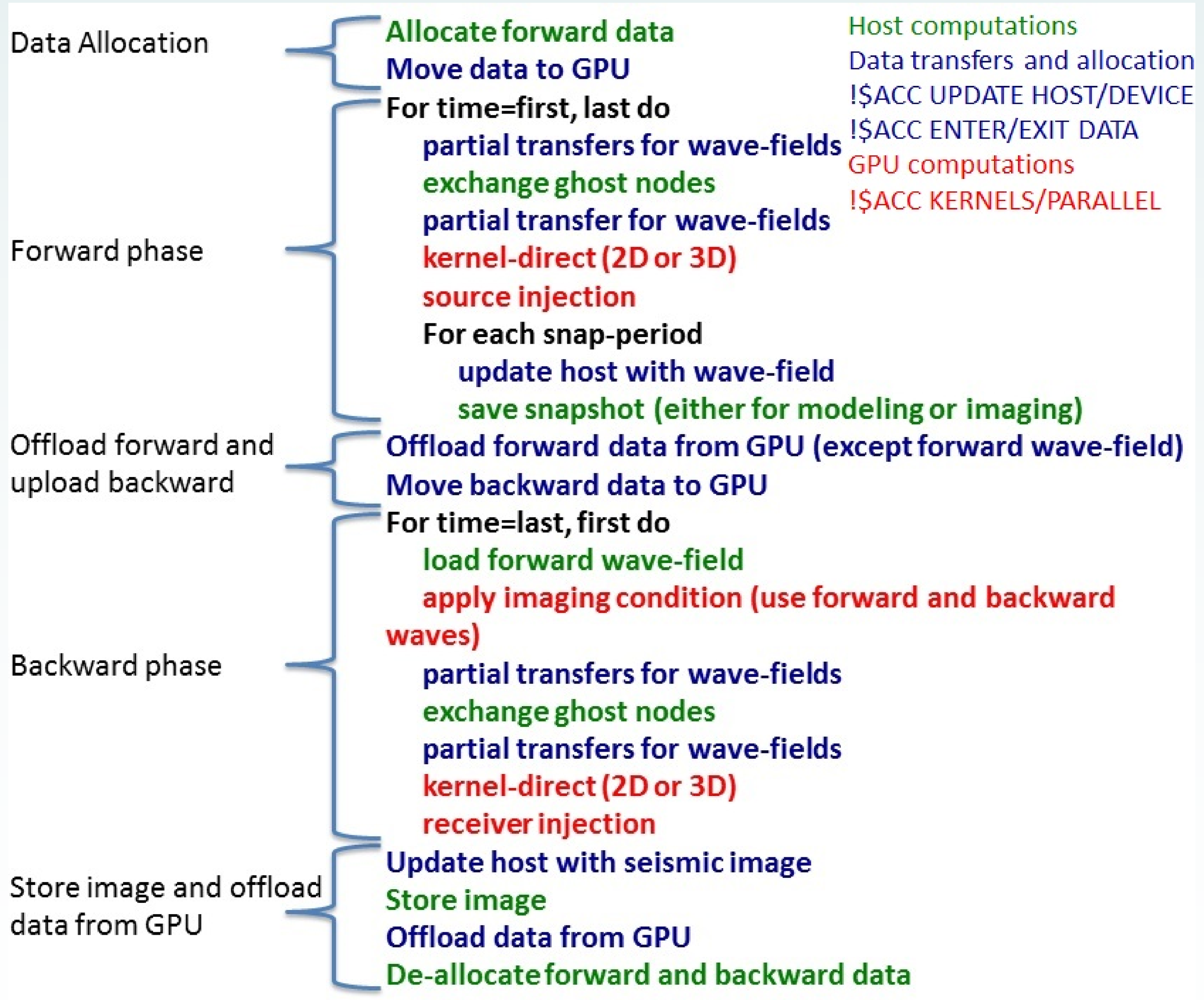


Fig. 7: User's responsibility

OpenACC implementation of RTM

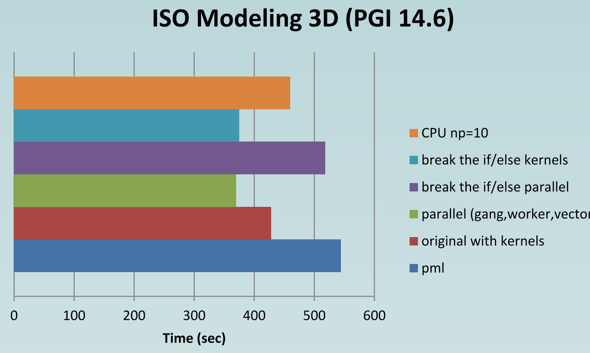


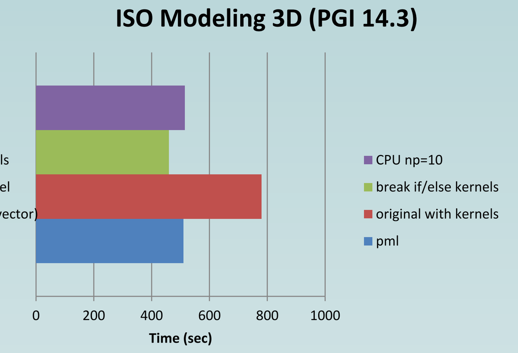
- Optimization techniques used: coalescing memory accesses, overlapping GPU computations via asynchronous operations, applying different loop scheduling schemas, code transformations, OpenACC optimizations

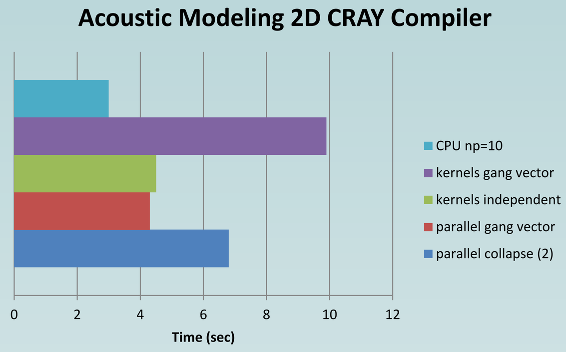
Evaluation

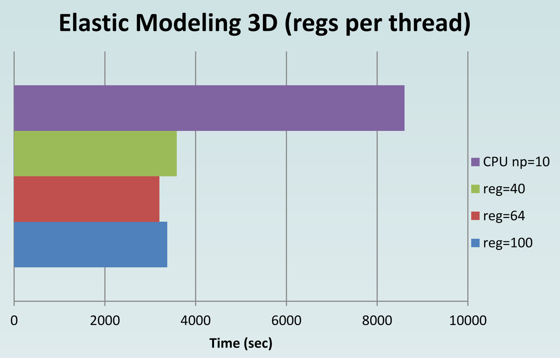
Table 1: Evaluation platform

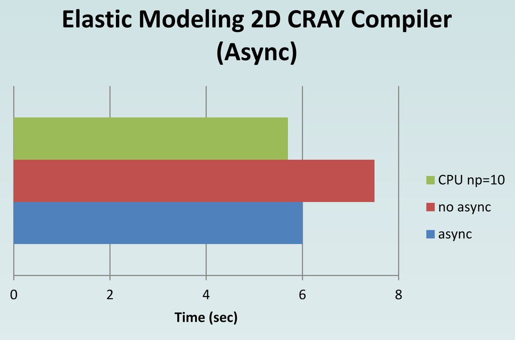
Software/Hardware	Facts
CPU/CRAY	Two Intel 10 core Xeon Ivy Bridge E5-2680 v2 @2.8 GHz, total 20 cores/64GB memory per node.
CPU/IBM	Two Intel Quad-core Xeon E5640 @2.8 GHz, 12MB L3 Caches, total 8 cores/24GB memory per node
GPU	1- Nvidia Tesla Kepler K40 on CRAY with 2880 CUDA cores, 12 GB GDDR5 Memory, ECC: disabled. 2- Nvidia Tesla Fermi M2090 on IBM with dedicated with 512 CUDA cores, 6 GB GDDR5 Memory, ECC: disabled
API and Languages	OpenACC 2.0, MPICH 3.1, FORTRAN 2003
Compilers	PGI Compiler 13.7 / 14.3 / 14.6, CRAY Compiler 8.2.6, NVIDIA CUDA 5.0 / 5.5











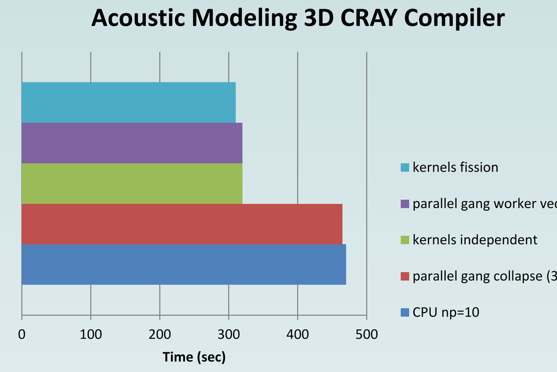


Table 2: Speedup results (RTM)

	CRAY cluster		IBM cluster
Model\Compiler	CRAY compiler	PGI compiler	PGI compiler
ISOTROPIC 2D	1.2	1.0	1.3
ACOUSITC 2D	2.4	2.0	7.9
ELASTIC 2D	1.7	1.5	2.3
ISOTROPIC 3D	1.1	1.2	1.1
ACOUSTIC 3D	1.3	1.3	10.8
ELASTIC 3D	3.0	2.9	x

Table 3: Speedup results (Modeling)

	CRAY cluster		IBM cluster
Model\Compiler	CRAY compiler	PGI compiler	PGI compiler
ISOTROPIC 2D	0.7	1.1	2.3
ACOUSTIC 2D	0.9	1.1	1.2
ELASTIC 2D	0.7	1.1	2.4
ISOTROPIC 3D	0.9	1.2	1.0
ACOUSTIC 3D	1.2	1.7	2.3
ELASTIC 3D	2.4	2.7	x

Conclusion

- OpenACC versions for 12 different seismic imaging cases used in the oil and gas industry were developed. Best speedup against a full socket MPI implementation: 3x on CRAY and 10x on IBM.
- OpenACC propagators exhibit different speedup behaviors depending on the intensity of computations.
- Code restructuring is crucial to enhance GPU utilization.
- Performance portability is still a big challenge.

A. Qawasmeh, M. Hughes, H. Calandra, and B. Chapman, "GPU technology applied to reverse time migration and seismic modeling via OpenACC" PMAM@PPoPP. ACM, 2015, pp. 75-85.

Acknowledgement: Many thanks go to TOTAL company for providing the computing resources.