Accelerating MATLAB with CUDA

Massimiliano Fatica
NVIDIA
mfatica@nvidia.com

Won-Ki Jeong
University of Utah
wkjeong@cs.utah.edu

Overview

- MATLAB can be easily extended via MEX files to take advantage of the computational power offered by the latest NVIDIA GPUs (G8x):
  - G80 is a massively parallel processor (up to 128 processors running at 1.5Ghz)
  - Current generation has support of IEEE-754 single precision, double precision will be available towards the end of the year.
- Programming the GPU for computational purpose was a very cumbersome task before CUDA:
  - CUDA is a C-like programming language and includes FFT and BLAS libraries. CUDA is freely available (Windows and Linux)
- This work shows the feasibility and benefits of this approach

NVMEX

- To interface CUDA and MATLAB we had to slightly modify the MEX infrastructures:

  CUDA files have a .cu suffix and need to be compiled with a specific compiler (nvcc)

  nvmex script and CUDA MEX examples available at (http://developer.nvidia.com/object/matlab_cuda.html)

- Initial focus on 2D FFTs:

  FFT-based methods could be often used in single precision.

- Application selected:

  Pseudo-spectral solution of Euler equation

Results

2D isotropic turbulence: 1024x1024 mesh, 400 RK4 steps on Windows, Opteron 250, Quadro FX5600.

	Runtime	Speed-up
Standard MATLAB	8098 s	
Overload FFT2 and IFFT2	4425 s	1.8x
Overload Non-linear term	735 s	11x
Overload Non-linear term , FFT2 and IFFT2	577 s	14x

Advection of elliptic vortex: 256x256, Linux, Opteron 250, Quadro FX5600.

	Runtime	Speed-up
Standard MATLAB	168 s	
Overload Non-linear term , FFT2 and IFFT2	14.9 s	11x