Introduction to GPU Computing

Matthis Hauschild

Universität Hamburg
Fakultät für Mathematik, Informatik und Naturwissenschaften
Fachbereich Informatik

Technische Aspekte Multimodaler Systeme

December 4, 2014

Table of Contents

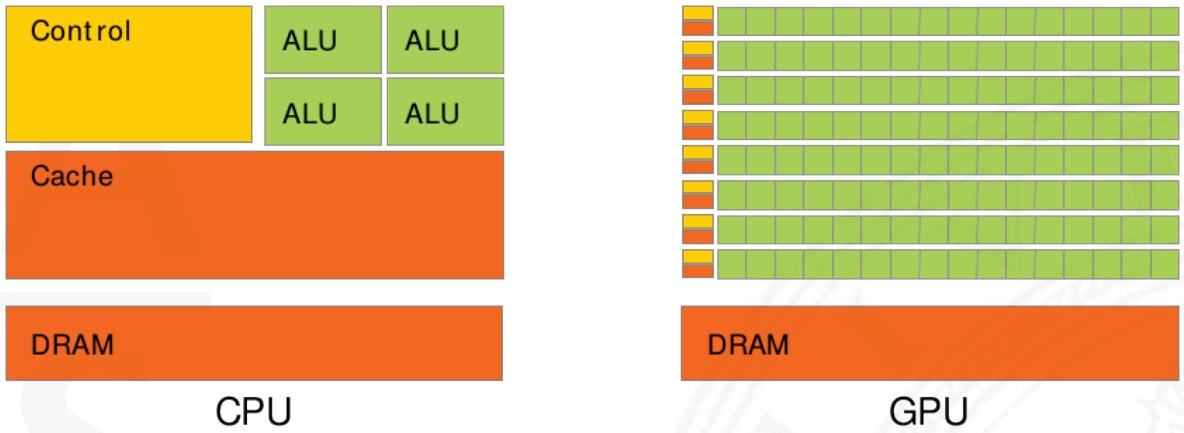
1. Architecture of a GPU
2. General-purpose computing on GPUs
3. Applications of GPGPU
4. Performance evaluation examples

What is a GPU

- Graphics processing unit
- Main GPU manufacturers
  1. Intel
  2. AMD
  3. Nvidia
- Performance characteristics:[1]
  - GPU architecture: 28 nm
  - GPU speed: $\sim$ 1 GHz
  - Memory amount: 8 GiB GDDR5
  - Memory bandwidth: 640 GiB/s

[1] based on the AMD Radeon R9 series (cf.[1])

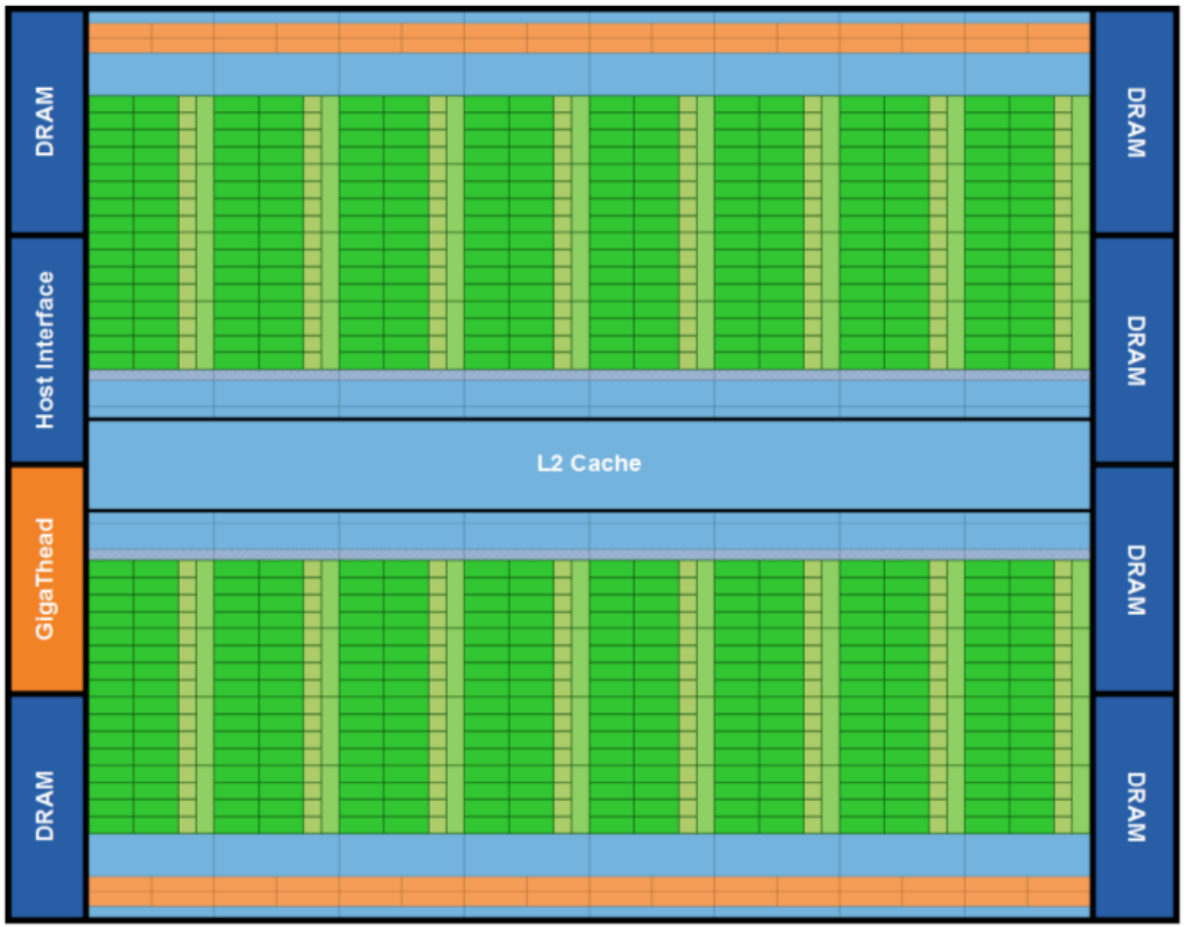
Difference between GPU and CPU[3]



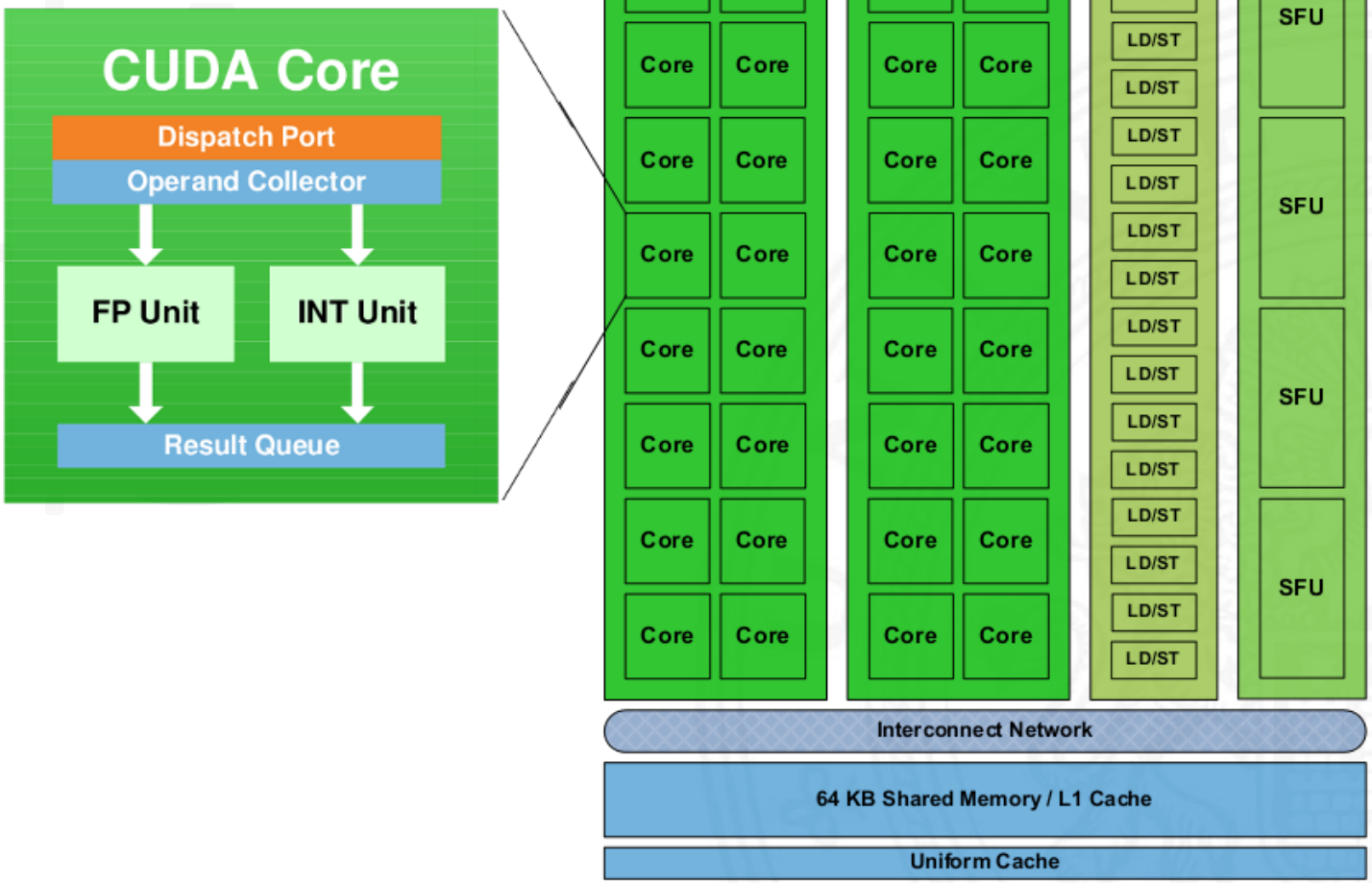
- CPU optimized for single thread execution
- GPU optimized for multiple data execution

Architecture of a GPU[4]

based on the Nvidia Fermi architecture:



Architecture of a GPU[4]



Architecture of a GPU[4]

Summary of the Nvidia Fermi architecture:

- 16 Streaming Multiprocessors (SM)
- 32 CUDA cores per SM
- = 512 CUDA cores $\Rightarrow$ 512 FMA op/clock

$\Rightarrow$ it is great for generating graphics, but what else could be done with it?

What is GPGPU[5]

- General-purpose computing on graphics processing units
- Using GPU for non-graphical computations
  - Good for data parallelism
  - Bad for instruction parallelism
- First use in LU factorization
- Became popular at 2001 with matrix multiplication
- Started using DirectX and OpenGL

GPGPU Frameworks

- **Brook** – One of the earliest GPU frameworks by Stanford University
- **CUDA** – Proprietary Nvidia-only framework
- **OpenCL** – Open source general framework by Khronos Group
- **C++ AMP** – Open C++ extension by Microsoft
- **OpenACC** – C, C++ and Fortran extension
- **ArrayFire** – Wrapper for CUDA, OpenCL, etc.

General applications of GPGPU

Again, GPGPU can only be superior to CPU computing, if the same algorithm is applied to a lot of data (data parallelism)
For example:

- k-nearest neighbor
- Fast Fourier Transform
- Segmentation
- Audio Processing
- CT reconstruction
- Weather forecasting
- Cryptography
- Database operations

Applications of GPGPU in Robotics[2]

For example:

- Generally many image processing tasks
- Frame transformation
- Inverse kinematic calculation
- 3D pose estimation
- Point-set registration

Performance evaluation examples

Test 1

- Sobel operator on a real image using OpenCL
- Measurement of the possible frames per second
- On GPU and CPU

Test 2

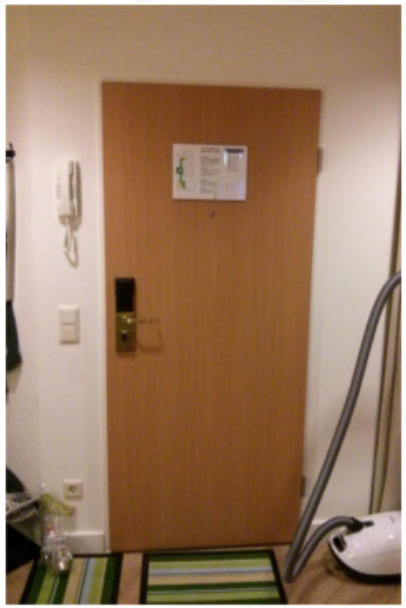
- Matrix multiplication of two squared matrices using OpenCL
- Measurement of time needed for calculation
- On GPU and CPU

Performance evaluation examples - System characteristics

- My CPU:
  - Model: AMD Phenom II X4 965
  - Clock speed: 3400 MHz
  - Misc: 4 Cores, SSE3
- My GPU:
  - Model: AMD Radeon HD 6950,
  - Memory: 2048 MB
  - Core clock: 800 MHz
  - Memory clock: 1250 MHz
  - Memory bandwidth: 160 GB/s
- My RAM: 8 GB

Performance evaluation examples - Test 1

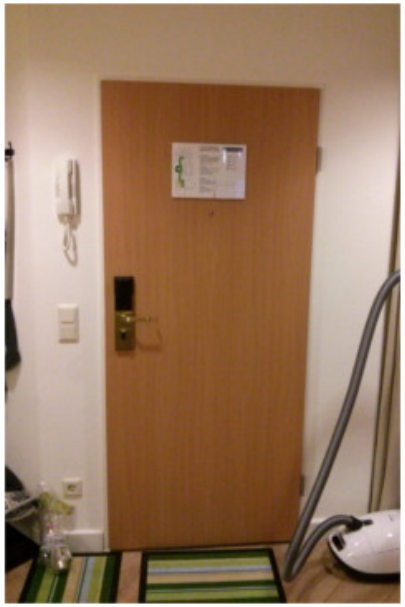
The Sobel operator:

1. 

$*$

-1	0	+1
-2	0	+2
-1	0	+1

= dx's

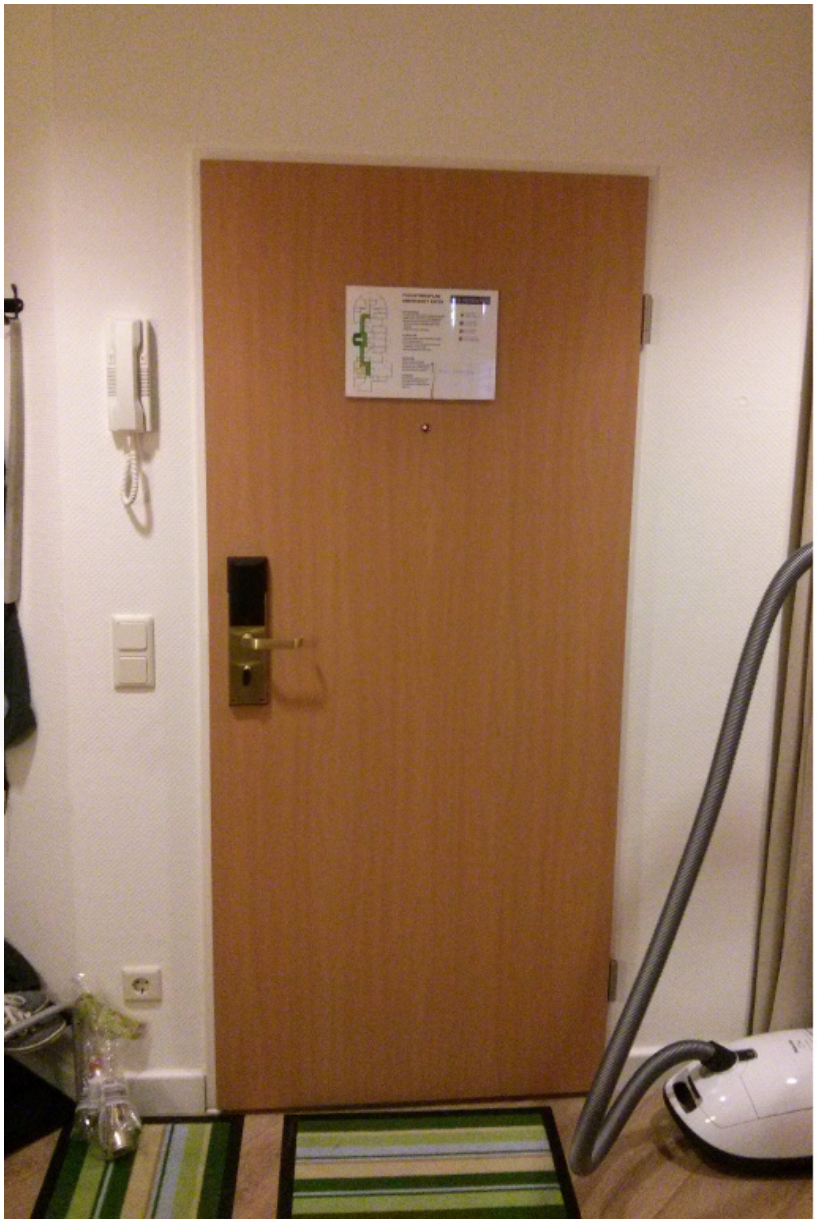
2. 

$*$

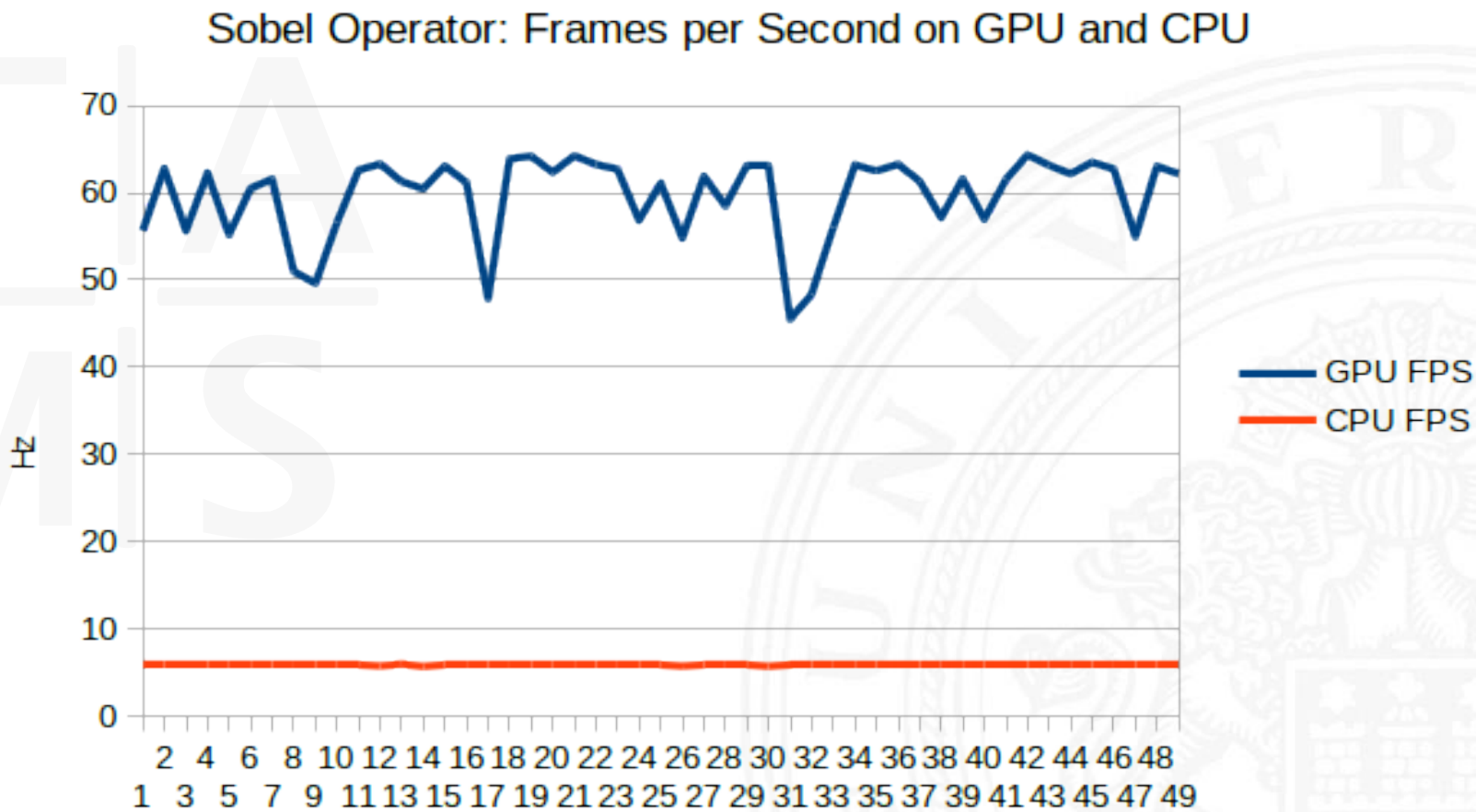
+1	+2	+1
0	0	0
-1	-2	-1

= dy's

3. $s = \sqrt{dx^2 + dy^2}$



Performance evaluation examples - Test 1



Performance evaluation examples - Test 2

Matrix Multiplication[2]:

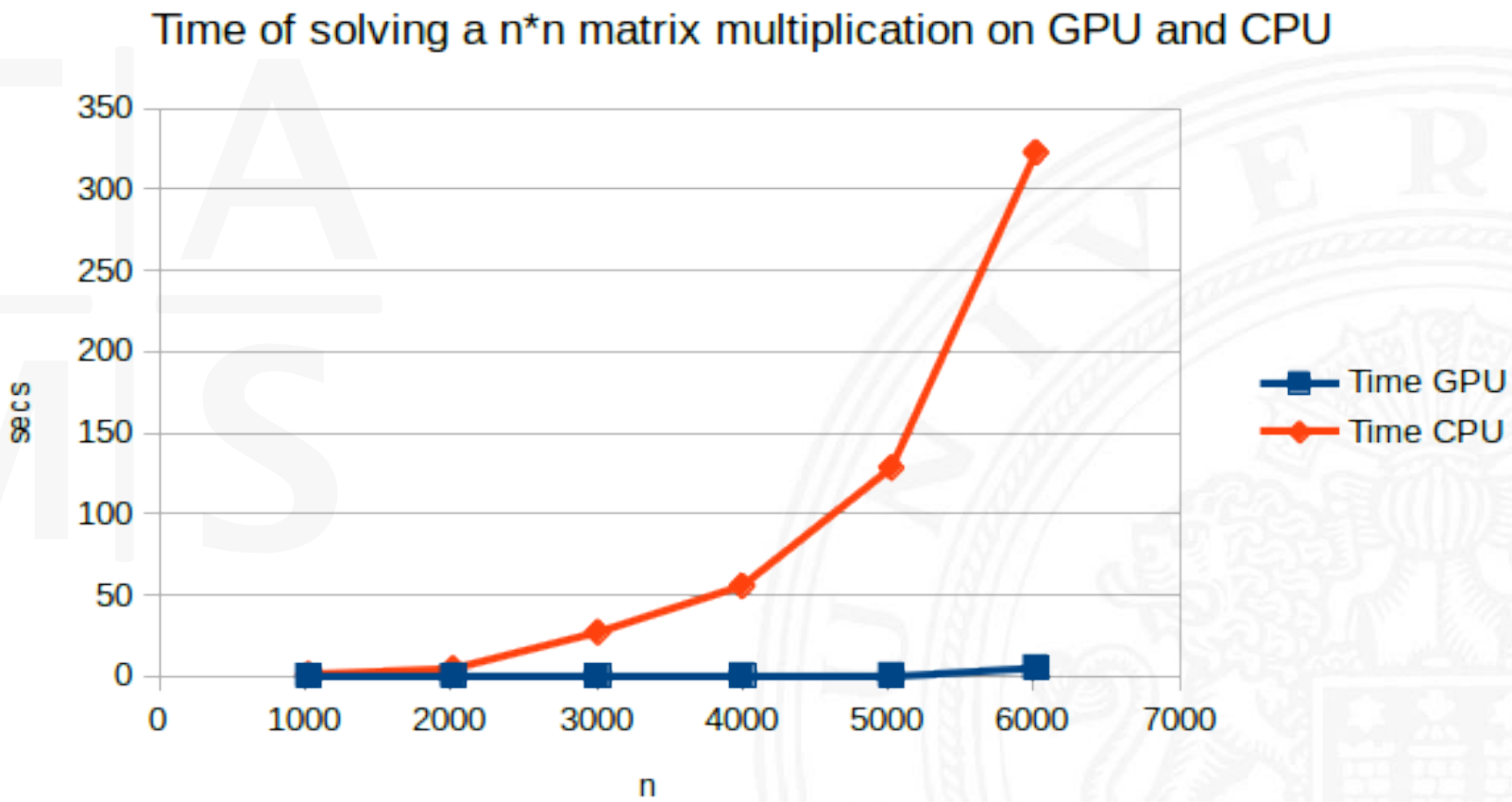
$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{pmatrix} \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \\ b_{31} & b_{32} \end{pmatrix} = \begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix}$$

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{pmatrix} \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \\ b_{31} & b_{32} \end{pmatrix} = \begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix}$$

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{pmatrix} \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \\ b_{31} & b_{32} \end{pmatrix} = \begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix}$$

[2] from `http://www.mathematrix.de/wp-content/uploads/matrixmul2.png`

Performance evaluation examples - Test 2



Thank you for your attention!

Matthis Hauschild
3hauschi@informatik.uni-hamburg.de

Universität Hamburg
Fakultät für Mathematik, Informatik und Naturwissenschaften
Fachbereich Informatik

Technische Aspekte Multimodaler Systeme

Bibliography

[1] AMD. *AMD Radeon™ R9 Grafikkartenserie*, 2014. http://www.amd.com/de-de/products/graphics/desktop/r9#.

[2] J. Bedkowski and A. Maslowski. *GPGPU computation in mobile robot applications*. Warsaw University of Technology, 2012.

[3] Nvidia. *CUDA C Programming Guide*, 2014. http://docs.nvidia.com/cuda/pdf/CUDA_C_Programming_Guide.pdf.

[4] Nvidia. *NVIDIA's Next Generation CUDA Compute Architecture: Fermi*, 2014. http://www.nvidia.de/content/PDF/fermi_white_papers/NVIDIA_Fermi_Compute_Architecture_Whitepaper.pdf.

[5] Wikipedia. *General-purpose computing on graphics processing units*, 2014. http://en.wikipedia.org/wiki/General-purpose_computing_on_graphics_processing_units.