Ускорение MATLAB на GPU

GPGPU

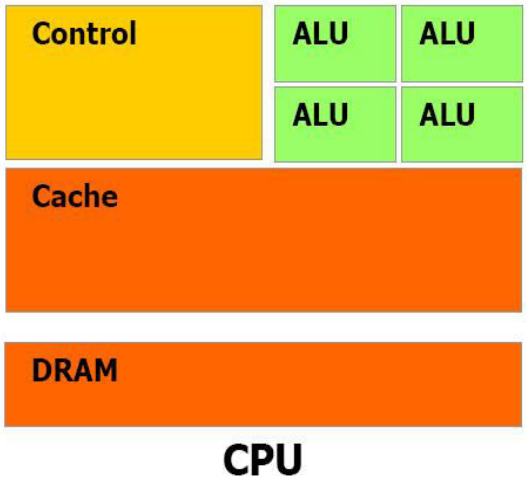
***G**eneral-**P**urpose **G**raphics **P**rocessing **U**nits* (2003 г.) – («GPU общего назначения») – техника использования графического процессора видеокарты для общих (неграфических) вычислений, которые обычно проводит центральный процессор.

Применение GPGPU:

- Вычислительная математика
- Вычислительная биология
- Вычислительная экономика
- Моделирование в физике
- Обработка сигналов…

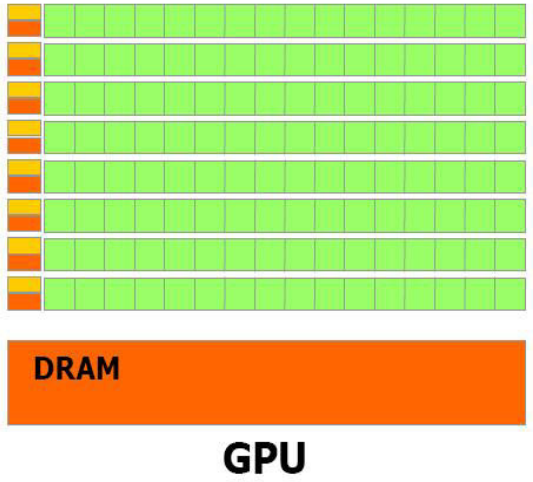
CPU

- Память оптимизирована под минимальную латентность (система “кэшей”).
- Много транзисторов “управления” (предсказание ветвлений, планировщики и пр.).
- Архитектура оптимизирована для программ со сложным управлением (эффективная обработка ветвлений).



GPU

- Память оптимизирована под максимальную пропускную способность.
- Большая часть транзисторов для вычислений.
- Архитектура оптимизирована для программ с большим объемом вычислений (параллелизм по данным типа SIMD).
- Латентность скрывается вычислениями во время запросов к памяти.



GPU

демонстрируют хорошие результаты в параллельной обработке данных:

- с одной и той же последовательностью действий, применяемых к большому объёму данных (многопоточные вычисления), что подразумевает меньшие требования к управлению исполнением,

- с высокой плотностью арифметики - высоким отношением числа арифметических операций к числу обращений к памяти, что означает возможность покрытия латентности памяти вычислениями.

CUDA

***C**ompute **U**nified **D**evice **A**rchitecture* (2007 г.) -
новая программно-аппаратная
архитектура NVIDIA для параллельных
вычислений на GPU, предоставляющая средства (toolkit)
для организации вычислений общего назначения на GPU

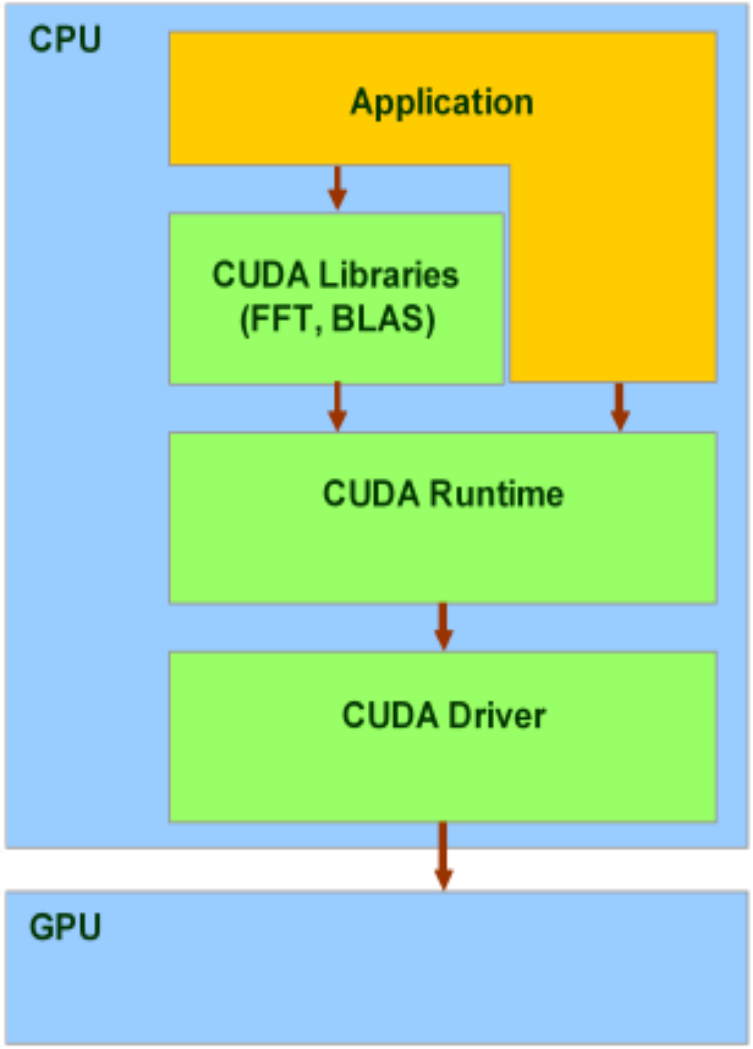
Присутствует в GPU NVidia:

- GeForce 8800 и выше,
- Quadro FX 5600/4600 и выше,
- Tesla серии 10,
- Tesla серии 20 (Fermi).

http://www.nvidia.ru/object/cuda_home_new_ru.html

CUDA Toolkit

- компилятор nvcc;
- библиотеки CuFFT и CuBLAS;
- профилировщик;
- отладчик gdb для GPU;
- API высокого уровня (CUDA Runtime) и API низкого уровня (CUDA Driver);
- руководство по программированию;
- CUDA Developer SDK (исходный код, утилиты и документация).



http://developer.nvidia.com/object/cuda_3_2_downloads.html

Вычислительный сервер на базе GPU NVidia Tesla

Конфигурация:

- Платформа SuperServer SYS-7046GT-TRF-TC4
- CPU 2 x Intel Nehalem 4Core X5570 2.93 GHz
- RAM 12 x 2 GB RAM
- HDD 8 x 500 GB SATA
- **GPU 4 x TESLA C1060 (4 x 240 ядер)**
- Пик. произв. (SP) **~3,73 TFlops** (4 GPU)
- Пик. произв. (DP) **312 GFlops** (4 GPU)

GPU NVidia Tesla C1060

- Total amount of device memory: **4** GB
- Number of multiprocessors: **30**
- Number of cores: **240**
- Clock rate: **1.30** GHz

Вычислительный сервер на базе GPU NVidia Tesla

Программное обеспечение:

- OS Gentoo Linux (kernel 2.6.37)
- Intel C/C++/F90/F95 (v.11.1)
- CUDA Toolkit (v.3.2)
- AccelerEyes Jacket (v.1.3) Multi-GPU License (4 GPU)
- Matlab (7.11.0.584 R2010b 64-bit) Concurrent Network License

AccelerEyes Jacket

accelerates MATLAB code on GPUs. With minimal knowledge and time, single threaded M-codes are transformed to GPU-enabled applications that fully leverage hardware. Thousands of MATLAB function syntaxes are supported.

Jacket is designed for engineers, scientists, and analysts who want maximum performance and maximum leverage of GPU resources, without hassling with low-level programming details. Jacket automatically translates M-code to high performance primitives required for best utilization of GPUs. All GPU-specific programming details are handled by Jacket, freeing the user to focus on science, engineering, and analytics.

http://www.accelereyes.com/products

AccelerEyes Jacket

пример кода Matlab + Jacket:

```
>>  N = 128; % matrix size
>>  M = 200; % number of tiled matrices
>>
>>  % Create Data
>>  [Ac Bc] = deal(complex( gones(N,N,M, 'single'),0));
>>
>>  % Compute 200 (128x128) FFTs
>>  gfor ii = 1:M
>>      Ac(:,:,ii) = fft2(Bc(:,:,ii));
>>  gend
>>
>>  % Bring the results back to CPU
>>  Ac = single(Ac);
```

MATLAB Acceleration on Tesla and Quadro GPUs

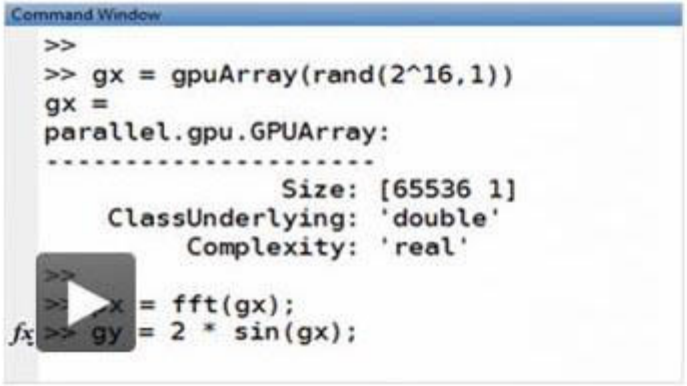
R2010b

NVIDIA and MathWorks have collaborated to deliver the power of GPU computing for MATLAB users. Available through the latest release of MATLAB 2010b, NVIDIA GPU acceleration enables faster results for users of the Parallel Computing Toolbox and MATLAB Distributed Computing Server. Visit MATLAB GPU Computing with NVIDIA CUDA GPUs for more information about GPU computing with MATLAB.

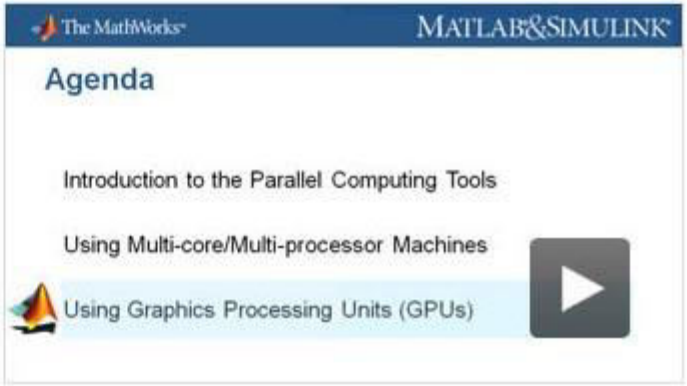
MathWorks

The latest release of Parallel Computing Toolbox and MATLAB Distributed Computing Server takes advantage of the CUDA parallel computing architecture to provide users the ability to

- Manipulate data on NVIDIA GPUs
- Perform GPU accelerated MATLAB operations
- Integrate users' own CUDA kernels into MATLAB applications
- Compute across multiple NVIDIA GPUs by running multiple MATLAB workers with Parallel Computing Toolbox on the desktop and MATLAB Distributed Computing Server on a compute cluster



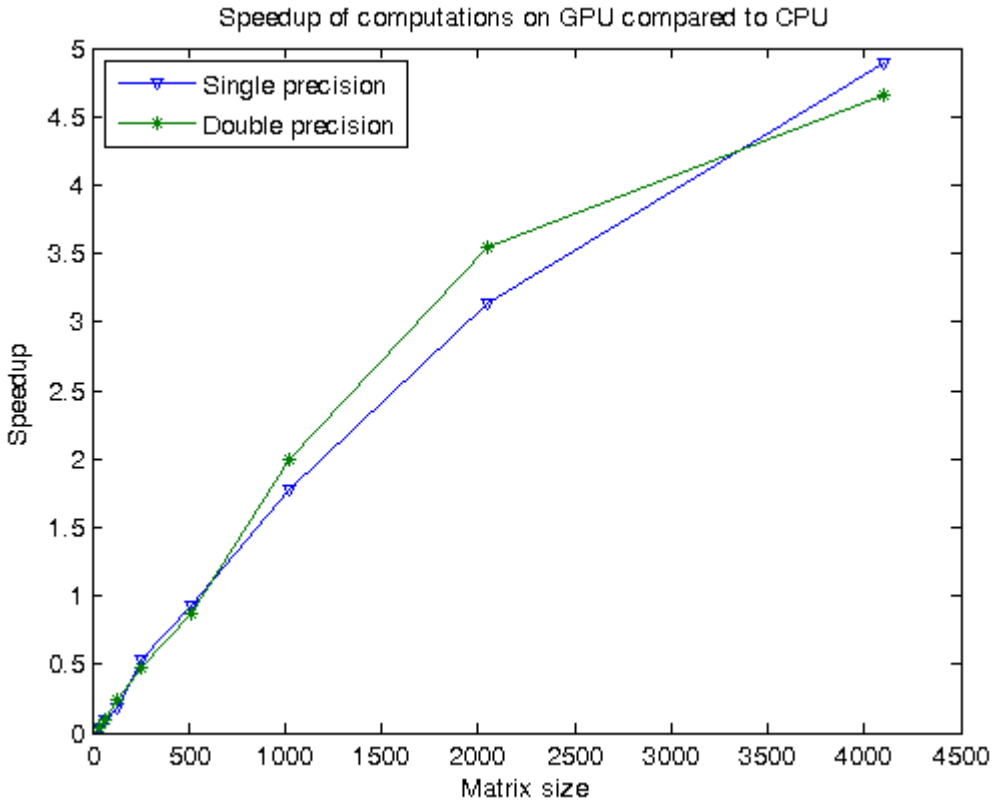
Brief Overview of GPU Computing with MATLAB



MATLAB Parallel Computing with GPUs

http://www.nvidia.com/object/tesla-matlab-accelerations.html

Ускорение MATLAB на GPU

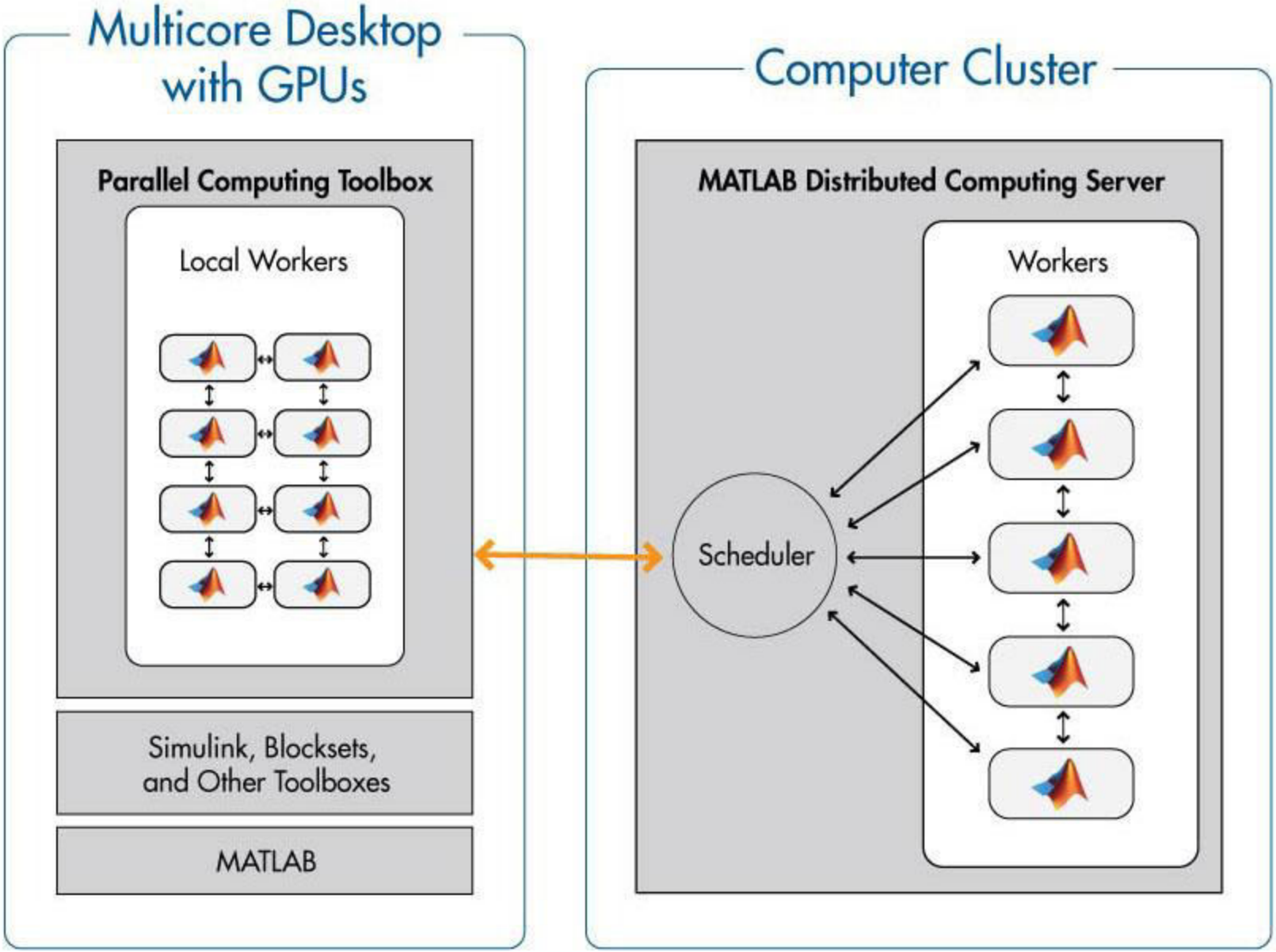


http://www.nvidia.com/object/tesla-matlab-accelerations.html

Parallel Computing Toolbox

lets you solve computationally and data-intensive problems using multicore processors, GPUs, and computer clusters. High-level constructs - parallel for-loops, special array types, and parallelized numerical algorithms let you parallelize MATLAB applications without CUDA or MPI programming. You can use the toolbox with Simulink to run multiple simulations of a model in parallel.
The toolbox provides eight workers (MATLAB computational engines) to execute applications locally on a multicore desktop. Without changing the code, you can run the same application on a computer cluster or grid (using MATLAB Distributed Computing Server).

http://www.mathworks.com/products/parallel-computing/


Multicore Desktop with GPUs
Parallel Computing Toolbox
Local Workers
Simulink, Blocksets, and Other Toolboxes
MATLAB
Computer Cluster
MATLAB Distributed Computing Server
Workers
Scheduler

Вычислительный сервер на базе GPU NVidia Tesla

Программное обеспечение:

- OS Gentoo Linux (kernel 2.6.37)
- Intel C/C++/F90/F95 (v.11.1)
- CUDA Toolkit (v.3.2)
- AccelerEyes Jacket (v.1.3) Multi-GPU License (4 GPU)
- Matlab (7.11.0.584 R2010b 64-bit) Concurrent Network License

Благодарю за внимание!

ИДСТУ СО РАН
Суперкомпьютерный центр коллективного пользования
тел.: 453062, e-mail: super@icc.ru, URL: www.mvs.icc.ru