Fast event generation system using GPU

Junichi Kanzaki (KEK)

ACAT 2013

May 16, 2013, IHEP, Beijing

Motivation

- The mount of LHC data is increasing.
  - $5fb^{-1}$ in 2011
  - $22fb^{-1}$ in 2012
- High statistics data
  -> Reduction of systematic errors becomes essential for good physics measurements.
- Better understandings of backgrounds from QCD multi-jet productions
  -> Fast event generation by changing model parameters

Overview

- Basic tests of HEGET (helicity amplitude library) with simple QED (n-photon) and QCD (n-jet) processes
- Development of GPU versions of VEGAS and BASES/SPRING
- Test of cross section computation and event generation with SM processes
- Summary & Prospect

Bibliography

- QED: K. Hagiwara, J. Kanzaki, N. Okamura, D. Rainwater and T. Stelzer, Eur. Phys. J. C66 (2010) 477, e-print arXiv:0908.4403.
- QCD: K. Hagiwara, J. Kanzaki, N. Okamura, D. Rainwater and T. Stelzer, Eur. Phys. J. C70 (2010) 513, e-print arXiv:0909.5257.
- MC integration (VEGAS & BASES): J. Kanzaki, Eur. Phys. J. C71 (2011) 1559, e-print arXiv:1010.2107.
- SM: submitted to Eur. Phys. J. C,
  e-print arXiv:1305.0708v2
- Event generation (SPRING): in preparation

Our GPU Environment

	C2075	GTX580	GTX285	GTX280	9800GTX
Streaming Processors	448	512	240	←	128
Global Memory	5.4GB	1.5GB	2GB	1GB	500MB
Constant Memory	64KB	64KB	64KB	←	64KB
Shared Memory/block	48KB	48KB	16KB	←	16KB
Registers/block	32768	32768	16384	←	8192
Warp Size	32	32	32	←	32
Clock Rate	1.15GHz	1.54GHz	1.30GHz	←	1.67GHz

- NVDIA GPUs + CUDA
- C2075: Peak floating point performance 1.03 TFLops (single), 515 GFlops (double)

Test with QED and QCD

- Test with simple final states:
  - n-photon production (QED)
  - n-jet production (QCD)
- Development of basic components to calculate cross sections on GPU (CUDA)
  - Amplitude calculation:
    Heget (HELAS in FORTRAN)
  - Phase space generation
  - Random number generation

* Simple event loop program to calculated cross sections

Test with QED and QCD

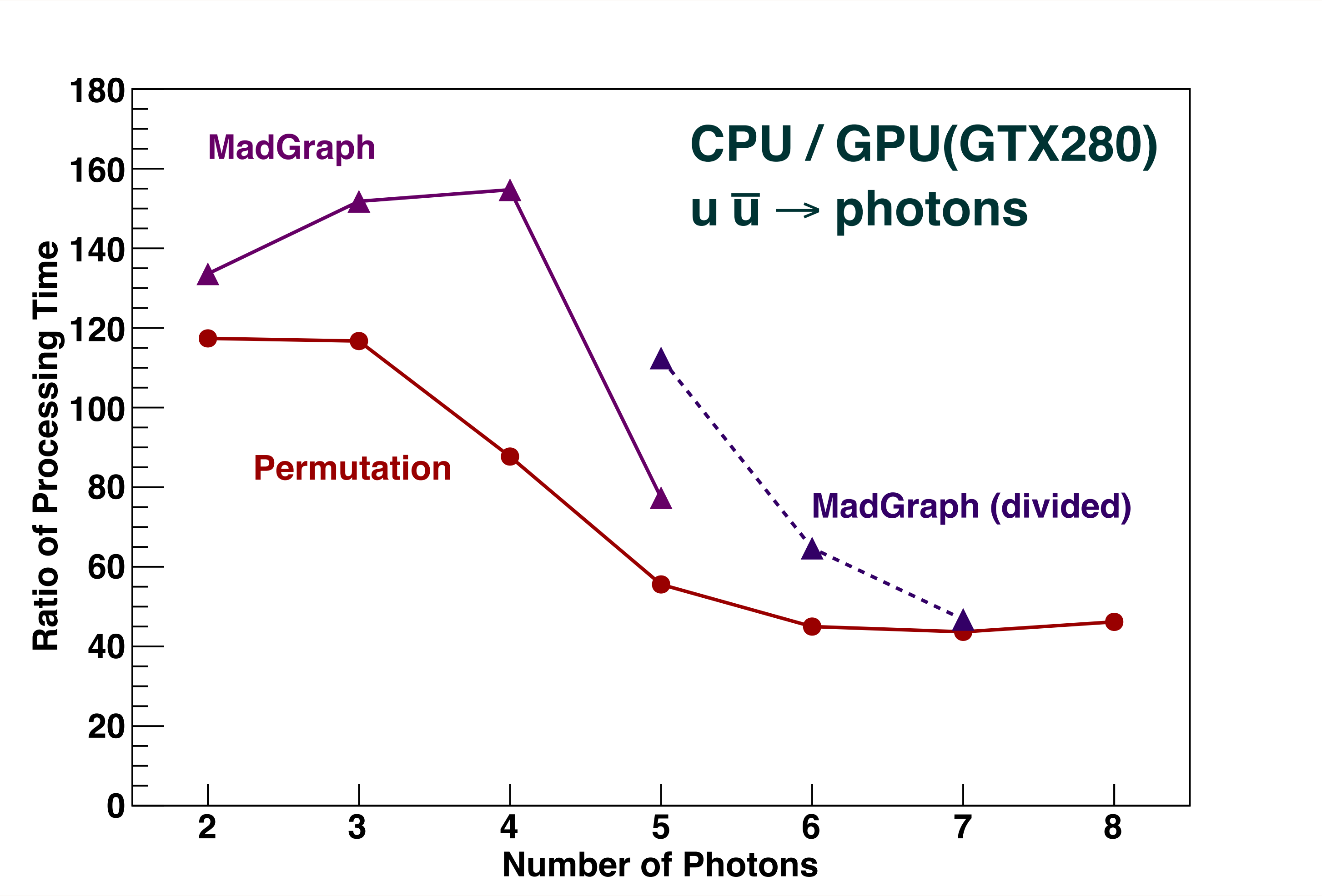
- Check the total cross sections with MadGraph
- Compare process time / loop between CPU and GPU.
- Learn and experience GPU computation:
  - double/single performance ratio
  - parameter dependence of performance: register allocation, no.of threads/block
  - loop unrolling

QED Processes

- uu~ -> n-photons
- Test with two kinds of amplitude:
  - MadGraph amplitude in FORTRAN -> C/CUDA
  - Amplitude by permutation of photons (short)
- Divide a long amplitude program into smaller pieces -> successive kernel calls

# photons	# diagrams = (# photons)!
2	2
3	6
4	24
5	120
6	720
7	5040
8	40320

Event process time ratio (QED)



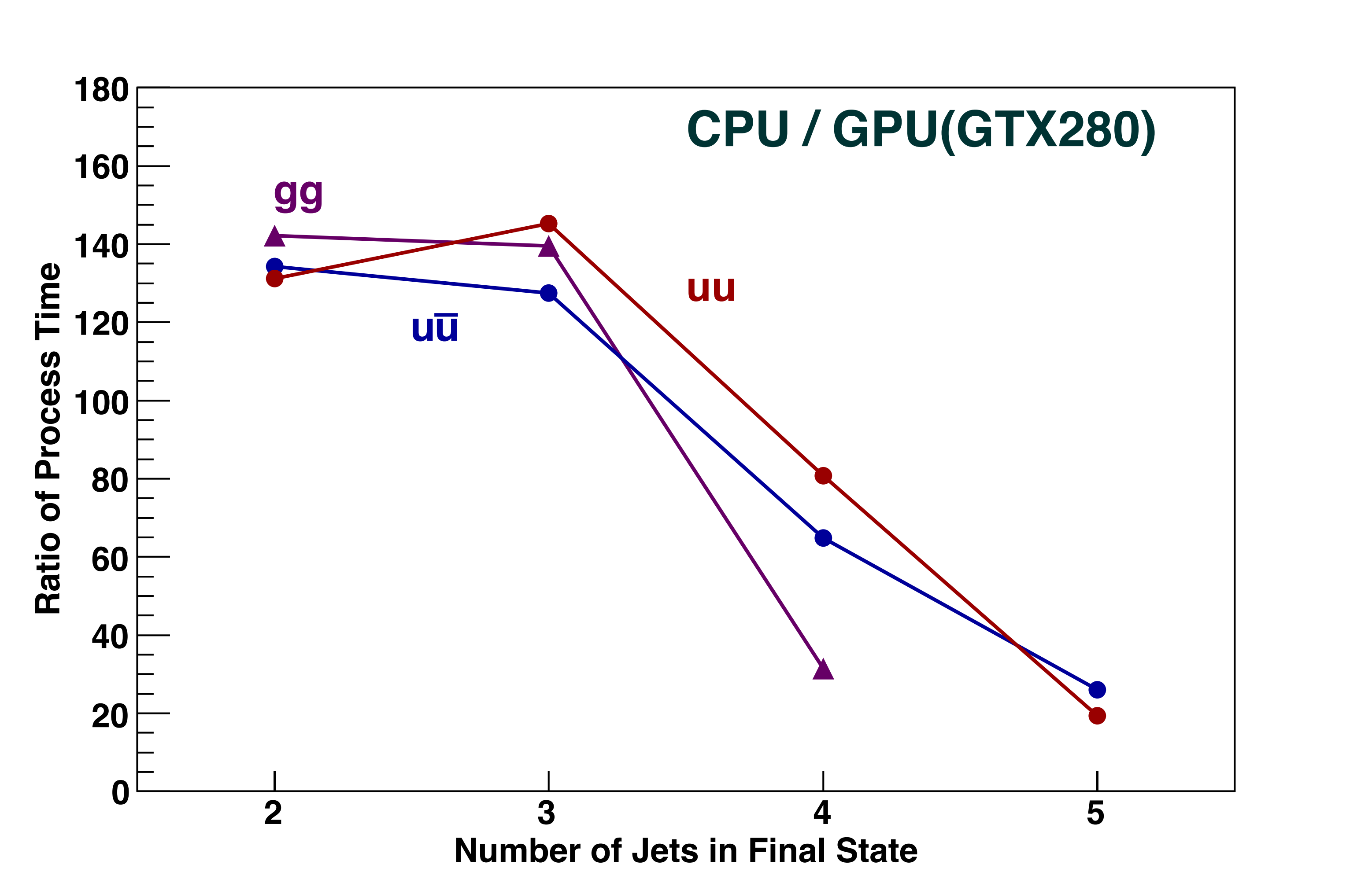
- Large reduction of process time / event loop from CPU to GPU (single precision)

QCD Processes

# final jets	gg → gluons		uu~ → gluons		uu → uu+gluons	
	#diagram	#color	#diagram	#color	#diagram	#color
2	6	6	3	2	2	2
3	45	24	18	6	10	8
4	510	120	159	24	76	40
5	7245	720	1890	120	786	240

- uu~>n-gluons, gg>n-gluons, uu>uu+gluons
- gg>5g: the program cannot be executed on GPU.

Ratio of Process Time (QCD)



- Performance degraded due to the size of amplitude and color factor multiplications.

Monte Carlo integration on GPU

- For the practical event generation on GPU
  -> GPU versions of BASES/SPRING
- Application of GPU to MC integration:
  each GPU thread evaluates function value at each space point

- Test of BASES programs using SM processes with decaying massive particles.
- Compare total process time of original FORTRAN on CPU and CUDA on GPU, and cross sections between MG5 and BASES (CPU and GPU).

SM Processes

- Decay of all massive particles:
  W>l(e, μ)ν, Z->ll (e, μ), t->W(lν)b,
  H->ττ
- Automatic conversion of MadGraph amplitude matrix.f -> CUDA functions (MG2CUDA):
- We fixed the kernel parameters:
  No. of register=64, the thread block size = 256
- Double precision computations

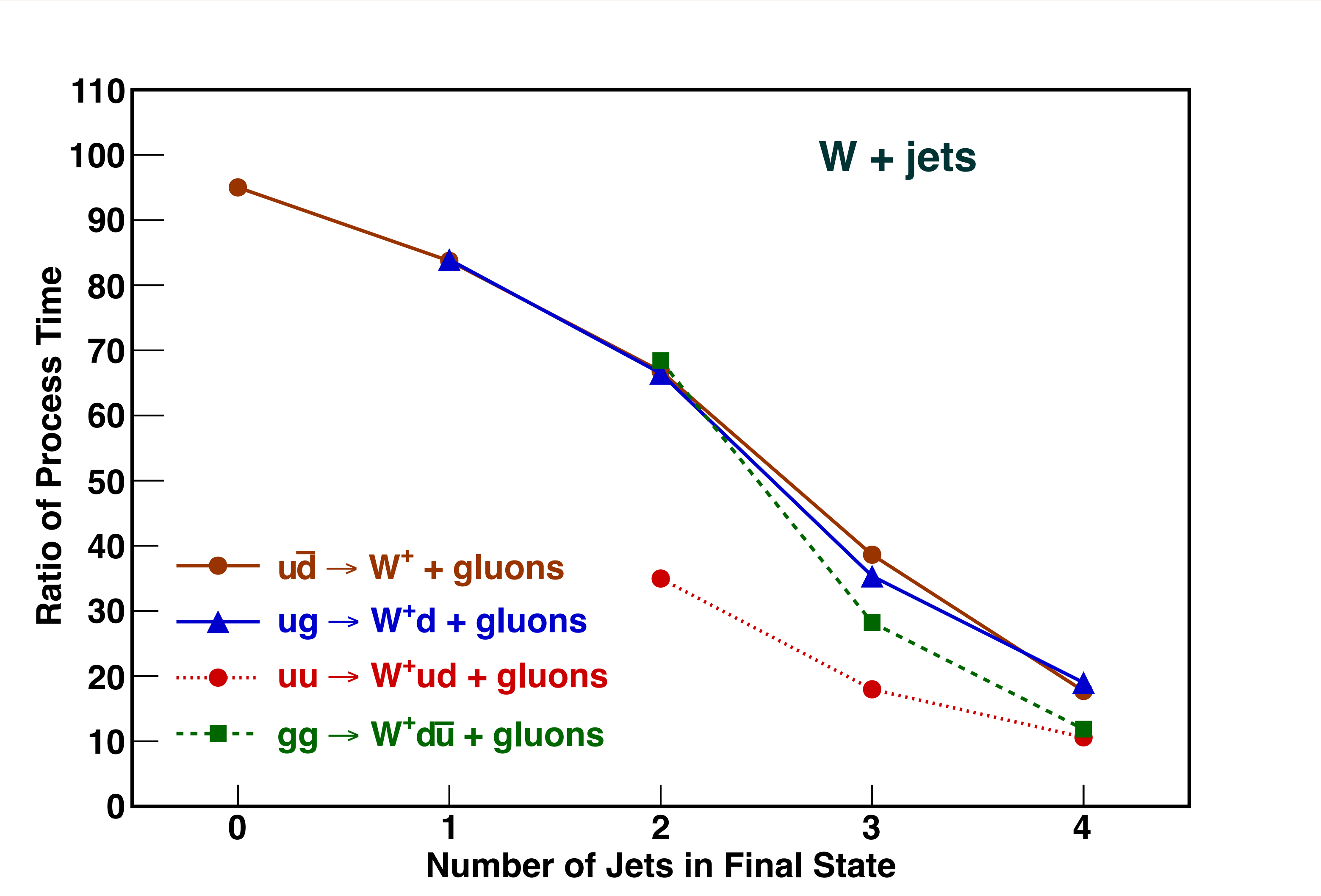
SM Processes

- W, Z + up to 4jets:
  - ud~>W^+, ug>W^+d, uu>W^+ud, gg->W^+du~
  - uu~>Z, ug>Zu, uu>Zuu, gg>Zuu~
- WW, WZ, WW + up to 3jets:
  - uu~>W^+W^-, ug>W^+W^-u, uu>W^+W^-uu, uu>W^+W^+dd, gg->W^+W^-uu~
  - ud~>W^+Z, ug>W^+Zd, uu>W^+Zud, gg>W^+Zdu~
  - uu~>ZZ, ug>ZZd, uu->ZZuu, gg>WWuu~
- tt~+up to 3jets: uu~>tt~, ug>tt~u, uu>tt~uu, gg>tt~

SM Processes (contn'd)

- HW,HZ+up to 3jets:
  - ud~>HW^+, ug>HW^+d, uu>HW^+ud, gg>HW^+du~
  - uu~>HZ, ug>HZu, uu>HZuu, gg>HZuu~
- Http+2jets: uu~>Htt~, ug>Htt~u, uu>Htt~uu, gg>Htt~
- H(WBF)+2jets: ud>Hud, uu>Huu, ug>Hudd~, gg>Huu~dd~
- HH+up to 3jets: ud->HHud, uu->HHuu
- HHH+up to 2jets: ud->HHHud, uu->HHHuu

Ratio of Total Integration Time



- Comparison of total execution time with double precision.

Event Generation by SPRING

- Generate unweighted events by BASES results
- One thread generates one event in a certain hyper-cell of multi-dimension space (acceptance-rejection):
  -> the most inefficient hyper-cell determines the total process time
- Iterative reuse of threads:
  threads that have finished event generation can be assigned to inefficient hyper-cell at the next iteration
  -> improves total performance

SPRING performance

- Total execution time [sec]:
  generation of unweighted 10^6 events

No. of gluons	FORTRAN	GTX580	CPU/GPU
0	9.72	0.346	28
1	43.2	0.768	56
2	4224.8	26.53	160

large improvement is expected for processes with more particles in its final state.

* Preliminary test in single precision

Summary & Prospect

- Program components of cross section computation and event generation based on MadGraph system can be executed on GPU with high performance:
  - GPU version of VEGAS and BAES/SPRING
- Improvement factor of performance can become between 10~100 for total execution time of BASES integration.
- Large improvement of SPRING can be expected.

* Hardware is improving and more applications of GPU to HEP software should be useful.