FEM Integration with Quadrature on GPUs

Matthew Knepley[1], Karl Rupp[2], Andy R. Terrel[3]

rupp@iue.tuwien.ac.at
http://karlrupp.net/

[1] University of Chicago, USA
[2] Vienna University of Technology, Austria
[3] Continuum Analytics, USA

March 18th, 2015

Introduction

Finite Element Method

Several basis functions per element

Evaluation of integrals on each element

General Weak Form

Residual formulation for test function ϕ

$$\int_{\Omega} \phi \cdot f_0(u, \nabla u) + \nabla\phi : \mathbf{f}_1(u, \nabla u) = 0.$$

Examples

Laplace: $f_0 \equiv 0$, $\mathbf{f}_1 \equiv \nabla u$

Poisson: $f_0 \equiv g$, $\mathbf{f}_1 \equiv \nabla u$

Introduction

$$\int_\Omega \phi \cdot f_0(u, \nabla u) + \nabla\phi : \mathbf{f}_1(u, \nabla u) = 0.$$

Element-Wise General Weak Form

Evaluation using quadrature

$$\sum_e \mathcal{E}_e^T \left[B^T W f_0(u^q, \nabla u^q) + \sum_k D_k^T W \mathbf{f}_1^k(u^q, \nabla u^q) \right] = 0$$

$\mathcal{E}$ ... global vector

W ... quadrature weights

B, D_k ... reduction operations for global basis coefficients

Parallelization Options

Across elements

Quadrature points

Basis functions

Parallelization Across Elements

- Large memory per thread
- Synchronizations with neighbor elements
- [Cecka et al. 2011; Taylor et al. 2008; Williams 2012]

Parallelization per Quadrature Point

- No memory overhead
- Too many synchronizations

Parallelization via Basis Functions

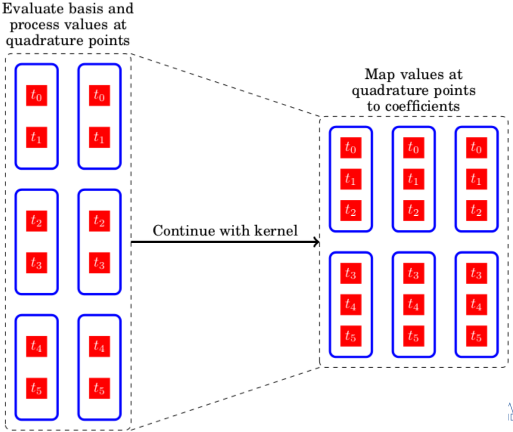
- Very little local memory
- Repeated loads of coefficients from global memory
- [Dabrowski et al. 2008]

Thread Block Works on Multiple Elements

Number of quadrature points N_q

Number of basis functions N_b

Minimum number of elements $\mathrm{LCM}(N_\mathrm{q}, N_\mathrm{b})$



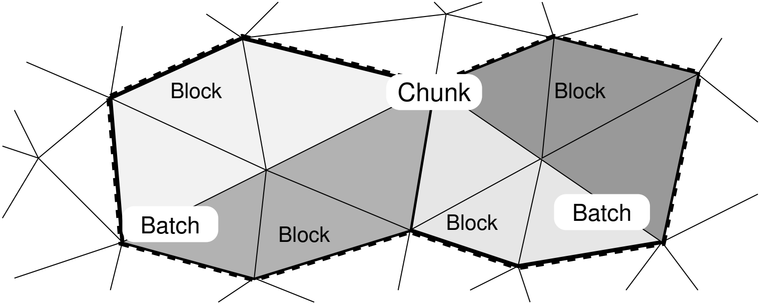
New Algorithm

High Level Decomposition

Chunks - Cells processed by each thread workgroup

Batches - Cells processed with one thread transposition

Blocks - Smallest unit of execution



Benchmark

OpenCL-enabled Hardware

- NVIDIA GTX 470
- NVIDIA GTX 580
- NVIDIA Tesla K20m
- AMD FirePro W9100
- (AMD A10-5800K)

Comparisons

- Single vs. double precision
- 2D vs. 3D

Invariants

- Variable coefficients
- First-order FEM
- Poisson equation

Choice of Block and Batch Numbers

NVIDIA GTX 470

Performance in GFLOPs/sec

Actual choice not very sensitive

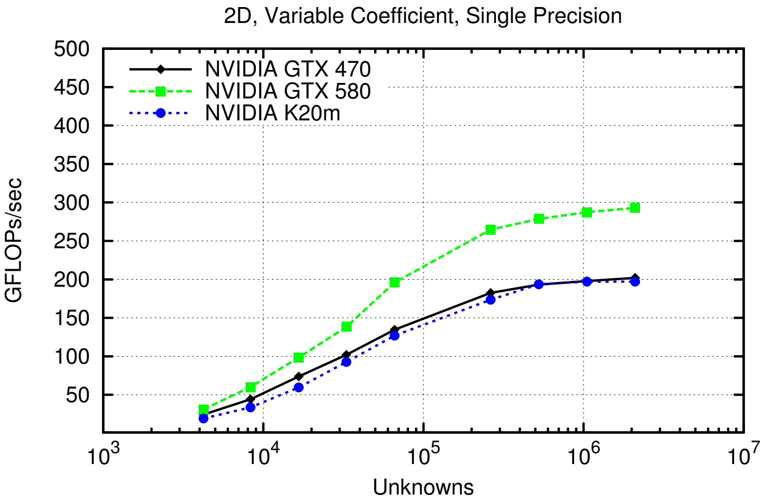
	Batches					
Blocks	16	20	24	28	32	36
4	113	120	118	122	**137**	119
8	109	116	113	120	108	117
12	102	112	110	109	115	113
16	108	100	99	111	130	106

(2D triangular mesh, variable coefficients, single precision, NVIDIA GTX 470)

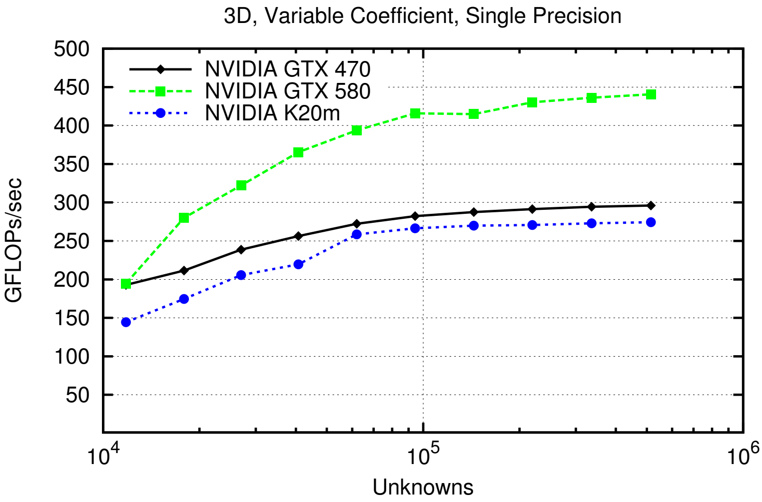
PETSc SNES ex12:

```
./ex12 -petscspace_order 1 -run_type perf -variable_coefficient field
-refinement_limit 0.00001 -show_solution false -petscfe_type opencl
-petscfe_num_blocks 4 -petscfe_num_batches 16
```

Benchmark

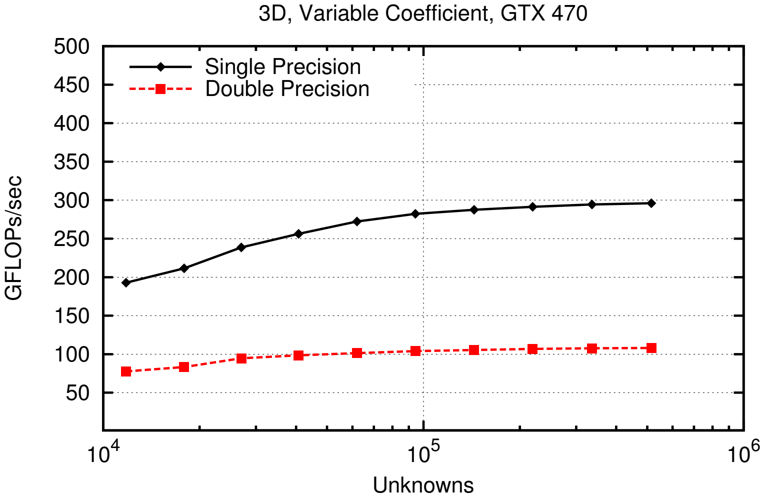

2D, Variable Coefficient, Single Precision
NVIDIA GTX 470
NVIDIA GTX 580
NVIDIA K20m
GFLOPs/sec
500
450
400
350
300
250
200
150
100
50
10^3
10^4
10^5
10^6
10^7
Unknowns

Benchmark

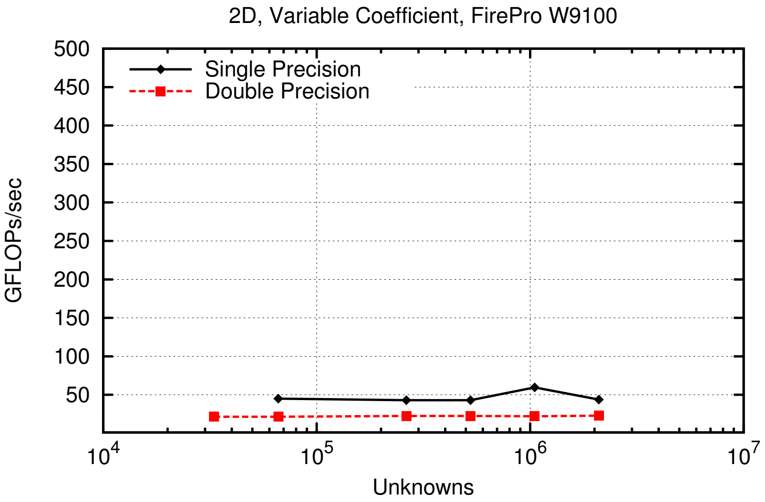


3D, Variable Coefficient, Single Precision
500
450
400
350
300
250
200
150
100
50
GFLOPs/sec
NVIDIA GTX 470
NVIDIA GTX 580
NVIDIA K20m
10^4
10^5
10^6
Unknowns

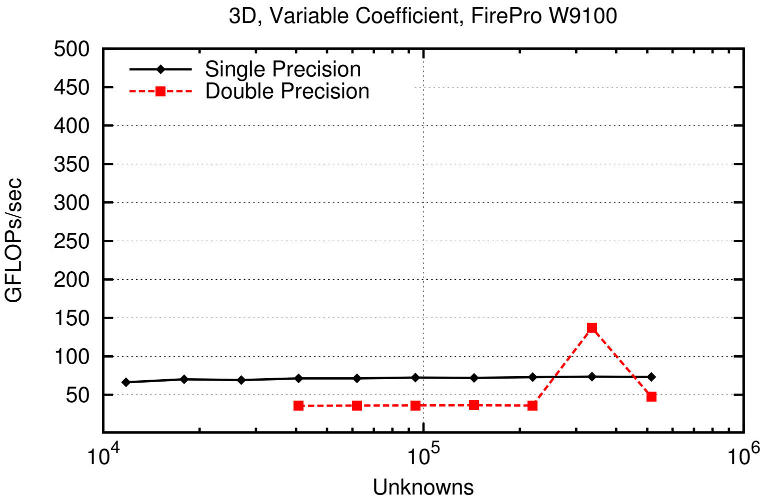
Benchmark


3D, Variable Coefficient, GTX 470
Single Precision
Double Precision
GFLOPs/sec
500
450
400
350
300
250
200
150
100
50
10^4
10^5
10^6
Unknowns

Benchmark


2D, Variable Coefficient, FirePro W9100
GFLOPs/sec
500
450
400
350
300
250
200
150
100
50
Single Precision
Double Precision
10^4
10^5
10^6
10^7
Unknowns

Benchmark



3D, Variable Coefficient, FirePro W9100
Single Precision
Double Precision
GFLOPs/sec
500
450
400
350
300
250
200
150
100
50
10^4
10^5
10^6
Unknowns

Performance Modeling

Limiting Factor?

GTX 470: 134 GB/sec memory bandwidth (theoretical)

GTX 470: 1088 GFLOPs/sec peak (theoretical)

Arithmetic Intensity

Count FLOPs and bytes loaded/stored

$$\beta = \frac{\left[(2+(2+2d)d)N_{\mathrm{bt}}N_{\mathrm{q}} + 2dN_{\mathrm{comp}}N_{\mathrm{q}} + (2+2d)dN_{\mathrm{q}}N_{\mathrm{bt}}\right]N_{\mathrm{bs}}N_{\mathrm{bl}}}{4N_{\mathrm{t}}\left((d^2+1) + N_{\mathrm{bt}} + (d+1)N_{\mathrm{q}}\right)}$$

2D Mesh, First-Order FEM, Single Precision

$\beta = 41/22 \approx 2$ FLOPs/Byte

GTX 470: $134 \times 41/22 = 250$ GFLOPs possible

GTX 470: 200 GFLOPs achieved (80 percent, cf. STREAM benchmark)

Summary and Conclusion

FEM Quadrature on GPUs

“Matrix-Free”

Higher arithmetic intensity

Performance Results

Good performance on NVIDIA GPUs and AMD APUs

5x improvements for discrete AMD GPUs desired

Performance Modeling

Performance limited by memory bandwidth

Excellent prediction accuracy

Reproducibility

PETSc, SNES tutorial, ex12