

Homework #4

cpe 631

Kyle Ray

February 14, 2018

Contents

[Purpose 2](#_Toc506564623)

[Compilation 2](#_Toc506564624)

[Problem 1 2](#_Toc506564625)

[Problem 2 4](#_Toc506564626)

[Appendix A Software Blocking Experiment Sim Results 6](#_Toc506564627)

[Mem-Config 6](#_Toc506564628)

[No Block 7](#_Toc506564629)

[Software Block Size 64 7](#_Toc506564630)

[Software Block Size 128 8](#_Toc506564631)

[Appendix B Daxpy Simulation Results 9](#_Toc506564632)

# Purpose

Become familiar with Multi2Sim – a simulation framework for CPU-GPU heterogenous computing platforms.

# Compilation

Each folder contains a make file which has the following options

1. make: Compiles and links the source for the application
2. make clean: Removes reports, text files, and compiled code
3. make run: Performs the m2s simulation for the problem and pipes the output to text files.

# Problem 1

Exploring the effectiveness of the blocking optimization for matrix multiplication. Blocking optimization theoretically should increase the performance of the system by increasing the amount of data reuse from the cache which in turn will lower the number of fetches back to main memory.

Architecture Setup:

1. Default x86 processor configuration
2. Modified memory hierarchy with the following:
   1. Split Instruction L1I Cache
      1. 4 KB, 64-byte blocks, 4-way set associativity, 4 CC latency, LRU
   2. Split Data L1D 4KB Cache with 4-way set associativity and 64-byte blocks
      1. 4 KB, 64-byte blocks, 4-way set associativity, 4 CC latency, LRU
   3. Unified L2U 23 KB Cache
      1. 32 KB, 64-byte blocks, 4-way set associativity, 10 CC latency, LRU

The application used in the experiment is a serial matrix multiplication program written in the C language and compiled with gcc. Square matrix multiplication with matrix sizes 256x256 were used as well as blocking sizes 16, 32, 64, and 128 to test the performance of the architecture mentioned above. The results are tabulated below.

According to some experiments that Intel has performed with blocking optimization they were seeing that block size 16 provided the best improvement for performance in a matrix multiplication of 4096x4096. The data presented below shows a very similar outcome.

**Table 1. Execution Times for Various Blocking Sizes (Matrix Multiplication 256x256)**

|  |  |
| --- | --- |
| Block Size | Execution Time (ns) |
| 0 | 8619930491.00 |
| 16 | 3207318191.50 |
| 32 | 4412285429.50 |
| 64 | 5708939374.50 |
| 128 | 7254412656.50 |

**Figure 1. Execution Times for Various Blocking Sizes (Matrix Multiplication 256x256)**

It does seem that software blocking optimization improves the performance of the application, and in this case a block size of 16 provides the best improvement.

# Problem 2

Evaluate the effectiveness of loop unrolling regarding execution time. Architecture setup is set to the Multi2Sim default setup. A C program performing a DAXPY kernel (aX + Y = Z) where X,Y,and Z are arrays of n elements and a is a constant 3.0. The program was ran with n = 2048 and the execution times were recorded for no unrolling up to four loop unrolls. The loop unrolls were executed in powers of 2 i.e. 2, 4, 8, and 16. The data from the runs is tabulated below.

**Table 2. Execution Times for Loop Unrolls (DAXPY with 2048 Elements)**

|  |  |
| --- | --- |
| Loop Unrolls | Execution Time (ns) |
| 0 | 187712.00 |
| 2 | 176189.00 |
| 4 | 178131.50 |
| 8 | 172611.00 |
| 16 | 171303.00 |

**Figure 2. Execution Time for Loop Unrolls (DAXPY with 2048 Elements)**

From the data above, we can see that loop unrolling in most cases does help the overall execution time. This makes sense because it is effectively cutting out many control instructions by working on more items in a single iteration. Unfortunately, it does look as if the benefit to loop unrolling is quickly approaching a limit and there will likely be an insignificant change if the technique is pursued further.

# Appendix A Software Blocking Experiment Sim Results

## Mem-Config

[CacheGeometry geo-l1]

Sets = 16

Assoc = 4

BlockSize = 64

Latency = 4

Policy = LRU

Ports = 2

[CacheGeometry geo-l2]

Sets = 128

Assoc = 4

BlockSize = 64

Latency = 10

Policy = LRU

Ports = 2

[Module mod-l1-0]

Type = Cache

Geometry = geo-l1

LowNetwork = net-l1-l2

LowModules = mod-l2-0

[Module mod-l2-0]

Type = Cache

Geometry = geo-l2

HighNetwork = net-l1-l2

LowNetwork = net-l2-mm

LowModules = mod-mm

[Module mod-mm]

Type = MainMemory

BlockSize = 256

Latency = 200

HighNetwork = net-l2-mm

[Network net-l1-l2]

DefaultInputBufferSize = 1024

DefaultOutputBufferSize = 1024

DefaultBandwidth = 256

[Network net-l2-mm]

DefaultInputBufferSize = 1024

DefaultOutputBufferSize = 1024

DefaultBandwidth = 256

[Entry core-0]

Arch = x86

Core = 0

Thread = 0

DataModule = mod-l1-0

InstModule = mod-l1-0

## No Block

; Multi2Sim 4.2 - A Simulation Framework for CPU-GPU Heterogeneous Computing

; Please use command 'm2s --help' for a list of command-line options.

; Simulation alpha-numeric ID: Phv8N

;

; Simulation Statistics Summary

;

[ General ]

RealTime = 7713.34 [s]

SimEnd = ContextsFinished

SimTime = 8619930500.00 [ns]

Frequency = 2000 [MHz]

Cycles = 17239861001

[ x86 ]

RealTime = 7713.33 [s]

Instructions = 921020048

InstructionsPerSecond = 119406

Contexts = 2

Memory = 19845120

FastForwardInstructions = 0

CommittedInstructions = 906150237

CommittedInstructionsPerCycle = 0.05256

CommittedMicroInstructions = 1946771319

CommittedMicroInstructionsPerCycle = 0.1129

BranchPredictionAccuracy = 0.9927

SimTime = 8619930491.00 [ns]

Frequency = 2000 [MHz]

Cycles = 17239860982

CyclesPerSecond = 2235073

## Software Block Size 64

; Multi2Sim 4.2 - A Simulation Framework for CPU-GPU Heterogeneous Computing

; Please use command 'm2s --help' for a list of command-line options.

; Simulation alpha-numeric ID: 4anXf

;

; Simulation Statistics Summary

;

[ General ]

RealTime = 9971.87 [s]

SimEnd = ContextsFinished

SimTime = 5400142058.00 [ns]

Frequency = 2000 [MHz]

Cycles = 10800284117

[ x86 ]

RealTime = 9971.87 [s]

Instructions = 1187938299

InstructionsPerSecond = 119129

Contexts = 2

Memory = 19845120

FastForwardInstructions = 0

CommittedInstructions = 1132177884

CommittedInstructionsPerCycle = 0.1048

CommittedMicroInstructions = 2351107992

CommittedMicroInstructionsPerCycle = 0.2177

BranchPredictionAccuracy = 0.9766

SimTime = 5400142030.50 [ns]

Frequency = 2000 [MHz]

Cycles = 10800284061

CyclesPerSecond = 1083075

## Software Block Size 128

; Multi2Sim 4.2 - A Simulation Framework for CPU-GPU Heterogeneous Computing

; Please use command 'm2s --help' for a list of command-line options.

; Simulation alpha-numeric ID: TcLTE

;

; Simulation Statistics Summary

;

[ General ]

RealTime = 7157.74 [s]

SimEnd = ContextsFinished

SimTime = 7254412684.00 [ns]

Frequency = 2000 [MHz]

Cycles = 14508825369

[ x86 ]

RealTime = 7157.74 [s]

Instructions = 1019704025

InstructionsPerSecond = 142462

Contexts = 2

Memory = 19845120

FastForwardInstructions = 0

CommittedInstructions = 994047515

CommittedInstructionsPerCycle = 0.06851

CommittedMicroInstructions = 2071445466

CommittedMicroInstructionsPerCycle = 0.1428

BranchPredictionAccuracy = 0.9877

SimTime = 7254412656.50 [ns]

Frequency = 2000 [MHz]

Cycles = 14508825313

CyclesPerSecond = 2027012

# Appendix B Daxpy Simulation Results

; Multi2Sim 4.2 - A Simulation Framework for CPU-GPU Heterogeneous Computing

; Please use command 'm2s --help' for a list of command-line options.

; Simulation alpha-numeric ID: AE5Fq

;

; Simulation Statistics Summary

;

[ General ]

RealTime = 0.43 [s]

SimEnd = ContextsFinished

SimTime = 187714.00 [ns]

Frequency = 2000 [MHz]

Cycles = 375429

[ x86 ]

RealTime = 0.43 [s]

Instructions = 138331

InstructionsPerSecond = 320484

Contexts = 1

Memory = 9121792

FastForwardInstructions = 0

CommittedInstructions = 106744

CommittedInstructionsPerCycle = 0.2843

CommittedMicroInstructions = 229683

CommittedMicroInstructionsPerCycle = 0.6118

BranchPredictionAccuracy = 0.9132

SimTime = 187712.00 [ns]

Frequency = 2000 [MHz]

Cycles = 375424

CyclesPerSecond = 869778

; Multi2Sim 4.2 - A Simulation Framework for CPU-GPU Heterogeneous Computing

; Please use command 'm2s --help' for a list of command-line options.

; Simulation alpha-numeric ID: AFV3I

;

; Simulation Statistics Summary

;

[ General ]

RealTime = 0.43 [s]

SimEnd = ContextsFinished

SimTime = 176431.00 [ns]

Frequency = 2000 [MHz]

Cycles = 352863

[ x86 ]

RealTime = 0.43 [s]

Instructions = 136934

InstructionsPerSecond = 316871

Contexts = 1

Memory = 9121792

FastForwardInstructions = 0

CommittedInstructions = 105740

CommittedInstructionsPerCycle = 0.3001

CommittedMicroInstructions = 222540

CommittedMicroInstructionsPerCycle = 0.6315

BranchPredictionAccuracy = 0.8983

SimTime = 176189.00 [ns]

Frequency = 2000 [MHz]

Cycles = 352378

CyclesPerSecond = 815418

; Multi2Sim 4.2 - A Simulation Framework for CPU-GPU Heterogeneous Computing

; Please use command 'm2s --help' for a list of command-line options.

; Simulation alpha-numeric ID: AHKDu

;

; Simulation Statistics Summary

;

[ General ]

RealTime = 0.40 [s]

SimEnd = ContextsFinished

SimTime = 178134.00 [ns]

Frequency = 2000 [MHz]

Cycles = 356269

[ x86 ]

RealTime = 0.40 [s]

Instructions = 136709

InstructionsPerSecond = 338955

Contexts = 1

Memory = 9121792

FastForwardInstructions = 0

CommittedInstructions = 105228

CommittedInstructionsPerCycle = 0.2954

CommittedMicroInstructions = 218956

CommittedMicroInstructionsPerCycle = 0.6146

BranchPredictionAccuracy = 0.8885

SimTime = 178131.50 [ns]

Frequency = 2000 [MHz]

Cycles = 356263

CyclesPerSecond = 883315

; Multi2Sim 4.2 - A Simulation Framework for CPU-GPU Heterogeneous Computing

; Please use command 'm2s --help' for a list of command-line options.

; Simulation alpha-numeric ID: Hlagz

;

; Simulation Statistics Summary

;

[ General ]

RealTime = 0.39 [s]

SimEnd = ContextsFinished

SimTime = 172611.00 [ns]

Frequency = 2000 [MHz]

Cycles = 345223

[ x86 ]

RealTime = 0.39 [s]

Instructions = 136243

InstructionsPerSecond = 351054

Contexts = 1

Memory = 9121792

FastForwardInstructions = 0

CommittedInstructions = 104972

CommittedInstructionsPerCycle = 0.3041

CommittedMicroInstructions = 217164

CommittedMicroInstructionsPerCycle = 0.6292

BranchPredictionAccuracy = 0.8831

SimTime = 172580.00 [ns]

Frequency = 2000 [MHz]

Cycles = 345160

CyclesPerSecond = 889365

; Multi2Sim 4.2 - A Simulation Framework for CPU-GPU Heterogeneous Computing

; Please use command 'm2s --help' for a list of command-line options.

; Simulation alpha-numeric ID: HmOEF

;

; Simulation Statistics Summary

;

[ General ]

RealTime = 0.38 [s]

SimEnd = ContextsFinished

SimTime = 171303.00 [ns]

Frequency = 2000 [MHz]

Cycles = 342607

[ x86 ]

RealTime = 0.38 [s]

Instructions = 135997

InstructionsPerSecond = 357318

Contexts = 1

Memory = 9121792

FastForwardInstructions = 0

CommittedInstructions = 104849

CommittedInstructionsPerCycle = 0.306

CommittedMicroInstructions = 216271

CommittedMicroInstructionsPerCycle = 0.6313

BranchPredictionAccuracy = 0.8808

SimTime = 171301.00 [ns]

Frequency = 2000 [MHz]

Cycles = 342602

CyclesPerSecond = 900151