Raspberry Pi Performance Benchmarking

Justin Moore
Salish Kootenai College

Overview

- Raspberry Pi Cluster Build
- Performance
- Performance Benchmark Tools
- Tuning
- Analysis
- Conclusion

Raspberry Pi Cluster – Model B

- CPU – Broadcom ARM11 76JZF-S 700MHz
  - Can overclock to 1000MHz
- RAM – 512MB
  - 448MB/64MB – CPU/GPU
- Linux OS
- 10/100 BaseT Ethernet port
- Size of a credit card
- Price - $35

Image: http://commons.wikimedia.org/wiki/File:RaspberryPi.jpg#mediaviewer/File:RaspberryPi.jpg

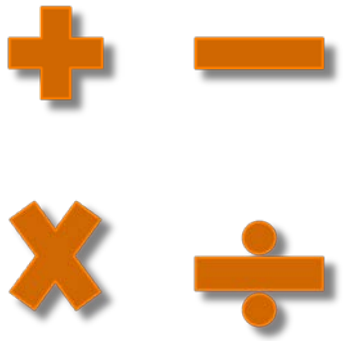
Raspberry Pi Cluster - Setup

- 4 Pi cluster
- USB hub – Manhattan 7 port USB 2.0
- SD card – Kodak 8GB

- Router – Western Digital N750
- NFS mounted external hard drive – 500GB Buffalo Inc.

Performance – Floating Point Operations

- How we measure performance
  - Busy CPU? Speed?
- What is a floating point operation (FLOP)?
  - Arithmetic operation
  - Formats
    - Single
    - Double

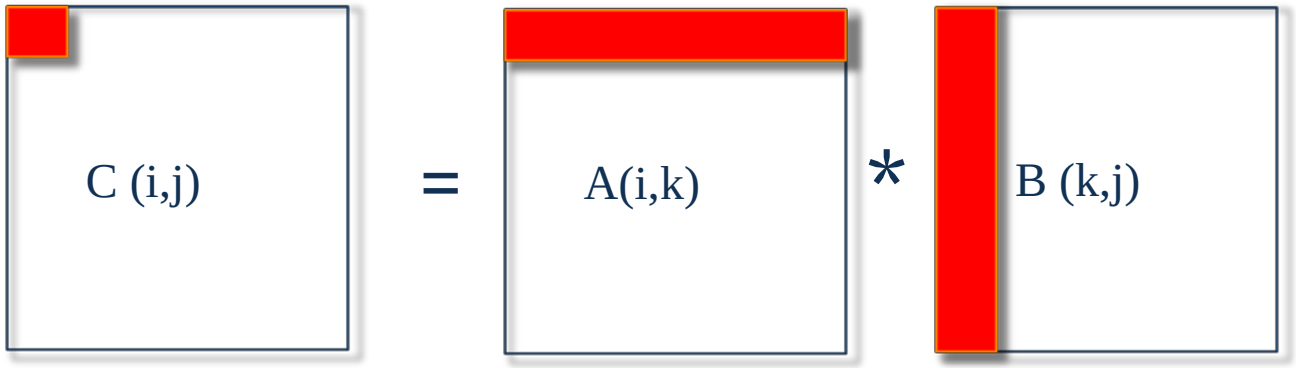


- Performance measurement by FLOPS
- How do we measure FLOPS?
  - General Matrix Multiplication (GEMM)
  - High computational intensity with an increase in matrix size

Performance Benchmarking

- Single Raspberry Pi
  - BLAS - Basic Linear Algebra Subprograms
  - ATLAS - Automatically Tuned Linear Algebra Software
    - Auto tunes BLAS for any system
- Raspberry Pi Cluster
  - MPI - Message Passing Interface
    - Standard API for inter-process communication
    - Facilitates parallel programming
    - MPICH 2-1.4.1p1
  - HPL - High Performance LINPACK
    - Tuned MPI
    - Combined with ATLAS
- Wrote Custom code
  - ATLAS
  - Added parallel capability
    - Compared with HPL

MyGEMM – Naïve Implementation



Where N = Matrix Size

```
For i = 1 to N
  For j = 1 to N
    For k = 1 to N
      C(i,j)=C(i,j)+A(i,k)*B(k,j)
```

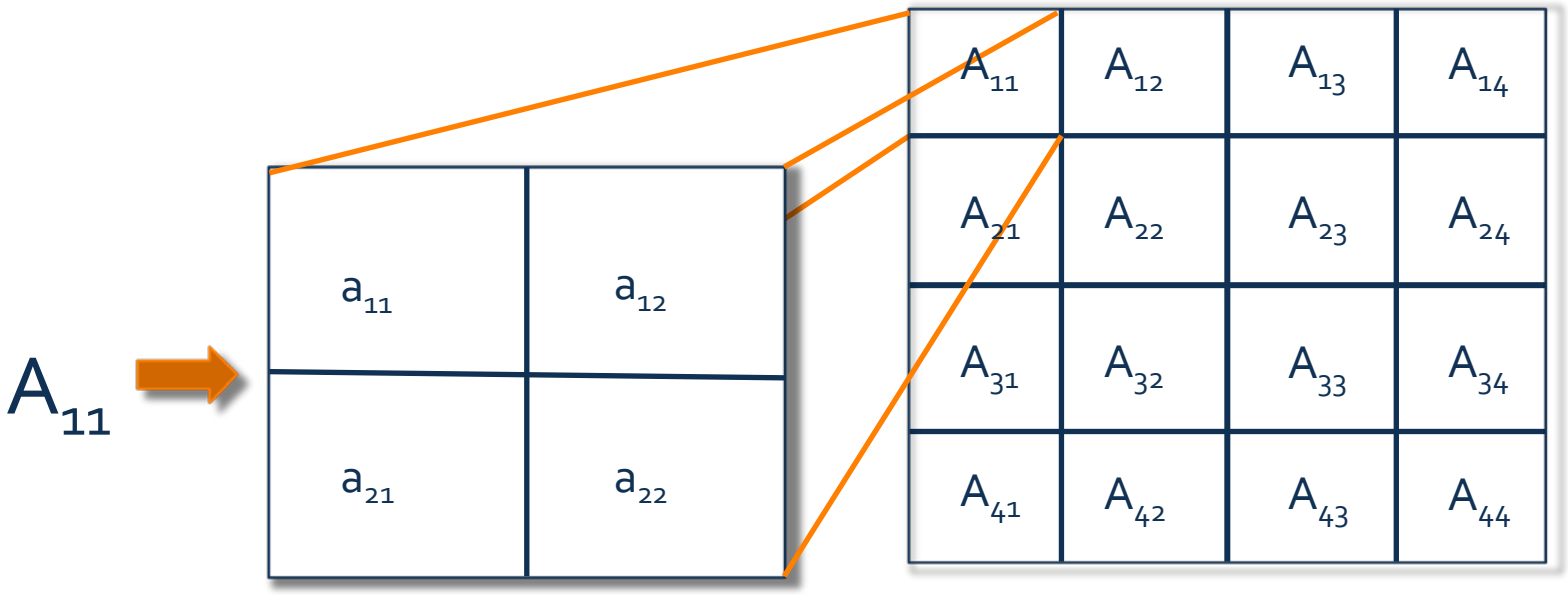
MyGEMM – Naïve Pitfalls

- GEMM – Naïve method inefficient
  - Two whole matrices are loaded in to memory
  - Cache is not used efficiently
    - Strides through the matrix

- If not Naïve, then what?

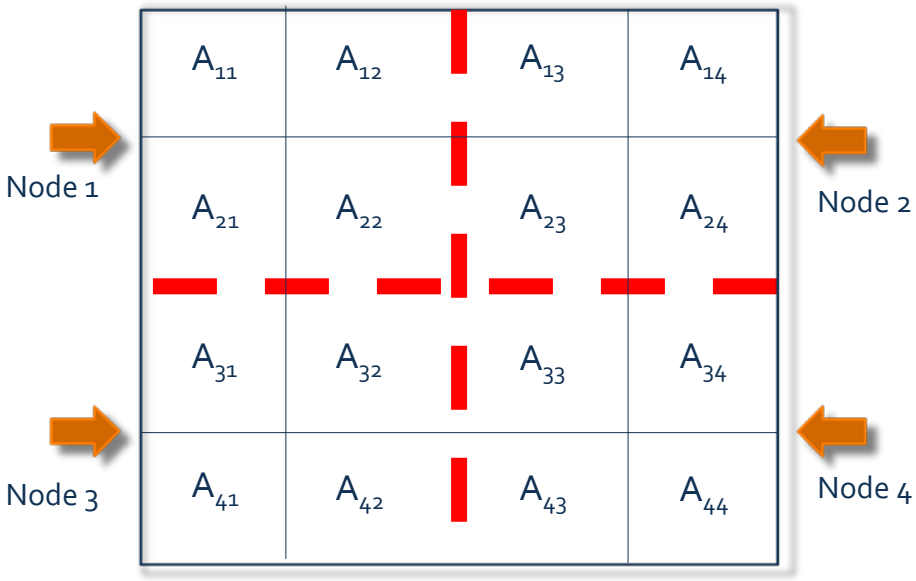
MyGEMM – Software Tuning

- What is block matrix multiplication
  - Matrix is split into smaller matrices/blocks
  - Shrinks matrix size to allow both A & B into fast memory



MyGEMM – Cluster Software Tuning

- How do we distribute the matrix multiplication?
  - MPI used to distribute blocks to nodes



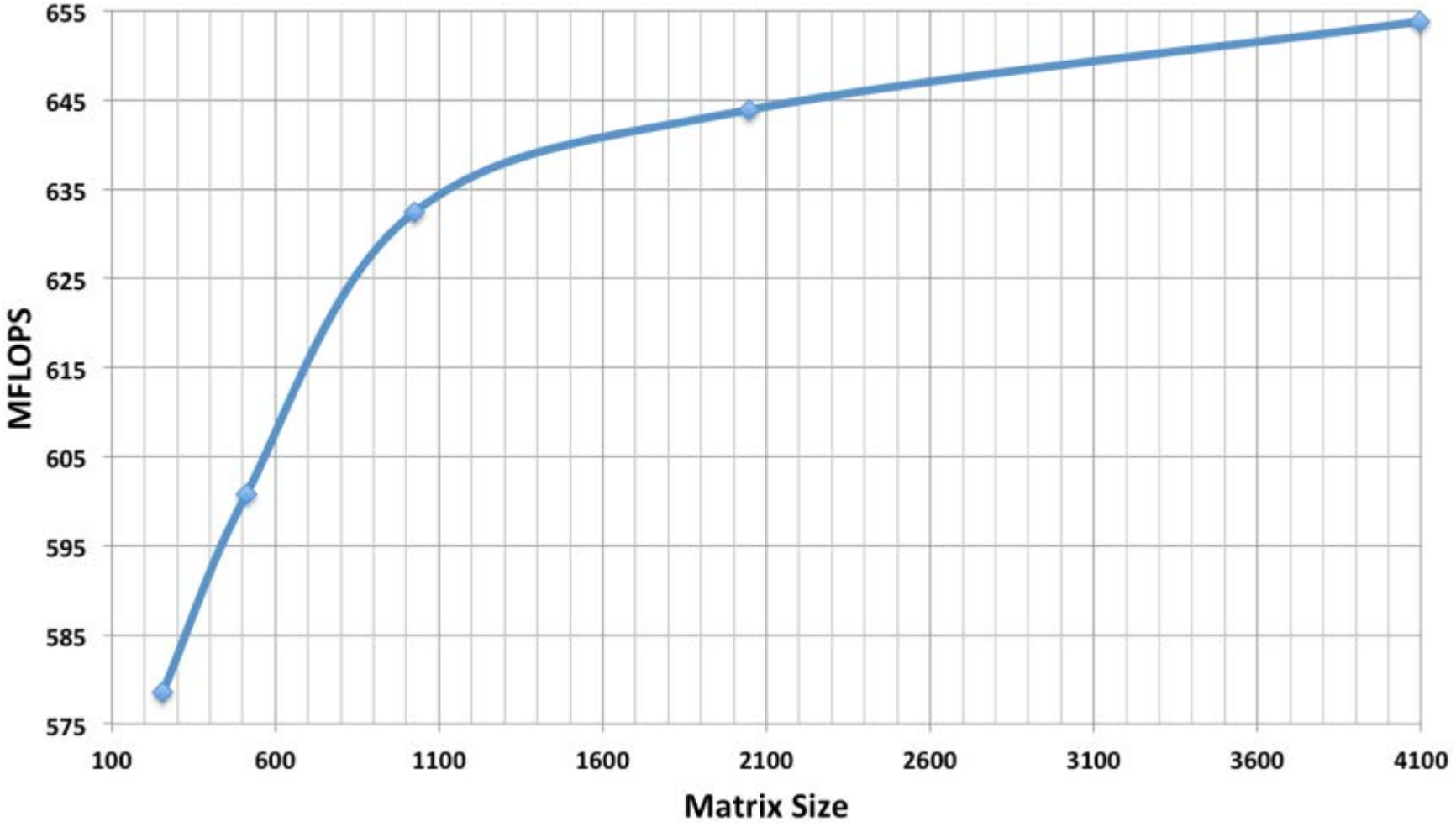
- MyGEMM allows for experimentation on block size

Hardware Tuning

- Raspberry Pi allows CPU overclocking and memory sharing between CPU and GPU
- Memory
  - Memory is shared between CPU and GPU
    - 512MB total onboard memory
    - Up to 496MB can be used for CPU
    - More memory = larger matrices
- CPU Clock
  - Up to 1000 MHz from 700MHz

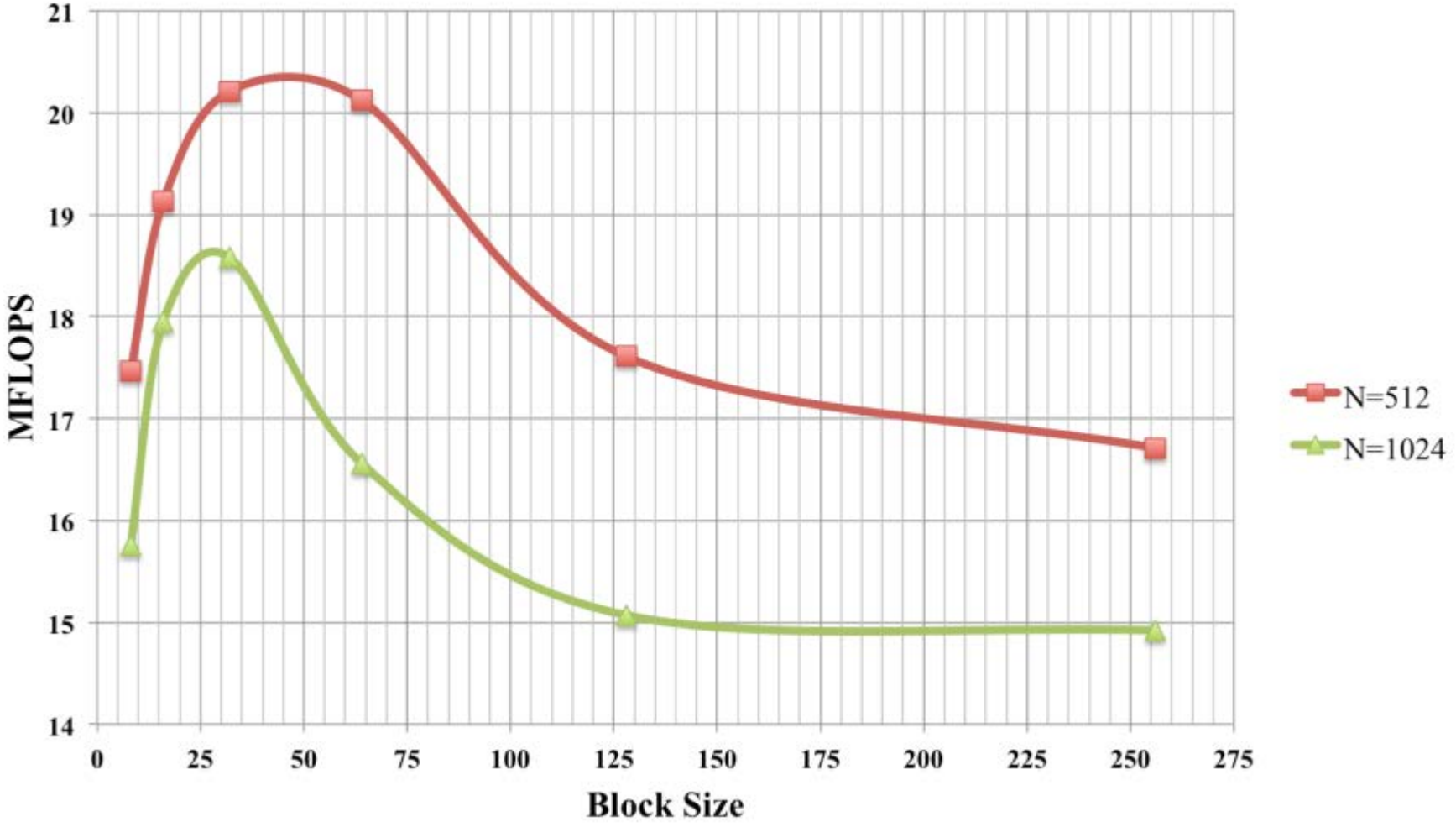
Computational Intensity

Single Precision, Single Node MyGemm w/ ATLAS



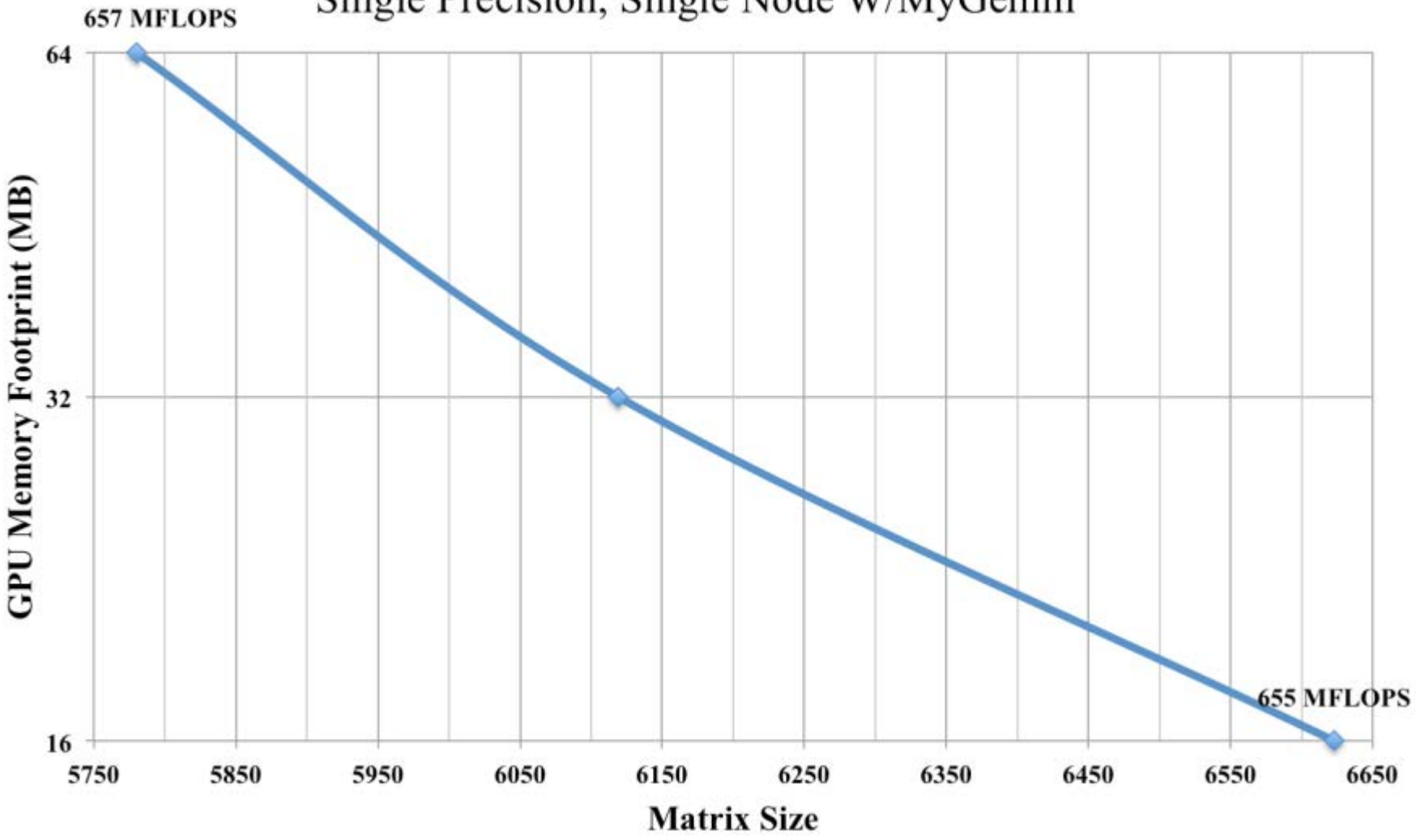
Blocking Efficiency

Single Precision, Single Node W/MyGemm



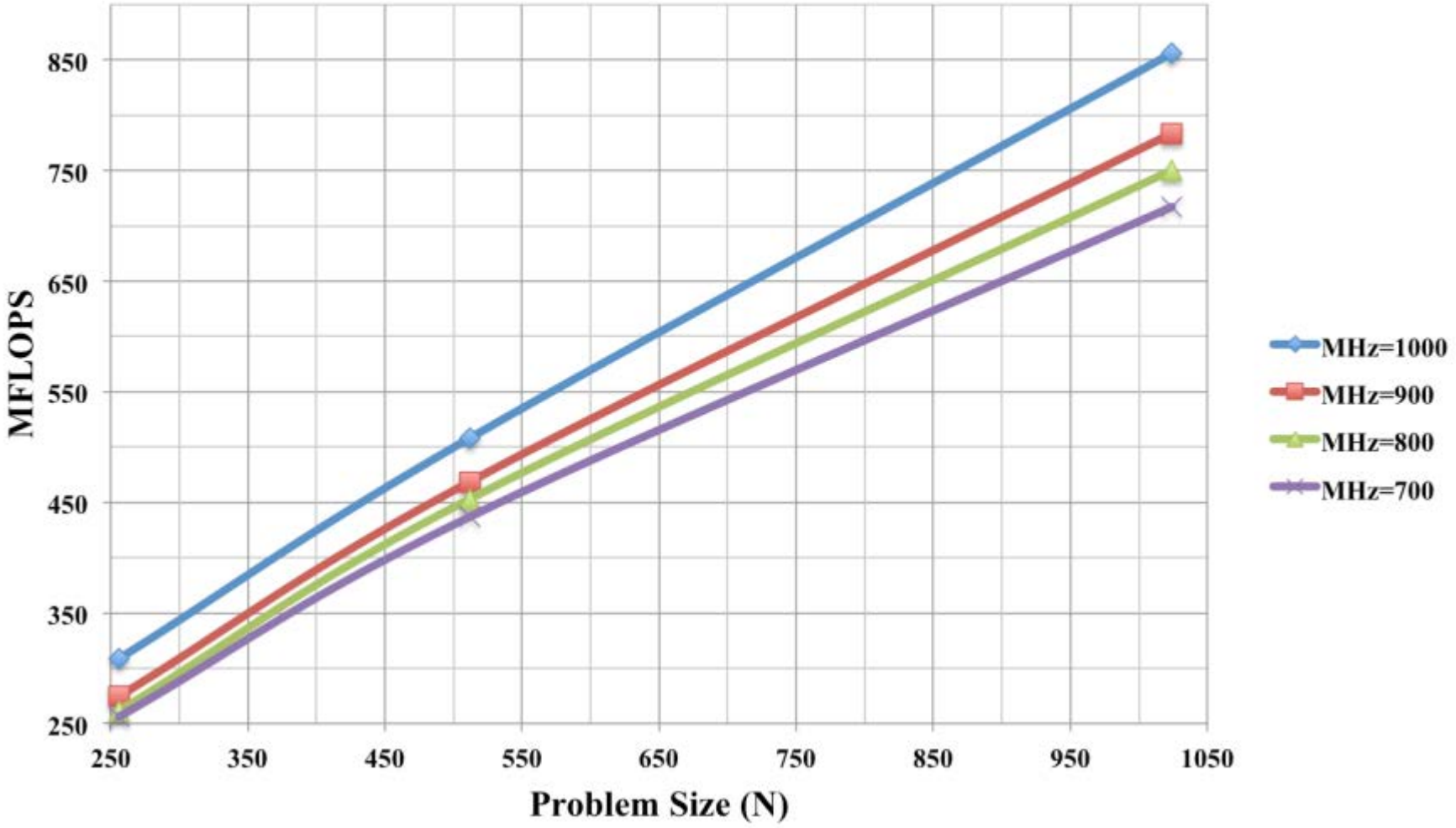
Variation of GPU Footprint

Single Precision, Single Node W/MyGemm



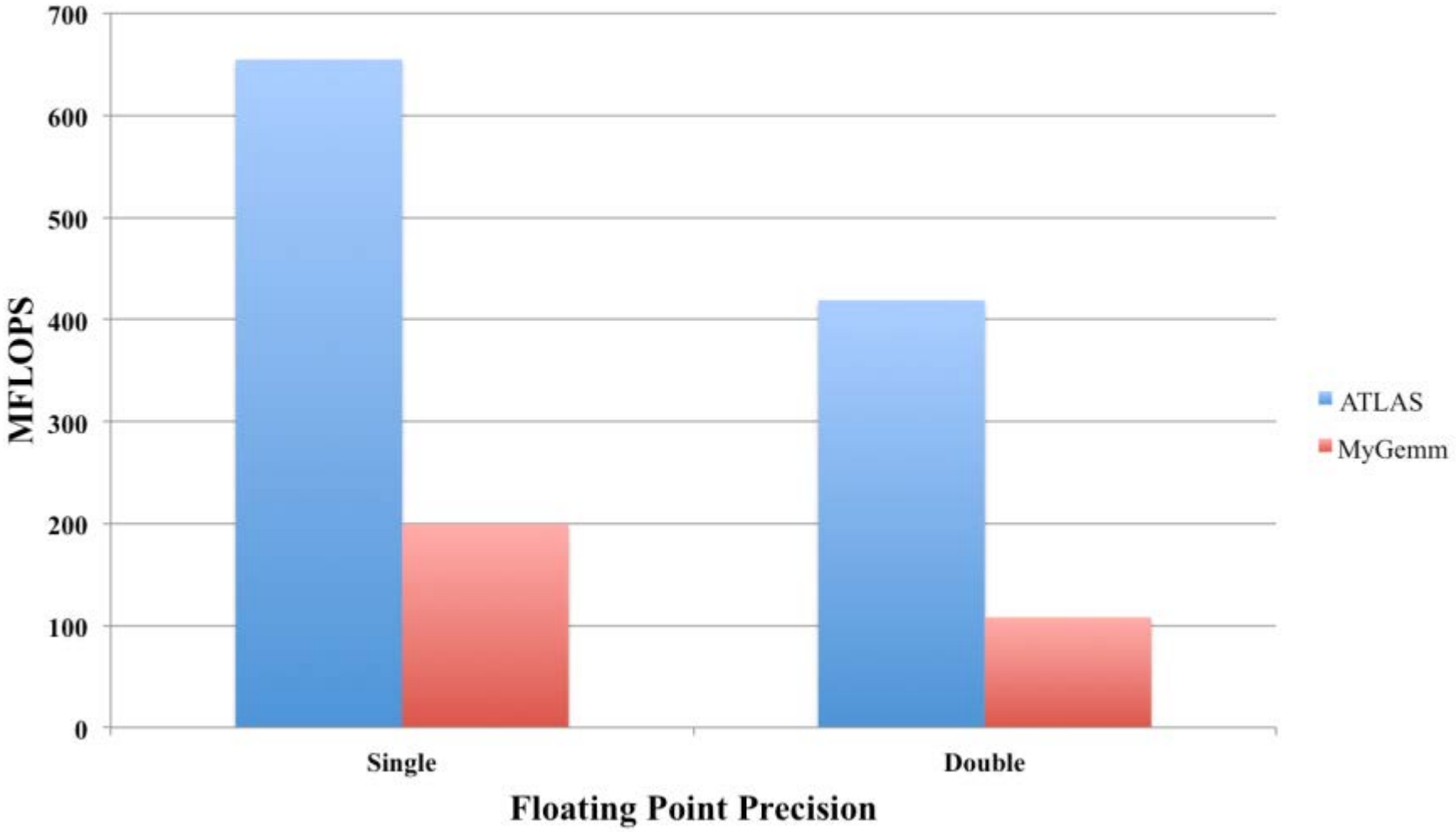
Clock Speed (MHz)

Single Precision, Single Node W/MyGemm & MPI



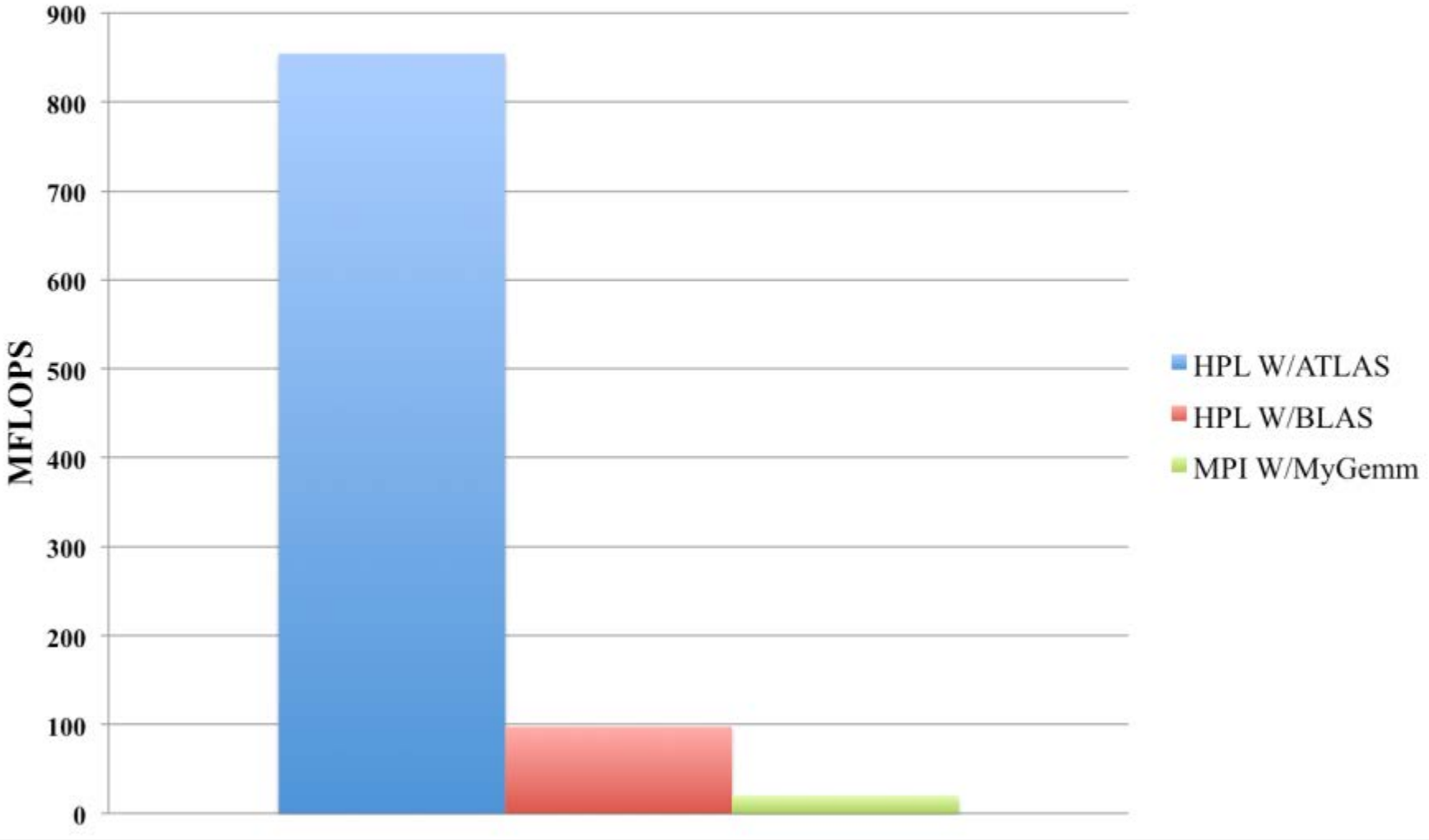
Performance Comparison

Single Node, W/ ATLAS & MyGemm



Performance Comparison

Raspberry Pi Cluster, Double Precision, HPL W/ ATLAS & BLAS



Conclusion

- Performance is dependent on key factors
  - Matrix Size
  - Block Size
  - RAM size
  - CPU speed
- Top 500 – 1993
  - T#292
    - Tied with General Motors

Conclusion – Pi Cluster vs. Yellowstone

	Pi Cluster ARM11, 32 bit, Double Precision	Yellowstone Sandy Bridge Xeon, 64 bit, Double Precision
Power	15.6 Watts	1.4 MWatts
Performance	836 MFLOPS	1.2 PFLOPS
Price	$250.00	$22,500,000.00
MFLOPS/W	54.8	875.3
MFLOPS/$	3.4	57.2

Thank You

- Dr. Richard Loft
- Raghu Raj Prasanna Kumar
- Amogh Simha
- Stephanie Barr

Questions?