Performance Study of Matrix Operations on Homogeneous and Heterogeneous Reconfigurable Computing Systems

Nahid Alam & Dr. Melissa Smith
Clemson University

Motivation & Goals

- Motivation:
  - Matrix operations prevalent in many scientific applications
    - Image and signal processing
    - Linear Solvers and LU Decomposition
  - Efficiency needed to support faster computations
- Goals:
  - Use FPGAs to exploit the parallelism of Matrix operations for higher throughput
  - Study the performance in single and multi-FPGA environment
  - Long term goals: Portability
    - Across multiple homogeneous platforms
    - Within a heterogeneous platform
  - Validate heterogeneous performance model

Platform & Tools

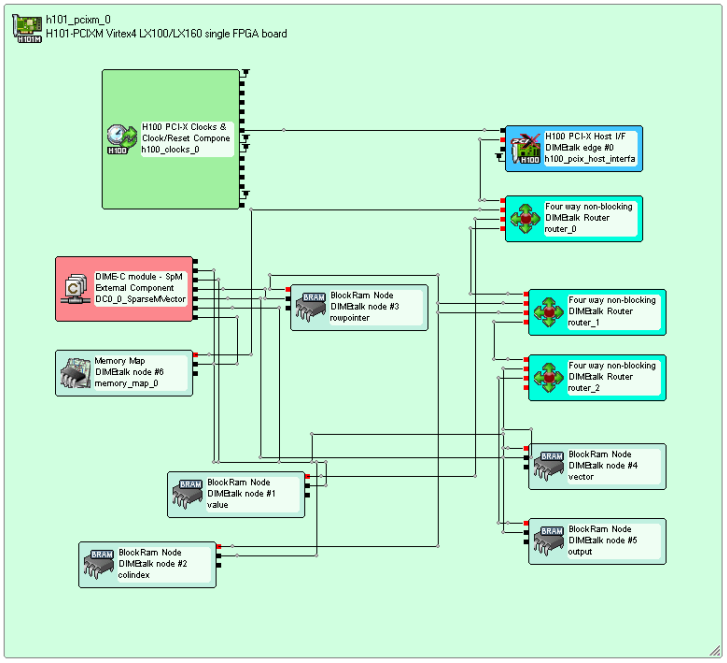
- Intel Xeon CPU 5130@ 2 GHz
  + 2 Nallatech H101-PCIXM FPGA accelerator cards
  - One V4LX100
  - 4 Bank DDR2 SSRAM (4 MBytes each)
  - 1 Bank DDR2 SDRAM (512 Mbytes)
  - 4 Channel Serial Communications
  - Supports floating-point single and double precision, IEEE754 norm
- DIMETalk System Design tool (DIME-C)
  - Advantages over HDL
    - Some auto parallelization
    - Makes programming less HDL intensive
  - Disadvantages
    - Less control than HDL
    - Lacks support for multidimensional array

Algorithms of Study

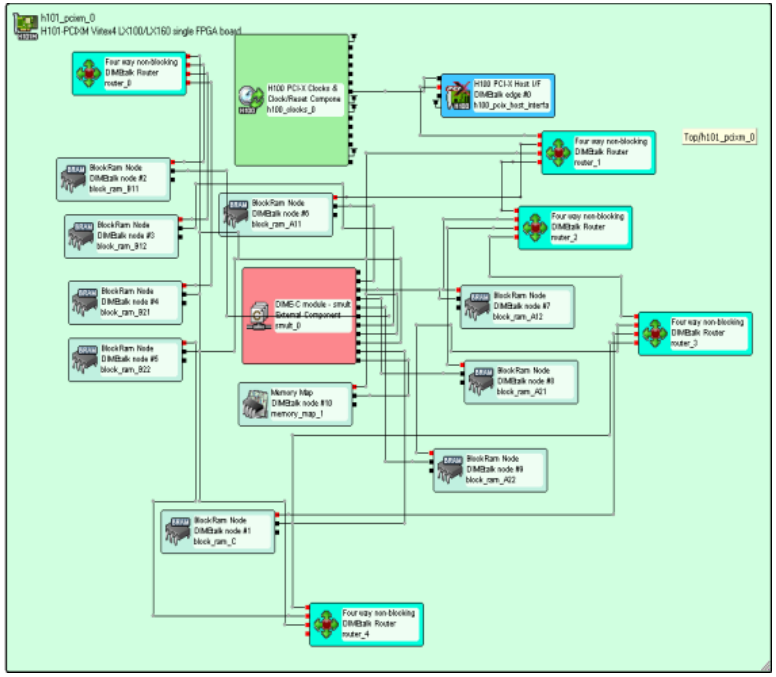
- Sparse Matrix Vector multiplication (SpMV)
  - Compress Row Storage (CRS) for Sparse Matrix
  - Sparse Matrix formulation outside the FPGA
  - Runtime: worst case O (m^2); m=actual # values in Sparse Matrix
- Conventional (dense) Matrix multiplication (CMM)
  - Runtime: worst case O(n^3)
- Block Matrix multiplication (BMM): Strassen approach
  - Concept is similar but exact approach was not followed due to the incapability of using recursion in DIME-C

Designing the system - Snapshot of 2 designs:

- Pink box - DIME-C module for computation inside the FPGA
- Block RAMs hold different formats of CRS data structure or different blocks in case of BMM; results also kept here
- Data routed to different components through the routers for computation

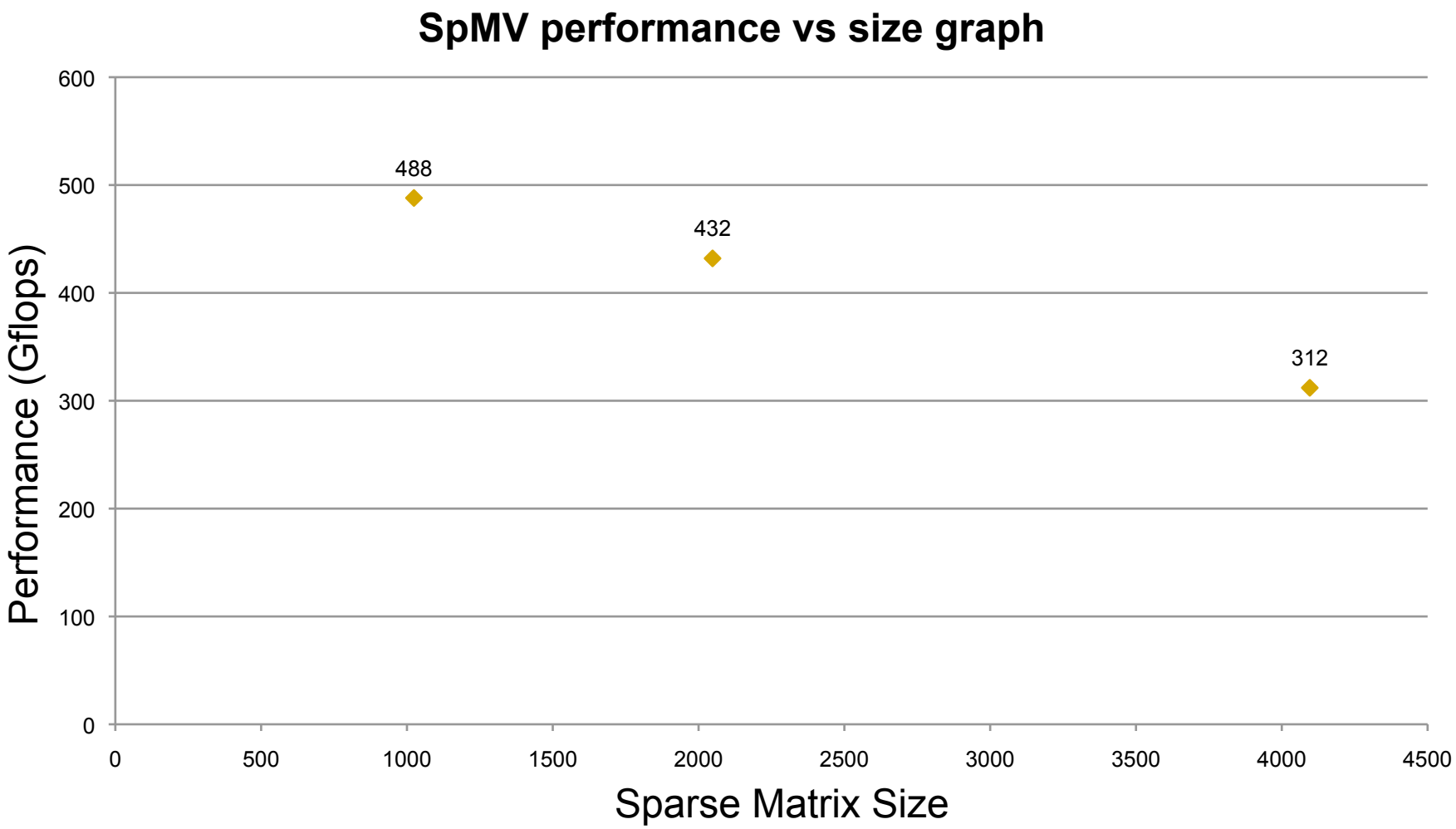


SpMV Design

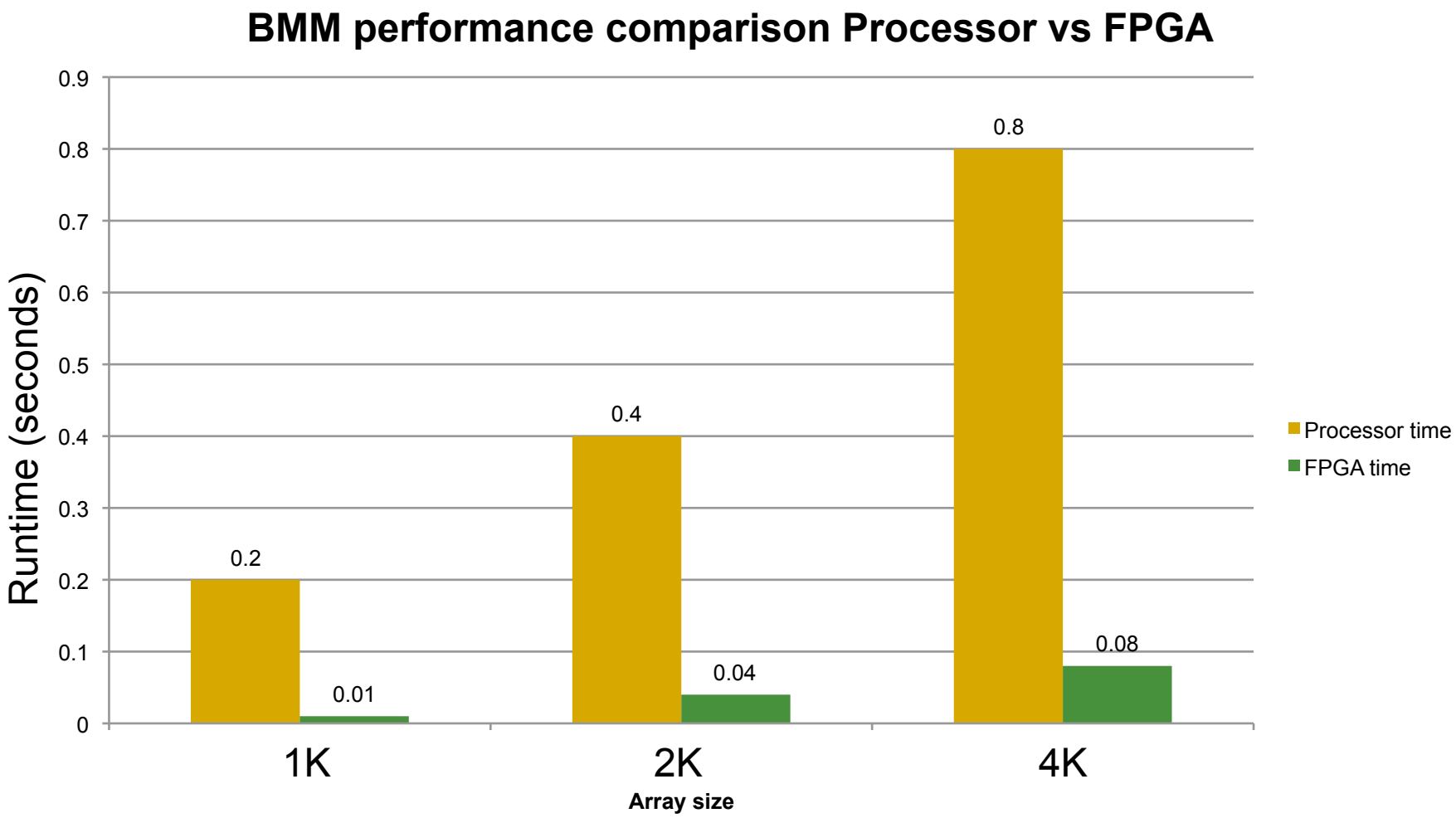


BMM Design

Results: Performance (SpMV)



Results: Performance (BMM)



Results: Single Node Resource Utilization

Algorithms	RAMB16s	Slices	DSPs
SpMV	18%	14%	4%
CMM	15%	10%	0%
BMM	47%	20%	0%

Bottleneck

- Target was to study Matrix operation on single and multi-node environment
  - Problem size may be too large to fit in a single node
  - Multi-node studies not complete due to some system issues
- DIME-C multi-dimensional array support
  - Must implement Matrices in a single dimension
  - Complicates implementations with larger sizes

Summary

- Future goals
  - Work on the current bottlenecks to study the performance over multi-node environment
  - Algorithmic improvement to exploit higher degree of parallelism
- Conclusion
  - This study is focused on performance and portability for a heterogeneous multi-node environment
  - Current state of study completed on single node
  - Multi-node analysis in progress