Parallel Processing Project: Max Bandwidth

Aman Hogan-Bailey

University of Texas at Arlington

CSE-5351: Parallel Processing

Contents

Theoretical Maximum Bandwidth (Part 1)

The `lscpu` command indicates that the UTA cluster given has an Intel® Xeon® Processor E5-2680 v4. According to Intel's ARK website, the theoretical maximum bandwidth for this processor is 76.8 GB/s (or 76,800 MB/s). This value, however, warrants verification based on the system's configuration.

The `dmidecode` command indicates the memory is DDR4-2400, with four memory channels and an 8-byte bus width. Given this configuration, the theoretical bandwidth for a single processor is as follows:

$$Bandwidth_P = TransferRate * BusWidth * MemChannel$$

$$Bandwidth_P = 2400\frac{MT}{s} * 8 * 4 = 76{,}800\frac{MB}{s}$$

$$Bandwidth_P = 76{,}800\frac{MB}{s} = 76.8\frac{GB}{s}$$

This value confirms the Intel ARK website's reported bandwidth. Furthermore, the system has two processors, each independently accessing four memory channels. Assuming optimal utilization of all channels, the total theoretical memory bandwidth for the entire system is:

$$Bandwidth_T = Bandwidth_P * NumOfProcessors$$

$$Bandwidth_T = 76{,}800\frac{MB}{s} * 2 = 153600\frac{MB}{s}$$

This figure represents the maximum data transfer rate the system could theoretically achieve under ideal conditions.

Max Bandwidth Testing

My experiments used a size of 1,000,000,000 or (1000 x 1000 x 1000) (1 billion) for the vector size. All results were compiled using level three GCC optimization. These results were relatively consistent among trials and were run through a visualization script and a bash script. Below are the results for parts 2-4.

Max Bandwidth Testing Boilerplate Code (Part 2)

The first experiment used the boilerplate code in class. The write bandwidth was 1,973 MB/s (1.3% max bandwidth), and the read bandwidth was 5,668 MB/s (3.7% max bandwidth). The results are in the first cell of *Table 1* and *Table 2*.

Write Bandwidth Experiments (Part 3)

The experiments indicate that using *Non-Temporal Writes + Setting Memory to Zero Before Timing* yields the highest write bandwidth, achieving a peak bandwidth of 117,639 MB/s (76% max bandwidth) at 16 threads. *No Optimization's* highest write bandwidth was 18,284 MB/s at max (56) threads. *Setting Memory to Zero Before Timing* had a maximum bandwidth of 57,873 MB/S at 16 threads. The maximum bandwidth for writes occurred between 16 and max (56) threads for all optimization types over several experiments (Table 1).

Across all optimization types, the write bandwidth more than quadrupled going from one thread to 16 threads. Similarly, for all optimization types, the bandwidth leveled off after 16 threads. The data gathered shows evidence to suggest that *Non-Temporal Writes + Setting Memory to Zero Before Timing yields* the highest bandwidth utilization at 76% and that the highest bandwidth occurs between 16 to max (56) threads (Figure 1).

Read Bandwidth Experiments (Part 4)

The experiments indicate that using an *Unroll Loop Size* of 1 and a thread size of 1 always yields the worst performance at around 5,668 MB/s. Doubling the *Unroll Loop Size* increases the bandwidth at a far less rate than doubling the *Number of Threads*. The highest bandwidths are achieved at an *Unroll Loop Size* greater than one and a *Number of Threads* of 16 and over. The highest bandwidth was with a *Number of Threads* of max (56) and an *Unroll Loop Size* of 16 at 134,474 MB/s (87% max bandwidth) (Table 2).

The data suggests that the thread count affects bandwidth more than unroll loop size. The data also suggests that *Number of Threads* of 16 and higher will achieve the highest read bandwidth (Figure 2).

Conclusions

In conclusion, this project assessed various optimizations to determine their effectiveness in achieving max memory bandwidth for both read and write operations on the UTA cluster. The theoretical maximum memory bandwidth is 153,600 MB/s. And experiments revealed that it is very possible to get close to these numbers with the maximum read bandwidth utilization at 87% with 134,474 MB/s using 56 threads and an Unroll and an *Unroll Loop Size* of 16. Similarly, the maximum write bandwidth was at 76% with 117,639 MB/s using *Non-Temporal Writes + Setting Memory to Zero Before Timing* and 16 threads.

For write operations, the highest bandwidth *used Non-Temporal Writes + Setting Memory to Zero Before Timing*, reaching 117,639 MB/s or at 76% of the theoretical maximum, with 16 threads. This optimization consistently outperformed others between 8 to 56 threads. Additionally, across all optimization types, the maximum write bandwidth was observed to generally be achieved between 16 and max (56) threads.

In read operations, the best performance was with 56 threads and an unroll loop size of 16, achieving a peak bandwidth of 134,474 MB/s, approximately 87% of the theoretical maximum. The experiments suggest that thread count has a more substantial impact on both read and write bandwidths than other factors such as unroll loop size. Specifically, the thread count was critical in optimizing bandwidth, with higher thread counts generally yielding better results for read operations.

When testing using larger sizes for the vector, the bandwidth for the reads and writes stayed stead between 70% and 80%.

Overall, the tested optimizations, particularly *Non-Temporal Writes + Setting Memory to Zero Before Timing*, significantly improved memory bandwidth from 1% to 76% utilization and yielded the best write bandwidth. And the thread counts also significantly increased the read bandwidth, going from 3.7% to 87%. Further refinement of these and other optimization strategies will be essential in closing the gap between theoretical and practical performance in high-performance computing environments.

References

None

Tables

Table 1

Write Bandwidth in MB/s

OPTIMIZATION TYPE	THREADS					
-	1	2	4	8	16	Max (56)
NO OPTIMIZATION	1973.435715	3770.683061	6946.821734	12593.10373	17088.52562	18284.50349
SET MEM TO ZERO	8485.809857	6085.867691	11436.62835	38367.61879	57873.26113	56287.83527
NON-TEMPORAL WRITES + SET MEM TO SET MEM TO ZERO	18314.72325	18273.78186	59054.07563	36031.0451	117639.4968	109589.7694

Table 2

Read Bandwidth in MB/s

UNROLL LOOP SIZE	THREADS					
-	1	2	4	8	16	Max (56)
1	5668.872	7705.241	19603.21	21153.94	72788.46	131654.9
2	7444.43	8811.526	24334.05	40502.61	88584.44	130854.9
4	7656.294	14301.64	23537.32	27733.94	57807.2	133795.7
8	7768.76	16093.93	26722.16	36270.66	60148.93	133459.1
16	7821.25	14376.95	27121.04	51648.46	45155.52	134474.7

Figures

Figure 1.

Figure shows line plot of Bandwidth and Number of threads across various optimizations.

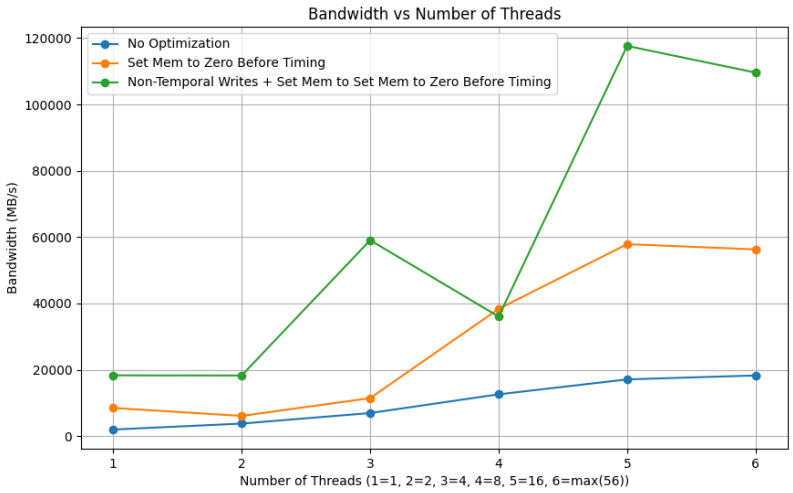

Figure 2.

Figure shows heatmap between Unroll loop size and number of threads.