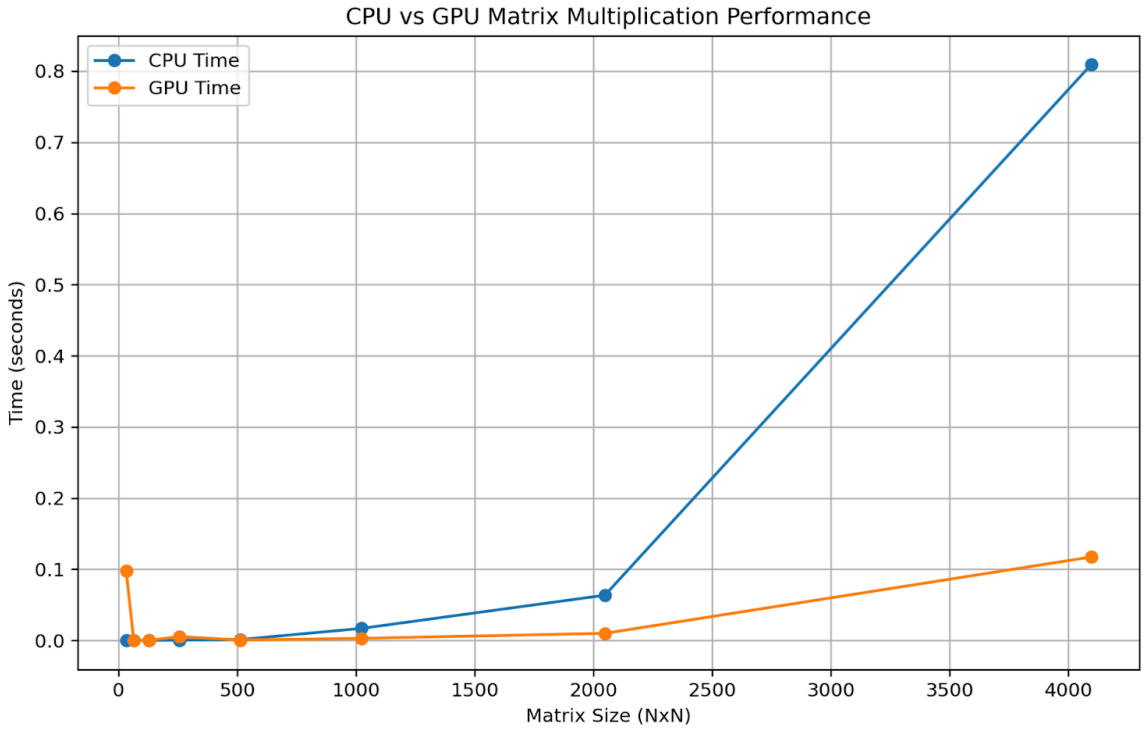
Raymond Zhu

Output of matrix multiplication for different matrix sizes:

```
Running experiment for matrix size: 32x32
CPU Time: 0.00003210 seconds
GPU Time: 0.09778350 seconds
Running experiment for matrix size: 64x64
CPU Time: 0.00002640 seconds
GPU Time: 0.00014430 seconds
Running experiment for matrix size: 128x128
CPU Time: 0.00019100 seconds
GPU Time: 0.00014820 seconds
Running experiment for matrix size: 256x256
CPU Time: 0.00038370 seconds
GPU Time: 0.00560030 seconds
Running experiment for matrix size: 512x512
CPU Time: 0.00132150 seconds
GPU Time: 0.00061000 seconds
Running experiment for matrix size: 1024x1024
CPU Time: 0.01702940 seconds
GPU Time: 0.00301420 seconds
Running experiment for matrix size: 2048x2048
CPU Time: 0.06360790 seconds
GPU Time: 0.01004580 seconds
Running experiment for matrix size: 4096x4096
CPU Time: 0.80914700 seconds
GPU Time: 0.11729100 seconds
```

CPU vs GPU Matrix Multiplication Performance Graph:

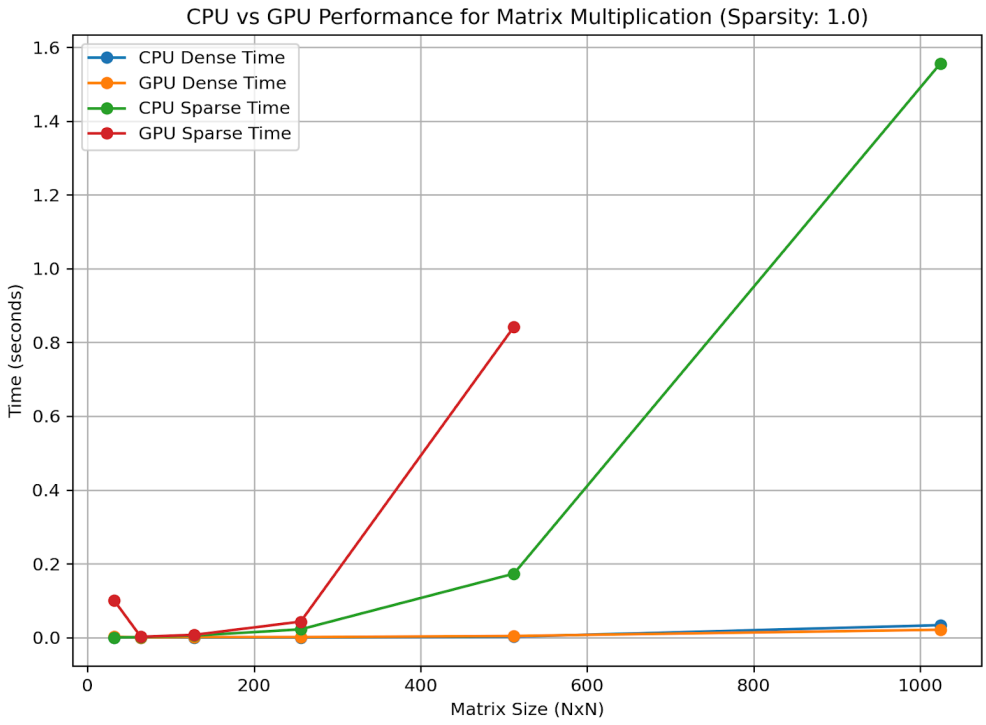


Output of Matrix Multiplication for Sparsity level 1.0:

```
Running experiments for sparsity level: 1.0

Running experiment for matrix size: 32x32, sparsity: 1.0
Results for matrix size 32x32, sparsity 1.0:
  CPU Dense Time: 0.00000000 seconds
  GPU Dense Time: 0.00200367 seconds
  CPU Sparse Time: 0.00000000 seconds
  GPU Sparse Time: 0.10035872 seconds
Running experiment for matrix size: 64x64, sparsity: 1.0
Results for matrix size 64x64, sparsity 1.0:
  CPU Dense Time: 0.00000000 seconds
  GPU Dense Time: 0.00000000 seconds
  CPU Sparse Time: 0.00100231 seconds
  GPU Sparse Time: 0.00198984 seconds
Running experiment for matrix size: 128x128, sparsity: 1.0
Results for matrix size 128x128, sparsity 1.0:
  CPU Dense Time: 0.00000000 seconds
  GPU Dense Time: 0.00100160 seconds
  CPU Sparse Time: 0.00500870 seconds
  GPU Sparse Time: 0.00699735 seconds
Running experiment for matrix size: 256x256, sparsity: 1.0
Results for matrix size 256x256, sparsity 1.0:
  CPU Dense Time: 0.00000000 seconds
  GPU Dense Time: 0.00102162 seconds
  CPU Sparse Time: 0.02199554 seconds
  GPU Sparse Time: 0.04296207 seconds
Running experiment for matrix size: 512x512, sparsity: 1.0
Results for matrix size 512x512, sparsity 1.0:
  CPU Dense Time: 0.00200200 seconds
  GPU Dense Time: 0.00399947 seconds
  CPU Sparse Time: 0.17300320 seconds
  GPU Sparse Time: 0.84220719 seconds
Running experiment for matrix size: 1024x1024, sparsity: 1.0
Results for matrix size 1024x1024, sparsity 1.0:
  CPU Dense Time: 0.03344941 seconds
  GPU Dense Time: 0.02096534 seconds
  CPU Sparse Time: 1.55603576 seconds
  GPU Sparse Time: Out of Memory
```

CPU vs GPU Performance for Matrix Multiplication (Sparsity 1.0):

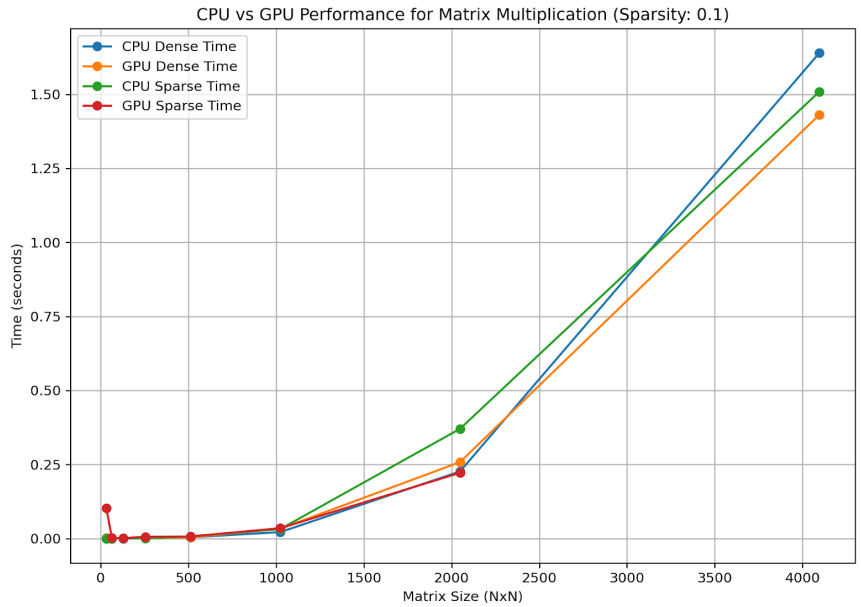


Output of Matrix Multiplication for Sparsity level 0.1:

```
Running experiments for sparsity level: 0.1

Running experiment for matrix size: 32x32, sparsity: 0.1
Results for matrix size 32x32, sparsity 0.1:
  CPU Dense Time: 0.00000000 seconds
  GPU Dense Time: 0.00200319 seconds
  CPU Sparse Time: 0.00000000 seconds
  GPU Sparse Time: 0.10310626 seconds
Running experiment for matrix size: 64x64, sparsity: 0.1
Results for matrix size 64x64, sparsity 0.1:
  CPU Dense Time: 0.00000000 seconds
  GPU Dense Time: 0.00000000 seconds
  CPU Sparse Time: 0.00000000 seconds
  GPU Sparse Time: 0.00201178 seconds
Running experiment for matrix size: 128x128, sparsity: 0.1
Results for matrix size 128x128, sparsity 0.1:
  CPU Dense Time: 0.00000000 seconds
  GPU Dense Time: 0.00099397 seconds
  CPU Sparse Time: 0.00099993 seconds
  GPU Sparse Time: 0.00100088 seconds
Running experiment for matrix size: 256x256, sparsity: 0.1
Results for matrix size 256x256, sparsity 0.1:
  CPU Dense Time: 0.00099897 seconds
  GPU Dense Time: 0.00100112 seconds
  CPU Sparse Time: 0.00099802 seconds
  GPU Sparse Time: 0.00599909 seconds
Running experiment for matrix size: 512x512, sparsity: 0.1
Results for matrix size 512x512, sparsity 0.1:
  CPU Dense Time: 0.00400186 seconds
  GPU Dense Time: 0.00400305 seconds
  CPU Sparse Time: 0.00699568 seconds
  GPU Sparse Time: 0.00700045 seconds
Running experiment for matrix size: 1024x1024, sparsity: 0.1
Results for matrix size 1024x1024, sparsity 0.1:
  CPU Dense Time: 0.02182221 seconds
  GPU Dense Time: 0.03300095 seconds
  CPU Sparse Time: 0.03200126 seconds
  GPU Sparse Time: 0.03499985 seconds
Running experiment for matrix size: 2048x2048, sparsity: 0.1
Results for matrix size 2048x2048, sparsity 0.1:
  CPU Dense Time: 0.22660279 seconds
  GPU Dense Time: 0.25815582 seconds
  CPU Sparse Time: 0.37053490 seconds
  GPU Sparse Time: 0.22199297 seconds
Running experiment for matrix size: 4096x4096, sparsity: 0.1
Results for matrix size 4096x4096, sparsity 0.1:
  CPU Dense Time: 1.63986135 seconds
  GPU Dense Time: 1.43103886 seconds
  CPU Sparse Time: 1.50903416 seconds
  GPU Sparse Time: Out of Memory
```

CPU vs GPU Performance for Matrix Multiplication (Sparsity 0.1):

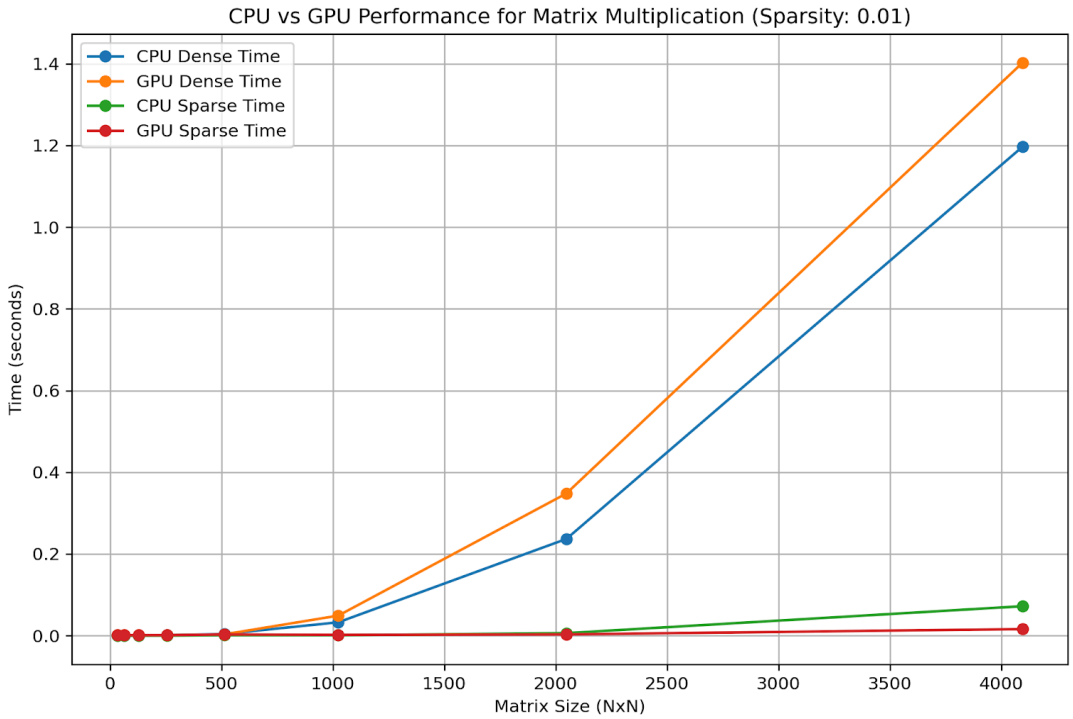


Output of Matrix Multiplication for Sparsity 0.01:

```
Running experiments for sparsity level: 0.01

Running experiment for matrix size: 32x32, sparsity: 0.01
Results for matrix size 32x32, sparsity 0.01:
  CPU Dense Time: 0.00000000 seconds
  GPU Dense Time: 0.00099874 seconds
  CPU Sparse Time: 0.00000000 seconds
  GPU Sparse Time: 0.00099993 seconds
Running experiment for matrix size: 64x64, sparsity: 0.01
Results for matrix size 64x64, sparsity 0.01:
  CPU Dense Time: 0.00000000 seconds
  GPU Dense Time: 0.00099730 seconds
  CPU Sparse Time: 0.00000000 seconds
  GPU Sparse Time: 0.00099945 seconds
Running experiment for matrix size: 128x128, sparsity: 0.01
Results for matrix size 128x128, sparsity 0.01:
  CPU Dense Time: 0.00000000 seconds
  GPU Dense Time: 0.00099850 seconds
  CPU Sparse Time: 0.00000000 seconds
  GPU Sparse Time: 0.00099921 seconds
Running experiment for matrix size: 256x256, sparsity: 0.01
Results for matrix size 256x256, sparsity 0.01:
  CPU Dense Time: 0.00101542 seconds
  GPU Dense Time: 0.00100756 seconds
  CPU Sparse Time: 0.00000000 seconds
  GPU Sparse Time: 0.00099897 seconds
Running experiment for matrix size: 512x512, sparsity: 0.01
Results for matrix size 512x512, sparsity 0.01:
  CPU Dense Time: 0.00400758 seconds
  GPU Dense Time: 0.00299478 seconds
  CPU Sparse Time: 0.00100780 seconds
  GPU Sparse Time: 0.00298047 seconds
Running experiment for matrix size: 1024x1024, sparsity: 0.01
Results for matrix size 1024x1024, sparsity 0.01:
  CPU Dense Time: 0.03255486 seconds
  GPU Dense Time: 0.04894948 seconds
  CPU Sparse Time: 0.00102448 seconds
  GPU Sparse Time: 0.00196481 seconds
Running experiment for matrix size: 2048x2048, sparsity: 0.01
Results for matrix size 2048x2048, sparsity 0.01:
  CPU Dense Time: 0.23669529 seconds
  GPU Dense Time: 0.34803724 seconds
  CPU Sparse Time: 0.00603962 seconds
  GPU Sparse Time: 0.00296760 seconds
Running experiment for matrix size: 4096x4096, sparsity: 0.01
Results for matrix size 4096x4096, sparsity 0.01:
  CPU Dense Time: 1.19720101 seconds
  GPU Dense Time: 1.40296888 seconds
  CPU Sparse Time: 0.07199860 seconds
  GPU Sparse Time: 0.01600194 seconds
```

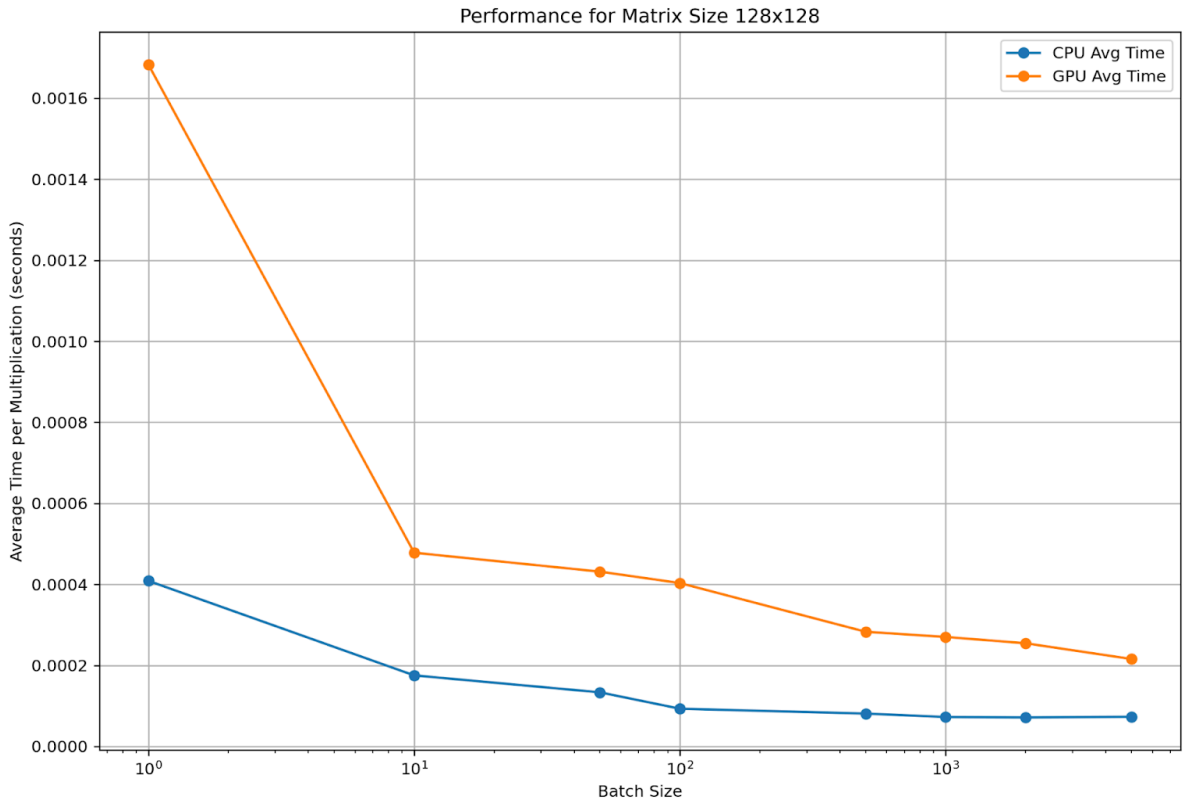
CPU vs GPU Performance for Matrix Multiplication (Sparsity 0.01):



Output of Matrix Multiplications in Batches (Matrix Size 128x128):

```
Running experiment for matrix size 128x128...
  Running batch of 1 multiplications...
    CPU Avg Time: 0.00040860 seconds, GPU Avg Time: 0.00168300 seconds
  Running batch of 10 multiplications...
    CPU Avg Time: 0.00017524 seconds, GPU Avg Time: 0.00047805 seconds
  Running batch of 50 multiplications...
    CPU Avg Time: 0.00013341 seconds, GPU Avg Time: 0.00043135 seconds
  Running batch of 100 multiplications...
    CPU Avg Time: 0.00009300 seconds, GPU Avg Time: 0.00040317 seconds
  Running batch of 500 multiplications...
    CPU Avg Time: 0.00008098 seconds, GPU Avg Time: 0.00028291 seconds
  Running batch of 1000 multiplications...
    CPU Avg Time: 0.00007248 seconds, GPU Avg Time: 0.00027014 seconds
  Running batch of 2000 multiplications...
    CPU Avg Time: 0.00007147 seconds, GPU Avg Time: 0.00025452 seconds
  Running batch of 5000 multiplications...
    CPU Avg Time: 0.00007305 seconds, GPU Avg Time: 0.00021566 seconds
```

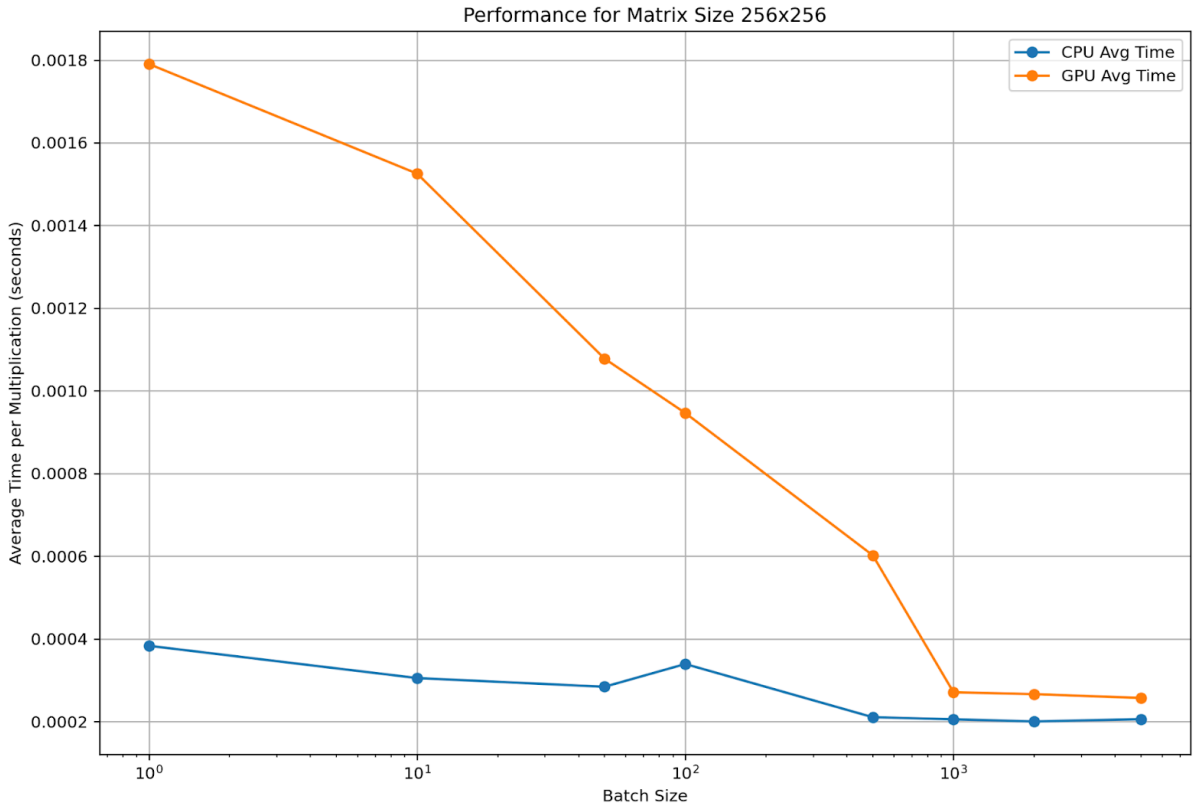
CPU vs GPU Average Time per Multiplication per Batch Size (Matrix Size 128x128):



Output of Matrix Multiplications in Batches (Matrix Size 256x256):

```
Running experiment for matrix size 256x256...
  Running batch of 1 multiplications...
    CPU Avg Time: 0.00038310 seconds, GPU Avg Time: 0.00179060 seconds
  Running batch of 10 multiplications...
    CPU Avg Time: 0.00030504 seconds, GPU Avg Time: 0.00152546 seconds
  Running batch of 50 multiplications...
    CPU Avg Time: 0.00028400 seconds, GPU Avg Time: 0.00107777 seconds
  Running batch of 100 multiplications...
    CPU Avg Time: 0.00033906 seconds, GPU Avg Time: 0.00094620 seconds
  Running batch of 500 multiplications...
    CPU Avg Time: 0.00021036 seconds, GPU Avg Time: 0.00060205 seconds
  Running batch of 1000 multiplications...
    CPU Avg Time: 0.00020527 seconds, GPU Avg Time: 0.00027065 seconds
  Running batch of 2000 multiplications...
    CPU Avg Time: 0.00020004 seconds, GPU Avg Time: 0.00026618 seconds
  Running batch of 5000 multiplications...
    CPU Avg Time: 0.00020560 seconds, GPU Avg Time: 0.00025691 seconds
```

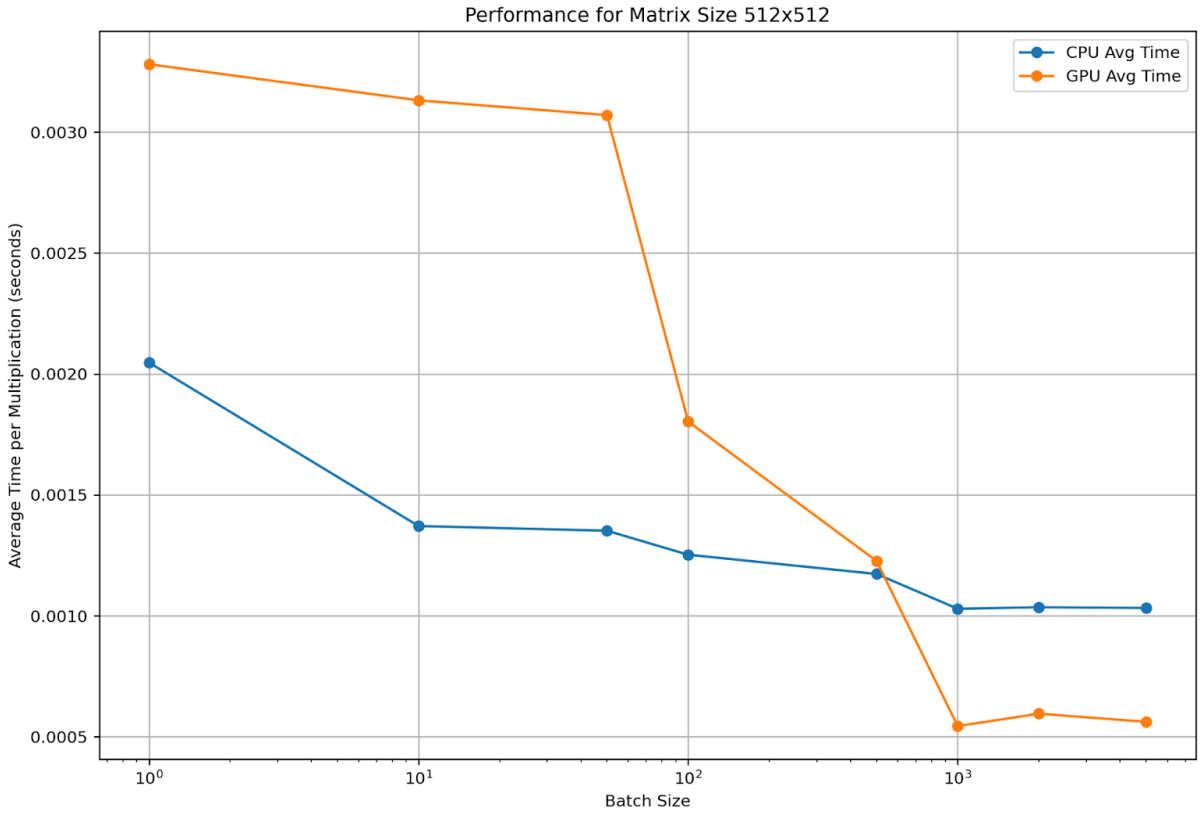
CPU vs GPU Average Time per Multiplication per Batch Size (Matrix Size 256x256):



Output of Matrix Multiplications in Batches (Matrix Size 512x512):

```
Running experiment for matrix size 512x512...
  Running batch of 1 multiplications...
    CPU Avg Time: 0.00204680 seconds, GPU Avg Time: 0.00328040 seconds
  Running batch of 10 multiplications...
    CPU Avg Time: 0.00137094 seconds, GPU Avg Time: 0.00313117 seconds
  Running batch of 50 multiplications...
    CPU Avg Time: 0.00135161 seconds, GPU Avg Time: 0.00307002 seconds
  Running batch of 100 multiplications...
    CPU Avg Time: 0.00125242 seconds, GPU Avg Time: 0.00180309 seconds
  Running batch of 500 multiplications...
    CPU Avg Time: 0.00117283 seconds, GPU Avg Time: 0.00122643 seconds
  Running batch of 1000 multiplications...
    CPU Avg Time: 0.00102905 seconds, GPU Avg Time: 0.00054403 seconds
  Running batch of 2000 multiplications...
    CPU Avg Time: 0.00103528 seconds, GPU Avg Time: 0.00059508 seconds
  Running batch of 5000 multiplications...
    CPU Avg Time: 0.00103223 seconds, GPU Avg Time: 0.00056159 seconds
```

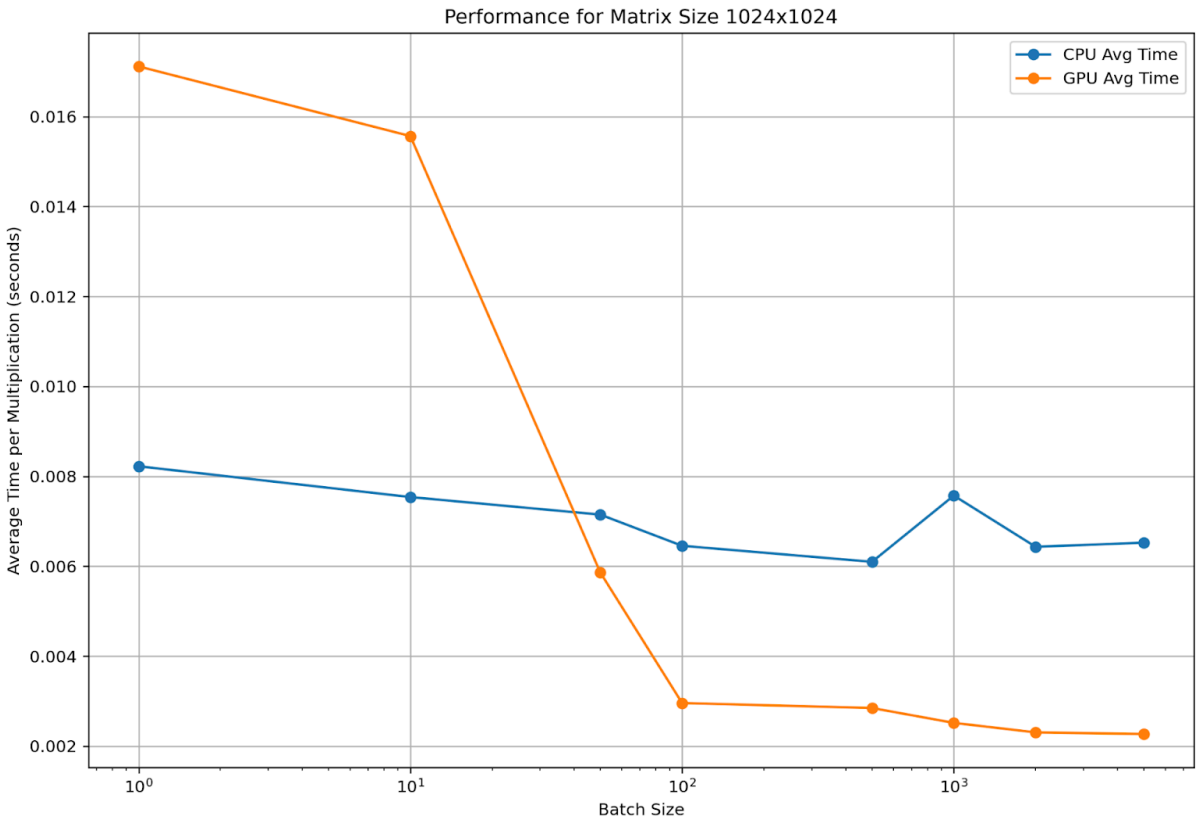
CPU vs GPU Average Time per Multiplication per Batch Size (Matrix Size 512x512):



Output of Matrix Multiplications in Batches (Matrix Size 1024x1024):

```
Running experiment for matrix size 1024x1024...
  Running batch of 1 multiplications...
    CPU Avg Time: 0.00822350 seconds, GPU Avg Time: 0.01711390 seconds
  Running batch of 10 multiplications...
    CPU Avg Time: 0.00753877 seconds, GPU Avg Time: 0.01556467 seconds
  Running batch of 50 multiplications...
    CPU Avg Time: 0.00714889 seconds, GPU Avg Time: 0.00586136 seconds
  Running batch of 100 multiplications...
    CPU Avg Time: 0.00645554 seconds, GPU Avg Time: 0.00295922 seconds
  Running batch of 500 multiplications...
    CPU Avg Time: 0.00609901 seconds, GPU Avg Time: 0.00284888 seconds
  Running batch of 1000 multiplications...
    CPU Avg Time: 0.00757160 seconds, GPU Avg Time: 0.00251810 seconds
  Running batch of 2000 multiplications...
    CPU Avg Time: 0.00643332 seconds, GPU Avg Time: 0.00230681 seconds
  Running batch of 5000 multiplications...
    CPU Avg Time: 0.00652431 seconds, GPU Avg Time: 0.00227072 seconds
```

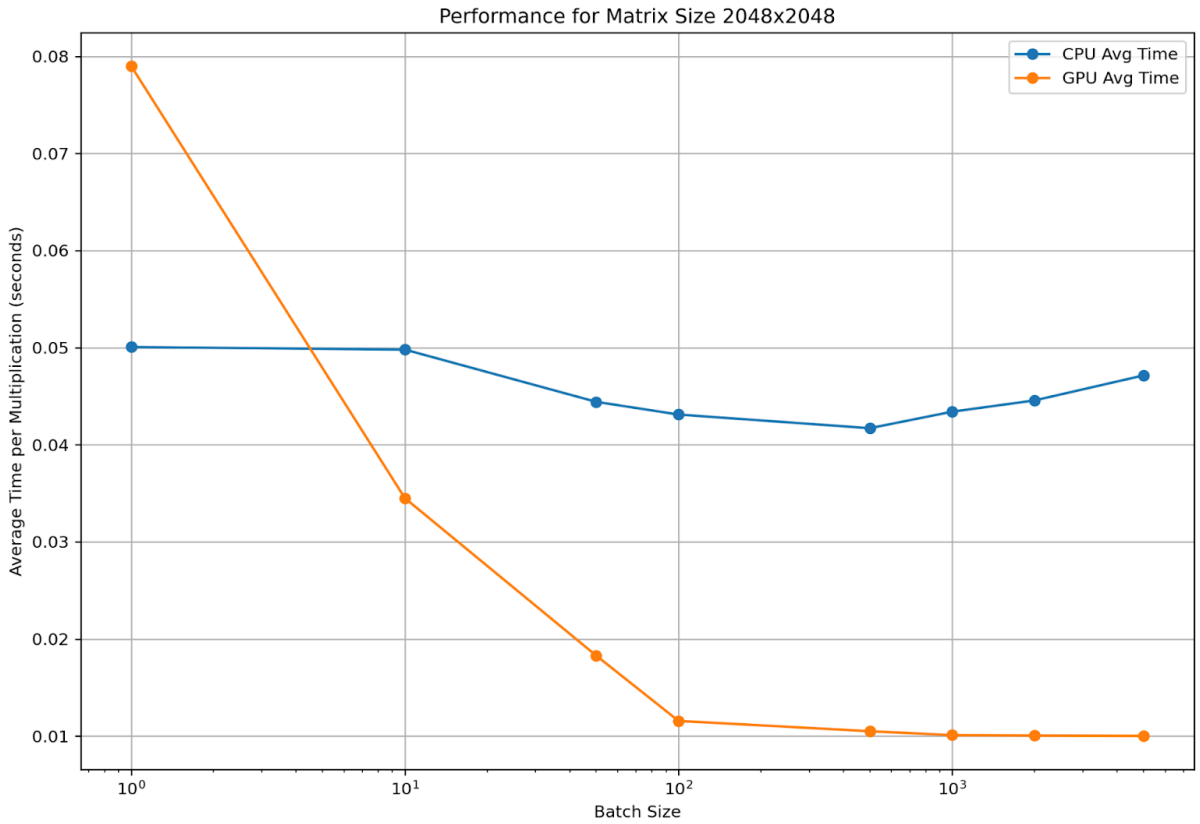
CPU vs GPU Average Time per Multiplication per Batch Size (Matrix Size 1024x1024):



Output of Matrix Multiplications in Batches (Matrix Size 2048x2048):

```
Running experiment for matrix size 2048x2048...
  Running batch of 1 multiplications...
    CPU Avg Time: 0.05005960 seconds, GPU Avg Time: 0.07898980 seconds
  Running batch of 10 multiplications...
    CPU Avg Time: 0.04979847 seconds, GPU Avg Time: 0.03450133 seconds
  Running batch of 50 multiplications...
    CPU Avg Time: 0.04441943 seconds, GPU Avg Time: 0.01828833 seconds
  Running batch of 100 multiplications...
    CPU Avg Time: 0.04310876 seconds, GPU Avg Time: 0.01156192 seconds
  Running batch of 500 multiplications...
    CPU Avg Time: 0.04170343 seconds, GPU Avg Time: 0.01049960 seconds
  Running batch of 1000 multiplications...
    CPU Avg Time: 0.04340517 seconds, GPU Avg Time: 0.01009902 seconds
  Running batch of 2000 multiplications...
    CPU Avg Time: 0.04456414 seconds, GPU Avg Time: 0.01005862 seconds
  Running batch of 5000 multiplications...
    CPU Avg Time: 0.04713722 seconds, GPU Avg Time: 0.01001515 seconds
```

CPU vs GPU Average Time per Multiplication per Batch Size (Matrix Size 2048x2048):



Output of Matrix Multiplication for Sparsity 1.0 for Matrix Size 2048x2048 Error Message:

```
Running experiment for matrix size: 2048x2048, sparsity: 1.0
Traceback (most recent call last):
  File "C:\Users\Ray Zhu\Code\Final_Project_test.py", line 172, in <module>
    main()
  File "C:\Users\Ray Zhu\Code\Final_Project_test.py", line 147, in main
    results.append((size, sparsity, *run_experiment(size, sparsity)))
                                     ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ray Zhu\Code\Final_Project_test.py", line 124, in run_experiment
    _, gpu_sparse_time = gpu_sparse_matrix_multiplication(A_sparse, B_sparse)
                         ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ray Zhu\Code\Final_Project_test.py", line 97, in gpu_sparse_matrix_multiplication
    C_sparse_gpu = A_gpu.dot(B_gpu)
                   ^^^^^^^^^^^^^^^^
  File "C:\Users\Ray Zhu\AppData\Local\Programs\Python\Python312\Lib\site-packages\cupyx\scipy\sparse\_base.py", line 341, in dot
    return self @ other
           ~~~~~^~~~~~~
  File "C:\Users\Ray Zhu\AppData\Local\Programs\Python\Python312\Lib\site-packages\cupyx\scipy\sparse\_base.py", line 130, in __matmul__
    return self.__mul__(other)
           ^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ray Zhu\AppData\Local\Programs\Python\Python312\Lib\site-packages\cupyx\scipy\sparse\_csr.py", line 159, in __mul__
    return cusparse.spgemm(self, other)
           ^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ray Zhu\AppData\Local\Programs\Python\Python312\Lib\site-packages\cupyx\cusparse.py", line 2057, in spgemm
    _cusparse.spGEMM_workEstimation(
  File "cupy_backends\cuda\libs\cusparse.pyx", line 5061, in cupy_backends.cuda.libs.cusparse.spGEMM_workEstimation
  File "cupy_backends\cuda\libs\cusparse.pyx", line 5072, in cupy_backends.cuda.libs.cusparse.spGEMM_workEstimation
  File "cupy_backends\cuda\libs\cusparse.pyx", line 1535, in cupy_backends.cuda.libs.cusparse.check_status
cupy_backends.cuda.libs.cusparse.CuSparseError: CUSPARSE_STATUS_INSUFFICIENT_RESOURCES: insufficient resources
```