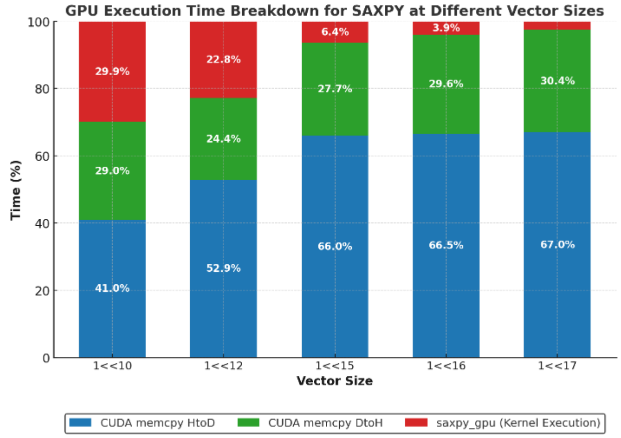
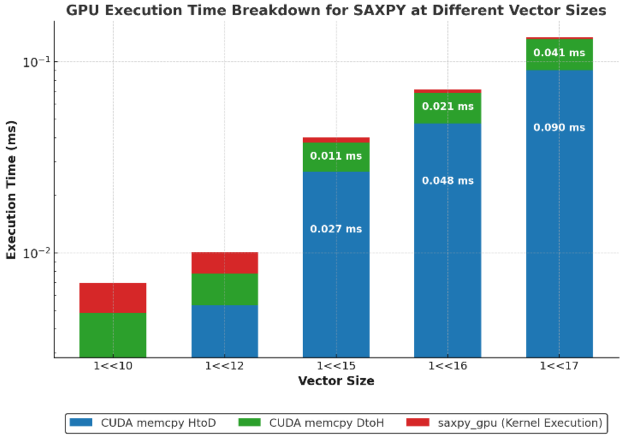
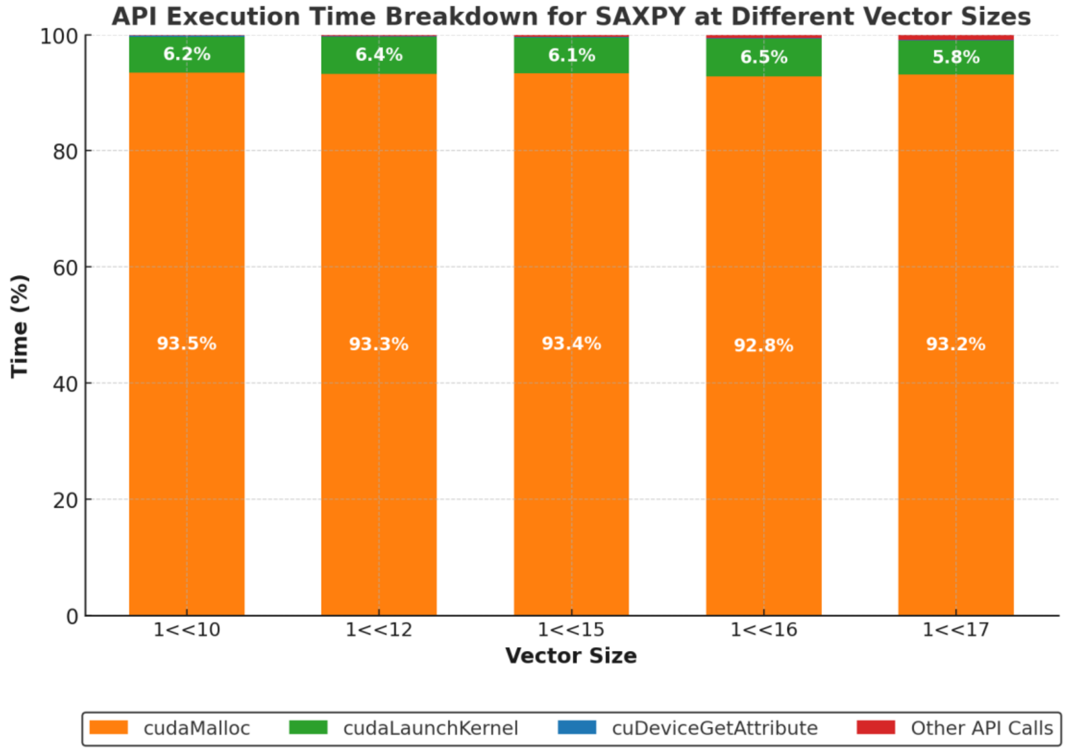
PART A: Single-precision A · X Plus Y (SAXPY)

Altering Vector Size (1<<10 to 1<<17)



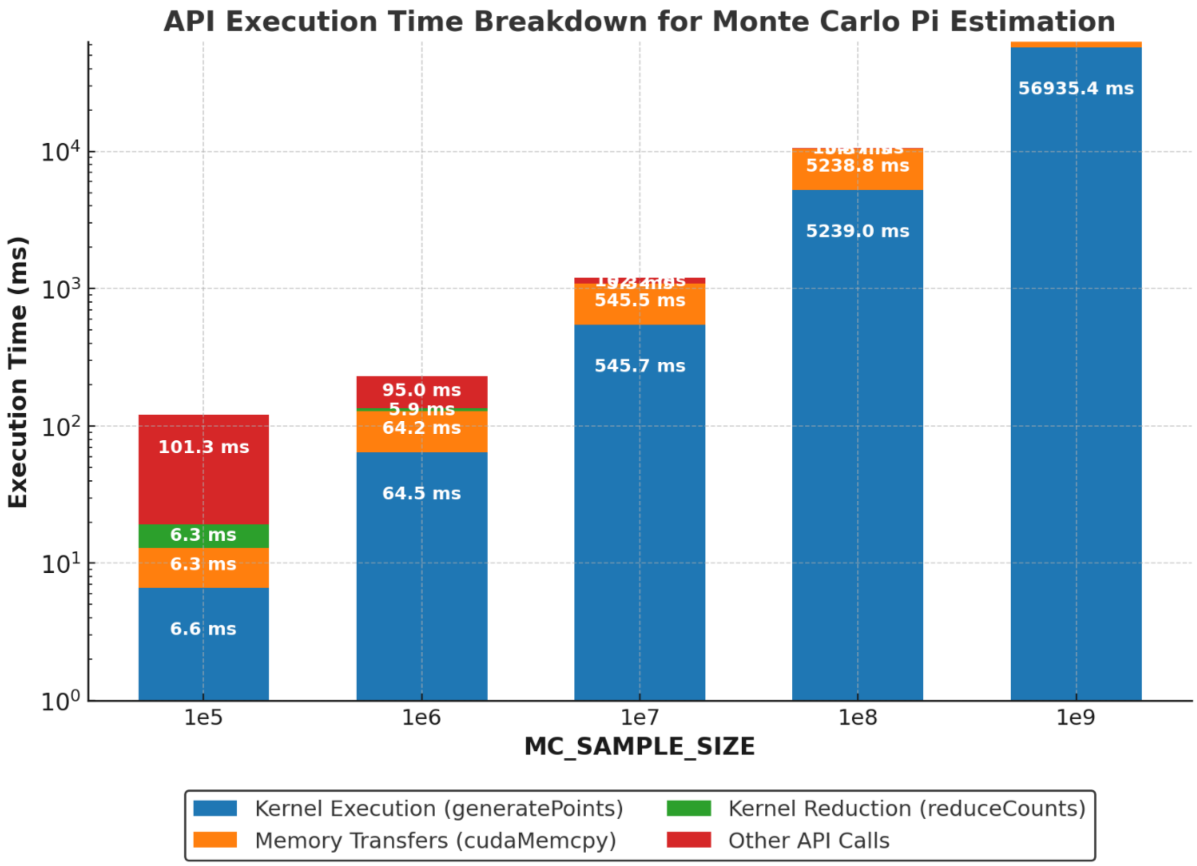


Based on the results above, we observe that increasing vector size leads to higher execution time, with a significant portion spent on memory copy operations. This indicates that performance is primarily limited by memory transfers rather than computation. Additionally, from the result below, cudaMalloc accounts for ~93% of API execution time, highlighting substantial memory allocation overhead. Both observations confirm that GPU execution for SAXPY is **memory-bound**.



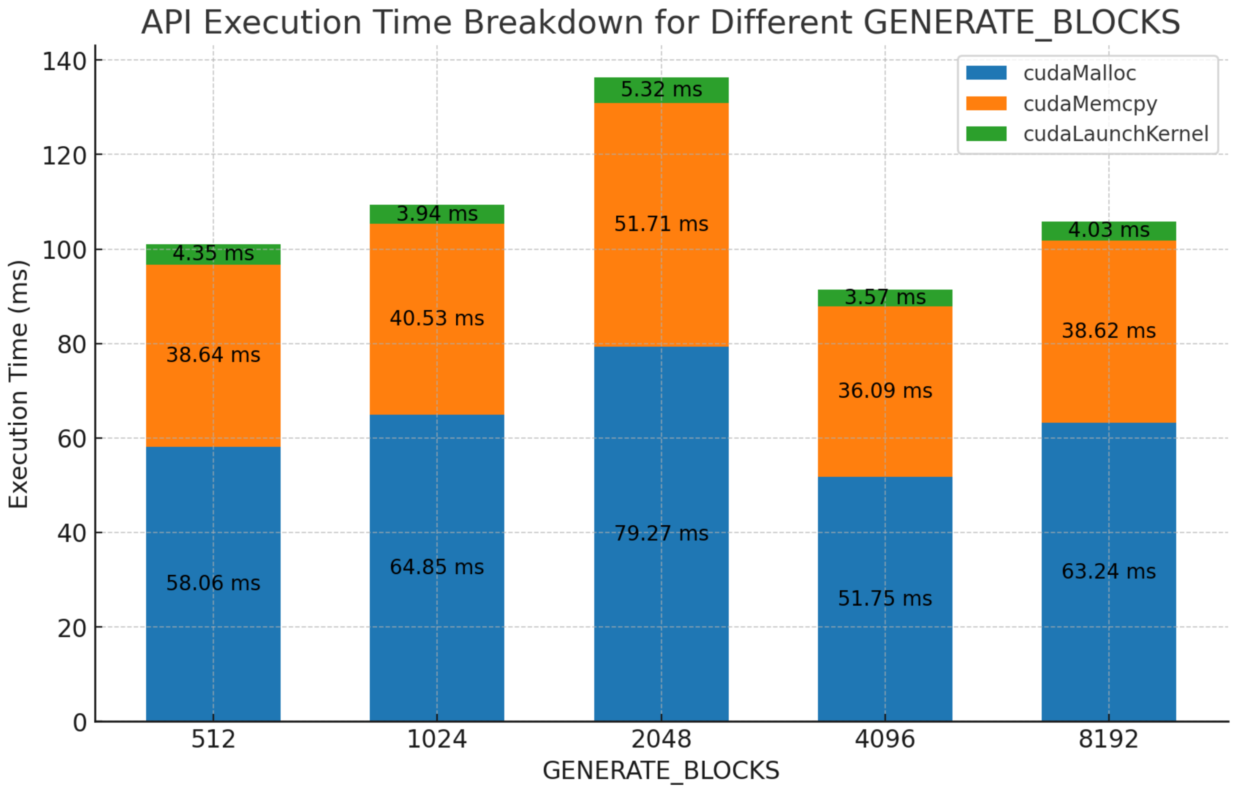
PART B: Monte Carlo estimation of the value of π

1. **Altering MC_SAMPLE_SIZE (1e5 to 1e9):**



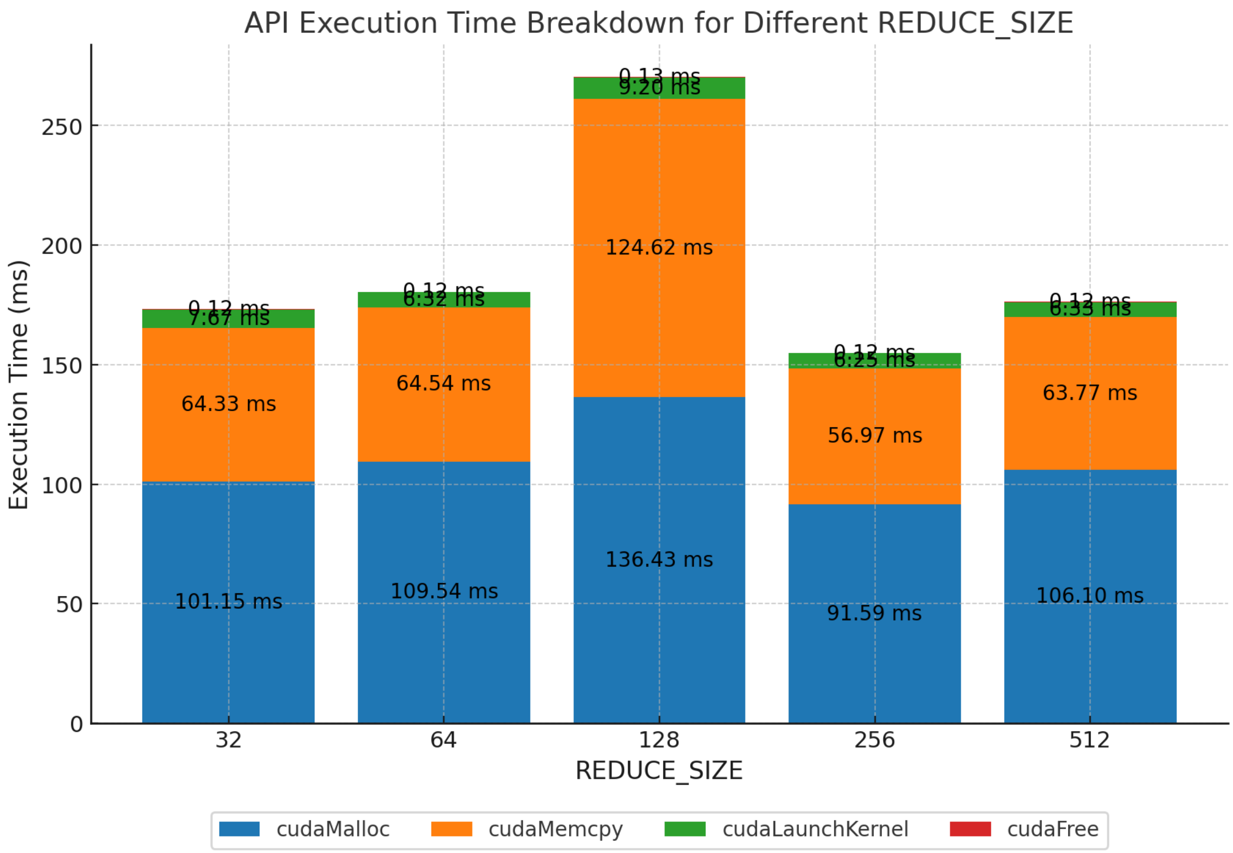
From the result, as MC_SAMPLE_SIZE increases, the execution time scales proportionally, dominated by kernel execution (generatePoints). For smaller sample sizes, memory transfers (cudaMemcpy) and other API calls contribute a noticeable portion to execution time. However, as sample size grows, kernel execution becomes the primary bottleneck, while memory overhead remains relatively constant. This suggests that Monte Carlo Pi estimation is **compute-bound** rather than memory-bound, with performance primarily limited by kernel execution rather than data transfers.

2. Altering GENERATE_BLOCKS (512 to 8192):



For API calls, the results indicate that increasing GENERATE_BLOCKS does not significantly impact execution time, suggesting that performance gains plateau beyond a certain point. To optimize performance, finding the best GENERATE_BLOCKS value requires balancing parallelism and computational efficiency, avoiding excessive kernel launches that may introduce overhead without substantial speedup.

3. Altering REDUCE_SIZE (32 to 512):



For API calls, with the GENERATE_BLOCKS fix at 4096. The results show that varying REDUCE_SIZE affects execution time, with REDUCE_SIZE = 128 leading to the highest overhead. This suggests that an optimal REDUCE_SIZE is needed to balance parallel reduction efficiency and kernel execution. Since GENERATE_BLOCKS is fixed at 4096, selecting a REDUCE_SIZE that minimizes memory transfer and synchronization overhead can improve performance.