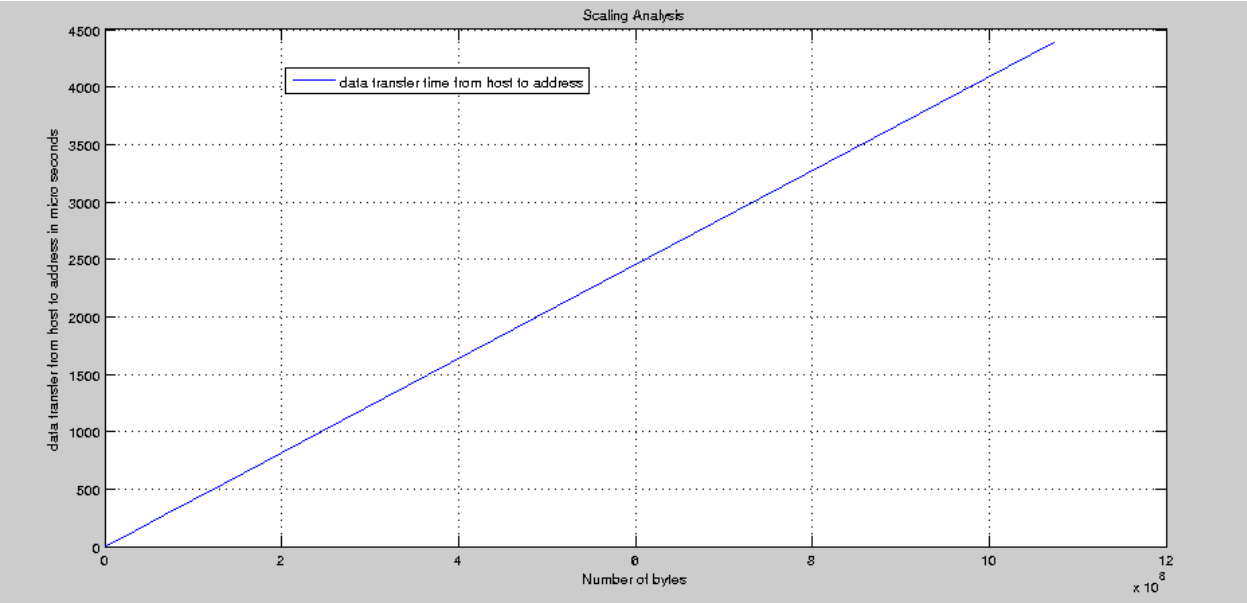
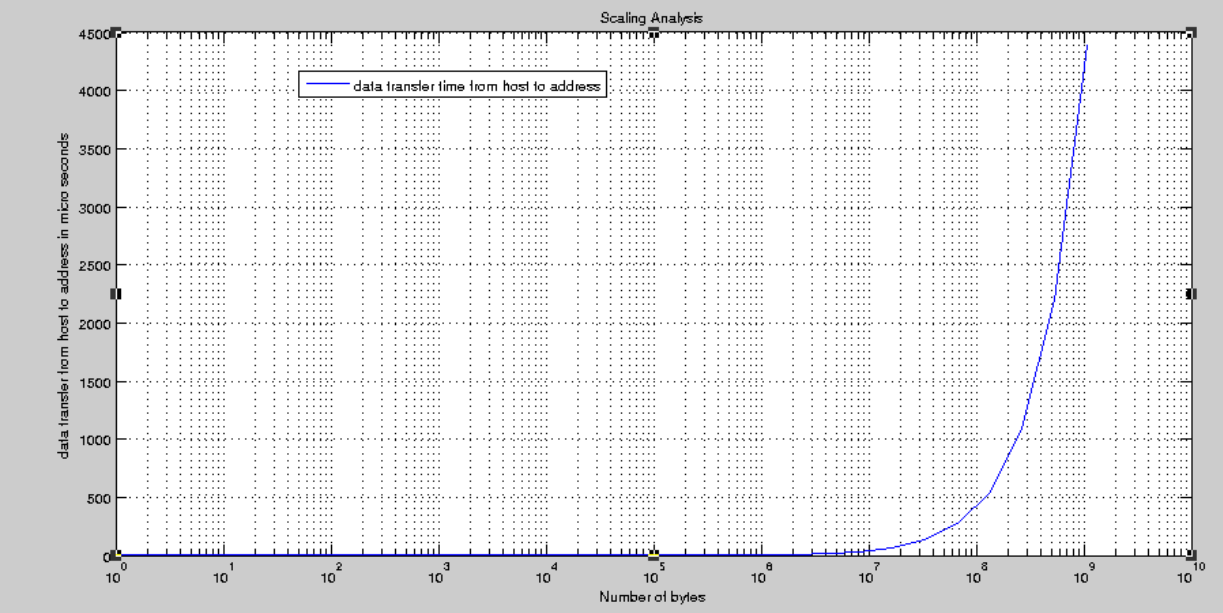
ME759
High Performance Computing for Engineering Applications
Assignment 7
Date Assigned: October 21, 2013
Date Due: October 28, 2013 – 11:59 PM

Problem 1.
Answer 1.1
Please find the code saved as 'p2t.cu'
Plot (linear scale):

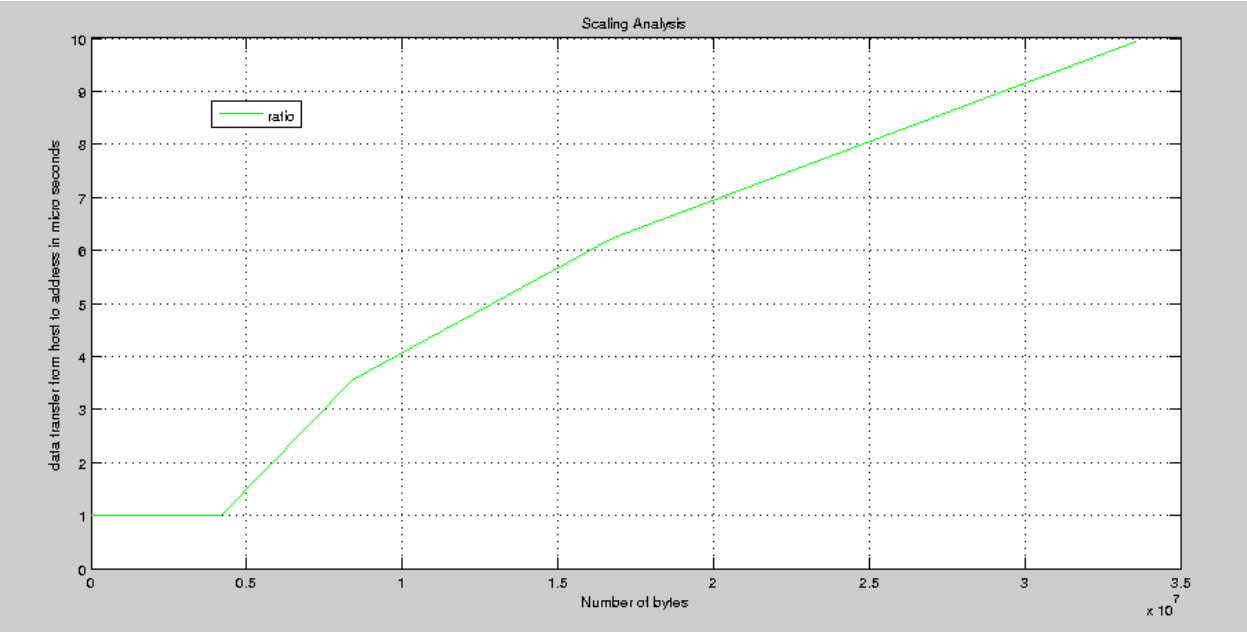


Plot (log scale)



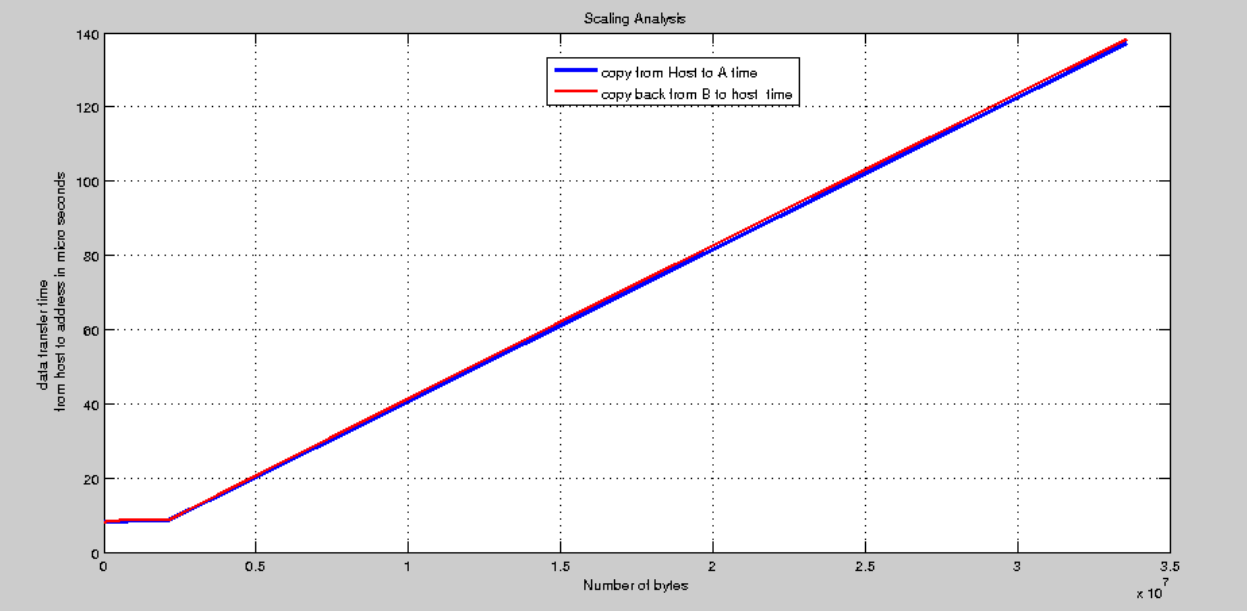
Answer 1.2
The ratio of non-pinned memory transfer to pinned memory transfer is continuously increasing after 2^8 no of bytes

Plot (linear scale)



Answer 1.3: While co-relating with 1.1, the time taken in data transfer from host to memory location A (on device) is more as compared to copying back from memory location B (device) to Host

Plot (linear scale): please find the plot below.



Problem 2.
Revisit the array reduction code you wrote a couple of weeks ago. Specifically, write a piece of code that does the following:

a) Takes as command line arguments two positive integers: the first one, call it N, indicates the size of the array to be reduced; the second one, call it M, indicates the max absolute value of any entry in the array to be generated. Example:
>> myProgr 100000 5
b) Generates on the host an array of N random double precision numbers in the range $-M$ to M (to get a random number in double precision you might first generate an integer then multiply it by 1.0).
c) Uses CUDA code to sum up all the double precision numbers in the array and compares the result against a version of the code that runs on the CPU. A message should be output to report the reduction result on the CPU and on the GPU. Note: there might be small differences in the CPU and GPU results, we'll discuss in class why this can happen.
d) Presents a timing report to indicate the amount of time spent on the CPU and GPU to reduce the array. The GPU should report inclusive time.
Report on the forum (under topic "Timing Results Assignment 7: Reduction Operation") the CPU and GPU times you get when you run your executable using the following input:
>> myProgr 50000000 5
For this problem
i) You should have some checks in place to indicate whether the memory allocation on the device was successful (the TA will try to use your code for an array of one trillion entries). In case of error, the code should exit graciously with a message to indicate the cause of failure
ii) You can use any material covered in class, recycle any code discussed in class, use any information you read from any external source (including CUDA SDK examples). However, no copy-and-paste of code from other

sources. Again, it's ok to copy and paste code from the lecture notes.
For this problem, the TA will list the winner of the speed competition on the forum.
There will be two champions:
1) Winner of the fastest CPU solution competition
2) Winner of the fastest GPU solution competition
The TA will use an array larger than $N = 1,000,000$ entries to evaluate your code, and $M \leq 10$.

Answer: Carried out modifications in Array reduction code of HW5. Please find the code named as "vector_reduction.cu", kernel code "vector_reduction_kernel2.cuh", computation code "vector_reduction_gold.cpp". The code is running fine for the values as mentioned in the problem statement on Euler.