This assignment must be entirely your own work. If any part of your assignment has been copied, then your mark may be reduced to zero. Any consultation (given/taken) must be acknowledged. Late assignments will be penalized at the rate of three marks per day. In fact, you should aim to submit the assignment several days before the due date because extensions will not be granted for reasons such as lack of resources.

Submit all Assignment4 written/printed solutions to your lecturer and any PDF/WORD document of your solutions (of the form xxxx_*.* : replace xxxx with your last name) to mganesh@mines.edu and to the grader breyes@mymail.mines.edu.

1. Show that the Amdahl's Law estimate for maximum speedup is same as that given by Gustafson-Barsis's Law. If a particular algorithm spends 10% of its time executing something sequential and if the other 90% has been parallelized perfectly, what will the maximum speedup be on any Sayers Lab machine using all four cores? Justify your answer.

2. Consider an algorithm with $T_P = T_{COMP,P} + T_{COMM,P}$ and $T_1 = PT_{COMP,P}$. Show that

$$S_P = P\left(1 + \frac{T_{COMM,P}}{T_{COMP,P}}\right)^{-1}, \qquad E_P = \left(1 + \frac{T_{COMM,P}}{T_{COMP,P}}\right)^{-1}. \tag{1}$$

(Note: We use notation introduced in lectures.) Let the algorithm be characterized by the data size $N$. Let $T_{COMP,P}$ and $T_{COMM,P}$, respectively, be proportional to $N/P$ and $P$. Let the proportionality constant for $T_{COMM,P}$ be ten times that for $T_{COMP,P}$. Let $N = 100,000$. Determine the maximum speedup, the associated number of cores $P$ (and hence experimental serial fraction) for this problem, by considering the speedup as a function of the number of processing cores $P$. Justify your answer. Determine the value of $P$ where the parallel efficiency goes below 10%. Justify your answer.

3. Consider the parallel summation problem $D_N = \sum_{i=1}^{N} d_i$ with $N$ divisible by $P$. Write down the parallel tasks $\widetilde{D}_j, j = 1, \cdots P$, so that $D_N = \sum_{j=1}^{P} \widetilde{D}_j$. Suppose that the communication time required for the final sum is proportional to $\log_2 P$ so that $T_P = N/P + c \log_2 P$. Determine the speedup and efficiency in this case. Is this algorithm scalable? Is this algorithm scalable with respect to memory? Justify your answer.

4. Consider a scalable algorithm characterized by the data size $N$. Let the algorithm be implemented using $P$ processing cores. Using the fact that the speedup is bounded by the inverse of the serial fraction, $f(N)$, of the algorithm, prove that $f(N) \leq c/P(N)$ for some proportionality constant $c$. Using this bound and Amdahl's Law prove that

$$f(N) \leq \frac{c}{[P(N) - 1] + P(N)/c}.$$

5. Consider a problem involving two parallel tasks $a$ and $b$, leading to a combined task $a + b$. For $c = a, b, a + b$, let $T_1^c$ and $T_P^c$ respectively denote the sequential time and parallel time for the task $c$. Assume that $T_1^{a+b} = T_1^a + T_1^b$ and $T_P^{a+b} \geq T_P^a + T_P^b$. Prove that the speedup for the combined tasks satisfies

$$S_P^{a+b} \leq \left[\frac{\lambda}{S_P^a} + \frac{1-\lambda}{S_P^b}\right]^{-1}, \qquad \lambda = \frac{T_1^a}{T_1^a + T_1^b}.$$