

Problem 5

Susie's implementation is better. In Joe's implementation, each process computes a chunk of neighboring rows of the Mandelbrot image, whereas in Susie's implementation, each process computes every p^{th} row. This means that some of Joe's processes will finish their tasks earlier than the others and sit idle, waiting to send their results (we would see an outside-in behavior, meaning the the first and the last processes finish working first, and then the second and second to last, etc.). This is because, close to the top and bottom edges, the computation takes much less time compared to the middle of the image. Susie's implementation distributes these areas to processes in a more balanced way, and hence will perform better. I would promote Susie.

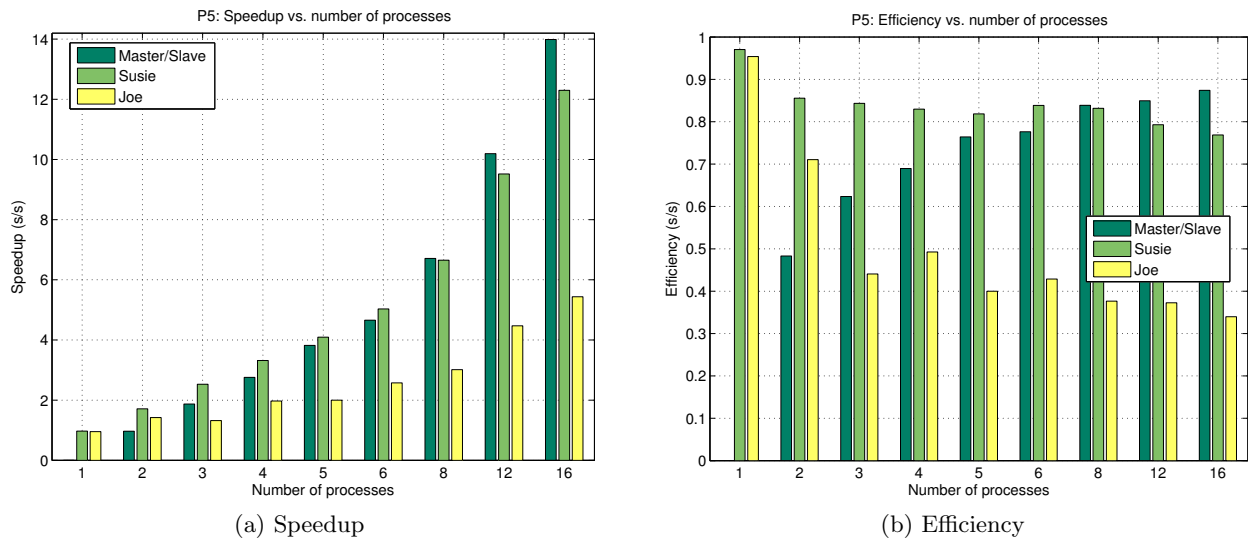


Figure 1: Speedup and efficiency plots for $P5$ executed on Resonance

In Figure 1 we see the Susie's algorithm is in fact a lot better than Joe's. The Master/Slave implementation catches up to and exceeds the performance of the other algorithms when there are 8 or more processes. The Master/Slave implementation really takes off with more cores because none of the processes (maybe except for the master) are wasting time sitting idle. They get assigned jobs once they finish their work. Even though the processes in Susie's algorithm terminate around the same time, as the number of processes increases, the cumulative time wasted idling gets noticable.