I plan to analyze my system with two experiments. Each experiment seeks to profile of a certain preemption policy's efficacy with respect to the number of resources (CPUs) available. The question that I would like to answer with my experiments is "does a learned, adaptive time quantum for each process minimize process wait time"?

The first experiment will vary the number of CPUs in the system while utilizing a fixed time quantum for each process. I will report the standard deviation and mean of the process wait times over the course of a run for some large number of steps T. This provides a baseline metric for process wait time.

The second experiment will use a reinforcement learning approach to learning a time quantum. I will again vary the number of CPUs but this time use adaptive preemption and take measures of the length of time processes were waiting during the course of the run. I plan to implement adaptive preemption using a linear function approximating agent. After a process leaves the system, it will be rewarded (maybe negatively) based on how long that process waited until it was completed relative to the amount of time it required to run. The wait time updates the agent's understanding of the system and will hopefully improve over time until it reaches an acceptable minimum.

Both experiments will report average metrics from many realizations of runs of the same parameter regime (i.e, for each number of CPUs, I will run $N = 10$ different simulations and take the average of the results from each simulation).