# 15-418/618 Spring 2021
# Exercise 4

| | |
|---|---|
| Assigned: | Thur., Mar. 4 |
| Due: | Mon., Mar. 15, 11:00 pm |

## Overview

This exercise is designed to help you better understand the lecture material and be prepared for the style of questions you will get on the exams. The questions are designed to have simple answers. Any explanation you provide can be brief—at most 3 sentences. You should work on this on your own, since that's how things will be when you take an exam.

You will submit an electronic version of this assignment to Canvas as a PDF file. For those of you familiar with the LaTeX text formatter, you can download the template and configuation files at:

http://www.cs.cmu.edu/~418/exercises/ex4.zip

Instructions for how to use this template are included as comments in the file. Otherwise, you can use this PDF document as your starting point. You can either: 1) electronically modify the PDF, or 2) print it out, write your answers by hand, and scan it. In any case, we expect your solution to follow the formatting of this document.

## Problem 1: Resource Oriented Scaling

John Gustafson proposed a new formulation of speedup, in response to concerns about applying Amdahl's Law to some problem domains. For Gustafson's law, let $s$ be the fraction of execution that is inherently sequential (with dependencies preventing parallel execution), and $p$ be the speedup of the execution part that can benefit from from the improvement of computing resources. Then the speedup can be expressed as $S = s + (1 - s)p$.

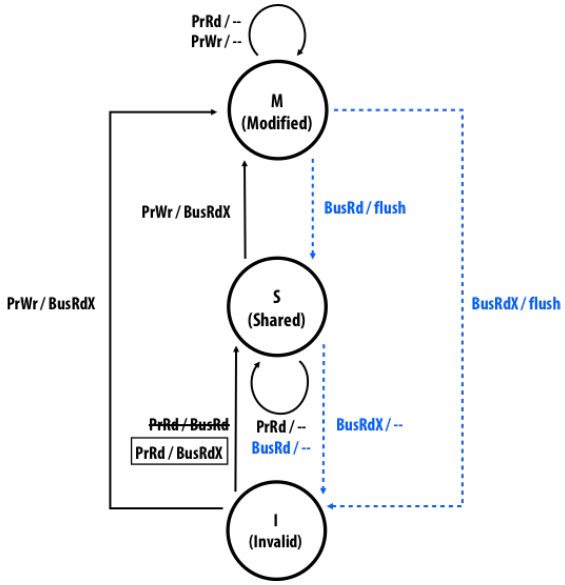
Compare briefly how the two scaling laws (Gustafson's and Amdahl's) relate to the scaling constraints taught in the lecture on workload-driven performance evaluation (**problem-constrained scaling, memory-constrained scaling and time-constrained scaling**). Which constraint(s) is similar to the scaling expressed by each law? Why?

http://www.cs.cmu.edu/~418/lectures/10_perfeval.pdf.

(Recall Amdahl's law: let s be the fraction of execution that is inherently sequential , then the maximum speedup in latency due to parallel execution is $S \leq 1/s$. More specifically, the speedup is $S \leq \frac{1}{s+\frac{(1-s)}{n}}$ where $n$ is the number of processors used.)

## Problem 2: Cache coherency

A. Your friend suggests modifying the MSI coherence protocol so that PrRd / BusRd behavior on the I-to-S transition is changed to PrRd / BusRdX, as is shown below:




Is the memory system still coherent? What impact does this change have on the system?

B. When we plot a parallel application's cache miss rate as a function of the cache block size, we often see a U-shaped curve where the miss rate initially decreases with larger block sizes, but then it starts to increase. Please explain this phenomenon.