concurrency scaling

Scaling

Scaling is how execution time or number of requests handled improves as resources are added.

Performance depends massively based on execution environment variables such as RAM, number of cores in CPU, program being executed, and the data the program is handling.

Amdahl's Law
Amdahl's law calculates the maximum speed up available by parallelisation based on what proportion of a program can be run in parallel.

$$s_{total} = \frac{1}{(1-p) + \frac{p}{s}}$$

- s_{total} = runtime
  - p = fraction of the program that benefits from parallelization
  - s = number of cores used
  - as s approaches infinity, $\frac{p}{s}$ approaches 0, and benefit becomes less noticeable

Strong Scaling
Strong scaling is when the same program with the same data has an execution time that scales linearly with the resources available to it.

For instance, a calculation happens twice as fast when twice the resources are allocated to it.

There will always be diminishing returns with this type of scaling.

Weak Scaling
Weak scaling is when a program can handle more data as more resources are allocated to it.

For instance, a server has the same response time when it receives twice as many requests but has twice the resources allocated to it.

This type of scaling does not necessarily have diminishing returns.