Amdahl's law

In Amdahl's law, computational workload W is fixed while the number of processors that can work on W can be increased.

Denote the execution rate of i processors as R_i, then in a relative comparison they can be simplified as $R_1 = 1$ and $R_n = n$. The workload is also simplified. We assume that the workload consists of sequential work αW and n parallel work $(1-\alpha)W$ where α is between 0 and 1. More specifically, this workload can be written in a vector form as, $W = (\alpha, 0, ..., 0, \alpha - 1)W$, or, $W_1 = \alpha W$, $W_n = (1-\alpha)W$, and $W_i = 0$ for all $i \neq 1, n$.

The execution time of the given work by n processors is then computed as,

$$T_n = \frac{W_1}{R_1} + \frac{W_n}{R_n}$$

Speedup of n processor system is defined using a ratio of execution time, i.e.,

$$S_n = \frac{T_1}{T_n}$$

Substituting the execution time in relation W gives,

$$S_n = \frac{W/1}{\dfrac{\alpha W}{1} + \dfrac{(1-\alpha)W}{n}} = \frac{n}{1+(n-1)\alpha} \qquad (1)$$

Eq.(1) is called the Amdahl's law. If the number of processors is increased infinity, the speedup becomes,

$$S_\infty = \frac{1}{\alpha} \qquad (2)$$

Notice that the speedup can NOT be increased to infinity even if the number of processors is increased to infinity. Therefore, Eq.(2) is referred to as a sequential bottle neck of multiprocessor systems.

Gustafson's Law

This law says that increase of problem size for large machines can retain scalability with respect to the number of processors.

Assume that the workload is scaled up on an n-node machine as, s

$$W' = \alpha W + (1-\alpha)nW$$

Speedup for the scaled up workload is then,

$$S_n' = \frac{Single\,\mathrm{Pr}\,ocessorExeutionTime}{n - ProcessorExecutionTime}$$

$$S_n' = \frac{(\alpha W + (1-\alpha)nW)/1}{\frac{\alpha W}{1} + \frac{(1-\alpha)nW}{n}} \quad (3)$$

Simplifying Eq.(3) produces the Gustafson's law:

$$S_n' = \alpha + (1-\alpha)n \quad (4)$$

Notice that if the workload is scaled up to maintain a fixed execution time as the number of processors increases, the speedup increases linearly. What Gustafson's law says is that the true parallel power of a large multiprocessor system is only achievable when a large parallel problem is applied.

Sun and Ni's Law

This one is referred to as a memory bound model. It turns out that when the speedup is computed by the problem size limited by the available memory in n-processor system, it leads to a generalization of Amdahl's and Gustafson's law.

For n nodes, assume the parallel portion of workload is increased by $G(n)$ times reflecting the increase of memory in n-node system, i.e.,

$$W^* = \alpha W + (1-\alpha)G(n)W$$

The memory bound speedup is then given as

$$S^* = \frac{\alpha W + (1-\alpha)G(n)W}{\alpha W/1 + (1-\alpha)G(n)W/n}$$

Simplification leads to the Sun and Ni's law:

$$S^* = \frac{\alpha + (1-\alpha)G(n)}{\alpha + (1-\alpha)G(n)/n}$$

Depending on $G(n)$, there are three cases:

Case 1: $G(n)=1$

$$S^* = \frac{\alpha + (1-\alpha)}{\alpha + (1-\alpha)/n} = \frac{1}{1+(n-1)\alpha}$$

Notice that this is the Amdahl's law

Case 2: *G(n)=n*

$$S^* = \frac{\alpha + (1-\alpha)n}{\alpha + (1-\alpha)n/n} = \alpha + (1-\alpha)n$$

Notice that this is the Gustafson's law

Case 3: *G(n)>n,*

Let *G(n)=m* where *m>n.* The speedup is then

$$S^* = \frac{\alpha + (1-\alpha)m}{\alpha + (1-\alpha)m/n} = \frac{\alpha + (1-\alpha)m}{m/n + (1-m/n)\alpha}$$

This is the case where the workload grows faster than the memory requirement. Notice that it produces slightly higher speedup than the case of Gustafson's workload.