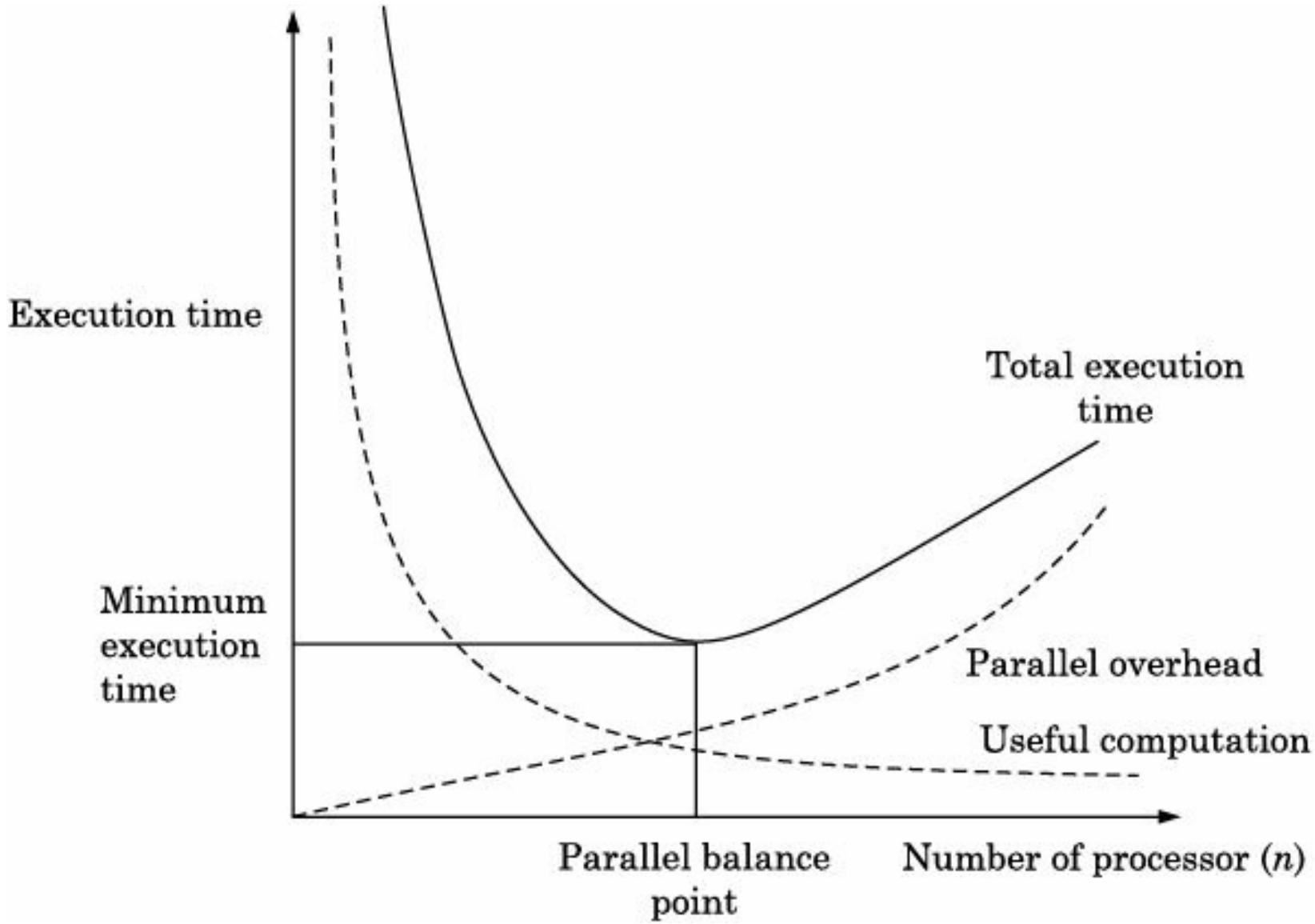
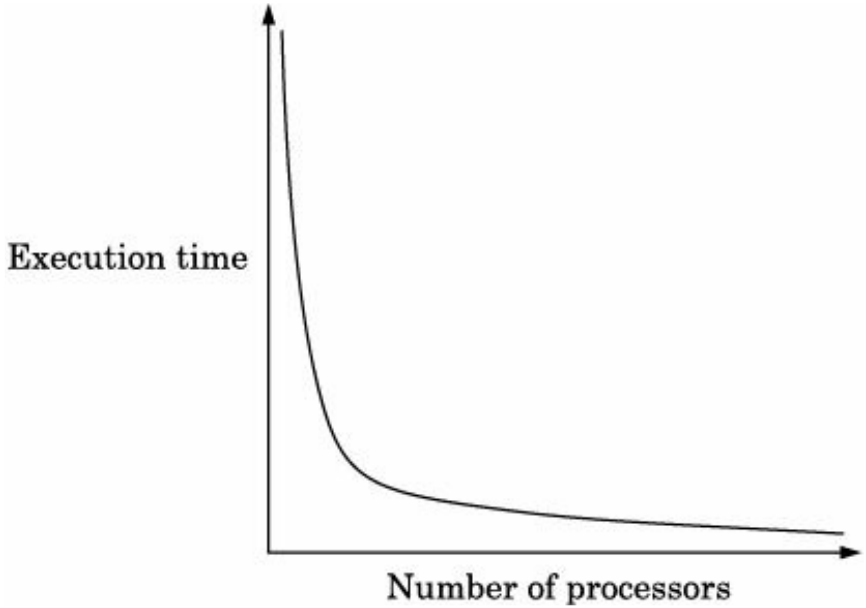
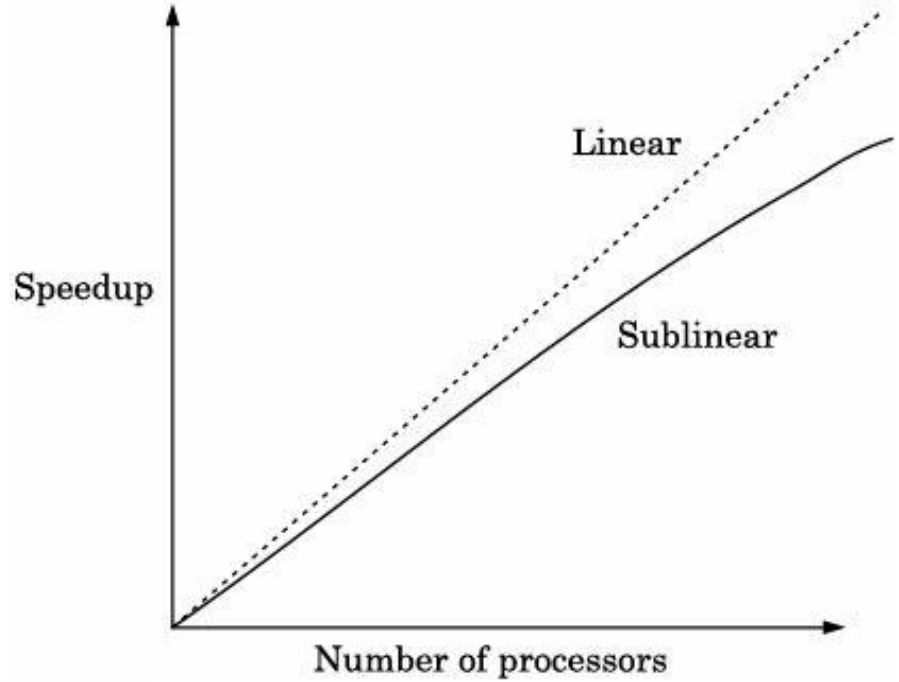

Execution time
Total execution time
Minimum execution time
Parallel overhead
Useful computation
Parallel balance point
Number of processor (n)

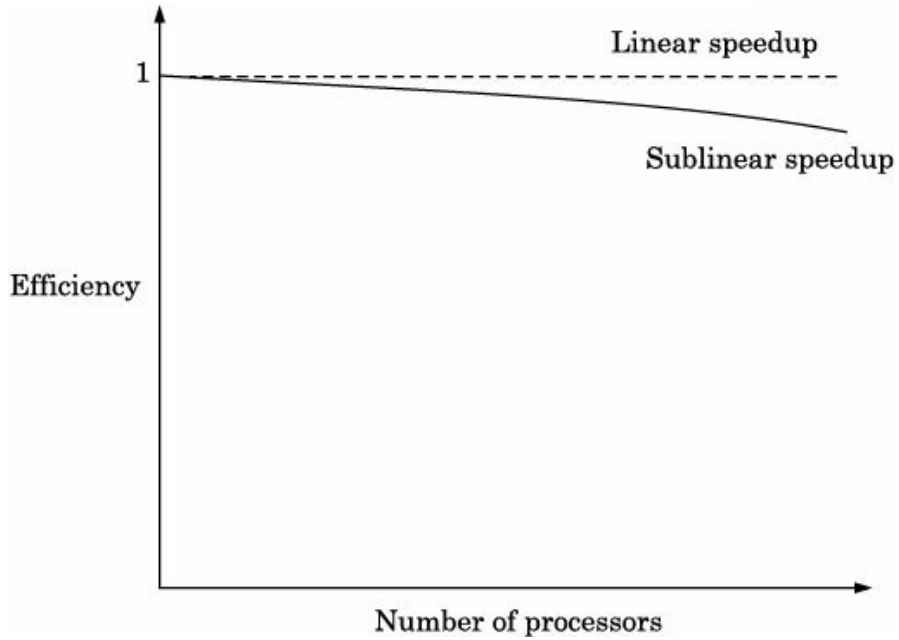
Performance Laws :

- Amdahl’s law is based on a fixed problem size or a fixed work load
- Gustafson’s law is for scaled problems, where the problem size increases with the increase in machine size

  (i.e., number of processors)
- Sun and Ni’s law is applied to scaled problems bounded by memory capacity.


Execution time
Number of processors


Linear
Speedup
Sublinear
Number of processors


Linear speedup
1
Sublinear speedup
Efficiency
Number of processors

Amdahl's law for fixed workload

Basics of performance evaluation:-

Seq Alg

⇓

It is evaluated in terms of its execution time which is expressed as func of i/p size.

Parallel Alg

⇓

Its execution time depends not only on i/p size but also on factors like Hw arch, no of processors etc.,

- parallel run time
- speed up
- efficiency

} performance Metrics

⇒ parallel run time

⇒ speed up

⇒ efficiency

} performance Metrics

① Parallel run time:

$T(n)$

n = no: processors.

when $n = 1$

$T(1)$ denotes Sequential runtime of prog on Single processor

⇒ parallel run time

⇒ speed up

⇒ efficiency

} performance Metrics

① Parallel run time:

$T(n)$ — n = no: processors.

when $n = 1$

$T(1)$ denotes sequential runtime of prog on single processor

② Speed up :-

It is the ratio of time taken to run on single processor to the comp with identical processors.

$$S(P) = \frac{T(1)}{T(n)} = \frac{T(s)}{T(P)}$$

$$\boxed{S(n) = \frac{T(1)}{T(n)}}$$

Efficiency:

As the ratio of speed up achieved & the no: of processors used to achieve it.

$$E(n) = \frac{S(n)}{n} = \frac{T(1)}{T(n) \cdot n}$$

$$E(n) = \frac{T(s)}{n \cdot T(p)}$$

Amdahl's law

⇒ Is a formula, used to find

(1) - max improvement possible by improving particular part of a sys

⇒ In HLC, computing it is mainly

(2) used to predict the

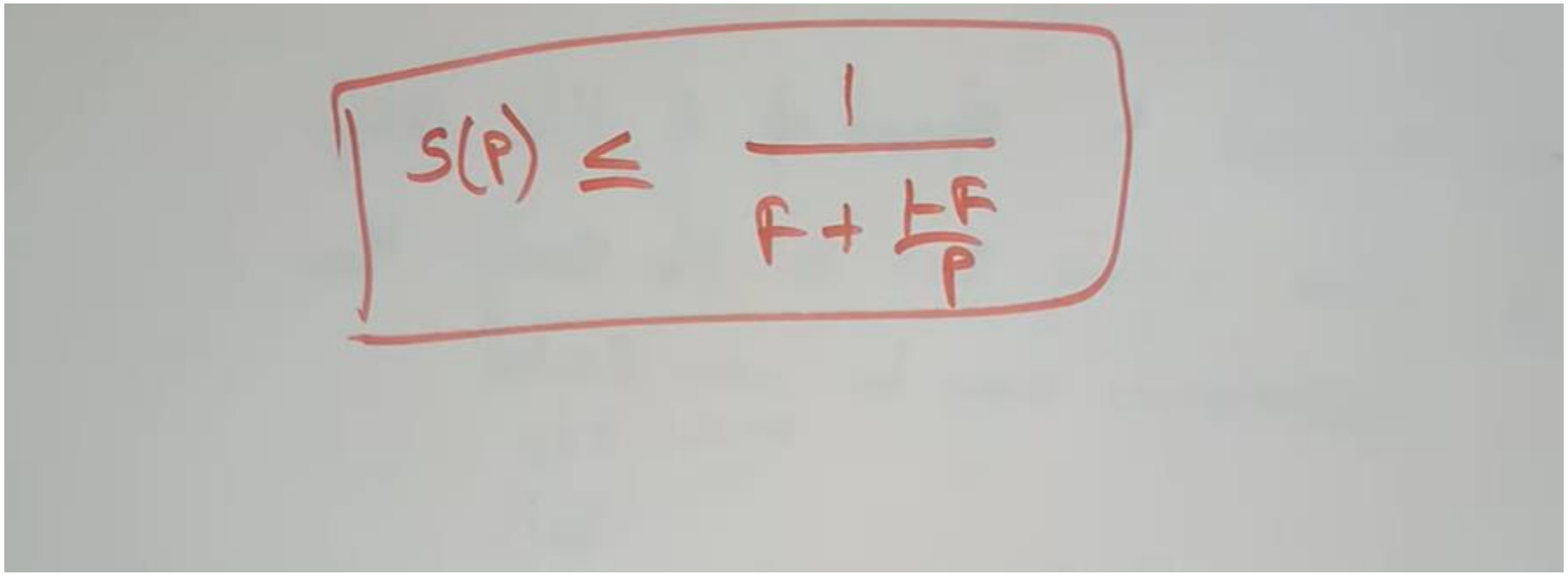
- theoritical max. speed up for prog processing using multiple processors.

⇒ related to speed up of ||le comp

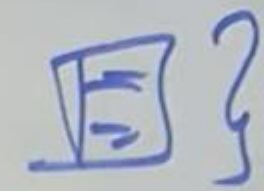
⇒ when a prog run on ||le comp then computation may be
- Serial
- ||le
- both

⇒ there will be a certain part of prog consider sequential fraction of prog → F

||le computation of prog = 1 − F


$S(p) \leq \frac{1}{f + \frac{1-f}{p}}$

$$S(P) \leq \frac{1}{F + \frac{1-F}{P}}$$



P = no: of processors

$$S(P) = \frac{T(s)}{T(P)} \cdot \frac{T(1)}{T(P)} \quad \text{——— (1)}$$

Serial.
parallel = seq + parallel

Execution time Single processor = T

" nth processor : sequential computing + nth computing time

$$\therefore T(P) = \frac{F \cdot T}{1} + \frac{(1-F) \cdot T}{P}$$

F
1-F

$$S(P) = \frac{T(S)}{T(P)}$$

$$= \frac{1}{F(1) + (1-F)\frac{1}{P}}$$

$$= \frac{1}{F + \frac{(1-F)}{P}} \quad \text{—— (2)}$$

$S(P) \leq 1$

$$\Rightarrow \boxed{S(P) \leq \frac{1}{F + \frac{(1-F)}{P}}}$$

$\frac{1}{2}$

$= 0.5$

0.25