一、基本指标

工作负载=执行时间+浮点运算数+指令数目

并行执行时间 T_{comput} 为计算时间，T_{paro} 为并行开销时间，T_{comm} 为相互通信时间

$$T_n = T_{comput} + T_{paro} + T_{comm}$$

通信开销Tcomm

点到点

* 通信开销 $t(m) = t_0 + m/r_\infty$
* 通信启动时间 t_0
* 渐近带宽 r_∞：传送无限长的消息时的通信速率
* 半峰值长度 $m_{1/2}$：达到一半渐近带宽所要的消息长度
* 特定性能 π_0：表示短消息带宽

$$t_0 = m_{1/2}/r_\infty = 1/\pi_0$$

整体通信

典型的整体通信有：

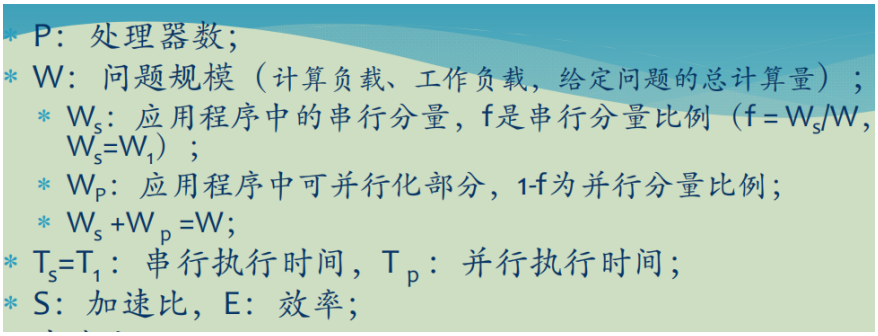
* 广播（Broadcasting）：处理器0发送m个字节给所有的n个处理器
* 收集（Gather）：处理0接收所有n个处理器发来在消息，所以处理器0最终接收了m n个字节；
* 散射（Scatter）：处理器0发送了m个字节的不同消息给所有n个处理器，因此处理器0最终发送了m n个字节；
* 全交换（Total Exchange）：每个处理器均彼此相互发送m个字节的不同消息给对方，所以总通信量为mn^2个字节；
* 循环移位（Circular-shift）：处理器i发送m个字节给处理器i+1，处理器n-1发送m个字节给处理器0，所以通信量为m n个字节。

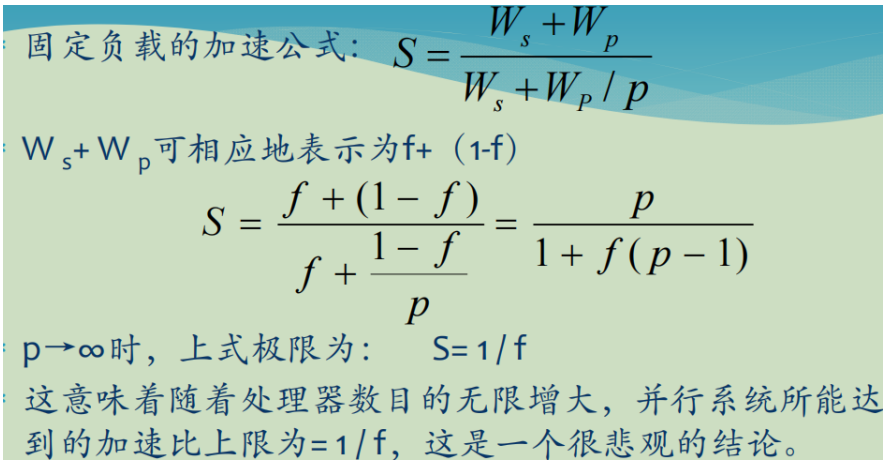
二、加速比性能定律

- 加速比：并行算法比串行算法快多少倍

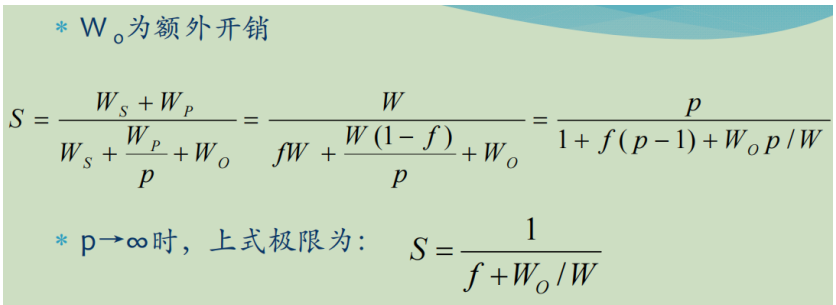
1、Amdahl定律

- 适用于固定计算负载，对实时性要求高的
- 固定计算分在可分布在多处理器上



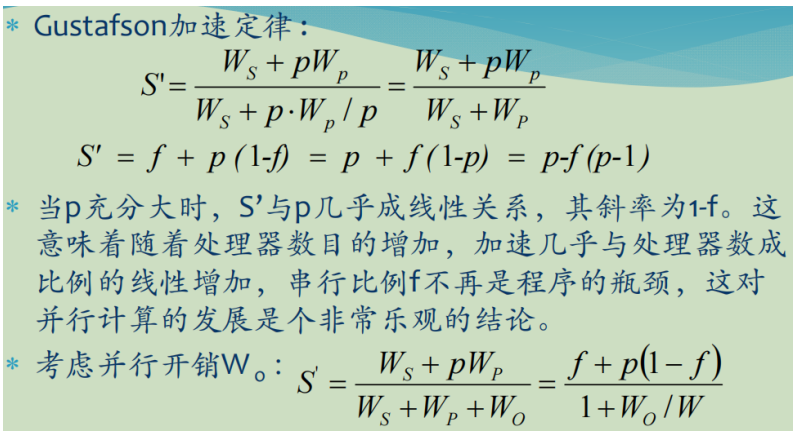


并行额外开销越大，加速越小



2、Gustafson

- 适用于可扩放问题
- 为提高精度，时间不变就加大计算量
- 增多处理器必须是增大了实际问题规模才有意义



3、Sun Ni

- 受限于存储器，以存储空间换更好的解

假定在单节点上使用了全部存储容量M并在相应于W的时间内求解之，此时工作负载W= fW + （1-f）W。

在p 个节点的并行系统上，能够求解较大规模的问题是因为存储容量可增加到pM。令因子G（p）反应存储容量增加到p倍时并行工作负载的增加量，所以扩大后的工作负载W = fW + （1-f）G（p）W。

* 存储受限的加速公式：

$$S'' = \frac{fW + (1-f)G(p)W}{fW + (1-f)G(p)W/p} = \frac{f + (1-f)G(p)}{f + (1-f)G(p)/p}$$

* 并行开销W_O:

$$S' = \frac{fW + (1-f)WG(p)}{fW + (1-f)G(p)W/p + W_O} = \frac{f + (1-f)G(p)}{f + (1-f)G(p)/p + W_O/W}$$

* G（p）=1时就是Amdahl加速定律；
* G（p）=p 变为 f + p（1-f），就是Gustafson加速定律
* G（p）>p时，相应于计算机负载比存储要求增加得快，此时 Sun和 N i 加速均比 Amdahl 加速和 Gustafson 加速为高。

4、加速比

- 线性：无通信开销的，矩阵运算，内积运算
- p/logp：分治类

* 通信密集类的应用问题：S = 1 / C （p）
* 超线性加速
* 绝对加速：并行算法与最佳串行算法
* 相对加速：同一算法在单机和并行机的运行时间

绝对加速针对科研

相对加速针对业界

三、可扩放性评测标准

可扩放性：在确定的应用背景下，性能随处理器数的增加而按比例提高的能力

> 能否有效利用增加的处理器，用可扩放性来度量

- 并行计算要调整的是处理数p和问题规模W

影响加速的因素

- 串行分量
- 并行额外开销（通信、等待、竞争、冗余操作和同步等）
- 处理器个数超过算法并发程度

增加问题规模可提高加速原因

- 并发度提升
- 额外开销增加幅度小于有效计算
- 串行分量占比减小

研究目的

* 确定解决某类问题用何种并行算法与何种并行体系结构的组合，可以有效地利用大量的处理器；
* 对于运行于某种体系结构的并行机上的某种算法当移植到大规模处理机上后运行的性能；
* 对固定的问题规模，确定在某类并行机上最优的处理器数与可获得的最大的加速比；
* 用于指导改进并行算法和并行机体系结构，以使并行算法尽可能地充分利用可扩充的大量处理器

1、等效率度量标准

等效率函数fE（p）：效率E不变时，W随p变化的函数

图 4.6 给出了 3 种等效率函数曲线，曲线 1 表示算法具有很好的扩放性，曲线 2 表示算法是可扩放的，曲线 3 表示算法是不可扩放的。

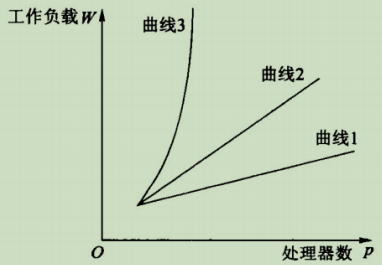
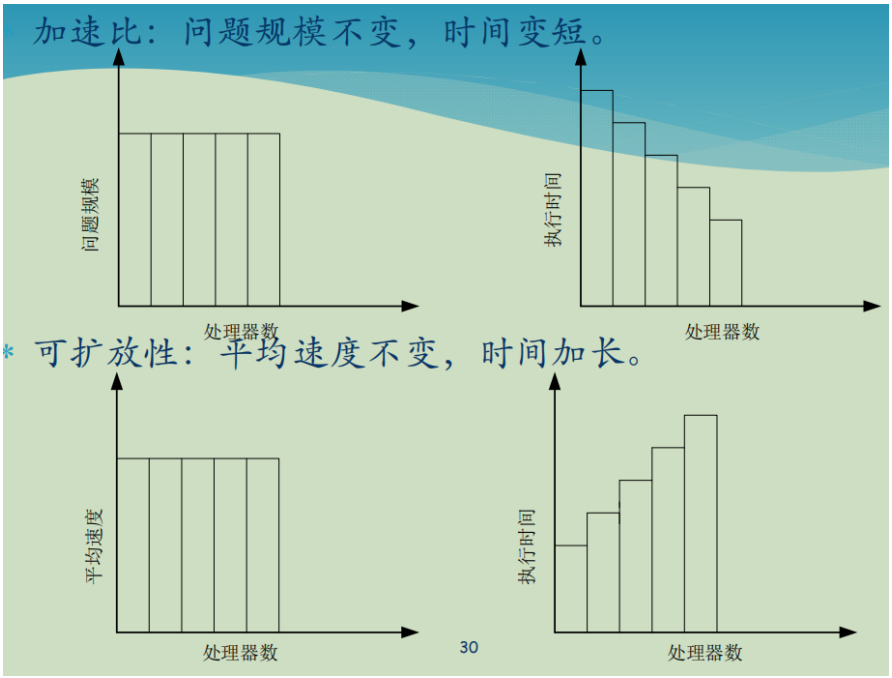


图 4.6 等效率函数曲线

- 适用于开销T0易于计算的情况
- 最大优点是，可用简单的、可定量计算的、少量的参数就能计算出等效率函数
- 缺点：共享存储的并行计算机中，T0则主要是非局部访问的读 / 写时间、进程调度时间、存储竞争时间以及高速缓存一致性操作时间等。很难计算！

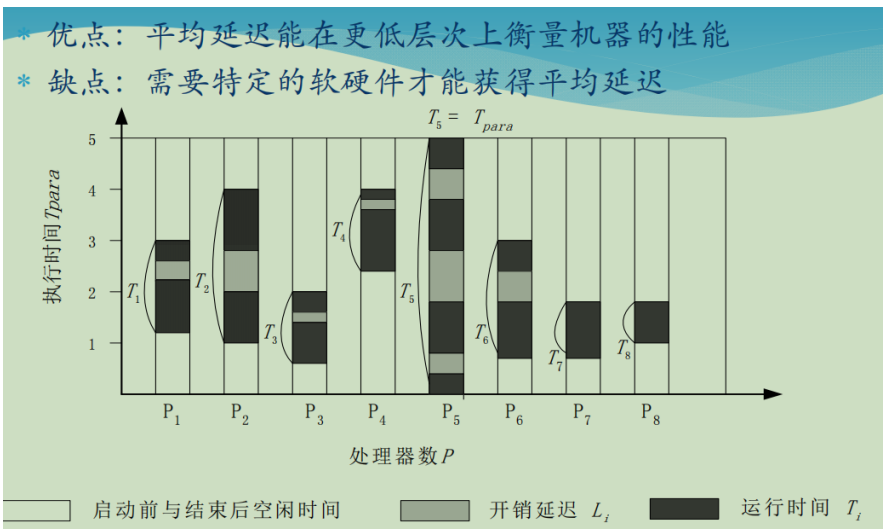
2、等速度度量标准

可扩放：运行在并行机上的算法，当处理器数目增加时，增大一定工作量能维持平均速度不变



- 最大优点：用机器性能速度来度量可扩放性，比较直观

3、平均延迟开销



4、总结比较

- 基本出发点都是抓住了影响算法可扩放性的基本参数 To

* 等效率度量标准
  * 在保持效率E不变的前提下，研究问题规模W如何随处理器个数p而变化。
* 等速度度量标准
  * 在保持平均速度不变的前提下，研究处理器个数p增多时应该相应的增加多少工作量W。
* 平均延迟度量标准
  * 在保持效率E不变的前提下，用平均延迟的比值来标志随处理器个数p增加需要增加的工作量W。
* 三种度量可扩放性的标准是彼此等效的。