

"CLASSICAL" PARALLEL PROGRAMMING

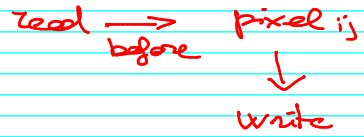
→ WHAT IS PARALLEL PROGRAMMING

→ phases SEQ → PAR

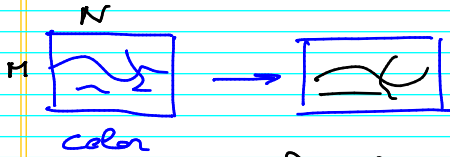
→ implementation details

use of 2 or more PE in combination solving a single problem

② coordination between // activities

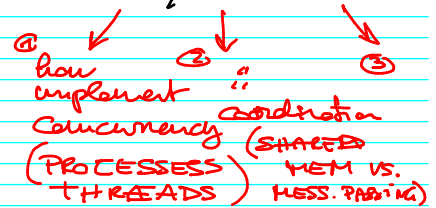


① concurrent activities which? how many?



$f: \text{color pixel} \rightarrow \text{bw pixel}$

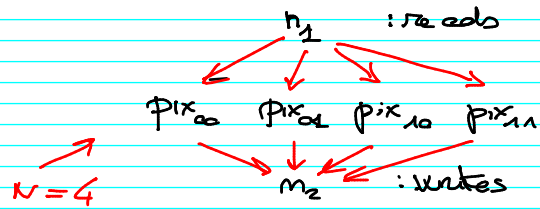
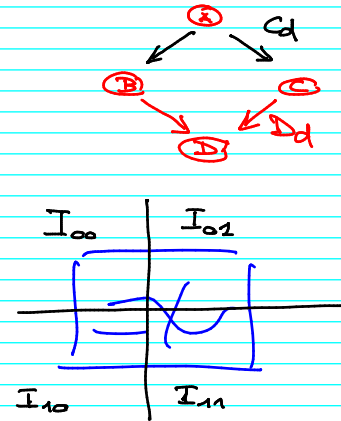
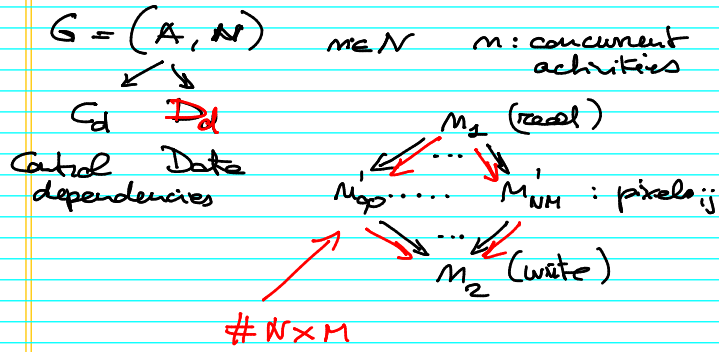
③ implementation takes into account hw/sw archie



$A = \left\{ \begin{array}{l} V_{ij} : \text{pixel } ij \xrightarrow{\text{color}} \text{bw } ij, \\ \text{read original image,} \\ \text{store BW image} \end{array} \right\}$

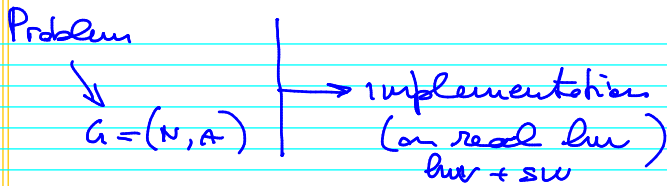
③ MAPPING SCHEDULING

Concurrent activities graph



```

for (i=0; i<4; i++)
    pthread_create ( ..... );
for (i=0; i<4; i++)
    pthread_join ( ..... );
    
```



FUNCTIONAL CODE

Specifies
what is to be
computed

depends on the
problem we consider

(DOMAIN SPECIFIC CODE)

APPLICATION

PROGRAMMERS

+

NON FUNCTIONAL
CODE

specifies
how the
parallel comp
is organized

(SPECIFIC OF THE
TARGET
ARCHITECTURE)

SYSTEM

PROGRAMMERS

Reason: faster applications

Prog: P solution t
(seq)



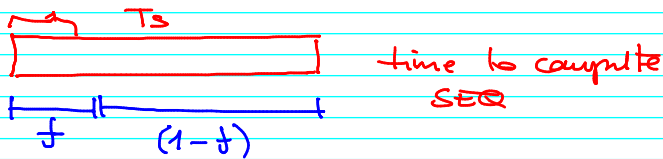
PAR solution using m PE
solution in t/m

$$\text{speedup}(m) = \frac{T(1)}{T(m)}$$

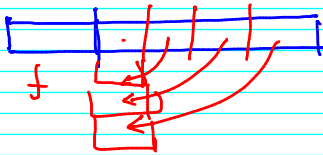
$$sp(m) = \frac{T_s}{fT_s + \frac{(1-f)T_s}{m}} =$$

$$\lim_{m \rightarrow \infty} sp(m) = \frac{T_s}{fT_s} = \frac{1}{f}$$

Amdahl Law



10% SEQ 90% CAN BE MADE PAR



$$\frac{fT_s + \frac{(1-f)T_s}{m}}{m}$$
$$\lim_{m \rightarrow \infty} = fT_s$$

OVERHEAD

↳ necessary (NON FUNCTIONAL CODE)

NON FUNCTIONAL CODE

↳ impairs speedup

* performance/parallel implementation

$$sp = \frac{T_s}{fT_s + \frac{(1-f)T_s}{m}} \rightarrow \frac{(1-f)T_s}{m} + T_{ov}(m)$$

* fault tolerance

ability to complete appl in presence of faults (err/svr)

$$\lim_{m \rightarrow \infty} sp(m) = \lim_{m \rightarrow \infty} \frac{T_s}{fT_s + \frac{(1-f)T_s}{m} + T_{ov}(m)}$$

HTBF

$$= \frac{T_s}{fT_s + T_{ov}(m)} = \frac{1}{f + \frac{T_{ov}(m)(1-f)}{T_s}}$$

↑↑
NON FUNCT CODE