30 Lezione

martedì 14 maggio 2019 14:08

DISCUSSION ABOUT THE PROJECT

→ Pick 1 exercise

2 kind of project

Application project	Framework project
• SWO	• BSP (sort)
• SKYLINE	• AUTONOMIC FARM
• FREE APPL	• FREE PAT

For the free choice: if you have particular interest in one field.
- Write a text to explain the idea
- Ask for permission

Provide 3 version of the application:

→ Sequential code (use it as reference to study the scalability)

→ C++ with thread
→ fastflow
} Compare with the sequential one

Sequential ① → C++ ② → fastflow ③ → Use the Building block

before going to C++ or fastflow solution do the analysis on the code (how much you can parallelise .... and the other analysis that we have seen in the course)

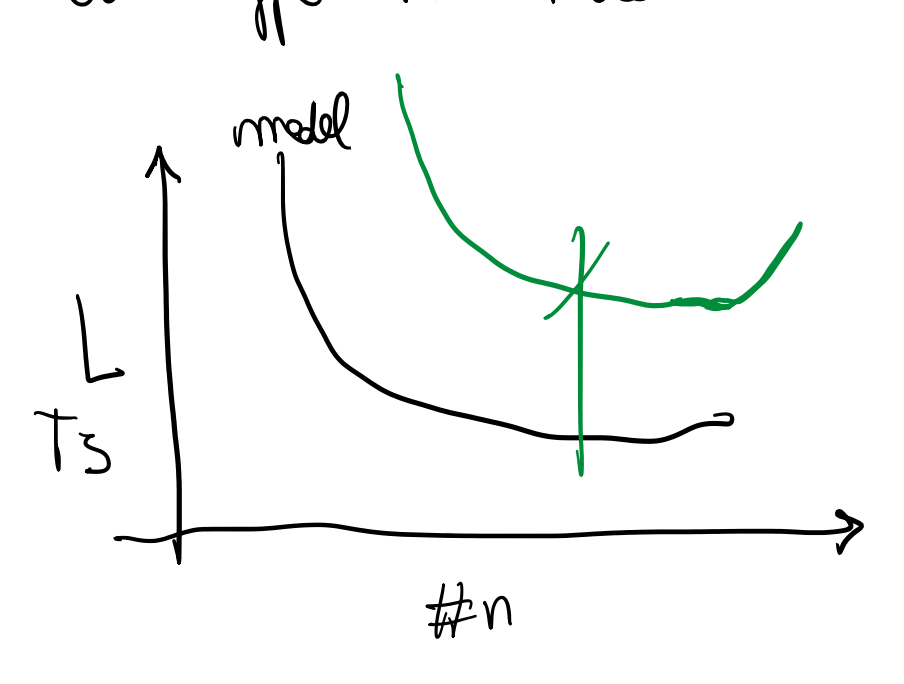
Good practice: the sequential code is not parallelized but is easy to parallelize.

- Limit the access to the state [14:30]

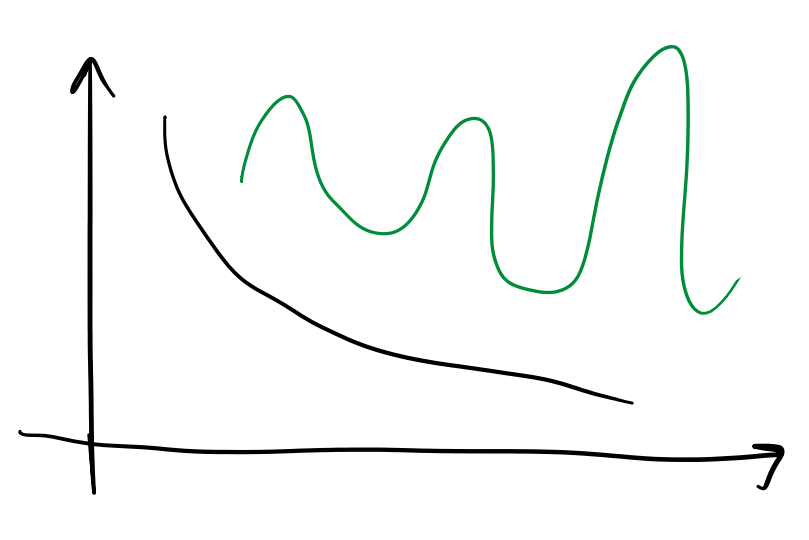
[14:33] ascoltare "is not mandatory but..."

$A_{C++\,par}$ $A_{ff\,par}$

Use the same data as the sequential one to understand what happen in the two different models.



What can happen?



Every model that we use for the composition of pattern are very unfair.

the point of the project is not to have something with a curve that is perfect in the theoretical method. the point is to understand what happen and have reason about the results.

You need to be able to understand what is going on. (Explain the bottleneck ....)

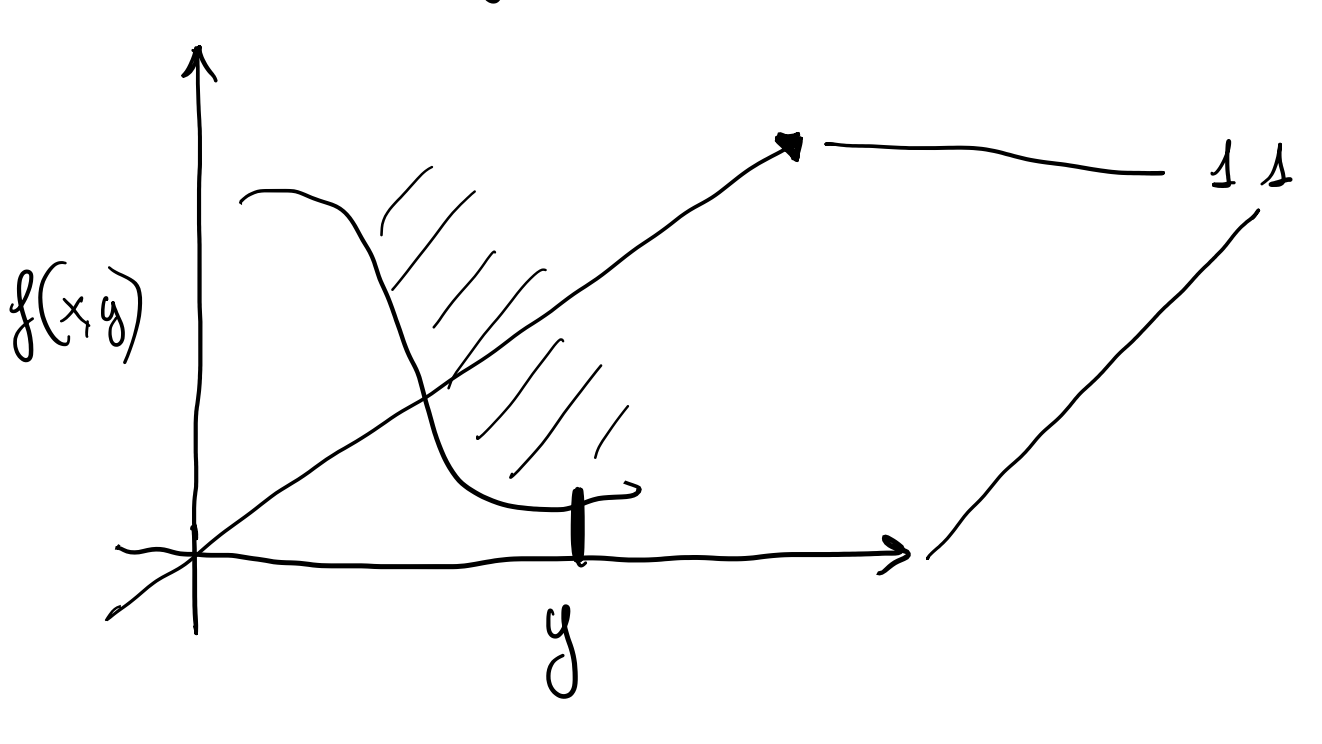
Framework project: understand what is going on. You have to do for the pattern the same things that you have to do for the application.

Measure: seq → {Tseq}

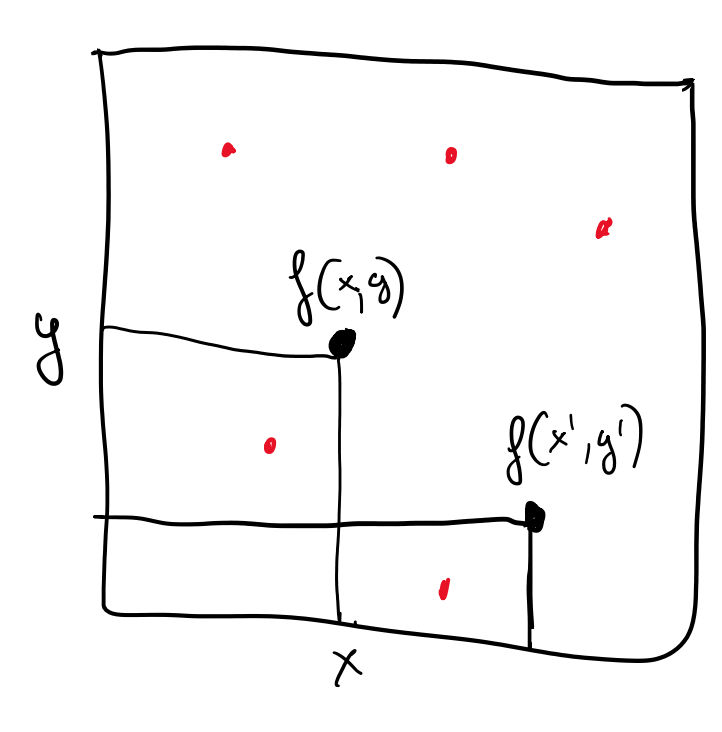
What happen if the ranges are ... [14:42]

Swarm Particle Optimization

Example: we use this to implement min $f(x,y)$



We want to search the minimum of the function;



I start with a random distribution of particles and every particle has an initial speed.

Move around this particles

this particles contributes maintaining a global minimum. At each iteration each particle update his speed and position and also update his contribution to the global minimum.

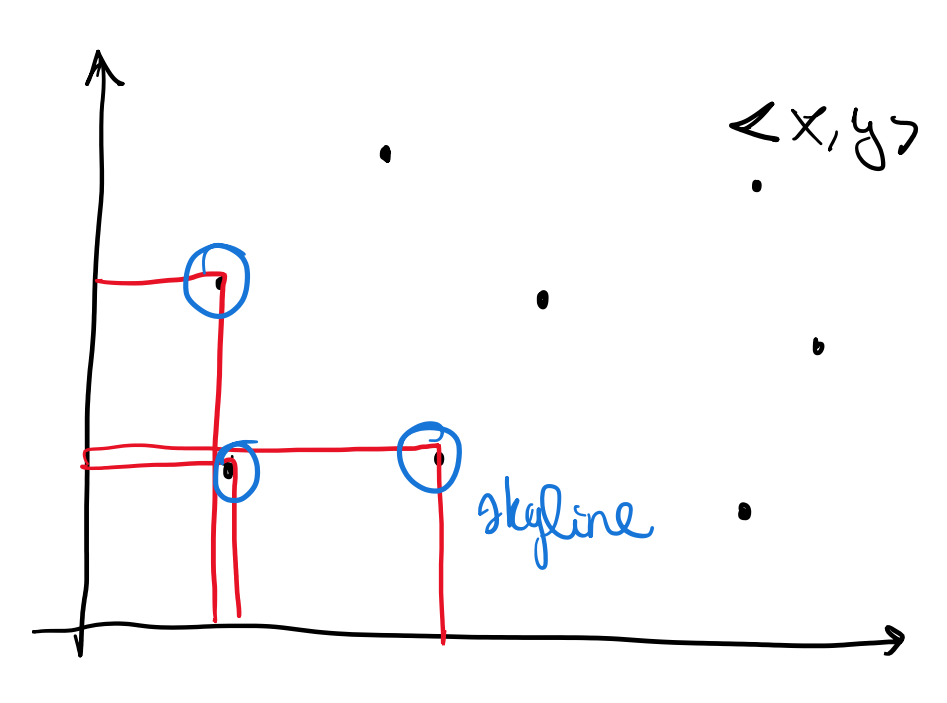
Iteration:

p_i $\vec{V_i}$

$$\vec{V_i}(t+1) = a\,V(t) + b \cdot R_1 (d_{local\ minimum}) + c \cdot R_2 (d_{global\ minimum})$$

$$p(t+1) = p(t) + v(t+1) \quad (t=1)$$

SKYLINE



the skyline is the set of tuples [5:25]

$t_1 = \langle a, b \rangle$ ($<_1$, $<_2$)

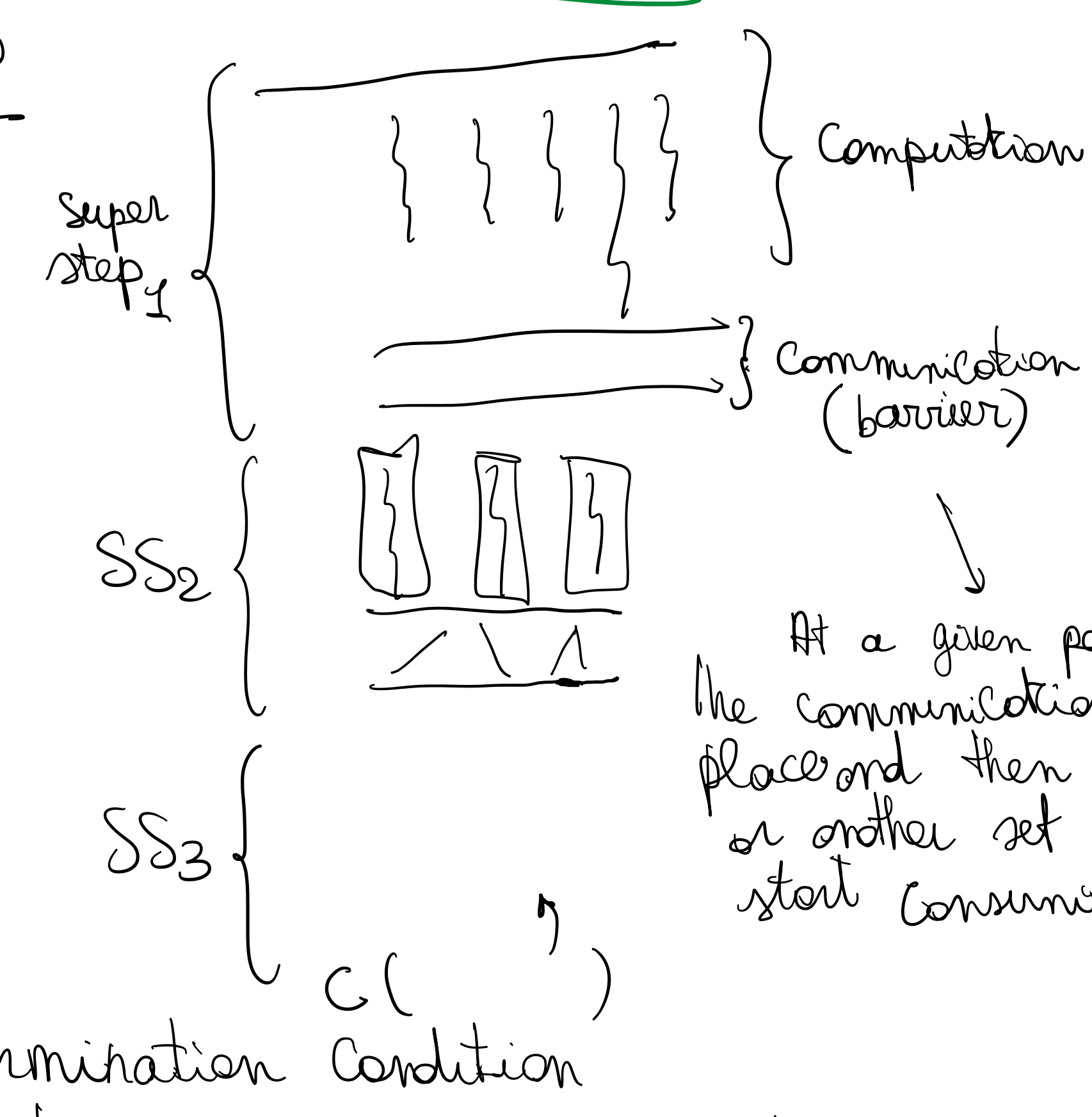
$t_2 = \langle c, d \rangle$

t_1 dominates t_2

iff $a <_1 c$ && $b <_2 d$

FRAMEWORK PROJECT

BSP

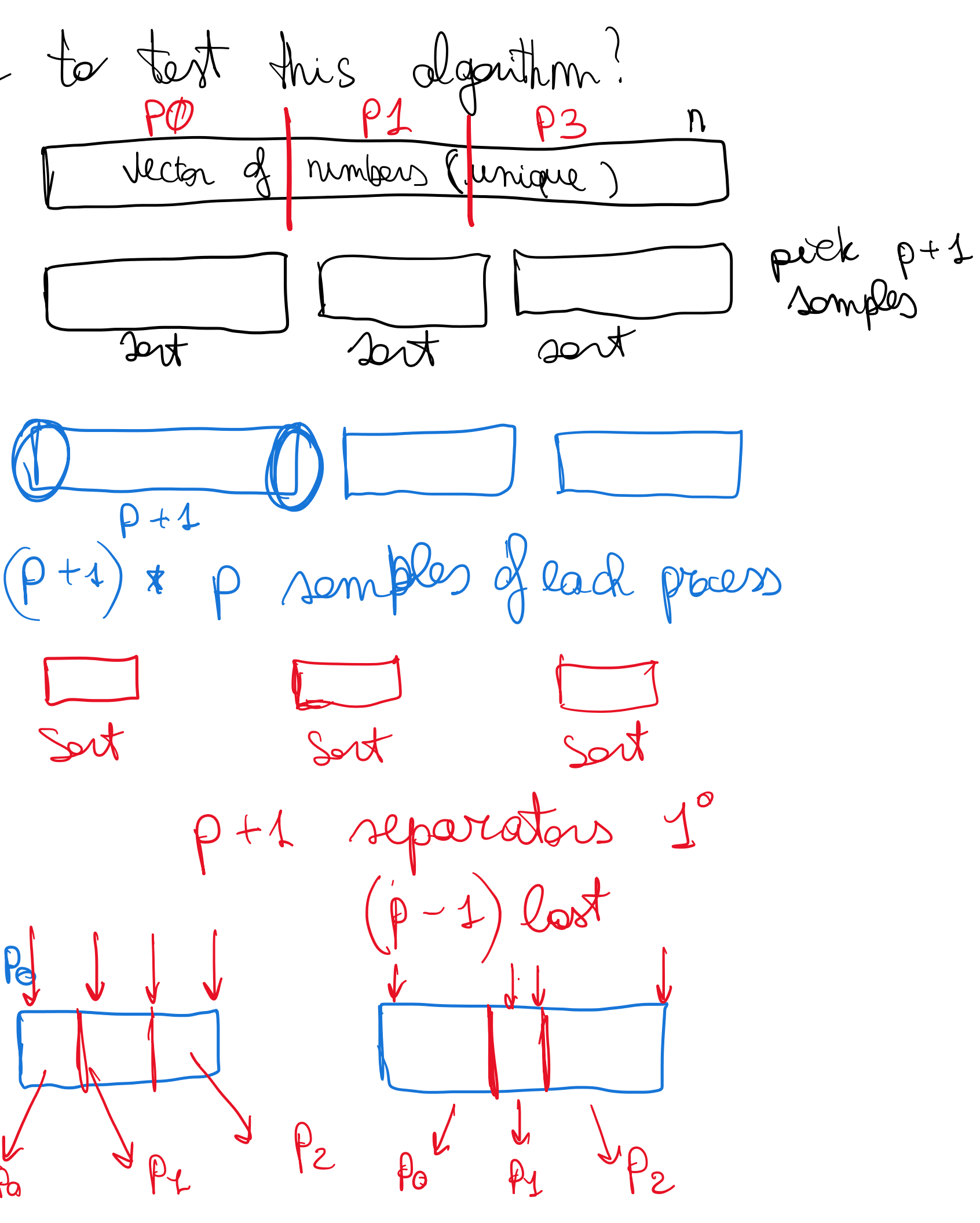


At a given point all the communications takes place and then the same or another set of threads start consuming the data

termination Condition
↳ if evaluted true stop the Computation

What can we use to test this algorithm?

→ Sort algorithm:



SS1

They uses the same number of p = parallel activities

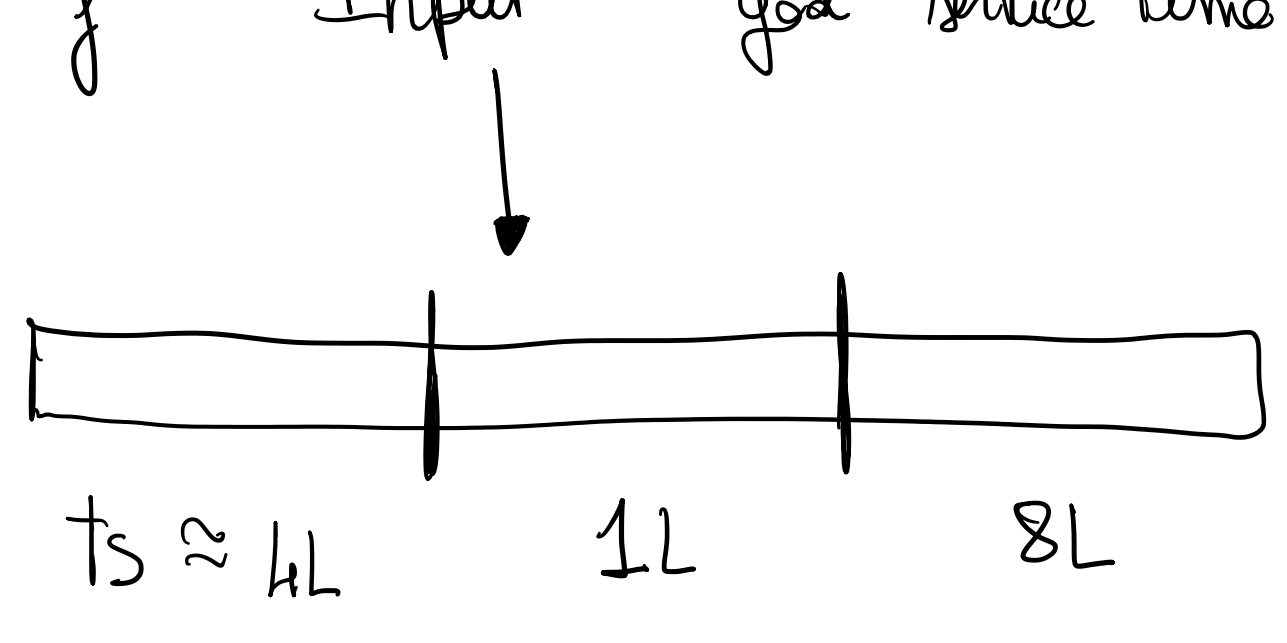
SS2

Suppose p = 3

SS3 — Sort all items received + mines

AUTONOMIC FARM

f Input goal service time (T_S) now



Workers: 4n n 8n

the goal has to be achieved best effort.

Compare what you get with fastflow.