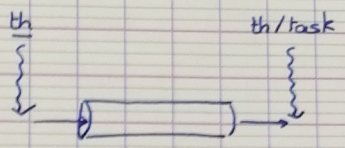
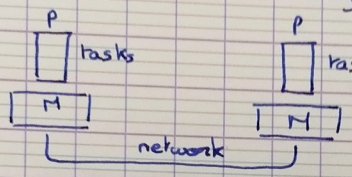
Failure :

1) pompe s'arrete d'exécuter
   1) bloqué
   2) exception



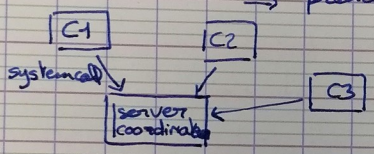
Detecteur de faute parfect ?

Comment ceux vivant vont s'organiser pour fchier ss les morts ?



Soit être humain
Soit process
Peu pas distinguer il répond pas de il est mort
prend decis°

2 types de système → prend decis° en cas de faute
→ prend pas decis°



enter(x)
leave(x)
x ϕ owner
└□□□ queue

Panne franche → soit marche bien soit marche pas
Detecteur de panne parfait → on finit par savoir

- Panne franche d'1 processus client
  soit client ds queue → on supprime le client de la queue
  soit client ds sect° critique → on redache le verrou et on donne au processus suivant

- Panne franche de 2 processus client idem

- Panne franche serveur
  stateless → il se souvient pas de son état après panne
  panne reboot recovery marche
  si serveur plante les clients envoient un msg au serveur

  statefull → sauvegarde état qqpart pour retrouver qd crash
  besoin état verrou + queue, sv qd il se passe qqc ex supp queue
  sv ds 1 fichier qu'on peut recuperer qd reboot

mais ça marche pas

↳ pdt reboot 1 client peut être mort

↳ serveur doit recréer des sockets

detect° fautes → ~~client~~ serveur passe en accept ou client essaient de se reconnecter

& ils savent que serveur reboot (reconstitut° connaiss)

vérifier que y ai pas de clients qui sont morts pdt panne

↳ fixer limites pour clients doivent faire 1 connect après reboot de 1 certaine période sinon ⚠

erreur dès l'encrypt° du fichier

si sv ap dit ce client je pense client a alors qu'il a pas (= infos)

si sv ap dit " je pense client a = que ce qu'il a vraiment

↳ synchronisation de que les ≠ parties savent

statelesss : recréat° connaiss idem statefull

serveur se rappel de rien & client se rappel → recuperer état des clients

↳ peut savoir qui était en sect° critique et qui étaient ds queue mais pas ordre

Si appli demande pas FIFO strict → pose pas de pb

Si appli demande FIFO strict → clients doivent connaître leur posit° ds la file.

Causalité : cause précède csq & choses se diffusent avec certain temps

Ordre total

↳ logical clock Lamport

evnt → +1 LC

P_i process send M(P_i, LC) LC++ (horloge du msg)

receive Max(LC, M(LC)) +1 (mon horloge)

Group{P_i}

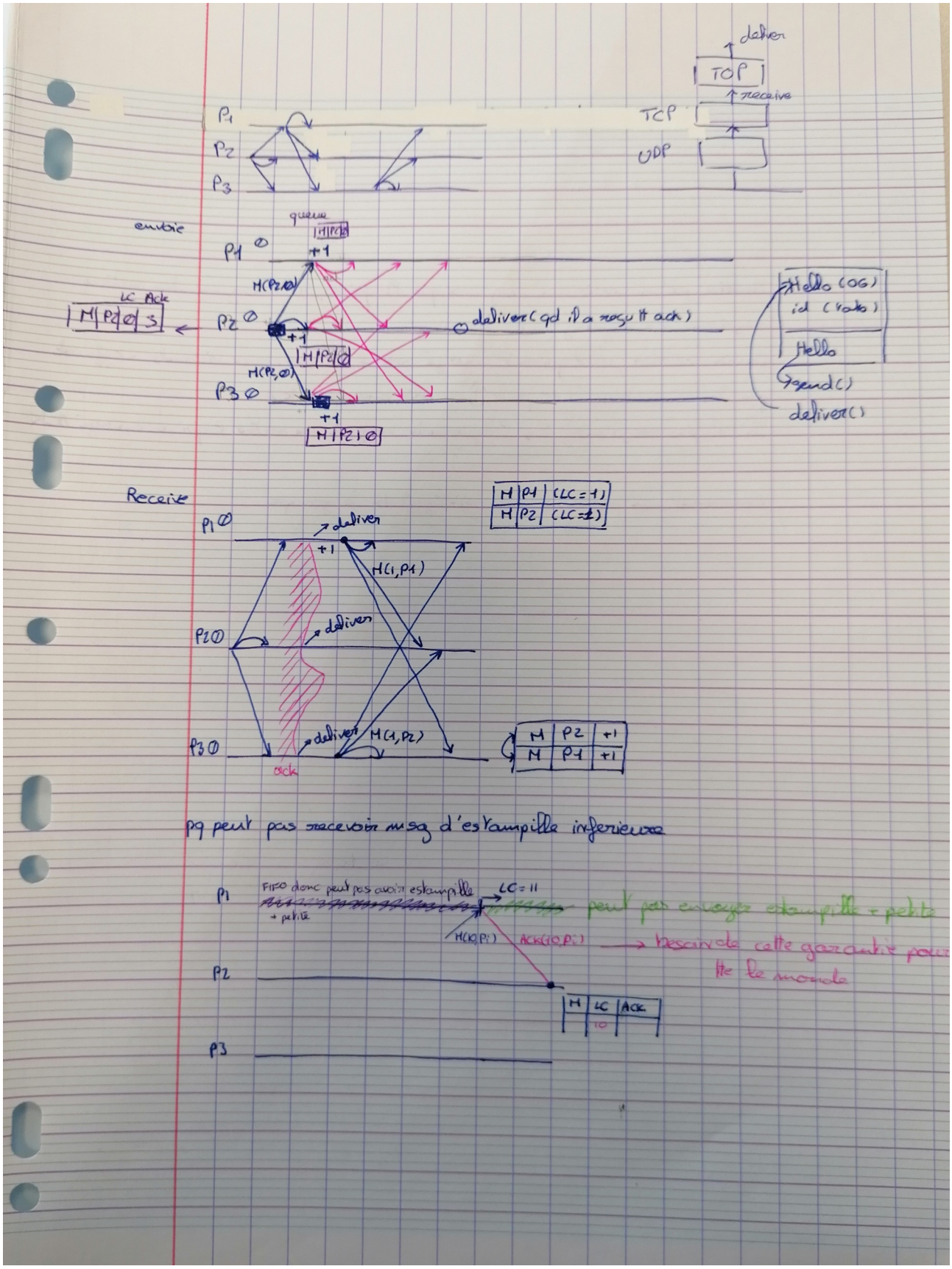
Multicast

0) Fifo / Totalless

1) send to all M(P_i, LC)

2) recept°
- queue ordered LC P_i+
- ack to all

3) deliver top & msg & all ack sommet pile a tt les acks



pq peut pas recevoir msg d'estampille inférieure

Qu'est ce qui se passe si un processus tombe en panne ?

Si P2 envoie un msg à P1 et que P1 tombe en panne avant d'envoyer ack P2 n'aura jamais le bon nb de ack → si panne detecté on attend l'ack de mauvais.

Si P2 panne ack sont envoyé à qui

Marche pdt l'envoie des msg ex que P1 qui reçoit msg → les autres reçoivent ack de P1 mais de msg

Besoin vecteurs ceux qui ont envoyé ack + vecteurs ceux vivants et faire que les 2 matchs (1P peut ☨ après avoir envoyé ack)

Si P pas reçu msg → P demande msg à un autre P qui a envoyé ACK

Comment un processus est réintégré ds le groupe