Systèmes et algorithmique répartis

ENSEEIHT/DIMA, master 2 Informatique
1h45, documents autorisés

décembre 2016

Toutes les réponses doivent être justifiées. Un simple "oui", "non" ou "42" est considéré comme une absence de réponse.
Dans chacune des parties, toutes les questions valent autant.

1 Calcul réparti et causalité (5 points)

On considère les échanges de messages entre 3 sites A, B, C représentés par le chronogramme suivant :

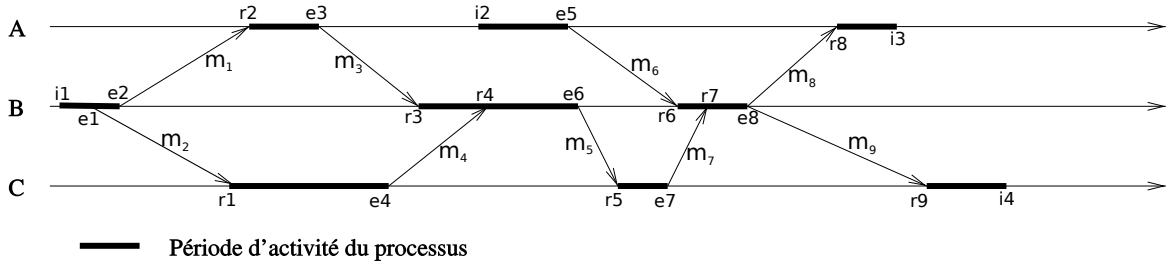


FIGURE 1 – Chronogramme des échanges

Questions

1. Dans le chronogramme de la figure (1), quelle particularité implique que ce dernier ne représente pas un calcul diffusant ?
2. Pourquoi certaines actions des processus (à préciser) ne peuvent pas être considérées dans ce chronogramme comme atomiques ?
3. Les événements i_2 et r_5 sont-ils causalement liés ?
4. Donner une coupure *cohérente* incluant les événements i_2 et e_4.
5. Donner une coupure *non cohérente* incluant les événements i_2 et e_4.
6. Déterminer la valeur de l'horloge vectorielle de l'événement r_4. Justifier cette valeur soit par raisonnement, soit en calculant les horloges des événements le précédant.
7. Déterminer l'histoire causale du message m_6 et en déduire si les délivrances r_6, r_7 respectent la causalité.

2 Problème : reprise après panne (15 points)

La prise en compte des défaillances est un trait caractéristique et essentiel des systèmes répartis. Le problème aborde ici la question de la capture d'un cliché et les protocoles de reprise après panne. Dans ce cas, les protocoles de reprise répartis s'appuient sur la construction d'un cliché formant un état global cohérent. Cet état global forme un *point de reprise* dans lequel on peut remettre le système (*rollback*) si un des sites est défaillant. L'état global est obtenu à partir des états locaux sauvegardés par chacun des sites. Le protocole de prise des états locaux et/ou de restauration doit assurer la cohérence de l'état global restauré.

Question

1. Les mécanismes de reprise après panne ne constituent qu'une classe de solutions au traitement des défaillances. Citer deux autres types de services/mécanismes/protocoles contribuant à la tolérance aux pannes.
2. La restauration d'un état passé cohérent construit à partir de la prise indépendante d'états locaux est sujette à l'effet domino. Expliquer en quoi consiste l'effet domino, et quelle est sa cause.

2.1 Variations autour de Chandy-Lamport

L'algorithme de Chandy-Lamport suppose que les canaux de communication utilisés par les sites sont FIFO. Pour lever cette restriction, on propose

- de numéroter les clichés (on suppose pour cela que l'on dispose d'un mécanisme permettant d'établir un ordre global sur les différentes prises de cliché).
- de superposer à chaque message applicatif le numéro du dernier cliché auquel le site émetteur du message a participé.

Questions

3. Sur quel mécanisme pourrait-on s'appuyer pour implanter une numérotation globale des différents clichés ?
4. Proposer une adaptation (simple) du protocole de Chandy-Lamport utilisant cette numérotation pour construire des coupures cohérentes sans supposer que les canaux de communication sont FIFO.
5. Montrer que cette adaptation simple permet de construire une coupure cohérente mais pas de capturer correctement les messages en transit.

(indications : considérer les deux cas du transparent II-27)

2.2 Protocole de Manivannan-Singhal [1]

Cet algorithme est basé sur la prise de clichés locaux indépendants par chacun des sites, mais il intègre les relations de causalité induites par les échanges de messages applicatifs pour forcer la prise de clichés locaux, qui éviteront l'effet domino et faciliteront le calcul des points de reprise.

On ne traite d'abord que de la cohérence de l'état global sans se préoccuper de la gestion des messages en transit.

2.3 Algorithme de construction des points de reprise

Chaque site possède un compteur sn_i (sequence number) pour numéroter ses clichés locaux. Un cliché local peut être pris spontanément (*When it is time to take a basic checkpoint*) ou forcé (sur réception d'un message). Chaque cliché local est affecté d'un numéro de séquence.

Data Structures at Process P_i
sn_i : integer (:= 0); {Sequence number of the latest checkpoint, initialized to 0. This is updated every time a new checkpoint is taken}

When process P_i sends a message M
$M.sn := sn_i$; {sequence number of the current checkpoint appended to the message M}
send (M);

When it is time for process P_i to take a basic checkpoint
$sn_i := sn_i + 1$;
Take checkpoint C;
$C.sn := sn_i$;

Process P_j, upon receiving a message M from process P_i
If $sn_j < M.sn$ then
$sn_j := M.sn$;
Take checkpoint C;
$C.sn := sn_j$;
Process the message.

La chronogramme figure 2 représente une exécution de ce protocole. Les traits verticaux représentent les prises de cliché spontanées, les carrés représentent les prises de cliché forcées. Les numéros représentent les numéros affectés aux différents clichés locaux. Dans ce qui suit, $C_{i,k}$ désigne le cliché local au site P_i, de numéro k.

Questions sur la figure

6. Montrer que $C_{2,3}$ et $C_{3,3}$ sont indépendants.

Questions dans le cas général

7. Montrer qu'un site i ne traite un message M que après qu'il a pris un cliché local avec un numéro $\geq M.sn$.
8. Montrer que pour tout message M et tout site i, $send(M) \in C_{i,m_i} \Leftrightarrow M.sn < m_i$.

1. *A low-overhead recovery technique using quasi-synchronous checkpointing*, D. Manivannan and M. Singhal, 16th International Conference on Distributed Computing Systems, 1996.

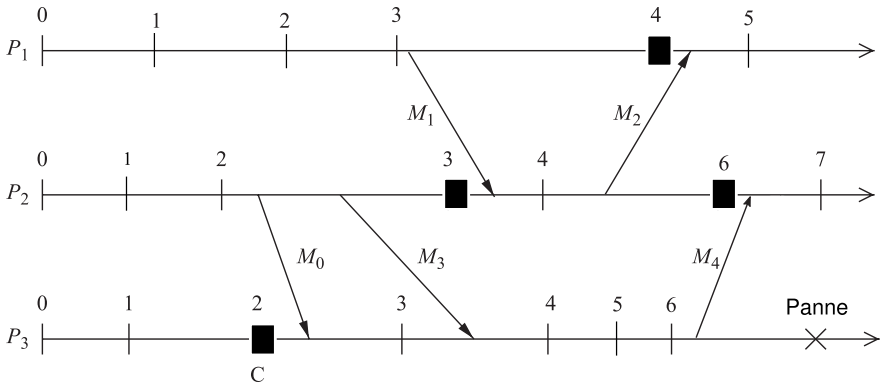


FIGURE 2 – Chronogramme pour Manivannan-Singhal

9. Montrer que pour tout message M et tout site i, $receive(M) \in C_{i,m_i} \Rightarrow M.sn < m_i$.
10. Montrer que l'inverse n'est pas nécessairement vraie.
11. Montrer, en s'appuyant sur la relation de causalité, que pour tous sites i, j avec $i \neq j$, $C_{i,k}$ est causalement indépendant de $C_{j,k}$.
12. En déduire que les $C_{\_,k}$ (avec k fixé), lorsqu'ils existent, forment une coupe cohérente. Illustrer ce résultat sur le chronogramme.
13. La numérotation des clichés pouvant comporter des trous, les $C_{\_,k}$ n'existent pas nécessairement pour tous les sites. Dans le cas général, l'événement $C_{i,k}$ est causalement indépendant de l'événement $C_{j,m}$, où m est le plus petit majorant de k parmi les numéros de cliché de j. Donner la coupe cohérente contenant $C_{1,5}$ et illustrant cette propriété.

2.4 Algorithme de récupération des points de reprise

Quand un site i souhaite construire un point de reprise (= un état global cohérent), il applique l'algorithme suivant. Ce point de reprise est identifié par le sn_i du site i qui a initié la construction du point de reprise.

When process P_i wants to collect a global checkpoint

send $request\_check\_point(i, sn_i)$ to all processes including process P_i;
After receiving $reply(j, m_j)$ from each process P_j,
Declare $S = \{C_{j,m_j} \mid 1 \leq j \leq N\}$ as a global checkpoint;

Process P_j, upon receiving $request\_check\_point(i, m)$ from Process P_i

If $(m > sn_j)$ then
$sn_j := m$;
Take checkpoint C;
$C.sn := sn_j$;
send $reply(j, sn_j)$ to process P_i;
Else
Find the earliest checkpoint C such that $C.sn \geq m$;
send $reply(j, C.sn)$ to P_i.

Questions

Quand un site i reçoit un message $reply(j, m_j)$ du site j en réponse à sa requête $request\_check\_point(i, m)$,

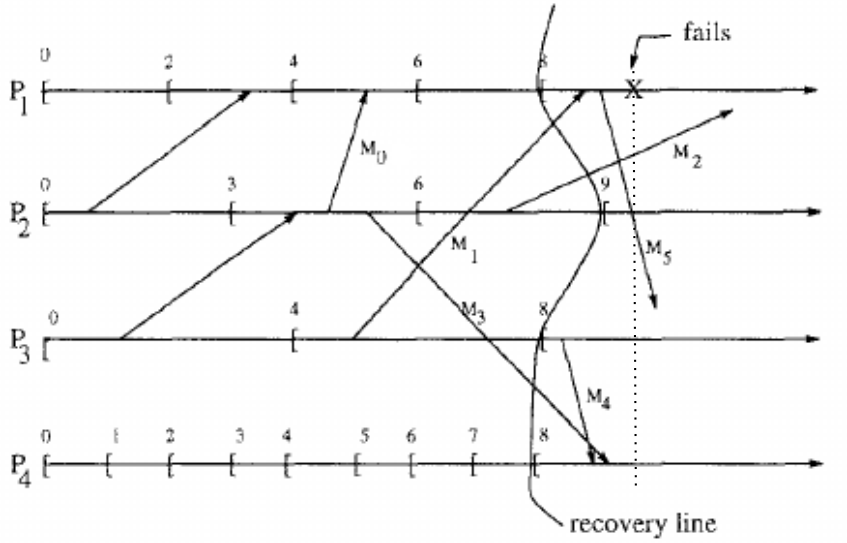
14. Montrer qu'il existe un cliché local C_{j,m_j} tel que $m_j \geq m$;
15. et montrer que pour tout cliché local C_{j,m'_j} pris avant la prise de C_{j,m_j} alors $m'_j < m$.

Correction globale

16. (sûreté) En utilisant (principalement) les propriétés 14, 15, 8, 9, 7, montrer que si un site i déclare comme état global $\{C_{1,m_1}, C_{2,m_2}, \ldots, C_{N,m_N}\}$, cet état global est effectivement cohérent.
17. (vivacité) Sous l'hypothèse que les messages arrivent en temps fini, montrer que si un site déclenche une prise d'état global, l'algorithme termine en temps fini.

2.5 Prise en compte des messages en transit

Les messages sont donnés au réseau de communication qui se charge de leur acheminement. Quand un message est délivré, il disparaît définitivement du réseau de communication. Noter que l'état du réseau de communication ne fait pas partie de l'état sauvegardé par la prise de cliché.



Supposons un calcul qui progresse, puis survient un retour en arrière (*rollback*) qui restaure un état global dans le passé. On peut supposer que la détection de la défaillance et la restauration de l'état passé sur l'ensemble des sites sont instantanées (ceci est en fait sans importance).

Sur la figure, les [indiquent un cliché local (avec son numéro), et la *recovery line* est le point de reprise global de numéro 8 qui est restauré après la défaillance de P_1. Au moment de cette restauration, un message peut être :

- traité : il a été émis et délivré avant l'état restauré (exemple : M_0). Inutile d'en parler dans la suite ;
- en retard : émis avant l'état global restauré, pas encore délivré au moment du retour en arrière ;

– en transit : émis après l'état global restauré, pas encore délivré au moment du retour en arrière (exemple : M_5) ;

– perdu : émis avant l'état global restauré, délivré entre cet état et le retour en arrière. La restauration de l'état passé produit alors une nouvelle exécution où ce message n'est plus délivré (exemple : M_1) ;

– orphelin : émis après l'état global restauré, délivré avant cet état global. La restauration de l'état passé produit une situation où ce message est délivré alors qu'il n'a pas été émis (dans l'exécution après restauration) ;

– dupliqué : message délivré deux fois *après* la restauration du point de reprise.

Questions

On considère le mécanisme de construction de l'état global présenté jusque là.

18. De quel(s) type(s) sont les messages M_2 et M_3 ?
19. Montrer qu'un retour en arrière ne peut pas produire de message orphelin.
20. Pourquoi n'est-il pas nécessaire de s'occuper des messages en retard ?
21. Montrer, sur un exemple, pourquoi un retour en arrière peut produire des messages perdus.

Pour rejouer les messages perdus après le retour en arrière, chaque site va garder trace des messages qu'il a reçus avant le retour en arrière. Ainsi, après un retour en arrière, il pourra rejouer leur délivrance.

Soit un point de reprise global identifié par un numéro k et correspondant à un cliché local $C_{i,k'}$ du site i (d'après l'algorithme, $k' \geq k$). Pour éviter les messages perdus après restauration à l'état $C_{i,k'}$, l'idée est de rejouer la délivrance des messages qui avaient été reçus après la prise de $C_{i,k'}$. Par exemple, après la restauration de $C_{1,8}$, le site 1 rejoue la délivrance de M_1.

22. Montrer que, si l'on rejoue la délivrance de tous les messages que i avait reçus après la prise de son cliché local, cela peut conduire à des messages dupliqués. Donner un exemple d'un tel message sur la figure.
23. Donner un critère, portant sur le site i et le numéro de séquence du message $M.sn$ qui permet de déterminer si la délivrance du message est inutile.
24. Montrer qu'un message en transit (comme M_5) conduit aussi à un message dupliqué (M_5 reçu deux fois après que l'on a restauré le point de reprise global 8).
25. Proposer une solution pour éviter cela.