

Systèmes et algorithmique répartis

ENSEEIH/3SN

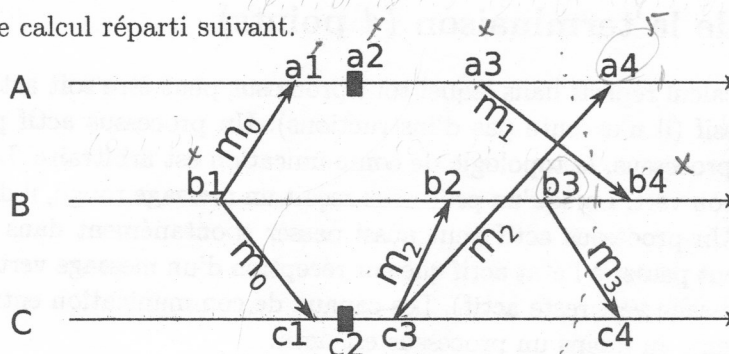
1h30, documents autorisés

7 décembre 2021

1 point par question, sauf indiqué en marge.

1 Questions de cours (6 points)

On considère le calcul réparti suivant.



1. Quels événements sont concurrents de a_3 ?
2. Quel est le passé causal de b_3 ?
3. L'exécution en temps absolu $b_1, a_1, a_2, a_3, c_1, c_2, c_3, b_2, b_3, b_4, c_4, a_4$ est-elle causalement conforme au calcul présenté ?
4. Donner l'horloge de Lamport de b_4 et a_4 .
5. Donner l'horloge vectorielle de b_4 et a_4 .
6. Donner une coupure cohérente incluant b_3 et a_4 .

2 Variation autour des horloges de Lamport (5 points)

On considère un système distribué où chaque machine s possède une horloge matérielle R_s qui avance approximativement en temps réel. Cette horloge matérielle est un entier strictement croissant, c'est-à-dire que si on considère deux lectures r_1 et r_2 de l'horloge, et que r_1 a lieu avant r_2 , alors nécessairement $r_1 < r_2$. Les horloges matérielles des sites ne sont pas synchronisées et avancent chacune à leur rythme, potentiellement irrégulier (l'écart $R_s - R_t$ entre deux machines s et t n'est pas constant). Sur chaque site s , on construit une horloge logique H_s avec des règles inspirées des horloges de Lamport :

1. Événement interne sur s : $H_s = \min(H_s + 1, R_s)$
2. Émission sur d'un message m : appliquer la règle 1 et envoyer $\langle H_s, m \rangle$
3. Réception sur s de $\langle dm, m \rangle$: $H_s \leftarrow \max(H_s, dm)$ puis appliquer la règle 1.

Pour un événement e ayant lieu sur le site s , on note $d(e)$ sa date = la valeur de H_s après l'application de la règle correspondante.

Questions

1. Indiquer la ou les différence(s) entre ces horloges et les horloges de Lamport.
2. Considérer deux événements e et e' ayant lieu sur la même machine tel que e a lieu avant e' . Montrer que $d(e') > d(e)$ en considérant attentivement les trois types d'événements.
3. Donner un chronogramme avec (au moins) un message m tel que $d(\text{émission}(m)) > d(\text{réception}(m))$.
4. Supposons que le délai temps réel de communication est borné par x pour n'importe quelle horloge matérielle, et que l'on peut synchroniser les horloges matérielles tel que l'écart entre deux horloges est borné par y ($|R_i - R_j| \leq y$). Existe-t-il une valeur de y pour garantir que $d(\text{émission}(m)) < d(\text{réception}(m))$ pour tout message?
5. On change la règle 1 pour $H_s = \max(H_s + 1, R_s)$. Montrer que, même si les horloges matérielles peuvent arbitrairement dériver, cette horloge logique a les mêmes propriétés que les horloges de Lamport : $e \prec e' \Rightarrow d(e) < d(e')$.

3 Détection de la terminaison (4 points)

On considère un calcul réparti dans lequel tout processus peut être soit actif (il exécute son programme), soit passif (il n'exécute pas d'instructions). Un processus actif peut envoyer des messages aux autres processus, la topologie de communication est arbitraire. Les messages sont de deux types : rouge ou vert. Lorsqu'un processus reçoit un message rouge, il devient passif (s'il ne l'était pas déjà). Un processus actif peut aussi passer spontanément dans l'état passif. Un processus passif ne peut passer à l'état actif que sur réception d'un message vert (et un processus actif qui reçoit un message vert reste actif). Les canaux de communication entre processus sont tous FIFO. Initialement, au moins un processus est actif.

Questions

1. Le calcul est terminé quand tous les processus sont passifs, et qu'aucun processus ne peut redevenir actif. Énoncer une propriété globale du système (état des processus et des canaux) qui caractérise la terminaison. Cette propriété est-elle stable?
2. Donner un algorithme permettant de détecter la terminaison, avec les hypothèses indiquées. (3 pt)

4 Réplication de données (5 points)

On considère un système réparti dont les sites sont fiables et reliés par un réseau maillé, fiable, asynchrone. Le nombre N de sites est fixe et leur identité connue. On souhaite réaliser un protocole de cohérence causale pour un ensemble de données partagées, dupliquées sur chacun des N sites. Chaque site s dispose donc pour chaque donnée partagée x d'une copie locale, notée x_s . Les opérations possibles sur la donnée x pour le site s sont :

- $x.\text{Read}_s()$ qui retourne la valeur de x déterminée par le protocole de cohérence ;
- $x.\text{Write}_s(v)$ qui fournit une valeur v à écrire dans x .

Questions

1. (1 point) On suppose que l'on dispose d'un service de diffusion causalement ordonnée, fournissant les opérations :

- $\text{diffusion_causale}(\text{msg})$ ¹, qui envoie le message msg à l'ensemble des N sites ;
- $\text{réception_causale}(\text{msg})$, une opération synchrone qui bloque le site appelant l'opération jusqu'à la délivrance d'un message msg par le service de diffusion causale ;
- $\text{sur_réception_causale}(\text{msg})$, un traitant asynchrone qui est appelé par le service de diffusion causale quand un message msg peut être délivré.

Implémenter le protocole de cohérence causale en définissant les messages échangés et en fournissant les algorithmes², pour une donnée partagée x et un site s :

- de l'opération $x.\text{Read}_s()$,
- de l'opération $x.\text{Write}_s(-)$,
- des traitants éventuellement associés aux délivrances de messages.

2. (2 points) On suppose maintenant que l'on dispose uniquement d'un service de communication point à point, fournissant les opérations :

- $\text{émettre}(\text{msg}, \text{dest})$, qui envoie le message msg à l'un des sites ;
- $\text{réception}(\text{msg})$, une opération synchrone qui bloque le site appelant jusqu'à la réception d'un message msg ;
- $\text{sur_réception}(\text{msg})$, un traitant asynchrone qui est appelé par le service de communication quand un message msg est réceptionné sur le site récepteur.

Implémenter le protocole de cohérence causale en définissant les messages échangés et en fournissant les algorithmes, pour une donnée partagée x et un site s :

- de l'opération $x.\text{Read}_s()$,
- de l'opération $x.\text{Write}_s(-)$,
- des traitants éventuellement associés aux réceptions de messages.

3. (2 points) Normalement, pour ce protocole, et sous les hypothèses choisies au départ, les opérations $x.\text{Read}_s()$ et $x.\text{Write}_s(-)$ devraient être non bloquantes, c'est-à-dire qu'elles devraient fournir un résultat directement, sans nécessiter de synchronisation (d'attente) par rapport aux autres sites. Expliquez pourquoi.

1. Pour cette opération, comme dans la suite, vous pouvez considérer si vous en avez besoin

- que l'émetteur et/ou le récepteur du message sont connus et peuvent être désignés à votre convenance ;
- que le message est une liste de valeurs du type de votre convenance.

2. Le niveau de détail attendu pour les algorithmes est celui adopté pour les différents algorithmes et protocoles présentés dans le cadre du cours.