Rapport de DM : Algorithme de Kumar

Kaba Sakho & Papa El Hadji G. Cisse

1 Février 2026

1 Question 1 : Implémentation et Simulation

Nous avons implémenté l'algorithme de Kumar en DistAlgo en suivant les variables d'état et les transitions de la Figure 1. Notre simulation valide que le moniteur ne déclare la terminaison que si tous les agents sont inactifs et qu'il n'y a plus de messages 'm' en transit.

Points clés de notre exécution :

— **Réactivation :** Un agent inactif qui reçoit un message 'm' repasse bien à l'état *active*. On le voit dans nos logs avec l'agent 3.

— **Canaux FIFO :** L'ordre des messages 't' est respecté grâce à `config(channel is fifo)`, ce qui est indispensable pour la preuve de monotonicité.

— **Détection et Arrêt :** Le moniteur attend d'avoir vu tous les agents et que les compteurs soient équilibrés ($mSent[p][q] = mRcvd[q][p]$) avant de déclarer la terminaison. Le programme s'arrête ensuite proprement via un signal 'stop' envoyé par le Main.

2 Question 2 : Analyse théorique

2.1 2.1 Invariants sur les compteurs

Pendant toute l'exécution, les propriétés suivantes sont conservées :

— $mSent[p][q] \le ds[p][q] \le p.aSent[q]$ et $mRcvd[p][q] \le dr[p][q] \le p.aRcvd[q]$: Comme les compteurs locaux ne font que croître et que les rapports 't' arrivent dans l'ordre (FIFO), la valeur connue par le moniteur ($mSent, mRcvd$) ou en transit (ds, dr) ne peut jamais dépasser la réalité locale de l'agent.

— **Conservation :** $p.aSent[q]$ est égal à la somme de $q.aRcvd[p]$ et du nombre de messages 'm' en transit entre p et q.

— **Monotonicité :** Si un message ('t', s, r) précède ('t', s', r') dans le canal, alors $s[p][q] \le s'[p][q]$ et $r[p][q] \le r'[p][q]$ car les variables locales ne diminuent jamais.

2.2 2.2 Preuve du Lemme (Coherent et Menace)

Soit $A \subseteq eseen$. On utilise les prédicats $Coherent(A)$ ($ds[p][q] = dr[q][p]$) et $Menace(A)$ (un agent de A est actif ou ses compteurs ont bougé).

Preuve par l'absurde : Supposons que $Coherent(A)$ et $Menace(A)$ soient vrais, mais qu'il n'y ait aucun échange avec l'extérieur, c'est-à-dire que pour tout $p \in A$ et $q \notin A$, on ait $dr[p][q] = p.aRcvd[q]$.

Si on fait le bilan sur A, le fait que $Menace(A)$ soit vrai implique une activité résiduelle. Mais si $Coherent(A)$ est vrai (équilibre interne) et qu'il n'y a pas de déséquilibre avec l'extérieur ($dr = aRcvd$), cette activité n'a aucune source possible. **Conclusion :** Il existe donc forcément $p \in A$ et $q \notin A$ tel que $dr[p][q] \neq p.aRcvd[q]$.

2.3 2.3 Conclusion, correction de l'algorithme

Supposons que le moniteur déclare la terminaison. Alors par condition de Kumar :

1. $eseen$ = ensemble de tous les agents
2. $Coherent(eseen)$ est vraie
3. $Menace(eseen)$ est fausse

Par contraposée du point (2.2), si Menace était vraie, un message transiterait vers l'extérieur, ce qui est impossible puisque *eseen* contient tout le monde. **Par conséquent :**

- aucun agent n'est actif
- aucun message n'est en transit
- tout message envoyé a été reçu

C'est la définition même de la terminaison globale. L'algorithme de Kumar est donc correct.