

Project report

Sujet : Projet ARA 2024–2025
étude expérimentale 1



Member :

Runlin ZHOU

Weida LIU

4 Nov 2024 - 7 Dec 2024

Contents

1	Introduction	1
2	la valeur de γ est négligeable par rapport à celle de α	3
3	la valeur de γ est comparable à celle de α	6
4	la valeur de γ est nettement supérieure à celle de α	9
5	Comparaison complète	12
6	Conclusion	16

1 Introduction

Dans ce projet, nous collectons des données sur les performances de l'algorithme de Naimi-Tréhel sous différentes charges en ajoutant de nouvelles structures de données et sections statistiques au code de l'algorithme de Naimi-Tréhel et nous les analysons sur la base de ces données afin d'évaluer les performances de l'algorithme dans différentes situations.

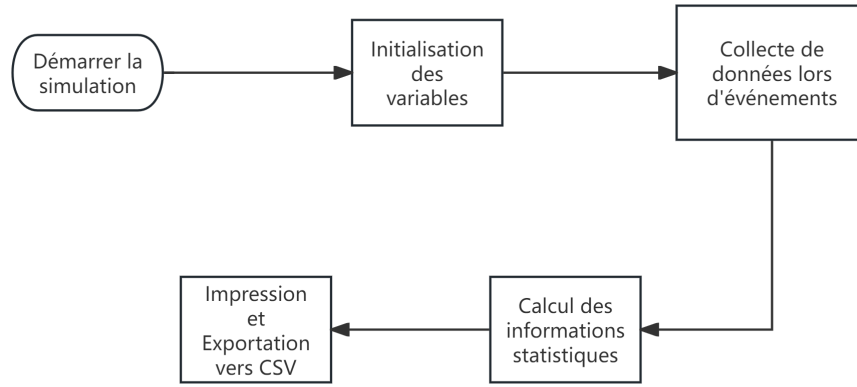
Nous utiliserons les paramètres suivants pour modifier ρ (charge) :

- le nombre de nœuds, noté N ,
Nous fixons $N = 10$, c'est une bonne représentation de la performance sans être trop lourd pour l'appareil
- le temps moyen passé en section critique, noté α ,
contrôlé par `protocol.mutex.timeCS` dans le fichier de configuration
- le temps moyen qu'un nœud attend avant de redemander l'accès à la section critique, noté β ,
contrôlé par `protocol.mutex.timeBetweenCS` dans le fichier de configuration
- le temps moyen pour transmettre un message entre deux nœuds, noté γ .
contrôlé par la moyenne des paramètres `protocol.transport.mindelay` et `protocol.transport.maxdelay`

Nous utiliserons les paramètres suivants pour comparer les performances :

- le nombre de messages applicatifs (token et request) par section critique,
- le nombre de messages `request` par nœud,
- le temps moyen nécessaire pour obtenir la section critique, c'est-à-dire le temps moyen qu'un nœud passe dans l'état `requesting`,
- le pourcentage de temps que le token passe dans chacun de ses états (U, T et N).

Afin de collecter les données, nous avons ajouté une série de codes de collecte de données au `NaimiTrehelAlgo` original, et le flux général de la collecte est le suivant :



Afin d'éviter que des valeurs aberrantes n'affectent les résultats, pour chaque ensemble de paramètres, on va effectuer la simulation 10 fois et prendrons la valeur moyenne comme résultat à analyser.

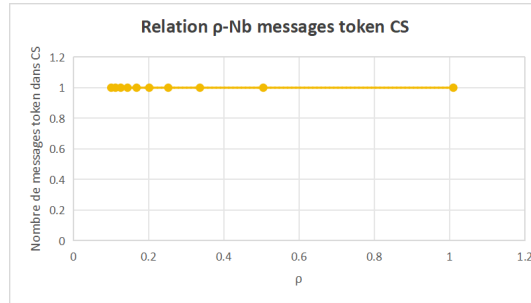
Ensuite, nous examinerons les performances de l'algorithme dans les trois cas suivants:

- $\gamma \ll \alpha$: cas où la valeur de γ est négligeable par rapport à celle de α ,
- $\gamma = \alpha$: cas où la valeur de γ est comparable à celle de α ,
- $\gamma \gg \alpha$: cas où la valeur de γ est nettement supérieure à celle de α .

2 la valeur de γ est négligeable par rapport à celle de α

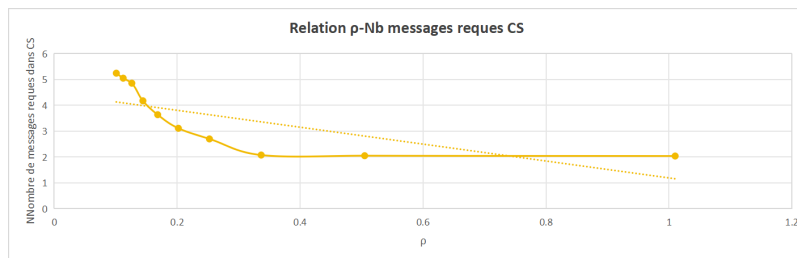
Pour réaliser $\gamma \gg \alpha$, nous avons choisi $\alpha = 100$, $\gamma = 1$ et décrémente β de 1 000 à 100 afin de modifier ρ (charge).

2.1 Nombre de messages de tokens dans les CS

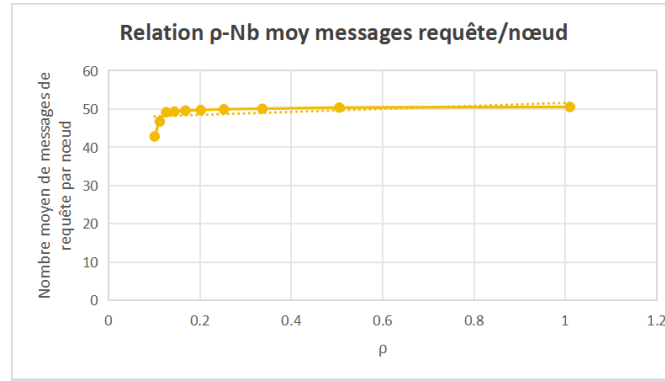


Le nombre de messages de tokens dans CS est resté stable à 1. Cela signifie que, quand le délai de transmission est négligeable, quelle que soit la charge, un seul message token est nécessaire pour chaque accès à CS, et le système maintient une efficacité de transmission élevée quelle que soit la charge.

2.2 Nombre de messages de demande dans le CS



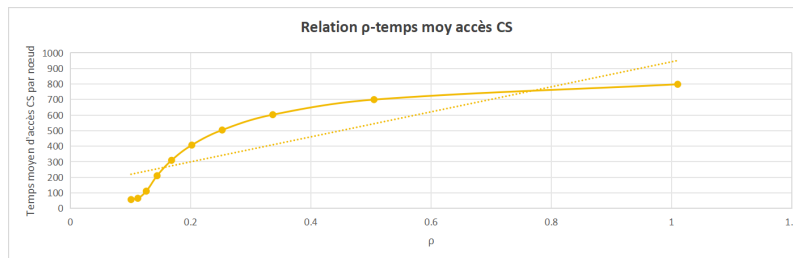
Avec l'augmentation de la charge, le nombre de messages de requête dans chaque CS diminue. Ensuite, la baisse ralentit. Quand la charge est élevée, ce nombre se stabilise. Globalement, la courbe reste stable et montre une légère tendance à la baisse. Quand le délai de transmission est négligeable, les nœuds obtiennent plus facilement des tokens après une demande. Cela montre que, dans un réseau efficace, le nombre de messages de demande dans CS ne dépend pas de la charge.



2.3 Nombre moyen de messages de demande par nœud

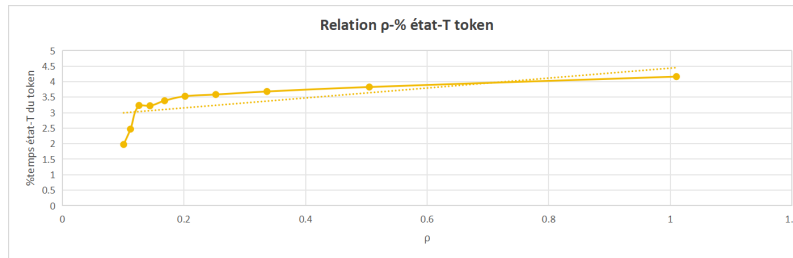
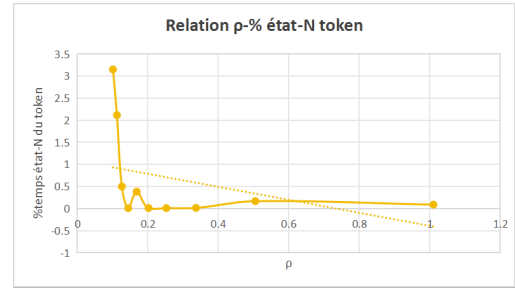
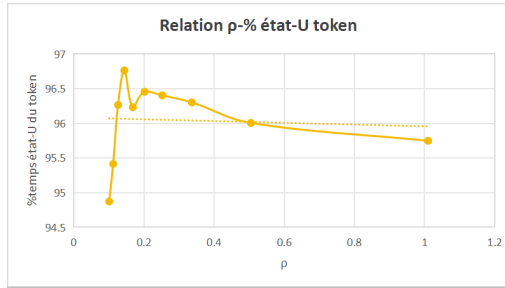
Avec l'augmentation de la charge, le nombre moyen de messages de requête par nœud augmente légèrement et la courbe reste globalement stable et montre une tendance à la hausse. Quand le délai de transmission est négligeable, les requêtes des nœuds sont satisfaites rapidement. Même si la charge augmente, il n'est pas nécessaire d'émettre trop de messages de requête.

2.4 Temps d'attente moyen pour qu'un nœud entre en CS



A mesure que la charge augmente, le temps d'attente moyen pour que chaque nœud entre dans CS augmente rapidement, puis ralentit lorsque les charges sont élevées. Mais globalement il reste à un niveau inférieur par rapport aux cas de $\gamma \gg \alpha$ et $\gamma = \alpha$. Cela signifie que, quand le délai de transmission est négligeable, même si la charge augmente, il est relativement facile pour le nœud d'obtenir le token.

2.5 Pourcentage de temps d'état symbolique (U, T, N)

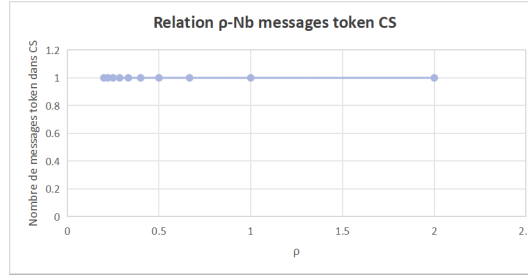


Le ratio d'états U domine toujours, tandis que les ratios d'états T et N sont très faibles. Lorsque la charge est légère (c'est-à-dire lorsque la charge est faible), le token passe la plupart de son temps dans l'état N inutilisé. À mesure que la charge augmente, la proportion de l'état N diminue considérablement, tandis que la proportion de l'état T. augmente. Cela signifie que, quand le délai de transmission est négligeable, le token n'a pas besoin d'attendre trop longtemps sur le réseau et il n'y a presque pas de temps d'inactivité. Le token est utilisé la plupart du temps.

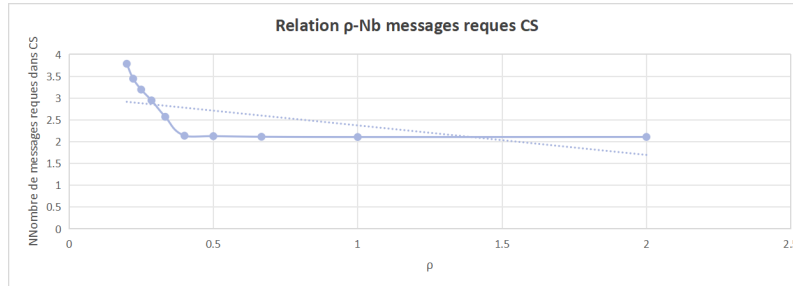
3 la valeur de γ est comparable à celle de α

Pour réaliser $\gamma = \alpha$, nous avons choisi $\alpha = 100$, $\gamma = 100$ et décrétementé β de 1 000 à 100 afin de modifier ρ (charge).

3.1 Nombre de messages de tokens dans les CS

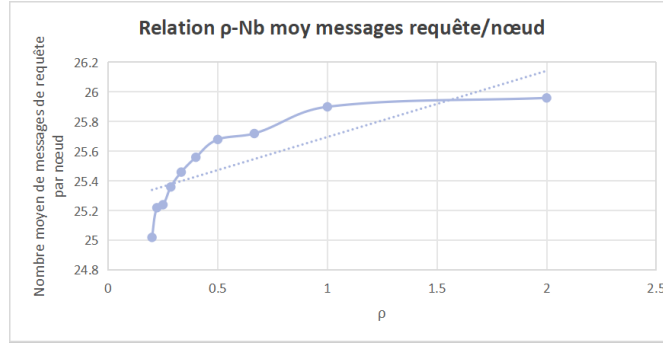


Le nombre de messages de tokens dans CS est resté stable à 1. Cela signifie que, quand le délai de transmission du message est du même ordre que le temps CS, quelle que soit la charge, un seul message token est nécessaire pour chaque accès à CS, et le système maintient une efficacité de transmission élevée quelle que soit la charge.



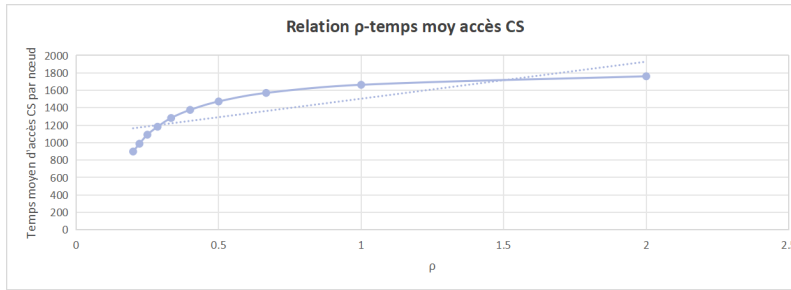
Quand la charge est faible, le nombre de messages de requête dans CS est plus grand, mais à mesure que la charge augmente, le nombre de messages de requête dans chaque CS diminue d'abord légèrement, puis diminue lorsque la charge est élevée. La tendance s'est ralentie et s'est progressivement stabilisée. Cela indique que lorsque le délai de transmission du message est du même ordre que le temps CS, le token peut être inactif lorsque la charge est faible et des messages de requête supplémentaires doivent être envoyés pour l'évoquer. Lorsque la charge augmente, les tokens circulent plus fréquemment et il est plus facile de trouver les tokens par nœuds, donc le nombre de requêtes diminue légèrement.

3.2 Nombre moyen de messages de demande par nœud



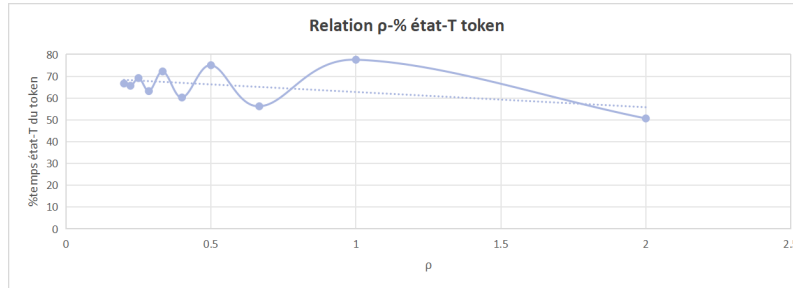
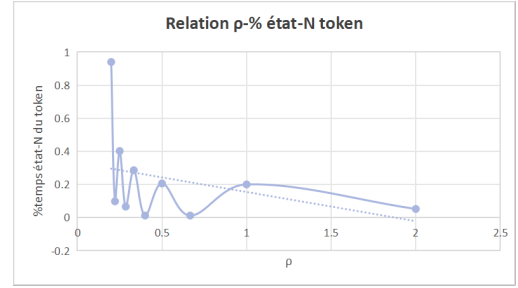
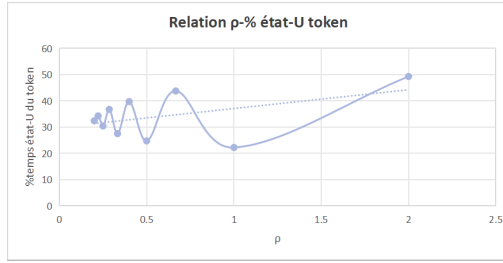
A mesure que la charge augmente, le nombre moyen de messages de requête par nœud augmente légèrement, puis la croissance ralentit à des charges plus élevées. La courbe reste globalement stable et tend à augmenter. Cela montre que lorsque le délai de transmission du message est du même ordre que le temps CS, la charge augmente, ce qui amène le nœud à émettre des requêtes plus fréquemment, et comme la vitesse de transmission n'est pas particulièrement lente, le système peut répondre à la requête pour dans une certaine mesure et ne le fera pas. Il y a une augmentation infinie du nombre moyen de messages de requête par nœud.

3.3 Temps d'attente moyen pour qu'un nœud entre en CS



A mesure que la charge augmente, le temps d'attente moyen pour que chaque nœud entre dans CS augmente rapidement et se situe généralement à un niveau plus élevé que le cas où $\gamma \ll \alpha$. Cela montre que lorsque le délai de transmission du message est de la même ampleur que le temps CS, la charge augmente et l'effet de file d'attente est évident. En même temps, puisque le temps de transmission ne peut être ignoré, le nœud a besoin de plus de temps pour attendre le message. message à arriver.

3.4 Pourcentage de temps d'état symbolique (U, T, N)

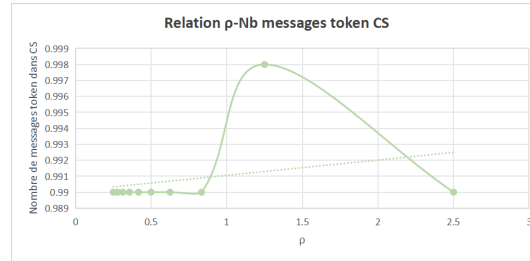


À partir des données et des images, nous pouvons voir qu'à mesure que la charge augmente, la proportion de l'état U montre une tendance globale à la hausse, la proportion de l'état T montre une tendance à la baisse, la proportion de l'état N montre une tendance à la baisse mais la différence n'est pas grande, et la proportion d'état U Lorsque $\gamma \ll \alpha$ est significativement plus petite, la proportion d'état T est significativement plus grande que lorsque $\gamma \ll \alpha$. Cela montre que lorsque le délai de transmission du message est de la même ampleur que le temps CS, le token doit passer plus de temps à attendre sur le réseau, mais n'est pas souvent inactif, de sorte que le token est dans l'état de transmission près de la moitié du temps.

4 la valeur de γ est nettement supérieure à celle de α

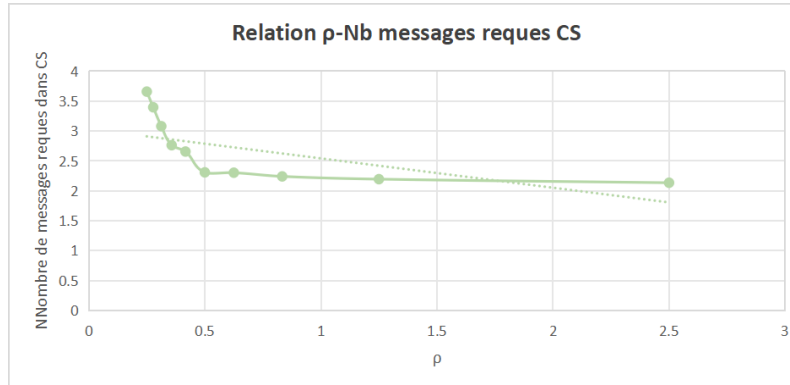
Pour réaliser $\gamma \gg \alpha$, nous avons choisi $\alpha = 50$, $\gamma = 200$ et décrément β de 1 000 à 100 afin de modifier ρ (charge).

4.1 Nombre de messages de tokens dans les CS



Le nombre de messages de tokens dans CS est toujours stable autour de 1 et ne change pas radicalement avec , mais il fluctue légèrement. Quand le délai de transmission du message est bien supérieur au temps d'exécution du CS, un seul message token est encore nécessaire à chaque fois pour entrer dans le CS.

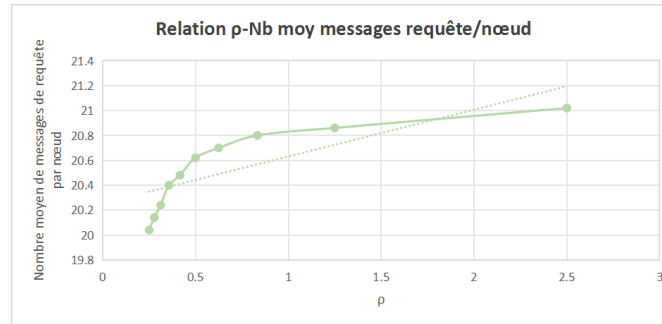
4.2 Nombre de messages de demande dans le CS



À partir des données et des images, nous pouvons voir que lorsque la charge est faible, le nombre de messages de requête dans CS est plus important. Cependant, à mesure que la charge augmente, le nombre de messages de requête dans chaque CS diminue d'abord, puis la tendance à la baisse ralentit. vers le bas et se rapproche progressivement de la stabilité. Cela montre que lorsque le délai de transmission du message est bien supérieur

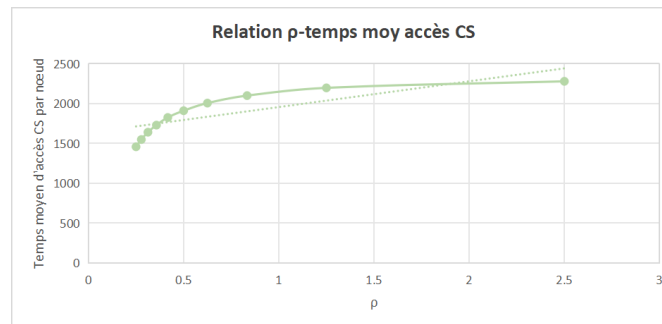
au temps d'exécution du CS, il est difficile pour le token d'apparaître à temps sur le nœud demandé. Sous une faible charge, les tokens inactifs nécessitent davantage de messages de requête pour le réveil et le positionnement. Lorsque la charge augmente, les tokens circulent plus fréquemment et il est plus facile pour les nœuds de trouver des tokens, alors le nombre de requêtes diminue légèrement.

4.3 Nombre moyen de messages de demande par nœud



Le nombre moyen de messages de requête par nœud augmente légèrement. Ensuite, il se stabilise. La courbe reste stable dans l'ensemble et montre une tendance à la hausse. Cela montre que, si le délai de transmission du message est bien plus long que le temps d'exécution du CS, chaque nœud doit émettre plus de requêtes pour obtenir le token. Avec l'augmentation de la charge, l'efficacité globale diminue.

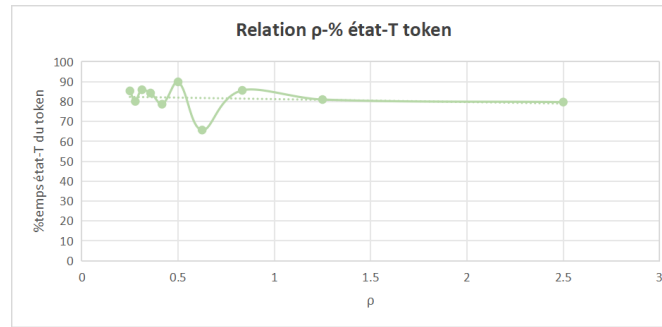
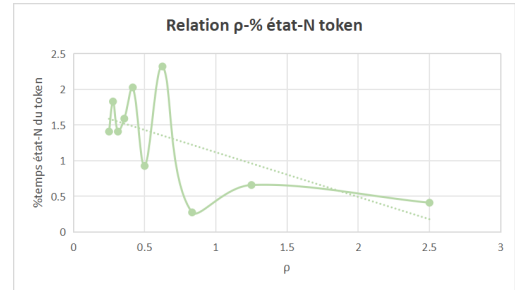
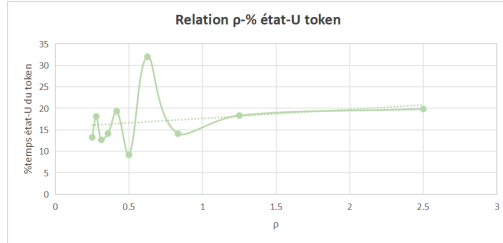
4.4 Temps d'attente moyen pour qu'un nœud entre en CS



À partir des données et des images, nous voyons que, quand la charge augmente, le temps d'attente moyen pour entrer dans CS augmente rapidement. Ce temps reste généralement plus élevé que dans les cas où $\gamma \ll \alpha$ et $\gamma = \alpha$. Cela montre que, si le délai de transmission

est beaucoup plus long que le temps d'exécution du CS, l'effet de file d'attente devient plus marqué avec l'augmentation de la charge. En même temps, à cause du délai de transmission, chaque nœud doit attendre plus longtemps pour recevoir le message.

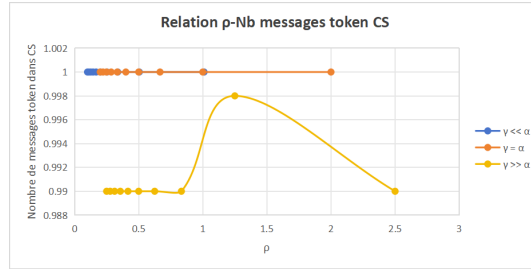
4.5 Pourcentage de temps d'état symbolique (U, T, N)



À partir des données et des images, nous pouvons voir qu'à mesure que la charge augmente, la proportion de l'état U montre une légère tendance à la hausse, la proportion de l'état T montre une légère tendance à la baisse, la proportion de l'état N montre une tendance à la baisse et la proportion de l'état U Elle est nettement plus petite que dans le cas de $\gamma \ll \alpha$ et $\gamma = \alpha$, et la proportion d'état T est nettement plus grande que dans le cas de $\gamma \ll \alpha$ et $\gamma = \alpha$. Cela montre que lorsque le délai de transmission du message est beaucoup plus long que le temps d'exécution du CS, le token est presque toujours en route, quelle que soit la façon dont la charge change, le goulot d'étranglement de la transmission rend le token rarement utilisé et inactif.

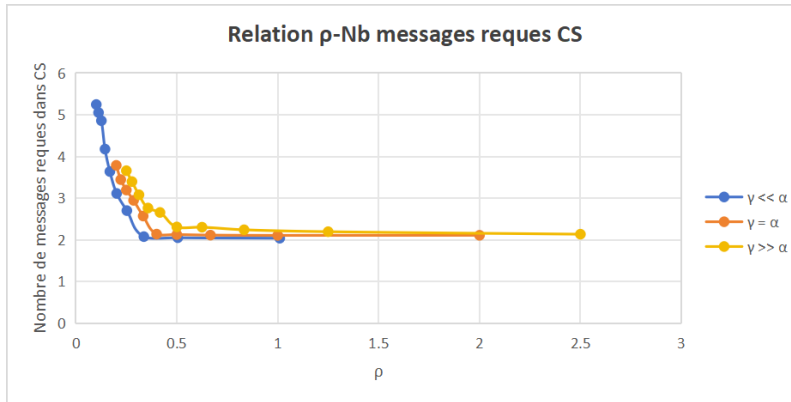
5 Comparaison complète

5.1 Nombre de messages à tokens dans les CS



Quelle que soit la façon dont les conditions de transmission et de charge changent, le nombre de messages de jeton dans CS est stable autour de 1, ce qui indique que chaque fois qu'un nœud entre avec succès dans CS, un seul message de jeton doit être livré en moyenne. Ceci est conforme au principe de l'algorithme Naimi-Trehel et vérification de l'exclusivité mutuelle des algorithmes.

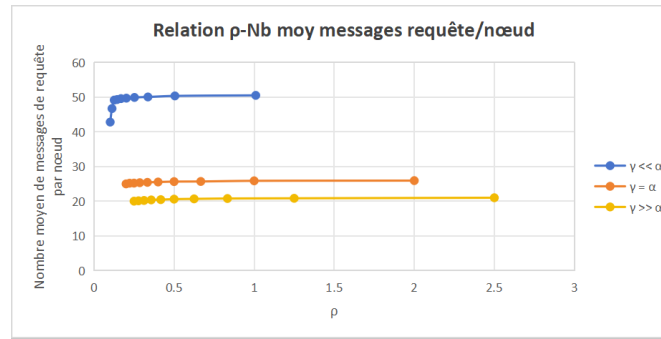
5.2 Nombre de messages de demande dans le CS



Dans les trois situations de transmission, la tendance du nombre de messages de requête dans CS avec charge est fondamentalement la même, et ils diminuent tous avec l'augmentation de la charge. La tendance à la baisse ralentit à des charges élevées et devient progressivement stable. est fondamentalement le même. Cela montre que quelle que soit la situation de transmission, à faible charge, le jeton est inactif. Lorsque le nœud a besoin du jeton, il doit envoyer plus de requêtes pour trouver le jeton, ce qui entraîne un nombre plus élevé de messages de requête dans CS. cette fois. À mesure que la charge augmente, les jetons

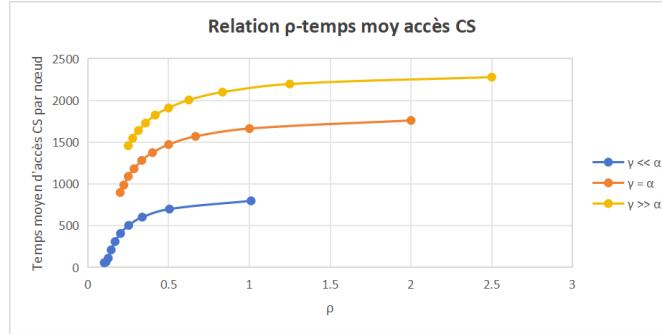
circulent entre les nœuds. À ce stade, les nœuds n'ont pas besoin d'envoyer plus de requêtes pour trouver des jetons. Par conséquent, le nombre de messages de requête dans CS diminue progressivement à ce stade. Lorsque atteint une certaine valeur, le système entre dans un équilibre dynamique et le jeton est presque toujours nécessaire au demandeur. Par conséquent, le nombre de messages de requête ne change plus de manière significative, mais reste à un niveau stable.

5.3 Nombre moyen de messages de demande par nœud



Dans les trois cas de transmission, la tendance du nombre de messages de requête dans CS avec charge est fondamentalement la même. Elle augmente d'abord puis se stabilise sous une charge élevée, et la plage de valeurs est relativement stable. Cependant, on peut constater que le nombre moyen de messages de requête par nœud dans le cas de « est beaucoup plus grand que dans les deux autres cas. Cela montre que lorsque le délai de transmission du message est négligeable par rapport au temps d'exécution du CS, la transmission du message est terminée presque instantanément, ce qui signifie que le nœud peut immédiatement envoyer la requête suivante après avoir terminé un CS. À cet égard, nous en tirons des leçons. trois Le nombre de fois où le cas a été saisi dans CS peut également être vu. Dans le même temps, cela montre également qu'à mesure que la charge augmente, les nœuds demanderont à entrer dans le CS plus fréquemment et, par conséquent, davantage de messages de requête seront envoyés. Cependant, en raison des limitations des capacités de traitement du protocole et des ressources système, cette croissance finit par se stabiliser.

5.4 Temps d'attente moyen pour qu'un nœud entre en CS

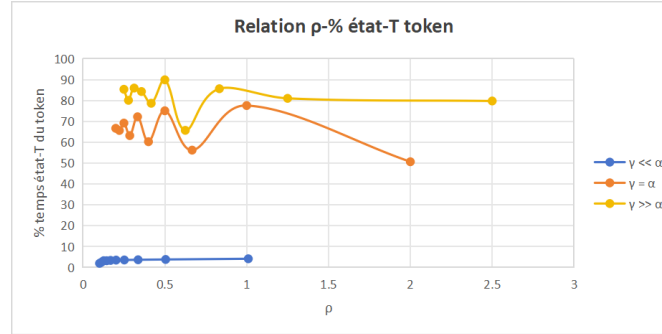
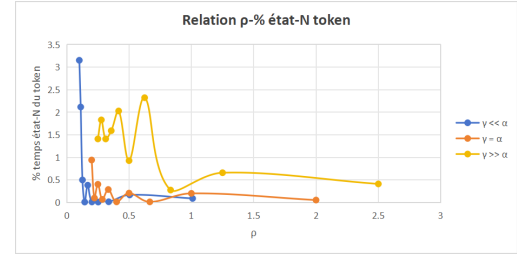
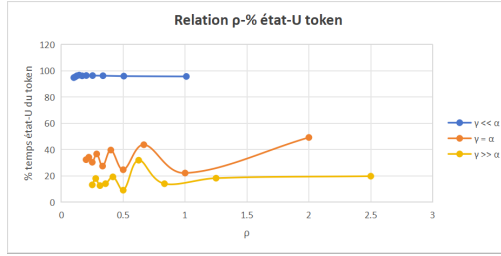


Dans les trois cas de transmission, le temps d'attente moyen des nœuds entrant dans CS a fondamentalement la même tendance avec les changements de charge. Ils augmentent tous rapidement au début, puis deviennent stables, et la plage de valeurs change considérablement. Dans le même temps, nous pouvons constater que dans le cas de \gg , le temps d'attente moyen pour que les nœuds entrent dans CS est beaucoup plus long que dans le cas de $=$, et beaucoup plus long que dans le cas de \ll .

Cela montre que plus le nœud entre dans la situation de transmission, plus le temps d'attente moyen du nœud entrant dans CS est long, car plus de temps est passé à attendre la transmission du jeton.

Dans le même temps, cela montre également qu'à mesure que la charge augmente au début, le temps d'attente moyen pour que les nœuds entrent dans CS augmentera progressivement, car à ce moment-là, les demandes sont plus intensives, il y a plus de concurrents dans la file d'attente, et cela Il est moins probable que le jeton atteigne immédiatement le nœud demandeur, ce qui entraîne une augmentation du temps d'attente. Cependant, à mesure que la charge continue d'augmenter et que la capacité de traitement du système devient progressivement saturée, la livraison des requêtes et des jetons s'équilibre progressivement. À ce stade, le temps d'attente moyen pour que les nœuds entrent dans CS se stabilise progressivement.

5.5 Pourcentage de temps d'état symbolique (U, T, N)



Lorsque «, le jeton est presque toujours utilisé. Lorsque », le jeton est presque toujours en transit. La situation de = se situe juste au milieu entre les deux.

Cela montre que lorsque le délai de transmission du message est négligeable par rapport au temps d'exécution de CS, le jeton est utilisé la plupart du temps.

Lorsque le délai de transmission du message est du même ordre que le temps CS, la répartition des états est plus équilibrée et plus affectée par la charge de requête. Par rapport au cas où «, plus de temps est consacré à la transmission, ce qui entraîne une réduction de l'efficacité du système.

Lorsque le délai de transmission du message est bien supérieur au temps d'exécution du CS, le jeton est presque toujours en cours de transmission. À ce moment-là, le temps de transmission devient un goulot d'étranglement, ce qui entraîne une efficacité extrêmement faible du système.

6 Conclusion

A partir de l'analyse ci-dessus, l'algorithme naimi-trehel:

En cas où la valeur de τ est négligeable par rapport à celle de α ($\tau \ll \alpha$), le token est utilisé la plupart du temps. Illustration où le délai de transmission du message est négligeable par rapport au temps d'exécution du CS, l'algorithme peut traiter efficacement les requêtes tout en assurant l'exclusivité mutuelle.

En cas où la valeur de τ est comparable à celle de α ($\tau \approx \alpha$), les performances se dégradent sensiblement, mais les performances restent dans des limites raisonnables.. Il y a un équilibre entre l'efficacité des demandes et l'efficacité de la transmission. Montre que l'algorithme reste bien adaptable dans des environnements où les délais de transmission des messages sont du même ordre que le temps CS.

En cas où la valeur de τ est nettement supérieure à celle de α ($\tau \gg \alpha$) il est proche de ne pas fonctionner correctement, les performances sont considérablement réduites, le jeton est en transit la plupart du temps. Les algorithmes ne conviennent pas aux environnements où les délais de transmission des messages sont beaucoup plus longs que le temps d'exécution du CS.