

Chloé DESDOUITS
Guillaume DUVILLIE
Swan ROCHER

M1 Informatique MOCA

TP d'algorithmes distribués

Table des matières

1	Introduction	2
2	Algorithme d'exclusion mutuelle de Naimi-Trehel	3
3	Extension tolérante aux pannes	4
4	Implémentation	5
5	Tests et résultats	6

Chapitre 1

Introduction

Afin de résoudre des problèmes complexes sur des données de grandes tailles, il existe plusieurs possibilités : utiliser un algorithme avec une bonne complexité, augmenter la puissance de l'ordinateur utilisé ou utiliser plusieurs ordinateurs interconnectés. Lorsque les deux premières possibilités ont été poussées à leur maximum, il est nécessaire d'utiliser des algorithmes dits distribués afin de répartir les calculs sur plusieurs machines.

Les algorithmes distribués sont divisés en plusieurs catégories. Il existe tout d'abord des algorithmes d'élection qui permettent de donner la priorité temporaire à un site par rapport aux autres. Il y a également des algorithmes d'exclusion mutuelle qui permettent à tous les sites de travailler sur la même ressource (section critique) et qui garantissent le fait que deux sites ne peuvent pas accéder en même temps à cette ressource. Enfin les algorithmes de terminaison permettent de détecter la fin du calcul distribué pour tous les sites et ainsi de terminer l'application.

Nous allons nous intéresser aux algorithmes distribués d'exclusion mutuelle. Ceux-ci sont jugés sur les critères suivants :

- la vivacité : le fait qu'un site qui demande à entrer en section critique puisse y entrer au bout d'un temps fini,
- la sûreté : la capacité de l'algorithme à faire en sorte qu'un seul site à la fois soit en section critique,
- le respect de l'ordre des demandes,
- la tolérance aux pannes,
- la complexité en nombre de messages échangés.

Nous allons tout d'abord étudier l'algorithme de Naïmi-Trehel [1] qui fait partie de la catégorie des algorithmes distribués d'exclusion mutuelle. Nous allons ensuite nous intéresser à une extension tolérante aux pannes de cet algorithme [2]. Nous expliquerons alors les choix effectués lors de l'implémentation de ces deux algorithmes. Enfin, nous exposerons les résultats de nos phases de test.

Chapitre 2

Algorithme d'exclusion mutuelle de Naimi-Trehel

Chapitre 3

Extension tolérante aux pannes

Chapitre 4

Implémentation

Chapitre 5

Tests et résultats

Bibliographie

- [1] M. Naimi, M. Trehel, and A. Arnold. A log (n) distributed mutual exclusion algorithm based on path reversal. *Journal of Parallel and Distributed Computing*, 34(1) :1–13, 1996.
- [2] J. Sopena, L. Arantes, M. Bertier, and P. Sens. A fault-tolerant token-based mutual exclusion algorithm using a dynamic tree. *Euro-Par 2005 Parallel Processing*, pages 644–644, 2005.