

Systemes distribués asynchrones tolérants aux pannes



“Consensus: the Big Misunderstanding” 1997



Rachid Guerraoui & André Schiper
Ecole Polytechnique Fédérale de Lausanne

Mardi 15 février 2004,
Christophe GRAVIER

Plan de la présentation

- 1- Introduction: le problème du consensus
- 2- Le consensus dans les protocoles d'accord
- 3- Les clarifications proposées par les auteurs
- 4- Conclusion

Problème du consensus

cadre socio-philosophique

1- Introduction: le problème du consensus

- Consensus: qui s'accorde, qui vit en harmonie
- Conflit: ce qui se heurte, s'oppose, se contrarie, se contredit
- Ex: l'état républicain maintient le conflit encadré par la loi et le consensus dans la recherche du bien commun.
- Le consensus est un phénomène
 - D'accord sur une décision par un groupe
 - Dans le cadre d'une action pour le bien commun

Problème du consensus

définition écolo-économique

1- Introduction: le problème du consensus

- un processus qui se déroule en plusieurs phases
 - utilisant diverses techniques
 - de discussion
 - d'analyse
 - de débat
 - grâce auquel un groupe arrive à
 - prendre des décisions
 - sans avoir recours au vote.

R. Tecchio, « la méthode consensus – présentation. Entre éthique, pragmatisme et esthétisme », In “passerelle éco n° 9”, 3 mars 2003

A partir d'une action non violente contre le G8 sur un sujet écologique. (proposition alternative)

Problème du consensus

ex. en systèmes distribués asynchrones (SDA)

2- Le consensus dans les protocoles d'accord

- Traduction de la pensée du consensus en informatique
- Soit $P = \{p_1, p_2, \dots, p_n\}$ et $V = \{V_1, V_2, \dots, V_3\}$ les valeurs initiales respectives associées à V .
- Etablir un consensus serait que P choisisse $v / v \in V$
- Intervient:
 - la notion de groupe
 - de décision commune sans vote (« non-violence »)
- Si l'ensemble V a:
 - 1 valeur: ce n'est pas un problème de consensus (dictature)
 - 2 valeurs: protocole d'accord binaire
 - 2+ valeurs: protocole d'accord multi-valué.
- Et si un des processus s'effondre ?

Résultat d'impossibilité de Fischer, Lynch et Paterson

2- Le consensus dans les protocoles d'accord

- Dans un système distribué asynchrone, si un processus est faux, il est impossible d'obtenir une décision basée sur la méthode du consensus
- Énoncé incontournable et non remis en cause ici
- Énoncé ne touchant pas les systèmes:
 - Synchrones
 - Partiellement synchrones
 - Distribués avec détecteurs de pannes (« oracles »)
- M. Fischer, N. Lynch and M. Paterson, « Impossibility of Distributed Consensus with One Faulty Process », “Journal of ACM”, 32:374-382, Avril 1985

Les clarifications de l'article

« M1: Consensus is for theoreticians only »

3- Les clarifications proposées par les auteurs

- Théoriciens: ne pas essayer systématiquement une résolution par consensus si l'on peut s'en passer
- Praticiens: systèmes « réels »
 - ont à résoudre des problèmes d'accord pratique
 - et ne se préoccupent pas du résultat d'impossibilité FLP
- Auteurs
 - Nécessité de pousser à l'implémentation
 - Si un système résout un problème de consensus, cette solution serait-elle factorisable ou au moins transposable ?

Les clarifications de l'article

« M2: Time-outs are enough »

3- Les clarifications proposées par les auteurs

- Logique: RI FLP s'applique aux systèmes asynchrones
 - utilisons des time-outs pour cadrer les réactions.
- Auteurs:
 - Ex: dans SDA, le time-out peut être le nombre inst/sec
 - Or le RI FLP s'applique dans systèmes distribués asynchrones.
 - Donc les time-outs ne sont pas suffisants !
- Le problème vient du fait que si les time-outs sont:
 - Trop hauts: ce n'est plus asynchrone (pas de menace RI FLP mais système non asynchrone)
 - Trop faibles: trop de « faux négatifs » (le système redevient asynchrone donc pas de consensus possible)

Les clarifications de l'article

« M3: There's no life after FLP »

3- Les clarifications proposées par les auteurs

- RI FLP \equiv impossible de trouver un algorithme permettant de résoudre un problème de consensus dans toutes les exécutions possibles du système.
- Auteurs:
 - Possible de trouver algo. \mathcal{A} satisfaisant \mathcal{A} -runs $\in \mathcal{A}$ -total-system-runs
 - Possible de trouver algo. \mathcal{A}' satisfaisant \mathcal{A}' -runs $\in \mathcal{A}'$ -total-system-runs
 - Possible de dire que $\mathcal{A} > \mathcal{A}'$ ssi \mathcal{A}' -runs $\subset \mathcal{A}$ -runs
- La proposition est de caractériser les « runs » sous lesquels un algorithme résout le problème du consensus.

Les clarifications de l'article

« M4: The failure detector model is unrealistic »

3- Les clarifications proposées par les auteurs

- Non discuté dans cet exposé
- Proposition
 - intéressante
 - mais nécessitant de nombreuses explications en aval dépassant le cadre (horaire) octroyé pour cet exposé.

Les clarifications de l'article

« M5: Time-free means inefficient »

3- Les clarifications proposées par les auteurs

- Il est généralement dit: « Solving consensus in a time-free model leads to a complex and inefficient solution »
- Auteurs:
 - La solution est généralement intrinsèquement complexe
 - La solution est généralement efficace !
 - Ne pas confondre complexité et efficacité
- Mesure de l'efficacité
 - Nombre de messages envoyés par l'algorithme
 - Nombre d'étapes nécessaire à l'établissement du consensus
 - 3 étapes: algorithme Chandra-Toueg (envoyer, renvoyer, décider)
 - 2 étapes: algorithme de décision précoce (envoyer + propo, renvoyer)
 - 1 étape: time-based (envoi et décision si aucune protestation)

Les clarifications de l'article

3- Les clarifications proposées par les auteurs

« M6: Asynchronous algorithms cannot be used for time critical applications »

- Auteurs:
 - Ces algorithmes:
 - ne peuvent être utilisés comme applications temps-critique
 - mais peuvent être immergés dans ces applications
 - Doivent être élaborés en 2 phases:
 - Phase de design
 - Phase d'implémentation

Doutes sur l'argumentation

Apport de ce « misunderstanding » ?

Conclusion

- Problème du consensus pour plusieurs domaines
 - info, socio, économie, politique, philo, écologie, ...
- Accent mis sur aide des praticiens informatiques dans la réflexion du consensus (factoriser les solutions)
 - pourquoi pas une réflexion trans-disciplinaire ?
- Clarifications sur le consensus dans les SDA
 - Réflexions argumentées et pertinentes
 - Dernier « misunderstanding » pas très décisif (apport ?)
 - Pas de proposition de contournement sauf M3: « there's no life after FLP » qui est plutôt une « demi vie »

Questions !

