Le consensus, au cœur du calcul distribué

Avancées récentes et questions ouvertes sur le consensus probabiliste

Matthieu Perrin
LS2N, Université de Nantes
Équipe GDD
matthieu.perrin@univ-nantes.fr

Accueil des étudiants de l'ENS Paris-Saclay
30/11/2018

GDD : Gestion des Données Distribuées

Web des données

- Web sémantique social
- Données liées
- Intégration des données

Protection des données

- Respect de la vie privée
- Confidentialité

LS2N

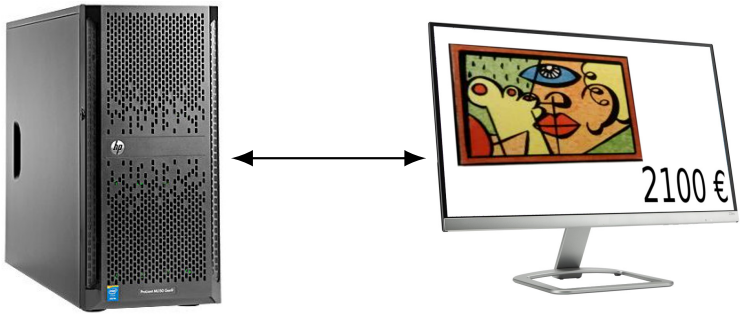
Création	1er janvier 2017
Siège	Nantes
Pays	France
Rattachement	CNRS École centrale de Nantes Université de Nantes IMT Atlantique Inria
Directeur	Claude Jard
Disciplines	Numérique, informatique, robotique, automatique, cognition, productique, traitement d'images
Site web	http://ls2n.fr/ [archive]

modifier

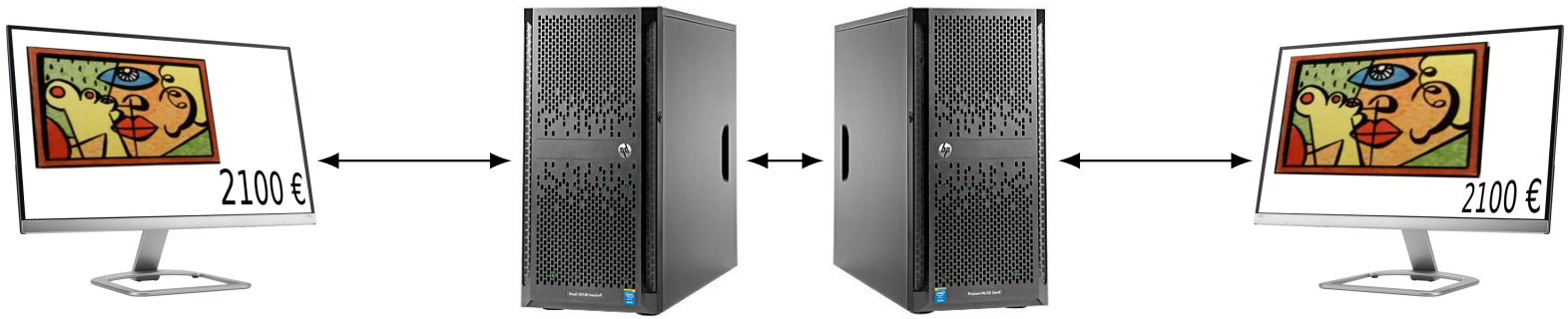
Fondement du calcul distribué

- Algorithmique du réparti
- Calculabilité et complexité
- Passage à l'échelle et tolérance aux pannes

Gestion d'un site de vente d'antiquités



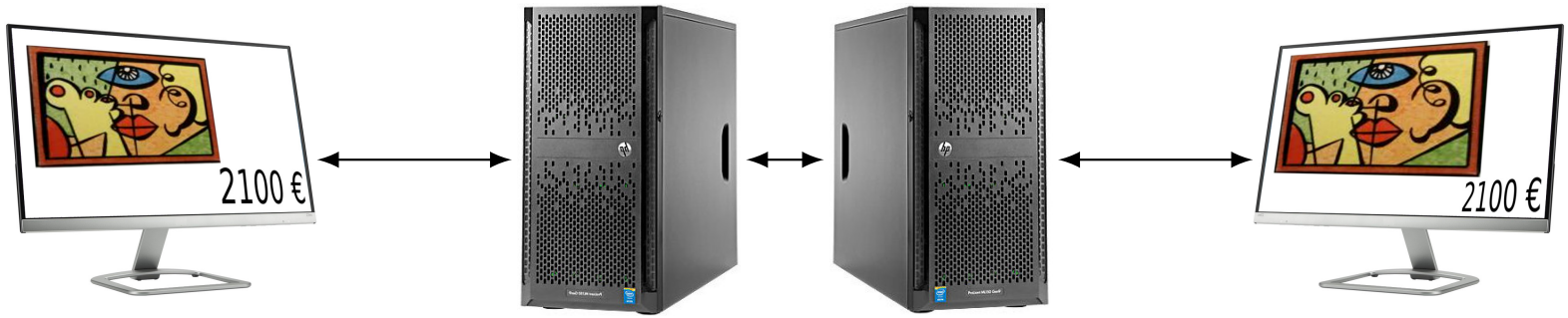
Gestion d'un site de vente d'antiquités



Avantages de la réplication

- Parallélise la charge
- Réduit la latence
- Résiste aux pannes

Gestion d'un site de vente d'antiquités



Avantages de la réplication

- Parallélise la charge
- Réduit la latence
- Résiste aux pannes

Difficultés de la réplication

- Cohérence des données ?
- Synchronisation nécessaire

Consensus !

Universalité du consensus

Reconstruire l'illusion qu'il n'y a qu'un seul serveur.

Autre exemple : Accord dans la blockchain

Problème scientifique

Modèle de calcul

- n processus
- Communication par messages
- Canaux asynchrones
- Au plus $t < \frac{n}{2}$ pannes

Spécification du consensus

Chaque processus propose une valeur (booléenne) et cherche à décider une valeur.

Accord : Au plus une valeur est décidée.

Validité : Toute valeur décidée a été proposée.

Terminaison : Tous les processus corrects décident une valeur.

Impossibilité du consensus

Théorème de Fischer, Lynch et Paterson

Pas de solution au consensus asynchrone, tolérante aux pannes et déterministe !

Impossibilité du consensus

Théorème de Fischer, Lynch et Paterson

Pas de solution au consensus asynchrone, tolérante aux pannes et déterministe !

Pistes de solution

Synchronie : Mais peu de systèmes sont synchrones.

Impossibilité du consensus

Théorème de Fischer, Lynch et Paterson

Pas de solution au consensus asynchrone, tolérante aux pannes et déterministe !

Pistes de solution

Synchronie : Mais peu de systèmes sont synchrones.

Détecteur de fautes : Que fait-on si le détecteur de fautes n'est pas fiable ?

Impossibilité du consensus

Théorème de Fischer, Lynch et Paterson

Pas de solution au consensus asynchrone, tolérante aux pannes et déterministe !

Pistes de solution

Synchronie : Mais peu de systèmes sont synchrones.

Détecteur de fautes : Que fait-on si le détecteur de fautes n'est pas fiable ?

Non-déterminisme : Ajoutons des nombres aléatoires !

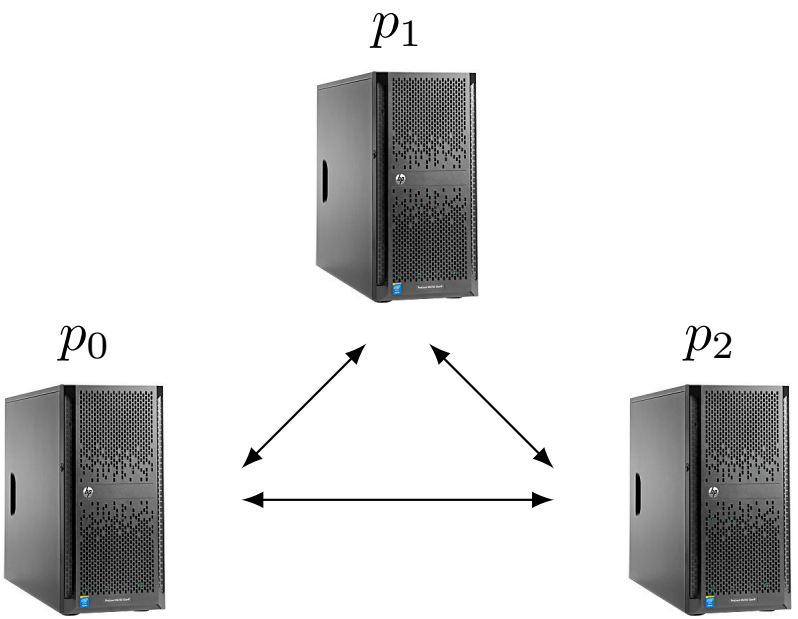
Consensus probabiliste

Accord : Au plus une valeur est décidée.

Validité : Toute valeur décidée a été proposée.

Terminaison probabiliste : Un processus correct décide avec probabilité 1.

Pourquoi le consensus est-il difficile ?



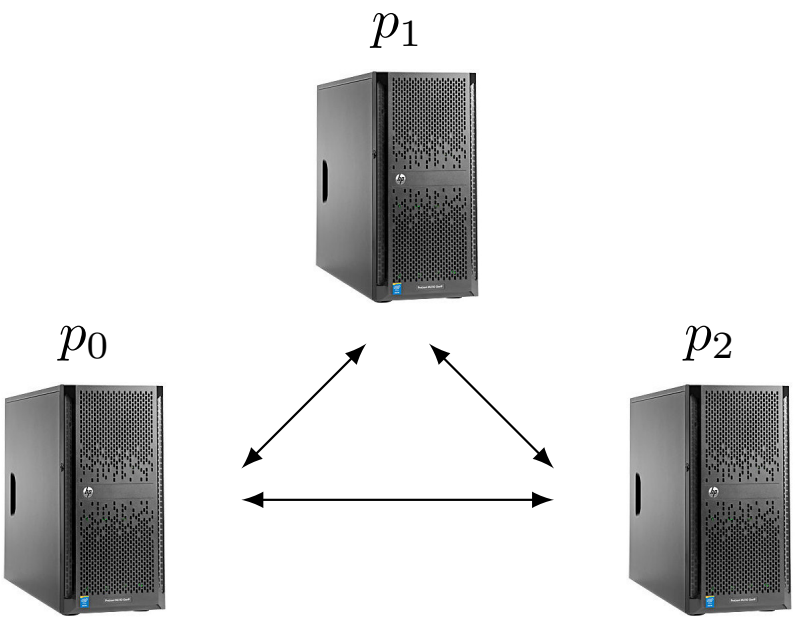
Point de vue de p_0

n = 3, t=1

- p_0 a proposé
- p_1 a proposé

p_0 peut-il attendre p_2 ?

Pourquoi le consensus est-il difficile ?



Point de vue de p_0

n = 3, t=1

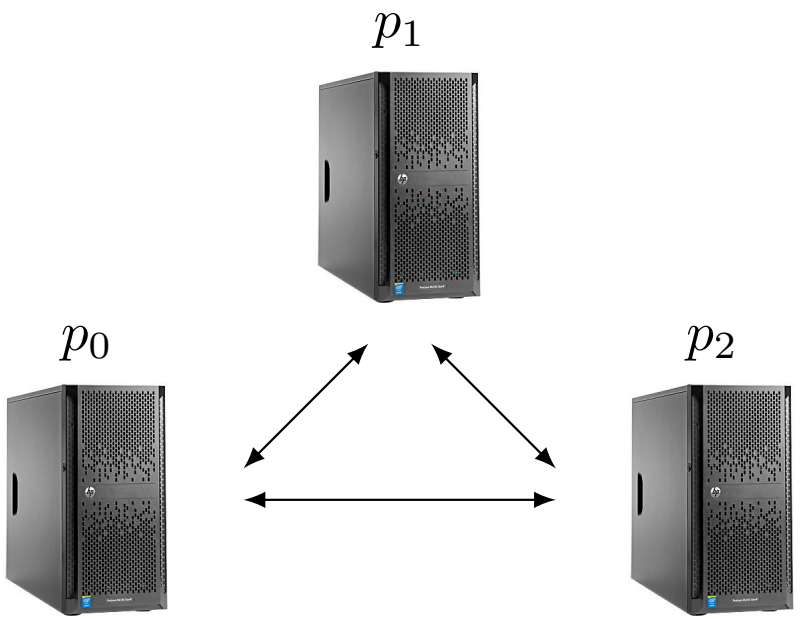
- p_0 a proposé
- p_1 a proposé

p_0 peut-il attendre p_2 ?

- non

Que peut faire p_0 ?

Pourquoi le consensus est-il difficile ?



Point de vue de p_0

n = 3, t=1

- p_0 a proposé
- p_1 a proposé

p_0 peut-il attendre p_2 ?

- non

Que peut faire p_0 ?

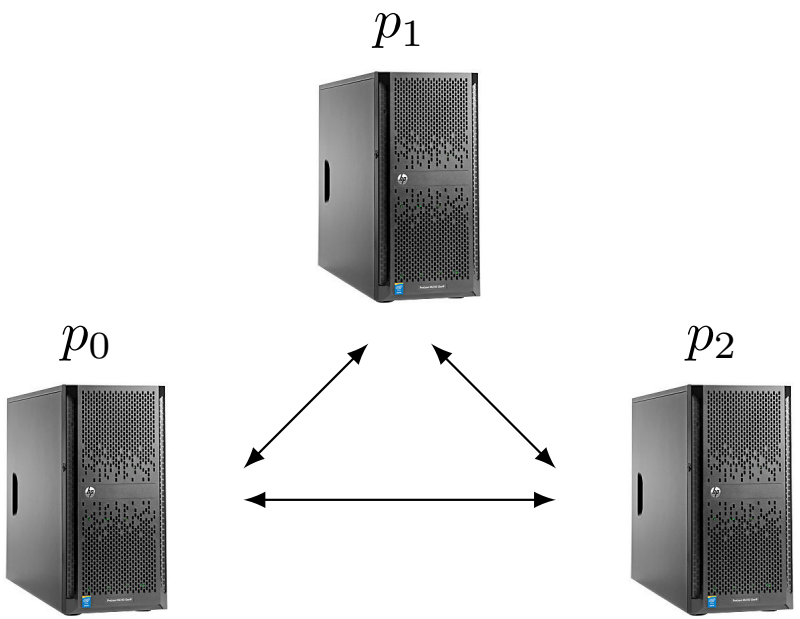
Situation 1

Point de vue de p_2

- p_2 a proposé
- p_0 a proposé

p_2 ne peut que décider

Pourquoi le consensus est-il difficile ?



Point de vue de p_0

n = 3, t=1

- p_0 a proposé
- p_1 a proposé

p_0 peut-il attendre p_2 ?

- non

Que peut faire p_0 ?

Situation 1

Point de vue de p_2

- p_2 a proposé
- p_0 a proposé

p_2 ne peut que décider

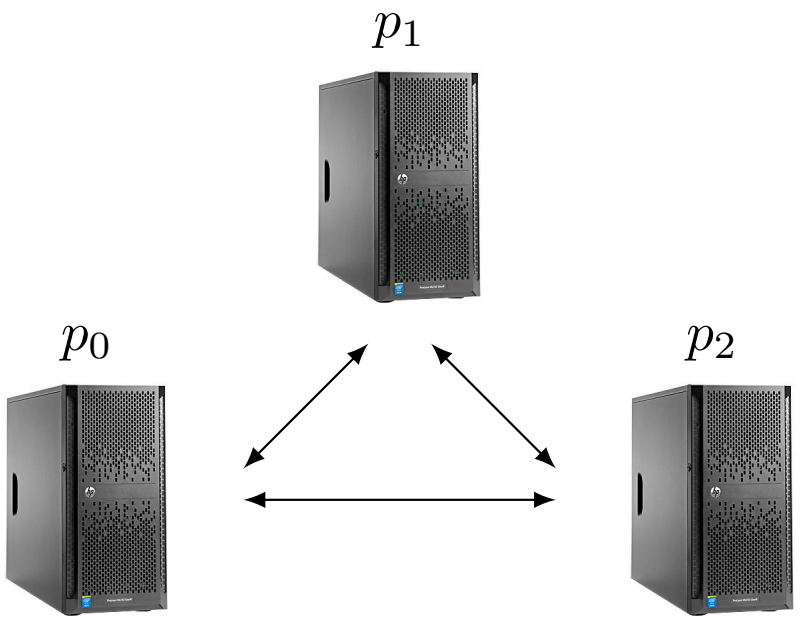
Situation 2

Point de vue de p_2

- p_2 a proposé
- p_1 a proposé

p_2 ne peut que décider

Pourquoi le consensus est-il difficile ?



Point de vue de p_0

n = 3, t=1

- p_0 a proposé
- p_1 a proposé

p_0 peut-il attendre p_2 ?

- non

Que peut faire p_0 ?

- dire "Je ne sais pas" $(Abort)$

Situation 1

Point de vue de p_2

- p_2 a proposé
- p_0 a proposé

p_2 ne peut que décider

Situation 2

Point de vue de p_2

- p_2 a proposé
- p_1 a proposé

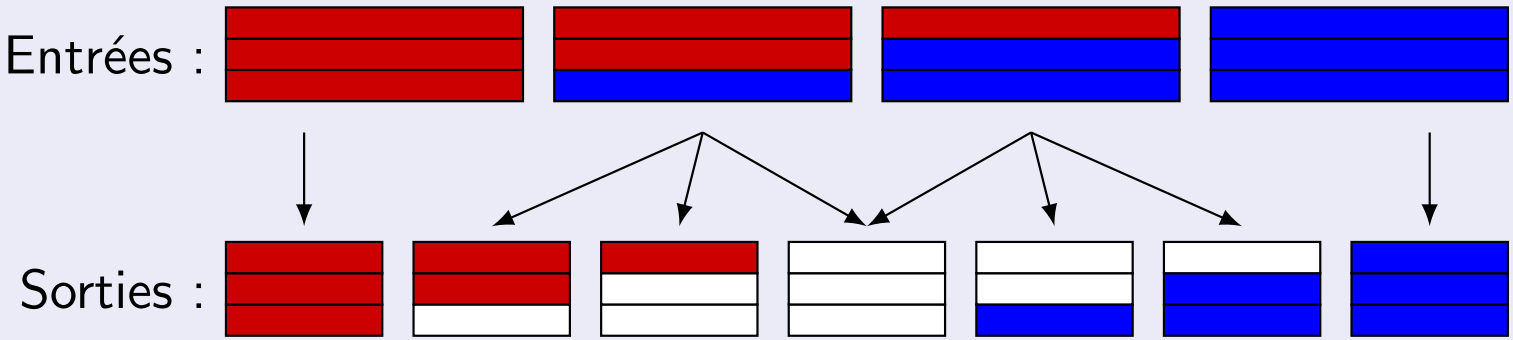
p_2 ne peut que décider

Pourquoi le consensus est-il difficile ?

Une étape de calcul

$\texttt{send}(v)$;
wait for $n - t$ messages;
if $\underline{\text{received} = \{w\}}$ **then** **return** w **else** **return** $\perp$;

Propriétés

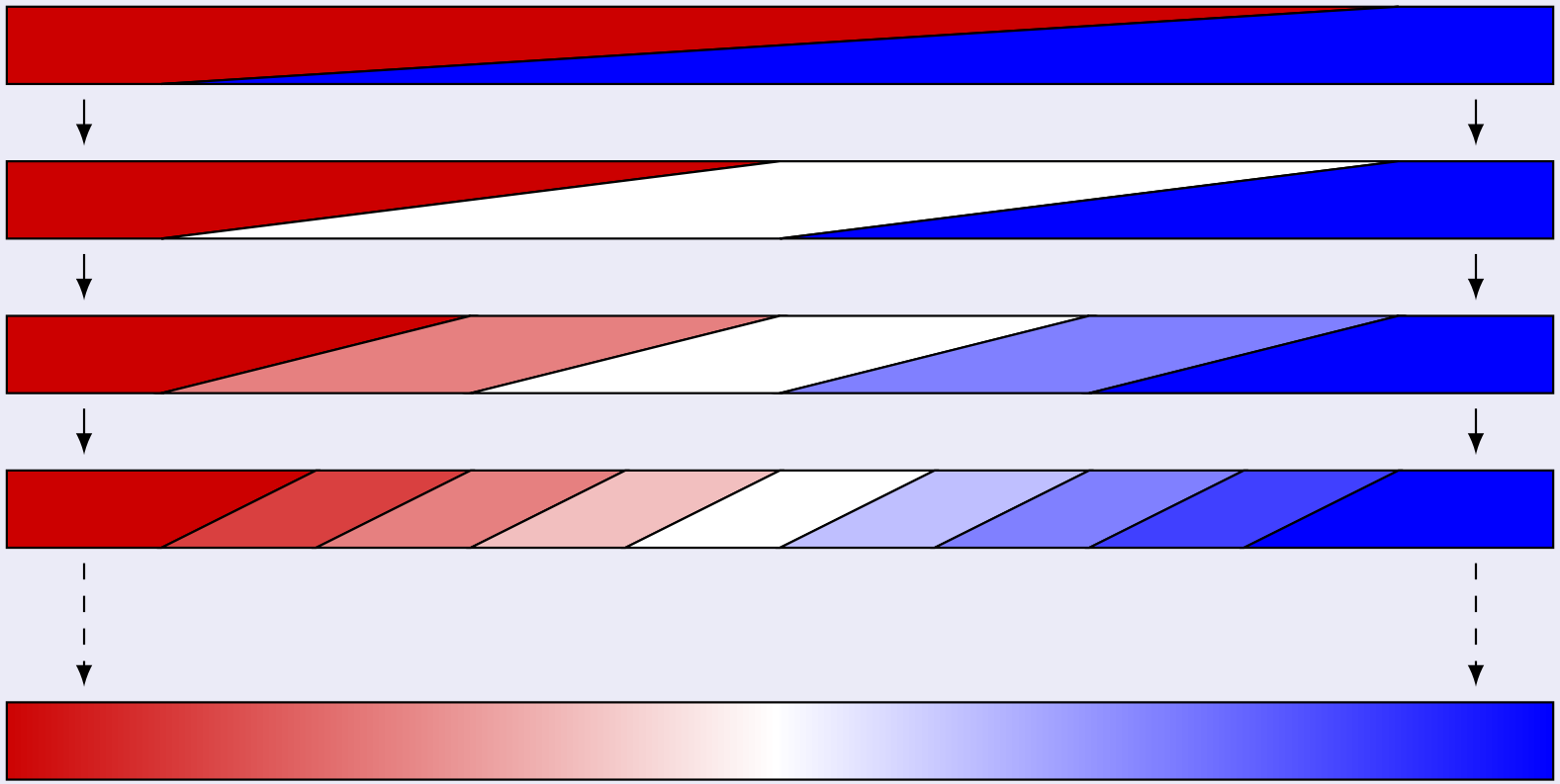


Conservation Si une seule valeur est proposée, l'accord est conservé

Pré-accord • et • sont exclusifs

Pourquoi le consensus est-il difficile ?

Itérations

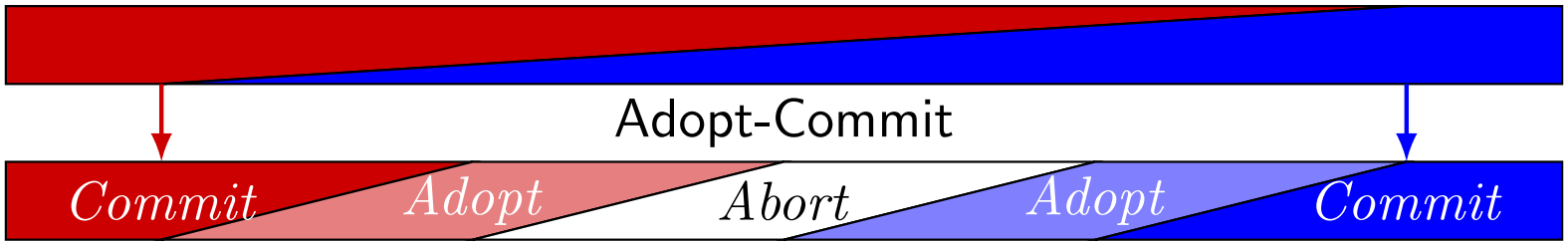


On ne peut jamais couper entre ● et ●.

Algorithme de Ben-Or

Abstraction Adopt-Commit

Sorties	*Commit*, *Adopt*, *Abort*, *Adopt*, *Commit*
Pré-accord	*Commit* et *Abort* exclusifs ; • et • exclusifs
Conservation	Si une seule valeur est proposée, tout le monde obtient *Commit*



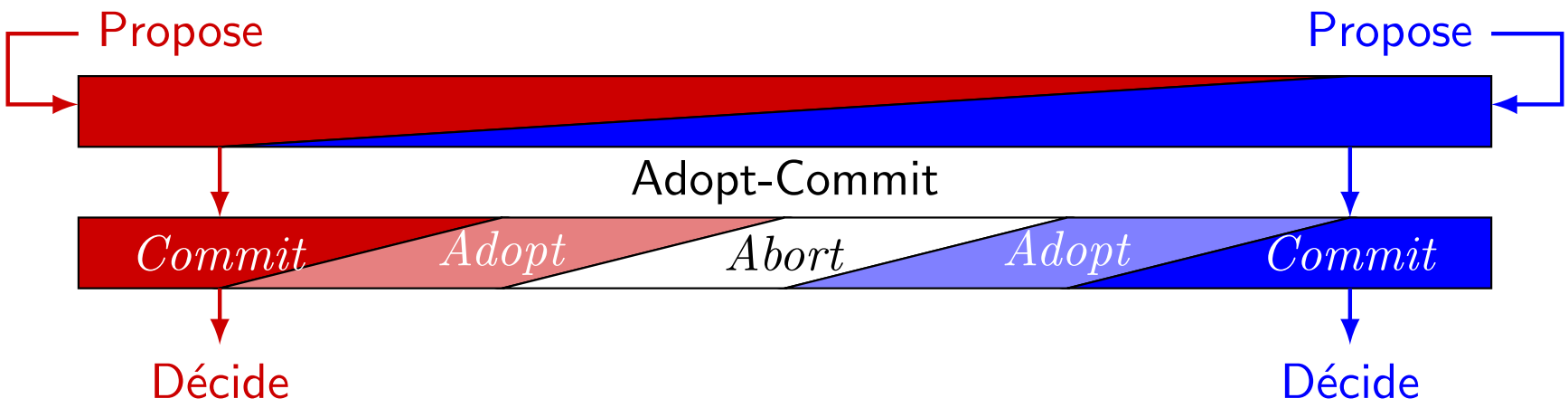
Algorithme de Ben-Or

Abstraction Adopt-Commit

Sorties $Commit$, $Adopt$, $Abort$, $Adopt$, $Commit$

Pré-accord $Commit$ et $Abort$ exclusifs ; • et • exclusifs

Conservation Si une seule valeur est proposée, tout le monde obtient $Commit$



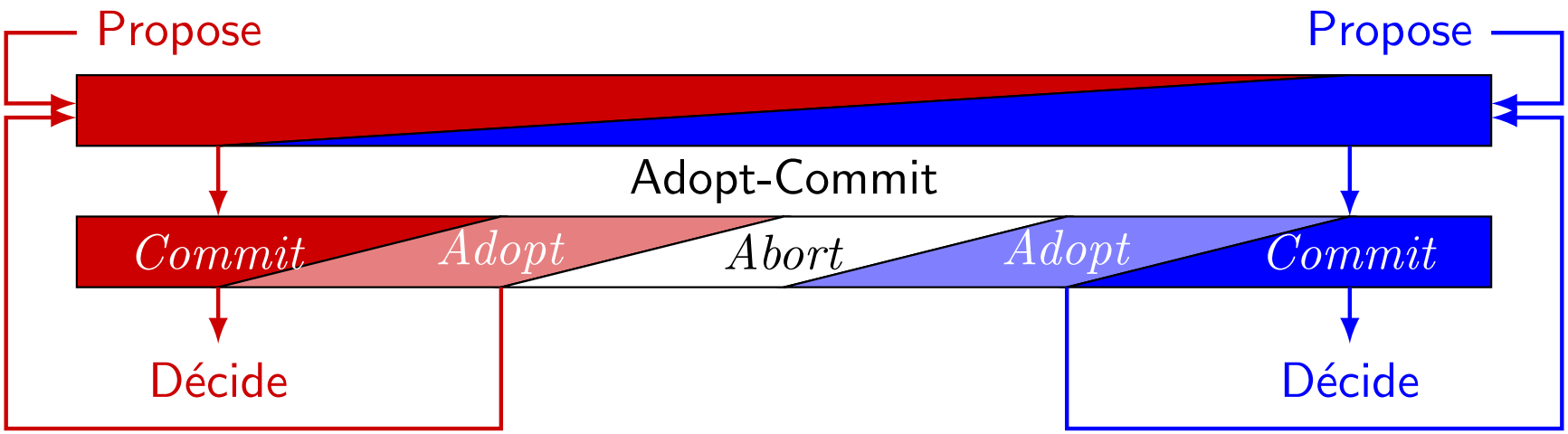
Algorithme de Ben-Or

Abstraction Adopt-Commit

Sorties $Commit$, $Adopt$, $Abort$, $Adopt$, $Commit$

Pré-accord $Commit$ et $Abort$ exclusifs ; ● et ● exclusifs

Conservation Si une seule valeur est proposée, tout le monde obtient $Commit$



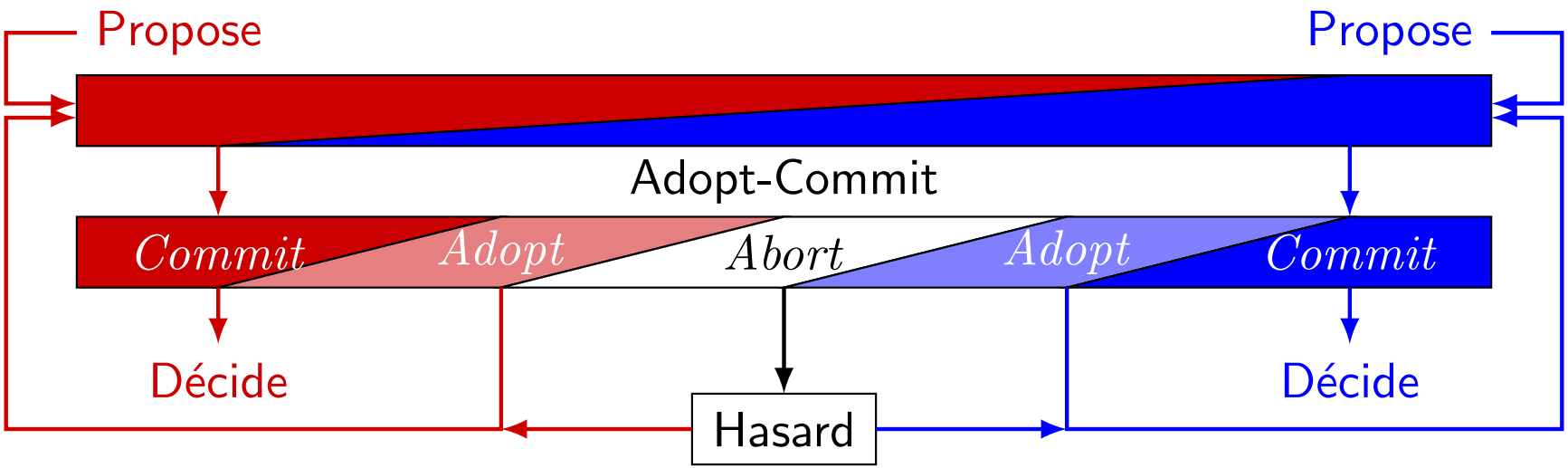
Algorithme de Ben-Or

Abstraction Adopt-Commit

Sorties $Commit$, $Adopt$, $Abort$, $Adopt$, $Commit$

Pré-accord $Commit$ et $Abort$ exclusifs ; ● et ● exclusifs

Conservation Si une seule valeur est proposée, tout le monde obtient $Commit$



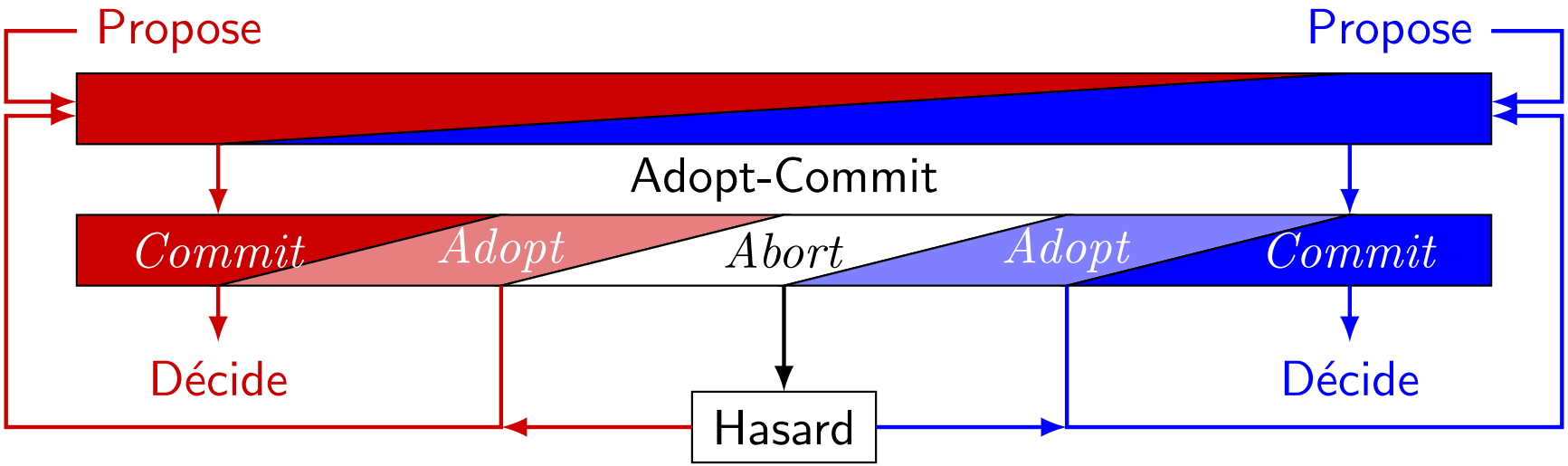
Algorithme de Ben-Or

Abstraction Adopt-Commit

Sorties $Commit$, $Adopt$, $Abort$, $Adopt$, $Commit$

Pré-accord $Commit$ et $Abort$ exclusifs ; • et • exclusifs

Conservation Si une seule valeur est proposée, tout le monde obtient $Commit$



Complexité moyenne

2^{n-1} rondes $\Rightarrow$ pièce commune ?

Pièce commune

Théorème central limite

$X_1, \dots X_k$ des variables aléatoires dans $\{-1, 1\}$:

$$\mathbb{P}\left(-\alpha\sqrt{k} < \sum_{i=1}^{k} X_i < \alpha\sqrt{k}\right) \xrightarrow[k\to\infty]{} f(\alpha)$$

Pièce commune

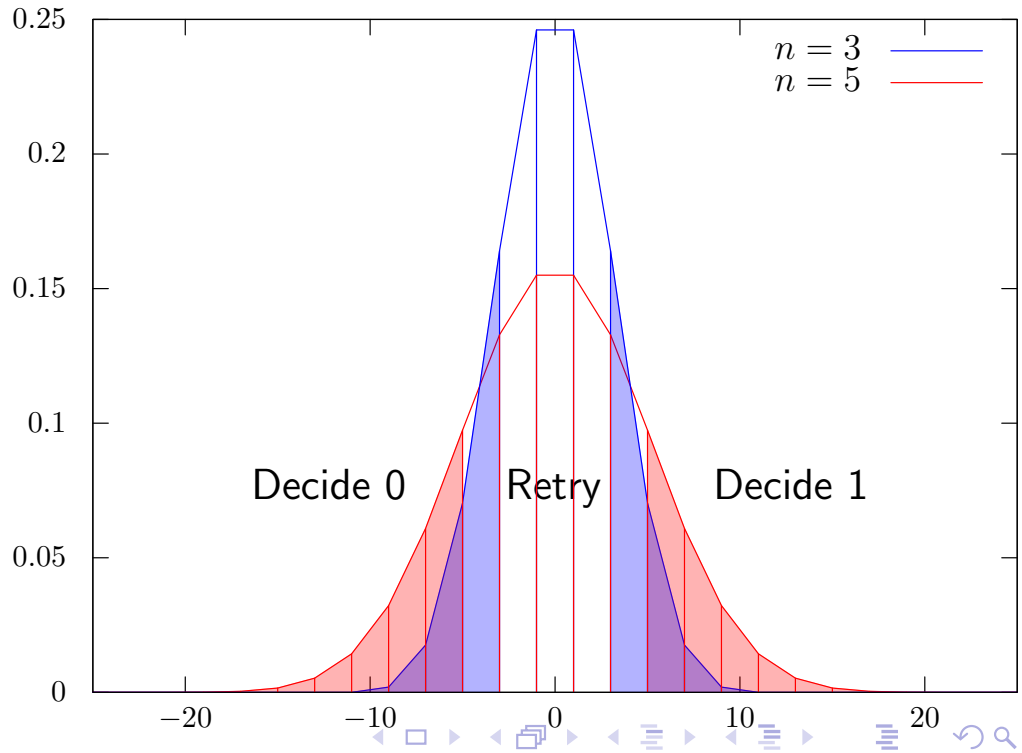
Théorème central limite

$X_1, \dots X_k$ des variables aléatoires dans $\{-1, 1\}$:

$$\mathbb{P}\left(-\alpha\sqrt{k} < \sum_{i=1}^{k} X_i < \alpha\sqrt{k}\right) \xrightarrow[k\to\infty]{} f(\alpha)$$

Algorithme de pièce commune

```
while cpt < n^2 do
    if random() then
        sum.increment()
    else
        sum.decrement()
    cpt.increment();
return sign(sum);
```



Stage d'Antoine Domenech (ENS de Lyon)

Remarques

- $\mathcal{O}(n^2)$ tirages sont nécessaires pour la pièce commune.
- En modifiant l'algorithme de la pièce commune, on obtient le consensus.

Stage d'Antoine Domenech (ENS de Lyon)

Remarques

- $\mathcal{O}(n^2)$ tirages sont nécessaires pour la pièce commune.
- En modifiant l'algorithme de la pièce commune, on obtient le consensus.

Spécification du consensus

Terminaison (avec probabilité 1)

Accord : au plus une valeur est décidée.

Validité : toute valeur décidée a été proposée

Spécification de la pièce commune

Terminaison (avec probabilité 1)

Accord : au plus une valeur est décidée

Hasard : toutes les valeurs ont une probabilité ρ

Stage d'Antoine Domenech (ENS de Lyon)

Remarques

- $\mathcal{O}(n^2)$ tirages sont nécessaires pour la pièce commune.
- En modifiant l'algorithme de la pièce commune, on obtient le consensus.

Spécification du consensus

Terminaison (avec probabilité 1)

Accord : au plus une valeur est décidée.

Validité : toute valeur décidée a été proposée

Spécification de la pièce commune

Terminaison (avec probabilité 1)

Accord : au plus une valeur est décidée

Hasard : toutes les valeurs ont une probabilité ρ

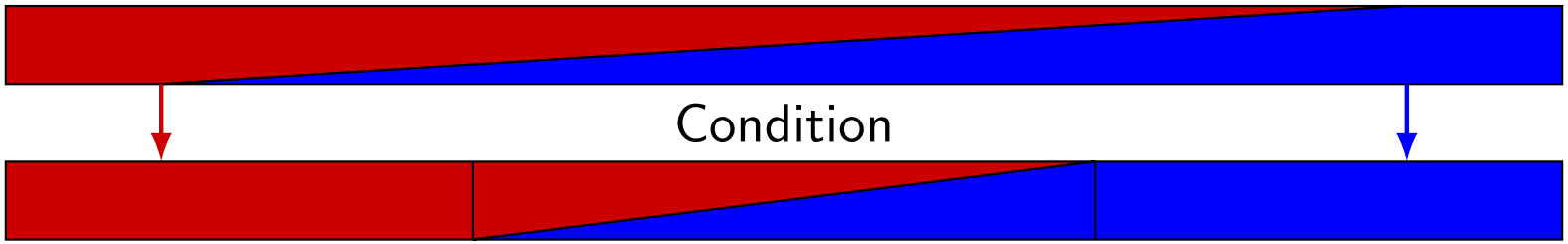
Questions ouvertes

Quelle est la complexité du consensus pobabiliste ?

Stage de Julien Weibel (ENS Paris)

Étendre la condition de terminaison

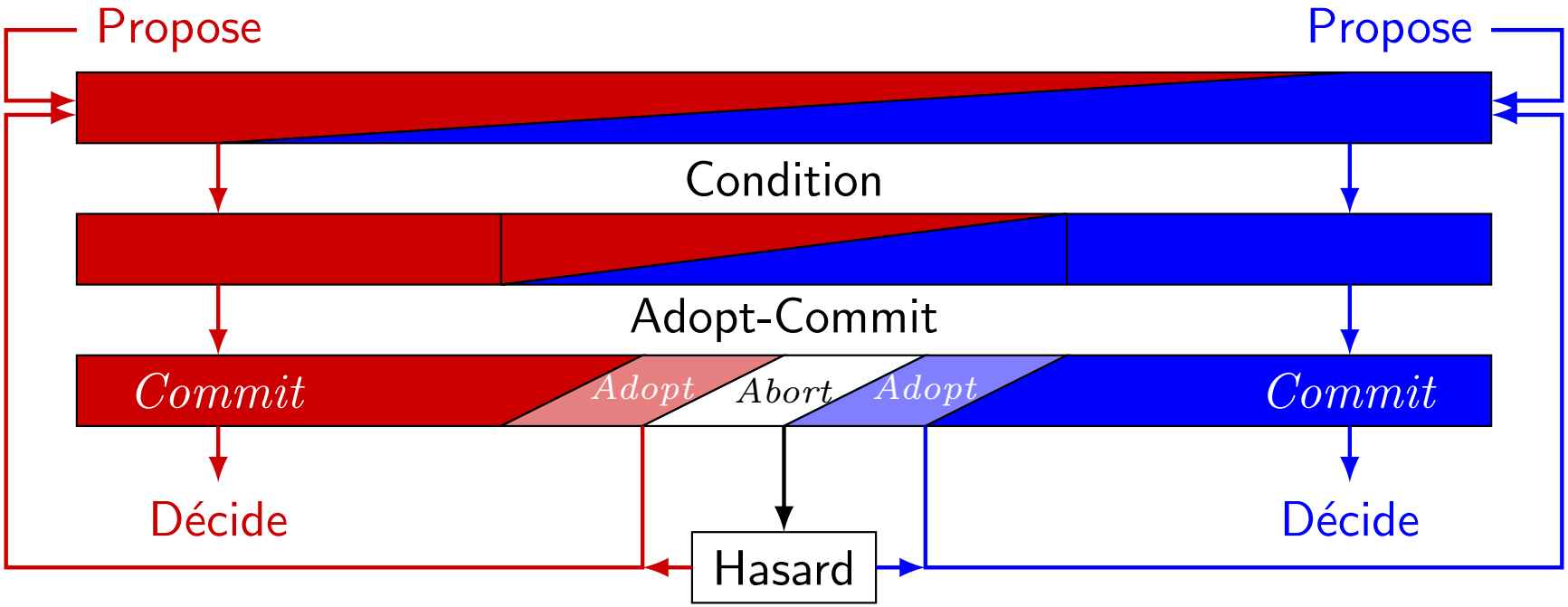
`send`(v);
wait for $n - t$ messages;
return most received value



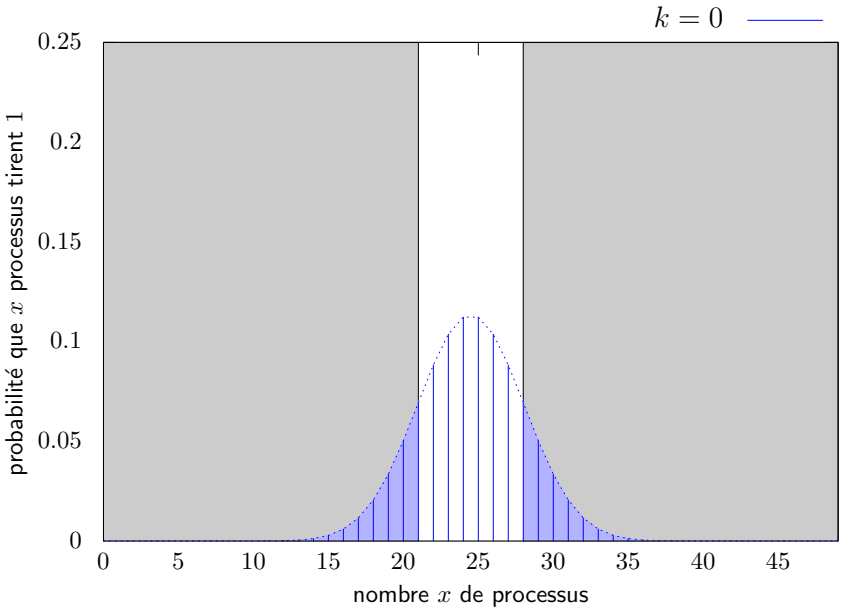
Stage de Julien Weibel (ENS Paris)

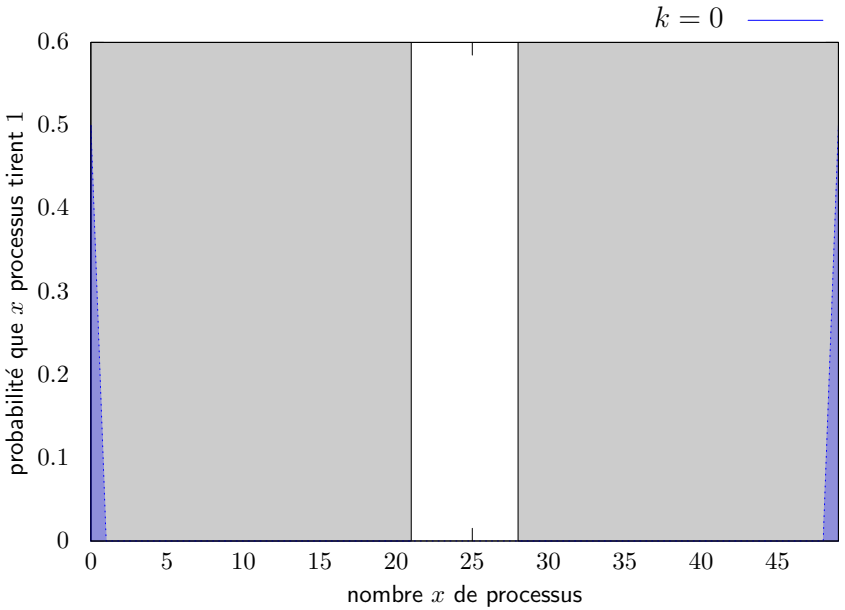
Étendre la condition de terminaison

`send`(v);
wait for $n - t$ messages;
return most received value

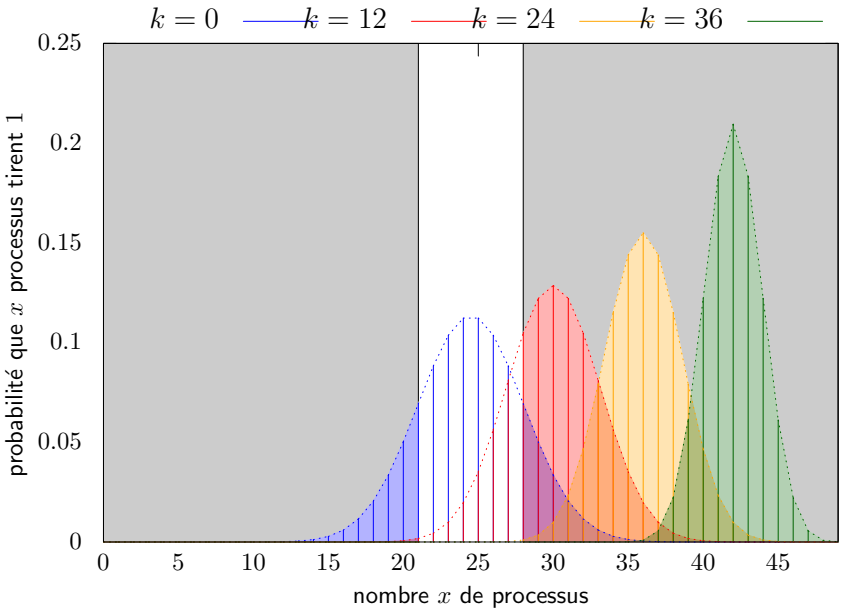


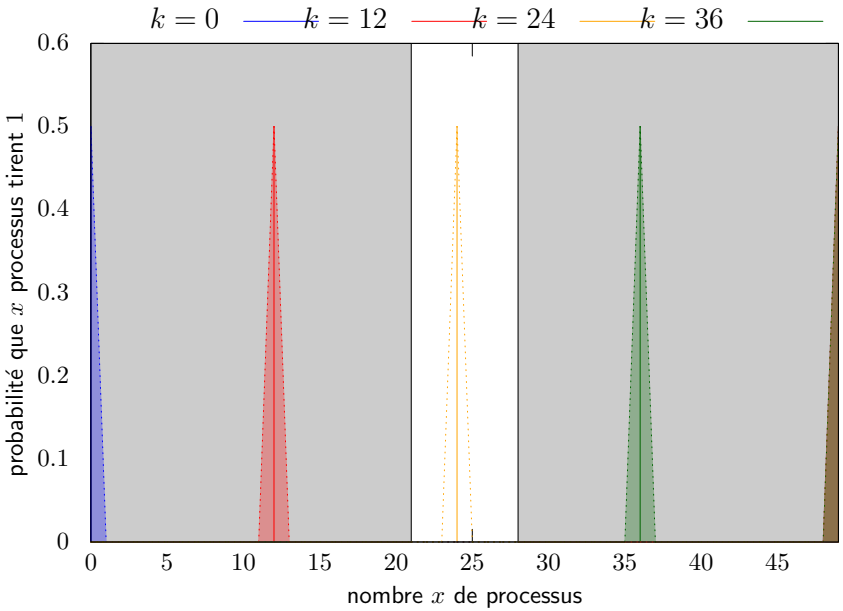
Stage de Julien Weibel (ENS Paris)



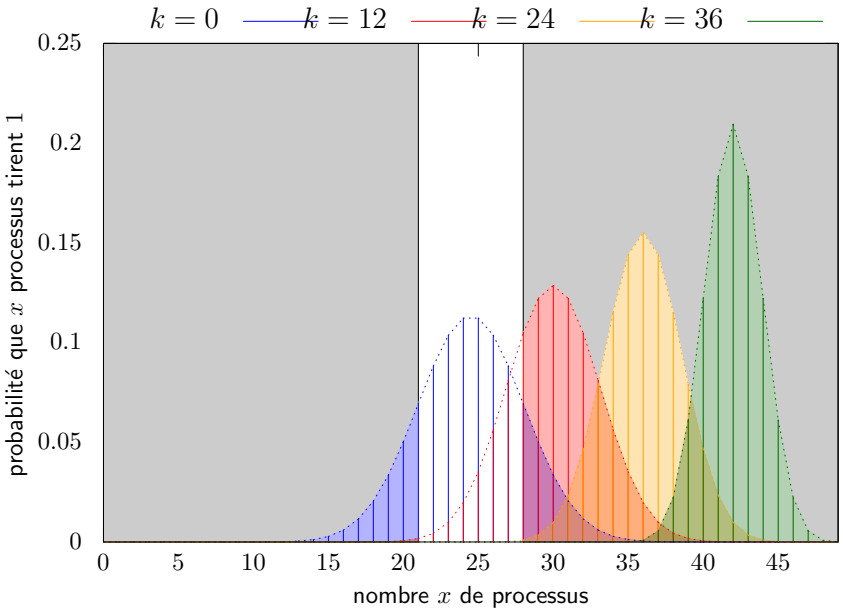


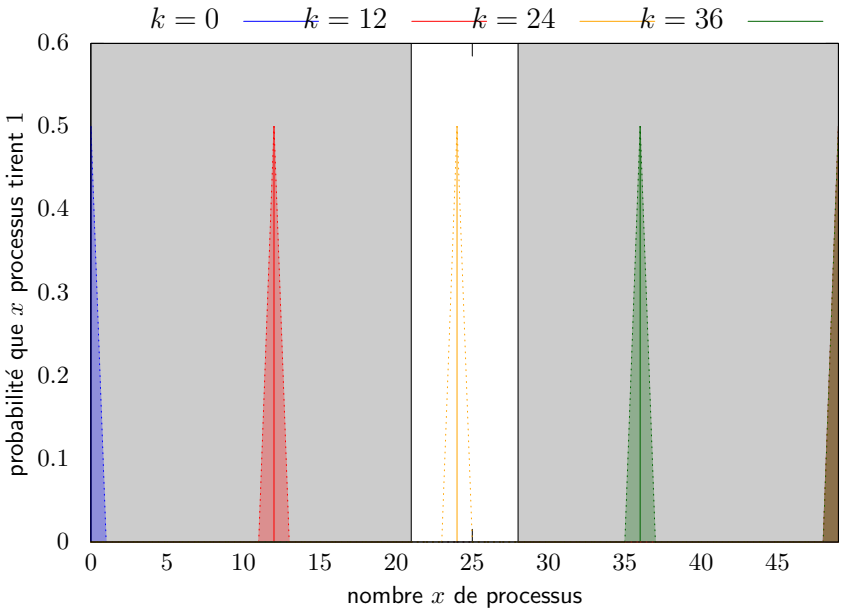
Stage de Julien Weibel (ENS Paris)





Stage de Julien Weibel (ENS Paris)





Observation

Quand t est assez petit, la pièce locale est plus efficace que la pièce commune !

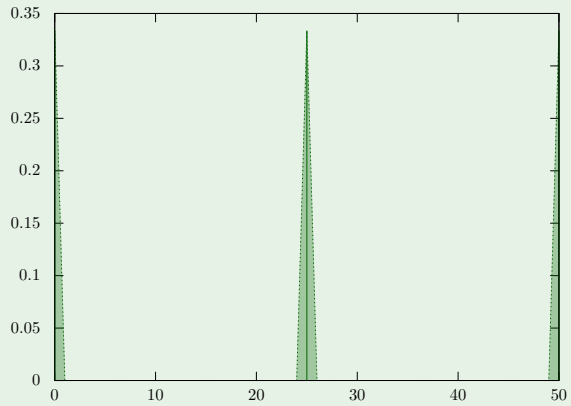
Stage de Julien Weibel (ENS Paris)

Travail de Julien

Problème : Pour t et n donnés, quelle est la pièce optimale ?

Contribution : Caractérisation de la pièce optimale en fonction de n et t.

Pièce optimale pour $t < \frac{n}{3}$



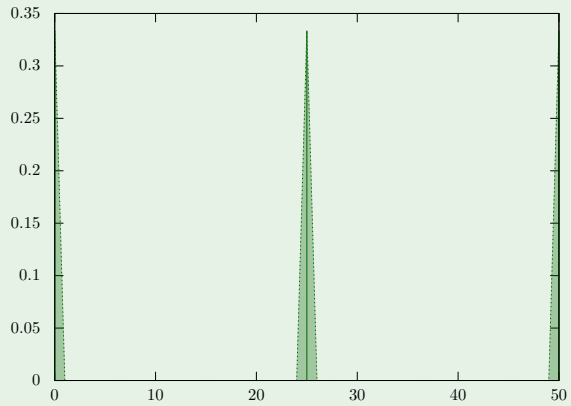
Stage de Julien Weibel (ENS Paris)

Travail de Julien

Problème : Pour t et n donnés, quelle est la pièce optimale ?

Contribution : Caractérisation de la pièce optimale en fonction de n et t.

Pièce optimale pour $t < \frac{n}{3}$



Question ouverte

Comment créer des pièces exotiques ?

Bien d'autres questions ouvertes

Adaptation à la puissance de l'adversaire

- L'asynchronie est gérée par un adversaire
- Algorithmes plus efficaces quand l'adversaire ne peut pas lire les messages

Question ouverte

Existe-t-il des algorithmes efficaces à la fois face à un adversaire fort et face à un adversaire faible ?

Bien d'autres questions ouvertes

Adaptation à la puissance de l'adversaire

- L'asynchronie est gérée par un adversaire
- Algorithmes plus efficaces quand l'adversaire ne peut pas lire les messages

Question ouverte

Existe-t-il des algorithmes efficaces à la fois face à un adversaire fort et face à un adversaire faible ?

Adaptation au nombre de participants

Souvent, tout le monde ne propose pas de valeur.

Question ouverte

Peut-on avoir une complexité qui dépend du nombre de participants ?

Conclusion

Le consensus est un beau problème.

- Utile en pratique
  - Constructions universelles
- Utile en théorie
  - Le ≪ problème de l'arrêt ≫ du distribué
- Passionnant en recherche
  - Des liens avec de nombreux problèmes mathématiques

Beaucoup de questions ouvertes

- Construction de pièces exotiques
- Complexité du consensus sans pièce commune
- Adaptation au contexte
- ...

Nous serions heureux d'y réfléchir avec vous !