

# Mise au point d'algorithmes répartis en environnement fortement variable, et expérimentation sur réseaux sans fil

Corine MARCHAND

Directeur de thèse : M<sup>me</sup> Brigitte PATEAU

Co-encadrant : M. Jean-Marc VINCENT

Laboratoire ID-IMAG (UMR 5132), Projet Apache.

Projet MIRRA : Contrat de collaboration INRIA - France Télécom R&D



Laboratoire  
Informatique et  
Distribution



Institut National  
Polytechnique  
de Grenoble



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



INSTITUT NATIONAL  
DE RECHERCHE EN  
INFORMATIQUE ET  
EN AUTOMATIQUE



GRENOBLE  
UNIVERSITÉ  
JOSEPH FOURIER  
SCIENCE TECHNOLOGIE MÉDECINE



## Environnement

### Environnement type



### Objets Communicants

- Ordinateurs portables
- Assistants personnels
- Téléphones cellulaires
- ...

### Réseau ad-hoc sans fil

- WIFI
- Bluetooth
- ...

### Principe

Les entités partagent des **services** et collaborent au **maintien** de la communauté

**Exemples** : Réunion de travail, jeux en réseau

- 1 Réseaux ambiants
- 2 Caractérisation de la variabilité
- 3 Algorithme de consensus en environnement variable
- 4 Détecteurs de défaillances et qualité de service
- 5 Analyse et évaluation du module de consensus
- 6 Conclusions et Perspectives

# Réseaux ambiants

## Applications sur réseaux sans fil

### Architecture dynamique

- Comportement des objets communicants
  - Connexions / Déconnexions
- Comportement de l'interconnexion
  - Communications non fiables
  - Variabilité des latences
- Hétérogénéité

⇒ Construction d'une infrastructure middleware distribuée.

### Offrir une palette d'outils middleware

- Garantir la disponibilité des services
- Gérer l'accès cohérent aux services

## Applications sur réseaux sans fil

### Architecture dynamique

- Comportement des objets communicants
  - Connexions / Déconnexions
- Comportement de l'interconnexion
  - Communications non fiables
  - Variabilité des latences
- Hétérogénéité

⇒ Construction d'une infrastructure middleware distribuée.

### Offrir une palette d'outils middleware

- Garantir la disponibilité des services
- Gérer l'accès cohérent aux services

## Applications sur réseaux sans fil

### Architecture dynamique

- Comportement des objets communicants
  - Connexions / Déconnexions
- Comportement de l'interconnexion
  - Communications non fiables
  - Variabilité des latences
- Hétérogénéité

⇒ Construction d'une infrastructure middleware distribuée.

### Offrir une palette d'outils middleware

- Garantir la disponibilité des services
- Gérer l'accès cohérent aux services

## Contexte général

### Problématique

Comment assurer le maintien de la cohérence de la communauté dans un réseau ambiant ?

### Moyen

- Prise de décision répartie.
- Conception et mise au point d'algorithmes répartis "adaptés".

### Réalisation

Construction de modules middleware adaptés à la dynamique de l'architecture.

→ A terme, masquer l'instabilité de l'architecture aux applications

## Contexte général

### Problématique

Comment assurer le maintien de la cohérence de la communauté dans un réseau ambiant ?

### Moyen

- Prise de décision répartie.
- Conception et mise au point d'algorithmes répartis "adaptés".

### Réalisation

Construction de modules middleware adaptés à la dynamique de l'architecture.

→ A terme, masquer l'instabilité de l'architecture aux applications

## Contexte général

### Problématique

Comment assurer le maintien de la cohérence de la communauté dans un réseau ambiant ?

### Moyen

- Prise de décision répartie.
- Conception et mise au point d'algorithmes répartis "adaptés".

### Réalisation

Construction de modules middleware adaptés à la dynamique de l'architecture.

→ A terme, masquer l'instabilité de l'architecture aux applications

## Contexte général

### Approche

- Adaptation des algorithmes existants.

### Méthodologie

- Caractérisation de l'environnement
  - Identification et quantification des particularités.
- Adaptation des algorithmes répartis
  - Adaptation interne → comportementale.
  - Mise au point et dimensionnement → réglage du paramétrage.
- Validation expérimentale

## Contexte général

### Approche

- Adaptation des algorithmes existants.

### Méthodologie

- Caractérisation de l'environnement
  - Identification et quantification des particularités.
- Adaptation des algorithmes répartis
  - Adaptation interne → comportementale.
  - Mise au point et dimensionnement → réglage du paramétrage.
- Validation expérimentale

# Caractérisation de la variabilité

## Caractérisation de la dynamique

### Adaptations nécessaires :

- Non fiabilité des communications
- Forte variabilité des temps de communication
- Le problème des déconnexions / reconnexions fréquentes
- Hétérogénéité des entités

### Objectif :

Exhiber et analyser les comportements de l'environnement.

- Niveau comportemental (connexion/déconnexion).
- 

### Campagnes d'expériences :

Expérimentations et analyse de performances.

## Caractérisation de la dynamique

### Adaptations nécessaires :

- Non fiabilité des communications
- Forte variabilité des temps de communication
- Le problème des déconnexions / reconnexions fréquentes
- Hétérogénéité des entités

### Objectif :

Exhiber et analyser les comportements de l'environnement.

- Niveau comportemental (connexion/déconnexion).
- Niveau communication (variabilité).

### Campagnes d'expériences :

Expérimentations et analyse de performances.

## Caractérisation de la dynamique

### Adaptations nécessaires :

- Non fiabilité des communications
- Forte variabilité des temps de communication
- Le problème des déconnexions / reconnexions fréquentes
- Hétérogénéité des entités

### Objectif :

Exhiber et analyser les comportements de l'environnement.

- Niveau comportemental (connexion/déconnexion).
- **Niveau communication (variabilité).**

### Campagnes d'expériences :

Expérimentations et analyse de performances.

## Caractérisation de la dynamique

### Adaptations nécessaires :

- Non fiabilité des communications
- Forte variabilité des temps de communication
- Le problème des déconnexions / reconnexions fréquentes
- Hétérogénéité des entités

### Objectif :

Exhiber et analyser les comportements de l'environnement.

- Niveau comportemental (connexion/déconnexion).
- **Niveau communication (variabilité).**

### Campagnes d'expériences :

Expérimentations et analyse de performances.

## Paramètre étudié : Latence

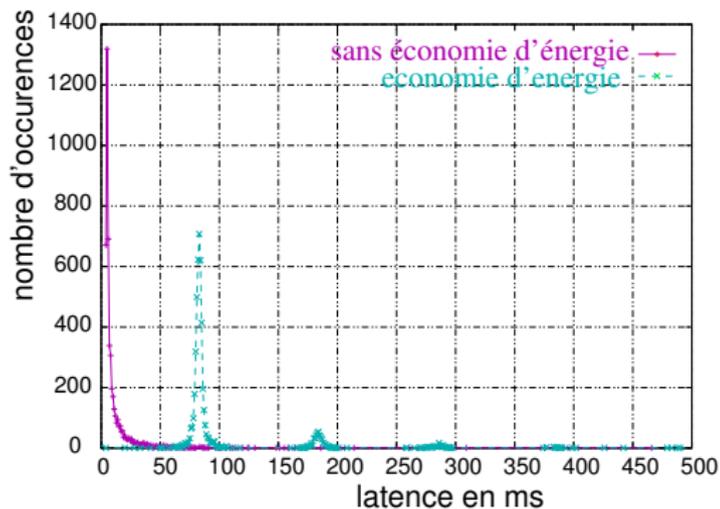
Expériences préliminaires : **Latence**, débit et pertes

## Contexte expérimental :

- Protocole 802.11b
- Mode ad-hoc

## Mesure de latence :

- PDAs 1<sup>ère</sup> génération
- ping (délai : 1s)
- Taille échantillon : 5000 mesures



Facteur influent : Économie d'énergie

⇒ Identification des facteurs influents

## Paramètres étudié : Latence

## Analyse factorielle des latences

## Facteurs retenus :

- Distance
- Nb obstacles
- Nb entités
- Charge réseau
- Type récepteur
- Type émetteur
- Économie d'énergie

+ Interactions

Restrictions : environnement de type salle de réunion

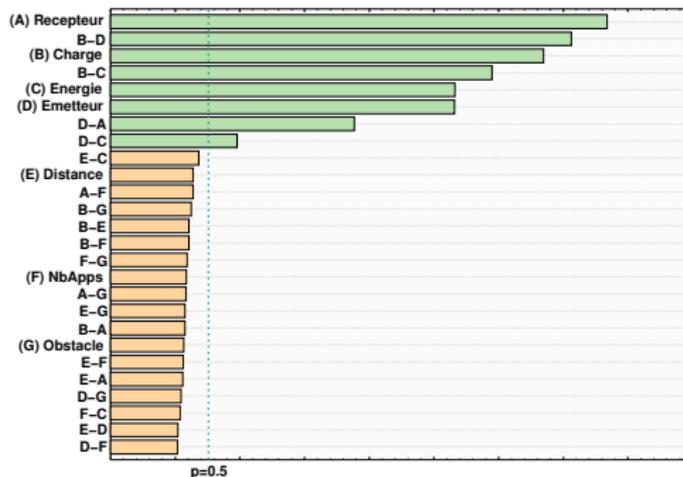
## Paramètres étudié : Latence

## Analyse factorielle des latences

## Facteurs retenus :

- Distance
- Nb obstacles
- Nb entités
- Charge réseau
- Type récepteur
- Type émetteur
- Économie d'énergie

+ Interactions



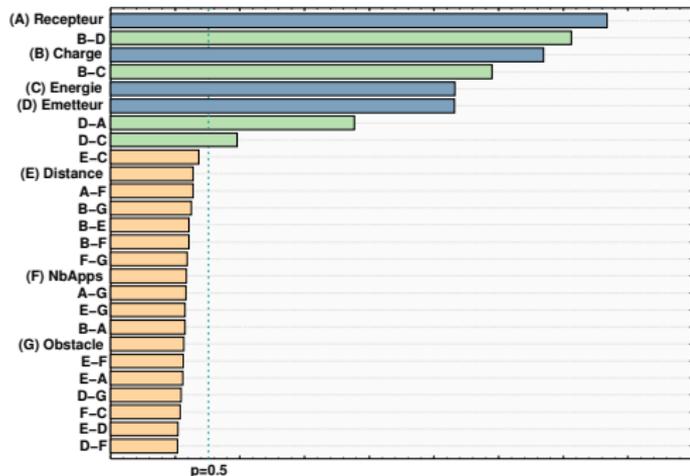
Restrictions : environnement de type salle de réunion

## Paramètres étudié : Latence

## Analyse factorielle des latences

## Facteurs prépondérants :

- Distance
- Nb obstacles
- Nb entités
- Charge réseau
- Type récepteur
- Type émetteur
- Économie d'énergie
- + Interactions



Restrictions : environnement de type salle de réunion

## Bilan

### Étude environnementale

- Niveau réseau d'interconnexion :
  - Identification des spécificités de l'environnement.
  - Quantification des performances escomptées.
- Niveau objets communicants :
  - Qualification des comportements.
  - Caractérisation de l'hétérogénéité

### Intérêt :

- Adaptation spécifique des algorithmes
- Réglage adéquat des paramètres

## Bilan

### Étude environnementale

- Niveau réseau d'interconnexion :
  - Identification des spécificités de l'environnement.
  - Quantification des performances escomptées.
- Niveau objets communicants :
  - Qualification des comportements.
  - Caractérisation de l'hétérogénéité

### Intérêt :

- Adaptation spécifique des algorithmes
- Réglage adéquat des paramètres

## Bilan

### Étude environnementale

- Niveau réseau d'interconnexion :
  - Identification des spécificités de l'environnement.
  - Quantification des performances escomptées.
- Niveau objets communicants :
  - Qualification des comportements.
  - Caractérisation de l'hétérogénéité

### Intérêt :

- Adaptation spécifique des algorithmes
- Réglage adéquat des paramètres

## Bilan

### Étude environnementale

- Niveau réseau d'interconnexion :
  - Identification des spécificités de l'environnement.
  - Quantification des performances escomptées.
- Niveau objets communicants :
  - Qualification des comportements.
  - Caractérisation de l'hétérogénéité

### Intérêt :

- Adaptation spécifique des algorithmes
- Réglage adéquat des paramètres

# Algorithme de consensus en environnement variable

## Le problème du consensus

### Problèmes de "cohérence applicative" :

- Gestion de groupe
- Continuité de service
- Gestion de ressources

### Classes d'algorithmes :

- Diffusion atomique
- Élection
- Consensus
- Enregistrements cohérents d'états
- ...

⇒ Brique de base : Algorithme de Consensus

## Le problème du consensus

### Problèmes de "cohérence applicative" :

- Gestion de groupe
- Continuité de service
- Gestion de ressources

### Classes d'algorithmes :

- Diffusion atomique
- Élection
- Consensus
- Enregistrements cohérents d'états
- ...

⇒ Brique de base : Algorithme de Consensus

## Le problème du consensus

### Problèmes de "cohérence applicative" :

- Gestion de groupe
- Continuité de service
- Gestion de ressources

### Classes d'algorithmes :

- Diffusion atomique
- Élection
- Consensus
- Enregistrements cohérents d'états
- ...

⇒ Brique de base : Algorithme de Consensus

## L'instabilité de l'environnement

### Modèles de l'environnement

- Au niveau des communications
  - Environnement asynchrone
  - Environnement non fiable
- Au niveau des défaillances
  - Défaillances définitives
  - Défaillances momentanées

## L'instabilité de l'environnement

### Modèles de l'environnement

- Au niveau des communications
  - Environnement asynchrone
  - Environnement non fiable
- Au niveau des défaillances
  - Défaillances définitives
  - Défaillances momentanées

## L'instabilité de l'environnement

### Modèles de l'environnement

- Au niveau des communications
  - Environnement asynchrone
  - Environnement non fiable
- Au niveau des défaillances
  - Défaillances définitives
  - Défaillances momentanées

## Le problème du consensus

### Impossibilité de Fisher, Lynch & Paterson

[Fischer-Lynch-Paterson 85]

#### Alternatives (Approches possibles)

- Algorithmes probabilistes [Canetti-Rabin 93]
- Algorithmes auto-stabilisants [Tixeuil 00]
- Approche partiellement synchrone [Dwork-Lynch-Stockmeyer 88]
- ....

⇒ Choix algorithmiques :

Consensus + Détecteurs de défaillances [Chandra-Toueg 96]

#### Intérêt

Concentrer l'analyse de la variabilité de l'environnement.

## Le problème du consensus

### Impossibilité de Fisher, Lynch & Paterson

[Fischer-Lynch-Paterson 85]

### Alternatives (Approches possibles)

- Algorithmes probabilistes [Canetti-Rabin 93]
- Algorithmes auto-stabilisants [Tixeuil 00]
- Approche partiellement synchrone [Dwork-Lynch-Stockmeyer 88]
- ....

⇒ Choix algorithmiques :

Consensus + Détecteurs de défaillances [Chandra-Toueg 96]

### Intérêt

Concentrer l'analyse de la variabilité de l'environnement.

## Le problème du consensus

### Impossibilité de Fisher, Lynch & Paterson

[Fischer-Lynch-Paterson 85]

### Alternatives (Approches possibles)

- Algorithmes probabilistes [Canetti-Rabin 93]
- Algorithmes auto-stabilisants [Tixeuil 00]
- Approche partiellement synchrone [Dwork-Lynch-Stockmeyer 88]
- ....

⇒ Choix algorithmiques :

Consensus + Détecteurs de défaillances [Chandra-Toueg 96]

### Intérêt

Concentrer l'analyse de la variabilité de l'environnement.

## Le problème du consensus

### Impossibilité de Fisher, Lynch & Paterson

[Fischer-Lynch-Paterson 85]

### Alternatives (Approches possibles)

- Algorithmes probabilistes [Canetti-Rabin 93]
- Algorithmes auto-stabilisants [Tixeuil 00]
- Approche partiellement synchrone [Dwork-Lynch-Stockmeyer 88]
- ....

⇒ Choix algorithmiques :

Consensus + Détecteurs de défaillances [Chandra-Toueg 96]

### Intérêt

Concentrer l'analyse de la variabilité de l'environnement.

## Les détecteurs de défaillances

### Fonctionnalité

Construction d'un état estimé de chaque site distant.

Vue locale de l'ensemble du système.

→ Liste des sites distants suspects (notion de "présence")

### Propriétés

- Exactitude (un processus correct ne doit pas être suspecté)
- Complétude (un processus incorrect doit être suspecté)

### Qualité de l'information et Réactivité

[Chen-Toueg-Aguilera 00]

- Erreurs non symétriques
- Risques associées différents

	<i>connecté</i>	<i>déconnecté</i>
<i>non suspecté</i>	ok	erreur type I
<i>suspecté</i>	erreur type II	ok

## Les détecteurs de défaillances

### Fonctionnalité

Construction d'un état estimé de chaque site distant.

Vue locale de l'ensemble du système.

→ Liste des sites distants suspects (notion de "présence")

### Propriétés

- Exactitude (un processus correct ne doit pas être suspecté)
- Complétude (un processus incorrect doit être suspecté)

### Qualité de l'information et Réactivité

[Chen-Toueg-Aguilera 00]

● Erreurs non symétriques

→ Risques associées différents

	<i>connecté</i>	<i>déconnecté</i>
<i>non suspecté</i>	ok	erreur type I
<i>suspecté</i>	erreur type II	ok

## Les détecteurs de défaillances

### Fonctionnalité

Construction d'un état estimé de chaque site distant.

Vue locale de l'ensemble du système.

→ Liste des sites distants suspects (notion de "présence")

### Propriétés

- Exactitude (un processus correct ne doit pas être suspecté)
- Complétude (un processus incorrect doit être suspecté)

### Qualité de l'information et Réactivité

[Chen-Toueg-Aguilera 00]

- Erreurs non symétriques
- Risques associées différents

	<i>connecté</i>	<i>déconnecté</i>
<i>non suspecté</i>	ok	erreur type I
<i>suspecté</i>	erreur type II	ok

## Algorithme de consensus

### Principe algorithmique

### Algorithme de Chandra & Toueg [Chandra-Toueg 96]

#### Hypothèses

- Communications fiables
- Majorité de processus corrects

#### Principe de fonctionnement

- Rounds asynchrones
- 1 coordinateur par round
- 4 étapes par round

## Algorithme de consensus

### Principe algorithmique

Algorithme de Chandra & Toueg [Chandra-Toueg 96]

#### Hypothèses

- Communications fiables
- Majorité de processus corrects

#### Principe de fonctionnement

- Rounds asynchrones
- 1 coordinateur par round
- 4 étapes par round

## Algorithme de consensus

### Principe algorithmique

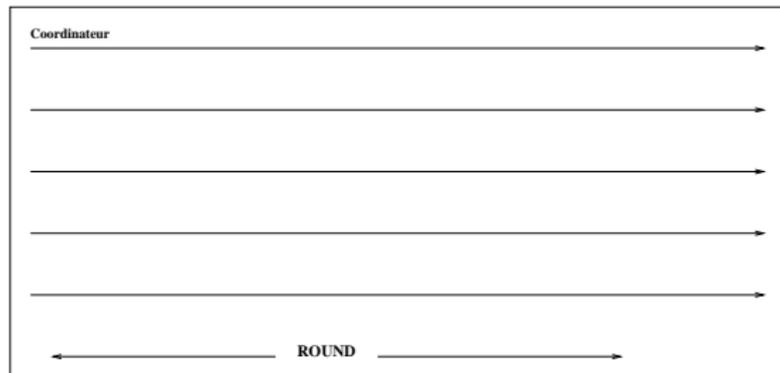
### Algorithme de Chandra & Toueg [Chandra-Toueg 96]

#### Hypothèses

- Communications fiables
- Majorité de processus corrects

#### Principe de fonctionnement

- Rounds asynchrones
- 1 coordinateur par round
- 4 étapes par round



## Algorithme de consensus

### Principe algorithmique

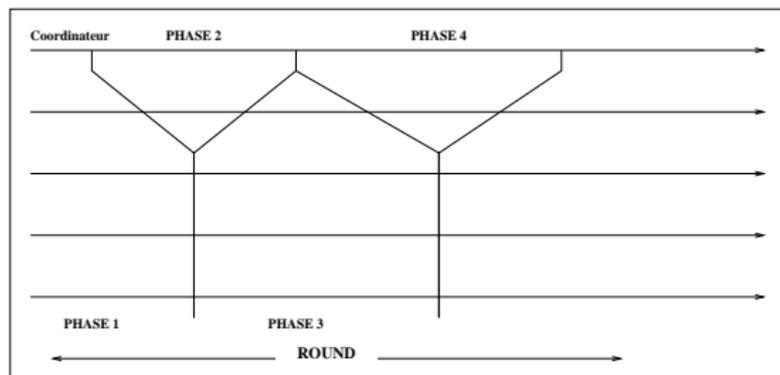
### Algorithme de Chandra & Toueg [Chandra-Toueg 96]

#### Hypothèses

- Communications fiables
- Majorité de processus corrects

#### Principe de fonctionnement

- Rounds asynchrones
- 1 coordinateur par round
- 4 étapes par round



## Algorithme de consensus

### Principe algorithmique

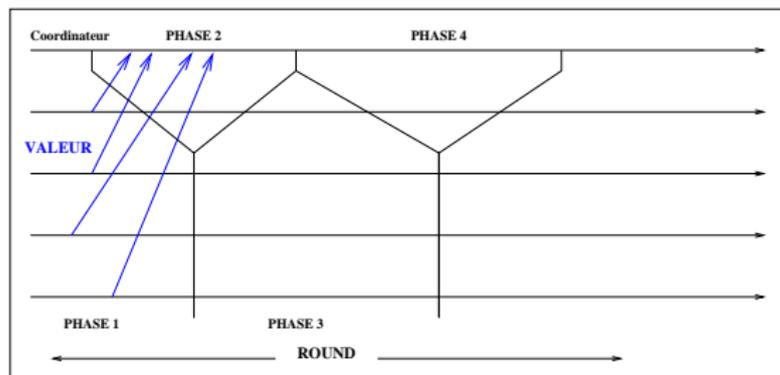
### Algorithme de Chandra & Toueg [Chandra-Toueg 96]

#### Hypothèses

- Communications fiables
- Majorité de processus corrects

#### Principe de fonctionnement

- Rounds asynchrones
- 1 coordinateur par round
- 4 étapes par round



## Algorithme de consensus

### Principe algorithmique

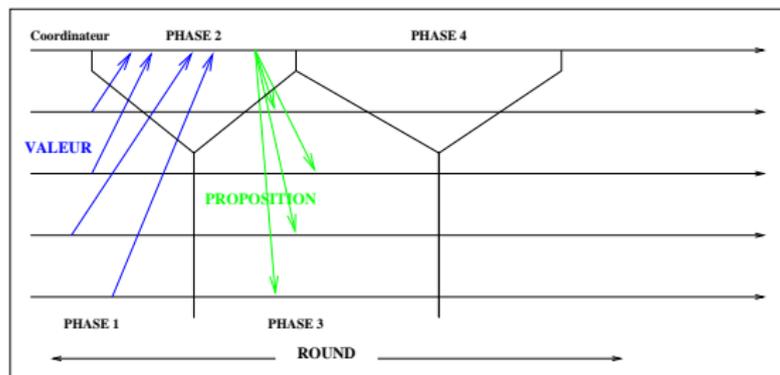
### Algorithme de Chandra & Toueg [Chandra-Toueg 96]

#### Hypothèses

- Communications fiables
- Majorité de processus corrects

#### Principe de fonctionnement

- Rounds asynchrones
- 1 coordinateur par round
- 4 étapes par round



## Algorithme de consensus

### Principe algorithmique

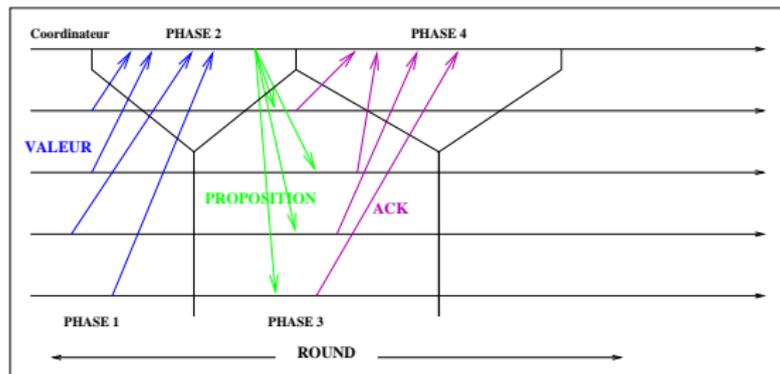
### Algorithme de Chandra & Toueg [Chandra-Toueg 96]

#### Hypothèses

- Communications fiables
- Majorité de processus corrects

#### Principe de fonctionnement

- Rounds asynchrones
- 1 coordinateur par round
- 4 étapes par round



## Algorithme de consensus

### Principe algorithmique

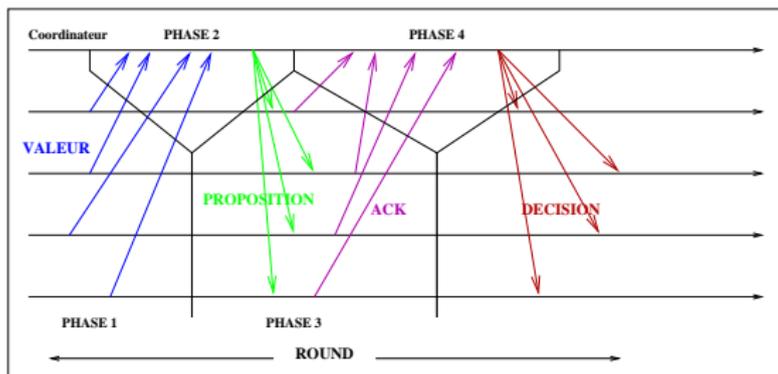
### Algorithme de Chandra & Toueg [Chandra-Toueg 96]

#### Hypothèses

- Communications fiables
- Majorité de processus corrects

#### Principe de fonctionnement

- Rounds asynchrones
- 1 coordinateur par round
- 4 étapes par round



## Adaptation de l'algorithme de consensus

### Fiabilisation des communications

- Module de retransmission périodique [Guerraoui-Oliveira-Schiper 97]
- Choix de la politique de réémission
  - Nombre de réémissions, Période de réémission, Type de messages réémis

### Gestion des entités

- Fluctuation du nombre d'entités présentes
  - ⇒ attentes longues
- Utilisation systématique des détecteurs de défaillances
  - ⇒ force le passage au round suivant en cas d'instabilité
- Interrogations des détecteurs couplées avec le module de retransmission

### Diffusion de la décision

- Protocole de diffusion agressive de la valeur de décision

## Adaptation de l'algorithme de consensus

### Fiabilisation des communications

- Module de retransmission périodique [Guerraoui-Oliveira-Schiper 97]
- Choix de la politique de réémission
  - Nombre de réémissions, Période de réémission, Type de messages réémis

### Gestion des entités

- Fluctuation du nombre d'entités présentes
  - ⇒ attentes longues
- Utilisation systématique des détecteurs de défaillances
  - ⇒ force le passage au round suivant en cas d'instabilité
- Interrogations des détecteurs couplées avec le module de retransmission

### Diffusion de la décision

- Protocole de diffusion agressive de la valeur de décision

## Adaptation de l'algorithme de consensus

### Fiabilisation des communications

- Module de retransmission périodique [Guerraoui-Oliveira-Schiper 97]
- Choix de la politique de réémission
  - Nombre de réémissions, Période de réémission, Type de messages réémis

### Gestion des entités

- Fluctuation du nombre d'entités présentes
  - ⇒ attentes longues
- Utilisation systématique des détecteurs de défaillances
  - ⇒ force le passage au round suivant en cas d'instabilité
- Interrogations des détecteurs couplées avec le module de retransmission

### Diffusion de la décision

- Protocole de diffusion agressive de la valeur de décision

## Adaptation de l'algorithme de consensus

### Fiabilisation des communications

- Module de retransmission périodique [Guerraoui-Oliveira-Schiper 97]
- Choix de la politique de réémission
  - Nombre de réémissions, Période de réémission, Type de messages réémis

### Gestion des entités

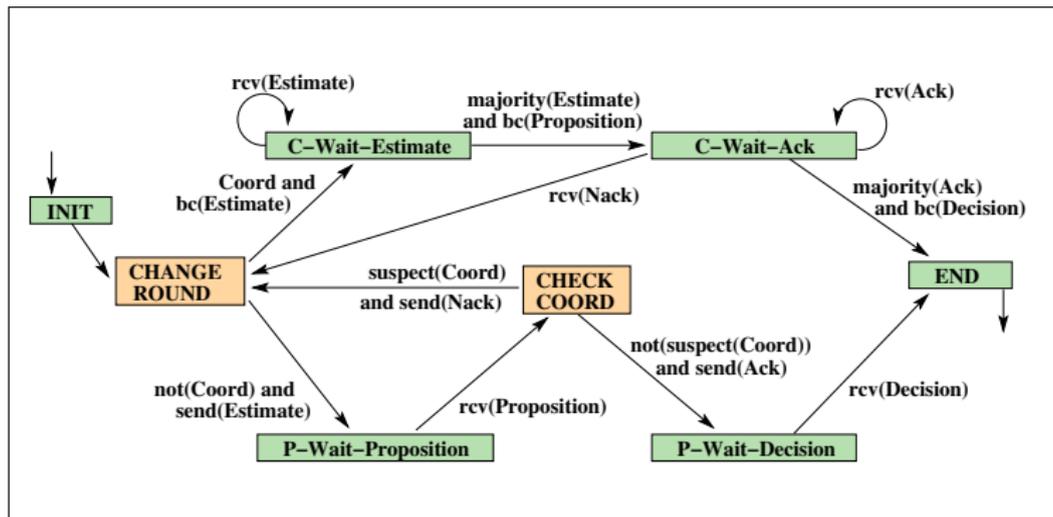
- Fluctuation du nombre d'entités présentes
  - ⇒ attentes longues
- Utilisation systématique des détecteurs de défaillances
  - ⇒ force le passage au round suivant en cas d'instabilité
- Interrogations des détecteurs couplées avec le module de retransmission

### Diffusion de la décision

- Protocole de diffusion agressive de la valeur de décision

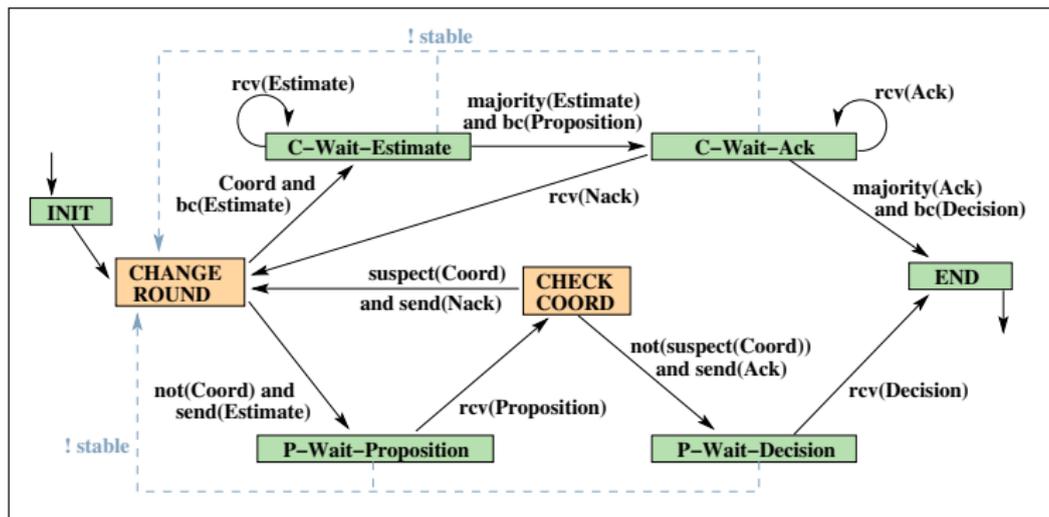
## Algorithme de consensus adapté

## Description de l'algorithme



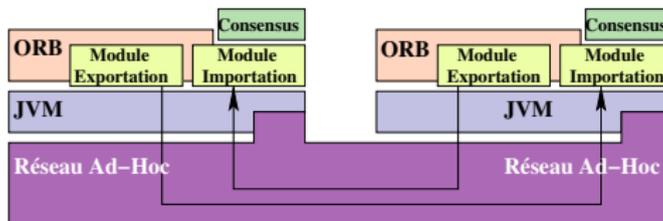
## Algorithme de consensus adapté

## Description de l'algorithme



## Implantation

### Intégration globale



### Plateforme :

Configuration matérielle des machines.

	PDA	Laptop
Architecture	0.2 GHz / 32 MB	1.0 GHz / 256 MB
OS	Linux Familiar (kernel 2.4)	Linux Debian (kernel 2.4)
JVM	Jeode EVM Version 1.10.2	Sun Java JRE 1.4.2.01-b06
Wireless Card	Compaq Orinoco 802.11b	

## Implémentation des détecteurs de défaillances

### Besoin d'informations sur les sites distants

- Exposer les données locales
- Analyser les données en provenance des sites distants

### Exportation d'informations

- Diffusion d'informations à la demande, par anticipation
  - Mécanisme de "heartbeat"

### Importation d'informations

- Collecte d'informations
- Estimation de l'état des sites distants

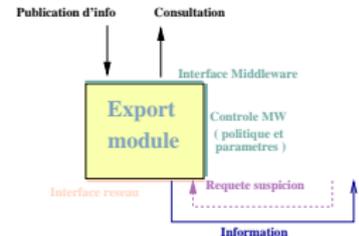
## Implémentation des détecteurs de défaillances

### Besoin d'informations sur les sites distants

- Exposer les données locales
- Analyser les données en provenance des sites distants

### Exportation d'informations

- Diffusion d'informations à la demande, par anticipation
  - Mécanisme de "heartbeat"



### Importation d'informations

- Collecte d'informations
- Estimation de l'état des sites distants

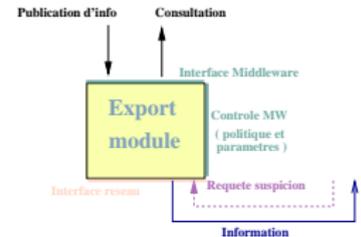
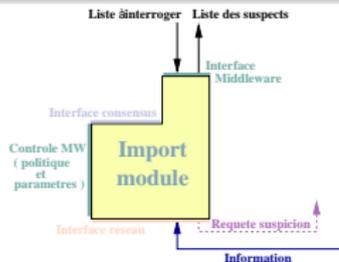
## Implémentation des détecteurs de défaillances

### Besoin d'informations sur les sites distants

- Exposer les données locales
- Analyser les données en provenance des sites distants

### Exportation d'informations

- Diffusion d'informations à la demande, par anticipation
  - Mécanisme de "heartbeat"



### Importation d'informations

- Collecte d'informations
- Estimation de l'état des sites distants

## Bilan

### Réalisations :

- Choix de l'algorithme consensus basé sur les détecteurs de défaillances
- Adaptation à l'instabilité environnementale
- Intégration Middleware

### Qualité du module de consensus

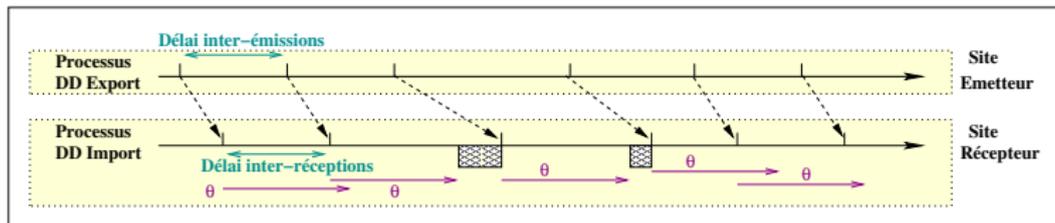
- Durée d'exécution du consensus
- Fonction de la qualité de service des détecteurs de défaillances
  - Capacité à estimer correctement l'état global
    - Fonction de la qualité de l'architecture sous-jacente

# Détecteurs de défaillances et qualité de service

## Paramétrage des détecteurs de défaillances

### Principe de fonctionnement :

Mécanisme implémenté : "heartbeat".



### Paramétrage

- Délais d'émissions des pulsations.
- Fonctions d'estimation de suspicions (temporisations).

[Bertier-Marin-Sens 03]

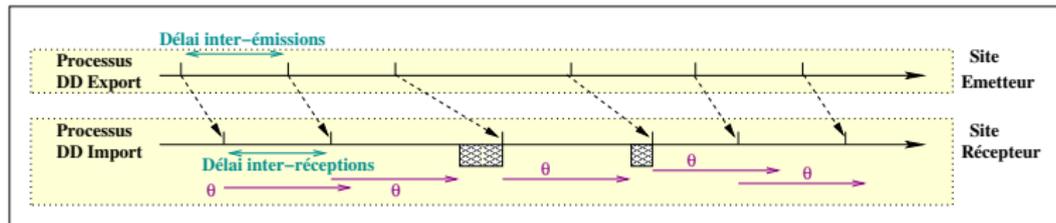
### Objectif

Estimer les valeurs de temporisations en fonction de la qualité de escomptée.

## Paramétrage des détecteurs de défaillances

### Principe de fonctionnement :

Mécanisme implémenté : "heartbeat".



### Paramétrage

- Délais d'émissions des pulsations.
- Fonctions d'estimation de suspicions (temporisations).

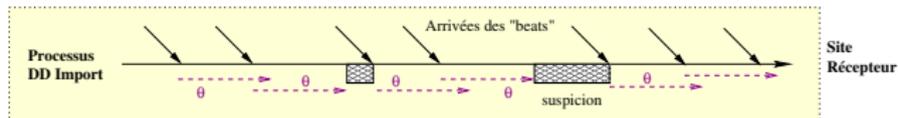
[Bertier-Marin-Sens 03]

### Objectif

Estimer les valeurs de temporisations en fonction de la qualité de escomptée.

## Modélisation des lois d'inter-arrivées des pulsations

**Hypothèses** : Arrivées des pulsations au niveau détecteur de défaillances.



Exemples : Distribution d'Erlang(2)

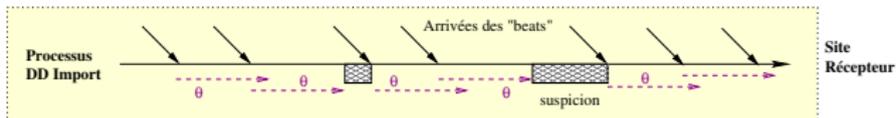
- Temporisation :  $\theta$
- Fréquence de réception des pulsations :  $\lambda$

• Taux de suspicion à tort :

$$\Phi(\theta) = e^{-2\lambda\theta} (1 + \theta)$$

## Modélisation des lois d'inter-arrivées des pulsations

**Hypothèses** : Arrivées des pulsations au niveau détecteur de défaillances.

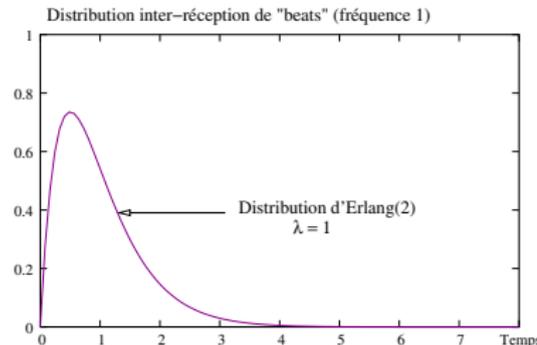


**Exemples** : Distribution d'Erlang(2)

- Temporisation :  $\theta$
- Fréquence de réception des pulsations :  $\lambda$

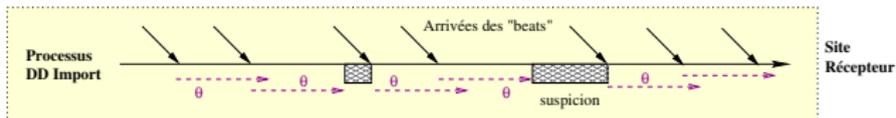
- Taux de suspicion à tort :

$$\Phi(\theta) = e^{-2\lambda\theta}(1 + \theta)$$



## Modélisation des lois d'inter-arrivées des pulsations

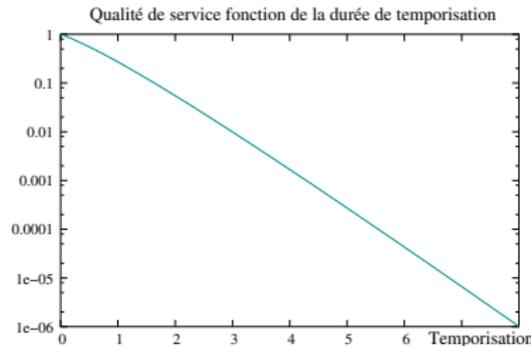
**Hypothèses** : Arrivées des pulsations au niveau détecteur de défaillances.



## Exemples : Distribution d'Erlang(2)

- Temporisation :  $\theta$
- Fréquence de réception des pulsations :  $\lambda$
- Taux de suspicion à tort :

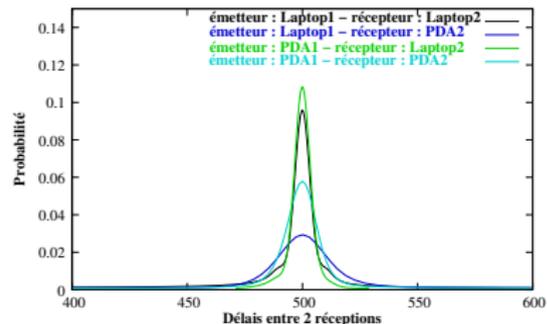
$$\Phi(\theta) = e^{-2\lambda\theta}(1 + \theta)$$



## Modélisation

### Observations expérimentales

Étude du délai de mise à jour de l'information des détecteurs de défaillances.



### Limitations du modèle

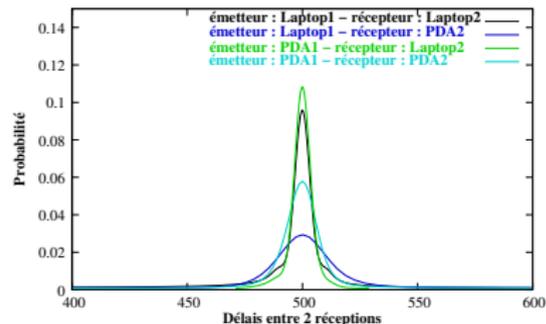
- Indépendance des inter-arrivées de pulsations

⇒ Corrélation temporelle des inter-arrivées

## Modélisation

### Observations expérimentales

Étude du délai de mise à jour de l'information des détecteurs de défaillances.



### Limitations du modèle

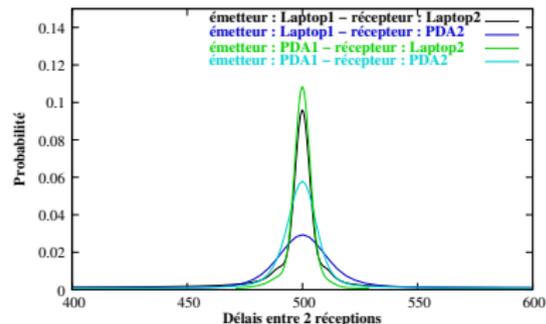
- Indépendance des inter-arrivées de pulsations

⇒ Corrélation temporelle des inter-arrivées

## Modélisation

### Observations expérimentales

Étude du délai de mise à jour de l'information des détecteurs de défaillances.



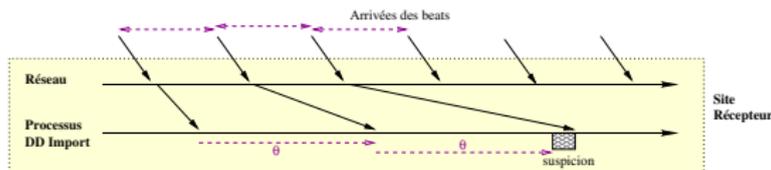
### Limitations du modèle

- Indépendance des inter-arrivées de pulsations

⇒ Corrélation temporelle des inter-arrivées

## Second modèle

**Hypothèse :** Contention à l'arrivée des pulsations.



Temporisation statique ( $\theta$ )

Modélisation par file d'attente

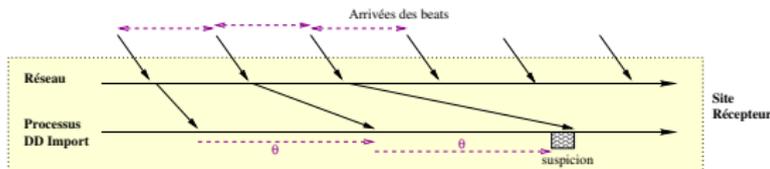
- Taux de suspicion à tort : (cas file d'attente D/M/1)

$$\Phi(\theta) = \frac{1}{A} \frac{1}{(2-\beta)\mu} e^{-\mu\theta} (e^{-\mu(1-\beta)A} + (1-\beta)e^{\mu A}) \quad \theta \geq A$$

- Temps moyen de remontée de la pile ( $\frac{1}{\mu}$ )
- Période entre deux pulsations ( $A$ )
- Probabilité que la pulsation précédente ait été traité à l'arrivée d'une pulsation ( $\beta$ )

## Second modèle

**Hypothèse :** Contention à l'arrivée des pulsations.



Temporisation statique ( $\theta$ )

Modélisation par file d'attente

- Taux de suspicion à tort : (cas file d'attente D/M/1)

$$\Phi(\theta) = \frac{1}{A} \frac{1}{(2-\beta)\mu} e^{-\mu\theta} (e^{-\mu(1-\beta)A} + (1-\beta)e^{\mu A}) \quad \theta \geq A$$

- Temps moyen de remontée de la pile ( $\frac{1}{\mu}$ )
- Période entre deux pulsations ( $A$ )
- Probabilité que la pulsation précédente ait été traité à l'arrivée d'une pulsation ( $\beta$ )

## Bilan

### Calibrage du paramétrage

- Réglage des temporisations en fonction de la qualité de service escomptée.

### Modélisation

- Fonction des lois d'inter-arrivées des "beats"
- Tient compte du type d'entité réceptrice
- Adaptable à diverses configurations environnementales

### Estimation des lois d'inter-arrivées des "beats"

Nécessite la prise en compte des particularités du réseau d'interconnexion.

→ Changements de mode imprévus du système.

## Bilan

### Calibrage du paramétrage

- Réglage des temporisations en fonction de la qualité de service escomptée.

### Modélisation

- Fonction des lois d'inter-arrivées des "beats"
- Tient compte du type d'entité réceptrice
- Adaptable à diverses configurations environnementales

### Estimation des lois d'inter-arrivées des "beats"

Nécessite la prise en compte des particularités du réseau d'interconnexion.

→ Changements de mode imprévus du système.

## Bilan

### Calibrage du paramétrage

- Réglage des temporisations en fonction de la qualité de service escomptée.

### Modélisation

- Fonction des lois d'inter-arrivées des "beats"
- Tient compte du type d'entité réceptrice
- Adaptable à diverses configurations environnementales

### Estimation des lois d'inter-arrivées des "beats"

Nécessite la prise en compte des particularités du réseau d'interconnexion.

→ Changements de mode imprévus du système.

# Analyse et évaluation du module de consensus

## Plateforme expérimentale

### Contexte

- réseau sous-jacent : 802.11b
- 6 entités présentes (4 PDAs, 2 portables)
- 1 entité = capteur réseau (1 PC)

### Paramétrage

- Période des heartbeats : 500 ms (fréquences : 2 pulsations / s)
- Temporisation : 1500 ms
- "Consensus Polling" : 2000 ms (interrogation DD + retransmission)

### Expériences

- 1000 consensus successifs

→ Observation de l'exécution de l'algorithme de consensus

## Plateforme expérimentale

### Contexte

- réseau sous-jacent : 802.11b
- 6 entités présentes (4 PDAs, 2 portables)
- 1 entité = capteur réseau (1 PC)

### Paramétrage

- Période des heartbeats : 500 ms (fréquences : 2 pulsations / s)
- Temporisation : 1500 ms
- "Consensus Polling" : 2000 ms (interrogation DD + retransmission)

### Expériences

- 1000 consensus réussis

→ Observation de l'exécution de l'algorithme de consensus

## Plateforme expérimentale

### Contexte

- réseau sous-jacent : 802.11b
- 6 entités présentes (4 PDAs, 2 portables)
- 1 entité = capteur réseau (1 PC)

### Paramétrage

- Période des heartbeats : 500 ms (fréquences : 2 pulsations / s)
- Temporisation : 1500 ms
- "Consensus Polling" : 2000 ms (interrogation DD + retransmission)

### Expériences

- 1000 consensus réussis

→ Observation de l'exécution de l'algorithme de consensus

## Plateforme expérimentale

### Contexte

- réseau sous-jacent : 802.11b
- 6 entités présentes (4 PDAs, 2 portables)
- 1 entité = capteur réseau (1 PC)

### Paramétrage

- Période des heartbeats : 500 ms (fréquences : 2 pulsations / s)
- Temporisation : 1500 ms
- "Consensus Polling" : 2000 ms (interrogation DD + retransmission)

### Expériences

- 1000 consensus réussis

→ Observation de l'exécution de l'algorithme de consensus

## Prise de mesure et Datation

### Phase d'instrumentation

- Enregistrements des événements estampillés à la date locale
- Datation globale ?

### Extrapolation d'une horloge globale (post-traitement des traces)

- Relation d'ordre causal :  $e_i \prec e_j$ 
  - $e_i$  avant  $e_j$  la même entité
  - $e_i$  = émission et  $e_j$  = réception d'un même message
  - $\exists k$  tel que  $e_i \prec e_k$  et  $e_k \prec e_j$
- Approximation dans un référentiel commun :  $d_{ref}^{e_i} = \alpha d_{loc}^{e_i} + \beta$

### Interprétation

- Représentation graphique (Pajé)
- Analyse statistique

## Prise de mesure et Datation

### Phase d'instrumentation

- Enregistrements des événements estampillés à la date locale
- Datation globale ?

### Extrapolation d'une horloge globale (post-traitement des traces)

- Relation d'ordre causal :  $e_i \prec e_j$ 
  - $e_i$  avant  $e_j$  la même entité
  - $e_i$  = émission et  $e_j$  = réception d'un même message
  - $\exists k$  tel que  $e_i \prec e_k$  et  $e_k \prec e_j$
- Approximation dans un référentiel commun :  $d_{ref}^{e_i} = \alpha d_{loc}^{e_i} + \beta$

### Interprétation

- Représentation graphique (Pajé)
- Analyse statistique

## Prise de mesure et Datation

### Phase d'instrumentation

- Enregistrements des événements estampillés à la date locale
- Datation globale ?

### Extrapolation d'une horloge globale (post-traitement des traces)

- Relation d'ordre causal :  $e_i \prec e_j$   
 $e_i$  avant  $e_j$  la même entité  
 $e_i =$  émission et  $e_j =$  réception d'un même message  
 $\exists k$  tel que  $e_i \prec e_k$  et  $e_k \prec e_j$
- Approximation dans un référentiel commun :  $d_{ref}^{e_i} = \alpha d_{loc}^{e_i} + \beta$

### Interprétation

- Représentation graphique (Pajé)
- Analyse statistique

## Prise de mesure et Datation

### Phase d'instrumentation

- Enregistrements des événements estampillés à la date locale
- Datation globale ?

### Extrapolation d'une horloge globale (post-traitement des traces)

- Relation d'ordre causal :  $e_i \prec e_j$   
 $e_i$  avant  $e_j$  la même entité  
 $e_i$  = émission et  $e_j$  = réception d'un même message  
 $\exists k$  tel que  $e_i \prec e_k$  et  $e_k \prec e_j$
- Approximation dans un référentiel commun :  $d_{ref}^{e_i} = \alpha d_{loc}^{e_i} + \beta$

### Interprétation

- Représentation graphique (Pajé)
- Analyse statistique

## Prise de mesure et Datation

### Phase d'instrumentation

- Enregistrements des événements estampillés à la date locale
- Datation globale ?

### Extrapolation d'une horloge globale (post-traitement des traces)

- Relation d'ordre causal :  $e_i \prec e_j$   
 $e_i$  avant  $e_j$  la même entité  
 $e_i =$  émission et  $e_j =$  réception d'un même message  
 $\exists k$  tel que  $e_i \prec e_k$  et  $e_k \prec e_j$
- Approximation dans un référentiel commun :  $d_{ref}^{e_i} = \alpha d_{loc}^{e_i} + \beta$

### Interprétation

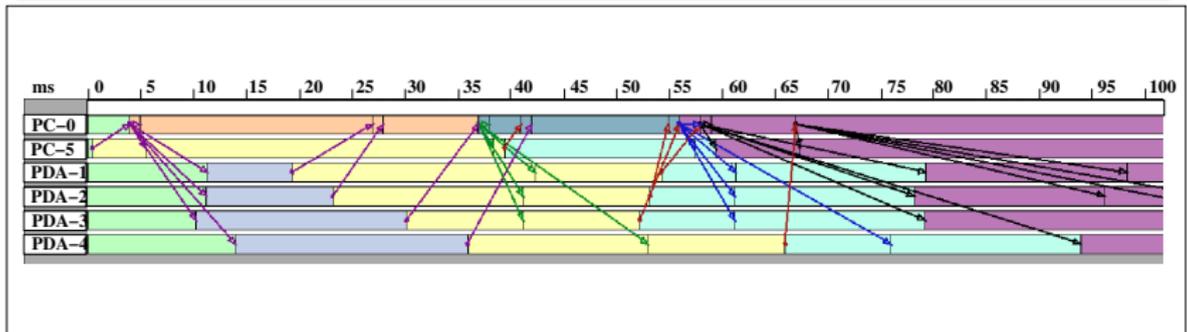
- Représentation graphique (Pajé)
- Analyse statistique

## Traces d'exécution

## Diagramme post-mortem d'un consensus ne donnant lieu à aucune suspicion

Initialisation : PC-5

1 seul round : coordinateur PC-0

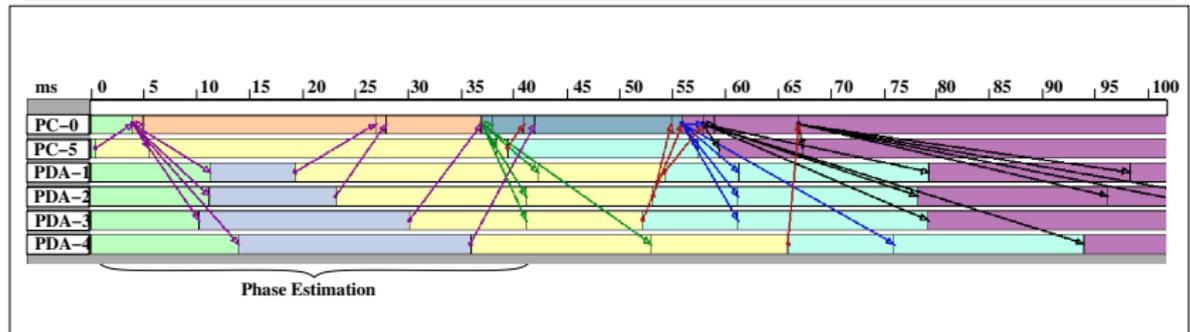
Retard de l'Ack de PDA-4  $\Rightarrow$  message Forward

## Traces d'exécution

## Diagramme post-mortem d'un consensus ne donnant lieu à aucune suspicion

Initialisation : PC-5

1 seul round : coordinateur PC-0

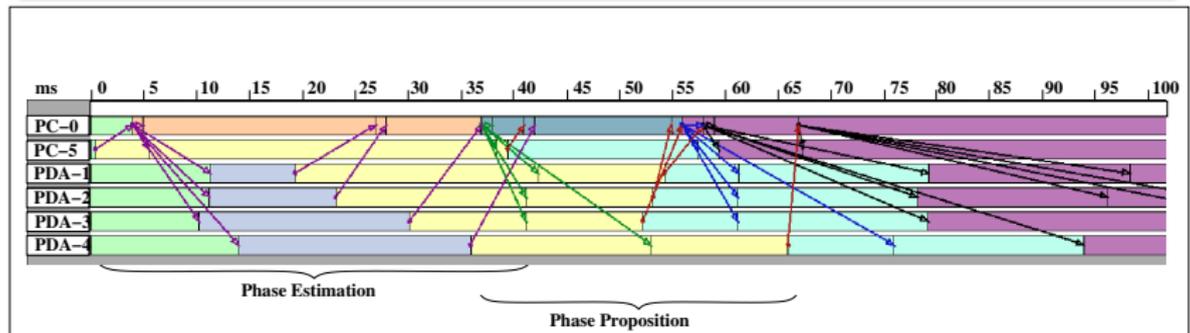
Retard de l'Ack de PDA-4  $\Rightarrow$  message Forward

## Traces d'exécution

## Diagramme post-mortem d'un consensus ne donnant lieu à aucune suspicion

Initialisation : PC-5

1 seul round : coordinateur PC-0

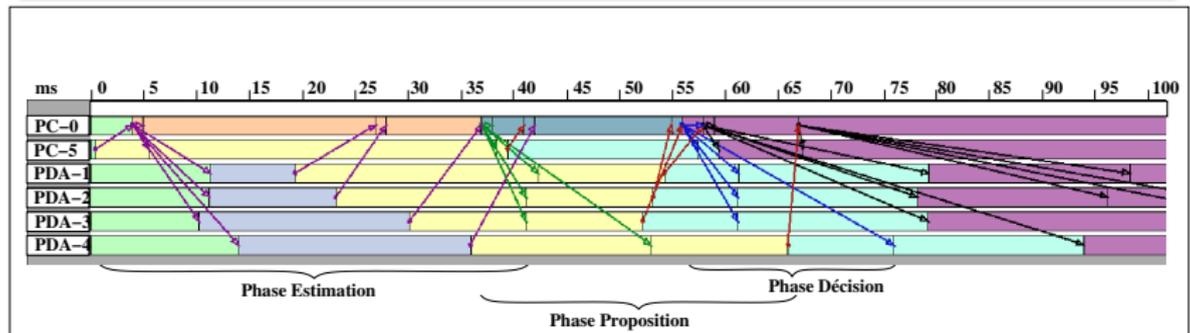
Retard de l'Ack de PDA-4  $\Rightarrow$  message Forward

## Traces d'exécution

## Diagramme post-mortem d'un consensus ne donnant lieu à aucune suspicion

Initialisation : PC-5

1 seul round : coordinateur PC-0

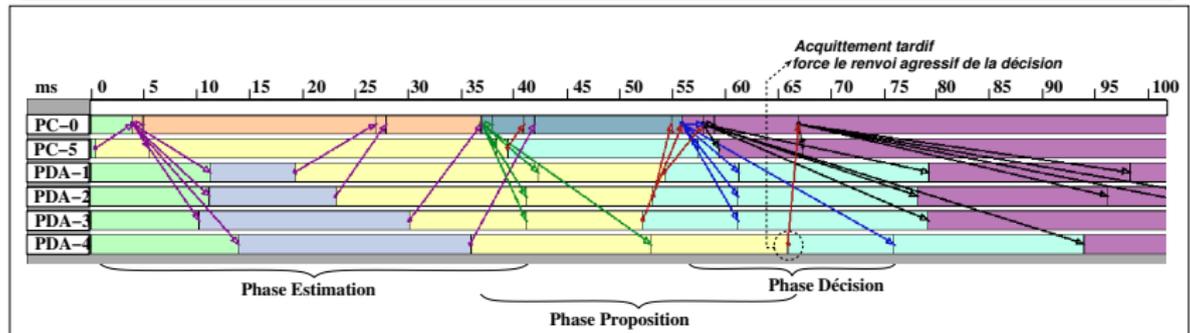
Retard de l'Ack de PDA-4  $\Rightarrow$  message Forward

## Traces d'exécution

## Diagramme post-mortem d'un consensus ne donnant lieu à aucune suspicion

Initialisation : PC-5

1 seul round : coordinateur PC-0

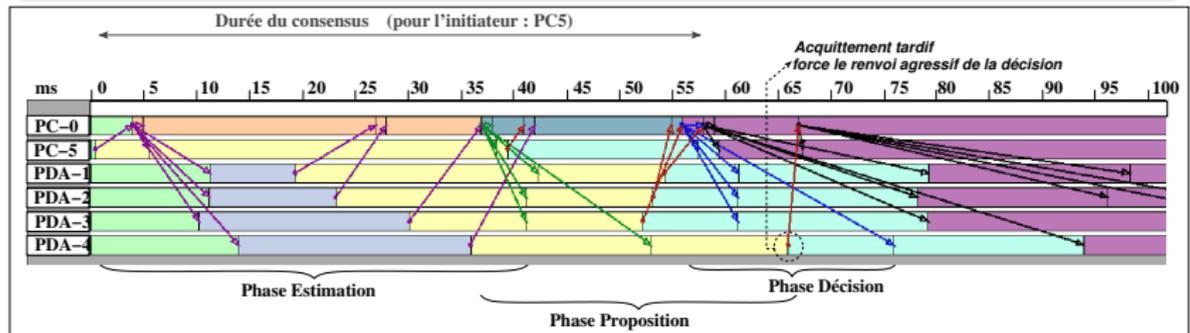
Retard de l'Ack de PDA-4  $\Rightarrow$  message Forward

## Traces d'exécution

## Diagramme post-mortem d'un consensus ne donnant lieu à aucune suspicion

Initialisation : PC-5

1 seul round : coordinateur PC-0

Retard de l'Ack de PDA-4  $\Rightarrow$  message Forward

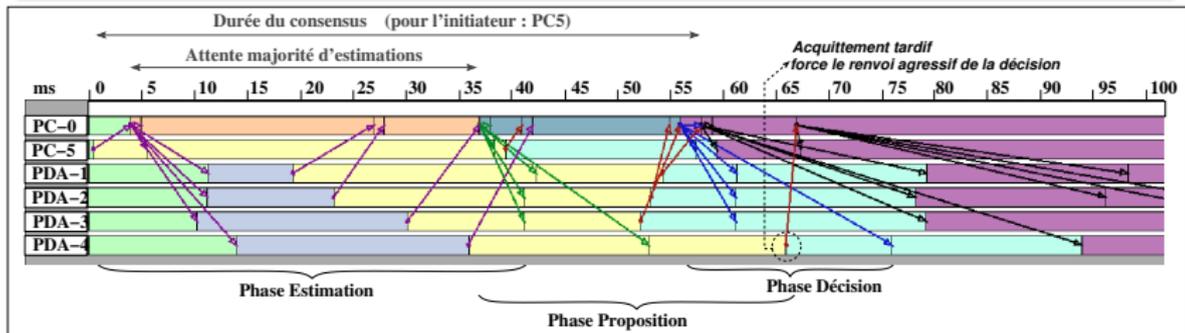
## Traces d'exécution

## Diagramme post-mortem d'un consensus ne donnant lieu à aucune suspicion

Initialisation : PC-5

1 seul round : coordinateur PC-0

Retard de l'Ack de PDA-4 ⇒ message Forward



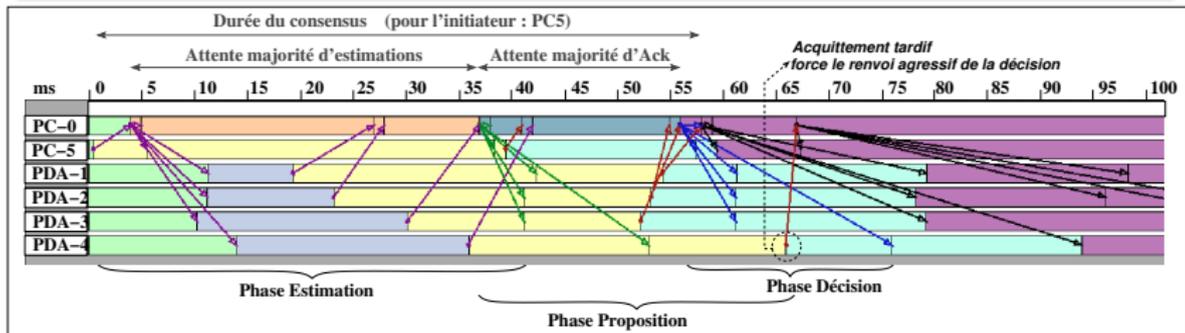
## Traces d'exécution

## Diagramme post-mortem d'un consensus ne donnant lieu à aucune suspicion

Initialisation : PC-5

1 seul round : coordinateur PC-0

Retard de l'Ack de PDA-4 ⇒ message Forward



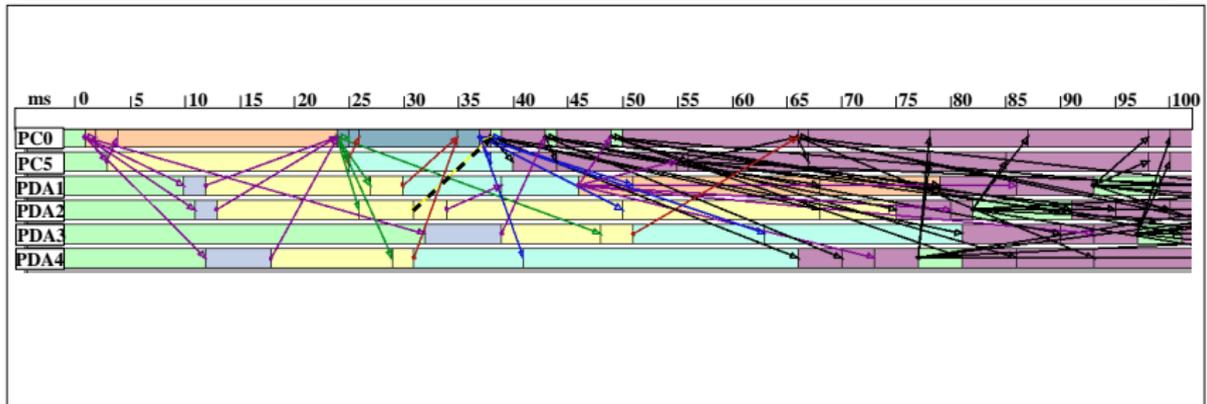
## Traces d'exécution

## Diagramme post-mortem d'un consensus illustrant une fausse suspicion

Décision au round 1

Message Nack émis pas PDA-2 parvient au coordinateur après décision

PDA-1 coordinateur du round 2



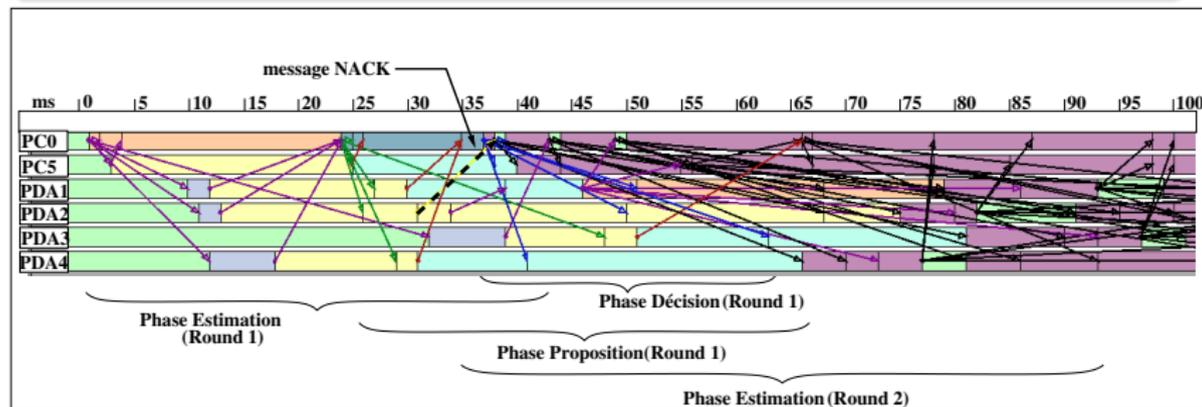
## Traces d'exécution

## Diagramme post-mortem d'un consensus illustrant une fausse suspicion

Décision au round 1

Message Nack émis par PDA-2 parvient au coordinateur après décision

PDA-1 coordinateur du round 2



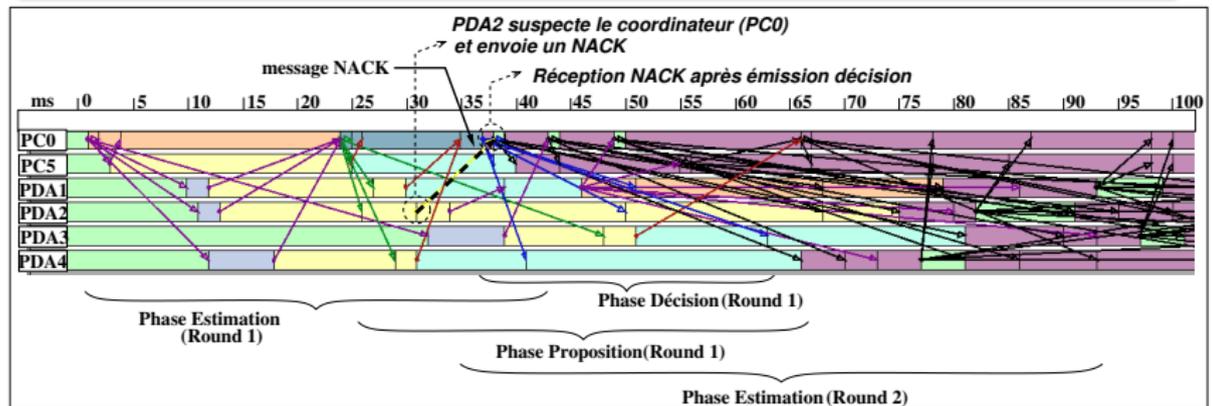
## Traces d'exécution

## Diagramme post-mortem d'un consensus illustrant une fausse suspicion

Décision au round 1

Message Nack émis pas PDA-2 parvient au coordinateur après décision

PDA-1 coordinateur du round 2



## Évaluations de performances

QoS au niveau système (utilisation des ressources)

QoS au niveau applicatif (latences)

## Évaluations de performances

### QoS au niveau système (utilisation des ressources)

- Nombre moyen de messages par entité
- Nombre moyen de messages émis suivant leur type

<i>ExpI</i>	Broadcast	Unicast	Total
<b>PC-0</b>	6.4	0	6.4
<b>PDA-1</b>	0	2	2
<b>PDA-2</b>	0	2	2
<b>PDA-3</b>	0	2	2
<b>PDA-4</b>	0	2	2
<b>PC-5</b>	0	2	2
Total	6.4	10	16.4

### QoS au niveau applicatif (latences)

## Évaluations de performances

### QoS au niveau système (utilisation des ressources)

- Nombre moyen de messages par entité
- Nombre moyen de messages émis suivant leur type

### QoS au niveau applicatif (latences)

- Distribution des différentes durées d'un consensus (ms)

<i>Exp 1</i>	initiateur	tous
$\bar{x}$	71	224
min	25	37
max	2063	2070
$\sigma$	190	231
$Q_1$	39	136
$Q_2$	45	199
$Q_3$	52	254

## Évaluations de performances

### QoS au niveau système (utilisation des ressources)

- Nombre moyen de messages par entité
- Nombre moyen de messages émis suivant leur type

### QoS au niveau applicatif (latences)

- Distribution des différentes durées d'un consensus (ms)
- Distribution du temps d'attente d'une majorité de réponses pour le coordinateur (ms)

<i>Exp 1</i>	Estimations	Acquittements
$\bar{x}$	37	26
min	1	7
max	2021	161
$\sigma$	143	20
$Q_1$	18	17
$Q_2$	21	21
$Q_3$	27	26

## Bilan

### Validation expérimentale du module de consensus

- Génération de traces d'exécution
- Analyse comportementale
- Évaluations temporelles

### Optimisations

- Module de consensus
  - Limiter les phases d'attente
  - Restreindre la diffusion de la décision
- Lien avec la qualité des détecteurs de défaillances

# Conclusions et Perspectives

## Conclusions

### Méthodologie d'analyse comportementale

- Caractérisation des éléments perturbateurs
- Analyse qualitative et quantitative

### Méthodes d'adaptation d'algorithmes

- Analyse des algorithmes existants
- Définition des propriétés attendues
- Modélisation (Détermination d'un paramétrage adéquat)

### Environnement d'expérimentation

- Validation expérimentale des algorithmes
- Evaluations de performances

## Conclusions

### Méthodologie d'analyse comportementale

- Caractérisation des éléments perturbateurs
- Analyse qualitative et quantitative

### Méthodes d'adaptation d'algorithmes

- Analyse des algorithmes existants
- Définition des propriétés attendues
- Modélisation (Détermination d'un paramétrage adéquat)

### Environnement d'expérimentation

- Validation expérimentale des algorithmes
- Evaluations de performances

## Conclusions

### Méthodologie d'analyse comportementale

- Caractérisation des éléments perturbateurs
- Analyse qualitative et quantitative

### Méthodes d'adaptation d'algorithmes

- Analyse des algorithmes existants
- Définition des propriétés attendues
- Modélisation (Détermination d'un paramétrage adéquat)

### Environnement d'expérimentation

- Validation expérimentale des algorithmes
- Evaluations de performances

## Perspectives

### Analyse environnementale

- Étude complémentaire
  - Approfondissement de l'analyse factorielle
  - Enrichissement de l'étude comportementale
- Adaptation de la méthodologie
  - Autres environnements (réseaux Bluetooth, réseaux de capteurs, ...)

### Algorithmique répartie

- Optimisation des détecteurs de défaillances
- Estimation multi-niveaux de l'état global
- Étude d'autres approches algorithmiques

### Mise en œuvre

- Mobilité (nomadisme)
- Mécanismes de routage
- Étude du passage à l'échelle

## Perspectives

### Analyse environnementale

- Étude complémentaire
  - Approfondissement de l'analyse factorielle
  - Enrichissement de l'étude comportementale
- Adaptation de la méthodologie
  - Autres environnements (réseaux Bluetooth, réseaux de capteurs, ...)

### Algorithmique répartie

- Optimisation des détecteurs de défaillances
- Estimation multi-niveaux de l'état global
- Étude d'autres approches algorithmiques

### Mise en œuvre

- Mobilité (nomadisme)
- Mécanismes de routage
- Étude du passage à l'échelle

## Perspectives

### Analyse environnementale

- Étude complémentaire
  - Approfondissement de l'analyse factorielle
  - Enrichissement de l'étude comportementale
- Adaptation de la méthodologie
  - Autres environnements (réseaux Bluetooth, réseaux de capteurs, ...)

### Algorithmique répartie

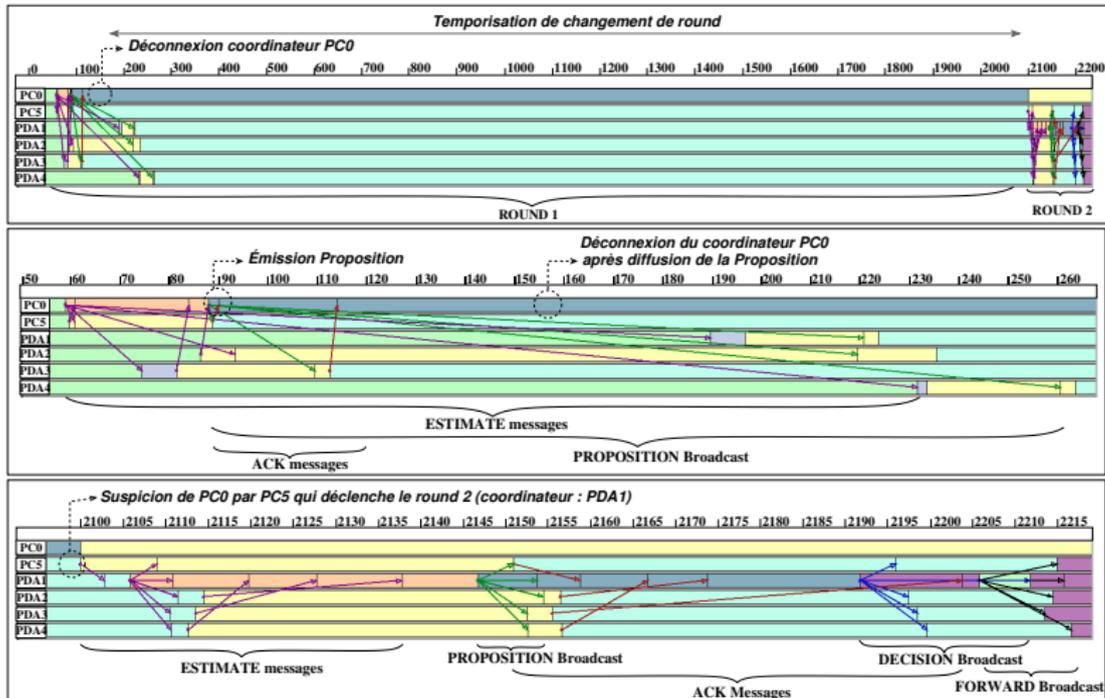
- Optimisation des détecteurs de défaillances
- Estimation multi-niveaux de l'état global
- Étude d'autres approches algorithmiques

### Mise en œuvre

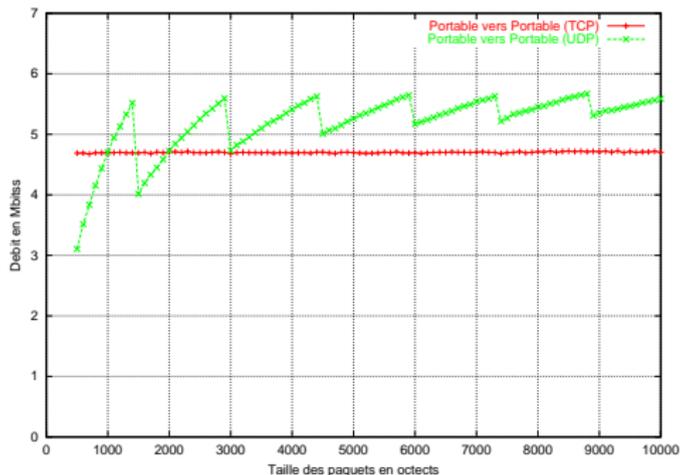
- Mobilité (nomadisme)
- Mécanismes de routage
- Étude du passage à l'échelle



## Impact d'une déconnexion

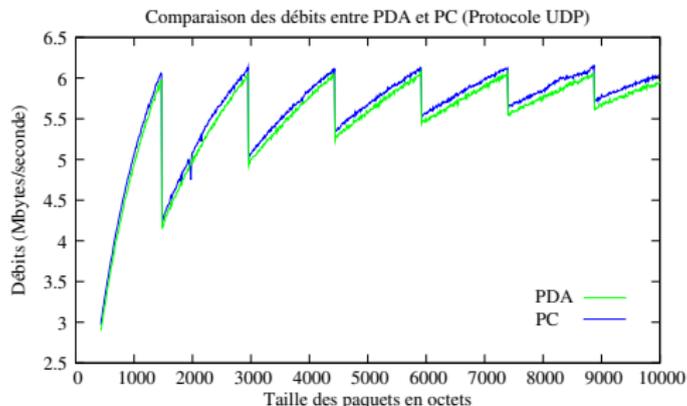


## Débits moyen pour les ordinateurs portables.



- Pour la courbe en mode UDP : des décrochages sont visibles tous les 1470 octets dûs au MTU (Maximum Transmission Unit).
- Le coût de l'utilisation de TCP est bien mis en évidence. En effet, le débit chute de 0.5 Mbits/s par rapport à UDP.

## Débits observés avec le protocole UDP pour un PDA (iPAQ) et un PC portable



- La courbe de débit pour le PDA moins régulière et parsemée de décrochages.
- plus faible robustesse de l'architecture des PDAs.
- Les différences observées entre PDA et PC portable représentent, un écart variant autour de 0.10Mb/s pour le protocole UDP et autour de 0.8Mb/sec pour le protocole TCP.