

# Réseaux et Systèmes (MIN 15112)

2025 - 2026

Zhiyi Zhang

[zhiyi.zhang@uvsq.fr](mailto:zhiyi.zhang@uvsq.fr)

<https://www.zhiyizhang.com>

# Contrôle d'accès

# Contrôle d'accès

- Partage du support & multiplexage
- Contrôle d'accès
- Aloha (Pure/Slotted)
- CSMA
- CSMA/CD

# Modes de communication

## Classification de modes de communication

Le support physique (canal de communication) entre la source A et la destination B :

- Unidirectionnel (simplex) : communication dans un seul sens, de A vers B uniquement, *e.g.*, radio, télévision
- Bidirectionnel (half-duplex) : communication dans les deux sens, mais alternativement, *e.g.*, talkie-walkie, câble partagé
- Bidirectionnel (full-duplex) : communication dans les deux sens en simultané, *e.g.*, téléphone

# Partage du support ?

## Réseau point à point (P2P)

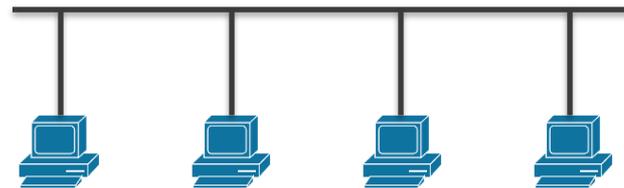
- Connexion directe entre deux équipements
- Le canal de transmission est réservé à ces deux équipements



# Partage du support ?

## Réseau à accès multiple

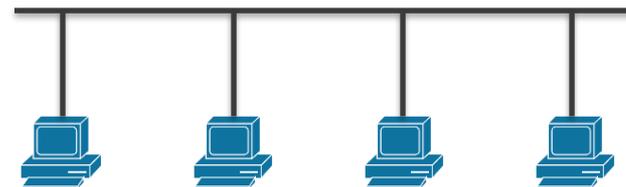
- Plusieurs équipements partagent le même canal de transmission
- Risque de collision ou d'interférences
- Contrôle d'accès nécessaire



# Partage du support ?

## Réseau à accès multiple

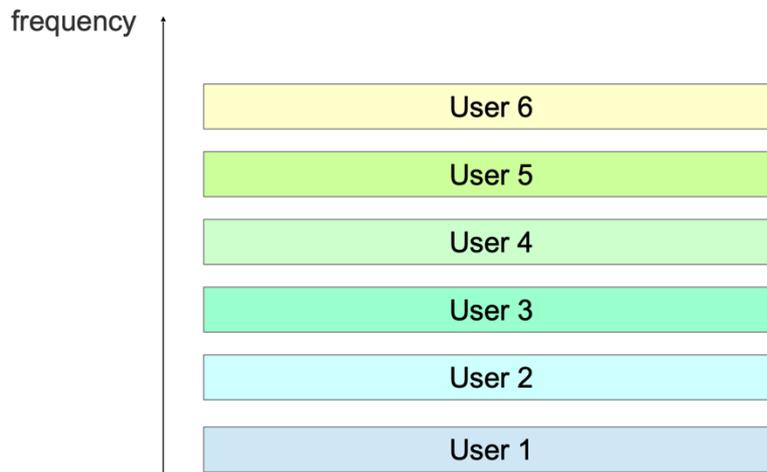
- Plusieurs équipements partagent le même canal de transmission
- Risque de collision ou d'interférences
- Contrôle d'accès nécessaire



**Comment partager le support ?**

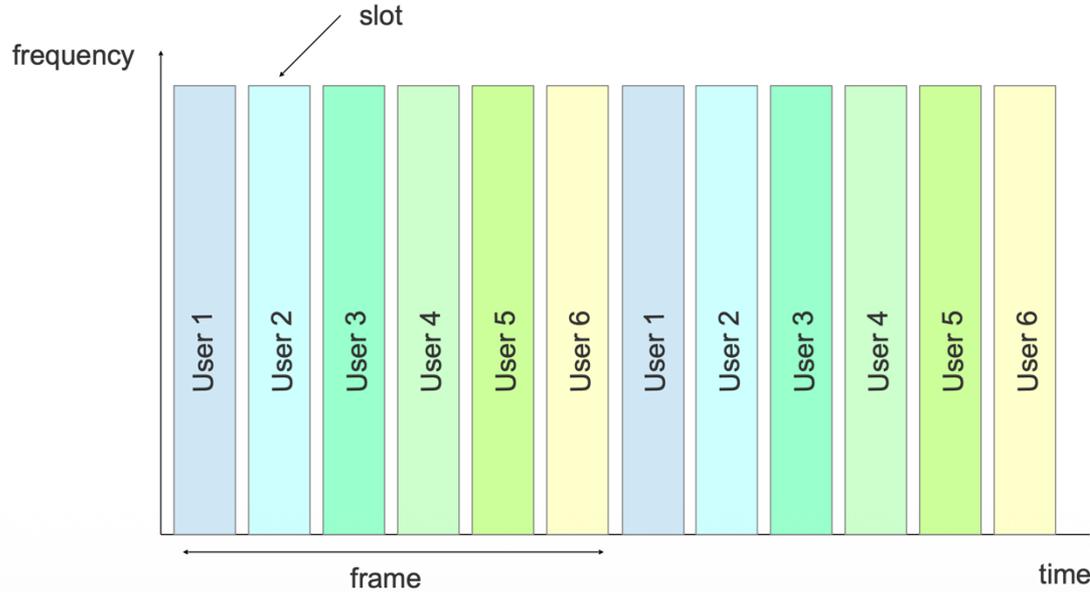
# Multiplexage

## FDMA – Frequency Division Multiple Access



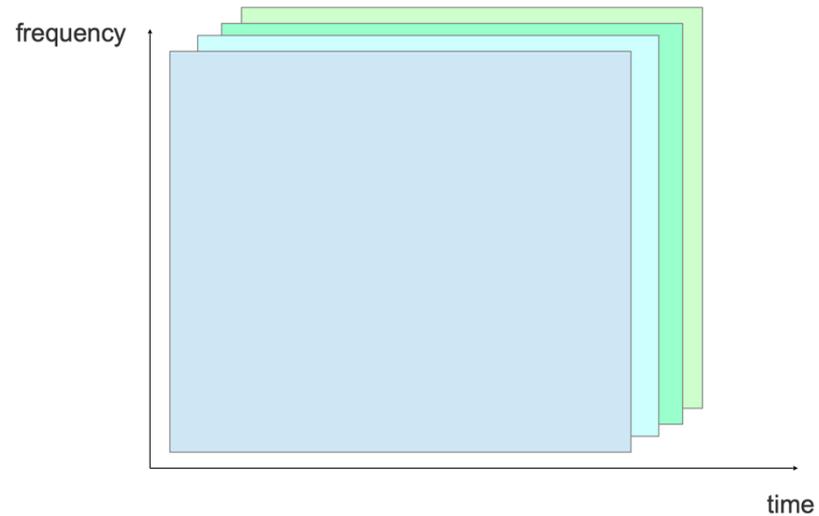
# Multiplexage

## TDMA – Time Division Multiple Access



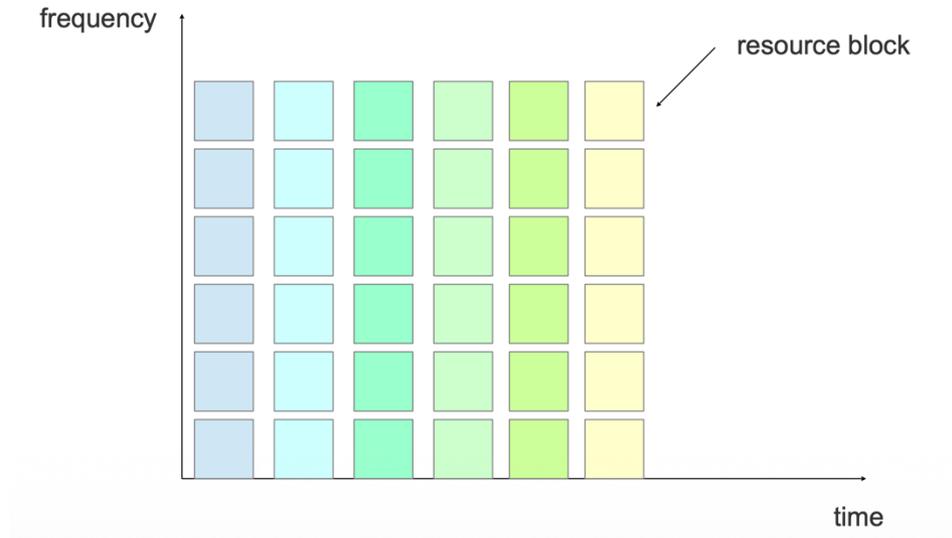
# Multiplexage

## CDMA – Code Division Multiple Access



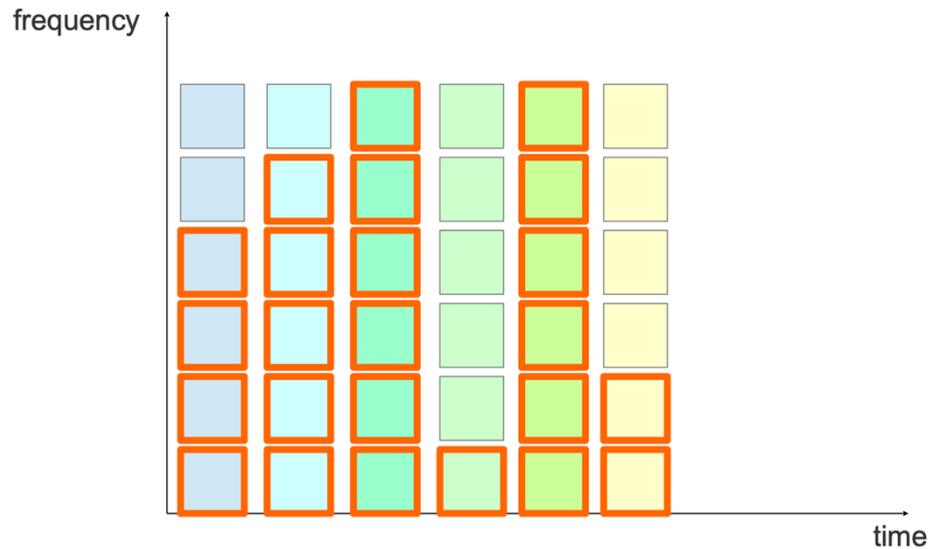
# Multiplexage

Multiplexage en fréquence et temporel (e.g. OFDM (Orthogonal Frequency-Division Multiplexing))



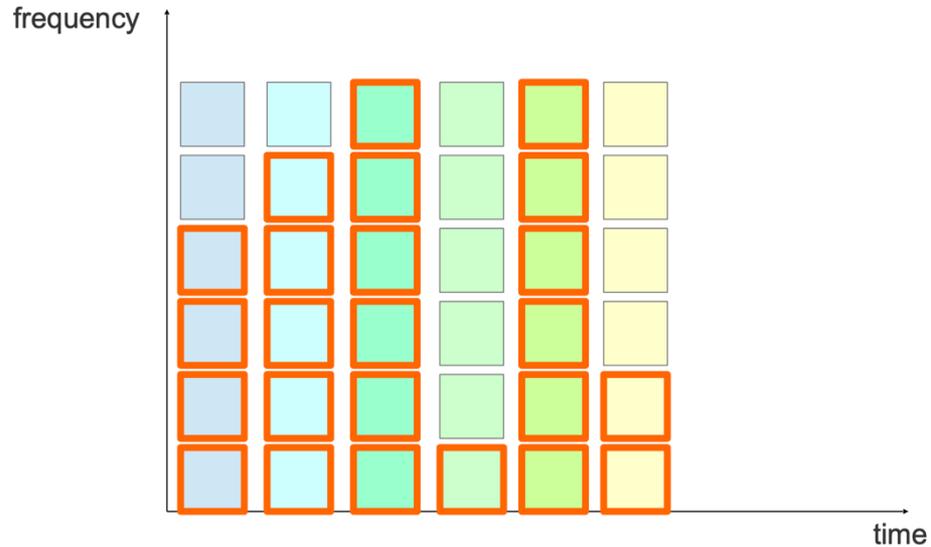
# Allocation statique

Problème ?



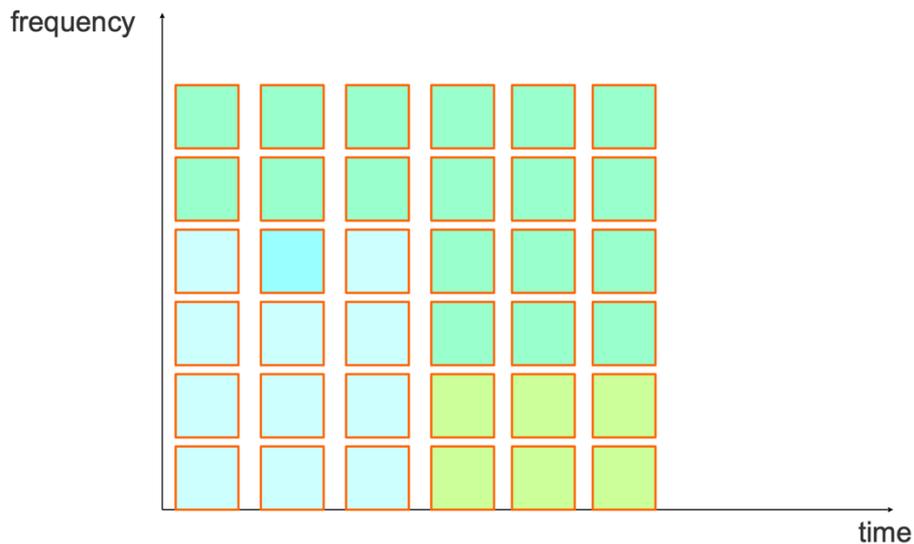
# Allocation statique

Problème : sous-utilisation des ressources



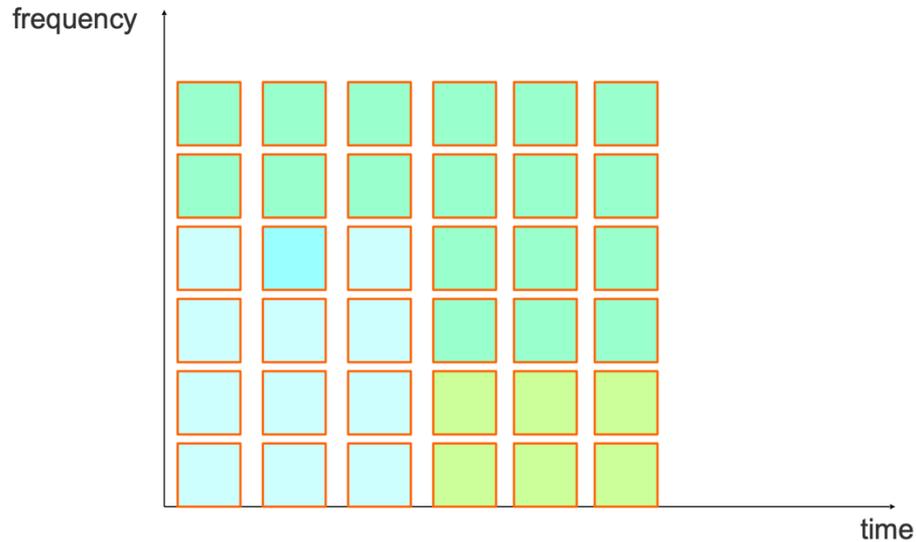
# Solutions dynamiques

Les ressources sont attribuées aux utilisateurs en cas de besoin



# Solutions dynamiques

Les ressources sont attribuées aux utilisateurs en cas de besoin



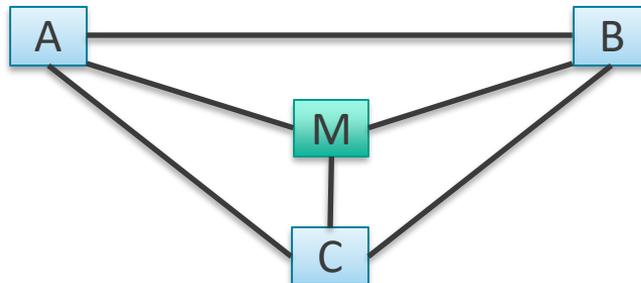
Mais...  
Comment ?

# Protocoles centralisés

- Un coordinateur attribue l'accès au canal aux utilisateurs
- Nécessite un trafic de contrôle supplémentaire
- Performance proche de l'optimal
- Mais, point unique de défaillance

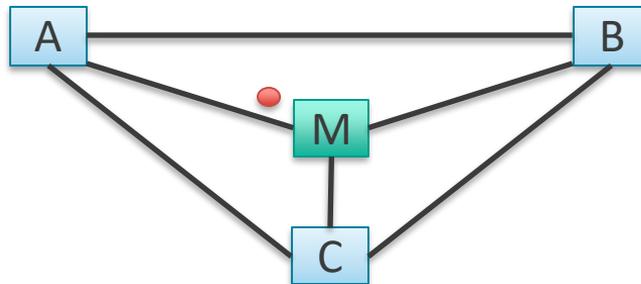
# Protocoles centralisés

- Un coordinateur attribue l'accès au canal aux utilisateurs
- Nécessite un trafic de contrôle supplémentaire
- Performance proche de l'optimal
- Mais, point unique de défaillance



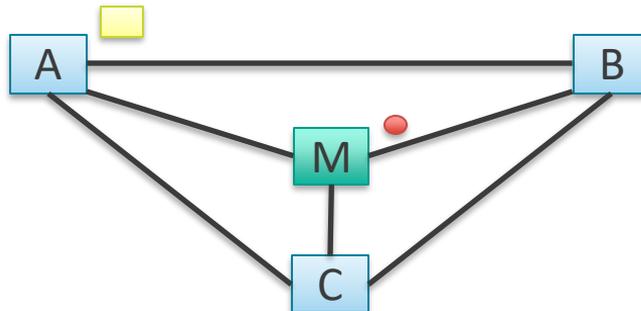
# Protocoles centralisés

- Un coordinateur attribue l'accès au canal aux utilisateurs
- Nécessite un trafic de contrôle supplémentaire
- Performance proche de l'optimal
- Mais, point unique de défaillance



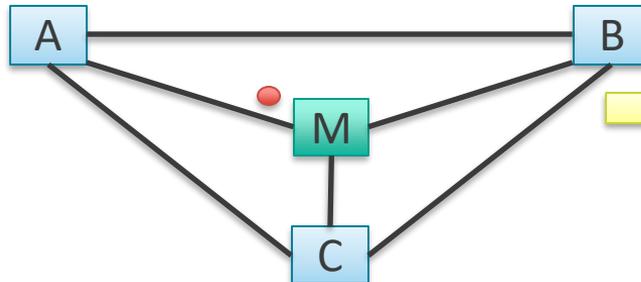
# Protocoles centralisés

- Un coordinateur attribue l'accès au canal aux utilisateurs
- Nécessite un trafic de contrôle supplémentaire
- Performance proche de l'optimal
- Mais, point unique de défaillance



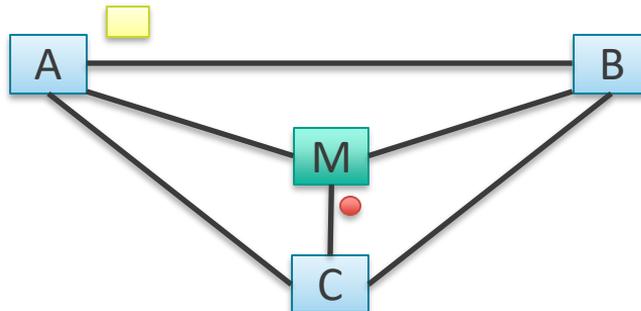
# Protocoles centralisés

- Un coordinateur attribue l'accès au canal aux utilisateurs
- Nécessite un trafic de contrôle supplémentaire
- Performance proche de l'optimal
- Mais, point unique de défaillance



# Protocoles centralisés

- Un coordinateur attribue l'accès au canal aux utilisateurs
- Nécessite un trafic de contrôle supplémentaire
- Performance proche de l'optimal
- Mais, point unique de défaillance

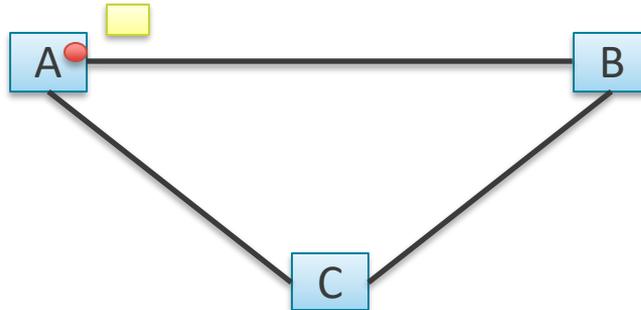


# Protocoles distribués

- Toutes les stations jouent un rôle similaire
- Un consensus est nécessaire
- Une synchronisation est nécessaire
- La robustesse du réseau est améliorée

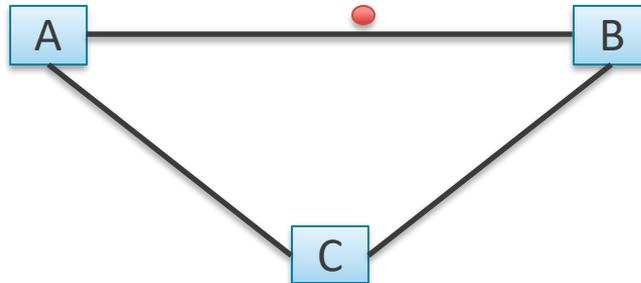
# Protocoles déterministes

- L'accès au support dans un temps fini garanti
- Basés sur un mécanisme de partage de jeton
- La station qui possède le jeton a le droit de transmettre



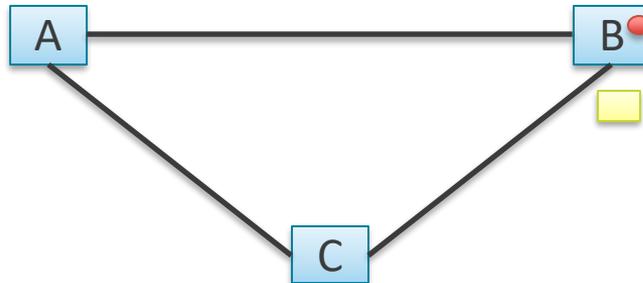
# Protocoles déterministes

- L'accès au support dans un temps fini garanti
- Basés sur un mécanisme de partage de jeton
- La station qui possède le jeton a le droit de transmettre



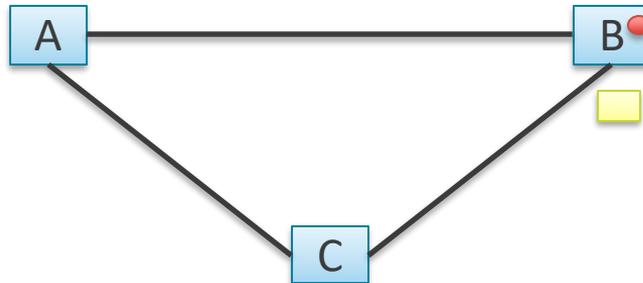
# Protocoles déterministes

- L'accès au support dans un temps fini garanti
- Basés sur un mécanisme de partage de jeton
- La station qui possède le jeton a le droit de transmettre



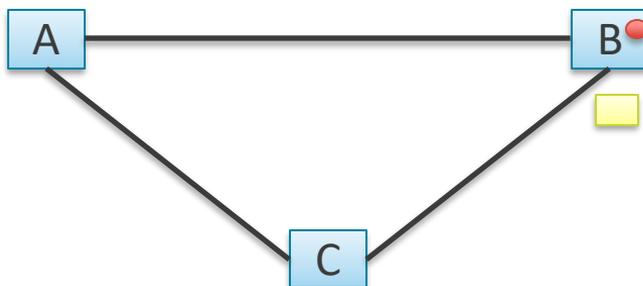
# Protocoles déterministes

- L'accès au support dans un temps fini garanti
- Basés sur un mécanisme de partage de jeton
- La station qui possède le jeton a le droit de transmettre
- Mais, sous-utilisation des ressources



# Protocoles déterministes

- L'accès au support dans un temps fini garanti
- Basés sur un mécanisme de partage de jeton
- La station qui possède le jeton a le droit de transmettre
- Mais, sous-utilisation des ressources



Exemples :

Token Bus (IEEE 802.4)

Token Ring (IEEE 802.5)

# Mécanismes aléatoires

- Bonne performance en moyenne
- Mais, aucune garantie concernant le temps d'accès au canal
- Des collisions sont possibles entre les émetteurs

# Protocoles MAC stochastique

## Aloha

- Années 60 - 70
- Le premier protocole de contrôle d'accès aléatoire
- Toutes les stations sont autorisées à émettre un paquet à tout moment

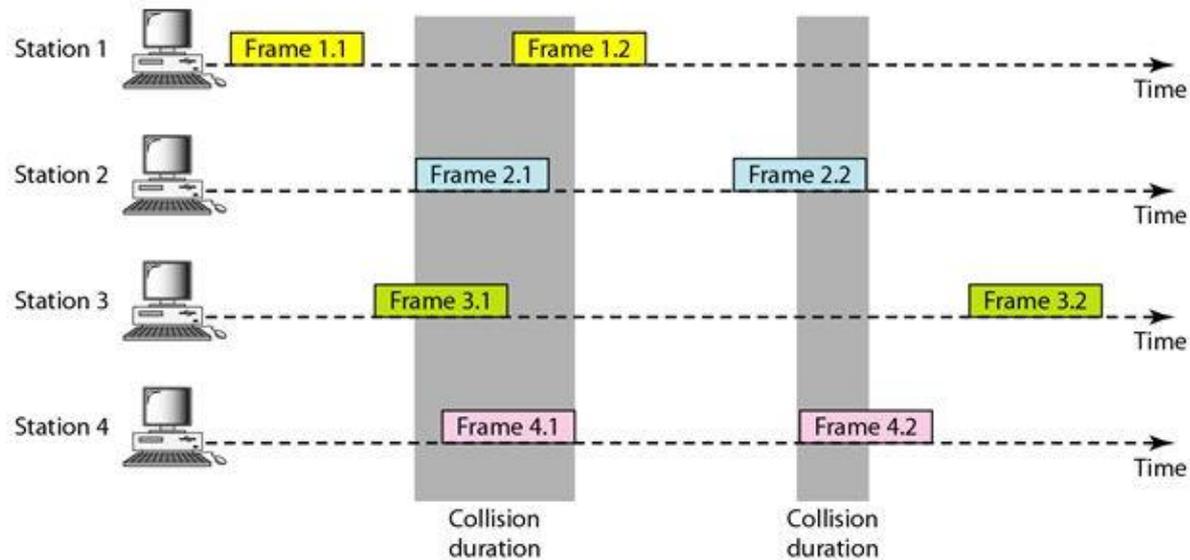
# Aloha

- Toutes les stations partagent le même canal

Problème ?

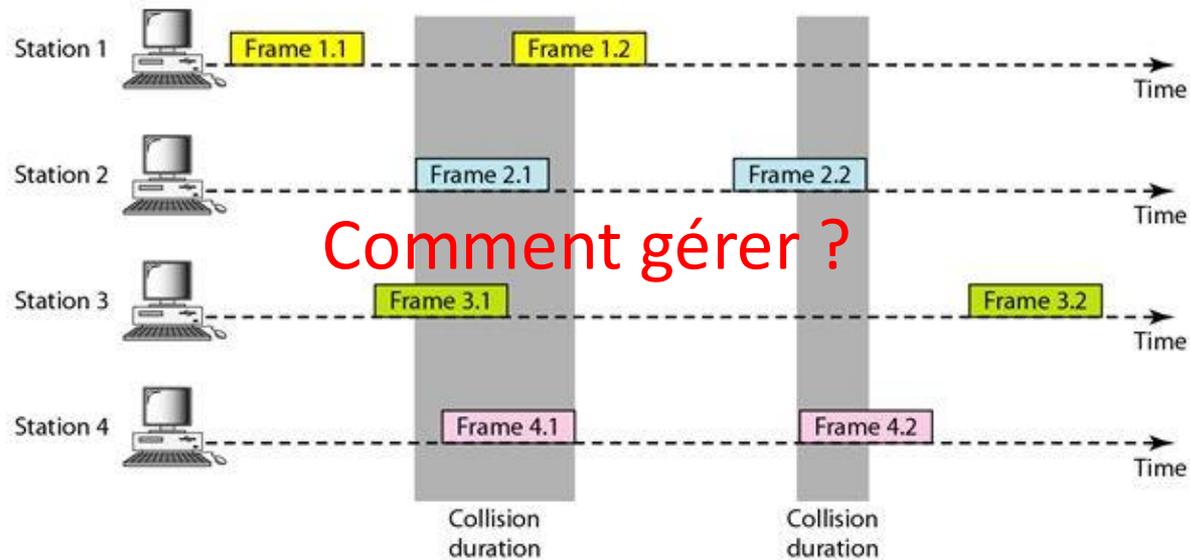
# Aloha

- Toutes les stations partagent le même canal



# Aloha

- Toutes les stations partagent le même canal



# Aloha

- Si la transmission est réussite, il y a un ACK
- Si pas de ACK après un time-out -> collision
- Si collision -> retransmission après une durée aléatoire

# Aloha

- Si la transmission est réussite, il y a un ACK
- Si pas de ACK après un time-out -> collision
- Si collision -> retransmission après une durée aléatoire

Pourquoi une durée aléatoire ?

# Aloha

- Bonne performance dans les réseaux peu chargés
- Les collisions se multiplient lorsque le charge augmente
- L'utilisation du canal maximale : 18,6% de la bande passante

# Aloha Ver. 2

Pure Aloha (Ver. 1) : Les stations sont autorisées à émettre à tout moment

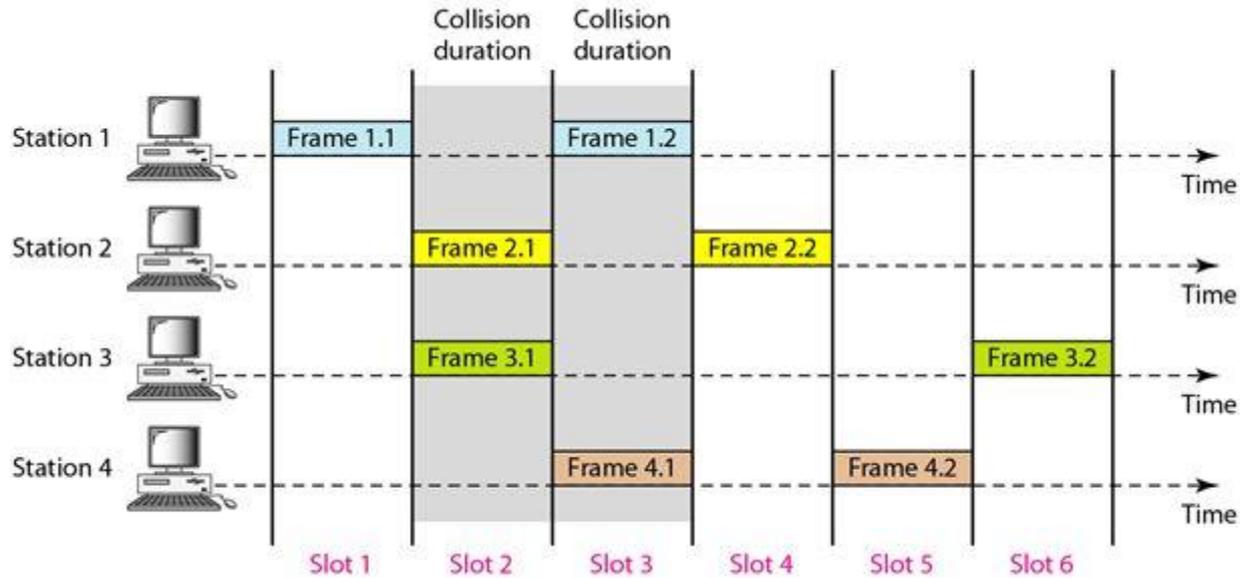
# Aloha Ver. 2

~~Pure Aloha (Ver. 1) : Les stations sont autorisées à émettre à tout moment~~

## Slotted Aloha

- Synchronisation temporelle dispo
- Le temps est découpé en créneau (slots) de durée égale à une trame
- Les stations ne peuvent émettre qu'au début d'un créneau
- Cela réduit la probabilité de collision
- L'utilisation du canal est doublée (36,8%)

# Slotted Aloha



# CSMA

## CSMA (Carrier Sense Multiple Access)

- Basé sur le mécanisme de Carrier Sense
- Écouter avant de parler (listen before you talk)
- Si le support est occupé, attendre et transmettre plus tard
- Le temps d'attente est déterminé par un back-off aléatoire

# CSMA

Trois variantes principales :

- 1-persistent CSMA
- Non-persistent CSMA
- p-persistent CSMA

# 1-persistent CSMA

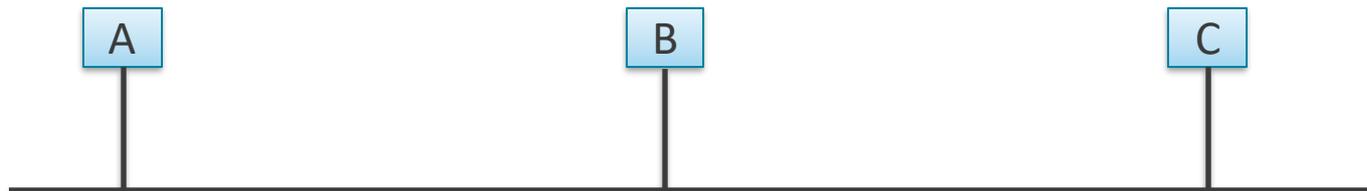
1-persistent CSMA :

- La station écoute le canal
- Si le canal est occupé -> écouter en continu
- Dès que le canal devient libre, la station envoie immédiatement

# 1-persistent CSMA

1-persistent CSMA :

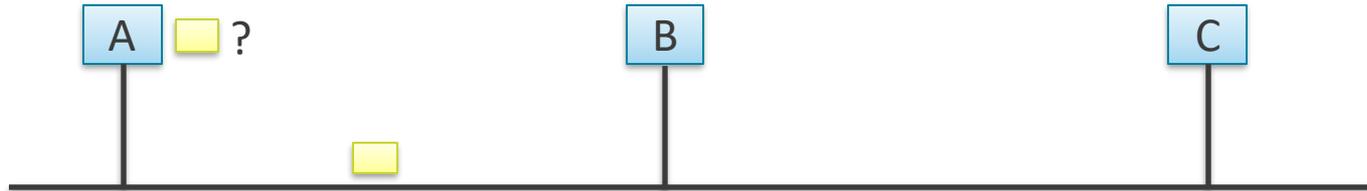
- La station écoute le canal
- Si le canal est occupé -> écouter en continu
- Dès que le canal devient libre, la station envoie immédiatement



# CSMA 1-persistent

1-persistent CSMA :

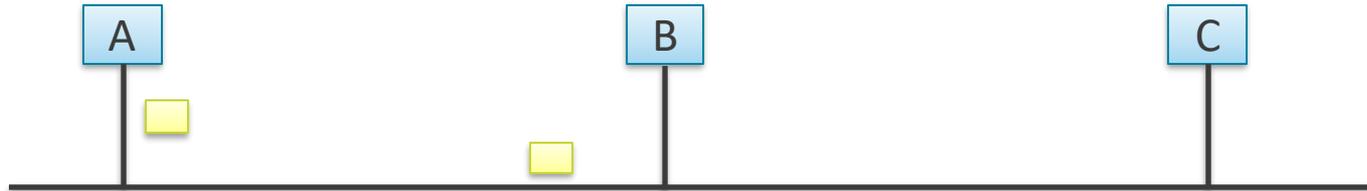
- La station écoute le canal
- Si le canal est occupé -> écouter en continu
- Dès que le canal devient libre, la station envoie immédiatement



# CSMA 1-persistent

1-persistent CSMA :

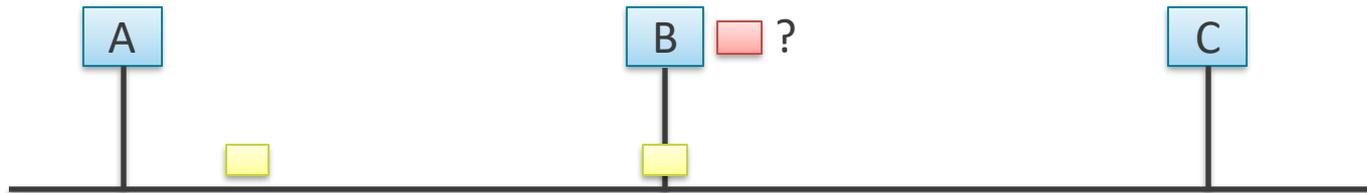
- La station écoute le canal
- Si le canal est occupé -> écouter en continu
- Dès que le canal devient libre, la station envoie immédiatement



# CSMA 1-persistent

1-persistent CSMA :

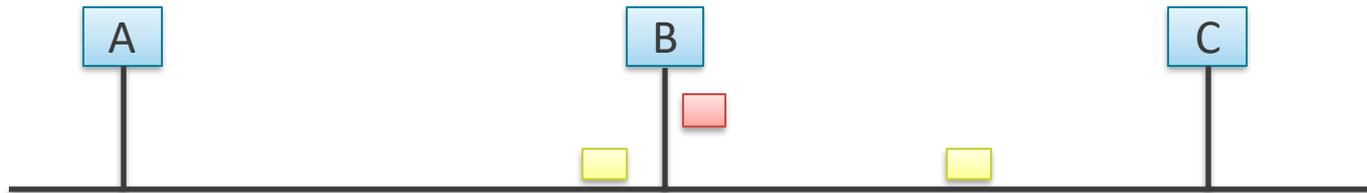
- La station écoute le canal
- Si le canal est occupé -> écouter en continu
- Dès que le canal devient libre, la station envoie immédiatement



# CSMA 1-persistent

1-persistent CSMA :

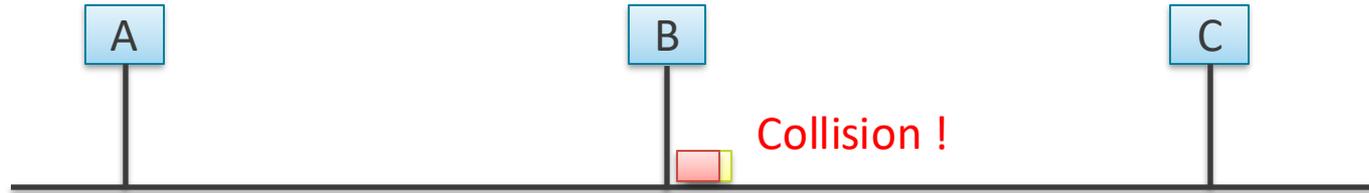
- La station écoute le canal
- Si le canal est occupé -> écouter en continu
- Dès que le canal devient libre, la station envoie immédiatement



# CSMA 1-persistent

1-persistent CSMA :

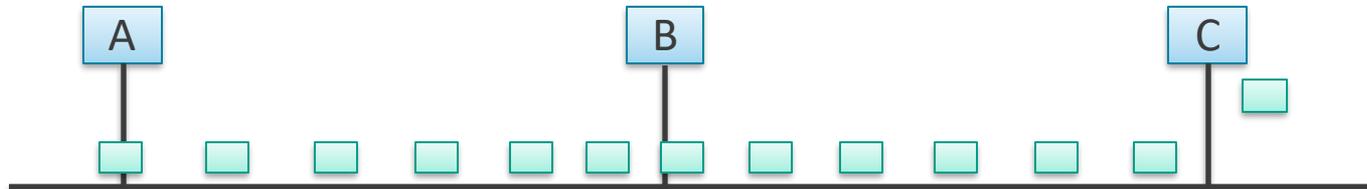
- La station écoute le canal
- Si le canal est occupé -> écouter en continu
- Dès que le canal devient libre, la station envoie immédiatement



# Non-persistent CSMA

Non-persistent CSMA (0-persistent) :

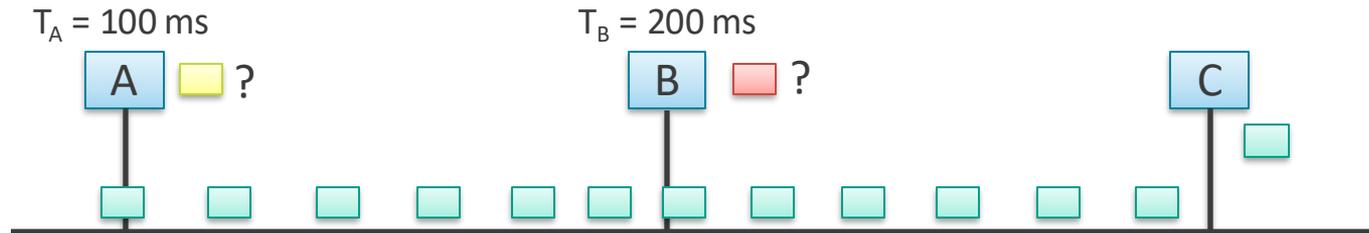
- La station écoute le canal
- Si le canal est libre, la station transmet immédiatement
- Si le canal est occupé, la station n'attend pas en continu
  - La station attend un temps aléatoire (back-off aléatoire)
  - La station réécoute le canal



# Non-persistent CSMA

Non-persistent CSMA (0-persistent) :

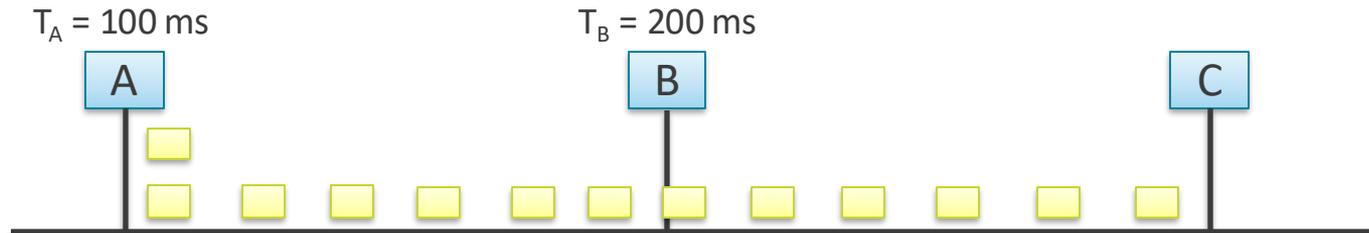
- La station écoute le canal
- Si le canal est libre, la station transmet immédiatement
- Si le canal est occupé, la station n'attend pas en continu
  - La station attend un temps aléatoire (back-off aléatoire)
  - La station réécoute le canal



# Non-persistent CSMA

Non-persistent CSMA (0-persistent) :

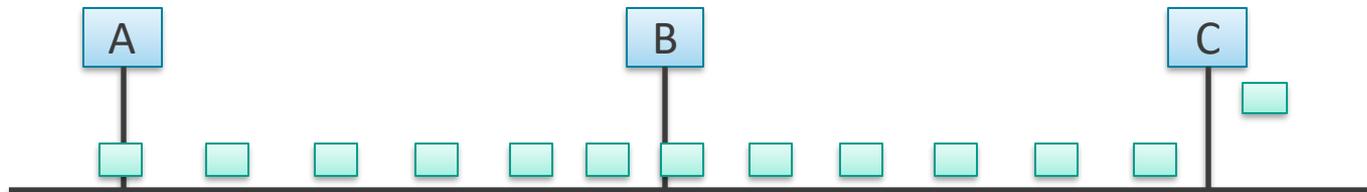
- La station écoute le canal
- Si le canal est libre, la station transmet immédiatement
- Si le canal est occupé, la station n'attend pas en continu
  - La station attend un temps aléatoire (back-off aléatoire)
  - La station réécoute le canal



# p-persistent CSMA

p-persistent CSMA :

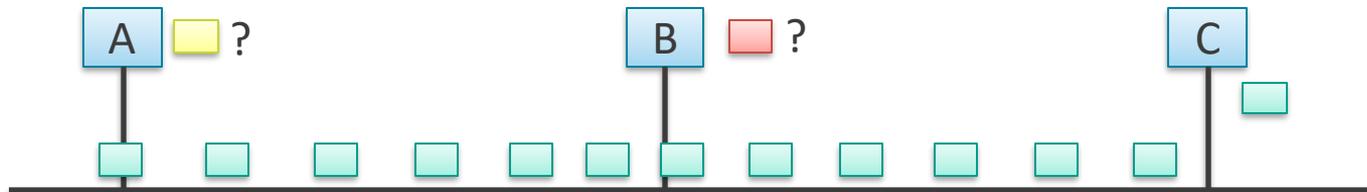
- Le temps est divisé en créneaux (slots) de même durée
- La station écoute le canal
- Le canal est occupé -> la station écoute et attend qu'il devienne libre
- Le canal est libre
  - La station envoie immédiatement avec une probabilité  $p$
  - La station attend encore un slot avec probabilité  $(1-p)$



# p-persistent CSMA

p-persistent CSMA :

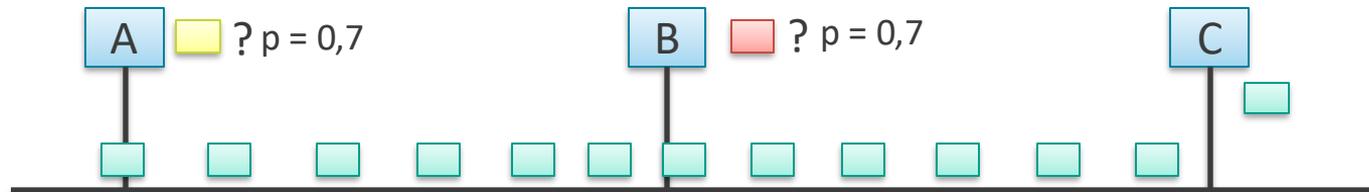
- Le temps est divisé en créneaux (slots) de même durée
- La station écoute le canal
- Le canal est occupé -> la station écoute et attend qu'il devienne libre
- Le canal est libre
  - La station envoie immédiatement avec une probabilité  $p$
  - La station attend encore un slot avec probabilité  $(1-p)$



# p-persistent CSMA

p-persistent CSMA :

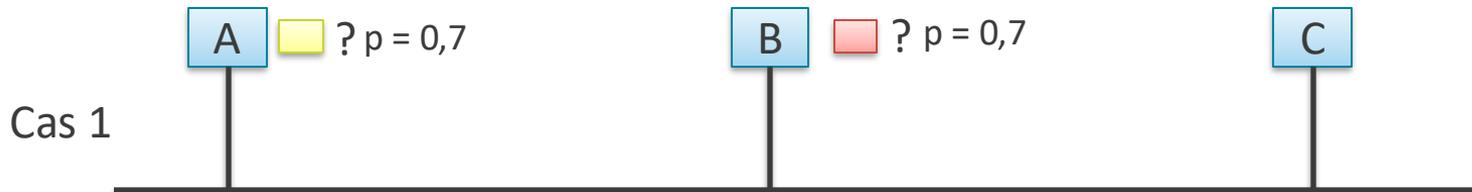
- Le temps est divisé en créneaux (slots) de même durée
- La station écoute le canal
- Le canal est occupé -> la station écoute et attend qu'il devienne libre
- Le canal est libre
  - La station envoie immédiatement avec une probabilité  $p$
  - La station attend encore un slot avec probabilité  $(1-p)$



# p-persistent CSMA

p-persistent CSMA :

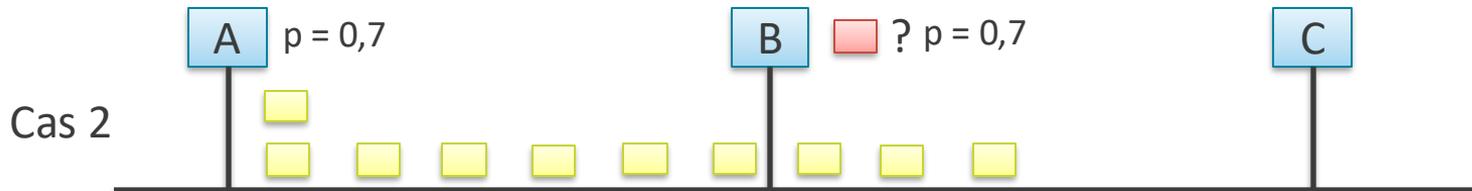
- Le temps est divisé en créneaux (slots) de même durée
- La station écoute le canal
- Le canal est occupé -> la station écoute et attend qu'il devienne libre
- Le canal est libre
  - La station envoie immédiatement avec une probabilité  $p$
  - La station attend encore un slot avec probabilité  $(1-p)$



# p-persistent CSMA

p-persistent CSMA :

- Le temps est divisé en créneaux (slots) de même durée
- La station écoute le canal
- Le canal est occupé -> la station écoute et attend qu'il devienne libre
- Le canal est libre
  - La station envoie immédiatement avec une probabilité  $p$
  - La station attend encore un slot avec probabilité  $(1-p)$





# Performance protocoles MAC

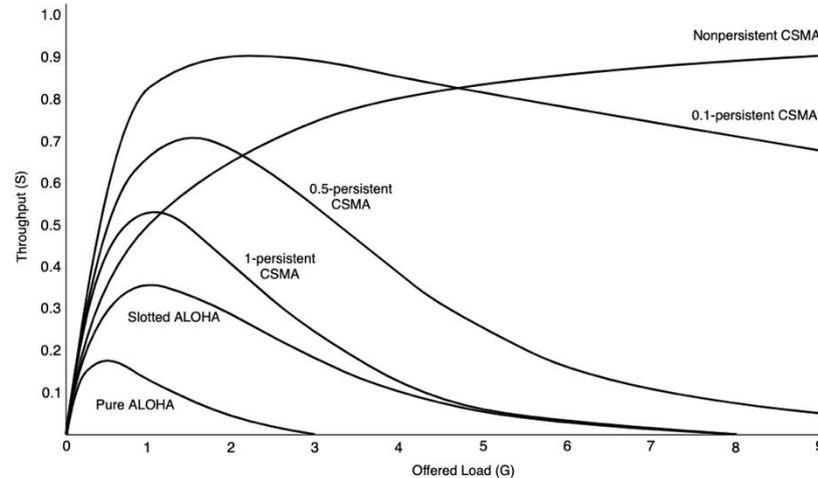


Figure 5.6 – Comparison of the channel utilization versus offered load for CSMA and ALOHA protocols.

# CSMA dans l'Ethernet ?

CSMA/CD (CSMA with Collision Detection)

CS : écouter le support pour détecter les transmissions en cours

CD : détecter une collision dès que possible et passer en back-off

Contexte :

- Couche : sous-couche MAC (Medium Access Control)
- Type de topologie : bus/câble partagé
- Plusieurs stations peuvent émettre en même temps -> collisions

# CSMA/CD

Comment on peut détecter une collision ?

# CSMA/CD

Comment on peut détecter une collision ?

- Chaque station écoute pendant qu'elle émet
- Si le signal reçu  $\neq$  le signal envoyé  $\rightarrow$  collision détectée
- Interruption immédiate de la transmission
- Émission d'un jam signal
- Back-off aléatoire avant de réessayer

Comment on peut détecter une collision ?

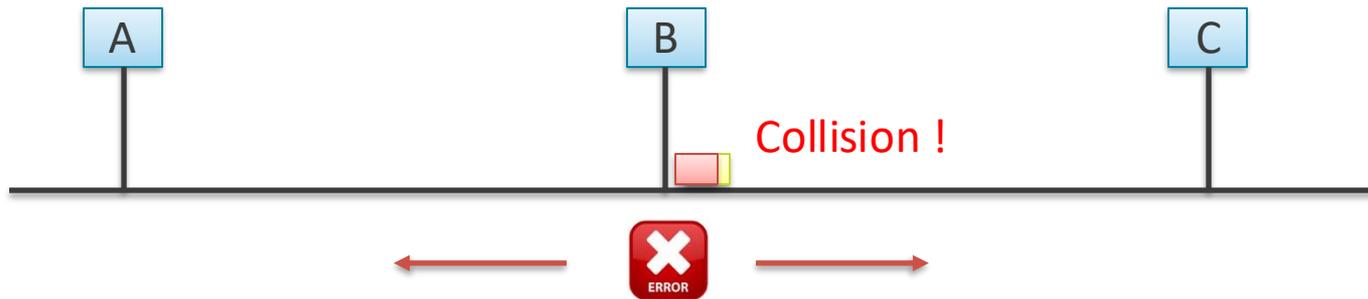
- Chaque station écoute pendant qu'elle émet
- Si le signal reçu  $\neq$  le signal envoyé  $\rightarrow$  collision détectée
- Interruption immédiate de la transmission
- Émission d'un jam signal
- Back-off aléatoire avant de réessayer

Pourquoi envoyer un jam signal ?

# CSMA/CD

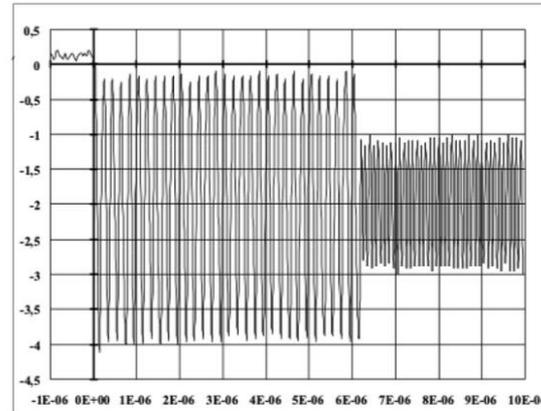
Le but de jam signal :

Propager l'information de collision à toutes les stations du réseau partagé.



# CDMA/CD

Collision -> Signal perturbé, amplitude anormale ou distorsion



# CSMA/CD

On revient sur la taille de trame.

Pourquoi on impose une taille minimale de trame ?

# CSMA/CD

On revient sur la taille de trame.

Pourquoi on impose une taille minimale de trame ?

L'émetteur doit détecter la collision avant la fin de la transmission.

Sinon, l'émetteur considère qu'il n'y a pas de collision.

# CSMA/CD

On revient sur la taille de trame.

Pourquoi on impose une taille minimale de trame ?

L'émetteur doit détecter la collision avant la fin de la transmission.

Sinon, l'émetteur considère qu'il n'y a pas de collision.

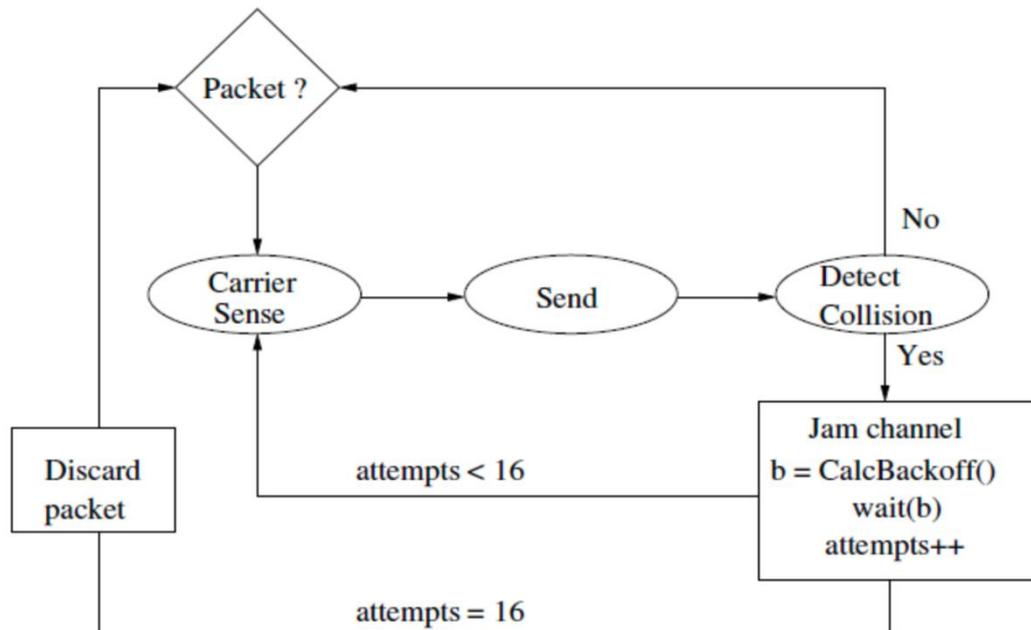
Au pire cas, la durée d'émission d'une trame doit être au moins deux fois le temps de propagation.

## BEB (Binary Exponential Back-off)

- Objectif : Lorsqu'une collision est détectée, les stations doivent retransmettre à des instants différents
- Le temps est découpé en créneaux
- Lors de la  $i^e$  collision consécutive, la station choisit aléatoirement une valeur  $b$  dans la fenêtre  $[0, 2^i - 1]$
- La station attend donc  $b$  créneaux avant de tenter une nouvelle transmission

# CSMA/CD

## Machine à états du protocole CSMA/CD



## Rôles de la couche MAC

- Partager les ressources (blocs de fréquences, slots, codes) entre les utilisateurs
- Gérer les collisions

## Clarification

- Allocation statique
- Allocation dynamique
  - Protocoles centralisés
  - Protocoles distribués
    - Protocoles déterministes
    - Protocoles stochastiques

# Résumé (Cours 3)

- Partage du support & multiplexage
- Contrôle d'accès
- Aloha (Pure/Slotted)
- CSMA (1-persistent, non-persistent, p-persistent)
- CSMA/CD