

Chapitre 5

Couche Liaison : Principes

Plan

- Introduction
- Délimitation de trames
- Détection/Correction d'erreurs
- Contrôle de flux

Introduction

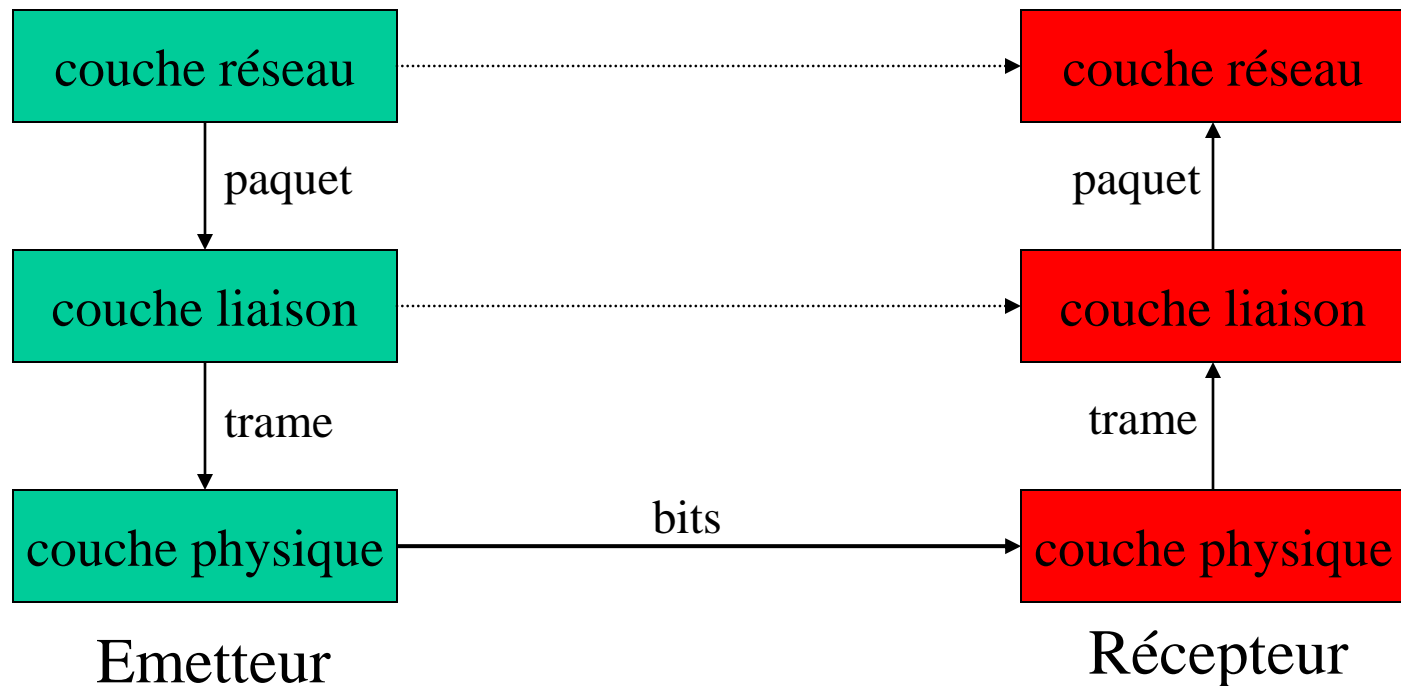
Couche liaison

- Cette couche doit assurer une transmission exempte d'erreurs sur un canal de communication.
- Les données sont fractionnées en trames.

Description

- La couche liaison récupère des paquets de la couche réseau.
- Pour chaque paquet, elle construit une (ou plusieurs) trame(s).
- La couche liaison envoie chaque trame à la couche physique.

Description



Services offerts

- Gestion (délimitation) de trames
- Contrôle d'erreurs
- Contrôle de flux
- Contrôle d'accès à un canal partagé (MAC)

Délimitation de trames

Délimitation des trames

- Il existe trois méthodes :
 - Compter les caractères
 - Utiliser des champs délimiteurs de trame
 - Ils se situent en début et en fin de trame
 - Des bits (ou caractères) de transparence sont nécessaires
 - Violier le codage normalement utilisé dans la couche physique

Compter les caractères

- On utilise un champ dans l'en-tête de la trame pour indiquer le nombre de caractères de la trame
- Problème : si la valeur du champ est modifiée au cours de la transmission
- Méthode rarement utilisée seule

Exemple

Trames émises

06	'S'	'U'	'P'	'E'	'R'	03	'L'	'E'	06	'C'	'O'	'U'	'R'	'S'
----	-----	-----	-----	-----	-----	----	-----	-----	----	-----	-----	-----	-----	-----



Trames reçues

06	'S'	'U'	'P'	'E'	'R'	04	'L'	'E'	06	43	'O'	'U'	'R'	'S'
----	-----	-----	-----	-----	-----	----	-----	-----	----	----	-----	-----	-----	-----

code ASCII de 'C'

Utiliser des délimiteurs

- Un fanion (délimiteur) est placé :
 - au début de chaque trame
 - à la fin de chaque trame (en fait, au début de la suivante)
- Un fanion (flag) = séquence particulière de bits
- Des bits de transparence sont alors nécessaires pour qu'une séquence binaire dans la trame ne corresponde accidentellement au fanion.

Exemple

- Fanion : 01111110
- Bit de transparence : 0 inséré après toute séquence de cinq 1 successifs dans la trame.
- Technique utilisée dans :
 - HDLC
 - PPP

Exemple

Données :

01011001111110

Trame :

01111110 010110011111010 01111110

Utiliser des fanions

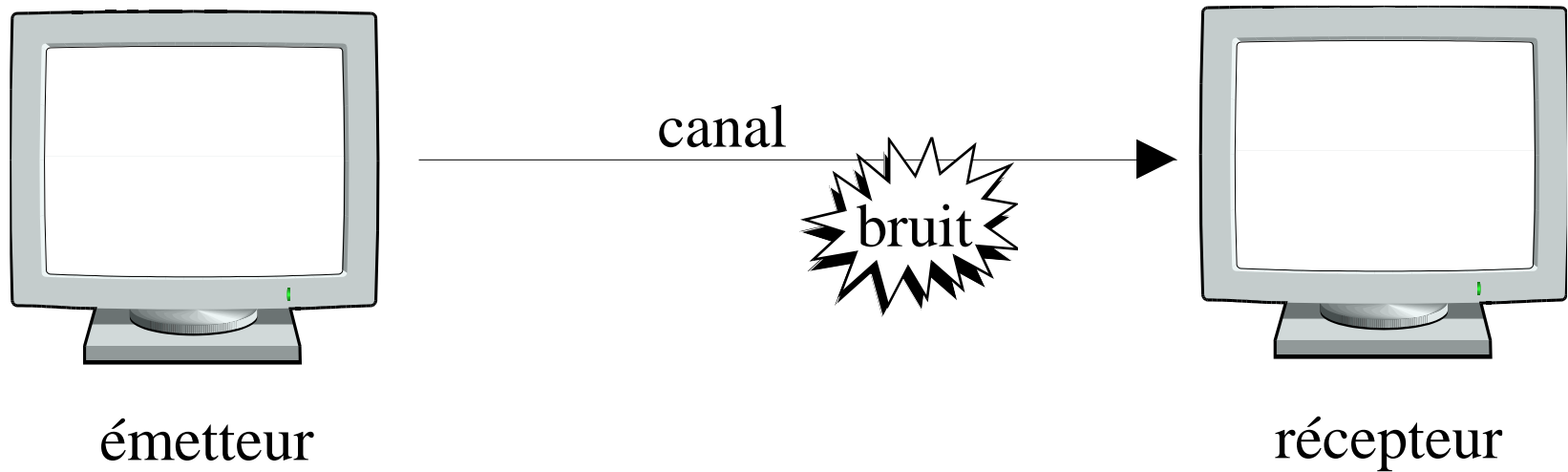
- Avantages
 - permet toujours de retrouver la synchronisation
 - permet l'envoi de trames de tailles quelconques
 - technique la plus simple
- Cette technique est utilisée également en considérant des caractères de délimitation et des caractères de transparence.

Violer le codage

- Utilisable lorsque le codage sur le support physique contient des redondances
- Par exemple :
 - 0 = impulsion positive puis négative
 - 1 = impulsion négative puis positive
 - On peut donc utiliser les combinaisons positive-positive et négative-négative pour délimiter les trames
- Utilisée dans la norme 802

Détection/Correction d'erreurs

Transmission d'information



Causes d'erreurs sur un canal

- rayonnement électromagnétique
 - relais
 - émetteurs
- câblage mal isolé
- effet de distorsion

Taux d'erreur sur un canal

$$\text{taux d'erreur} = \frac{\text{nombre de bits erronés}}{\text{nombre de bits émis}}$$

- 10^{-9} pour les réseaux locaux
- 10^{-5} pour le RTC (Réseau Téléphonique Commuté)
- taux élevé pour le téléphone sans fil

Types d'erreurs de transmission

- Erreurs isolées
 - simples à détecter
 - simples à corriger
 - proportion élevée de blocs affectés
- Erreurs en rafales
 - difficiles à détecter
 - difficiles à corriger
 - proportion faible de blocs affectés

Exemple

- taille des blocs échangés : 1000 bits
- taux d'erreur : $1/1000$
- erreurs isolées
 - la plupart des blocs en erreur
- erreurs en rafale (de longueur 100)
 - un bloc sur 100 en erreur en moyenne

Deux stratégies possibles

La destination peut :

- détecter les erreurs, puis demander une retransmission
 - code détecteurs d'erreurs
- détecter et corriger les erreurs
 - codes correcteurs d'erreur

Principe des codes

- Exploiter la redondance d'informations
⇒ ajouter des bits de contrôle aux bits de données
- Corriger est plus difficile que détecter
⇒ plus de bits de contrôle

Suprématie des codes détecteurs

Les codes détecteurs sont plus souvent utilisés que les codes correcteurs mais :

- lorsque le canal est unidirectionnel
- lorsque la distance est élevée (satellite)
- lorsque les erreurs sont isolées

les codes correcteurs peuvent (ou doivent) être utilisés.

Code de contrôle de parité

Principe : un seul bit (dit de parité) est ajouté aux bits de données.

- parité paire : le nombre de bits à 1 du mot formé doit être pair.
- parité impaire : le nombre de bits à 1 du mot formé doit être impair.

Exemple

- parité paire

100 0001 : bits de données

+ 0 : bit de contrôle

= 0100 0001 : mot de code

- parité impaire

101 0001 : bits de données

+ 1 : bit de contrôle

= 1101 0001 : mot de code

Contrôle de flux

Contrôle de flux

- Utilisation d'acquittements
- Gestion de temporisateurs
- Numérotation des trames
- Limitation du nombre de trames pouvant être envoyées par l'émetteur

Protocole 1

- Hypothèses
 - mémoire tampon infinie
 - canal parfait (pas de pertes ni d'erreurs)
 - protocole mono-directionnel
 - un émetteur
 - un récepteur

Emetteur

Tant que vrai répéter

p ← coucheReseau.donnerPaquet()

t ← construireTrame(p)

couchePhysique.prendreTrame(t)

Fin tant que

Récepteur

Tant que vrai répéter

t ← couchePhysique.donnerTrame()

p ← extrairePaquet(t)

coucheReseau.prendrePaquet(p)

Fin tant que

Protocole 2

- Protocole de type « envoyer et attendre »
- Hypothèse levée : mémoire tampon infinie
- Principe
 - le récepteur envoie une trame d'acquittement après chaque trame reçue
 - l'émetteur attend de recevoir un acquittement avant d'émettre la trame suivante

Emetteur

Tant que vrai répéter

p ← coucheReseau.donnerPaquet()

t ← construireTrame(p)

couchePhysique.prendreTrame(t)

couchePhysique.attendreAquittement()

Fin tant que

Récepteur

Tant que vrai répéter

t ← couchePhysique.donnerTrame()

p ← extrairePaquet(t)

coucheReseau.prendrePaquet(p)

couchePhysique.envoyerAcquittement()

Fin tant que

Protocole 3

- Hypothèse levée : canal parfait
 - ⇒ des trames peuvent être erronées
 - ⇒ des trames peuvent être perdues
- Principe
 - Utiliser une méthode de détection d'erreurs
 - Le récepteur émet une trame d'acquittement si la trame arrivée est correcte.
 - L'émetteur ré-émet une trame si aucun ack reçu et si un certain délai de temporisation a expiré

Emetteur

Tant que vrai répéter

p ← coucheReseau.donnerPaquet()

t ← construireTrame(p)

booléen ack ← faux

Tant que ack = faux répéter

couchePhysique.prendreTrame(t)

ack ← couchePhysique.attendreAquittement()

Fin Tant que

Fin tant que

Récepteur

Tant que vrai répéter

t ← couchePhysique.donnerTrame()

Si estCorrecte(t) alors

p ← extrairePaquet(t)

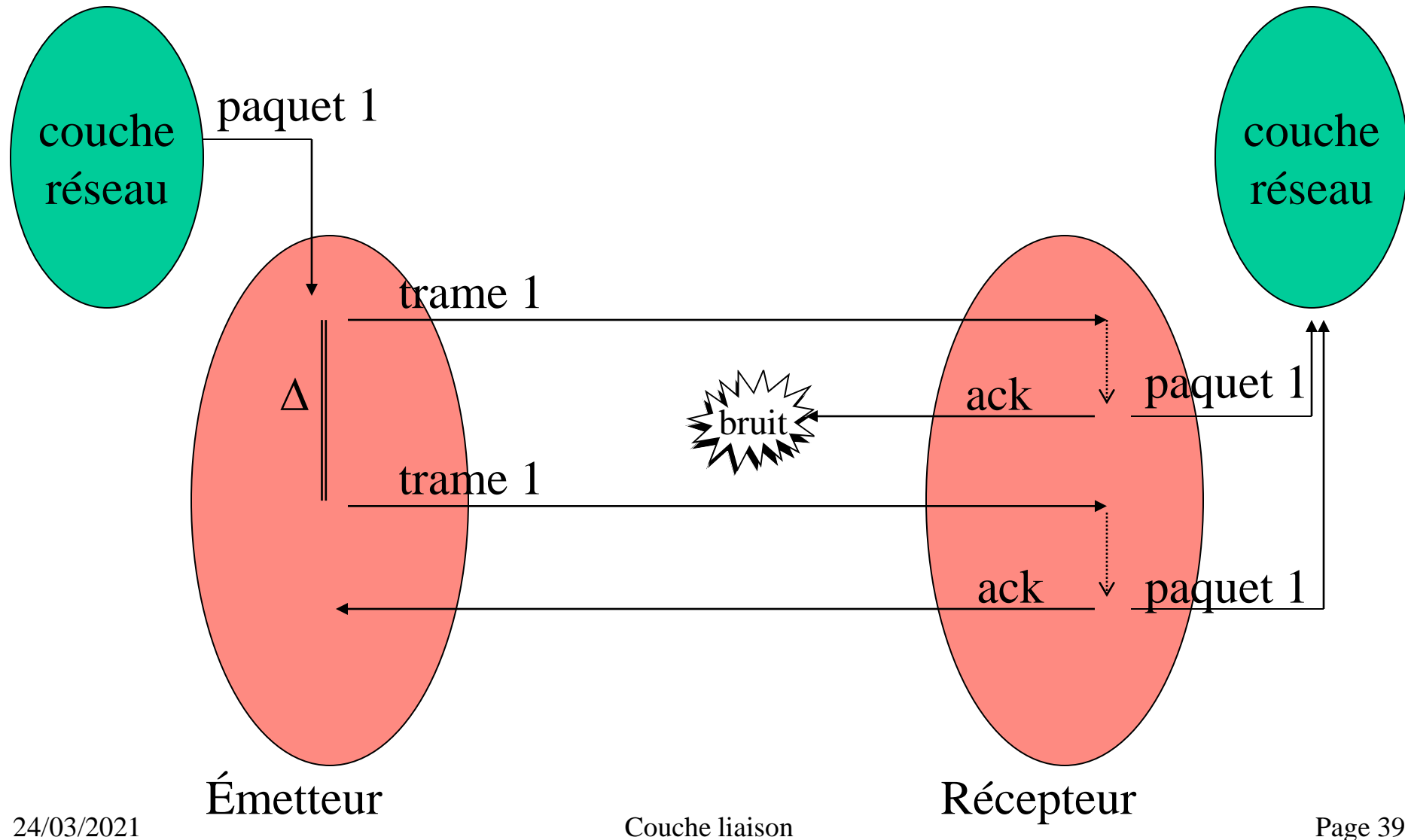
coucheReseau.prendrePaquet(p)

couchePhysique.envoyerAcquittement()

Fin si

Fin tant que

Problème



Solution

- En considérant le problème précédent :
 - il s'avère nécessaire de numéroté les trames pour distinguer deux trame successives.
- De plus, comme on le verra en TD :
 - il est préférable que la trame d'acquittement contienne le numéro de la trame qui est acquittée.

Protocole 4

- Protocole à fenêtres d'anticipation (sliding windows).
- Deux fenêtres sont gérées par chaque entité de couche liaison. En effet :
 - Toute entité émettrice possède une fenêtre d'anticipation appelée fenêtre d'émission
 - Toute entité réceptrice possède une fenêtre d'anticipation appelée fenêtre de réception

Fenêtre d'émission

- La fenêtre d'émission indique la liste des numéros de trames dont on attend l'acquittement.
- Elle possède une taille maximale maxE indiquant le nombre maximal de trames qui peuvent être envoyées sans se préoccuper des acquittements.
- Elle possède une taille courante variable curE de valeur inférieure ou égale à maxE. On a toujours :

$$0 \leq \text{curE} \leq \text{maxE}$$

Bornes

- La fenêtre est représentée par deux frontières (bornes).
 - la frontière droite (borne supérieure) est incrémentée de 1 à chaque envoi
 - la frontière gauche (borne inférieure) est incrémentée de 1 à chaque acquittement reçu correspondant à la trame associée à cette frontière

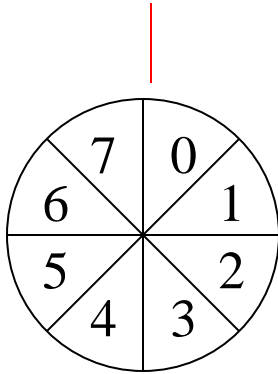
Mémoires tampons

- Côté émetteur, maxE mémoires tampons sont nécessaires pour stocker les trames.
- En effet, il est possible qu'il faille envoyer à nouveau une ou plusieurs trames.

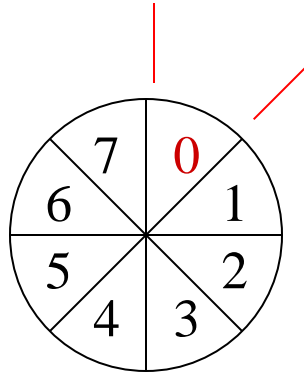
Exemple

- Considérons que :
 - les trames soient numérotés de 0 à 7 (sur 3 bits)
 - $\text{maxE} = 3$
 - $\text{curE} = 0$ (initialement)

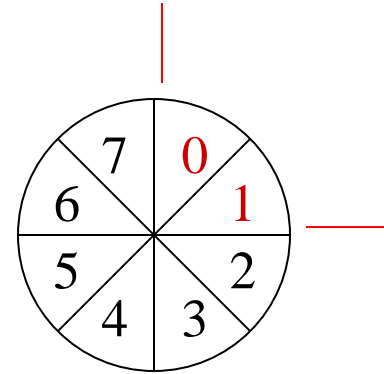
Exemple



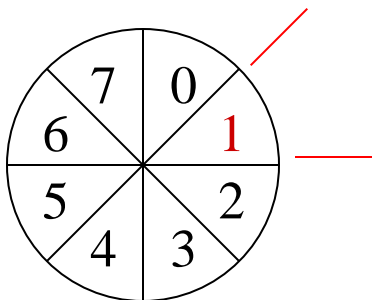
e1)



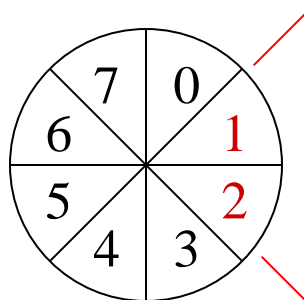
e2)



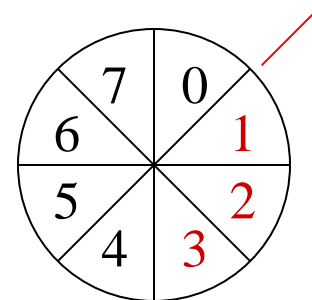
e3)



e4)



e5)



e6)

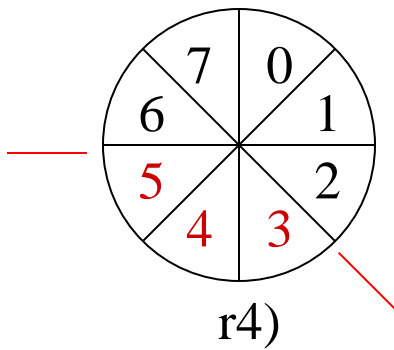
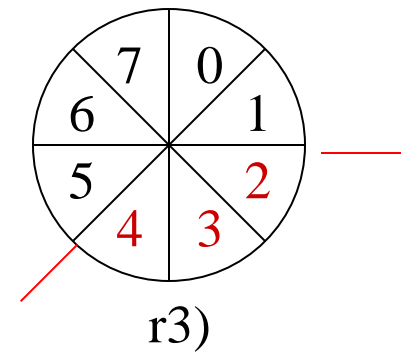
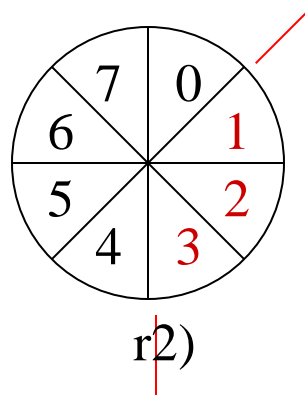
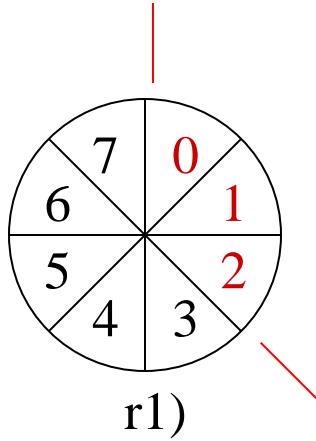
Fenêtre de réception

- La fenêtre de réception indique la liste des numéros de trame attendus.
- La fenêtre de réception possède une taille R qui ne varie pas (sauf cas particuliers).

Bornes

- La fenêtre est représentée par deux frontières (bornes).
- les deux frontières sont toutes deux incrémentées de 1 chaque fois qu'une trame correcte correspondant à la frontière droite (borne inférieure) est reçue.

Exemple



Pipelining

- Utiliser des fenêtrage d'anticipation permet d'utiliser la technique de pipelining.
- Cela consiste à envoyer plusieurs trames successivement sans attendre de recevoir les trames d'acquittement.
- Il est important de bien régler la largeur des fenêtres pour améliorer l'efficacité du pipelining.

Calcul de l'efficacité

- Temps d'émission $T_e(t)$ = temps nécessaire pour que la trame t passe sur le canal.

$$T_e(t) = \text{length}(t)/d \text{ où :}$$

d = débit du canal (en bits/s)

$\text{length}(t)$ = longueur de la trame t (en bits)

- Temps de propagation T_p = temps nécessaire pour qu'un bit passe de l'émetteur au récepteur.
- Temps de traitement T_t = temps nécessaire au récepteur pour traiter une trame.

Calcul de l'efficacité

- Hypothèse : toutes les trames sont de longueur m .
- Temps d'attente T_a = temps nécessaire pour recevoir l'acquittement d'une trame émise.

$$T_a = T_e(t) + T_p + T_t + T_e(\text{ack}) + T_p \text{ où :}$$

- t désigne une trame quelconque
- ack désigne une trame quelconque d'acquittement.
- Temps d'émission maximale $T_{em} =$ temps maximal pendant lequel il est possible d'émettre des trames. $T_{em} = \max E * m / d$

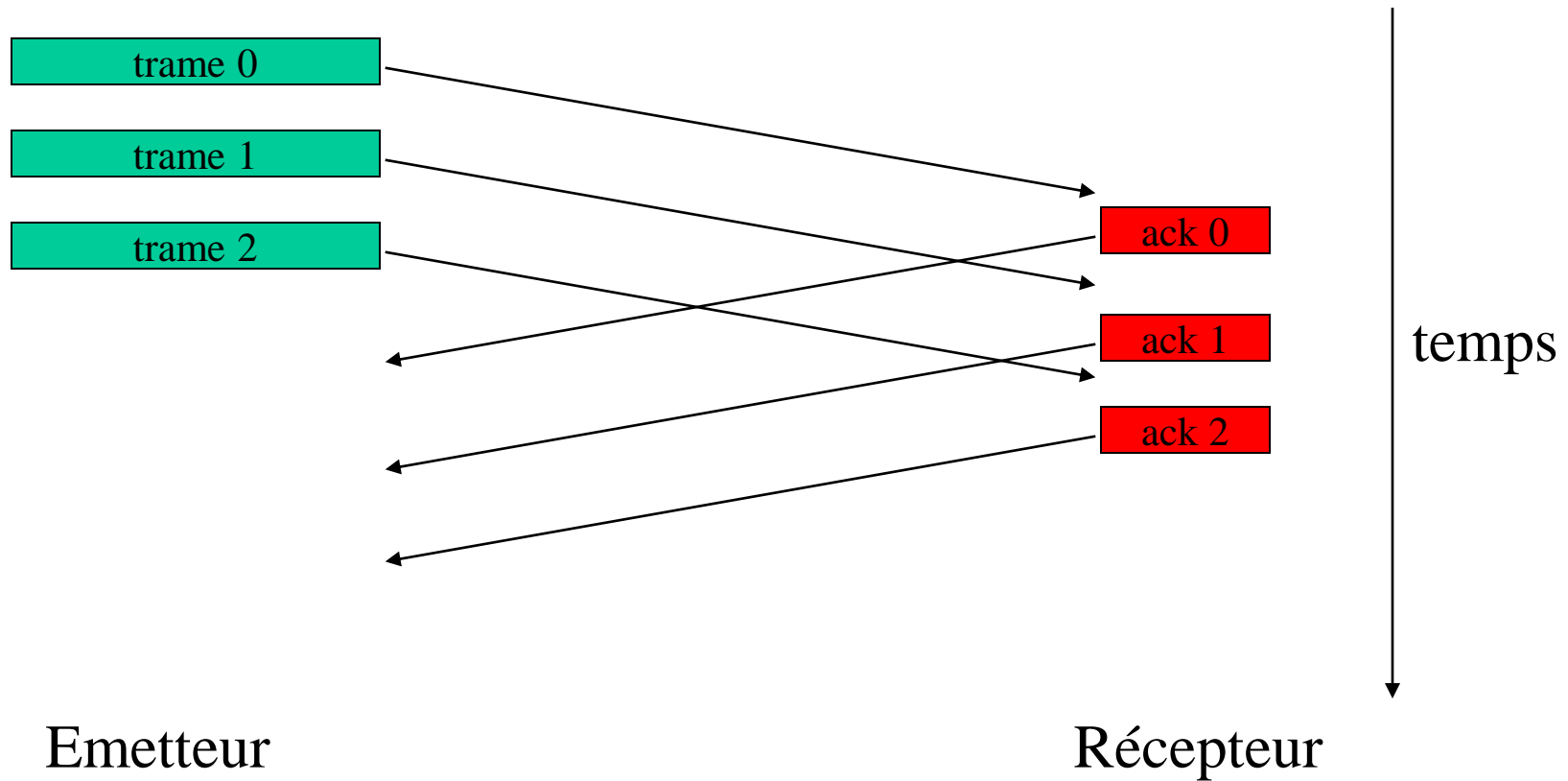
Efficacité

- Efficacité :
100% si $T_{em} \geq T_a$
 T_{em}/T_a sinon
- NB : calcul théorique ne prenant en compte
 - ni les erreurs de transmission
 - ni les problèmes de surcharge du récepteur

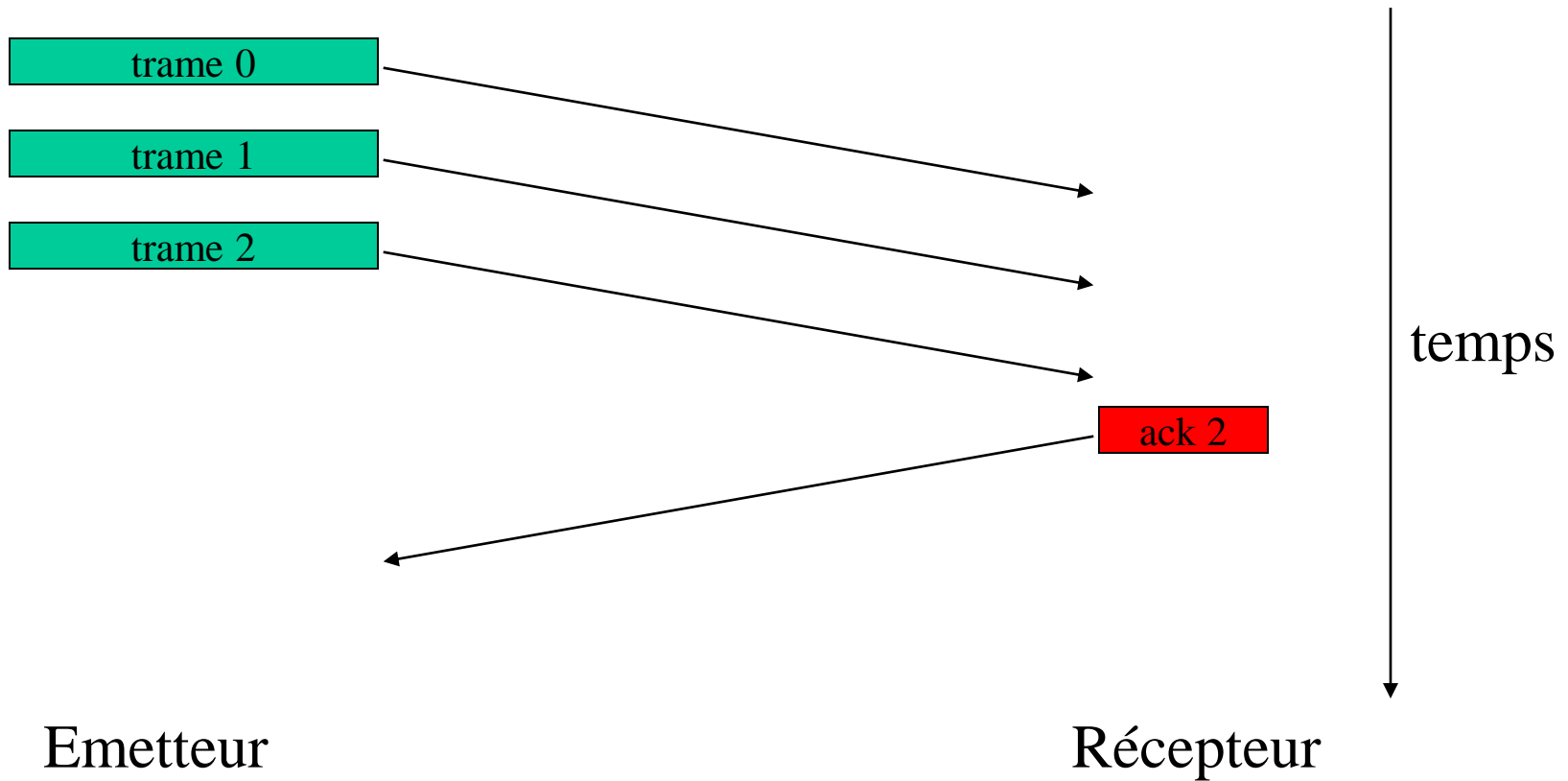
Acquittements

- Lorsque plusieurs trames doivent être acquittées, il est possible :
 - d’envoyer un acquittement « individuel » pour chaque trame
 - d’envoyer un acquittement « collectif » en indiquant
 - le plus grand numéro de trame parmi celles qui sont acquittées
 - ou le numéro de la prochaine trame attendue.

Acquittement « individuel »



Acquittement « collectif »



Contrainte

- Il est nécessaire que :

$$\max E < n$$

où n représente le nombre de numéros de trames (allant de 0 à $n-1$) pouvant être utilisés.

- En effet, comme le récepteur peut envoyer plusieurs fois le même acquittement, une ambiguïté peut apparaître.

Exemple

$\text{maxE} = 8$ et $n = 8$

- L'émetteur envoie une série de trames numérotées de 0 à 7,
- L'émetteur reçoit un acquittement « collectif » avec le numéro 7,
- L'émetteur envoie une nouvelle série de trames numérotées de 0 à 7,
- L'émetteur reçoit un acquittement « collectif » avec le numéro 7.

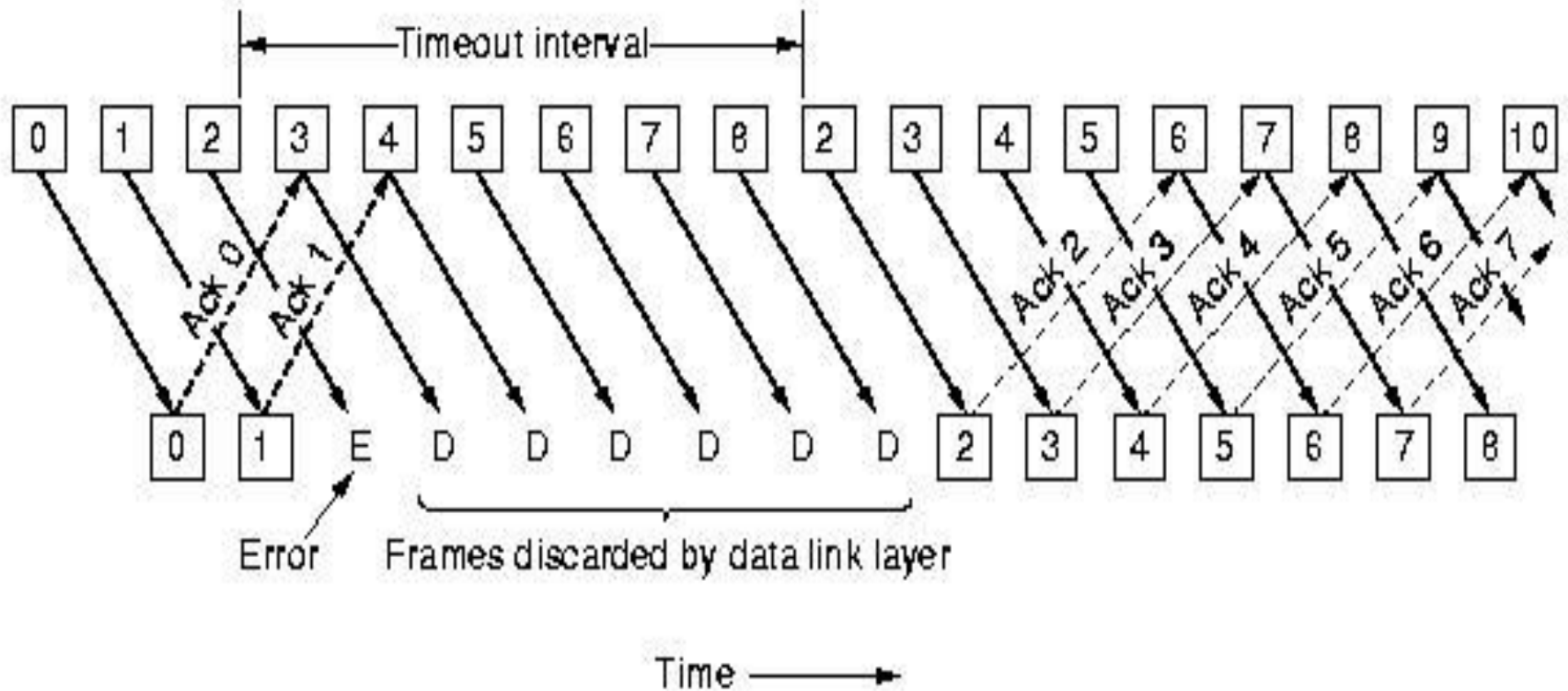
Erreurs de transmission

- Si une trame située au milieu d'une série est perdue ou erronée ?
- Deux techniques de rejet sont possibles :
 - technique du rejet total
 - technique du rejet sélectif

Technique de rejet total

- Le récepteur rejettent toutes les trames qui suivent celle qui est erronée.
 - inconvénient : le canal est mal exploité
 - avantage : pas besoin de mémoires tampons
- Cette technique correspond à l'utilisation d'une fenêtre de réception de taille 1.

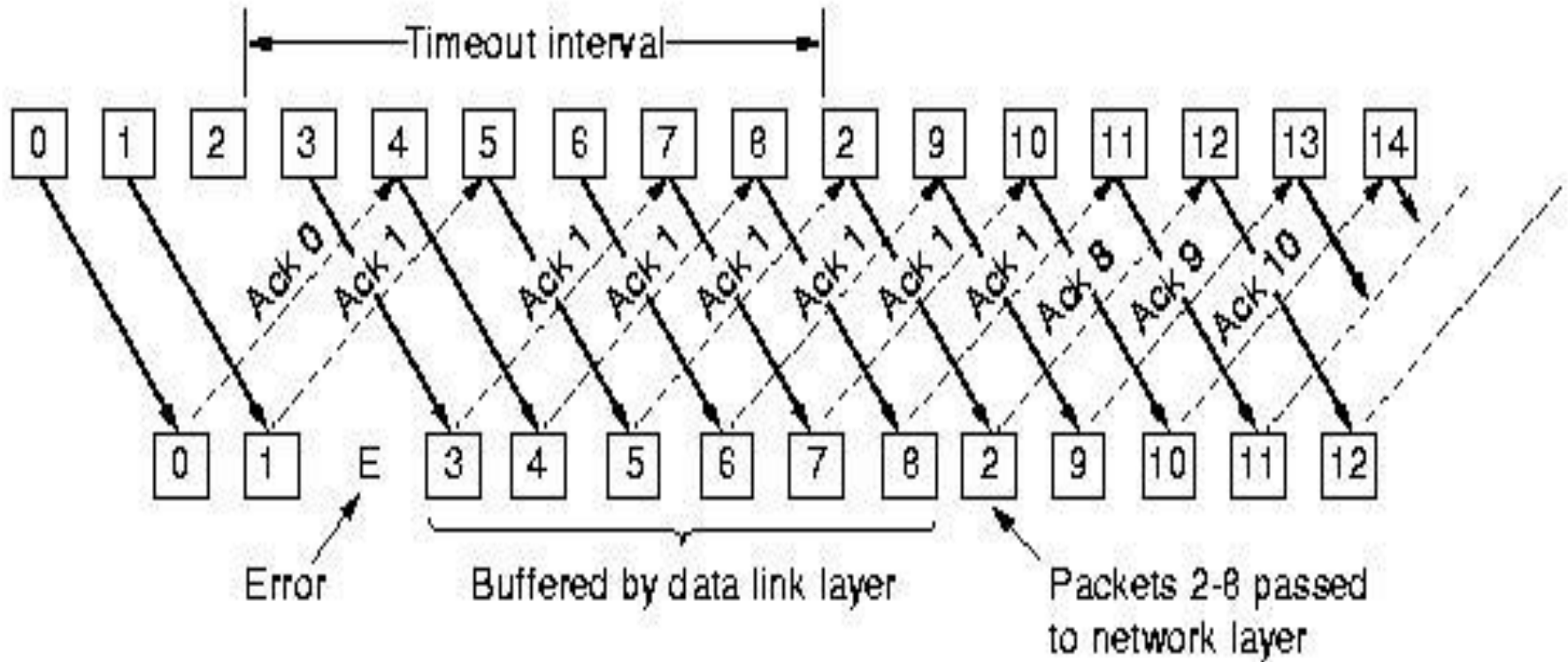
Exemple



Technique de rejet sélectif

- Le récepteur accepte les suivantes (en les stockant) jusqu'à une certaine limite donnée par R.
 - avantage : le canal est mieux exploité
 - inconvénient : besoin de mémoires tampons
- Cette technique correspond à l'utilisation d'une fenêtre de réception de taille supérieure à 1.
- Le récepteur utilise un acquittement « collectif ».

Example



Technique de rejet sélectif

- La technique de rejet sélectif impose une nouvelle contrainte sur la valeur de $\max E$: il faut choisir une valeur au plus égale à $n/2$
- En effet, un problème survient lorsque une nouvelle fenêtre au niveau de l'émetteur se superpose à une ancienne.

Superposition

- Lorsque la communication est bidirectionnelle (un émetteur et un récepteur de chaque côté), il est possible d'utiliser un champ des trames de données pour indiquer un acquittement.
- Il s'agit d'une technique de superposition (piggybacking)
- Avantage : quelques bits versus une trame
- Inconvénient : attendre une trame de données
- NB : si pas de trames de données à envoyer dans un certain délai, envoyer une trame de contrôle.