Sécurité des Réseaux

Jean-Yves Antoine
LI - Université François Rabelais de Tours
Jean-Yves.Antoine AT univ-tours.fr

Sécurité des réseaux

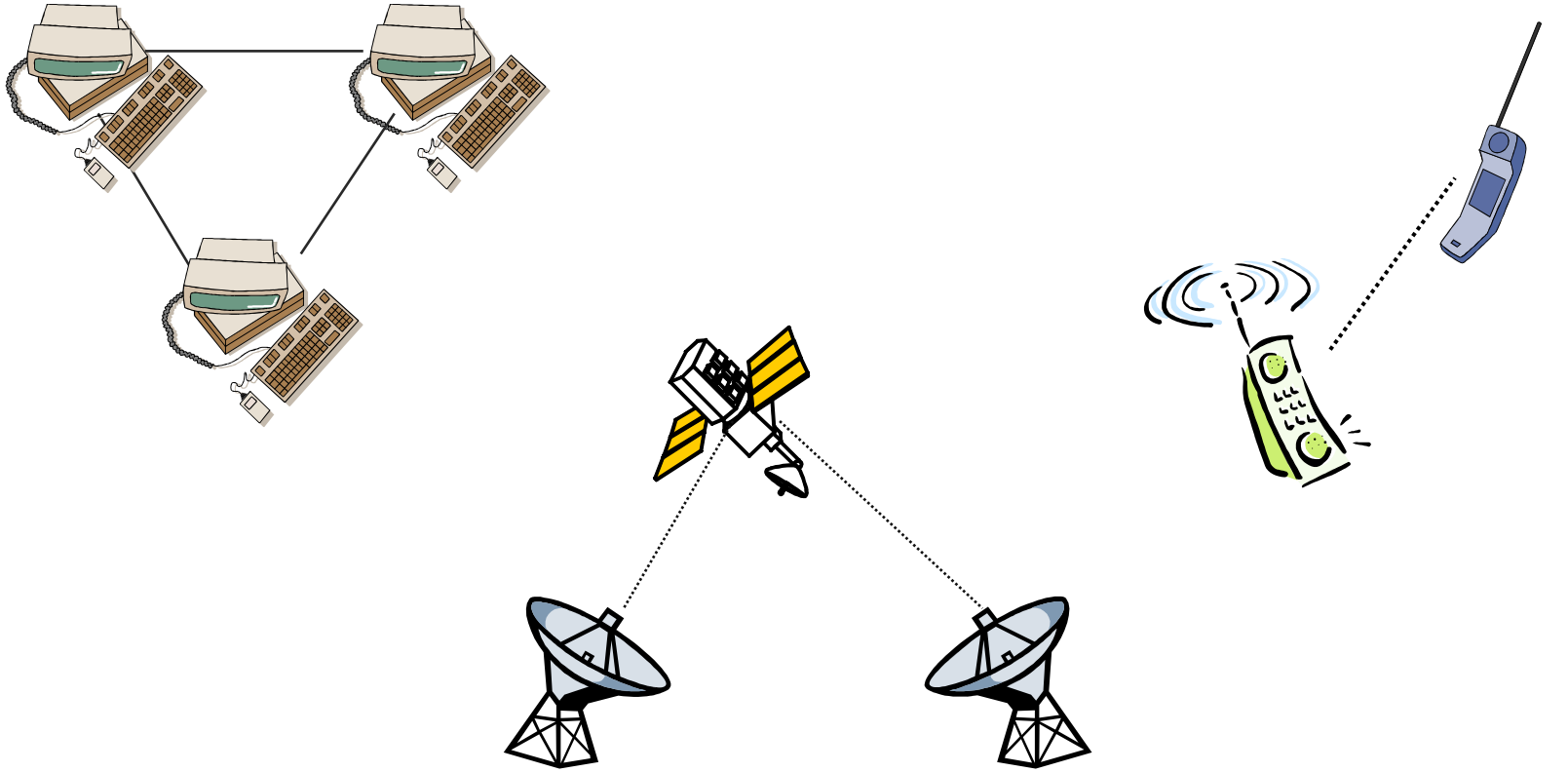
Source : Byte

Codage : introduction

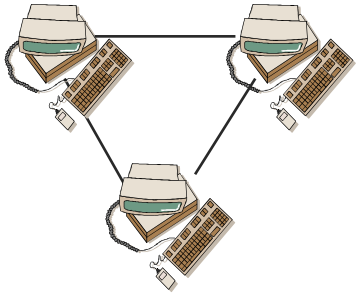
UNIVERSITÉ FRANÇOIS - RABELAIS TOURS

Objectif

Détection et correction les erreurs lors de la transmission de données



Transmission



Équipements fiables — taux d'erreur négligeable

Lignes de transmission sources principales d'erreurs

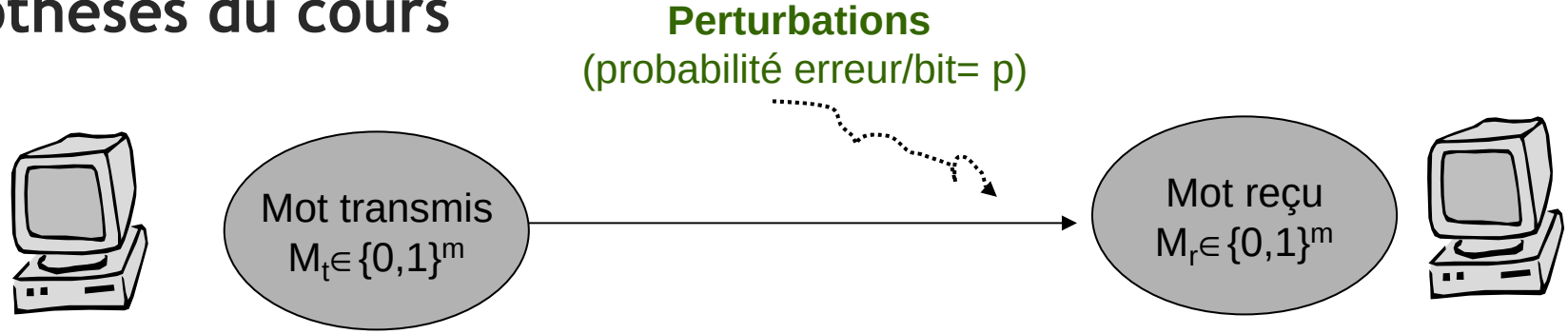
Erreurs de transmission

bruit thermique gaussien — agitation thermique des différents composants du système de transmission

bruit impulsif — étincelles de rupture dans les relais ou les équipements de commutation, foudre, surtensions sur les lignes du secteur

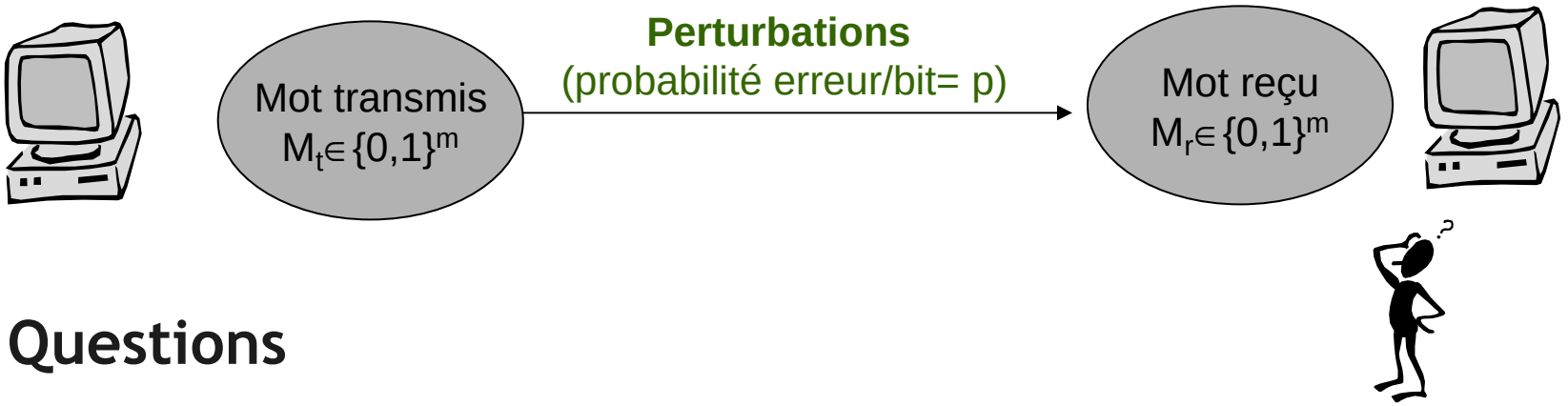
Taux d'erreurs bruts dépassant nettement ce qui est admissible pour l'utilisateur ⇨ détection et correction d'erreurs

Hypothèses du cours



- On se limite à la transmission d'**informations binaires.**
- On se limite aux **codages par blocs** : l'information à transmettre est découpée en mots binaire de longueur fixée (m)
- **Perturbation** sur la ligne de transmission :
  - **probabilité p** qu 'un bit soit mal transmis
  - transmission des différents bits assimilée à des phénomènes aléatoires indépendants
  - on suppose que la transmission n'affecte par la longueur du mot

Cas d'école



Questions

- probabilité que le mot reçu soit égal au mot transmis ?
- probabilité d'une erreur sur au moins un bit ?

Conséquences

- impossibilité de déceler a priori une erreur pour le receveur
- nécessité de la mise en place d'un codage des mots à transmettre pour faciliter la détection d 'erreur à la transmission

Codes correcteurs d'erreurs

Codes détecteurs

S'assurer que si le mot reçu n 'est pas le mot transmis,
le receveur est en mesure de le détecter

Codes correcteurs

S'assurer que si le mot reçu n 'est pas le mot transmis, le receveur est en mesure de le détecter et de le corriger pour retrouver l'information initialement transmise

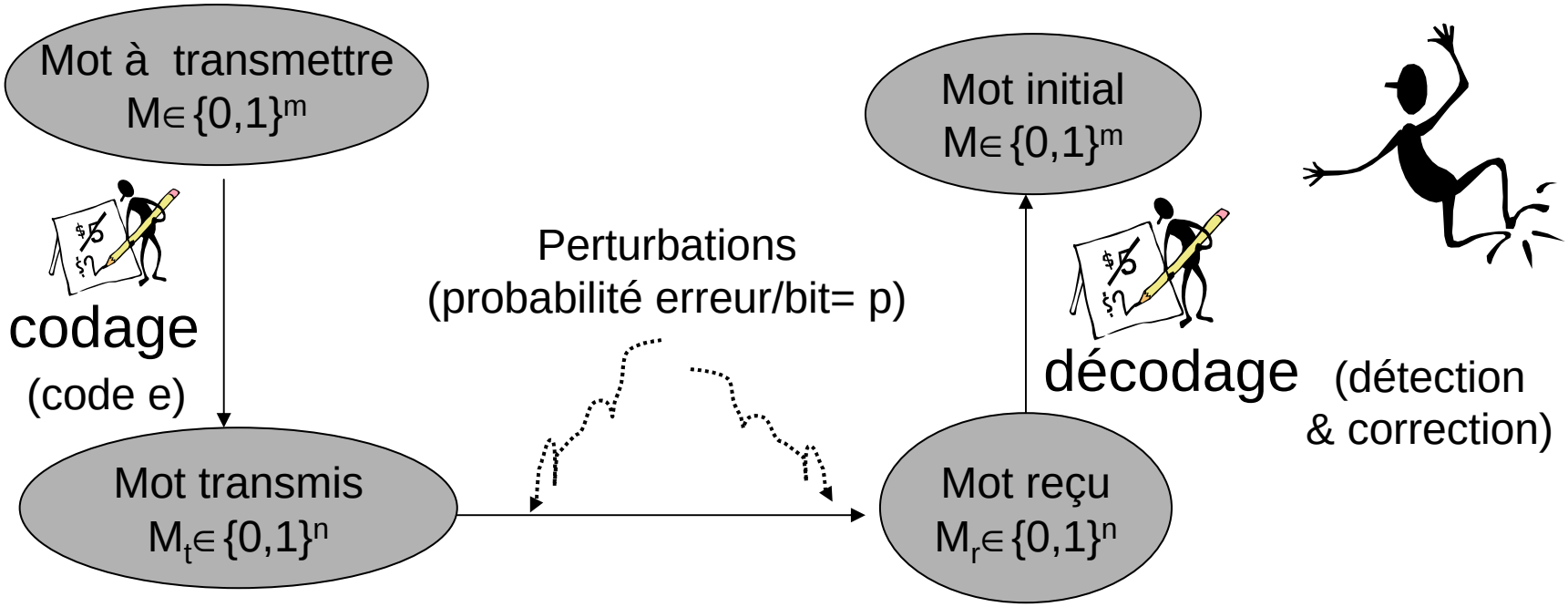
Principe

Introduction d'une redondance dans les données transmises

Exemples de codes basiques

- codage par contrôle de parité
- codage par répétition

Codage / décodage



- **Transmission sans erreur** — le mot reçu est un mot reconnu correct par le code (cf infra : *mot de code*)
- **Transmission avec erreur** — détection ou correction : recherche du mot transmis le plus probable

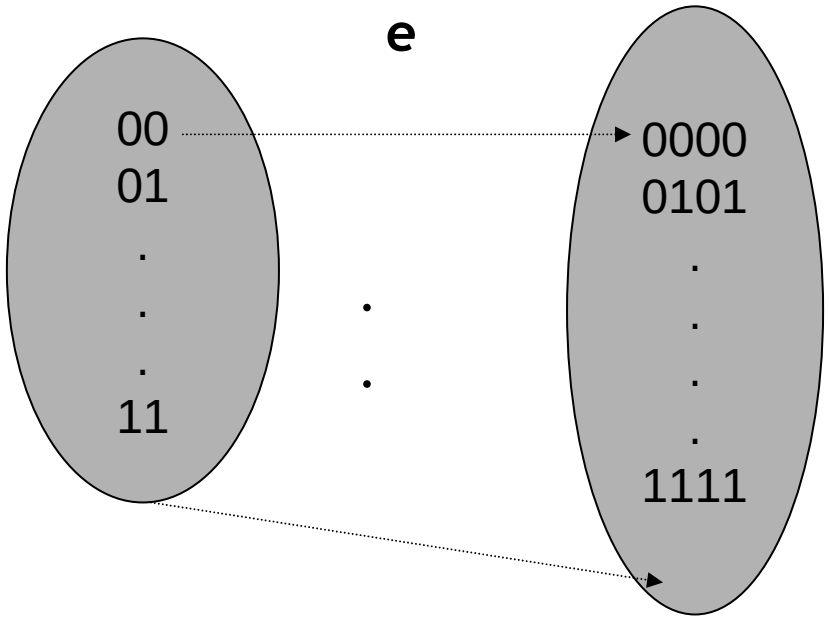
Attention, on peut recevoir un mot de code en présence d'erreur

Exemple : deux erreurs avec un contrôle de parité de bit

Code : définition

On appelle code de longueur n toute application **injective** tq

e : $\{0,1\}^m \rightarrow \{0,1\}^n$



- ✓ Les mots de f ($\{0,1\}^m$) sont appelés par définition **mots du code**
- ✓ **Rendement du code** — r = m / n

Exemple historique — sonde MARINER 9 (1972)

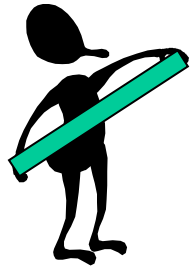
Distance de Hamming

Notion très utile pour déterminer la **capacité de détection ou de correction** d'un code (nombre d 'erreurs détectées ou corrigées)

Distance (rappel)

On appelle distance définie sur un ensemble E toute application d
$E \times E \rightarrow \Re$ tq $\forall a,b,c \in E$ on a

- $d(a,b) \geq 0$
- $d(a,b)=0 \Leftrightarrow a=b$
- $d(a,b)=d(b,a)$
- $d(a,c) \leq d(a,b)+d(b,c)$



Distance de Hamming

Soit a et b deux éléments de $\{0,1\}^n$, on appelle distance de Hamming de *a* à *b* l'application d tq :

$d: \{0,1\}^n \times \{0,1\}^n \rightarrow \mathbf{N}$
a,b d(a,b) = nombre de bits différents entre a et b

Distance minimale d'un code

Plus petite distance qui sépare deux mots de code différents

Théorème : code k-détecteur

Soit $e: \{0,1\}^m \rightarrow \{0,1\}^n$ un code de distance minimale d.

Alors le code e détecte tous les messages erronés dont le nombre d'erreurs est compris entre 1 et k si et seulement si $k < d$

Un tel code est dit code k-détecteur ($k = d-1$)

Exemples

- codage par contrôle de parité
- codage par répétition

Correction par maximum de vraisemblance

Soit $e: \{0,1\}^m \rightarrow \{0,1\}^n$ un code

- Si le mot reçu est un mot de code, on le laisse tel quel
- Si le mot reçu n'est pas un mot de code, on examine sa distance à tous les mots de code et on décide que le mot envoyé est l'antécédent du mot de code le plus proche (s'il n 'y en a qu 'un seul)

Théorème : code t-correcteur

Soit $e: \{0,1\}^m \rightarrow \{0,1\}^n$ un code de distance minimale d

Alors, dans le cas d'un décodage par maximum de vraisemblance, tout message qui comporte au plus t erreurs est correctement corrigé si et seulement si $d \geq 2t+1$

Un tel code est dit code t-correcteur avec $t = E((d-1)/2))$

Propriétés générales des codes

Borne de Hamming

On considère un code e : $\{0,1\}^m \rightarrow \{0,1\}^n$ qui est t-correcteur, de distance minimale d et de rendement r = n - m

$$2^m.\left(\sum_{i=0}^{i=t} C_n^t\right) \leq 2^n$$

avec $C_n^t = n!/t!(n-t)!$

Nb mots corrigeables à distance t

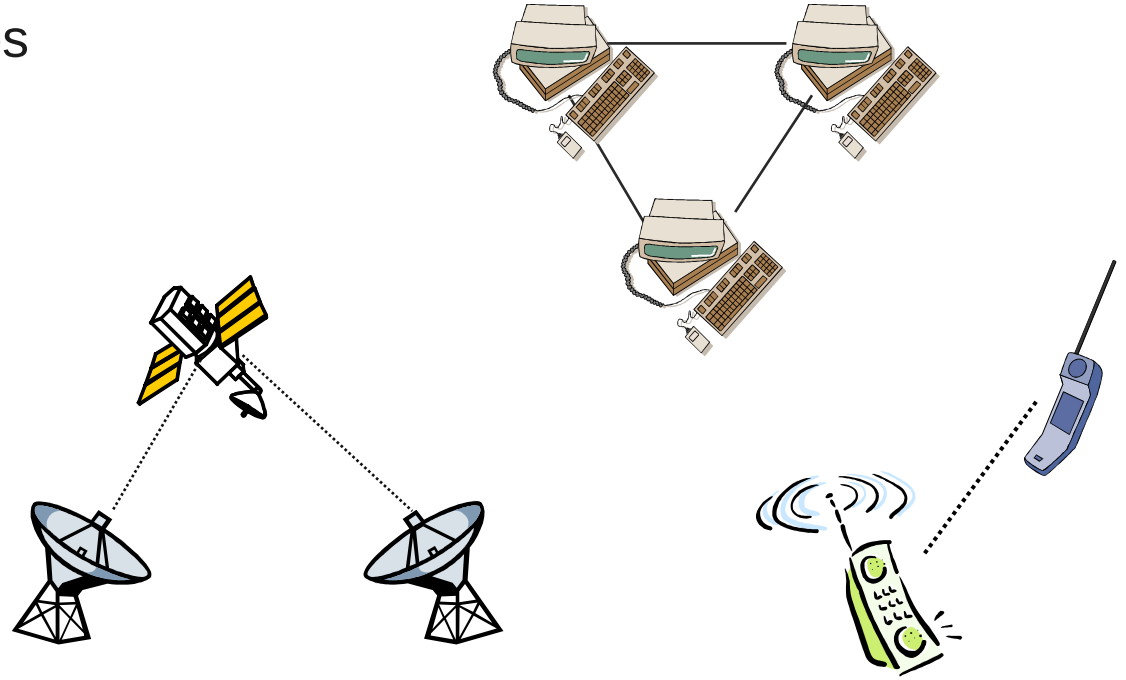
Nb mots transmis

Code parfait

On parle de **code parfait** si et seulement si il y a égalité dans l'inégalité de Hamming. Dans ce cas, tous les mots transmis sont des décodables à une distance t de correction

Différents types de code

- ✓ Codes systématiques
- ✓ Codes linéaires
- ✓ Codes polynomiaux
- ✓ Codes cycliques



⇨ Choix du code suivant les besoins, le domaine d'utilisation (nature des perturbations, taux d'intégrité visé…)

⇨ connaissance des propriétés (capacité de détection, rendement…) théoriques des codes

Bibliography

Main references

B. Martin (2004) Codage, cryptologie et applications. PPUR (Presses Polytechniques et Universitaires Romandes), Lausanne, Suisse. ISBN 2-88074-569-1

Exercice

Etude d'un code

On considère l'application e définie de $\{0,1\}^3$ vers $\{0,1\}^8$ donnée par le tableau suivant:

b	e(b)
000	00000000
001	10111000
010	00101101
011	10010101
100	10100100
101	10001001
110	00011100
111	00110001

1- Étude du code

a) S'agit-il d'un code?

b) Quelle est sa distance minimale ?

c) Que peut-on affirmer sur le nombre d'erreurs détectées?

d) Que peut-on affirmer sur le nombre d'erreurs corrigées?

2- Comment décoder le mot reçu (11100101) par maximum de vraisemblance?

3 – Ce code est-il un code parfait ?