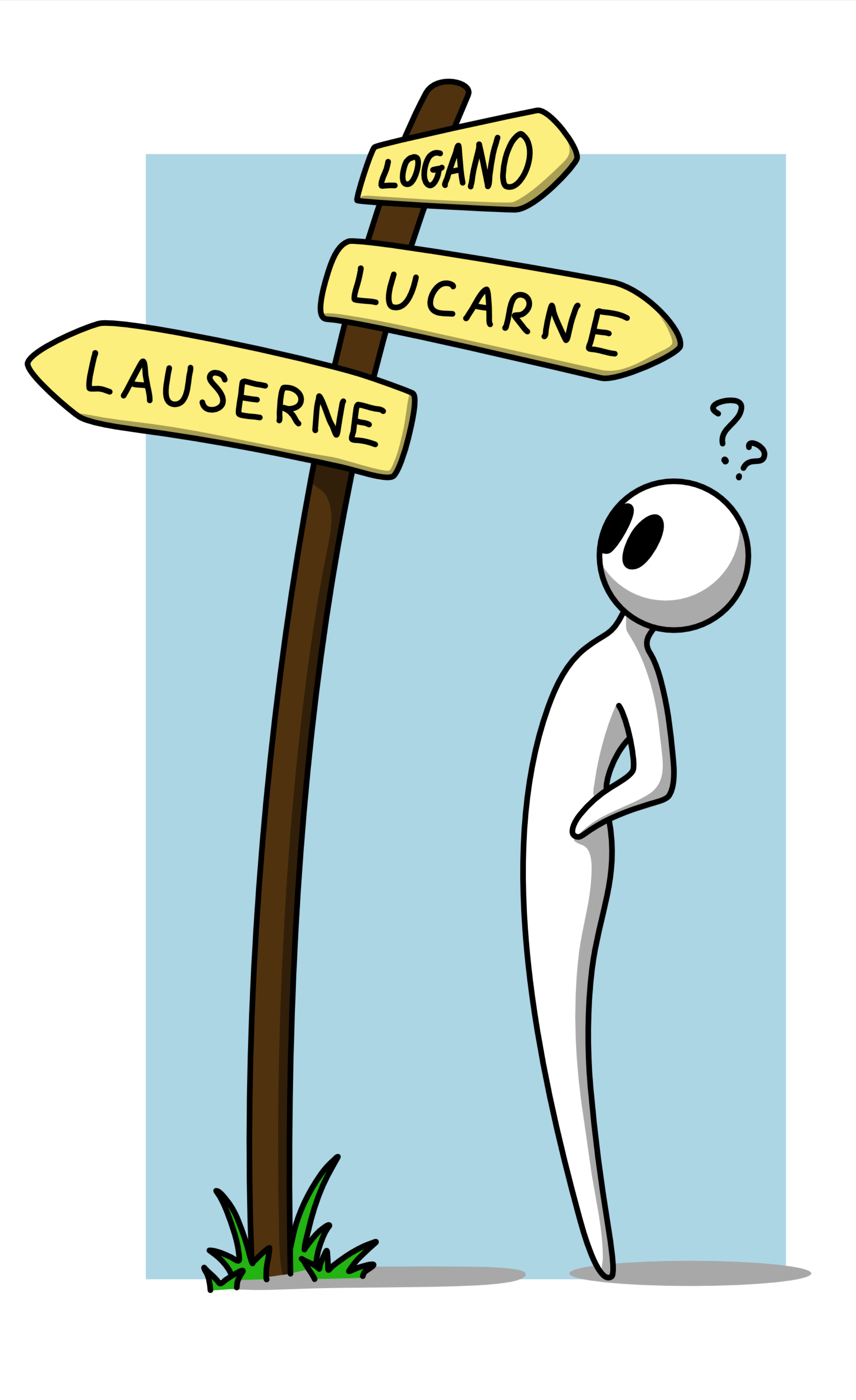
EPFL



Information, Calcul et Communication

Correction d’erreurs : principes de base

Olivier Lévêque

École polytechnique fédérale de Lausanne

Comment transmettre des données ?

Il existe deux moyens de transmission :

- A travers le **temps** : enregistrement sur un support
  (et les données peuvent lues plus tard)
- A travers **l'espace** :
  - par câble électrique ou fibre optique (téléphonie, internet)
  - dans l'air (téléphonie sans fil, wifi)

Problème commun : des erreurs surviennent régulièrement !

- lors de l'écriture ou de la lecture de données
- lors de la transmission par câble électrique, fibre optique ou onde électromagnétique

Exemple

- Vous désirez communiquer la direction à prendre (N,S,E,W) à un ami perdu.

Code binaire utilisé :

N	S	E	W
11	10	01	00

- Vous indiquez à votre ami d'aller au nord en envoyant 11.

- Si votre ami reçoit :

11 Tout se passe bien : votre ami se dirige vers le nord.

1? Un **effacement** survient : votre ami ne sait pas s'il doit se diriger vers le nord ou vers le sud.

10 Une **erreur** survient : votre ami se dirige alors vers le sud !

EPFL

Solution : ajouter de la redondance !

« *Mieux vaut prévenir que guérir* » :

→ Ajouter de la redondance au message que l’on veut envoyer

→ *Mieux :* Ajouter **un minimum de redondance,**
pour prévenir **un maximum d’erreurs**

Exemples de redondance dans la vie courante

- Un père à ses enfants :

« Mettez vos chaussures… mettez vos chaussures, j'ai dit ! »

- Vous à votre ami :

« Pour venir chez moi, c'est simple :
tu tournes dans la 2e rue à droite après le feu rouge, juste après la station-service ;
j'habite au numéro 3, dans l'immeuble bleu avec les grands balcons »

Les informations en rouge sont redondantes (donc inutiles en théorie…).

Corriger un effacement

- Codage par répétition

N	S	E	W
1111	1100	0011	0000

  - 11?1 → N
  - Simple mais coûteux : il faut envoyer 4 bits au lieu de 2 !

- Bit de parité

N	S	E	W
110	101	011	000

  - 1?0 → N
  - Le dernier bit correspond à la somme modulo 2 des deux premiers.

Pour envoyer un message de n bits et corriger un effacement, il faut ajouter :

$n\ bits$	$1\ bit\ de\ parit\acute{e}$

La taille finale du message est :

$2n\ bits$	$n + 1\ bits$

Corriger une erreur

- Doubler chaque bit
  - 1101 → N ou S ??

N	S	E	W
1111	1100	0011	0000

- Tripler chaque bit
  - 111101 → N

N	S	E	W
111111	111000	000111	000000

On applique ici la **règle de la majorité** : 111101 → N 111100 → S

Mais à nouveau, cette solution est très coûteuse :

Chaque bit est triplé, on aura donc besoin de $3n$ bits
pour envoyer un message de n bits à l'origine !

Corriger une erreur

Un meilleure méthode : ajouter *plusieurs* bits de parité

Voici un essai :

- Lorsque $n = 4$: on ajoute 2 bits de parité
  - le premier indique la parité de la somme des bits 1 et 2
  - le deuxième indique la parité de la somme des bits 1 et 3

Exemple : Pour envoyer 1101, on envoie 110101.

- Si on reçoit 100101 :
  - bit 1 et 2 : $1 + 0 = 1 \neq 0$
  - bit 1 et 3 : $1 + 0 = 1$

On déduit que le bit 2 est faux :
Le message d'origine est donc 110101

Corriger une erreur (suite)

Un meilleure méthode : ajouter *plusieurs* bits de parité

Voici un essai :

- Lorsque $n = 4$: on ajoute 2 bits de parité
  - le premier indique la parité de la somme des bits 1 et 2
  - le deuxième indique la parité de la somme des bits 1 et 3

Problème : Pour envoyer 1101, on envoie toujours 110101.

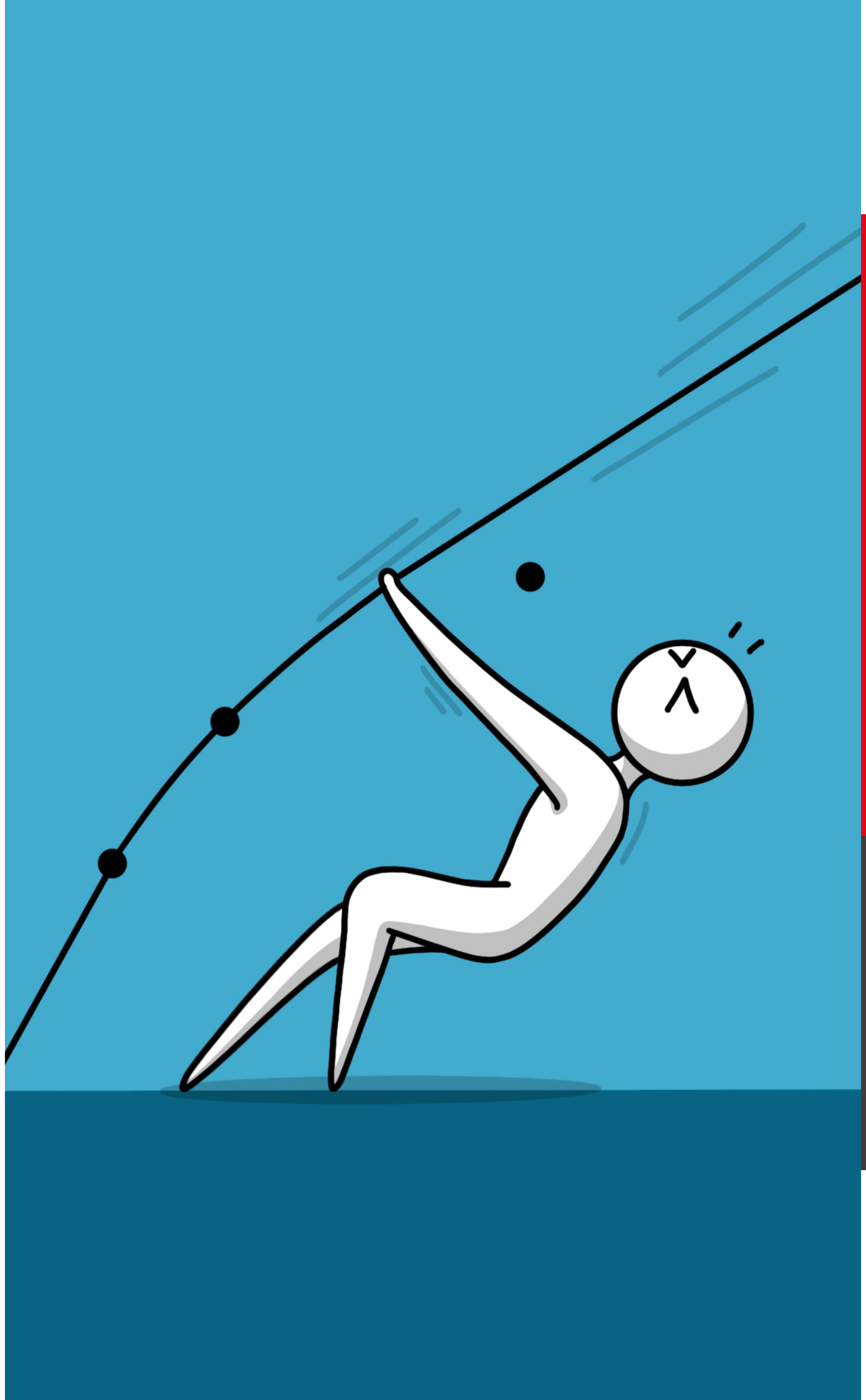
- Si l'erreur survient sur le bit 4, on reçoit alors 110001, mais les deux bits de parité sont corrects → erreur indétectée !

- Pour réparer ce problème, *3 bits de parité* sont en fait nécessaires → code de Hamming

Corriger plusieurs effacements ou erreurs ?

Tout l'art de la théorie du codage (70 ans d'histoire…) consiste à choisir parcimonieusement les bits de parité pour corriger un nombre maximum d'erreurs…

EPFL



Information, Calcul et Communication

Les codes de Reed-Solomon

Olivier Lévêque

École polytechnique fédérale de Lausanne

Introduction

- Deux personnes (disons Alice et Bob) cherchent à communiquer, mais une proportion non-négligeable des informations transmises par Alice sont effacées/bruitées avant d’être reçues par Bob:

- Les codes de Reed-Solomon sont un bon moyen de gérer une telle situation.
- Pour simplifier leur description, nous allons supposer qu’Alice et Bob travaillent avec des moyens de communication capables de manipuler des *nombres réels* (et non des bits).

Protocole de communication : Envoi

- *Avant de communiquer*, Alice et Bob se mettent d'accord sur un ensemble de n nombres réels *tous différents* :

$$\{t_1, t_2, t_3, \dots, t_n\}$$

- Alice désire envoyer un message x composé d'une suite de nombres réels :

$$x = \{x_1, x_2, x_3, \dots, x_k\} \quad \text{avec} \quad k < n$$

- Elle définit le polynôme suivant :

$$P(t) = \sum_{i=1}^{k} x_i \cdot t^{i-1} \qquad \deg(P) \leq k - 1$$

- et envoie finalement le message :

$$y = \{y_1 = P(t_1),\ y_2 = P(t_2),\ y_3 = P(t_3), \dots,\ y_n = P(t_n)\}$$

Protocole de communication : Réception

- Bob reçoit le message y. On suppose que $n - k$ nombres du message sont effacés; il reçoit donc que k nombres de y. Pour simplifier, admettons de plus que Bob ne reçoive que les k premiers nombres $\{y_1, y_2, y_3, \dots, y_k\}$.
- Le but de Bob est de retrouver $\{x_1, x_2, x_3, \dots, x_k\}$ à partir de $\{y_1, y_2, y_3, \dots, y_k\}$

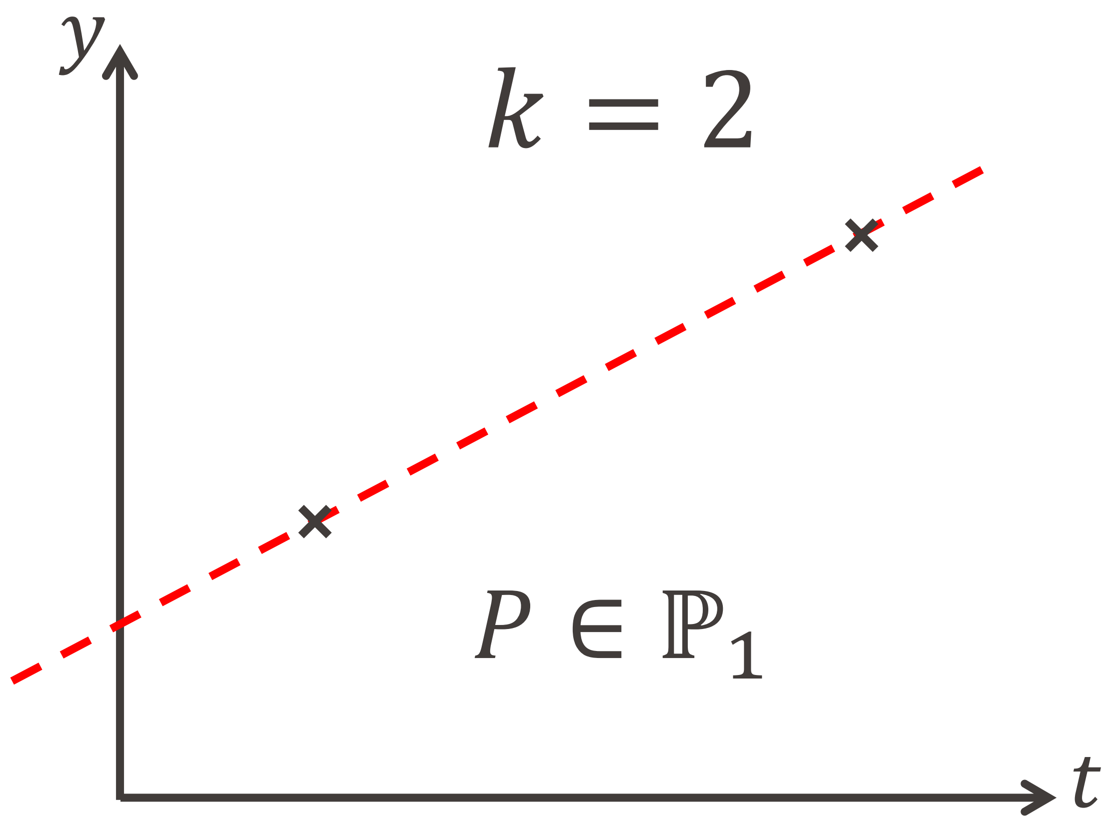
Rappelons que $y_j = P(t_j) = \sum_{i=1}^{k} x_i \cdot t_j^{\,i-1}$ pour $1 \leq j \leq k$

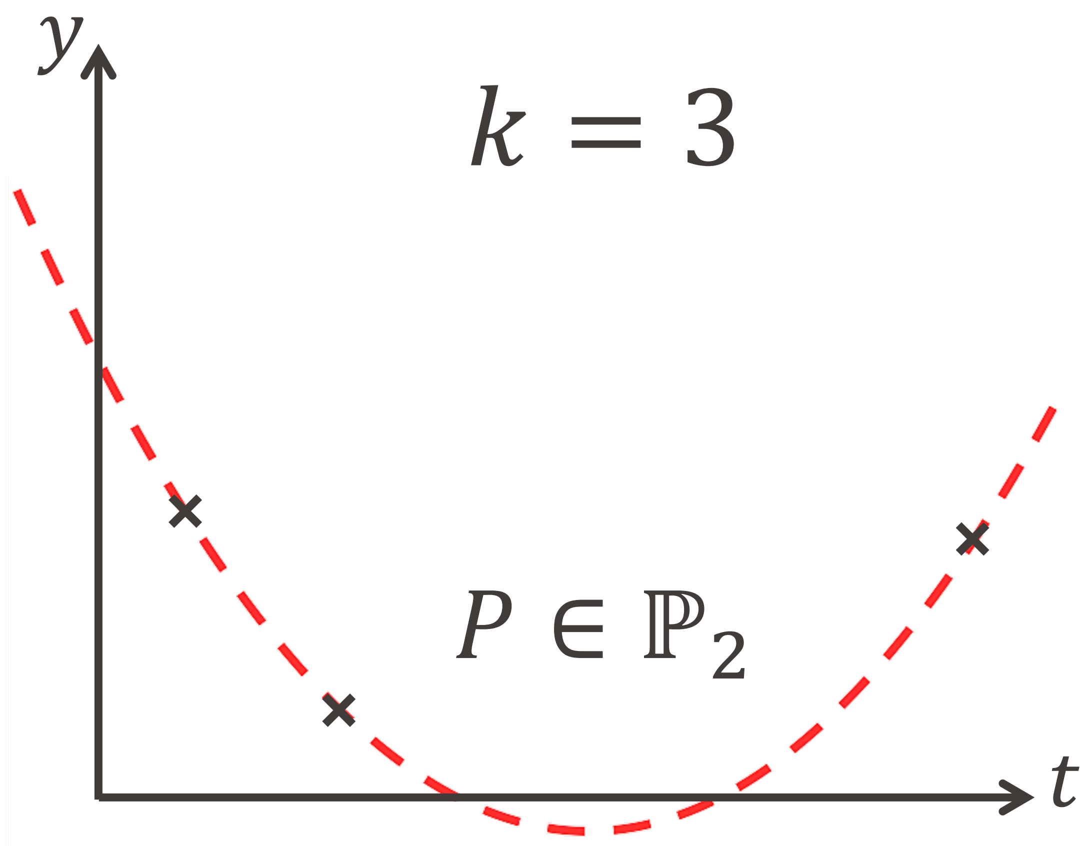
- Il s'agit donc de résoudre un système linéaire de k équations à k inconnues!

Exemple pour $k = 3$ et $t = \{1, 2, 3\}$: $\begin{cases} x_1 \cdot 1 + x_2 \cdot 1 + x_3 \cdot 1 = y_1 \\ x_1 \cdot 1 + x_2 \cdot 2 + x_3 \cdot 4 = y_2 \\ x_1 \cdot 1 + x_2 \cdot 3 + x_3 \cdot 9 = y_3 \end{cases}$

Une autre façon de visualiser le problème

- En réalité, Bob reçoit les coordonnées des points (t_j, y_j) $\forall\, 1 \leq j \leq k$. Il s'agit donc de trouver l'unique polynôme P tel que $\deg(P) \leq k-1$ passant par les k points différents.



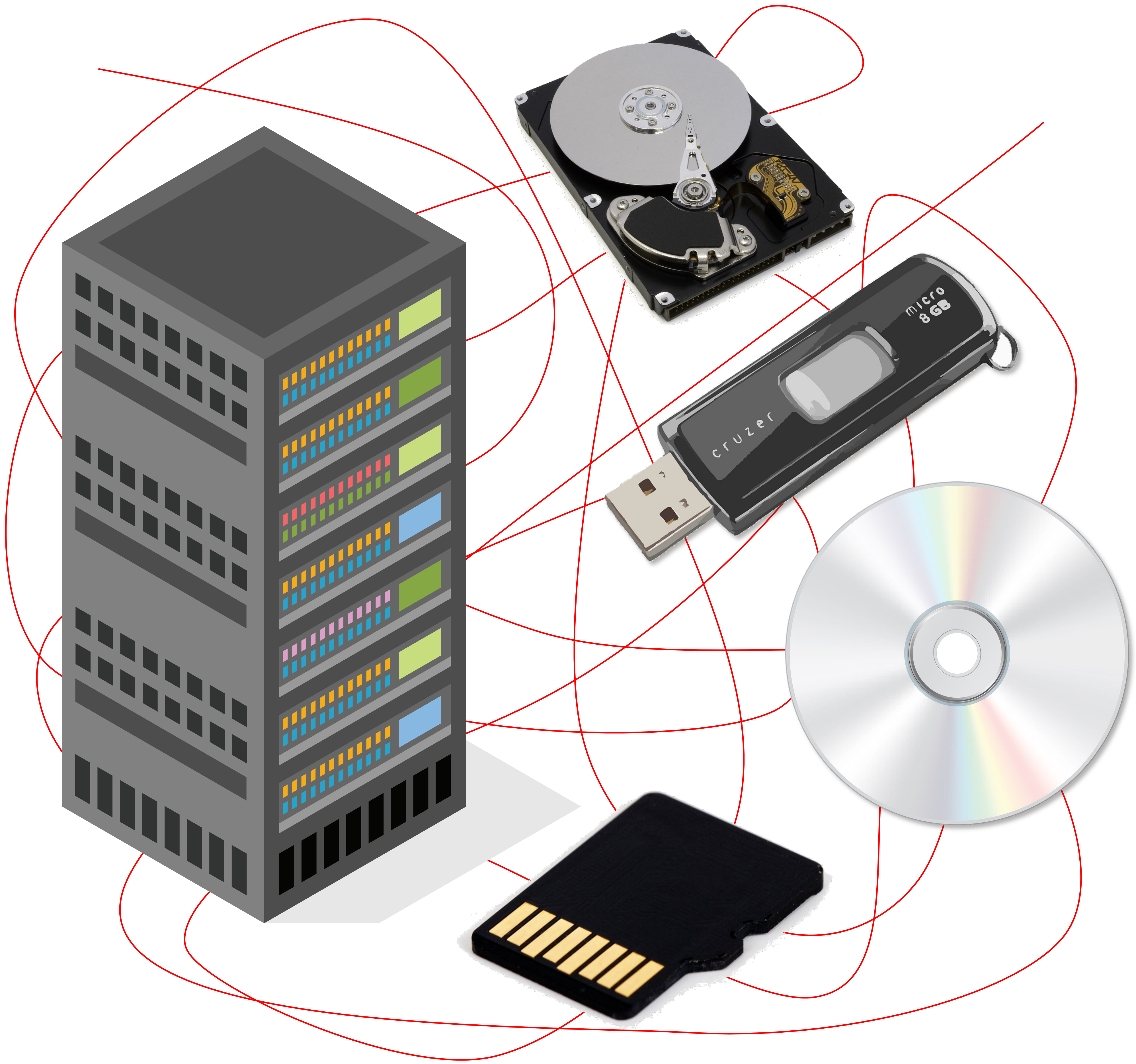


- Ainsi, Bob retrouve le message envoyé x (composé des coefficients du polynôme P).

En pratique

- On ne travaille pas avec des nombres réels, mais avec des nombres entiers modulo p (où p est un nombre premier) : $\{0,1,2,\ldots,p-1\}$, ou plus généralement des nombres appartenant à un groupe fini.
- Et les mêmes principes concernant les polynômes s'appliquent dans ce cadre.

Application : Stockage de données et codes-barres 2D



QR code

Aztec Code

MaxiCode

Data Matrix

PDF-417