Securing multimedia content delivery

Digital watermarking part. I

gaetan.leguelvouit@b-com.com

b com

Trois définitions

La **dissimulation d'information** – *data hiding* – est l'art de cacher des informations dans un document hôte.

La **stéganographie** – *steganography* – consiste à établir un canal de communication secret en dissimulant des informations.

- du grec *steganos* : caché derrière quelque chose,
- et *graphia* : écriture

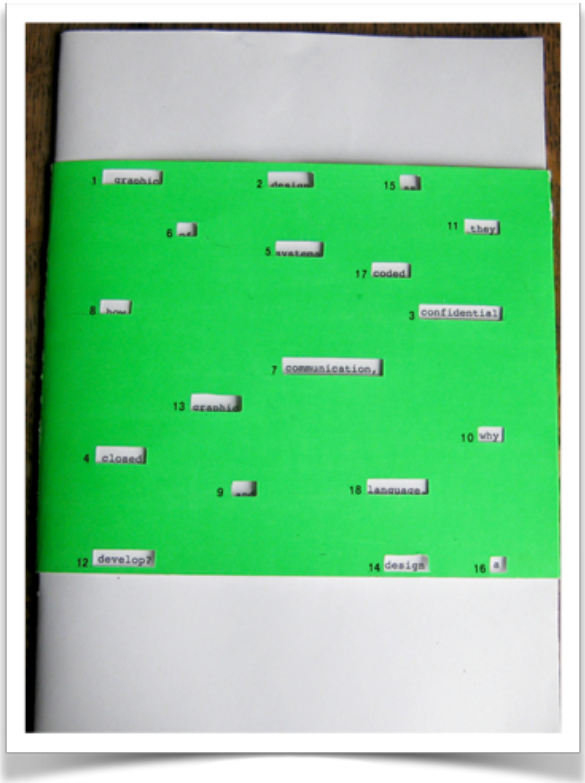
Le **tatouage numérique** – *digital watermarking* – consiste à insérer des informations robustes dans un document.

Historique de la stéganographie

- Hérodote et l'esclave tatoué au $5^{ème}$ siècle av. J.C.
- Les boules de soie chinoise
- L'encre sympathique dès le 1^{er} siècle av. J.C., avec du lait ou du jus de citron
- Les acrostiches : voir Horace de Corneille

Historique de la stéganographie

- La **grille de Cardan** (vers le $16^{ème}$ siècle) : une feuille découpée révèle le message secret enfoui dans un text anodin
- Essor de la **stéganographie numérique** : utilisation d'images, des blocs non pleins des systèmes de fichiers, des options de formats, etc.



Historique du tatouage

- Les premiers filigranes apparaissent en Italie, pour identifier les fabriques de papier (1282)
- Au $18^{\text{ème}}$ siècle, utilisés pour **authentifier** la monnaie et cartains documents ; apparition des premières **contrefaçons**
- Tatouage de musique en 1954, avec insertion de morse
- Tatouage numérique en 1988

Tatouage numérique

Documents : texte, images, vidéo, audio, (2D,3D) objets, dans dessinée, cartes, bases de données, logiciels, . . .

Des applications diverses

- communication : cachée, synchronisation, auto-correction
- protection des droits (premier usage, l'essentiel de ce cours)
- intégrité
- traçabilité
- protection contre la copie
- contenu enrichi (métadonnées, index, etc.)
- ...

Remarques

- Dans la stéganographie, le document hôte n'a aucune valeur ; il doit juste paraitre anodin
- Pour le tatouage, le document doit conserver toute sa valeur (ne pas être dégradé) ; les informations marquées concernent généralement le doument (authentification, identificaion)

- Pour la suite, nous nous focalisons sur le tatouage – si possible – robuste et invisible

Vecteur hôte

- Le document hôte est le support du marquage
- Mais généralement, le document n'est pas modifié directement : on utilise une sous-partie ou une autre représentation, qui définit le **vecteur hôte**
  - Ex. : les coefficients haute fréquence de la tranformée de Fourier d'une image, 1 échantillon sonore sur 10 d'un son, etc.
- On note ce vecteur $\mathbf{x} \in \mathbb{R}^m$

Message à insérer

- Le message est l'information qui sera transmise via le tatouage
- Pour simplifier, on considère que le message est binaire :

$$\mathbf{m} \in \{0,1\}^n$$

- La dimension n est le nombre de **bits utiles** du tatouage

Marque

- La marque (le tatouage) est la mise en forme du message. C'est une forme de **codage du message**. On note

$$\mathbf{w} \in \mathbb{R}^m$$

- Le tatouage se fait en ajoutant $\mathbf{w}$ au vecteur hôte $\mathbf{x}$: c'est-à-dire $\mathbf{y} = \mathbf{x} + \mathbf{w}$

Distortion d'insertion

- On peut quantifier la distortion d'insertion du tatouage par l'erreur quadratique moyenne

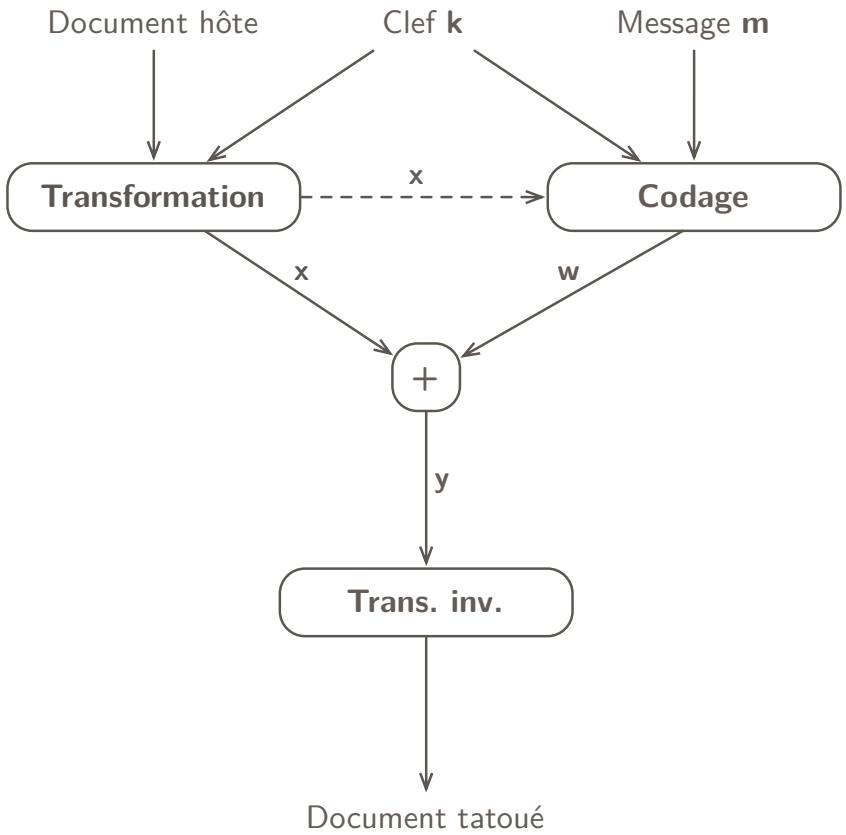
$$\mathrm{MSE}(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{m} (\mathbf{x}[i] - \mathbf{y}[i])^2$$

- . . . À partir de laquelle on définit le PSNR

$$\mathrm{PSNR} = 10 \log_{10} \left[\frac{d^2}{\mathrm{MSE}} \right]$$

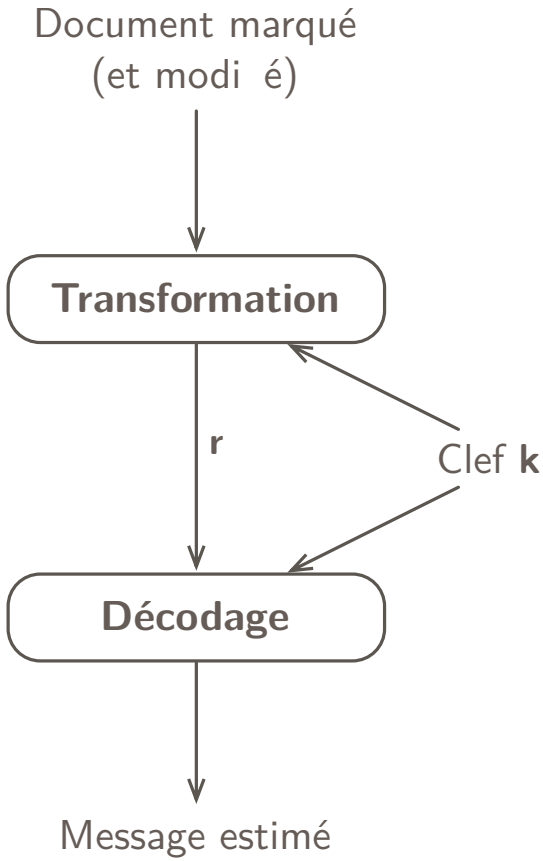
Fonction d'insertion

- La fonction de codage de **m** peut prendre en compte le vecteur hôte **x** : c'est le **codage informé** (voir partie suivante)
- Transformatio et codage utilisent une clef secrète **k**



Fonction de lecture

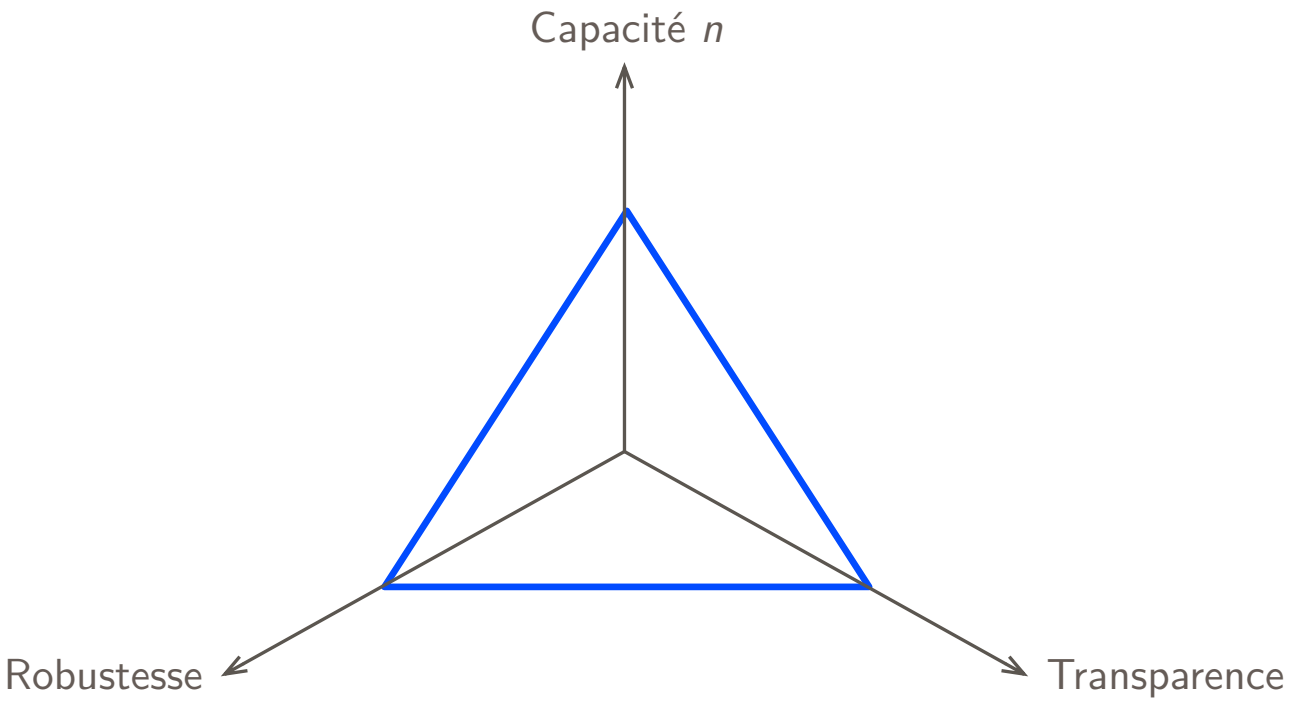
- La fonction de transformation est le même que lors de l'insertion
- La fonction de décodage doit être capable de **corriger les erreurs** dues à la modification du document marqué
- La clef secrète **k** est la même que lors de l'insertion : tatouage **symétrique**



Sur les attaques

- Une attaque est la mofication – intentionnelle ou pas – du document tatoué, qui se traduit par la modification de $\mathbf{y}$. Par exemple :
  - ajout de bruit gaussien
  - conversion N/A,
  - compression avec perte
- Souvent modélisée par un ajout de bruit $\mathbf{r} = \mathbf{y} + \mathbf{z}$.
- La force de l'attaque est quantifiée par l'EQM ou le PSNR

Compromis



Capacité

Quelques ordres de grandeur :

- traçage de copie : 22 bits
- indexation : 128 bit
- enrichissement de contenu : plusieurs centaines de bits/image
- synchronisationaudio-video : 8 bits/image
- ...

Sur la clef secrète

Le **principe de Kerckoff** : la sécurité doit uniquement s'appuyer sur un paramètre inconnu de l'attaquant, c'est-à-dire une clef :

- tous les utilisateurs légitimes partagent le même secret
- on suppose que les attaquants connaissent l'algorithme

En pratique, la clef est utilisée comme graine d'un générateur pseudo-aléatoire :

- pour sélectionner les échantillons qui seront marqués
- pour introduire de l'aléa dans les fonctions de codage et décodage

Représentation d'une image

Un tableau de pixels, chacun typiquement encodé sur 3 octets :

- (R, G, B) : rouge, vert et bleu
- (Y, U, V) : Y est la luminance, U et V les chrominances.

$$\begin{pmatrix} Y \\ U \\ V \end{pmatrix} = \begin{pmatrix} 0,299 & 0,587 & 0,144 \\ 0,596 & -0,274 & -0,322 \\ 0,211 & -0,523 & 0,312 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix}$$

Tatouage LSB

1. Sélectionner un ensemble de pixels
2. Remplacer le bit de poids faible par le bit du message à insérer

Très simple à implémenter, bonne capacité, mais faible robustesse (la moindre modification va modifier les LSB).

Patchwork Bender [95], Pitas [96]

Prenons deux ensembles A, B de p pixels (sélection grâce à une clef), avec les luminances $\{a_1, \ldots, a_p\}$ et $\{b_1, \ldots, b_p\}$.
En moyenne :

$$S = \frac{1}{n} \sum_{i=1}^{p} (a_i - b_i) \simeq 0$$

Patchwork Bender [95], Pitas [96]

Prenons deux ensembles A, B de p pixels (sélection grâce à une clef), avec les luminances $\{a_1, \ldots, a_p\}$ et $\{b_1, \ldots, b_p\}$.
En moyenne :

$$S = \frac{1}{n} \sum_{i=1}^{p} (a_i - b_i) \simeq 0$$

Idée : modifions les luminances $a'_i = a_i + C$, $b'_i = b_i - C$.

$$S' = \frac{1}{n} \sum_{i=1}^{p} (a'_i - b'_i) = S + 2C \simeq 2C$$

Patchwork Bender [95], Pitas [96]

Prenons deux ensembles A, B de p pixels (sélection grâce à une clef), avec les luminances $\{a_1, \ldots, a_p\}$ et $\{b_1, \ldots, b_p\}$.
En moyenne :

$$S = \frac{1}{n} \sum_{i=1}^{p} (a_i - b_i) \simeq 0$$

Idée : modifions les luminances $a'_i = a_i + C$, $b'_i = b_i - C$.

$$S' = \frac{1}{n} \sum_{i=1}^{p} (a'_i - b'_i) = S + 2C \simeq 2C$$

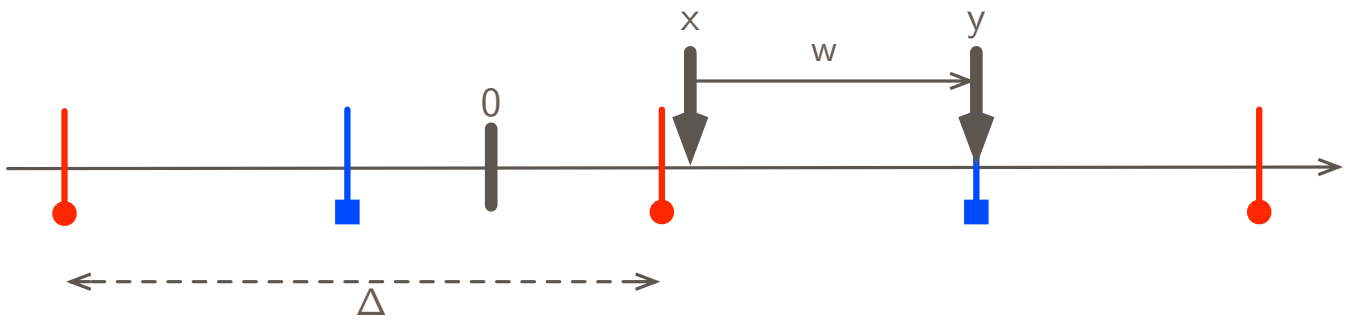
Nous introduisons un biais statistique.

Quantification scalaire

- Les éléments de $\mathbf{x}$ sont quantifiés en fonction du bit à insérer :

$$\mathbf{y}[i] = \Delta \times \text{round}\left(\frac{\mathbf{x}[i] + \mathbf{d}[i] + \mathbf{m}[i]\Delta/2}{\Delta}\right) - \mathbf{d}[i] - \frac{\mathbf{m}[i]\Delta}{2}$$

- Le vecteur $\mathbf{d}$ est le *dithering* et il est généré à partir de la clef
- Δ détermine la puissance de la marque
- La lecture de la marque est aussi une fonction d'arrondi



Dans JPEG Koch [95]

Voir tableau.

Étalement de spectre

- Soit une matrice **G** de dimensions $n \times m$ composée de ± 1, générée pseudo-aléatoirement depuis une clef
- L'objectif est de moduler ces porteuses statistiquement orthogonales pour construire la marque **w** (les 0 de **m** sont remplacés par -1) :

$$\mathbf{w}[i] = \alpha \sum_{j=1}^{n} \mathbf{m}[j] \times \mathbf{G}[i][j]$$

- α détermine la puissance de la marque

Étalement de spectre

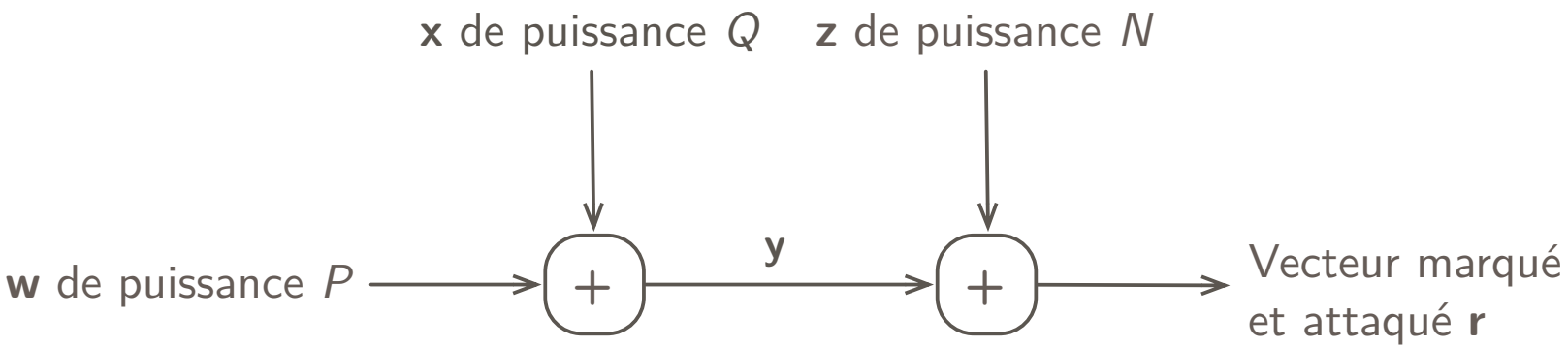
- La lecture se fait en corrélant les porteuses de $\mathbf{G}$ avec le vecteur tatoué-attaqué $\mathbf{r}$:

$$\mathbf{c}[i] = \sum_{i=1}^{m} \mathbf{r}[i] \times \mathbf{G}[i][j]$$

- Si la corrélation est positive, on décode 1, sinon 0

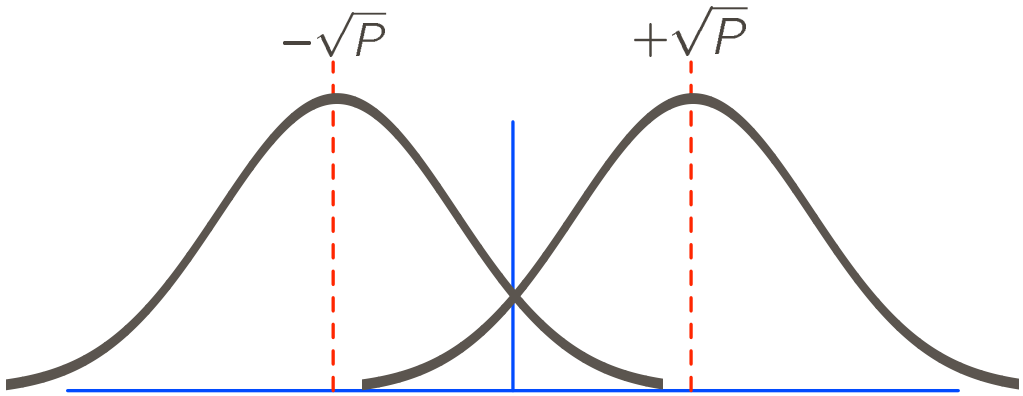
Canal de communication

- Le tatouage chercher à transmettre une information via une marque bruitée par un vecteur hôte et une attaque
- On retrouve les caractéristiques d'un canal de communication



Retour sur l'étalement de spectre

- Le théorème central limite montre que l'étalement de spectre définit un **canal gaussien** avec $P = m^2\alpha^2$



- On en déduit la probabilité d'erreur par bit

$$P_e = \frac{1}{2} \times \text{ercf}\left[\sqrt{\frac{P}{2(Q+N)}}\right]$$

Ordres de grandeur

- Marquage de 12 coefficients par bloc DCT de l'image Lena : on observe $Q = 6,8 \times 10^7$
- Puissance de la marque avec $\alpha = 1/2$: $P = 6,0 \times 10^8$
- Attaque par ajout de bruit gaussien t.q. PSNR $= 30$ dB : $N = 3,2 \times 10^6$

Probabilité d'erreur par bit : $\mathrm{P_e} = 2,34 \times 10^{-3}$

Notion de capacité

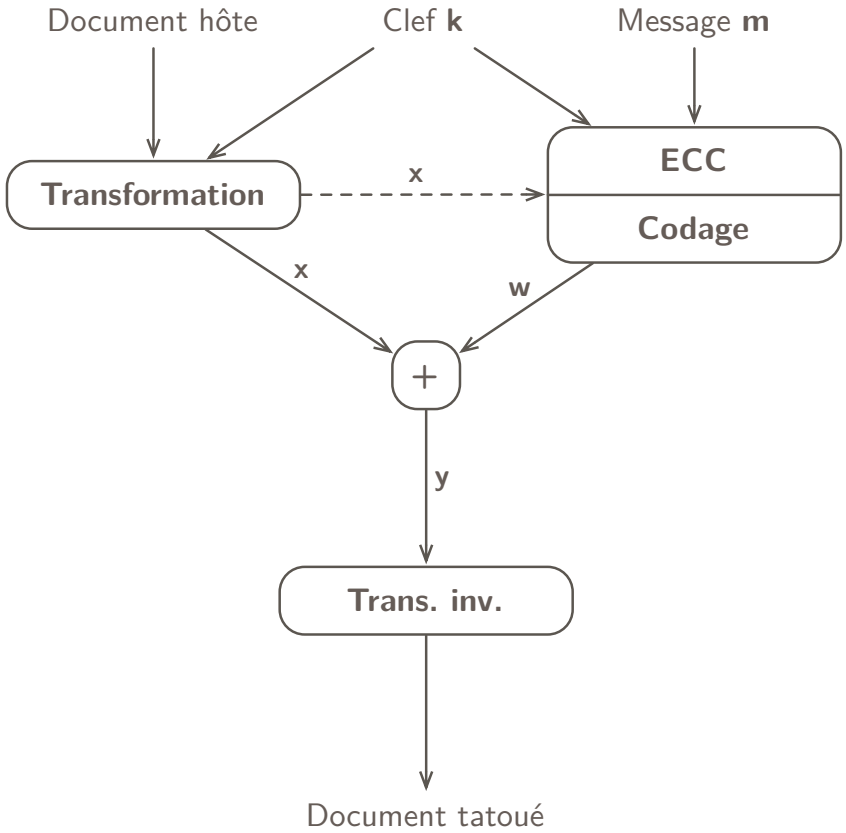
- La capacité d'un canal est le nombre de bits utiles par symbole que l'on peut transmettre sans erreur
- Pour le canal gaussien (et donc le tatouage par ÉdS) :

$$C = \frac{1}{2} \log_2 \left[1 + \frac{P}{Q + N} \right]$$

- C'est le rapport n/m maximal

Codage canal

- Pour s'approcher de cette borne (et donc améliorer la robustesse), on utilise des codes correcteurs d'erreurs (ECC)
- Les meilleurs codes sont très proches de la borne de capacité (i.e. $P_e < 10^{-5}$) : turbo-codes, LDPC

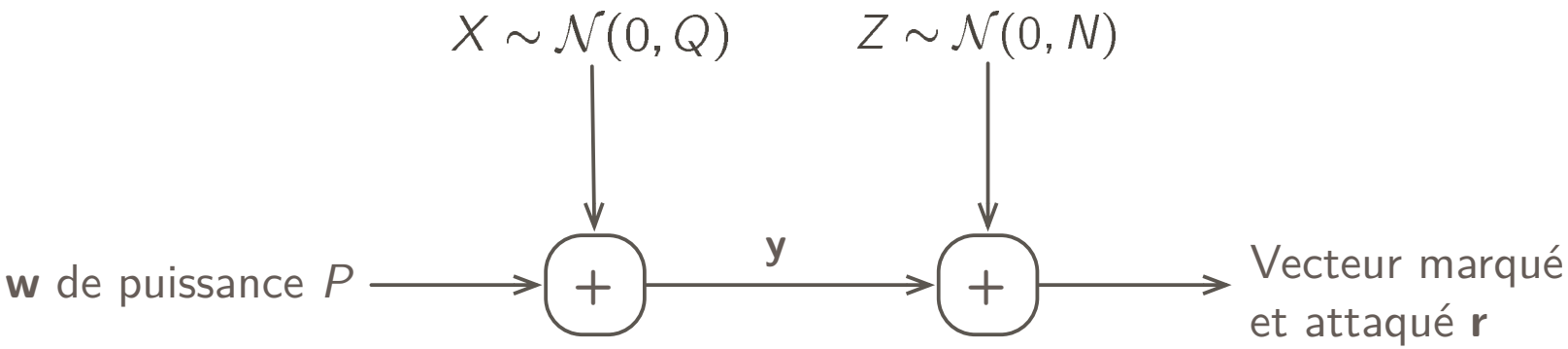


Peut-on faire mieux ?

[Spoiler]

Oui.

Schéma idéal de Costa



- L'information adjacte – *side information* – est l'information disponible à l'encodage : ici le vecteur hôte **x**
- En 1983, Costa a montré pour le cas gaussien que cette information n'avait aucune influence sur la capacité du canal :

$$C = \frac{1}{2} \log_2 \left[1 + \frac{P}{N} \right]$$

- Il propose une démonstration constructive

Les registres défaillants

- Un registre de 3 bits
- Avant l'écriture, un des 3 bits est bloqué
  - la position défaillante et sa valeur change aléatoirement
  - le récepteur ne sais pas quelle position était bloquée lors de l'écriture

Combien de bits peut-on transmettre de façon fiable ?

Les registres défaillants

Code correcteur : **dictionnaire** qui associe un **mot de code** à un message

- Ajoute de la redondance pour pouvoir rattraper les erreurs de transmission
- Décodage : recherche du mot de code le plus proche (le plus probable) des informations reçues

Les registres défaillants

Si on utilise le dictionnaire $0 \rightarrow 000$ et $1 \rightarrow 111$

- distance minimale du code $= 3$
- on peut corriger une erreur, ce qui convient à notre problème

Si on utilise le dictionnaire $00 \rightarrow 000$, $01 \rightarrow 001$, $10 \rightarrow 100$ et $11 \rightarrow 111$

- distance minimale du code $= 1$
- on ne peut corriger qu'une erreur

On peut transmettre 1 bit utile par cette approche

Les registres défaillants

Notre nouveau dictionnaire associe plusieurs mots de code à un message pour s'adapter à l'état initial du regisre.

Message	**Mots de code**	
00	000	111
01	011	100
10	110	001
11	010	101

- Encodage = mot de code le plus proche de l'état initial **dans le sous-dictionnaire du message** à transmettre
- Décodage = mot de code le plus proche des informations reçues **dans le dictionnaire entier**

Les registres défaillants

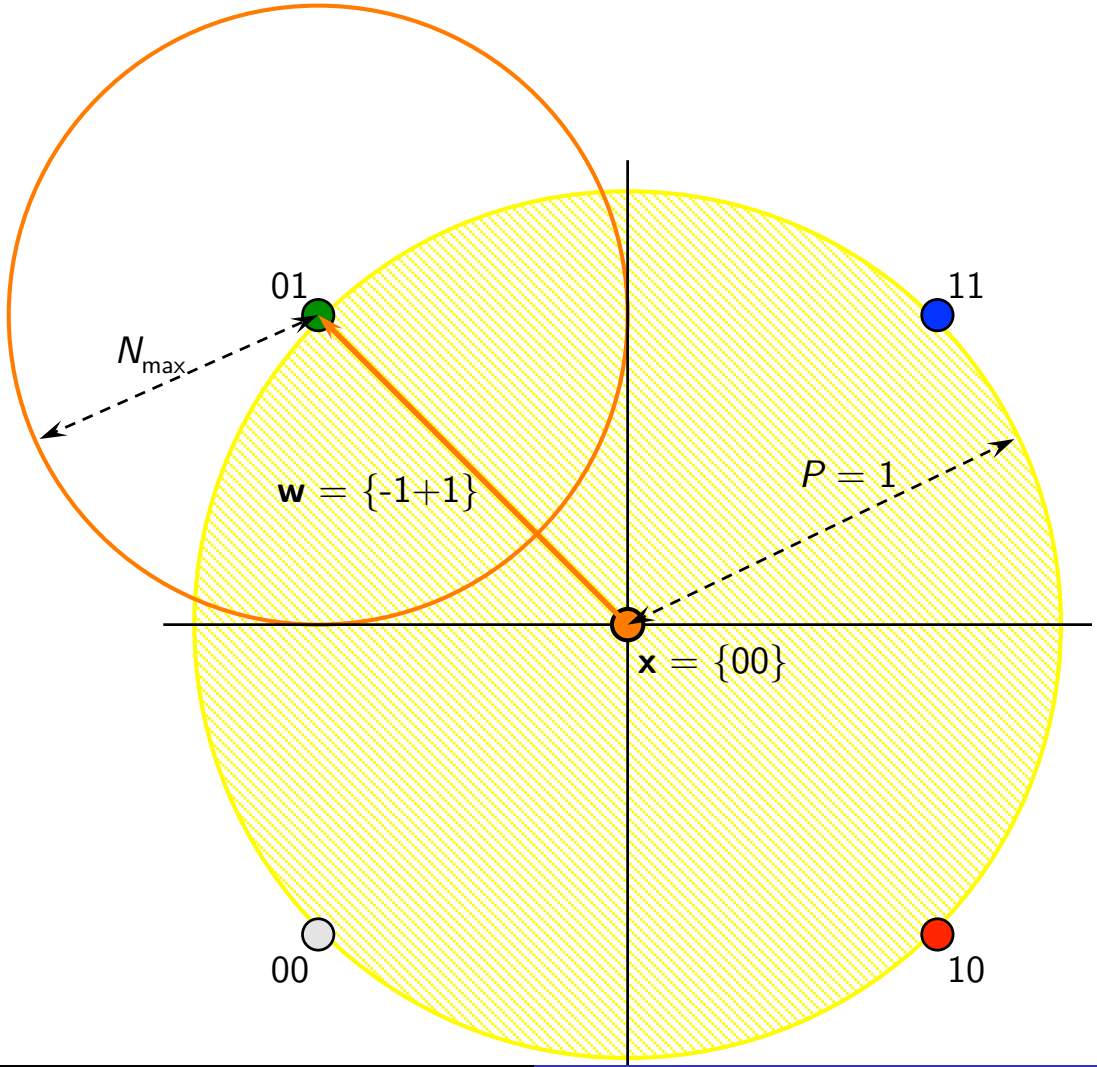
On cherche à transmettre le message 10. L'état initial est –1–.

Message	Mots de code	
00	000	111
01	011	100
10	110	001
11	010	101

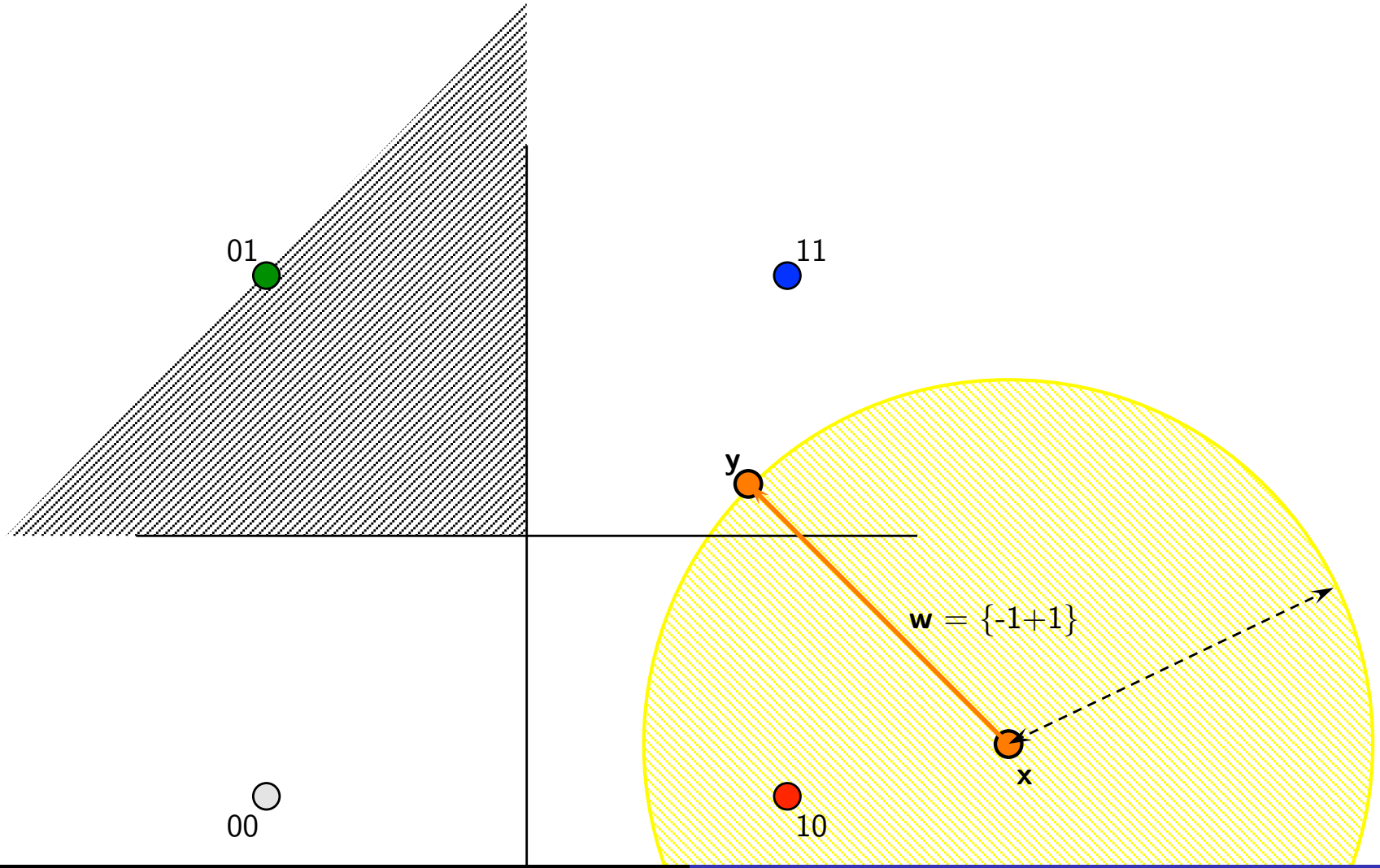
- Dans le sous-dictionnaire de 10, le mot le plus proche de –1– est 110
- Le récepteur cherche à quel sous-dictionnaire appartient 110, et décode le message 10

On peut donc transmettre 2 bits utiles

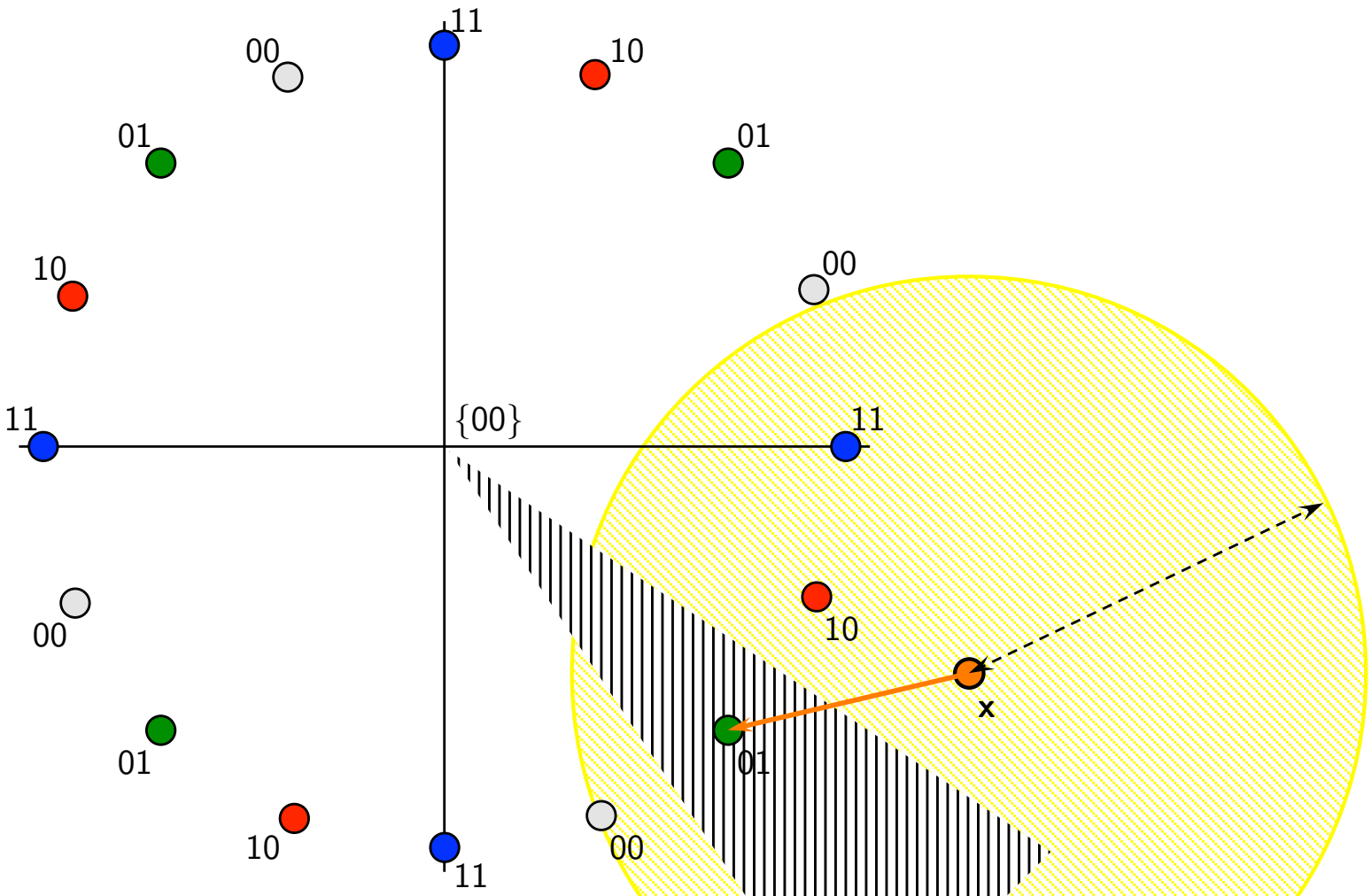
Interprétation géométrique



Interprétation géométrique



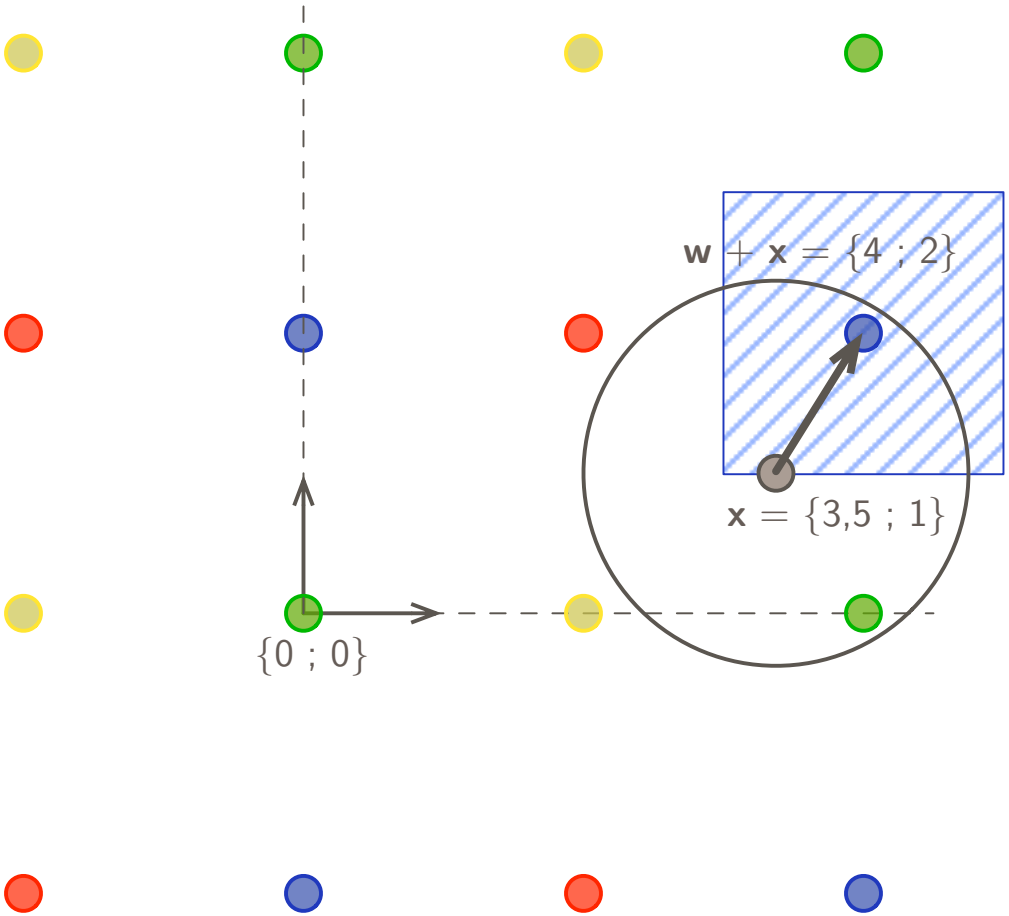
Interprétation géométrique



Le schéma idéal

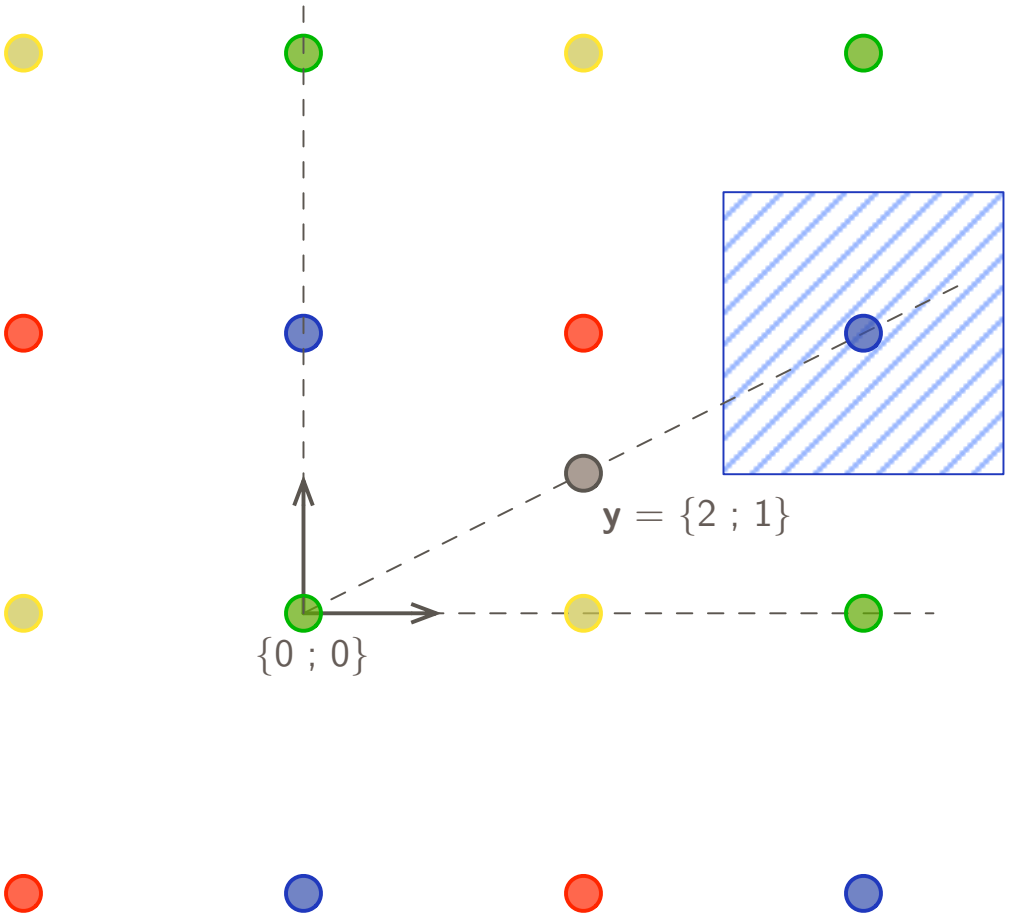
- Le schéma idéal de Costa comprend un **codage informé** (dictionnaire surjectif) et une insertion informée (diriger le vecteur hôte vers le mot de code)
- Mais en pratique, pour des dimensions courantes ($n > 100$), impossible de reprendre la construction aléatoire du dictionnaire comme dans la démonstration

En pratique : la quantification



La quantification associe plusieurs mots de code à un message : c'est un codage informé

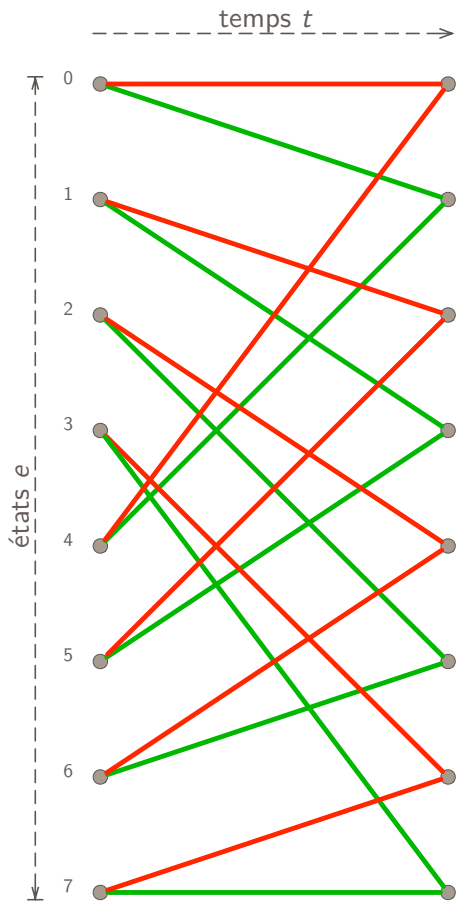
En pratique : la quantification



Mais un simple changement d'échelle décale la marque

En pratique : la quantification

- Un treillis est un graphe composé de $e \times t$ **états** et de **transitions**
- Chaque transition correspond à un bit à encoder et est valuée par un symbole émis
- À chaque message correspont un chemin, et donc une suite de symboles : c'est le mot de code
- Décodage = trouver le chemin le plus probable



En pratique : la quantification

- Pour associer plusieurs mots de code à un message, on multiplie les transitions
- L'encodage consiste à retrouver le chemin le plus proche de **x** qui correspond au bon message
- Décodage = idem, mais treillis plus grand