Théorie de l'information et codage **2010/2011**

Cours 6 — 22 mars 2011

Enseignant: Marc Lelarge *Scribe: Lucas Boczkowski*

Pour information

- Page web du cours
  `http://www.di.ens.fr/~lelarge/info11.html`

6.0.1 Le théorème de codage de canal

On sait déjà que l'on ne trouvera pas de code permettant de transmettre avec une probabilité d'erreur moyenne arbitrairement faible si $R > C$. Le théorème suivant donne un résultat positif (mais non constructif) dans le cas $R < C$.

Théorème 6.0.1 *Pour un canal discret sans mémoire de fonction capacité-coût $C(\beta)$, pour tout $\beta_0 \geq \beta_{min}$, et des réels $\beta > \beta_0$, $R < C(\beta_0)$, $\epsilon > 0$, pour n suffisamment grand, il existe un code $C \stackrel{def}{=} \{\underline{x}_1, \ldots, \underline{x}_M\}$ de longueur n et une règle de décodage, tel que :*
(a) chaque mot code $\underline{x}_i$ est β-admissible
(b) $M \geq 2^{\lceil Rn \rceil}$
(c) $P_E^{(i)} < \epsilon$ pour tout $i = 1, 2, \ldots, M$.

Démonstration. La preuve se décompose en deux parties. On va d'abord définir une fonction de décodage f pour un (M, n) code arbitraire. Puis on va montrer qu'il existe un code satisfaisant aux propriétés $(a), (b), (c)$ en utilisant un argument probabiliste.

L'espace probabilisé qui nous intéresse d'abord est $\Omega \stackrel{def}{=} \mathcal{X}^n \times \mathcal{Y}^n = \{(\underline{x}, \underline{y}) \Big/ \underline{x} \in \mathcal{X}^n, \underline{y} \in \mathcal{Y}^n\}$
On munit Ω de la mesure de probabilité :

$$p(\underline{x}, \underline{y}) = p(\underline{x})p(\underline{y}|\underline{x})$$

où

- $p(\underline{x}) = p(x_1) \ldots p(x_n)$ et $p(.)$ atteint la capacité $c(\beta_0)$
- $p(\underline{y}|\underline{x}) = \prod_{i=1}^n p(y_i|x_i)$

Soit $R < R' < C(\beta_0)$ et $\mathcal{T} = \{(\underline{x}, \underline{y}) \Big/ I(\underline{x}; \underline{y}) \geq nR'\} \subseteq \Omega$ où

$$I(\underline{x}; \underline{y}) = log_2 \frac{p(\underline{y}|\underline{x})}{p(\underline{y})} \text{ avec, } p(\underline{y}) = \sum_{\underline{x}} p(\underline{x}, \underline{y}) = \sum_{\underline{x}} p(\underline{x})p(\underline{x}|\underline{y}).$$

On définit encore

$$B = \{\underline{x} \ / b(\underline{x}) \leq n\beta\}$$

$$\mathcal{T}^\star = \{(\underline{x}, \underline{y}) \in \mathcal{T} \Big/ \underline{x} \in B\}$$

Pour un code $C = \{\underline{x}_1, \ldots, \underline{x}_M\}$ de longueur n, on définit la règle de décodage : pour chaque $\underline{y}$, on pose $S(\underline{y})$ l'ensemble des « bons candidats »

$$S(\underline{y}) = \{\underline{x} \Big/ (\underline{x}, \underline{y}) \in \mathcal{T}^\star\} \subseteq B$$

Lorsque l'on reçoit $\underline{y}$, si $S(\underline{y})$ est réduit à un élément $\underline{x}_i$, alors on pose $f(\underline{y}) = \underline{x}_i$ sinon on pose $f(\underline{y}) =?$
Pour le code C et avec cette règle de décodage, si $\underline{x}_i$ est transmis et $\underline{y}$ est reçu, on a :

$$P_E^i \leq P(\underline{x}_i \notin S(\underline{y})) + \sum_{j \neq i} P(\underline{x}_j \in S(\underline{y}))$$

Définissons les fonctions indicatrices Δ et $\overline{\Delta}$ comme suit

$$\Delta(\underline{x}, \underline{y}) = \begin{cases} 1, & \text{si } (\underline{x}, \underline{y}) \in \mathcal{T}^\star \\ 0, & \text{sinon} \end{cases} \quad \text{et} \quad \overline{\Delta} = 1 - \Delta$$

Cela permet de réecrire la majoration sur P_E^i de manière explicite comme une fonction (déterministe) du code $C = \{\underline{x}_1, \ldots, \underline{x}_M\}$:

$$P_E^i \leq \sum_{\underline{y}} \overline{\Delta}(\underline{x}_i, \underline{y}) p(\underline{y}|\underline{x}_i) + \sum_{j \neq i} \sum_{\underline{y}} \Delta(\underline{x}_j, \underline{y}) p(\underline{y}|\underline{x}_i) \stackrel{def}{=} Q_i(\underline{x}_1, \ldots, \underline{x}_M)$$

Voici maintenant la deuxième partie de la preuve (où l'on montre l'existence). L'idée du « random coding » est de définir une distribution de probabilités sur C, sous laquelle l'espérance des Q_i -ce sont alors des v.a- tende vers 0, pour $M = 2^{\lceil Rn \rceil}$. L'argument sera du type : « si en moyenne une quantité est petite, alors il existe des tirages qui la rendent effectivement petite ». On fait le choix suivant pour la distribution de probabilité sur les codes C :

$$p(\underline{x}_1, \ldots, \underline{x}_M) = \prod_{i=1}^{M} p(\underline{x}_i)$$

où $p(\underline{x}_i) = \prod_{k=1}^{n} p(x_{ik})$ atteint $C(\beta_0)$.

Remarque 6.0.1 – *l'intuition est la suivante : la loi $p(\underline{x})$ en entrée du canal permet de maximiser l'information mutuelle entre la sortie du canal et l'entrée, ce qui 'signifie' que les mots code $\underline{x}$ ayant une 'forte' probabilité $p(\underline{x})$ sont bien transmis. La loi choisie sur les codes met plus de 'poids' sur les mots code transmis de manière fiable et donc la 'densité' de tels mots code est plusélevée.*

– *Nous avons défini des codes jusqu'à présent comme des ensembles. Ici, la définition d'un code est un M-uplet. En particulier, il peut y avoir deux mot-codes identiques qui vont donc systématiquement donner une erreur avec la règle de décodage que nous avons définie. Pour des raisons de symmétries, il est cependant plus facile d'effectuer les calculs avec cette distribution et de modifier le code pour résoudre ces problèmes a posteriori, ce que nous ferons à la fin de la preuve.*

En prenant la moyenne par rapport à la loi sur les codes, on obtient :

$$\mathbb{E}[Q_i] = \underbrace{\mathbb{E}\Big[\sum_{\underline{y}} \overline{\Delta}(\underline{x}_i, \underline{y}) p(\underline{y}|\underline{x}_i)\Big]}_{E_1} + \sum_{j\neq i} \underbrace{\mathbb{E}\Big[\sum_{\underline{y}} \Delta(\underline{x}_j, \underline{y}) p(\underline{y}|\underline{x}_i)\Big]}_{E_2^{(j)}}$$

Il s'agit à présent de majorer E_1 et $E_2^{(j)}$ par des quantités tendant vers 0.

$$\begin{aligned} E_1 &= \sum_{\underline{x}_1,\cdots,\underline{x}_M} p(\underline{x}_1 \dots p(\underline{x}_M) \sum_{\underline{y}} \overline{\Delta}(\underline{x}_i, \underline{y}) p(\underline{y}|\underline{x}_i) \\ &= \sum_{\underline{x},\underline{y}} p(\underline{x}) p(\underline{y}|\underline{x}) \overline{\Delta}(\underline{x}_i, \underline{y}) \\ &= P((\underline{x}, \underline{y}) \notin \mathcal{T}^\star) \\ &\leq P(\underline{x}, \underline{y}) \notin \mathcal{T}) + P(\underline{x} \notin B) \end{aligned}$$

Ce qui se réecrit

$$E_1 \leq P(I(\underline{x}, \underline{y}) < nR') + P(b(\underline{x}) > n\beta)$$

Mais $b(\underline{x}) = \sum_{k=1}^{n} b(x_k)$ est une somme de v.a i.i.d de moyenne $\mathbb{E}[b(X)] \leq \beta_0 < \beta$. Par la Loi Faible des Grands Nombres (LFGN)

$$\lim_{n\to\infty} P(b(\underline{x}) > n\beta) = 0$$

Par la propriété sans mémoire du canal, on peut écrire

$$I(\underline{x}, \underline{y}) = \sum_{k=1}^{n} \log \frac{p(y_k|x_k)}{p(y_k)} = \sum_{k=1}^{n} I(x_k, y_k)$$

Donc sous la probabilité $p(\underline{x}, \underline{y})$, nous avons à nouveau une somme de v.a. i.i.d de moyenne :

$$\mathbb{E}[I(x_k, y_k)] = I(X, Y) = C(\beta_0) > R'$$

Et on peut conclure que $P(I(\underline{x},\underline{y}) < nR') \to 0$ par la LFGN. Passons à $E_2^{(j)}$

$$\begin{aligned} E_2^{(j)} &= \sum_{\underline{x}_j,\underline{y}} p(\underline{x}_j)\Delta(\underline{x}_j,\underline{y}) \sum_{\underline{x}_i} p(\underline{x}_i)p(\underline{y}|\underline{x}_i) \\ &= \sum_{\underline{x},\underline{y}} p(\underline{x})\Delta(\underline{x},\underline{y})p(\underline{y}) \\ &\leq \sum_{\underline{x},\underline{y}\in\mathcal{T}} p(\underline{x})p(\underline{y}) \end{aligned}$$

Par définition de $\mathcal{T}$, $p(\underline{x})p(\underline{y}) \leq p(\underline{x},\underline{y})2^{-R'n}$. Donc

$$E_2^{(j)} \leq 2^{-R'n} \sum_{\underline{x},\underline{y}\in\mathcal{T}} p(\underline{x},\underline{y}) \leq 2^{-R'n}$$

et $\sum_{j\neq i} E_2^{(j)} \leq M2^{-R'n}$. Pour $M = 2^{\lceil Rn\rceil+1}$ (ce choix de M paraît arbitraire mais va être justifié dans la suite) et $R' > R$, de dernier majorant tend vers 0 quand n tend vers l'infini.
Donc pour n assez grand, $\mathbb{E}[Q_i] \leq \epsilon$ avec $M = 22^{\lceil Rn\rceil+1}$. On définit l'erreur moyenne par :

$$P_E(C) = P_E(\underline{x}_1,\ldots,\underline{x}_M) = \frac{1}{M}\sum_{i=1}^{M} P_E^i(\underline{x}_1,\ldots,\underline{x}_M)$$

Pour $M = 2^{\lceil Rn\rceil+1}$ et n grand, la majoration de chacun des $\mathbb{E}[Q_i]$ donne :

$$\mathbb{E}[P_E] < \epsilon.$$

Donc en particulier il existe un code $\{\underline{x}_1,\ldots,\underline{x}_M\}$ avec probabilité d'erreur moyenne

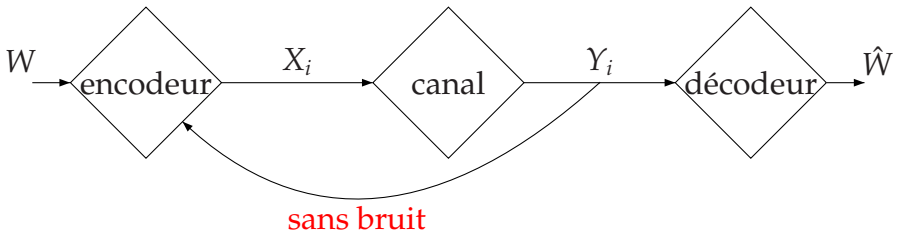
$$P_E(\underline{x}_1,\ldots,\underline{x}_M) < \epsilon$$

Attention : il peut cependant exister des mots code $\underline{x}_i$ avec

$$b(\underline{x}_i) > n\beta \text{ ou } P_E^i > \epsilon.$$

Pour régler ce problème on procède de la façon suivante. On sait qu'au plus la moitié des mots codes $\underline{x}_i$ sont tels que $P_E^{(i)} \geq \epsilon$ sinon $P_E \geq \epsilon/2$ (ce qui fournit une contradiction quitte à changer $\epsilon \leftarrow \frac{\epsilon}{2}$ partout).
Si on supprime les mots code ayant une probabilité d'erreur $P_E^i \geq \epsilon$, on obtient un nouveau code avec $M \geq 2^{\lceil Rn\rceil}$ mots code. De plus le fait de supprimer des mots code ne peut faire que diminuer la probabilité d'erreur des mots code restant. Donc pour chacun des mots code restant, on a $P_E^i < \epsilon$.
De plus, automatiquement, si $b(\underline{x}_i) > n\beta$ alors $\underline{x}_i$ n'est pas dans le code obtenu. En effet, pour tout $\underline{y}$, comme $S(\underline{y}) = \{\underline{x}, (\underline{x},\underline{y}) \in \mathcal{T} \,\&\, b(\underline{x}) \in B\}$ alors $\underline{x}_i \notin S(\underline{y})$ pour tout $\underline{y}$ et donc $P_E^{(i)} = 1$.
Finalement le code obtenu vérifie les trois propriétés attendues. □

6.0.2 Canal avec feedback



Dans un canal avec feedback, au moment de l'encodage du $(n+1)$-ème bit, on sait quels ont été les n premiers bits reçus. Le but est de montrer qu'un tel canal a la même capacité que s'il n'y avait pas de feedback. Clairement $C_{\text{feedback}} \geq C$.
Un $(2^{nR}, n)$ code avec feedback est une suite de fonctions $\{1, \dots, 2^{nR}\} \times \mathcal{Y}^{i-1} \to \mathcal{X}$ et une fonction de décodage $g : \mathcal{Y}^n \to \{1, \dots, 2^{nR}\} \cup \{?\}$.
Si W est uniformément distribuée dans $\{1, \dots, 2^{nR}\}$

$$P_E^{(n)} = P(g(Y^{(n)}) \neq W)$$

donc

$$\begin{aligned} nR = H(W) &= H(W|\hat{W}) + I(W;\hat{W}) \\ \text{(Fano)} &\leq 1 + P(W \neq \hat{W})nR + I(W;\hat{W}) \\ \text{(Data Processing Ineq.)} &\leq 1 + P_E^{(n)}nR + I(W;Y^{(n)}) \end{aligned}$$

et

$$\begin{aligned} I(W;Y^{(n)}) &= H(Y^{(n)}) - H(Y^{(n)}|W) \\ &= H(Y^{(n)}) - \sum_{i=1}^{n} H(Y_i|Y_1, \dots, Y_{i-1}, W) \\ &\leq \sum_i H(Y_i) - \sum_i H(Y_i|X_i) \\ &= \sum_{i=1}^{n} I(X_i, Y_i) \leq nC, \end{aligned}$$

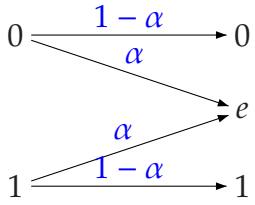
où la première inégalité vient du fait que X_i est une fonctions des $Y_1, \dots, Y_{i-1}, W$ donc $H(Y_i|Y_1, \dots, Y_{i-1}, W) = H(Y_i|Y_1, \dots, Y_{i-1}, W, X_i)$ et que sachant X_i, Y_i est indépendante des $Y_1, \dots, Y_{i-1}, W$ donc $H(Y_i|Y_1, \dots, Y_{i-1}, W, X_i) = H(Y_i|X_i)$.

Donc $nR \leq 1 + P_E^{(n)}nR + nC$ soit

$$R \leq \frac{C}{1 - P_E^{(n)}} + \frac{1}{n(1 - P_E^{(n)})}$$

Donc pour tout $(2^{nR}, n)$ code avec feedback tel que $P_E^{(n)} \to 0$ quand $n \to \infty$, on doit avoir $R \leq C$, ce qui suffit à conclure en vertu du théorème de codage de canal.

Canal à effacement



On cherche à déterminer la capacité C de ce canal.

$$C = \max I(X;Y) = \max_{p(x)} H(Y) - H(\alpha)$$

Posons $\pi := P(X = 1)$, alors $H(Y)$ se réecrit

$$H(Y) = H((1-\pi)\alpha, \alpha, \pi(1-\alpha)) = H(\alpha) + (1-\alpha)H(\pi)$$

et donc $C = 1 - \alpha$.
En fait, on savait qu'on ne pourrait pas faire mieux grâce au canal analogue avec feedback. Cependant pour le canal à effacement avec feedback, la méthode de transmission suivante est naturelle : le symbole reçu est transmis dans le canal feednback, donc si e a été reçu, on peut réemettre jusqu'à transmission du bit voulu. Dans ce cas on pourra transmettre au taux $1-\alpha$ car la transmission d'un bit prend un temps aléatoire suivant une loi géométrique de paramètre de moyenne $\frac{1}{1-\alpha}$. Donc pour transmettre k bits, il faut de l'ordre de $n = \frac{k}{1-\alpha}$ utilisations du canal (lorsque k est grand).