Codage et Cryptanalyse linéaire

Application au DES

B. Gérard Le Bobinnec - J-P. Tillich

C2 - 18 mars 2008

Plan

Plan

Rappels sur le DES

Entrées :

- Message de 64 bits : $P = (L_0||R_0)$.
- Clé de 56 bits $K \longrightarrow (K_1, \ldots, K_{16})$ sous clés de 48 bits.

Sorties :

- Chiffré de 64 bits : $C = (L_{16}||R_{16})$.

Schéma de Feistel

Pour $0 \leq i \leq 15$ calculer :

$$L_{i+1} = R_i \quad , \quad R_{i+1} = L_i \oplus F(R_i, K_{i+1}).$$

Renvoyer $C = (L_{16}||R_{16})$.

Remarque

Pour déchiffrer C il suffit de lui appliquer le DES en inversant l'ordre des sous-clés.

Philosophie de l'attaque linéaire

- Attaque à clairs connus.
- Exploite le manque de non-linéarité d'un chiffrement.

Equation

$$\langle P, \pi \rangle \oplus \langle C, \gamma \rangle \oplus c = \langle K, \kappa \rangle$$

$$\langle A, B \rangle = \bigoplus_k A_k . B_k .$$

- $\pi \in \mathbb{F}_2^{64}$: masque d'entrée.
- $\gamma \in \mathbb{F}_2^{64}$: masque de sortie.
- $\kappa \in \mathbb{F}_2^{56}$: masque de clé.

Philosophie de l'attaque linéaire

- Attaque à clairs connus.
- Exploite le manque de non-linéarité d'un chiffrement.

Equation probabiliste

$$\mathcal{P}\left(\langle P, \pi\rangle \oplus \langle C, \gamma\rangle \oplus c = \langle K, \kappa\rangle\right) = \frac{1}{2} + \varepsilon.$$

$$\langle A, B\rangle = \bigoplus_k A_k.B_k.$$

- $\pi \in \mathbb{F}_2^{64}$: masque d'entrée.
- $\gamma \in \mathbb{F}_2^{64}$: masque de sortie.
- $\kappa \in \mathbb{F}_2^{56}$: masque de clé.

Attaque directe [Matsui EUROCRYPT'93]

$$\mathcal{P}(\langle P, \pi \rangle \oplus \langle C, \gamma \rangle \oplus c = \langle K, \kappa \rangle) = \frac{1}{2} + \varepsilon.$$

- Récupérer $N = O(1/\varepsilon^2)$ couples de messages.
- Calculer T le nombre de couples pour lesquels la partie gauche de l'équation vaut 1.
- Si $T > N/2$ alors on devine que $\langle K, \kappa \rangle = 1$, 0 sinon.

Remarque

Ne permet de retrouver qu'un bit d'information sur la clé.
Une des motivations pour utiliser plusieurs équations.

Attaque par distingueur [Matsui EUROCRYPT'93]

$$\mathcal{P}(\langle P, \pi \rangle \oplus \langle C, \gamma \rangle \oplus \langle F(P, K_1), \gamma' \rangle \oplus c = \langle K, \kappa \rangle) = \frac{1}{2} + \varepsilon.$$

$\langle F(P, K_1), \gamma' \rangle$ est en fait une fonction de C et de $\tilde{K}$, $\tilde{K}$ étant composée de 6 bits de K.

- Pour chacune des 2^6 possibilités pour $\tilde{K}$ calculer T le nombre de couples pour lesquels la partie gauche de l'équation vaut 1.
- Parmi les T repérer celui qui maximise $|T - N/2|$.
- On devine que la clé partielle utilisée est le $\tilde{K}$ correspondant.
- De plus, si $T > N/2$ alors on devine que $\langle K, \kappa \rangle = 1$, 0 sinon.

Remarque

Permet de retrouver 7 bits d'information sur la clé.
Impose des contraintes sur l'équation utilisée.

Schéma général d'une cryptanalyse linéaire

Cryptanalyse en 3 étapes :

- Extraction de l'information.
- Analyse de l'information.
- Recherche exhaustive (sur les bits restants).

Attaque par distingueur

- Calcul des compteurs.
- Tri des compteurs (et donc des candidats pour $\tilde{K}$).
- Recherche des $56 - 7$ bits non devinés.

On va s'intéresser à la phase d'analyse de l'information en cas d'utilisation de plusieurs équations.

Plan

Etat de l'art

- Masques de supports disjoints [Matsui CRYPTO'94] .
- Plusieurs équations avec le même masque de clé [Kaliski, Robshaw CRYPTO'94] [Choi, Hong, Hong, Lee 2005] .
- Plusieurs équations avec un masque de clé quelconque [Biryukov, De Cannière, Quisquater CRYPTO'04] .

Problématique de la cryptanalyse linéaire multiple

$$K \longrightarrow (\langle K, \kappa_j \rangle)_{1 \leq j \leq n} = \tilde{K} \quad \underset{\text{analyse}}{\overset{\text{extraction}}{\longrightarrow}} \quad (\mathcal{P}(\tilde{K}_j = 0))_{1 \leq j \leq n}$$

Notre point de vue : problème de décodage sur un canal gaussien.
$\tilde{K} \in \mathcal{C}$, code de longueur n et dimension k.

Canal à Bruit Blanc Additif Gaussien

Modèle

$$(X_j)_{1\leq j\leq n} \xrightarrow{\text{canal}} (Y_j)_{1\leq j\leq n}$$

$$X_j = (-1)^{\tilde{K}_j} \quad , \quad Y_j = X_j + N_j \quad , \quad N_j \sim \mathcal{N}(0, \frac{1}{4N\varepsilon_j^2})$$

Les N_j sont indépendants.

Soit :

$$T_j = \sum_{(P,C)} < \pi_j, P > \oplus < \gamma_j, C > \oplus b_j.$$

Alors la cryptanalyse correspond à ce modèle si on considère :

$$Y_j = \frac{N - 2T_j}{2N\varepsilon_j}.$$

Etude du nombre de couples nécessaires

Etude du nombre de candidats plus vraisemblables que la bonne clé.

Junod : 21 attaques par distingueur [Junod SAC'01] :

> *'We observe [...] a pessimistic rank expected value.'*

Phénomène connu en théorie des codes correcteurs :
Avec une probabilité exponentiellement faible la taille de l'ensemble est exponentiellement grande.

La taille de l'ensemble varie entre 1 et 2^k alors que le logarithme de la taille varie entre 0 et k.

Il est donc préférable de calculer l'espérance du logarithme de la taille.

Solution : Entropie

X et Y deux variables aléatoires discrètes à valeurs dans $\mathcal{X}$ et $\mathcal{Y}$.
On note $p(x)$(resp. $p(y)$) loi de proba de X(resp. Y).

L'entropie de X correspond au nombre de bits d'incertitude sur X :

$$\begin{aligned}
\mathcal{H}(X) &= -\sum_{x\in\mathcal{X}} p(x)\log_2(p(x)) \\
\mathcal{H}(X|Y) &= -\sum_{x\in\mathcal{X},y\in\mathcal{Y}} p(x|y).p(y)\log_2(p(x|y)) \\
\mathcal{I}(X;Y) &= \sum_{x\in\mathcal{X},y\in\mathcal{Y}} p(x|y).p(y)\log_2\left(\frac{p(x|y)}{p(x)}\right) \\
&= \mathcal{H}(X) - \mathcal{H}(X|Y)
\end{aligned}$$

Borne sur l'entropie

Inégalité essentielle

Si $p(\tilde{\mathbf{K}}|\mathbf{Y}) = \prod_{j=1}^{n} p(\tilde{K}_j|Y_j)$:

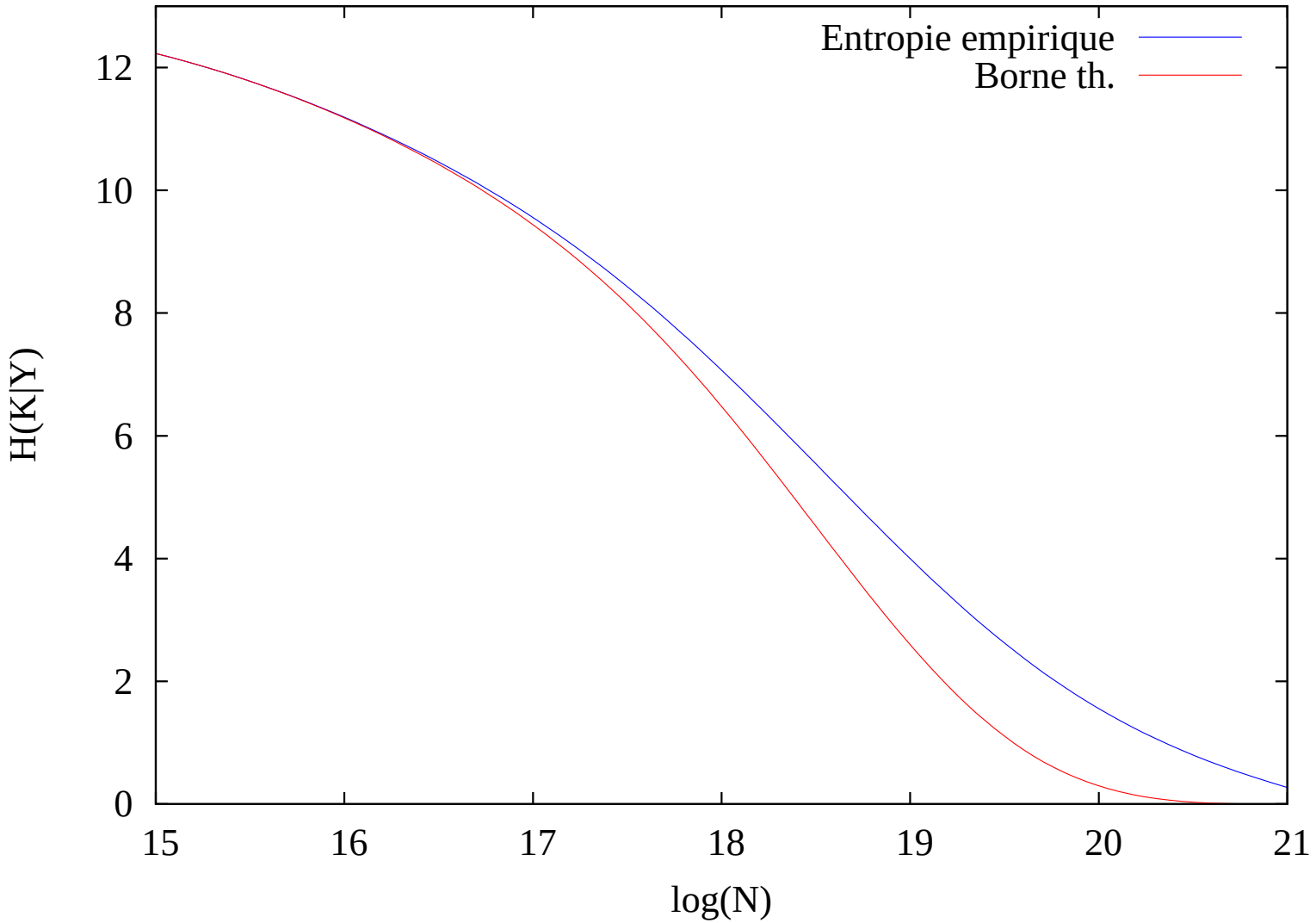
$$\mathcal{I}(\tilde{\mathbf{K}};\mathbf{Y}) \leq \sum_{j=1}^{n} \mathcal{I}(\tilde{K}_j;Y_j) \quad , \quad \mathcal{H}(\tilde{\mathbf{K}} \mid \mathbf{Y}) \geq \mathcal{H}(\tilde{\mathbf{K}}) - \sum_{j=1}^{n} \mathcal{I}(\tilde{K}_j;Y_j).$$

Application à l'attaque directe

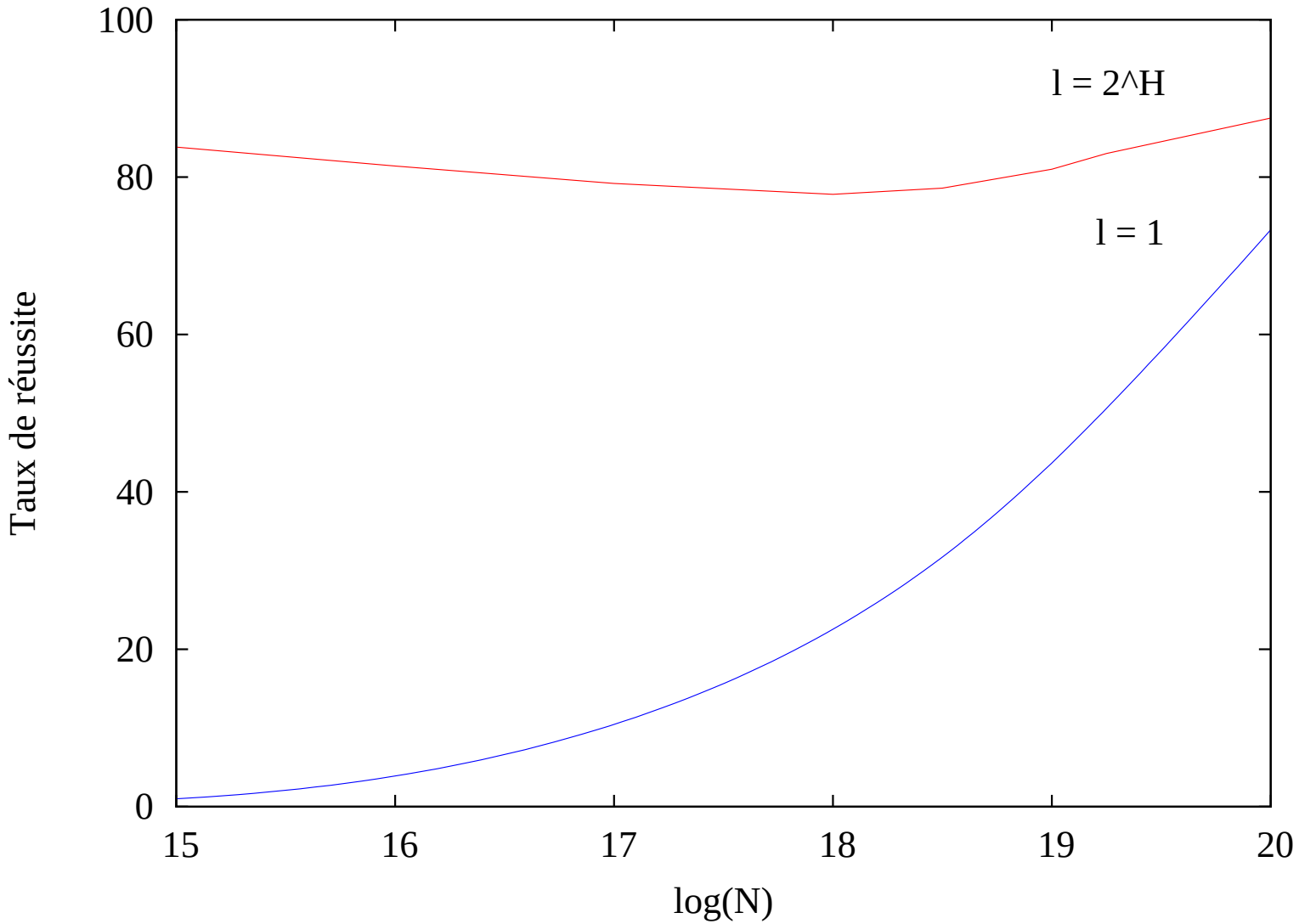
$$\tilde{\mathbf{K}} = (\tilde{\mathbf{K}})_j = <\mathbf{K}, \kappa_j> \quad , \quad \mathcal{H}(\tilde{\mathbf{K}} \mid \mathbf{Y}) \geq k - \sum_{j=1}^{n} \mathbf{Cap}(\sigma_j^2) \quad , \quad \sigma_j^2 = \frac{1}{4N\varepsilon_j^2}$$

$$\mathbf{Cap}(\sigma^2) = \int_{-1}^{1} \frac{\sigma}{\sqrt{2\pi}(1-t^2)} e^{-\frac{(1-\sigma^2 tanh^{-1}(t))^2}{2\sigma^2}} \log_2(1+t)dt.$$

Domaine de pertinence de la borne



Entropie et taille de liste



Plan

Phase d'analyse

$$K \longrightarrow X = \left((-1)^{<K,\kappa_j>}\right)_j \overset{\text{canal}}{\longrightarrow} Y = (X_j + N_j)_j \overset{\text{décodage}}{\longrightarrow} K'$$

Décodage au maximum de vraisemblance

$K' \in \mathcal{C}$ qui maximise $P(\mathbf{K} = K' \mid \mathbf{Y} = Y)$.

Algorithme naïf :

- Calculer les 2^k mots de code.
- Calculer leurs vraisemblances.
- Renvoyer le mot le plus vraisemblable.

Algorithme naïf $\Leftrightarrow$ méthode actuelle de cryptanalyse.

Solution : algorithme de décodage

Algorithme de décodage par résonnance stochastique [Valembois 2000] .

Avantages

- Ne regarde que peu de mots de code ‘typiques’.
- Efficace (utilisation de tables de hachage).

Désavantage

- Pas de résultat théorique sur sa complexité.

Pour retrouver 23 bits sur 8 tours :

- Algorithme naïf : 2^{23} mots de codes.
- Résonnance stochastique : 2^{12} mots de code.

Recherche d'approximations

- 'Branch and bound Algorithm'.
- Décodage des Reed-Muller d'ordre 1 [Loidreau, Tavernier] .

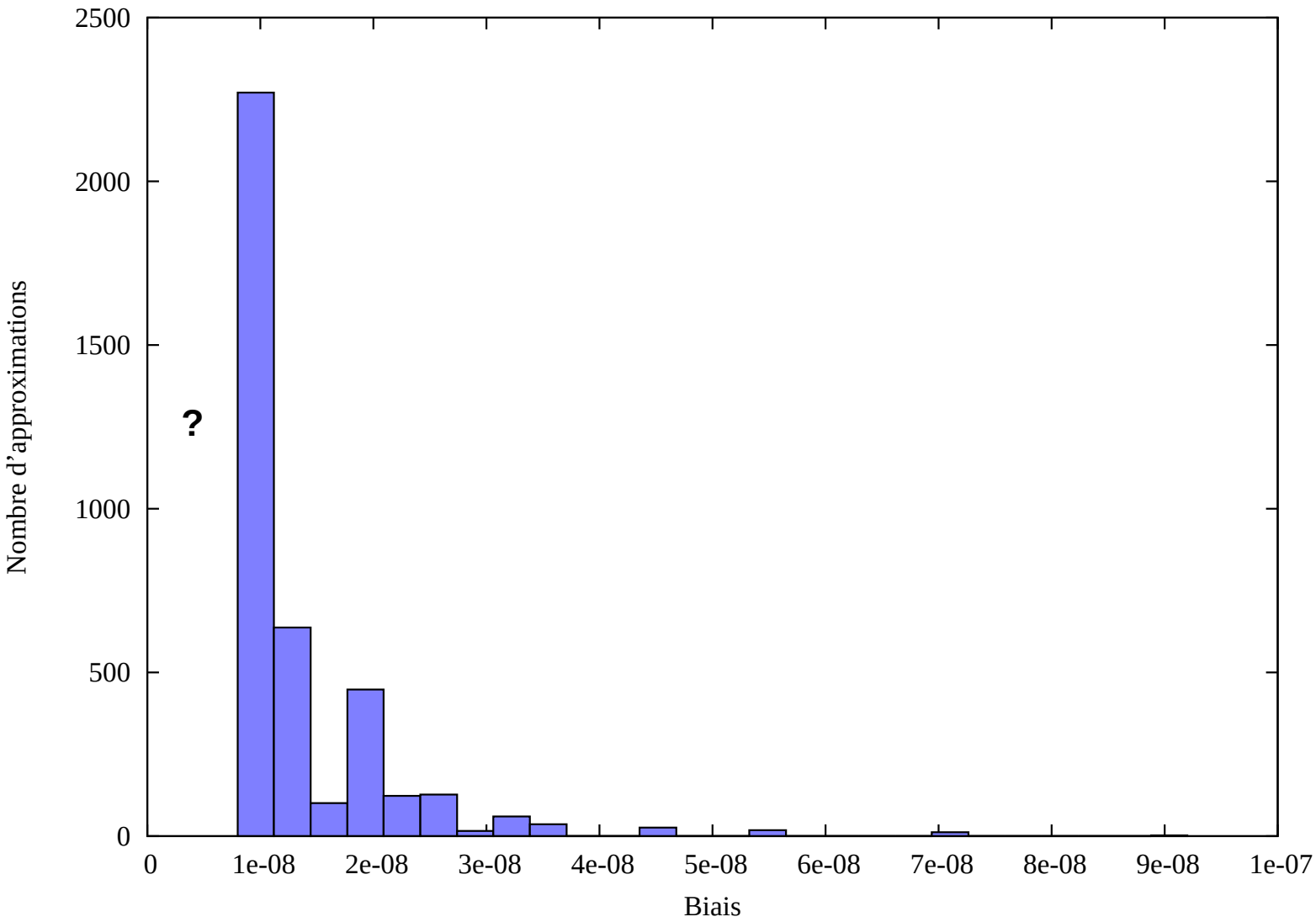
Avantages des Reed-Muller

- Fonctionne même en cas de Key Scheduling non linéaire.
- Pas de 'Hull Effect'.

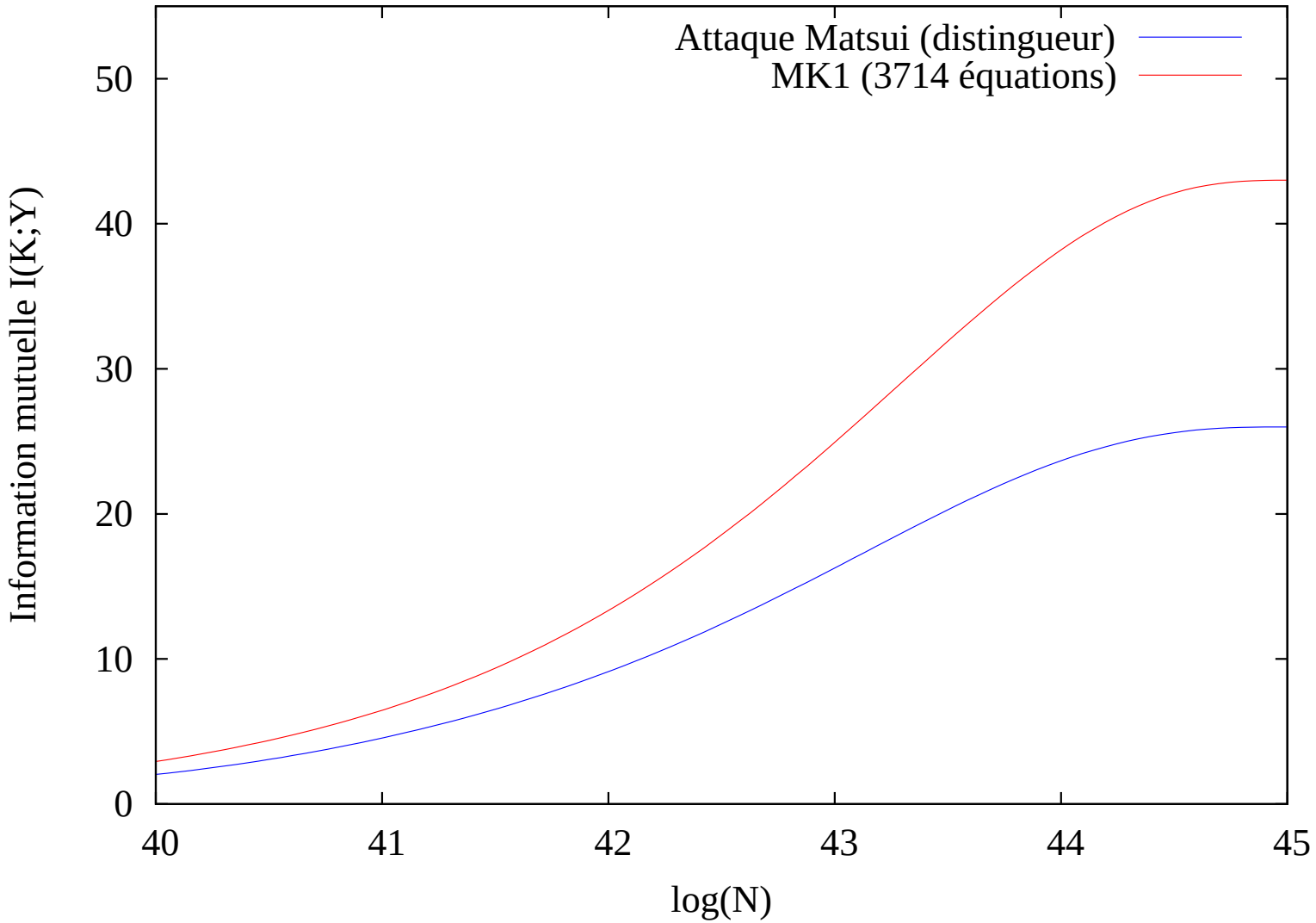
Désavantages

- Nécessité d'avoir un bon masque de sortie (désavantage ?).
- Actuellement : trop long et volumineux pour des équations sur 16 tours.

Approximations trouvées



Potentiel de ce type d'attaque Vs distingeurs



Plan

Conclusion

- On a vu un moyen d'estimer la quantité d'information gagnée lors d'une cryptanalyse.
- On peut diminuer la complexité de l'attaque directe en utilisant des techniques de codage.

- Perspectives :
  - Pour utiliser cette borne il suffit de pouvoir calculer $\mathcal{I}(\tilde{K}_j; Y_j)$. Regarder d'autres cryptanalyses (statistique, ...).
  - Attaquer des formes réduites de candidats AES.
  - Réussir à utiliser la technique des Reed-Muller pour des biais plus petits.
  - Etudier la complexité de l'algorithme de Valembois.

Conclusion

- On a vu un moyen d'estimer la quantité d'information gagnée lors d'une cryptanalyse.
- On peut diminuer la complexité de l'attaque directe en utilisant des techniques de codage.

- Perspectives :
  - Pour utiliser cette borne il suffit de pouvoir calculer $\mathcal{I}(\tilde{K}_j; Y_j)$. Regarder d'autres cryptanalyses (statistique, ...).
  - Attaquer des formes réduites de candidats AES.
  - Réussir à utiliser la technique des Reed-Muller pour des biais plus petits.
  - Etudier la complexité de l'algorithme de Valembois.