Chiffrement à l'aide de Codes Correcteurs et Corrélations

A Cryptosystem Using Error Correcting Codes and Correlations

par Sami Harari

Laboratoire Modélisation et Signal
Université de Toulon et du Var
B.P.132 83957 La Garde cedex France
e-mail :harari@univ-tln.fr

résumé et mots clés

Ce travail propose un algorithme de chiffrement à clés secrètes, utilisant un ensemble de codes correcteurs. A cette fin il introduit une nouvelle classe de codes aléatoires décodables. Contrairement au cryptosystème de McEliece, les clés doivent en être gardées secrètes. Le taux de transmission en est cependant plus élevé, très voisin de 1 si les dimensions des codes sont grandes; le volume d'une clé de chiffrement est également plus réduit que dans le système de référence. Ce travail représente une amélioration d'un système déjà présenté dans [1].

Code correcteur, Cryptosystème, Clés secrètes, Corrélation, Codes concaténés.

abstract and key words

In this document a new secret key cryptosystem is presented. It uses a set of particular error correcting codes. Its information rate is close to 1 if the dimension of the code is large. The volume of the keys is lower than that of McEliece and Niederreiter cryptosystems. This work is an improvement of a system presented in [1].

Code correcteur, Cryptosystème, Clés secrètes, Corrélation, Codes concaténés.

1. introduction

Ce travail reprend, en le généralisant, un système de chiffrement à clé secrète à l'aide de codes correcteurs proposé dans [1], qui a un très faible taux de transmission.

Dans [2] McEliece avait introduit un système de cryptographie à clé publique utilisant des codes correcteurs. Le message y était codé en mots de code, en utilisant une matrice génératrice G du code, le secret assuré par le choix d'un vecteur à composantes binaires tirées aléatoirement, additionné au mot du code.

Des attaques en temps exponentiel ont été développées contre ce système. Elles sont inefficaces si les dimensions sont suffisamment grandes.

Niederreiter avait proposé dans [3] un système analogue où l'information était contenue dans le vecteur d'erreur, le mot du code servant à masquer celle ci.

De récents travaux ont montré l'équivalence des deux systèmes du point de vue de la solidité. Celui de Niederreiter est plus performant en taux de transmission.

Dans ce travail, le cryptosystème est à clés secrètes et utilise un ensemble de codes aléatoires de distance minimale connue. Comme dans le système de Niederreiter le cryptogramme est constitué de la somme d'un mot du code et d'un vecteur de l'espace ambiant, l'information étant portée par ce dernier, le mot du code servant à masquer l'information.

Pour chacun des codes aléatoires retenus, il existe un algorithme de correction d'erreur permettant de corriger de nombreuses configurations de poids supérieur au demi poids minimal. Le

code, ainsi que le mot du code, sont choisis aléatoirement dans les ensembles correspondants.

L'étude de la robustesse du système montre en particulier que les attaques contre le système de McEliece ne sont pas valables contre le présent système.

Quelle que soit la capacité de correction du code, il est possible de caractériser et d'énumérer toutes les configurations corrigibles par le décodeur.

Ce cryptosystème possède la propriété remarquable que son taux de transmission n'est fonction que de la redondance du code. Il est sensiblement plus élevé que les taux de ceux cités précédemment.

2. corrélation de vecteurs

Pour sa mise en œuvre le nouveau cryptosystème utilise les propriétés de corrélation des vecteurs qui sont exposées ci-après.

Définition 1

Soit $\vec{v} = (v_0, \ldots, v_{n-1})$ un vecteur binaire de longueur n. Le support S de $\vec{v}$ est le sous ensemble des indices de $\{0, \ldots, n-1\}$ pour lesquels $v_i = 1$.

Définition 2

Le poids d'un vecteur binaire $\vec{v} = (v_0, \ldots, v_{n-1})$, noté $w(\vec{v})$, est la somme arithmétique de ses composantes :

$$w(\vec{v}) = \sum_{i=0}^{n-1} v_i.$$

Définition 3

Soit $\vec{v} = (v_0, \ldots, v_{n-1})$ et $\vec{v'} = (v'_0, \ldots, v'_{n-1})$ deux vecteurs binaires de longueur n. Le vecteur de corrélation $h(\vec{v}, \vec{v'})$ de $\vec{v}$ et $\vec{v'}$, est défini par $h(\vec{v}, \vec{v'}) = (v_0.v'_0, \ldots, v_{n-1}.v'_{n-1})$

De la propriété évidente que

$$\text{supp}\,(h(\vec{v}, \vec{v'})) = \text{supp}\,(\vec{v}) \cap \text{supp}\,(\vec{v'})$$

il suit aisément que

$$w((h(\vec{v}, \vec{v'})) \leq \inf\,(w(\vec{v}), w(\vec{v'}))$$

3. définition des codes aléatoires utilisés

Les codes aléatoires utilisés ont tous les mêmes paramètres (n, k), cependant leur distance minimale n'est pas constante. Ils admettent des matrices génératrices ayant la propriété suivante : les supports de tout couple de vecteurs lignes sont disjoints.

Soit

$$1 < n_1 < \ldots < n_{k-1} < n_k = n$$

une suite strictement croissante de k entiers.

Définition 4

Un code utilisable $C(n; n_1, n_2, \ldots, n_k)$ est un code binaire linéaire de longueur n, ayant pour matrice génératrice G une matrice dont le $i$ème vecteur ligne $\vec{g}_i$ a ses composantes nulles exceptées celles d'indice $n_i, n_i + 1, \ldots, n_{i+1} - 1$.

De tels codes ont pour poids minimum

$$\mu = \inf_i\{n_{i+1} - n_i\}$$

Une permutation appliquée aux colonnes de la matrice génératrice fournit un code ayant même polynôme énumérateur de poids et, en particulier, même distance minimale. La propriété des supports est également conservée par permutation.

Lemme 1

Le polynome énumérateur de poids $A(z)$ d'un code C de type $(n; n_1, n_2, \ldots, n_k)$ ayant une matrice génératrice G dont chaque vecteur ligne $\vec{g}_j$ $0 \leq j \leq k-1$ est de poids $w_j = n_{j+1} - n_j$ vaut :

$$A(z) = 1 + \sum_{j=0}^{k-1} z^{w_j} + \sum_{0 \leq i \neq j \leq k-1} z^{w_i + w_j} + \ldots + \sum_{0 \leq j_0 \neq j_1 \neq \ldots \neq j_{k-2}} z^{j_0 + \ldots + j_{k-2}} + z^n$$

Preuve se déduit aisément de celle de [1].

4. décodage des codes aléatoires utilisés

Le lemme suivant caractérise les configurations d'erreurs corrigibles pour de tels codes.

Lemme 2 (corrélation algébrique).

Soit C un code utilisable de matrice génératrice G constituée de vecteurs lignes $\vec{g}_j \quad j = 0 \ldots k-1$ de poids μ_j. Soit $\vec{r} = \vec{c} + \vec{e}$ la somme d'un mot de code et d'un vecteur d'erreur ayant la propriété que $w(\vec{e}_j) < \mu_j/2$, où $\vec{e}_j = h(\vec{e}, \vec{g}_j)$ $0 \leq j \leq k-1$. Un algorithme pour retrouver $\vec{c}$ à partir de $\vec{r}$ est le suivant :

- Calculer les vecteurs de corrélation $\vec{\epsilon}_j = h(\vec{r}, \vec{g}_j) \quad j = 0 \ldots k-1$ du mot reçu avec chacun des vecteurs lignes $\vec{g}_j$ de la matrice génératrice. Si $w(\vec{\epsilon}_j) < \mu_j/2$ poser $\lambda_j = 0$ sinon $\lambda_j = 1$.

- Avec les coefficients λ_j déterminés ci dessus, le mot le plus proche de $\vec{r}$ pour la distance de Hamming est

$$\vec{c} = \sum_{j=0}^{k-1} \lambda_j \vec{g}_j.$$

Preuve

Soit

$$\vec{c} = \sum_{j=0}^{k-1} \lambda_j \vec{g}_j$$

et soit $\vec{e}$ un vecteur satisfaisant aux conditions du lemme.

Si pour un indice j l'égalité $\lambda_j = 1$ est vraie pour $\vec{c}$ alors le poids du vecteur de corrélation de $\vec{c}$ et $\vec{g}_j$ est égal à μ_j. Il s'en suit que $w(\epsilon_j) > \mu_j/2$. Si par contre $\lambda_j = 0$ alors le vecteur de corrélation de $\vec{c}$ et $\vec{g}_j$ vaut $\vec{0}$ et par conséquent $w(\epsilon_j) < \mu_j/2$.

Cette méthode permet de déterminer les coordonnées du mot de code le plus proche de $\vec{r}$ sur chacun des supports. En effet, l'égalité suivante est vérifiée pour les codes utilisés :

$$d(\vec{r}, \vec{c}) = \sum_{i=0}^{k-1} d(h(\vec{r}, \vec{g}_i), \vec{g}_i).$$

Etant donné que les termes du membre de droite sont positifs et que la méthode proposée pour le calcul des λ_i minimise chacun d'entre eux, il en résulte que le membre de gauche est minimum.

Etant donné l'unicité du mot de code le plus proche d'un vecteur, la démonstration est achevée.

5. le cryptosystème

Le nouvel algorithme utilise les propriétés énoncées. Il fait intervenir des matrices génératrices de plusieurs codes. Le chiffrement consiste à calculer un mot d'un code choisi aléatoirement, puis celui-ci est perturbé par des erreurs correspondant à l'information à chiffrer.

5.1. les clés de chiffrement et déchiffrement

Si A et B désirent échanger de l'information de manière secrète il faut qu'un centre habilité choisisse des paramètres n et k et une famille de permutations, $\sigma_j \quad j = 1, \ldots, l$ de l'ensemble $\{0, \ldots, n-1\}$ ainsi que des partitions $\mathcal{P}_j \quad j = 1, \ldots, l$ de l'ensemble des indices $\{0, \ldots, n-1\}$. Le centre attribue ces permutations et ces partitions à A et B. Ces données sont gardées secrètes par les deux intervenants. Pour chacune de ces partitions $\mathcal{P}_j$, A et B calculent les matrices $G_j \quad j = 1, \ldots, l$ des codes aléatoires correspondants. Chacune de ces matrices a la propriété que tout couple de vecteurs lignes est à supports disjoints. Les deux intervenants calculent ensuite les permutées de ces matrices à l'aide des permutations $\sigma_j \quad j = 1, \ldots, l$ qui sont encore notées $G_j \quad j = 1, \ldots, l$. Cet ensemble est noté S.

Ces données, qui sont propres au cryptosystème et doivent être gardées secrètes, sont appelées les *clés du système*. Toutefois, pour mettre en œuvre l'algorithme, une donnée supplémentaire doit être connue du chiffreur et du déchiffreur : l'indice (ou le numéro) de la matrice utilisée pour chiffrer un message. Cette quantité, qui doit être également gardée secrète, est appelée la *clé de message*.

5.2. le chiffrement

Il comporte trois étapes.

1. L'intervenant A choisit la matrice G_i, pour la valeur de l'indice donnée par la clé de message, dans l'ensemble S. Cette matrice a des vecteurs lignes $\vec{g}_{ji} \quad 0 \leq j \leq k-1$ de poids μ_{ji}.
2. A choisit alors une combinaison linéaire à coefficients $(0, 1)$ des vecteurs lignes $\vec{g}_{ji}$, soit

$$\vec{r} = \sum_{j=0}^{k-1} \lambda_j \vec{g}_{ji}.$$

   $\vec{r}$ doit être de poids voisin de $n/2$. Ce vecteur s'obtient par un tirage aléatoire des coefficients λ_j effectué avec un générateur de densité 1/2.
3. Au message à chiffrer est associé $\vec{m}$ un vecteur de longueur n dont le poids sur le support de chaque $\vec{g}_{ji}$ est inférieur à $\lfloor \mu_{ij}/2 \rfloor$ pour tout j. Le cryptogramme est le vecteur

$$\vec{c} = \vec{r} + \vec{m}$$

   qui est alors transmis à B.

Remarque

La construction de $\vec{m}$ présuppose l'existence d'un algorithme de codage à poids constant. Un tel algorithme est donné dans [7]. Il est rappelé au paragraphe consacré à la réalisation.

5.3. le déchiffrement

Le déchiffrement comporte trois étapes. B doit d'abord déterminer le code utilisé et ensuite $\vec{m}$ en effectuant des calculs pour trouver $\vec{r}$.

1. Détermination de la matrice G_i : A l'aide de la clé de message le destinataire du cryptogramme B choisit la matrice G_i dans l'ensemble S.

2. Discrimination algébrique : B calcule, à l'aide de $\vec{c}$, le mot du code

$$\vec{r} = \sum_{j=0}^{k-1} \lambda_j \vec{g}_{ji}$$

qui a été utilisé par A. Ceci veut dire qu'il doit trouver les coefficients $\lambda_j, \quad j = 0, \ldots, k-1$.
A cette fin B calcule, pour tous les j, la suite des vecteurs de corrélation $h(\vec{c}, \vec{g}_{ji})$ de $\vec{c}$ avec chacun des vecteurs lignes $\vec{g}_{ji}$ de G_i. Si, pour chaque indice j, le poids du résultat est supérieur à $\lfloor \mu_{ji}/2 \rfloor$, il attribue la valeur 1 à λ_j sinon il pose $\lambda_j = 0$.

3. Reconstitution de l'information : Au terme de cette étape, B possède une estimation de $\vec{r}$ donnée par

$$\vec{r} = \sum_{j=0}^{k-1} \lambda_j \vec{g}_{ji}$$

dont il fait la somme, composante à composante, avec $\vec{c}$ pour en déduire $\vec{m}$.
Pour déduire le message à partir de $\vec{m}$, B applique un algorithme inverse de l'algorithme de « codage à poids constant » aux diverses suites binaires, obtenues en restreignant $\vec{m}$ aux supports des vecteurs $\vec{g}_{ji}$ pour toutes les valeurs de j.

Il est à noter que les seuls calculs qui doivent être effectués pour décoder sont des calculs de corrélation et de poids.

6. bornes sur le taux de transmission

Définition 5

Le taux de transmission d'un cryptosystème par bloc qui, à m bits de clair associe un cryptogramme de n bits, est le rapport m/n.

Le système présenté introduit peu de redondance, contrairement aux autres systèmes de chiffrement utilisant les codes. Pour en faciliter l'étude il est utile de considérer d'abord un cas particulier.

Définition 6

Une matrice de chiffrement est dite régulière si tous ses vecteurs ligne sont à supports disjoints et de même poids.

6.1. cas des matrices régulières

Chacune des matrices $G_j \quad j = 1, \ldots, l$ est constituée de k vecteurs de longueur n de même poids dont les supports sont deux à deux disjoints. Il s'en suit que $\mu = n/k$. Soit $t = \lfloor \frac{\mu}{2} \rfloor$. Un vecteur $\vec{e}$ correspond à un message décodable pour la matrice G_j si pour tout vecteur ligne $\vec{g}_{ij}$ le poids de $h(\vec{e}, \vec{g}_{ij})$ est inférieur à t. Le nombre de tels vecteurs $\vec{e}$ est facile à dénombrer : Pour un des vecteurs lignes $\vec{g}_{ij}$ de poids μ il y en a

$$1 + \binom{\mu}{1} + \binom{\mu}{2} + \ldots + \binom{\mu}{(\mu-1)/2} = 2^{\mu-1}$$

si μ est impair ou bien

$$1 + \binom{\mu}{1} + \binom{\mu}{2} + \ldots + \binom{\mu}{\mu/2} = 2^{\mu-1} - \frac{1}{2}\binom{\mu}{\mu/2+1}$$

si μ est pair.

Ce nombre concerne les composantes de $\vec{e}$ correspondant au support d'un unique vecteur ligne. Pour les autres vecteurs lignes de la matrice génératrice, la même énumération s'applique. Etant donné que les configurations peuvent être choisies indépendamment les unes des autres, les choix des composantes des configuration d'erreurs corrigibles sur les différents supports des vecteurs lignes n'interfèrent pas entre eux, ce qui montre qu'il y en a au plus $2^{(\mu-1).k}$.

Il en résulte que le taux de transmission du cryptosystème est approximé par la formule :

$$R \approx \frac{\log_2(2^{(\mu-1).k})}{n} = \frac{(\mu-1)k}{n} = \frac{k.(\mu-1)}{k.\mu} = \frac{\mu-1}{\mu}$$

6.2. cas général

Chacune des matrices $G_j \quad j = 1, \ldots, l$ est constituée de vecteurs de longueur n et de k lignes dont les supports sont disjoints et chaque ligne est de poids μ_{ji}. Comme dans le cas précédent un vecteur $\vec{e}$ correspond à un cryptogramme décodable pour une matrice G_j si pour tout vecteur ligne $\vec{g}_{ij}$ le poids de $h(\vec{e}, \vec{g}_{ij})$ est inférieur à $\lfloor \frac{\mu_{ji}}{2} \rfloor$.

L'ensemble de tels vecteurs $\vec{e}$ est facile à dénombrer sur chacun des supports : pour un des vecteurs lignes $\vec{g}_{ij}$ il y en a :

$$1 + \binom{\mu_{ji}}{1} + \binom{\mu_{ji}}{2} + \ldots + \binom{\mu_{ji}}{(\mu_{ji}-1)/2} \approx 2^{\mu_{ji}-1}$$

L'égalité est atteinte si μ_{ij} est impair. Pour l'ensemble des vecteurs de la matrice génératrice, les choix des configurations d'erreurs corrigibles, sur les différents supports, n'interfèrent pas entre eux. Les configurations peuvent donc être choisies indépendamment les unes des autres. De ce fait il y en a au total :

$$2^{(\mu_{j0}-1)} \ldots 2^{(\mu_{jk-1}-1)}.$$

Le taux de transmission du cryptosystème est alors approximé par la formule :

$$R \approx \frac{\log_2(2^{(\mu_{j0}-1)} \ldots 2^{(\mu_{jk-1}-1)})}{n} = \frac{(\mu_{j0}-1) + \ldots + (\mu_{jk-1}-1)}{n} = \frac{n-k}{n}$$

Remarque

Cette approximation montre que le taux de transmission du cryptosystème ne dépend pas de la distance minimale du code, mais seulement de sa redondance, en raison de l'algorithme de décodage qui est utilisé.

7. justification de la méthode

Le schéma de cryptographie repose sur le principe suivant : L'unicité du décodage pour un code correcteur. Le code C utilisé est une concaténation de codes à répétition de longueur variable. Un mot du code, ici $\vec{r}$, est modifié en $\vec{c}$ par l'addition, composante par composante, d'une configuration d'erreur $\vec{e}$, corrigible sur chacun des sous codes définissant la concaténation. Il s'ensuit que le décodage par maximum de vraisemblance de $\vec{c}$, appliqué à chacun des sous-codes de la concaténation, donnera toujours $\vec{r}$ même si le poids de la configuration d'erreur $\vec{e}$ est supérieur au rayon de correction du code C. Cette propriété de décodage au delà de la distance minimale des codes concaténés est bien connue et fort utilisée.

8. solidité du système

L'objectif que peut se fixer un cryptanalyste est de rechercher l'ensemble des matrices secrètes G_j pour $j = 1, \ldots, l$ en observant des cryptogrammes. En effet une fois ces matrices obtenues, il lui est alors possible de déchiffrer les cryptogrammes en appliquant l'algorithme de déchiffrement.

Pour déterminer la complexité de cette recherche il est utile d'étudier celle d'une cryptanalyse, à clair connu, qui utilise les propriétés de la classe de codes utilisée. La probabilité de succès de ce type d'attaque sera donnée comme une fonction des paramètres du système.

8.1. cryptanalyse à clair connu

8.1.1. matrice de chiffrement unique

Si l'ensemble S des matrices de chiffrement est réduit à une seule matrice, et que l'adversaire enregistre l'ensemble des cryptogrammes transitant sur la ligne, il lui est possible d'obtenir des informations. Si en plus le message en clair $\vec{i}$ est connu cela revient à choisir $\vec{i} = 0$ car le système est linéaire. Dans ce cas, les cryptogrammes sont des mots du code. Il est possible de définir récursivement le produit de Hadamard d'un ensemble de l vecteurs comme étant la corrélation de n'importe lesquels d'entre eux avec la corrélation des $l - 1$ restants. Ainsi, si $l = 3$, le produit de Hadamard $h(\vec{c}_i, \vec{c}_j, \vec{c}_k)$ de $\vec{c}_i, \vec{c}_j, \vec{c}_k$ est défini par

$$h(\vec{c}_i, \vec{c}_j, \vec{c}_k) = h(\vec{c}_i, h(\vec{c}_j, \vec{c}_k)).$$

Soit $\vec{c}_1, \vec{c}_2, \vec{c}_3 \ldots$ une suite de mots du code, récupérée par le cryptanalyste. Il lui est alors possible de construire la suite de vecteurs suivants :

$$h(\vec{c}_1, \vec{c}_2), \quad h(\vec{c}_1, \vec{c}_2, \vec{c}_3), \ldots$$

Cette suite est constituée de vecteurs dont les poids décroissent vers 0, en ayant pour valeurs des indices non nuls du polynome énumérateur de poids. Or ces codes ont la propriété que le produit de corrélation de deux mots du code est encore un mot du code.

L'avant dernier vecteur de la suite précédente est donc un vecteur ligne de la matrice G. Puisque G est constituée de k vecteurs lignes, au moins $\log_2 k$ produits de corrélation sont nécessaires pour obtenir un des vecteurs lignes de la matrice G.

En renouvelant le choix des vecteurs initiaux au moins k fois cette méthode permet d'obtenir l'ensemble des vecteurs lignes de la matrice G.

Une estimation de la matrice G s'obtient donc après le calcul de $O(k.\log_2 k)$ produits de corrélation. Cette cryptanalyse est très efficace puisque la valeur de k qui est envisagée est $k \approx 100$.

8.1.2. cas d'un ensemble de matrices régulières

Supposons que l'ensemble S soit constitué de l éléments $G_1, \ldots, G_l$ dont tous les vecteurs lignes sont de même poids μ.

Dans l'hypothèse où l'adversaire a accès à une suite $(\vec{c}_i)$ de cryptogrammes, la suite des poids des vecteurs de corrélation qu'il peut calculer est décroissante et tend vers 0. Il existe donc t tel que

$$h(\vec{c}_1, \vec{c}_2, \ldots, \vec{c}_{t+1}) = 0$$

et

$$\vec{c} = h(\vec{c}_1, \vec{c}_2, \ldots, \vec{c}_t) \neq 0.$$

Si tous les vecteurs $\vec{c}_i$ de la suite appartiennent au même code, le cryptanalyste réussira à déterminer un vecteur ligne de l'une des matrices comme dans le cas précédent. Toutefois si les vecteurs intervenant dans les produits de Hadamard appartiennent à plus d'un code, la suite construite par le cryptanalyste convergera vers un vecteur n'appartenant à aucun des codes. Par conséquent il ne peut en tirer d'information correspondant à son objectif.

Pour parvenir à reconnaître les vecteurs $\vec{c}_i$ provenant d'un même code, P. Camion [5] suggère la méthode suivante : Deux vecteurs $\vec{c}_i$ et $\vec{c}_j$ proviennent presque surement d'un même code si le poids

du vecteur somme est un multiple de μ. Cette remarque permet de regrouper les vecteurs d'un même code et la détermination de la matrice génératrice s'effectuera en appliquant la méthode de cryptanalyse du cas de la matrice de chiffrement unique.

8.1.3. cas général

Si les matrices sont régulières, mais que les $\mu_i = \text{poids}(\vec{g}_{ji})$ sont distincts et inconnus, la méthode suggérée ne s'applique plus. En effet le cryptanalyste ne dispose alors d'aucune information lui permettant de regrouper les mots d'un même code. Seules des hypothèses probabilistes permettent d'estimer la probabilité de succès de cette cryptanalyse dans le cas général.

La probabilité qu'un vecteur $\vec{c}_i$ appartienne à une matrice particulière G_j vaut $1/l$. Les matrices étant choisies indépendamment les unes des autres, la probabilité qu'un couple de vecteurs appartienne au même code vaut $p_2 = l^{-2}$, plus généralement la probabilité que t vecteurs appartiennent au même code vaut $p_t = l^{-t}$. On constate alors que la probabilité d'obtention d'un vecteur ligne d'une des matrices génératrices, par calcul de corrélation itéré, vaut

$$P = l^{-k}.$$

Si $k \approx 60$ et $l \approx 50$ cette probabilité est de l'ordre de 50^{-60} et peut donc être considérée comme négligeable.

8.2. cryptanalyse à l'aide des seuls cryptogrammes

Dans l'hypothèse plus réaliste où le cryptanalyste ne dispose que de cryptogrammes ne correspondant à aucun clair connu, il lui est possible d'utiliser une telle tentative de cryptanalyse. Si les vecteurs d'erreur associés au texte en clair sont de petit poids, la probabilité de succès de l'attaque, bien que non nulle, sera inférieure à celle obtenue aux paragraphes précédents. Elle fournira des fragments de vecteurs lignes.

Il est possible de se prémunir contre de telles éventualités en restreignant le codage des informations aux erreurs de poids minoré, au prix d'une diminution du taux de transmission.

Il convient de remarquer que les attaques développées contre le système de McEliece sont sans objet pour ce cryptosystème. En effet, pour mettre en œuvre celles-ci, il est nécessaire de connaître les matrices génératrices des codes qui sont utilisées. Avec le présent algorithme cela n'est guère possible car les mots de code qui transitent sur la ligne de communication sont toujours entachés d'erreur. La probabilité d'arriver à reconstituer complètement une des matrices génératrices est donc un évènement de probabilité négligeable.

9. avantages par rapport à d'autres systèmes utilisant des codes

9.1. le chiffrement

Les codes utilisées font partie d'une famille de grande cardinalité. Un grand choix de paramètres est possible, en fonction des contraintes de l'application envisagée.

Le chiffrement dans le sytème de McEliece n'est soumis à aucune contrainte particulière. Il consiste à effectuer le produit d'un vecteur à 512 composantes par une matrice pour obtenir un vecteur à 1024 composantes.

Dans le cas de la version Niederreiter, une application de transcodage est nécessaire, par ce qu'il est classique d'appeler « codage à poids constant ». Cette opération utilise du calcul multiprécision en raison de la valeur des entiers qui interviennent dans le calcul. Elle est préalable à un calcul de syndrome qui est également un produit matriciel de même caractéristique algorithmique.

Dans le nouveau système, une opération de transcodage du même type est également nécessaire pour calculer un cryptogramme à partir d'un clair. Toutefois, appelée plusieurs fois avec des paramètres sensiblement inférieurs, elle est d'exécution beaucoup plus rapide et ne nécessite pas de multiprécision. Elle est suivie de deux produits matriciels faisant intervenir des quantités de dimensions inférieures à celles des algorithmes de référence.

9.2. le déchiffrement

Dans les deux systèmes de référence le déchiffrement comporte la phase de décodage d'un code de Goppa, qui est un algorithme de complexité quadratique dans le nombre d'erreurs. Dans le cas de Niederreiter le décodage est suivi d'une opération de reconstitution d'entiers à partir du codage à poids constant, qui fait intervenir des entiers en multiprécision.

Pour le nouvel algorithme le déchiffrement comporte deux calculs. Le premier consiste en une opération de décodage de codes concaténés par décision majoritaire. Ceci implique qu'il est possible d'appliquer une stratégie de type « diviser pour régner » en effectuant un décodage par sous ensemble de composantes correspondant aux supports des vecteurs lignes de la matrice génératrice. Une réduction de complexité en résulte et une parallélisation est possible.

Chacun des décodages est une opération de complexité presque linéaire. Elle consiste à calculer des produits de corrélation et effectuer une décision majoritaire après un calcul de poids, sur des codes de petite dimension.

Le décodage se termine par une opération de reconstitution d'entiers à partir du codage à poids constant. Là encore, les valeurs des paramètres qui interviennent en font une opération très rapide.

10. détails de réalisation et valeurs numériques

Pour définir chaque matrice génératrice secrète, il faut fournir des données permettant de construire les vecteurs lignes de la matrice génératrice ainsi que les permutations des colonnes.

10.1. les vecteurs lignes de matrice génératrice

Les vecteurs lignes de la matrice génératrice sont caractérisés par le fait qu'ils sont à supports disjoints. La matrice initiale est formée de « marches » de longueurs variables. Pour décrire cette structure, il est nécessaire et suffisant de donner les coordonnées des extrémités des supports de chaque vecteur ligne. Ceci revient à se donner, pour une matrice de dimension k et de longueur n, une suite croissante de $k-2$ entiers inférieurs à n.

Ceci est illustré dans l'exemple suivant où les paramètres retenus sont $n = 150$, $k = 20$, la suite définissant la matrice étant $(0, 6, 18, 22, 26, 30, 34, 54, 58, 64, 80, 84, 88, 94, 98, 110, 114, 132, 136, 146, 150)$. Dans toutes les représentations la valeur 1 est représentée par un point, la valeur zéro par un blanc.

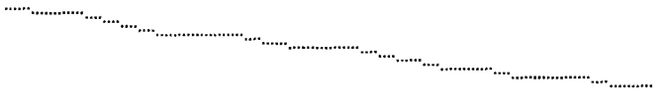


Figure 1. – Une matrice aléatoire non permutée.

L'application d'une permutation aux colonnes de la matrice génératrice préserve la propriété caractéristique de la matrice génératrice tout en détruisant la structure de « marche » de la matrice initiale. Ceci est illustré dans la figure 2.

Figure 2. – Une matrice aléatoire permutée.

Il est possible, mais non nécessaire, de masquer la propriété caractéristique des supports, en remplacant la matrice génératrice permutée par une matrice équivalente. Une matrice équivalente à celle de la figure 2 est donnée à la figure 3.

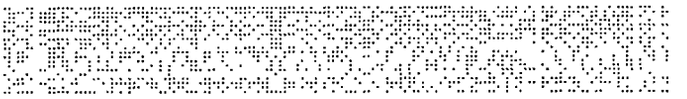


Figure 3. – Une matrice d'aspect aléatoire équivalente à la précedente.

10.2. la permutation des colonnes de la matrice génératrice

Chaque permutation des colonnes de la matrice génératrice, qui est précalculée pour chacune des matrices intervenant dans le cryptosystème, est donnée par l'image de la suite des indices des coordonnées des vecteurs. Cela revient à se donner une suite de n entiers pour décrire une permutation des n coordonnées.

10.3. codage à poids constant

Le cryptosytème suppose qu'il est possible de transformer une suite binaire sans contrainte particulière en vecteurs binaires de longueur p et de poids h, les paramètres p et h prenant plusieurs valeurs correspondant aux poids des vecteurs lignes des matrices génératrices G_i.

Etant donné une suite binaire et des paramètres p et h, il faut d'abord subdiviser la suite en blocs de longueur $\lfloor \log_2 \binom{p}{h} \rfloor$ bits. Chacun de ces blocs doit être considéré comme la représentation binaire d'un entier n, $0 \leq n \leq \binom{p}{h}$. Pour transformer de tels entiers en vecteurs binaires de longueur p et de poids h, il est possible de se servir de l'application induite par l'ordre lexicographique des vecteurs : Si n est supérieur à $\binom{p-1}{h}$ le premier bit du vecteur est mis à 1, sinon il est mis à 0, en adoptant la convention que $\binom{n}{k} = 0$ si $n < k$. Il faut ensuite mettre à jour les valeurs de p et de h et itérer p fois afin de déterminer les p bits du résultat. Ceci conduit à l'algorithme suivant :

- entrée : n, p, h, sortie $\vec{y}$
- pour i allant de 1 à p faire
- si $n \geq \binom{p-i}{h}$ alors
- mettre $y_i = 1$
- effectuer $n = n - \binom{p-i}{h}$
- effectuer $h = h - 1$
- sinon mettre $y_i = 0$
- renvoyer $\vec{y}$

L'algorithme inverse qui, à un vecteur de longueur p et de poids h, renvoie un entier n inférieur ou égal à $\binom{p}{h}$ est de même complexité.

- entrée : $\vec{y}$,p, h; sortie n.
- $n = 0$

- pour i allant de 1 à p faire
- si $y_i = 1$ alors
- poser $n = n + \binom{p-i}{h}$
- poser $h = h - 1$
- renvoyer n.

10.4. taux de transmission réalisables

Le choix des paramètres d'un cryptosystème repose sur plusieurs considérations. Pour que l'attaque par recherche exhaustive de tous les mots d'un code ne soit pas possible, la dimension du code doit être supérieure à un seuil de sécurité. La valeur $k = 70$ a été retenue.

Si le taux de transmission attendu est de 0,9, la longueur n du code doit être voisine de 700. Ce sont ces paramètres qui ont été retenus dans les tableaux de résultats qui vont suivre.

Le premier présenté montre que le taux de transmission du cryptosystème ne s'améliore que lentement lorsque la valeur de n croit. Les valeurs présentées sont obtenues grâce au choix d'une suite particulière définissant un code utilisable. Ils ont toutefois une portée plus générale car les nombres de ce tableau ne varient pas de manière significative pour d'autres choix de suites d'entiers définissant des code de même paramètres. Il est à noter que les résultats obtenus sont très voisins des maxima donnés au paragraphe 7.

longueur du code	dimension du code	nombre de bits du clair	taux de transmission
601	70	527	0,88
700	70	622	0,89
800	70	716	0,90
901	70	817	0,91
1000	70	902	0,90

Figure 4. – Tableau de performances : version générale.

Comme il a été dit auparavant, les cryptogrammes associés aux messages donnant lieu à des configurations d'erreur de petit poids fournissent plus d'information à un éventuel cryptanalyste que les autres. Les résultats donnés au tableau suivant sont obtenus en utilisant les mêmes codes que précédemment, en imposant une contrainte de poids d' erreur sur les supports de chaque vecteur ligne.

Seules sont codées les informations dont le vecteur d'erreur est de poids au moins égal au poids du support divisé par 4, pour les vecteurs lignes de poids supérieur à 4. Il en résulte une baisse du taux de transmision du cryptosystème. De telles configurations sont peu nombreuses.

Au vu des résultats numériques obtenus il apparaît que cette condition, qui augmente la résistance aux attaques en cryptanalyse, est peu contraignante en taux de transmission.

longueur du code	dimension du code	nombre de bits du clair	taux de transmission
601	70	518	0,86
700	70	609	0,87
800	70	696	0,87
901	70	805	0,89
1000	70	882	0,88

Figure 5. – Tableau de performances : version modifiée.

10.5. volume des clés

Dans le cadre d'une réalisation pratique, il est légitime de se préoccuper du volume des clés du cryptosystème et de message. L'étude est faite pour un système de 50 matrices, chacune étant de longueur 700 et de dimension 70.

Chaque code est défini par une suite de 70 entiers définissant les supports des vecteurs lignes de la matrice génératrice. Les entiers de la suite, qui sont tous inférieurs à la longueur de bloc (ici 700), sont codables sur 10 bits, soit un volume total de 700 bits.

La permutation associée à chaque code est réprésentée par la donnée de 700 entiers inférieurs à 700, qui sont codés sur 10 bits chacun, soit sur un volume de stockage de 7000 bits.

Au total chaque matrice est représentée par 7700 bits. Si l'ensemble S des matrices est de cardinalité 50, le volume global pour décrire le cryptosystème est de 385 000 bits soit moins de 50 Koctets.

La clé de message est codée sur un octet.

Il est possible de réduire ce volume de 25 % environ en codant les différences entre les entiers au lieu de leurs valeurs absolues.

Il est à noter que le volume des clés pour le système de Niederreiter est de l'ordre de 300 Koctets et celui de McEliece de 600 Koctets pour un taux de transmission de 1/2. Ces deux systèmes ne comportent pas de clé de message.

11. implantation logicielle ou matérielle

L'algorithme de décodage utilisé donne lieu à des réalisations informatiques très performantes, sous forme logicielle et matérielle, puisqu'il repose sur le calcul de corrélation de vecteurs. Ces calculs sont possibles en utilisant des opérations booléennes sur les mots de 32 ou 64 bits des machines les plus récentes. Cet algorithme a aussi la propriété d'être invariant quand les permutations sont appliquées aux indices des vecteurs. C'est le même algorithme qui est utilisé quel que soit l'ensemble de matrices S retenu, dès lors que la propriété de support disjoint est satisfaite.

Lors de réalisations informatiques logicielles des débits de 100 Kb/s ont été facilement atteints sur micro ordinateur de moyenne gamme, sans parallelisation du déchiffrement. Ces premières valeurs permettent d'entrevoir des débits de l'ordre de la dizaine de mégabits par seconde pour des versions réalisées sous forme de circuit intégré ASIC.

12. conclusions

Le cryptosystème présenté est cryptographiquement solide et de faible complexité algorithmique. Il donne lieu à des réalisations performantes, qui sont susceptibles d'être accélérées par un calcul en parallèle de certaines opérations au déchiffrement.

Il est comparable en taux de transmission et en complexité de calcul aux autres systèmes à clés secrètes.

BIBLIOGRAPHIE

[1] S.Harari « A correlation cryptographic scheme » *Proceedings EUROCODE 1990* Springer Verlag LNCS, pp 180–192, 1991.

[2] R.J. McEliece « A public key cryptosystem based on algebraic coding theory » *JPL Technical Report, CA May 1978.*

[3] H. Niederreiter « Knapsack type cryptosystems and algebraic coding theory », *Problems of control and Information theory* vol. 15, no. 2, pp 159-166, 1986.

[4] P.J. Lee and E.F. Brickell « An observation on the security of McEliece's Public Key Cryptosystem » *Eurocrypt 1988 Davos Switzerland* pp.153 - 157, May 1988.

[5] P. Camion « private communication » April 1991.

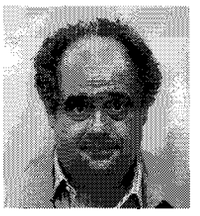
[6] T. Cover « An observation on the security of McEliece's Public Key Cryptosystem » *IEEE Transactions on Information Theory* pp.153 - 157, May 1973.

[7] B. Chor and R.Rivest « A knapsack type public key cryptosystem based on arithmetic in finite fields » *IEEE Transactions on Information Theory* vol. 34, pp. 901 - 909, Sept. 1988.

Manuscrit reçu le 25 Avril 1996

L' AUTEUR

Sami HARARI



Après des études de mathématiques à la Faculté des Sciences de Paris terminées par une thèse du 3e cycle en algèbre des catégories obtenue en 1971, l'auteur est nommé assistant en mathématiques à la faculté des Sciences d'Amiens en 1976, puis Maître Assistant en informatique à l'université de Toulon en 1984. Il soutient une habilitation à diriger des recherches sur les liens entre codage et cryptographie en 1992. Il est nommé Professeur en informatique en 1993. Il a collaboré avec plusieurs organismes de recherche d'état ainsi que les industriels du secteur. Il est titulaire de trois brevets d'invention dont deux portent sur des dispositifs cryptographies.