Systèmes dynamiques et chiffrement en continu

Floriane Anstett, Gilles Millérioux, Gérard Bloch

Systèmes dynamiques et chiffrement en continu

F. Anstett * G. Millerioux* G. Bloch*

1 Introduction

Depuis 1993, de nombreuses méthodes utilisant des systèmes dynamiques pour masquer une information ont été proposées. Les modèles dynamiques non linéaires peuvent exhiber des comportements complexes et, parmi eux, les comportements chaotiques se distinguent par leur grande sensibilité aux conditions initiales et à la valeur des paramètres du modèle. Ainsi, les signaux chaotiques sont des signaux à large spectre, imprédictibles à long terme et qui présentent des propriétés statistiques proches de l'aléatoire bien qu'issus de systèmes déterministes. Il est donc raisonnable de penser que les systèmes dynamiques et les comportements chaotiques puissent être utilisés à des fins de cryptage en garantissant des propriétés requises par les cryptosystèmes comme la confusion et la diffusion. Ainsi, le masquage chaotique [1], la modulation paramétrique [2], l'approche par systèmes inverses [3] ont été largement étudiés. Un aperçu de ces diverses méthodes est donné dans [4].

Cependant, jusqu'à présent, ces travaux relevaient plus de la sténographie que du cryptage proprement dit. Très peu de travaux ont réellement fait un lien entre les algorithmes de chiffrement standard et ceux basés sur la génération de séquences chaotiques. On notera cependant des études intéressantes dans [5][6]. L'objectif de cette note est de proposer un schéma original de chiffrement mettant à profit des propriétés des systèmes dynamiques, en particulier chaotiques, tout en essayant de répondre à quelques problèmes rencontrés en cryptographie usuelle.

2 Chiffrement en continu par inclusion

Avant d'exposer le principe du chiffrement en continu dit par inclusion, on rappelle tout d'abord la méthode usuelle de chiffrement symétrique en continu (*key stream cipher*) [7], qui est illustrée sur la Figure 1(a) . Le chiffrement en continu usuel obéit, côté émetteur, à :

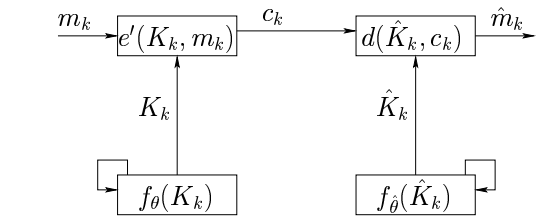
$$\begin{cases} K_{k+1} = f_\theta(K_k) \\ c_k = e'(K_k, m_k) \end{cases} \quad (1)$$

Ce chiffrement consiste à diviser un texte clair en blocs de longueur égale appelés symboles : on les note m_k. La fonction de chiffrement e' change pour chaque symbole, car elle dépend d'une clé variant dans le temps K_k et obéissant à une dynamique f_θ, paramétrée par θ qui joue le rôle d'une clé statique. En général, le symbole clair m_k et chiffré $c_k = e'(K_k, m_k)$ sont des mots binaires, la fonction e' est un simple XOR et la clé dynamique est générée itérativement par des registres à décalage bouclés. Le symbole chiffré c_k est disponible en sortie.

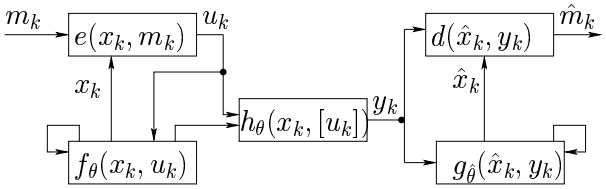
De façon générale, le chiffrement en continu par inclusion obéit, côté émetteur, à :

$$\begin{cases} x_{k+1} = f_\theta(x_k, u_k) \\ u_k = e(x_k, m_k) \\ y_k = h_\theta(x_k, [u_k]) \end{cases} \quad (2)$$

$[u_k]$ signifie que h_θ peut dépendre de u_k, mais pas nécessairement. Le principe est illustré sur la Figure 1(b).



(a)



(b)

FIG. 1: Chiffrement en continu, (a) usuel, (b) par inclusion

Chaque symbole m_k d'un texte clair et appartenant à un alphabet fini $\mathcal{M} = \{m_0, \ldots, m_N\}$ est chiffré successivement aux instants k. Le chiffrement de m_k s'effectue à l'aide d'une fonction $e : \mathcal{X} \times \mathcal{M} \to \mathcal{U}$ qui dépend d'une clé dynamique $x_k \in \mathcal{X}$. x_k est un état interne de dimension strictement supérieure à 1. Il en résulte un symbole chiffré $u_k = e(x_k, m_k) \in \mathcal{U}$. Le flot $\{x_k\}_{k=1,\ldots}$

*Université Henri Poincaré - Nancy 1, Centre de Recherche en Automatique de Nancy (CRAN, CNRS UMR 7039), CRAN - ESSTIN, 2 Rue Jean Lamour 54519 Vandoeuvre-Les-Nancy (France). Corresponding author E-mail: millerioux@esstin.uhp-nancy.fr

est généré par une dynamique non linéaire chaotique f_θ, où $\theta \in \Theta$ est un paramètre statique. Une caractéristique du chiffrement en continu par inclusion est que le symbole chiffré u_k n'est pas émis directement. En effet, seule la quantité scalaire $y_k = h_\theta(x_k, [u_k]) \in \mathcal{Y}$ est transmise, u_k étant injecté dans la dynamique f_θ. L'intérêt de cette caractéristique sera discuté dans les paragraphes suivants.

Ce principe général a été partiellement exposé dans [3][8][9]. Dans cette note, on propose une structure particulière où f_θ et h_θ sont des fonctions modulo caractérisées par des matrices, respectivement, $A_\theta \in \mathbb{Z}^{n\times n}$, $B_\theta \in \mathbb{Z}^n$ et $C_\theta \in \mathbb{Z}^{1\times n}$, $D_\theta \in \mathbb{Z}$, telles que :

$$\left\{ \begin{array}{ll} x_{k+1} = (A_\theta x_k + B_\theta u_k) & \mod 1 \\ y_k = (C_\theta x_k + D_\theta u_k) & \mod 1 \end{array} \right. \quad (3)$$

Ces matrices dépendent d'un paramètre $\theta \in \Theta = \mathbb{N}$ qui joue le rôle de clé statique et on a $\mathcal{X} = [0,1[^n \subset \mathbb{R}^n$, $\mathcal{Y} = [0,1[\subset \mathbb{R}$.

3 Récupération du texte clair

La récupération du texte clair nécessite une synchronisation des séquences des clés dynamiques à l'émission et à la réception.

$$\left\{ \begin{array}{l} \hat{K}_k = f_{\hat\theta}(\hat{K}_k) \\ \\ \hat{m}_k = d(\hat{K}_k, c_k) \end{array} \right. \text{(a)} \quad \left\{ \begin{array}{l} \hat{x}_{k+1} = g_{\hat\theta}(\hat{x}_k, y_k) \\ \\ \hat{m}_k = d(\hat{x}_k, y_k) \end{array} \right. \text{(b)} \quad (4)$$

Pour le chiffrement en continu usuel, la fonction d est telle que $\hat{m}_k = m_k$ si $\hat{K}_k = K_k$. Le flot de clés étant issu de récurrences autonomes, les générateurs de clés doivent être initialisés de part et d'autre de façon identique. Ainsi, la valeur initiale z_0 joue le rôle de la clé statique et on doit donc avoir $\hat\theta = \theta = z_0$.

Pour le chiffrement en continu par inclusion, la fonction de déchiffrement d est telle que $\hat{m}_k = m_k$ si $\hat{x}_k = x_k$. Contrairement au cas précédent, il n'est pas nécessaire que les deux générateurs de clés aient le même état initial x_0. En effet, $g_{\hat\theta}$ est choisie de telle sorte que si la clé statique vérifie $\hat\theta = \theta$, alors $\hat{x}_k = x_k$ quelle que soit $\hat{x}_0$ et indépendamment de u_k, plus précisément :

$$\begin{array}{ll} \lim_{k\to\infty} \|x_k - \hat{x}_k\| = 0, & \forall \hat{x}_0, \forall u_k \quad \text{(a)} \\ \text{ou } \exists k_f, \|x_k - \hat{x}_k\| = 0, & \forall \hat{x}_0, \forall u_k, \forall k > k_f \quad \text{(b)} \end{array} \quad (5)$$

Des structures particulières de $\hat{f}_{\hat\theta}$ assurant une *synchronisation globale indépendante de* u_k ont été introduites dans un contexte qui s'apparente au chiffrement dans [3][9][10].

Un premier avantage, par rapport au chiffrement en continu usuel, est que si la synchronisation est perdue pour une raison quelconque, cette méthode permet une resynchronisation automatique. Ce point est principalement du à l'inclusion de u_k dans la dynamique du système.

4 Cryptanalyse

Concernant la cryptanalyse, on aborde ici le problème particulier de la reconstruction de la clé.

Pour le chiffrement en continu standard, le recours à un registre à décalage à rétroaction linéaire (*Linear Feedback Shift Register*, LFSR) afin de générer des séquences de bits pseudo-aléatoires est un mécanisme très souvent utilisé. Cependant, Massey a prouvé en 1969 que la reconstruction de la totalité d'une séquence peut s'effectuer à partir d'un fragment d'une longueur finie et liée à la complexité linéaire du LFSR. Cela peut se révéler un problème majeur, car il est parfois possible, par une attaque à texte clair choisi, d'accéder à la séquence pseudo-aléatoire. Ainsi, si l'on se reporte au schéma de la Figure 1(a), on constate qu'en forçant m_k à 0 alors que e est la fonction XOR, on obtient effectivement $c_k = K_k$.

Pour l'algorithme de chiffrement en continu par inclusion, une question essentielle est également celle de la possibilité de recouvrement de la clé statique θ permettant de retrouver la séquence $\{x_k\}_{k=1,\ldots}$ et donc le texte clair. Cela rejoint le problème de l'identifiabilité paramétrique, notion empruntée à la théorie du contrôle. Le problème est de déterminer si le signal transmis y_k contient assez d'informations pour permettre à un pirate de reconstruire θ. Pour cela, il convient d'obtenir une relation liant l'entrée m_k, la sortie y_k, ainsi que la clé θ. Si l'on parvient à éliminer les états internes x_k et obtenir une relation entrée/sortie unique ayant pour forme générale :

$$\mathcal{L}_\theta(y_k, y_{k-1}, \ldots, m_k, m_{k-1}, \ldots) = 0 \quad (6)$$

avec $\mathcal{L}_\theta$ une fonction linéaire en θ, alors une attaque à texte clair choisi permettrait de tout reconstruire. Dans certains cas, il existe une infinité de paramètres θ et donc de séquences $\{x_k\}_{k=1,\ldots}$ conduisant à la même sortie y_k, c'est-à-dire à la relation (6). Dans d'autres cas, l'introduction d'une nonlinéarité f_θ rend impossible l'obtention de la relation (6). On se reportera en particulier à l'exemple étudié dans [11] dans le cas de la linéarité de type modulo (3) où la robustesse de l'algorithme de chiffrement à une attaque à texte clair choisi est ainsi prouvée.

5 Discussion

Une maquette de transmission d'images vidéo en temps réel via Internet basée sur le mécanisme de chiffrement par inclusion a été réalisée et présentée dans [12]. La Figure 2 montre l'image parfaitement restituée lorsque la clé est identique (a) et l'image incorrectement restituée lorsque la clé diffère (b).

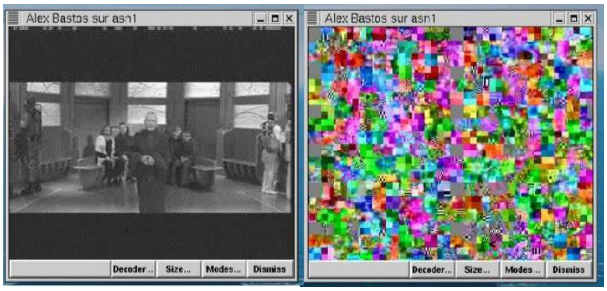


FIG. 2: (a) : clés identiques, (b) : clés différentes

En conclusion, l'usage d'un système dynamique pour la génération de la clé a permis d'aboutir à un mécanisme de chiffrement viable en pratique, assurant une synchronisation automatique et permettant d'obtenir des propriétés de robustesse sur le plan de la cryptanalyse paramétrique.

Néanmoins, différents points restent à considérer. Les algorithmes de chiffrement en continu utilisent des séquences binaires. L'algorithme proposé utilise des flots de clé réelles. Ainsi, il est théoriquement possible de générer des séquences chaotiques (équidistribuées dans la cas de récurrences modulo [11]). Que cela confère un avantage majeur du point de vue de la cryptanalyse statistique reste une conjecture à prouver. De plus, les effets de troncature dus à une implémentation hardware devront donc être examinés (périodicité et distribution des séquences).

Le chiffrement est robuste vis-à-vis de l'identifiabilité stricte de la clé statique. Néanmoins, il ne peut pas encore être, à ce stade, considéré comme complètement sécurisé. Une étude de la sensibilité par rapport à la clé doit être menée pour résister par exemple à une attaque gloutonne. Quelles autres attaques standards conviendrait-il de tester ?

On espère que cette note permettra d'engager une réelle discussion sur le lien possible entre systèmes dynamiques, chaos et cryptographie, ce qui fait partiellement défaut à ce jour.

Références

[1] K. M. Cuomo, A. V. Oppenheim, and S. H. Strogatz. Robustness and signal recovery in a synchronized chaotic system. *International Journal of Bifurcation and Chaos*, 3(6) : 1629–1638, 1993.

[2] H. Dedieu, M. P. Kennedy, and M. Hasler. Chaos shift keying : modulation and demodulation of a chaotic carrier using self-synchronizing Chua's circuits. *IEEE Trans. Circuits Syst. II : Anal. Digit. Sign. Process.*, 40 : 634–642, 1993.

[3] U. Feldmann, M. Hasler, and W. Schwarz. Communication by chaotic signals : the inverse system approach. *Int. J. of Circuit Theory Appl.*, 24 : 551–579, 1996.

[4] T. Yang. A survey of chaotic secure communication systems. *Int. J. of Computational Cognition*, 2(2) : 81–130, 2004. (available at `http ://www.YangSky.com/yangijcc.htm`).

[5] F. Dachselt, K. Kelber, J. Vandewalle, and W. Schwarz. Chaotic versus classical stream ciphers – a comparative study. In *Proc. of Int. Symp. on Circuits and Systems ISCAS'98*, volume IV, pages 518–521, Monterey, June 1998.

[6] L. Kocarev. Chaos-based cryptography : a brief overview. *IEEE Circuits and Systems Magazine*, 1(3) : 6–21, 2001.

[7] A. Menezes, P. Van Oorschot, and S. Vanstone. *Handbook of Applied Cryptography*. CRC Press, 2001.

[8] K.-Y. Lian, T.-S. Chiang, and P. Liu. Discrete-time chaotic systems : applications in secure communications. *International Journal of Bifurcation and Chaos*, 10(9) : 2193–2206, 2000.

[9] G. Millerioux and J. Daafouz. Unknown input observers for message-embedded chaos synchronization of discrete-time systems. *International Journal of Bifurcation and Chaos*, 14(4) : 1357–1368, 2004.

[10] G. Millerioux and J. Daafouz. An observer-based approach for input independent global chaos synchronization of discrete-time switched systems. *IEEE Trans. Circuits. Syst. I : Fundamental Theo. Appl.*, 50(10) : 1270–1279, 2003.

[11] L. Rosier, G. Millérioux, and G. Bloch. Synchronization of chaos for a class of dynamical systems on the $n-$torus. *Systems and control letters*, 2004. submitted.

[12] G. Millerioux, G. Bloch, J. M. Amigo, A. Bastos, and F. Anstett. Real-time video communication secured by a chaotic key stream cipher. In *Proc. of IEEE 16th Europ. Conference on Circuits Theory and Design ECCTD'03*, volume III, pages 245–248, Krakow, Poland, Sept. 1-4 2003.