Chiffrement évolutionniste

Fouzia Omary

Département de mathématiques et informatique Faculté des sciences-Rabat, Université MohammedV Maroc omaryfouzia@yahoo.fr

Abderrahim Tragha

Département de mathématiques et informatique Faculté des sciences Ben Msik Casablanca, Université HassanII-Mohammedia Maroc a.tragha@univh2m.ac.ma

Abdelghani Bellaachia

Département d'informatique, Université Georges Washington, Washington DC 20052 Etats-Unis bell@gwu.edu

Aboubakr Lbekkouri

Département de mathématiques et informatique Faculté des sciences-Rabat, Université MohammedV Maroc omaryfouzia@yahoo.fr

Abdelaziz Mouloudi

Département de mathématiques et informatique Faculté des sciences Kenitra, Université Ibn Tofail-Maroc mouloudi_aziz@hotmail.com

Dans cet article, nous présentons une application des algorithmes évolutionnistes à la cryptographie et notamment au chiffrement symétrique. Tout d'abord, nous avons ramené le problème de chiffrement à un problème d'optimisation. Puis, nous lui avons donné une formalisation semblable à celle utilisée pour la résolution, par algorithme évolutionniste, du problème du voyageur de commerce. Notre attention s'est portée ensuite sur le codage des chromosomes, sur l'établissement de la fonction d'évaluation et sur le choix des opérateurs génétiques adaptés. Du côté cryptographique, grâce à la clé symétrique générée par notre propre algorithme évolutionniste, nous illustrons les processus de chiffrement et de déchiffrement du message préalablement brouillé. Des exemples d'applications seront donnés à la fin de cet article, suivis d'une discussion d'évaluation de ce travail en comparaison avec d'autres algorithmes de même catégorie.

algorithmes génétiques, évolutionnistes, cryptographie, chiffrement, clés.

In this article, we present an application of evolutionist algorithms to cryptography and especially to symmetrical ciphering. First of all, we reduced ciphering problem to optimisation problem. Then, we assign to it a formalisation similar to the one used for the resolution of the TSP (travelling salesman problem) via evolutionary algorithm. Our attention focussed then on coding the chromosomes, establishing the fitness function and the choice of the adapted genetic operators. On the cryptographic side, we illustrate the processes of ciphering and decoding the message previously scrambled, thanks to the symmetrical key generated by our own evolutionist algorithm. Examples of applications will be given at the end of this article followed in the same way by an evaluative discussion of this work in comparison with other algorithms category.

Genetic algorithms, evolutionist, cryptography, ciphering, keys.

1 Introduction

Jadis, la cryptographie, la science du secret, a été strictement réservée aux milieux diplomatiques et militaires pendant plus de 3000 ans. Mais avec le développement de l'informatique et l'évolution des réseaux de communications, elle s'est imposée dans tous les domaines. Cependant, les vrais algorithmes de chiffrement sont peu nombreux, citons-en quelques exemples :

DES (Florin et Natkin, 2002), algorithme symétrique, mis au point en 1973, dont la clé secrète est de 64 bits. Une recherche exhaustive de la clé (1998) a permis aux crypto-analyseurs de casser ce fameux algorithme.

AES (Stinson, 2003), développé en 1997, a remplacé le DES. Il est symétrique et peut supporter des clés de longueurs égales à 128, 192 et 256 bits.

IDEA (Florin et Natkin, 2002), algorithme symétrique, assez récent, avec une clé de 128 bits.

RSA (Labouret, 2001), mis au point en 1970, est asymétrique et se base sur l'arithmétique modulaire pour définir sa clé publique et sa clé privée. Cet algorithme résiste toujours à la cryptanalyse.

Le PGP (Menezes, Oorschot et al., 1996) et (Manuel PGP 1998), mis au point en 1992, est une combinaison des meilleures fonctionnalités de la cryptographie asymétrique et de la cryptographie conventionnelle. PGP est un système hybride, il est actuellement, le crypto-système dominant.

Parallèlement, les domaines d'application des algorithmes évolutionnistes n'ont cessé de s'élargir grâce à leur grande efficacité (rapidité, simplicité de leurs opérateurs…). Ceci nous a incités à concevoir et à réaliser un système de chiffrement fondé sur les techniques des algorithmes évolutionnistes.

Cet article est structuré comme suit : dans la section 2, nous avons ramené le problème de chiffrement d'un message M à un problème d'optimisation. Ensuite, nous avons défini un codage adapté du problème**,** permettant ainsi une résolution qui s'inspire de celle du problème de voyageur du commerce (Caux, Pierreval et al. 1995) (en anglais TSP). Puis, nous avons construit notre algorithme en définissant la fonction d'évaluation adéquate, en précisant la nature des individus aptes à la sélection et en choisissant les opérateurs génétiques adaptés à ce problème. Le processus de déchiffrement est détaillé dans la section 3 et les résultats expérimentaux des différents exemples traités sont présentés dans la section 4. Dans la section 5, nous présentons les avantages de notre algorithme en comparaison avec des algorithmes importants du domaine de la cryptographie, avant de conclure en section 6.

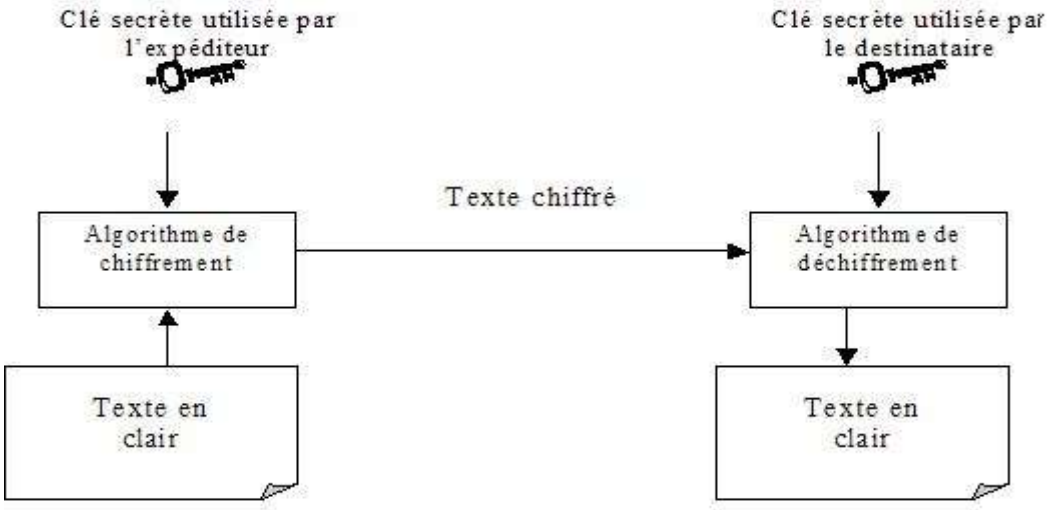


Figure 1: Algorithme de chiffrement symétrique

2 Description de notre algorithme de chiffrement

Soit M_0 le message à chiffrer. M_0 est une suite de *l* caractères. Ces derniers appartiennent à l'ensemble des 256 caractères du code ASCII .
Il est à noter que ce message peut être formé uniquement par des nombres (par exemple des codes bancaires d'un certain nombre de personnes), comme il peut être un mélange de nombres et de phrases littéraires y compris la ponctuation, etc.
Nous appliquons d'abord à M_0 un brouillage. Ce dernier peut être effectué par combinaison de plusieurs méthodes simples comme les substitutions, les permutations, le chiffrement affiné, etc. Cela nécessite une clé secrète symétrique. Le message brouillé sera désigné par M.

2.1 Formalisation du problème

Soient $c_1,c_2,\ldots,c_m$ les différents caractères de M ($l \leq m \leq 256$). Désignons par L_i ($1 \leq i \leq m$) la liste des différentes positions du caractère c_i dans M avant le chiffrement et par card(L_i) le nombre des occurrences de c_i dans M .
Remarque : $L_i \cap L_i$ est vide, $\forall i,j \in [1,k]$.
$L_1, L_2,\ldots, L_m$ est une partition de l'ensemble {1, 2, …, m}.
Le message M peut être représenté par le vecteur ci-dessous :

(c_1,L_1)	(c_2,L_2)	…	(c_m,L_m)

Tableau 1 : Original-Ch

Le but de notre travail est de modifier le plus possible les fréquences d'apparition des caractères dans le message M et d'établir le plus de désordre dans leurs positions. Pour cela, nous changeons itérativement la répartition des listes sur les différents caractères de M de telle manière que la différence entre le cardinal de la nouvelle liste L'_i affectée au caractère c_i et le cardinal de la liste initiale L_i soit maximale. Nous sommes alors devant un problème d'optimisation. Or les algorithmes évolutionnistes sont très efficaces pour la résolution de ce genre de problème, nous allons donc les utiliser, notamment ceux appliqués aux problèmes de permutations (Caux, Pierreval et al. 1995). Ces algorithmes se présentent sous plusieurs versions, la plus utilisée est celle décrite en section 2.2.

2. 2 Squelette de l'algorithme

Etape 0 : Définir un codage du problème
Etape 1 : Créer une population initiale P_0 de q individus{$X_1, X_2, \ldots, X_q$}
i := 0
Etape 2 : Evaluation des individus
Soit F la fonction d'évaluation. Calculer $F(X_i)$ pour chaque individu X_i de P_i
Etape 3 : Sélection
Sélectionner les meilleurs individus (au sens de F) et les grouper par paire.
Etape 4 : Application des opérateurs génétiques
1-Croisement : Appliquer l'opération de croisement aux paires sélectionnées
2-Mutation : Appliquer la mutation aux individus issus du croisement
Ranger les nouveaux individus obtenus (de 1 et 2) dans une nouvelle génération P_{i+1}
Répéter les étapes 2,3 et 4 jusqu'à l'obtention du degré de performance souhaité.

2.3 Notre système de chiffrement

Etape 0 : Codage
Un individu (ou chromosome) est un vecteur de taille m. Les gènes sont les listes L_{p_i} ($1 \le i \le m$). Le $i^{è}$ gène L_{p_i} contient les nouvelles positions que prendra le caractère c_i.

Etape 1 : Création de la population initiale P_0 composée de q individus ($X_1, X_2, \ldots, X_q$).
Nous désignons par Original-Ch le chromosome dont les gènes sont les listes $L_1, L_2, \ldots, L_m$ (placés dans cet ordre) qui représentent le message avant l'application de l'algorithme.
Nous appliquons q permutations sur Original-Ch afin d'obtenir une population initiale constituée de q solutions potentielles au problème.
i :=0.

Etape 2 : Evaluation des individus
Soit X_j un individu de P_i dont les gènes sont $L_{j_1}, L_{j_2}, \ldots, L_{j_m}$.

Nous définissons la fonction d'évaluation F sur l'ensemble des individus X_j par :

$$F(X_j) = \sum_{i=1}^{m} \left| card\,(L_k) - card\,(L_i) \right| \qquad [1]$$

Etape 3 : Sélection des meilleurs individus
Nous utilisons la méthode classique de la roulette (Florin et Natkin, 2002) permettant de retenir les individus les plus forts. Rappelons le processus :
On affecte à chaque individu X_i une probabilité d'apparition $p(X_i)$, (ou force relative) par :

$$p(Xi) = \frac{F(X_i)}{\sum_{k=1}^{q} F(X_k)} \qquad [2]$$

La sélection d'un individu se fait de la manière suivante :

Soit $q_i = \sum_{i=1}^{i} p(X_i)$ [3]

la probabilité d'apparition cumulée d'un individu X_i et soit r un nombre aléatoire compris entre 0 et 1, l'individu retenu est:

$$\begin{cases} X_1 & si \quad q_1 \le r \quad ou \\ X_i & si \quad q_{i-1} < r \le q_i \end{cases} \qquad [4]$$

où q est le nombre d'individus dans la population.
Ce processus est répété q fois. Avec ce principe, un individu fort peut être sélectionné plusieurs fois. Par contre, un individu faible a moins de chance d'être sélectionné.
Et nous introduisons une fonction de contrôle qui va éliminer les individus pour lesquels seulement une minorité de gènes ont changé de valeur par rapport au chromosome initial Original-Ch.
Puisque nous nous sommes ramené à un problème de permutations avec contraintes, nous appliquons alors les opérateurs génétiques adaptés à ce genre de problème.
Etape 4 : Applications des opérateurs génétiques

Croisement MPX (Maximal Preservative X)
Ce croisement a été proposé par (Mühlenbein et Schlierkamp-Voosen, 1993) pour le problème du voyageur de commerce. L'idée de cet opérateur est d'insérer une partie du

chromosome d'un parent dans le chromosome de l'autre parent de telle façon que le croisement résultant soit le plus proche possible de ses parents. C'est un croisement à deux points. Les deux fils sont obtenus de manière symétrique. Nous en illustrons le fonctionnement par un exemple.

Exemple :

Parent 1 :	a	b	c	d	e	f	g	h	i	j	k	l
Parent 2:	c	b	a	g	h	i	j	l	k	f	d	e

[i]

La zone de croisement est comprise entre les positions 5 et 9. La première étape consiste à recopier la zone de croisement du parent1 sur le fils1. Ensuite, les gènes du fils1 qui ne sont pas dans la zone de croisement sont complétés de la façon suivante:

Le i^{è} gène du parent2 est recopié sur le i^{è} gène du fils1 si cette copie respecte les contraintes(ne crée pas une tournée incohérente). Sinon, le i^{è} gène du parent1 est recopié sur le i^{è} gène du fils1 si cette copie ne crée pas de doublons. Si les deux cas précédents ne peuvent pas s'appliquer, le i^{è} gène du fils1 reçoit un gène de la zone de croisement du parent2 qui respecte les contraintes (premier non pris).

Fils 1 :	c	b	a	d	e	f	g	h	i	j	k	l
Fils 2 :	a	b	c	d	h	i	j	l	k	f	e	g

[ii]

Ce croisement est appliqué aux individus sélectionnés avec un taux bien précis. D'après (Labouret, 2001) le meilleur taux est de l'ordre de 60% à 100%.

Mutation de transposition

Nous choisissons la mutation qui consiste à permuter aléatoirement deux gènes d'un chromosome. Cet opérateur est appliqué aux individus issus du croisement avec un taux adapté, de préférence de 0,1% à 5% (Labouret, 2001).

Placer la nouvelle progéniture dans une nouvelle population P_{i+1}.

Répéter les étapes 2, 3 et 4 jusqu'à un critère d'arrêt.

Définir la condition d'arrêt

La fonction F est bornée car $0 \leq F(X) \leq 2*l$ pour tout individu X, en fait :

$$\sum_{i=1}^{m} \left| card\ (L_{k_i}) - card\ (L_i) \right| \leq \sum_{i=1}^{m} (card\ (L_{k_i}) + card\ (L_i)) \leq 2*l \qquad [5]$$

Théoriquement et mathématiquement parlant, la fonction F admet un maximum puisqu'elle est bornée. D'après certains résultats de recherche (Khan Phang, 1988) la convergence d'une fonction d'évaluation est assurée, mais peut être vers une valeur proche de Max, déterminée expérimentalement. En fait, cela se confirme dans les expériences que nous avons entamées et que nous interpréterons par la suite.

Dernière phase de l'algorithme

Désignons par Final-Ch la solution finale donnée par notre algorithme évolutionniste. A partir de Original-Ch et Final-Ch, nous construisons notre clé symétrique. Cette clé sera appelée clé génétique.

3 Déchiffrement

Le déchiffrement du message chiffré M' se fait en deux étapes :

Dans la première étape, nous allons représenter le texte chiffré, par un vecteur de listes comme nous l'avons fait pour le message en clair. Désignons alors par c'_1, c'_2, …, c'_m les différents caractères de M' et par L'_1, L'_2…, L'_m leurs listes respectives de position dans M'. Grâce à la clé génétique, les caractères vont retrouver leurs listes de position correspondantes

dans le message en clair. En effet, la clé est une permutation de {1,2,…,m} que nous pouvons représenter par un vecteur noté Clef, de taille m telle que :

Clef(1)=p_1 , Clef(2)=p_2 , …, Clef(i)=p_i , …, Clef(m)=p_m où

Le caractère c'$_{p_1}$ va être associé à la liste L_1'.

Le caractère c'$_{p_2}$ va être associé à la liste L_2'.

……

Le caractère c'$_{p_m}$ va être associé à la liste L'$_m$.

Ainsi, nous obtenons le message M.

Dans la deuxième étape, nous procédons au déchiffrement de M pour obtenir M_0. Le passage de M_0 à M se fait par le biais d'opérations ou fonctions simples de la cryptographie (substitution simple, décalage, chiffrement affiné…) et dont la clé est secrète et symétrique.

4 Expérimentation

Nous avons appliqué notre algorithme sur des messages de différentes tailles; mais nous n'avons retenu que des exemples modèles (quatre). Et pour chacun d'eux, nous avons fait l'application pour différentes tailles de population et nous en avons retenu les meilleures. Puis dans un tableau récapitulatif, nous avons enregistré les résultats importants, à savoir : valeur de la convergence de la fonction d'évaluation, nombre de générations atteint lors de cette convergence.

4.1 Les messages modèles

Le brouillage utilisé pour les quatre messages ci-dessous est un chiffrement affiné défini par : F(x)=ax+b où a=-1 , b=255 et x est le code ASCII du caractère du message initial.

Message 1 : Taille=1026

Travelling salesman problem (TSP) is an NP-complete problem, implying that there is no known algorithm to solve it exactly in better than O(an) time (for n cities). In fact, there are (n-1)!/2 possible combinations, so to try every one requires factorial time with the problem size. Just to remind, there are cities and given distances between them. Travelling salesman has to visit all of them, but he does not want to travel very much. The task is to find a sequence of cities to minimize travelled distance. In other words, find a minimal Hamiltonian tour in a complete graph of n nodes. A method was implemented for a 6-city problem. The optimal tour found for the 6 cities was: 1- 2 - 5 - 4 - 6 - 3 - 1 which has distance 9.242640 units.The processor times for the 5 and 6 city runs were below the timer resolution. The time for each combination can be estimated as 2e-07 seconds. The number of combinations for the 29 cities problem is 8.84e30. This gives a time estimate of 1.77e24 seconds, or 5.60e16 years.

Forme cryptée

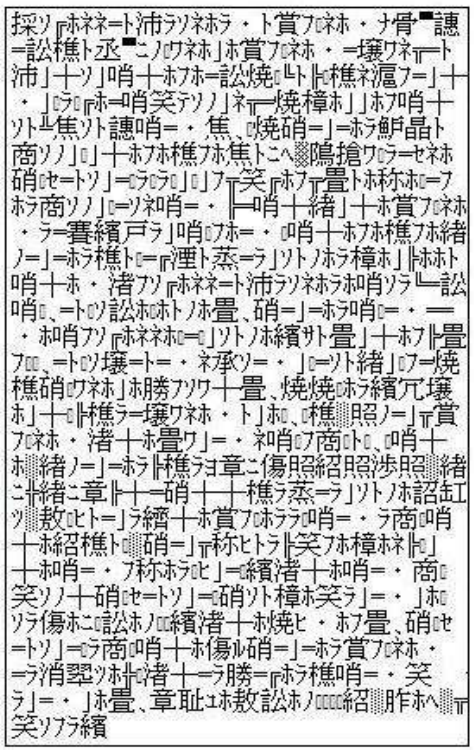


Figure 2. Le chiffrement du message 1

La clé est:
8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 4 5 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 6 7 51 23 52 0 1 2 3 25 29 24 27 28 30 26 31

Message2 Taille: 684 Forme cryptée

Astronauts In August 1983 Guion Bluford became the first African American to go into space, while serving on a mission aboard the Challenger space shuttle. Bluford said that the blastoff of the shuttle was like riding in a high-speed elevator through a bonfire. He also recognized that, "From a black perspective, my flight on the shuttle represented another step forward. "Astronaut Mae Jemison became the first African American woman to travel in space when she flew on the space shuttle Endeavor in a September 1992 mission. After her space flight, Jemison resigned from NASA and established the Jemison Group, a company that researches, develops, and markets advanced technologies.

Figure 3. Chiffrement du message 2.

La clé est:
8 9 10 11 12 13 14 15 16 17 18 19 6 7 22 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 21 1 23 24 25 2 5 4 20 26 0 3 28 27

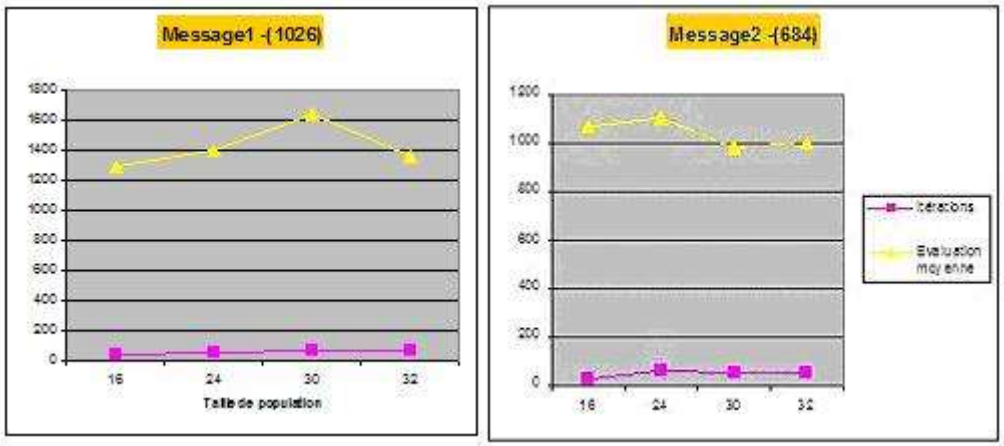


Figure 4. Résultats expérimentaux des messages 1 et 2.

Message 3 Taille: 1149

Abraham Lincoln, the 16th president of the United States, guided his country through the most devastating experience in its national history--the CIVIL WAR. He is considered by many historians to have been the greatest American president. Abraham Lincoln was born Sunday, February 12, 1809, in a log cabin near Hodgenville, Kentucky. He was the son of Thomas and Nancy Hanks Lincoln. Thomas Lincoln was a. Abraham had gone to school briefly in Kentucky and did so again in Indiana. He attended school with his older sister... Both of Abraham's parents were members of a Baptist congregation which had separated from another church due to opposition to slavery. When Abraham was 7, the family moved to southern Indiana As Abraham grew up, he loved to read and preferred learning to working in the fields. This led to a difficult relationship with his father who was just the opposite. Abraham was constantly borrowing books from the neighbors. Lincoln's declining interest in politics was renewed by the passage of the Kansas-Nebraska Act in 1854. In 1860 he furthered his national reputation with a successful speech at the Cooper Institute in New York.

Forme cryptée

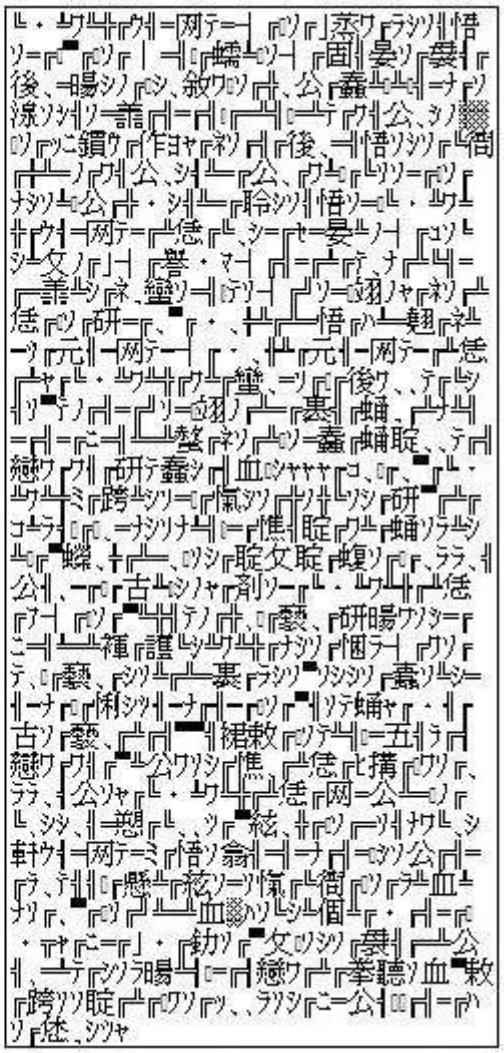


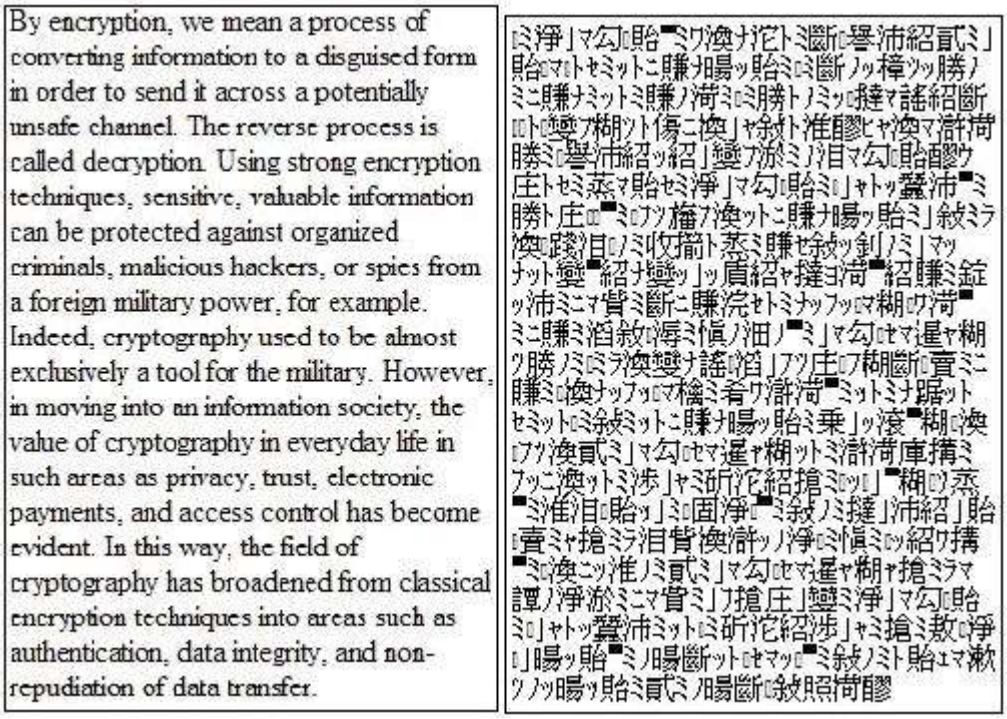
Figure 5. Le chiffrement du message 3

La clé est:

15 16 17 18 19 20 21 22 23 24 25 26 27 50 51 4 5 6 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 7 8 9 10 11 12 13 14 28 29 30 0 1 2 3 52 53

Message 4 Taille: 803

Forme cryptée



By encryption, we mean a process of converting information to a disguised form in order to send it across a potentially unsafe channel. The reverse process is called decryption. Using strong encryption techniques, sensitive, valuable information can be protected against organized criminals, malicious hackers, or spies from a foreign military power, for example. Indeed, cryptography used to be almost exclusively a tool for the military. However, in moving into an information society, the value of cryptography in everyday life in such areas as privacy, trust, electronic payments, and access control has become evident. In this way, the field of cryptography has broadened from classical encryption techniques into areas such as authentication, data integrity, and non-repudiation of data transfer.

Figure 6. Le chiffrement du message 4

La clé est :
9 10 11 12 13 14 15 16 17 18 19 24 25 26 27 28 29 30 31 32 33 0 1 2 3 22 4 5 7 8 23 20 21 6

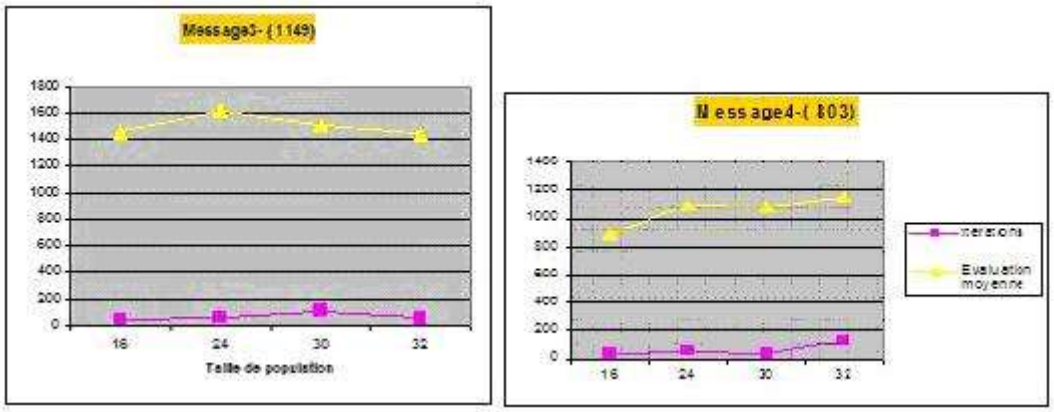


Figure 7. Résultats expérimentaux des messages 3 et 4

4.2 Résultats retenus

		Taille de la Population			
		16	24	30	32
Message1 1026 car.	Nombre de générations	35	58	71	64
	Valeur de CV	1288	1394	1638	1354
Message2 684 car.	Nombre de générations	26	58	46	53
	Valeur de CV	1068	1100	982	1002
Message3 1149 car.	Nombre de générations	44	58	111	53
	Valeur de CV	1458	1622	1512	1444
Message4 803 car.	Nombre de générations	34	51	38	130
	Valeur de CV	886	1096	1088	1160

Tableau 2 : Tableau récapitulatif des résultats

4.3 Interprétation

Pour la plupart des exemples traités dans nos expériences, nous constatons que les meilleures valeurs de l'optimum (valeur de convergence) sont atteintes pour une population de taille 24 (message 2 et message 3 dans figure1). Dans certains cas, les tailles 30 et 32 ont donné aussi de bons résultats, mais le nombre de générations dans ce cas a été doublé, ce qui est coûteux du point de vue temps. Nous conseillons alors de prendre 24 comme taille de la population.
Nous citerons ci-dessous les quatre messages modèles et donnerons les opérations utilisées pour brouiller le message et la clé de session générée par notre algorithme évolutionniste.

5 Discussion

Nous allons comparer notre algorithme avec l'un des algorithmes symétriques les plus connus, le DES.
En général, les algorithmes symétriques les plus connus (DES, IDEA, AES) font des chiffrements par blocs. Le nôtre chiffre le message tout entier en une seule prise, ce qui est moins coûteux.
Le DES a une clé de 64 bits (mais ne sont utilisés que les 56 bits). Cette taille si petite a favorisé son attaque par une recherche exhaustive de la clé (ou attaque par force brute). Dans notre algorithme, nous pouvons toujours supposer que le message contient plus de 30 caractères différents, donc la taille de la clé est au moins égale à 240 bits, taille résistante actuellement aux attaques. Dans le cas où le texte contient moins de vingt différents caractères nous pouvons le compléter par d'autres. En outre, notre clé est une clé de session (variable d'un message à l'autre) et est générée par notre propre algorithme. Par contre, celle du DES ne l'est pas. Or les clés de session sont plus résistantes aux attaques que les autres.

Autre point important à évoquer, on peut penser à une crypto-analyse par l'étude des fréquences d'apparition des caractères dans le message. Ce genre d'attaque s'appuie sur des études en linguistique déterminant les fréquences d'apparition des lettres d'un texte écrit dans une langue naturelle. Or, rappelons que nos messages ne sont pas exclusivement des messages littéraux sans ponctuation, ils peuvent contenir tous les caractères possibles du code ASCII. Puisqu'il n'y a aucune étude des fréquences de tous les caractères du code ASCII, une attaque de ce genre est à rejeter, d'autant plus que le brouillage initial que nous avons effectué rend l'attaque plus difficile puisque le crypto-analyste ne peut avoir une idée préalable sur les types des caractères du message.

6 Conclusion

Notre travail a apporté deux contributions simultanées. La première est l'élargissement de l'espace d'application des algorithmes évolutionnistes. Ces derniers n'ont prouvé leur efficacité qu'en crypto-analyse. En fait, l'unique et le meilleur algorithme de chiffrement asymétrique « Knapsack cipher » (Muller, 2003), inspiré de la résolution du problème du sac à dos, a été attaqué en utilisant les algorithmes génétiques. La deuxième contribution est la conception d'une nouvelle formalisation du problème de chiffrement par l'innovation d'un algorithme de chiffrement évolutionniste. Ce dernier bénéficie de tous les avantages des algorithmes évolutionnistes (faible coût, simplicité des opérations, performance…) et possède des qualités que nous avons mentionnées dans la discussion ci-dessus. Toutefois, montrer qu'un système de chiffrement est sûr reste une tâche très délicate (excepté le système à masque jetable). La plupart des crypto-analystes pensent même que tous les algorithmes de chiffrement sont cassables à long terme, mais l'essentiel est de concevoir son système de chiffrement de telle manière que la durée de vie souhaitée pour le texte chiffré soit inférieure à celle mise par le crypto-analyste pour casser ce système.

Caux, C., Pierreval, H., Portmann, M. (1995). Les algorithmes génétiques et leur application aux problèmes d'ordonnancement. *APII*, 29, 4-5, 409-443.
Clark, J. (2003). *Nature-Inspired Cryptography: Past, Present and Future.*
IEEE, 3, 1647-1654
Florin,G., Natkin, S. (2002*). les techniques de la cryptographie.* CNAM .
Goldberg, D. (1989). *Genetic algorithms in search optimisation & Machine Learning*. Addison-Wesley. Publishing Company, Inc.
Grenfenslette, J. (1986). Optimisation of control parameters for genetic algorithms.
IEEE translation on SMC. 16, 1, 122-128.
Khan Phang, C. (Octobre 1988). *Algorithmes heuristiques et évolutionnistes.*
Thèse de doctorat, université de Lille.
Labouret, G. (2001). *Introduction à la cryptographie*.
http://www.hsc.fr/ressources/cours/crypto/index.html.fr
Menezes, A., Van Oorschot, P., Vanstone, S. (1996). *Handbook of aplied cryptography.* CRC Press.
Muller, D. (2003). *Le chiffre de Merkle-Hellman*
http://www.apprendre -en-ligne.net/crypto/knapsack
Mühlenbein, H, Schlierkamp-Voosen, D. (1993). Predictive Models for the Breeder Genetic Algorithm-I, continuous Parameter Optimization. *Evolutionary Computation*, 1, 1, 25-49.
Shannon**, C.** (1949). *Communication* Theory of Secrecy Systems**.**
Bell Systems Technical Journal.
Stinson, D. (2002). *Cryptography - Theory and practice.*
Florida: CRC Press, Inc.
Tisserand, R (2000). *Etat de l'art sur la cryptographie*.
http://www.mines.u-nancy.fr/~tisseran/cours/cryptographie_Renaud.pdf
Zimmermann, P. (1994). *Guide de l'utilisateur de PGP Network-Associates, Inc. (USA)*