IFT-4001 (Hiver 2016) Projet d'exploration

Alexandre Cormier (111 101 150)
Alexandre Picard-Lemieux (111 103 625)
Patrick Côté (111 103 743)
Vincent Beaudoin (111 103 778)

24 avril 2016

1 Introduction

Utilisée depuis l'antiquité, la cryptographie sert à protéger des messages. Celle-ci se fait souvent à l'aide de secrets ou de clés. Cette cryptographie a principalement été utilisée à des fins militaires, commerciales ou simplement pour protéger la vie privée. L'opération de déchiffrer un message peut donc avoir des conséquences assez importantes sur la vie des gens, car ces informations peuvent mener à des actions destructives ou créatives.

L'optimisation combinatoire est une branche de l'informatique et des mathématiques appliquées. C'est la recherche d'une solution au coût minimal d'un problème dont l'espace des solutions est discret. Nous ferons une étude de la viabilité de la programmation par contraintes pour attaquer la cryptographie classique par substitution.

2 Description du problème

Le chiffrement par substitution est une façon de cacher un message en remplaçant chaque lettre par une autre. On parle de substutition monoalphabétique lorsque qu'une lettre est toujours remplacée par la même autre lettre. Autrement, on parle de substitution polyalphabétique.

Un exemple de chiffrement par substitution monoalphabétique est le chiffre de César. Pour ce chiffre, on associe à chaque lettre sa position dans l'alphabet, commençant par 0. La clé est une lettre et chaque lettre du message est décalée d'un nombre de positions correspondant à la clé. Par exemple, le message ABC chiffré avec la clé B devient BCD. Les calculs de décalage se font en modulo 26, tel que XYZ chiffré avec la même clé devient YZA.

Le chiffre de vigenère est un exemple de chiffrement par substitution polyalphabétique. La clé est composée d'une ou plusieurs lettres et le message est placé divisé en partie de la même longueur que la clé. Si les différentes parties du message sont placées l'une en-dessous de l'autre, il suffit d'appliquer le chiffre de César à chaque colonne i avec la i^{eme} lettre de la clé. Par exemple, le message ABCD chiffré avec la clé BC devient BDDF.

La substition est toujours une composante à la base de la cryptographie moderne, mais ces méthodes de chiffrement simples ne sont plus utilisées comme systèmes cryptographiques à part entière puisqu'elles sont vulnérables à différentes attaques. Il est possible, notamment, d'analyser la fréquence relative des lettres dans le message chiffré et comparer avec la fréquence relative des lettres dans la langue du message d'origine pour trouver les clés les plus probables.

Dans le cadre de ce projet, nous étudierons une méthode alternative, utilisant la programmation par contrainte, pour retrouver la clé de tels chiffrements à partir du message chiffré. Il est évidemment impossible pour un programme automatisé de trouver la clé de façon certaine, car il n'y a pas nécessairement moyen de reconnaître le message d'origine lorsque la bonne clé est trouvée.

Par contre, nous pourrons ordonner les clés les plus probables selon, notamment, la fréquence relative de chaque lettre dans le message déchiffré.

L'objectif ici est d'étudier la viabilité d'une telle approche, par contrainte, pour l'attaque du chiffrement par substitution. Nous étudierons l'efficacité de l'approche selon la longueur du message et, dans le cas du chiffre de Vigenère, de la longueur de la clé.

Par exemple, en prenant le message encrypté LXFOPVEFRNHR, le solveur doit trouver que la clé la plus probable est LEMON et que le texte original est ATTACKATDAWN. L'implémentation utilisée utilisera les fréquences des lettres trouvées d'une langue donnée. [1] Cette implémentation sera donc plus efficace si la chaine est longue et si elle utilise des vrais mots de la langue choisie.

3 Approche(s) proposée(s)

On prend n qui est la longueur de la chaine à déchiffrer, c qui est la lonueur de la clé, l qui est le nombre de lettre dans l'alphabet, et p qui est la précision de la fréquence.

On déclare une variable pour chaque lettre de la chaine cryptée. Ainsi, la variable X_i représente la valeur de la lettre de la chaine cryptée à la position i pour $1 \leq i \leq n$. Le domaine de chaque variable est l'ensemble des entiers entre 0 et $l - 1$.

On déclare une variable pour chaque complément de clé. Ainsi, la variable K_i représente la valeur de la lettre de la clé à la position i pour $1 \leq i \leq c$. Le domaine de chaque variable est l'ensemble des entiers entre 0 et $l - 1$.

On déclare une variable pour le texte intermédiaire. Ainsi, la variable I_i représente la valeur de la la lettre de la chaine à la position i pour $1 \leq i \leq n$. Le domaine de chaque variable est l'ensemble des entiers entre 0 et $2 * (l - 1)$.

On déclare une variable pour le texte brut. Ainsi, la variable B_i représente la valeur de la lettre de la chaine à la position i pour $1 \leq i \leq n$. Le domaine de chaque variable est l'ensemble des entiers entre 0 et $l - 1$.

On déclare une variable pour les comptes. Ainsi, la variable C_i représente le compte de chaque lettre dans la chaine à la position i pour $1 \leq i \leq l$. Le domaine de chaque variable est l'ensemble des entiers entre 0 et n.

On déclare une variable pour les comptes multipliés. Ainsi, la variable M_i représente le compte multiplié par la précision de la fréquence de chaque lettre dans la chaine à la position i pour $1 \leq i \leq l$.

1. Wikipedia, [En ligne]. `https://en.wikipedia.org/wiki/Letter_frequency` (Page consultée le 3 avril 2016)

Le domaine de chaque variable est l'ensemble des entiers entre 0 et $n * p$.

On déclare une variable pour les fréquences multipliées. Ainsi, la variable F_i représente la fréquence multipliée de chaque lettre dans la chaine à la position i pour $1 \leq i \leq l$. Le domaine de chaque variable est l'ensemble des entiers entre 0 et p.

On déclare une variable pour les fréquences multipliées négatives du langage. Ainsi, la variable N_i représente la fréquence multipliée négative de chaque lettre de l'alphabet à la position i pour $1 \leq i \leq l$. Le domaine de chaque variable est l'ensemble des entiers entre $-p$ et 0.

On déclare une variable pour les différences. Ainsi, la variable D_i représente la différence entre les fréquences multipliées et les fréquences multipliés du language à la position i pour $1 \leq i \leq l$. Le domaine de chaque variable est l'ensemble des entiers entre $-p$ et p.

On déclare une variable pour les différences absolues. Ainsi, la variable A_i représente valeur absolue de la différence entre les fréquences multipliées et les fréquences multipliés du language à la position i pour $1 \leq i \leq l$. Le domaine de chaque variable est l'ensemble des entiers entre 0 et p.

On déclare une variable S pour la somme des différence des fréquences. Le domaine de la variable est l'ensemble des entiers entre 0 et $l * p$.

Pour chaque lettre i de la chaine, nous avons ces contraintes.

$$X_i + K_{i\%c} = I_i \tag{1}$$

$$I_i \% l = B_i \tag{2}$$

Pour chaque lettre i de l'alphabet, nous avons ces contraintes.

$$Compte(i, B, C_i) \tag{3}$$

$$C_i * p = M_i \tag{4}$$

$$M_i / n = F_i \tag{5}$$

$$F_i + N_i = D_i \tag{6}$$

$$A_i = |D_i| \tag{7}$$

Pour finir, nous avons cette contrainte.

$$A_1 + A_2 + ... + A_l = S \tag{8}$$

La fonction objective est de minimiser S. L'heuristique par défaut a été utilisé.

Nous avons donc $n + c + n + n + l + l + l + l + l + l + 1 \in \Theta(n + c + l)$ variables.

Nous avons donc

$$\begin{aligned} &nl + cl + n(2l - 1) + nl + l(n + 1) + l(np + 1) + l(p + 1) + l(p + 1) + l(2p + 1) + l(p + 1) + (lp + 1) \\ &= nl + cl + 2nl - n + nl + ln + l + nlp + l + lp + l + lp + l + 2lp + l + lp + l + lp + 1 \\ &= 5nl + cl - n + 6l + nlp + 6lp + 1 \\ &\in \Theta(nlp + cl) \end{aligned}$$

valeurs.

Nous avons n contraintes de type (1) et (2). Nous avons l de type (3), (4), (5), (6) et (7). Nous avons une contrainte de type (8). Nous avons donc au total $2n + 5l + 1 \in \Theta(n + l)$.

Ceci est donc notre modèle finale. Lors de nos premières itérations, nous avions mis que la fréquence des lettres était restée dans des variables réelles. Par contre, grâce à la variable de précision, nous convertissons les fréquences, qui sont réelles, en entiers.

4 Protocole d'expérimentation

Nous avons choisi de faire des tests sur des instances de différentes longueurs de message et de clé, des messages qui respectent bien la distribution de lettres anglaises, d'autre qui ne la respectent pas. Les métriques que nous allons observer durant les expérimentations sont le temps de réponse, le nombre de retours arrière, ainsi que le nombre de solutions trouver.

La première instance est une instance qui est considérée comme étant un message long ayant une clé de longueur 4 et qui devrait avoir une distribution correcte de la fréquence des lettres.
Voici le texte brut : FRIEN DSROM ANSCO UNTRY MENLE NDMEY OUREA RSICO METOB URYCA ESARN OTTOP RAISE HIMTH EEVIL THATM ENDOL IVESA FTERT HEMTH EGOOD ISOFT INTER REDWI THTHE IRBON ESSOL ETITB EWITH CAESA RTHEN OBLEB RUTUS HATHT OLDYO UCAES ARWAS AMBIT IOUSI FITWE RESOI TWASA GRIEV OUSFA ULTAN DGRIE VOUSL YHATH CAESA RANSW ERDIT HEREU NDERL EAVEO FBRUT USAND THERE STFOR BRUTU SISAN HONOU RABLE MANSO ARETH EYALL ALLHO NOURA BLEME NCOME ITOSP EAKIN CAESA RSFUN ERAL

Voici le texte encrypté : YKBXG WLKHF TGLVH NGMKR FXGEX GWFXR HNKXT KLBVH FXMHU NKRVT XLTKG HMMHI KTBLX ABFMA XXOBE MATMF XGWHE BOXLT YMXKM AXFMA XZHHW BLHYM BGMXK KXWPB MAMAX BKUHG XLLHE XMBMU XPBMA VTXLT KMAXG HUEXU KNMNL ATMAM HEWRH NVTXL TKPTL TFUBM BHNLB YBMPX KXLHB MPTLT ZKBXO HNLYT NEMTG WZKBX OHNLE RATMA VTXLT KTGLP XKWBM AXKXN GWXKE XTOXH YUKNM NLTGW MAXKX LMYHK UKNMN LBLTG AHGHN KTUEX FTGLH TKXMA XRTEE TEEAH GHNKT UEXFX GVHFX BMHLI XTDBG VTXLT KLYNG XKTE

La deuxième instance est une instance qui est considérée comme étant un message d'une courte longueur ayant une clé de longueur 1 et qui a une bonne distribution de la fréquence des lettres. De plus, cette instance va être réutilisée pour faire des tests avec des clés de longueur 2,3,4,5.
Voici le texte brut : GENIUS WITHOUT EDUCATION IS LIKE SILVER IN THE MINE

La troisième instance est une instance qui est considérée comme un long message ayant une très bonne distribution de la fréquence des lettres que nous avons testées sur une clé de 1 et de deux.
Voici le texte brut : HEREUPON LEGRAND AROSE WITH A GRAVE AND STATELY AIR AND BROUGHT ME THE BEETLE FROM A GLASS CASE IN WHICH IT WAS ENCLOSED IT WAS A BEAUTIFUL SCARABAEUS AND AT THAT TIME UNKNOWN TO NATURALISTS OF COURSE A GREAT PRIZE IN A SCIENTIFIC POINT OF VIEW THERE WERE TWO ROUND BLACK SPOTS NEAR ONE EXTREMITY OF THE BACK AND A LONG ONE NEAR THE OTHER THE SCALES WERE EXCEEDINGLY HARD AND GLOSSY WITH ALL THE APPEARANCE OF BURNISHED GOLD THE WEIGHT OF THE INSECT WAS VERY REMARKABLE AND TAKING ALL THINGS INTO CONSIDERATION I COULD HARDLY BLAME JUPITER FOR HIS OPINION RESPECTING IT

La quatrième instance est une instance ayant une longueur courte et qui ne respect pas très bien la distribution des lettres. Nous l'avons testé sur une clé de longueur 1.
Voici le texte brut : ZJXQKB

5 Résultats

Voici les résultats sans l'heuristique que nous avons créée.

Instances	**Résultats**				
	Longueur clé	**Retour arrière**	**Temps de réponse**	**Nb solutions**	**Présent**
1	4	559 281	87,187 secondes	9061	Oui
2	1	55	0,007 secondes	19	Oui
2	2	1369	0,306 secondes	172	Oui
2	3	68 345	1,944 secondes	1 576	Oui
2	4	6 607 263	114,837 secondes	11 646	Oui
2	5	1 611 566 903	6h49m47s	292	Oui
3	1	51	0,047 secondes	15	Oui
3	2	1 345	0,535 secondes	155	Oui
4	1	63	0,198 secondes	17	Non

Voici les résultats avec l'heuristique que nous avons créée. * Les résultats sont ceux sans l'heuristique.

Instances	Résultats				
	Longueur clé	**Retour arrière**	**Temps de réponse**	**Nb solutions**	**Présent**
1	4	219 421	93,992 secondes	158	Oui
2	1	55	0,373 secondes	21	Oui
2	2	1411	0,306 secondes	172	Oui
2	3	68 345	1,944 secondes	1 576	Oui
2	4	6 607 263	114,837 secondes	11 646	Oui
2	5	1 611 566 903	6h49m47s	292	Oui
3	1	51	1,018 secondes	18	Oui
3	2	1323	4,099 secondes	65	Oui
4	1	63	0,198 secondes	17	Non

6 Discussion

7 Conclusion