Petite introduction à la cryptologie

MAÎTRISE DE MATHÉMATIQUES

J.-P. Zanotti

Université de Toulon et du Var/Faculté des Sciences et Techniques

CHAPITRE 1

Cryptographie à clef secrète

1. Principes fondamentaux de la cryptogaphie

La *cryptologie*, ou science du secret, est devenue depuis une trentaine d'années une discipline scientifique à part entière qui connaît un engouement spectaculaire à la mesure du rôle de plus en plus important que jouent les réseaux de communication au sein de notre société moderne. La *cryptographie* concerne les techniques d'écriture de messages secrets tandis que la *cryptanalyse* en fait l'analyse pour les valider ou les invalider.

La cryptographie et la cryptanalyse s'appuient sur de nombreuses théories mathématiques et informatique dont les plus importantes sont

- la théorie de l'information
- la combinatoire
- l'algorithmique
- la théorie de la complexité et de l'approximation
- l'arithmétique et la théorie des nombres
- la théorie des probabilités
- l'algèbre discrète

Il est difficile de parler de cryptographie sans parler de codage et pour le non-spécialiste les deux termes sont souvent synonymes. L'ambiguïté est évidemment entretenue par l'importance de leur héritage scientifique commun. Les objectifs respectifs des deux sont pourtant à première vue radicalement opposés : le codage s'emploie à rajouter des informations à un message pour le rendre plus intelligible alors que la cryptographie tente par tous les moyens de le rendre incompréhensible ! En réalité les deux théories se complètent, car le message chiffré doit être intelligible pour pouvoir être déchiffré. Il est en effet de peu d'intérêt d'employer le javanais au téléphone pour ne pas dévoiler le contenu d'une conversation si votre interlocuteur (qui comprend le javanais) n'entend que le bruit d'une cafetière électrique...

Les problèmes qui relèvent de la cryptographie peuvent être regroupés en trois thèmes majeurs :

(1) La *confidentialité*, qui est toujours l'objectif principal et qui amène à développer et à étudier les techniques permettant de communiquer des informations confidentielles entre plusieurs entités. Deux grandes classes de protocoles sont employés : les protocoles à *clef privée* ou *secrète* et les protocoles à *clef publique*.

(2) L'*intégrité* ou *authentification*, ou comment s'assurer qu'un document n'a pas été altéré par une tierce personne.

(3) L'*identification* qui consiste à certifier l'origine d'un message.

On doit également citer le problème tout aussi important de la *signature* des documents mais il s'agit de la conjonction des deux derniers :

$$\text{signature} \equiv \text{intégrité et identification.}$$

Plus récemment, des problèmes de *partage de secret* ont vu le jour. Pour donner une idée de cette "nouvelle" problématique, imaginons que dans une hypothétique défense européenne, les m chefs d'états disposent d'une clef leur permettant de déclencher une attaque nucléaire. Tout le monde s'accordera à dire qu'il est inconcevable qu'une seule clef puisse autoriser l'accès à cette ressource. D'un autre coté, il peut être hasardeux de conditionner l'accès à une telle ressource uniquement si les m clefs sont activées ! On peut donc souhaiter que l clefs suffisent, avec bien sûr $1 < l < m$. C'est de ce type de questions que traite le partage de secret.
Le volume restreint d'heures de ce cours ne nous permettra pas d'étudier ni même d'aborder toutes les problématiques de la cryptographie aussi passionnantes soient-elles, nous allons donc nous concentrer sur la confidentialité qui est historiquement le premier problème sur lequel s'est agrégé la cryptographie et qui reste encore de nos jours le plus important. Une première partie est consacrée à l'étude de quelques protocoles à clef secrète simples et c'est la cryptanalyse du système de Vigenère qui clot ce chapitre. Le reste du cours est principalement dédié à l'étude des systèmes de chiffrement à clef publique à travers RSA. La cryptographie à clef publique est indissociable de la compréhension de la théorie de la complexité qui met en lumière toute le problématique du temps de calcul, il nous a donc semblé indispensable d'en présenter les principales notions.

Le cours est constitués en cinq parties de tailles sensiblement égales :

(1) Cryptographie à clef secrète : les méthodes naïves, notion de clef.

(2) Le système de Vigenère et sa cryptanalyse.

(3) Cryptographie à clef publique : le système RSA.

(4) Algorithmes et complexité. Quelques éléments de la théorie de la complexité : problèmes P, NP et NP-complets. Le théorème de Cook.

(5) Analyse de RSA et problèmes relatifs à la primalité/décomposition.

2. Cryptographie à clef secrète

Comme nous l'avons déjà dit quelques lignes plus haut, l'un des principaux objectifs de la cryptographie est de permettre à deux entités A et B, plus connues des cryptographes sous leurs noms d'emprunt *Alice* et *Bob*, de communiquer des informations *secrètes*, autrement dit, des informations qui ne doivent être intelligibles que par eux.
La cryptographie ne consiste pas à protéger *physiquement* les informations qu'Alice souhaite transmettre à Bob et qui constituent ce que l'on appelle le *message clair*, mais à les transformer de manière à ce que seul Bob soit capable d'en retrouver le contenu, du moins en principe... Ceci ne signifie pas qu'aucune contre-mesure physique n'est envisagée pour protéger une transmission (on cache bien le contenu d'une lettre dans une enveloppe), mais que la cryptographie se concentre sur les transformations adéquates à appliquer au message pour assurer le secret et ceci peut bien entendu intégrer des contraintes bien

physiques (brouillage du spectre des consommations électriques des composants, chronogrammes, etc...). Une hypothèse de base de la cryptographie est donc que le vecteur de transmission appelé *canal*, téléphone, internet, cable, papier ou autres, peut *toujours* être piraté par un intrus O qui s'appelle généralement *Oscar*. On voit bien avec l'exemple de l'enveloppe que le secret n'est pas bien gardé, il suffit à Oscar de l'ouvrir pour prendre connaissance du contenu du message.

Les premières méthodes de chiffrement, dont certaines sont vieilles de plus de 4000 ans, sont toutes basées sur le principe dit d'une *clef privée*, à savoir que le chiffrement et le déchiffrement du message dépendent d'une quantité k supposée secrète au sens où elle n'est connue que de l'expéditeur Alice et du destinataire Bob. Bien entendu, cela sous-entend qu'il faut trouver un autre canal que celui que l'on veut protéger pour transmettre cette clef, mais ceci est une autre histoire que nous aborderons plus loin quand nous étudierons la cryptographie à clef publique. Dans la cryptographie à clef secrète, on suppose donc toujours que la clef k transite de A vers B par un canal considéré comme sûr. Le lecteur perspicace rétorquera que si ce canal sûr existe, pourquoi ne pas l'utiliser pour transmettre un message clair ? Essentiellement pour deux raisons, la première est que la clef peut-être de taille réduite comparée au message à protéger, l'autre est qu'elle peut être utilisée plusieurs fois, ceci tant qu'elle est considérée comme sûre. Le schéma ci-dessous résume le protocole de chiffrement à clef secrète :

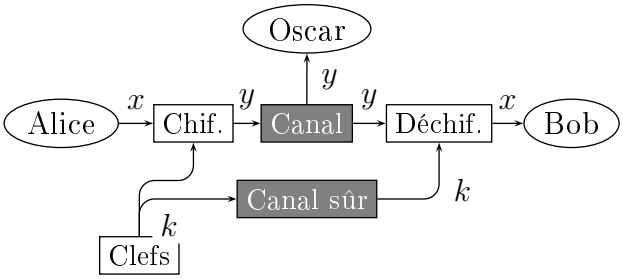

FIG. 1 – Protocole de chiffrement à clef secrète.

On peut à présent donner une vision un peu plus formelle d'un système de chiffrement à clef secrète :

DÉFINITION 1. *Un* système cryptographique *ou* protocole de chiffrement à clef secrète *est un quintuplet* (P, C, K, E, D) *où :*

- P *est l'espace (fini) des messages* clairs *;*
- C *est l'espace (fini) des messages* chiffrés *;*
- K *est l'espace (fini) des* clefs *;*

D'autre part, E *est une partie de* C^P *et* D *une partie de* P^C *telles que pour toute clef* $k \in K$*, il existe une unique* fonction de chiffrement e_k *de* E *et une unique* fonction de déchiffrement d_k *de* D *telles que*

$$\forall x \in P, \quad d_k(e_k(x)) = x. \tag{1}$$

La signification de l'équation (1) tombe évidemment sous le sens si l'on veut pouvoir retrouver le message initial envoyé par Alice, mais mathématiquement parlant, que peut-on

en tirer comme informations sur les fonctions de chiffrement et de déchiffrement ? Trivialement les fonctions e_k et d_k doivent être injectives sans quoi le déchiffrement d'un texte chiffré y pourrait provenir de plusieurs clairs et dans l'autre cas, deux chiffrés distincts pourraient être déchiffrés en un même message clair.
En général les ensembles P et C sont constitués par des mots de longueur l fixée construits sur un alphabet fini A. On dit dans ce cas que le système de chiffrement est un système de chiffrement par *blocs* car les textes clairs sont tous de la même longueur l. Comme les protocoles de chiffrement et de déchiffrement sont quasi exclusivement utilisés à travers des systèmes informatiques, l'alphabet binaire $A = \{0, 1\}$ est évidemment majoritaire.

3. Chiffrement par décalage circulaire, ou chiffrement de César

Le principe est très simple et consiste à se fixer un entier k compris entre 0 et 25 et à remplacer chaque lettre d'un message par la k-ème lettre suivante dans l'ordre alphabétique en revenant au début de l'alphabet dès que l'on a atteint la lettre z (on évite généralement la valeur $k = 0$). Par exemple pour $k = 3$, le message `ALLO` est chiffré en `DOOR`. Pour qui est un peu familier avec l'arithmétique modulaire, tout ceci se formalise avantageusement de la manière suivante : les trois ensembles P, C et K sont tous égaux à $\mathbf{Z}/26\mathbf{Z}$. Ainsi le message ou texte clair au sens de la définition 1 est réduit à une unique lettre. Tout le monde l'aura compris, et il s'agit là d'un principe commun aux systèmes de chiffrement par blocs, pour chiffrer un "message" au sens commun du terme, il suffit de le morceler et de répéter le processus de chiffrement sur l'ensemble des blocs obtenus, ici les lettres du texte.
La fonction de chiffrement $e_k : \mathbf{Z}/26\mathbf{Z} \to \mathbf{Z}/26\mathbf{Z}$ est donc définie tout simplement par

$$e_k(x) = x + k$$

et la fonction de déchiffrement par

$$d_k(y) = y - k.$$

Il s'agit donc d'une simple addition dans $\mathbf{Z}$, éventuellement suivie d'une réduction modulaire.
Un tel système est-il sûr, comme le pensait l'empereur Jules César qui en était friand ? Bien entendu non, et la raison est élémentaire, l'espace des clefs K est beaucoup trop petit pour espérer conserver le secret très longtemps. Il est clair que si la clef k est choisie de manière aléatoire, entre 1 et 25 (le cas 0 est sans grand intérêt), en moyenne il faudra 12.5 tentatives pour trouver la bonne clef, ce qui même à la main est tout à fait envisageable. On en déduit donc une première condition pour qu'un système de chiffrement à clef secrète soit sûr :

(2) l'espace des clefs K doit être grand.

Malheureusement, cette condition, si elle est nécessaire, est loin d'être suffisante comme nous pourrons le constater avec d'autres systèmes cryptographiques. Présisons néanmoins ce que l'on entend par "grand". Aujourd'hui une machine grand public, à moins de 10KF, permet d'effectuer environ 10^9 opérations élémentaires sur 32 bits, soit environ 2^{30} opérations. Comme il est relativement simple de mettre ce type de machines en réseau, on

augmente aisément de 10 la puissance de 2 avec 1024 machines et on peut encore l'augmenter de 19 en calculant durant une semaine complète (une durée d'une semaine correspond à environ 2^{19} secondes), on arrive donc approximativement à 2^{60} opérations de base. Il est donc admis aujourd'hui que la longueur d'une clef binaire doit être d'au moins 128 bits, ce qui s'interprète de la manière suivante : le cardinal de l'espace des clefs K doit être supérieur à 2^{128}. Si cette contrainte de taille est à moduler en fonction du système de chiffrement, il faut avant tout l'interpréter comme une borne inférieure qui permet d'éviter l'attaque empirique du système, à savoir l'essai de toutes les clefs possibles. Notons que la très grande majorité des attaques de systèmes cryptographiques consistent en substance à éliminer à chaque tentative de déchiffrement, une grande partie des clefs en sus de celle qui a échoué, ceci en exploitant les failles conceptuelles du système. Ainsi, si un système cryptographique a un espace des clefs sur 128 bits (soit 2^{128} clefs), mais qu'un pirate est capable d'éliminer 2^{80} clefs à chaque tentative de déchiffrement, il ne faut finalement que 2^{48} tentatives pour "casser" le système, ce qui tout à fait envisageable.

S'il est certain que la connaissance du fonctionnement d'un système cryptographique est un atout non négligeable pour l'attaquer, il est inconcevable aujourd'hui de baser la sécurité d'un système sur l'absence d'informations sur son fonctionnement ; d'une part parce qu'il est fort difficile de garder ce type d'informations secrètes et d'autre part, parce que l'expérience montre qu'un attaquant déterminé pourra toujours en comprendre les mécanismes à terme. C'est la leçon que viennent d'apprendre les banques à leur dépend au sujet de la carte bancaire. Pour conclure ce paragraphe, citons donc le *principe de Kerckhoff* :

(3) Oscar connaît le système cryptographique.

4. Chiffrement par permutation ou substitution

C'est un système un peu plus complexe que le précédent puisque l'on substitue chaque lettre de l'alphabet par une autre lettre. Pour satisfaire la condition (1), il est évident que l'on ne peut pas substituer deux lettres distinctes par une même lettre, autrement dit chaque lettre de l'alphabet doit être remplacée par une lettre différente, on réalise donc une bijection entre les lettres et leurs substitution. Exemple :

```
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
   |   |                       | | |
A Z E R T Y U I O P Q S D F G H J K L M W X C V B N+
```

Ainsi le message `SECRET` est chiffré en `LTEKTM`. Formalisons un peu tout ceci : on rappelle que si E est un ensemble fini, on note $S(E)$ l'ensemble des bijections de l'ensemble E sur lui-même et ces bijections sont appelées des *permutations*. Si l'on munit cet ensemble de la loi de composition des applications, on obtient un groupe, c'est le *groupe des permutations de E*. Notons qu'il n'est pas commutatif. Ce groupe est isomorphe au groupe $\mathfrak{S}_n$ des permutations de l'intervalle $[1, n]$ de $\mathbf{N}$. En effet, si E est fini, cela signifie par définition qu'il existe un entier $n \geq 0$ appelé *cardinal* de E et une bijection φ de E dans $[1, n]$. Donc si $s \in S(E)$, on lui associe la permutation $\sigma \in \mathfrak{S}_n$ définie par

$$\sigma = \varphi \circ s \circ \varphi^{-1}.$$

Les espaces des clairs et des chiffrés sont tout deux égaux à l'alphabet $A = \{a, \ldots, z\}$ et l'espace des clefs est $S(A)$. On notera au passage qu'il n'est pas indispensable d'identifier A à $\mathbf{Z}_{26}$ puisque l'on n'exploite pas les propriétés algébriques de cet anneau contrairement au chiffrement de César. La fonction de chiffrement est donc associée à une permutation $s \in S(A)$:

$$e_s(x) = s(x),$$

et la fonction de déchiffrement est obtenue à partir de la permutation inverse s^{-1} qui existe et est unique puisque $S(A)$ est un groupe.

$$d_s(y) = s^{-1}(y).$$

On peut noter que la permutation inverse s^{-1} est très facile à calculer à partir de la permutation s.
Cette fois l'espace des clefs est bien plus important puisque le cardinal de $S(A)$ est égal à celui de $\mathfrak{S}_n$ qui vaut $n!$. Dans le cas qui nous intéresse, $n = 26$ et la formule de Stirling nous donne une estimation qui évite les n produits.

$$n! = \sqrt{2\pi n}\, n^n e^{-n}\big(1 + \Theta(\frac{1}{n})\big).$$

Ceci nous donne pour $n = 26$ un nombre très proche de 4×10^{26} qui est compris entre 2^{88} et 2^{89}. Donc du point de vue de la condition que nous avons mise en évidence dans le chiffrement de César, même si l'on atteint pas 2^{128}, on peut considérer que l'espace des clefs est grand, en tout cas suffisamment pour éviter toute recherche exhaustive de la clef ! Ce système est-il pour autant beaucoup plus sûr que le précédent ? En réalité, malgré l'apparente complexité du chiffrement, il est relativement simple de retrouver la clef, ou au moins une clef partielle. Comment procéder ? On peut mesurer très facilement la distribution statistique des lettres ou des groupes de lettres dans une langue en collectant et en analysant un grand nombre d'articles ou de textes. Pour le français, on obtient les pourcentages suivants à partir de 40 grands classiques de la littérature :

Ces 26 lettres ont été classées en 6 groupes dans l'ordre décroissant des probabilités :

(1) `E` : proba. $\simeq 0.16$;

(2) `A I S T N R U` : $0.065 \leq$ proba. ≤ 0.095 ;

(3) `L O` : proba. $\simeq 0.05$;

(4) `D M P C V` : $0.021 \leq$ proba. ≤ 0.034 ;

(5) `Q G B F J H` : proba. $\simeq 0.01$.

(6) `Z Y X W K` : proba. < 0.004.

On peut pousser l'analyse plus loin en étudiant les probabilités d'apparition des digrammes (voire des trigrammes, i.e. respectivement groupes de 2 et 3 lettres). On obtient le classement suivant pour les 30 digrammes les plus fréquents, toujours dans l'ordre décroissant :

```
ES EN DE LE NT RE ON AI IT ER AN TE QU EL SE
OU LA IS NE RA IL UR TI NS TA IN ET ME CE ED
```

x	$P(x)$	x	$P(x)$
A	9.42	N	7.15
B	1.02	O	5.14
C	2.64	P	2.86
D	3.39	Q	1.06
E	15.87	R	6.46
F	0.95	S	7.90
G	1.04	T	7.26
H	0.77	U	6.24
I	8.41	V	2.15
J	0.89	W	0.00
K	0.00	X	0.30
L	5.34	Y	0.24
M	3.24	Z	0.32

d	$P(d)$	D	$P(d)$
ES	2.3809	OU	1.1782
EN	2.1248	LA	1.1737
DE	1.9570	IS	1.1314
LE	1.8835	NE	1.0668
NT	1.7009	RA	1.0201
RE	1.6622	IL	1.0178
ON	1.6207	UR	0.9688
AI	1.6051	TI	0.9154
IT	1.4596	NS	0.9132
ER	1.4470	TA	0.8790
AN	1.3957	IN	0.8723
TE	1.2940	ET	0.8634
QU	1.2569	ME	0.8619
EL	1.2220	CE	0.8456
SE	1.1864	ED	0.8449

TAB. 2 – Distributions des lettres de l'alphabet en français et distribution des 30 digrammes les plus fréquents exprimé en %.

Exercice 1 [2/5] Déchiffrez le message suivant en retrouvant la clef secrète. Pour cela utilisez les fréquences de distribution des lettres isolées, des digrammes :

```
YIFQFMZRWQFYVECFMDZPCVMRZWNMDZVEJBTXCDDUMJ
NDIFEFMDZCDMQZKCEYFCJMYRNCWJCSZREXCHZUNMXZ
NZUCDRJXYYSMRTMEYIFZWDYVZVYFZUMRZCRWNZDZJJ
XZWGCHSMRNMDHNCMFQCHZJMXJZWIEJYUCFWDJNZDIR
```

5. Le chiffrement de Vigenère

Les chiffrements par décalage ou par permutation sont des chiffrements dits *monoalphabétiques* car un symbole particulier x est toujours chiffré par le même symbole y et cette caractéristique consitue le talon d'Achille de cette méthode. Le chiffrement de Vigenère tente de remédier à ce problème en autorisant des chiffrés différents pour un même symbole. Le principe est le même que pour le chiffrement par décalage, mais au lieu de tronçonner le message en suite de symboles à chiffrer, on tronçonne le message en mots $x = x_1x_2 \dots x_m$ de longueur m dont chaque symbole est chiffré avec un décalage spécifique. Une clef k est donc un m-uplet $(k_1, \dots, k_m)$ de $\mathbf{Z}/26\mathbf{Z}^m$ et la fonction de chiffrement est donnée par

$$e_k(x) = (x_1 + k_1, \dots, x_m + k_m),$$

les additions se faisant dans $\mathbf{Z}/26\mathbf{Z}$ naturellement. Le déchiffrement s'obtient en remplaçant les $+$ par des $-$, ou en chiffrant y avec la clef $-k$. Dans ce schéma, on préfère habituellement représenter la clef comme un mot ou une phrase [1], appelée *mot/phrase de passe* en reprenant l'ordre naturel des lettres de l'alphabet : $A \equiv 0$, $B \equiv 1$, etc... Par exemple, avec la phrase "Elvis est vivant", correspondant aux décalages

1. privée des séparations entre mots

```
E  L  V  I  S  E  S  T  V  I  V  A  N  T
4  11 21 8  18 4  18 19 21 8  21 0  13 19
```

on chiffre le texte “Comment chiffrer avec Vigenère” en enlevant au préalable tous les espaces et les accents :

```
     COMMENTCHIFFRE RAVECVIGENERE
+    ELVISESTVIVANT ELVISESTVIVANT
=    GZHUWRLVCQAFEX VLQMUZAZZVZRR
```

L'espace n'est là que pour matérialiser la segmentation du message en blocs. Cette méthode est une réponse au problème du chiffrement monoalphabétique et la taille de l'espace des clefs est trivialement 26^m, donc suffisamment grande pour peu que la phrase clef soit de taille au moins 20.

Nous allons voir que malgré les apparences, ce système de chiffrement n'est pas plus sûr que les autres (il a tout de même fallu quelques dizaines d'années pour en venir à bout...) La cryptanalyse se fait en trois étapes : la première consiste à retrouver la longueur de la phrase clef m ; la seconde à déterminer suffisamment de décalages relatifs entre les k_i (i.e. des valeurs $k_j - k_i$) pour que tous les k_i puissent s'exprimer relativement à un k_l particulier ; et pour conclure tester les 26 valeurs possibles pour k_l (qui fixent alors les $m-1$ autres valeurs).

(1) Détermination de m, la longueur de la clef. Le test de *Kasiski* présenté en 1863 consiste à repérer dans le texte chiffré des motifs identiques de longueur au moins trois et de noter leurs positions (i.e. la position de la première lettre). L'apparition d'un même motif à plusieurs endroits du message chiffré s'explique de deux façons : c'est une coïncidence ou alors les motifs en clair correspondants sont distants (en nombre de positions) d'un multiple de m. Si le chiffré est assez long et que l'on est capable de déterminer plusieurs motifs identiques, il est raisonnable de conjecturer que la valeur de m est égale au pgcd de tous les décalages entre motifs identiques.
On peut valider ou invalider l'hypothèse sur la longueur m de la clef en calculant l'*indice d'autocoïncidence* d'une chaîne de caractères :

Définition 2. *Soit $x = x_1 \dots x_n$ une séquence sur un alphabet fini $A = \{a_0, \dots, a_{l-1}\}$ de cardinal l dont on connaît la distribution en fréquence $f_1, f_2, \dots, f_l$ (i.e. la lettre a_i apparaît f_i fois dans x). On appelle indice d'*autocoïncidence $\omega(x)$ *la probabilité que deux symboles pris au hasard dans x soient égaux.*

Comment calculer cette quantité ? Attention, il ne s'agit pas de calculer la probabilité que la chaîne contiennent deux fois le même caractère, sans quoi sauf en ayant des caractères tous différents, la probabilité serait 1. Cette probabilité est la somme des probabilités que les deux caractères soient égaux à l'un des n caractères de A sur tous les caractères de A : la probabilité que le premier caractère soit égal à a_i est f_i/l, et la probabilité que le second caractère soit égal à a_i est $(f_i - 1)/(l-1)$, d'où

$$\omega(x) = \frac{1}{l(l-1)} \sum_{i=0}^{l-1} f_i(f_i - 1). \tag{4}$$

Si A désigne l'alphabet d'une langue, le français par exemple, et p_i la probabilité d'apparition de la lettre a_i, on peut définir l'indice d'autocoïncidence de la langue A par

$$\omega_A := \sum_{i=0}^{l-1} p_i^2. \tag{5}$$

Un simple calcul à partir de la table 2 nous donne pour le français ($l = 26$) :

$$\omega_A \approx 0.076. \tag{6}$$

Ainsi, avec un texte $x = x_1 \dots x_n$ assez long, il est raisonnable de penser que $\omega(x) \approx \omega_A$. Supposons à présent que l'on utilise un chiffrement par permutation, i.e. pour $\sigma \in \mathfrak{S}_{|A|}$, on analyse $x_{\sigma(1)} \dots x_{\sigma(k)}$, alors évidemment

$$\sum_{i=0}^{l-1} p_{\sigma(i)}^2 = \sum_{i=0}^{l-1} p_i^2 = \omega_A. \tag{7}$$

Ceci reste évidemment vrai pour le chiffrement cyclique de César qui est un cas très particulier du chiffrement par permutations. (L'ensemble des clefs de César est un sous-ensemble de 26 permutations parmi les 26! possibles).
Dans le cas d'un texte totalement aléatoire, l'indice d'autocoïncidence $\omega_{\text{aleat.}}$ se calcule aisément (la probabilité d'apparition des l lettres est égale à $1/l$) :

$$\omega_{\text{aleat.}} = l(\frac{1}{l})^2 = \frac{1}{l} \approx 0.038 \quad \text{pour } l = 26. \tag{8}$$

En quoi ces dernières remarques nous permettent-elle d'avancer dans notre analyse du chiffrement de Vigenère ? Principalement pour la détermination de la longueur m des blocs. Notons $y = y_1 \dots y_n$ le chiffré de x. Supposons que l'on range les lettres y_i de y dans une matrice de m' lignes en remplissant la matrice en colonnes plutôt qu'en lignes, i.e. la première colonne à gauche contient du haut vers le bas $y_1 \dots y_{m'}$, la seconde $y_{m+1} \dots y_{2m'}$ etc. Eventuellement la dernière colonne peut-être incomplète si $m' \not| k$. De deux choses l'une, soit $m' = m$ et dans ce cas, toutes les lettres de la $i$ème ligne l_i de la matrice ont été chiffrées par césar avec la même clef k_i et dans ce cas l'autocoïncidence $\omega(l_i)$ de la $i$ème ligne doit être proche de 0.076 ; dans l'autre cas, les lettres ont été chiffrées avec des clefs différentes ce qui va nécessairement perturber le spectre des distributions des lettres et rapprocher l'indice $\omega(l_i)$ de l'autocoïncidence d'un texte aléatoire. Ci-dessous la clef est `KEY` sur le texte `ABCDEFGHIKLM` (les espaces ne sont là que pour mettre les blocs en valeur) :

```
  ABC DEF GHI JKL M
+ KEY KEY KEY KEY K
= KFA NID QLG TOJ W
```

En rangeant les chiffrés en colonnes :

```
KNQTW  =  ADGJM + K
FILO   =  BEHK  + E
```

```
ADGJ   =  CFIL  + Y
```

On suppose à présent que l'on connaît la valeur de m. On peut se concentrer sur le calcul du mot de passe k.

DÉFINITION 3. *Soit $x = x_1 \dots x_n$ et $x' = x'_1 \dots x'_{n'}$ deux séquences de longueurs respectives n et n' sur un alphabet fini $A = \{a_0, \dots, a_{l-1}\}$ de cardinal l dont on connaît les distributions en fréquence $f_1, f_2, \dots, f_l$ et $f'_1, f'_2, \dots, f'_l$ (i.e. la lettre a_i apparaît f_i fois dans x et f'_i fois dans x'). On appelle indice d'*intercoïncidence *$\omega(x, x')$ la probabilité qu'un symbole de x soit égal à un symbole de x' tout deux pris au hasard.*

Le même raisonnement que pour l'indice d'autocoïncidence nous donne :

$$\omega(x, x') = \frac{1}{nn'} \sum_{i=0}^{l-1} f_i f'_i. \tag{9}$$

Pour une langue donnée, la notion d'intercoïncidence coïncide (!) avec la notion d'autocoïncidence. Sans chiffrement, deux lignes l_i et l_j de la matrice doivent avoir une intercoïncidence proche de 0.076, mais une fois chiffrées que se passe-t-il ?
Supposons que l'on note y et y' les chiffrés du même texte x de longueur n avec deux clefs différentes k et k'. La probabilité qu'une lettre y_i de y soit égale à un symbole a_i est bien sûr égale à la probabilité que le symbole x_i (avant chiffrement en y_i) soit égal à a_{i-k} puisque $y_i = x_i + k \pmod{l}$ (l'indice est calculé modulo l). Par exemple, pour une clef partielle $k_j = 3$, la probabilité qu'un symbole du chiffré soit égal à la lettre `L` (11 dans $\mathbf{Z}/26\mathbf{Z}$) est égal à la probabilité que le symbole clair soit la lettre `I` ($11 - 3 = 8$). Autrement dit,

$$\begin{aligned} \omega(l_i, l_j) &= \sum_{h=0}^{l-1} p_{h-k_i} p_{h-k_j} \\ &= \sum_{h=0}^{l-1} p_h p_{h+k_i-k_j}. \end{aligned}$$

Dans ces égalités ci-dessus, le calcul des indices se fait naturellement modulo l (26 pour l'alphabet usuel). Cette valeur ne dépend, comme on s'y attend, que du décalage relatif entre les lignes l_i et l_j, i.e. de $k_i - k_j$. Si l'on calcule les indices d'intercoïncidence avec des décalages compris entre 0 et 13 pour la langue française, on obtient le tableau 3.
Dans cette table, on remarque que l'indice d'intercoïncidence oscille entre 0.035 et 0.049 pour tout décalage relatif non-nul, alors qu'il atteint 0.076 avec un décalage relatif nul. Le reste de la cryptanalyse coule maintenant de source, on dispose de m lignes, on peut donc les comparer deux-à-deux en décalant circulairement la seconde pour toutes les valeurs comprises entre 0 et 13. La bon décalage relatif fera apparaître une valeur d'intercoïncidence proche de 0.076, alors que les 12 autres donneront des valeurs significativement plus faibles.
Comme on dispose de m lignes, cela donne $13m(m-1)/2$ comparaisons possibles. On choisira bien entendu les comparaisons les plus pertinentes en terme d'intercoïncidence pour retrouver les $m-1$ décalages relatifs. Il manque une dernière relation pour retrouver

déc.	inter. ω	déc.	inter. ω
0	**0.076**	7	0.043
1	0.035	8	0.049
2	0.038	9	0.029
3	0.041	10	0.035
4	0.033	11	0.037
5	0.046	12	0.031
6	0.027	13	0.039

TABLE. 3 – Intercoïncidence entre une séquence et ses décalés.

les m inconnues, il suffit de tester les 26 clefs partielles possibles pour k_1, et pour chaque hypothèse faite sur k_1, déduire les $m-1$ autres clefs partielles k_2 à k_m.

Travaux pratiques 2 [É]crivez un programme *php* pour automatiser la cryptanalyse du système de Vigenère. Pour cela, vous devrez au prálable calculer le spectre des fréquences des lettre en français, en analysant des textes libres de droits diffusés sur internet. Les signes diacritiques seront systématiquement oubliés, ainsi "â" sera interprété comme un "a" (idem à=a, ç=c, é=e, œ=o+e, etc.) Vous comparerez les résultats obtenus avec ceux donnés dans ce cours. Déchiffrez alors le texte suivant :

```
PIGLOWLRRVEMFJUIYRYEXAJIJAIXROFAEAFOMAWZWODYFLWKSBXUIVBSMSNYIEKYFYMRYDMMFIXTZWVL
SDLRGEMPKRMUHHMHKWVCKKOIJIDEFRUVPSZVGHZANJWFTOWSIKXBEIRUDCFZWELIZKMWGXYSWKREMNRS
REFAVUXHNQJMAWLKIHIKEAMRHPACAFYVMIEOAKTKKXSRBSMSZRNTCJIQRQERZWUXBURHFAXWDRVINEJI
IVVIQYLKEGYGOCIYDEGVIVZZJZRXPQTEWZHOVOLYWAVBXSDPAFMOHVEUFKEKZMTZIIEOADMEETLXHQYQ
PWFISSITROVWUAHVIDNKIQRYEWSMWXXSBJKRSMVVFRLSTLHPIZNBWZHAWDIUYNEEDIQFJAIVXMEFSXIF
SIGZTGJEPDPAMQVHHYSGMGXJKOVZRLJWCGLONWFGPXVTDIXGRWKGCDSIDACOUSPYISLWIQHSEMMOISEA
REOGOAMPNCPHXVQJJWYOKHCEZZGUFSUIDXZICUIHYJHYBYUIVBSLRRDMEPXKPLEVQJWFWXFUIVREKOYL
TUOJFYMKENCLWILZFWVTPGNTXEBTJEHXUIQNKGYOKRHUOJXIPLLELNEWLHRKMGFKXGRHAGMDAZHGRZXJ
IDFGAOJJISJSBTBMLERWVEYJQQHOERUAJIHAFZZKVHAEOAMZZISXMIHYMPQXSIHKXI
```

6. Chiffrement affine

Le chiffrement affine, qui n'est pas d'un grand intérêt pratique n'est présenté ici que pour introduire quelques notions qui vont être reprises à travers le protocole de chiffrement à clef publique RSA. Comme son nom l'indique, le chiffrement affine consiste à utiliser une fonction de chiffrement affine $x \mapsto ax+b$. On se place, comme pour le chiffrement de César, dans l'anneau $\mathbf{Z}_{26}$. L'espace des clairs est encore confondu avec l'espace des chiffrés et vaut $\mathbf{Z}_{26}$ tandis que l'espace des clefs est *en première approche* l'ensemble $\mathbf{Z}_{26} \times \mathbf{Z}_{26}$. La fonction de chiffrement est donnée par

$$e_k(x) = ax + b \mod 26 \quad \text{avec} \quad k = (a,b). \tag{10}$$

Même s'il apparaît immédiatement que l'espace des clefs est beaucoup trop réduit pour pouvoir en tirer un système concluant, on va tout de même étudier ce système pour en tirer quelques informations. Comment déchiffrer un message y une fois passé dans la moulinette e_k ? Nous avons vu que les fonctions de chiffrement et de déchiffrement doivent être toutes deux injectives. La question est donc posée pour la fonction e_k définie par l'équation (10).

Autrement dit, pour tout $y \in \mathbf{Z}_{26}$, l'équation

$$ax + b \equiv y \pmod{26} \tag{11}$$

doit admettre une unique solution.

Cette équation est équivalente à $ax \equiv y - b \pmod{26}$ et comme y décrit $\mathbf{Z}_{26}$, $y - b$ également (une translation est une bijection). La recherche des soluttions à cette dernière équation revient se ramène à l'étude, plus simple, pour tout $y \in \mathbf{Z}_{26}$ des solutions de l'équation

$$ax \equiv y \pmod{26}.$$

La réponse à cette question est donnée par le théorème suivant (cf. Hardy & Wright "An Introduction to the Theory of Numbers", pp. 51 th. 57,pp. 94) :

THÉORÈME 4. *Soient a, b et m trois entiers et soit $d = (a, m)$ le pgcd de a et m. Alors la congruence*

$$ax \equiv b \pmod{m}$$

admet exactement d solutions si $d|b$, et n'en admet aucune dans le cas contraire.

Dans notre cas $m = 26$, et en appliquant le théorème on doit trouver une unique solution quand $(a, m) = 1$. La question est maintenant de trouver quelles sont les entiers a inférieurs à m et premiers avec m ? Cette fois la réponse est donnée par Euler, mais il nous faut rappeler quelques autres résultats. Le premier est le théorème assurant l'unicité de la décomposition en produit de facteurs premiers d'un entier n, souvent appelé "théorème fondamental de l'arithmétique" tant il est capital.

THÉORÈME 5 (Théorème fondamental de l'arithmétique). *Tout entier n se décompose de manière unique, à une permutation près, en un produit de facteurs premiers.*

Preuve . Si n est premier, le théorème est prouvé, donc supposons que n ne soit pas premier. Parmi les diviseurs de n différents de 1, considérons le plus petit p_1. Celui-ci est nécessairement premier, sans quoi tout diviseur de p_1 serait un diviseur de n donc plus petit que p_1 ce qui est absurde. On peut donc écrire $n = p_1 n_1$ avec $n_1 < n$. On recommence le même raisonnement avec n_1, et ainsi de suite. La suite des n_i ainsi définie est strictement décroissante, elle ne peut donc continuer indéfiniment d'où le résultat sur la décomposition.
Rien dans cette décomposition ne permet d'affirmer qu'elle est unique. Supposons qu'il existe des entiers admettant plusieurs décompositions en produits de facteurs premiers, et soit n le plus petit d'entre eux. On note

$$n = p_1 p_2 p_3 \dots p_k \quad \text{et} = \quad n = q_1 q_2 q_3 \dots q_l$$

deux décompositions différentes de n en supposant que les facteurs p_i sont rangés dans l'ordre croissant, ainsi que les q_j. S'il existait deux premiers p_i et q_j égaux, alors l'entier $n/p_i = n/q_j$, strictement inférieur à n, admettrait deux décompositions distinctes ce qui est impossible puisque n était le plus petit. Comme $n \geq p_1^2$ et $n \geq q_1^2$ et que $p_1 \neq q_1$, alors $n > p_1 q_1$. D'autre part, $p_1|n$ et $q_1|n$ donc $p_1 q_1|n$ soit $q_1|\frac{n}{p_1}$. Comme $n/p_1 < n$, il n'admet qu'une décomposition $p_2 p_3 \dots p_k$, ainsi q_1 est l'un des p_i ce qui n'est pas possible. □

Si n est un entier positif, on considère la *fonction indicatrice d'Euler* $\varphi : \mathbf{N} \to \mathbf{N}$ définie par

$$\varphi(n) = \#\{a \in \mathbf{N}, \quad 0 < a < m,\ (a, m) = 1\}.$$

THÉORÈME 6. *Soit n un entier positif et $\prod_{i=1}^{r} p_i^{\nu_i}$ sa décomposition en produit de facteurs premiers. Alors*

$$\varphi(n) = \prod_{i=1}^{r} p_i^{\nu_i - 1}(p_i - 1).$$

En appliquant le théorème, on obtient $\varphi(26) = (13-1)(2-1) = 12$, il y a donc 12 entiers premiers avec 26, ce sont les entiers

$$1,\ 3,\ 5,\ 7,\ 9,\ 11,\ 15,\ 17,\ 19,\ 21,\ 23,\ 25.$$

Finalement l'espace des clefs pour le chiffrement affine n'est pas $\mathbf{Z}_{26} \times \mathbf{Z}_{26}$ comme nous l'avions estimé en introduction, mais l'ensemble

$$K_{26} = \{(a, b) \in \mathbf{Z}_{26} \times \mathbf{Z}_{26},\ (a, 26) = 1\}$$

qui est de cardinal $\varphi(m)m$, soit dans notre situation $12 \times 26 = 312$. Se pose maintenant le problème du déchiffrement, autrement dit pour y fixé en supposant que $(a, m) = 1$, il faut calculer le seul x qui satisfait (11), ou encore résoudre

$$ax \equiv y - b \pmod{26}.$$

Bien sûr, si le calcul se faisait dans un corps, la réponse serait immédiate, il suffit de diviser par a les deux membres de l'égalité pour aboutir au résultat, mais malheureusement nous travaillons dans un anneau. Ainsi pour diviser par a, il faut nous assurer au préalable qu'il admet bien un inverse et pour cela il nous faut encore un peu de théorie. On rappelle que si A désigne un anneau, on note A^* l'ensemble des éléments inversibles de A, autrement dit, l'ensemble des $x \in A$, tels qu'il existe un unique $x' \in A$ pour lequel $xx' = x'x = 1_A$. On montre trivialement que cet ensemble est un sous-groupe multiplicatif de $A^\times$. On a par exemple $\mathbf{Z}^* = \{-1, 1\}$.

Exercice 3 [2/5] Soit m un entier positif. Montrez que pour tout élément a non-nul de $\mathbf{Z}/m\mathbf{Z}$, les trois propriétés suivantes sont équivalentes :

(1) $a \in (\mathbf{Z}/m\mathbf{Z})^*$;

(2) a n'est pas un diviseur de 0 dans $\mathbf{Z}/m\mathbf{Z}$, autrement dit le seul élément b de $\mathbf{Z}/m\mathbf{Z}$ tel que $ab = 0$ est $b = 0$;

(3) $(a, m) = 1$.

À la lumière de cet exercice, l'ordre du groupe $(\mathbf{Z}/m\mathbf{Z})^*$ est donc $\varphi(m)$ et tous les éléments a premiers avec m admettent un inverse a^{-1}. Parfait, mais tout ceci ne nous dit pas comment calculer *explicitement* l'inverse a^{-1} de a ! C'est l'algorithme d'Euclide étendu qui nous permet d'y arriver.

ALGORITHME 1 : EUCLIDE ÉTENDU

Entrée : deux entiers positifs a et m.
Sortie : deux entiers d et b tels que $d = (a, m)$ et $ba \equiv d \pmod{m}$.
Règles :

(12a) $$(m, a) \mapsto (m, a, 0, 1)$$

(12b) $$(r, r', t, t') \mapsto (r', r - qr', t', t - qt') \quad \text{si } r' \neq 0 \quad (\text{avec } q = r \div r')\,;$$

(12c) $$(r, r', t, t') \mapsto (r, t) \quad \text{si } r' = 0.$$

Exercice 4 [3/5] Exécutez l'algorithme "à la main" pour $a = 6$ et $m = 33$. Démontrez la justesse de l'algorithme d'Euclide étendu. Pour cela, on note $(r_i, r_{i+1}, t_i, t_{i+1})$ le quadruplet obtenu à chaque application de la règle (12b), avec

$$r_0 = m,\ r_1 = a, \quad t_0 = 0,\ t_1 = 1.$$

On veut donc montrer d'une part qu'il existe un entier k tel que $r_{k+1} = 0$ et que $r_k = (a, m)$. D'autre part, on veut montrer que $t_k a \equiv r_k \pmod{m}$. Montrez par récurrence que

$$\forall j \in [0, k], \quad r_j \equiv t_j a.$$

En déduire que si $(a, m) = 1$ alors $t_k = a^{-1} \mod m$.

CHAPITRE 2

Théorie de l'information

1. Confidentialité parfaite

Dans l'étude de la sécurité d'un système cryptographique, il y a deux grandes approches théoriques. La première est la sécurité *calculatoire* qui fait appel à la difficulté pratique qu'il y a à effectuer certains calculs. La théorie sous-jacente est la théorie de la complexité que nous aborderons plus loin dans ce cours. La seconde que nous allons aborder dans les lignes qui suivent est la sécurité *inconditionnelle*, totalement indépendante des moyens de calculs dont on dispose. Le cadre théorique naturelle est la théorie de l'information de Shannon[1]. Pour étudier la sécurité inconditionnelle, il faut préciser quel est le type d'attaque que va tenter Oscar :

(1) chiffré connu : Oscar connaît des chiffrés y uniquement ;

(2) clair connu : Oscar connaît des couples clairs-chiffrés (x, y) ;

(3) clair choisi : Oscar peut choisir x et obtenir y par une machine de chiffrement ;

(4) chiffré choisi : Oscar peut momentanément déchiffrer un message y en x.

Les 4 attaques ont été donnée dans l'ordre croissant des moyens qui s'offrent à Oscar pour découvrir la clef k, la dernière étant rarement réalisée.

2. Mini lexique sur les probabilités

On rappelle brièvement quelques notions de probabilité : quand on fait une *expérience*, on observe un *résultat* pour lequel on a une *description* (M ou F pour désigner le sexe de l'enfant issu d'un accouchement, la sécu préfère 1 ou 2, les numéros 1 à 49 d'une boule lors d'un tirage du loto, les résultats sous forme de couples d'un match de foot, etc.).

L'ensemble Ω des descriptions est appelé *espace des épreuves* et doit être en adéquation avec les résultats de l'expérience menée, ainsi $\Omega = \{0,1\}^n$ est un espace adapté à la description de l'expérience consistant à lancer n fois une pièce de monnaie et à noter les valeurs successives *pile* 0 ou *face* 1. .
Il faut quelques hypothèses théoriques pour que tout ceci nous permettent de faire des calculs : une partie $\mathcal{F} \subseteq \mathcal{P}(\Omega)$ est appelée *tribu* ou *σ-algèbre* sur Ω si elle possède les propriétés suivantes :

(1) $\varnothing \in \mathcal{F},\ \Omega \in \mathcal{F}$;

(2) $\forall A \in \mathcal{F},\ \overline{A} \in \mathcal{F}$;

1. décédé il y a quelques jours à la date où je rédige ces lignes

(3) $\forall (A_i)_{i\in I} \in \mathcal{F}$, où I est au plus dénombrable,

$$\bigcup_{i\in I} A_i \in \mathcal{F}.$$

Par exemple, $\mathcal{P}(\Omega)$ et $\{\varnothing, \Omega\}$ sont deux tribus sur Ω, respectivement appelées tribu *triviale* et tribu *grossière* sur Ω. Un *évenement* est un élément de la tribu $\mathcal{F}$. Par exemple, la suite des lancers contient au moins deux tiers de *pile*. On dit qu'une epreuve $\omega \in \Omega$ *réalise* l'évènement $A \subset \Omega$ si $\omega \in A$. Ainsi les trois épreuves 010, 100 et 001 réalisent lévènement "au moins deux tiers de pile" pour $n = 3$. Une épreuve qui appartient à un évènement A de $\mathcal{F}$ est appelé une *réalisation* de A. Généralement une tribu est choisie *a posteriori*, plus précisément, quand on cherche à modéliser un phénomène probabiliste, on considère les évènements intéressants à étudier et on construit la plus petite tribu qui les contient.

DÉFINITION 7. *Soit Ω un ensemble quelconque et $\mathcal{F}$ une tribu sur Ω. On appelle* probabilité sur $(\Omega, \mathcal{F})$ *toute fonction $P : \mathcal{F} \to [0, 1]$ qui satisfait :*

(1) $P(\varnothing) = 0,\ P(\Omega) = 1$ *;*

(2) P *est* sous σ-additive, *i.e.* $\forall (A_i)_{i\in I} \in \mathcal{F}$, *où* I *est au plus dénombrable,*

$$P\big(\bigcup_{i\in I} A_i\big) \leq \sum_{i\in I} P(A_i).$$

avec égalité si et seulement si les A_i sont deux-à-deux disjoints (σ-additivité).

Dans ce cas, le triplet $(\Omega, \mathcal{F}, P)$ est appelé espace probabilisé.

Une application $X : \Omega \to E$ est appelée *variable aléatoire continue* si $E = \mathbf{R}$ et *discrète* (v.a.d. en abrégé) si E est au plus dénombrable. Dans ce dernier cas, il faut également que

$$\forall x \in E, \quad \{\omega \in \Omega,\ X(\omega) = x\} \in \mathcal{F}$$

DÉFINITION 8. *Une fonction $p : E \to \mathbf{R}_+$ est appelée* distribution de probabilité *sur E si elle satisfait aux conditions suivantes :*

(1) $\forall x \in E,\ p(x) \in [0, 1]$ *;*

(2) $\sum_{x\in E} p(x) = 1.$

Si X est une v.a.d., on dira que p est la distribution de probabilité de la v.a.d. X si elle satisfait de plus :

$$P(\{\omega \in \Omega;\ X(\omega) \in F\}) = \sum_{x\in F} p(x).$$

Notons que $\forall F \subseteq E$, $\{\omega \in \Omega,\ X(\omega) \in F\} \in \mathcal{F}$ car E étant au plus dénombrable, F l'est également et la propriété (3) des tribus permet de conclure.
Si A et B sont deux évènements d'un espace probabilisé, on définit $P(A|B)$ appelée *probabilité conditionnelle* de A sachant P (ou encore probabilité de A si B) par

$$P(A|B) := \frac{P(A \cap B)}{P(B)}. \tag{13}$$

Les évènements A et B sont dits *indépendants* si $P(A|B) = P(A)$, autrement dit si l'occurrence de l'évènement B ne modifie pas la probabilité de l'évènement A. L'égalité (13) nous donne dans ce cas $P(A \cap B) = P(A)P(B)$. Cette égalité est souvent appelé formule de Bayes, alors qu'il ne s'agit que d'une définition. Notons que la fonction définie par $P_B(A) := P(A|B)$ est également une probabilité.

3. Retour à la théorie de l'information

à intégrer.

CHAPITRE 3

Chiffrement à clef public : RSA

Le protocole RSA est l'un des premiers systèmes de chiffrement à clef publique. Il a été proposé en 1977 par trois chercheurs Rivest, Shamir et Adleman dont RSA est l'acronyme. Sous une apparente simplicité, ce protocole cache une mine de résultats profonds sur la nature des nombres et le calcul. L'arithmétique sur $\mathbf{Z}$ et les nombres premiers y jouent un rôle central.

Ceci explique le nombre important de résultats d'arithmétique présentés ici et avec lesquels il est indispensable de se familiariser pour essayer de comprendre ce protocole en profondeur. La plupart des résultats sont élémentaires, en ce sens qu'ils découlent plus ou moins directement des définitions et des propriétés intrinsèques des objets manipulés, mais ils sont rarement triviaux. Nous n'avons abordé que les algorithmes les plus simples qui tournent autour de RSA. Le lecteur devra se tourner vers des ouvrages plus complets pour approfondir cette lecture, notamment sur les algorithmes de factorisation ou les tests de pseudo-primalité.

Le lecteur est supposé être familiarisé avec les principaux résultats d'algèbre du premier cycle universitaire et avoir quelques notions d'algorithmique ou de programmation. Nous n'avons pas développé (pour le moment) les prolongements naturels vers la théorie de la complexité qui sont pourtant incontournables (mais incompatibles avec le volume horaire du présent cours), notamment depuis que trois chercheurs indiens ont pu montrer en août 2002 que le problème de la primalité était dans la classe P des problèmes polynomiaux :

"Étant donné N un entier naturel, N est-il premier ?"

Ce résultat est à rapprocher du résultat de V. Pratt de 1978 qui montrait que ce problème de décicions appatient à NP $\cap$ co-NP. Il démontre que si N est premier il existe une preuve de sa primalité (on dit un *certificat*) que l'on peut vérifier en temps polynomial. La question duale

"Étant donné N un entier naturel, N est-il un entier composite ?"

c'est-à-dire existe-t-il deux entiers $p > 1$ et $q > 1$ teks que $N = pq$. Dans ce cas, un certificat évident est justement le couple (p, q) dont la vérification consiste à calculer le produit pq et s'assurer que $pq = N$ ce qui se fait en temps quadratique avec l'algorithme appris à l'école primaire par exemple.

1. Quelques mots sur les groupes et les anneaux

On rappelle qu'un *magma* est un couple $(G, \star)$ constitué d'un ensemble G et d'une loi de composition interne $\star$. Un *groupe* est un magma associatif, unifère et tel que tout élément est symétrisable, c'est-à-dire si la loi $*$ satisfait respectivement les trois propriétés suivantes :

(1) La loi $*$ est *associative*, i.e. $\forall (x,y,z) \in G^3$, $(x \star y) \star z = x \star (y \star z)$;

(2) Il existe un *élément neutre* e pour $\star$, i.e. $\forall x \in G$, $e \star x = x \star e = x$;

(3) Pour tout $x \in G$, il existe un *symétrique* x', i.e. $x \star x' = x' \star x = e$.

Par soucis d'économie, on dit "le groupe G" plutôt que le groupe $(G, \star)$. Si la loi $\star$ est commutative, le groupe est dit *commutatif* ou *abélien*. Si le cardinal de l'ensemble G sous-jacent est fini, le groupe est dit *fini* et le cardinal du groupe, noté $\#G$ (parfois $|G|$) est appelé *ordre* du groupe, sinon le groupe est dit *infini*. On réserve habituellement la notation additive $+$ aux groupes commutatifs et dans un cadre général, on privilégie la notation multiplicative car elle allège considérablement les écritures puisque le symbole de multiplication est quasi systématiquement omis quand aucune confusion n'est possible : on écrit ab plutôt que $a \times b$ ou $a.b$. Dans le cas où la loi est additive (resp. multiplicative), le symétrique x' d'un élément x est plutôt appelé *opposé* (resp. *inverse*) et noté $-x$ (res. x^{-1}). Pour les mêmes raisons, l'élément neutre pour une loi similaire à l'addition (resp. multiplication) est souvent noté 0 (resp. 1).

Un sous-ensemble H d'un groupe G, est un *sous-groupe* de G, s'il constitue un groupe pour la loi induite par celle de G. On peut caractériser un sous-groupe au moins de trois manières équivalentes :

(1) H est stable, $e \in H$ et $\forall x \in H$, $x^{-1} \in H$;

(2) H est stable, $H \neq \varnothing$ et $\forall x \in H$, $x^{-1} \in H$;

(3) $H \neq \varnothing$ et $\forall (x,y) \in H \times H$, $xy^{-1} \in H$.

Si H est réduit à $\{e\}$ de G, c'est le sous-groupe dit *trivial*. S'il est différent de G tout entier, H est un appelé un sous-groupe *propre*. Dans un groupe additif G totalement ordonné, on note $G_+ := \{x \in G,\ x \geq 0\}$. Un groupe totalement ordonné est dit *archimédien* s'il satisfait

$$\forall (x,y) \in G_+ \times G_+ \setminus \{e\},\ \exists n \in \mathbf{N}, \quad x \leq ny. \tag{14}$$

Exercice 5 [$\frac{2}{5}$] Montrez que le groupe additif de $\mathbf{Z}$ est archimédien.

Exercice 6 [$\frac{2}{5}$] Montrez que l'intersection d'une famille quelconque $(H_i)_{i \in I}$ de sous-groupes d'un groupe G est un sous-groupe de G.

Soient G un groupe d'élément neutre e et A une partie de G. On note $\mathrm{gr}(A)$ l'intersection [1] de tous les sous-groupes de G qui contiennent A, et on dit également que A est une partie génératrice de H. Dans le cas où A est réduite à un élément a, on écrit $\mathrm{gr}(a)$ ou encore (a) plutôt que $\mathrm{gr}(\{a\})$ et on dit que le groupe est *monogène*. Considérons l'ensemble

$$G_a := \{a^n,\ n \in \mathbf{Z}\}$$

où $a^n =$ désigne le produit de n exemplaires de a avec la convention $a^0 = e$, et a^{-n} désigne le produit de n exemplaires de a^{-1} dont on peut aisément vérifier qu'il s'agit de l'inverse de x^n. L'ensemble G_a est donc un sous-groupe de G d'après la propriété (1) des sous-groupes. L'intersection de tous les sous-groupes de G qui contiennent a est le plus petit sous-groupe de G (cf. exercice 6) donc inclus dans G_a, mais tout sous-groupe de G

1. c'est alors le "plus petit" sous-groupe de G contenant A. L'inclusion n'étant pas une relation d'ordre total, l'existence d'un plus petit élément n'est pas systématique.

qui contient a contient nécessairement toutes les puissances de a, donc G_a est inclus dans tout sous-groupe de G. Finalement

LEMME 9. *Soit a un élément d'un groupe G. Alors* $\mathrm{gr}(a) = \{a^n,\ n \in \mathbf{Z}\}$.

Deux cas se présentent alors : toutes les puissances de a sont distinctes, auquel cas le groupe (a) est *infini* (ce qui n'est possible bien entendu que dans le cas où le groupe G est lui-même infini), soit il existe deux entiers n et m, que l'on peut supposer positifs avec $n > m$ sans restreindre la généralité (pourquoi ?), et tels que $a^n = a^m$. Dans ce cas $a^{n-m} = e$ et le sous-ensemble $\{s > 0,\ a^s = e\}$ de $\mathbf{N}$ est non-vide, il admet donc un plus petit élément d. Ainsi, pour tout entier n de $\mathbf{Z}$, on a $a^n = a^r$ si r désigne le reste de la division euclidienne de n par d. D'autre part les éléments $a^0 = e,\ a^1, a^2, \ldots, a^{d-1}$ sont deux-à-deux distincts, sans quoi cela contredirait la définition de l'entier d, et constituent ainsi le groupe (a) qui est donc *fini*. Les puissances successives de a constituant une suite périodique, un cycle de cette suite forme le groupe (a). Un groupe monogène fini est donc appelé *groupe cyclique*.

DÉFINITION 10. *Soit a est un élément d'un groupe G. On appelle* ordre *de a, l'ordre du sous-groupe monogène (a) engendré par a.*

Exercice 7 [$\frac{2}{5}$] Soit H un sous-groupe d'un groupe G. Montrez que les relations $R_g(x,y) \equiv x^{-1}y \in H$ et $R_g(x,y) \equiv yx^{-1} \in H$ sont deux relations d'équivalence sur G et que les ensembles $aH = \{ah,\ h \in H\}$ et $Ha = \{ha,\ h \in H\}$ constituent les classes d'équivalence pour ces relations (appelées respectivement classes à gauche et à droite de H).

THÉORÈME 11. *Soit G un groupe fini et H un sous-groupe de G. Alors $\#H \mid \#G$.*

Preuve. Soit a un élément de G et H un sous-groupe de G. Les classes à gauche $aH = \{ah,\ h \in H\}$ constituent (propriété élementaire des relations d'équivalence) une partition de G. D'autre part, l'application $x \mapsto ax$ de H dans aH est bijective, les différentes classes sont donc toutes équipotentes. On sait que la somme des cardinaux des éléments d'une partition d'un ensemble fini est égale au cardinal de l'ensemble, d'où

$$r|H| = |G|$$

si r désigne le nombre des classes de la famille. L'entier r s'appelle l'*index* du sous-groupe H et son ordre divise donc celui de G. □

COROLLAIRE 12 (Lagrange). *L'ordre de tout élément d'un groupe fini G divise l'ordre du groupe G. De plus*

$$\forall x \in G, \quad x^{\#G} = e. \tag{15}$$

Preuve. La première partie est une conséquence directe du théorème 11. Si l'on note d l'ordre d'un élément $x \in G$, par définition, on a $x^d = e$. Comme $d \mid \#G$, il existe un entier r tel que $\#G = rd$, donc

$$x^{\#G} = x^{rd} = \left(x^d\right)^r = e^r = e.$$

□

PROPOSITION 13. *Soit H une partie finie non-vide d'un groupe G d'élément neutre e. Alors H est un sous-groupe de G si et seulement si $e \in H$ et H est stable pour la loi considérée.*

Preuve. La condition est par définition nécessaire, montrons qu'elle est suffisante. Il suffit de montrer que si $e \in H$ et $\forall (x,y) \in H \times H, xy \in H$ alors $\forall x \in H,\ x^{-1} \in H$. Comme H est stable, pour tout x de H et tout $n \in \mathbf{Z}$, $x^n \in H$, l'ordre d de x est donc nécessairement fini puisque H est fini. Si $d = 1$, alors H est réduit à $x^0 = e$ l'élement neutre. Si $d > 1$, on a $x^{d-1}x = xx^{d-1} = e$, l'inverse de x est donc x^{d-1} et appartient à H par stabilité. □

On rappelle qu'un anneau $(A, +, \times)$ est constitué d'un ensemble A et de deux lois de composition internes, l'*addition* et la *multiplication* (ou *produit*) telles que :

(1) $(A, +)$ est un groupe commutatif d'élément neutre noté 0 ;

(2) $(A, +)$ est un magma associatif unifère d'élément neutre noté 1 ;

(3) le produit est distributif sur l'addition.

Si le produit est commutatif, l'anneau est dit commutatif. Dans la suite, nous ne considèrerons que des anneaux commutatifs. Le sous-ensemble de A des éléments qui admettent un inverse constitue un groupe pour le produit, on l'appelle le *groupe des inversibles* et on le note A^*. Par exemple $\mathbf{Z}^* = \{\pm 1\}$. Dans un anneau commutatif, un élément non-nul a tel que $ab = 0$ avec $b \neq 0$ est appelé *diviseur de zéro*. Un anneau sans diviseurs de zéro est dit *intègre*.

Si tous les éléments non-nuls de l'anneau A sont inversibles alors A est appelé un *corps*. Un *idéal à gauche* (resp. à droite) est un sous-groupe additif I tel que $AI = I$ (resp. $IA = I$). Un idéal à gauche et à droite est dit *bilatère*, ce qui est toujours le cas si l'anneau est commutatif. Un idéal I bilatère est dit *principal* s'il existe un élément a tel que $I = aA$, on le note (a) (cf exercice 8). Un anneau *principal* est un anneau dont tous les idéaux sont principaux. Un anneau *archimédien* est un anneau dont le groupe additif est archimédien.

Exercice 8 [$\frac{2}{5}$] Montrer que l'idéal (a) est bien le groupe monogène engendré par (a) d'où la cohérence de la notation.

Exercice 9 [$\frac{2}{5}$] En utilisant le résultat de l'exercice 5, montrez en corollaire que l'anneau $\mathbf{Z}$ est muni de la division *euclidienne*, i.e. pour tout couple $(a, b) \in \mathbf{Z} \times \mathbf{Z} \setminus \{0\}$, il existe un unique couple $(q, r) \in \mathbf{Z}^2$ tel que

$$a = bq + r, \quad \text{et} \quad 0 \leq r < |b|. \tag{16}$$

2. Quelques notions d'arithmétique

L'arithmétique est la branche des mathématiques qui s'intéresse plus particulièrement aux nombres. L'anneau $\mathbf{Z}$ des entiers relatifs y joue un rôle central.

PROPOSITION 14. *Les sous-groupes additifs non-triviaux de $\mathbf{Z}$ sont de la forme $n\mathbf{Z}$ avec $n > 1$.*

Preuve . Soit H un sous-groupe additif non-trivial de $\mathbf{Z}$. Il contient donc au moins un élément non-nul et son opposé. L'ensemble des entiers positifs de H est non-vide et admet donc un plus petit élément n. Par division euclidienne, tout élément x de H peut s'écrire

$nq + r$ avec $0 \leq r < n$. Comme $n \in H$, par stabilité tout multiple de n est dans H, donc $nq \in H$ et donc $nq + r - nq = r \in H$. Ceci n'est possible que pour $r = 0$ sans quoi n ne serait pas le plus petit élément de H puisque $r < n$ et ainsi tous les entiers de H sont des multiples de n. □

En notant que $(n) = n\mathbf{Z}$, les $n\mathbf{Z}$ sont des sous-groupes monogènes de $\mathbf{Z}$ et comme $\mathbf{Z}n\mathbf{Z} = n\mathbf{Z}$, ce sont les seuls idéaux de $\mathbf{Z}$. Ainsi, tout idéal de $\mathbf{Z}$ est *principal*, on a donc

COROLLAIRE 15. *L'anneau* $\mathbf{Z}$ *est un anneau principal.*

L'application $n \mapsto n1_A$ de $\mathbf{Z}$ dans un anneau A d'élement neutre 1_A est un morphisme dont le noyau est un idéal de $\mathbf{Z}$ donc de la forme $n\mathbf{Z}$ d'après ces derniers résultats. L'entier n en question est appelé la *caractéristique de l'anneau* A.

Exercice 10 [$\frac{2}{5}$] Montrez que la caractéristique d'un anneau A ne peut être nulle que si A est infini. Déduire du corollaire ci-dessus qu'un anneau A non-nul, sans diviseurs de zéro a pour caractéristique 0 ou un nombre premier p.

THÉORÈME 16. *Soit* G *un groupe monogène. S'il est fini d'ordre* n *(donc cyclique), il est isomorphe à* $\mathbf{Z}/n\mathbf{Z}$*, sinon il est isomorphe à* $\mathbf{Z}$*. Dans les deux cas, il est commutatif.*

Exercice 11 [$\frac{2}{5}$] Démontrez le théorème 16.

Remarque 1. Il est tentant d'en déduire que le groupe multiplicatif d'un corps fini est commutatif et que par conséquent tout corps fini est commutatif. Mais la chronologie n'est pas respectée : le groupe multiplicatif d'un corps fini est cyclique *parce que le corps est commutatif.*

Si I et J désignent deux idéaux d'un anneau A, on appelle *idéal somme* noté $I + J$ le plus petit idéal de A qui contient toutes les sommes $au + bv$, $a \in I$, $b \in J$.

3. Identité de Bezout et algorithme d'Euclide

THÉORÈME 17. *Soient* a *et* b *deux entiers. L'idéal somme* $a\mathbf{Z}+b\mathbf{Z}$ *est égal à l'idéal* $(a, b)\mathbf{Z}$ *où* (a, b) *désigne le pgcd de* a *et* b*.*

Preuve. En effet, $\mathbf{Z}$ étant principal, il existe un entier d tel que $a\mathbf{Z} + b\mathbf{Z} = d\mathbf{Z}$. Montrons qu'il s'agit du pgcd de (a, b). Puisque $a \in a\mathbf{Z} + b\mathbf{Z}$ et $b \in a\mathbf{Z} + b\mathbf{Z}$, $d|a$ et $d|b$. Soit e un diviseur de a et b, alors $a\mathbf{Z} \subseteq e\mathbf{Z}$ et $b\mathbf{Z} \subseteq e\mathbf{Z}$, ainsi $a\mathbf{Z} + b\mathbf{Z} \subseteq e\mathbf{Z}$, soit $d\mathbf{Z} \subseteq e\mathbf{Z}$ ce qui entraîne $e|d$. □

On en déduit immédiatement le corollaire suivant qui généralise la célèbre identité de Bezout :

COROLLAIRE 18 (Identité de Bezout). *Soient* a*,* b *deux éléments de* $\mathbf{Z}$ *et* d *un entier. Il existe deux entiers relatifs* u *et* v *tels que*

$$au + bv = d. \tag{17}$$

si et seulement si $(a, b) \mid d$.

Preuve. La condition est nécessaire, en effet s'il existe deux entiers u et v tels que $au+bv = d$, alors $d \in (a,b)\mathbf{Z}$ d'après le théorème 17 et ainsi $(a,b) \mid d$. Elle est suffisante car si $(a,b) \mid d$, alors $d \in (a,b)\mathbf{Z}$ et $(a,b)\mathbf{Z} = a\mathbf{Z} + b\mathbf{Z}$ assure l'existence de u et v. □

Attention à l'identité de Bezout, si pour deux entiers a et b on peut exhiber un d tel que (17), cela ne permet pas d'affirmer que $(a,b) = d$ mais seulement $(a,b) \mid d$. Cependant ce résultat est vrai si $d = 1$. Contre-exemple : $2 \times 3 + 4 \times 2 = 14$, mais $(2,4) \neq 14$.

Nous allons à présent étudier un algorithme fondamental, l'algorithme d'Euclide étendu qui est une version modifié de l'algorithme d'Euclide permettant de calculer, non seulement le pgcd des deux entiers a et b, mais également deux entiers u et v qui satisfont l'équation (17).

Rappelons pour commencer l'algorithme d'Euclide. Il est basé sur le résultat élémentaire suivant : si d est un diviseur de a, alors $d|b \Leftrightarrow d|(a-b)$, ce qui prouve que $(a,b) = (a-b,b)$. Le calcul du pgcd de a et b se ramène donc au calcul du pgcd de a et $a-b$. Il n'y a bien entendu aucune raison de s'arrêter en si bon chemin et on peut répéter l'opération, c'est-à-dire soustraire b tant que le résultat reste supérieur à b. Algorithmiquement, on réalise une boucle "tant que" dans laquelle on réalise la réécriture $(a,b) \mapsto (a-b,b)$, la boucle s'achevant dès que $a-b < b$. Le lecteur perspicace se convaincra aisément que ce processus s'apparente à la division euclidienne de a par b sous sa forme soustractive, autrement dit la succession des réécritures de cette boucle peut avantageusement être remplacée par la simple règle de réécriture $(a,b) \mapsto (r,b)$, où $a = bq + r$ et $0 \leq r < b$.

Maintenant que le premier terme du couple est inférieur au second, il suffit de les permuter et de recommencer le processus jusqu'à ce que mort s'ensuive, i.e. jusqu'à ce que le reste de la division euclidienne soit nul. L'algorithme complet s'exprime alors avec les règles de réécriture suivantes :

$$\begin{aligned}(a,b) &\mapsto (b,a), \quad \text{si } a < b.\\ &\mapsto (b, a \mod b), \quad \text{si } b \neq 0.\\ &\mapsto a, \quad \text{si } b = 0.\end{aligned}$$

La première règle ne sert qu'à l'initialisation du processus pour assurer que le premier terme du couple est supérieur au second. La dernière règle donne le pgcd de a et b. On peut éviter la première règle à l'aide de l'observation suivante : si $a < b$, alors la division euclidienne de a par b donne $a = 0b + a$ et l'application de la règle $(a,b) \mapsto (b, a \bmod b)$ transforme le couple (a,b) en (b,a) et réalise donc la permutation rendant obsolète la première règle. Finalement l'algorithme d'Euclide s'écrit :

$$(a,b) \mapsto (b, a \mod b), \quad \text{si } b \neq 0. \tag{18a}$$

$$\mapsto a, \quad \text{si } b = 0. \tag{18b}$$

On peut à présent présenter l'algorithme d'Euclide étendu, mais pour cela, il nous faut disposer des restes obtenus lors des différentes divisions euclidiennes. L'application des règles de réécritures ci-dessus jusqu'à l'obtention du pgcd de a et b induit la séquence

d'équations suivante :

$$r_0 = q_1 r_1 + r_2 \qquad \text{avec } r_0 := a \text{ et } r_1 := b, \tag{19a}$$
$$r_1 = q_2 r_2 + r_3, \tag{19b}$$
$$\vdots \tag{19c}$$
$$r_{i-2} = q_{i-1} r_{i-1} + r_i, \tag{19d}$$
$$\vdots \tag{19e}$$
$$r_{n-2} = q_{n-1} r_{n-1} + r_n, \tag{19f}$$
$$r_{n-1} = q_n r_n + r_{n+1}, \tag{19g}$$
$$r_n = q_{n+1} r_{n+1} \qquad \text{avec } d := r_{n+1}. \tag{19h}$$

À partir de la séquence des restes r_i et des quotients q_i, on se propose de construire deux séquences u_i et v_j telles que

$$\forall i \in [1, n], \quad d = u_{i-1} r_{i-1} + v_i r_i. \tag{20}$$

Si l'on peut construire de telles suites, les entiers u et v recherchés seront respectivement les termes u_0 et v_1. Nous allons construire les valeurs de ces deux suites en partant des deux derniers termes u_{n-1} et v_n déterminés par l'équation (19g) où l'on a isolé r_{n+1}, c'est-à-dire d le pgcd de a et b :

$$d = r_{n-1} - q_n r_n, \quad \text{donc } u_{n-1} = 1 \text{ et } v_n = -q_n. \tag{21}$$

On vient donc de réaliser l'égalité (20) pour $i = n$. À partir de cette même égalité, on peut remplacer r_i par sa valeur obtenue à l'aide de l'équation (19d) et obtenir

$$\begin{aligned} d &= u_{i-1} r_{i-1} + v_i r_i \\ &= u_{i-1} r_{i-1} + v_i (r_{i-2} - q_{i-1} r_{i-1}) \\ &= v_i r_{i-2} + (u_{i-1} - q_{i-1} v_i) r_{i-1} \end{aligned}$$

On en déduit finalement que

$$\forall i \in [2, n], \quad u_{i-2} = v_i \quad \text{et} \quad v_{i-1} = u_{i-1} - v_i q_{i-1}.$$

En réécrivant ces deux égalités sous forme matricielle, on obtient pour tout $i \in [2, n]$,

$$\begin{pmatrix} u_{i-2} \\ v_{i-1} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & -q_{i-1} \end{pmatrix} \begin{pmatrix} u_{i-1} \\ v_i \end{pmatrix} \tag{22}$$

Pour $i = 2$ cela nous donne :

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} u_0 \\ v_1 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & -q_1 \end{pmatrix} \begin{pmatrix} u_1 \\ v_2 \end{pmatrix}. \tag{23}$$

Puis en appliquant $n - 1$ fois l'égalité (20) on conclut avec

$$\begin{pmatrix} u \\ v \end{pmatrix} = \left(\prod_{i=1}^{n} Q_i \right) \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \quad \text{où} \quad Q_i := \begin{pmatrix} 0 & 1 \\ 1 & -q_i \end{pmatrix} \tag{24}$$

Cette dernière égalité nous montre qu'il est inutile de conserver les quotients et les restes successifs de l'algorithme d'Euclide pour construire les entiers u et v une fois le pgcd

calculé, mais que l'on peut au fur et à mesure calculer le produit des matrices pour obtenir les entier u et v en même temps que le pgcd.

L'algorithme d'Euclide étendu s'en déduit directement, il suffit de calculer un produit matriciel à chaque division euclidienne. En notant

$$\Pi_k = \begin{pmatrix} a & a' \\ b & b' \end{pmatrix} \tag{25}$$

la matrice produit $Q_1.Q_2 \dots Q_k$ des k premières matrices Q_i, $i \in [1,k]$ on a $\Pi_{k+1} = \Pi_k Q_{k+1}$ et

$$\Pi_{k+1} = \begin{pmatrix} a' & a - a'q_{k+1} \\ b' & b - b'q_{k+1} \end{pmatrix}. \tag{26}$$

La programmation de cet algorithme revient donc à calculer deux fonctions affines à chaque division euclidienne fournissant un nouveau quotient q_i.

Exemple 2. On veut calculer $(71, 25)$, on a la séquence

$$\begin{aligned} 71 &= \mathbf{2} \times \mathbf{25} + \mathbf{21} \\ 25 &= \mathbf{1} \times \mathbf{21} + \mathbf{4} \\ 21 &= \mathbf{5} \times \mathbf{4} + \underline{\mathbf{1}} \\ 4 &= \mathbf{4} \times \mathbf{1} \end{aligned}$$

Donc $d = \underline{\mathbf{1}}$, $q_1 = 2$, $q_2 = 1$ et $q_3 = 5$. On a donc la séquence de matrices :

$$\begin{pmatrix} 0 & 1 \\ 1 & \mathbf{-2} \end{pmatrix} \xrightarrow{\mathbf{1}} \begin{pmatrix} 1 & -1 \\ -2 & 3 \end{pmatrix} \xrightarrow{\mathbf{5}} \begin{pmatrix} -1 & \mathbf{6} \\ 3 & \mathbf{-17} \end{pmatrix}$$

et ainsi $u = 6$ et $v = -17$, soit

$$71 \times \mathbf{6} + 25 \times (\mathbf{-17}) = 1.$$

Lemme 19 (Lamé). *Soient a et b deux entiers tels que $a > b > 0$ de pgcd d. Si l'algorithme d'Euclide usuel applique n fois la règle (18a) (et donc 1 fois la règle (18b)), alors*

$$a \geq dF_{n+2}, \quad b \geq dF_{n+1}, \tag{27}$$

où $(F_n)_{n \in \mathbf{N}}$ est la suite de Fibonacci définie récursivement par

$$F_{n+1} := F_n + F_{n-1}, \quad \text{avec } F_0 = 0 \text{ et } F_1 = 1. \tag{28}$$

Preuve. Par récurrence, si $n = 1$, cela signifie que a est un multiple de b et donc que $d = (a,b) = b$. Dans ce cas, $F_2 = 1$ et $F_3 = 2$. D'autre part, $dF_2 = bF_2 = b \leq b$ et $dF_3 = 2b \leq a$ puisque $a = kb$ avec $k > 1$ car a est un multiple de b et $a > b$. Supposons à présent que l'on a appliqué la première règle $n+1$ fois, alors la première application de cette règle sur le couple (a,b) a donné le couple $(b, a-qb)$ auquel on peut appliquer l'hypothèse de récurrence pour en déduire que

$$b \geq dF_{n+2}, \quad a - qb \geq dF_{n+1}.$$

Comme $a - qb < a - b$, on a $a - b \geq dF_{n+1}$. En additionnant cette dernière inégalité avec $b \geq dF_{n+2}$, on arrive à $a \geq dF_{n+1} + dF_{n+2}$, soit $a \geq dF_{n+3}$. □

LEMME 20. *On a*

$$F_n = \frac{1}{\sqrt{5}}\left(\phi^n - (-\phi)^{-n}\right), \tag{29}$$

où ϕ est le nombre d'or $(\sqrt{5}+1)/2$.

Preuve. La définition récursive (28) de la suite de Fibonacci nous donne :

$$\begin{pmatrix} F_{n-1} \\ F_n \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} F_{n-2} \\ F_{n-1} \end{pmatrix} \quad \forall n > 0. \tag{30}$$

Ainsi, en appliquant cette égalité n fois, on a

$$\begin{pmatrix} F_{n-1} \\ F_n \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}^n \begin{pmatrix} F_0 \\ F_1 \end{pmatrix} \tag{31}$$

Le problème se ramène donc au calcul d'une puissance d'un endomorphisme. La théorie montre (cf. t. 1 p. 420) que si les p valeurs propres de l'endomorphisme sont λ_i de multiplicité r_i, alors le terme général de la suite est une combinaison linéaire des suites $n \mapsto n^k \lambda_i^n$ avec $1 \leq i \leq p$ et $0 \leq k \leq r_i - 1$. Ici le polynôme caractéristique est

$$\begin{vmatrix} X & 1 \\ 1 & X-1 \end{vmatrix} = X^2 - X - 1.$$

Les deux valeurs propres sont

$$\phi = \frac{1+\sqrt{5}}{2}, \quad \phi' = \frac{1-\sqrt{5}}{2}.$$

Comme $\phi\phi' = -1$, on a $F_n = a\phi^n - b(\phi)^{-n}$ et les deux premiers termes de la suite nous donnent :

$$\begin{aligned} a + \quad & b = 0 \\ \phi a - \phi^{-1} b &= 1 \end{aligned}$$

On en déduit que $a = 1/\sqrt{5}$ et $b = -a$ et le résultat en découle. □

COROLLAIRE 21. *Soient a et b deux entiers tels que $a \geq b \geq 0$. L'algorithme d'Euclide demande au plus*

$$\left\lfloor \log_\phi \sqrt{5} \frac{b}{d} \right\rfloor. \tag{32}$$

divisions euclidiennes où $d = (a, b)$ et ϕ est le nombre d'or $(\sqrt{5}+1)/2$.

Preuve. L'expression (29) peut s'écrire

$$F_n = \frac{\phi^n}{\sqrt{5}} - \frac{(-1)^n}{\phi^n \sqrt{5}}. \tag{33}$$

Pour tout n on a clairement $\phi^n\sqrt{5} > 2$, donc

$$\frac{\phi^{n+1}}{\sqrt{5}} - \frac{1}{2} < F_{n+1}.$$

donc d'après (27),

$$\frac{\phi^{n+1}}{\sqrt{5}} - \frac{1}{2} < \frac{b}{d}. \tag{34}$$

Ainsi, on en déduit

$$\begin{aligned} n &< \log_\phi \sqrt{5}\left(\frac{b}{d} + \frac{1}{2}\right) - 1 \\ &< \log_\phi \sqrt{5}\frac{b}{d} \end{aligned}$$

Comme $\log_\phi \sqrt{5}$ n'est pas entier, le résultat en découle. □

L'identité de Bezout nous permet de démontrer le résultat important :

THÉORÈME 22. *Soit m un entier positif. Soit $a \in \mathbf{Z}/m\mathbf{Z}$, les trois propriétés suivantes sont équivalentes :*

(1) $a \in (\mathbf{Z}/m\mathbf{Z})^*$;

(2) *a n'est pas un diviseur de 0 dans $\mathbf{Z}/m\mathbf{Z}$, autrement dit le seul élément b de $\mathbf{Z}/m\mathbf{Z}$ tel que $ab = 0$ est $b = 0$;*

(3) $(a, m) = 1$.

Preuve. (1) ⇒ (2) est évident. Montrons que (2) ⇒ (3) par contraposition : supposons que a ne soit pas premier avec m, il existe donc $k \neq \pm 1$ tel que $a = ka'$ et $m = km'$ et ainsi $am' = a'm$ donc $am' \equiv 0 \pmod m$ et a est alors un diviseur de 0. Reste à montrer que (3) ⇒ (1) : si a est premier avec m, d'après Bezout, il existe u et v tels que $au + vm = 1$, donc $au \equiv 1 \pmod m$ et ainsi u est l'inverse de a modulo m. □

COROLLAIRE 23. *L'anneau commutatif $\mathbf{Z}/n\mathbf{Z}$ ($n \geq 1$) est intègre si et seulement si n est un nombre premier.*

Si on réduit l'équation (17) modulo b, on en déduit que u est l'inverse de a modulo b, i.e. que u est l'inverse de a dans l'anneau $\mathbf{Z}/b\mathbf{Z}$. Ainsi le calcul de cet inverse se fait également avec l'algorithme d'euclide étendu. Il nous faut à présent calculer le coût d'un division euclidienne puisque c'est l'opération fondamentale utilisée dans l'algorithme d'Euclide. Le lemme suivant nous sera utile dans toute la suite.

LEMME 24. *Soit N un entier positif et soit $b > 1$ un entier. Le nombre de chiffres, noté $|N|$, de l'entier N dans son écriture en base b est donné par*

$$|N| = \lfloor \log_b N \rfloor + 1. \tag{35}$$

Preuve. Soit n le nombre de chiffres de N, cela signifie que dans l'écriture de N en base b ci-dessous, le chiffre a_{n-1} est non-nul :

$$N = a_0 + a_1 b + a_2 b^2 + \cdots + a_{n-1} b^{n-1}.$$

Comme tous les a_i sont positifs, on peut minorer N par le plus petit entier à n chiffres, c'est à dire b^{n-1} (tous ses chiffres a_i sont nuls sauf $a_{n-1} = 1$) et le majorer par le plus

grand entier à n chiffres en base b, c'est-à-dire l'entier dont tous les chiffres sont égaux à $b-1$ soit

$$\sum_{i=0}^{n-1}(b-1)b^i = (b-1)\sum_{i=0}^{n-1} b^i = b^n - 1.$$

on a donc l'encadrement

$$b^{n-1} \leq N \leq b^n - 1.$$

L'inégalité de gauche nous permet d'affirmer que $n-1 \leq \log_b N$, soit $n-1 \leq \lfloor \log_b N \rfloor$ car n est entier, donc

$$n \leq \lfloor \log_b N \rfloor + 1.$$

L'autre inégalité large se transforme en l'inégalité stricte $N < b^n$, soit $\log_b N < n$ et $\lfloor \log_b N \rfloor < n$ que l'on restransforme en inégalité large car il s'agit d'entiers :

$$\lfloor \log_b N \rfloor + 1 \leq n.$$

□

4. La division euclidienne

Toutes les machines disposent généralement d'un jeu d'instructions arithmétiques de base, à savoir les 4 opérations. Le seul problème est que ce jeu d'instructions s'applique uniquement sur des entiers dont la taille ne dépasse pas celle d'un registre machine, autrement dit 32 bits sur les PC actuels et au mieux 64 bits pour la prochaine génération. On se convaincra aisément que le produit P de deux nombres entiers N et M de n chiffres et m chiffres en base b respectivement est un entier qui s'écrit sur $n+m$ chiffres dans la base b. Ceci montre que pour la multiplication, si la machine de référence est sur k bits, on ne peut utiliser l'instruction produit que si la taille deux entiers est inférieure à $k/2$. Pour fixer les idées, on aura donc intérêt sur un PC à choisir la base $b = 2^{16}$ de manière à ce que les deux chiffres d'un produit de deux nombres à un chiffre tienne sur un registre.

Exercice 12 [$\frac{1}{5}$] Soient N et M deux entiers représentés respectivement sur n et m chiffres en base b. Calculez le nombre de chiffres du produit NM. Même question pour une division.

L'objectif de ce qui suit est de réaliser un algorithme de division eulidienne sur des nombres arbitrairement grand représentés en base b en s'appuyant uniquement sur les instructions d'une machine capable de travailler sur des nombres d'un ou deux chiffres en base b. Typiquement en langage C, la base est 2^{16}, un nombre est représenté par un `short int` sur 16 bits *mais on suppose toujours que les calculs se font vers un* `long int` *sur* 32 *bits de manière à ce qu'un produit puisse être effectué sans dommage*. On note donc c_+, c_-, $c_\times$, et $c_\div$ le coût des opérations correspondantes sur une machine de référence en base b.

Pour ne pas faire de digressions trop importantes sur le calcul dit multiprécision, nous allons balayer "rapidement" les complexités des algorithmes les plus simples (donc pas nécessairement les plus efficaces) réalisant ces 4 opérations fondamentales sur des entiers dont le nombre de chiffres est arbitrairement grand.

Commençons par l'addition (ou la soustraction) sur deux entiers de même taille n en base b, $x = (x_1 x_2 \dots x_n)_b$ et $y = (y_1 y_2 \dots y_n)_b$. Le principe est directement calqué sur la

méthode apprise en cours élémentaire. On additionne les chiffres x_n et y_n modulo b et on propage la retenue si nécessaire. La complexité est donc linéaire en nombre de chiffres.

Le principe de la division euclidienne est quant à lui calqué sur l'algorithme "manuel" où l'on détermine à chaque étape un nouveau chiffre du quotient (dans l'ordre décroissant des puissances de la base). Toute la subtilité de l'algorithme que nous allons décrire consiste à déterminer une stratégie efficace pour le choix de ce chiffre. La référence pour la question est comme souvent Knuth, t.2, p. 255-262. On suppose que l'on cherche à calculer le quotient q et le reste r de la division euclidienne d'un entier x par un entier y. On suppose que x s'écrit sur $n+m$ chiffres et y sur n chiffres en base b, soit

$$x = (x_1 x_2 \dots x_{n+m})_b \text{ et } x = (y_1 y_2 \dots y_n)_b,$$

où les indices des chiffres suivent l'ordre d'écriture naturel (i.e. x_{n+m} et y_n sont les chiffres des unités). Déterminons le nombre de chiffres nécessaires au maximum pour l'écriture du quotient q. Le quotient q est maximum quand le dividende x est maximum et le diviseur y est minimum, donc pour $x = b^{n+m} - 1$ et $y = b^{n-1}$. On a donc

$$q = \left\lfloor \frac{b^{n+m} - 1}{b^{n-1}} \right\rfloor = b^{m+1} - 1.$$

D'après le lemme 24, le nombre de chiffres de q vaut

$$1 + \left\lfloor \log_b(b^{m+1} - 1) \right\rfloor = m + 1.$$

Dans la suite, on notera $q = (q_0 q_1 \dots q_m)$. L'algorithme "manuel" qui permet de calculer les $m+1$ chiffres q_0, q_1,... du quotient q dans cet ordre est basé sur $m+1$ applications du même principe de base. On se contentera donc de décrire ce principe pour l'obtention du premier chiffre q_0. On considère le nombre $x' := (x_1 x_2 \dots x_n)_b$ constitué par les n premiers chiffres de x. S'il est strictement plus petit que y, on lui rajoute un chiffre de plus, i.e. $x' = (x_1 x_2 \dots x_{n+1})_b$. On calcule le quotient de x' par y qui est le premier chiffre q_0 de q. On soustrait alors $q_0 b^m y$ (ou $q_0 b^{m-1} y$ si l'on a dû ajouter le chiffre x_{n+1} à x') à x et on recommence le même processus entre le nouveau x ainsi obtenu et y. Le nœud de l'algorithme consiste donc à évaluer le plus rapidement possible le quotient x'/y.

Bien entendu, du point de vue de la programmation, puisque l'on ne sait effectuer les opérations arithmétiques de base que sur 1 ou deux chiffres en base b, la question se ramène à savoir comment déterminer le quotient x'/y sans avoir à considérer tous les chiffres des nombres x' et y. Les deux lemmes suivants permettent de répondre à cette question et sont prouvés dans «*Knuth*»

LEMME 25. *Soient $x = (x_0 x_1 x_2 \dots x_n)_b$ et $y = (y_1 y_2 \dots y_n)_b$ deux entiers positifs tels que $x/y < b$. Alors l'entier*

$$\hat{q} := \min\left(\left\lfloor \frac{x_0 + x_1 b}{y_1} \right\rfloor, b - 1 \right). \tag{36}$$

vérifie

$$\hat{q} \geq \left\lfloor \frac{x}{y} \right\rfloor. \tag{37}$$

Autrement dit, l'estimation du quotient de x et y à l'aide des deux premiers chiffres de x et du premier chiffre de y est une estimation par excés.

Le lemme suivant explicite les conditions pour que cette estimation soit très proche de la bonne valeur du quotient :

LEMME 26. *Avec les notations introduites ci-dessus, si*

$$y_1 \geq \left\lfloor \frac{b}{2} \right\rfloor \tag{38}$$

alors on a

$$\hat{q} - 2 \leq \left\lfloor \frac{x}{y} \right\rfloor \leq \hat{q}. \tag{39}$$

Le fait important à noter est que l'erreur sur l'estimation du quotient ne dépend absolument pas de la base, uniquement du premier chiffre de y. Bien entendu, en moyenne la condition $y_1 \geq \lfloor b/2 \rfloor$ n'est satisfaite qu'une fois sur deux. Pour cette raison, avant de développer une procédure de calcul basée sur l'estimation $\hat{q}$, on normalise x et y en les multipliant tous les deux par l'entier

$$d := \left\lfloor \frac{b}{y_1 + 1} \right\rfloor, \tag{40}$$

ce qui ne change bien entendu rien au quotient de la division eulidienne de x par y, mais qui assure que le premier terme du diviseur satisfait bien la condition (38). On peut vérifier que cette normalisation (peu coûteuse puisque l'on multiplie x et y avec un nombre d'un seul chiffre) n'introduit aucun nouveau chiffre dans y et au plus un chiffre dans x.

Une fois la normalisation effectuée, l'algorithme est alors quasiment identique à celui que l'on fait "a la main", à deux différences près : la première concerne la détermination du chiffre courant du quotient q qui se fait empiriquement à la main avec n ou $n+1$ chiffres de x et les n chiffres de y alors qu'ici, ce chiffre est une estimation $\hat{q}$ qui ne fait intervenir que les deux premiers chiffres de x et le premier chiffre de y. Les inégalités (39) montrent qu'en soustrayant $\hat{q}b^m y$ ou $\hat{q}b^{m+1}y$ à x, on peut éventuellement obtenir un nombre négatif, auquel cas il suffira au pire de rajouter 2 fois y pour relancer le processus sur le bon résultat. Une fois le quotient et le reste déterminés, il ne reste plus qu'à les diviser par l'entier d qui avait servi à la normalisation. On calcule évidemment $\hat{q}$ au plus $m+1$ fois, c'est à dire $|y|$.

Exercice 13 [$\frac{3}{5}$] Exprimer la complexité de l'algorithme de la division euclidienne en fonction des constantes c_+, c_-, etc...

5. Retour à Z

Quel est le cardinal et plus généralement, quel est la structure du groupe $(\mathbf{Z}/n\mathbf{Z})^*$. On note $\varphi(n)$ ce cardinal appelé *indicateur d'Euler* de n qui indique donc le nombre d'entiers premiers avec n positifs et inférieurs à n d'après le théorème 22. Il nous faut encore quelques résultats avant de nous attaquer au calcul de $\varphi(n)$.

THÉORÈME 27 (Gauss). *Soient a, b et n trois entiers. Si n est premier avec a et $n \mid ab$, alors $n \mid b$.*

Preuve. Comme $(n, a) = 1$, on a $(nb, ab) = b$, mais $n \mid nc$ et par hypothèse, $n \mid ab$, donc n divise le pgcd de nb et ab soit b. □

THÉORÈME 28 (Euclide). *Soient a et b deux entiers et p un nombre premier. Si $p|ab$ alors $p|a$ ou $p|b$.*

Preuve. Si $p \nmid a$, alors $(p, a) = 1$ et Bezout nous donne l'existence de deux entiers u et v tels que $au + pv = 1$. On en déduit $abu + pbv = b$, or $p|ab$ et $p|pb$ donc $p|b$. □

THÉORÈME 29 (Théorème fondamental de l'arithmétique). *Soit n un entier, alors n s'écrit de manière unique sous la forme*

$$n = \prod_{i=1}^{k} p_i^{r_i}, \tag{41}$$

où tous les exposants r_i sont strictement positifs et les nombres p_i sont tous premiers et strictement croissants.

Preuve. L'écriture (41) s'obtient aisément une fois que l'on a montré que n s'écrit comme un produit de nombres premiers (pas nécessairement distincts). Il suffit alors d'ordonner ces nombres dans l'ordre croissant. Si n est premier, il n'y a rien à montrer, sinon n admet au moins un diviseur dans l'intervalle $[1, n]$ et on note d le plus petit de ces diviseurs. Alors d est nécessairement premier, sans quoi il ne pourrait être le plus petit diviseur de n. On note alors p_1 ce diviseur premier de n et on recommence le même processus avec l'entier $n_1 = n/p_1$. On obtient ainsi une suite décroissante d'entiers et le processus s'arrête dès que n_i est premier.

Rien dans cette décomposition ne permet d'affirmer qu'elle est unique. Supposons qu'il existe des entiers admettant plusieurs décompositions en produits de facteurs premiers, et soit n le plus petit d'entre eux. On note

$$n = p_1 p_2 p_3 \dots p_k \quad \text{et} = \quad n = q_1 q_2 q_3 \dots q_l$$

deux décompositions différentes de n en supposant que les facteurs p_i sont rangés dans l'ordre croissant, ainsi que les q_j. S'il existait deux premiers p_i et q_j égaux, alors l'entier $n/p_i = n/q_j$, strictement inférieur à n, admettrait deux décompositions distinctes ce qui est impossible puisque n était le plus petit. Comme $n \geq p_1^2$ et $n \geq q_1^2$ et que $p_1 \neq q_1$, alors $n > p_1 q_1$. D'autre part, $p_1|n$ et $q_1|n$ donc $p_1 q_1|n$ soit $q_1|\frac{n}{p_1}$. Comme $n/p_1 < n$, il n'admet qu'une décomposition $p_2 p_3 \dots p_k$, ainsi q_1 est l'un des p_i ce qui n'est pas possible. □

THÉORÈME 30 (Des restes chinois). *Soit $I = [1, k]$ et $(m_i)_{i \in I}$ une famille d'entiers strictement positifs et premiers entre eux dans leur ensemble. Soit m le produit des m_i. Soit Z l'anneau produit des $\mathbf{Z}/m_i\mathbf{Z}$, alors l'application $\pi : \mathbf{Z}/m\mathbf{Z} \to Z$ définie par*

$$\pi(x) = (x \bmod m_1, \dots, x \bmod m_k). \tag{42}$$

est un isomorphisme d'anneaux.

Preuve. La loi multiplicative dans Z étant définie composante à composante, il est aisé de vérifier qu'il s'agit bien d'un homomorphisme. Pour montrer qu'il est injectif, supposons que $\pi(x) = \pi(y)$, ceci équivaut à $\pi(x-y) = 0$ ce qui signifie que les restes des k divisions euclidiennes de $x-y$ par les m_i sont tous nuls et donc que $(x-y)$ est multiple de chaque m_i. Les m_i étant premiers entre eux, on en déduit que leur produit m divise $(x-y)$ et qu'ainsi $(x-y) \equiv 0 \pmod{m}$, donc $x = y$ dans $\mathbf{Z}/m\mathbf{Z}$. La surjectivité est immédiate puisque l'on a une application injective entre deux ensembles de même cardinal fini. Cependant nous allons donner une autre preuve de la surjectivité qui permet de calculer l'entier x à partir de ses résidus.

Soit $(r_1, r_2, \ldots, r_k) \in \mathbf{Z}$, on cherche un antécédent x à ce vecteur. On définit $\tilde{m}_i := m/m_i$ pour $i \in I$. On a clairement $(\tilde{m}_i, m_i) = 1$, on note alors μ_i l'inverse de $\tilde{m}_i$ modulo m_i qui existe d'après Bezout (on étudiera la preuve constructive de Bezout à l'aide de l'algorithme d'Euclide étendu qui permet d'exhiber cet inverse). Considérons l'entier

$$x := \sum_{i=1}^{k} r_i \mu_i \tilde{m}_i \mod m.$$

Pour tout $i \in I$, si l'on calcule x modulo m_i, tous les termes $r_j \mu_j \tilde{m}_j$ sont nuls si $j \neq i$ puisque $m_i | \tilde{m}_j$, par contre si $j = i$, comme μ_i est l'inverse de $\tilde{m}_i$ modulo m_i, le terme $r_i \mu_i \tilde{m}_i$ vaut r_i d'où le résultat. □

Le théorème des restes chinois est plus souvent connus à travers ce corollaire direct (qui est historiquement le bon) :

COROLLAIRE 31. *Soit $I = [1, k]$ et $(m_i)_{i \in I}$ une famille d'entiers strictement positifs et premiers entre eux dans leur ensemble. Soit m le produit des m_i. Alors pour toute famille $(r_i)_{i \in I}$ d'éléments de Z, le système d'équations*

$$x \equiv r_i \pmod{m_i}, \tag{43}$$

admet une unique solution modulo m.

LEMME 32. *Avec les mêmes hypothèses que pour le théorème des restes chinois, on a*

$$(\mathbf{Z}/m\mathbf{Z})^* \simeq \prod_{i=1}^{k} (\mathbf{Z}/m_i\mathbf{Z})^*. \tag{44}$$

Preuve . Ce résultat découle d'un résultat simple plus général sur les inversibles d'un anneau produit. □

COROLLAIRE 33. *Soit $n > 1$ un entier et $\prod_{i=1}^{k} p_i^{r_i}$ sa décomposition en produit de facteurs premiers. Alors*

$$\varphi(n) = n \prod_{p|n} (1 - \frac{1}{p}) \tag{45}$$

où p décrit l'ensemble des nombres premiers qui divisent n.

Preuve. En appliquant la proposition ci-dessus, les $p_i^{r_i}$ sont premiers entre-eux, le groupe des inversibles $(\mathbf{Z}/n\mathbf{Z})^*$ est donc isomorphe au produit des groupes $(\mathbf{Z}/p_i^{r_i}\mathbf{Z})^*$. L'indicateur d'Euler de ces groupes est facile à calculer, en effet, les seuls entiers positifs strictement inférieurs à $p_i^{r_i}$ qui ne lui sont pas premiers sont les multiples de p_i, il y en a exactement $p_i^{r_i-1}$. On a donc

$$\begin{aligned}\varphi(n) &= \prod_{i=1}^{k} \varphi(p_i^{r_i}) = \prod_{i=1}^{k} (p_i^{r_i} - p_i^{r_i-1}) = \prod_{i=1}^{k} p_i^{r_i}(1 - \frac{1}{p_i}) \\ &= n \prod_{p|n} (1 - \frac{1}{p}).\end{aligned}$$

□

THÉORÈME 34 (Euler). *Soit $n > 0$ un entier et un a un entier premier avec n, alors*

$$a^{\varphi(n)} \equiv 1 \pmod{n}. \tag{46}$$

Preuve. On regarde a dans $\mathbf{Z}/n\mathbf{Z}$, comme il est premier avec n, c'est un inversible et on conclut avec l'égalité 15 du théorème de Lagrange. □

Si on choisit $n = p$ où p est premier, on obtient en corollaire le très célèbre

COROLLAIRE 35 (Petit théorème de Fermat). *Soit p un nombre premier et a un entier premier avec p, alors*

$$a^{p-1} \equiv 1 \pmod{p}. \tag{47}$$

Preuve. Si p est premier, $\varphi(p) = p - 1$, d'où le résultat. □

Attention, la réciproque est malheureusement fausse, on ne peut déduire qu'un entier p est premier s'il satisfait la congruence (47), quand bien même celle-ci est satisfaite pour *tous* les entiers a premiers avec p. Par exemple, considérons l'entier $m = 561 = 3 \times 11 \times 17$. Soit a un entier premier avec 3, 11 et 17, donc avec 561. Le petit théorème de Fermat 35 nous donne $a^2 \equiv 1 \pmod 3$, $a^{10} \equiv 1 \pmod{11}$, $a^{16} \equiv 1 \pmod{17}$, mais 2, 10 et 16 divisent tous les trois 560, donc $a^{560} \equiv 1$ pour les trois modulos 3, 11 et 17, et comme ils sont premiers entre eux deux-à-deux, $a^{560} \equiv 1 \pmod{561}$ (si $a \equiv 1 \pmod m$ et $a \equiv 1 \pmod{m'}$ alors $a \equiv 1 \pmod{mm'}$ si $(m, m') = 1$). Les nombres qui satisfont la congruence pour tout a et qui ne sont pourtant pas premier sont les nombres de *Carmichael*.

La déception de ne pouvoir déduire de ce théorème un algorithme de test de primalité est (faiblement) compensée par le résultat important :

THÉORÈME 36. *Soit m un entier positif impair et a un entier premier avec m. Soit d l'ordre multiplicatif de a modulo m. Si $a^{m-1} \equiv 1 \pmod m$, alors $d \mid m - 1$. Si de plus $d = m - 1$, alors m est premier.*

Preuve. Soit d l'ordre multiplicatif de a modulo m. C'est par définition l'ordre d du sous-groupe monogène (a) de $(\mathbf{Z}/m\mathbf{Z})^\times$, donc $d \mid \varphi(m)$ d'après le théorème 12 de Lagrange. Comme $\varphi(m) \leq m - 1$, on a $d \leq m - 1$. On a $a^{m-1} = a^r(a^d)^q$ où a est le quotient de la division de $m - 1$ par d et r le reste. Comme $a^d \equiv 1 \pmod m$ et $a^{m-1} \equiv 1 \pmod m$, nécessairement $a^r \equiv 1 \pmod m$, ce qui contredirait que d est l'ordre multiplicatif de a

puisque $r < d$, sauf si $r = 0$, donc $d \mid (m-1)$. Si $d = m-1$, on en déduit que $\varphi(m) = m-1$ et ainsi que m est premier. □

6. Le protocole RSA

La définition d'un système de chiffrement à clef publique est identique à la définition d'un système cryptographique à clef secrète. Ce qui distingue les deux systèmes est que dans le cas du chiffrement à clef publique, une *partie* de la clef est rendue publique et seule cette partie est nécessaire pour le chiffrement. Autrement dit, la clef k est un couple (k_p, k_s) constitué d'une partie *publique* k_p indispensable au chiffrement et d'une partie *secrète* k_s inutile au chiffrement mais indispensable au déchiffrement. En pratique, le déchiffrement n'est censé être possible qu'en connaissant k_s. L'assymétrie sur l'utilisation de la clef dans le chiffrement et le déchiffrement d'un système à clef publique fait qu'on les appelle souvent systèmes *assymétriques* par opposition aux systèmes *symétriques* à clef secrète. On peut donc dire que la fonction de chiffrement utilise k_p et que la fonction de déchiffrement utilise k_p et k_s.

La cryptanalyse des systèmes à clef publique consiste donc à chercher comment déchiffrer rapidement uniquement à l'aide de la partie k_p de la clef ou encore de reconstituer k_s. Voici le protocole RSA :

PROTOCOLE RSA

Étape 1 [calcul de la clef] Bob se donne deux nombres premiers p et q, et calcule leur produit $N = pq$. Il se donne ensuite un entier e premier avec $\varphi(N) = (p-1)(q-1)$ et rend publiques les deux quantités N et e. Il garde les quantités p et q secrètes.

Étape 2 [chiffrement] Alice veut envoyer un message. Elle récupère les quantités publiques N et e laissées par Bob puis code son message sous la forme d'un entier naturel x strictement inférieur à N et premier avec N. Elle réalise le chiffrement suivant :

$$x \mapsto x^e \mod N \tag{48}$$

et transmet le chiffré $y := x^e \mod N$ sur le canal.

Étape 3 [déchiffrement] Bob calcule d l'inverse de e modulo $\varphi(N)$ et fait ensuite le déchiffrement suivant :

$$y \mapsto y^d \mod N(= x) \tag{49}$$

et retrouve ainsi le message clair x.

Montrons que Bob retrouve bien le message clair x. Si Bob calcule d l'inverse de e modulo $\varphi(n)$, cela signifie que $ed \equiv 1 \pmod{\varphi)(N}$, autrement dit que $ed = \lambda\varphi(N) + 1$, et ainsi

$$y^d \equiv x^{ed} \equiv x^{\lambda\varphi(N)+1} \pmod N$$

soit

$$y^d \equiv x(x^{\varphi(N)})^\lambda \equiv x \pmod N \tag{50}$$

d'après le théorème d'Euler 34.

La clef du système RSA est un quintuplet $k = (p, q, e, d, N)$ dont la partie publique est le couple (e, n) et la partie secrète est le triplet (p, q, d). seuls e et N sont nécessaires au chiffrement et sont rendus publics.

Dans les 4 sections suivantes, nous allons étudier brièvement les problèmes liés à la sécurité du système puis successivement celles liées aux étapes 1 à 3 du protocole RSA.

7. Oscar et RSA

La clef de voute du protocole repose sur le secret constitué par les entiers p et q. En effet, si Oscar est capable de déterminer les facteurs p et q de n, alors il en déduit $\varphi(N)$ et il lui est alors très facile de calculer d grâce à l'algorithme d'Euclide étendu. Déchiffrer le message d'Alice devient alors un jeu d'enfant. Une première question est donc : quelle est la difficulté de la factorisation d'un nombre entier ?

D'après le principe désormais bien connu de Kerckhoff, Oscar connaît parfaitement le fonctionnement du protocole et sait que $\varphi(N) = (p-1)(q-1) = N-(p+q)+1$. Supposons qu'il soit en mesure de calculer $\varphi(N)$, alors il en déduit la somme $S := p+q = N-\varphi(N)+1$ et il dispose donc à la fois du produit N et de la somme S de p et de q. Il suffit alors de résoudre l'équation

$$x^2 - Sx + N = 0$$

pour retrouver p et q et ainsi "casser" le système. Si RSA est sûr, il faut croire que la factorisation d'un entier N ou le calcul de son indicateur d'Euler $\varphi(N)$ sont des problèmes difficiles. C'est ce que nous allons brièvement étudier à présent.

Une première technique empirique pour factoriser N est basée sur le théorème fondamental de l'arithmétique qui nous suggère de calculer le reste des divisions entières avec tous les nombres premiers p inférieurs à N. Si le reste n'est jamais nul, N est premier, sinon on a trouvé un des facteurs. En remarquant que si l'on n'a trouvé aucun diviseur de N inférieur ou égal à $\sqrt{N}$, alors il est inutile de continuer. En effet, s'il existait un entier d strictement supérieur à $\sqrt{n}$ tel que $d \mid n$, alors n/d serait un diviseur de n strictement inférieur à $\sqrt{n}$, il aurait donc déjà été testé au travers des nombres premiers de sa décomposition.
La première question mérite d'étudier quelques problèmes connexes, par exemple sur le nombre de nombres premiers. Sont-ils en nombre fini ? Si non, y-en-a-t-il beaucoup ? La réponse à la première question a été donnée par Euclide et s'appuie sur le lemme suivant :

LEMME 37. *Tout nombre entier $n > 1$ est divisible par un nombre premier.*

Preuve. Soit $n_0 := n$. Si n_0 est premier le résultat est acquis, sinon il existe deux entiers n_1 et k_1 tels que $n_0 = k_1 n_1$ et $1 < n_1 < n$. Si n_1 est premier on a fini, sinon on peut recommencer le raisonnement pour $n_1 = k_2 n_2$, et de proche en proche créer ainsi une suite strictement décroissante $(n_k)_{k\in\mathbf{N}}$. La suite étant nécessairement finie l'un des n_i est premier. □

THÉORÈME 38 (Euclide). *Il y a une infinité de nombres premiers.*

Preuve. Par l'absurde : supposons qu'il y en ait un nombre fini n et notons les $p_1, p_2, \ldots, p_n$. On définit l'entier $p := 1 + p_1 p_2 \ldots p_n$ qui n'est divisible par aucun des p_i puisque $p \equiv 1 \pmod{p}_i$ pour tout i d'après le théorème de la division euclidienne. Dans ce cas, soit p

est premier, auquel cas la preuve est achevée, soit il contient un facteur premier nécessairement différents des p_i. Dans les deux cas, on a exhibé un nombre premier qui n'est pas dans la liste $p_1, p_2, \ldots, p_n$ ce qui est impossible. □

Remarque 3. les 5 premiers entiers p construits de cette manière sont premiers mais $1 + 2.3.5.7.11.13 = 30031 = 59.509$.

On définit alors $\pi(x)$, le nombre d'entiers premiers inférieurs à x. La preuve du théorème suivant est relativement difficile et demanderait des développements trop importants, nous nous contenterons de son énoncé :

THÉORÈME 39 (des nombres premiers).

$$\lim_{x\to\infty} \pi(x)\frac{\log(x)}{x} = 1.$$

COROLLAIRE 40 (de raréfaction des nombres premiers).

$$\lim_{x\to\infty} \frac{\pi(x)}{x} = 0.$$

On rappelle que la fonction exponentielle $x \mapsto e^x$ est définie par la somme de la série convergente

$$e^x := \sum_{k=0}^{\infty} \frac{x^k}{k!}. \tag{51}$$

Comme on a $e^x/x^n > x/(n+1)!$, la fonction exponentielle croit plus rapidement que n'importe quelle puissance de x. Inversement, la fonction réciproque $\ln x$ tend donc vers l'infini plus lentement que n'importe quelle puissance de x.

Comme la sécurité du protocole repose sur la difficulté supposée de la factorisation, il est évident que Bob a tout intérêt à ne pas choisir des premiers p et q déséquilibrés en taille, c'est-à-dire en nombre de chiffres. En effet si l'un des deux nombres p ou q est trop petit, un simple crible permettra de le détecter et on suppose que la situation qui donnera le plus de fil à retordre à Oscar est celle où $|p| \approx |q|$ (cf. lemma 24 pour le nombre de chiffres d'un entier en base b).

LEMME 41. *Soient a et b deux nombres entiers. Alors*

$$|a|.|b| - 1 \leq |ab| \leq |a|.|b|.$$

Le nombre de chiffres d'un produit est majoré par le produit des nombres de chiffres.

Preuve. En exercice. □

Nous allons montrer que la méthode empirique consistant à calculer le reste de la division euclidienne de n par tous les nombres premiers inférieurs à $\sqrt{n}$ est totalement inopérante et que la connaissance du nombre de chiffres de p ou q n'est pas d'un grand secours. Le nombre $\pi(\sqrt{n})$ de premiers inférieurs à $\sqrt{n}$ est donné par le théorème de raréfaction des nombres premiers :

$$\pi(\sqrt{n}) = \frac{\sqrt{n}}{\ln\sqrt{n}} = \frac{2\sqrt{n}}{\ln n}. \tag{52}$$

Et finalement, sans même tenir compte du coût d'une division euclidienne, on pourraît, à tort, en conclure que l'algorithme empirique a un coût très faible, bien meilleur qu'un algorithme linéaire ! Malheureusement, il ne faut pas perdre de vue que la complexité d'un algorithme s'exprime en fonction de la *taille* des données de l'algorithme, soit $|n|$ le nombre de chiffres de l'entier n. Pour fixer les idées et pour faire des estimations simples, nous choisirons la base 2. Le lemme 24, nous dit que $|n| \approx \log_2 n$, soit $n \approx 2^{|n|}$ que l'on injecte dans (52) :

$$\pi(\sqrt{n}) = \frac{2\sqrt{2^{|n|}}}{\ln 2^{|n|}} = 2^{\frac{|n|}{2} - \log_2 |n| - \log_2(\ln 2) + 1}. \tag{53}$$

Autrement dit, dès que n contient plus d'une centaines de chiffres binaires, le nombre de divisions euclidienne dépasse la capacité de calcul des machines actuelles. Actuellement, RSA est utilisé en pratique avec des entiers n de 2048 bits (chiffres binaires).

La situation ne s'améliore pas si l'on sait que les premiers à tester doivent contenir $|n|/2$ chiffres. Le plus petit nombre entier m à $|n|/2$ chiffres (on suppose n pair pour simplifier le raisonnement) s'écrit $10\ldots0$ en binaire avec une séquence de $|n|/2 - 1$ zéros, soit

$$m = 2^{|n|/2-1}, \tag{54}$$

et le plus grand nombre entier M à $|n|/2$ chiffres sécrit $11\ldots1$ en binaire avec une séquence de $|n|/2$ uns qui est égal à

$$M = 2^{|n|/2} - 1. \tag{55}$$

Le nombre de nombre premiers à $|n|/2$ chiffres binaires est donc

$$\pi(M) - \pi(m) \approx \frac{2^{\frac{|n|}{2}}}{\frac{|n|}{2}\ln 2} - \frac{2^{\frac{|n|}{2}-1}}{\frac{|n|}{2}\ln 2} \tag{56}$$

$$= 2^{\frac{|n|}{2} - \log_2 |n| - \log_2(\ln 2)}. \tag{57}$$

Et on voit bien que l'on ne change absolument l'ordre de grandeur des calculs par rapport à la situation (53). On a gagné que la moitié des calculs, ce qui est faible pour une complexité exponentielle !

On vient donc de montrer que l'approche empirique du crible d'Erathostène pour la factorisation de n n'était pas réaliste, mais on peut légitimement se demander comment Bob fait pour construire les premiers p et q indispensables à la mise en place du protocole ?

8. Algorithmes liés au calcul de la clef : pseudo-primalité,

Comment trouver les entiers premiers p et q pour le protocole RSA, si l'on est incapable dans l'état actuel des connaissances de savoir rapidement si un nombre est premier ou non ? La solution consiste à lâcher du lest sur les certitudes. On cherche des algorithmes qui vont permettre d'affirmer qu'un nombre N est premier avec une certaine probabilité. Dit comme ceci, celà n'a pas grand sens, il faut donc préciser un peu la méthodologie.

On se donne un problème de décision, c'est-à-dire un problème qui attend une réponse positive ou négative. Le problème de la *primalité* est un problème de décision fondamental en théorie de la complexité : l'instance du problème est un entier N, et la question est

"N est-il premier ?". Le problème *complémentaire* (ou *dual*) d'un problème de décision est le problème constitué de la même instance, mais où la question est la négation de la question initiale. Le complémentaire du problème de la primalité est le problème de la *factorisation* : l'instance du problème est un entier N, et la question est "N est-il composé, i.e. existe-t-il deux entiers u et v strictement supérieurs à 1 tels que $N = uv$? Il faut noter qu'il est possible de prouver qu'un entier est premier sans que cela se traduise pour autant par des tentatives de factorisation et symétriquement qu'il est possible de prouver qu'un entier est composé sans pour autant être capable de calculer ses facteurs.

On appelle algorithme de *Monte-Carlo positif* un algorithme qui répond à un problème de décision et dont la réponse est certaine si elle est positive et incertaine si elle est négative. On appelle *probabilité d'erreur* de l'algorithme, la probabilité ϵ que l'algorithme ait répondu négativement à la question alors que la réponse aurait du être positive. On définit de manière totalement symétrique les algorithmes de Monte-Carlo *négatifs*. On peut donc considérer qu'un tel algorithme réalise un test sur un objet (N dans le cas de la primalité) et que l'objet réussi ou non le test en question (si N est premier le test est "réussi"). Si l'on fait passer à l'objet plusieurs tests (l'algorithme dépend d'une quantité choisie au hasard) et que tous s'avèrent positifs, on pourra raisonnablement penser que l'objet a une bonne probabilité de satisfaire la propriété liée au test (la primalité pour N). Nous calculerons évidemment cette probabilité en fonction de la probabilité d'erreur ϵ du test et du nombre de fois où l'objet aura été testé.

Le test de Solovay-Strassen est un algorithme de type Monte-Carlo *négatif* avec une probabilité d'erreur $1/2$ pour le problème de la primalité, et par conséquent de type Monte-Carlo *positif* pour le problème dual de la factorisation. Il est basé sur le calcul du symbole de Jacobi $\left(\frac{a}{N}\right)$. Il nous faut introduire un certain nombre de notations et de définitions.

Soient r et m deux entiers. On dit que r est un résidu quadratique modulo m, s'il existe un entier a tel que $r \equiv a^2 \pmod{m}$. C'est la façon particulièrement pompeuse de dire que $r \mod m$ est un carré dans $\mathbf{Z}/m\mathbf{Z}$! Le produit de deux résidus quadratiques reste évidemment un résidu quadratique. Soit a un entier et p un nombre premier, on définit le *symbole de Legendre* de a et p par

$$\left(\frac{a}{p}\right) := \begin{cases} 0 & \text{si } p \mid a, \\ 1 & \text{si } p \nmid a \text{ et } a \text{ est résidu quadratique modulo } p, \\ -1 & \text{si } p \nmid a \text{ et } a \text{ non-résidu quadratique modulo } p. \end{cases} \tag{58}$$

LEMME 42. *Soit $\mathbf{F}_q$ le corps fini à q élément avec q impair (i.e. le corps n'est pas de caractéristique 2). Alors la moitié des éléments de $\mathbf{F}_q^\times$ sont des carrés. D'autre part,*

$$a^{(q-1)/2} = \begin{cases} 1 & \textit{si } a \textit{ est un carré,} \\ -1 & \textit{sinon.} \end{cases} \tag{59}$$

Preuve. On considère la surjection $f : x \mapsto x^2$ de $\mathbf{F}_q^\times$ dans l'ensemble des carrés $C := \{x^2,\ x \in \mathbf{F}_q^\times\}$. En prouvant que pour tout $y \in C$, il existe exactement deux antécédents à y, il suffira d'appliquer le principe des Bergers pour conclure : si f est une surjection de A

dans B pour laquelle il existe une constante k (ici $k = 2$) telle que $\forall y^i nB$, $|f^{-1}(y)| = k$, alors $\#A = k\#B$. En effet, $a^2 = b^2$ si et seulement si $(a+b)(a-b) = 0$ soit $a = b$ ou $a = -b$. L'opposé d'un nombre a ne peut être a lui-même, en effet $a = -a$ équivaut à $2a = 0$ ce qui n'est pas possible dans un corps de caractéristique différente de 2.
Pour l'autre partie du résultat, on sait que pour tout élément a non-nul, $a^{q-1} = 1$, ce qui entraîne $(a^{(q-1)/2})^2 = 1$. L'équation $X^2 - 1$ admet deux solutions distinctes (pour les mêmes raisons de caractéristique) qui sont ± 1, donc $a^{(q-1)/2} = \pm 1$. Si a est un carré, i.e. $a = x^2$, alors $a^{(q-1)/2} = x^{q-1} = 1$. □

Exercice 14 [$\frac{2}{5}$] Montrez que dans un corps fini de caractéristique 2, tout élément est un carré.

On en déduit le critère d'Euler qui constitue la base du test de pseudo-primalité de Solovay-Strassen :

COROLLAIRE 43 (Critère d'Euler). *Soit p un nombre premier impair. Parmi les $p-1$ éléments de $\mathbf{F}_p^\times$, la moitié sont des carrés, et pour tout entier a, on a*

$$a^{(p-1)/2} \equiv \left(\frac{a}{p}\right) \pmod{p}. \tag{60}$$

On généralise le symbole de Legendre $\left(\frac{n}{m}\right)$ aux entiers m impairs et ≥ 3 en décomposant m en produit de facteurs premiers $m_1 \dots m_r$ et en posant :

$$\left(\frac{n}{m}\right) := \prod_{i=1}^{r} \left(\frac{n}{m_i}\right). \tag{61}$$

Le symbole ainsi défini s'appelle le symbole de *Jacobi*. Par construction, la valeur de ce symbole ne peut prendre d'autres valeurs que celles du symbole de Legendre, $0, \pm 1$.
Le test de Solovay-Strassen est le suivant : on veut tester si un entier N est premier. On tire un entier $1 \leq a < N$ au hasard, on calcule $\left(\frac{a}{N}\right)$ et $a^{(N-1)/2}$. Si $\left(\frac{a}{N}\right) \equiv a^{(N-1)/2} \pmod{N}$, l'algorithme répond N est premier, sinon il répond N est factorisable.

Exercice 15 [$\frac{2}{5}$] Montrer que le test de Solovay-Strassen est un algorithme de Monte-Carlo positif pour le problème de la factorisation.

Ce symbole satisfait les propriétés suivantes : il est bi-multiplicatif :

$$\left(\frac{n}{mm'}\right) = \left(\frac{n}{m}\right)\left(\frac{n}{m'}\right) \quad \text{et} \quad \left(\frac{nn'}{m}\right) = \left(\frac{n}{m}\right)\left(\frac{n'}{m}\right). \tag{62}$$

Exercice 16 [$\frac{2}{5}$] Démontrez les propriétés suivantes du symbole de Jacobi :

(1) Si m est un entier impair et si $a \equiv b \pmod{m}$ alors

$$\left(\frac{a}{m}\right) = \left(\frac{b}{m}\right)$$

(2) Si m est un entier impair alors,

$$\left(\frac{2}{m}\right) = \begin{cases} 1 & \text{si } m \equiv \pm 1 \pmod 8, \\ -1 & \text{si } m \equiv \pm 3 \pmod 8. \end{cases}$$

(3) Si m est un entier impair alors,

$$\left(\frac{ab}{m}\right) = \left(\frac{a}{m}\right)\left(\frac{b}{m}\right).$$

On admettra enfin un dernier résultat très important obtenu par Gauss :

THÉORÈME 44 (Loi de réciprocité quadratique de Gauss). *Soient p et q deux nombres premiers impairs distincts. Alors*

$$\left(\frac{p}{q}\right) = \begin{cases} \left(\frac{q}{p}\right) & \text{si } p \equiv 1 \pmod 4 \text{ ou } q \equiv 1 \pmod 4, \\ -\left(\frac{q}{p}\right) & \text{sinon.} \end{cases} \tag{63}$$

La loi de réciprocité s'etend immédiatement à deux entiers n et m impairs positifs et premiers entre eux par bi-multiplicativité du symbole de Jacobi. On définit deux fonctions signes ϵ et ω, de $1 + 2\mathbf{Z} \to \{\pm 1\}$ par

$$\epsilon(n) := \begin{cases} 1 & \text{si } n \equiv 1 \pmod 4, \\ -1 & \text{si } n \equiv 3 \pmod 4. \end{cases} \qquad \omega(n) := \begin{cases} 1 & \text{si } n \equiv 1, 7 \pmod 8, \\ -1 & \text{si } n \equiv 3, 5 \pmod 8. \end{cases} \tag{64}$$

Le symbole de Jacobi peut se calculer par un algorithme rapide basé sur la loi de réciprocité quadratique :

ALGORITHME 2 : SYMBOLE DE JACOBI

Entrée : deux entiers positifs u et v, v impair.
Sortie : $\left(\frac{u}{v}\right)$.
Règles :

$$u \geq 0 : \quad (u, v) \mapsto (u, v, 1) \tag{65a}$$
$$u < 0 : \quad (u, v) \mapsto (-u, v, \epsilon(v)) \tag{65b}$$
$$u \text{ pair} > 0 : (u, v, \epsilon) \mapsto (u/2, v, \epsilon\omega(v)) \tag{65c}$$
$$u \text{ impair} > 1 : (u, v, \epsilon) \mapsto (v - (v \div u)u, u, \epsilon\theta(u, v)) \tag{65d}$$
$$u = 1 : (u, v, \epsilon) \mapsto \epsilon \tag{65e}$$
$$u = 0 : (u, v, \epsilon) \mapsto 0 \tag{65f}$$

Exercice 17 [$\frac{3}{5}$] Calculez la complexité de cet algorithme en vous basant sur l'étude de la complexité de l'algorithme d'Euclide étendu.

[Développer sur l'algorithme binaire, la preuve etc...Introduire l'algorithme square & multiply et les améliorations (chaînes d'additions, arbres, etc.), preuve etc...la réduction modulaire et son coût, lien avec l'algorithme d'Euclide.]

THÉORÈME 45. *Soit m un entier positif impair. L'ensemble*

$$H = \{a \in (\mathbf{Z}/m\mathbf{Z})^*, \ \left(\frac{a}{m}\right) \equiv a^{(m-1)/2} \pmod m\} \tag{66}$$

est un sous-groupe propre de $\mathbf{Z}/m\mathbf{Z}$.

Preuve . Pour montrer que l'ensemble *fini* H est un groupe, il suffit de montrer qu'il contient l'élément neutre et qu'il est stable (cf. proposition 13). L'entier 1 est toujours un résidu quadratique et $1^{(m-1)/2} \equiv 1 \pmod{m}$, il appartient bien à H. Par multiplicativité du symbole de Jacobi, si $a \in H$ et $b \in H$, $ab \in H$. L'ensemble H est donc un sous-groupe de G.

Reste à montrer que H est un sous-groupe propre, autrement dit qu'il existe au moins un entier $a \in (\mathbf{Z}/N\mathbf{Z})^*$ qui n'appartient pas à H. Pour cela, on distingue deux situations complémentaires :

(1) $m = p_1 \dots p_r$ où les p_i sont des nombres premiers tous distincts ;

(2) $m = p^k q$ avec p premier, $k \geq 2$, q impair et $(p, q) = 1$.

Dans le premier cas, comme m est impair, p_1 est différent de 2 et d'après le lemme 42 la moitié des éléments non-nuls de $\mathbf{F}_{p_1}$ sont des carrés, on peut ainsi se donner u un non-résidu quadratique modulo p_1. Les p_i étant premiers entre eux deux-à-deux, le théorème des restes chinois 30 nous permet d'affirmer qu'il existe un unique entier a qui satisfait le système de congruences :

$$\begin{aligned} a &\equiv u \pmod{p_1} \\ a &\equiv 1 \pmod{p_i}, \quad 2 < i \leq r. \end{aligned}$$

Par multiplicativité du symbole de Jacobi, on en déduit immédiatement que $\left(\frac{a}{m}\right) = -1$. Montrons à présent que $a^{(m-1)/2} \not\equiv 1 \pmod{m}$. Comme $a \equiv 1 \pmod{p_i}$, on a $a^{(m-1)/2} \equiv 1 \pmod{p_i}$ et les p_i étant deux-à-deux premiers entre eux, on en déduit que $a^{(m-1)/2} \equiv 1 \pmod{p_2 \dots p_r}$. Si $a^{(m-1)/2} \equiv -1 \pmod{m}$ alors m divise $a^{(m-1)/2} + 1$ et $p_2 \dots p_r$ également, soit $a^{(m-1)/2} \equiv -1 \pmod{p_2 \dots p_r}$ au lieu de $a^{(m-1)/2} \equiv 1 \pmod{p_2 \dots p_r}$. Donc $a \notin H$.
Dans le second cas, finir...

□

9. Algorithmes liés au chiffrement : l'exponentiation, calcul modulaire, calcul de pgcd

Le protocole RSA impose clairement de disposer d'une bibliothèque de routines arithmétiques pour réaliser les calculs relativement complexes mis en œuvre. Comment calculer une exponentielle ? Par exemple, $123^{1070181}$?

10. Algorithmes liés au déchiffrement : calcul d'un inverse