# Mathematics models for security


Description

**Antoine Puissant**

Enseignant : M. Filiol

2014 - 2015

**Résumé**

Mini-projet à rendre le 30 novembre.

# Table des matières

# 1 Théorie des graphes

## 1.1 Définitions et généralités

**Les *p-graphes* (orientés)** C'est un couple $G(X,U)$. $X$ est l'ensemble des nœuds et $U$ l'ensemble des arcs. $U$ est une famille et non un ensemble. Ainsi on peut avoir plusieurs fois un élément représenté : $U = u_1, u_2, u_2, u_3, \ldots$.
GRAPHE
Pondérer un graphe c'est faire une application C pour laquelle $C : U \rightarrow R$.
Boucle $u \in U|(x_i, x_i)$
— $w^+(x)$ c'est l'ensemble des arcs qui arrivent sur le nœud
— $w^-(x)$ c'est l'ensemble des arcs qui partent du nœud
— $w(x) = w^+(x) \bigcup w^-(x)$
— $|X| = N$
— $|U| = M$

**1-graphes** Ce sont des *p-graphes* avec $p = 1$.

**Densité** La densité d'un graphe orienté est égal au cardinal de $U$ divisé par $N^2$ : $d = \frac{|U|}{N^2} = M$

**Multi-graphes** Ce sont des *p-graphes* sans les « flèches ». Ici, on n'a pas la notion de sens entre les noeuds liés. On a alors le modèle suivant : $G = (X, E)$ où $E$ représente les arêtes.
Ici, la densité est égale à $d = \frac{M}{\frac{N(N-1)}{2}}$.

# 2 Parties de graphes

Un sous-graphe de G c'est la partie engendrée par A soit, pour un graphe $G = (X,U)$ où $A = X$, $SG_A(G) = (A, A^2 \bigcap U)$.
Un graphe partiel de G est un sous graphe ayant le même nombre de noeuds mais on restreint le nombre d'arc. Soit, pour notre graphe $G = (X,U)$, on a $V \subset U$. Ce qui nous donne $SG_V(G) = (X,V)$.

# 3 Parcours et connexité

On a la notion de chemin. C'est une séquence de transition entre des noeuds. Un arc est un chemin de valeur 1. Un chemin fermé est un circuit, un cycle $(u_1, u_2, u_3, \ldots, u_1)$.
Un parcours élémentaire est un cycle, un chemin, ou une chaîne ne passant au plus qu'une fois par un sommet donné.

## 3.1 Fermeture transitive d'un graphe

On a $G = (X, \Gamma)$. $\Gamma : X \rightarrow P(x)$ soit $x \rightarrow y|(x,y) \in U$.
La fermeture transitive de x c'est l'ensemble des points y que je peux joindre en fonction d'une longueur donnée : $lim(x \bigcup \Gamma(x) \bigcup \Gamma(\Gamma(x)))$.

### 3.2 Connexité

Un graphe est dit connexe ssi $\forall (x, y) \in X^2$ il existe un chemin allant de $x$ à $y$. $\forall y, y \in \Gamma(x)$.
GRAPHE 3

### 3.3 Point d'articulation (Cut point, articulation point)

C'est un sommet qui augmente le nombre de composante connexes si on l'enlève.

### 3.4 Un isthme

C'est un arc qui augmente le nombre de composantes connexes si on l'enlève.

### 3.5 Un graphe fortement connexe

Pour un graphe orienté $G(X, U)$ est dit fortement connexe ssi $\forall x, y; (x, y) \in Uet(y, x) \in U$.

## 4 Parcours eulériens et hamiltoniens

Ces problèmes ont été rendus célèbres par les problèmes de tourmé ou du voyageur de commerce, etc...
Un graphe dual c'est un graphe dans lequel on échange les arcs et les noeuds.

### 4.1 Parcours eulériens

Un parcours dans un graphe est eulérien ssi il passe une et une seule fois par toutes les arêtes de G.
GRAPHE
Ce graphe est un parcours eulérien. En effet, il est possible de passer par les noeuds une seule fois.

**Théorème 1.** *Théorème d'Euler*
*Un multigraphe admet un parcours eulérien si et seulement si :*
- *il est connexe*
- *il possède 0 ou 2 sommets de degré impair.*
  - *S'il a 0 sommet de degré impair, le parcours est un cycle*
  - *S'il a 2 sommets de degré impair, le parcours est obligatoirement une chaîne reliant ces deux points.*

### 4.2 Parcours hamiltonien

Un parcours $\mu$ est hamiltonien s'il passe une et une seule fois par tous les sommets de G.
Ce parcours est connu grâce au problème du voyageur de commerce (PVC) [1].

1. Travelling Sales Man (TSM)

# 5 Graphes particuliers

## 5.1 Graphes symétriques

Si $(x, y) \in U \Leftrightarrow (y, x) \in U$

## 5.2 Graphes antisymétriques

## 5.3 Graphes complets

$\forall x \in X, \forall y \in X, (x, y) \in U$

**Théorème 2.** *La clique*
*La clique d'un graphe G est tout sous-graphe complet.*

## 5.4 Graphes bipartites / multipartites

$G = (X_1, X_2, \mu)$
$X_1 \bigcup X_2 = X$
$\forall (x, y) \in U, X \in X_1 et Y \in X_2$
GRAPHE
Dans le cas des graphes bipartites $K_{n_1,n2}$. Un graphe $K_{2,3}$ est un graphe dont les partites ont des point qui sont touts liés.

## 5.5 Graphes planaires

G est planaire ssi on peut le dessiner sans croisement d'arêtes.
Tous les graphes tels que $|X| < 5$ sont planaires.

**Exercice 1.** *Montrer que $K_{2,3}$ est planaire.*

CORRECTION = GRAPH
$K_5$ est le plus petit graphe complet non planaire.
$K_{3,3}$ est le plus petit graphe complet k-partite non planaire.

**Théorème 3.** *Théorème de Kuratowski*
*$G = (X, E)$ et planaire*
*Ssi il ne contient aucun sous-graphe réductibles à $K_5$ ou $K_{3,3}$.*
*Un sous-graphe $G_1$ est réductible à un sous-graphe $G_2$ si on peut rendre $G_1$ égal à $G_2$ en réduisant les noeuds et en contractant chaque sommet de degré 2.*

## 5.6 Les arbres

Ce sont des graphes connexe et sans cycle. Cela implique que $M = N - 1$.
GRAPHE
Pour un arbre, il suffit d'enlever une arête pour augmente le nombre de composantes connexes.
Une arborescence, c'est un arbre orienté.
GRAPHE

### 5.6.1 Les anti-arborescences

Une anti-arborescence est un arbre dans lequel le sens des arcs a été inversé.

### 5.6.2 Les arbres recouvrants

On a : $G = (X, E)$ et $T$, un graphe partiel connexe de G.
$T$ est un arbre connexe recouvrant [2] de $G$. On retrouve ces problèmes dans les arbres généalogiques, dans les hiérarchies, dans les réseaux, etc...

2. Spanning tree

# 6 Classes de problèmes en théorie des graphes

## 6.1 Introduction et rappels

Pour résoudre un problème, il faut savoir combien de temps la résolution va durer. On va alors calculer la complexité d'un algorithme (et non d'un programme).
On va retrouver deux grands types de problèmes :

**Les problèmes ≪ faciles ≫** Ce sont des problèmes que l'on va pouvoir résoudre dans une durée finie. Le nombre $f(n)$ d'opération est un polynôme (de manière générale). Par exemple, pour de la multiplication matricielle, on a $f(n) = O(n^3)$.
On se place ici dans les problèmes de classe P.

**les problèmes ≪ difficiles ≫** Ce sont des problèmes où le $f(n)$ est exponentiel ($Z^n$).
On se place ici dans des problèmes NP (Non déterministe Polynomial). Cela veut dire que l'on va pouvoir résoudre le problème en temps humain mais on va pouvoir vérifier la solution en temps fini (pour une solution donnée).

## 6.2 POC et PE

Un POC[3] est un problème dans lequel on chercher à trouver la valeur minimale $s^*$ d'une application $f$ ($f(x) \in \mathbb{N} ou \mathbb{R}$) sur un ensemble fini $S$ :

$$f(s^*) = \min_{s \in S} f(S) \tag{1}$$

$f$ est la fonction économique ou fonction de coût.
Un PE[4] est un problème dans lequel $f : S \to 0, 1$. On cherche s'il existe un élément $s \in S$ tel que $s$ satisfasse une propriété P. On peut dire, trivialement, que la réponse de ce problème est ≪ oui ≫ ou ≪ non ≫.
On peut retrouver dans cet ensemble les problèmes de décisions.
Ici, on peut avoir :

$$f(S) = 0 \Leftrightarrow s\ verifie\ P \tag{2}$$

Cela revient à calculer le minimum de $f(s)$ :

$$\min_{s \in S} f(S) \tag{3}$$

Ainsi, un PE peut être exprimé en tant que POC.
Un POC peut être exprimé en tant que PE si l(on fixe des contraintes.

**Exercice 2.** *On a $G = (X, U, C)$, 2 sommets $t$ et $s \in G$ $(t \neq s)$.*
*But : Trouver un chemin de coût minimal de $s$ à $t$.*
$f : S \to \mathbb{R}$
*chemin $\mu \Leftrightarrow f(\mu) = \sum_{(ij) \in \mu} C_{ij}$*

3. Problème d'Optimisation Combinatoire
4. Problème d'Existence

## 6.3 Classes de problèmes de graphes faciles

### 6.3.1 Les problèmes d'exploration de graphes

On va retrouver des problèmes de type PE : $G = (X, U), s, t$ de $X$.
Question : existe-t-il un chemin entre $t$ et $s$ ? (PE et si contrainte c'est un POC)
On a ici une complexité en $O(M)$. C'est ainsi en fonction du nombre d'arcs.

### 6.3.2 Chemins de coût optimal

On a un graphe $G = (X, U, C), s, t \in X$.
Question : trouver un chemin optimal de $s$ à $t$.
La résolution générale de ce type de problème est faite par l'algorithme de Bellman en $O(NM)$. C'est ainsi en fonction du nombre de liens et de nœuds.
Si c'est sans circuit, on trouve une complexité en $O(M)$.
Si, $\forall (i, j) \in Ui, j > 0$, alors on va utiliser l'algorithme de Dijkstra et on a un complexité de $O(N^2)$.

### 6.3.3 Problèmes de flots

Ici, on a plutôt un POC. Il est souvent représenté par le problème de *Max Flow*.
On a $G = (X, U, C, s, t)$. $C_{i,j}$ désigne une capacité.
But : maximiser, optimiser le débit du flot pouvant s'écouler de $s$ à $t$ avec la contrainte que $\varphi_{ij} \leqslant C_{ij}$.
On va alors utiliser l'algorithme de Ford-Fulkerson avec un complexité de $O(N.M^2)$.

### 6.3.4 Arbres recouvrants

On a ici un POC souvent représenté par le problème du *Maximum spanning tree*.
On a $G = (X, E, W)$.
But : trouver un arbre recouvrant de poids minimal.
On va utiliser l'algorithme de Prim en $O(N^2)$ ou l'algorithme de Kruskal en $O(M.log_2 N)$.

### 6.3.5 Problèmes de couplage

On se place ici sur un POC.
On a $G = (X, E)$. $C \subset E$ est un couplage si $H$ couple d'arêtes de $C$ n'ont aucun nœuds en communs.
GRAPHE

### 6.3.6 Problèmes eulériens et chinois (POC)

On a $G = (X, E, W)$ ou $G = (X, U, C)$.
On utilise des algorithmes en $O(M)$.
Pour les problèmes chinois, c'est une dégradation. On doit trouver un passage d'au moins une fois par tous les arcs pour un coût minimal.

### 6.3.7 Tests de planarité (PE)

On veut savoir si le graphe est planaire ou pas. On va beaucoup s'en servir dans les réseaux électriques.
On va utiliser l'algorithme d'Hopcrof et Tarjon qui a une complexité en $O(N)$.

### 6.3.8 Tests de bipartisme

On est ici en $O(M)$.

## 6.4 Problèmes difficiles de la théorie des graphes

### 6.4.1 Stable maximal (undependant sct

On a $G = (X, E)$, $S \subset X$ $\forall x, y, \in S, (x, y)! \in E$.
GRAPHE

### 6.4.2 Transversal minimal (POC)

On va retrouver cela dans le problème du vertex cover.
On a $G = (X, E)$, $T \subset X$, $\forall x \in G$, $e = (i, j)$ avec $i \in T$ ou $j \in T$.
GRAPHE

### 6.4.3 La clique maximale (POC)

On a $G = (X, E)$. On dit que $Q \subset X$ est une clique si $\forall (i, j) \in X^2, (i, j) \in E$.
Un stable maximal dans $G$ équivaut à une clique dans $G^2$

### 6.4.4 Problèmes de coloration maximale (POC)

Problème des quatre couleurs.
$G = (X, E)$ est k-colorables si avec $k$ couleurs distinctes on puisse colorier tous les nœuds de telle façon que deux nœuds adjacents soient de couleur différente.
On appelle le nombre chromatique d'un graphe $\min_G(k) = s$.

# 7 Vision algébrique de la théorie des graphes

## 7.1 Matrices d'adjacences

On a $G = (X, E)$ un graphe simple avec $|X| = n$.
La matrice d'adjacence de $G$ est un matrice de taille $n$. On a $A = [a_{ij}]_{i,j \leqslant n}$.

$$a_{ij} = \begin{cases} 1 \ si \ (i,j) \in E \\ 0 \ sinon \end{cases} \tag{4}$$

## 7.2 Spectre de graphes

Spectre de graphe $G = (X, E)$ est noté de la matière suivante :

$$Spec \ G = \begin{pmatrix} \lambda_0 & \lambda_1 & \dots & \lambda_{s-1} \\ m(\lambda_0) & m(\lambda_1) & \dots & m(\lambda_{s-1}) \end{pmatrix} \tag{5}$$

## 7.3 Polynôme caractéristique

...
Propriété :

- $C_1 = 0$
- $-C_2 =$ nombre d'arêtes de G
- $-C_3 =$ nombre de triangles dans G $\Leftrightarrow K_3$

Proposition : Le nombre de chaînes/cycles de taille $l$ dans $G$ joignant $x_i$ et $x_j$ est la valeur des $a_{ij}$ de $A^l$.
Proposition : Soit $G$ un graphe connexe de matrice d'adjacence $A$ et de diamètre $d$. Alors, la dimension de $A(G)$ est au moins égale à $d + 1$.
Corollaire : Soit un graphe $G$ connexe tel que $|x| = n$ et de diamètre $d$. On a $v$ a valeur propre avec $d + 1 \leqslant v \leqslant n$

# 8 Fonction booléennes

C'est :

$$\left\{\{0;1\}^2, \wedge, \vee, -\right\} Def : \mathbb{F}_2^n \to \mathbb{F}_2 \tag{6}$$

On a ici les opérateurs $\ll$ ou $\gg$, $\ll$ et $\gg$ et la complémentation.
On a ici un corps fini [5].
Table de vérité c'est un vecteur.
On a $\vee_{\alpha \in \mathbb{F}_2^n} x^\alpha$.
si $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_n)$, alors $x^\alpha = x_1^\alpha . x_2^{\alpha 2} . x_3^{\alpha 3} \ldots . x_n^{\alpha n}$
CFN : Conjonctive normal form.
$\wedge(\vee_{\alpha i}\ x_i^{\alpha i})$
$(x_1 \vee \overline{x_2} \vee x_3)(x_2 \vee \overline{x_1} \vee \overline{x_3})$
Problème 3 sat [6]. Résolvable par les sat solveur.
Une formule conjonctuve n'est pas normale. Les formes dijonctives normales ne sont pas unique. Ce phénomène est génant.
On va alors utiliser la forme algébrique normale.

## 8.1 Forme algébrique normale

Cela s'écrit de la manière suivante : $\bigoplus_{\alpha \in \mathbb{F}_2^n}\ x^\alpha$
On a alors : $x_1.x_2 \oplus x_1.x_3 \oplus x_2.x_3$
Dans ce cas, $x_i^{\alpha i} = x_1$ si $\alpha i = 1$ ou $x_i^{\alpha i} = 1$ ou $x_i = 0$

$f : \mathbb{F}_{\not\varkappa}^{\ltimes} \to \mathbb{F}_{\not\varkappa}$
Le poids de Hamming de f est noté de la manière suivante :

$$\omega t(f) = \left\| \left\{ x \in \mathbb{F}_{\not\varkappa}^{\ltimes}, \right\} \right\| \tag{7}$$

? ? ?
Soit f une fonction associant .... On appelle transformée de Walsh Fourier de f, la fonction $\hat{f}$ la fonction allant de $\mathbb{F}_2^n$ à $\mathbb{C}$ définie par :

$$\hat{f}(u) = \sum (-1)^{f(x)} . (-1)^{<x.u>} \tag{8}$$

Leame : On a $\hat{f} = 2^n.2.ham(f, < x, u >)$
On a vu que $\hat{f}(0) = 4$
$ham(x, y) = \omega t(x \oplus y$
Définition : Toutes les fonctions $\mathbb{F}_2$ linéaires sur $\mathbb{F}_2^n$ sont de la forme $< x, u >, \forall u$, soit :

$$Ex_i.u_i \tag{9}$$

**Théorème 4.**

$$f(x) = 2^n.(-1)^{f(x)}, \forall x \tag{10}$$

*On a une quasi involutivité*

**Exercice 3.** *Toute fonction $\mathbb{F}_2$ linéaire non constante est équilibrée.*

5. Fined field
6. satisfaisant

**Théorème 5.** *Théorème de Parceval.*
*Soit* $f : \mathbb{F}_2^{\ltimes} \to \mathbb{F}_{\not\ltimes}$. *Alors,*

$$\sum_{u \in \mathbb{F}_2^n} (\hat{f}(u))^2 = 2^{2n} \tag{11}$$

Calcul tensoriel = matrices de matrices.
Calcul de la transformée de WF :
Méthode naïve : $f(u) = \sum (-1)^{f(x)-2}$

## 8.2 Transformée de WF rapide : algorithme de Cooley-Tukey

Spectre de fourier pour la fonction tel que .... On calcule la totalité des valeurs du spectre de Fourier en on complexité logarithmique en temps et exponentielle en mémoire.

$$Spectre_{WF}(f) = (0; 4; 4; 0; 4; 0; 0; -4) \tag{12}$$

Application :
$f$ équilibrée : $\hat{f}(000) = 0$

$$P(f(u) =< x, u >) = \frac{1}{2}.\left(1 - \frac{\hat{f}(u)}{2^n}\right) \tag{13}$$

$$P(f(u) =< x, u >) = \frac{1}{2}.\left(1 - \frac{4}{8}\right) \tag{14}$$

$$P(f(u) =< x, u >) = \frac{1}{2}.\left(\frac{3}{2}\right) \tag{15}$$

$$P(f(u) =< x, u >) = \frac{3}{4} \tag{16}$$

**Exercice 4.** *Démonstration d'un corollaire :*

$$HAM(f, < x, u >) + HAM(f, < x, u > \oplus 1) = 2^n \tag{17}$$

# 9 Codes de Reed Muller - Décodage par Fast Fourier Transform (FFT)

On prend comme ensemble de base un sous-espace vectoriel de dimension $dim(E) = 2^n : E = \mathbb{F}_2^{2^n}$. On a ici un code linéaire. Et $A$ est l'ensemble des fonctions $\mathbb{F}_2$ affines.
On pose $f(x)_{x\in\mathbb{F}_2^n} =< u, x > +1$.
Les codes de Reed Muller c'est l'ensemble des fonctions affines. Ainsi, le code de Reed Muller est noté de la manière suivante :

$$R(1,n) = \{f(n), x \in \mathbb{F}_2^n, f(n) =< A, x > +a, A \in \mathbb{F}_2^n, a \in \mathbb{F}_2^n\} \tag{18}$$

## 9.1 Codage

Soit $w$ un mot de code de taille $n+1$ que l'on souhaite transmettre. On le note de la manière suivante : $(w_{n+1}, w_n, \ldots, w_1)$.
On transmet alors $f(.) =< A, x > +a$ où $A = w_{n+1}, \ldots, w_2$ et $a = w_1$. On transmet ensuite la table de vérité de la fonction.

### 9.1.1 Exemple

On veut coder (101). On envoie alors la table de vérité de $< (1,0), x > \oplus 1$.

| $x_2$ | $x_1$ | $< (1,0), w > +1$ |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |

(on fait $+1$ à $x_2$)
Décodage sur le papier.

## 9.2 Transformée de Reed Muller

$TV_f \to ANF(f)$
Soit $f : \mathbb{F}_2^n \to \mathbb{F}_2$.

$$ANF(f) = \oplus A_\alpha . x\alpha \tag{19}$$

Si $x = (x_n, \ldots, x_1)$ et $\alpha = (\alpha_n, \ldots, \alpha_1)$, alors

$$x^\alpha = (x_i^{\alpha_i})_{i\in\mathbb{N}\leqslant n} \tag{20}$$

On en déduit alors la proposition suivante :

**Propriété 1.**

$$a_\alpha = \oplus_{\beta\leqslant\alpha} f(\beta) \tag{21}$$

# 10 Projet

MCS = Minimum Cut Set
Analyse de l'article, élaboration du rapport. Expliquer le concept, résumer.
Choisir une ville de taille moyenne en France (taille équivalente à Laval). On applique alors la méthode donnée.
Mini rapport et code source à rendre le 30 novembre 2015.
Pour les cartes, prendre OpenStreetMap.

# Table des figures