

# Fusion Chapitres

Wilfried Ehounou

1875



# Contents

<b>1</b>	<b>Introduction</b>	<b>5</b>
<b>2</b>	<b>Mesures et Étude de l'Art</b>	<b>7</b>
2.1	État De L'Art . . . . .	7
2.2	Les réseaux de flots et Mesures . . . . .	7
2.2.1	Graphe de flots: Réseau électrique . . . . .	7
2.2.2	Grandeurs physiques et caractéristiques . . . . .	7
2.2.3	Loi de conservation dans un réseau électrique . . . . .	7
2.2.4	Contraintes dans les réseaux électriques . . . . .	7
2.3	Problématique . . . . .	7
2.3.1	Problème: Line-Distance . . . . .	7
<b>3</b>	<b>Corrélation de mesures</b>	<b>9</b>
3.1	Mesures: des séries temporelles . . . . .	10
3.1.1	Séries Temporelles: Définitions et Propriétés . . . . .	10
3.1.2	Particularités des mesures . . . . .	10
3.2	Étude de l'art sur la corrélation de séries temporelles . . . . .	10
3.2.1	Corrélation sur les séries entières: {W,D}DTW, TWE, MSM, CID, DTDe, COTE . . . . .	10
3.2.2	Corrélation par intervalles: TSF, TSBF, LPS . . . . .	10
3.2.3	Corrélation par parties significatives(shapelets): FS, ST, LS . . . . .	10
3.2.4	Corrélation par agrégation des features: BOP, SAXVSM, BOSS, DTW <sub>F</sub> . . . . .	10
3.3	Proposition d'une méthode de corrélation de mesures électriques . . . . .	10
3.4	Formalisation et calcul de la Matrice de corrélation . . . . .	10
<b>4</b>	<b>Matrice de Correlation : un Line graphe</b>	<b>11</b>
4.1	État de l'art: les line graphes . . . . .	11
4.2	Proposition d'algorithmes . . . . .	11
4.2.1	line couverture . . . . .	11
4.2.2	Determination du graphe de corrélation et sa line couverture . . . . .	11
4.2.3	algorithme de couverture . . . . .	13
4.2.4	algorithme de correction . . . . .	13
4.2.5	Global algorithm behavior . . . . .	15
4.3	Correction particulière de Graphes : Graphes Iourtes . . . . .	15

<b>5 Simulation des algorithmes sur des réseaux théoriques et réelles</b>	<b>17</b>
5.1 Définitions . . . . .	17
5.2 Simulation sur les données théoriques . . . . .	18
5.2.1 Objectifs . . . . .	18
5.2.2 Données: Génération aléatoires de graphes . . . . .	18
5.2.3 Résultats . . . . .	20
5.3 Simulation sur les données réelles . . . . .	23
5.3.1 Réseau électrique: datacenter d'un opérateur de télécommunications . . . . .	23
5.3.2 Grandeurs et Mesures . . . . .	23
5.3.3 Matrice de corrélation . . . . .	23
5.3.4 Application des algorithmes de couverture et de correction . . . . .	23
5.3.5 Discussion sur le réseau proposé . . . . .	23

# **Chapter 1**

## **Introduction**

Il ya de cela 10 ans, on etudie les grpahes dans la longueur bla bla bla



# **Chapter 2**

## **Mesures et Étude de l'Art**

### **2.1 État De L'Art**

### **2.2 Les réseaux de flots et Mesures**

#### **2.2.1 Graphe de flots: Réseau électrique**

Modélisation d'un réseau électrique

#### **2.2.2 Grandeurs physiques et caractéristiques**

#### **2.2.3 Loi de conservation dans un réseau électrique**

#### **2.2.4 Contraintes dans les réseaux électriques**

### **2.3 Problématique**

#### **2.3.1 Problème: Line-Distance**





# Chapter 3

## Corrélation de mesures

### 3.1 Mesures: des séries temporelles

#### 3.1.1 Séries Temporelles: Définitions et Propriétés

#### 3.1.2 Particularités des mesures

### 3.2 Étude de l'art sur la corrélation de séries temporelles

#### 3.2.1 Corrélation sur les séries entières: {W,D}DTW, TWE, MSM, CID, DTDC, COTE

Dynamic Time Warping: weight WDTW, derivative DDTW

Time Warp Edit TWE

Move-split-merge MSM

Complexity invariant Distance CID

Derivative Transform Distance DTDC

Elastic Ensemble EE

Collective of Transform Ensembles COTE

#### 3.2.2 Corrélation par intervalles: TSF, TSBF, LPS

Time Series Forest TSF

Time Series Bag of Features TSBF

Learned Pattern Similarity LPS

#### 3.2.3 Corrélation par parties significatives(shapelets): FS, ST, LS

Fast Shapelets FS

Shapelet Transform ST

Learned Shapelets LS

#### 3.2.4 Corrélation par agrégation des features: BOP, SAXVSM, BOSS,

DTW

# Chapter 4

## Matrice de Correlation : un Line graphe

### 4.1 État de l'art: les line graphes

### 4.2 Proposition d'algorithmes

#### 4.2.1 line couverture

Considérons un réseau électrique modélisé par un DAG et une matrice  $\mu_c$  dans laquelle chaque ligne/colonne est associée à un arc du DAG. Une case de  $\mu_c[i, j]$  contient le niveau de corrélation de mesures entre l'arc  $i$  et l'arc  $j$ , une valeur entre 0 et 1. Idéalement, si ces deux arcs partagent une extrémité en commun, soit l'extrémité initiale ou soit l'extrémité finale comme dans la figure ??, la valeur est proche de 1 sinon elle est proche de 0. Toutefois des erreurs de mesures peuvent apparaître.

À partir de  $\mu_c$  et d'une valeur de seuil choisie  $s \in [0, 1]$ , on considère la matrice  $M$  de même dimension que  $\mu_c$  dans laquelle  $M[i, j] = 1$  ssi  $\mu_c[i, j] \geq s$ , sinon  $M[i, j] = 0$ .

Soit  $G = (V, E)$  le graphe dont  $M$  est la matrice d'adjacence. Idéalement, l'ensemble des liens ayant une extrémité commune forme une clique dans  $G$ .

**Définition 1.** Un graphe  $G = (V, E)$  est un graphe de corrélationssi il est le line graphe d'un DAG. L'ensemble  $C$  est dit une **couverture de corrélation** de  $G$

#### 4.2.2 Determination du graphe de corrélation et sa line couverture

**Définition 2.** Une ambiguïté est un graphe isomorphe à l'un des graphes de la figure 4.1. Le sommet  $X$  est appelé le **point d'ambiguité**.

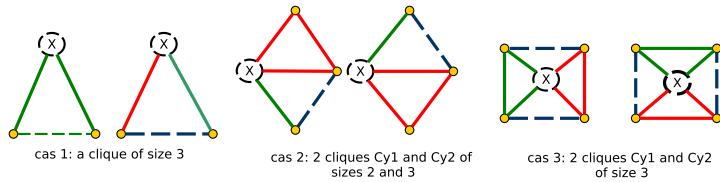


Figure 4.1: ambiguity presentation at point X

**Lemma 1.** Si un graphe de corrélation  $G$  admet deux couvertures de corrélation, alors il existe au moins un sommet  $u$  de  $G$  tel que  $G[\{u\} \cup \Gamma_G(u)]$  est une ambiguïté dont  $u$  est le point.

*Proof.* Consider two correlation covers  $C$  and  $C'$  of  $G$ . Note that it exists at least one vertex  $v$  of  $G$  that is not covered by the same clique(s) in  $C$  and  $C'$ . Let  $c_1$  and  $c_2$  ( $c_2$  can be empty) partitionning  $\{v\} \cup \Gamma_G(v)$  in  $C$ . Let  $c_3$  and  $c_4$  be two cliques (different from  $c_1$  and  $c_2$ ) partitionning  $\{v\} \cup \Gamma_G(v)$ . To each  $i \in \{1, 2\}$  and  $j \in \{3, 4\}$ , we denote by  $c_{i,j}$  the intersection of  $c_i$  and  $c_j$ . Each vertex  $w \in c_{i,j}$  is covered by at most two cliques of  $G$  in  $C$  whose  $c_i$  is one on them. Because  $c_j$  is a clique, the vertex  $v$  is neighbour to all vertices of  $c'_{i',j}$ , for  $i' \neq i$ . The edges between these vertices are in  $c'_i$ , then each edge  $[w,z]$  for all  $z \in c'_{i',j}$  forms a clique matching to a flow network. Then a cardinal of each set  $c_{i,j}$  is equal to 1. We denote by  $v_{i,j}$  the only vertex belonging to  $c_{i,j}$ . It is possible to get  $v_{1,3} = v_{1,4}$  or  $v_{2,3} = v_{2,4}$ . If two equalities are checked in, the vertex  $v$  was covered by one clique of cardinality 3 or not by two cliques. So, the only possible cases are showed by a figure 4.2. The vertex  $v$  is the point of an ambiguity that is isomorphic to  $G[\{u\} \cup \Gamma_G(u)]$ .  $\square$

On peut deduire le corollaire suivant :

**Corollary 2.** *Si un graphe de correlation admet deux couverture de correlation, alors il est isomorphe à un des graphes de la figure 4.2.*

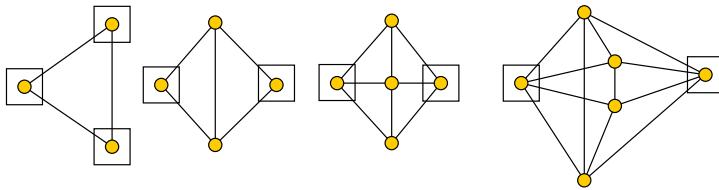


Figure 4.2: possible graphs with two covers and framed ambiguity points

En Effet, si  $G[u \cup \Gamma_G(u)]$  est une ambiguïté, chaque arête, qui n'est pas liée au point d'ambiguïté, doit être une arête d'une et une seule autre ambiguïté de  $G$ . Et chaque sommet d'une ambiguïté, qui n'est pas un point d'ambiguïté, doit appartenir à une et une seule autre ambiguïté de  $G$  dont il n'est pas non plus le point d'ambiguïté. De plus, chaque arête, n'étant pas couvert par les deux configurations de cliques possibles dans une ambiguïté (les arêtes en pointillées dans la figure 4.2), doivent être dans la même situation dans une autre ambiguïté à laquelle elles appartiennent. Ces contraintes font que si un graphe contient une ambiguïté induite, alors il ne peut être que dans un cas de la figure 4.2.

**Définition 3.** *Soit  $G$  un graphe et  $u$  un sommet de  $G$ . Une partition de  $\Gamma_G(v)$  en deux cliques  $C_{u1}, C_{u2}$  est cohérentessi chaque sommet  $v$  de  $C_{u1}$  (resp  $C_{u2}$ ) a au plus un voisin dans  $C_{u2}$  (resp  $C_{u1}$ ).*

Le résultat suivant est un corollaire direct de la preuve du lemme ??

**Lemma 3.** *Soit  $G$  un line graphe,  $u$  un sommet de  $G$  et une partition cohérente de  $u \cup \Gamma_G(v)$  en deux cliques  $C_{u1}, C_{u2}$ . Si l'une de ces deux cliques est de cardinal supérieur ou égal à 4, alors cette partition cohérente est unique.*

Étant donné une partition cohérente  $C_{u1}, C_{u2}$  pour un sommet  $u$  de  $G$ , la fiabilité de cette cohérence est  $F(C_{u1}, C_{u2}) = \min_{[x,y] \in E(C_{u1} \cup E(C_{u2}))} \mu_C([x, y])$

**Définition 4.** *Une line-couverture d'un graphe non orienté connexe  $G$  est un ensemble de cliques maximales de  $G$  telles que chaque sommet de  $G$  appartient à une ou deux de ces cliques et que chaque arête de  $G$  soit couverte par exactement une de ces cliques.*

Il a été montré qu'un graphe  $G$  est un *line graph* ssi il admet une *line-couverture*. Basée sur cette idée et sur des sous-graphes exclus, un algorithme linéaire en fonction du nombre d'arêtes a été proposé pour décider si un graphe est un *line graph* et si oui en fournir la *line-couverture*. Cette algorithme est l'**algorithme de couverture**.

### 4.2.3 algorithme de couverture

### 4.2.4 algorithme de correction

We consider the initial graph  $G$  with vertices  $v$  such that this neighborhood does not form a clique i.e  $cliq(v) = -1$ . We suggest an algorithm which modify the initial set  $E_C$  by adding or deleting edges in order to get a *linegraph*. Given that vertices  $v$  are labeled to  $cliq(v) = -1$  in a given order  $z_1, z_2, \dots, z_t$ . Let  $E_C^i$  be the edge set of  $G$  after the  $(i-1)$ -th vertices had been already treated in that order. In the following chapter, we denote by  $z = z_i$  a vertex and by  $\mathcal{C}^i$  the set of cliques of  $G_C$  at the  $i$ -th step. Thus  $E_C^1 = E_C$  et  $\mathcal{C} = \mathcal{C}^1$ .

Let  $C(z) = \{C_1, \dots, C_k\}$  be a set of all cliques from  $\mathcal{C}^i$  of size at least 3 which  $z$  belongs to. We notice that the intersection of each pair of cliques from  $C(z)$  gives the vertex  $z$ . We define by  $S(z)$  the union of a set of all adjacent vertices  $v$  of  $z$  in the cliques  $K_2 = \{v, z\} \in \mathcal{C}^i$  of size 2 and a set of all adjacent vertices  $v$  of  $z$  such that  $[v, z]$  is not covered by any clique of  $\mathcal{C}^i$ .

$$S(z) = \{v \in \Gamma_G(z) \mid \{z, v\} \in \mathcal{C}^i\} \cup \{v \in \Gamma_G(z) \mid \{z, v\} \notin E(c), \forall c \in \mathcal{C}^i\} \quad (4.1)$$

with  $E(c)$ , an edge set of clique  $c$ .

**Définition 5.** Two cliques  $C$  and  $C'$  from  $\mathcal{C}(z)$  are **contractables** if any edge  $[u, v]$  of  $E_C^i$  such that  $u \in C$  and  $v \in C'$  is not covered by a clique (other than  $\{u, v\}$ ) in  $\mathcal{C}$ .

**Définition 6.** A clique  $C \in \mathcal{C}^i$  is **neighbour** of  $z$  if  $C \notin \mathcal{C}(z)$  and  $card(C \cap S(z)) \geq 2$ . The dependency of a clique  $C$  neighbour of  $z$  is the set  $D_z(C) \subset \mathcal{C}(z)$  such as  $C' \in D_z(C)$  and  $C' \cap C \cap \Gamma_G(z) \neq \emptyset$ . A clique  $C$  is **increasing** if and only if it is neighbour of  $z$  and  $D_z(C)$  is contractable or empty.

$$voisine(z) = \{C \in \mathcal{C}^i \mid C \notin \mathcal{C}(z) \& card(C \cap S(z)) \geq 2\} \quad (4.2)$$

$$D_z(C) = \{C' \in \mathcal{C}(z) \mid C' \cap C \cap \Gamma_G(z) \neq \emptyset\} \quad (4.3)$$

We denote by an **increasement** of  $z$ , an union of a vertex  $\{z\}$ , an increasing clique  $C$  of  $z$  and a contraction of cliques of  $D_z(C)$ .

**Définition 7.** We denote by a **compression** de  $z$ , a triplet  $T$  represented by  $\pi_1, \pi_2, \pi_s$  defined by:

- $\pi_1$  (resp.  $\pi_2$ ) can be one of two forms:

1.  $\pi_1$  (respectively  $\pi_2$ ) is the union of  $\{z\}$ , a subset  $C_1$  (resp.  $C_2$ ) of cliques from  $\mathcal{C}(z)$  such as any pair  $C$  and  $C'$  of  $C_1$  (resp.  $C_2$ ) is contractable and a subset  $S_1$  (resp.  $S_2$ ) of vertices  $v \in S(z)$  not belonging to any clique such as:

$$\forall v \in S_1, \forall x \in C_1, \exists C' \in \mathcal{C} \text{ t.q. } card(C') > 2 \text{ et } \{v, x\} \subset C'$$

2.  $\pi_1$  (resp.  $\pi_2$ ) is an increasement of  $z$ .

- $\pi_1$  and  $\pi_2$  can not be simultaneously reduced to  $\{z\}$  and  $\pi_1 \cap \pi_2 = \{z\}$ .
- $\pi_S = \Gamma_G(z) - ((\pi_1 \cap \Gamma_G(z)) \cup (\pi_2 \cap \Gamma_G(z)))$  such as edge set  $\{[z, v] \in E_C^i : v \in \pi_S\}$  is not disconnecting.
- the triplet  $\pi_1 \cap \Gamma_G(z)$ ,  $\pi_2 \cap \Gamma_G(z)$ ,  $\pi_S \cap \Gamma_G(z)$  is a 3-partition of  $\Gamma_G(z)$ .

Such a compression always consists for example in whether  $\pi_1 = \{z\} \cup C_i \in C(z)$ ,  $\pi_2 = \emptyset$ ,  $\pi_s = \gamma_G(z) - (\gamma_G(z) \cup C_i)$  if  $C(z)$  is not empty. Else,  $\pi_1 = \{z\} \cup \{v \in \gamma_G(z)\}$ ,  $\pi_2 = \emptyset$ ,  $\pi_s = \gamma_G(z) - \{v\}$  is another compression. Another example compression is given in the figure 4.3.

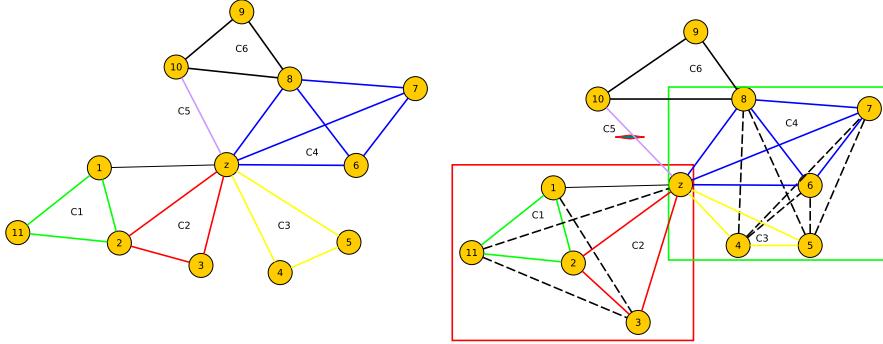


Figure 4.3: an compression example

The  $\pi_1, \pi_2, \pi_S$  compression cost is defined by:

$$c(T) = |\{\{u, v\} \in \pi_1 : [u, v] \notin E_C^i\}| + |\{\{u, v\} \in \pi_2 : [u, v] \notin E_C^i\}| + |\pi_S|$$

Let  $Cost(z)$  be the minimum cost of a correlation cover of  $v$ .

Our goal is to modify  $G_C$  such as  $z$  will be covered by one or two cliques from  $\pi_1$  and  $\pi_2$ . To do this, the cost of this change  $c(T)$  takes into account the edges to add (linked to  $\pi_1$  and  $\pi_2$ ) and to remove (linked to  $\pi_S$ ).

Thus, **applying a compression**  $T = \pi_1, \pi_2, \pi_s$  consists in adding in  $E_0$  the edges defined by the sets  $\{\{u, v\} \in \pi_1 : [u, v] \notin E_C^i\}$  (would be covered by the clique  $\pi_1$ ) and  $\{\{u, v\} \in \pi_2 : [u, v] \notin E_C^i\}$  (would be covered by the clique  $\pi_2$ ), and to delete the edges  $\{[z, v] \in E_C^i : v \in \pi_S\}$ . From then on, vertex  $z$  belongs to two cliques  $\pi_1$  et  $\pi_2$ . We realize the following updates to obtain  $\mathcal{C}^{i+1}$  and  $E_C^{i+1}$ :

- delete all cliques  $\mathcal{C}_z$  covered by  $\pi_1$  in  $\mathcal{C}^i$ .
- delete all cliques of cardinal 2 covered by  $\pi_1$  and  $\pi_2$  in  $\mathcal{C}^i$ ,
- add  $\pi_1$  and  $\pi_2$  in  $\mathcal{C}^i$ , delete from  $E_C^i$  all edges  $\{[z, v] \in E_C^i : v \in \pi_S\}$ .
- assign  $Clq(z)$  to 1 (if  $\pi_1$  or  $\pi_2$  is empty) or 2 (else).

This procedure has the following properties:

**Propriété.** We consider a compression operation. Let  $\mathcal{C}^{i+1}$  the set obtained from  $\mathcal{C}^i$  after a compression.

1. each edge of  $G_C$  covered by one or two cliques in  $\mathcal{C}^i$  remains in  $\mathcal{C}^{i+1}$ .

2. each edge covered by only one clique in  $\mathcal{C}^i$  and not deleted in  $\mathcal{C}^i$  remains in  $\mathcal{C}^{i+1}$ .
3. the vertex  $z$  is covered by one or two cliques in  $\mathcal{C}^{i+1}$  (the number of vertices covered increases by 1 compared to the number in  $\mathcal{C}^i$ ).

Thus, for each vertex  $z_i$  of list  $z_1, z_2, \dots, z_t$  taken in that order, we consider a minimum compression cost  $c_m^i$  and we apply it. The property above ensure at the end of process, we obtain a correlation graph  $G_C^t = (V, E_C^t)$  whose the modified set  $\mathcal{C}$  is a correlation cover. The line-distance verifies  $DL(G_C^0, G_C^t) \leq \sum_{1 \leq i \leq t} c_m^i$ . We notice that during a  $j > 1$  step, the vertex  $z_j$  and its neighborhood is covered by one or two cliques after the processing of  $j - 1$  previous vertices. If no compression is applied to it, we have an identity compression and thus  $c_m^i = 0$ .

#### 4.2.5 Global algorithm behavior

In complexity, the two algorithms process once each vertex in the graph. The complexity of vertex processing is exponential depending of each vertex degree and the cliques to which the vertex belongs, here again depends on the size and the number of cliques. Global algorithm is thus **pseudo-polynomial** depending on the degree of the graph. Given the initial graph, an algorithm running is a way to select vertices in the **covered algorithm** and then the order to take labeled vertices  $z$  such as  $cliq(z) = -1$  in the correction algorithm.

### 4.3 Correction particulière de Graphes : Graphes Iourtes



# Chapter 5

## Simulation des algorithmes sur des réseaux théoriques et réelles

### 5.1 Définitions

**Définition 8.** Une erreur de corrélation est l'existence ou l'absence de corrélation entre deux arêtes (ou arcs) lorsque, respectivement, il n'existe pas de corrélations ou il en existe une.

L'absence de corrélation (corrélation fausse négative) est désignée par 0 dans la matrice d'adjacence tandis que l'existence de corrélation (corrélation fausse positive) a une valeur 1 dans cette matrice.

**Définition 9.** Une corrélation entre arêtes (ou arcs) est l'existence d'un sommet commun aux arêtes (ou arcs). Ce sommet commun peut être source, destination ou intermédiaire comme présenté dans la figure 5.1

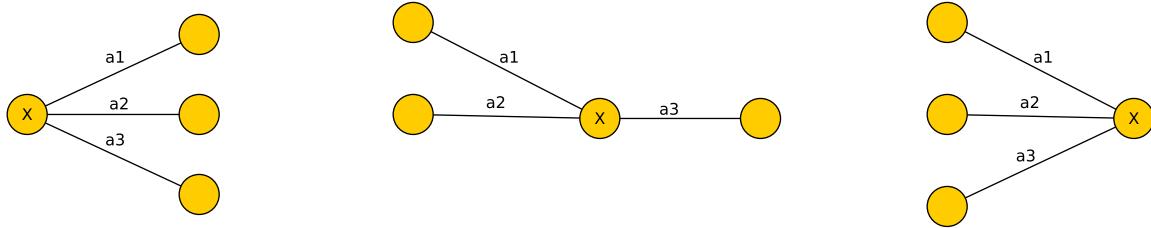


Figure 5.1: De la gauche à la droite: sommet  $X$  source, sommet  $X$  intermédiaire, sommet  $X$  destination

**Définition 10.** métrique: distance de Hamming

La métrique utilisée pour différencier deux graphes est la distance de Hamming. La distance de Hamming est le nombre d'arêtes (ou arcs) différentes entre deux graphes ayant le même ensemble de sommets.

Ainsi, une distance de Hamming égale à 0 signifie que les deux graphes sont identiques. Tandis que une distance de Hamming égale à  $k$  signifie qu'il a  $k$  arêtes différentes entre ces deux graphes.

**Définition 11.** fonction de coût d'un sommet

La fonction de coût d'un sommet est le coût de chaque arête ajoutée ou supprimée lorsqu'on applique l'algorithme de correction sur ce sommet.

Le coût d'une arête peut être

- unitaire : l'ajout et la suppression valent 1.
- normal: la suppression est la probabilité de l'arête et l'ajout vaut 1 moins cette probabilité.
- carré : la suppression est la probabilité au carré de l'arête et l'ajout vaut 1 moins cette probabilité au carré.
- quadratique : la suppression est la probabilité de l'arête à la puissance 4 et l'ajout vaut 1 moins cette probabilité à la puissance 4.

## 5.2 Simulation sur les données théoriques

### 5.2.1 Objectifs

Les travaux réalisées dans cette partie ont pour but de montrer que les algorithmes proposés (couverture et correction) fournissent un graphe non orienté de distance de Hamming minimale lorsque la matrice d'adjacence de ce graphe contient plus de corrélations *fausses positives* que de corrélations *fausses négatives* et peu d'erreurs de corrélations. Le nombre d'erreurs doit être inférieur à 6 pour une probabilité de corrélations fausses positives supérieure à 0.8 . Pour se faire, nous montrons que les sommets, n'appartenant à aucune couverture (sommets  $\in \text{sommets\_1}$ ) doivent être corrigés avec la méthode de **permutation aléatoire minimum** afin d'avoir de meilleurs résultats.

### 5.2.2 Données: Génération aléatoires de graphes

#### Génération de réseaux de flots

La structure de données utilisée, pour le graphe du réseau de flots, est une *matrice d'adjacence*. Cette matrice d'adjacence est une matrice carrée de  $n$  sommets. On définit le nombre de sommets  $n$  et le degré moyen  $\alpha$  du graphe. La probabilité d'existence d'une arête est de  $\text{proba} = \frac{\alpha}{N}$ . Toutefois, si le graphe obtenu à partir de la matrice d'adjacence n'est pas connexe, on choisit aléatoirement un sommet de chaque composante connexe et on ajoute une arête entre ces sommets.

Pour orienter les arêtes, on réalise un tri topologique avec un parcours en largeur *Breath First Search (BFS)* du graphe non orienté généré, à partir de certains sommets choisis comme des sources. Chaque sommet  $x$  a un ordre topologique  $D_x$  et l'arête  $a_{xy}$  devient soit l'arc  $a_{xy}$  si  $D_x < D_y$  soit l'arc  $a_{yx}$  si  $D_x > D_y$ . Le graphe obtenu est alors orienté (un *Directed Acyclic Graph DAG*).

L'ajout des flots sur chaque arc se fait par un parcours en largeur (BFS). On définit les valeurs minimales et maximales des grandeurs physiques. Ces valeurs sont sélectionnées selon le réseau énergétique à modéliser. À titre d'exemple, les valeurs choisies pour des grandeurs électriques sont: les intensités  $I = [150, 200]$ , les tensions  $U = [220, 250]$ , les puissances  $P = [33000, 62500]$ .

On débute par les sommets sources dont on génère une valeur aléatoire comprise dans l'un des intervalles de ces grandeurs. Chaque arc sortant du sommet source reçoit un flot égal à la somme des flots sur les arcs entrants du sommet source multipliée par le facteur  $\epsilon$  (désignant les pertes joules) et divisée par le degré sortant de ce sommet si nous avons comme grandeurs les intensités et les puissances. Dans le cas de grandeurs comme les tensions, le flot de chaque arc sortant est le flot

multipliée par le facteur  $\epsilon$ . On propage les valeurs des grandeurs physiques jusqu'à ce qu'on arrive aux sommets puits. L'application de ces règles de flots permettent de vérifier la *loi de conversation des noeuds*.

### Génération de line graphes sous jacent au réseau de flots non orienté

Nous nommons les arcs du graphe et nous construisons la matrice de corrélation selon la notion de corrélation entre arcs (0 aucun sommet en commun sinon 1). La matrice binaire de corrélation est la *matrice d'adjacence du line graphe* sous jacent au graphe non orienté de notre réseau de flots généré. Cette matrice est symétrique et est notée  $matE$ .

Si la matrice  $matE$  ne contient *aucune erreur de corrélation* alors elle admet une couverture en cliques c'est-à-dire que chaque sommet est contenu dans deux cliques au maximum et chaque arête est couverte par une seule clique. Dans le cas contraire, nous utilisons l'algorithme de correction afin de proposer une couverture pour chaque sommet n'appartenant à aucune clique.

Notre but est de proposer une couverture en cliques des matrices  $matE$  erronées afin que la distance de Hamming entre le line graphe de matrice d'adjacence  $matE$  et le line graphe générée soit minimum.

### Prise en compte de l'erreur de corrélation dans la matrice $matE$

On génère 500 graphes de flots de  $n = 30$  sommets ayant un degré moyen  $\bar{d} = 5$ . On en déduit également 500 line graphes de 150 sommets. On introduit trois paramètres  $k, p, prob$ :

1.  $k$  désigne le nombre de corrélations erronées à ajouter dans la matrice  $matE$ . Dans nos simulations,  $k \in [1, 9]$ .
2.  $p$  désigne le type d'erreurs à réaliser, soit corrélation *fausses positives* (ajout d'arêtes) soit corrélation *fausses négatives* (suppression d'arêtes) soit les deux. Cette variable  $p \in [0, 1]$  varie par pas de 0.1. Par exemple
  - si  $p = 0 \rightarrow$  on supprime uniquement des arêtes dans le line graphe.
  - si  $p = 0.5 \rightarrow$  on ajoute et supprime équiprobablement des arêtes.
  - si  $p = 1.0 \rightarrow$  on ajoute uniquement des arêtes dans le line graphe.
3.  $prob$  désigne la probabilité associée à une corrélation selon le type d'erreurs effectué dans  $matE$ . Ce paramètre est important car les valeurs de corrélations calculées ne sont pas binaires mais probabilistes. Nous en reparlerons dans la sous-section 5.2.3.

Pour ajouter des erreurs à notre matrice  $matE$  de corrélation correcte, on tire aléatoirement  $k$  cases non encore modifiées. Nous mettons chaque case et sa case symétrique à 1 si la probabilité de la case (généré selon la loi uniforme) est inférieure ou égale à  $p$ . Si cette case est déjà à 1, on choisit une autre case.

Nous décidons de modifier  $k \in [1, 9]$  corrélations, selon  $p = 0.5$ , dans la matrice  $matE$  et nous appliquons les algorithmes de couverture et de correction. À la fin de l'algorithme de couverture, s'il existe des sommets du line graphe non couverts par *une ou deux cliques*, on les ajoute à l'ensemble *sommets\_1* et nous appliquons l'algorithme de correction sur chaque sommet de *sommets\_1* selon les méthodes suivantes:

- méthode 1 : degré minimum avec remise.

Elle consiste à sélectionner le sommet de degré minimum dans l'ensemble *sommets\_1*, à appliquer l'algorithme de correction afin de modifier *matE* et enfin à re-exécuter les deux algorithmes sur la matrice *matE* modifiée.

- méthode 2 : coût minimum avec remise.

Elle consiste à sélectionner le sommet de coût minimum dans l'ensemble *sommets\_1*, à appliquer l'algorithme de correction afin de modifier *matE* et enfin à re-exécuter les deux algorithmes sur la matrice *matE* modifiée.

- méthode 3 : coût minimum avec permutation des sommets de *sommets\_1*.

Elle consiste à choisir une permutation dont les sommets sont classés par ordre croissant de leur coût de modification de la matrice *matE* et à appliquer l'algorithme de correction sur cette permutation.

- méthode 4 : degré minimum avec permutation des sommets de *sommets\_1*.

Elle consiste à choisir une permutation dont les sommets sont classés par ordre croissant de leur degré et à appliquer l'algorithme de correction sur cette permutation.

- méthode 5 : permutation aléatoire des sommets de *sommets\_1*.

Elle consiste à choisir aléatoirement  $N$  permutations puis à appliquer l'algorithme de correction et à sélectionner la permutation ayant un coût et une distance de Hamming minimum.

Considérons  $k \in [1, 9]$  le nombre d'erreurs de corrélations dans *matE*,  $\alpha \in [1, 5]$  le nombre de fois qu'on applique  $k$  erreurs sur la matrice *matE<sub>k</sub>*, *LG<sub>k,α</sub>* le line graphe dont on a modifié  $k$  corrélations  $\alpha$  fois et *LG'<sub>k,α</sub>* le line graphe fourni par les algorithmes de couverture et de correction à partir du line graphe *LG<sub>k,α</sub>*.

On note les matrices d'adjacence de *LG<sub>k,α</sub>* et *LG'<sub>k,α</sub>* respectivement *matE<sub>k,α</sub>* et *matE'<sub>k,α</sub>*.

En comparant

1. *LG* et *LG'<sub>k,α</sub>*, on obtient la distance de Hamming notée *DH<sub>k,α</sub>*.
2. *LG<sub>k,α</sub>* et *LG'<sub>k,α</sub>*, on a la distance line notée *DL<sub>k,α</sub>*.

On définit par les variables *moy\_DH* et *moy\_DL*, les moyennes respectives des distances de Hamming (notée *DH<sub>k,α</sub>*) et des distances line (notée *DL<sub>k,α</sub>*) pour une valeur donnée de  $k$  et pour tout  $\alpha \in [1, 5]$ .

$$moy\_DH_k = \sum_{\alpha=1}^5 DH_{k,\alpha} \quad moy\_DL_k = \sum_{\alpha=1}^5 DL_{k,\alpha} \quad (5.1)$$

### 5.2.3 Résultats

Nous décrirons les distributions des distances line et de Hamming moyennées (*moy\_DL* ou *moy\_DH*) pour une méthode de correction (aléatoire) puis nous comparons les cinq méthodes de correction en se basant sur les distances de Hamming moyennées. Nous expliquons le choix de la méthode de *permutation aléatoire* et montrons que les algorithmes (couverture et correction) proposent de meilleurs résultats lorsque la matrice de corrélation possède plus de corrélations *faux positives* que de corrélations *faux négatives* et aussi peu d'erreurs de corrélations ( $k < 6$ ).

Nous présentons également l'impact de la fonction de coût dans les distributions de distances de Hamming.

### Distribution de la méthode de permutation aléatoire

La figure 5.2 représente les histogrammes de la méthode de permutation aléatoire pour une probabilité  $p = 0.5$  et une fonction de coût normal  $F_1$ . À gauche de cette figure, nous avons les distributions des distances lines et à droite celle des distances de Hamming. Ces histogrammes partent d'une corrélation modifiée (en haut de la figure 5.2) à 9 corrélations modifiées (en bas de la figure 5.2).

Dans la représentation d'une distribution, chaque batonnet correspond au pourcentage de graphes associé à une distance moyennée (line ou Hamming). À titre d'exemple, pour  $k = 2$  corrélations modifiées, le pic de l'histogramme est à  $moy\_DL = 2$  pour les distances lines et de  $moy\_DH = 0$  pour les distances de Hamming. Les pourcentages de graphes pour  $moy\_DL = 2$  et  $moy\_DH = 0$  sont identiques c'est-à-dire 50%.

Nous avons, dans les histogrammes, une droite verticale  $y = k$  (en rouge) correspondant à  $k$  erreurs de corrélations. On constate que les pics sont à gauche sur la droite  $y = k$  pour  $k \leq 6$  corrélations modifiées. Cela signifie que le pourcentage de graphe dont  $moy\_DL = k$  et  $moy\_DH = 0$  est le plus élevé (on retrouve en général le line graphe initial) et entraîne une asymétrie de la distribution des distances pour  $k \leq 6$ . En revanche, au-delà de  $k > 6$ , l'algorithme de correction retrouve peu de corrélations modifiées. Ce qui place le pic à droite de la droite  $y = k$  et fournit une distribution gaussienne des distributions.

Le fait que les distributions de distance line et de Hamming soient, toutes les deux, asymétriques (pour  $k \leq 6$ ) soient symétriques (pour  $k > 6$ ) nous interrogent sur l'évolution des distributions de distance line par rapport à celle de Hamming. Pour cela, on calcule leur corrélation par la fonction  $F$  définie en 5.2 et dont la fonction de répartition cumulée est dans la figure 5.3.

$$F_{k,\alpha} = \frac{|moy\_DL_{k,\alpha} - moy\_DH_{k,\alpha}|}{\max(moy\_DL_{k,\alpha}, moy\_DH_{k,\alpha})} \quad (5.2)$$

Si  $moy\_DL$  et  $moy\_DH$  évoluent de manière opposé (l'un est supérieur à 0, l'autre est égal à 0), la fonction  $F$  est égale à 1. Par contre, s'ils sont très corrélés (i.e évoluent identiquement) alors  $F = 0$ . Ainsi lorsqu'elles évoluent dans le même ordre, on obtient la courbe racine carrée. Tel est le cas dans la figure 5.3 pour  $k > 6$ . Les courbes tendent vers la fonction racine carrée quand  $k$  devient grand. Par ailleurs  $F \approx 0$  quand  $k$  est très petit (fig 5.3,  $k \leq 6$ ).

### Comparaison des méthodes de correction

Nous recherchons la meilleure méthode de correction parmi les cinq méthodes énumérées plus haut. Pour se faire, on dispose des distributions de distances line et de Hamming, des histogrammes, des fonctions de répartitions de ces distributions et aussi des moyennes de distances line/Hamming associées à  $k$  corrélations érronées pour chacune des méthodes. Nous utilisons la moyenne des distances de Hamming pour la comparaison de méthodes parce que la distance de Hamming permet d'évaluer la différence entre le graphe de base et celui prédit par nos algorithmes.

Rappelons qu'on a  $p = 0.5$  et la fonction de coût est unitaire ( $F_0$ ). La figure 5.4 affiche les courbes des différentes méthodes pour des distances de Hamming moyennées en fonction des  $k$  erreurs de corrélations.

Nous constatons que la pire des méthodes est celle de degré minimum avec remise (en rouge avec un rond) car elle est au dessus des autres et la meilleure est celle de *de permutation aléatoire* (en jaune avec un triangle) car elle propose des line graphs ayant le nombre minimum d'arêtes

différentes pour  $\forall k$ .

Nous retenons, pour la suite, la méthode de permutation aléatoire comme méthode de correction des sommets n'appartenant à aucune couverture (sommets  $\in$  sommets\\_1).

### Influence des erreurs de corrélations sur les distributions

Faisons varier la variable  $p \in [0, 1]$  par pas de 0.1 dans le but de visualiser l'impact de corrélations *fausses positives* et *fausses négatives* dans l'exécution des algorithmes. Rappelons que l'ajout et la suppression d'arêtes ont le même coût de traitement c'est-à-dire 1. La figure 5.5 résume l'évolution des types d'erreurs de corrélations ( $p$ ) pour des distances de Hamming *moy\_DH* en fonction de  $k \in [1, 9]$  erreurs de corrélations.

Nous constatons que les algorithmes donnent de meilleurs résultats pour  $p = 1$  et de mauvais résultats pour  $p = 0$ . En d'autres termes, lorsqu'on ajoute que des corrélations *fausses positives* i.e  $p = 1$  dans la matrice *matE*, les algorithmes proposent, dans la majorité des cas, un line graphe dont ces arêtes sont supprimées. Cela s'illustre dans la figure 5.6 où l'ajout de 5 corrélations *fausses positives* influencent très peu les line graphs proposés puisqu'ils sont identiques aux line graphs initiales dans 45% des cas. En revanche, ce pourcentage baisse avec beaucoup de corrélations *fausses positives*. Tel est le cas pour  $k = 9$  corrélations où le taux est de 24%.

Par ailleurs, le mauvais résultat obtenu pour des probabilités  $p < 0.5$  s'explique par la correction de très peu de corrélations (inférieure à 1/3 des corrélations modifiées) entraînant l'ajout d'arêtes au voisinage des sommets à traiter (sommets  $\in$  sommets\\_1).

Nous pensons que le meilleur compromis est la probabilité  $p = 0.8$  parce que, pour peu de corrélations modifiées ( $k < 5$ ), les lines graphes produits et générés diffèrent de  $k < 5$  arêtes correspondant aux  $k$  corrélations effectuées et au-delà  $k \geq 5$ , le nombre d'arêtes différentes est fonction du nombre de corrélations faites multiplié par 1.5. Cela signifie qu'il faut, dans la matrice de corrélation, 20% de corrélations *fausses negatives* et 80% de corrélation *fausses positives*.

Que se passe-t-il si on priorise l'ajout de corrélations *fausses negatives* à chaque traitement c'est-à-dire la suppression d'arêtes. Dans ce cas, l'ajout d'arêtes *fausses positives* a un coût moins important que celui des *fausses negatives*.

### Impact de la fonction de coût sur les distributions

Nous définissons quatre fonctions de coût: unitaire ( $n=0$ ), normal ( $n=1$ ), carré ( $n=2$ ), quadratique ( $n = 4$ ) selon l'expression suivante

$$F_n = \begin{cases} \text{prob}[(a_i, a_j)]^n \\ (1 - \text{prob}[(a_i, a_j)])^n \end{cases} \quad (5.3)$$

ou  $\text{prob}[(a_i, a_j)]^n$ , la corrélation entre les arêtes  $a_i$  et  $a_j$ , correspond au coût d'ajout de corrélations fausses négatives,  $(1 - \text{prob}[(a_i, a_j)])^n$  au cout d'ajout de corrélations fausses positives.

Étant donné que nous avons utilisé des matrices binaires dans la génération de line graphs, nous assignons des probabilités pour chaque type de corrélation tel que:

- $\text{prob}[(a_i, a_j)] = [0, 0.5[$  : corrélation vrai négative i.e  $0 \rightarrow 0$
- $\text{prob}[(a_i, a_j)] = [0.5, 0.79[$ : corrélation fausse négative i.e  $1 \rightarrow 0$
- $\text{prob}[(a_i, a_j)] = 0.8$  : corrélation fausse positive i.e  $0 \rightarrow 1$

- $\text{prob}[(a_i, a_j) = ]0.8, 1]$  : corrélation vrai positive i.e  $1 \rightarrow 1$

La figure 5.7 affiche les courbes des différentes fonctions de coût en fonction de  $k$  corrélations modifiées pour  $p = 0.5$ .

Les 4 courbes sont superposées pour  $k \leq 5$  et au-delà de  $k > 5$ , les courbes carrée ( $n=2$ ) et normal ( $n=1$ ) ont les plus petites distances moyennées de Hamming. Pour  $k > 9$ , la courbe unitaire ( $n=0$ ) évolue de manière exponentielle tandis que les courbes carrée ( $n=2$ ) et normal ( $n=1$ ) ont un rapport de 2 entre les distances de Hamming et le nombre  $k$  de corrélations modifiées. On en déduit que la pire fonction de coût est la fonction unitaire ( $n=0$  en jaune).

Toutefois, nous ne pouvons pas choisir la meilleure fonction de coût car les fonctions des courbes *normal* et *carré* évoluent identiquement. On peut conclure qu'utiliser les probabilités de corrélations pour le calcul du coût améliore significativement les distances moyennées de Hamming.

On choisit pour la suite la fonction de cout normal  $F_1$  pour le coût de chaque arête.

## 5.3 Simulation sur les données réelles

- 5.3.1 Réseau électrique: datacenter d'un opérateur de télécommunications**
- 5.3.2 Grandeurs et Mesures**
- 5.3.3 Matrice de corrélation**
- 5.3.4 Application des algorithmes de couverture et de correction**
- 5.3.5 Discussion sur le réseau proposé**

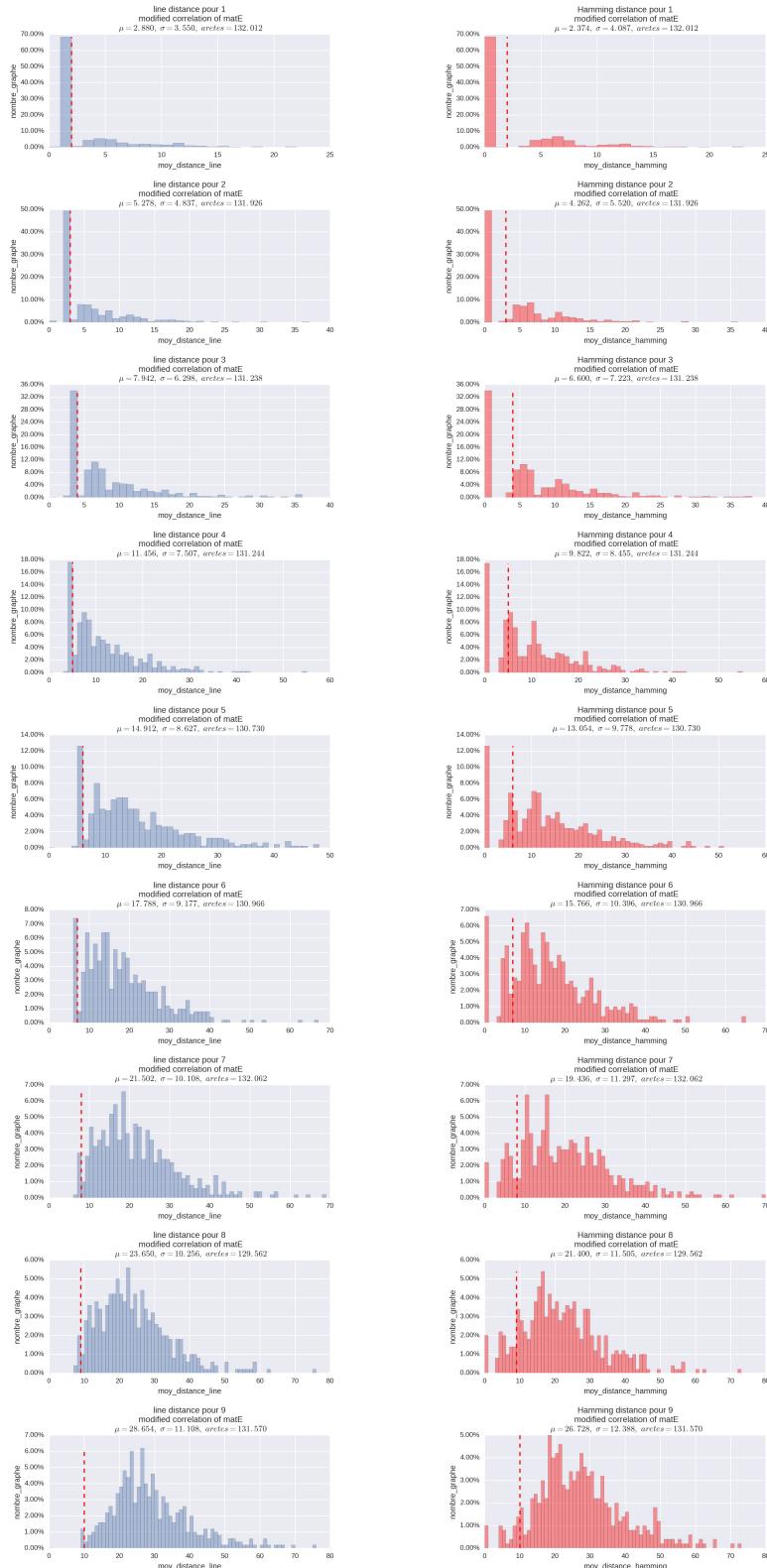


Figure 5.2: Méthode permutation aléatoire avec coût unitaire : distribution des distances line  $moy\_DL$  et de Hamming  $moy\_DH$  pour  $k \in [1, 9]$  corrélations alterées

Figure 5.3: Méthode coût minimum avec remise : fonction de répartition cumulée des distances line  $moy\_DL$  et de Hamming  $moy\_DH$  pour  $k \in [1, \dots, 9]$  de corrélations alterées

Figure 5.4: Comparaison des différentes méthodes de correction de sommets pour  $k \in [1, 9]$  corrélations modifiées

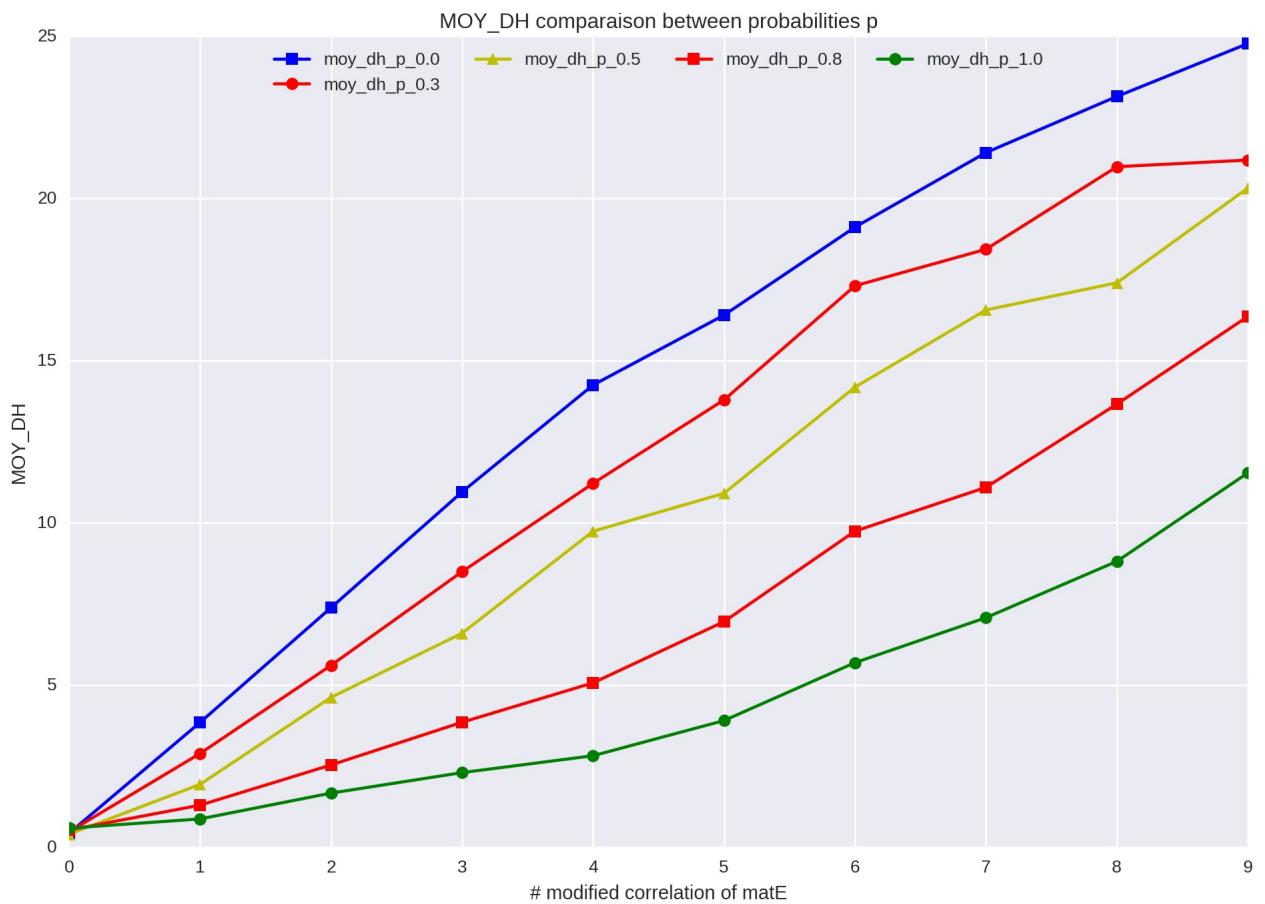


Figure 5.5: Comparaison des différentes probabilités d'ajout  $k \in [1, 9]$  de corrélations fausses positives et fausses négatives sur la méthode de permutation aléatoire

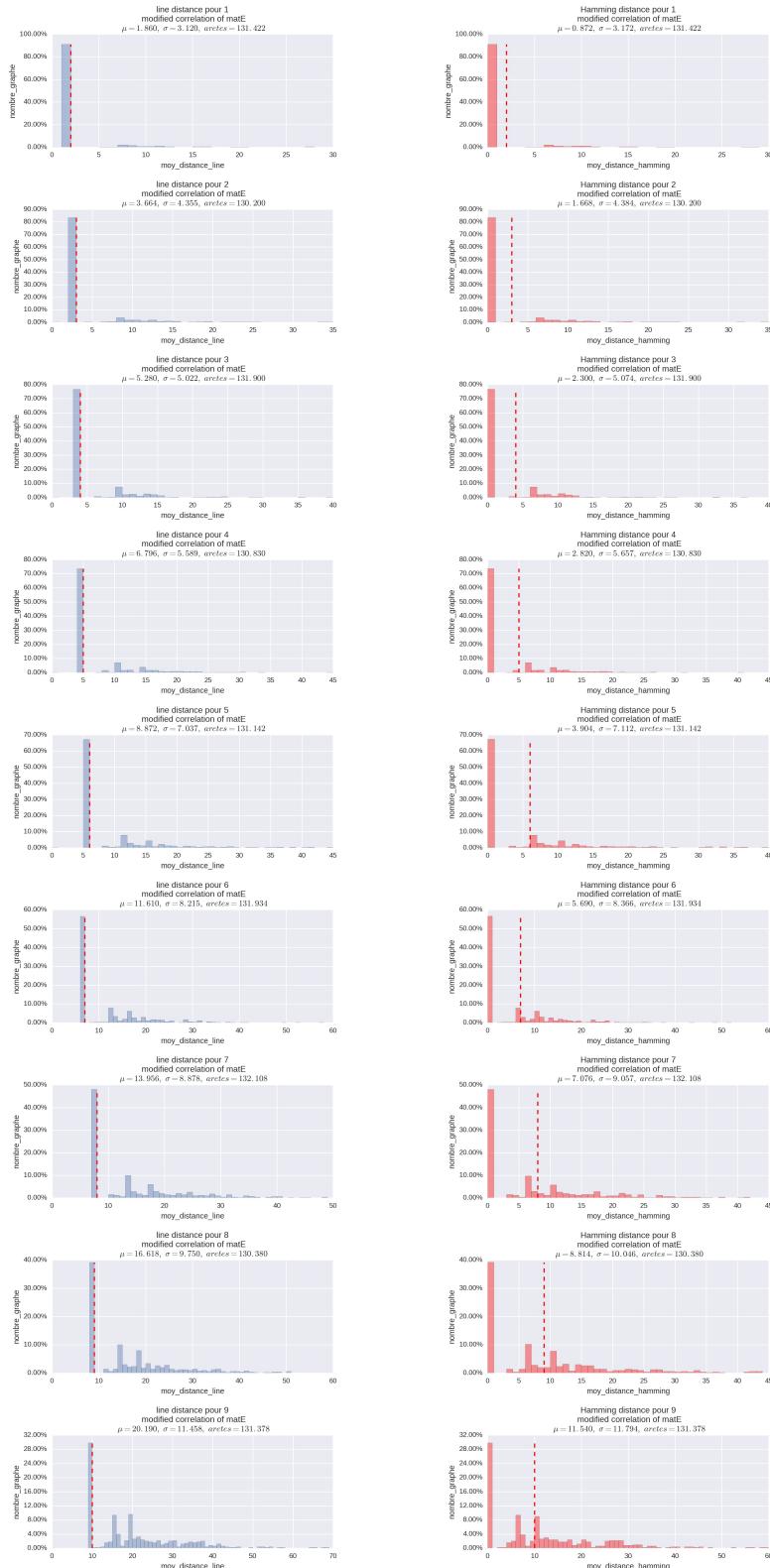


Figure 5.6: distribution des distances line et de Hamming pour  $p = 1.0$  et  $k \in [1, 9]$  corrélations modifiées

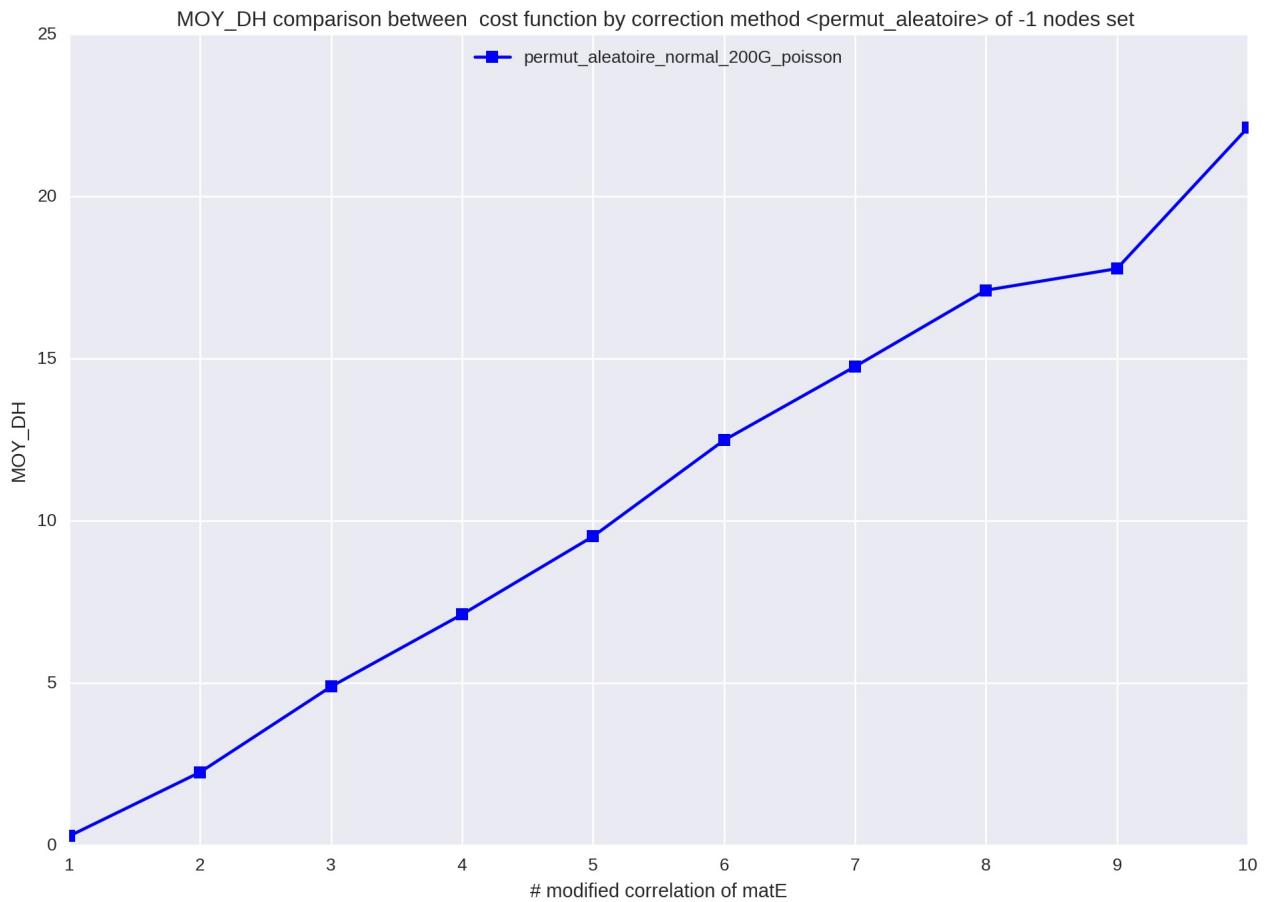


Figure 5.7: Comparaison des différentes fonction de coût sur l'ajout de  $k \in [1, 9]$  de corrélations fausses positives et fausses négatives pour une probabilité  $p = 0.5$  avec la méthode de permutation aléatoire



# Contents