Université Pierre et Marie Curie : Master Ingénierie de la Robotique et Systèmes Intelligents.

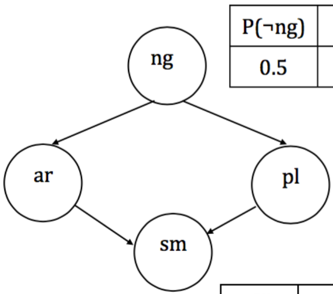
TP N° 2 : Réseaux Bayésiens

1. Calculs préliminaires

Q.1, Q.2 et Q.3 : Feuilles soumises en début de séance.

Introduction : L'objectif de ce TP est de se familiariser avec les probabilités conjointes et conditionnelles, le calcul d'inférence et les réseaux Bayésiens à travers trois scénarios. Pour cela, nous allons utiliser une toolbox « Bayes Toolbox for Matlab » développé par Kevin Murphy : docteur au MIT (Massachussetts Institute of Technology Artificial Intelligence Lab) et professeur adjoint à l'Université de la Colombie-Britannique de Columbia (UBC).

Q4. Lors des calculs préliminaires, nous avons déterminé à laide du raisonnement Bayésien un ensemble de résultats liés à des événements météorologiques (pluie ou ¬ pluie ; nuageux ou ¬ nuageux) ou matériels (arrosage ou ¬ arrosage) ainsi que leurs conséquences sur l'état d'un sol (mouillé ou ¬ mouillé). Désormais, nous souhaitons vérifier ces résultats en simulant les réseaux Bayésiens proposés ci-dessous.



P(¬ng)	P(ng)
0.5	0.5

ng	P(¬ar)	P(ar)
F	0.5	0.5
V	0.9	0.1

ng	P(¬pl)	P(pl)
F	0.8	0.2
V	0.2	0.8

ar	pl	P(¬sm)	P(sm)
F	F	1	0
V	F	0.1	0.9
F	V	0.1	0.9
V	V	0.01	0.99

On note la probabilité de tout les noeuds du graphe (les quatre événements du graphe sont vrais simultanément) :

$$P(ng, ar, pl, sm) = P(ng) \,.\, P(ar \mid ng) \,.\, P(pl \mid ng, ar) \,.\, P(sm \mid ng, ar, pl)$$

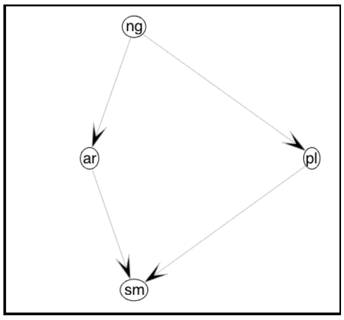
Le raisonnement Bayésien permet de simplifier cette expression en utilisant les relations d'indépendance conditionnelles. Ainsi, la « *pluie* » ne dépendant pas de l'« *arrosage* » sachant « *nuageux* » et « *sol mouillé* » ne dépendant pas de « nuageux » sachant *« arrosage »* et *« pluie »* , la probabilité conjointe peut s'écrire :

$$P(ng, ar, pl, sm) = P(ng) \,.\, P(ar \mid ng) \,.\, P(pl \mid ng) \,.\, P(sm \mid ar, pl)$$

† Indices : ng : « nuageux » ; ar : « arrosage » ; pl : « pluie » ; sm : « sol mouillé ».

<u>*Vérification des résultats de Q1 et Q3 :*</u>

(Implémentation Matlab soumise en annexe 1)




```
P(pl|sm)=
    0.7079

P(ar|sm)=
    0.4298

P(ar|sm,ng)=
    0.1304
```

2. Diagnostics

Contexte : Dans cette partie nous sommes amenés à effectuer l'étude d'un diagnostic fait sur une maladie rare touchant 1% de la population.

Deux variables binaires peuvent être associées aux différents événements :
L'état du patient : (malade ou ¬malade) et Résultat du diagnostic : (positif : T ou négatif : ¬T)

Pour effectuer cette étude nous supposons que :
- Le test indique positif dans 95% des cas où le patient est vraiment malade.
- Le taux de «faux-positifs» du test est de 5%.
- Le taux de «faux-négatifs» du test est de 5%.
- La maladie est assez rare et n'affecte que 1% de la population.

Q5.

M	*P(¬T)*	*P*(T)
FAUX	0.95	0.05
VRAI	0.05	0.95

P(M) « Probabilité d'être malade » : 0.01
P(¬M) « Probabilité de ne pas être malade » : 0.99
P(T|M) « Test positif sachant patient malade » : 0.95
P(¬T|¬M) « Test négatif sachant patient non-malade » : 0.95
P(¬T|M) « Test négatif sachant patient malade » : 0.05
P(T|¬M) « Test positif sachant patient non-malade » : 0.05

Q6. *Probabilité que le patient soit réellement malade en effectuant un test (T1) :*

$$P(M|T1) = \frac{P(T|M) \text{ x } P(M)}{P(T)} = \frac{P(T|M) \text{ x } P(M)}{P(T|M) \text{ x } P(M) + P(T|\neg M) \text{ x } P(\neg M)} = \frac{0.95 \times 0.01}{0.95 \times 0.01 + 0.05 \times 0.99} = 0.16$$

Q7. Nous effectuons sur le patient un second test (T2) qui s'avère être de nouveau positif. En connaissant les résultats des tests (T1 et T2) nous allons déterminer la probabilité que le patient soit malade :

$$P(M|T1,T2) = \frac{P(T2|M) \text{ x } P(M|T1)}{P(T2|M) \text{ x } P(M|T1) + P(T2|\neg M) \text{ x } P(\neg M|T1)}$$

Afin de résoudre l'équation ci-dessus nous devons au préalable déterminer $P(\neg M \mid T1)$:

Mise à jour des paramètres :

$P(M \mid T1) = 0.16$
$P(\neg M \mid T1)$ = À déterminer
$P(T2 \mid M) = 0.95$ « T2 indépendant de T1 »
$P(T2 \mid \neg M) = 0.05$ « T2 indépendant de T1 »

$$P(\neg M|T1) = \frac{P(T1|\neg M) \text{ x } P(\neg M)}{P(\neg M)}$$

$$P(\neg M|T1) = \frac{P(T1|\neg M) \text{ x } P(\neg M)}{P(T1|\neg M) \text{ x } P(\neg M) + P(T1|M) \text{ x } P(M)}$$

$$P(\neg M|T1) = \frac{0.05 \times 0.99}{0.05 \times 0.99 + 0.95 \times 0.01} = 0.83$$

† Mise à jour des paramètres : $P(\neg M \mid T1) = 0.83$

Probabilité que le patient soit malade en connaissant les résultats de deux tests (T1 et T2) positifs :

$$P(M|T1,T2) = \frac{P(T2|M) \text{ x } P(M|T1)}{P(T2|M) \text{ x } P(M|T1) + P(T2|\neg M) \text{ x } P(\neg M|T1)}$$

$$P(M|T1,T2) = \frac{0.95 \times 0.16}{0.95 \times 0.16 + 0.05 \times 0.83}$$

$$P(M|T1,T2) = 0.78$$

Q8. Ici, une implémentation des réseaux Bayésiens correspondant aux deux scénarios précédents sera faite afin de confirmer les résultats des applications (Q.6 et Q.7). Par la suite, nous allons augmenter la probabilité que la population soit touchée par cette maladie P(M) à 50% et créer de nouveaux réseaux afin d'observer le comportement des probabilités liées aux deux scénarios ci-dessus dans un contexte différent (Cf.Plan : page suivante).

† Nous allons d'abord réaliser le calcul des probabilités montrant que le patient est réellement malade en effectuant un test (T1) puis deux tests (T1,T2) dans une situation où 50% de la population est touchée par cette maladie c'est-à-dire : P(M|T1) et P(M|T1,T2) avec P(M) = 0.5.

†† Pour faciliter la lisibilité du rapport, l'implémentation des réseaux Bayésiens sera mise en annexe.

Plan :

† Calcul P(M|T1) avec P(M) = 0.1 — (Effectué - Q.6)
† Calcul P(M|T1,T2) avec P(M) = 0.1 — (Effectué - Q.7)

1. Calcul P(M|T1) avec P(M) = 0.5 — (Suite)
2. Calcul P(M|T1,T2) avec P(M) = 0.5 — (Suite)

3. Confirmation des résultats par l'implémentation des réseaux Bayésiens pour P(M|T1) et P(M|T1,T2) avec P(M) = 0.1
4. Confirmation des résultats par l'implémentation des réseaux Bayésiens pour P(M|T1) et P(M|T1,T2) avec P(M) = 0.5

1. Calcul de la probabilité de P(M|T1) avec P(M) = 0.5 —

Mise à jour des paramètres :

$P(M) = 0.5$
$P(\neg M) = 0.5$
$P(M|T1) = 0.95$
$P(T1|M) = 0.95$
$P(T1|\neg M) = 0.05$

$$P(M|T1) = \frac{P(T1|M) \times P(M)}{P(T1)}$$

$$P(M|T1) = \frac{P(T1|M) \times P(M)}{P(T1|M) \times P(M) + P(T1|\neg M) \times P(\neg M)}$$

$$P(M|T1) = \frac{0.95 \times 0.5}{0.95 \times 0.5 + 0.05 \times 0.5}$$

$$P(M|T1) = 0.95$$

2. Calcul de la probabilité de P(M|T1,T2) avec P(M) = 0.5

$$P(M|T1,T2) = \frac{P(T2|M) \times P(M|T1)}{P(T2|M) \times P(M|T1) + P(T2|\neg M) \times P(\neg M|T1)}$$

Mise à jours des paramètres :
$P(M) = 0.5$
$P(\neg M) = 0.5$
$P(M|T1) = 0.95 = P(M|T2)$
$P(T1|M) = 0.95 = P(T2|M)$
$P(T1|\neg M) = 0.05 = P(T2|\neg M) = 0.05$

P(¬M|T1) : à déterminer
*** Se rapporter au cadrant de droite*

$$P(\neg M|T1) = \frac{P(T1|\neg M) \times P(\neg M)}{P(T1)}$$

$$P(\neg M|T1) = \frac{P(T1|\neg M) \times P(\neg M)}{P(T1|\neg M) \times P(\neg M) + P(T1|M) \times P(M)}$$

$$P(\neg M|T1) = \frac{0.05 \times 0.5}{0.05 \times 0.5 + 0.95 \times 0.5}$$

$$P(\neg M|T1) = 0.05$$

Probabilité que le patient soit malade après deux tests positifs (T1,T2) :

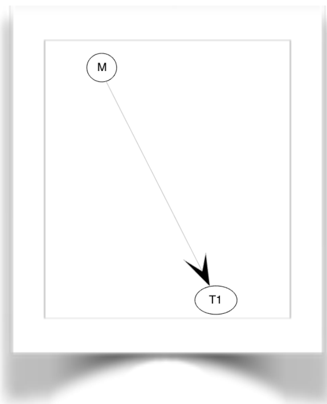
$$P(M|T1,T2) = \frac{P(T2|M) \text{ x } P(M|T1)}{P(T2|M) \text{ x } P(M|T1) + P(T2|\neg M) \text{ x } P(\neg M|T1)}$$

$$P(M|T1,T2) = \frac{0.95 \text{ x } 0.95}{0.95 \times 0.95 + 0.05 \times 0.5}$$

$$P(M|T1,T2) = 0.997238$$

3. Résultats des réseaux Bayésiens venant avérer les calculs de P(M|T1) et P(M|T1,T2) avec P(M) = 0.1 (Implémentation Matlab soumise en annexe 2 et 3)

Réseaux Bayésien de P(M|T1):



Résultat implémentation Matlab :

```
Command Window
P(M|T1)=
    0.1610
```

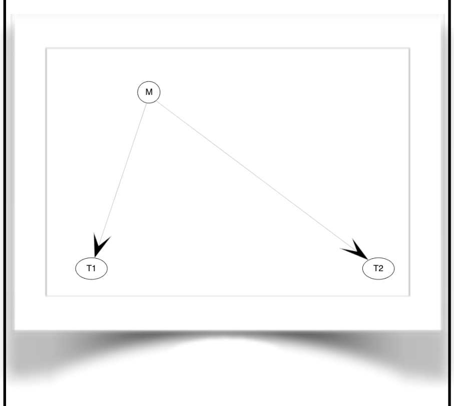
Application numérique Q.6 :

$$P(M|T1) = \frac{P(T|M) \text{ x } P(M)}{P(T)}$$

$$P(M|T1) = \frac{P(T|M) \text{ x } P(M)}{P(T|M) \text{ x } P(M) + P(T|\neg M) \text{ x } P(\neg M)}$$

$$P(M|T1) = \frac{0.95 \times 0.01}{0.95 \times 0.01 + 0.05 \times 0.99} = 0.16$$

Réseaux Bayésien de P(M|T1,T2):



Résultat implémentation Matlab :

```
Command Window
P(M|T1,T2)=
    0.7848
```

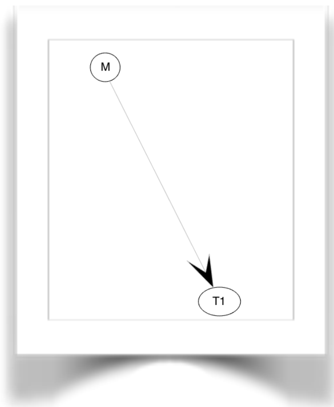
Application numérique Q.7 :

$$P(M|T1,T2) = \frac{P(T2|M) \text{ x } P(M|T1)}{P(T2|M) \text{ x } P(M|T1) + P(T2|\neg M) \text{ x } P(\neg M|T1)}$$

$$P(M|T1,T2) = \frac{0.95 \times 0.16}{0.95 \times 0.16 + 0.05 \times 0.83} = 0.78$$

4. Résultats des réseaux Bayésiens venant avérer les calculs de P(M|T1) et P(M|T1,T2) avec P(M) = 0.5 (Implémentation Matlab soumise en annexe 4 et 5)

Réseaux Bayésien de P(M|T1):



Résultat implémentation Matlab :

```
Command Window
P(M|T)=
    0.9500
```

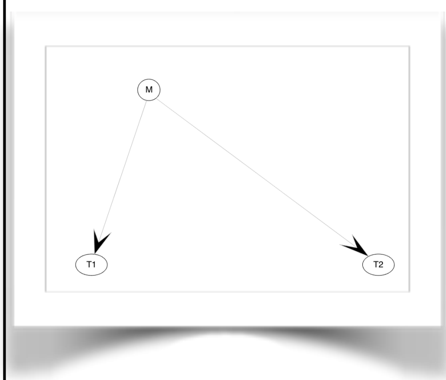
Application numérique Q.8.1 :

$$P(M|T1) = \frac{P(T1|M) \text{ x } P(M)}{P(T1)}$$

$$P(M|T1) = \frac{P(T1|M) \text{ x } P(M)}{P(T1|M) \text{ x } P(M) + P(T1|\neg M) \text{ x } P(\neg M)}$$

$$P(M|T1) = \frac{0.95 \times 0.5}{0.95 \times 0.5 + 0.05 \times 0.5} = 0.95$$

Réseaux Bayésien de P(M|T1,T2):



Résultat implémentation Matlab :

```
Command Window
P(M|T1,T2)=
    0.9972
```

Application numérique Q.8.2 :

$$P(M|T1,T2) = \frac{P(T2|M) \text{ x } P(M|T1)}{P(T2|M) \text{ x } P(M|T1) + P(T2|\neg M) \text{ x } P(\neg M|T1)}$$

$$P(M|T1,T2) = \frac{0.95 \text{ x } 0.95}{0.95 \times 0.95 + 0.05 \times 0.5} = 0.997238$$

Observations : Nous remarquons que si la probabilité que la population soit touchée par la maladie augmente, il suffit d'un test pour être quasiment sûr que le patient soit réellement malade. Dans ce contexte, le second test T2 et la probabilité que la patient soit réellement malade tend vers une valeur certaine - $P(M|T1,T2) \simeq 1$.

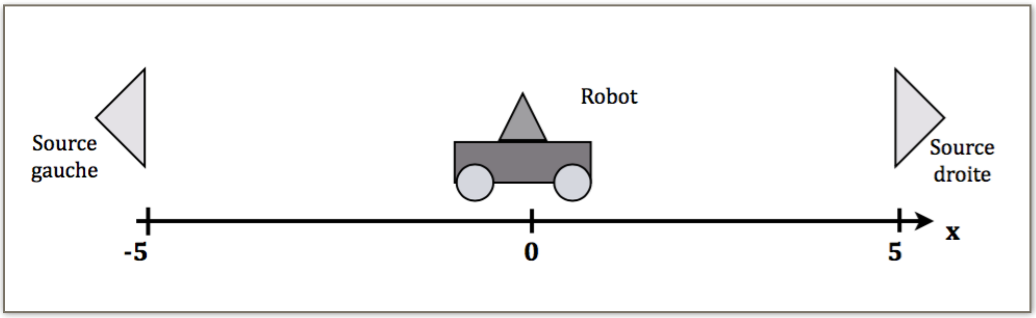
3. Détection d'un son via des réseaux Bayésiens dynamiques

Un réseau Bayésien dynamique ou temporel (DBN pour Dynamic Bayesian Network) est un modèle statistique et stochastique qui étend la notion de réseau bayésien. À la différence de ces derniers, un réseau Bayésien dynamique permet de représenter l'évolution des variables aléatoires en fonction d'une séquence discrète. Un réseau Bayésien est un modèle graphique probabiliste qui, à partir de variables aléatoires structurées en un graphe orienté acyclique, permet de calculer des probabilités conditionnelles liées à ces variables. Les réseaux Bayésiens dynamiques étendent ce processus en prenant en compte l'évolution des variables aléatoires, généralement dans le temps ou en fonction de la position dans notre cas d'étude. En effet, nous allons utiliser les réseaux Bayésiens dynamiques afin de déterminer la décision que devra prendre un robot souhaitant atteindre des sources sonores, l'une sera disposée à sa droite l'autre à sa gauche en se déplaçant sur un axe sachant qu'il se trouve à un endroit initial pouvant varier à chaque itération, donc pour des endroits pouvant être différents étant donnés que le robot se déplace.

Q.9 Probabilités conditionnelles à différentes positions du robot sur l'axe :

Contexte : Nous savons que les capteurs du robot sont peu performants. Ainsi les probabilités d'entendre le son à droite (ED) ou à gauche (¬ED) alors que le son a été émis à droite (D) ou à gauche (¬D) sont affichés dans le tableau ci-dessous :

Représentation graphique du robot dans son environnement :



D	P(¬ED)	P(ED)
F	exp(-Dg/14)	1-exp(-Dg/14)
V	1-exp(-Dd/14)	exp(-Dd/14)

$$P(\neg ED \mid \neg D) = \exp(-Dg/14)$$
$$P(ED \mid \neg D) = 1 - \exp(-Dg/14)$$
$$P(\neg ED \mid D) = 1 - \exp(-Dd/14)$$
$$P(ED \mid D) = \exp(-Dd/14)$$

Pour X =0 :

$$P(\neg ED \mid \neg D) = \exp(-5/14) = 0.699$$
$$P(ED \mid \neg D) = 1 - \exp(-5/14) = 0.3002$$
$$P(\neg ED \mid D) = 1 - \exp(-5/14) = 0.3002$$
$$P(ED \mid D) = \exp(-5/14) = 0.699$$

Le robot se situant au milieu de l'axe : le calcul des probabilités nous montre la cohérence de la prise de décision que pourrait faire celui-ci. En effet, la probabilité d'avoir entendu un son à gauche sachant qu'il a été émis à gauche s'avère plus élevé que la probabilité d'avoir entendu un son à droite sachant qu'il a été émis à gauche. (Ligne 1 et ligne 2)

Pour X = -3 :

$$P(\neg ED \mid \neg D) = \exp(-2/14) = 0.8668$$
$$P(ED \mid \neg D) = 1 - \exp(-2/14) = 0.1331$$
$$P(\neg ED \mid D) = 1 - \exp(-7/14) = 0.3934$$
$$P(ED \mid D) = \exp(-7/14) = 0.606$$

Le robot se situant sur la partie gauche de l'axe à la position X=-3 : Le calcul des probabilités nous montre la cohérence de la prise de décision que pourrait faire celui-ci. En effet, la probabilité d'avoir entendu un son à gauche sachant qu'il a été émis à gauche est dans ce cas, nettement plus élevé que la probabilité d'avoir entendu un son à droite sachant qu'il a été émis à gauche. (Ligne 1 et ligne 2). On note, que l'emplacement du robot sur l'axe à une influence sur les probabilités. Plus le robot sera proche de la source sonore émettante plus sa probabilité de s'y diriger deviendra certaine. C'est d'ailleurs ce que nous allons étudier dans le cas suivant.

Pour X = 10:

$$P(\neg ED \mid \neg D) = \exp(-10/14) = 0.4895$$
$$P(ED \mid \neg D) = 1 - \exp(-10/14) = 0.510$$
$$P(\neg ED \mid D) = 1 - \exp(0/14) = 0$$
$$P(ED \mid D) = \exp(0/14) = 1$$

Le robot se situant sur la partie droite de l'axe à la position X =5, donc à la fin de l'axe : Le calcul des probabilités nous montre une fois de plus la cohérence de la prise de décision que pourrait faire le robot.

Si nous regardons les résultats des deux dernières probabilités : la probabilités d'avoir entendu un son à gauche sachant qu'il a été mis à droite est nettement plus faible que la probabilité d'avoir entendu un son à droite sachant qu'il a été émis à droite (ligne 3 et ligne 4). On note, que l'emplacement du robot sur l'axe à une influence sur les probabilités.

Dans notre cas le robot se trouve sur l'extremum de droite, de ce fait, regardons les deux dernières lignes (ligne 3 et ligne 4) en ayant conscience de l'emplacement de celui-ci.

- Pour la probabilité que le robot entendent un son à gauche sachant qu'il a été émis à droite étant à l'extremum de droite sur l'axe (ligne 3). Dans ce cas, nous remarquons que le robot ne peut se tromper dans sa décision étant donné que $P(\neg ED \mid D) = 0$ cela implique que dans ce contexte le robot n'a aucune chance de se déplacer vers le côté gauche.

- Pour la probabilité que le robot entendent un son à droite sachant qu'il a été émis à droite étant à l'extremum de droite sur l'axe (ligne 4). Dans ce cas, $P(ED \mid D) = 1$ nous montre que dans ce contexte le robot « sait » pertinemment qu'il est se trouve du bon côté de l'axe et a atteint son objectif.

Q.10 Création du réseau Bayésien correspondant au scénario précédent avec une variable X représentant la position du robot (initialement à 0).

† Implémentation Matlab mise en annexe 6, elle regroupera cette question ainsi que les deux suivantes.

Calcul de la probabilité que le son ait été émis à gauche s'il a été entendu à gauche et la probabilité que le son ait été émis à gauche s'il a été entendu à droite :

$$P(\neg D \mid \neg ED) = \frac{P(\neg ED \mid \neg D) \times P(\neg D)}{P(\neg ED)}$$

$$P(\neg D \mid \neg ED) = \frac{P(\neg ED \mid \neg D) \times P(\neg D)}{P(\neg ED \mid \neg D) \times P(\neg D) + P(\neg ED \mid D) \times P(D)}$$

$$P(\neg D \mid \neg ED) = \frac{0.699 \times 0.5}{0.699 \times 0.5 + 0.3002 \times 0.5} = 0.69956$$

$$P(\neg D \mid ED) = \frac{P(ED \mid \neg D) \times P(\neg D)}{P(ED)}$$

$$P(\neg D \mid ED) = \frac{P(ED \mid \neg D) \times P(\neg D)}{P(\neg ED \mid \neg D) \times P(\neg D) + P(ED \mid D) \times P(D)}$$

$$P(\neg D \mid ED) = \frac{0.3002 \times 0.5}{0.3002 \times 0.5 + 0.699 \times 0.5} = 0.30044$$

Q.11 Supposons que la source droite émette un son (D vrai). Le robot a comme objectif de se déplacer vers la source qui a selon lui le plus de chance d'être active. Itérativement, une observation (ED ou ¬ED) est générée et le robot détermine, en accord avec sa position X actuelle, quelle source (D ou ¬D) a le plus probablement émis le son.

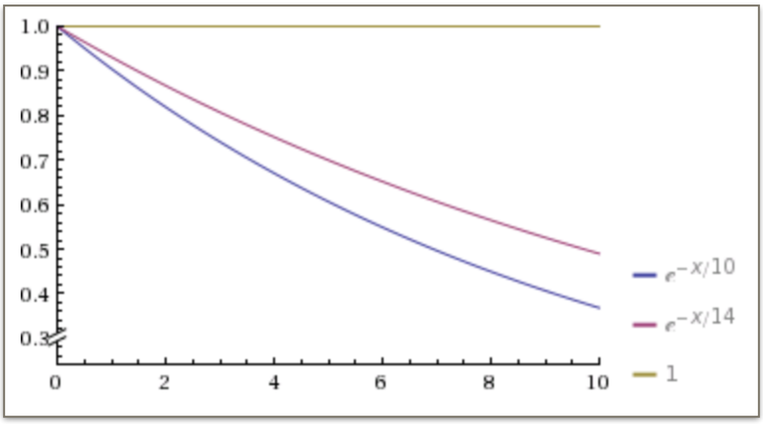
Il se déplace ensuite de ΔX=±1 selon la source la plus probable. Notre objectif étant que le robot atteigne une des deux sources (X=-5 ou X=5) nous allons ajouter une boucle à notre programme permettant au robot de réaliser le but escompté. À chaque itération, nous allons déterminer les probabilités P(¬ED|D) = FAUX et *P(ED|D) = VRAI et observer le comportement de celle-ci.*

Enfin, les probabilités que le son ait été émis à gauche ou à droite respectivement P(¬D|ED) et *P(D|ED) (si le son a été entendu à droite) et P(¬D|¬ED)* et *P(D|¬ED) (si le son a été entendu à gauche) seront déterminé à chaque itération afin que le robot puisse se déplacer dans la direction la plus probable.*

† Afin de faciliter la lisibilité du rapport l'implémentation sera mise en annexe.

Q.12 Le robot dispose d'un capteur modélisé par une expression de la forme *exp(-Dd/14).* Sachant que le robot se déplace selon la réponse de son capteur qui induit les probabilités conditionnelles, modifier l'expression du capteur devrait avoir un impact direct sur les décisions que prendrait le robot. Pour cela, nous allons tracer l'expression du capteur sous différentes formes :

- Expression du capteur inchangée : *exp(-Dd/14) - Courbe rouge*
- Expression du capteur avec un diviseur inférieur *: exp(-Dd/10) - Courbe bleu*
- Expression du capteur avec un diviseur supérieur *: exp(-Dd/Inf) - Courbe jaune « Caca-d'oie »*



Le robot se déplaçant sur une distance sur un intervalle [-5,5] nous représentons le signal sur un intervalle de [0,10].

Les résultats obtenus pour les différentes modifications de l'expression du capteur nous montrent que si le diviseur du capteur diminue le robot aura tendance à prendre des décisions moins certaine que celle qu'il prendrait pour une expression initiale. En revanche, plus le diviseur de l'expression du capteur est élevé plus la décision que prendra le robot sera juste. En effet, nous pouvons voir que dans le cas où le diviseur tend vers l'infini le résultat de l'expression qu'utilise le robot pour calculer ses probabilités conditionnelles est égal à 1 ce qui signifie que lorsque les probabilités d'avoir entendus le son à droite ou à gauche sachant qu'il a été émis à droite ou à gauche seront certaines à chaque itération et le robot se déplacera toujours dans la bonne direction selon que le bruit ait été émis à droite ou gauche de l'axe.

La décision que prend le robot se base uniquement sur la réponse d'un capteur et ne peut être certaine étant donné la pression du capteur, la qualité, les défauts de fabrications etc. Néanmoins, plusieurs techniques peuvent être mises à disposition afin que le robot puisse prendre la bonne décision à chaque itération. Une solution serait d'allumer une lampe à l'endroit où le son aurait été émis et d'allier un capteur sonore plus un capteur lumineux. Dans ce cas, le robot pourrait décider de prendre une décision de déplacement en fonction de la réponse de son capteur sonore et de la réponse de son capteur lumineux venant infirmer ou confirmer une décision que devrait prendre le robot. Une autre solution serait de décupler les capteurs sonores afin que le robot puisse prendre une décision en fonction des différents résultats des capteurs (cette manipulation a été réalisée dans la seconde partie du TP lorsque que nous doublons les tests afin de savoir si le patient était réellement malade ou non).

Conclusion : Lors de ce TP, nous avons pu utiliser les réseaux Bayésiens dans différents contextes, un premier contexte météorologique visant à définir les probabilités conditionnelles en fonction d'événements météorologiques ou matériels définissent la probabilité qu'un sol soit mouillé ou non, un second relatif à une maladie au sein d'une population et un dernier contexte concernant la prise de décision de déplacement qu'effectuerait un robot à travers l'utilisation d'un réseau Bayésien dynamique. Ces trois cas de figure, nous ont permis de voir et prendre conscience de l'utilité des réseaux Bayésiens au quotidien et d'avoir un premier contact avec l'implémentation de ces réseaux à travers Matlab. Pour finir, je tenais à remercier Pr. Coninx pour ses explications des plus utiles et instructives.

† Implémentation Matlab annexe 1

```
clear all, close all, clc

%% Construct Bayesian net
N = 4;
dag = zeros(N,N);
ng = 1; ar = 2; pl = 3; sm = 4;
dag(ng,[pl ar]) = 1;
dag(pl,sm) = 1;
dag(ar,sm)=1;

%% Node size: binary
node_sizes = 2*ones(1,N);

%% Creation of Bayesian net
bnet = mk_bnet(dag, node_sizes);
%% Visualize net
names={'ng','ar','pl','sm'};
draw_graph(bnet.dag,names);

%% Define parameters
bnet.CPD{ng} = tabular_CPD(bnet, ng, [0.5 0.5]);
bnet.CPD{pl} = tabular_CPD(bnet, pl, [0.8 0.2 0.2
0.8]);
bnet.CPD{ar} = tabular_CPD(bnet, ar, [0.5 0.9 0.5
0.1]);
bnet.CPD{sm} = tabular_CPD(bnet, sm, [1 0.1 0.1
0.01 0 0.9 0.9 0.99]);

%% Définition de l'algorithme d'inférence
engine = jtree_inf_engine(bnet);

evidence = cell(1,N);

% Q1 P(pl|sm)
evidence{sm} = 2;

[engine, loglik] = enter_evidence(engine, evidence);

marg = marginal_nodes(engine, pl);

disp('P(pl|sm)=')
disp(marg.T(2))

% Q2 P(ar|sm)
evidence{sm} = 2;
[engine, loglik] = enter_evidence(engine, evidence);
marg = marginal_nodes(engine, ar);
disp('P(ar|sm)=')
disp(marg.T(2))
```

† Implémentation Matlab annexe 2 et 3

Annexe 2 : P(M|T1,T2) avec P(M) = 0.1 :

```
clear all, close all, clc
%% Construct Bayesian net
N = 2;
dag = zeros(N,N);
M=1;T1=2;
dag(M,T1) = 1;


%% Node size: binary
node_sizes = 2*ones(1,N);

%% Creation of Bayesian net
bnet = mk_bnet(dag, node_sizes);

%% Visualize net
names={'M','T1'};
draw_graph(bnet.dag,names);

%% Define paramters
bnet.CPD{M} = tabular_CPD(bnet, M, [0.99 0.01]);
bnet.CPD{T1} = tabular_CPD(bnet, T1, [0.95 0.05
0.05 0.95]);


%% Définition de l'algorithme d'inférence
engine = jtree_inf_engine(bnet);

evidence = cell(1,N);

%  P(M|T)
evidence{T1} = 2;

[engine, loglik] = enter_evidence(engine, evidence);

marg = marginal_nodes(engine, M);

disp('P(M|T)=')
disp(marg.T(2))
```

Annexe 3 : P(M|T1,T2) avec P(M) = 0.1 :

```
clear all, close all, clc
%% Construct Bayesian net
N = 3;
dag = zeros(N,N);
M=1;T1=2;T2=3;
dag(M,T1) = 1;
dag(M,T2)=1;

%% Node size: binary
node_sizes = 2*ones(1,N);

%% Creation of Bayesian net
bnet = mk_bnet(dag, node_sizes);

%% Visualize net
names={'M','T1','T2'};
draw_graph(bnet.dag,names);

%% Define paramters
bnet.CPD{M} = tabular_CPD(bnet, M, [0.99
0.01]);
bnet.CPD{T1} = tabular_CPD(bnet, T1, [0.95
0.05 0.05 0.95]);
bnet.CPD{T2} = tabular_CPD(bnet, T2, [0.95
0.05 0.05 0.95]);

%% Définition de l'algorithme d'inférence
engine = jtree_inf_engine(bnet);

evidence = cell(1,N);

%  P(M|T)
evidence{T1} = 2;
evidence{T2} = 2;
[engine, loglik] = enter_evidence(engine, evidence);

marg = marginal_nodes(engine, M);

disp('P(M|T1,T2)=')
disp(marg.T(2))
```

† Implémentation Matlab annexe 4 et 5

Annexe 4 : P(M|T1,T2) avec P(M) = 0.5 :

```
clear all, close all, clc

%% Construct Bayesian net
N = 2;
dag = zeros(N,N);
M=1;T1=2;
dag(M,T1) = 1;

%% Node size: binary
node_sizes = 2*ones(1,N);

%% Creation of Bayesian net
bnet = mk_bnet(dag, node_sizes);

%% Visualize net
names={'M','T1'};
draw_graph(bnet.dag,names);

%% Define parameters
bnet.CPD{M} = tabular_CPD(bnet, M, [0.5 0.5]);
bnet.CPD{T1} = tabular_CPD(bnet, T1, [0.95
0.05 0.05 0.95]);


%% Définition de l'algorithme d'inférence
engine = jtree_inf_engine(bnet);

evidence = cell(1,N);

%  P(M|T)
evidence{T1} = 2;

[engine, loglik] = enter_evidence(engine,
evidence);

marg = marginal_nodes(engine, M);

disp('P(M|T)=')
disp(marg.T(2))
```

Annexe 5 : P(M|T1,T2) avec P(M) = 0.5 :

```
clear all, close all, clc

%% Construct Bayesian net
N = 3;
dag = zeros(N,N);
M=1;T1=2;T2=3;
dag(M,T1) = 1;
dag(M,T2)=1;

%% Node size: binary
node_sizes = 2*ones(1,N);

%% Creation of Bayesian net
bnet = mk_bnet(dag, node_sizes);

%% Visualize net
names={'M','T1','T2'};
draw_graph(bnet.dag,names);

%% Define paramters
bnet.CPD{M} = tabular_CPD(bnet, M, [0.5 0.5]);
bnet.CPD{T1} = tabular_CPD(bnet, T1, [0.95
0.05 0.05 0.95]);
bnet.CPD{T2} = tabular_CPD(bnet, T2, [0.95
0.05 0.05 0.95]);

%% Définition de l'algorithme d'inférence
engine = jtree_inf_engine(bnet);

evidence = cell(1,N);

%  P(M|T)
evidence{T1} = 2;
evidence{T2} = 2;
[engine, loglik] = enter_evidence(engine,
evidence);

marg = marginal_nodes(engine, M);

disp('P(M|T1,T2)=')
disp(marg.T(2))
```

† Implémentation Matlab annexe 6

```
clear all
clc

% Random position of robot
x=0;

while 1
   %Calculate distance to right
   dd=abs(5-x);

   % Calculate P(ED|D)
   P_ED_D=exp(-dd/14);
   obs=Observation(P_ED_D);

   % Construct Bayesian net according to current info
   [PD,P_not_D]=PCD_Bayes_Net(x,obs);
   disp('P_D')
   disp(PD)
   disp('P not D')
   disp(P_not_D)

   % Bayesian reasoning, based on highest possibility
   if PD < P_not_D
      %Left movement
      disp('Move left')
      x=x-1;
   else
      % Right movement
      disp('Move right')
      x=x+1;
   end
   if x == 5 || x == -5
   break
   end
```