

Aide à la décision et diagnostic par réseaux bayésiens d'un robot mobile commandé en réseau

Amine MECHRAOUI¹, Jean-Marc THIRIET¹, Sylviane GENTIL¹

¹Laboratoire GIPSA-Lab

GIPSA-Lab UMR 5216

BP 46, F-38402 Saint Martin d'Hères Cedex, France.

amine.mechraoui, jean-marc.thiriet, sylviane.gentil@gipsa-lab.grenoble-inp.fr

Résumé— Ce papier traite de la co-conception de systèmes mobiles commandés en réseau. Il faut adapter le réseau de communication et sa qualité de service (QoS) aux besoins de l'application, vue ici comme la qualité de commande ou de diagnostic (QoC). Ce travail prend en compte les contraintes apportées par le réseau sur les algorithmes de commande et diagnostic (contraintes pouvant évoluer dynamiquement), et les manières dont le réseau peut s'adapter dynamiquement aux besoins ponctuels des algorithmes de diagnostic, en fonction des contraintes d'environnement, de l'état du système et de sa criticité. Les réseaux bayésiens (BN) sont utilisés pour la prise de décision et pour le diagnostic. Un robot mobile communicant est traité comme exemple d'architecture dynamique.

Mots-clés— Systèmes commandés en réseau, Réseaux sans fil, Réseaux bayésiens, Diagnostic distribué, Robot mobile, Qualité de Service, Qualité de Contrôle.

I. INTRODUCTION

Les applications de robotique en extérieur se multiplient. Le besoin actuel de développer des véhicules intelligents et multi-tâches a favorisé l'orientation vers les systèmes commandés en réseau sans fil (Wireless Networked Control systems (WNCS)) [1]. La communication entre les robots s'avère capitale pour assurer un bon fonctionnement. Cependant, l'intégration du réseau dans la boucle de commande entraîne des problèmes qui influencent les performances du système [2].

L'objectif principal de cette étude est de trouver un compromis entre la commande et le diagnostic d'une part et le réseau d'autre part. Il faut adapter le réseau aux contraintes dynamiques de la commande et du diagnostic et aussi adapter l'algorithme de commande et de diagnostic aux contraintes dynamiques du réseau ce qui nous amène à résoudre un problème de co-conception. Pour cela, on va définir une architecture de communication entre un véhicule mobile se déplaçant dans un espace 2D et plusieurs stations au sol (réseau infrastructure). Le robot peut communiquer avec la ou les stations pour échanger les informations nécessaires à sa mission. En fonction de la position du mobile et du signal reçu de chaque station, le véhicule pourra être en regard d'une ou plusieurs stations, ou d'aucune. Lorsqu'il ne sera plus en face d'une station il sera complètement autonome. Lorsqu'il sera en face d'une ou plusieurs stations, il doit pouvoir communiquer avec l'une de ces stations en prenant en compte deux critères principaux, à savoir : la Qualité de Service (QoS) et la Qualité de Contrôle (QoC). Nous considérons la QoS comme étant la capacité à fournir un support de communication conforme aux exigences en matière de temps et de bande

passante et aussi de garantir un niveau acceptable de pertes de paquets défini pour un usage particulier. La QoC désigne la capacité de garantir la stabilité, la précision, le temps de réponse, l'allure des transitoires et la robustesse d'un système. La quantification de ces deux critères est liée fortement à l'usage du système, au nombre d'informations échangées et aussi au réseau sans fil utilisé. L'évaluation de la QoS de la station courante peut aider à décider de déclencher ou non la procédure de changement de station courante pour assurer plus de garanties de transmission.

L'aspect stochastique et non déterministe des réseaux sans fil rend difficile de trouver un modèle analytique qui traduit fidèlement son comportement. La théorie des probabilités peut être une bonne solution pour quantifier la QoS. Par ailleurs, la causalité entre les différents paramètres du réseau d'une part et la QoS et la QoC d'autre part pousse à utiliser la théorie des graphes pour modéliser le comportement du réseau. L'outil qui permet le mariage entre la théorie des graphes et la théorie des probabilités est le réseau bayésien (BN : Bayesian Network) [3], [4].

L'article est organisé comme suit. Dans un premier temps, nous allons présenter le modèle du robot ainsi que l'architecture de commande via un réseau sans fil. Puis, l'approche proposée pour qualifier par un BN la QoS du réseau et la QoC du robot sera détaillée. Ensuite, l'utilisation de cette approche pour l'aide à la décision et le diagnostic sera présentée avec quelques scénarios. Enfin, une conclusion achève cet article.

II. ARCHITECTURE DE COMMANDE VIA UN RÉSEAU SANS FIL

Cette section se focalise sur la présentation du système choisi, à savoir, le robot Khepera III (voir Fig. 1) et son modèle cinématique et dynamique. Le robot Khepera III est un véhicule autonome, muni de deux roues motrices indépendantes.

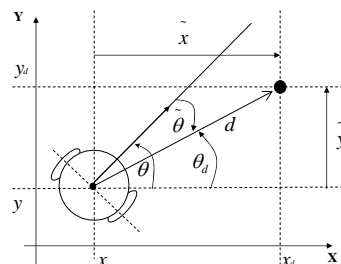


Fig. 1. Robot Khepera III et son modèle

A. Modèle du robot

Le robot étudié est unicycle. Il est actionné par deux roues indépendantes. Le déplacement du robot est décrit par un vecteur $M = [x \ y \ \theta]^T$ où le couple (x, y) désigne les coordonnées du centre de masse du robot par rapport au repère $R(O, \vec{x}, \vec{y})$ et l'angle θ représente l'orientation du véhicule par rapport à l'axe (O, \vec{x}) de ce même repère. On considère que l'axe de rotation passe par le centre de masse du robot. Le modèle cinématique est donné par les équations suivantes :

$$\dot{x} = v \cos \theta, \quad \dot{y} = v \sin \theta, \quad \dot{\theta} = \omega \quad (1)$$

v représente la vitesse linéaire du robot et ω sa vitesse angulaire. Le modèle cinématique du robot est tout à fait classique [5] et [6].

Le modèle dynamique du robot est caractérisé par la fonction de transfert des deux moteurs. Elle est trouvée par identification sur le prototype et est du premier ordre :

$$\frac{\omega_{(left, right)}}{U} = \frac{K}{\tau s + 1} \quad (2)$$

où U représente la tension appliquée à chaque moteur et $\omega_{(left, right)}$ représente la vitesse angulaire de chaque roue.

B. L'architecture de commande via un réseau

Deux niveaux de commande sont implémentés. Le premier niveau concerne la commande des deux roues en vitesse angulaire en utilisant un contrôleur PI discret pour chacune des roues motrices (cette boucle est purement locale). Le deuxième niveau de commande concerne la commande en vitesse linéaire et angulaire du robot en utilisant les lois de commande non-linéaires proposées par [6]. Le réseau sans fil a été intégré dans cette boucle de commande entre le contrôleur et les consignes de vitesse angulaire des roues d'une part et entre les capteurs (algorithme qui calcule en fonction des mesures des encodeurs la position du robot) et le contrôleur d'autre part.

L'outil de simulation des systèmes temps réel TrueTime [7] a été utilisé pour simuler le comportement du réseau sans fil. Deux tâches ont été programmées. La première est la tâche contrôleur qui génère un flux contrôleur et la deuxième concerne la tâche capteur qui, à son tour, génère le flux capteur. La tâche contrôleur est guidée par les événements, c'est-à-dire que le contrôleur calcule et envoie la consigne $(\omega_{r_k}, \omega_{l_k})$ à condition qu'il ait reçu toutes les mesures qui sont envoyées par le capteur $(\hat{x}_k, \hat{y}_k, \hat{\theta}_k)$.

B.1 Architecture de commande

Pour que le robot puisse rejoindre la position désirée définie par $[x_d \ y_d \ \theta_d]$, des erreurs de suivi (Eq. 3b) et d'orientation (Eq. 3c) sont calculées :

$$\tilde{x} = x_d - x, \quad \tilde{y} = y_d - y \quad (3a)$$

$$d = \sqrt{\tilde{x}^2 + \tilde{y}^2} \quad (3b)$$

$$\tilde{\theta} = \theta_d - \theta = \tan^{-1}(\tilde{y}/\tilde{x}) - \theta \quad (3c)$$

Les lois de commande suivantes sont considérées [6] :

$$v = \frac{v_{max}}{1 + |d|} d \cos(\tilde{\theta}) \quad (4a)$$

$$\omega = \frac{v_{max}}{1 + |d|} \cos(\tilde{\theta}) \sin(\tilde{\theta}) + K_{\tilde{\theta}} \tanh(k_{\tilde{\theta}} \tilde{\theta}) \quad (4b)$$

avec v_{max} la vitesse linéaire maximale que le robot peut atteindre ($v_{max} = 0.5m/s$) et $K_{\tilde{\theta}}, k_{\tilde{\theta}}$ des constantes.

B.2 Le protocole d'accès au médium

Le mode d'accès au médium utilisé dans notre cas est le CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance). Le principe du CSMA/CA est simple (voir Fig.2). Avant qu'une station ne décide d'émettre, elle vérifie que le canal de transmission n'est pas occupé. La vérification de la disponibilité du médium s'effectue par la couche physique. Si la détection a montré que le médium est libre, la station peut envisager de transmettre l'information. Sinon, elle doit attendre jusqu'à ce qu'elle reçoive que le canal est libre. Ce mécanisme est complété par un mécanisme pour générer des délais aléatoires avant transmission (random backoff ou encore CW pour Contention Windows). Chaque station exécute cet algorithme localement afin de déterminer à quel moment elle peut commencer sa transmission. Cette méthode nécessite l'emploi d'un espace de temps, d'une durée minimale spécifiée entre les transmissions. Cet espace est appelé IFS (Inter Frame Space) ou espace intertrame. Quatre IFS ont été définis par

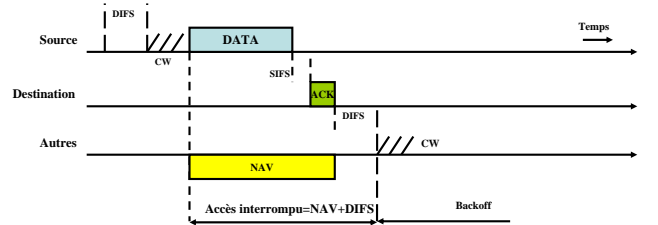


Fig. 2. Méthode d'accès CSMA/CA

le standard. Le SIFS (Short IFS) est le plus court de tous. Il est utilisé pour la transmission des trames d'acquittement (ACK) et les trames issues d'une même station. Le DIFS (DCF IFS) est le plus utilisé (avec le SIFS). Il est utilisé en mode DCF (Distributed Coordination Function) comme temps minimal d'attente avant transmission.

TABLE I
SPÉCIFICATIONS RÉSEAU

Paramètres	Valeur
Débit	802.11b
Taille de chaque trame	64bits
Période d'envoi (T_e)	400ms
Puissance émise (P_s)	-3dBm
Seuil de la puissance reçue (P_r)	-48dBm
Nombre de retransmissions en cas de perte de paquets	0
CW_{min}	3
SIFS	10μs
Slot Time (T_s)	20μs
$DIFS = 2T_s + SIFS$	50μs
Nombre max de retransmissions en cas de collision (k)	7

III. PROBLÈME DE CO-CONCEPTION

Le problème de co-conception dans notre cas, est d'essayer d'adapter le réseau aux contraintes de la commande (assurer une bonne QoC) et aussi aux contraintes du diagnostic pour détecter et localiser les différents défauts qui peuvent nuire au bon fonctionnement des différentes parties du système global. Il faut par ailleurs adapter la commande et le diagnostic aux besoins du réseau pour assurer toujours

une bonne QoS du réseau. Dans [5], une solution a été proposée comportant d'une part une stratégie de changement de station pour adapter le réseau aux contraintes de la commande, d'autre part une stratégie de commutation de contrôleur vers un contrôleur interne dégradé pour adapter la commande et assurer un minimum de QoC. Le défi maintenant est de décider quand le robot a vraiment besoin de changer de station et aussi de contrôleur. Pour cela, avant de déclencher la commutation de contrôleur ou le changement de station (Handover), on a besoin d'un outil d'aide à la décision qui, en fonction des différents paramètres qui influencent le réseau, peut qualifier la QoS du réseau et aussi la QoC du robot en estimant leurs états qualitatifs, à savoir, bons, dégradés ou mauvais. Nous avons aussi besoin d'identifier et localiser la cause de la dégradation de ces deux paramètres pour l'éviter ou la corriger. Dans cette section, nous proposons d'utiliser les BN pour l'aide à la décision et le diagnostic.

A. Les réseaux bayésiens (BN)

Un réseau bayésien est défini par [4] :

- Un graphe acyclique orienté $G, G=(V,E)$, où V est l'ensemble des nœuds de G , et E l'ensemble des arcs de G ,
- Un espace probabilisé fini (Ω,Z,P) ,
- Un ensemble de variables aléatoires associées aux nœuds du graphe et définies sur (Ω,Z,P) , tel que :

$$P(V_1, V_2, \dots, V_n) = \prod_{i=1}^n P(V_i|C(V_i)) \quad (5)$$

$C(V_i)$ est l'ensemble des causes (parents) de V_i dans le graphe G .

Les BN sont à la fois des modèles de représentation des connaissances et aussi des machines à calculer les probabilités conditionnelles (CPT : Conditional Probability Table). Il est donc nécessaire de distinguer deux approches différentes pour modéliser les connaissances par BN : l'approche qualitative et l'approche quantitative.

A.1 Approche qualitative

Dans cette partie, nous allons présenter l'aspect qualitatif qui est basé sur la théorie des graphes et représente la notion de causalité, c'est-à-dire, le lien de causes à effets. Nous avons étudié tous les paramètres qui influencent la QoS du réseau et la QoC du robot. La QoC est classiquement influencée par trois paramètres : la QoS du réseau, l'algorithme de commande et l'état du robot. Dans notre cas, l'algorithme de commande ne change pas, donc il a pas d'influence sur la QoC.

En se référant à [1] et [2], la QoS est influencée essentiellement par deux paramètres, à savoir, le retard entre le moment de transmission et le moment de réception, et le pourcentage de pertes de paquets (voir Fig.3). La puissance du signal reçu (Request Signal Strength Indicator (RSSI)) ainsi que le Bit Error Rate (BER) et la taille des buffers jouent un rôle sur le nombre de paquets perdus. Le BER a une valeur fixe suivant le réseau sans fil utilisé, donc on peut l'enlever de notre BN car on n'utilise qu'un seul réseau. On considère, dans notre cas, que le retard influence le nombre de pertes de paquets car si le retard dépasse une période

d'échantillonnage, on considère que le paquet est perdu [8]. Le retard considéré ici est le temps de propagation d'un paquet, le retard dû à l'attente de paquet dans la file d'attente n'est pas traité pour l'instant. Dans un BN, chaque nœud possède un ou plusieurs modes (ou états). La Table III présente les modes de chaque nœud du BN proposé.

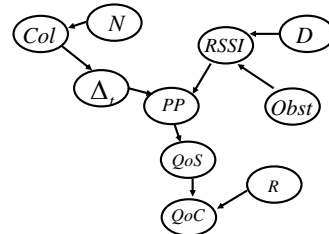


Fig. 3. Réseau bayésien du système

TABLE II

SIGNIFICATION DES DIFFÉRENTS NŒUDS DU BN

Nœuds	Signification
QoC	Qualité de Contrôle
QoS	Qualité de Service
R	Etat du robot
PP	Perte de Paquets
RSSI	Request Signal Strength Indicator
Δ_t	Retard agissant sur la QOS
Obst	Obstacle
D	Distance entre la station et le robot
Col	Collision
N	Nombre de nœuds

TABLE III

MODES DE CHAQUE NŒUD DU BN

Nœuds	Modes
QoC	Bonne, dégradée, mauvaise
QoS	Bonne, dégradée, mauvaise
R	Bon, dégradé, défaillant
PP	$0\% \leq PP < 40\%$, $40\% \leq PP < 60\%$, $60\% \leq PP < 100\%$
RSSI	$RSSI > -48dBm$, $RSSI < -48dBm$
Δ_t	$\Delta_t < T_e$, $\Delta_t \geq T_e$
Obst	Oui, Non
D	$D < Seuil$, $D \geq Seuil$
Col	≤ 0.8 , > 0.8
N	Nombre de nœuds

A.2 Approche quantitative

L'aspect quantitatif des BN consiste à associer à chaque nœud une table de probabilité qui le caractérise. Elle représente la spécification de l'ensemble des probabilités de la variable pour chacune de ses valeurs possibles connaissant chacune des valeurs de ses parents. Les tables de probabilités sont calculées généralement soit par expertise, statistiquement, soit grâce à un historique du système qu'on veut traiter. Dans notre cas, on utilise une étude statistique des différents paramètres du système.

Dans cette section, une étude a été faite pour déterminer toutes les CPT *a priori* nécessaires pour le BN proposé. Avant toute étude, une hypothèse a été adoptée. Nous avons besoin de déterminer les différents modes ou états de la QoS en fonction du pourcentage de pertes de paquets. Ce choix est arbitraire et peut être changé. En effet, la Table IV présente la CPT *a priori* de la QoS en fonction

des pertes de paquets.

TABLE IV

PROBABILITÉS *a priori* DE LA QoS EN FONCTION DES PERTES DE PAQUETS

$PP(\%)$	$0 \leq PP < 40$	$40 \leq PP < 60$	$60 \leq PP < 100$
$P(QoS = b PP)$	1	0	0
$P(QoS = d PP)$	0	1	0
$P(QoS = m PP)$	0	0	1

$b = \text{bonne}, d = \text{degradee}$ et $m = \text{mauvaise}$

1. Influence des pertes de paquets sur la QoC du robot : On va étudier *a priori* de façon statistique l'influence des pertes de paquets sur la QoC du robot de façon à discriminer trois états {Bonne QoC, QoC dégradée, Mauvaise QoC}. Pour cela, on a simulé les pertes sur TrueTime suivant l'Algorithme 1, *rand* est une fonction Matlab qui donne une valeur aléatoire η suivant une loi uniforme. η est la proposition de pertes de paquets, comprise entre 0 et 1. En ligne, le paramètre PP évoluera en fonction d'autres paramètres du réseau (Fig.3).

Algorithm 1 Simulation des pertes de paquets

```

PP = rand
if PP > η then
Calcul la valeur de la commande et prend en compte le changement de
cette valeur
end if

```

Nous avons choisi un critère quadratique (Eq. 6) pour déterminer les intervalles des différents états de la QoC. On a tracé l'évolution de ce critère en fonction du taux de pertes de paquets sur la Fig. 4. Cette courbe nous a aidé pour la définition des trois états.

$$QoC_{crit\u00e9rion} = \sum_{i=0}^k \epsilon_x^2 + \epsilon_y^2 \quad (6)$$

ϵ_x et ϵ_y sont les erreurs entre les points de la courbe de référence (à 0% de pertes de paquets) et les points des autres courbes (en fonction du pourcentage des pertes de paquets) à l'instant t suivant l'axe X et Y respectivement. k est le nombre de points représentant chaque courbe. La Fig. 5 montre les différentes trajectoires du robot en fonction de l'état de la QoC, à savoir : bonne, dégradée ou mauvaise, et l'évolution de ces trajectoire suivant l'axe X et Y . La Fig.6 montre l'évolution des vitesses linéaire, angulaire et de l'orientation du robot avec réseau et une station en fonction des trois états possibles de la QoC.

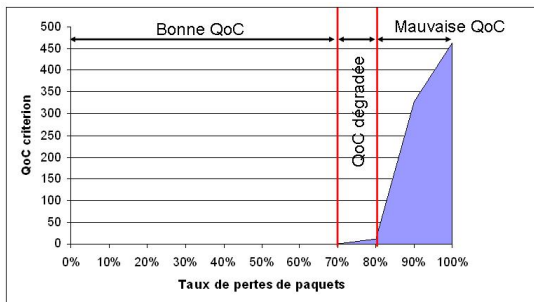


Fig. 4. Dégradation de la QoC en fonction des pertes de paquets

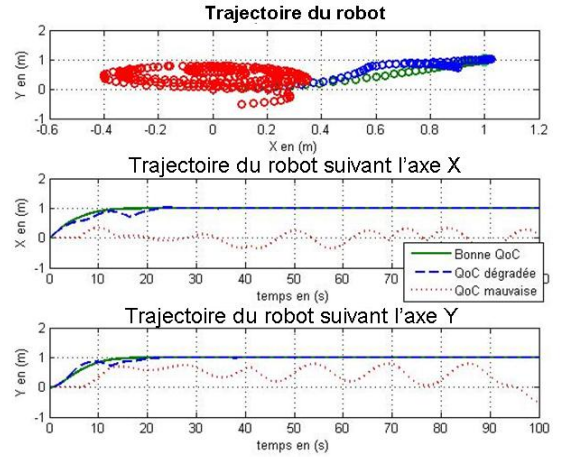


Fig. 5. Différentes trajectoires du robot en simulation en fonction de taux de pertes de paquets fixé dans TrueTime

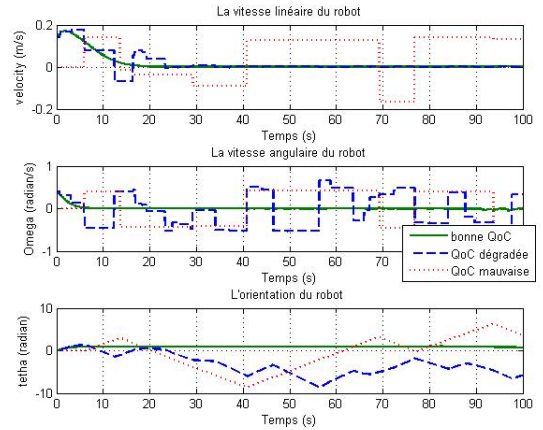


Fig. 6. Evolution des vitesses linéaire, angulaire et de l'orientation du robot avec réseau et une station

En se basant sur les résultats de la Fig.4 et l'hypothèse de la Table IV, la CPT *a priori* de la QoC en fonction de la QoS a été définie (Table V).

TABLE V

CPT *a priori* DE LA QoC EN FONCTION DE L'ÉTAT DU ROBOT ET LA QoS DU RÉSEAU SANS FIL

R	Normal			dégradé			défaillant		
	b	d	m	b	d	m	b	d	m
$P(QoC = b QoS, R)$	1	1	0.25	0	0	0	0	0	0
$P(QoC = d QoS, R)$	0	0	0.25	1	1	1	0	0	0
$P(QoC = m QoS, R)$	0	0	0.5	0	0	0	1	1	1

$b = \text{bonne}, d = \text{degradee}, m = \text{mauvaise}$ et $R = \text{Robot}$.

2. Influence de la distance et la présence d'obstacle sur la puissance du signal reçu (RSSI) : Dans cette partie, une étude de l'influence de la distance et la présence d'obstacle sur la RSSI a été faite sur TrueTime pour définir la CPT *a priori* du nœud RSSI. Pour cela, on a simulé un robot communicant avec une station et se déplaçant à partir d'un point initial $(x_0, y_0) = (0, 0)$ vers une cible qui est choisie d'une façon à ce que le robot à un moment sorte de la zone de couverture de la station. Nous avons pu remarquer que le robot perd complètement la communi-

tion après 18m. Donc, le rayon de courbure de la zone de couverture de la station est de 18m. $D = 18m$ sera pris comme un seuil de distance pour que le robot perde la communication. Concernant les obstacles, pour simplifier notre étude, on admet que si un obstacle se trouve entre la station et le robot, la communication sera interrompue. En conséquence, on a pu extraire des informations sur la distance et les obstacles sur la Table VI.

TABLE VI

CPT *a priori* DE LA RSSI EN FONCTION DE LA DISTANCE ENTRE LE ROBOT ET LA STATION ET LA PRÉSENCE D'OBSTACLES

Obs D	Non		Oui	
	< seuil	> seuil	< seuil	> seuil
$P(RSSI \geq -48 Obs, D)$	1	0	0.3	0
$P(RSSI < -48 Obs, D)$	0	1	0.7	1

3. Influence du retard et du RSSI sur le pourcentage des pertes de paquets : Dans cette partie, nous assumons que si le RSSI est inférieur à un seuil ($seuil = -48dBm$) les paquets transmis sont perdus. Aussi, si le retard est supérieur à une période d'échantillonnage, le paquet est considéré comme ancien et il sera jeté automatiquement. En conséquence, nous pouvons déterminer les probabilités *a priori* des différents états des pertes (PP) en fonction du retard (Δ_t) et du $RSSI$. La Table VII présente toutes ces probabilités.

TABLE VII

CPT *a priori* DES PERTES DE PAQUET EN FONCTION DU RETARD ET DU RSSI

Δ_t $RSSI(dBm)$	$\Delta_t < T_e$		$\Delta_t > T_e$	
	≥ -48	< -48	≥ -48	< -48
$P(0\% \leq PP < 40)$	1	0	0	0
$P(40\% \leq PP < 60)$	0	0	0	0
$P(60\% \leq PP < 100)$	0	1	1	1

4. Étude du délai de transmission : Dans cette partie, nous nous intéressons à l'étude du délai de transmission afin de trouver la CPT *a priori* du nœud retard (Δ_t) en fonction des paramètres ayant une influence directe sur ce dernier. Le terme retard ou délai de transmission désigne le temps que met un paquet pour être acheminé d'une source vers une destination. Ce temps prend en compte principalement deux sous-délais : le délai dans la file d'attente (Buffer) des nœuds et le délai de transmission (τ) du paquet sur le médium physique. Dans notre cas, on traite seulement le délai de propagation du paquet sur le médium physique. Le problème de la synchronisation des horloges a été traité dans de nombreux travaux [9] mais, pour ne pas compliquer notre étude, nous supposons que toutes les horloges des nœuds sont synchronisées. En se basant sur l'étude de [10], qui a proposé une méthode d'estimation de retard dans 802.11 en utilisant le mode DCF, on peut estimer le retard de transmission entre deux nœuds à partir des données de la charge du réseau, la longueur des trames et aussi la taille de la fenêtre de contention (CW) qui est un temps aléatoire entre deux transmissions successives. Soit n le nombre de retransmissions associé à la probabilité de collision p . D'après [10], n est calculé de la façon suivante :

$$n = \frac{6p^8 - 7p^7 + p}{1 - p} \quad (7)$$

La probabilité de collision est calculée en utilisant l'approximation proposée par [10] suivante :

$$p = \frac{2CW_{min}(m-1)}{(CW_{min}+1)^2 + 2CW_{min}(m-1)} \quad (8)$$

m étant le nombre de nœuds voulant transmettre en même temps.

$$\tau = \sum_{k=0}^n (DIFS + (1/2)2^k CW_{min} + T_m) \quad (9)$$

T_m est calculé de la façon suivante :

$$T_m = \Gamma / Debit \quad (10)$$

Γ est la taille de la trame.

La courbe de " p " en fonction de " N " et la courbe du " τ " estimé en fonction de " p " ont été tracées pour définir les seuils de " N " et de " Col " pour que le retard dépasse une période d'échantillonnage " $Te = 400ms$ ". On a remarqué que la probabilité de collision dépasse 0.8 si le nombre de nœuds dépasse 12 nœuds, le seuil de probabilité de collision qui fait que le retard dépasse $Te = 400ms$ est 0.8. On peut donc extraire les deux tableaux de CPT *a priori* des nœuds Δ_t et Col . La Table VIII présente la CPT *a priori* du retard en fonction de la probabilité de collision. La Table IX présente la CPT *a priori* de la collision en fonction du nombre de nœuds voulant communiquer.

TABLE VIII

CPT *a priori* DU RETARD EN FONCTION DE LA PROBABILITÉ DE COLLISION

Col	≤ 0.8	> 0.8
$P(\Delta_t \leq Te)$	1	0
$P(\Delta_t > Te)$	0	1

TABLE IX

CPT *a priori* DE LA COLLISION EN FONCTION DU NOMBRE DE NŒUDS VOULANT COMMUNIQUER

N	≤ 12	> 12
$P(Col \leq 0.8)$	1	0
$P(Col > 0.8)$	0	1

5. Détermination des probabilités marginales des nœuds sans parent : Dans cette partie, les probabilités marginales des nœuds sans parent seront déterminées. Pour le nœud " $Robot$ ", on considère qu'un diagnostiqueur interne utilisant les méthodes de diagnostic classiques, peut détecter et localiser les défaillances liées au robot (moteurs, capteurs ...). Pour le BN que nous avons proposé, ce qui nous intéresse est l'état du robot en général (normal, dégradé ou défaillant). En ce qui concerne les deux nœuds D et N , ce sont des nœuds mesurés, mais on doit déterminer des probabilités *a priori* initiales. Et comme à l'état initial, on ne peut pas connaître la distance ni le nombre de nœuds communicants, nous avons mis une probabilité incertaine pour chaque état. Cette probabilité sera changée dès que l'on mesure les valeurs de chaque variable.

IV. AIDE À LA DÉCISION ET DIAGNOSTIC

Après avoir déterminé toutes les probabilités conditionnelles et marginales *a priori*, nous pouvons passer maintenant à comment prendre une décision et comment diagnostiquer le système en calculant les probabilités *a posteriori* en fonction des évidences ou des mesures.

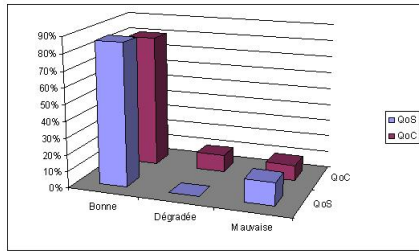


Fig. 7. Etat du BN avec une connaissance sur la distance et le nombre de nœud communicants

A. Aide à la décision

Prendre une décision dans notre application est très important, que ce soit au niveau de changement de station ou bien de la commutation de contrôleur. Pour cela, on a besoin d'estimer ou de déterminer l'état de la QoC et de la QoS pour pouvoir prendre une décision adéquate à la situation. En fonction des observations sur le système, les tables de probabilités *a posteriori* $P(QoS|Observations)$ et $P(QoC|Observations)$ seront calculés par inférence en utilisant l'algorithme d'inférence JLO [4]. Supposons qu'on ait des informations sur la distance entre le robot et la station et aussi sur le nombre de nœuds qui veulent communiquer (robots et stations). En actualisant les tables de probabilités de ces deux nœuds, les probabilités *a posteriori* de chaque nœud du réseau seront calculées et le résultat est présenté sur la Fig.7 Nous remarquons qu'avec ces évidences, la QoC et la QoS sont dans un état "bon". Changeons maintenant nos évidences sur le RSSI. Le robot sort de la zone de couverture de la station donc la valeur du RSSI est inférieure au seuil. La Fig.8 montre qu'avec cette connaissance, la QoC est mauvaise avec une probabilité de 49% et la QoS est mauvaise à 100%. En connaissant des informations sur le système, nous avons pu qualifier la QoC du robot et la QoS du réseau. En conséquence, on peut prendre une décision en fonction de l'état de la QoC et la QoS de changer la station ou bien de commuter le contrôleur vers un contrôleur embarqué.

B. Diagnostic

Si nous observons une dégradation de la QoC, nous pouvons en utilisant le BN proposé détecter et localiser la cause de la dégradation de la QoC. Avec les connaissances ou les évidences adoptées, à savoir, la distance est inférieure au seuil, le nombre de nœuds est inférieur au seuil aussi et il n'y a pas d'obstacle, le BN a pu localiser la cause. La seule raison qui peut causer cette dégradation est la défaillance du robot.

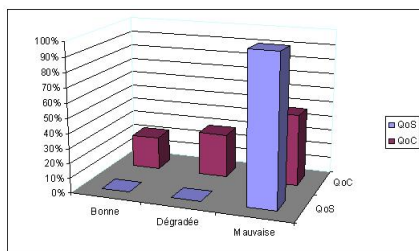


Fig. 8. Etat du BN avec une connaissance sur la distance et le nombre de nœud communicants et le RSSI

V. CONCLUSION

Cette article présente la simulation d'un robot mobile commandé à travers un réseau sans fil en utilisant Simulink/TrueTime. Cette simulation permet de préparer une expérimentation avec un robot Khepera mais présente quelques limitations dues à TrueTime (pas de *Buffer*). Nous avons proposé une approche basée sur les réseaux bayésiens dans le but de qualifier à la fois la QoC et la QoS, et de permettre l'aide à la décision et le diagnostic d'un système. Le choix de cette approche est justifié par la possibilité d'utiliser le même modèle pour l'aide à la décision et le diagnostic du système.

Pour déterminer les probabilités *a priori* de chaque nœud, nous avons traité dans ce papier l'influence des différents paramètres sur chacun des nœuds du BN proposé. Nous avons présenté ensuite quelques scénarios en ajoutant à chaque fois des observations. Le BN proposé a été utilisé pour le diagnostic et l'aide à la décision. L'utilisation d'un réseau bayésien dynamique (DBN) sera nécessaire pour la prise de décision et le diagnostic du système en ligne [11]. L'objectif le plus intéressant de cette étude est d'utiliser après le même modèle du réseau bayésien proposé pour pronostiquer ou anticiper la dégradation de la QoS du réseau et la QoC du robot pour éviter qu'elle ne se produise. L'implémentation de ce BN dans notre simulateur du robot commandé en réseau est en cours.

Vu la complexité de calculer les probabilités *a posteriori* de chaque nœud du BN, la distribution du BN est envisageable pour la suite et la communication entre sous BN sera assurée par le réseau, ce qui nous amène à étudier l'influence de l'intégration du réseau sur la performance du réseau bayésien distribué.

RÉFÉRENCES

- [1] J.P. Hespanha, P. Naghshtabrizi, and Y. Xu. A survey of recent results in networked control systems. In *Proceeding of IEEE special issue on Technology of Networked Control Systems*, volume 95, pages 138–162, 2007.
- [2] L. Li and F-Y Wang. Control and communication synthesis in networked control systems. In *International journal of intelligent control and systems*, volume 13, pages 81–88, 2008.
- [3] J. Pearl. *Probabilistic reasoning in intelligent systems : Networks of plausible inference*. Morgan Kaufmann Publishers, 1988.
- [4] P. Naïm, P.H. Willemin, P. Leray, O. Pourret, and A. Becker. *Réseaux bayésiens*. Eyrolles, 3rd edition, 2004.
- [5] A. Mechraoui, Z. H. Khan, J-M. Thiriet, and S. Gentil. Co-design for wireless networked control of an intelligent mobile robot. In *International Conference on Informatics in Control, Automation and Robotics (ICINCO)*, Milan, Italy, 2009.
- [6] J. M. Toibero, R. Carelli, and B. Kuchen. Switching control of mobile robots for autonomous navigation in unknown environments. In *Proceeding of IEEE International Conference on Robotics and Automation*, Roma, Italy, 2007.
- [7] M. Anderson, D. Henriksson, and A. Cervin. Truetime 1.5 : reference manual. Technical report, 2007.
- [8] C. Berbra, S. Gentil, S. Lesecq, and J.M. Thiriet. Co-design for a safe network control dc motor. In *Proceeding of 17th IFAC World Congress, Seoul, South Korea*, 2008.
- [9] B. Wehbi, A. Laouiti, and A. R. Cavalli. Efficient time synchronization mechanism for wireless multi hop networks. In *19th annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, Cannes, France*, 2008.
- [10] C. Sarr, C. Chaudet, G. Chelius, and I. G. Lassous. Amélioration de la précision pour l'estimation de la bande passante résiduelle dans les réseaux ad hoc basés sur ieee 802.11. In *JDIR*, 2007.
- [11] A. Mechraoui. On-line distributed bayesian decision and diagnosis of wireless networked mobile robots. In *Note interne, Gipsa-Lab*, 2010.