



**HAL**  
open science

## Diagnosis architecture reconfiguration for a networked mobile robot

Insaf Sassi, Alexia Gouin, Jean-Marc Thiriet

► **To cite this version:**

Insaf Sassi, Alexia Gouin, Jean-Marc Thiriet. Diagnosis architecture reconfiguration for a networked mobile robot. ESREL 2017 - 27th European Safety and Reliability Conference, Jun 2017, Portorož, Slovenia. 10.1201/9781315210469-373 . hal-01558498

**HAL Id: hal-01558498**

**<https://hal.science/hal-01558498>**

Submitted on 7 Jul 2017

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Diagnosis architecture reconfiguration for a networked mobile robot

I. Sassi, A. Gouin & JM. Thiriet

*Univ. Grenoble Alpes, CNRS, Gipsa-Lab, F-38000 Grenoble, France*

**ABSTRACT:** Wireless networked robots are mobile systems communicating to a control station via a wireless network. These robots move in a distributed infrastructure while receiving orders from control stations to reach their target. A distributed and collaborative diagnosis architecture is a good solution to monitor and diagnose the different components behavior of such distributed system due to the unreliability of wireless networks. Modular Bayesian Network (MBN), as a distributed Bayesian Network, can manage causal uncertain factors (stochastic failures dependencies) and infer information coming from different environments and from devices having memory and calculation limitations. MBN was proposed in our previous work for distributed and collaborative diagnosis. However, the robots can receive orders to switch to autonomous operating mode or to a good performing network (a new control station) to avoid system performance degradation. The robot is no more connected to the station of origin. The diagnosis strategy must consequently be updated. This paper proposes a procedure to reconfigure the diagnosis architecture and the MBN when the mobile robot changes its operating mode i.e, switching from embedded controller to distant controller.

## 1 INTRODUCTION

Distributed control and diagnosis have become essential for systems having memory and calculation limitations and energy consumption constraints like in the case of mobile robotic systems. Networked mobile robots have been increasingly deployed in a large number of applications. A wireless network provides the communication between the mobile robot and the control station where the diagnosis and decision making results are made. Using wireless networks offers advantages like flexibility, lack of wiring, mobility and fast deployment (Li et al. 2015). However, the stochastic behavior of such networks does not guarantee a reliable information transmission, which can lead to problems that affect the Wireless Networked Control System (WNCS) performance as proved in (Simon et al. 2010).

A Bayesian Network (BN) for diagnosis of wireless networked mobile robot was consequently proposed in our previous work (Sassi et al. 2016) to avoid performance degradation. The proposed BN multilayer structure was inspired from many BN for diagnosis such as Przytula and Choi (2007) model. Besides, we introduced a Modular Bayesian Network (MBN) as a distributed diagnosis method to reduce the on-board processing overhead (i.e. load calculations) on the robot and because of the incomplete vision about the environment from both robot and station sides. The proposed MBN permits to determine the cause

of Quality of Control (QoC) degradation for a networked mobile robot, whether it comes from the robot internal state, the network or the controller. Based on the QoC state and the network Quality of Service (QoS) state, the remote station can order the robot to switch to its embedded controller to maintain the system in an operational state. In this case, the robot is no more connected to the station. It becomes totally independent of the station until it finds a new one that guarantees a good QoS and potentially a better QoC. The network state does not impact the system performance after switching to embedded controller. The diagnosis architecture must thus be updated. There are three Bayesian modules, in the proposed MBN, communicating via virtual linking between shared nodes (Hwang and Cho 2009): the control module, the operative module and the network module as explained in (Sassi et al. 2016). When the robot becomes autonomous, the network module does not take part in the diagnosis procedure anymore. Once it finds a new control station, the network module must be reintegrated. Our study objective is to define a procedure to switch from a monolithic BN (singly connected Bayesian structure) to a distributed BN (modularized Bayesian structure). Besides, BN modules can be included in the global Bayesian network (or removed) and consequently shared nodes are added (or removed) in a Bayesian module. Conditional probabilities tables must be updated in order to adjust the new BN structure. The extension of the probability ta-

ble in the reconfiguration procedure is inspired from the extension and combination phase of the BNs consensus procedure in (Hu and Wang 2013). The joint probabilities distributions extension is studied in order to define the combination procedure.

The case study is a simulated networked mobile robot moving toward a target position. The mobile system can be autonomous or connected to a control station. In this work, when the networked robot switches to the embedded controller (autonomous behavior), the QoS node is removed from the control Bayesian sub-net. The control Bayesian module and the operative Bayesian module are assembled and combined in one Bayesian structure implemented on the robot. The network Bayesian module and the QoS node are reintegrated once the robot finds a new station. The reintegration method will be detailed in this work.

The paper is organized as follows. Section 2 presents some review basics of BNs and MBN. Section 3 describes the procedure of BNs quantitative and qualitative extension. Section 4 is dedicated to diagnosis architecture reconfiguration procedure. A combination and diagnosis reconfiguration scenario is presented in section 5. Finally, conclusions and perspectives are given in section 6.

## 2 BACKGROUND

### 2.1 Bayesian Networks

BNs are directed acyclic graphs that represent causal relationships between a set of random variables (Pearl 1988). The causal dependencies are presented by edges between the nodes corresponding to discrete/continuous variable. Each node is associated with a probability distribution. The causal relationship is represented by the conditional probability table of a variable knowing the state of its parents nodes. Let  $G = (V, E)$  be a directed acyclic graph (DAG) where  $V$  is the set of variables and  $E$  is the set of edges in the BN. The joint probability distribution corresponding to the BN variables can be written as follows:

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

where  $pa(V_i)$  is the set of  $V_i$  parents.

The d-separation in a BN is a rule to describe the conditions where the information is blocked between two distinct variables (nodes)  $X$  and  $Y$ . There are three types of connections between BN nodes (Naïm et al. 2007): serial, converging and diverging connections.  $X$  and  $Y$  are d-separated by  $Z$  means that  $Z$  blocks the information path between  $X$  and  $Y$ . It implies that knowing the state of  $X$  (apply a hard evidence on  $X$ ) does not change our knowledge about  $Y$  state. For all paths between  $X$  and  $Y$  in serial or diverging connection, if  $Z$  is instantiated, the information circuit

between  $X$  and  $Y$  is blocked. Meanwhile, in case of converging connection, if neither  $Z$  nor any of its descendants has received evidence, the information path is not open between  $X$  and  $Y$ . The d-separation property is useful to optimize the computation complexity in the inference procedure in MBNs as explained later in section 2.2.

### 2.2 Modular Bayesian Networks

Hwang and Cho (2009) proposes a MBN as a solution for complex and large scale systems in order to reduce computational complexity. It is designed for mobile environment and devices with resource limitations. MBN, as defined in (Lee et al. 2015) and (Lim et al. 2016), guarantees more extensibility and efficiency. Besides, it is possible to reuse BN modules in different applications. MBN is a distributed Bayesian structure obtained by dividing one large BN into several Bayesian modules using the modularization procedure detailed in (Hwang and Cho 2009). The modularization consists in defining the modules by selecting the corresponding variables to each domain. An acyclic sub-graph is then built for each module based on the global acyclic graph of the monolithic BN. Each obtained BN module is thus a sub-graph of the monolithic BN. Bayesian Modules has the same conditional probability tables (CPTs) definition as the monolithic BN. Every Bayesian module (Bayesian sub-net) presents a sub-domain of the global system and shares variables (set of nodes) with other Bayesian subnets. Finally, MBN  $\Omega$  is defined as following (Lee et al. 2015):

- 2-tuple  $(\Psi, R)$  where  $\Psi$  is the set of BN modules in  $\Omega$  and  $R$  represents the causality between the BN modules.
- Two modules  $\Psi_i = ((V_i, E_i), P_i)$  and  $\Psi_j = ((V_j, E_j), P_j)$  have influence on each other if there is a link  $R = \{(\Psi_i, \Psi_j) | i \neq j, V_i \cap V_j \neq \emptyset\}$  representing a causal-effect dependency between two modules.

The d-separation property between BN modules is similar to the d-separation concept between nodes in a BN for the three types of connection.  $\Psi_i$  and  $\Psi_j$  are d-separated by the shared node  $S$  means that any changes in  $\Psi_i$  or  $\Psi_j$  do not affect the other connected module. BN inference algorithm is used for inference in MBN structure. The inference in MBN is a two-step inference. The first step consists in applying hard evidence in every Bayesian Module. The result of the first step inference is then communicated between the connected modules. Virtual nodes are added as a child to every shared node to pass information between the BN modules as explained in (Hwang & Cho 2009). The virtual node has two possible states; yes and no and its CPT is defined according to the received soft evidence of the shared variable. A hard evidence is

then applied on the virtual node and the BN module performs a second inference. If two connected modules  $\Psi_i$  and  $\Psi_j$  are d-separated, there is no need to re-inference the other module e.g  $\Psi_j$  because the changes in  $\Psi_i$  do not affect  $\Psi_j$ .

### 3 BAYESIAN NETWORKS COMBINATION PROCEDURE

Consensus BNs is the result of combining several BNs learned from different experts or data bases. Combination in consensus consists in combination of graph models and aggregation of probability distributions (Del Sagrado and Moral 2003). Hu and Wang (2013) propose a combination method to obtain a consensus BN to model the reactive oxygen species regulatory pathway. It consists in joining a set of graphical models for qualitative combination and aggregation of probability distributions for quantitative combination. Their algorithm can be applied to the com-

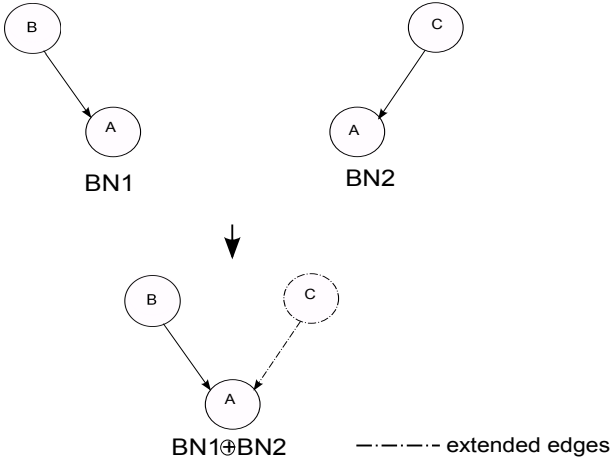


Figure 1: Bayesian Network extension: qualitative extension

ination of BNs defined over different variables. The extension phase in aggregation and combination algorithm is used in the diagnosis reconfiguration procedure. Let  $BN_1$  and  $BN_2$  be two BNs to be combined by extending  $BN_1$  into  $BN_1 \oplus BN_2$  and  $V_1$  and  $V_2$  be their variables sets verifying the following conditions:

- $V_1 \neq V_2$
- $V_1 \cap V_2 \neq \emptyset$

The extension procedure is described as following:

- **Qualitative extension:** Let  $G_1$  be the DAG of  $BN_1$  and  $G_2$  be the DAG of  $BN_2$ .  $G_1$  is extended into  $G_1 \oplus G_2$  such that  $G_1 \oplus G_2 = \{V_1 \cup V_2, E_1 \cup E\}$  where E is the set of edges  $\{(C, A) \in E_2 | C \in V_2 - V_1, A \in V_1\}$ . Solid lines in  $BN_1 \oplus BN_2$  represent the edges in  $BN_1$  and the dashed lines are the extended edges as illustrated in figure 1. The variables in the resulted graph should be consistent with each other which means that  $G_1 \oplus G_2$  is acyclic.

- **Quantitative extension:** This step consists in computing the CPT of each node in  $G_1 \oplus G_2$ . If  $A \in V_2 - V_1$  or  $A \in V_1 - V_2$  then the CPT of A does not change. If  $A \in V_1$  and C is such as  $(C, A) \in E_2 | C \in V_2 - V_1$ , conditional probabilities of A are computed using equation 2 where  $A, B \in V_1, C \in V_2 - V_1$ .

$$P(A = a_i | B = b_j, C = c_k) =$$

$$\frac{P_1(A = a_i | B = b_j) \times \frac{P_2(C=c_k | A=a_i, B=b_j)}{P_2(C=c_k | B=b_j)}}{\sum_{i=1}^n P_1(A = a_i | B = b_j) \times \frac{P_2(C=c_k | A=a_i, B=b_j)}{P_2(C=c_k | B=b_j)}} \quad (2)$$

where we have:

- $P_1$  and  $P_2$  are the probabilities distributions in  $BN_1$  and  $BN_2$  respectively.
- B and C are independent in  $BN_2$  because there is no dependency between B and C and because B and A are disconnected.

Given the independence properties between the variables, equation 2 can be simplified to obtain equation 3:

$$P(A = a_i | B = b_j, C = c_k) =$$

$$\frac{P_1(A = a_i | B = b_j) \times \frac{P_2(C=c_k | A=a_i)}{P_2(C=c_k)}}{\sum_{i=1}^n P_1(A = a_i | B = b_j) \times \frac{P_2(C=c_k | A=a_i)}{P_2(C=c_k)}} \quad (3)$$

$$\text{where } \frac{P_2(C=c_k | A=a_i)}{P_2(C=c_k)} = \frac{P_2(A=a_i | C=c_k)}{P_2(A=a_i)}$$

### 4 DIAGNOSIS ARCHITECTURE RECONFIGURATION PROCEDURE

The distributed modular BN in (Sassi et al. 2016) is a BN for diagnosis of a networked control system which is the networked robot. Three BN modules are used in order to perform diagnosis and determine the cause of the system degradation as shown in figure 2. A BN module has a multilayer structure where we have three types of nodes e.g, Observations nodes which are used to monitor the system behavior and to detect faults occurrence. The operative module is implemented on the robot while the control and the network Bayesian modules are on the station side. Both network Bayesian module and operative Bayesian module affect the control Bayesian module. They share QoS and robot state nodes with the control module. These shared nodes are the communication tool between the Bayesian modules by adding the corresponding virtual nodes as child to perform the

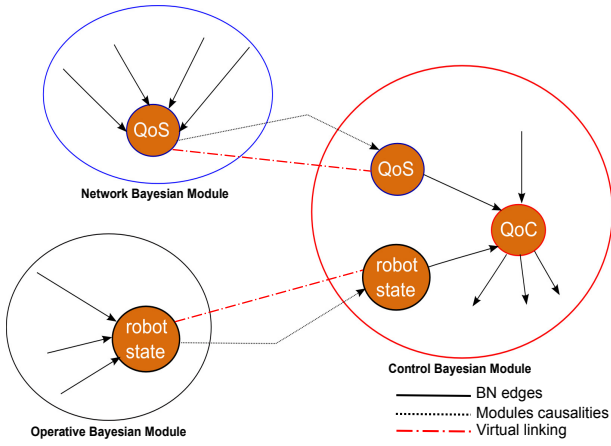


Figure 2: Modular Bayesian Network for diagnosis

diagnosis.

The robot switches to its embedded controller once the QoC becomes bad because of the QoS degradation as explained in our previous work (Sassi et al. 2016). It becomes totally autonomous until it finds another control station that guarantees a good service. The network module does not intervene in the diagnosis procedure and the network state does not impact the system performance in this case. The shared QoS node is removed from the QoC parents set because it does not affect the QoC state anymore. The diagnosis architecture should thus be updated to adjust the new system architecture. A monolithic BN is used combining the operative module and the control module and re-configuring the QoC conditional probabilities table because only the robot state and the controller state impact the QoC state as shown in figure 3. The statistical estimation and observations are

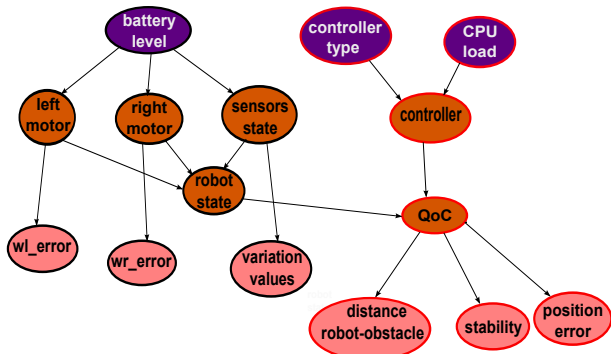


Figure 3: The embedded Bayesian Network

performed defining the probabilities values in the embedded monolithic BN. Once the robot is connected to a new control station, the network module takes action again. Besides, the monolithic BN is modularized using the procedure described in section 2 in order to re-obtain the distributed modular BN. Once the monolithic BN is modularized into operative and control modules, the marginal probability of the robot state  $P(robotstate)$  is computed using the inference algorithm in the operative module. The control module is then adopted by the new station.

The  $BN_1$  (the control Bayesian module) is combined to  $BN_2$  as shown in figure 4. The BN combination

procedure, which consists in qualitative and quantitative BN extension as explained in section 3, permits to reintegrate the QoS node to QoC parents set. The condition probabilities tables of the QoC is extended using the equation 3 for quantitative extension. The QoS marginal probability is computed using inference algorithm in the network Bayesian Module.

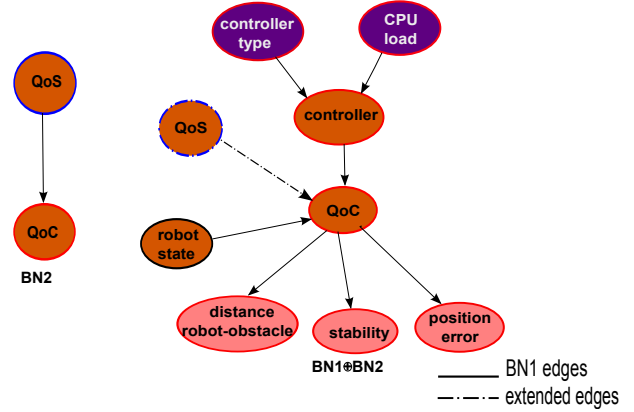


Figure 4: Extended Bayesian Control Module

## 5 COMBINATION SCENARIO

The objective of the integration of a consensus step and the modularization procedure is to find a solution for diagnosis architecture configuration and update. It consists in configuring the on-board monolithic BN implemented on the robot to have a distributed Bayesian structure. The distributed BN is a MBN where we have three modules as explained in section 4. The simulation scenario of figures 5 and 6

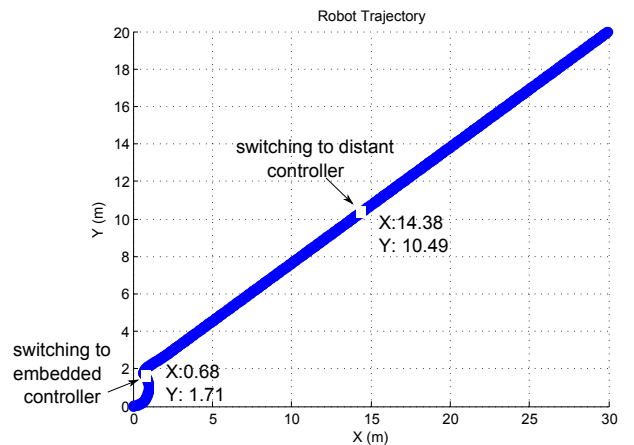


Figure 5: Robot trajectory

represents the robot behavior during its mission: moving toward the target position. Truetime toolbox (2.0 version) (Cervin et al. 2010) is used to simulate the networked system while BNT toolbox (Murphy et al. 2001) is used for MBN implementation. The robot has a mission to achieve a target position (30,20) starting from its initial position (0,0). The robot receives control data from the first station whose position is (1,1). The network conditions in the first network are bad because of a high packet loss rate (packet

Table 1: CPT of QoC state given the controller and the robot states

R	good			degraded			bad		
C	good	degraded	bad	good	degraded	bad	good	degraded	bad
$P(QoC=good R,C)$	1	1	0	0	0	0	0	0	0
$P(QoC=degraded R,C)$	0	0	0	1	1	0	0	0	0
$P(QoC=bad R,C)$	0	0	1	0	0	1	1	1	1

loss rate=80%). This network fault causes QoC degradation which corresponds to posterior probabilities  $P(QoC = bad) = 1$  and  $P(QoS = bad) = 0.87$  after setting hard evidences on the corresponding observations nodes. This degradation is observed when the robot is in  $(X, Y) = (0.68, 1.71)$  in figure 5 and it does not navigate towards the target position. This deviation from the expected behavior is detected because the position error does not decrease and the stability is not guarantee as shown in figure 6. The resulted hard evidences on observations nodes are  $P(stability = no) = 1$  and  $P(positionerror = nodecrease) = 1$ . Using the decision making algorithm defined in (Sassi et al. 2016), the robot switches to embedded controller as shown in figures 5 and 6. The on-board BN

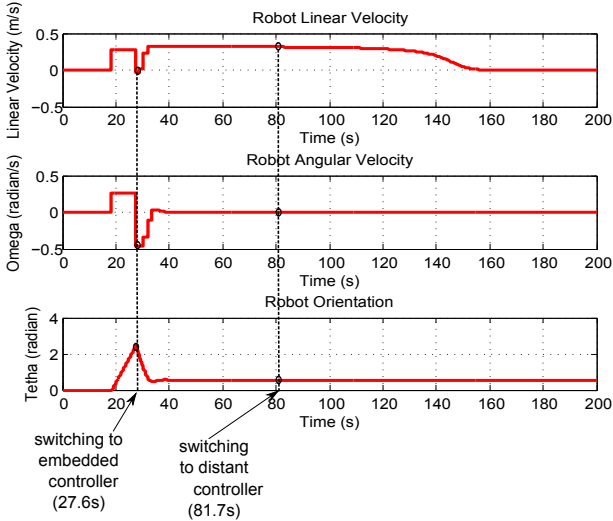


Figure 6: Robot velocities and orientation during navigation

supervises the QoC and the internal robot state. The QoC depends only on the robot state  $\mathbf{R}$  and the controller performance  $\mathbf{C}$  as shown in the CPT of QoC in table 1. The robot keeps performing statistical parameters estimations and observations.

Once the robot detects the second station (25,25), the robot state marginal probability is  $P(R) = [0.9 \ 0.07 \ 0.03]$  which is sent with the observations variables values ( $P(stability = yes) = 1$  and ( $P(positionerror = decrease) = 1$ . It switches to distant controller at 81.7s. The switching process used to reconnect to the control station (a new distant controller) guarantees the continuity and the stability of the robot performance as shown in figure 6. The cause-effect dependency between the QoC and the QoS is quantified in the new network and the CPT of QoC given QoS state is represented in table 2. The QoS-QoC dependency is integrated in the CPT

Table 2: CPT of QoC state given the QoS

QoS	good	degraded	bad
$P(QoC = good QoS)$	1	0	0
$P(QoC = degraded QoS)$	0	1	0.25
$P(QoC = bad QoS)$	0	0	0.75

of QoC as shown in table 3 (prior CPT) using equation 3. The prior probabilities of QoS in  $BN_2$  and in  $BN_1 \oplus BN_2$  is the result of executing the inference algorithm in the Bayesian network module such that  $P(QoS) = [0.8 \ ; \ 0 \ ; \ 0.2]$ . The marginal probabilities of QoC in  $BN_2$  are computed such that  $P(QoC) = [0.8 \ ; \ 0.05 \ ; \ 0.15]$ . We use equation 3 to calculate each value in the CPT of table 3 making some approximations in probabilities values ( $1 \simeq 0.999$ ,  $0 \simeq 0.0001$  or  $0 \simeq 0.0009$ ) to avoid having zero in the denominator of equation 3:

$$P(QoC = d|R = g, C = g, QoS = d) =$$

$$\frac{P_1(QoC = d|R = g, C = g) \times \frac{P_2(QoC=d|QoS=d)}{P_2(QoC=d)}}{\sum_{s=g,d,b} P_1(QoC = s|R = g, C = g) \times \frac{P_2(QoC=s|QoS=d)}{P_2(QoC=s)}} \quad (4)$$

where g,d and b refer to the variables possible states which are good, degraded and bad respectively. The CPT of QoC is partially represented in table 3 where the robot state is good.

$$P(QoC = d|R = g, C = g, QoS = d) =$$

$$\frac{0.0009 \times \frac{0.999}{0.05}}{0.999 \times \frac{0.0009}{0.8} + 0.0009 \times \frac{0.999}{0.05} + 0.0001 \times \frac{0.0001}{0.15}} = 0.94$$

## 6 CONCLUSION

A diagnosis configuration procedure is proposed in this paper to guarantee a coherence with system architecture update. The monolithic BN on-board is used to collect statistical data about the intern state of the robot and the causal dependencies. The autonomous behavior persists until a new control station takes control of the robot. It implies that the control structure is no more embedded but it is a distributed one. A distributed diagnosis architecture becomes a mandatory to monitor all the system components in order

Table 3: CPT of QoC state given the robot, the controller and the QoS states

R	good								
C	good			degraded			bad		
QoS	good	degraded	bad	good	degraded	bad	good	degraded	bad
$P(\text{QoC}=\text{good}   \text{R}, \text{C}, \text{QoS})$	1	0.06	0	1	0.06	0	0.16	0	0
$P(\text{QoC}=\text{degraded}   \text{R}, \text{C}, \text{QoS})$	0	0.94	0.9	0	0.94	0.9	0.02	0.96	0
$P(\text{QoC}=\text{bad}   \text{R}, \text{C}, \text{QoS})$	0	0	0.1	0	0	0.1	0.82	0.04	1

to maintain the operational state. This diagnosis update requires the modularization of the monolithic BN and the integration of new elements: the network Bayesian module, the inclusion of the dependency between QoS and QoC in the control module and the CPT extension of the QoC. CPT update of QoC is presented in this work to apply the extension steps as explained previously.

The combination procedure of control and operative Bayesian modules will be detailed in future work. The defined procedure and the MBN for diagnosis will be implemented on the real system. The Bayesian method will be then adopted for the case of multi-robots to define a collaborative diagnosis policy.

## REFERENCES

- Cervin, A., D. Henriksson, & M. Ohlin (2010). Truetime 2.0 betareference manual. *Department of Automatic Control, Lund University (June 2010)*.
- Del Sagrado, J. & S. Moral (2003). Qualitative combination of bayesian networks. *International Journal of Intelligent Systems* 18(2), 237–249.
- Hu, L. & L. Wang (2013). Using consensus bayesian network to model the reactive oxygen species regulatory pathway. *PLoS one* 8(2), e56832.
- Hwang, K.-S. & S.-B. Cho (2009). Landmark detection from mobile life log using a modular bayesian network model. *Expert Systems with Applications* 36(10), 12065–12076.
- Lee, S.-H., K.-M. Yang, & S.-B. Cho (2015). Integrated modular bayesian networks with selective inference for context-aware decision making. *Neurocomputing* 163, 38–46.
- Li, X., D. Li, J. Wan, A. V. Vasilakos, C.-F. Lai, & S. Wang (2015). A review of industrial wireless networks in the context of industry 4.0. *Wireless Networks*, 1–19.
- Lim, S., S.-H. Lee, & S.-B. Cho (2016). A modular approach to landmark detection based on a bayesian network and categorized context logs. *Information Sciences* 330, 145–156.
- Murphy, K. et al. (2001). The bayes net toolbox for matlab. *Computing science and statistics* 33(2), 1024–1034.
- Naim, P., P. Willemin, P. Leray, O. Pourret, & A. Becker (2007). *Les réseaux bayésiens*. (3<sup>ème</sup> édition ed.). Paris: Eyrolles.
- Pearl, J. (1988). Probabilistic reasoning in intelligent systems: Networks of plausible reasoning.
- Przytula, K. W. & A. Choi (2007). Reasoning framework for diagnosis and prognosis. In *Aerospace Conference, 2007 IEEE*, pp. 1–10. IEEE.
- Sassi, I., A. Gouin, & J.-M. Thiriet (2016). A bayesian network for diagnosis of networked mobile robots. In *European Safety and Reliability Conference 2016*.
- Simon, D., Y.-Q. Song, & C. Aubrun (2010). *Co-design approaches to dependable networked control systems*. John Wiley & Sons.