

A Multi-agent system with Distributed Bayesian Reasoning for Network Fault Diagnosis

Álvaro Carrera, Javier Gonzalez-Ordás, Javier García-Algarra, Pablo Arozarena,
and Mercedes Garijo

Abstract In this paper, an innovative approach to perform distributed Bayesian inference using a multi-agent architecture is presented. The final goal is dealing with uncertainty in network diagnosis, but the solution can be of applied in other fields. The validation testbed has been a P2P streaming video service. An assessment of the work is presented, in order to show its advantages when it is compared with traditional manual processes and other previous systems.

1 Introduction

Network and service management is a part of the business core for a telecommunication operator. It involves different processes; troubleshooting is a critical one. Network operation is rather expensive and so, a main concern for all the players of this market. On the other hand, operation excellence strongly affects the customer experience, and thus, may have impact in its retention.

Diagnosis automation supported by artificial intelligent techniques can help to reduce operation expenditure, avoiding unnecessary human intervention. In the case of fault management, automation means to be able to detect, diagnose and repair possible problems in the system. The main objectives of these fault management automation are scalability and ability to infer with incomplete and inaccurate information. The approach proposed in this paper faces these problems with a multi-agent architecture which uses distributed Bayesian inference as reasoning mechanism. Multi-agent paradigm is a very common approach in diagnosis systems [5, 10, 1, 4] and cognitive networks [2, 8].

Álvaro Carrera · Mercedes Garijo
Universidad Politécnica de Madrid, Madrid, Spain e-mail: a.carrera@dit.upm.es, mga@dit.upm.es

Javier García-Algarra · Javier González-Ordás · Pablo Arozarena
Telefónica I+D, Madrid, Spain e-mail: algarra@tid.es, javiord@tid.es, pabloa@tid.es

This work extends previous results [3] where a multi-agent system with Bayesian reasoning was proposed for fault diagnosis in Virtual Private Networks. This article presents how the system has evolved for a home scenario and which aspects have been improved.

To properly frame this study, a P2P streaming scenario was chosen. In this scenario, there are a multimedia provider user and a multimedia client user. Many faults may occur both in connection and in services. The system is designed to provide to an end-user or an operator the result of the diagnosis made upon receipt of a notification of a symptom of failure. The result is expressed in percentages representing the certainty of the occurrence of a given hypothesis.

2 Multi-agent Architecture

This section describes the architecture which has been designed for dealing with fault management in a scenario as presented above. One of the main design goals of this architecture has been the integration of capabilities in order to manage uncertainty. To handle it, diagnostic agents take their decisions based on a Bayesian network which models uncertainty about network faults, as described in sect. 3.1.

The following agent types have been identified:

Bayesian Agents offer reasoning capability to the system. There are two types: Diagnosis Agents and Belief Agents. *Diagnosis Agents* are the main responsables of the diagnosis. They request and recollect evidences or beliefs to infer the root failure cause. Their behavior is controlled by a Bayesian network that models the information about the scenario. *Belief Agents* offer their expert knowledge about different domains (like one Home Area Network). Each belief agent can be expert in a section of the network, in a concrete service, etc.

Scenario Agents offer real-time information required to know the status of the scenario. There are two types: Monitor agents and Observation agents. *Monitor agents* detect symptoms in the scenario and notify to the appropriate diagnosis agent. *Observation agents* offer services to know the status of scenario components, devices, services, etc.

Support Agents offer a support layer to all other agents. Support agents are classified as Yellow Pages Agent, Knowledge Agents and Storage Agent. *Yellow Pages Agents* offer to all agents a directory to search services or agents and their capabilities. *Knowledge Agents* offer a way to generate and deploy new knowledge in the system on the fly. They execute the Expectation Maximization (EM) learning algorithm [7] to improve the performance of the system and deploy appropriate knowledge in Bayesian agents. *Storage Agents* offer a service to store diagnosis in database to perform self-learning process.

In sake of reusability, most of the agents listed in this section are completely generic. Only scenario agents should be implemented to be adapted.

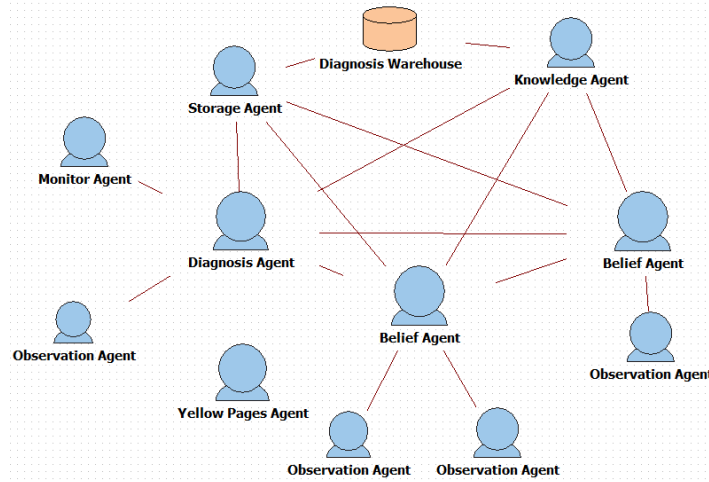


Fig. 1 Agents Hierarchy

Figure 1 shows the interactions between agents in the scenario presented in sect. 1. To clarify the diagram, the interaction with Yellow Pages Agent has been omitted.

3 Bayesian inference and learning

Bayesian agents of the system perform two types of inferences:

Centralized inference: The system performs centralized Bayesian inference when request or receive observations or evidences. Every time an agent receives new environment information, it performs local inference. After this, when an agent has resolved all their possible local actions; it could request a set of beliefs to other Bayesian agent, then this pair of agents performs Distributed Bayesian inference.

Distributed inference: When an agent receives a request of a set of beliefs, it should update its own Bayesian network (more exactly, its Bayesian inference engine) to share beliefs using the method described in section 3.1.

3.1 Distributed Bayesian Reasoning

In this study, a distributed way to perform Bayesian reasoning is presented. This system performs Bayesian inference to deduce the root failure cause using a multi-agent architecture, as it is shown in the previous sections (see sect. 2). As the complexity of the diagnosis scenarios grows, scalability may become a problem. To support this, the system is able to distribute the inference process in several smaller Bayesian

networks [9] to delegate parts of diagnosis in agents specialized in different problems, regions, services, etc. As mentioned above, the inference process combines two inference strategies. This section details further each one.

Centralized inference is a method in which some kind of evidences or observations are used to calculate the probability of a hypothesis to be true. Distributed inference adds to this method the chance to use beliefs instead of observations. In other words, centralized inference uses as inputs discrete observation (for example true or false, yes or no, etc.), but distributed inference uses uncertain observations as inputs too (for example, true with confidence equals to 0.65). Thus, beliefs (or uncertain observations) can be shared and propagated remotely to perform distributed inference.

The method used to perform distribute inference is a variation of the Virtual Evidence Method (VEM) algorithm [6]. This method is a simple way to set a desired belief in a target node. It consists in adding a child node to any other node (Target node) in the Bayesian model and using equation 1 (see below), thus getting control of the target node. To reach the desired value of the target node, the conditional probability table of the child node should be modified following the instructions shown below. These changes in the model and in the probability tables are performed in execution time by system agents.

This child node should have a fixed structure, two states: trigger state (TS) and non-trigger state (NTS). Conditional probability table of child node should be filled with the following equation:

$$P(ChildNode = TS|TargetNode = SX) = \frac{P_{desired}(TargetNode = SX)}{P_{current}(TargetNode = SX)} \cdot P(aux) \quad (1)$$

In this equation, there is one degree of freedom, $P(aux)$, that is used to assure the probability is always less than or equals to 1.0. This equation should be instantiated one time per one state of target node. After this process is finished, trigger state is added as an observation to child node, inference is performed and target node has the desired belief in their states.

In this study, flexibility offered by this method is used to perform distributed diagnosis reasoning. During the diagnosis process, agents perform centralized inference until all their possible actions have been solved. Then, they share their beliefs about several hypotheses with other agents that are able to continue with diagnosis process in other relevant areas, in other words, one agent delegates part of the diagnosis in other agents that are specialized in some relevant hypotheses and areas of knowledge.

4 Evaluation

A testbed has been built to evaluate the performance of the system. This testbed simulates the real scenario thus allowing complete management of all systems involved in the study case (see the last paragraph of sect. 1).

In this testbed, to access device configuration and network or service information, several external tools have been used (monitoring and management tools).

We have tested the diagnosis system causing connectivity problems, jitter, dropping packages, configuring the service with wrong data, etc. Then, we evaluate if the system reach valid conclusions and how many time waste in each diagnosis.

Some relevant results are shown below:

Table 1 Diagnosis Results Ranking

Diagnosis Confidence	Diagnosis Result	
	Correct	Incorrect
Reliable	51%	0%
Likely	26%	7%
Uncertain	8%	8%

Table 1 shows the diagnosis results of the system. Each row represents the following confidence: reliable (0.9 - 1), likely (0.3 - 0.9) and uncertain (0.01 - 0.3). These results can be improved using more data as input in the self-learning process. Actually, we are recollecting more data from testbed to generate more accurate Bayesian networks.

Table 2 Diagnosis Time Ranking

Diagnosis duration (in seconds)	Percentage
10-30	79%
30-50	15%
other	6%

Table 2 shows the time wasted since a symptom is detected to diagnosis is finished. Comparing this work with previous works [3], now the system improves the diagnosis time comparing with the previous approach and the system delegates portions of diagnosis in several physical places.

The system wastes less time because the agents interchange less messages through the core network using the beliefs request capability offered by the distributed Bayesian inference, and delegating portions of diagnosis offers extra flexibility to access to private regions or domains (like Home Area Network), because these portions of diagnosis can be offered like services and consumed externally from any diagnosis agent.

5 Conclusions and Future Work

In this paper, we have demonstrated how a multi-agent system with distributed Bayesian reasoning for network fault diagnosis can be applied to a relevant P2P network scenario. One of the most relevant results are the flexibility of the architecture, that can be adapted easily to other scenarios. Furthermore, the use of probabilistic approaches provides an answer even when the system has barely environment information. And finally, the scalability is provided by the multi-agent paradigm and by the distributed Bayesian inference.

In the future, we plan to further explore some of the challenges identified. One of them is to improve the parallelism between agent tasks to improve the system performance. Another is to add autonomic recovery capabilities to the system, once a conclusive diagnosis has finished.

Acknowledgement

This research has been funded by the Spanish Ministry of Science and Innovation through the project and T2C2 (TIN2008-06739-C04-03/TSI).

References

1. Dheedan, A., Papadopoulos, Y.: Multi-agent safety monitor. hull.ac.uk (2003) (2009)
2. Fortuna, C., Mohorcic, M.: Trends in the development of communication networks: Cognitive networks. *Computer Networks* **53**(9), 1354–1376 (2009)
3. García-Algarra, F.J., Arozarena-Llopis, P., García-Gómez, S., Carrera-Barroso, A.: A lightweight approach to distributed network diagnosis under uncertainty. In: INCOS '09: Proceedings of the 2009 International Conference on Intelligent Networking and Collaborative Systems, pp. 74–80. IEEE Computer Society, Washington, DC, USA (2009)
4. Hongyan, S., Xuefeng, J.: Study on large-scale rotating machinery fault intelligent diagnosis multi-agent system. *IEEE* (2008)
5. Lee, G.J., Clark, D.D., Smith, A.C.: CAPRI: A Common Architecture for Distributed Probabilistic Internet Fault Diagnosis. Ph.D. thesis, MIT (2007)
6. Pan, R., Peng, Y., Ding, Z.: Belief update in bayesian networks using uncertain evidence. 2006 18th IEEE International Conference on Tools with Artificial Intelligence ICTAI06 (3), 441–444 (2006)
7. Peña, J.: An improved bayesian structural EM algorithm for learning bayesian networks for clustering. *Pattern Recognition Letters* **21**(8), 779–786 (2000)
8. Smart, P.R., Sieck, W., Braines, D., Huynh, T.D., Sycara, K., Shadbolt, N.R.: Modelling the dynamics of collective cognition: A network-based approach to socially-mediated cognitive change (2010)
9. Xiang, Y., Poole, D., Beddoes, M.P.: Multiply sectioned bayesian networks and junction forests for large knowledge-based systems. *Computational Intelligence* **9**(2), 171–220 (1993)
10. Zhang, G., Huang, S., Yuan, Y.: The Study of Elevator Fault Diagnosis Based on Multi-Agent System. *IEEE* (2009)