

Expert System to Troubleshoot the Wireless Connection Problems

Dr. Raja'a A. Khalid[#], Rafah O. Jassim[#]

[#]Al-Nahrain University, Iraq / Baghdad

Abstract— Expert system is an important method to help using the expertise of the human in more than one place in high demanding fields. In this paper, an expert system was designed to troubleshoot the wireless connection problems and reach an accurate decision about the cause. Java and Prolog used connected by JIProlog. A survey investigates the wireless connection problems performed by about 40 expert, and a case study performed in Al-Nahrain University campus to test the system.

Keywords— Expert system, Wireless network, Java, Prolog, JIProlog.

I. INTRODUCTION

Artificial Intelligence (AI) is the field that seeks to "build systems that exhibit intelligent behaviour and perform complex tasks with a level of competence that is equivalent or superior to the level currently exhibited by human experts" [1]. Expert systems are one of the subfields of artificial intelligence. Since the 1960's, when development of artificial intelligence began, expert systems have been utilized to help mankind with problems within a limited scope. An Expert System is an interactive computer based decision tool that uses both facts and heuristics to solve difficult decision making problems, based on knowledge acquired from a human expert, i.e., an expert system is a computer program that simulates the thought process of a human expert to solve complex decision problems in a specific domain [2], [1].

Intelligent systems have been developed for a variety of domains, such as troubleshooting of electrical and/or mechanical equipment, identification of software/hardware problems and integrated circuit failures, as well as fault-detection in nuclear power systems. It is desirable for the intelligent system to identify the possible causes that could explain the symptoms and propose suitable solutions. In order to perform the tasks, an intelligent system must collect adequate domain knowledge constantly and dynamically emulates human reasoning and decision makings based on the most updated knowledge [3]. Expert systems are typically very domain specific. The developer of such a system must limit his or her scope of the system to just what is needed to solve the target problem. Special tools or programming languages are often needed to accomplish the specific objectives of the system [1].

Expert systems offers the benefits of increase the probability, frequency, and consistency of making good decisions, help distribute human expertise, facilitate real-time, low-cost expert-level decisions by the nonexpert [1]. Expert systems have a number of major system components

and interface with individuals in various roles. The major components are (as illustrated in Fig. 1):

- A declarative representation of the expertise, often in IF THEN rules, and it is known as (Knowledge base).
- The data that is specific to a problem being solved, which is referred as (Working storage).
- The (Inference engine), which is the code at the core of the system that derives recommendations from the knowledge base and problem-specific data in working storage.
- The (User interface), and it is the code that controls the dialog between the user and the system [2], [4].

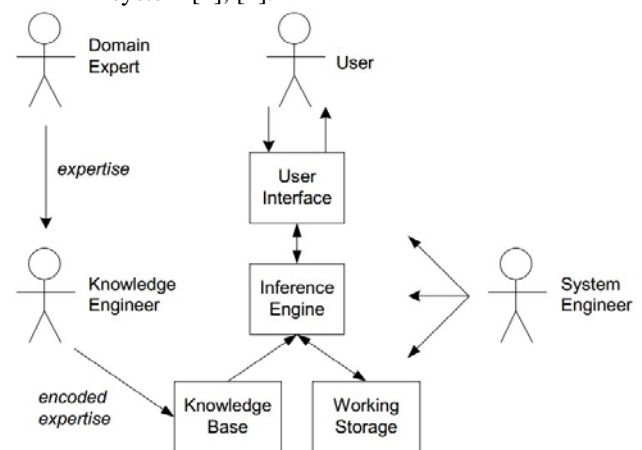


Fig. 1 Expert system components and human interfaces [4]

II. PROBLEM DESCRIPTION

With the increasing deployment of in-building wireless networking solutions using Wi-Fi and the increasing penetration and use of Wi-Fi enabled devices, including smart phones, tablet PCs smart TVs, games consoles and security systems, the demand on Wifi connection increased rapidly. In 2012, Cisco estimated that 28% of mobile data traffic indoors was carried using Wi-Fi. This is predicted to rise to 48% by 2017. A more recent report commissioned by the European Commission identified that over 71% of all wireless data traffic that was delivered to smart phones and tablets in the EU was delivered via Wi-Fi [5].

III. METHODOLOGY

The expert system mainly consists of three parts: The knowledge base, the inference engine, and the user interface. In order to interpolate the expertise into the system, PROLOG is used as the knowledge base. PROLOG is considered to be a declarative programming language, i.e. when implementing the solution to a problem, instead of specifying how to achieve a certain goal in a certain situation, we specify what the situation (rules and facts) and the goal (query) are, and let the Prolog interpreter derive the solution for us, this feature made Prolog widely used in AI system compared to C and C++ [8]. Likewise, Java is used as user friendly interface that explain the available options to the user and show the results after reaching the decision. Choosing Java was due to its effectiveness in implementing the user interface, and its platform independency [9]. On the other hand, JIProlog (Java Internet Prolog) is used to communicate between the Java and the Prolog. JIProlog enables calling Prolog predicates from Java without dealing with native code (JNI) and it allows invoking Java methods from Prolog in the same way you call predicates. JIPEngine was used, which is the main class, and it supplies the methods to initialize the Prolog interpreter, submit and manage queries, consult/unconsult files, and to set several interpreter properties. In addition, JIPQuery, JIPSyntaxErrorException, JIPTerm, JIPTermParser, and JIPVariable classes have also been used. The queries submitted synchronously, and then the query exits when the goal is satisfied or when it fails. In other words, the method waits until the interpreter has finished its work [10]. The full system flowchart is shown in Fig. 3, the system starts by initializing the Java interface and prepare the first question for the user by communicating the Prolog database via JIProlog. When the user answers the question, the Java consults the database (the flowchart in Fig. 2 to decide the next available question. The user will be notified with the decision that will be reached according to the available data.

IV. RESULTS

This section describes the results of building the algorithm of the expert system based on the flowchart shown in Fig. 3. Netbeans software used to build the Java user interface and the connection with the Prolog via the JIProlog as described earlier in section III. On the other hand, SWI-Prolog is used to build the Prolog database using the problem description flowchart shown in Fig. 2, SWI-Prolog used as it is open source and easy to use. JIProlog use synchronous call in which the Java consult the Prolog on every single answer.

The main Java will prepare a main window to ask the user the database questions as shown in Fig.4, the result will be dilated as shown in Fig. 5. The process reinitialized if the user have other enquiries, otherwise, the software aborted.

A case study performed in the networking laboratory of the college of information engineering of Al-Nahrain University (Iraq / Baghdad) using 48 devices to be connected for the 2.4 GHz Wifi network in the lab (the devices contain 35 PCs, and 13 laptops). 79 % of the devices had connection problems were (given that some devices didn't have the wireless security code). The study shows a success rate of 85 % in the devices that used the system.

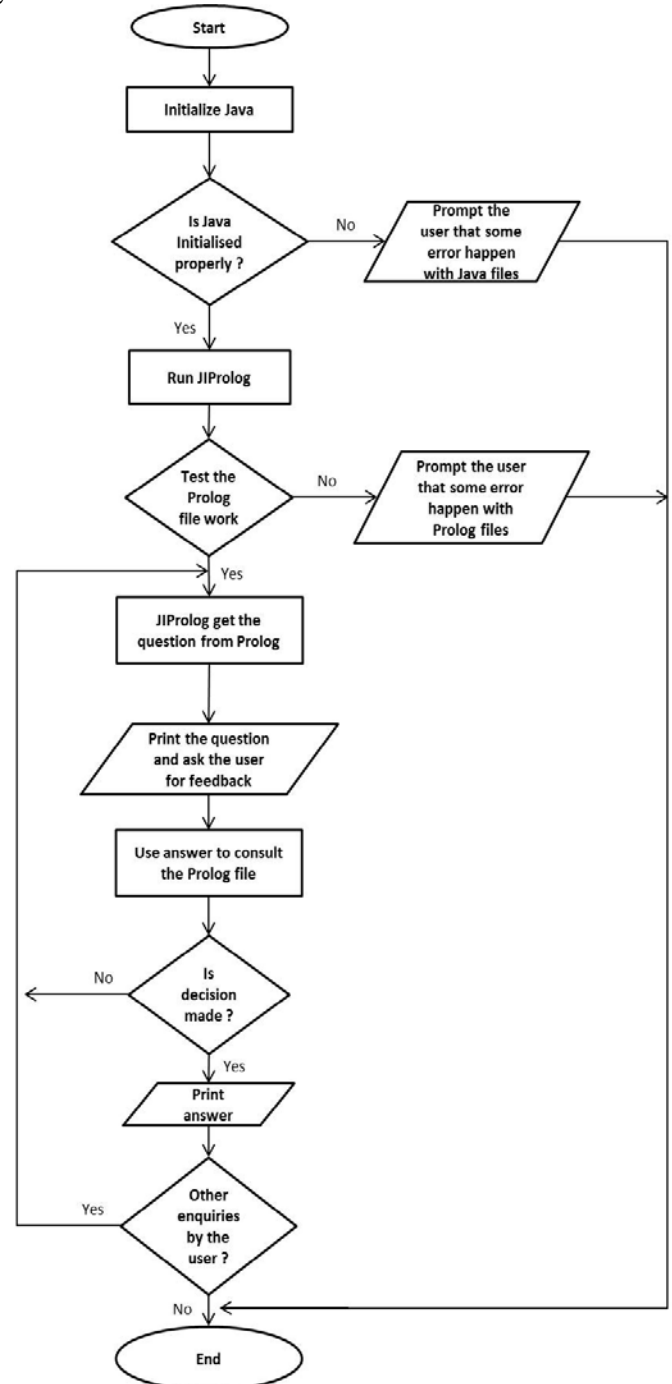


Fig. 3 Algorithm complete flow chart

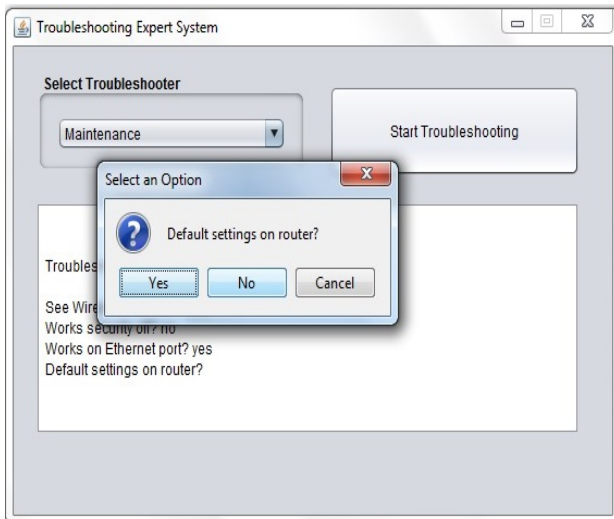


Fig. 4 Java Interface display

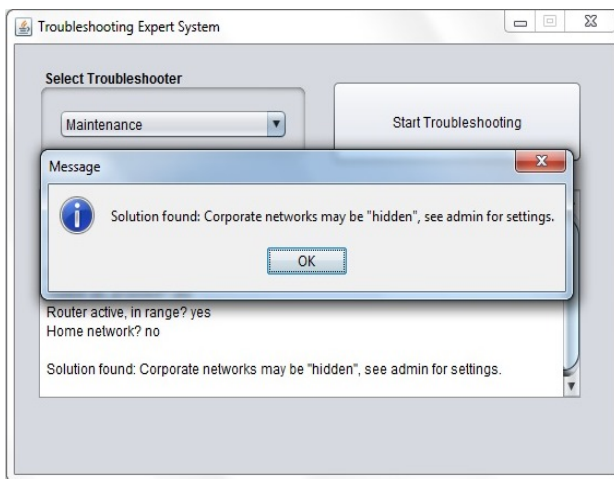


Fig. 4 Result display

V. CONCLUSIONS

The number of devices that use the wireless connection is increasing, and maintaining the wireless connection for the devices will need extreme efforts. In this paper, an expert system that uses the expertise of more than 40 experts in maintaining the wireless connection after a survey done in the Iraq has been designed. A case study on 48 devices performed in the laboratories of the college of information engineering in Al-Nahrain University/Baghdad. The results show a success rate of 85% out of the 79% of the devices that faced connection problems.

ACKNOWLEDGMENT

We would like to thank the college of Information Engineering for providing the college facilities to perform the case study.

REFERENCES

- [1] A. B. Badiru and J. Cheung, *Fuzzy engineering expert systems with neural network applications*, vol. 11. John Wiley & Sons, 2002.
- [2] S. Deepthi, S. Rao, N. Kumar, and S. Naidu, *Troubleshooting wireless mesh networks using rule-based expert system*, IJECT, vol. 2, pp. 22–25, Jun 2011.
- [3] Y. Hao, A. Trappey, C. Ku, G. Lin, P. Ho, and B. Hung, *Develop an intelligent equipment maintenance system using cyber-enabled jess technology*, in *Mechatronics, 2005 ICM'05. IEEE International Conference on*, pp. 927–932, IEEE.
- [4] D. Merritt, "Building expert systems in Prolog," 2000.
- [5] J. Marcus and J. Burns, "Study on impact of traffic off-loading and related technological trends on the demand for wireless broadband spectrum," wik-Consult sur mandat de la DG CONNECT, UE, 2013.
- [6] L. Ericsson, "More than 50 billion connected devices," 2011.
- [7] M. Rosenthal, *The Laptop Repair Workbook: An Introduction to roubleshooting and Repairing Laptop Computers*. Foner Books, 2008.
- [8] U. Endriss, "Lecture notes an introduction to prolog programming," 2007.
- [9] S. R. El-Beltagy, M. Rafea, and A. Rafea, "Practical development of internet Prolog applications using a java front end," in *The 2nd International workshop on logic programming tools for Internet applications*, in conjunction with ICLP97, 1997.
- [10] U. Chirico, "JIProlog 3.0.3 reference manual," 2007.