Glass-Box: An intelligent flight data recorder And real-time monitoring system

Krishna M. Kavi and Mohamed Aborizka
Department of Electrical and Computer Engineering
The University of Alabama in Huntsville
Huntsville, AL 35899
Voice: 256-890-6380; FAX: 256-890-6803
email: kavi@ece.uah.edu

1. Introduction

In this paper we describe a new approach to the design of next-generation flight data recorders (FDRs or Black boxes) and real-time monitoring of airplanes using recent information, artificial intelligence, learning and network technologies. In this new system, ground-based *intelligent software agents* communicate with their on-board counterparts to continuously collect flight data. The data collected from numerous flights can be correlated and *data-mined* to construct scenarios that could lead to unsafe incidents. Such analyses can be utilized by the intelligent software agents to detect potentially unsafe conditions in real-time and provide early warning to the pilot.

Over 90 years ago, philosopher George Santayana said, “those who cannot remember the past are condemned to repeat it”. Although modern aircraft designs rely on past failures to improve their safety, there exist no comprehensive data that can be used to correlate and track the complex set of events that may lead to an accident. According to NTSB annual review of aircraft accident data, a total of 1,935 U.S registered general aviation aircraft were involved in 1,907 accidents during the calendar year 1996. Of these, 360 accidents (involving 366 aircraft) resulted in fatalities. NTSB also concluded that pilots were either the sole cause or a contributing factor in 77 percent of all accidents and 83 percent of fatal accidents. It is our contention that the pilot is often unable to recognize potential problems early enough to take preventive measures, or the pilot is provided with insufficient time to explore alternate actions. This is partly because of the complex set of events leading to the problem and partly because the pilot is furnished with very limited information that is based only on the current flight (with no benefit of correlating the information to similar problems in other flights). Even the most experienced pilot, sometimes, may not be able make appropriate decisions under such time-critical conditions. Our intelligent agent based system will be able to track potential problems in real-time and provide early warning to the pilot. Data mining of previously collected data both from normal flights and accidents can form the basis for suggesting safe actions that can prevent accidents. Even in case of mechanical failures, our real-time data collection and monitoring can aid in the detection of failures or the identification of maintenance schedules.

2. Background and Related Research

Our project uses and extends the state-of-the-art research in the design of Flight Data Recorders, learning and artificial intelligence, software agent based programming paradigms, multi-agent based control systems, and real-time monitoring. In this section we describe briefly these related research areas.

2.1. Fight Data Recorders

Flight Data Recorders (FDR) were first introduced in the 1950s. The purpose of an airplane FDR system is to collect and record data from a variety of airplane sensors onto a medium designed to survive an accident. Depending on the age of an airplane, the FDR system may consists of an analog or a digital flight data acquisition unit and a digital recorder. FDR may have a tape or solid-state memory. The data collected in the FDR system can help investigators determine whether an accident was caused by a pilot error, an external events (such as weather), or an airplane system (such as mechanical or electrical failures). The first generation of FDRs recorded only five parameters including acceleration, air speed, compass heading, time and altitude. This data was embossed onto steal foil, and was capable of recording for 400 hours at which time the foil must be replaced. The second generation FDRs introduced in the 1970s aimed to collect more data but they were unable to process the incoming sensor data in a timely manner. This led to the development of a separate flight data acquisition unit (FDAU). The FDAU processes sensor data, digitizes and formats it for transmission to the FDR. The second generation of digital FDR (DFDR) used (magnetic) recording tape. Most DFDRs can record and store up to 18 operational parameters for a period 25 hours. Third generation FDRs (known as SSFDR) were introduced in 1990 and used solid-state technologies for recording data. These recorders can record up to 256 operational parameter for 25 hours [2].

2.2. Knowledge Discovery and Data Mining

The amount of data being collected by FDRs exceeds the ability of human beings to analyze without the aid of automated analysis techniques. Data mining is an analysis process used to explore large data volumes to find consistent patterns and/or systematic relationships among variables. The knowledge discovery process takes the raw results from data mining and transforms them into useful and understandable information by applying various artificial intelligence techniques. Learning algorithms are an integral part of knowledge discovery [3]. Machine learning is one of the artificial intelligence techniques that has contributed to knowledge discovery [4]. There are several knowledge discovery techniques in use, including:

- Probabilistic and statistical techniques.
- Bayesian classification, inductive logic, data cleaning/pattern discovery, and decision tree analysis.
- Genetic algorithms.
- Neural networks.
- Hybrid approaches that integrate two or more of these techniques [5].Probabilistic and statistical techniques.
- Bayesian classification, inductive logic, data cleaning/pattern discovery, and decision tree analysis.
- Genetic algorithms.
- Neural networks.
- Hybrid approaches that integrate two or more of these techniques [5].

2.3. Software Agents

Intelligent agents are autonomous software entities that can navigate heterogeneous computing environments. They achieve their specified goals either alone or by working with other agents [6]. They often use intelligence and learning abilities to achieve their tasks and perform their assigned tasks autonomously. Agents may also interact with their environment (or other agents) and adapt dynamically to changing conditions. Intelligent agents are increasingly being used in a wide variety of applications, ranging from comparatively small systems such as email filters, to large, open, complex, mission critical systems such as air traffic control [7].

The programming paradigm that uses software agents differs drastically from other models used in distributed processing, including remote procedure calls and client-server models. The difference stems from the autonomy of software agents. For example, in classical client-server architecture, the server provides specific services to one or more clients (both servers and clients are computer systems). In agent systems, however, an agent can provide services at one time, adopting the role of server, while requesting services at other times, acting as a client. In addition, agents possess autonomy and thus differ from classical servers or clients. Even when donning the role of a server, an agent will be an active participant in a client-server system and adapts to changing requirements and resources. The advantages of the new programming paradigm are numerous, including reduced network traffic (an agent need not maintain a continuous contact with its creator), tolerance of frequent network disconnection (agent can contact its creator when the connection is available), migration of tasks to data sources (this is achieved because of the autonomy), asynchronous processing, and decentralized structure [8].

2.4. Intelligent Multi-Agent Based Systems

Using a single agent to solve a complex problem may not always be feasible. In complex systems, an individual agent may be required to gather and analyze enormous amounts of knowledge. This may lead to situations where an agent may not be able to find any solutions to a given problem. Fortunately, most complex systems are distributed in nature and lend themselves to the use of multiple distributed agents. In multi-agent based systems, the knowledge needed to solve a problem is not available with a single agent, but distributed among the various agents. Each agent of a multi-agent system can either follow its own objectives and only contact other agents to obtain information, or contribute to a coordinated solution of the overall problem. Every individual agent is associated with a precisely defined task that is particularly suitable to the agent and whose solution is within the agent's capabilities. This task decomposition and distribution permits the processing of very complex problems. Existing single agent based systems can easily be integrated into a larger multi-agent based system, permitting incremental development of intelligent systems. Distributed problem solving with multi-agents systems is appropriate only when agents in the system are capable of communicating and cooperating with each other [8][9][10]. Due to the complexity of the systems invested by our research, multi-agent based approach will be utilized in our designs.

2.5 Multi-Agent Based Real Time Systems

Real-time systems present numerous challenges to a designer. They often have many stringent requirements on processing, timely response, power and size specifications. Certain r-al time systems may benefit from multi-agent based technologies, since multi-agent systems facilitate adaptability, reactivity and autonomy. Some of the multi-agent-based real-time systems include [11]:

- Surveillance systems.
- Network management systems.
- Manufacturing processing and monitoring systems.
- Multimodal interfaces.
- Remote cooperative work.
- Interactive television.

2.6. Real-time monitoring

Real-time monitoring is utilized in many critical systems such as nuclear reactors and air traffic control systems. In many cases, monitoring systems check for exceptional (or data outside of safe regions) conditions. In complex systems, the detection of exceptional conditions may not be sufficient to prevent potential accidents. It is necessary to correlate the conditions with complex interactions among the subsystems. The correlation may be in the form of defining a finite-state model for the system. The states of the model will track the system both in operational (good and safe) and problem (unsafe) conditions. Such models can then be utilized to define the set of state transitions that can lead to accidents. These models can be very detailed involving thousands of states. However, caution should be exercised to balance the number of states and the processing needed for real-time monitoring. While more states allow for more accurate monitoring, they also require more processing.

3. Glass Box – An intelligent flight data recorder and monitoring system

3.1. Multi-Agent Based FDR

We propose to design a multi-agent based FDR and monitoring system. In this section we outline a framework for such a system. Our goal is develop a system that can help a pilot make appropriate decisions using the knowledge base generated from data collected from other flights, the experience of pilots and other experts. In a full-fledged system, data from both normal flights and flights with problems will be collected and mined off-line for any patterns (or sequence of events) that identify unsafe situations. The real-time monitoring system can look for such patterns during a flight and provide the pilot with early warnings. Figure 1 shows our preliminary multi-agent based FDR architecture. It consists of three layers: the agent layer, the coordinator layer, and the knowledge base layer. The agent layer defines all the necessary agents including a data acquisition agent(s), agents to model the various subsystems of an aircraft, a decision agent and an agent to interface with the pilot.

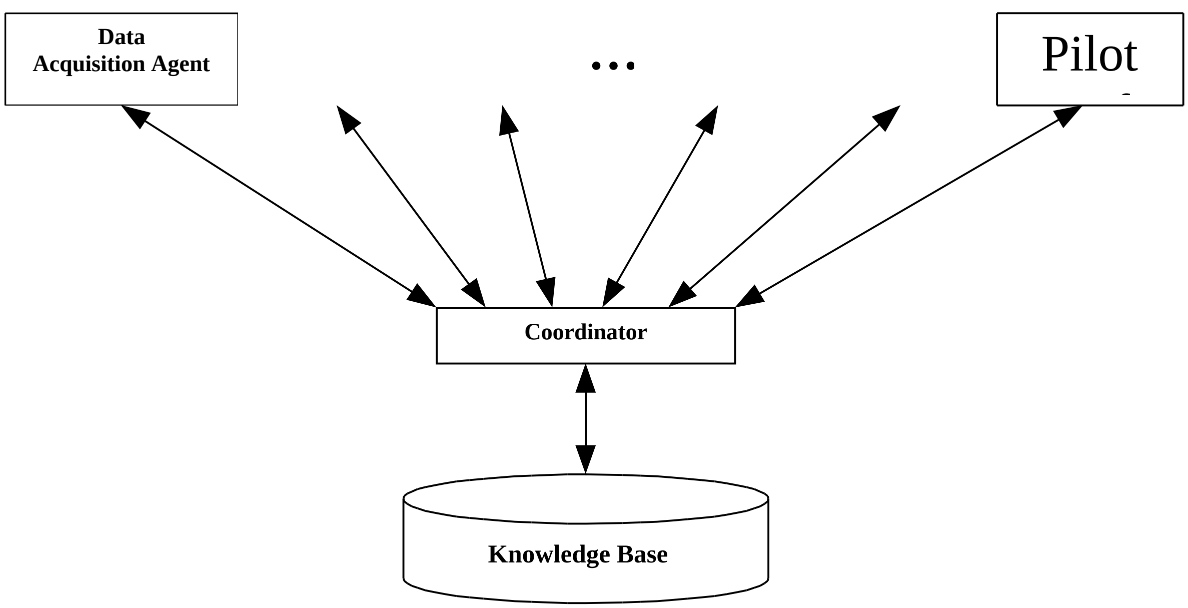

Figure 1: Architecture of Glass Box

Only the data acquisition agent(s) and the agent that interfaces with the pilot are resident on the aircraft itself, while the other agents are located at ground based computing systems. The coordinator layer manages the communication between different agents. The knowledge base layer stores common knowledge used by the different agents and the coordinator. The functions of the various functions is described below.

- The data acquisition agent(s) receives data from FDAU and also checks for any exceptional changes.
- The tracking agent continuously checks the communication between agents on the plane and agents on the ground station server. In the full-fledged implementation, the tracking agent may migrate from one ground-based system to another to maintain a constant contact with the agents onboard.
- The electrical, mechanical, …, agents are responsible for the detection of potential unsafe conditions in various aircraft subsystems.
- The decision agent is responsible in arriving at a final decision based on the individual decisions of the various agents. All agents utilize the knowledge-base and finite state models for reaching their decisions.
- The pilot interface agent displays warning messages and suggest appropriate actions to the pilot [12].

3.2. Glass Box in Action

Figure 2 shows a scenario for our multi-agent based FDR and monitoring system. As we mentioned above, only the data acquisition agent(s) and the agent that interfaces with the pilot are resident on the aircraft. All the other agents will be running on a ground-based computing server. This approach minimizes changes to current FDR designs and aircraft computer systems. All the major agents (data acquisition agent, interface agent, coordinating agent, decision agent, tracking agent) will be launched as an airplane starts its flight. As the data acquisition agent receives data from FDAU and detects any exceptional changes, the agent will transmit the changes to the coordinating agent. The coordinating agent may decide to launch specific subsystem agents (e.g., mechanical or electrical agent) to check the knowledge base for potentially problematic patterns. During this analysis, an agent may request for additional flight data. This request is communicated to the data acquisition agent onboard. *Note that such dynamic data collection is not feasible with current FDRs.* The results (or decisions) from these various agents are communicated to the decision agent, which takes a final decision and communicates the decision to the interface agent. The interface agent in turn provides necessary information to the pilot.

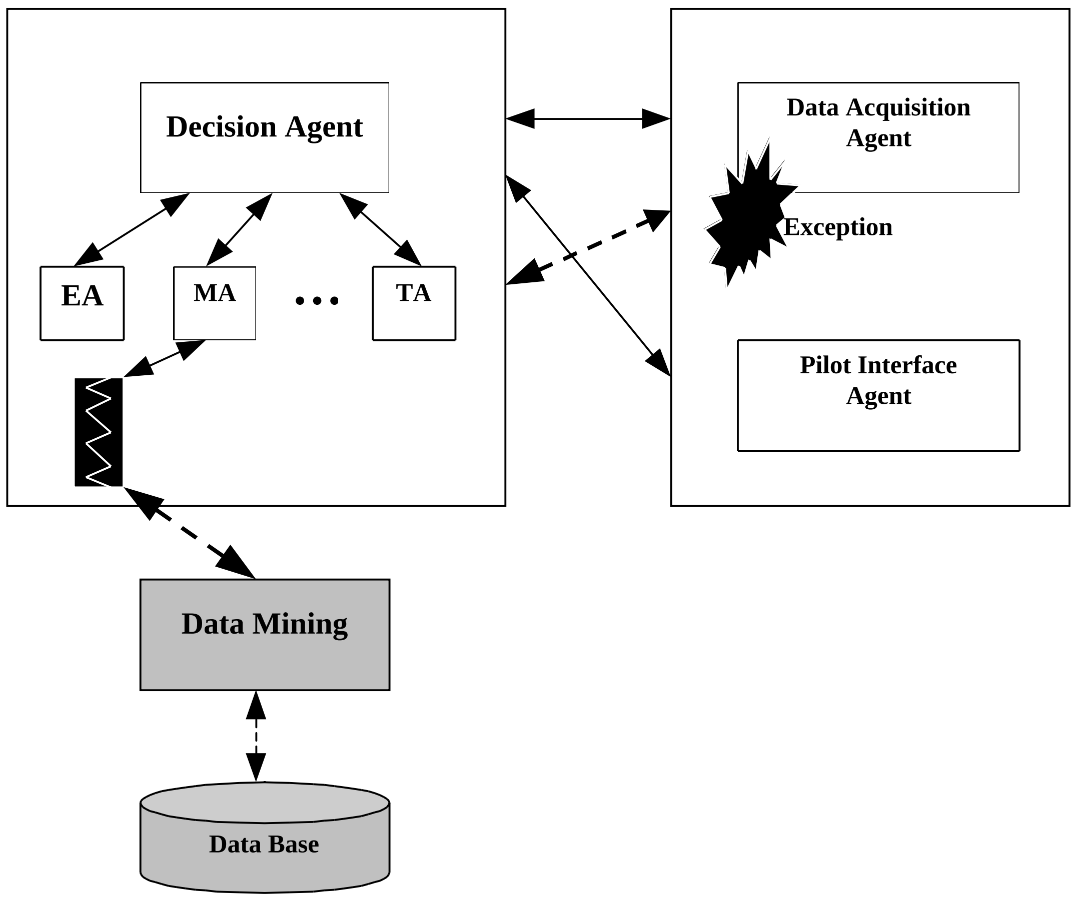

Figure 2: Glass Box in Action

3.4. Feasibility of the Proposed Design.

There are several requirements that must be met before our proposed intelligent FDRs and monitoring systems become reality. Since the aircraft resident agents must communicate with ground-based agents on (potentially) a continuous basis, a data communication network that spawns the globe is needed. However such a network is already being designed and will be operational in the near future. The ATN (Aeronautical Telecommunications Network) is a worldwide (digital) data network intended to provide a full and flexible support for seamless connectivity among mobile units, aircrafts, ground based computer systems and other service providers like the Aeronautical Operational Control (AOC), Aeronautical Administrative Communications (AAC), and Aeronautical Passenger Communications (APC), government authorities that provide Air Traffic Control (ATC) and flight information services (FIS). This network allows a collection of dissimilar transmission networks and interconnecting computers to operate as a single, cohesive, and virtual data network. Figure 3 shows the ATN infrastructure [13]. Our preliminary computations indicate that the data transmission requirements of the proposed intelligent FDR and monitoring systems are well within the capabilities of the ATN infrastructure.

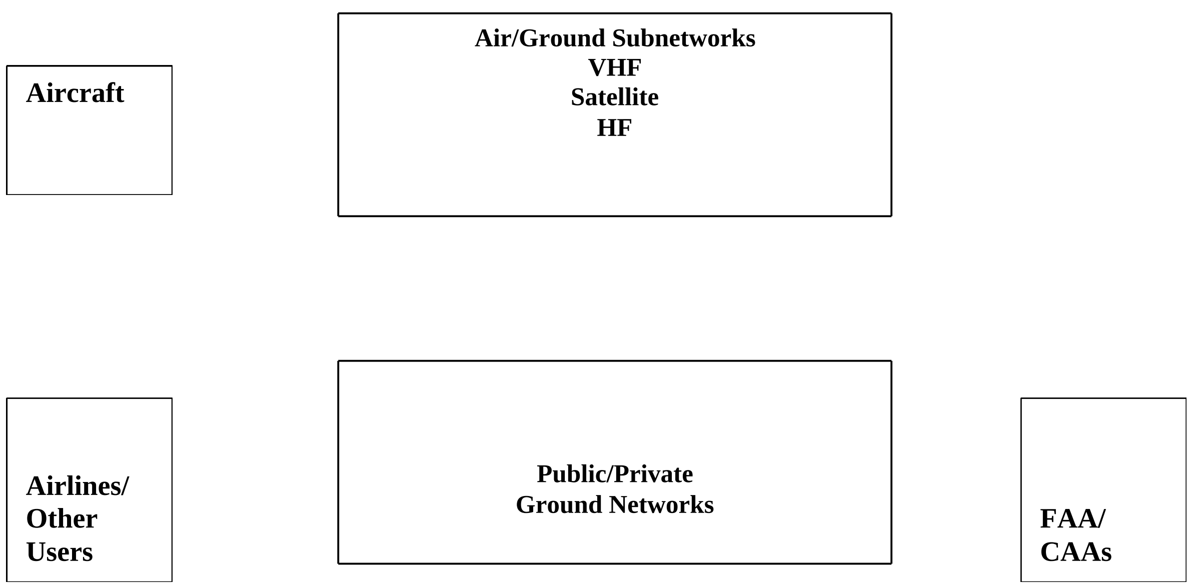

Figure 3: Aeronautical Telecommunications Network

As described in Section 2, the technologies for the development of intelligent and learning agents and data mining of large amounts of data already exist. Our initial analyses also indicate the modern computing systems are more than adequate to perform (off-line) data mining computations on the flight data collected.

4. Summary and Future Work

In this paper we described an approach for the design of intelligent FDRs and monitoring systems. Our approach uses a multiple agents to monitor the data collected by FDRs in real-time and use knowledge bases to detect potentially unsafe conditions. We feel that there are numerous advantages with our proposed design. We list a few of these advantages below.

- Currently, it is often difficult to locate FDRs after a crash, and located black boxes are often damaged. In the proposed system, there is no need to locate FDR after a crash since flight data is transmitted to ground-based computing systems in real-time. Real-time recording of data can also detect misreading or conflicting data due to failures of the FDR itself.

- Using flight data from several hundreds or even thousands of flights, it is possible to more accurately predict scenarios that may cause unsafe conditions. At present such analyses are not conducted.

- Using above analyses, it will be possible to develop intelligent agents to detect unsafe conditions in real-time to provide early warnings to the pilot. The information gathered from data mining can also be used to design safer aircrafts since it is now possible to identify sequences of events or combinations of factors that can cause unsafe situations. Traditional approaches to the design of control systems require precise mathematical models; however, such models may not be possible or difficult for describing complex interactions that may occur during a flight. Results from data-mining analyses can be used to develop fuzzy logic based control systems that can adjust to a variety of factors including weather-related, geographical data and individual pilot's judgments.

- Software agents can automatically log maintenance requirements using flight data.

We feel that the approach is feasible since all the necessary technology and network already exists. We are currently gathering flight accident data from the National Transportation Safety Board (NTSB) to build initial knowledge base. Using this knowledge base we will build an intelligent agent based system to aid in the accident investigations. A full-fledged intelligent FDR and real-time monitoring systems will be proposed based on the experience gained during this time.

5. References

[1]. Information on Flight data recorders can be obtained from the following web site. http://www.bath.ac.uk/~en8clf/desdev.htm.

[2]. Information on Flight data recorders can be obtained from the following web site. http://www.boeing.com/commercial/aeromagazine/aero_02/s/s01/story.html

[3] Brachman, R.J., and Anand, T. The Process Of Knowledge Discovery In Databases: A Human-Centered Approach. *In Advances In Knowledge Discovery and Data Mining*, eds. U.M. Fayyad, G. Piatesky-Shapiro, P. Smyth, and R. Uthurusamy, AAAI Press/MIT Press, Menlo Park, CA., 1996, PP 37-57.

[4] Fayyad, U.M., Piatesky-Shapiro, G., Smyth, P. From Data Mining To Knowledge Discovery: An Overview. *In Advances In Knowledge Discovery and Data Mining*, eds. U.M. Fayyad, G. Piatesky-Shapiro, P. Amyth, and R. Uthurusamy, AAAI Press/MIT Press, Menlo Park, CA., 1996, PP 37-57.

[5] Knowledge Discovery In Databases: Tools and Techniques. Web Site: http://www.acm.org/crossroads/xrds5-2/kdd.html

[6] Hedberg, S. *IEEE Expert*, v11, n6, 1996.

[7] Applications of Intelligent Agents. 1998. Web Site: http://www.springer.de/comp/lncs/agent/jennings.pdf

[8] Walter Brenner, Rudiger Zarnekow, and Harmut Witting. *Intelligent Software Agents.* Springer Publishing Co. 1998.

[9] Gerhard Weiss. *Multi Agent Systems*. MIT Press. 1999.

[10] Gheorghe Tecuci. *Building Intelligent Agents*. Academic Press Publishing Co. 1998.

[11] Real-time Multi-Agent Systems. 1999. Web Site: http://www.fipa.org/cfp/cfp6.html

[12] Huaiqing Wang, and Chen Wang, Intelligent Agents in the Nuclear Industry, *IEEE Computer* , v30, n11, 1997.

[13] Aeronautical Telecommunications Network. Web Site: http://www.atnsi.com/

[14] Joseph Tardo and Luis Valente. Mobile Agent Security and Telescript. In *Proceeding of COMPCON Spring ‘96*, pages 58-63. IEEE, 1996.

[15] Dag Johanse, Robert van Renesse, and Fred B. Shneider. An Introduction to the TACOMA Distributed System. *Technical Report 95-23*, Department of Computer Science, University of Tromos , June 1995.

[16] Robert S. Gray. Agent Tcl: A flexible and secure mobile-agent system. In *Proceeding of the fourth annual Tcl/Tk Workshop (TCL 96)*, July 1996.

[17] David Wong, Noemi Paciorek, and Dana Moore. Java-based Mobile Agents. *Communications of the ACM*, 42(3):92-102, March 1999.

[18] Danny Lange and Mitsuru Oshima. *Programming and Deploying Java Mobile Agents with Aglets*. Addison-Wesley Publishing Co.1998.

[19] David Wong, Noemi Paciorek, and Tom Walsh. Concordia: An Infrastructure for Collaborating Mobile Agents, *In Proceeding of Workshop on Mobile Agents* MA '97, Berlin, 1997.

[20] Anand R. Tripathi, Neeran M. Karnik, Ram D. Singh, and Tanvir Ahmed. Development of Mobile Agent Applications with Ajanta. *Technical Report*, Department of Computer Science, University of Minnesota, 1999.