ADAPTIVE RISK MANAGEMENT IN DISTRIBUTED SENSOR NETWORKS

Floriano Caprio
Siemens IT Solutions and Services spa, Centro direzionale Collina Liguorini, Avellino, Italy

Rossella Aiello, Giancarlo Nota
Dipartimento di Matematica e Informatica, University of Salerno, Fisciano (SA), Italy

Keywords: Risk Management, Multiagent System.

Abstract: The risk management in a distributed sensor network charged to put environmental variables under control is receiving great attention in recent years. We propose a framework that considers an high level model together with a distributed system based on adaptive agents able to handle the complete risk lifecycle at various levels of responsibility.

The paper first describes the risk modeling problem in a distributed sensor network, then introduces three fundamental agent types: the risk monitoring, the local monitoring and the global monitoring, used to build a network that supports risk management in a distributed environment. Then, the adaptive management of risk exposure is described in terms of a decision process based on a tight cooperation among Local Monitoring Agents.

The framework is general enough to be applied in several appication domain.

1 INTRODUCTION

The problem of risk management is well understood in many fields where a large body of knowledge has been developed to cope with risk exposure. Examples are enterprise risk management (Institute of Risk Management, 2002; COSO, 2004), project risk management (Project Management Institute, 2004; Camara et al., 2006), software risk management (Boehm, 1991; Williams et al., 1997; Han and Huang, 2007) among others.

Even though risk management is a topic that is assuming increasing relevance in many distributed contexts, such as environmental control, health systems , etc. the topic of distributed risk management has received little coverage in literature (Grabowski et al., 2000; Schaller and Vaz, 1997) and there is a need to investigate in such research field.

Due to the different application domain characteristics there is no universally accepted definition of risk, even if the concepts of *uncertainty* (an event may or may not happen) and *loss* (an event has unwanted consequences or losses) are common to many definition of risk (Rosenberg et al., 1999). *Risk exposure*, the fundamental definition of risk management, is then obtained combining the probability of an unwanted event and the amount of loss that arise when the unwanted event happens:

Re = p(unwanted event)* loss(unwanted event)

Many models have been proposed in the literature in order to gain knowledge about the better way to manage risks (Boehm, 1991; Williams et al., 1997; Rosenberg et al., 1999). In this paper we will refer to the model shown in fig. 1 known as “The SEI Risk Management Paradigm” conceived to represent continuous risk management activities that can be applied to any development process. It is a knowledge management model that can be applied in other contexts as well; we borrow the concept of the SEI model to propose a risk model in a distributed sensor network where the survaillance of given environmental variables is required to avoid or mitigate risks.

The purpose of the activities represented in the model is:

Identify: consider risks before they become problems

Analize: convert data into decision-making information

Caprio F., Aiello R. and Nota G. (2008).
ADAPTIVE RISK MANAGEMENT IN DISTRIBUTED SENSOR NETWORKS.
In *Proceedings of the Tenth International Conference on Enterprise Information Systems*, pages 315-320
DOI: 10.5220/0001738703150320

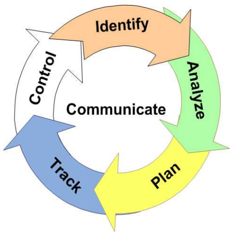

Figure 1: The SEI Risk Management Paradigm.

Plan: decide what should be done about a risk or a set of related risks

Track: acquire risk status and record it

Control: decide for the best reaction when the risks probability increases or when unwanted events happens,

where the communication is a cross-activity in the sense that data or information handled by a certain activity can be communicated to the involved stakeholders with the purpose of maintaining risk and risk loss under control.
Our concern in this paper is to investigate on models and technologies that can provide support to risk management systems in a geographically distributed environment. The purpose is to provide a general setting built on common principles and mechanisms so that dynamic and adaptive risk management can be pursued. The paper discusses an agent-based approach to risk management in a distributed sensor network and is structured as follows: in section 2 we model the risk management in a distributed sensor network. The framework described in section 3 maps this model with a multiagent system while section 4 introduces the description of the autonomous agent architecture.

2 RISK MODELING IN DISTRIBUTED SENSOR NETWORKS

A sensor network can be used in a wide scenario of applications ranging from environmental monitoring, climate control, structural monitoring, medical diagnostics, disaster management or emergency response (Culler et al., 2004). For their own nature, these monitoring applications have to deal with intrinsic risk components that must be continuously controlled. Therefore, an effective and efficient risk modeling phase assumes great relevance for the prevention and the mitigation of undesiderable and/or dangerous events.
Note that, in order to keep the model as general as possible, the concept of sensor is here used in a broad sense, and indicates an individual unit that inputs data (Wiener, 2000). Then, a sensor could be an electronic device if we model environmental risk management as well as an human being that collects data for an enterprise risk management system.
Figure 2 shows the model of a distributed sensor network for risk management. Each local monitoring node LM_1, LM_2,...LM_j of the network is responsible for the risk monitoring of a specific locations and is connected to others by a communication infrastructure. Considerating the generic local monitoring node LM_i, sensors S^i_1,...S^i_n captures data from the environment and send them to the associated node for the data analysis and risk evaluation. At a given time instant, a node LM_i manages one or more risk lifecycles that are associated to specific risks to be monitored. A higher level node, Global Monitoring GM, is responsible for the global monitoring of the network; apart from the typical functions assigned to local monitoring nodes, it executes further functions:

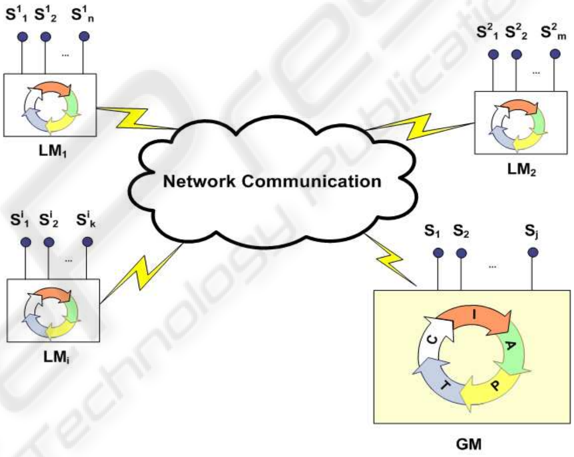

Figure 2: A distributed sensor network.

- identifies risk types;
- analyzes risks;
- plans and configures LM_i
- assigns responsibilities to LM_i
- capture alarms that involve two or more local monitoring nodes.

For example, an instance of the nodes shown in fig. 2 could be a sensor network part of a system that is charged of supervising environmental risks of fire, air pollution, etc. in a forest, and the nodes LM_1, LM_2 and LM_n are local monitors placed in particular areas of the territory to put under surveillance various types of risk. Each node is connected to sensors S_1,...,S_k measuring, for example, Temperature, Humidity,

CO_2 level and so on. A preliminary risk analysis phase could establish that the risk of fire increases when the temperature gets over a certain threshold (e.g. 45 C°) while humidity decreases. In this way, local monitors evaluate at distinct time instants the information retrieved from the sensor network, estimating in real time the risk level and making a mitigation action when a fire risk probability increases (for example, sending an alarm to local authorities).

3 A FRAMEWORK FOR RISK MANAGEMENT

The framework proposed in this paper considers the model presented in the previous section together with a distributed network of smart autonomous agents that provides automatic support to the model implementation. According to Wooldridge and Jennings definition (Wooldridge and Jennings, 1995; Jennings, 2001), agents are computer-based systems having the following characteristics:

- can operate autonomously, because they have control both over their internal state and over their behaviour;
- have the capability to communicate and interact with other agents;
- receive inputs from the environment and react to them in order to satisfy the design goals;
- exhibit goal-directed behaviour by taking the initiative.

The distributed agent architecture fits well the general model described in the previous section. Each local monitoring node is now realized by a Local Monitoring Agent that aims to manage one or more risk tipologies associated to the encompassing environment.

Because the number of possible risks for each considered domain can be very high, a single agent, able to manage a great number of risks, could become very complex. Furthermore, as the knowledge acquisition for a single type of risk may involves a substantial amount of communication with many others agents, it is convenient introduce a simplified light agent, specialized to handle a single risk type: the Risk Monitor Agent. Therefore, while an autonomous agent must coordinate the global behaviour of all its Risk Monitor Agents and decide the correct and more efficient action to perform when risks arise, each Risk Monitor Agent restricts its work to the monitoring of a single risk tipology.

In the following, the generic Local Monitoring and Risk Monitoring Agents will be denoted with A and R respectively. Risk Monitor Agents are created by the agent who own them on the basis of assigned risk types, chosen among those registered during the identification phase managed by the Global Monitor Agent (G). For example, in the case of surveillance of some environmental variables in a forest described in the previous section, there are n agent A_1, A_2,...A_n that perform local monitoring. Each A_i creates an agent R devoted to evaluation of fire risk, another for air pollution risk, and so on.

When R signals a critical situation, A must evaluate the current state and take a decision in order to resolve the problem or mitigate the risk. If not enough information are available to decide autonomously, A could require other information from neighbouring agents.

The identification and analysis of risk types is managed by the agent G that, having the complete knowledge of the initial planned agent distribution, instantiates the network configuration. Then, the risk lifecycle of a local agent A performs the following activities:

Identify: A receives by G the risk types under its responsibility and creates an agent R dedicated for each risk type to monitor;

Analyse: an agent R evaluates the inputs from the sensors and historical data exploiting the fuzzy rules to eventually suggest a mitigation action;

Plan: the agent A uses data analyis received from R to decide the better strategy to follow (e.g. plan a mitigation action or start a cooperation with other agents to “learn” more about);

Track: risk status data are collected by R and registered in the Historical Data Repository;

Control: A decides to handle the risk locally or to perform an escalation action. It has a global overview of all its identified risk types and establishes if there exists risk correlation between two or more of them;

Communicate: the communication (internal with/between Rs and external with other As) is guaranteed through the sending and receiving of messages.

The capabilities of an agent A are:

- generate a new local R, replicate or migrate a local R to another A, split a local R in two or more parts, merge two or more local Rs in a new one;
- accept or deny a remote replication or migration of an agent R;

- maintain explicit belief models of itself and other agents and be able to reason with incomplete, inconsistent and uncertain information;
- have a set of capabilities (which can change dynamically) that permit to "*learn from environment*";

The main features of the of the operator "replication", "migration", "split" and "merge" together with the motivation for their introduction are discussed below.

3.1 Replication

Replication allows multiple instances of the same R to be created in order to share the new risk types with another A. If A_i, having a risk type R, asks for a support to a remote A_j, it can replicate its agent R to A_j, sharing its local knowledge. For example, if A_i is evaluating the risk of a certain type of bacteric infection by means of R_k, A_i can ask to the neighbour agent A_j, that monitors only cancer risks, to host a replication of R_k in its own environment in order to collect information about the same infection in a given surrounding area.

3.2 Migration

Agent "Migration" allows to move the complete R environment from an agent A to another in order to undertake a distribute decision. When an agent receives a new event and it has no possibility to evaluate the correct action to perform the risk mitigation, then it starts a speculative action ((Kitamura and Murao, 2004)) that evaluates if a cooperative action becomes necessary. In the cooperative action the agent uses the Migration function to move its whole environment (DB, Fuzzy Logic, Program Code, etc.) to a neighbour agent.
Migration of Risk Monitor Agent is triggered when a migration message is processed (sent by another agent in the computation) or when a decision agent determines that the R should be migrated. When the A lifecycle is expiring, it can also decide to migrate all its Rs to the near A to avoid that the local knowledge be lost.

3.3 Split and Merge

If the risk is too complicate to resolve or there is a need to do analysis at a finer level, then A can split R in two or more parts respectively with a subrisk to evaluate. Consider, for example, a scenario where R receives the responsibility of collecting data about Meningitis occurrences in a system that monitors the risk of epidemies in a specific territory; the analysis could reveal that the risk associated to a *pneumococcus meningitis* epidemy is faster increasing compared with those induced by *meningococcus* or *Haemophilus B*. In this case, A can decide to split R, isolating the cases of *pneumococcus* meningitis to better evaluate this specific risk in the area.
On the contrary, if starting from the analysis of two risks managed, for example, by R_1 and R_2 can be inferred a new risk, A can merge R_1 and R_2 in R_3. Moreover, after a migration or replication to a new environment, same versions of the same risk type might already exist; in this case, a merge can be performed to save resources.

4 THE AUTONOMOUS AGENT ARCHITECTURE

The architectural model of an Autonomous Agent is shown in fig. 3. It includes a certain number of Rs each one uniquely identified by its Rid, that combined with Lid allows the retrieval of a Risk Monitor Agent over the network.

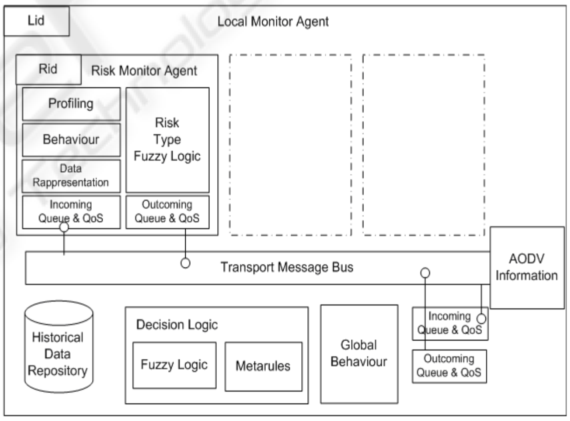

Figure 3: The Architecture of a Local Monitoring Agent.

The role played by the components of an A is described as follows. A *Risk Monitor Agent*, tracks risk status and risk identification, uploading the content of its captured data into the *Historical Data Repository*. The Profiling of a Risk Monitor Agent specify the characteristics of the risk type to handle. Local Monitoring Agent can adopt different profiling strategies for their R, based on the way profiles will be used. Generally, they profile risk type but also processing power, memory, storage, latency and bandwidth, where messages are sent to and received from, as well as the time spent to process or send a message. On the basis of the profiled information, the *Decision Logic* component decides how the risk monitoring agents must be distributed. The Fuzzy Logic component in the Local Monitor Agent is locally used from each Risk Monitor Agent to evaluate the specific risk type; R suggests a correct mitigation action that in many cases can be more effective when the collaboration with other Rs is tight. The final mitigation action is performed from the A that, on the basis of the historical information and the R suggestion, starts the Decision Process. The decision process is based on the Fuzzy logic for the Risk Type evaluation and on the Meta Rules for the infrastructure environment and profiling evaluation. Fig. 4 illustrates the macro decision process in an Autonomous Agent. The *Transport Message Bus* enables both the internal and external communication. Message priorities based on the associated weighed risk are also considered to manage the message queue allowing for urgent reconfiguration messages and QoS.

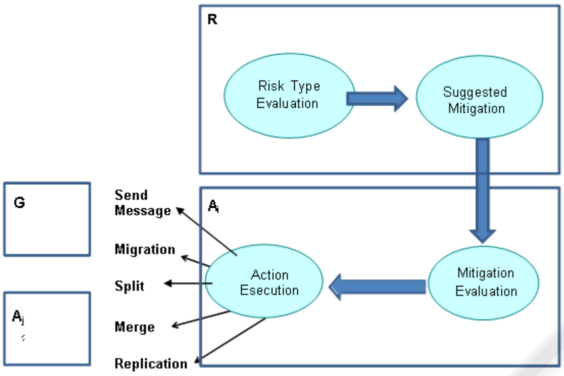

Figure 4: The decision process of an Autonomous Agent.

From the consideration that risks can be very often well evaluated by observing phenomena happened in neighbouring areas, such as fires, inundations, epidemics and so on, AODV algorithm is selected to allow communication. This algorithm can be used in dynamic and mobile network where the neighbour list can change every time and it builds routes between nodes only as desired by source nodes.

5 CONCLUSIONS

The paper proposes a framework for an adaptive risk management in a distributed sensor network that considers an abstract model together with an agent-based distributed system that realizes it. The system is composed of a certain number of agents able to proactively monitor the risks and adapt their behaviour, "learning" from the environment the more effective mitigation action for each risk.

The agent decisional process uses fuzzy rules and fuzzy logic inference to map human concepts and rules, simulating reasoning mechanisms as proposed in (He et al., 2003; Chrysanthakopoulos et al., 2004). The introduction of light agents devoted to risk monitoring allows the system to decompose the complexity of risks handling. Moreover, the capability of a Local Monitoring Agent to split and merge simpler agents allows to focus the attention at the right granularity level.

The presented framework is part of a project for the supervising of environmental risks in specific areas (called cluster) of "Regione Campania" in the South of Italy. The project, funded by POR Campania 2000-2006 Misura 6.2 "Società dell'Informazione", is almost completed. Some hw/sw devices concerning the video sensor network have already been installed for each cluster and include day&nigth and/or infrared cameras (fig. 5) together with sensors for meteo survey, electromagnetism, earthquake, air pollution, etc. responsible for the capture of data that local monitoring agents will evaluate. The implementation of our framework is in progress and an early prototype is expected in the next months.

While the proposed model provides some important benefits deriving especially from the automation of several activities of the risk lifecycle, limitations exist. As the risk management strategy is defined by humans, failures may be latent in the system and the identification of errors or mistakes can be underevaluated and hidden in the multiagent system. They must be considered as further risk types difficult to identify and resolve. Furthermore, the decisional behaviour of agents contains itself a certain degree of risk that must be considered in the implementation phase as suggested in (Vytelingum et al., 2004; Lorenz et al., 2005).

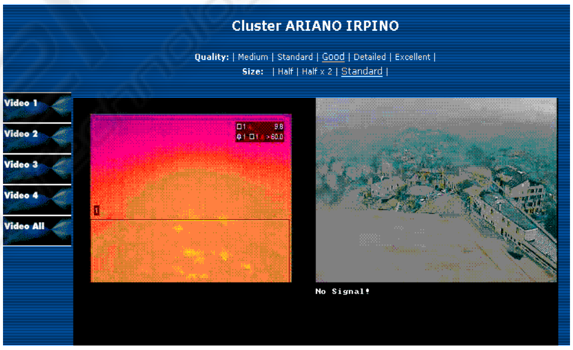
Cluster ARIANO IRPINO
Quality: | Medium | Standard | Good | Detailed | Excellent |
Size: | Half | Half x 2 | Standard |
Video 1
Video 2
Video 3
Video 4
Video All
9.8
>60.0
No Signal!

Figure 5: An infrared termocamera for the video surveillance of a cluster.

REFERENCES

Boehm, B. W. (1991). Software risk management: Principles and practices. *IEEE Software*, 08(1):32–41.

Camara, M. S., Kermad, L., and Mhamedi, A. E. (2006). Risk prediction in erp projects: Classification of reengineered business processes. In *CIMCA '06: Proceedings of the International Conference on Computational Inteligence for Modelling Control and Automation and International Conference on Intelligent Agents Web Technologies and International Commerce*, page 213, Washington, DC, USA. IEEE Computer Society.

Chrysanthakopoulos, G., Fox, W. L. J., Miyamoto, R. T., II, R. J. M., El-Sharkawi, M. A., and Healy, M. (2004). A fuzzy-logic autonomous agent applied as a supervisory controller in a simulated environment. *IEEE T. Fuzzy Systems*, 12(1):107–122.

COSO (2004). Enterprise risk management - integrated framework.

Culler, D., Estrin, D., and Srivastava, M. (2004). Guest editors' introduction: Overview of sensor networks. *Computer*, 37(8):41–49.

Grabowski, M., Merrick, J. R. W., Harrald, J. R., Mazzuchi, T. A., and van Dorp, J. R. (2000). Risk modeling in distributed, large-scale systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, 30(6):651–660.

Han, W.-M. and Huang, S.-J. (2007). An empirical analysis of risk components and performance on software projects. *Journal of Systems and Software*, 80(1):42–50.

He, M., fung Leung, H., and Jennings, N. R. (2003). A fuzzy-logic based bidding strategy for autonomous agents in continuous double auctions. *IEEE Trans. on Knowl. and Data Eng.*, 15(6):1345–1363.

Institute of Risk Management (2002). A Risk Management Standard.

Jennings, N. R. (2001). An agent-based approach for building complex software systems. *Commun. ACM*, 44(4):35–41.

Kitamura, Y. and Murao, T. (2004). Risk management methods for speculative actions. In *AAMAS '04: Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 1250–1251, Washington, DC, USA. IEEE Computer Society.

Lorenz, M., Gehrke, J. D., Langer, H., Timm, I. J., and Hammer, J. (2005). Situation-aware risk management in autonomous agents. In *CIKM '05: Proceedings of the 14th ACM international conference on Information and knowledge management*, pages 363–364, New York, NY, USA. ACM.

Project Management Institute (2004). A Guide to the Project Management Body of Knowledge (PMBOK Guide) - Third Edition.

Rosenberg, L. H., Hammer, T., and Gallo, A. (1999). Continuous risk management at nasa. In *Applied Software Measurement / Software Management Conference*, San Jose, California.

Schaller, M. and Vaz, A. (23-25 Mar 1997). Derma: a distributed enterprise risk management architecture. *Computational Intelligence for Financial Engineering (CIFEr), 1997., Proceedings of the IEEE/IAFE 1997*, pages 22–28.

Vytelingum, P., Dash, R. K., David, E., and Jennings, N. R. (2004). A risk-based bidding strategy for continuous double auctions. In de Mntaras, R. L. and Saitta, L., editors, *European Conference on Artificial Intelligence*, pages 79–83. IOS Press.

Wiener, N. (2000). *Cybernetics: Or Control and Communication in Animal and the Machine*. MIT Press, Cambridge, MA, USA.

Williams, R., Walker, J., and Dorofee, A. (1997). Putting risk management into practice. *IEEE Softw.*, 14(3):75–82.

Wooldridge, M. and Jennings, N. (1995). Intelligent Agents: Theory and Practice. *The Knowledge Engineering Review*, 10(2):15/152.