ROMA
TRE
UNIVERSITÀ DEGLI STUDI

DOCTORAL SCHOOL OF ENGINEERING
COMPUTER SCIENCE AND AUTOMATION
DEPARTMENT

XXV Cycle

Agile Knowledge Management for Situation Awareness

PhD student
Giusj Digioia

Tutor
Prof. **Stefano Panzieri**
Università Roma Tre

June 2013

Abstract

In the information age, access to data is easily achieved thanks to the development of new advanced sensors and information sources, able to measure all kind of features, to acquire several kind of information and to transfer those data fast and effectively all over the world.

Taking into account newly developed sensors and networks, the access and communication of information are not unyielding and crucial as their analysis, aggregation and elaboration.

It is within this context, that of Data Fusion finds its applicability. At the end of the 1990, Data Fusion doctrine is formalized as the ensemble of scientific techniques and algorithms, properly implemented in a single framework that is able to: (a) support human operators to gather huge quantities of heterogeneous data (some of which may not be synchronized) from sensors observing the scenario of interest; (b) detect and classify objects acting in the scenario; (c) understand the relationships among them, and the intent, and threats that they could cause; (d) foresee future evolutions of the scenario; and (e) take the best decisions in order to maximize human operator utility.

The above mentioned process has been formalized in the Joint Directors of Laboratories (JDL) model, and progressively revised mainly to give greater emphasis to the human operator as an active sensor within the process, and to evaluate the quality of the whole inference process.

Within the Data Fusion process, the goal of Situation Awareness (JDL Level 2) is to recognize relationships existing among objects observed in the scenario, in order to recognize situations of interest, and evaluate their threats. Hence, Situation Awareness should help human operators to be

aware of the context they are observing, especially when the scenario is wide, or phenomena observed are complex and evolve fast.

The focus of this PhD thesis is exactly Situation Awareness, and in particular knowledge management: in order to recognize situations and infer them from observations, knowledge models describing situations of interest must be effective, correct, and should be able to catch relevant and discriminant aspects.

The definition of a good knowledge model is crucial for effective Situation Awareness, and it is usually hard because it requires experience in the domain, or the availability of huge quantities of data to be input to learning algorithms (that generates usually models difficult to interpretate). Moreover, once the model has been defined, the evaluation of its quality is difficult, especially in real-time, because the truth about the observed situation is not known.

The goal of this thesis has been the investigation about effective knowledge management for correct inferences, and in particular the following aspects of knowledge management have been considered:

- real-time knowledge model construction with regard to specific situations or events of interest, adopting Data Mining techniques;

- real-time knowledge model refinement, according to metrics expressing the adequacy of the model to the observations gathered.

Knowledge models employed in Situation Awareness usually differ from each depending on the mathematical approach adopted (Bayesian approach refers to Bayesian Networks, Hidden Markov Models requires a probabilistic inference algorithm, Evidence theory refers to cause-effects models). In this work, real-time model construction has been apply to Hidden Markov Models; while real-time knowledge refinement has been investigated with regard to Evidence Theory.

Moreover, considerations derived from the implementation of Situation Awareness frameworks within the military context and critical infrastructure protection domain have been reported.

Majour results of this research can be summarized in the characterization of the *agility* measure, able to quantify the capability of a model to revise itself by evaluating inconsistencies, contradictions and errors, and taking into account uncertainty of information employed.

Model agility has be identified as a powerful feature in JDL Level 4 Process Refinement, because it can guide and improve the overall data collection process, eventually cueing the user or the system to search for lacking information.

Main features identified for an agile model are the following:

- an agile model does not require to be perfect since its construction: it can be obtained with imperfect knowledge of the whole system, because it is able to learn from its experience;
- agility extends the model lifetime: agile models are able to manage a greater number of scenarios that maybe were not even included when the model was created;
- an agile model is more resilient, more robust, and able to perform better and wider range of real life scenarios.

Investigations about agility measure within Evidence Theory, have highlighted the inability of knowledge models and algorithms to recognize time-dependent situations. In this regard, the trend of the empty set mass has been identified as an agility measure, able to identify the fitness of the model to the observed situations, and in particular model inadequacy to describe, and hence recognize, time-dependent patterns. It has been shown how to employ the measure for model review and correction, in order to allow in Evidence Theory dynamic pattern recognition, besides to static classification.

Finally, research conducted for this PhD thesis have lead to the definition of a system architecture combining Data Mining and Data Fusion techniques in order to allow the construction of knowledge models able to recognize effectively situations of interest, that can be specified by the user in real-time. In the proposed framework Data Mining approach is employed to define correlations among data stored in databases, and events or objects of interest for the user; mined correlations are employed to build in real-time knowledge models to be adopted in the Situation Awareness process.

Contents

Chapter 1

Introduction

1.1 Multi-sensor Data Fusion

In the information age, access to data is easily achieved thanks to the development of new advanced sensors and information sources, able to measure all kind of features, to acquire several kind of information and to transfer those data fast and effectively all over the world.

Taking into account newly developed sensors and networks, the access and communication of information are not unyielding and crucial such as their analysis, aggregation, and elaboration.

Within this context, the aim of the data fusion discipline finds its applicability. At the end of 1990's, Data Fusion doctrine is formalized as the ensemble of scientific techniques and algorithms, properly implemented in a framework, able to support human operators to gather huge quantities, heterogeneous and eventually not synchronized data from a set of sensors observing the scenario of interest, even wide and complex; to detect and classify objects existing and interacting in the scenario; to understand relationships among them, their intent, and the threats that they could cause; to foresee future evolutions for the scenario and to take best decisions in order to maximize human operator utility.

Hence, Data Fusion techniques combine data from multiple sensors and related information to achieve more specific inferences than could be achieved by using a single, independent sensor. The concept of multi-sensor Data

Fusion is hardly new. As humans and animals have evolved, they have developed the ability to use multiple senses to help them survive. For exemple, assessing the quality of an edible substance may not be possible using only the sensor of vision; the combination of sight, touch, smell, and taste is far more effective. Similary, when vision is obstructed by structures and vegetation, the sense of hearing can provide advanced warnings of impending dangers. Thus, multisensory data fusion is naturally performed by animals and humans to assess more accurately the surranding environment and to identify threats, thereby improving their chances of survival. While the concept of Data Fusion is not new, the emergence of new sensors, advanced processing techniques, and improved processing hardware have made real-time fusion of data increasingly viable.

Applications for multisensor data fusion are widespread. Military applications include automated target recognition, guidance for autonomous vehicles, remote sensing, battle-field surveillance, and automated threat recognition systems, such as identification-friend-foe-neutral (IFFN) systems. Non-military applications include monitoring of manufacturing processes, condition-based maintenance of complex machinery, robotics, and medical applications. Techniques to combine or fuse data are drawn from a diverse set of more traditional disciplines, including digital signal processing, statistical estimation, control theory, artificial intelligence, and classic numerical methods. Historically, data fusion methods were developed primarily for military applications. However, in recent years, these methods have been applied to civilian applications and a bidirectional transfer of technology has begun.

The following sections provide an overview on models formalizing the data fusion process, with a particular focus on processes involved in the *Situation Awareness*, the main topic of this PhD thesis. Moreover, an overview about disciplines, techniques and open researches involved in the Data Fusion process is reported.

1.2 Models for Data Fusion Frameworks

In order to improve communications of scientists and experts working on Data Fusion, a formalization of the processes involved has been required.

In past years, several models have been proposed, but the most popular one is definitively the JDL Data Fusion model, which was formalized by the Joint Directors of Laboratories (JDL) Data Fusion Working Group. Recently, scientists have formulated evolutions of the JDL model and have employed it as the basis for new models, focused on particular aspects.

Because of the importance of the JDL model, the Data Fusion models described hereafter, will be discussed in its context.

1.2.1 DIKW Hierarchy and Abstraction of Knowledge

The traditional Data Information Knowledge and Wisdom (DIKW) hierarchy 1.1 organizes data, information, knowledge, and wisdom in layers with an increasing level of abstraction and addition of knowledge, starting from the bottommost data layer. The hierarchy bears some resemblance to the JDL data fusion model in the sense that both start from raw transactional data to yield knowledge at an increasing level of abstraction.

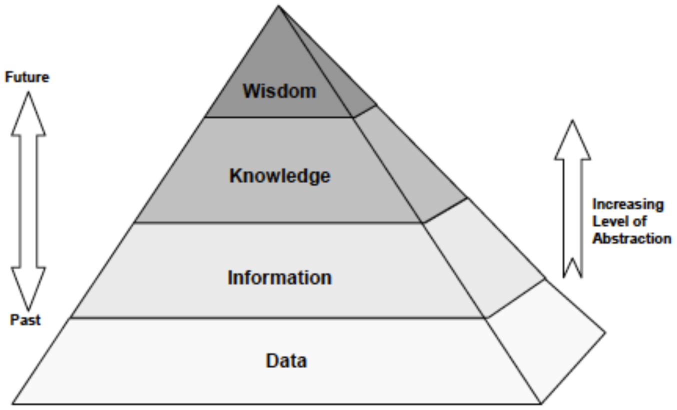


Figure 1.1: Data Information Knowledge and Wisdom hierarchy

Layers characterizing the model are:

- *Data Layer* - Data are transactional, physical, and isolated records of activity (e.g., Signal Intelligence (SIGINT) reports, facts, or figures obtained from experiments or surveys). Data are the most basic level and by themselves have little purpose and meaning.

- *Information Layer* - Information is semantic interpretation of data or represents relationships between data with meaning and purpose (e.g., tank unit at a certain battlefield location, presence of an enemy unit in a defensive posture).
- *Knowledge Layer* - Knowledge is general awareness or possession of information, facts, ideas, truths, or principles. Knowledge is generally personal and subjective.
- *Wisdom Layer* - Wisdom is knowledge of what is true or right, coupled with just judgment as to action. Wisdom is the knowledge and experience needed to make the right decisions and judgments in actions.

Thus *data* are the basic unit of *information*, which in turn is the basic unit of *knowledge*, which in turn is the basic unit of *wisdom*.

1.2.2 OODA Loop

One of the first Command, Control, Communications, Computers, and Intelligence (C4I) architectures is the Observe-Orient-Decide-Act (OODA) Loop [2], shown in Figure 1.2.

The OODA architecture was developed by Col. John Boyd, USAF [1] during the Korean War while referring to the ability possessed by fighter pilots that allowed them to succeed in combat. Observations in OODA refer to scanning the environment and gathering information from it, orientation is the use of the information to form a mental image of the circumstances, decision is considering options and selecting a subsequent course of action, and action refers to carrying out the conceived decision.

1.2.3 Rasmussen Information Processing Hierarchy

Rasmussen's three-tier model of human information processing [3] [4], is shown in Figure 1.3. The arch in Rasmussen's skill, rule, knowledge (SRK) model represents the flow of information through the human decision maker.

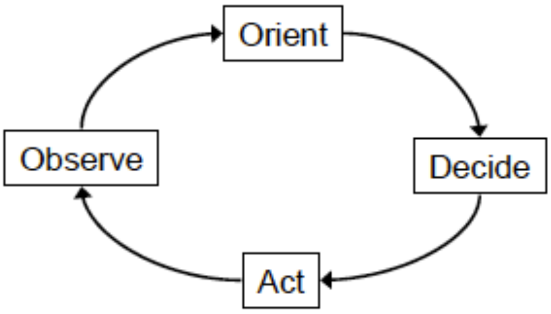


Figure 1.2: Boyd's OODA Loop

The left side of the arch corresponds to stimulus processing, and the right side corresponds to motor processing. Processing is divided into three broad categories, corresponding to activities at three different levels of complexity.

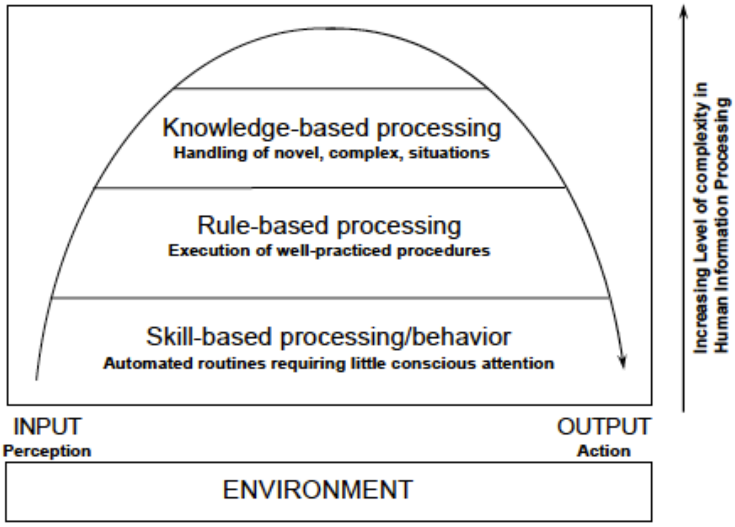

Knowledge-based processing
Handling of novel, complex, situations
Rule-based processing
Execution of well-practiced procedures
Skill-based processing/behavior
Automated routines requiring little conscious attention
INPUT
Perception
OUTPUT
Action
ENVIRONMENT
Increasing Level of complexity in Human Information Processing

Figure 1.3: Rasmussen hierarchy of human information processing

- *Skill-Based Processing* - At the lowest level is skill-based sensorimotor behavior, such as perceptual feature extraction and hand-eye coordination. This level represents the most automated, largely unconscious level of skilled performance (e.g., identification of a tank by looking at raw sensor reports).

- *Rule-Based Processing* - At the next level is rule-based behavior, exemplified by procedural skills for well-practiced tasks such as the identification of an enemy unit composition based on its numbers and relative locations.

- *Knowledge-Based Processing* - Knowledge-based behavior represents the most complex cognitive processing used to handle novel, complex, situations where no routines or rules are available for handling situations. Examples of this type of processing include interpretation of unusual behavior, and generating a Course of Action (COA) based on enemy unit size and behavior.

The Generic Error Modeling System (GEMS) [6] is an extension of Rasmussen's approach, which describes the competencies needed by workers to perform their roles in complex systems. GEMS describes three major categories of errors: skill-based slips and lapses, rule-based mistakes, and knowledge-based mistakes. See [5] for an instantiation of the information processing hierarchy to implement an agent to amplify human perception and cognition.

1.2.4 JDL model and its developments

The first version of the model is a two-layer hierarchy, Figure 1.4. At the top level, the data fusion process is conceptualized by sensor inputs, human-computer interaction, database management, source preprocessing, and four key subprocesses:

- **Level 1** processing (*Object Refinement*) is aimed at combining sensor data to obtain the most reliable and accurate estimate of an entity's position, velocity, attributes, and identity;

- **Level 2** processing (*Situation Refinement*) dynamically attempts to develop a description of current relationships among entities and events in the context of their environment;
- **Level 3** processing (*Threat Refinement*) projects the current situation into the future draw inferences about enemy threats, friend and foe vulnerabilities, and opportunities for operations;
- **Level 4** processing (*Process Refinement*) is a meta-process that monitors the overall data fusion process to asses and improve real-time system performance.

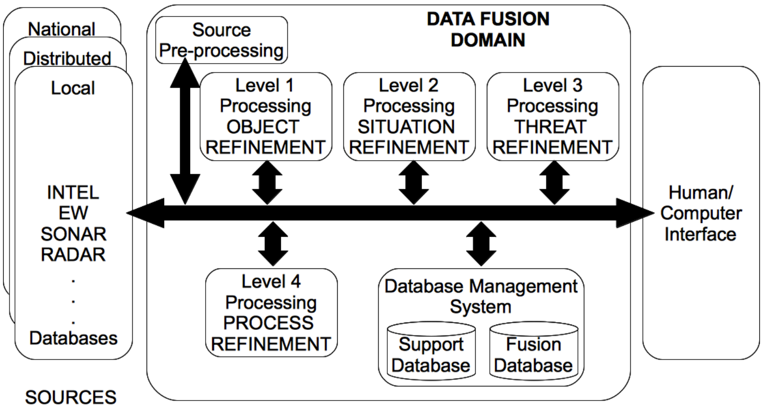

National
Distributed
Local
INTEL
EW
SONAR
RADAR
Databases
SOURCES
Source Pre-processing
DATA FUSION DOMAIN
Level 1 Processing OBJECT REFINEMENT
Level 2 Processing SITUATION REFINEMENT
Level 3 Processing THREAT REFINEMENT
Level 4 Processing PROCESS REFINEMENT
Database Management System
Support Database
Fusion Database
Human/ Computer Interface

Figure 1.4: JDL model

As it is noticeable, sources of information for the overall framework are supposed to be both local and distributed. The output of the elaborations is presented to the final user through proper Human Computer Interface. The Database Management System represents the set of databases and tools necessary to provide information for the fusion process (e.g., pre-defined and learned knowledge models or fusion rules); to archive data to be fused and to archive results of elaborations. The Source Pre-Processing module aims to perform all operations required by specific sensors before data elaboration (e.g., word extraction from audio signals).

The most mature area of data fusion process is *Level 1* processing, e.g., using multisensor data to determine the position, velocity, attributes, and identity of individual objects or entities. Determining the position and velocity of an object based on multiple sensor observations or tracks, and estimating the position and velocity of a target. Multi-sensor target tracking is dominated by sequential estimation techniques such as Kalman filter. Challenges in this area involve circumstances in which there is a dense target environment, rapidly maneuvering targets, or complex signal propagation environments (e.g., involving multipath propagation, cochannel interference, or clutter). However, single-target tracking in excellent signal-to-noise environments for dynamically well behaved target is a straightforward, easily solved problem. Current research focuses on solving the assignment and maneuvering target problem. Techniques such as multiple-hypothesis tracking (MHT), probabilistic data association methods, random set theory, and multiple criteria optimization theory are being used to resolve these issues. Some researchers are utilizing multiple techniques simultaneously, guided by a knowledge-based system capable of selecting the appropriate solution based on algorithm performance. A special problem in Level 1 processing involves the automatic identification of targets based on observed characteristics or attributes. To date, object recognition has been dominated by feature-based methods in which a feature vector (e.g., representation of the sensor data) is mapped into feature space areas for several classes of target). More research is needed in this area to guide the selection of features and to incorporate explicit knowledge about target classes. For example, syntactic methods provide additional information about makeup of target. In addition, some limited research is proceeding to incorporate contextual information, such as target mobility with respect to terrain, to assist in target identification.

Level 2 and *Level 3* fusion (situation refinement and threat refinement) are currently dominated by knowledge-based methods such as rule-based blackboard systems. In these areas, the most relevant problem is related to the representation and construction of models of knowledge in specific domains of interest. A system able to reason about situations and threats cannot be generalized, it needs to know entities and relations characterizing the operative context. This seems to be the most relevant limitation for level 2 and 3 systems. Research must be done to identify a standard methodology to represent knowlodge and to investigate algorithms able to handle those models, and with uncertain and heterogeneous information, in order to infer situations and threats. Recently, the ontological approach seems to be accepted as a promising way to represent knowledge about entities and relations existing among them; moreover probabilistic algorithms start to be investigated to manage ontologies and reasoning about them are being investigated. Other promising approaches are driven from Bayesian framework, Evidence Theory, Neural Networks and Markov Models.

Bayesian Net (BN) approach relies on Bayes theory and allows building of graphical models, highlighting cause relations about entities of the domain; however, some cons are related to the huge number of parameters needed to build BNs and to high computational load in cases of BNs with many variables.

Evidence Theory (ET) allows managing fuzzy information with efficient algorithms, capable of discovering contradictions among information acquired and to highlight incompleteness of models adopted. The counterpart is that it is not well suited for time-dependent modeling of situations and threats, despite *Markov Model* (MM) approach.

MMs allow to model dynamic situations and threats and in literature algorithms exist to employ asynchronous, heterogeneous information and to refine model parameters as well as new data are acquired.

Finally, *Neural Network* (NN) approach has the advantage of building good models, eventually non-linear, but it is usually impossible to identify a semantic meaning to them and much training data are required for their definition.

Level 4 processing, which assesses and improves the performance and operation of an ongoing data fusion processes, has a mixed maturity. For single sensor operations, techniques from operations research and control theory have been applied to develop effective systems, even for complex single sensors such as phased array radars. In contrast, situations that involve multiple sensors, external mission constraints, dynamic observing environments, and multiple targets are more challenging. To date, considerably difficulty has been encountered in attempting to model and incorporate mission objectives and constraints to balance optimized performance with limited resources, such as computing power and communication bandwidth, and other effects. Methods from utility theory are being applied to develop measures of system performance and measures of effectiveness. Knowledge-based systems are being developed for context-based approximate reasoning. Significant improvements will result from the advent of smart, self-calibrating sensors, which can accurately and dynamically assess their own performance.

Recent developments on the JDL model have led to its revision as shown in Figure 1.5, where *Level 5* User Refinement has been introduced, as explained in [31].

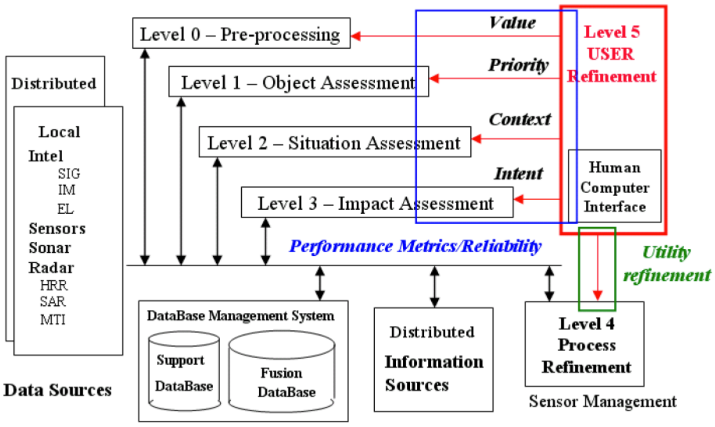


Figure 1.5: JDL model review

1.3 Situation Awareness

1.3.1 Endsley model for Situation Awareness

While the JDL model is a functional model for the data fusion process, it does not model it from a human perspective. A human operator/analyst becomes aware of certain situations (or achieves situation awareness) based on either the results from his or her own cognitive processes or the results produced by an automated Situation Assessment (SA) tool built on some processing model such as the JDL model. Such an automated SA tool does not have to mimic the complex human cognitive processes (in fact the JDL model and many other computational models do not). Endsley in [7] formally defines situation awareness as *the perception of the elements in the environment within a volume of time and space, the comprehension of their meaning, and the projection of their state in the future.* The three levels of Situation Awareness (SAW), namely *perception*, *comprehension*, and *projection*, as shown in Figure 1.6, parallel the corresponding levels in the JDL model. The levels in the JDL model can be viewed as processes producing results to help a human operator became aware of the situation at hand.

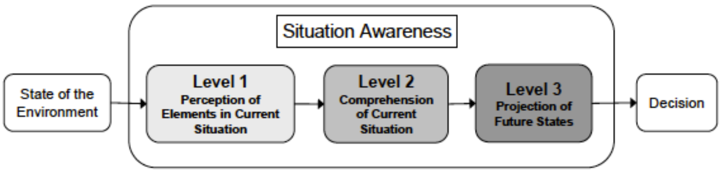

Situation Awareness
State of the Environment
Level 1
Perception of Elements in Current Situation
Level 2
Comprehension of Current Situation
Level 3
Projection of Future States
Decision

Figure 1.6: Situation Awareness model

In accordance to Endsley model

- *Level 1* represents the Perception of Elements in Current Situation - A human operator needs to perceive the status, attributes, and dynamics of relevant elements in the environment, including the current and historical movements of both friendly and hostile units, their types, terrain, and weather. The information overloading problem can easily rise in a dynamic, high-tempo, operational environment.
- *Level 2* represents the Comprehension of Current Situation - A human operator needs to comprehend the situation based on a synthesis of perceived elements along with the mission at hand. The level provides a holistic picture of the environment relating the disjointed perceived elements. For example, upon observing some probing activities by an

adversarial force in an area within an urban environment, the operator quickly determines the seriousness of the problem in terms of a coordinated attack by combining other contextual knowledge, which may include high-value targets in the area, previous activities, and capabilities. The degree of comprehension by a human operator varies according to the operator's level of experience. A novice operator may be capable of achieving the same Level 1 situation awareness as a more experienced operator, but may not be able to integrate various perceived elements along with the mission in order to comprehend the situation.

- *Level 3* represents the Projection of Future States - A human operator needs to understand what will happen in the near future based on both understanding of dynamics of the perceived elements and comprehension of the current situation. This level provides the knowledge and time necessary to decide on a suitable course of action to deal with threatening situations.

1.3.2 Open Issues

Within SAW the following research fields can be investigated:

- *knowledge model management* - before recognizing undergoing situations, humans need to know situations. If humans don't have experience of what they are observing, evidences will remain uncorrelated observations, and they will not be able to progress from looking to understanding. In this regard, it is crucial that the management of knowledge employed in the inference process, and in particular issues related to reference model representation, learning, definition and refinement, must be considered.

- *inference algorithms* - once the knowledge model has been defined, it is necessary to fit evidences gathered in the reference model in order to identify the best-fitting pattern for the evidences gathered. In this regard several issues should be addressed such as the synchronization of evidences acquired, the trustworthiness of information sources, the management of missing or uncertain information, the management of cause and effect, and the temporal relations among evidences.

- *performance evaluation* - one of the most difficult tasks is related to the evaluation of elaborations performed by a SAW framework. In this regard it is necessary to identify metrics to assess the quality and correctness of inferences. This can usually be done only if the truth is known and hence it is employable as a reference for elaborations. If the truth is not completely known or it is ambiguous, as it is usually the case of reality, then the evaluation of SAW elaborations is not a trivial task.

- *SAW results visualization* - the aim of the data fusion discipline is to increase the awareness of the user when it is not able to quickly understand what is happening in the nearby context. This problem could arise when there is too much data, the observation field is too wide, or processes observed are complex. In order to achieve the goal in data fusion, a crucial aspect is to consider the presentation of elaboration results to the user. In this regard, a good balance must be achieved between providing the user with many details and with a syntesis of the scenario. Too many details could affect the ability to quick uderstand the scenario, but they help the analysis process; synthesis helps users to have the overview about the observed scenario, but could omit relevant details to the user. Therefore, human computer interfaces must be accurately studied to be effective for users in accordance with their mission and should employ alternative tools to ensure effectiveness. In certain cases, the overall goal of the data fusion process can be achieved by just adopting good visualization.

1.4 PhD thesis research topics: motivations and goals

Main focus of this PhD thesis is the SAW discipline. The work started from addressing practical issues derived from the application of SAW methodologies and to the implementation of the SAW framework, in the military domain and in the critical infrastructure protection field. Requirements coming from final users, architecture definition and choices about methodologies to employ have been addressed and reported in this PhD thesis.

Taking into account experiences derived from the implementation issues, the focus turned towards knowledge model management. The interest arises from the firm belief that dogmatic reference models and closed world assumption are not suitable for SAW; and from the importance of effective reference models to achieve the right understanding of observed situations. To this purpose, tools supporting users in model definition, refinement and updating are of great importance.

With this regard, agility metrics to evaluate the effectiveness of knowledge models have been defined, and the analysis of agility has been conducted for different knowledge models employed by different inference methodologies.

A final analysis has regarded the comparison and the evaluation of synergies existing between Data Mining and Data Fusion, especially with the purpose of knowledge model definition and learning.

Chapter 2

Agile Models for Situation Awareness

A relevant issue felt in the domain of Situation Awareness is related to the definition of models describing situations and threats of interest. Actually, the widely adopted approaches are based on two phases: employ training data as input of learning algorithms, and then validate the built model through other sets of data, gathered from the field. Model construction is therefore considered as an off-line process, and model correction is contemplated in terms of little adjustments in real-time applications. Great advantages could be derived by the employment of agile models, able to revise themselves evaluating model inconsistencies, contradictions and errors, or taking into account user information.

In this work, the analysis of model agility is conducted with regard to the Evidence Theory approach, a technique adopted in the domain of Situation Awareness to perform automated reasoning on time-independent models and static pattern recognition. In particular, possible metrics to highlight online model inconsistencies will be investigated and evaluated in a case study driven from the Critical Infrastructure Protection (CIP) domain.

2.1 Overview on Knowledge Representation for Situation Awareness

Within the context of Information Fusion, a capability required to automate reasoning systems, is to understand relationships among objects of interests and to assess the threats they could cause. This capability, together with others related to the data fusion domain [32], is well defined as Level 2 and 3 of the Joint Directors of Laboratory (JDL) model, i.e., Situation Assessment and Threat Assessment, also referred to as Situation Awareness.

Since the Information Fusion Theory has gained relevance, many approaches have been studied to perform inference, starting from rough data as Probabilistic Bayesian Networks [8], Evidence Theory [58], Neural Networks [35] and Markov Models [9]. Each of the mentioned approaches allows to classify and recognize situations and threats, previously modeled in a proper way, employing even uncertain and imprecise information gathered from the field. In particular, depending on the empirical structure of the knowledge, different models and inferring techniques are better suited to represent the domain of interest: for example, hierarchical knowledge is well represented by cause and effect Bayesian Nets; flat and time-independent knowledge can be managed with Evidence Theory models; time-dependent patterns are well recognized by Markov Models. Finally, complex, non-linear systems can be modeled by Neural Networks.

The success of all techniques depends strictly on the adequateness of the model describing the world of interest, and the corresponding techniques, which contemplate model learning and upgrading. For example, the Viterbi algorithm [10] allows Hidden Markov Model parameter re-estimation as well as observations. Bayesian Network learning algorithms exist for parameter estimation or network structure estimation [11]. Finally, Neural Networks can be defined after a training and validation phase.

All mentioned techniques require a previous definition of a model for the area of interest, usually this task is performed off-line. Then the model is applied in application, thanks to data gathered from the observed reality. The obtained model has the ability to work only for the specific case for which it has been created and it can be only tuned with regard to some parameters during the on-line learning process. We think that a model should be tested also in real-time context where the model can change during time for some reasons, such as when the environment changes. The model should be also *agile* in order to evaluate reshaping information. By this way, the model can also be obtained with imperfect knowledge of the whole system, and it could then be able to learn from its experience. In this paper, the main idea is to determine metrics able to understand evolutions in the observed situations and make Evidence Theory able to change idea about classified situations.

Knowledge representation is felt as a crucial issue by the Situation Awareness community, as demonstrated by Endsley in [15], where wrong inferences are correlated to possible errors in knowledge modeling. Many efforts have been undertaken to define a standard model representation and to build a sort of knowledge base, at least for specific domains of real world. In this sense, ontologies [21] seem to have gained wide consensus in the community, but further works are required before a considerable number of domains will be covered, and further works will be needed to adapt ontologies to the different inferring techniques. Up to now, experiments can be found in [48], demonstrating the applicability of the probabilistic approach to ontological models.

In order to measure the effectiveness of Information Fusion systems, several studies have been conducted to define metrics for each of the JDL levels, see [12] and [13]. In general, the following metrics are identified for the Situation Awareness evaluation: *timeliness*, *confidence*, *cost*, *accuracy*, *throughput*. The evaluation of each metric is not trivial, as well as the definition of the best practice to adopt once the metrics have been computed (e.g., if the accuracy of a system is low, how is it possible to improve it?). In several works, model refinement is regarded as an off-line task for human operators, as in [23]. Metrics evaluation usually requires the comparison between inferences elaborated by the system and reality. This kind of comparison is possible only in off-line validation processes and not in real-time operations, when reality is not known and must be assessed. For these reasons, the mentioned metrics are not suitable for real-time evaluations of the system; such kind of metrics cannot rely on reality, but must take into account only the intrinsic characteristics of the model itself.

In this work, we take into account Evidence Theory, as a simple technique to score different situations. Smets' Transferable Belief Model is a mathematical representation of the Evidence Theory related to the concept of belief measures. This method allows the user to score different situations because of some information called *evidence*. The method considers the uncertainty of the model, defining not a probability measure but rather a confidence interval. This interval has lower and upper bounds, which represent the *belief* and the *plausibility* measures, respectively. The greatest disadvantage of the Evidence Theory is the computational complexity due to the definition of the power set. The power set is the set of all possible subset of all the considered situations.

2.1.1 Related Works

In this work we focus on the crucial aspect of knowledge model definition and management. In particular, we assume that a model for a certain domain of interest is given (through learning algorithms or defined by experts), and we investigate metrics able to highlight in real-time modeling errors and inconsistencies. We believe that it is of great importance for the user to understand as soon as possible if the knowledge model employed by the system is inadequate, so that model correction can be performed manually by the user, or automatically by the system. Model correction techniques go beyond the goals of this work, therefore few considerations on the topic are provided. The main focus is on the knowledge model employed by Evidence Theory, characterized by a flat structure, suitable for classification of static situations and threats. Simulation results are reported to show characteristics of inferences and define metrics for real-time model effectiveness evaluation are reported.

To our knowledge, in literature problems of real-time model correction and metrics have been addressed in different works, but from a different perspective. In real-time learning, the definition of an initial model and its upgrading through real-time experiences in real world is contemplated. In this approach, if the initial model defines few and simple actions, there is a good chance that its validity is granted and modeling errors are unlikely to occur. Despite this, simple initial models need a huge number of real-time adjustments before being effective, and in the meanwhile, agent behaviour could seem inadequate. Complex initial model definition requires high initial efforts, therefore the chance of modeling errors and inconsistencies is high. The advantage is fast availability of effective models for real world operations and few real-time adjustment requirements. In summary, the complexity of the initial model in real-time learning algorithms should be a trade off between the mentioned aspects.

Works dealing with real-time learning approach mainly focus mainly on model upgrading policies and real-time performances, rather than on model error correction and metrics for well-built models, as in [14] and [18]. In both papers the adequateness of the initial model is not at all discussed at all.

In [16], [24] and [17], are proposed different learning algorithms to build Markov Models, describing respectively a strategic game, automata behaviour and a generic dynamic system, respectively, are proposed. All approaches are focused on a learning strategy that does not count on an initial model, therefore the problem of model validation is not taken into account.

In [19] and [20] the problem of repairing incorrect knowledge after off-line model construction are discussed. Despite our work, model correction is performed off-line, and is based on the analysis of inconsistencies in the learned model. Once inconsistencies are identified, they are corrected or inserted in the model as exceptions, so that the final model provided to the inference system is coherent and well-defined.

Another research domain that could be correlated to the study proposed in this work is the one of anomalies detection, in fact an anomly can be regarded as a mismatch from a given model. The common approach adopted in this field and also in [22], is to define off-line a model for normal situations and a different model for abnormal situations, so that in real-time operations, anomalies can be recognized as well as normal situations. The need for abnormal behaviour models arises to avoid that real-time false alarms are generated each time that discrepancies with the models occur. Works in anomaly detection field mainly focus on off-line abnormal model definition techniques, despite of the proposed study whose goal is to highlight in real-time a mismatch between world representation and the on-going situation. The mismatch could occur even between reality and an abnormal behaviour model, suggesting modeling errors even in anomaly representation.

2.2 Evidence Theory Applied to Situation Awareness

The term *Evidence Theory* was coined by Shafer in [58], reinterpreting the work of Dempster [25] on how to represent and aggregate epistemic uncertainty.

Evidence Theory [58] found its application in the domain of Situation Awareness [26], as a framework to classify static, time-independent patterns of situations. The approach consists in putting in relation evidences gathered from the field with causes that could have generated those evidences. Each time evidences are acquired, the set of possible causes becomes smaller and smaller, until the identification of the most plausible one. Evidences can be heterogeneous and even asynchronous, and they can be treaten also as fuzzy variables [45].

Evidence theory has been applied in many contexts, like statistical inference, fault diagnosis, and risk analysis. Other application fields include image processing and pattern identification or recognition. In [27] a target track identification based on a radar information has been implemented, with the aim to identify hostile flying objects.

In the rest of this section Evidence Theory knowledge model characteristics and inference algorithms are presented.

2.2.1 Knowledge Representation

Knowledge model employed in Evidence Theory is typically characterized by a flat structure that can be represented as a bipartite graph $G = (\Omega, \Phi, \Lambda)$, see Figure 2.1, where:

- Ω represents the set of situations to be classified and that should be mutually exclusive and exhaustive;

- Φ is the set of evidence that can be gathered from the scenario;
- Λ contains direct edges in the form $(\omega_i; \phi_j)$, where $\omega_i \in \Omega$ and $\phi_j \in \Phi$.

Edges express correlation between situations and evidences. When specific evidence is acquired from the field, the corresponding situations are supported. In Evidence Theory, model structure is assumed to be fixed, that is why time-dependent patterns cannot be represented.

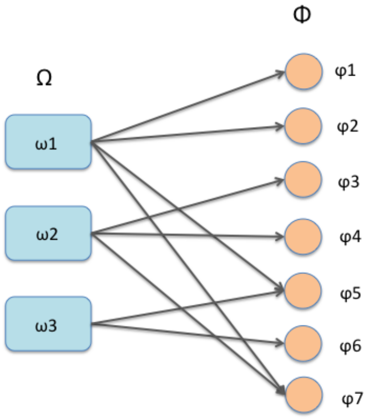


Figure 2.1: Evidence Theory knowledge model

For example, let $\Omega = \{\omega_1 \cdots \omega_n\}$ be a finite set of possible values of a variable ω, where the elements ω_i are assumed to be mutually exclusive and exhaustive (e.g., different positions, different behaviors, different situations, etc.). Suppose that only vague evidence is available in order to distinguish between the different values; for instance, during a crime investigation, a witness has seen a long haired subject in the nearby of the crime scene, while another witness has heard a female voice. These two observations apply to subsets of the suspects, and there is the need to compose them in order to determine the guilty. From a set theoretical point of view, this means that, for each observation, a value is assigned to the corresponding subset of suspects, and these values are composed for the single suspect by considering the values associated to all the subsets of the suspects that contain that specific person. Note that, in principle, all the subsets of the suspects have to be considered, and the resulting set, namely *Power Set*, has a number of elements that is exponential in the number of suspects. Specifically, if the generic subset of suspects is denoted as γ_i, the power set originated by the set Ω, is denoted by Γ or 2^Ω and is defined as $\Gamma = \{\gamma_1 \cdots \gamma_{|\Gamma|}\}$, and contains every subset $\gamma_i \subseteq \Omega$, see Figure 2.2.

In this framework, the focus is on quantifying the belief of propositions of the form: *the true value ω is contained in γ_i.*

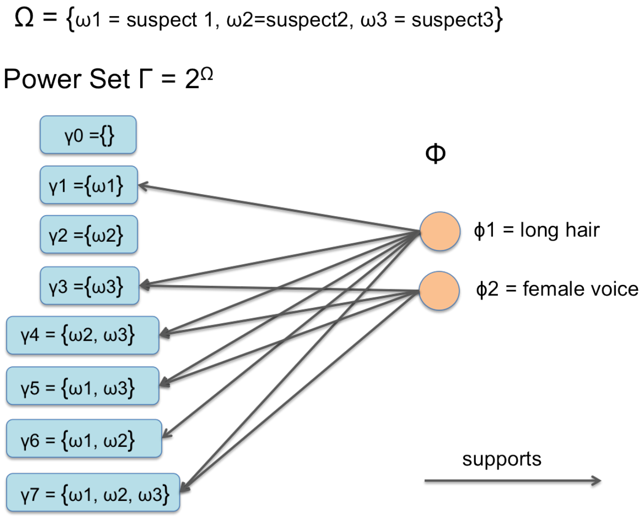


Figure 2.2: Evidence Theory model example

2.2.2 Inference Algorithm

Let us now introduce a basic belief assignment, that is a choice for the beliefs of sets γ_i.

Basic Belief Assignment - A function $m : 2^\Omega \to [0, 1]$ is called a basic belief assignment (BBA) if

$$\begin{aligned} m(\emptyset) &= 0 \quad (2.1) \\ \sum_{\gamma_a \in \Gamma} m(\gamma_a) &= 1 \quad (2.2) \end{aligned}$$

Thus for $\gamma_a \in \Gamma$, $m(\gamma_a)$ is the part of belief that supports exactly γ_a, i.e. the fact that the true value of ω is in γ_a. However, due to the ambiguity of observations, there is no insight about the subsets of γ_a. The first condition reflects the fact that no belief should be committed to the empty set $\emptyset$ and the second condition reflects that the total belief has measure one.
Notice that $m(\gamma_a)$ and $m(\gamma_b)$ can be both equal to zero even if $m(\gamma_a \cup \gamma_b) \neq 0$. Furthermore, $m(\cdot)$ is not monotone under inclusion, i.e. $\gamma_a \subset \gamma_b$ does not imply $m(\gamma_a) < m(\gamma_b)$.
Let us now define a belief function.
Belief Function - A function $Bel : 2^{\Omega} \rightarrow [0, 1]$ is called belief function over Ω if it satisfies the following relationship:

$$Bel(\gamma_a) = \sum_{\gamma_b \subseteq \gamma_a} m(\gamma_b) \quad (2.3)$$

This function quantifies the total specific amount of belief supporting the event, and it is often taken into account in the decision making process after data aggregation has been performed [28].

The main criticism to Shafer formulation concerns the application of the Dempster-Shafer (DS) combination rule. In fact, whenever there is a strong conflict between sources to be combined, the straightforward application of DS combination rule can lead to pathological behaviors, eventually reinforcing the opinion with minimum belief [29].

To face such an issue, Philip Smets [30] proposed the *Transferable Belief Model* (TBM). The TBM theory, like the Shafer formulation, relies on the concept of BBA, but relaxes the assumption of $m(\emptyset) = 0$. This allows to explicitly take into account the level of contradiction in the information sources.

Within the TBM model it is possible to combine different and contradictory information sources by composing the masses associated to each source, by means of the so called Smets operator. In the TBM, the combination rule is, defined as follows:

$$m_{ij}(\gamma_a) \triangleq (m_i \otimes m_j)(\gamma_a) = \sum_{\substack{\gamma_b, \gamma_c \\ \gamma_b \cap \gamma_c = \gamma_a}} m_i(\gamma_b) m_j(\gamma_c). \tag{2.4}$$

The main drawback of these approaches however, is that the power set is exponential in the number of elements of Ω. Such an issue often limits the applicability of these methodologies, nevertheless in the literature some approaches aimed to keep the complexity down have been introduced [58]. Moreover, computational advantages can be obtained reducing the power set only to those subsets supported by evidences acquired at each step of the inference algorithm.

2.3 Agility Analysis

In this section, mass distribution on the knowledge model, during real-time evidence acquisition, is discussed. The analysis of mass distribution characteristics leads to the identification of metrics that can be employed to improve Situation Awareness process.

2.3.1 Inability to discriminate situations

First matter taken into account is related to the capability of evidences contemplated in the model to discriminate situations of interest. For example, if the goal of a system is to classify a military platform and the only evidence taken into account by the model is velocity measurement, the system could be able to distinguish between an aircraft and a wheeled means of transport, but it probably will not be able to discriminate between a car and a motorbike. This kind of shortcoming in classification, could be imputed to:

- the lack of evidences available from field sensors;
- a wrong knowledge model.

The effect of such a kind of ineffective classification at run-time is that the inference algorithm posts great part of mass, and consequently belief, on a not-atomic subset, $\gamma_i : |\gamma_i| > 1$, of the power set Γ.

When a model is well-defined and evidences gathered are sufficient to classify situations of interest, as well as new evidences are acquired, the mass distribution converges towards an atomic subset of the power set. If this does not happen, and mass distribution converge towards a non-atomic subset, it means that one of the two mentioned cases are occurring.

In case of lacking evidence, to increase user awareness, it could be effective to cue the user or the system, to search for new, missing evidences. The following optimization function could be used to maximize awareness on situations:

$$f(x) = \alpha\Delta\frac{m(\gamma_i)}{\sum m(\gamma_j)} + \beta search\_effort \tag{2.5}$$

where γ_i are all atomic subsets of the power set Ω, γ_j are the not-atomic ones, Δ expresses the variation of mass distribution on the subsets and $search\_effort$ is a measure of the effort to search for new evidences. Hence, maximizing $f(x)$, means to maximize convergence of mass distribution towards atomic subsets, i.e., precise classifications, with less effort as possible.

Indeed, if a model cannot distinguish between two or more situations, it is easily recognizable as their support is represented exactly by same evidences. In this case, it is worth to re-analyze knowledge about the domain of interest.

2.3.2 Mass re-allocation

If the model is running in an on-line mode, mass re-allocation can be adopted in order to allow the algorithm to change its idea about classifications. The contradiction is represented in the mass of the empty set. The universal set mass is the index of the total ignorance obtained from data.

In this work we consider the Dempster-Shafer combination rule for the aggregation of data coming from the field with last results elaborated by the inference algorithm.

Results obtained are affected by the previous knowledge generated by past evidences. The idea is to move all the mass from the empty set to the universal one. In this way, at each step, new situations inside the model can be considered.

2.3.3 Closed world vs Open world assumption

A very strict assumption in Dempster-Shafer formulation is that situations of interest must be exhaustive and mutually exclusive. Building such a kind of models is very difficult, if not impossible, unless to restrict automated reasoning to a small and well-known domain, where the closed world assumption is feasible. The closed world assumption states that all possible situations are modeled and that any other situation cannot exist.

In order to avoid such a strict and not realistic assumption, Smets introduced the *empty set* $\{\emptyset\}$ among plausible subsets of the power set Γ. In Smets formulation, the empty set mass is directly dependent on conflicting evidence acquired from the field: if $m(\emptyset) > 0$

- it might mean that there is some underlying conflict between the sources that are combined, in order to produce the BBA m;
- the open world assumption is supported: Ω might not be exhaustive, i.e., it might not contain all the possibilities. Under this interpretation, being $\{\emptyset\}$ the complement of Γ, the mass $m(\emptyset) > 0$ represents modeling errors, signifying that the truth might not be contained in Ω.

The mass of the empty set can be computed as follows:

$$m_{ij}(\emptyset) = 1 - \sum_{\substack{\gamma_a \neq \emptyset \\ \gamma_a \in \Gamma}} m_{ij}(\gamma_a) \tag{2.6}$$

If $m(\emptyset) > 0$, because of inconsistencies in evidence acquired up to that time, it might mean that:

- there are problems in sources gathering evidences: for example, one of the source produces wrong output measures;
- the situation observed evolves with time: for example, in time interval $[t_0, t_1]$ evidence acquired by the system correctly support a certain situation γ_i, then, when the situation evolves and becomes γ_j, new evidences gathered in time interval $[t_1, t_2]$, supporting γ_j, result to be in contrast with those related to $[t_0, t_1]$, and cause the empty set mass to increase and converge towards 1.

In the mentioned cases, the knowledge model the system refers to is correct, and the mass of the empty set increases because of malfunctioning sensors, or because the system reasons about time-dependent situations, adopting a time-independent model. In this case, a solution to allow the system to classify a certain situation γ_i, and then correctly classify γ_j, without erroneously thinking that evidence are in conflict, is to transfer the empty set mass to the greater subset of the power set Γ, i.e., the one expressing the highest degree of ignorance (all values of Ω are plausible), so that the system can *change itsidea* about on-going observed situation.

Another reason explaining $m(\emptyset) > 0$ is that the knowledge model employed does not suit situations observed. In this case, transferring the empty set mass to the ignorance set does not lead to a correct classification of a new situation, but causes again the increase of $m(\emptyset)$. For what stated before, such a kind of cycles can be regarded as a metric to identify modelling errors and trigger learning process for real-time model correction.

2.4 A Case Study within Critical Infrastructure Protection

In this section, a simple case study, driven from the Critical Infrastructure domain, as well as experimental results supporting previous considerations are presented.

Consider the case study depicted in Figure 2.3: a power grid is controlled by a Supervisory Control And Data Acquisition (SCADA) system through some Remote Terminal Units (RTUs). The connection between SCADA system and RTUs is granted by a telecommunication infrastructure (TLC). Another infrastructure (i.e. train transport system) depends on the power grid controlled by the SCADA.

We assume that 3 possible types anomalies can be detected by 3 different intelligent sensors: alarms generated by the SCADA (X1), TLC network alarms (X2), power grid alarms (X3). The resulting anomaly vector is $v = [X1, X2, X3]$. For what concerns the causes, 4 events have been considered: power grid failure (H1), train transport system failure (H2), cyber attack to

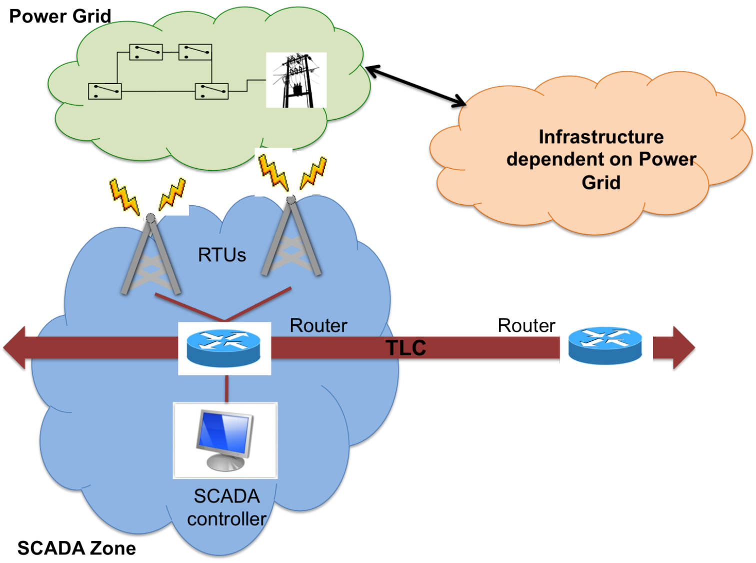


Figure 2.3: Case Study

the telecommunication system (H3), telecommunication system failure (H4). The model for the case study presented is shown in Figure 2.4, and it is the bipartite graph $G = (H, X, E)$ where H and X are the set of 4 possible causes and 3 faults, respectively and E only contains direct edges in the form (h_i, x_j), where $h_i \in H$ and $x_j \in X$.

For the particular case study, the variables of interest are: $\Omega = \{H1, H1, H3, H4\}$ and the power set: $2^{\Omega} = \Gamma : \{\emptyset, \{H1\}, \{H2\}, \{H3\}, \{H4\}, \{H1, H2\}, \{H1, H3\},$
$\{H1, H4\}, \{H2, H3\}, \{H2, H4\}, \{H3, H4\},$
$\{H1, H2, H3\}, \{H1, H2, H4\}, \{H2, H3, H4\},$
$\{H1, H3, H4\}, \{H1, H2, H3, H4\}\}$.

In order to apply the TBM framework, it has been provided a criterion for the assignment of masses to the elements of Γ, starting from the anomaly vector v has been provided. In particular, let Ψ_j be the subset of Ψ supported by the j-th failure (i.e., the causes which have an outgoing edge that goes into the j-th failure node), then it has been assigned to Ψ_j a mass equal to α, representing the reliability of the sensor registering the failure; and $1 - \alpha$ to

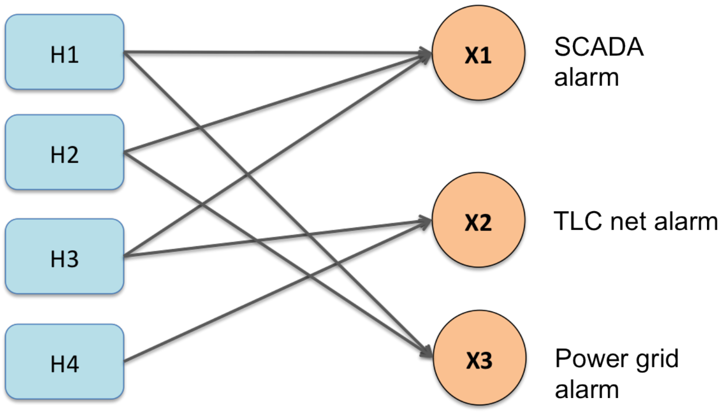


Figure 2.4: Case StudyModel

the *universal set*, i.e. the bigger subset expressing the maximum ignorance, $\{H1, H2, H3, H4\}$.

In the following examples, the reliability of sensor X1 is α, the reliability of X2 is β, the reliability of X3 is γ. The corresponding subsets supported by each sensor and that can be derived by the model are: X1 supports $H1, H2, H3$, X2 supports $H3, H4$, X3 supports $H1, H2$. This means that if the SCADA alarm raises, the classification system will assess that one among $H1, H2$ and $H3$ is the possible cause, without specifying the exact one.

In order to speed-up computations, each time evidence is gathered, only a portion of the power set is taken into account, that is the portion supported by the evidence acquired:

$$\Psi = \{\gamma_i \in \Gamma | \exists (h_i, x_j) \in E \text{ with } v_j > 0\} \tag{2.7}$$

which is the subset of Γ supported by a non-null faults, according to the graph G. In this way it is possible to reduce the size of the power set by considering only 2^{Ψ}.

Let us consider a first example for the case study introduced. We will refer to it as *CaseStudyA*. Suppose that a cyber attack occurs. It causes suddenly the alarms of SCADA and TLC systems. Suppose that, after a while, the TLC alarm goes down as dangerous traffic is no further detected and later, also the SCADA alarm ends as an operator reset the system to a normal condition. The anomaly vector at different sample time assumes the following values:

- T0: $v = [\alpha, \beta, 0]$
- T1: $v = [\alpha, \beta, 0]$
- T2: $v = [\alpha, 0, 0]$
- T3: $v = [\alpha, 0, 0]$

where it is assumed that the reliability of the SCADA sensor is $\alpha = 0.6$ and that of the IDS sensor watching the TLC network is $\beta = 0.9$.

The *belief* of the subsets of the power set are plotted in Figure 2.5 and mass distribution supporting the belief is shown in Figure 2.6.

It can be noticed that the classification system is successful in identifying the atomic subset $H3$, corresponding to the cyber attack, as the possible cause of alarms registered. All other subsets containing $H3$, present the same belief as $H3$. Such correct classification has been possible as the system received two consistent evidences at time T0 and T1 and then, even with the only evidence on $X1$, the classifier keeps on believing its previous estimate. It can be noticed that the mass distribution is more focused on $H3$ rather than the belief measure, that is properly a possibility measure.

Let us consider now another example related to the same case study: *CaseStudyB*, where the same conditions than *CaseStudyA* are verified, apart from the acquisition of an inconsistent measure at time $T2$. Suppose that only at time $T2$, $X3$ sensor rises a false alarm related to the power grid. The anomaly vector at different time samples is therefore the following:

- T0: $v = [\alpha, \beta, 0]$
- T1: $v = [\alpha, \beta, 0]$
- T2: $v = [\alpha, 0, \gamma]$
- T3: $v = [\alpha, 0, 0]$

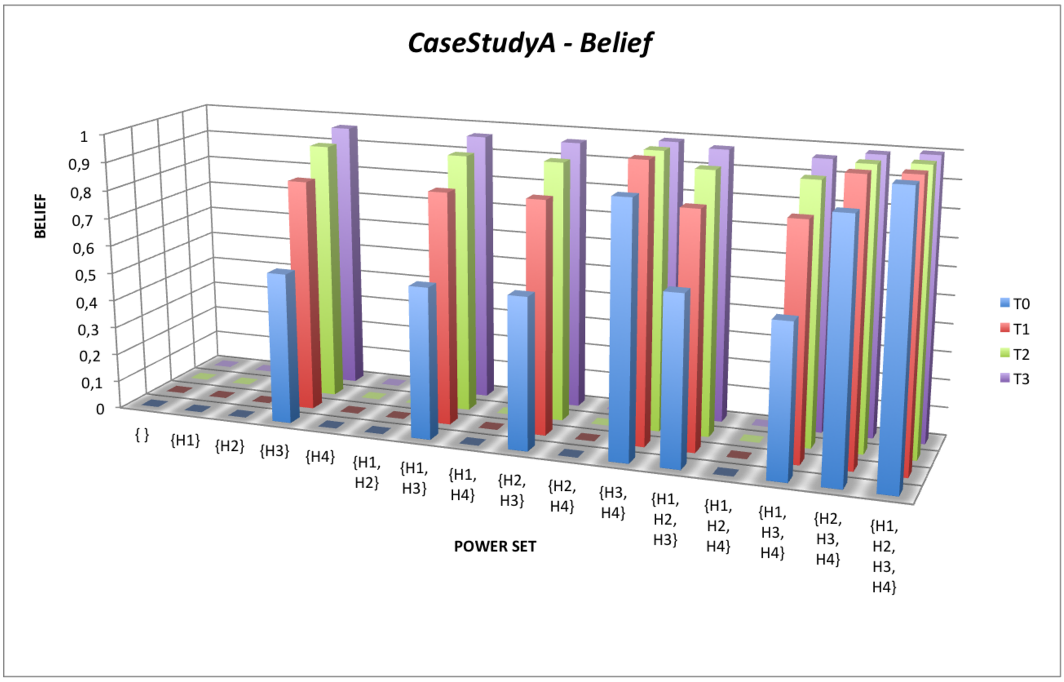


Figure 2.5: *CaseStudyA*: Beliefs supporting the power set

where $\gamma = 0.3$. The *belief* of the subsets of the power set are plotted in Figure 2.7 and mass distribution supporting the belief is shown in Figure 2.8.

It can be noticed that the system registers correctly an inconsistency in mass distribution at $T2$ and, consequently, at $T3$, due to the false alarm of $X3$, but it keeps on estimating correctly a cyber attack, $H3$, as cause of the alarms.

In the example named *CaseStudyC*, it has been assumed that at first a cyber attack occurs, arising $X1$ and $X2$ alarms, then, due to the attack, a failure on the power grid occurs causing alarm $X3$. The vector of anomaly is reported hereafter:

- T0: $v = [\alpha, \beta, 0]$
- T1: $v = [\alpha, \beta, 0]$

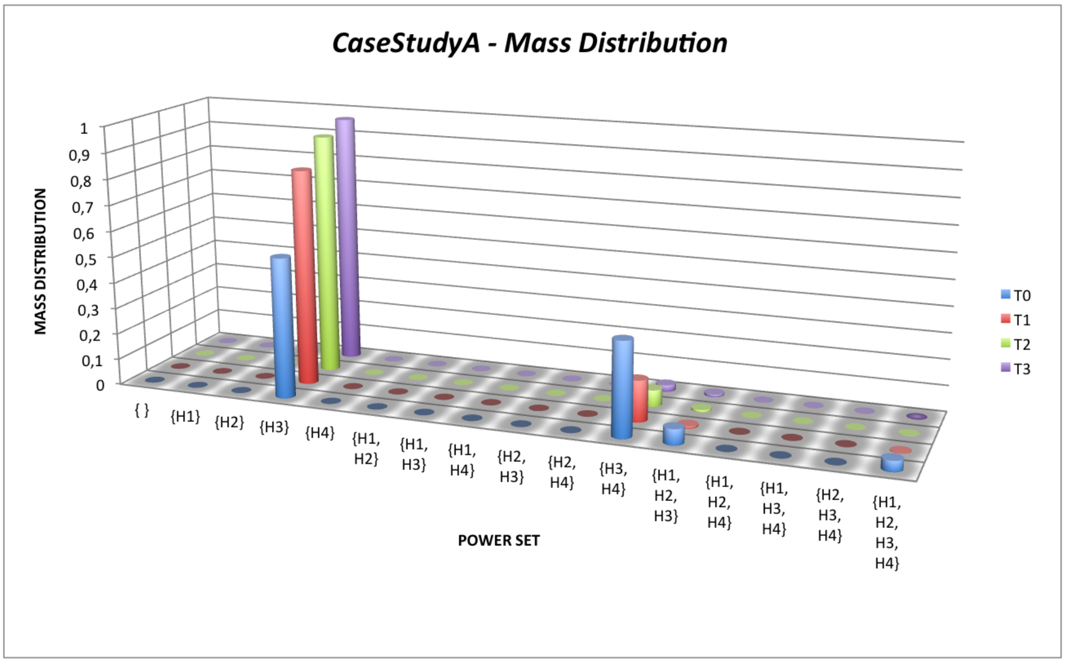


Figure 2.6: *CaseStudyA*: Mass distribution on the power set

- T2: $v = [\alpha, 0, 0]$
- T3: $v = [\alpha, 0, 0]$
- T4: $v = [0, 0, \gamma]$
- T5: $v = [0, 0, \gamma]$

where $\gamma = 0.7$. The *belief* of the subsets of the power set are plotted in Figure 2.9 and the mass distribution supporting the belief is shown in Figure 2.10.

It can be noticed that at the beginning, the classifier estimates correctly $H3$ as the most plausible cause of $X1$ and $X2$ alarms, but when $X3$ is rised, it registers high inconsistencies on the mass distribution at time samples $T4$ and $T5$, without being able to *change the idea* about the belief of a new cause of power grid alarm. This is coherent with what was stated in

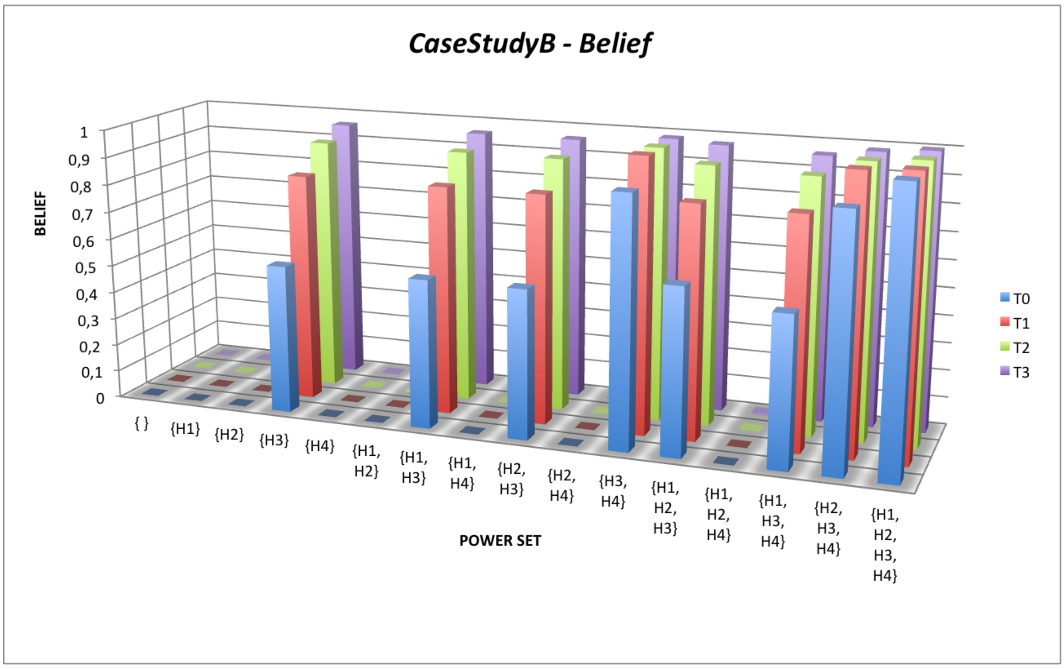


Figure 2.7: *CaseStudyB*: Beliefs supporting the power set

the previous section: Evidence Theory framework is well suited for time-independent situation recognition, but not dynamic pattern recognition.

In order to allow the system to classify $H2$ correctly, as consequent event of $H3$, in example *CaseStudyD*, we apply the mass re-distribution described in previous section. In particular, up to time $T3$ the system classifies $H3$ as cause of alarms; at time $T4$ X3 is acquired and results in contrast to previous estimations; at time $T5$, when $X3$ is again acquired, the inconsistency expressed by the empty set mass as it gets close to 1 and the system cannot classify $H2$ as the right cause. At time $T6$ the mass redistribution is applied, i.e. the empty set mass is added to the universal set $\{H1, H2, H3, H4\}$ mass, and the system changes idea about estimation and can identify correctly the subset $\{H2, H1\}$ as possible cause of $X3$ alarm, while the belief of $H3$ decreases. The anomaly vector for the example is:

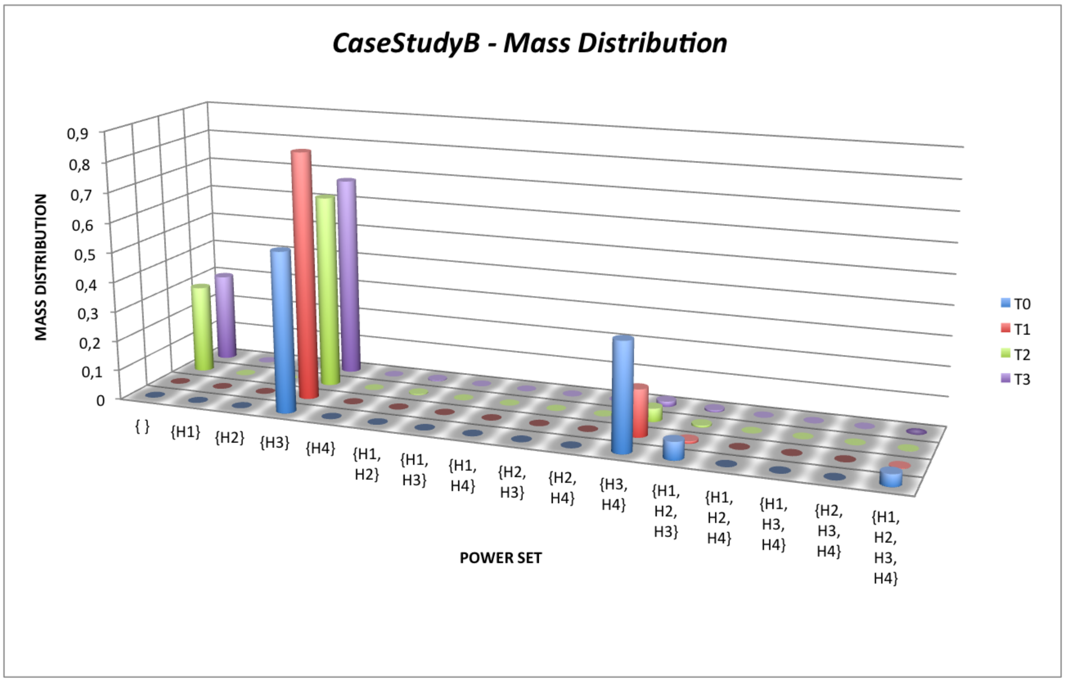


Figure 2.8: *CaseStudyB*: Mass distribution on the power set

- T0: $v = [\alpha, \beta, 0]$
- T1: $v = [\alpha, \beta, 0]$
- T2: $v = [\alpha, 0, 0]$
- T3: $v = [\alpha, 0, 0]$
- T4: $v = [0, 0, \gamma]$
- T5: $v = [0, 0, \gamma]$
- T6: mass re-distribution
- T7: $v = [0, 0, \gamma]$
- T8: mass re-distribution

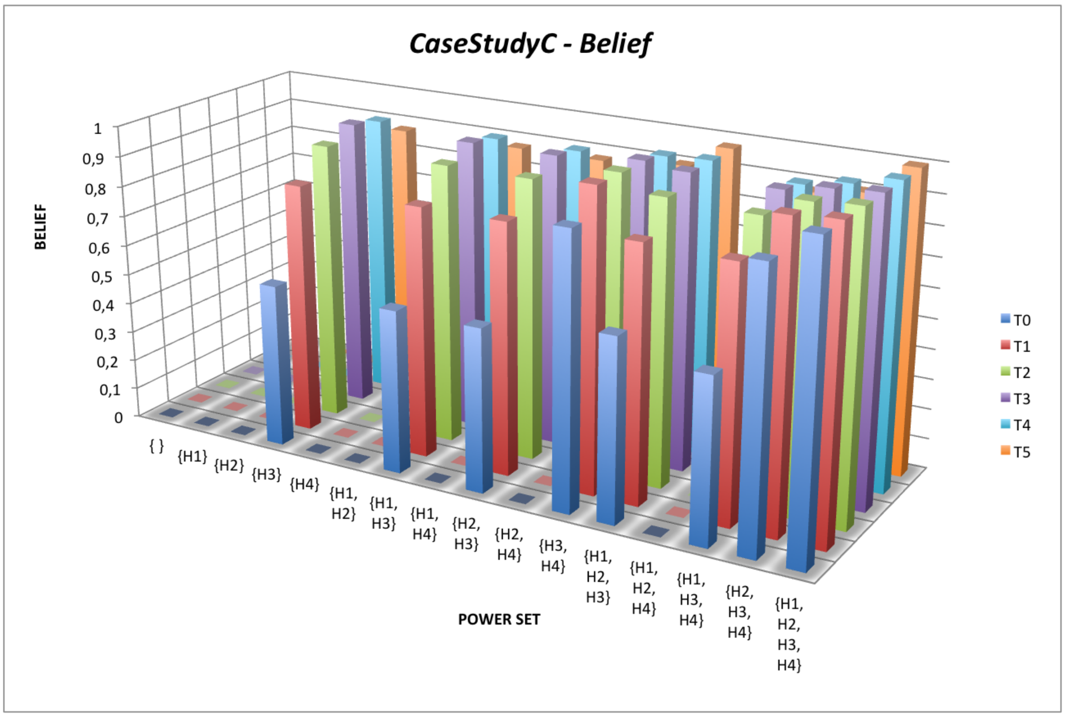


Figure 2.9: *CaseStudyC*: Beliefs supporting the power set

- T9: $v = [0, 0, \gamma]$

Note that at time $T8$ mass distribution is applied a second time in order to further reduce the inconsistency with previous estimation and it should be applied until $m(\emptyset) \to 0$.

The *belief* of the subsets of the power set are plotted in Figure 2.11 and mass distribution supporting the belief is shown in Figure 2.12.

It can be noticed, that the mass re-distribution allows to the classifier to change its idea about its classifications and to recognize different situations in different times. This achievement allows the employment of Evidence Theory in dynamic pattern recognition, besides to the domain of static classification.

In the previous section we assessed that the measurement of inconsistency related to the empty set mass is a useful metric to recognize a well-structured

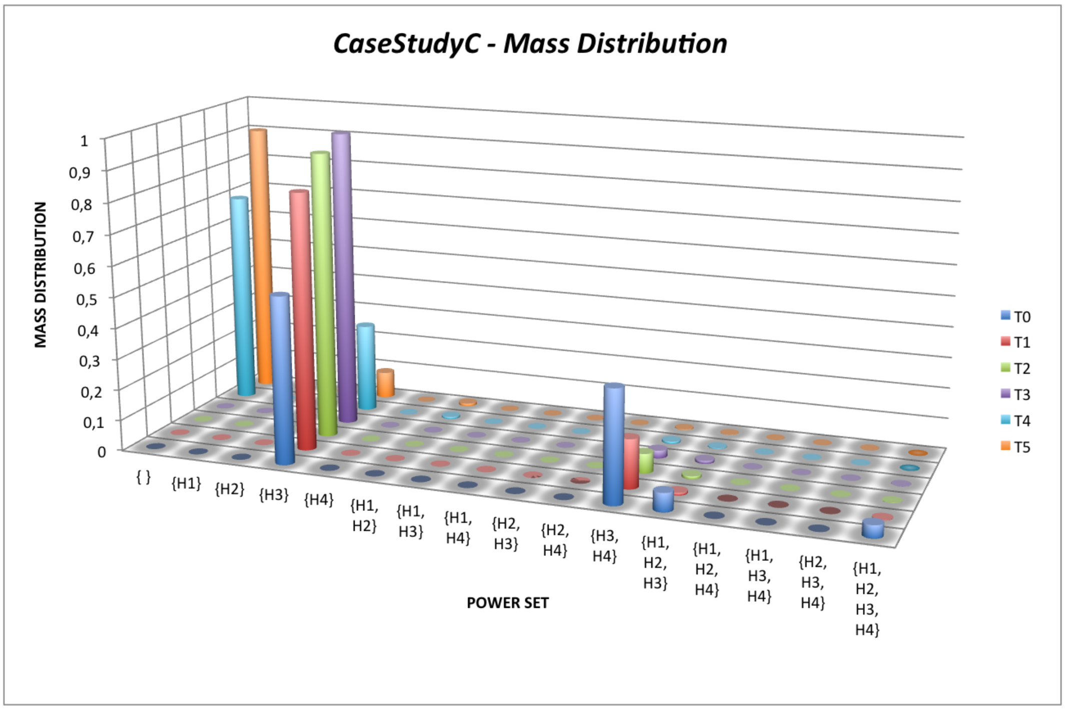


Figure 2.10: *CaseStudyC*: Mass distribution on the power set

model. Imagine that in *CaseStudyD*, at time $T4$ an evidence in contrast with a previous estimation is acquired, but, despite of *CaseStudyD*, the evidence that follows is not coherent among them as they do not fit the model: even applying mass re-distribution, the inconsistency will never go to 0 and will tend towards 1, unless continuous redistribution of inconsistent mass is applied. Such kind of trend of $m(\emptyset)$ is therefore a useful metric to highlight ineffective models for observed situations.

One possible action to take in these cases is to modify the knowledge model employed as a reference. In our example, the problem stems from raising alarms X2 and X3 at the same time. In fact, the intersection of the related hypotheses of these alarms is the empty set. For this reason, a possible approach is to increase the frame of discernment, adding a new hypothesis H5. This hypothesis H5 is linked by two edges that go into X2

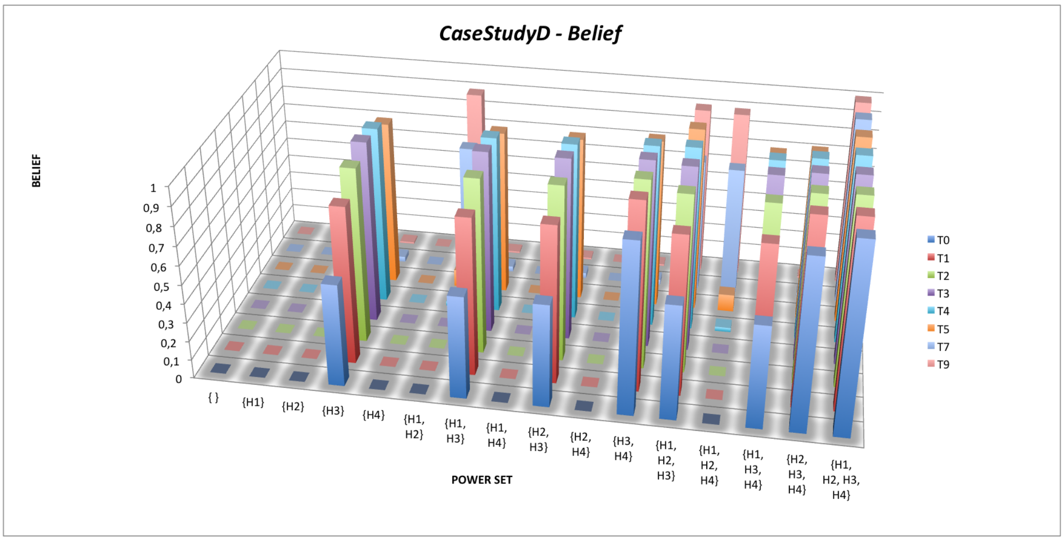


Figure 2.11: *CaseStudyD*: Beliefs supporting the power set

and X3 alarms. The new knowledge model is depicted in Figure 2.13.

Let now consider the new knowledge model and the following vector of anomalies:

T0: $v = [\alpha, \beta, 0]$

T1: $v = [\alpha, \beta, 0]$

T2: $v = [\alpha, 0, 0]$

T3: $v = [\alpha, 0, 0]$

T4: $v = [0, 0, \gamma]$

T5: $v = [0, 0, \gamma]$

T6: mass re-distribution

T7: $v = [0, 0, \gamma]$

T8: mass re-distribution

T9: $v = [0, 0, \gamma]$

T10: $v = [0, \beta, \gamma]$

T11: mass re-distribution

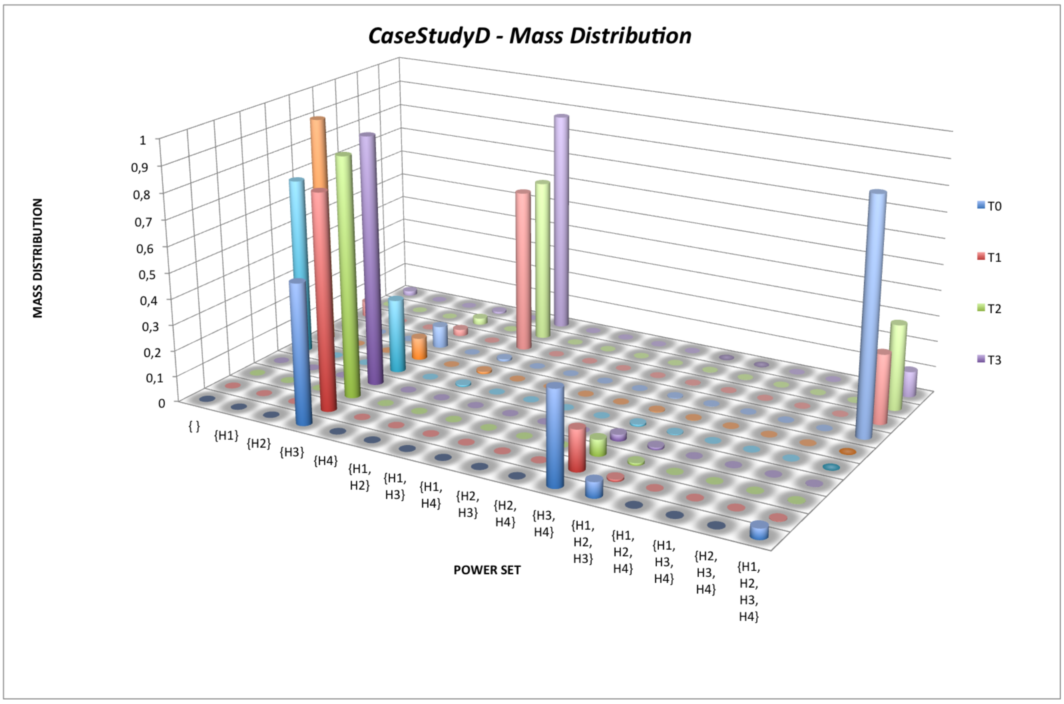


Figure 2.12: *CaseStudyD*: Mass distribution on the power set

T12: $v = [0, \beta, \gamma]$
T13: mass re-distribution with new hypothesis in the knowledge model
T14: $v = [0, \beta, \gamma]$
T15: mass re-distribution with new hypothesis in the knowledge model
T16: $v = [0, \beta, \gamma]$

The mass re-allocation process is still necessary to allow the evaluation to evolve. As shown in Table 2.1, applying the mass transfer is not enough to model possible causes occurring in the field. As soon as it is noticed that the empty set mass increases and decreases with mass re-distribution, at time step T13, the new hypothesis $H5$ is introduced and, at T14, the higher value of mass is allocated on H5 hypothesis. The empty-set mass is reduced, and this is still true at time T16, after another execution of mass-reallocation.

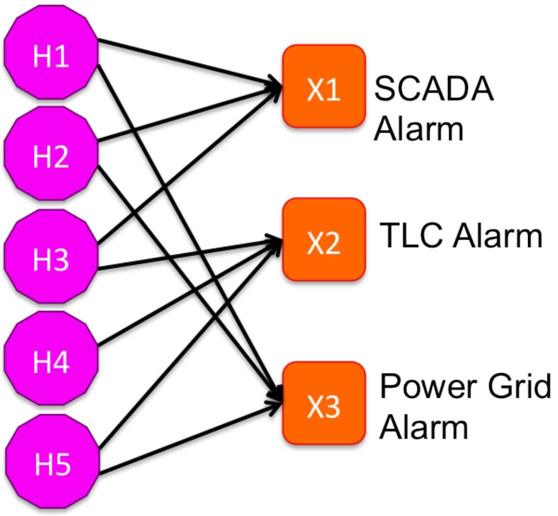


Figure 2.13: The new knowledge model, considering a frame of discernment of 5 hypotheses

The label associated to H5 can be the fire explosion described in the initial part of the example, but in real-time context and with automatic procedures, identifying the meaning associated to H5 can be very difficult without the help of human operators.

Table 2.1: Mass Assignment considering at T13 the new knowledge model

	T9	T10	T11	T12	T13	T14	T15	T16
$\{\emptyset\}$	0.0187	0.8733	0	0.6575	0	0.2520	0	0.0650
$\{H1\}$	0	0	0	0	0	0	0	0
$\{H2\}$	0	0	0	0	0	0	0	0
$\{H3\}$	0.0078	0.0024	0.0024	0.0007	0.0007	0.0002	0.0002	0.0001
$\{H4\}$	0	0	0	0	0	0	0	0
$\{H5\}$	-	-	-	-	0	0.4142	0.4142	0.7512
$\{H1, H2\}$	0.8734	0.0943	0.0943	0.0708	0.0708	0.0089	0.0089	0.0009
$\{H1, H3\}$	0	0	0	0	0	0	0	0
$\{H1, H4\}$	0	0	0	0	0	0	0	0
$\{H1, H5\}$	-	-	-	-	0	0	0	0
$\{H2, H3\}$	0	0	0	0	0	0	0	0
$\{H2, H4\}$	0	0	0	0	0	0	0	0
$\{H2, H5\}$	-	-	-	-	0	0	0	0
$\{H3, H4\}$	0.0002	0.0270	0.0270	0.2447	0.2447	0.0805	0.0805	0.0244
$\{H3, H5\}$	-	-	-	-	0	0	0	0
$\{H4, H5\}$	-	-	-	-	0	0	0	0
$\{H1, H2, H3\}$	0.0001	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
$\{H1, H2, H4\}$	0	0	0	0	0	0	0	0
$\{H1, H2, H5\}$	-	-	-	-	0	0.0460	0.0460	0.0236
$\{H1, H3, H4\}$	0	0	0	0	0	0	0	0
$\{H1, H3, H5\}$	-	-	-	-	0	0	0	0
$\{H1, H4, H5\}$	-	-	-	-	0	0	0	0
$\{H2, H3, H4\}$	0	0	0	0	0	0	0	0
$\{H2, H3, H5\}$	-	-	-	-	0	0	0	0
$\{H2, H4, H5\}$	-	-	-	-	0	0	0	0
$\{H3, H4, H5\}$	-	-	-	-	0	0.1775	0.1775	0.1266
$\{H1, H2, H3, H4\}$	0.0998	0.0030	0.8763	0.0263	0.0263	0.0008	0.0008	0.0000
$\{H1, H2, H3, H5\}$	-	-	-	-	0	0	0	0
$\{H1, H2, H4, H5\}$	-	-	-	-	0	0	0	0
$\{H1, H3, H4, H5\}$	-	-	-	-	0	0	0	0
$\{H2, H3, H4, H5\}$	-	-	-	-	0	0	0	0
$\{H1, H2, H3, H4, H5\}$	-	-	-	-	0.6575	0.0197	0.2718	0.0082

Chapter 3

Data Mining and Situation Awareness: synergies for knowledge model definition and refinement

3.1 Data mining application to Situation Awareness

One of the most felt issue in the Information age is related to the availability of huge quantities of data and the inability to correlate them to targets of interest, or to employ them for knowledge model construction that could lead to the definition of preventivate actions and countermeasures.

Management of information stored in Databases (DB) in order to discover hidden correlations, clusters of data and related descriptions is addressed by the Data Mining (DM) discipline; whereas critical situation recognition and threat evaluation, starting from heterogeneous observations is an issue addressed by Situation Awareness doctrine.

Both approaches deal with classification, but DM is generally applied off-line, on a set of data that can eventually be prepared for subsequent statistical analysis. Such kind of data structures leads to the definition and refinement of the so-called Data Warehouse (DW), where different classification techniques can be applied. Indeed, SAW manages on-line acquisition of data and tries to fit them to a previously defined model, describing the domain of interest. In other words, DM allows extracting distinguishing features of data; SAW tries to give a meaning to data, explaining why it has been gathered/observed.

The application of both approaches in the same system could be done as follows:

- adopting DM techniques to discover if and how information stored in DBs are correlated to a specific target of interest (e.g. bomb attack event);
- employing mined correlation to build knowledge models, related to the target of interest;
- adopting those models for Situation, Threat Assessment and Process Refinement (JDL levels 2, 3 and 4).

In this chapter, it is proposed a system architecture contemplating the application of DM and SAW approaches as described before.

3.2 Related Works

Only in recently published works researchers on possible influences between Data Mining and Data Fusion domains can be found.

In [34] data mining, and in particular an event co-reference process, is adopted to identify descriptions of the same event across sentences, documents, or structured databases. In McConky's work the goal is to understand if different textual descriptions, stored in different DBs, refer to the same event, through the application of a customized event extraction technique and the evaluation of an event similarity measure. Also in the proposed architecture, the correlation between an event of interest and others, stored in different DBs, must be evaluated, but the focus is on causal, temporal and spatial correlation, rather than on similarities. Despite McConky's work, information managed in our work are properly structured for data mining classification, so that issues related to natural language interpretation are not taken into account.

Moreover, while our work presents an overview of a wide system architecture, in [34] the focus is on the implementation details of textual event description correlation and user Situation Awareness is supposed to be increased by presenting to the user the collection of all existing descriptions of the same event. In this work, SAW is regarded in a more complex view: user SAW is related to understanding the undergoing situation and to evaluating its threat. In this regard, the proposed system architecture supports the user through the application of inference algorithms on agile knowledge models, which are refined through relations mined in DB records.

In [40] it is proposed a mixed approach combining data mining and Bayesian Network approach, where BNs are built and validated employing data stored in DBs, and through a refinement process performed by the user. In contrast to the work presented herein, Data Mining is meant as a learning process more than a way to discover implicit correlations among data.

In the work of Salerno [41], from the comparison and analysis of the JDL and Endsley models for Situation Awareness, a new framework for SAW is proposed. Within this framework, Data Mining techniques are mentioned for their potential to discover relationships between entities in a database and employ them to generate predictive models capable of describing what has been examined in terms of an abstract mathematical formalism (usually, a graph-theoretic construct). Nevertheless, details on Data Mining application within the architecture are not elaborated upon.

Another case study in which Data Mining is applied to SAW is reported in [42], where Data Mining is integrated with information visualization techniques. The so called visual data mining approach aims to integrate the user in the knowledge discovery process using effective and efficient visualization techniques, in order to discover anomalies in maritime traffic.

Finally, [43] presents an Advanced Driving Assistance System that analyses situational driver behavior and proposes real-time countermeasures to minimize fatalities/casualties. The system is based on Ubiquitous Data Mining (UDM) concepts. It fuses and analyses different types of information from crash data and physiological sensors to diagnose driving risks in real time. UDM is meant as the process of analyzing data from distributed and heterogeneous sources, with mobile devices or within sensor networks.

3.3 An architecture combining Data Mining and Data Fusion

In this section the overview of a system combining DM and DF techniques within the intelligence domain is presented. As mentioned before, the proposed architecture aims to combine Data Mining techniques to inference algorithms, specifically employed for Situation/Threat Assessment and pattern recognition.

The goal of Data Mining is to discover relationships among data stored in different and huge databases, and to define clusters of records, characterized by similarities with regard to certain kind of relations. Situation Awareness methodologies allow the user to recognize situations of interest, observing data acquired real-time from different and heterogeneous sensors.

The link between the two approaches can be summarized as follows: Data Mining approach is employed to define correlations among data stored in databases, and events or objects of interest for the user; mined correlations are employed to build knowledge models adopted in the Situation Awareness process.

Data Mining techniques employed in this work refer to supervised classification, where main features describing cluster of information are known and algorithm goal is to assess the belonging of data to each cluster. In particular, clusters will be defined according to temporal, spatial and causal relations.

Indeed, Situation Assessment techniques employed in this work refer to the Hidden Markov Model, which is able to describe pattern of dynamic and time-dependent situations through a graph of nodes and edges, representing states and relationships among them. In particular, the output of the data mining process (i.e., relationships among data stored) is employed to build and refine iteratively refine the Hidden Markov Model adopted in the inference process, where observations from the field feed the model and allow it to estimate the on-going situation, to project it, and to evaluate its threat according to the related impact.

Finally, relationships discovered in the Data Mining phase are regarded as cues for evidence search, contributing to JDL Level 4 functions, i.e. Process Refinement [32]. In next paragraphs, the details of the system architecture and the process of combining the methodologies as previously described, are reported.

3.3.1 System architecture overview

In Figure 3.1 is depicted the overall system architecture. The input of the system are observations gathered from the field and a Target Of Interest (TOI), specified by the user. Observations are continuously stored in intelligence databases and feed the Hidden Markov Model employed in the Situation Awareness process. A TOI represents a target of particular interest for a user, such as a specific location, a military base, or a particular event. When a user is interested in analyzing a specific TOI, the user can define this TOI and provide it as input to the proposed system in order to discover everything that could be correlated to it; this will help in developing a course-of-action to prevent related threats.

The outputs of the process are basically a TOI correlation tree and a Hidden Markov Model. The first is built by the Data Mining module, discovering different levels and kinds of correlations, among DB records and the specific TOI. The final TOI correlation tree is defined by taking into account previous correlation trees built for the same kind of TOI. The HMM is generated by the Agile Model Construction Module, that translates relationships discovered among DB records in the states and transitions of the HMM. This is then employed for pattern recognition and threat assessment related to the specific TOI. The system contemplates also an Evidence Search Module for SAW refinement. When the system is operational, it manages a set of HMMs and related TOI correlation trees, which correspond to a set of type of TOIs. Observations are employed to feed HMMs and alert the user about threats related to TOIs. Each time that a specific TOI becomes of greater interest (the user defines it as input of the system), a refinement process is started and leads to HMM and correlation tree refinement. In this way, models of the most critical TOIs are exactly those model most refined, according to stored observations and previous analysis.

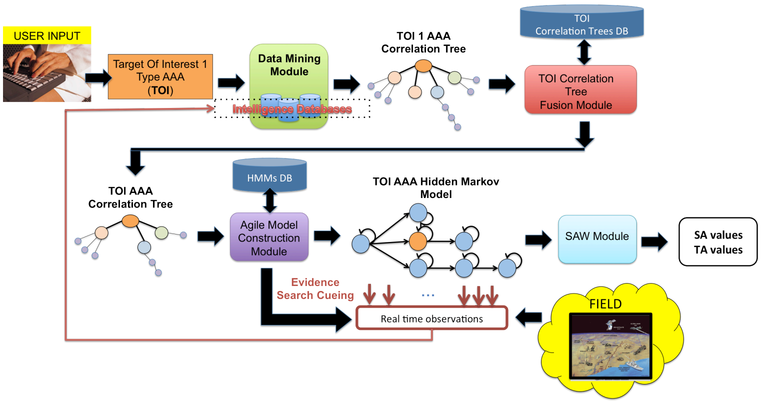


Figure 3.1: System Architecture Overview

3.3.2 Architecture components

Main components of system architecture are listed below:

- ***Target Of Interest*** **(TOI)** - It represents a target of particular interest for the user, such as a particular military base, a particular radar, platform, or a particular event. For interest, we mean the need of the user to gather all correlated information from intelligence DBs, to monitor it and to prevent possible threats related to it. The definition of TOIs depends on the particular records stored in system DBs. A TOI can be characterized be the following information:
  - Type : it specifies the type of a TOI, such as event, location, city, and action. All TOIs of the same type refer to the same correlation tree and, consequently HMM. For example, *Bomb explosion* could be the type of the TOI related to a real event of bomb explosion.

  - Name : it describes the specific TOI.
  - Time : this field defines the temporal information related to the TOI, such as the date, and time range.
  - Location : it indicates the geographical location related to the TOI.

- ***Data Mining Module*** - The main purpose of this module is to mine correlations among records of intelligence DBs and the TOI specified by the user. Data Mining techniques taken into account in this work refer to unsupervised and supervised classification, allowing to define data clusters, according to a set of given variables. The variables chosen in this work express temporal, causal, and spatial correlation.

  - Intelligence Databases : they represent the Data Warehouse and Data Marts (Data Marts contain a specific subset of data of interest from a Data Warehouse), on which the Data Mining module is based. They could be related to any kind of intelligence, such as Image Intelligence (IMINT), Communications Intelligence (COMINT), Human Intelligence (HUMINT), Signal Intelligence (SIGINT), but an intermediate process is required to prepare them for the following type of analysis:

    - *Temporal analysis*: two records are temporarily correlated if the distance between their temporal feature is within a specific time range. In particular, different kind of temporal correlations can be defined: given a time interval among day, week, month, year, the records could be fully overlapping, partially overlapping, sequential.
    - *Spatial analysis*: two records are spatially correlated if the distance between their spatial feature is within a specific geographic range. In particular, different kind of spatial correlation can be defined: city, region, country, or areas of different radius.
    - *Causal analysis*: two records are causally correlated if the probability that A causes B is higher that a certain threshold

$\delta : P(B|A) = N_{B,A}/N_A > \delta$, where $N_{B,A}$ is the number of times B occurred within a certain time range, starting from the occurrence of A, and N_A is the total number of times A occurred.

- ***TOI Correlation Tree*** - A correlation tree represents the tree of all records or TOIs correlated to a specific one, according to at least one of the relations mentioned before. The root of a correlation tree is the TOI specified by the user, the nodes connected to the root with one link represent the 1st level correlated TOIs, the nodes that are distant 2 links from the root, represent the TOIs correlated with the first-level-correlated TOIs, and so on, see Figure 3.2. The weight of the links expresses the degree of correlation with the up-level node.

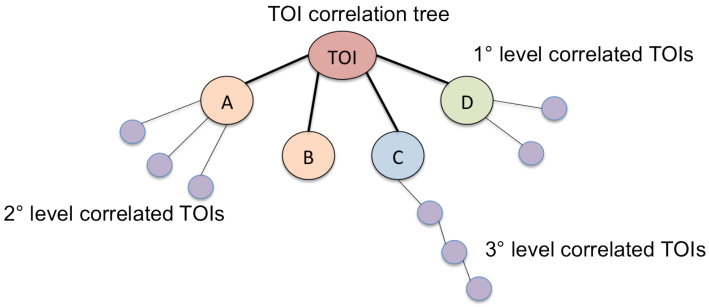


Figure 3.2: Example of TOI correlation tree

The size of the tree depends on the size of DBs and on the level of correlation to be investigated. TOI correlation trees can be related to specific TOIs, or to types of TOIs. In particular, the system first generates a specific TOI tree; then the Fusion Module compares it to the correlation tree of the TOI type; finally it updates the correlation tree for the specific type of TOI and stored it as a reference in a proper DB. In next sections, the TOI correlation tree run-time construction is described.

- ***TOI Correlation Tree DB*** - it stores the correlation trees generated by the system,according to the type of TOIs defined as input by the user. Archived TOI type trees represent the reference trees, starting from which HMMs are built. Their construction is performed by the TOI Fusion Module.

- ***TOI Correlation Tree Fusion Module*** - the goal of this module is to build the correlation trees related to the type of TOIs analyzed by the system. It takes as input a TOI correlation tree, specific for a particular TOI; it compares it with the tree of the corresponding type of TOI; it updates the archived tree for that type, with the newly generated TOI tree. The update process follows the rules described below, for each correlation level and for each node:

  - If the specific TOI tree contains a node whose type it is not contained in the TOI type tree, it is added with the same degree of correlation and with the same kind of relationships (e.g., causal, spatial, temporal);
  - If the specific TOI tree contains a node whose type is already contained in the TOI type tree, its degree of correlation is increased, and the set of relationships is updated;
  - If the type TOI tree contains a type of node that is not contained in the TOI specific tree, its degree of correlation with the up-level node is decreased, while the set of relationships is not updated;

- ***Agile Model Construction Module*** - this module's goal is to derive, from the TOI type correlation trees, Hidden Markov Models with which monitor possible threats related to analyzed TOIs. HMMs of interest are therefore those related to TOI type like events or actions (e.g., bomb attack, radar installation), whose threats need to be evaluated. Translation rules employed to build a HMM from a TOI correlation tree reflect the correlation relationships and weights mentioned above. In particular tree nodes correspond to the states of the HMM, while transition edges and probability reflect the weight of temporal, spatial, and causal correlations among tree nodes.

- ***Evidence Cueing Module*** - Once an HMM is defined, the JDL level 4 Process Refinement, a process in which sensors and inference algorithms are refined can start. In the proposed architecture, the refinement process is regarded as cueing the search for evidence that could be hidden from the observation process. In particular, the module highlights the kind of evidence the user should look for, according to the type of TOI related to each node of the HMM.

- ***SAW Module*** - The SAW module task is to take all observations coming from the field and to feed the HMMs stored in the HMM DB, in order to estimate on-going plans related to TOIs. As the approach adopted refers to Markov Theory, the inference process employs the Viterbi algorithm [33], which given a sequence of observations, estimate the most probable path and sequence of states, followed up to a certain time. The probability of a certain path corresponds to the Situation Assessment value, which expresses the confidence that a certain situation is undergoing. The SAW module then computes the Threat Assessment value, and expresses the impact that a certain situation could have, e.g.,

$$\text{TA} = \text{SA value} \times \text{damage caused.}$$

3.3.3 Run-Time Process

In a real-time context, two main processes can be identified, the SAW process and the Model Construction process. The former runs continuously, feeding HMMs stored in DBs with observations from the field, then archived in Intelligence DBs; the latter is triggered by the user, when the construction and refinement of a specific TOI tree is required. The Model Construction process affects the SAW process each time an HMM is updated. As noticed before, most refined models are related to TOIs of greater interest for the user. In Figure 3.3 the run-time process is depicted.

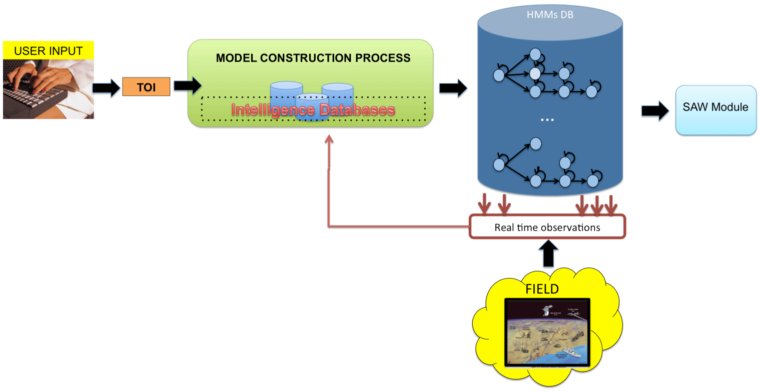


Figure 3.3: Run-time process

3.4 Data Mining Process and Target Of Interest Classification

3.4.1 Data Mining Overview

The Data Mining (DM) goal is to select, analyze and model huge quantities of data, in order to discover underlying and hidden relationships of specific interest to the DB user. DM can be regarded as an inductive methodology for information andknowledge retrieval from empirical data to general and theoretic rules to apply in wider contexts, in order to achieve a certain goal, rather than simple knowledge modeling.

The Data Mining discipline intersects different research fields such as artificial intelligence, automatic learning, DB administration, statistical, and mathematical theory. A common issue in the DM process is that data employed have often been gathered for different purposes than the one of being analyzed. For this reason, usually DB records are *dirty*, sometimes incomplete and describing features which are not always independent. Accordingly, if DM techniques would be applied on a DB that has not been previously prepared, it probably would not produce any reliable result. Such a kind of data adjustment leads to the need for the Data Warehouse (DW), which is an integrated collection of well structured data, oriented to the interest of the user.

A DW is characterized by the following elements:

- Data transformation layer;
- Data preparation and storage layer;
- Data interpretation and analysis layer;
- Data presentation layer.

The DW construction process requires manual intervention, but, after initial preparation of data, it can be partially automatized when new data need to be stored. The information analysis process usually followed in the DM approach is called Sampling, Exploring, Modifying, Modeling, and Assessment (SEMMA), proposed for the first time by the developers of the software SAS, specific for DM analysis. The SEMMA process can be summarized as follows:

- **S (Sampling)** - the huge quantity of data available is sampled in order to work on a representative subset of data;
- **E (Exploring)** - first and general data analysis are conducted in order to highlight anomalies in data, hidden dependencies (applying for example the Chi-square test), or to evaluate the harmonization of the data set to analyze;
- **M (Modifying)** - data are modified for the need to introduce particular codification and variables, or to transform quantitative variables in qualitative, and vice versa;
- **M (Modeling)** - dependency models among variables are estimated;
- **A (Assessment)** - most explicative and general models are selected.

The strength of a model is measured not only on the sampled data employed for analysis, but especially on other sets of data, stored for validation purposes. The first three phases of the SEMMA method are mainly related to the DW definition; last two phases are focused on the classification and correlation process. In the next section, a few details on classification process from a theoretical point of view are provided, and in particular, it describes how the DM process is applied in the proposed framework, with details on DW construction process and algorithms employed.

3.4.2 Classification Process

The goal of a classification process is to identify clusters of records in a DW, so that all records in the same cluster are similar among them and dissimilar from records belonging to different clusters.

As stated before, the classification process requires a similarity measure definition. The similarity can be measured with regard to records of the DW, or to features (columns) characterizing records. In the first case, the similarity is expressed by the correlation measure, in the second case, several measures for distances can be adopted, as Euclidean, Manhattan, Chebyshev or Mahalanobis distance.

The correlation measure is a statistical value, between -1 and 1, computed on a statistically representative set of data, where:

- -1 expresses the inverse correlation between the records;
- 0 expresses the independence of records;
- 1 expresses the direct correlation of records.

Among distances to measure similarity of features, the most common one is the Euclidean distance.

There are mainly two kind of classification:

- *Unsupervised classification* - clusters which records could belong to are not known;
- *Supervised classification* - the belonging cluster of each record is known.

The aim of unsupervised classification is therefore to identify cluster, starting from data stored in the DW. The result of unsupervised classification can and it is usually employed in supervised classification, whose aim is to express the model for each cluster, in terms of record features. The process followed in the unsupervised classification can be summarized as follows:

1. Identification of those features allowing to better discriminate clusters, given a set of records;

2. Definition of clusters and record aggregation, accordingly to the discriminating features identified.

For the first step of the process, a largely adopted technique is called Principal Component Analysis (PCA). It contemplates the computation of a correlation matrix V and related eigenvalues l and eigenvectors a, so that: $V\,a = l\,a$, among DW records.

The set of all eigenvectors represents the whole information related to DW records, but among them it is possible to identify a subset able to carry main distinguishing information of DW records. The subset corresponds to the principal components allowing to classify records. In fact, once the eigenvectors have been computed, it is possible to project records on the related orthogonal subspaces, and to highlight those components that can better classify records.

After the PCA process, the most relevant features for cluster identification are defined and a methodology for records aggregation must be applied. This process is called *Cluster Analysis* and different techniques exist. Hierarchical techniques defines clusters accordingly to a matrix of distances and the output is represented by a dendrogram, i.e. a graphic representation of clusters on the DW record set, like the example shown in Figure 3.4.

A non-hierarchical technique is the *K-average* method and its fuzzy version, the *C-average* method, whose output contains a description of clusters and cluster assignation for each DW record, for further details refer to [36]. In supervised classification the cluster that each DW record belongs to is known and the goal is to discover the dependency among record features and the cluster, that is to express the model of a cluster in terms of record features: $Y = f(X)$, where Y is the cluster and X are the most relevant features expressing the belonging of a record to the cluster. Once f(X) is given for a certain set of data, the same function can be applied for the classification of new different records.

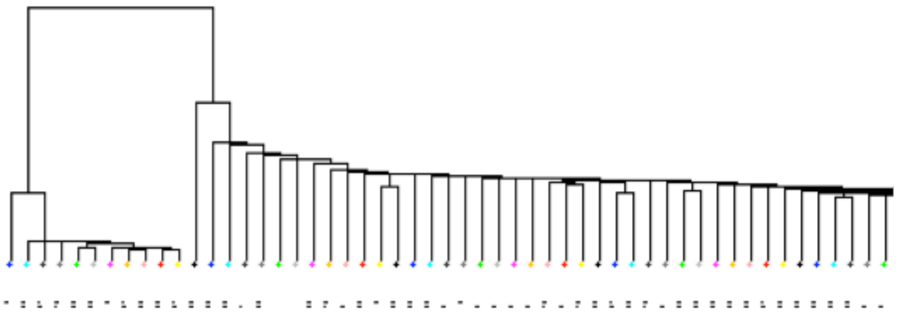


Figure 3.4: Example of dendrogram

Examples of supervised classification techniques are:

- ***Statistical methods***
  - *Linear Logistic Regression* - it is a type of analysis used for predicting the outcome of a binomial dependent variable, based on a linear function of one or more predictor variables.
  - *Linear Discriminant Analysis* - it estimates the coefficient of a linear discriminant function, through the maximization of the similarity among records of a group and of the dissimilarity between different groups.
  - *Classification Trees* - they are employed when explicative features X are qualitative, and they employ the ID3 algorithm in order to identify the most convenient path to take a decision, accordingly to what happened in the past.
- ***Learning methods***
  - *Neural Networks* - they allows estimation of non-linear functions by explaining the dependency of features and clusters, [35].

Once a classification technique has been applied, it is necessary to define thresholds (usually depending on the frequency with which records belong to each clusters) and, according to them, define the belonging cluster for each record. Moreover, the classification process quality should be evaluated through an appropriate metric, as the *T-student*. In next sections the Linear Discriminant Analysis is presented, as it is the methodology employed in the proposed system architecture. The descriptions of others techniques can be deepened in [36].

3.4.3 Linear Discriminant Analysis

The application of the Linear Discriminant Analysis is convenient when the following hypotheses are true:

- Explicative features X are quantitative variables;
- Explicative features X are characterized by normal distribution in each cluster;
- Variance and Covariance matrix are homogeneous within groups.

In this methodology it is assumed that the discriminant function $f(X)$ is linear, of the form:

$$g_i = \beta_1 x_{1i} + \cdots + \beta_p x_{pi} \tag{3.1}$$

where g represents a score, to compare with a certain threshold to define the belonging of the record i to the cluster, while the coefficients β must be estimated by the algorithm.

In particular, in the linear discriminant analysis, the coefficients are the components of the eigenvector, related to the maximum eigenvalue of the following matrix:

$$D_W^{-1}(D_B) \tag{3.2}$$

where D_W is the matrix expressing the deviation within a cluster, while D_B is the matrix expressing the deviation between clusters. In order to get a good classification, the expression Deviation Between/Deviation Within should be maximized (considering the maximum eigenvalue), exactly how the linear discriminant analysis required. In this way, records of the same clusters are as much as possible similar among them and dissimilar with regard to other records of different clusters. The computation of the deviation matrix can be performed as follows:

$$D_W = \sum_{g=1}^{G}(\sum_{i=1}^{i=n_g}(x_i^g - \bar{x}_g)^2) \tag{3.3}$$

$$D_B = \sum_{g=1}^{G}(\bar{x}^g - \bar{x})^2 n_g \tag{3.4}$$

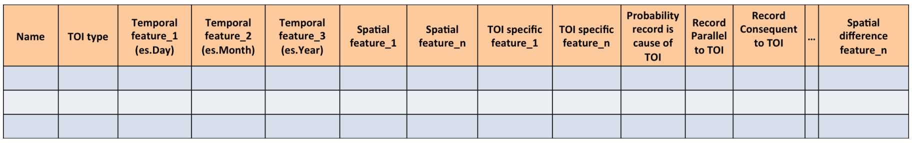
A simple threshold can be defined as the average of the scores characterizing two clusters.

3.4.4 Data Mining for Intelligence Classification

This section describes how DM methodologies are applied to the proposed framework. In the proposed system architecture, DM process is employed for the evaluation of correlations between a specific TOI, and others data stored in the intelligence DBs, accordingly to temporal, causal and spatial features.

As stated before, the first step in the DM regards the Data Warehouse definition. In this work, details of the DW construction process are omitted, as it is dependent on the particular and classified structure of intelligence DBs. The output of the first three steps of the SEMMA process is similar to Figure 3.5 which shows an example of a DW record. It is mainly characterized by the TOI variables already mentioned, such as name, type, temporal and spatial information, in addition to other variables specific for each TOI. A DW record also includes a set of variables that are necessary for cluster analysis, and which are computed each time a new TOI is specified as input. Examples of ancilary variables include the *probability that the record is associated with the specific TOI*, whose value is computed as previously described .

Even if the initial definition of the DW requires human intervention. For example, once the DW record structure has been defined, the DW update



Name	TOI type	Temporal feature_1 (es.Day)	Temporal feature_2 (es.Month)	Temporal feature_3 (es.Year)	Spatial feature_1	Spatial feature_n	TOI specific feature_1	TOI specific feature_n	Probability record is cause of TOI	Record Parallel to TOI	Record Consequent to TOI	…	Spatial difference feature_n

Figure 3.5: Example of a DW record

process can be automated, so that when new observations are acquired from the field, they can be automatically stored and structured as required by the correlation analysis process.

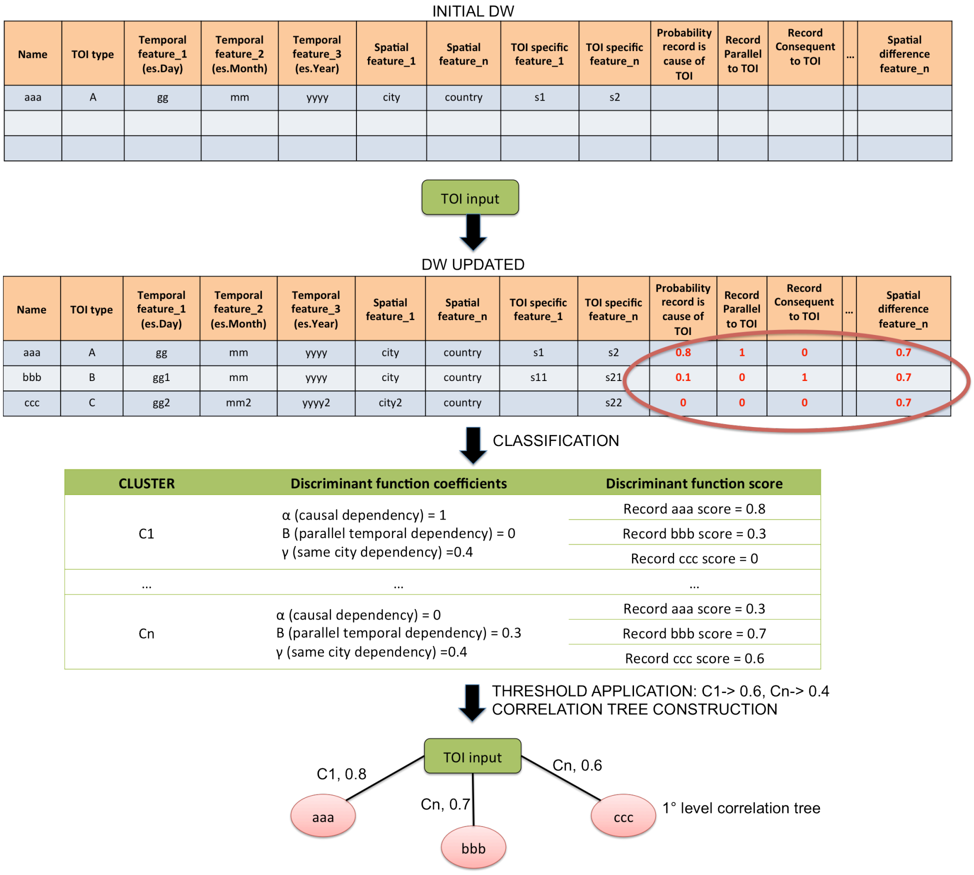
According to the run-time process characterizing the proposed system, when the user specifies a TOI of interest, the steps that should be followed by the DM module are the following, see Figure 3.6:

1. *DW update* - the variables required for spatial, causal, and temporal correlation with the TOI in input are computed, for all the DW records;

2. *Unsupervised classification* - cluster analysis is applied with regard to the variables added in the DW for spatial, temporal, and causal correlation, in order to identify clusters in the DW. Note that in this case, the Principal Component Analysis is not necessary, as the relevant features for classification are already known.

3. *Supervised classification* - linear discriminant analysis is applied to define discriminant functions of all clusters identified in the unsupervised classification. Discriminant functions are computed for each record of the DW, and scores and coefficients are stored for TOI correlation tree construction.

4. *TOI tree construction* - for each cluster, a threshold is defined expressing the level of correlation of the record with the specific TOI. All records characterized by a score higher than the related threshold are added at the first level of TOI correlation tree. The edge linking it with the TOI in input is characterized by:



INITIAL DW

Name	TOI type	Temporal feature_1 (es.Day)	Temporal feature_2 (es.Month)	Temporal feature_3 (es.Year)	Spatial feature_1	Spatial feature_n	TOI specific feature_1	TOI specific feature_n	Probability record is cause of TOI	Record Parallel to TOI	Record Consequent to TOI	...	Spatial difference feature_n
aaa	A	gg	mm	yyyy	city	country	s1	s2					

TOI input

DW UPDATED

Name	TOI type	Temporal feature_1 (es.Day)	Temporal feature_2 (es.Month)	Temporal feature_3 (es.Year)	Spatial feature_1	Spatial feature_n	TOI specific feature_1	TOI specific feature_n	Probability record is cause of TOI	Record Parallel to TOI	Record Consequent to TOI	...	Spatial difference feature_n
aaa	A	gg	mm	yyyy	city	country	s1	s2	0.8	1	0		0.7
bbb	B	gg1	mm	yyyy	city	country	s11	s21	0.1	0	1		0.7
ccc	C	gg2	mm2	yyyy2	city2	country		s22	0	0	0		0.7

CLASSIFICATION

CLUSTER	Discriminant function coefficients	Discriminant function score
C1	α (causal dependency) = 1 Β (parallel temporal dependency) = 0 γ (same city dependency) =0.4	Record aaa score = 0.8 Record bbb score = 0.3 Record ccc score = 0
...	...	...
Cn	α (causal dependency) = 0 Β (parallel temporal dependency) = 0.3 γ (same city dependency) =0.4	Record aaa score = 0.3 Record bbb score = 0.7 Record ccc score = 0.6

THRESHOLD APPLICATION: C1-> 0.6, Cn-> 0.4
CORRELATION TREE CONSTRUCTION

Figure 3.6: Exemplification of the DM process in the proposed framework

- The coefficients of the linear discriminant function, expressing how the record is correlated to the input TOI (for example, if the coefficient of the *Probability that the record is cause of the specific TOI* is zero, it means that there is no causal correlation between the record and the TOI, while, if it is different from zero, it means that causal correlation exists and depends on the coefficient value).
- The discriminant function score, expressing how much the record is correlated to the TOI, given the coefficient of the linear dis-

criminant function, i.e. a specific kind of correlation (for example, causal and partially parallel temporal correlation).

5. The size of the TOI correlation tree is increased repeating iteratively steps 1-4, using as input TOI a node of the just-generated correlation tree. Tree expansion could require high computation effort for the system, that is why the expansion of the TOI correlation tree should be stopped at the very first levels, sufficient for the proposed system analysis.

A final consideration must be done on the TOI correlation tree construction. The frequency with which TOIs are given in the input to the system is supposed to be considerably lower than that with which observations are stored in the DW, and cluster analysis is strongly influenced by data employed by supervised and unsupervised classification. This means that new observations could lead to changes in cluster models that are effectively applied only when a specific TOI is given as input to the system. As HMMs for SAW are refined only when a TOI correlation tree is updated, independently by the observation storage process, a HMM will be refined as much as the related TOI is of interest for the user.

3.5 Situation Awareness and Agile Model Construction

In this work, the SA technique adopted refers to Markov Theory and contemplates the refinement of the knowledge model on the base of hidden relationships existing on data and discovered by the Data Mining module. The adoption of Hidden Markov Models (HMM) for pattern recognition is motivated by HMM capability to model plans, causal, and temporal relations among states, and by the existence of well known algorithms for path estimation among model states.

3.5.1 Hidden Markov Models

Markov Models (MM) [37] are powerful instruments able to model a system that may assume discrete states, providing a prediction on the likelihood of sequences of states that identify a pattern or behavior. The relation among the different states is represented by means of a graph structure, where the nodes represent the states and the edges represent the allowed transitions.

The main assumption of such a methodology is that, at each time step, the system evolves, changing the state with a given probability, without memory of the past decisions; hence the edges are characterized by a weight that represents the transition probability. Therefore, the MM formalism allows calculation of the likelihood of sequences of states, assessing the possible behaviors along the temporal dimension. Note that the MM formalism does not impose any constraint on the topological structure of the graph, hence there is the possibility to model cyclic behaviors and also self pointing edges that represent the persistence of a given state over more time steps.

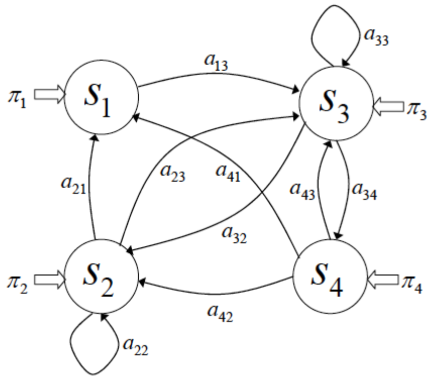
Markov Models find applications in many contexts. For instance in the financial context this instrument has been applied for the analysis of credit risk spread [38], and while in [39] the different responses to psychological tests were modeled using MMs.

MMs can be fed with observation, eventually noisy, acquired from the field, in order to influence the evolution of the system. A Markov Model with n states is defined as the 4-tuple $\{S, x(k), A, x_0\}$ where:

- $S = \{s_1, \cdots, s_n\}$ is the set of the states;
- $x(k) = [x_1(k), \cdots, x_n(k)] \in [0,1]^n$ is the probability vector associated to the states at time step k, i.e., $x_i(k)$ represents the probability that the system at time k is in the state i. Clearly it is always verified that $\sum_{i=1}^{n} x_i(k) = 1$.
- A is the state transition matrix, whose elements aij represent the probability of passing from state i to state j, i.e. the probability $p(x_j(k) = 1 | x_i(k-1) = 1)$. Due to the probabilistic structure of $x(k)$, the sum of transition probabilities must be 1 (i.e., the sum along each row of A is equal to 1).

- $x_0 \in [0,1]^n$ is the initial probability vector associated to the states at time step k = 0. Clearly the sum of these probabilities is equal to 1.

Figure 3.7 reports an example of MM with 4 states. Note that the presence of self-pointing edges for the states S1 and S4 represent the probability to persist in the corresponding state.



$$A = \begin{bmatrix} 0 & 0 & a_{13} & 0 \\ a_{21} & a_{22} & a_{23} & 0 \\ 0 & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & 0 \end{bmatrix}$$

Figure 3.7: Example of Markov Model

Within the standard MM framework, each state corresponds to an observable event; however, in many real cases, the states of the system are not directly observable. Instead, a set of indirect observations may be available, and these observations may be affected by noise. The exact motion of a vehicle, for instance, might not be directly observable. However a noisy sensorial information or a witness may provide an, although vague, hint for the reconstruction of the state. In order to face the challenge of unobservable states, in the literature the MMs have been suitably extended; the result is a Hidden Markov Model (HMM).

An HMM is described by the 7-tuple $\{S, x(k), A, x_0, O, y(k), B\}$ where $\{S, x(k), A, x_0\}$ are the same of MMs and $O, y(k), B$ are defined as:

- $O = \{o_1, \cdots, o_m\}$ is the set of m types of observation;
- $y(k) \in [0,1]^m$ is the probability vector associated to the observations at

time step k, i.e., $y_i(k)$ is the probability associated to the observation i; again $\sum_{i=1}^{m}(y_i(k) = 1)$.

- B is a $n \times m$ observation likelihood matrix, whose elements b_{ij} represent the probability of observing o_j while being in the state i, i.e., $p(y_j(k) = 1|x_i(k) = 1)$.

On HMM the most widely approach adopted for state path estimation in the Viterbi Algorithm, whose details can be found in [33].

3.5.2 Agile Model Construction Module: from Target Of Interest correlation trees to Hidden Markov Models

A crucial aspect in the proposed architecture is the construction of the HMM DB employed by the system for SA. The task is accomplished by the Agile Model Construction module that is supposed to take input from the TOI correlation tree of a specific type of TOI (e.g., bomb attack), look into the DB for the related HMM, and either substitute it or generate a new corresponding HMM to be employed in SA.

HMMs are generated only for those TOIs representing dangerous events or actions that, if monitored, could be prevented. In the construction of the model, the TOI input represents the end node of the HMM. To each node in the correlation tree, corresponds a state in the HMM, while to each link in the correlation tree corresponds a set of transition edges, according to the cluster correlation model. If a node in the tree is correlated to the TOI input, this means that it belongs to a specific cluster and that its correlation value is higher than a certain threshold. The model of the cluster is summarized by the coefficients of the linear discriminant function estimated by the supervised classification. The coefficients are then used to assign a weight and define edges in the HMM.

3.5.3 Threat Assessment

Threat Assessment is performed by evaluating the risk related to the estimated situation. In general, risk depends on the probability that a certain situation occurs and on its impact:

$$\text{Risk} = \text{Probability} \times \text{Impact}$$

In regard to the proposed system architecture, the probability of a situation corresponds to the SA value, the impact is intended as the quantification of the damage that offensive situations cause and it should take into account enemy offensive capabilities and friend defensive capabilities, as well as the influence of the environment in estimated situations.

3.5.4 Process Refinement

As stated before, the aim of process refinement is to employ results of other JDL levels, for tuning system processes and improving the quality of elaborations. In this regard, in the proposed architecture, an Evidence Search Cueing module has been introduced. It suggests to the user new evidence to be searched, according to the states of HMMs being analyzed.

Chapter 4

Situation Awareness Applications

In this section an overview on two Situation Awareness frameworks that have been designed and developed within Military domain and Critical Infrastructure Protection domain is provided .

In this regard, the state-of-the-art, system architecture, methodologies, and application issues are described for both proposed frameworks.

The first system described is called INFUSION. It is a system for the evaluation of situations and threats in simulated military scenarios and for scenario projection, in order to support the decision making process in strategic and tactical context. The system deals with fuzzy variables acquired from the on-going scenario; it adopts the Evidence Theory approach to fuse information and classify situations. Moreover, it evaluates threats by measuring the risk of on-going situations on the target of interest. INFUSION is able to foresee possible future scenarios, through the projection of the item of the simulated scenario to a desired time or position. Trajectory projection of items depends on their intent, estimated through the Bayesian approach.

The second work described in this section refers to Critical Infrastructures (CI). In the case of a natural disaster or malicious event, it is vital for the decision makers, operators and stakeholders involved in CI protection, to quickly understand to which extent the actual situation is critical and possibly what are the causes and the expected effects. In this regard, the application of Situation Awareness to CI might increase comprehension of complex interdependent behaviours. In this section an approach to understand the possible causes of outages in different and interconnected infrastructures, based on the evidence of detected failures or attacks is provided. Moreover, causes inferred are used to estimate possible undetected failures that, together with those detected, provide a better understanding of the infrastructure vulnerability, and the impact of outages. Such a kind of analysis is regarded as a useful support to identify effective countermeasures, in order to mitigate risks related to malfunctioning behaviour of critical infrastructures. The analysis is conducted within the cyber security domain.

4.1 INFUSION system

One of the key factors for success on the battlefield nowadays is the effective use of information. Usually commanders and analysts, in dynamic tactical scenarios, have to manage a huge amount of data coming from different sources related to the environment, these data may include uncertain and incomplete information, which is difficult to correlate and analyze. Within this context, tools able to collect data and perform automated reasoning could provide a considerable support to the decision-making processes and to what-if analysis.

The purpose of INFUSION system is to support users in assessing both on-going and future situations and threats related to a terrestrial battlefield. At the moment, INFUSION elaborates data coming from scenarios simulated in a test bed developed for military training. The theoretical approach adopted for evaluations draws the basis from Dempster-Shafer theory of evidence [58] and from fuzzy logic [45] to fuse uncertain information and classify on-going situations; from expert systems theory [44] to evaluate threats related to situations; from Bayesian Net (BN) theory [8] to estimate enemies intent and project the current scenario. Models of situations and criteria related to threats have been defined accordingly to NATO military doctrine [50].

In literature, few works describe implemented Data Fusion frameworks. In [46] ALARM system is described as a system for situation and threat evaluation in near real-time. In order to facilitate fast computations, the approach adopted is based on a simple comparison of acquired information with a checklist of criteria describing small events. The combination of events defines more complex situations, whose threats are evaluated with regard to the scope of situations. The system is thought to perform only a first-pass processing of inflowing information, while the most of the analysis effort is left to human operators. Also INFUSION can evaluate situations and threats in near real-time, however it refers to a recognized theoretical approach that, despite to checklists, allows the user to refine evaluations when new information is available and to handle uncertain and incomplete information.

In [47] a threat assessment process is described. The methodology is based on the Cognitive Work Domain Analysis to model enemy intent and capabilities through a Bayesian Net, and adopts the probabilistic approach to infer enemy intent. INFUSION also adopts BNs to estimate enemy intent, but it is used for enemy trajectory prediction, and therefore to foresee possible future scenarios. Threats are related not to enemy intent, but to the risk of on-going situations related to friend forces. PROGNOS system described in [48] employs probabilistic ontologies in distributed system architecture as a means to provide semantic interoperability within an intrinsically complex and uncertain environment. The main focus is the situation model definition and algorithm able to manage the model; therefore despite of INFUSION, the evaluation of risk and scenario projection are not considered.

The distinguishing features implemented for INFUSION can be designated as the capability to perform fast and light computations, suitable to warn the operator on imminent situations and threats; the capability to foresee possible future scenarios and allow the user to perform what-if analysis; the capability to operate in a distributed simulated environment, compatible with the High Level Architecture (HLA) standard [49], thereby allowing to test the quality of evaluations and their processing time, and to train operators.

4.1.1 System Architecture

INFUSION has been developed with the purpose of implementing functionalities related to levels 2 and 3 of the JDL model, i.e., Situation and Threat Assessment. In particular, INFUSION is able to analyze battlefield scenarios adequately simulated in a HLA environment, that is a distributed synthetic environment for military simulation.

The proposed system operates in HLA as a federated system for supporting commander decisions. In this regard it knows the characteristics of the battlefield and acquires all information related to the platforms, whose identification and tracking is supposed to be already assessed.

In Figure 4.1 is illustrated the configuration of the operating environment of INFUSION. The main components of the overall architecture are:

- ***Scenario Generator and Animator*** (SGA) - it allows the following operations:
  - Scenario creation and animation;
  - Management of platforms, sensors, weapons;
  - Mission planning;
  - Scenario and Data distribution;
  - Viewer Integration.
- ***INFUSION engine*** - it performs the elaborations on Situations and Threats in simulated and projected scenarios.
- ***INFUSION Human Computer Interface*** (HCI) - it reports the results of elaborations related to Situation and Threat Assessment and allows the user to specify the necessary parameters to request the projection of simulated scenarios.

All data exchanged among federated nodes conform to the HLA standard definition.

The main functionalities implemented by the INFUSION system are:

- Situation and Threat Assessment in simulated scenarios - INFUSION acquires from SGA the simulated scenario and related items, and through

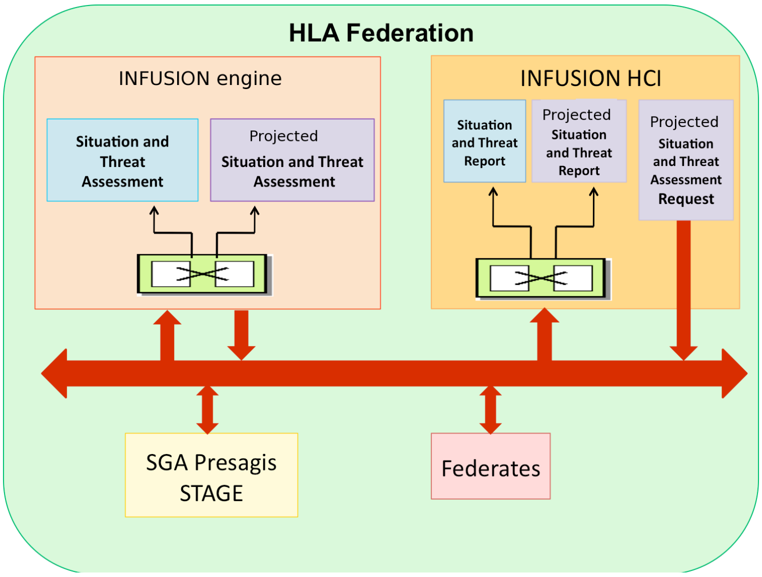

HLA Federation
INFUSION engine
Situation and Threat Assessment
Projected Situation and Threat Assessment
INFUSION HCI
Situation and Threat Report
Projected Situation and Threat Report
Projected Situation and Threat Assessment Request
SGA Presagis STAGE
Federates

Figure 4.1: INFUSION operating environment

the HCI provides, real-time estimations on on-going situations and threats, previously modeled;

- Situation and Threat Assessment in projected scenarios - INFUSION estimates the intents of the items in the simulated scenario; it generates possible future scenarios, based on intents estimated; and analyzes situations and threats in projected scenarios, in order to support what-if analysis. In particular, the user can request the projection of the simulated scenario in two different elaboration modes, corresponding to different kind of what-if analysis:

  1. *Time Mode* - the user specifies a temporal interval $\triangle T1$ and an item of friendly coalition. Starting from the simulated scenario at time T0, when the elaboration request occurs, INFUSION estimates a set of possible projected scenarios, related to time $T0 + \triangle T1$; it evaluates future situations and threats; and highlights to the user those threats minimized by each scenario,

regarding the specified item.

2. *Position Mode* - the user specifies a desired position for a certain item of friendly coalition. INFUSION projects the simulated scenario from time T0, when the elaboration request occurs, to time $T0 + \triangle T2$, when the specified item achieves the desired position; then it evaluates the future situations and threats.

The *Time Mode* elaboration helps the user to evaluate future situations and threats for a particular item of friendly coalition, answering the question *"What may happen to FriendAlpha in $\triangle T1$?"* and *"Where should FriendAlpha move to be safer?*. The *Position Mode* elaboration helps the user to evaluate the situations and threats to which a particular item of friendly coalition is subject, if it moves toward a certain position. The *Position Mode* elaboration helps the user to answer the question *"What may happen to FriendAlpha if it moves in PositionBravo?"*.

The specification of a particular friendly item to analyze is necessary to achieve near real-time performances of system. In fact, in *Time Mode* elaborations, INFUSION considers a set of possible future positions for the friendly item, corresponding to reachable positions in time interval $\triangle T1$; for each of the mentioned positions, the system generates and analyzes projected scenarios. If the system would consider more than one friendly item, it should evaluate a huge number of projected scenarios, corresponding to all possible configurations of the items, without degrading system performance. In this case, a possible solution for the reduction of the computational complexity could be reducing the number of reachable positions, considering only the most likely ones for each item.

4.1.2 Situation and Threat Assessment in Simulated Scenarios

As already mentioned in previous sections, INFUSION is able to assess situations and threats in a simulated scenario. Situations and threats recognized by INFUSION have been defined accordingly to NATO military doctrine. Threats are evaluated considering the risk of on-going situations with regard to each friendly item.

SA and TA are performed continuously, each time updated information is acquired from SGA. The patterns of situations and threats analyzed by the system are described hereafter:

- *Engagement* - an item engages another one, the target, when it moves towards the target and the rifle range of its weapon is smaller than the relative distance between the items. The threat related to this situation depends on the capability of kill of the weapon, with respect to the target platform, and to the visibility of the target with respect to the topography.
- *Collision* - a collision between an item and another one, its target, can be estimated considering the bearing of the trajectory followed by the item, compared to the one of its target, the relative speeds of the items, the visibility of the target with regard to the topography. The threat related to this situation depends on the relative mass of the platforms.
- *Encirclement* - the situation and threat of encirclement can be evaluated considering the number of items surrounding a certain target, their relative positions and bearing, as well as the engagement of the target by each of the items involved in the encirclement.

The information available to the system for SA and TA are reported hereafter:

- Digital Terrain Elevation Data (DTED);
- Item ID;
- Item coalition (Friend/Hostile);
- Item position;
- Item speed;
- Item direction of movement;
- Item weapon information;
- Item platform information.

The mentioned data are managed as fuzzy variables, as it is assumed that they are not completely accurate, but they carry a certain degree of uncertainty. This uncertainty is then employed as a measure of trustiness in all evaluations.

4.1.3 Evidence Theory to assess Situations in INFUSION system

Criteria describing patterns and information acquired from the scenario are employed by the system to classify the on-going situation. The approach adopted refers to Evidence Theory, allowing the identification of causes that generate evidence acquired from the field. Evidence can be related to heterogeneous information and acquired at different times, even asynchronously.

With regard to INFUSION, the correspondence between evidence (information available) and causes (patterns of situations) is defined in accordance to a set of criteria, drawn from military doctrine, and describing situations which are not supposed to change with time.

The formal description of the algorithm is reported hereafter. Let a bipartite graph G = (V1, V2, E) where:

- V1 represents the set of patterns of situations (Engagement, Collision, Encirclement) that are considered mutually exclusive and exhaustive. Note that this assumption can seem quite strict, but for the purpose of this work it is assumed they are exhaustive for the domain of interest, in fact the enemy can be involved in only those three situations; all other kind of actions generate evidence that will support the null set representing ignorance or contradiction.

- V2 is the set of fuzzified information available from the scenario.

- E contains direct edges in the form $(v1_i; v2_j)$, where $v1_i \in V1$ and $v2_j \in V2$, see Figure 4.2.

Edges express correlation between situations and evidence. When specific evidence is acquired from the field, the corresponding situations are supported. In Evidence Theory, the structure of G (model of situations) is assumed to be fixed, therefore time dependent patterns cannot be modeled.

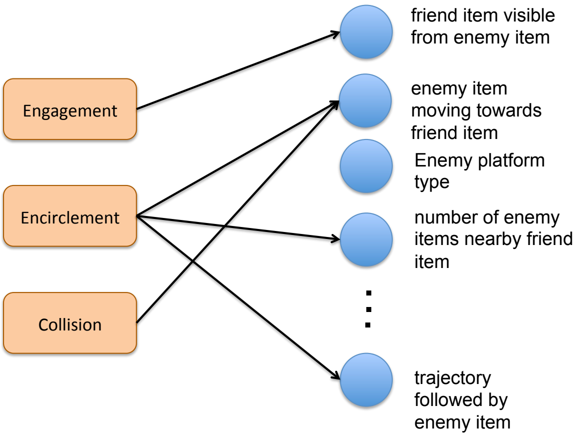

Engagement
Encirclement
Collision
friend item visible from enemy item
enemy item moving towards friend item
Enemy platform type
number of enemy items nearby friend item
trajectory followed by enemy item

Figure 4.2: Correspondence between Situations and Evidences in Evidence Theory

Let $x0 \in R^n$ be the vector of information acquired from the scenario at a certain time stamp, where the $i-th$ component $x_i0 \in [0,1]$ represents the trustiness of related information.

Let Γ be the power set containing all possible subsets of situation patterns, including the null set. The cardinality of Γ is $2|V1|$. Usually computations are performed on the whole power set, in INFUSION it has been introduced the reduced power set Ω, i.e. a set containing all subsets of Γ related only to those elements of V1 supported by the evidence vector. The cardinality of Ω is therefore smaller than $2|V1|$ and improves computation performances.

Let *Basic Belief Assignment* (BBA) be a function $m : \Gamma \to [0,1]$, expressing the part of belief supporting a subset of Γ, also named as mass. BBA function must ensure that the total belief is equal to 1:

$$\sum_{\gamma_a \in \Gamma} m(\gamma_a) = 1$$

Note that the BBA employed by the system takes into account the trustiness of information acquired and the cardinality of patterns supported by the specific information evidence. Moreover, contradictory information supports the null set according to Smets idea of open world assumption and contradiction quantification [30] (i.e. not all situations could be modeled, information acquired could be uncertain and therefore contradictory). When the mass of the null set reaches high values it means that the on-going situations has not been modeled, or that information acquired is in contrast with the past one, i.e., the on-going situation has changed.

In both cases INFUSION redistributes the null set mass on the power set allowing the algorithm to *change idea* about its evaluations.

The algorithm implemented by INFUSION system executes the following steps:

1. It acquires the evidence vector $x0$ from the simulated scenario;

2. It computes masses for each subset of the reduced power set Ω, through the BBA and in accordance to $x0$.

3. It computes the belief (Bel) that a certain situation is going on through the following equation:

$$Bel(\gamma_a) = \sum_{\gamma_b \subseteq \gamma_a} f(\gamma_b)$$

The subset of Γ with the highest belief, is the most likely to be performed by enemy forces against a certain friendly coalition item.

4.1.4 Threat Assessment in INFUSION system

Threat Assessment is performed taking into account the risk of on-going situations, with regard to each friendly item. In general, risk depends on the probability that a certain situation occurs and on its impact:

$$\text{Risk} = \text{Probability} \times \text{Impact}$$

With regard to INFUSION, the mentioned probability is the result of Situation Assessment process, the impact is estimated taking into account the following elements, for each friendly item:

- enemy offensive capabilities and friend defensive capabilities;

- influence of environment in operations (e.g., visibility between two platforms depends on terrain elevation).

Impact quantifies the damage that offensive actions cause on friendly forces. INFUSION performs TA with regard to on-going situations and also to the predicted ones, so that threats can be adopted by users as a measure in action planning and decision making.

4.1.5 Situation and Threat Assessment in Projected Scenarios

Besides evaluate situations and threats in simulated scenarios, INFUSION is able to evaluate them in projected scenarios, in order to help the user in *what-if* analysis and to support the decision making process.

Through INFUSION HCI, the user can request a scenario projection according to two different elaboration modes.

The detailed steps followed in both computation modes are described below:

- *Time Mode* - parameters specified: friendly item and $\triangle T1$

  1. A set of possible future positions for the specified item is computed (*if-positions*). The positions are achievable by the item at time $T0 + \triangle T1$
  2. The intent of enemy coalition items, with regard to the specified one, is estimated. Two possible intents can be estimated, offensive and not-offensive
  3. According to the estimated intent of enemy items and to different positions of a specified friendly item, several scenarios are generated (*what-if scenarios*) and, for each of them, an enemy item trajectory is computed in the time interval $\triangle T1$
  4. Situations and threats are evaluated in the projected scenarios and the most favorable scenarios, with regard to the selected item, and are highlighted by the system.

- *Position Mode* - parameters specified: friendly item and its desired future position

  1. The specified item is projected in the specified position (if-position)
  2. The temporal interval $\triangle T2$ is computed, according to the time necessary to the item to achieve the specified position
  3. The intent of enemy coalition items, with regard to the specified one, is estimated. Two possible intents can be estimated, offensive and not-offensive
  4. According to the estimated intents of enemy items and to the position of the specified item, their position at time $T0 + \triangle T2$ are computed
  5. Situations and threats are evaluated in the projected scenario

In Figure 4.3 both the elaboration modes are illustrated.

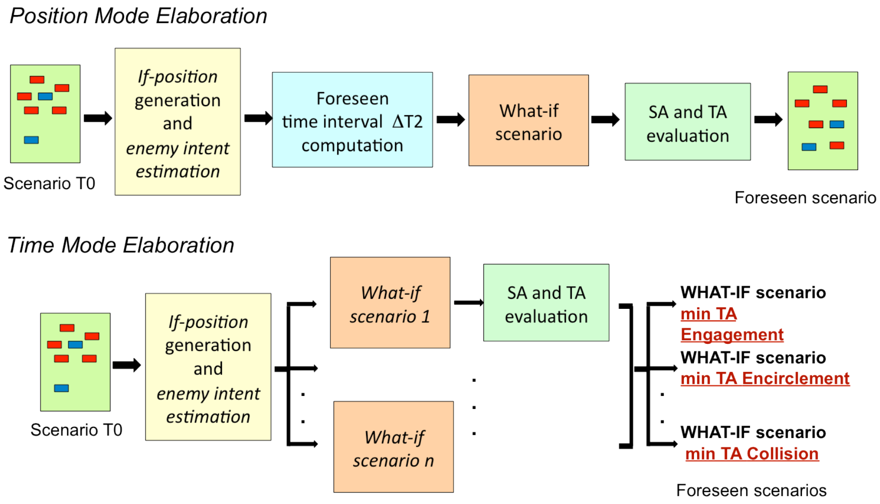


Figure 4.3: Scenario projection algorithm: elaboration modes

4.1.6 Intent Estimation with Bayesian Networks

A fundamental step in the algorithm for scenario projection is the intent estimation of enemy items, with regard to the friendly one specified by the user. In particular, given a simulated scenario, INFUSION associates an intent to each enemy item, between offensive and not-offensive, and according to that, the system projects item trajectories.

In order to perform intent estimation, the Bayesian approach has been adopted. The reader could notice that inference technique adopted for intent estimation is different from the one adopted for Situation Assessment. The main reason for this difference is that the interference technique depends on the empirical knowledge structure related to situations and intent. Knowledge about intent is hierarchical, therefore a cause-effect model, like BNs is well suited for its representation; on the other hand, knowledge about situations that INFUSION classifies is flat, exactly like the model structure employed in Evidence Theory approach.

A Bayesian Net (BN) is a directed acyclic graph, which represents the probabilistic dependencies among a set of random variables. Each node, representing a random variable in the BN, defines alternative propositions. The propositions must be mutually exclusive and collectively exhaustive, that is, all possibilities must be represented. The directed connections between nodes represent the conditional probability of inferring the existence of one node (being pointed to), given the existence of another node. Each node can have multiple inputs, that is, many parents. Each node in the network stores its probability distribution, given its direct parents and any associated evidence. BNs provide a formal method for reasoning about partial belief under conditions of uncertainty. In this method, propositions are given numerical parameters signifying the degree of belief accorded them from a given set of knowledge. The parameters are combined and manipulated accordingly to the rules of probability theory. In the context of Bayesian statistics, the probability is interpreted as the degree of belief. The BN employed by INFUSION is depicted in Figure 4.4.

Leaf nodes are those on which evidence can be posted directly, according to data gathered from the scenario. A priori probabilities, the *Conditional*

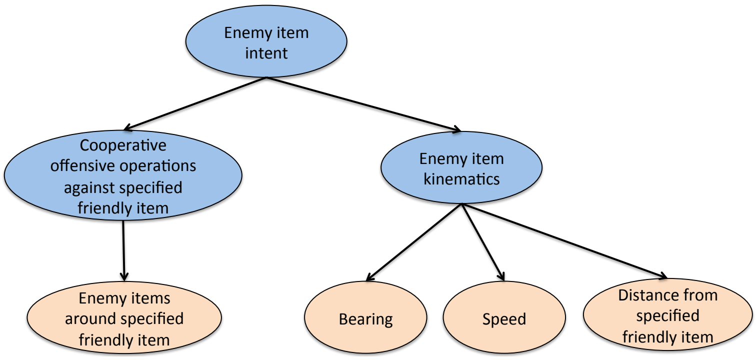

Enemy item intent
Cooperative offensive operations against specified friendly item
Enemy item kinematics
Enemy items around specified friendly item
Bearing
Speed
Distance from specified friendly item

Figure 4.4: Bayesian Net employed for intent estimation

Probability Tables (CPTs), as well as the parameters employed for computations, have been defined by experts working in the military domain. The description of nodes in the BN is reported hereafter:

- *Enemy items around specified friendly item* - the node expresses the presence of enemy items in the nearby of the specified friendly item. The related states are *many* and *few*.
- *Cooperative offensive operations against specified friendly item* - the node expresses the presence of ongoing cooperative offensive operations against the specified friendly item. The related states are *yes* and *no*.
- *Bearing* - the node indicates if the enemy item is moving towards the specified friendly item. The related states are *offensive* and *not-offensive*.
- *Speed* - the node represents the enemy moving speed. The related states are *high* and *low*.
- *Distance from specified friendly item* - the node expresses the distance between the enemy item and the specified friendly item. The related states are *near* and *far*.

- *Enemy item kinematics* - the node expresses the features of movement of the enemy item with regard to the specified friendly item. The related states are *offensive* and *not-offensive.*
- *Enemy item intent* - the node represents the intent of the enemy item with regard to the specified friendly item. The related states are *offensive* and *not-offensive.*

When a leaf node receives evidence from the scenario, it updates its own belief, and then propagate its evidence to its neighboring nodes, in order to update the belief in the whole net and estimate the intent of the related enemy item with regard to the friendly one being analyzed. The evidence propagation algorithm adopted by INFUSION is described hereafter [8]. Given an evidence e_x on the node X of the BN, the algorithm revises the belief vector for each node, i.e. it computes $p(Y|e_x)$, for each node Y. For each node X in the BN, with p number of states, the following variables have been defined, as represented in 4.5:

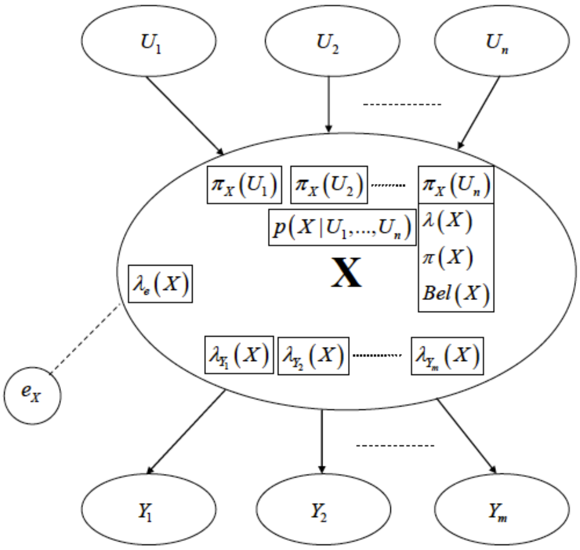


Figure 4.5: BN node structure

- p-ary π vector $\pi(X)$
- p-ary λ vector $\lambda(X)$
- p-ary belief vector $Bel(X)$
- p-ary evidence vector $\lambda_e(X)$
- $p \times n$ CPT $p(X|U_1, \cdots, U_n)$, if X has n parents $U_1, \cdots, U_n$
- q-ary parent π vector $\pi X(U_i)$, for each parent U_i with q number of states
- p-ary child λ vector $\lambda_{Y_i}(X)$, for each child Y_i.

At first, the algorithm initializes the BN performing the following steps:

- set the π and belief vectors of each root node X, without any parents, to the related prior probability distribution
- compute the π and belief vectors of the rest of the nodes performing the following:
  - set each parent π vector $\pi_X(U_i)$ to the π vector of U_i
  - set its π vector $\pi(X)$ to 4.1 if X has parents $U_1, U_2, \cdots, U_n$

    $$\sum p(X|u_1, u_2, \cdots, u_n) \prod \pi_X(U_i) \tag{4.1}$$

    where the sum is performed on $u_1, u_2, \cdots, u_n$, and the product on $i = 1$ to n
  - set its belief vector $Bel(X)$ equal to its π vector

Once the BN has been initialized, the algorithm operates as follows, each time an evidence $\lambda_e(X)$ on the node X is acquired from the scenario:

- revise the beliefs of X:
  - compute $\lambda(X)$ as the product of all its child λ vectors and the evidence vector
  - compute $\pi(X)$ as the product of all its parent π vectors

  - compute the belief vector of X as the product of its π and λ vectors

- generate messages from X on the update of its evidence vector:
  - to each parent U_i of X:

    $$\alpha \sum \lambda(X) \sum p(x|u_1, \cdots, u_{i-1}, u_{i+1}, \cdots, u_n) \prod \pi_X(u_k) \quad (4.2)$$

    where α is a normalizing constant, the first sum is performed on x, the second one on $u_1, \cdots, u_{i-1}, u_{i+1}, \cdots, u_n$, while the product on $k \neq i$
  - to each child Y_i of X:

    $$\frac{\alpha Bel(X)}{\lambda_{Y_i}(X)} \quad (4.3)$$

    where α is a normalizing constant.

- generate same messages as 4.2 and 4.3:
  - from X to each child Y_i other than Y_j, on the update of λ_{Y_j} vector
  - from X to each parent U_i other than U_j , on the update of $\pi_X(U_j)$ vector.

4.1.7 Scenario Projection

Once the intent of each enemy item, with regard to the friendly one specified by the user, has been estimated, INFUSION can generate the projected scenarios.

With regard to *Time Mode* elaboration, the system projects six different scenarios in which the friendly item is supposed to move in six different directions, spanning 360° around it. With regard to *Position Mode* elaboration, the system foresees the scenario according to the desired position specified by the user for a friendly item.

In particular, for both enemy and friendly items, the system adopts a linear velocity model in order to compute item trajectories in $\triangle T1$ or $\triangle T2$, depending on the elaboration mode specified. The speed, assumed as constant, is the one recorded for the item at time $T0$, when the request of elaboration occurs. The computation of trajectories takes into account the elliptical shape of the earth and the DTED.

The most relevant factors affecting trajectory projection are coalition and intent estimated for each item. In particular:

- if the item belongs to the friendly coalition, it is supposed to keep on moving as at time $T0$, i.e. same direction and speed;
- if the item belongs to the enemy coalition and its estimated intent is:
  - *not-offensive*, it is supposed to keep on moving as at time $T0$;
  - *offensive*, it is supposed to follow the movement of its target.

Note that dependency of enemy trajectory with the intent towards only one friendly item is a simplification: enemy trajectory could also be affected by enemy intent towards other friendly items, by the operation it is conducting or by environment constraints.

4.1.8 Simulation Results

In this section are reported simulation results obtained with regard to different scenarios generated by the SGA named Presagis-STAGE. INFUSION HCI is integrated with the Geographic Information System (GIS) NASA World Wind.

For simulations showed in this section, INFUSION has been installed on a Quad Core Server, with 4 GB RAM, HDD 250 Gb, 2 monitors and O.S. Window 7. Let us consider a scenario, named *Scenario1*, containing in $T0$ two friendly items and ten enemy items, whose relative positions are depicted in Figure 4.6. From the GIS, items belonging to the friendly coalition are marked in blue, those belonging to the enemy coalition are marked in red, if offensive towards *Friend 1*, and in yellow if not-offensive, have been printed.

Results of SA and TA elaborations performed by INFUSION reported that *Friend 1* is engaged by seven enemy items, but not encircled because of the big distance; moreover *Friend 2* is engaged by two enemy items and risks to collide with one of them.

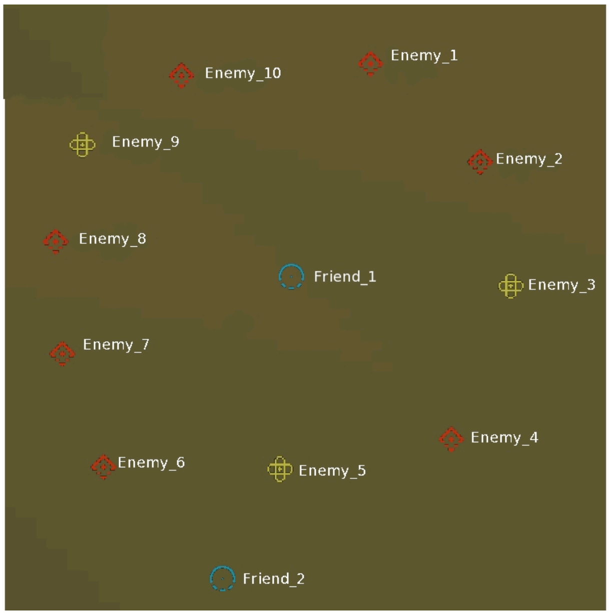


Figure 4.6: *Scenario1*

Starting from the same scenario, INFUSION has been requested to foresee six possible future scenarios, 1 minute after. The results of elaborations are depicted in Figure 4.7.

In each foreseen scenario, the item *Friend 1* moves in six different directions, the enemy items whose behavior has been estimated to be *offensive*, follow their target, i.e. *Friend 1*; those estimated to be *not-offensive* keep on moving as of time $T0$.

The scenario projected in Figure 4.7 (a) results to be the worst one for the item *Friend 1*, in fact, it is subject to the risks of engagement, collision and encirclement from *Enemy 1*, *Enemy 2* and *Enemy 10.* The best scenario indicated by the system is the one in Figure 4.7 (f), where *Friend 1* is subject to the smallest engagement threat. All the other foreseen scenarios report the collision and engagement threat.

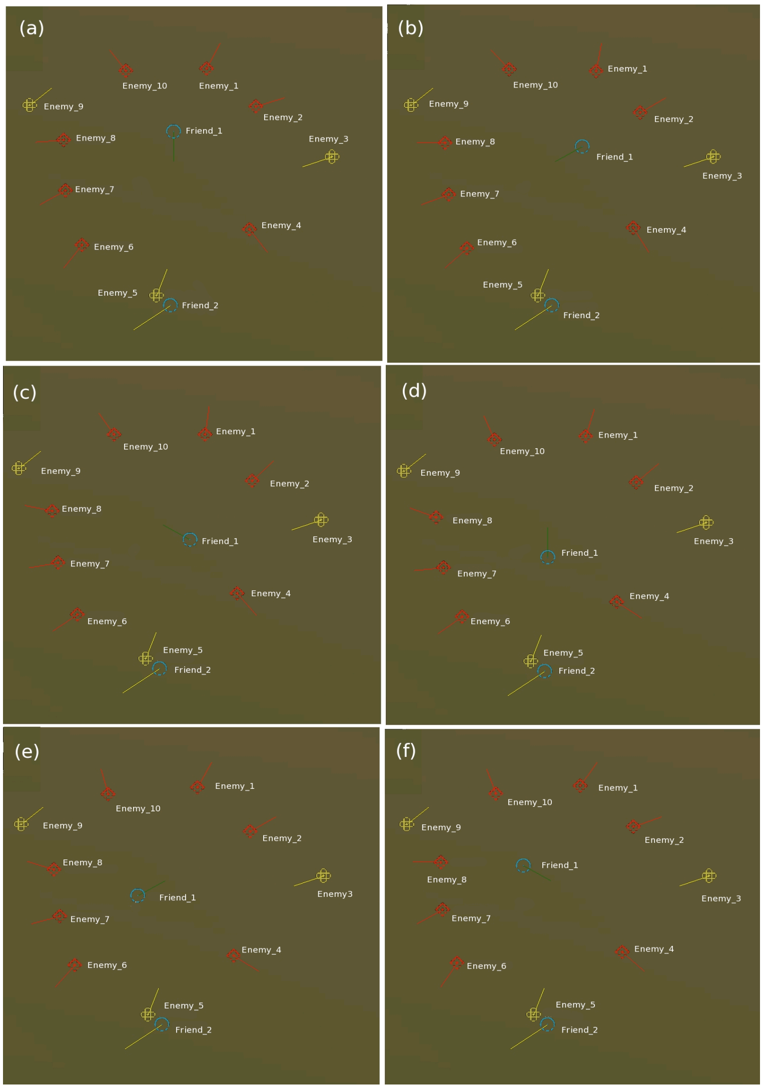


Figure 4.7: *Scenario1* projections

4.2 An Approach for Critical Infrastructure Protection from Cyber Attacks

The industrial control systems and, therefore, also the control center of large infrastructures are able to collect a large number of data and to show them to the operator. The operator is able to understand what is happening and then choose possible actions to be performed on the equipment under control. These systems are able to react to internal damage to the controlled system itself, but are not able to react to external damage if it is too large, such as which can result from natural events.

Critical infrastructures encompass a number of sectors, including many basic necessities of our daily lives, such as food, water, public health, emergency services, energy, transportation, information technology and telecommunications, banking and finance, postal services and shipping. Critical infrastructures are particular industrial systems, controlled by Supervisor Control and Data Acquisition (SCADA). SCADA are systems integrate data from large number of remote locations. The data are concentrated and acted upon by some logic processing at a central station.

Critical infrastructures have another problem: the interconnection between the different infrastructures. Interdependencies means bi-directional dependency between the two infrastructures. Interdependencies can be of different kinds: geographical, physical, cyber, or logic [59]. The geographic interdependencies are due to proximity among facilities equipments. Two infrastructures are physically interdependent if the state of each depends upon the material output of the other. The cyber interdependencies are related to information technology infrastructure, and due to increase use of computer and automation in the SCADA systems. Logic interdependency is related to some mechanism not previously explained, as legal or policy regimes.

All critical infrastructures increasingly rely on computers and networks for their operations. Many of the infrastructures' networks are also connected to the public Internet. While the Internet has been beneficial to both public and private organizations, the critical infrastructures' increasing reliance on networked systems and the Internet has increased the risk of cyber attacks that could harm the nation's infrastructures.

The SCADA systems were first introduced in the 80's and 90's and they are still used. These systems, including those installed until some years ago, did not consider the security aspects. These systems were designed with a monolithic structure, isolated from the outside world and with proprietary standards for communication between control centers and field devices.

Over time, because of the rapid growth of the Internet and telecommunications networks, the SCADA system has changed often, coming to a distributed structure, with standardized and well documented communications, like TCP/IP and Modbus. These SCADA systems are usually also connected to a corporate network. In addition, these SCADA systems exchange data in the clear with no encryption or authentication algorithms; however, encryption can degrade the communications performance.

However, in recent years, there is a growing need to evaluate the SCADA systems also from the point of view of security. This need arises due the great importance of these systems on the welfare of citizens and nations. In 2010, the discovery of Stuxnet [51] demonstrated how computer attack on industrial control systems and SCADA systems are possible. Stuxnet is able to infect Windows computers, and to recognize the industrial control systems, using special root-kit. In 2010 and in 2011, the number of SCADA vulnerability disclosures and exploits has dramatically increased by many orders of magnitude. Terry McCorkle and Billy Rios have found 100 SCADA bugs in 100 days, thanks to free software available on-line. [52]

Impact evaluation of cyber attacks and their consequences are very difficult to define. The problem is highly complex due also to interdependencies existence. In fact, the cascading effects are sometimes not easy to find, especially with the growing complex interdependencies of modern telecommunications.

Currently, the impact assessment of faults must also consider the possibility of evaluating the effects of cyber attacks, which are realistic within the Critical Infrastructure Protection. The introduction of firewalls, intrusion detection systems (IDS) and degree of separation between the business network and the control system network is a good step toward increasing security, but may not be enough.

In fact, the resilience of facilities can be achieve by means of information exchange among States and infrastructure owners. The information can be transmitted using national and international agencies, like Computer Emergency Response Teams (CERTs), or early warning and alerting networks, as European Information Sharing and Alert System (EISAC), American National Cybersecurity and Communications Integration Center (NCCIC) or the Australian Cyber Security Operations Centre (CSOC). [53]

All the agencies, listed above, were created with the goal of intermediating with infrastructure operators in case of possible cyber attacks. Each infrastructure suspected of being under attack, warns its agency or CERT that shares information with other agencies and infrastructures that may be affected in the attack. In addition, they provide mitigation mechanisms and countermeasures.

The architecture, described in next sections, is able to provide an early warning of possible cascading faults and cyber threats. It is also able to define the level of risk to the infrastructure, defined at different levels of abstraction, and using metrics such as Quality of Service (QoS). The architecture can also show possible countermeasures and adapt the software of existing perimetric equipment, such as firewalls, IDSs, and Intrusion Prevention Systems (IPSs).

4.2.1 Real Time Global Awareness

The proposed architecture, see Figure 4.8, is designed to help facility operators understand what is happening and what could happen within their infrastructures. A portion of this structure was conceived and tested within FP7 EU project MICIE [54] (Tool for systeMIc risk analysis and secure mediation of data exchanged across linked CI information infrastructurEs).

Within MICIE, the aim of improving cooperation among infrastructures has led to the establishment of a fully distributed architecture where the potential effects of failure among infrastructures are exchanged in a secure manner. Security is guaranteed by the presence of particular gateways using communication protocols with encoding and encryption. Security is also ensured by the transmission of data, which include the levels of quality of particular services. [55]

The MICIE system is able to define the risk level of each infrastructure and of each service and equipment. Then, the impact of faults are evaluated in a distributed manner. Each facility assesses its consequences after failures, at all levels of abstraction, including the quality of services. Then, these values are exchanged among connected facilities and they are considered as input values for the other facilities, able to understand the values of Quality

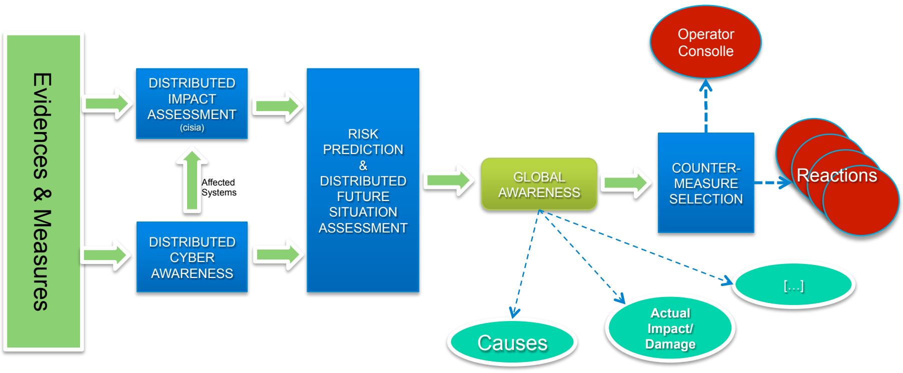

Evidences & Measures
DISTRIBUTED IMPACT ASSESSMENT (cisia)
Affected Systems
DISTRIBUTED CYBER AWARENESS
RISK PREDICTION & DISTRIBUTED FUTURE SITUATION ASSESSMENT
GLOBAL AWARENESS
COUNTER-MEASURE SELECTION
Operator Consolle
Reactions
[...]
Actual Impact/ Damage
Causes

Figure 4.8: Proposed architecture for critical infrastructure protection by cyber threats

of Service (QoS) and to evaluate repercussions.

This system is going to be upgraded to also integrate cyber awareness. [57], [56] For cyber awareness, we refer to the ability of the system to alert the user of faults, generated by intrusions, and cyber attacks in general. In fact, awareness is a process that leads to increased knowledge of the system, the causes that generated failures and the quality of services to customers. So awareness helps to make decisions based on better knowledge of what is happening because of the integration of all available data. In the proposed architecture, cyber awareness can be realized by means of Intrusion Detection and Prevention Systems, or by equipment, able to detect anomalous packet flows among SCADA telecommunication networks. These equipment send their outputs to a SCADA control centre.

The framework is now able to include the cyber threats, catalogued by the *Distributed Cyber Awareness* module depicted in Figure 4.8, in the module able to evaluate impact assessment. The module able to evaluate impact assessment is labeled *Distributed Impact Assessment* and shown in Figure 4.8. Both modules are fed with all values generated by the field equipment and then received by the SCADA control centre.

The *Risk Prediction and Distributed Future Situation Awareness* module, also depicted in Figure 4.8, is devoted to evaluating metrics and generating realistic predictions for all equipment and services described in the scenario. This module has been already presented inside the MICIE project, but the metrics are very simple and can be updated using fuzzy logic for example.

All values and data are distributed among the various interconnected infrastructures. Data can be subdivided in two groups: the one directly obtained by field sensors and cyber sensors, and the other obtained by assessment and prediction.

All these data are aggregated in order to achieve global awareness, as in Figure 4.8. The term *global* denotes the fact that all data to reach the best understanding are available. This module can run algorithms in order to (a) find the causes of faults, (b) backtrack of malware and virus, or (c) determine the actual impact caused by the faults.

In Figure 4.8, there is also a module for countermeasure selection or adaptation. This module is able to show a list of all applicable countermeasures to facility operators, to select contingency strategies at each possible level of abstraction. Countermeasures may include all instruments in the scenario that can be configured by commands, such as a firewall that allows raising the level of danger or communication policies between control centers and Remote Terminal Units (RTUs.)

In the following sections some aspect of this architecture is shown.

4.2.2 Mixed Holistic-Reductionistic Model: an approach for impact assessment and prediction

In this section, we introduce a Mixed Holistic Reductionistic approach (MHR). In such a perspective, the best aspects of holistic and reductionistic approaches are maintained: the interdependencies among elementary components are modelled with the reductionistic method, and the relations at high level are modelled through the holistic vision.

Between these two levels, there is another level called *Service Provider* (SP). These entities are demanded to provide an aggregate resource or service to reductionistic elements, and their values are considered as the Quality of Service.

With a reductionistic perspective, each infrastructure is decomposed into a web of interconnected elementary entities (or blocks); these entities receive and generate resources, and may propagate failures according to proximities of different natures; therefore, their behaviour depends on the interactions (mutual or not) with the other reductionistic elements. Moreover, their capability to correctly operate also depends on the availability and quality of some aggregate resources (or services) provided by SPs.

Service Providers are introduced as functional blocks demanded to provide specific, yet high level, functions to reductionistic elements belonging to the same or different infrastructure. Analogously to reductionistic elements, SPs require and provide (aggregate) resources and may suffer and propagate some failures; this allows modeling of complex and high-level failures (e.g. the effects of cyber attacks) that, instead, are very complex to model with a mere reductionistic perspective. The operativeness of each SP is largely influenced by the operative condition of the infrastructures, and by the policies and management strategies adopted in the specific context by the infrastructure's stakeholders.

Holistic blocks represent the holistic view of the infrastructures, and they interact with other holistic entities exchanging their operativeness. In this case the failure block allows modelling specifically some events like malicious behaviours, that should be very difficult to model at different abstraction levels. Holistic blocks have the duty to influence the operative conditions of SPs on the base of the feedbacks received from reductionistic elements and considering also the overall status of the infrastructure itself. Moreover every holistic node must provide adequate management service to SPs, by means of the definition and execution of adequate control actions (i.e., flow redirections, parameter configuration, event-driven suspension/reactivation/recovery, etc ..) in order to react to adverse events which may cause a degradation or denial of the aggregate resources provided by SPs and generate cascading propagation of faults.

Finally, a holistic node must be aware of the operativeness of its own SPs, in order to obtain a complete knowledge of the status of the infrastructure itself and then update the overall operativeness accordingly. In Figure 4.9 there is an example of MHR application, considering two infrastructures: the telecommunication network and the power grid.

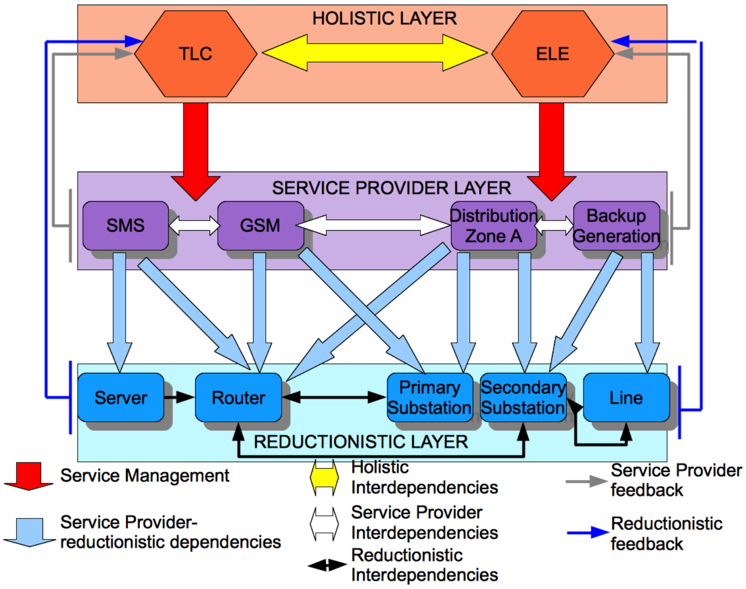


Figure 4.9: An MHR example, with its agents and layers

The overall system of system is decomposed into a set of n entities, and the spreading mechanisms of m resource typologies and k classes of failure are considered.

Such framework considers multiple interconnection matrices, which represent the different typologies of interaction; the result is a multi-graph, which allows performing complex topological and dynamical analyses.

Within this approach, all the elements follow a common general model:

- Elements exist in order to produce, and transport or consume tangible or intangible resources (goods, services, policies, management, etc);
- Elements may suffer faults or failures;
- Different faults may be propagated (or propagate their negative effects) according to proximities of different nature;

- The capability of each element to provide the required resources may depend on its operative condition, which is based on the availability of the resources it requires and on the severity of the failures that affect it.

Moreover, in order to effectively represent the uncertainty of human operators and actors, all the variables describing the dynamics of entities are expressed by Fuzzy numbers (FN). Fuzzy numbers can be seen as the most natural way to introduce model and data uncertainty in a technical vernacular.

Moreover, the interdependency is modeled by means of multiple adjacency matrices, resulting in a multi-graph. Finally, each quantity is modelled by means of Triangular Fuzzy Numbers, allowing encoding vague information and providing an estimation of the certainty of the simulation/prediction.

The model is able to manage faults and also some characterizations as of which the type of fault, such as cyber fault or failures due to earthquakes. So the MHR model can handle different propagation due to different fault causes: the propagation pattern of a malware or a cyber intrusion is completely different from the spreading of a fire.

The numerical value of each entity is considered also as a risk of malfunctioning. The holistic node of a telecommunication network has as fuzzy value, the risk value of the entire working facility. The value represents both the risk evaluation process and the deriving uncertainty of this process.

4.2.3 A case study

In this section, a simple case study is reported in order to support the effectiveness of the proposed approach within the cyber domain. The reference scenario described hereafter will be adopted for simulation results. Let us consider the following three infrastructures:

- A power grid providing electricity to both civil and government customers (i.e. police offices, houses, etc..)

- A telecommunication network, connecting power control rooms and the Remote Terminal Unit (RTUs) of the mentioned power grid, connected to the tele-controlled breakers. The telecommunication network is a typical SCADA network.

- A telecommunication infrastructure connected to the SCADA network for packet forwarding to very distant RTUs, and also for feeding mobile customers, as policemen around the city.

A SCADA system is a system specifically oriented to industrial system control and management. A RTU (Remote Terminal Unit) is an electronic device located in the field and connected with the SCADA by means of a telecommunication network, or in some case by means of serial socket.

Connections among infrastructures are shown in Figure 4.10. The power grid is controlled by the SCADA control system through the telecommunication network. SCADA network is, then, interconnected to telecommunication network, by means of specific gateways realizing the switching interfaces.

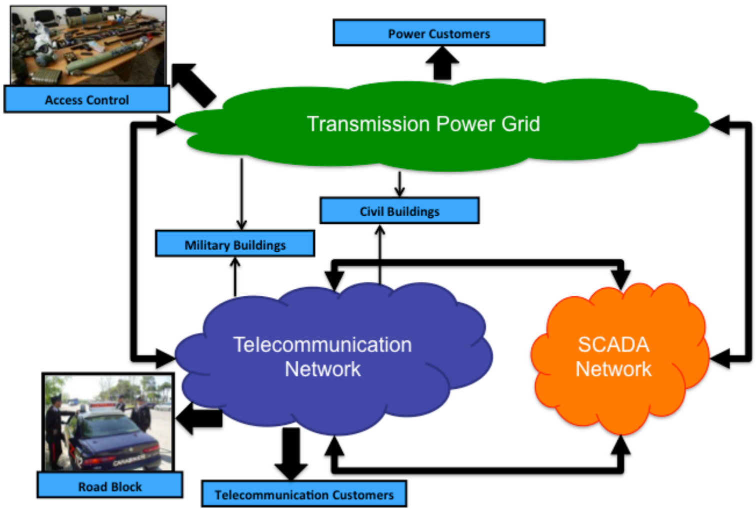


Figure 4.10: Interdependencies among critical infrastructures considered in the case study

In the telecommunication network, information is exchanged among control center and telecontrolled circuit switches. In case of failures on the power grid, the reconfiguration of all networks is necessary; this procedure is realized through command packets addressed to SCADA and to RTUs, in order to open and close the necessary switches. Moreover, all telecommunication equipment fed by the transmission power grid are fed by UPS (Uninterruptable Power Supply) system, not vulnerable to power grid failures.

Breakers are components employed in the electrical circuit interruption in case of suspicious power flow. They are characterized by quick response and little closing and opening times, like milliseconds. Breakers are employed in protection and control strategies, but they can also be employed for operation and maintenance purposes. Switches are devices adopted to physically separate power grid elements belonging to interconnected networks. Their response time to open/close orders are longer than the one of breakers. Consequently, they are regarded as elements to be operative after the intervention of breakers. The combined action of breakers and switches guarantees the complete physical and galvanic isolation of interconnected elements.

The next section reports how the MHR approach can be applied to the scenario described before and depicted in Figure 4.10; where possible end users of power grid and telecommunication network services are indicated. The three infrastructures have some interfaces, in fact some equipment are strictly related one to another, e.g. the RTUs of SCADA network with the telecontrolled breakers, or are the same agents, like for the telecommunication infrastructures.

In 4.10, some possible customers are depicted: some buildings, both for civil and military purposes, are fed by telecommunication network and by power grid distribution.

4.2.4 Simulation Results

In this section, simulation results related to the implementation of the scenario according to the MHR model are presented.

In the implemented model, the Service Layer models the infrastructure capability to feed end users depicted in Figure 4.10, and it is able to reconfigure the infrastructure, triggering specific routines. following figures 4.11 through 4.15 show the evolution of operative level of infrastructure components when a cyber attack occurs. The simulated attack is supposed to cause the denial of service of a telecommunication router, then effects are propagated among infrastructures as described hereafter. Finally, we will assume that a second fault on the power grid distribution network occurs, and we will analyze its effects on the already affected infrastructure.

Notes that the MHR approach manipulates fuzzy number, and especially Triangular Fuzzy Numbers (TFNs). These numbers are a way to express uncertainty. The triangular fuzzy number is defined by four (crisp) numbers: the left, the medium, the right value and the height. These values define an area, instead of a single number and are those plotted in the following figures.

In Figure 4.11, the operative level of the affected telecommunication router is depicted. The decreasing over time of the operation of the router is due to the nature of the fault itself: a cyber attack is assumed to cause the DoS (Denial of Service), if the affected component is a telecommunication node.

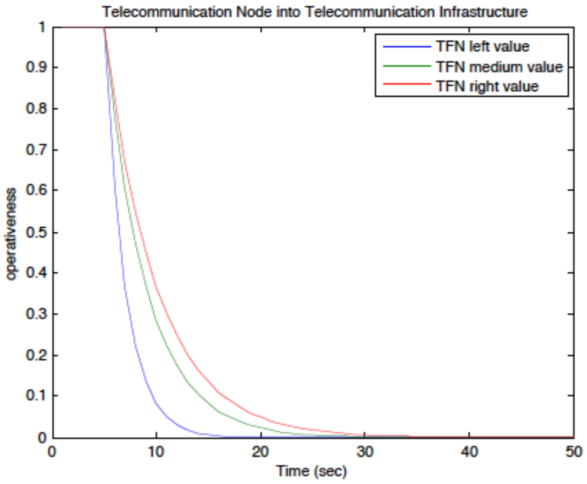


Figure 4.11: Operative level of a telecommunication node

The DoS of the telecommunication router has repercussions on some other agents in the models, belonging not only to the telecommunication network, but also to the other two infrastructures, as the SCADA network. If the affected telecommunication router is an interface between the two infrastructures, the effect of faults is directly transmitted to the interconnected elements in the SCADA network. While the affected telecommunication node is not directly connected to the SCADA network, as assumed in the proposed case study, fault effect will cause possible delays on the packet transmission and eventually packet dropping, as Figure 4.12

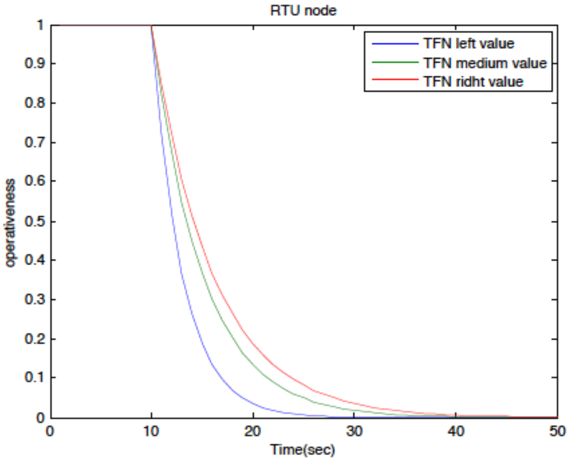


Figure 4.12: Operative level of a SCADA network agent

In Figure 4.13, the telecommunication reconfiguration service operativeness is depicted. As it can be noticed, the trend is strictly connected to the nature of fault. The DoS attack on the telecommunication router causes dropping of several packets. As the reconfiguration service can accomplish its task controlling infrastructure components by packages sent, the DoS of the telecommunication network strongly affect its QoS.

Repercussion of the cyber attack can be registered also in the power grid. In fact, in case of a second fault on the distribution power grid, the ability of the power SCADA to reconfigure the network is very reduces: the power grid reconfiguration requires packet transmission from the SCADA to the tele-controlled circuit breaker.

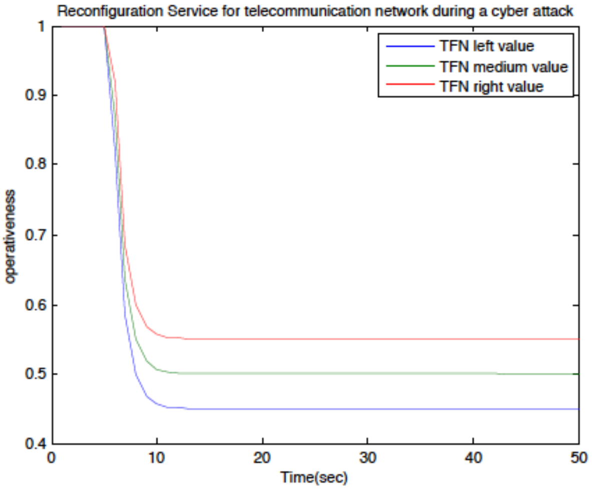


Figure 4.13: Operative level of the telecommunication reconfiguration service

This process aims to identify and isolate the fault area and then reconfigure the network by means of open/close commands. If packet transmission is not granted, the power grid reconfiguration service is strongly affected. Figure 4.14 presents the operative level of the reconfiguration process for power grid. The reconfiguration process QoS decreases rapidly, because some configurations are not allowed, due to troubles in the communication channel.

Finally, the operative level of telecommunication service to an end user (i.e. the police building) is displayed in Figure 4.15. As it can be noticed, the telecommunication delay felt by users is slightly longer than the one in 4.11. From what was presented in this section, it is possible to conclude that a cyber attack on a scenario like the one described in this section, can significantly affects strongly telecommunication nodes and reconfiguration services of all interconnected infrastructures. With this regard, effective countermeasures should be addressed to introduce redundancy in telecommunication channels.

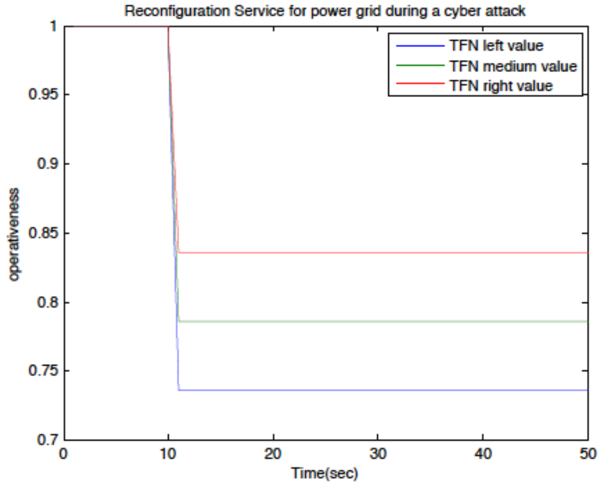


Figure 4.14: The operative level of the reconfiguration service in the power grid infrastructure

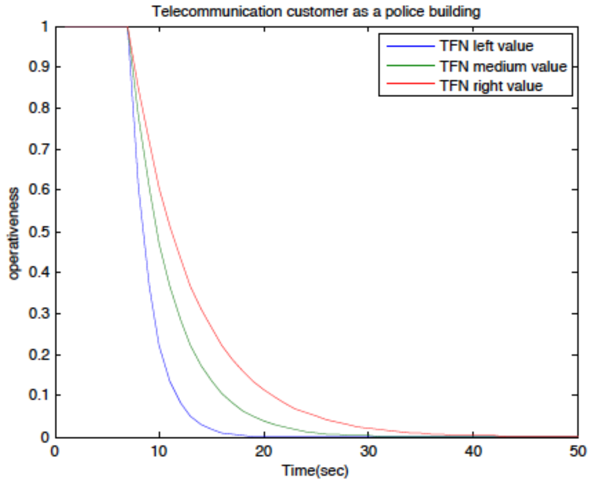


Figure 4.15: The operative level of a telecommunication customer

Conclusions

The topic of this PhD thesis is the Situation Awareness discipline, intended as the perception of the elements in the environment within a volume of time and space, the comprehension of their meaning, and the projection of their state in the future.

Thanks to advances in data acquisition, transfer and archiving technologies, the chance to realize systems able to support human operators in being aware of the observed situations, possible threats and future evolutions has become realistic.

In the development of such kind of systems, several issues have been raised and many scientific approaches have been formalized.

Within this work an overview of most common approaches adopted in Situation Awareness has been discussed. In particular Bayesian networks have been described, as a powerful tool able to perform statistical inference and to determine the probability of a behavior or situation, given the probability of atomic events or the probabilistic sensorial information. Then the MMs have been presented as a compact cyclic representation aimed to represent the behaviors or situations by means of a sequence of state transitions. Finally Evidence Theory has been presented has a data aggregation procedure able to cope with different and contradictory information sources.

Among open issues highlighted within SAW, related for example to inference algorithms, performance evaluation, effective visualization of analysis results, the one related to knowledge management has been addressed. In fact, knowledge modeling in Situation Awareness is regarded as a basic requirement for correct inference about on-going situations and threats.

With this regard, the *agility* of a model has been defined as a feature of those knowledge models able to highlight inconsistencies and contradictions in information acquired and to employ them, together with human advices, to revise themselves.

Despite of learning approaches, the focus has been on identifying metrics for on-line evaluation and on-line correction of the model. In particular, the characteristics of the model employed in Evidence Theory approach have been analyzed. Features of models unable to discriminate situations have been presented. The mass of the empty set has been indicated as an effective metrics to express the fitness of the model to the observed situation. Moreover, a variation to the rule for mass combination has been introduced, in order to allow the system to recognize time-dependent situations, and to change its idea about previous estimation.

Some indications have been provided in order to use the mentioned metrics for model refinement and simulation results related to a simple case study, driven from the Critical Infrastructure domain, have been presented.

Model agility has be identified as a powerful feature in JDL Level 4 Process Refinement, because it can guide and improve the overall data collection process, eventually cueing the user or the system to search for lacking information.

Main features identified for an agile model are the following:

- an agile model does not require to be perfect since its construction: it can be obtained with imperfect knowledge of the whole system, because it is able to learn from its experience;

- agility extends the model lifetime: agile models are able to manage a greater number of scenarios that maybe were not even included when the model was created;

- an agile model is more resilient, more robust, and able to perform better and wider range of real life scenarios.

Beside to the definition of agility, this work has investigated the chance to adopt Data Mining approach within Situation Awareness in order to allow the construction of knowledge models able to recognize effectively situations of interest, that can be specified by the user in real-time. With this regard a system architecture has been proposed. Data mining techniques are adopted to mine hidden relations among data stored in intelligence databases. The output of the Data Mining process is then used to build knowledge models employed for situation and threat assessment. Both databases and Situation and Threat Assessment processes are fed with observations gathered from the field.

The mentioned Data Mining process is triggered by the specification of a particular target of interest by the user. Once this target is given as input, the system Data Warehouse is updated through the computation of variables expressing spatial, causal and temporal correlation between each record and the target itself. Then the unsupervised classification is performed in order to identify clusters of data that are characterized by the same kind of correlation with the specific target. When clusters have been identified, the supervised classification, and in particular the linear discriminant analysis, is applied to estimate models of each cluster as linear functions whose coefficients expresses the dependency of the correlation score with the spatial, causal and temporal variables, introduced in the DW. The output of the Data Mining process (clusters of records and their models) is employed to build Hidden Markov Models for the recognition of situations of interest. The proposed architecture contemplates the refinement of knowledge models each time user requires the analysis of a certain target. Therefore, HMMs related to most critical TOIs are also those more frequently refined.

Finally, considerations have been derived from the application and implementation of SAW methodologies into the military domain and the critical infrastructure protection.

Within military context, the INFUSION system has been developed and presented. INFUSION is a tool able to perform situation and threat assessment in scenarios simulated by Presagis-STAGE, and to project the simulated scenario, according to the Situations and threats recognized at the moment by the system are those of encirclement, engagement and collision. The generation of future possible scenarios takes into account the presence of relieves, the intent estimated for each item, and a constant velocity model for trajectory prediction.

The implementation of the mentioned Situation Awareness framework has lead to the quantification of threat as the product between the probability that a certain situation occurs and its impact.

Major difficulties in INFUSION development have been related to model definition for situations and threats, and to parameter tuning of the Bayesian Net employed for enemy intent estimation. In both cases, we referred to the opinion of experts in the military domain, but further works have to be done to automate the learning process from simulated datasets.

With regard to Critical Infrastructure Protection, Situation Awareness techniques have been applied in order to increase the awareness about causes of malfunctioning, such as natural disasters or malicious events, as cyber attacks. Interdependent infrastructures have been modeled though the Mixed-Holistic Reductionist approach in order to evaluate the effects of the following cyber attacks, occurring to two interconnected infrastructures, that is a telecommunication and power grid infrastructure: denial of service, denial of access, denial of control and manipulation of view.

Our belief is that the framework of Situation Awareness suits the context of protection of Critical Infrastructures, in fact the understanding of malfunctioning behavior causes allows to estimate possible not detected failures, whose identification is crucial to evaluate the vulnerability of infrastructures and the impact of outages. Detected failures, together with not detected but estimated failures, are relevant to project the state of infrastructures, better perform risk assessment and undertake the most effective countermeasures.

4.2.5 Further Research Activities

Topics encompassed in this work touch different aspects of SAW discipline, hence further researches have been indicated for different domains and listed below:

- Analysis related to agility of knowledge models adopted in inference techniques, other than Evidence Theory.
- Implementation and validation of the system whose architecture aims to combine data Mining approach with Situation Awareness techniques.

- INFUSION system improvements in order to allow the following capabilities:
    - contemplate artificial and natural obstacles, meteorological conditions and timing;
    - evaluate new situations and threats in urban scenarios;
    - define new behavioral models, for items and clusters in the scenario, and their projection;
    - profile the visualization of Situation Awareness elaboration accordingly to the role of each INFUSION user, in order to provide him with the most relevant information.

Bibliography

Bibliographic references

[1] USAF (1998). Air Force Pamphlet 14-210, *Intelligence, USAF Intelligence Targeting Guide*, Department of Defense.

[2] OODA Loop. (2001). http://www.d-n-i.net/fcs/ppt/boyds-ooda-loop.ppt.

[3] Rasmussen, J. *Skills, rules and knowledge: Signals, signs and symbolism, and other distinctions in human performance models*, IEEE Transactions on Systems, Man, and Cybernetics, 12:257-266, 1983.

[4] Rasmussen, J. *Information Processing and Human Machine Interaction: An Approach to Cognitive Engineering*, North Holland, NY, 1986.

[5] Das, S., and Grecu, D. (2000). *COGENT: Cognitive agent to amplify human perception and cognition*, Proceedings of the 4th International Conference on Autonomous Agents, Barcelona.

[6] Reason, J. (1990). *Human Error*, Cambridge University Press, Cambridge, UK.

[7] Endsley, M. R. (1988). *Design and evaluation for situation awareness enhancement*, Proceedings of the 32nd Annual Meeting of the Human Factors Society, 97-101.

[8] Das et al., *High Level Data Fusion*, Artech House Publisher, 2008.

[9] T. Damarla, *Hidden Markov Model as a Framework for Situational Awareness*, IEEE 11th International Conference on Information Fusion, 2008, 1–7.

[10] J. Zhao, *Hidden Markov Models with Multiple Observation Processes*, Honours Thesis, Nov, 2007.

[11] J. Chenga, R. Greinera, J. Kellya, D. Bellb and W. Liub, *Learning Bayesian networks from data: An information theory based approach*, Artificial Intelligence, volume 137, 2, pages 43-90, 2002.

[12] E. Blash and S. Plano, *DFIG Level 5 (User Refinement) issues supporting Situational Assessment Reasoning*, 5th International Conference on Information Fusion, 2005.

[13] G.P. Tadda and J.S. Salerno, *Overview of Cyber Situation Awareness*, Cyber Situational Awareness, Advances in Information Security, Volume 46. ISBN 978-1-4419-0139-2, Springer-Verlag US, 2010, p. 15.

[14] Hester, Todd and Quinlan, Michael and Stone, Peter, *A Real-Time Model-Based Reinforcement Learning Architecture for Robot Control*, CoRR volume abs/1105.1749, 2011.

[15] M.R. Endsley, *Toward a Theory of Situation Awareness in Dynamic Systems*, volume 37, The Journal of the Human Factors and Ergonomics Society, Human Factors and Ergonomics Society, 1995, pages 32-64.

[16] E. Dereszynski, J. Hostetler, A. Fern, T. Dietterich, T. Hoang and M. Udarbe, *Learning Probabilistic Behavior Models in Real-time Strategy Games*, Journal of Artificial Intelligence Research, volume 4, (1996), pages 237-285.

[17] A.A. Malikopoulos, P.Y. Papalambros and D.N. Assanis, *A Real-Time Computational Learning Model for Sequential Decision-Making Problems Under Uncertainty*, Journal of Dynamic Systems, Measurement, and Control, American Society of Mechanical Engineers, New York, NY, USA, volume 131, (2009), 041010.1-04101.8.

[18] M. Molineaux, D.W. Aha, and P. Moore, *Learning Continuous Action Models in a Real-Time Strategy Environment*, Proceedings of FLAIRS Conference, 2008, pp.257-262.

[19] S.E. Friedman and K.D. Forbus, *Repairing Incorrect Knowledge with Model Formulation and Metareasoning*, Proceedings of the 22nd International Joint Conference on Artificial Intelligence.

[20] R. Chang, W. Brauer, M. Stetter, *Modeling semantics of inconsistent qualitative knowledge for quantitative Bayesian network inference*, Neural Netw 21: 182 - 192.

[21] V.A.M. Tamma and T.J.M. Bench-Capon, *A Knowledge Model to Support Inconsistency Management when Reasoning with Shared Knowledge*,

[22] L. Portnoy, E. Eskin and S. Stolfo, *Intrusion detection with unlabeled data using clustering*, Proceedings of ACM CSS Workshop on Data Mining Applied to Security (DMSA-2001), pages 76–105, 2001, Philadelphia, PA.

[23] J. Geller, H. Gu, Y. Perl and M. Halper, *Semantic refinement and error correction in large terminological knowledge base*, Data Knowl. Eng., volume 1, pages 1-32, April 2003, Elsevier Science Publishers B. V., Amsterdam, The Netherlands.

[24] S. Verwer and M. De Weerdt, *An Algorithm For Learning Real Time Automata*, BNAIC, BNVKI, 2007, Mahammad Mehdi Dastani and Edwin de Jong, pages 411-412, 1568-7805.

[25] A.P. Dempster, *The Annals of Mathematical Statistics*, Upper and lower probabilities induced by a multivalued mapping, volume 38, 1967.

[26] D. Smith, S. Singh, *Approaches to Multisensor Data Fusion in Target Tracking: A Survey*, IEEE Transactions on Knowledge and Data Engineering, volume 18, no. 12, Dec 2006.

[27] H.Leung, and Jiangfeng Wu, *Bayesian and Dempster-Shafer target identification for radar surveillance*, IEEE Transactions on Aerospace and Electronic Systems, April 2000, volume 36, no.2, pages 432-447.

[28] P.Smets, R.Kennes, *The transferable belief model*, Artificial intelligence, Elsevier, volume 66, no.2, pages 191-234, 1994.

[29] L.A.Zadeh, *A Simple View of the Dempster-Shafer Theory of Evidence and its Implication for the Rule of Combination*, AI Magazine, volume 7, pages 85-90, 1986.

[30] Smets, P., *The combination of evidence in the transferable belief model*, IEEE Transactions on Pattern Analysis and Machine Intelligence, volume 12, no.5, pages 447-458, 1990.

[31] A. N. Steinberg, C. Bowman, and F. White, *Revisions to the JDL Data Fusion Model*, NATO/IRIS Conf. October, 1998

[32] Hall, David L. and Llinas, J. *Handbook of multisensor data fusion*, CRC Press, 2001

[33] James, Z., *Hidden Markov Model with Multiple Observation Processes*, Honour Thesis, 2007

[34] McConky, K., Nagi, R., Sudit, M., Hughes, W., *Improving Event Coreference By Context Extraction and Dynamic Feature Weighting*, Proc. IEEE Multidisciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support, 38-43, 2012

[35] Fausett, L., *Fundamentals of Neural Networks*, New York, 1994

[36] Nisbet, R., Elder, J., Miner, G., *Handbook of Statistical Analysis and Data Mining Applications*, Academic Press, Elsevier, 2009

[37] Markov, A.A. *Rasprostranenie zakona bol'shih chisel na velichiny, zavisyaschie drug ot druga*, Izvestiya Fiziko-matematicheskogo obschestva pri Kazanskom universitete 2-ya seriya, 15, 135-156, 1906

[38] Jarrow, Lando, and Turnbull] Jarrow, R.A., D. Lando, and S.M. Turnbull. *A Markov model for the term structure of credit risk spreads.*, Review of Financial Studies 10(2), 481-523, 1997

[39] Miller, G. *Finite markov processes in psychology*, Psychometrica 17(2), 149-167, 1952

[40] Stark, R.F., Farry, M., Pfautz, J., *Mixed-Initiative Data Mining with Bayesian Networs*, Proc. IEEE Multidisciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support, 107-110, 2012

[41] Salerno, J., Hinman, M., Boulware, D., *Building a Framework for Situation Awareness*, Proc. IEEE Information Fusion, 107-110, 2004

[42] Riveiro, M., Falkman, G., Ziemke, T., *Improving maritime anomaly detection and situation through interactive visualization*, Proc. IEEE Information Fusion, 46-54, 2008

[43] Krishnaswamy, S., Loke, S.W., Rakotonirainy, A., Horovitz, O., and Gaber, M.M., *Towards Situation Awareness and Ubiquitous Data Mining for Road Safety: Rationale and Architecture for a Compelling Application*, Proc. of Conference on Intelligent Vehicles and Road Infrastructure, 16-17, 2005

[44] J. Fox, *Expert Systems and Theories of Knowledge*, Kluwer Academic Publisher, London.

[45] J.C. Bezdek and S.K. Pal, *Fuzzy Models for Pattern recognition*, cap 1 IEEE Press, NY, 1992.

[46] Rein et al., *ALARM for Early Warning: A Lightweight Analysis for Recognition of Menace*, 13th International Conference on Information Fusion, Edinburgh, UK, 2010.

[47] X.T. Nguyen, *Threat Assessment in Tactical Airborne Environments*, Proceedings of the Fifth International Conference on Information Fusion, Annapolis, USA, 2002.

[48] Carvalho et al., *PROGNOS: Predictive Situational Awareness with Probabilistic Ontologies*, 13th International Conference on Information Fusion, Edinburgh, UK, 2010.

[49] *1516 IEEE Standard for Modeling and Simulation High Level Architecture (HLA) Framework and Rules*, 2010.

[50] NATO ATP-35(B), *Land Forces Tactical Doctrine*, Dec 1, 1995.

[51] Falliere, N., O' Murchu, L., Chien, E. (2011) 'W32. Stuxnet Dossier', *White Paper, Symantec Corp., Security Response*, Version 1.4

[52] Rios, B., and McCorkle, T., '100 Bugs in 100 days: an analysis of ICS (SCADA) Software', *DerbyCon 2011*, Session

[53] European Commission (2011) 'Achievements and next steps: towards global cyber-security', *Communication from the Commission to the European Parliament, the Council, the European economic and social Committee and the Committee of the Regions on Critical Information Infrastructure Protection*

[54] FP7 MICIE project (2010), http://www.micie.eu

[55] Foglietta, C., Oliva, G. and Panzieri, S., (2011) 'Online Distributed Evaluation of Interdependent Critical Infrastructures', *Nonlinear Estimation and Applications to Industrial Systems Control* , Nova Pubblications, To Appear.

[56] Rieger, C. G., Gertman, D. I., McQueen, M. A., (2009) 'Resilient control systems: Next generation design research', *Conference on Human System Interactions*, pp.632–636

[57] Barford, P., Dacier, M., Dietterich, T. G., Fredrikson, M., Giffin, J., Jajodia, S., Jha, S., Li, J., Liu, P., Ning, P., Ou, X., Song, D., Strater, L., Swarup, V., Tadda, G., Wang, C. and Yen, J. (2010), 'Cyber SA: Situational Awareness for Cyber Defense', *Cyber Situational Awareness*, Vol. 46, pp.3–13.

[58] Shafer, G. (1976), 'A mathematical Theory of Evidence', *A Mathematical Theory of Evidence*

[59] S. M. Rinaldi, *Modeling and simulating critical infrastructures and their interdependencies*, in Proceedings of the 37th annual Hawaii international conference on System Science, 2004, p. 8.

References of Published Works

[60] Digioia, G., Arisumi, H., Yokoi, K., *Trajectory Planner for a Humanoid Robot passing through a door*, Proc. IEEE 14th International Conference on Humanoid Robots, Paris, 2009

[61] Digioia, G., Foglietta, C., Oliva, G., Panzieri, S., *Countermeasures Selection via Evidence Theory*, 6th International Conference on Critical Information Infrastructures Security, CRITIS2011

[62] Digioia, G., Panzieri, S., *INFUSION: a System for Situation and Threat Assessment in Current and Foreseen Scenarios*, CogSIMA2012 IEEE International Multi-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support, New Orleans, USA

[63] Digioia, G., Panzieri, S., *Homeland situation awareness through mining and fusing heterogeneous information from intelligence databases and field sensors*, SPIE Defense, Security and Sensing 2012, Baltimore, USA

[64] Digioia, G., Foglietta, C., Panzieri, S., *An Agile Model for Situation Assessment: how to make Evidence Theory able to change idea about classifications*, IEEE International Conference on Information Fusion, Singapore, 2012

[65] Digioia, G., Foglietta, C., Panzieri, S., Falleni, A., *Mixed-Holistic Reductionist Approach for Impact Assessment of Cyber Attacks*, European Intelligence and Security Informatics Conference, EISIC2012, Sweden, August 2012

[66] Digioia, G., Panzieri, S., *Critical Infrastructure Protection: Threats Mining and Assessment*, International Defence and Homeland Security Simulation Workshop, DHSS2012, Austria, September 2012

[67] Digioia, G., Foglietta, C., Oliva, G., Panzieri, S., Setola, R., *Moving from Measuring to Undersanding: Situation Awareness in Homeland Security*, Effective Surveillance for Homeland Security: Balancing Technology and Social Issues, CRC Press, Taylor and Francis eds., 2013

[68] Digioia, G., Foglietta, C., Oliva, G., Panzieri, S., *Aware on-line interdependency modeling via evidence theory*, International Journal of Critical Infrastructures, To Appear

[69] Digioia, G., Panzieri, S., *Knowledge Uncertainty Management in Agile Models: How do Situation Awareness techniques deal with it?*, SPIE Defense, Security and Sensing 2013, Baltimore, USA