



Universitetet
i Stavanger

FACULTY OF SCIENCE AND TECHNOLOGY

MASTER'S THESIS

| | |
|---|---|
| Study program/specialization: Master in Computer Sciences | Spring semester, 2009 Open / Confidential |
| Author: Rui Máximo Esteves | (signature author) |
| Instructor(s): Prof. Dr. Chunming Rong; Eng ^o Tomasz Włodarczyk Supervisor(s): Eng. Einar Landre (Statoil Hydro) | |
| Title of Master's Thesis: Data fusion algorithms for assessing sensors' accuracy in an oil production well - A Bayesian approach. Norwegian title: | |
| ECTS: 30 | |
| Subject headings: Data Fusion; Bayesian Network; Sensor Accuracy; Fault Diagnosis; Oil Production Well | Pages: + attachments/other: Stavanger, Date/year |

Acknowledges

The author would like to express his special gratitude to:

Prof. Dr. Chunming Rong from University of Stavanger;

Eng^o Einar Landre from Statoil Hydro;

Eng^o Tomasz Wlodarczyk from University of Stavanger;

Eng^o Mohammad Rajaieyamchee from University of Stavanger;

Eng^o Terje Kastad from University of Stavanger.

Abstract

Oil industry faces an underutilization problem of the captured data during the extracting process. This issue is a consequence of the lack of information regarding sensors' accuracy. One effect can be a serious obstacle in the development of computer assisted decision systems.

In a production well, it can be experienced the inexistence of sensor redundancy and enough information to assess credible probabilities. In this situation, we have to strongly depend of the experts' ability to provide alternatives based on their understanding. These skills can be a critical limitation and turns particularly difficult the establishment of a prediction model.

With this work we propose a Bayesian Network approach as a promissory data fusion technique for surveillance of sensors accuracy. We proved the usefulness of this method when it seems there isn't enough feasible data to construct a model. In presence of certain data constrains we suggest an inversion of the causal relationship. This approach can be a possible solution to help the expert in accessing conditional probabilities.

Index of Contents

| | |
|--|-----|
| Acknowledges..... | i |
| Abstract..... | ii |
| Index of Contents..... | iii |
| Index of figures..... | v |
| Index of tables..... | vi |
| Chapter I - INTRODUCTION..... | 1 |
| Thesis Overview..... | 2 |
| Aim of the thesis..... | 4 |
| Chapter II - THEORY..... | 5 |
| Data Fusion taxonomy..... | 6 |
| Introduction..... | 6 |
| JDL Based Taxonomy..... | 8 |
| Raw data level..... | 9 |
| Methods..... | 9 |
| Feature data level..... | 11 |
| Methods..... | 11 |
| Decision level..... | 15 |
| Methods..... | 15 |
| Hard and Soft Decision Taxonomy..... | 23 |
| Why Bayesian?..... | 25 |
| Bayesian theory..... | 27 |
| Bayesian Network..... | 29 |
| Applications..... | 36 |
| Limitations..... | 38 |
| Chapter III - THE MODEL..... | 40 |
| Hypothesis..... | 41 |
| Methodology..... | 42 |
| Assumptions..... | 43 |
| Data considerations..... | 44 |
| Experiments..... | 46 |
| Approach A..... | 47 |
| Approach B..... | 50 |
| Test and results..... | 54 |
| Chapter IV - CONCLUSIONS..... | 56 |
| Conclusions..... | 57 |
| Further developments..... | 58 |
| References..... | 59 |
| ANNEXES..... | 63 |
| ANNEX A- Exemple of Kalman's filter applied to robotic football..... | 64 |
| ANNEX B- Example applied to pattern recognition..... | 66 |
| ANNEX C- Simple example of Dempster Shafer..... | 69 |
| ANNEX D- Example of D. S. applied to aircraft detection..... | 71 |

| | |
|--|----|
| ANNEX E- Example of Neural Networks applied to pattern recognition..... | 73 |
| ANNEX F- Example of Fuzzy Logic applied to a temperature control device..... | 77 |
| ANNEX G- Genetic Algorithms Uses..... | 84 |
| ANNEX H- GA example applied to the Traveling Salesman problem..... | 88 |
| ANNEX I- Forward chaining expert system example..... | 90 |
| ANNEX J- Backward chaining expert system example..... | 92 |
| ANNEX L- Expert Systems uses..... | 94 |
| ANNEX M- Conditional Probabilities Tables..... | 97 |

Index of figures

| | |
|--|----|
| Figure 1- A simple feed Forward Neural Network..... | 13 |
| Figure 2- A simple neuron | 13 |
| Figure 3- The GA reproductive cycle | 18 |
| Figure 4- The architecture of Hearsay III- a speech understanding system | 22 |
| Figure 5- BN example to elucidate different types of queries. | 30 |
| Figure 6- BN example of an inverse graph approach. | 31 |
| Figure 7. Example of serial connection. | 32 |
| Figure 8. Example of d-separation- Z nodes are ascendants of X and Y..... | 32 |
| Figure 9. Example of d-separation..... | 33 |
| Figure 10- The tree problem | 34 |
| Figure 11- Bayesian Network according to approach A..... | 47 |
| Figure 12- Example of one simulation..... | 48 |
| Figure 13- Example of an incoherent simulation..... | 49 |
| Figure 14- Bayesian Network according to approach B..... | 50 |
| Figure 15- Example of BN under the same conditions as approach A..... | 51 |
| Figure 16- Experiment to reduce the CPT's tables complexity. | 52 |
| Figure 17- The final model including the sensor age condition | 53 |
| Figure 18. Kalman filter application..... | 64 |
| Figure 19- Decision Boundaries | 68 |
| Figure 20- The Neural Network..... | 73 |
| Figure 21 A simple block diagram of the control system..... | 77 |
| Figure 22- Typical control system response | 78 |
| Figure 23- The rule structure & rule matrix..... | 80 |
| Figure 24- The features of a membership function..... | 81 |
| Figure 25- Example errors | 82 |
| Figure 26- Early stage solution example | 89 |
| Figure 27- Optimal solution..... | 89 |
| Figure 28- Overview of the GA Performance | 89 |

Index of tables

| | |
|--|----|
| Table 1- JDL Based Taxonomy | 8 |
| Table 2 Advantages and disadvantages of Genetic Algorithms | 19 |
| Table 3- Hard and Soft Decision Taxonomy | 23 |
| Table 4- P(Sick) | 34 |
| Table 5- P(Dry) | 34 |
| Table 6- P(Loses Sick, Dry)..... | 35 |
| Table 7- Analysis of data | 45 |
| Table 8- Classification into conditions' states | 54 |
| Table 9- Results presented by the model | 55 |
| Table 10- Mass assignments for the various aircraft | 71 |

Chapter I - INTRODUCTION

The chapter pretends to give a short overview of this work. It starts with the description of the background and the importance of the thesis. A general outline of the work is given. The chapter finishes with the aim's definition.

Thesis Overview

In oil & gas extraction process each reservoir can be divided into homogeneous zones.

From an IT perspective an oil & gas zone can be seen as a closed uniform environment that contains some mixture of hydrocarbons under the same pressure and temperature conditions. Each well has a set of sensors to measure environmental conditions such as temperature and pressure. These conditions are distinct in the head of the well and in the reservoir (hole). A choke placed between these two places control this difference.

Statoil Hydro stated that the pressure gauges become to loose performance with time. As wells lifetime goes on, the measures became more uncertainty. The estimated lifecycle of a well can be more than 10 years. On the other hand, working at high temperatures can reduce the sensor lifetime to 2-3 years.

Until this point in time, Statoil does not have information about the accuracy of the measurements provided by the sensors. They suspect that one or more may be inaccurate but they cannot identify which one. Consequently, the usefulness of this data has been very limited. Statoil wants to have more information about the accuracy of the sensors' measurements in order to increase their reliability.

In a production well, the quality of sensors' measurements is an issue which the relevant attention has not been given. According to domain' specialists, studies in this field could benefit the oil sector by providing a better control of the extraction process. This understanding should be one basic stone in the developing of decision support systems. The usefulness of complex systems can be questioned when there is no information about the data accuracy they rely on.

Since there is no sensor redundancy, we had to solve the problem using information from the different sensors' types and data fusion techniques. We faced the challenge of inexistent consistent data to access probabilities and to determine correlations between variables.

In the begging of this work we made a revision about data fusion methods. The aim was to find one that suits our special needs. The starting point was classifying them according

to data fusion levels. We stated that our problem was in the feature level and Bayesian Networks could be a promise method.

However, the examples found in the literature generally assume existence of data to assess probabilities or the ability of an expert to easily express them as believes. As this was not the case, we established then the following hypothesis:

H: In absence of data we may use Bayesian Network for sensor accuracy surveillance.

Latter we experimented different ways to construct the Bayesian Network. The aim was to test if the hypothesis was true or not.

We tested without success the conventional approach to design the Bayesian Network structure. However, we found more plausible to construct the model by reversing the causal direction of the relationships. In this way, one can easier express the knowledge of the expert when in presence of certain data constraints.

To develop this work we had several meetings with experts in the domain.

Aim of the thesis

The present model attempts to solve the following oil extraction problem.

In each production well we have three different sensors (S)¹ that can be inaccurate:

- bht (S1) – borehole temperature;
- bhp (S2) – borehole pressure;
- whp (S3) – well head pressure.

The model's aim is to access probabilities for the sensors' accuracy on a production well.

The challenge is doing it with no trustable data to construct the model.

We want to investigate if this can be possible using data fusion techniques and expert's knowledge.

¹ We didn't find a relation between well head temperature and the other variables, as so, the model can not predict the accuracy.

Chapter II - THEORY

We start our theoretical revision by a classification of several data fusion methods into taxonomy. For those who are not familiar with them, we provided some examples as annexes.

There is an explanation of the reasons why Bayesian was the chosen method. We then presented more carefully its subjacent theory. A review of similar studies is shown. The chapter ends with an overview of the Bayesian limitations.

Data Fusion taxonomy

Introduction

There are some distinct data fusion's taxonomies. One of the most well known was developed by the Joint Director of Laboratories (JDL) from the U.S. Department of Defense. The JDL model was developed for military proposes and consist of four levels of data fusion:

1. identification and description of the objects;
2. interactive process to fuse spatial and temporal entities relationships;
3. combination of the activity and capacity of the enemy forces to infer their force;
4. related with all other levels and is responsible for regulation of the fusion process.

This model has been used also for other fields as image processing. However, given its specific nature is difficult to use in other domains. More generic model has been proposed by other authors with its base on JDL.

[1] presented the Data Fusion Architecture (DFA) in which the division of levels is taking in consideration the difference between data and variable. According to these authors, data can be defined as a measurement of the environment that is generated by a sensor or other type of source and variable is determined by an analysis of the data (feature extraction). A single type of data can provide one or more variables².

² From an image (raw data) an application can determine whether the image contains a person (variable = person-present?), an animal (variable = animal-present?) or an object (variable = object-present?).

Example extracted from I. Carvalho, H.S., et al. *A general data fusion architecture*,. in *Proceedings of the Sixth International Conference of Information Fusion*. 2003.

DFA presents three levels of data fusion:

1. data oriented;
2. task oriented (variable);
3. mixture of data and variable fusion.

The levels differentiate whether the fusion process is made before any data analysis (at the data level), after the data has been analyzed (at the variable level), or is done on a combination of raw data and variables (at the mixture level).

Others authors proposed variations of JDL model. In the next pages, a taxonomy based on [2] [3] will be presented.

JDL Based Taxonomy

In table 1 we present 3 level taxonomy based on JDL model. However this methods' classification should not be seen as rigid. Depending on the application, some of them can be used at several levels.

| JDL Based Taxonomy | |
|---------------------------|-----------------------|
| Level | Methods |
| Raw Data | Kalman Filtering |
| | Figure of Merit |
| | Gating |
| Feature data | Bayesian Theory |
| | Dempster-Shafer |
| | Neural Networks |
| | Clustering Algorithms |
| | Template Methods |
| Decision | Fuzzy Logic |
| | Genetics Algorithms |
| | Expert Systems |
| | Blackboard Systems |

Table 1- JDL Based Taxonomy

Raw data level

In this level the data fusion is processed directly from the sensor data. According to [3], when multisensor data is commensurate (i.e. data from the same nature which is measuring the same physical phenomena) then the raw sensor can be directly combined. The data association can be done by correlation of one set of sensor observations with another set of observations.

Methods

Kalman Filtering

The Kalman Filter can be defined as: “a set of mathematical equations that provides an efficient computational (recursive) means to estimate the state of a process, in a way that minimizes the mean of the squared error” [4]. According to these authors “The filter can be very powerful in several aspects it supports estimations of past, present, and even future states, and it can do so even when the precise nature of the modeled system is unknown”. This feature can be used in target positioning by removing the noise from sensor signals in order to better determine the present and future positions [2]. It uses a recursive solution in that each updated estimate of the state is computed from the previous estimate and the new input data. This lead to an efficient computing solution as only the previous estimate requires storage.

Kalman filters are based on linear dynamical systems discretised in time. It is assumed that the system and the measures are affected by White Gaussian noise. This means the noise is not correlated in time, and thus we can assume that at each discrete time, the noise affecting the system and measures are independent of past or future values [5]. However, we can extend the use of Kalman filtering to non linear systems through a

linearization procedure. The resulting filter is referred to as the extended Kalman filter (EKF) [6].

This filter is used for vision tracking on robotics, real time traffic-control algorithm, and autonomous driving systems.

The steps to use the Kalman filter for vision tracking are:

1. Initialization ($k=0$). In this step it is looked for the object in the whole image due we do not know previously the object position. We obtain this way x_0 . Also we can considerer initially a big error tolerance.
2. Prediction ($k>0$). In this stage using the Kalman filter we predict the relative position of the object, such position is considered as search center to find the object.
3. Correction ($k>0$). In this part we locate the object (which is in the neighborhood point predicted in the previous stage) and we use its real position (measurement) to carry out the state correction using the Kalman filter finding this way.

The steps 2 and 3 are carried out while the object tracking runs. [6]

In annex A is presented an example of a Kalman filter's application to robotic football.

Other methods

[2] made reference to Figure of Merit and Gating as used with the aim to decide which observations should be considered and which are outliers. The basic principle relies in considering only the observations that are below a defined threshold.

Feature data level

When multisensor data is not commensurate then can be fused on a feature/ state vector level. The aim of this level is the extraction of the representative features from the raw data³. On this level, one should extract the features from the various sensor observations and combine them into a single feature vector [3]. This feature vector should be a synthesis of more meaningful information for guiding human decision-making.

Methods

Bayesian theory

According to [2] Bayesian theory is one of the most common techniques employed in level two of data fusion. These authors encourage the use of Bayesian: "The use of multiple sensors in data fusion projects can produce conflicting data which, in turn, can cause decision problems. Application of the Bayesian theorem in such cases has proven successful in overcoming this challenge. It models the unknown system state by using probabilistic functions to determine an appropriate set of actions". Since a certain level of uncertainty is generally associated with sensor's data, it can be improved by quantifying the uncertainty behind each sensor decision and then comparing with some predetermined decision threshold level.

In Annex B is presented an example of Bayesian theory applied to pattern recognition.

³ Kessler and White explain this concept comparing to the cartoonist's use of key facial characteristics to represent the human face. 3. Kessler, O. and F. White, *Data Fusion Perspectives and Its Role in Information Processing*, in *Handbook of Multisensor Data Fusion - Theory and Practice*, M.E. Liggins, D.L. Hall, and J. Llinas, Editors. 2009, CRC Press: Boca Raton.

Dempster-Shafer

Dempster-Shafer (or theory of belief functions) can be considered as a generalization of the Bayesian theory of subjective probability. However, in opposition to the Bayesian theory DS does not requires probabilities for each question of interest, belief functions allow us to base degrees of belief for one question on probabilities for a related question [7].

“Dempster-Shafer allows alternative scenarios for the system, such as treating equally the sets of alternatives that have a nonzero intersection: for example, we can combine all of the alternatives to make a new state corresponding to “unknown”. But the weightings, which in Bayes’ classical probability theory are probabilities, are less well understood in Dempster-Shafer theory. Dempster-Shafer’s analogous quantities are called masses, underlining the fact that they are only more or less to be understood as probabilities” [8].

Object recognition is one of the uses of this method, and has been applied to detection of ship wakes from synthetic aperture radar images [9], robotic, automated guided vehicles [2], and other uses as: color image segmentation [10]; representing the uncertainty inherent in the characterization of containerized radiological waste [11].

To better understand DST’s concept, it is presented a simple example in ANNEX C and in ANNEX D an example applied to aircraft detection:

Neural Networks

The simplest NN is known as a Perceptron which is a system with an input and an output layer.

A Feed Forward Neural Network is a system with an input, an output and at least one hidden intermediate layer which is formed by simple computational units interlinked called neurons.

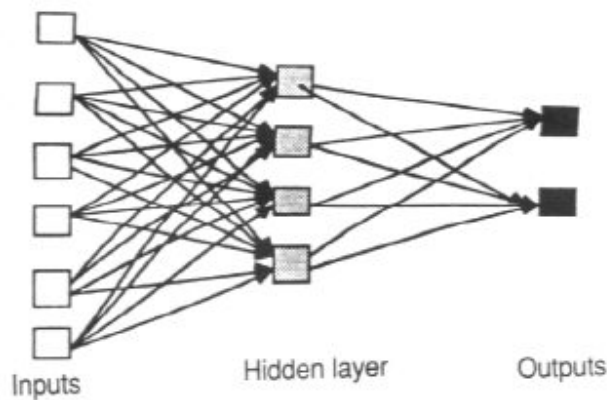


Figure 1- A simple feed Forward Neural Network

After a training process the neutrons establish synapses (weights) between them and the network should have the ability to respond to newer situations [12].

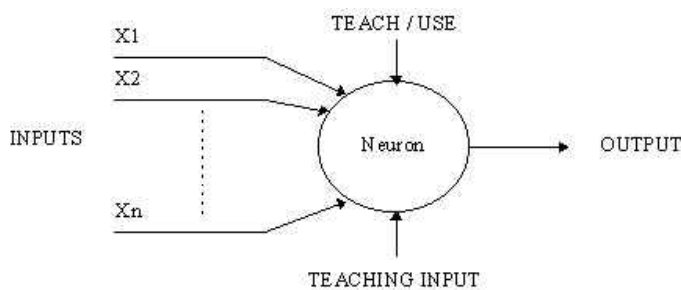


Figure 2- A simple neuron

This network has some advantages over either Bayesian or DSER methods. The most relevant is the ability to process data fusion without the need of a priori information on a parallel way [2]. Neural is been widely applied on different nature forecasts: weather; traffic; internet traffic, stock market among others. However, there are innumeros different applications with this method: Traveling Saleman's Problem (only to a certain degree of approximation); Medicine; Electronic Nose; Security; Loan Applications and Character Recognition Image Compression [13].

In annex E there is an example illustrating the application to pattern recognition.

Other methods

Other techniques for feature data fusion less used are clustering algorithms and template methods [14].

Decision level

In this situation the fusion occurs at the decision level. According to [14], it can also be called as postdecision or postdetection fusion. This level can be achieved applying Boolean operators or using a heuristic score over combinations of decisions from independent sensors detection or classification paths.

Methods

Fuzzy Logic

Fuzzy Logic is a method appropriate to model situations where the boundaries are not clearly identified. The fuzziness can be present in abstract and concrete situations and this theory allows specifying their relevant attributes and relationships [15]. Related to the drifting of the perfect calibration of sensors, [16] referred the usefulness of Fuzzy Logic “in capturing the desired behavior of the classification algorithm for diverse and nonlinear sensor responses is that we can blend information according to our human expert knowledge”. According to this author the inputs to the fuzzy logic could be outputs from other algorithms, such as neural networks, or other inference logic networks.

[17] described the following features about FL that makes it a particularly good choice for many control problems:

- It is inherently robust since it does not require precise, noise-free inputs and can be programmed to fail safely if a feedback sensor quits or is destroyed.
- The output control is a smooth control function despite a wide range of input variations.
- Since the FL controller processes user-defined rules governing the target control system, it can be modified and tweaked easily to improve or drastically alter

system performance. New sensors can easily be incorporated into the system simply by generating appropriate governing rules.

- FL is not limited to a few feedback inputs and one or two control outputs, nor is it necessary to measure or compute rate-of-change parameters in order for it to be implemented. Any sensor data that provides some indication of a system's actions and reactions is sufficient. This allows the sensors to be inexpensive and imprecise thus keeping the overall system cost and complexity low.
- Because of the rule-based operation, any reasonable number of inputs can be processed (1-8 or more) and numerous outputs (1-4 or more) generated. However it would be better to break the control system into smaller chunks and use several smaller FL controllers distributed on the system, each with more limited responsibilities.
- FL can control nonlinear systems that would be difficult or impossible to model mathematically. This opens doors for control systems that would normally be unfeasible for automation.

The Fuzzy Logic can be used on several fields as:

- on selection of the most suitable material for a particular application; hydrodynamic lubrication; elastohydrodynamic lubrication, fatigue and creep; cumulative fatigue damage analysis; reliability assessment; process control; total risk and reliability with human factors; system condition auditing; reframing standards using fuzzy sets for improved quality control; correlation of statistical and automatic process control; road transport logistics [18];
- on measuring automobile speeds and congestion levels, operating automatic trains using predictive logic, and selecting paths in autonomous vehicle navigation systems [2].

[17] suggests the following steps to design an FL system:

1. Definition of the control objectives and criteria: What am I trying to control? What do I have to do to control the system? What kind of response do I need? What are the possible (probable) system failure modes?
2. Determination of the input and output relationships. One should choose a minimum number of variables for input to the FL engine (typically error and rate-of-change-of-error).
3. Break the control problem down into FL rules. The problem should be split into a series of IF X AND Y THEN Z rules that define the desired system output response for given system input conditions.
4. Creation of FL membership functions. The memberships defines the meaning (values) of Input/Output terms used in the rules.
5. Development of necessary pre- and post-processing FL routines if implementing in S/W, otherwise program the rules into the FL H/W engine.
6. System test: evaluate the results, tune the rules and membership functions, and retest until satisfactory results are obtained.

In annex F there is an example applied to a temperature control device.

Genetics Algorithms

Genetics Algorithms are another method used at decision level. It consists on stochastic optimizations which simulates the process of natural evolution. These algorithms are suitable for very complex systems, including multiple objectives optimization.

GA can be viewed as a family of computational models inspired by Darwin's evolution theory. Potential solutions are encoded on a simple chromosome-like data structure and follow a reproductive cycle represented on next figure.

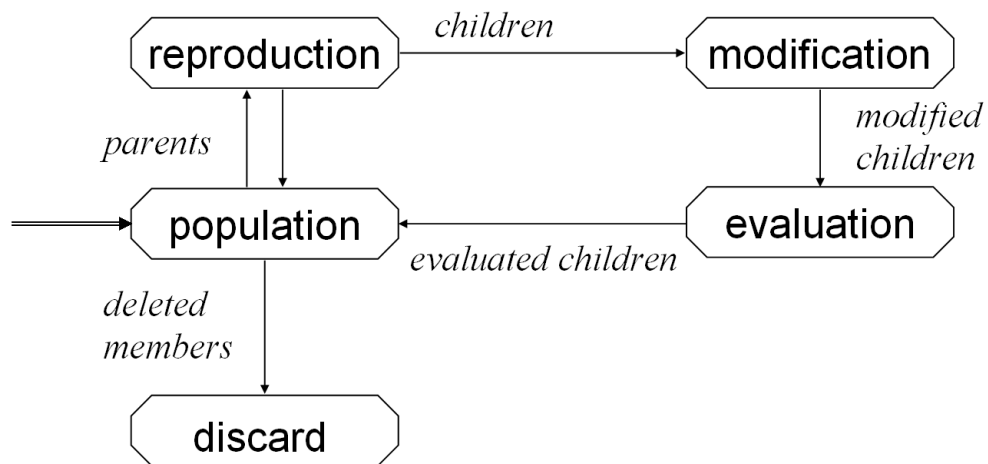


Figure 3- The GA reproductive cycle

An implementation of a GA usually starts with a population of chromosomes on which we select a set of parents for reproduction. The selected parents generate modified children by genes' recombination. A crossover occurs when genes came from a fusion of two different parents; however the recombination can be done by mutation of a single chromosome. The resulting children are used to form a new population that we hope to be better. The selection process is done evaluating their fitness (the more suitable they are the more chances they have to reproduce). This is repeated until some condition is satisfied. [19], [20]

| | | | |
|----------|-------------------|--------------------------|---------|
| Parent 1 | (0 1 1 0 1 0 0 0) | (0 1 0 0 1 0 0 0) | Child 1 |
| Parent 2 | (1 1 0 1 1 0 1 0) | (1 1 1 1 1 0 1 0) | Child 2 |

As with any method, the GA has its advantage and disadvantages. On next table there is a resume of the most important ones [21].

| Advantages | Disadvantages |
|---|--|
| Only uses function evaluations | Cannot use gradients |
| Easily modified for different problems | Cannot easily incorporate problem specific information |
| Handle noisy functions very well | Not good at identifying local optima |
| Handles large, poorly understood search spaces easily | No effective terminator |
| Good for multi-modal problems | Not effective for smooth uni-modal functions |
| Return a suite of solutions | Needs to be coupled with a local search technique |
| Very robust to difficulties in the evaluation of the objective function | |
| Easily parallelized | |

Table 2 Advantages and disadvantages of Genetic Algorithms

Genetics can be used in a hierarchical fuzzy model for pattern extraction and to neuro-fusion models complexity reduction. They can be used as an optimization technique, to extraction of knowledge, in combination with fuzzy rules, fuzzy membership, and with neural networks and fuzzy-logic.

A neuro-fuzzy-genetic model was proposed for data mining and fusion in the area of geoscience and petroleum reservoirs. The use of a neuro-fuzzy DNA model was propose for extraction of knowledge from seismic data and mapping the wireline logs into seismic data and reconstruction of porosity [12]. A list with more uses can be found on annex G.

In annex H one can find an example applied to traveling Salesman problem.

Expert Systems

Expert System is another method that can be used at decision level. A rule-based expert system is a set of rules that can be applied to a collection of facts in a repeatedly way by an engine. These rules represent heuristics that define a set of actions to be taken in a given situation and facts represent circumstances that describe a certain situation in the real world. [22]

The Expert Systems are present in oil industry applications such as: "Extra Pair of Eyes" (autonomous intelligent controlling systems); Pipeline and Production Supervision; Plant-wide network supervision and optimization; Abnormal Situations Management; Environment: Supervision and Control; Online-Analyzer verification and value inference; Planning, simulating and control of biochemical processes [23]. [24] have studied an expert system where a crude oil distillation column is designed to predict the unknown values of required product flow and temperature in required input feed characteristics. The system is also capable to optimize the distillation process with minimizing the model output error and maximizing the required oil production rate with respect to control parameter values. In combination with expert system the model also use neural networks and genetics algorithms.

In simple rule-based systems, there are two kinds of inference, forward chaining and backward chaining. In annex I and J one can find an example of forward and backward chaining systems.

More examples of expert systems applications are presented in annex L.

Blackboard Systems

A blackboard system is an architecture that can integrate multiple problem solving modules (referred to as knowledge databases). This type of integrated problem solvers can make use of more than one problem system in an attempt to overcome the inherent limitations of a single heuristic expert system. The problems solvers may also use different technologies. For example, a system might integrate a heuristic rule based reasoning system with a case-based reasoning system and possibly a model based system. These architectures can be used for a wide range of tasks such as classification, design, diagnosis, repair etc. [25]

Picture 15 represents a blackboard architecture for a speech understanding system. In this picture, one can see a set knowledge sources (solving modules) sharing a blackboard that is a common global database. The contents of the blackboard are often structured hierarchically and called hypotheses. Knowledge sources respond to changes on the blackboard, and interrogate and subsequently directly modify the blackboard. This modification results form the creation, modification and solution of hypotheses. The knowledge sources have the possibility to communicate and cooperate with each other through the blackboard. In blackboard architecture, each knowledge source responds only to a certain class or classes of hypotheses. These hypotheses, that a knowledge source responds to, often reflect the different levels in the blackboard's hierarchy. The blackboard holds the state of the problem solution, while the knowledge sources make modifications to the blackboard when appropriate. [25]

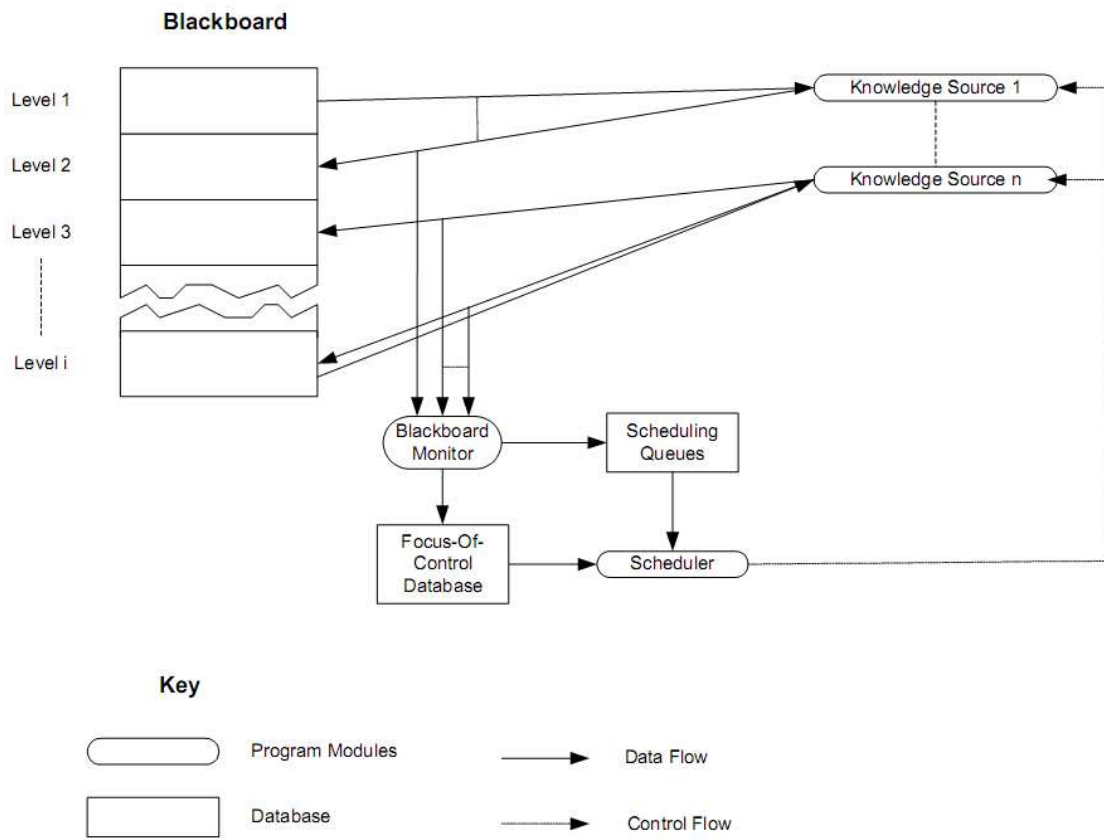


Figure 4- The architecture of Hearsay III- a speech understanding system

Hard and Soft Decision Taxonomy

Apart from the JDL’s based taxonomy, some authors suggest a different classification of methods regarding the decision process.

| Hard and Soft Decision Taxonomy | | |
|--|----------------------------|--|
| Decision Type | Method | Description |
| Hard decision | Boolean | Apply logical AND, OR to combine independent decisions [14]. |
| | Weighted sum score | Weight sensors by inverse of covariance and sum to derive score function [14]. |
| | M-of-N | Confirm decision based on m-out-of-n sensors that agree [14]. |
| Soft decision | Bayesian | See chapter <i>Taxonomy</i> |
| | Dempster-Shafer | |
| | Fuzzy variable | |
| | Neural networks | |
| | Genetics algorithms | |
| | Expert Systems | |
| | Blackboard Systems | |

Table 3- Hard and Soft Decision Taxonomy

This classification cluster methods inside two basic groups [14]:

- hard decisions which consist on a single optimum choice
- soft decisions, in which decision uncertainty in each sensor chain is maintained and combined with a composite measure of uncertainty.

In opposition to hard computing, soft computing is tolerant to imprecision, uncertainty, and partial truth. According to [12] soft computing is tractable, robust, efficient and inexpensive.

Why Bayesian?

According to the presented JDL based taxonomy, our data fusion problem belongs to feature level.

The method chose was the Bayesian Networks.

The main advantages of using Bayesian in data fusion summarized by [26, 27] are:

1. Bayesian statistics is a coherent system for quantifying objective and subjective uncertainties.
2. Bayesian provides principled methods for the model estimation and comparison and the classification of new observations.
3. Bayesian statistics provides a natural way to combine different sensor observations.
4. Bayesian statistics provides principle methods for dealing with missing information.
5. Bayesian provides a definition of "personal probability" which satisfies the same set of fundamental axioms which classical statisticians insist must hold for relative frequencies. This fact allows to focuses as much attention on the decision-maker as on the process or phenomenon under study.

Analyzing the data from the sensors it is very difficult to assess the probabilities in a classical way. Since we do not have sensor redundancy on each well, there is no way to confirm if the measure is correct or not. As so, assessing frequencies from the data is a though task and frustrating in a certain point.

The basic premise of Bayesian statistics is that all unknowns are treated as random variables and that the knowledge of these quantities is summarized via a probability distribution. [26]

Specifically Bayesian Networks extends the advantage of being able to reason in the presence of uncertainty, prior assumptions, and incomplete data. According to [28], they

can intermix expert judgment, statistical distributions, and observations in a single model. Further, they are able to learn from evidence in order to update their prior beliefs.

BN models have several advantages over regression-based models. BNs do not rely on point values of parameters that have been derived through some “best fit” procedure. Instead, the whole distribution of a variable is included. Similarly, BN models do not just predict a single value for a variable; they predict its probability distribution. By taking the marginal distributions of variables of interest, we get a ready-made means of providing quantitative risk assessment.

We didn't choose Neural Networks because we do not have data for the learning process.

Bayesian theory

To a better understanding of the Bayesian Networks let us start first with some notions about Bayesian theory.

Bayesian dates from the eighteen century and gains its roots with the English Reverend Thomas Bayes work [29]

This theory presents two important concepts: the Bayesian probabilities and the theorem (also known as rule).

In opposition to the frequency concept, the Bayesian can be related with partial beliefs in a different form to face probabilities. A probability can be thought as a quantitative measure of the strength of one's knowledge or of one's beliefs. This way, we can assess them using experts' knowledge and without having historical data. With this concept it is possible to deal with subjective beliefs and use them into a mathematical model.

Other idea subjacent to Bayesian is the conditionality. Instead of a classical approach, Bayes uses the notion of a probability of an event as a consequence of other events' probabilities.

An example of a conditional probability statement is that, given event B, the probability for event A to happen is x.

$$P(A|B) = x$$

This does means that $P(A) = x$ when B is true and everything else is irrelevant to A.

The Bayesian theorem is

$$P(A|B) = P(A | B) = \frac{P(B | A)P(A)}{P(B)}$$

Where [29]:

$P(A)$ is the prior probability or marginal probability of A. It is “prior” in the sense that it does not take into account any information about B;

$P(A|B)$ is the conditional probability of A, given B. It is also called the posterior probability because it is derived from or depends upon the specified value of B;

$P(B|A)$ is the conditional probability of B given A;

$P(B)$ is the prior or marginal probability of B, and acts as a normalizing constant;

Intuitively, Bayes' theorem in this form describes the way in which one's beliefs about observing 'A' are updated by having observed 'B';

Bayesian Network

According to [30] a Bayesian network (BN) consists of:

- a set of variables and a set of directed edges linking the variables;
- each variable has a set of mutually exclusive finite states;
- the edges express dependency relationships between the variables, forming a DAG (direct acyclic graph).
- for each variable A with parents B_1, \dots, B_n ; there is a conditional probability table (CPT) $P(A|B_1, \dots, B_n)$ to quantify the dependency. For variables without parents, the table is related to unconditional (also called marginal) distributions.

In a BN, when we observe a variable, the observation can be entered into the model by reducing their marginal probability distribution to a probability of one for the observed state and zero to the remaining states⁴. The presence of this new evidence updates the curve probability distribution of its children and the distributions of its parents. Applying Bayes's theorem, observations are propagated recursively through the model, updating their beliefs about probable causes and so learn from the evidence entered into the model. [28]

The Bayesian Networks allows any node to serve as either a query or an evidence variable. This is a very powerful characteristic and allows the usage of the network in several directions. Let's consider the BN presented on the next figure to better elucidate the different kinds of inference.

⁴ Some times the variable can be represented by a small interval containing the value in the continuous case. The discretization chosen must be a compromise between the accuracy of the evidence added and computational feasibility.

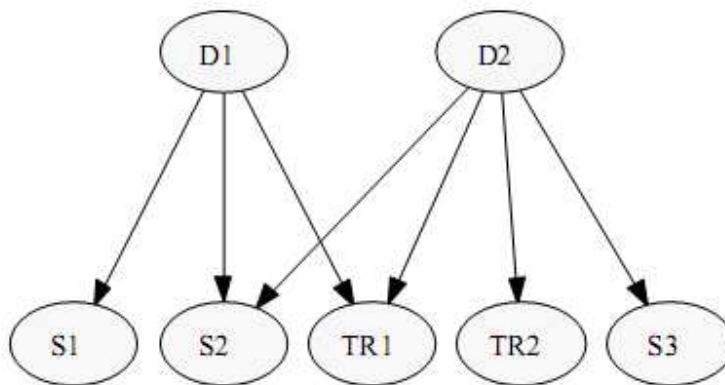


Figure 5⁵- BN example to elucidate different types of queries.
Example taken from the medical context where D1 and D2 are two different diseases; S1, S2, S3 symptoms; TR1, TR2 tests results realized to the patient.

There are four distinct kinds of inference that can be performed [31]:

- diagnostic inference (from effects to causes). Ex: given a symptom S1 infer the probability of the pathology D1, $P(D1|S1)$;
- causal inference (from causes to effects). Ex: given disease D2 find the most likely symptoms, $P(Si|D2)$;
- inter-causal inference (between causes of a common effect). Ex: given S2 infer $P(D1|S2)$, but adding evidence that D2 is true makes the probability of D1 go down. Although D1 and D2 nodes are independent the presence of one makes the other less likely;
- mixed inferences (combining two or more of the above).

According to [32], the structure of the directed graph is a representation of a factorization of the joint probability distribution. As many factorizations are possible, there are many graphs that are capable of encoding the same joint probability distribution. Of these, those that minimize the number of arcs should be more ideal.

[32] refer that from the point of view of knowledge engineering, graphs that reproduce the causal structure of the domain are particularly convenient, as they in general reflect

⁵ From: 31. Milho, I. and A. Fred, *A User-Friendly Development Tool for Medical Diagnosis Based on Bayesian Networks*. 2008, unknown.

the expert's understanding of the domain in a better way. These kinds of graphs also improve interaction with a human expert at the model building stage and are readily extendible with new information. Finally, these authors states that causal models facilitate user insight once a model is employed.

However, this causal approach to structuring the problem may present a lesser intuitive conditional probabilities assessment. This may become a serious difficulty when there is no sufficient data to help assessing these probabilities and we have to rely solely on the expert's believes. In this situation, we may have to choose the graph structure that match the available information in despite of the one who represents the relations in the most intuitive way.

We can find in the literature both approaches for the same kind of problem. [33] presented an example of a BN to diagnose Pneumonia. According to their idea the arrows indicate all of the conditional relationships between findings and diagnosis.

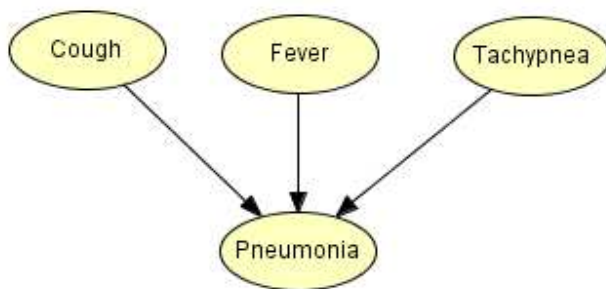


Figure 6⁶- BN example of an inverse graph approach.
In this approach arrows goes now from the symptom to the disease.

Serial connection is an import concept related to BN which help to introduce d-separation. To explain serial connection let us consider the example in next figure. In the present network, A has influence in B and B has influence in C. Consequently, evidence on A will influence B and B will transmit it to C, forming a communication channel. However, if the state of B is known, and the evidence is inserted in the network, A and C will became independent and communication channel is broken.

⁶ 33. Berner, E.S., H. K. J., and M.J. Ball, *Clinical Decision Systems. Theory and Practice*. 1996, New York: Springer-Verlag

As a conclusion, evidence may be transmitted through a serial connection unless the state of the variable in the connection is known. [30]

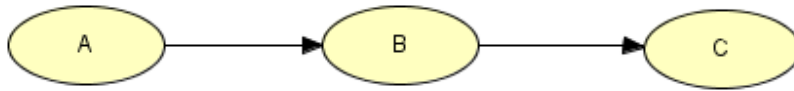


Figure 7. Example of serial connection.
When B is instantiated with evidence, the communication channel between A and B became blocked.

According to [34], a group of nodes Z is said to d-separate the disjoint groups of nodes X and Y when either the nodes Z are ascendants of both groups X and Y, or Z is an intermediate group of nodes. In the next two figures, we can see an example of both types of d-separation.

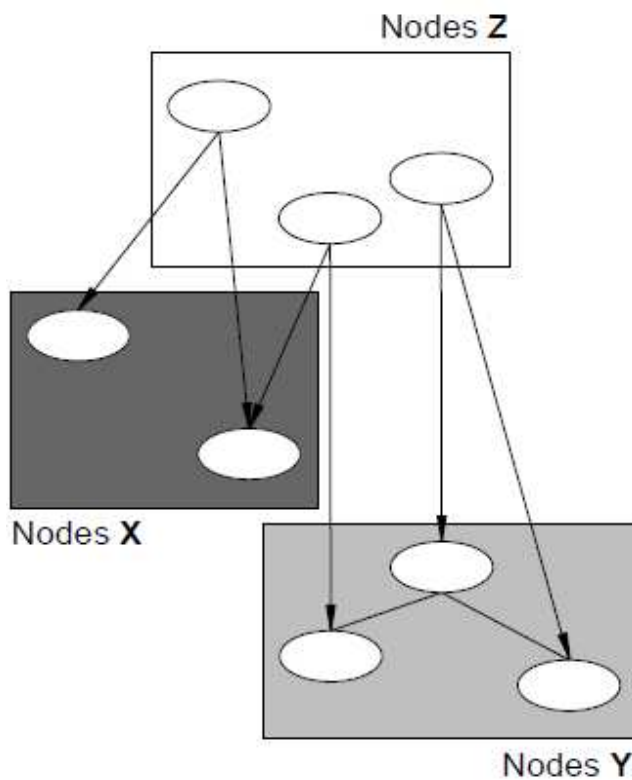


Figure 8⁷. Example of d-separation- Z nodes are ascendants of X and Y

⁷ From: 34. Carlos, A., T. Lorenzo, and R.-A. Juan, *Graphical models for problem solving*. Comput. Sci. Eng., 2000. 2(4): p. 46-57.

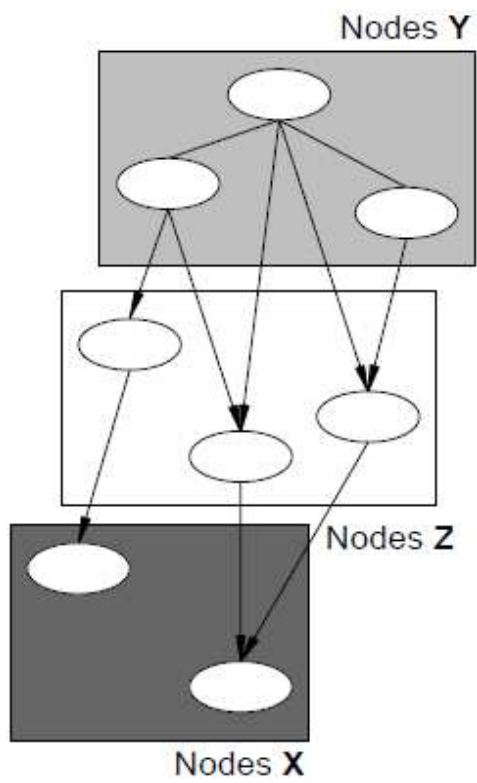


Figure 9. Example of d-separation.
Z is an intermediate group of nodes [34].

Bayesian Network – a simple example

To better understand the Bayesian Network concept lets consider the simple example⁸:

- we have a tree loosing its leaves and we want to know why;
- we know that if the tree is dry, this is can be the justification;
- however the losing of leaves can be an indication of a disease.

The problem can be represented with the Bayesian Network presented on next figure.

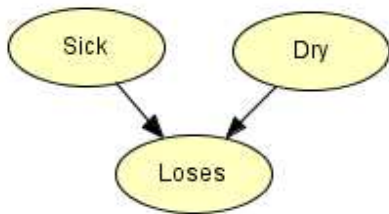


Figure 10- The tree problem

Let’s consider the following CPTs assessed by an expert in the domain.

| Sick="sick" | Sick="not" |
|-------------|------------|
| 0.1 | 0.9 |

Table 4- P(Sick)

| Dry="dry" | Dry="not" |
|-----------|-----------|
| 0.1 | 0.9 |

Table 5- P(Dry)

⁸ Based on example extracted from: 35. Manual, R. 2009, HUGIN 7.0

| | Dry="dry" | | Dry="not" | |
|-------------|-------------|------------|-------------|------------|
| | Sick="sick" | Sick="not" | Sick="sick" | Sick="not" |
| Loses="yes" | 0.95 | 0.85 | 0.90 | 0.02 |
| Loses="no" | 0.05 | 0.15 | 0.10 | 0.98 |

Table 6- P(Loses | Sick, Dry)

We can now determine some useful information:

- probability of Loses in a specific state.
e.g.: $P(\text{Loses} = \text{"yes"}) = 0.1832$
- marginal probability of Loses given evidences about Sick and/ or Dry states;
e.g.: $P(\text{Loses} = \text{"yes"} \mid \text{Sick} = \text{"sick"}, \text{Dry} = \text{"dry"}) = 0.95$;
- inferences about the probability of each parent, given evidence about the child's state;
e.g.: $P(\text{Sick} = \text{"sick"} \mid \text{Losses} = \text{"yes"}) = 0.49$
- inferences about the probability of one parent, given evidence about the child and other parent;
e.g.: $P(\text{Sick} = \text{"sick"} \mid \text{Losses} = \text{"yes"}, \text{Dry} = \text{"dry"}) = 0.11$

Applications

There are innumerable applications of the Bayesian theory which covers distinct fields of science.

However, Bayesian Networks approach' is still a relatively new research area. The main reason is because it requires significant computational power. Only in last few decades we started observing practical applications.

Bayesian Networks contributed with useful improvements on fields such as fault diagnosis and sensor accuracy.

In 1999, [36] presented a model to diagnose faults in airplane turbines with BN.

[37], in 2002, used Bayesian Networks coupled with multivariate state estimation to provide both fault detection and fault diagnostic capabilities for the Space Shuttle Main Engines. In this study, the sensors information is validated with residual estimation techniques. Then, if a fault occurs, a probability is assigned to the component that had the failure; finally, Bayesian networks are applied for diagnosis.

[38], developed a model to diagnose faults in networks of electric power distribution in the same year.

In 2003, [39] proposed a distributed solution using Bayesian Networks for the detection of environmental features in wireless sensor networks.

In the same year, [40] presented a model to predict the final quality of a software product. With BN they constructed a prediction model that focuses on the structure of the software development process explicitly representing complex relationships between metrics, and handling uncertain metrics, such as residual faults in the software products.

[41] in 2006 proposes the construction of a Bayesian network for valves failure diagnosis in industrial systems. The authors used some simulated data to train a network to learn the structure and parameters. Then, they designed the diagnosis probabilistic inference through the poly-trees algorithm. The network presents the valves failure probabilities according to the evidences that show up in entrance sensors.

[42] in the same year, developed a fault diagnosis in Autonomous Underwater Vehicles based on Bayesian Networks.

Other field that benefited with Bayesian Network was medicine. [43] stated in 1998 a large number of health care applications use DAGs.

In fact, we can find sophisticated BN approach to clinical dating from 1993. For example, [44] used a Bayesian network with continuously valued nodes to propose an optimal schedule of a certain drug delivery. To achieve this, CPT's were replaced with conditional density functions. This author used BN to infer the model parameters from a population and to probabilistically adapt it to a specific patient (taking into account the person's history). This information is then used to help in defining an optimal policy of drug delivery.

[32] presented in 1999 a Bayesian Network model for diagnosis of liver disorders and [45] in the year of 2000 a system to management of infectious disease.

[31] developed a user-friendly web based development tool for medical diagnosis based.

[45] stressed in 2004 the importance of Bayesian networks and other probabilistic graphical models as methods for discovering patterns in biomedical data and also as a basis for the representation of the uncertainties underlying clinical decision-making.

All this applications follow the "classical causal effect" approach. In fact it seems that the "inverted" approach has not been sufficiently studied.

Limitations

Bayesian Networks have some limitations in what concerns the difficulties to obtain the necessary parameters. [37] experience some of them during their work related to fault diagnostic capabilities for the Space Shuttle Main Engines:

1. prior probabilities of failure for each of the components are obtained from engineers and reliability test data;
2. conditional probabilities for some of the nodes are obtained from past reports and engineering estimates;
3. BN require tremendous numbers of parameters. For each node that has parents, a conditional probability is required for each state with regard to each combination of parent states. A single node may require hundreds of values.
4. conditional probabilities for multiple failure modes were not available, so Liu and Zhang calculated by averaging the values for each of the participating single failure cases.

Other crucial aspect that can turn into a limitation is the quality of the prior beliefs. [46] wrote that a Bayesian Network is only as useful as this prior knowledge is reliable. The author expresses that either an excessively optimistic or pessimistic expectation of the quality of these prior beliefs will distort the entire network and invalidates the results. He also emphasizes that selecting the proper distribution model to describe the data has a notable effect on the quality of the resulting network.

[47] experience certain limitations on Bayesian Networks for modeling genetic regulatory networks. The most important limitation they feel is the caution with which we must interpret the model. Observing the graph is easy to mislead between physical and statistical influences between variables.

[48] stated that mixed joint influences are more difficult to model, especially when one parent contributes with a positive and other with a negative influence.

Bayesian Networks may be difficult to model problems when causal relationships between variables are complex and there isn't available enough data to the network learn. It is the case of forward loops. However, these situations can be solved using more complex approach as Dynamic Bayesian Networks[49].

Chapter III - THE MODEL

The chapter starts with the formulation of the formal hypothesis on which relies the model. An outline of the methodology followed is presented. The necessary assumptions and considerations about the data's quality are then exposed. There is also a description of the necessary experiments realized. The chapter ends with the test and its results.

Hypothesis

With this work we intended to create a model to estimate sensor accuracy using data fusion. We believe that this could be done using Bayesian Networks.

The examples found in the literature generally assume one of the following:

- the existence of data to assess probabilities;
- the ability of an expert to easily express them as believes.

Unfortunately these conditions were not present. As so, we faced an extra challenge besides the regular problem modeling. We wanted to test if the application of BN was possible with such constraints.

From a formal point of view our aim could be formulated through the hypothesis:

H: In absence of data we may use Bayesian Network for sensor accuracy surveillance.

Following a certain methodology we tried to investigate its veracity.

Methodology

To test our hypothesis, the present methodology was followed:

- identification of the variables and their dependency relationships;
- delineation of the Bayesian Network structure;
- estimation of all the conditional probability tables (CPT) necessities to our BN;
- determination of the model inputs. The inputs are evidences we can observe through analysis of the well logs. In the present model we call it conditions.
- query the BN using information about conditions to obtain the sensors' accuracy probabilities;

If the results were according to our expectations the hypothesis would be considered as true.

Assumptions

The model is based on a set of conditions established with the help of experts in the domain.

If the sensors are correct they should obey to several conditions⁹ (C):

- $b_{ht}/b_{hp} = k_{te}$ (C1). The coefficient between the temperature and pressure should be constant. As a consequence of the ideal gas law, this should be valid for a certain time period;
- $b_{hp} > \min$ (C2). The borehole pressure should be over a minimum reference value;
- $b_{hp} - w_{hp} > \text{diffP}$ (C3). If the well is in production, there should be a pressure difference between the b_{hp} and b_{ht} ;
- $db(b_{hp}/w_{hp}) = k_{te}$ (C4). Relation between pressures on the choke should be constant for a stable choke aperture value.

We assume that the well is in production.

⁹ The aim of these conditions is to be generic and easy to understand despite of their accuracy. These definitions were not consensual between the experts. In order to implement the model to a specific reservoir, we suggest studying more specific and detailed conditions.

Data considerations

Statoil Hydro provided a log from 2 platforms, each one with 4 production wells connected to the same reservoir.

The measurements were from the borehole pressure (bhp), borehole temperature (bht), choke aperture, well head pressure (whp) and well head temperature (wht). These values were collected in 5 mins intervals over a month.

After analyzing the data applying some statistical figures of merit we observed the following problematic situations:

C1:

- A-4 presents clearly an irregular value; A-2, A-3; B-1 and A-1 are not so bad. We do not have information about the B-2 value.

C2:

- 2 wells presented an abnormal low bhp (one bellow 15 bars and the other negative (!));
- 1 well does not present bh values, as so, it had to be excluded from the analysis;

C3:

- A-2 and A-3 presents a higher pressure value at the well head than at borehole.

C4:

- A-1, A-2, A-3, A-4 presents a big variance between borehole and well head pressure at a stable choke.

The results are summarized in the next table.

| Conditions | WELL | | | | | | | |
|----------------------------------|------|------|------|-----|-----|-----|-----|-----|
| | A-1 | A-2 | A-3 | A-4 | B-1 | B-2 | B-3 | B-4 |
| C1 bht/bhp s ¹⁰ /mean | 6% | 0% | 0% | 26% | 9% | ?? | 3% | 0% |
| C2 Bhp mean (bar) | 279 | 14 | -100 | 187 | 180 | N/A | 220 | 230 |
| C3 Bhp –whp (bar) | 198 | -132 | -205 | 126 | 127 | ?? | 166 | 152 |
| C4 Bhp/whp s/mean | 24% | 23% | -24% | 24% | 12% | ?? | 6% | 32% |

Table 7- Analysis of data
Irregular situations represented by italic bold.

In this study we face a feasible data absence problem. As there is no acceptable data¹¹, it limits the model developing in these important aspects:

- determination of correlations between sensor values;
- determination of any probabilities;
- using the BN learning abilities to help establishing relations between variables.

Therefore, we have to solve the problem using merely domain expert's knowledge.

Consequently, we face the challenge of adapting the model in an easy way to incorporate this information.

¹⁰ Standard deviation.

¹¹ Details will be presented in the "Data considerations" section.

Experiments

In presence of the data constraints we experiment two different approaches in order to test our hypothesis. These approaches were based on different conditional probabilities concepts presented by [50]:

- A: where causal probabilities are those of the form $P(\text{TestResult}=\text{fail} \mid \text{Sensor}=\text{bad})$, indicating the likelihood that a particular test condition outcome is caused by the state of a certain sensor.
- B: where diagnostic probabilities are those of the form $P(\text{Sensor}=\text{bad} \mid \text{TestResult}=\text{fail})$, indicating the likelihood that a particular sensor is bad based on the fact that a certain condition test has failed.

In both approaches we used for C2 ($\text{bhp} > \text{min}$) and C3 ($\text{bhp} - \text{whp} > \text{diffP}$) boolean variables which represents the probability of the condition been satisfied or not. The same logic was followed for the sensors variables¹². We used 3 states variables for both approaches C1 ($\text{bhp}/\text{bhp} = \text{kte}$) and C4 ($\text{db}(\text{bhp}/\text{whp}) = \text{kte}$) pretending to model a bad; a good and an intermediate result of the conditions. Liu and Zhang, (2002) also used 3 states variables.

To model our network we used the Hugin Lite 7.1 software which can be found in:

<http://www.hugin.com>

¹² The sensor variable has a probability associated to Ok and Not_Ok states. One should interpret one has the complement of the other.

Approach A

It considers the sensors as parents and the conditions as children.

The logic behind is to reflect the sensors as the causes of the conditions' state.

The network provides the sensors' probabilities using Bayesian inference.

This modeling style seems to be the classical approach to fault diagnosis in engineering.

One possible network representation is shown in next figure.

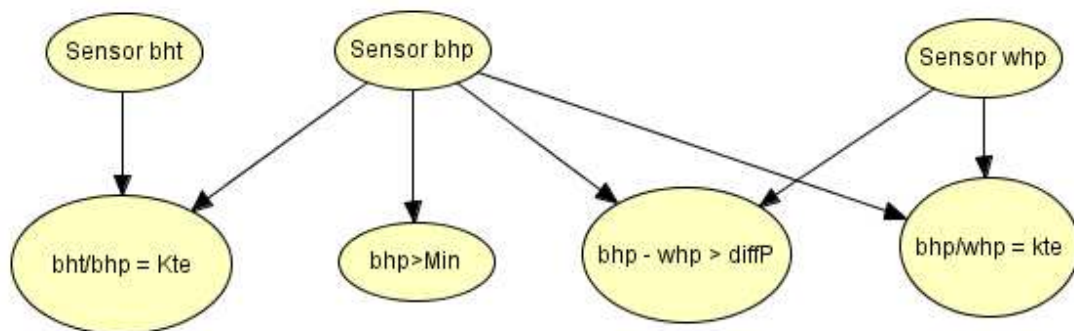


Figure 11- Bayesian Network according to approach A.

This approach has as advantage CPTs with few variables (maximum 3 variables).

However the children CPTs are not intuitive as they are in the form $P(\text{Condition}|\text{Sensor})$.

There is another problem related with the d-separation of the conditions. This states that the conditions are independent from each others given the sensors probabilities.

Despite the fact this approach seems to be simpler at a first glance, it became harder to access the CPTs in order to express the expert knowledge in a coherent way.

[50] stated this problem as “domain experts often experience difficulty arriving at the conditional probabilities in the causal direction, which are needed for the network design, as opposed to the probabilities in the diagnostic direction, which reflect their natural way of thinking.”

However, if we have considerable amount of feasible data it is possible to bypass this problem using the learning abilities of Bayesian Networks.

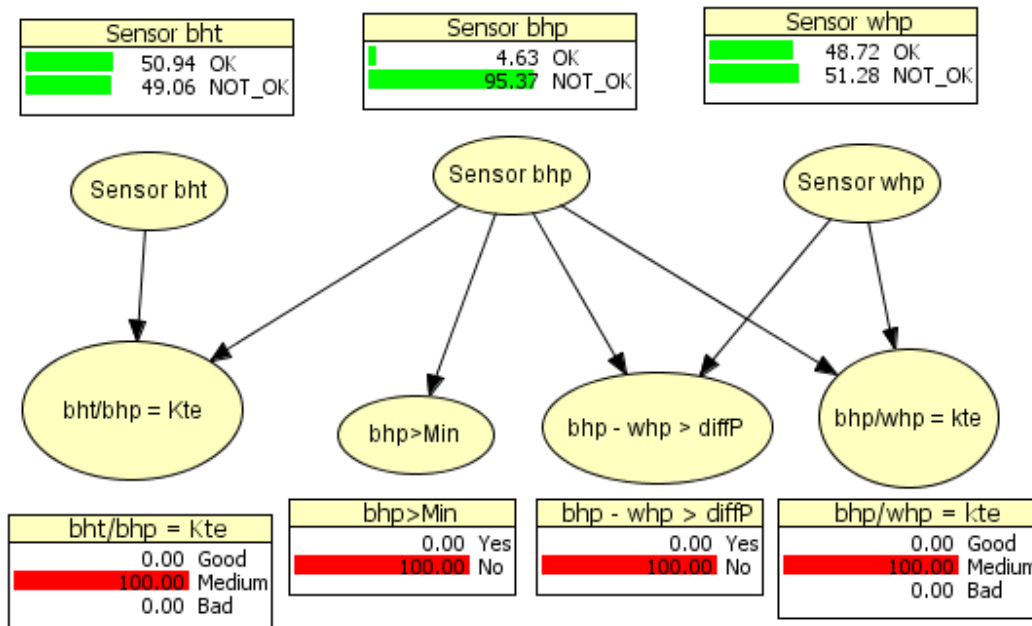


Figure 12- Example of one simulation¹³

In the last figure we can see a simulation of the BN using rough CPTs. Since $bhp > Min$ is false, the model can assume that Sensor bhp is not ok. So, bhp by itself justifies why the others conditions are not good. With this set of evidences, the model cannot clearly decide if others sensors are ok or not. This was according our expectations. Now, what if we realize that $bht/bhp = Kte$ is good? Since bhp is not working ok, we are not expecting good unless bht is also not ok.

¹³ In dark grey the evidences inserted. In light grey the probabilities provided by the model.

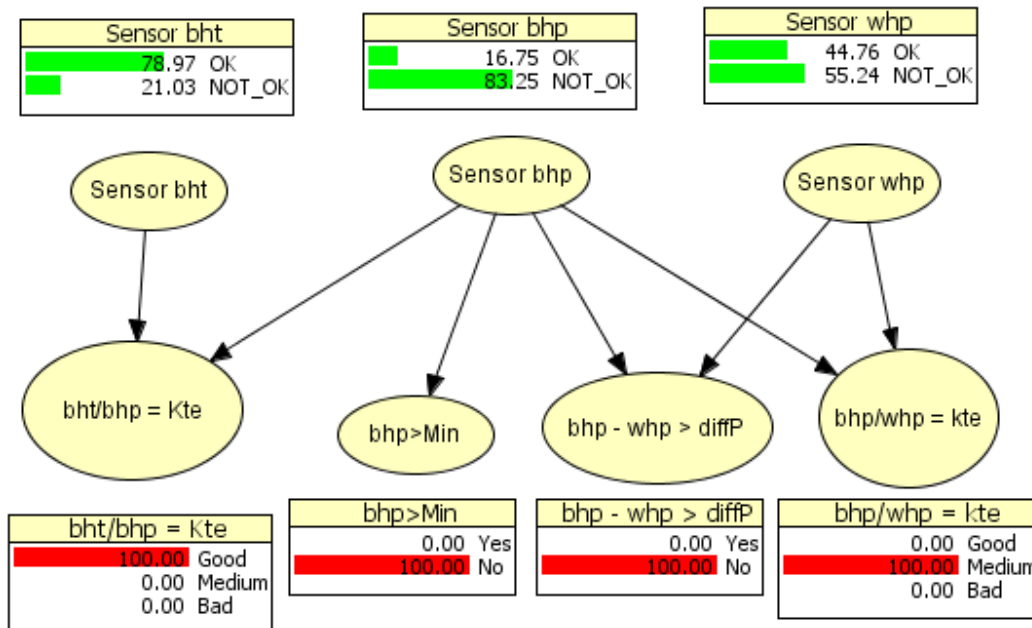


Figure 13- Example of an incoherent simulation.

As we can observe the model does not behave as expected. The justification is related with the conditions' d-separation. We could add more dependencies' relationships to improve the behavior. However, in that way we'll start to get complexes and non intuitive CPT.

One could easily think that given only the prior component probabilities ($P(C)$, $P(C')$), and the diagnostic conditional probabilities ($P(C|IT)$, $P(C'|IT)$), it is possible to uniquely determine the causal probabilities ($P(T|IC')$, $P(T'|IC')$ or ($P(T|IC)$, $P(T'|IC)$). However, as [50] proved this is not possible.

Approach B

This approach considers the conditions as parents and the sensors as children.

The model follows the logic of the “symptom \rightarrow diagnostic”, whereas the conditions are symptoms to diagnose the sensors’ accuracy.

This kind of approach is also found in the medical diagnostic context [33].

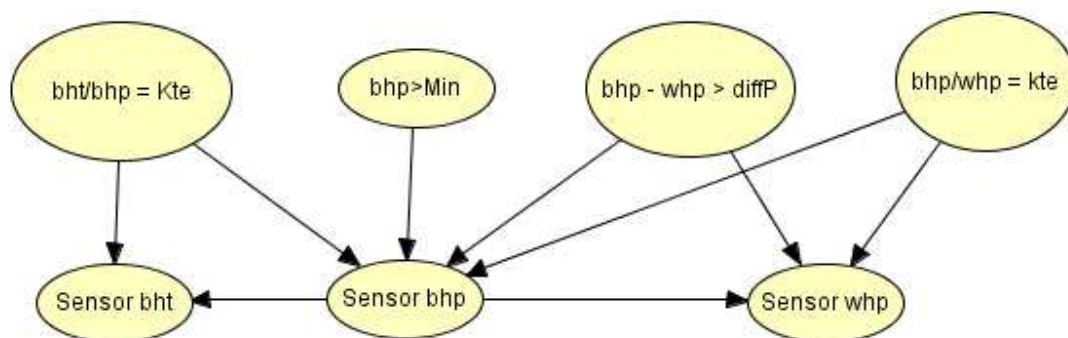


Figure 14- Bayesian Network according to approach B.

The network has now links between the sensors to express the interrelation between them and the conditions. At a first glance, these connections may suggest an erroneous physical relationship between the sensors. In the model context, one should interpret this variable as the diagnostic about the sensor and not the physical state. Even if these two concepts may seem similar, they differ in practical aspects because the knowledge of one diagnostic may influence the other sensor diagnostic. [47] alerts for some precaution in Bayesian models interpretations. They state that even BN are highly interpretable structures for representing statistical dependencies, they can be easily misleading if interpreted incorrectly. These authors stressed about the importance of the distinction between statistical interaction and physical interaction.

The mixed joint influence in Sensor bht from C1 and Sensor bhp was difficult to model as they have negative synergies. In a similar way we experienced the same problem in Sensor whp.

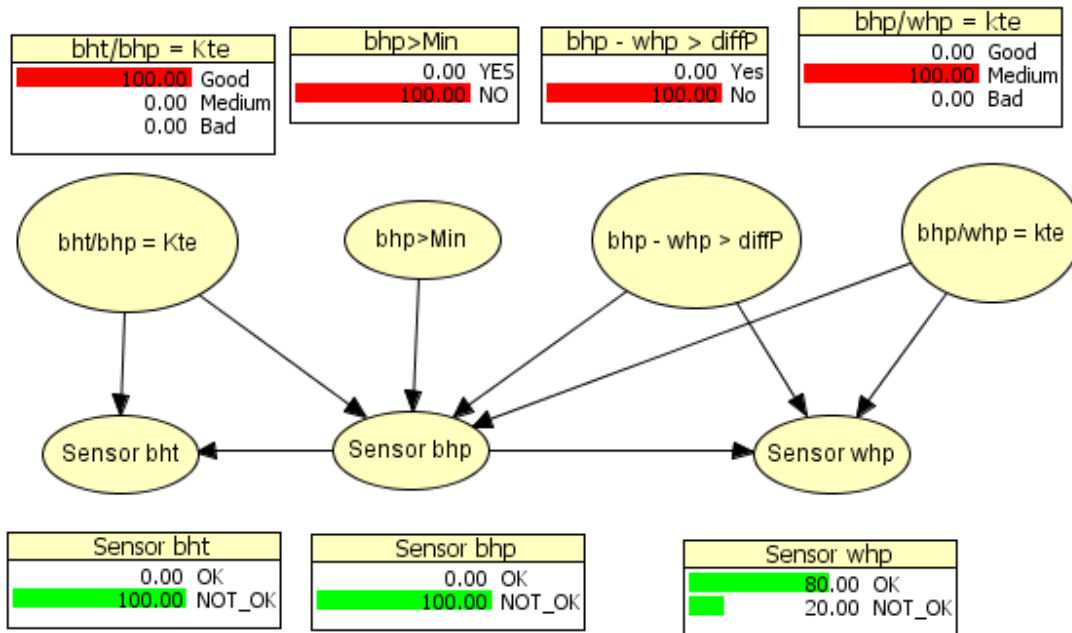


Figure 15- Example of BN under the same conditions as approach A.

We can now realize that BN assumes both sensors are not ok. As bhp is not ok, the model expects bhp/whp to be bad. Since this condition is just medium, whp can not be working perfectly.

As this approach seems more suitable to solve our given problem we tried to improve it adding an extra condition to expresses the age of sensors.

The age condition has two states which are:

- Old- when selected decrease the sensor accuracy;
- Neutral- when selected the sensor accuracy is not affected by this condition.

We also experimented adding more intermediate layers in a tentative to get CPTs with less variables and more easy to define. As suggested by [30] this was done by adding mediating variables.

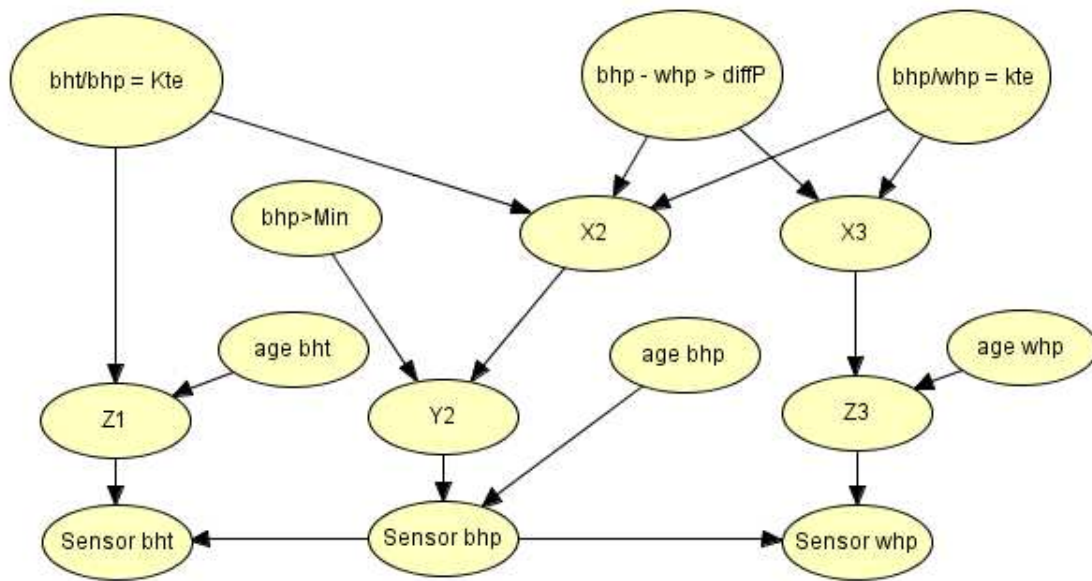


Figure 16- Experiment to reduce the CPT’s tables complexity.
 This was done by adding mediating variables Z1, Y2, X2, Z3, X3.

Even each CPT became easier to define, the overall model behavior was more difficult to delineate. These results are confirmed by [50] which state that multilayer networks are often very sensitive to conditional probabilities. These authors alert to the fact that probabilities have to be defined with greater accuracy because small perturbations in their values may result in radically different diagnostic conclusions.

They also refer: “In the choice between simple Bayesian networks or two-level Bayesian networks and a multilevel network one needs to carefully consider the expected diagnostic benefits versus the increased cost of the knowledge engineering, testing, and real-time execution.”

In consequence of the unsatisfactory results of our multilevel approaches the final model has only two levels as presented in the next figure.

The required CPTs can be found in annex M.

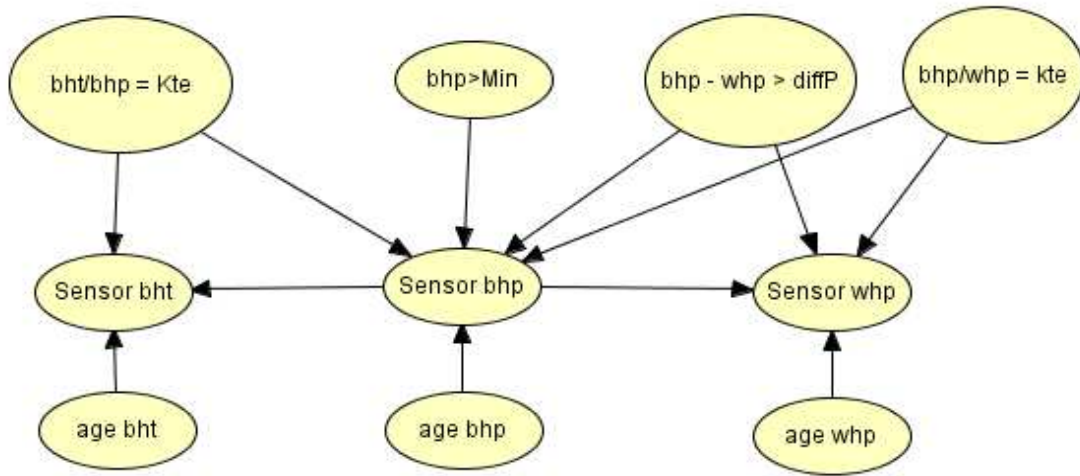


Figure 17- The final model including the sensor age condition

Test and results

We tested the final model to the problematic data described in the topic “data considerations”.

We classified the data into the conditions’ states which are the model’s inputs.

The criterion for the first and last condition is: Good <5%; 5% <= Medium < 20%; Bad >= 20%. The diffP is 120 bars and Min is 20 bars.

As we do not have information about the sensor age we set that condition on Neutral state.

| | | WELL | | | | | | |
|-----------|----------------------------|-------------|------------|------------|------------|------------|------------|------------|
| | Condition | A-1 | A-2 | A-3 | A-4 | B-1 | B-3 | B-4 |
| C1 | bht/bhp = kte | Med | Good | Good | Bad | Med | Good | Good |
| C2 | Bhp > Min | Yes | No | No | Yes | Yes | Yes | Yes |
| C3 | Bhp –whp > diffP | Yes | No | No | Yes | Yes | Yes | Yes |
| C4 | Bhp/whp = kte | Bad | Bad | Bad | Bad | Med | Med | Bad |

Table 8- Classification into conditions’ states¹⁴

The results are presented in the next table. As mentioned before, the variable sensor gives the probabilities of the state OK and their complementary NOT_OK. If sensor is measuring data correctly the state OK is more probable than NOT_OK. An equal probability distribution occurs when the model does not have enough information to decide about the sensor accuracy.

¹⁴ Due to lacking information we excluded B-2.

| A-1 | A-2 / A-3 | A-4 |
|---|---|---|
| <div style="border: 1px solid black; padding: 2px; margin-bottom: 5px;"> <p style="text-align: center; background-color: #ffffcc;">Sensor whp</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 60%;"></div> <div style="width: 35%;"> <p>58.22 OK</p> <p>41.78 NOT_OK</p> </div> </div> </div> <div style="border: 1px solid black; padding: 2px; margin-bottom: 5px;"> <p style="text-align: center; background-color: #ffffcc;">Sensor bht</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 60%;"></div> <div style="width: 35%;"> <p>56.00 OK</p> <p>44.00 NOT_OK</p> </div> </div> </div> <div style="border: 1px solid black; padding: 2px;"> <p style="text-align: center; background-color: #ffffcc;">Sensor bhp</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 60%;"></div> <div style="width: 35%;"> <p>20.00 OK</p> <p>80.00 NOT_OK</p> </div> </div> </div> | <div style="border: 1px solid black; padding: 2px; margin-bottom: 5px;"> <p style="text-align: center; background-color: #ffffcc;">Sensor whp</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 60%;"></div> <div style="width: 35%;"> <p>100.00 OK</p> <p>0.00 NOT_OK</p> </div> </div> </div> <div style="border: 1px solid black; padding: 2px; margin-bottom: 5px;"> <p style="text-align: center; background-color: #ffffcc;">Sensor bht</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 60%;"></div> <div style="width: 35%;"> <p>0.00 OK</p> <p>100.00 NOT_OK</p> </div> </div> </div> <div style="border: 1px solid black; padding: 2px;"> <p style="text-align: center; background-color: #ffffcc;">Sensor bhp</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 60%;"></div> <div style="width: 35%;"> <p>0.00 OK</p> <p>100.00 NOT_OK</p> </div> </div> </div> | <div style="border: 1px solid black; padding: 2px; margin-bottom: 5px;"> <p style="text-align: center; background-color: #ffffcc;">Sensor whp</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 60%;"></div> <div style="width: 35%;"> <p>64.11 OK</p> <p>35.89 NOT_OK</p> </div> </div> </div> <div style="border: 1px solid black; padding: 2px; margin-bottom: 5px;"> <p style="text-align: center; background-color: #ffffcc;">Sensor bht</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 60%;"></div> <div style="width: 35%;"> <p>90.00 OK</p> <p>10.00 NOT_OK</p> </div> </div> </div> <div style="border: 1px solid black; padding: 2px;"> <p style="text-align: center; background-color: #ffffcc;">Sensor bhp</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 60%;"></div> <div style="width: 35%;"> <p>10.00 OK</p> <p>90.00 NOT_OK</p> </div> </div> </div> |
| B-1 | B-3 | B-4 |
| <div style="border: 1px solid black; padding: 2px; margin-bottom: 5px;"> <p style="text-align: center; background-color: #ffffcc;">Sensor whp</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 60%;"></div> <div style="width: 35%;"> <p>48.00 OK</p> <p>52.00 NOT_OK</p> </div> </div> </div> <div style="border: 1px solid black; padding: 2px; margin-bottom: 5px;"> <p style="text-align: center; background-color: #ffffcc;">Sensor bht</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 60%;"></div> <div style="width: 35%;"> <p>52.00 OK</p> <p>48.00 NOT_OK</p> </div> </div> </div> <div style="border: 1px solid black; padding: 2px;"> <p style="text-align: center; background-color: #ffffcc;">Sensor bhp</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 60%;"></div> <div style="width: 35%;"> <p>40.00 OK</p> <p>60.00 NOT_OK</p> </div> </div> </div> | <div style="border: 1px solid black; padding: 2px; margin-bottom: 5px;"> <p style="text-align: center; background-color: #ffffcc;">Sensor whp</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 60%;"></div> <div style="width: 35%;"> <p>45.00 OK</p> <p>55.00 NOT_OK</p> </div> </div> </div> <div style="border: 1px solid black; padding: 2px; margin-bottom: 5px;"> <p style="text-align: center; background-color: #ffffcc;">Sensor bht</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 60%;"></div> <div style="width: 35%;"> <p>50.00 OK</p> <p>50.00 NOT_OK</p> </div> </div> </div> <div style="border: 1px solid black; padding: 2px;"> <p style="text-align: center; background-color: #ffffcc;">Sensor bhp</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 60%;"></div> <div style="width: 35%;"> <p>50.00 OK</p> <p>50.00 NOT_OK</p> </div> </div> </div> | <div style="border: 1px solid black; padding: 2px; margin-bottom: 5px;"> <p style="text-align: center; background-color: #ffffcc;">Sensor whp</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 60%;"></div> <div style="width: 35%;"> <p>40.56 OK</p> <p>59.44 NOT_OK</p> </div> </div> </div> <div style="border: 1px solid black; padding: 2px; margin-bottom: 5px;"> <p style="text-align: center; background-color: #ffffcc;">Sensor bht</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 60%;"></div> <div style="width: 35%;"> <p>50.00 OK</p> <p>50.00 NOT_OK</p> </div> </div> </div> <div style="border: 1px solid black; padding: 2px;"> <p style="text-align: center; background-color: #ffffcc;">Sensor bhp</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 60%;"></div> <div style="width: 35%;"> <p>50.00 OK</p> <p>50.00 NOT_OK</p> </div> </div> </div> |

Table 9- Results presented by the model

The results confirm our initial expectations regarding which sensor is most probable to be the cause of the data problem on each well.

These outcomes reinforce the possibility to construct BN even in absence of data. The initial hypothesis was proven to be true. As so, Bayesian is a technique suitable for surveillance of sensors' accuracy in a production well.

Chapter IV - CONCLUSIONS

In this chapter we present the final conclusions. Some considerations about practical usages are given. It ends with some suggestions for further developments.

Conclusions

With this work we could conclude that our initial hypothesis is true. Therefore Bayesian Network is a suitable method for surveillance of sensors' accuracy in a production well.

We found that when facing situations of missing feasible data to construct the model and limited expert knowledge it may become easier to invert the causal structure. This can be achieved by slightly modifying the variables and the relations meanings in order to express expert' believes. However, this should only be done if the model is still easily understandable.

The “inverted” approach has been rarely used, though it can be a better alternative under special constraints. It can be especially useful helping the expert in the critical task of assessing his believes of probabilities.

Despite our satisfactory results, one should take into consideration the following aspects were simplifications:

- the conditions that variables should obey;
- the probabilities assessment;
- the number of states on each condition;
- the classification criterions of this states.

As so, before applying this model in practice we present the following suggestions:

- revise the conditions;
- include more conditions;
- increase the conditions states and improve the criterions;
- revise the CPTs.

Further developments

It would be interesting to extend this model in a way to use the measurements from the others wells.

Since the several wells are all connected to the same reservoir, one could try to find correlations between the several sensors.

If we can find these correlations we could incorporate a sensor redundancy approach into this model.

It would also be interesting trying a Dynamic Bayesian Network approach to include the evolution of the several sensor measurements over time.

References

1. Carvalho, H.S., et al. *A general data fusion architecture*,. in *Proceedings of the Sixth International Conference of Information Fusion*. 2003.
2. Dailey, D.J., P. Harn, and P.-j. Lin, *Its Data Fusion*. 1996, Washington State Transportation Center (TRAC).
3. Kessler, O. and F. White, *Data Fusion Perspectives and Its Role in Information Processing*, in *Handbook of Multisensor Data Fusion - Theory and Practice*, M.E. Liggins, D.L. Hal., and J. Llinas, Editors. 2009, CRC Press: Boca Raton.
4. Welch, G. and G. Bishop, *An Introduction to the Kalman Filter*. 2006, Department of Computer Science; University of North Carolina; Chapel Hill.
5. Silva, J.M.L.d., *Sensor fusion and behaviours for the CAMBADA Robotic Soccer Team*, in *DETI*. 2008, University of Aveiro: Aveiro.
6. Cuevas, E., D. Zaldivar, and R. Rojas. *Kalman filter for vision tracking*. 2005 [cited; Available from: <http://page.mi.fu-berlin.de/zaldivar/files/tr-b-05-12.pdf>.
7. Shafer, G. *Dempster-Shafer Theory*. [cited; Available from: <http://www.glennshafer.com/assets/downloads/articles/article48.pdf>.
8. Koks, D. and S. Challa, *An Introduction to Bayesian and Dempster-Shafer Data Fusion*. 2005, Australian Government – Department of Defence.
9. Tunaley, J.K.E., T.M. Sibbald, and M.T. Rey-Cousins. *Use of the Dempster-Shafer algorithm in the detection of ship wakes from synthetic aperture radar images*. in *Acoustics, Speech, and Signal Processing, 1991. ICASSP-91., 1991 International Conference on*. 1991.
10. Mena, J.B. and M. J.A., *Color image segmentation using the Dempster-Shafer theory of evidence for the fusion of texture*. 2003, Alcalá University, ISPRS Archives, Munich. p. pp. 17.-19.
11. Bridges, S., J. Hodges, and B. Wooley. *Preliminary Results in the Use of Dempster Shafer Theory for a Radiological Waste Characterization Expert System*. 1996 [cited; Available from: ftp://ftp.cs.msstate.edu/publications/tech_reports/961212.ps.Z
12. Nikraves, M. and F. Aminzadeh, *Soft Computing and Intelligent Data Analysis in Oil Exploration*, in *Soft Computing and Intelligent Data Analysis in Oil Exploration*, M. Nikraves, F. Aminzadeh, and L.A. Zadeh, Editors. 2003, Elsevier: Amesterdam.
13. Clabaugh, C., D. Myszewski, and J. Pang. *Applications of neural networks*. 2000 [cited; Available from: <http://www-cse.stanford.edu/classes/sophomore-college/projects-00/neural-networks/Applications/index.html>.
14. Waltz, E. and T. Waltz, *Principles and Practice of Image and Spatial Data Fusion*, in *Handbook of Multisensor Data Fusion - Theory and Practice*, M.E. Liggins, D.L. Hal., and J. Llinas, Editors. 2009, CRC Press: Boca Raton.
15. Steinberg, A.N., *Foundations of Situation and Threat Assessment*, in *Handbook of Multisensor Data Fusion - Theory and Practice*, M.E. Liggins and D.L. Hal., Editors. 2009, CRC Press: Boca Raton.

References

16. Swanson, D.C., *Adapting Data Fusion to Chemical and Biological Sensors*, in *Handbook of Multisensor Data Fusion - Theory and Practice*, M.E. Liggins and D.L. Hall, Editors. 2009, CRC Press: Boca Raton.
17. Kaehler, S.D. *Fuzzy Logic Tutorial - An Introduction*. 1998 [cited; Available from: <http://www.seattlerobotics.org/encoder/Mar98/fuz/flindex.html>].
18. Harris, J., *Fuzzy Logic Applications in Engineering Science*. Intelligent Systems, Control and Automation: Science and Engineering. Vol. 29. 2006, Dordrecht: Springer.
19. Whitley, D. *A Genetic Algorithm Tutorial*. [cited; Available from: http://samizdat.mines.edu/ga_tutorial/ga_tutorial.ps].
20. Williams, W. *Genetics Algorithms: A Tutorial*. [cited; Available from: <http://www.dbai.tuwien.ac.at/staff/musliu/ProblemSolvingAI/Class9GATutorial.pdf>].
21. Carter, J.N., *Introduction to using genetic Algorithms*, in *Soft Computing and Intelligent Data Analysis in Oil Exploration*, M. Nikravesh, F. Aminzadeh, and L.A. Zadeh, Editors. 2003, Elsevier: Amsterdam.
22. Soe, S.M.M. and M.P.P. Zaw. *Design and Implementation of Rule-based Expert System for Fault Management in PROCEEDINGS OF WORLD ACADEMY OF SCIENCE, ENGINEERING AND TECHNOLOGY 2008*.
23. Gross, M.S., *Survey of Real-time Expert Systems with Productive Applications in the CPI and the Approach of DEA Mineraloel AG*, in *ESIT '99 - European Symposium on Intelligent Techniques*. 1999: Orthodox Academy of Crete, Greece.
24. Motlaghi, S., F. Jalali, and M.N. Ahmadabadi, *An expert system design for a crude oil distillation column with the neural networks model and the process optimization using genetic algorithm framework*. Expert Systems with Applications: An International Journal archive, 2008. **35**(4).
25. Hunt, J. *Blackboard Architectures*. 2002 [cited].
26. Mitchell, H.B., *Multi-Sensor Data Fusion. An introduction*. 2007, Berlin: Springer –Verlag.
27. Raeside, D.E., *Bayesian Statistics: Mathematical Tools for Computers*. Computer, 1975: p. pp. . 65-73.
28. Hearty, P., et al., *Predicting Project Velocity in XP Using a Learning Dynamic Bayesian Network Model*. IEEE Trans. Softw. Eng., 2009. **35**(1): p. 124-137.
29. *Bayes' theorem*. [cited; Available from: http://en.wikipedia.org/wiki/Bayes'_theorem].
30. Jense, F.V., *Bayesian networks and decision graphs*. Statistics for engineering and information science. 2001, New York: Springer.
31. Milho, I. and A. Fred, *A User-Friendly Development Tool for Medical Diagnosis Based on Bayesian Networks*. 2008, unknown.
32. Onisko, A., M.J. Druzdzal, and M.H. Wasyluk, *A Bayesian Network Model for Diagnosis of Liver Disorders*, in *Research Report CBMI-99-27*, C.f.B. Informatics, Editor. 1999, University of Pittsburgh.
33. Berner, E.S., H. K. J., and M.J. Ball, *Clinical Decision Systems. Theory and Practice*. 1996, New York: Springer-Verlag
34. Carlos, A., T. Lorenzo, and R.-A. Juan, *Graphical models for problem solving*. Comput. Sci. Eng., 2000. **2**(4): p. 46-57.

References

35. Manual, R. 2009, HUGIN 7.0
36. Mast, T.A., et al. *Bayesian belief networks for fault identification in aircraft gas turbine engines*. in *Control Applications, 1999. Proceedings of the 1999 IEEE International Conference on*. 1999.
37. Edwina, L. and Z. Du, *Diagnosis of Component Failures in the Space Shuttle Main Engines Using Bayesian Belief Networks: A Feasibility Study*, in *Proceedings of the 14th IEEE International Conference on Tools with Artificial Intelligence*. 2002, IEEE Computer Society.
38. Chien, C.F., S.L. Chen, and Y.S. Lin, *Using Bayesian Network for Fault Location on Distribution Feeder*. *Power Engineering Review, IEEE*, 2002. 22(1): p. 72-72.
39. Krishnamachari, B., et al., *On the Complexity of Distributed Self-Configuration in Wireless Networks*. *Telecommunication Systems* 2004. Volume 22, Numbers 1-4 / January, 2003: p. 33-59.
40. Amasaki, S., et al. *A Bayesian belief network for assessing the likelihood of fault content*. in *Software Reliability Engineering, 2003. ISSRE 2003. 14th International Symposium on*. 2003.
41. Ramirez, J.C. and A.S. Piqueras, *Learning Bayesian Networks for Systems Diagnosis*, in *Proceedings of the Electronics, Robotics and Automotive Mechanics Conference - Volume 02*. 2006, IEEE Computer Society.
42. Changting, S., Z. Rubo, and Y. Ge, *Fault Diagnosis of AUV Based on Bayesian Networks*, in *Proceedings of the First International Multi-Symposiums on Computer and Computational Sciences - Volume 2 (IMSCCS'06) - Volume 02*. 2006, IEEE Computer Society.
43. Lauritzen, S.L. and D.J. Spiegelhalter, *Local computations with probabilities on graphical structures and their application to expert systems*, in *Readings in uncertain reasoning*. 1990, Morgan Kaufmann Publishers Inc. p. 415-448.
44. Bellazzi, R., *Drug delivery optimization through Bayesian networks: an application to erythropoietin therapy in uremic anemia*. *Comput. Biomed. Res.*, 1993. **26**(3): p. 274-293.
45. Lucas, P.J.F., et al., *A Probabilistic and Decision-Theoretic Approach to the Management of Infectious Disease at the ICU*. *Artificial Intelligence in Medicine*, 2000.
46. Niedermayer, D. *An Introduction to Bayesian Networks and their Contemporary Applications*. 1998 [cited; Available from: <http://www.niedermayer.ca/papers/bayesian/>].
47. Hartemink, A.J., et al., *Bayesian methods for elucidating genetic regulatory networks*. *Intelligent Systems, IEEE*, 2002. **17**(2): p. 37-43.
48. Chang, R., M. Stetter, and W. Brauer, *Quantitative Inference by Qualitative Semantic Knowledge Mining with Bayesian Model Averaging*. *Knowledge and Data Engineering, IEEE Transactions on*, 2008. **20**(12): p. 1587-1600.
49. Murphy, K.P., *Dynamic Bayesian Networks*, in *A dissertation submitted in partial satisfaction of the requirements for the degree of Doctor of Philosophy*. 2002, University of California: Berkeley.
50. Przytula, K.W. and D. Thompson. *Construction of Bayesian networks for diagnostics*. in *Aerospace Conference Proceedings, 2000 IEEE*. 2000.

References

51. CAMBADA, C.A.M.r.w.A.D.A. *CAMBADA Robotic Soccer* 2009 [cited; Available from: <http://www.ieeta.pt/atri/cambada>.
52. Riggi, F. and R. Harmouche, *A web project on Bayes Decision Rule - Discrete Features*. 2006.
53. Ardis, P. *Notes on Dempster Shafer Belief Functions*. [cited; Available from: <http://www.cs.rochester.edu/~ardis/DempsterShafer.pdf>.
54. Stergiou, C. and D. Siganos. *NEURAL NETWORKS. SURPRISE* 1996 [cited; Available from: http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.html.
55. Alba, E. and C. Cotta. 1998 [cited; Available from: <http://www.lcc.uma.es/~ccottap/semEC/>.
56. Paine, J. *Expert Systems*. 1996 [cited; Available from: <http://www.j-paine.org/students/lectures/lect3/node1.html>
57. *THE APPLICATIONS OF EXPERT SYSTEMS*. [cited 2009 May]; Available from: http://www.wtec.org/loyola/kb/c1_s2.htm.

ANNEXES

ANNEX A- Exemple of Kalman's filter applied to robotic football

Object tracking is used in robotic football for determination of ball's position in the field. The robot's trepidation and environmental factors provides noise to the measures. The CAMBADA¹⁵ robotic football team uses Kalman to filter this noise and the results can be seen on the picture 1 [51].

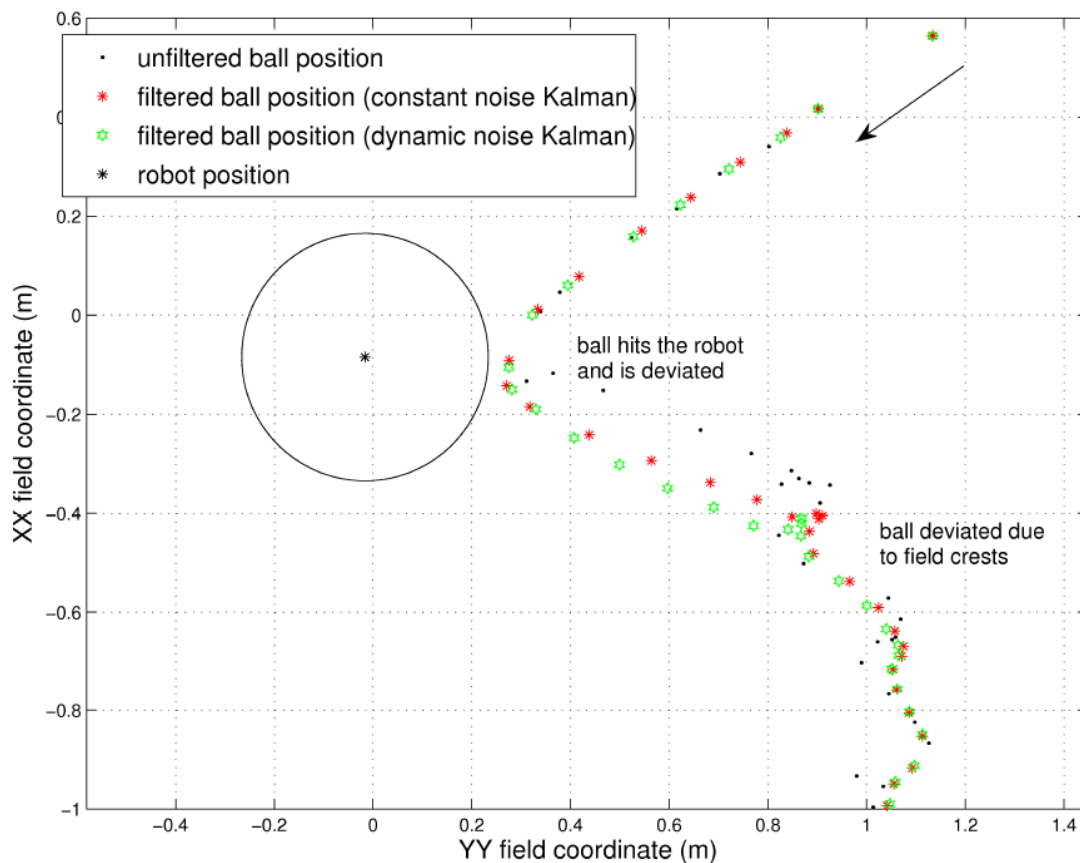


Figure 18. Kalman filter application

The read measures of the position are represented with black dots. Positions filtered by the Kalman filter with constant noise represented with stars. Positions filtered by the

¹⁵ CAMBADA- The 2008's robotic soccer world championship winner from University of Aveiro; Portugal - uses Kalman filter to help tracking the position and velocity of the ball.

ANNEXES

Kalman filter with dynamic noise¹⁶ represented with hexagrams. The robot position is represented by the star in the circumference center. [5]

¹⁶ Dynamic noise increases with ball's distance

ANNEX B- Example applied to pattern recognition¹⁷

Let's consider a three dimensional binary feature vector $\mathbf{X}=(x_1,x_2,x_3) = (0,1,1)$ that we will attempt to classify with one of the following classes:



and let's say that the prior probability for class 1 is $P(\omega_1)= 0.6$ while for class 2 is $P(\omega_2)= 0.4$. Hence, it is already evident that there is a bias towards class 1.

Additionally, we know that likelihoods of each independent feature is given by p and q where:

$$p_i = P(x_i=1|\omega_1) \text{ and } q_i = P(x_i=1|\omega_2)$$

meaning that we know the probability (or likelihood) of each independent feature given each class - these values are known and given:

$$p = \{0.8, 0.2, 0.5\} \text{ and } q = \{0.2, 0.5, 0.9\}$$

therefore, the discriminant function is $g(\mathbf{x}) = g_1(\mathbf{x}) - g_2(\mathbf{x})$ or by taking the log of both sides:

¹⁷ Extracted from 52. Riggi, F. and R. Harmouche, *A web project on Bayes Decision Rule - Discrete Features*. 2006.

ANNEXES

$$g(x) = \log \frac{p(\mathbf{x}|\omega_1)}{p(\mathbf{x}|\omega_2)} + \log \frac{p(\omega_1)}{p(\omega_2)}$$

however, since the problem definition assumes that X is independent, the discriminant function can be calculated by:

$$g(x) = \sum_{i=1}^d w_i x_i + w_0$$

with

$$w_i(x) = \ln \frac{p_i(1 - q_i)}{q_i(1 - p_i)} \quad i = 1, \dots, d$$

$$w_1 = \ln \frac{0.8(1-0.2)}{0.2(1-0.8)} = 2.77; \quad w_2 = \ln \frac{0.2(1-0.5)}{0.5(1-0.2)} = -1.39;$$

$$w_3 = \ln \frac{0.5(1-0.9)}{0.9(1-0.5)} = -2.19$$

$$w_0 = \ln \left(\frac{0.6}{0.4} \right) + \ln \left(\frac{1-0.8}{1-0.2} \right) + \ln \left(\frac{1-0.2}{1-0.5} \right) + \ln \left(\frac{1-0.5}{1-0.9} \right) = 1.0986$$

$$g(x) = 2.77x_1 - 1.39x_2 - 2.19x_3 + 1.0986$$

After inputting the x_i values into the discriminant function, the answer $g(\mathbf{x}) = -2.4849$. Therefore this belongs to class 2. Below is a plot of the decision boundary surface.

ANNEXES

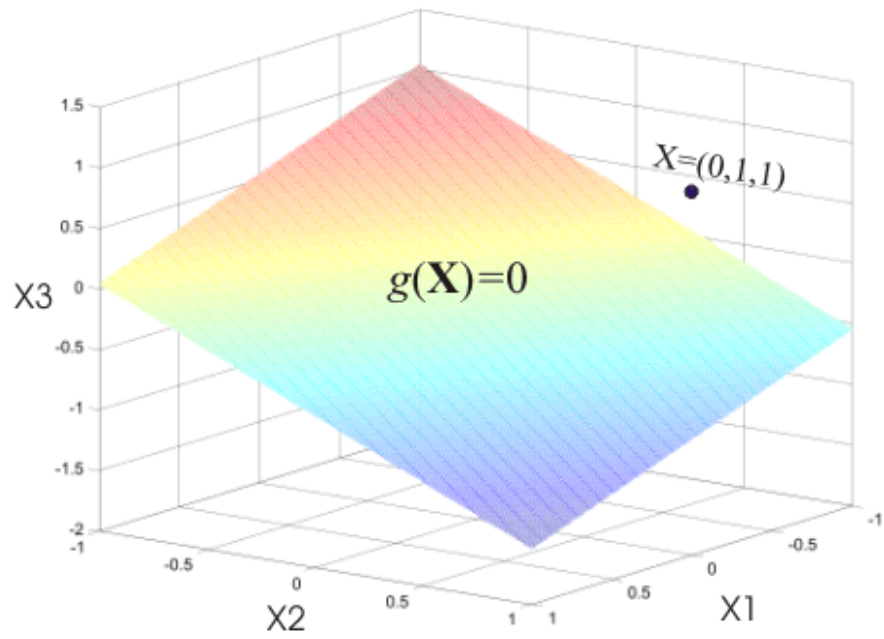


Figure 19- Decision Boundaries

All points above the plane belong to class ω_2 since if $X = (0,1,1)$, $g(\mathbf{x}) = -2.4849 < 0$.

ANNEX C- Simple example of Dempster Shafer¹⁸

Say we park our car in our usual spot but we do not know whether a friend might be picking it up during the day. Say Sue comes into the office and says your car is not there. But we know that Sue is absent minded and so only 90% of the time is she looked in the right spot. Thus, 90% of the time we have good evidence that the car is gone, whereas the other 10% we have no information one way or the other. Thus, we might believe that the probability that the car is missing is at least .9, and might be up to 1.0. This we capture with what Brachman and Levesque call a possibility interval of [.9, 1.0] - where the lower bound is called the belief in the proposition, and the second is the plausibility of it being true, say if all the uncertainty were resolved in favor of our hypothesis. We write the first term as $Bel(p)$, and the second as $Pl(p)$, and we can relate them with the formula

$$Bel(p) = 1 - Pl(\neg p)$$

i.e., the first indicates how certain we are about p , where the second indicates how high the probability could go given how certain we are that $\neg P$. In this example, we have no evidence that the car is still there, so the interval is [.9, 1.0].

Say we have another friend Bill, who also says that the car is missing. But we know that Bill only looks in the right spot 80% of the time. Thus, looking at just his evidence, we would have the possibility interval [.8, 1.0]. Now consider if both Sue and Bill tell me that the car is missing

- how might I combine this evidence?

¹⁸ Extracted from 53. Ardis, P. *Notes on Dempster Shafer Belief Functions*. [cited; Available from: <http://www.cs.rochester.edu/~ardis/DempsterShafer.pdf>].

ANNEXES

It is true that the probability that both are reliable is $.8 * .9 = .72$, but this isn't the number we are interested in, since only one of them has to be reliable for us to know that the car is missing. The probability that both Bill and Sue are unreliable is $.1 * .2 = .02$, so 98% of the time one of them is reliable and my car will be missing. Thus, my new degree of belief is .98, and the possibility interval is $[.98, 1.0]$. In this case, all my evidence was consistent, and as long as we believe the two pieces of evidence are independent, our conclusion seems justified.

What happens when evidence is inconsistent? Say Sue says the car is missing, and Bill says the car is there (but might have misrecognized it when he looked in the wrong spot). Thus, for Sue we have the possibility interval of $[.9, 1.0]$ as before, and for Bill we have $[0, .2]$. We have to consider four cases:

- Sue is reliable, Bill is reliable - impossible since they both can't be right!
- Sue is reliable, Bill is not, with probability $.9 * .2 = .18$. In this case, the car is gone.
- Bill is reliable, Sue is not, with probability $.1 * .8 = .08$. In this case the car is there.
- Both are unreliable, with probability $.2 * .1 = .02$. In this case, we do not know.

How do we convert this into a possibility interval? Essentially we normalize. Since we know at least one is unreliable, the sum of the three non-zero probabilities above should be 1, but the above calculations only give us $.18 + .08 + .02 = .28$. We fix this by normalizing, and thus we believe that the car is missing with $.18 / .28 = .643$ and that the car is not missing as $.08 / .28 = .286$. Thus the possibility interval for the car being missing is $[.643, 1 - .286]$, which equals $[.643, .714]$.

ANNEX D- Example of D. S. applied to aircraft detection¹⁹

Let's consider the following example from aircraft detection. The system has two sensors observing a target and the possible states are:

- detection of an F-111,
- detection of an F/A-18,
- detection of an P-3C Orion,
- “fast”, if we cannot distinguish between an F-111 and an F/A-18.
- “unknown”, where a decision as to what the aircraft is does not appear to be possible at all.

The sensors notice an aircraft and allocate the masses m_1 and m_2 as set in Table 2.

| | Sensor 1 | Sensor 2 | Fused masses |
|-------------|---------------|---------------|-------------------|
| Target type | (mass m_1) | (mass m_2) | (mass $m_{1,2}$) |
| F-111 | 30% | 40% | 55% |
| F/A-18 | 15% | 10% | 16% |
| P-3C | 3% | 2% | 0.4% |
| Fast | 42% | 45% | 29% |
| Unknown | 10% | 3% | 0.3% |
| Total mass | 100% | 100% | 100% |

Table 10- Mass assignments for the various aircraft²⁰

The third column holds the final masses fused using Dempster's rule of combination.

¹⁹ Extracted from: 3. Kessler, O. and F. White, *Data Fusion Perspectives and Its Role in Information Processing*, in *Handbook of Multisensor Data Fusion - Theory and Practice*, M.E. Liggins, D.L. Hall, and J. Llinas, Editors. 2009, CRC Press: Boca Raton.

²⁰ Dempster requires the totals equal to 100%.

ANNEXES

Exemplifying to the F-111:

$$m_{1,2}(F-111) \propto m_1(F-111) m_2(F-111) + m_1(F-111) m_2(\text{Fast}) + m_1(\text{Fast}) m_2(F-111)$$

$$m_2(\text{Unknown}) + m_1(\text{Fast}) m_2(F-111) + m_1(\text{Unknown}) m_2(F-111)$$

$$= 0.30 \times 0.40 + 0.30 \times 0.45 + 0.30 \times 0.03 + 0.42 \times 0.40 + 0.10 \times 0.40$$

$$= 0.47$$

The other masses were calculated on a similarly way. The third column was normalized by dividing each by their sum yields the final mass values.

The fusion reinforces the idea that the target is an F-111 and, together with our initial confidence in its being a fast aircraft, means that we are more sure than ever that it is not a P-3C. Even most of the mass is assigned to the two fast aircrafts, the amount of mass assigned to the “fast” type is not as high as we might expect.

ANNEX E- Example of Neural Networks applied to pattern recognition²¹

Pattern recognition can be implemented by using a feed-forward neural network that has been trained accordingly. During training, the network is trained to associate outputs with input patterns. When the network is used, it identifies the input pattern and tries to output the associated output pattern. The power of neural networks comes to life when a pattern that has no output associated with it, is given as an input. In this case, the network gives the output that corresponds to a taught input pattern that is least different from the given pattern.

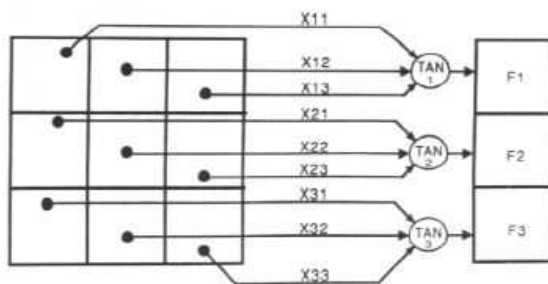


Figure 20- The Neural Network

For example:

The network of above figure is trained to recognize the patterns T and H. The associated patterns are all black and all white respectively as shown below.

²¹ Extracted from: 54. Stergiou, C. and D. Siganos. *NEURAL NETWORKS*. SURPRISE 1996 [cited; Available from: http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.html.

ANNEXES



If we represent black squares with 0 and white squares with 1 then the truth tables for the 3 neurons after generalization are;

| | | | | | | | | | |
|------|--|---|---|---|---|---|---|---|---|
| X11: | | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| X12: | | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| X13: | | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| | | | | | | | | | |
| OUT: | | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |

Top neuron

| | | | | | | | | | |
|------|--|---|-----|---|-----|-----|---|-----|---|
| X21: | | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| X22: | | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| X23: | | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| | | | | | | | | | |
| OUT: | | 1 | 0/1 | 1 | 0/1 | 0/1 | 0 | 0/1 | 0 |

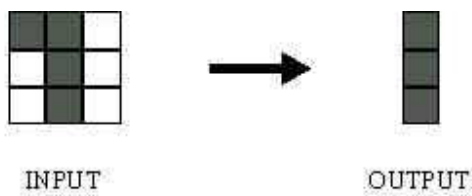
Middle neuron

ANNEXES

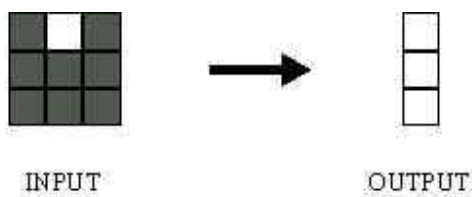
| | | | | | | | | | |
|------|--|---|---|---|---|---|---|---|---|
| X21: | | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| X22: | | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| X23: | | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| | | | | | | | | | |
| OUT: | | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 |

Bottom neuron

From the tables the following associations can be extracted:

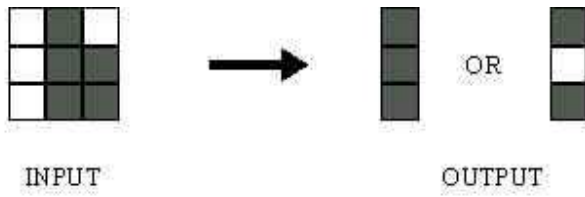


In this case, it is obvious that the output should be all blacks since the input pattern is almost the same as the 'T' pattern.



Here also, it is obvious that the output should be all whites since the input pattern is almost the same as the 'H' pattern.

ANNEXES



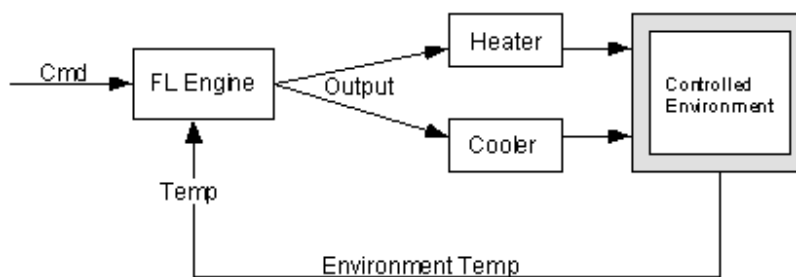
Here, the top row is 2 errors away from a T and 3 from an H. So the top output is black. The middle row is 1 error away from both T and H so the output is random. The bottom row is 1 error away from T and 2 away from H. Therefore the output is black. The total output of the network is still in favour of the T shape.

ANNEX F- Example of Fuzzy Logic applied to a temperature control device²²

On the following example we will try to exemplify the application of the four firsts steps.

1) Definition of the control objectives and criteria

We want to design a simple proportional temperature controller with an electric heating element and a variable-speed cooling fan. A positive signal output calls for 0-100 percent heat while a negative signal output calls for 0-100 percent cooling. Control is achieved through proper balance and control of these two active devices.



Cmd: Targettemperature
 Temp: Feedback Sensor in controlled environment
 Error: Cmd-Temp (+ = too cold, - = too hot)
 Error-dot: Time derivative of Error (+ = getting hotter, - = getting cooler)
 Output: HEAT or NO CHANGE or COOL

Figure 21 A simple block diagram of the control system

2) Determination of the input and output relationships

It is necessary to establish a meaningful system for representing the linguistic variables in the matrix. For this example, the following will be used:

"N" = "negative" error or error-dot input level

²² Extracted from 17. Kaehler, S.D. *Fuzzy Logic Tutorial - An Introduction*. 1998 [cited; Available from: <http://www.seattlerobotics.org/encoder/Mar98/fuz/flindex.html>].

ANNEXES

"Z" = "zero" error or error-dot input level

"P" = "positive" error or error-dot input level

"H" = "Heat" output response

"-" = "No Change" to current output

"C" = "Cool" output response

Define the minimum number of possible input product combinations and corresponding output response conclusions using these terms. For a three-by-three matrix with heating and cooling output responses, all nine rules will need to be defined. The conclusions to the rules with the linguistic variables associated with the output response for each rule are transferred to a matrix.

ERROR IN SIMPLE CONTROL SYSTEM

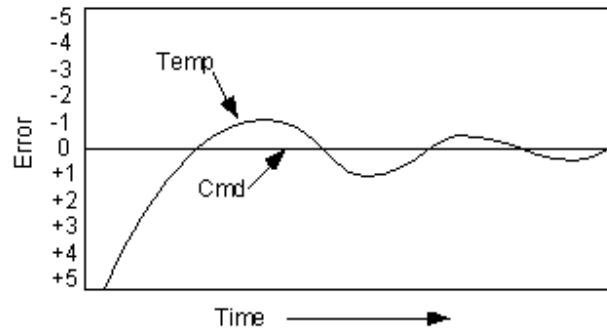


Figure 22- Typical control system response

Figure above shows what command and error look like in a typical control system relative to the command setpoint as the system hunts for stability. Definitions are also shown for this example.

ANNEXES

Definitions:

INPUT#1: ("Error", positive (P), zero (Z), negative (N))

INPUT#2: ("Error-dot", positive (P), zero (Z), negative (N))

CONCLUSION: ("Output", Heat (H), No Change (-), Cool (C))

INPUT#1 System Status

Error = Command-Feedback

P=Too cold, Z=Just right, N=Too hot

INPUT#2 System Status

Error-dot = $d(\text{Error})/dt$

P=Getting hotter Z=Not changing N=Getting colder

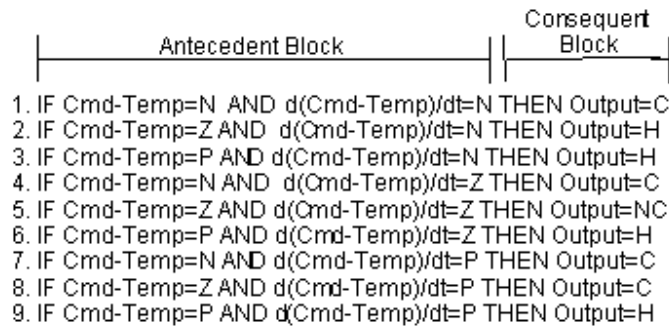
OUTPUT Conclusion & System Response

Output H = Call for heating NC = Do not change anything C = Call for cooling

3) Break the control problem down into FL rules

Linguistic rules describing the control system consist of two parts; an antecedent block (between the IF and THEN) and a consequent block (following THEN). By making this type of evaluation, usually done by an experienced operator, fewer rules can be evaluated, thus simplifying the processing logic and perhaps even improving the FL system performance.

ANNEXES



| | | Error -(Cmd-Temp) | | |
|------------------------------|---|-------------------|---------|--------|
| | | N | Z | P |
| Error-dot - (d(Cmd-Temp)/dt) | N | 1 C | 2 H | 3 H |
| | Z | 4 C | 5 NC | 6 H |
| | P | 7 C | 8 C | 9 H |

Figure 23- The rule structure & rule matrix

Additional degrees of error and error-dot may be included if the desired system response calls for this. This will increase the rule base size and complexity but may also increase the quality of the control. Figure above shows the rule matrix derived from the previous rules.

4) Creation of FL membership functions

The membership function is a graphical representation of the magnitude of participation of each input. It associates a weighting with each of the inputs that are processed, define functional overlap between inputs, and ultimately determines an output response. The rules use the input membership values as weighting factors to determine their influence on the fuzzy output sets of the final output conclusion. Once the functions are inferred, scaled, and combined, they are defuzzified into a crisp output which drives the system.

ANNEXES

There are different memberships functions associated with each input and output response.

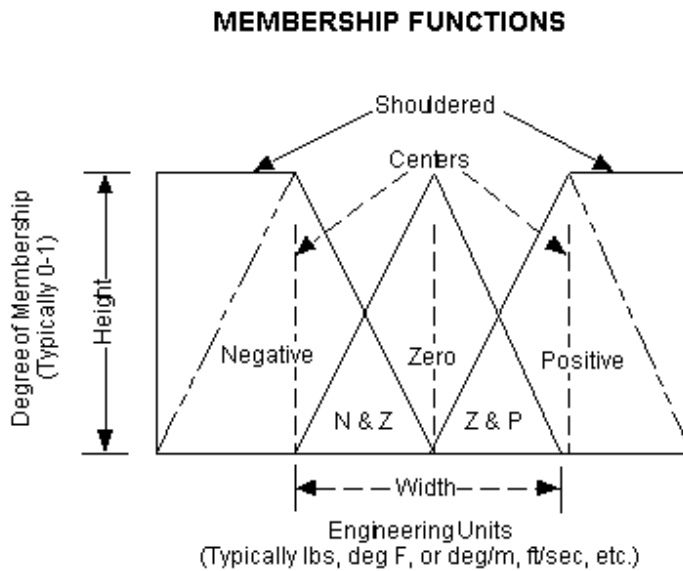


Figure 24- The features of a membership function

Figure above illustrates the features of the triangular membership function which is used in this example because of its mathematical simplicity. Other shapes can be used but the triangular shape lends itself to this illustration.

The degree of membership (DOM) is determined by plugging the selected input parameter (error or error-dot) into the horizontal axis and projecting vertically to the upper boundary of the membership function(s).

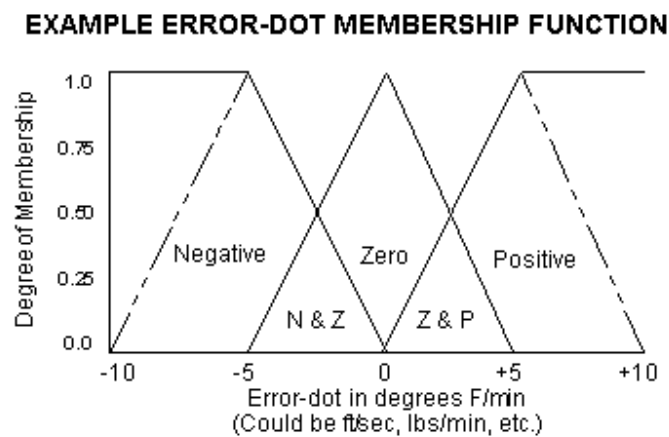
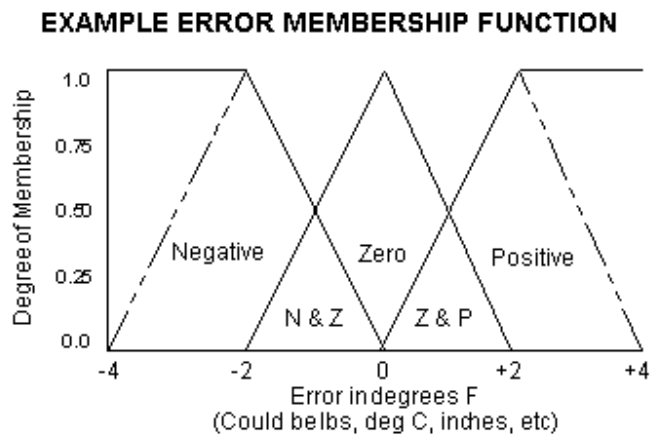


Figure 25- Example errors

In figure above, consider an "error" of -1.0 and an "error-dot" of +2.5. These particular input conditions indicate that the feedback has exceeded the command and is still increasing.

The degree of membership for an "error" of -1.0 projects up to the middle of the overlapping part of the "negative" and "zero" function so the result is "negative" membership = 0.5 and "zero" membership = 0.5. Only rules associated with "negative" & "zero" error will actually apply to the output response. This selects only the left and middle columns of the rule matrix.

For an "error-dot" of +2.5, a "zero" and "positive" membership of 0.5 is indicated. This selects the middle and bottom rows of the rule matrix. By overlaying the two regions of

ANNEXES

the rule matrix, it can be seen that only the rules in the 2-by-2 square in the lower left corner (rules 4,5,7,8) of the rules matrix will generate non-zero output conclusions. The others have a zero weighting due to the logical AND in the rules.

ANNEX G- Genetic Algorithms Uses²³

- **Genetic Algorithms in Parametric Design of Aircraft**, by Mark F. Bramlette and Eugene E. Bouchard. The authors discuss optimizing aircraft designs when the task is posed as that of optimizing a list of parameters. They have approached the problem with a number of optimization algorithms, including a genetic algorithm using real number representation. They also discuss the performance of each algorithm and describe some innovative techniques used in their quite successful genetic algorithm, including the technique of generating a large number of initial population members and then working only with the best ones.
- **Dynamic Anticipatory Routing in Circuit-Switched Telecommunications Networks**, by Louis Anthony Cox, Jr., Lawrence Davis, and Yuping Qiu. The objective of the study is to optimize the routing of telephone networks in order to minimize costs to US West. It compares the performance of an order-based genetic algorithm with several other optimization techniques on this problem. The authors conclude that the genetic algorithm is a highly successful technique when the problem is complex, but hybridization of these algorithms can lead to better performance than using any of them in isolation.
- **A Genetic Algorithm Applied to Robot Trajectory Generation**, by Yuval Davidor. He shows how to apply genetic algorithm techniques to the task of planning the path which a robot arm is to take in moving from one point to another. Davidor uses variable-length chromosomes in his solution, and devises some novel and interesting crossover operators.
- **Genetic Algorithms, Nonlinear Dynamical Systems, and Models of International Security**, by Stephanie Forrest and Gottfried Mayer-Kress, concerns a problem posed by current research in chaotic models of real processes.

²³ Source: 55. Alba, E. and C. Cotta. 1998 [cited; Available from: <http://www.lcc.uma.es/~ccottap/semEC/>].

- Chaotic models of international arms races and economic competition seem to model some features of the real-world processes better than some other more traditional models have done. The authors use a genetic algorithm to find good settings of the parameters of Mayer-Kress's models in order to enhance their performance on the models.
- **Strategy Acquisition with Genetic Algorithms**, by John J. Grefenstette. He experiments with SAMUEL, a genetic algorithm that learns techniques for maneuvering a simulated airplane in order to evade a simulated missile. The genetic algorithm he describes employs several techniques of interest, including variable-length chromosomes composed of rules that form a production system. A chromosome is evaluated by using those rules to maneuver the airplane in simulated interactions between airplanes and missiles. Grefenstette has built knowledge of the production rule domain into his operators in clever ways.
 - **Genetic Synthesis of Neural Network Architecture**, by Steven A. Harp and Tariq Samad, that describes techniques for encoding neural network architectures on binary chromosomes. The authors use variable-length chromosomes and a variety of other novel techniques. This is a good place to begin in learning how to combine neural networks and genetic algorithms.
 - **Air-Injected Hydrocyclone Optimization Via Genetic Algorithm**, by Charles L. Karr, that describes the solution of a design problem by a genetic algorithm using the bit string representation technique. Karr represents the design of an air-injected hydro cyclone as a list of parameters. An interesting feature of his approach is the use of a new operator called "simplex reproduction". Karr shows that a genetic algorithm using this operator is quite effective as a search technique for finding design parameter combinations.
 - **A Genetic Algorithm Approach to Multiple Fault Diagnosis**, by Gunar E. Liepens and W. D. Potter, which discusses the use of a genetic algorithm for finding the most plausible combination of causes for alarms in a microwave communication system. The authors use binary chromosomes to represent solutions to a problem that they show is a type of set covering problem. They

- show how to incorporate knowledge about set covering optimization into their genetic algorithm in novel ways, yielding a high-performance hybrid solution to the problem.
- **A Genetic Algorithm for Conformational Analysis of DNA**, by C. B. Lucasius, M. J. J. Blommers, L. M. C. Buydens, and G. Kateman. It is a development of a genetic algorithm for determining the structure of a sample of DNA based on spectrometric data about the sample. An interesting "cascaded" evaluation technique that greatly enhances the efficiency of their evaluation function is used. The authors use bit strings to encode molecular structures. Their evaluation function measures the degree to which each decoded structure conforms to the data that have been collected about the sample. The genetic algorithm evolves a description of molecular structure that is in agreement with the data collected. The problem of determining bimolecular structure occupies a central position in the worlds of fundamental and applied chemistry today.
 - **Automated Parameter Tuning for Sonar Information Processing**, by David J. Montana. An application of genetic algorithms to two problems associated with interpreting passive sonar data. The first is a parameterization problem. To solve it, Montana uses a floating-point version of OOGA to find good parameter settings for the algorithms employed in the process of interpreting sonar data. The second problem is a classification problem. For this problem, a genetic algorithm is used to train neural networks classifying sonar signals in various ways. In this second system, Montana and Davis experiment with a number of domain-based operators, including the use of back propagation -a neural network technique- as a genetic algorithm operator. This application is useful if you are interested in hybrid genetic algorithms, real number representations for parameterization, or neural networks.
 - **Interdigitation: A Hybrid Technique for Engineering Design Optimization**, by Gilbert Syswerda. An application of a genetic algorithm to the problem of scheduling activities in a laboratory in which each activity may affect the others in a variety of ways. Syswerda has been implementing this system under contract

ANNEXES

to the U. S. Navy. The genetic algorithm uses an order-based chromosome to represent its schedule. The chromosome is decoded with a decoder that incorporates a good deal of knowledge about the scheduling domain.

- **The Traveling Salesman and Sequence Scheduling: Quality Solutions Using Genetic Edge Recombination**, by Darrell Whitley, Timothy Starkweather, and Daniel Shaner. The authors describe a technique for solving the traveling salesman problem, a well-known combinatorial optimization problem. Their solution includes novel and ingenious representation, crossover, and repair mechanisms. They also show how similar techniques can be applied to the problem of scheduling a Hewlett-Packard production line.

ANNEX H- GA example applied to the Traveling Salesman problem²⁴

Traveling Salesman is a classical non deterministic problem where the aim is to find a tour of a given set of cities so that:

- each city is visited only once
- the total distance traveled is minimized

The chromosome representation is an ordered list of city numbers:

1) London 3) Dunedin 5) Beijing 7) Tokyo
2) Venice 4) Singapore 6) Phoenix 8) Victoria
CityList1 (3 5 7 2 1 6 4 8)
CityList2 (2 5 7 6 8 1 3 4)

Example of a mutation involving the reorder of the CityList2:

Before: (5 8 **7** 2 1 **6** 3 4)
After: (5 8 **6** 2 1 **7** 3 4)

On next two figures one can see the results of a GA applied to an example with 30 cities. These pictures evidence the algorithm's convergence from a solution with a total distance of 941 to an optimal one with just 420. The algorithm's performance is showed on the third figure.

²⁴ Extracted from 20. Williams, W. *Genetics Algorithms: A Tutorial*. [cited; Available from: <http://www.dbai.tuwien.ac.at/staff/musliu/ProblemSolvingAI/Class9GATutorial.ppt>.

ANNEXES

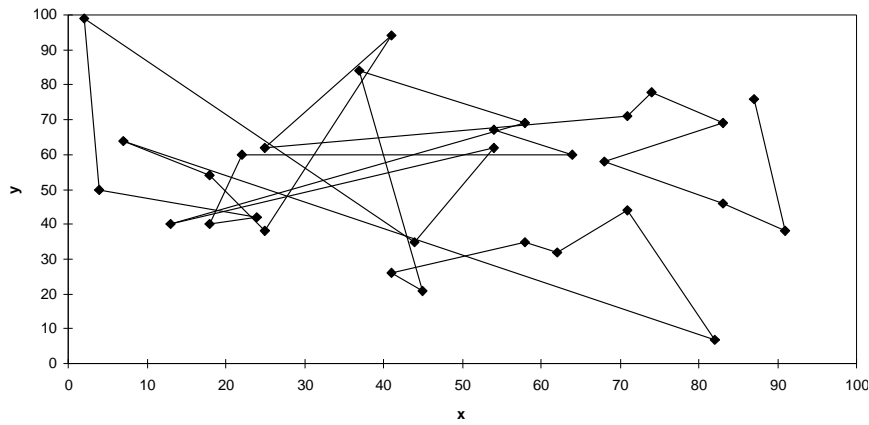


Figure 26- Early stage solution example

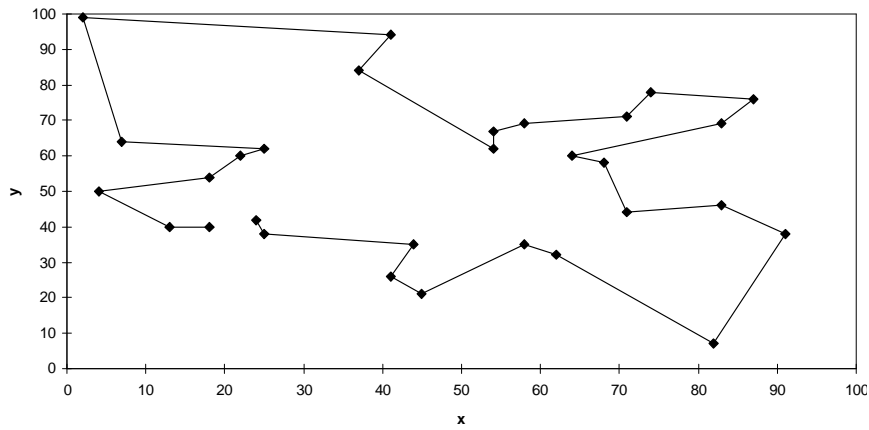


Figure 27- Optimal solution

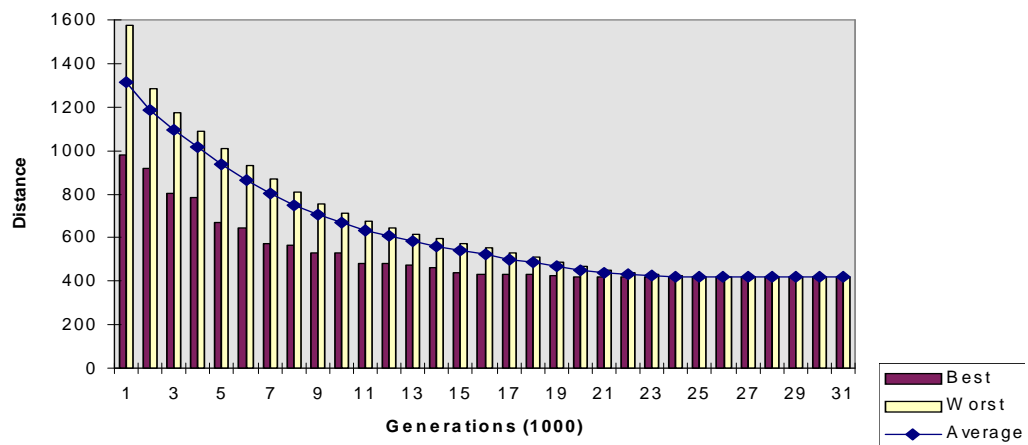


Figure 28- Overview of the GA Performance

ANNEX I- Forward chaining expert system example²⁵

In this inference type, rules are evaluated as new data enter the system. In case a rule evaluation becomes true, actions may add new data to memory, thus triggering more rules. And so on.

Consider a system with three rules:

If someone is a third year, then they need a job.

If someone is a third year, then they live in.

If someone needs a job, they will become an accountant.

And we put the following fact into database:

John is a third year.

Being a forward chain, the system is constantly on watch for new data. As soon as this data arrives, the system searches all the rules for any whose conditions weren't true before but are now. It then adds their conclusions to system.

In this case, rules 1 and 2 have conditions which match this new fact. So the system will immediately create and add the two facts below.

- John needs a job.
- John lives in.

²⁵ Extracted from 56. Paine, J. *Expert Systems*. 1996 [cited; Available from: <http://www.j-paine.org/students/lectures/lect3/node1.html>]

ANNEXES

These facts in turn can trigger rules. As each arrives, the system would look for yet more rules that are made true. In this case, the fact *John needs a job* would trigger rule 3, resulting in the addition of another fact to system:

John will become an accountant.

The fact *John lives in* would not trigger anything else though.

ANNEX J- Backward chaining expert system example²⁶

In a backward chaining the system needs to know the value of a piece of data. It searches for rules whose conclusions mention this data. Before it can use the rules, it must test their conditions. This may entail discovering the value of more pieces of data, and so on. This is also called goal-directed inference, or hypothesis driven, because inferences are not performed until the system is made to prove a particular goal (i.e. a question).

This is a lazy kind of inference. It does no work until absolutely necessary, in distinction to forward chaining, where the system eagerly awaits new facts and tries applying conditions as soon as they arrive.

So, if we had this knowledge base again

If someone is a third year, then they need a job.

If someone is a third year, then they live in.

If someone needs a job, they will apply to be an accountant.

and we were to add

John is a third year

the system would do nothing at all.

But if we were then to ask the question

Is there anyone who is going to become an accountant?

the system would try to answer. It would begin by searching either for a fact that gives the answer directly, or for a rule by which the answer could be inferred. To find such a rule, it searches the entire knowledge base for rules whose conclusions, if made true, will answer the question.

²⁶ Extracted from 56. Ibid. [cited.

ANNEXES

In this example, there are no facts giving the answer; there's one rule whose conclusion, if true, would supply an answer, and that's rule 3.

The system next checks the rule's conditions. Is there anyone who needs a job? As with the original question, we look either for a fact that answers directly, or for a rule. There are no facts, but rule 1 is relevant.

So we now check *its* conditions. Is there a third year? This time, there is a fact that answers this: *John is a third year*. So we've proved rule 1, and that's proved rule 3, and that's answered the question.

ANNEX L- Expert Systems uses²⁷

The seven main fields of applications:

Diagnosis and Troubleshooting of Devices and Systems of AI Kinds

This class comprises systems that deduce faults and suggest corrective actions for a malfunctioning device or process. Medical diagnosis was one of the first knowledge areas to which ES technology was applied (for example, see Shortliffe 1976), but diagnosis of engineered systems quickly surpassed medical diagnosis. There are probably more diagnostic applications of ES than any other type. The diagnostic problem can be stated in the abstract as: given the evidence presenting itself, what is the underlying problem/reason/cause?

Planning and Scheduling

Systems that fall into this class analyze a set of one or more potentially complex and interacting goals in order to determine a set of actions to achieve those goals, and/or provide a detailed temporal ordering of those actions, taking into account personnel, materiel, and other constraints. This class has great commercial potential, which has been recognized. Examples involve airline scheduling of flights, personnel, and gates; manufacturing job-shop scheduling; and manufacturing process planning.

Configuration of Manufactured Objects from Subassemblies

Configuration, whereby a solution to a problem is synthesized from a given set of elements related by a set of constraints, is historically one of the most important of expert system applications. Configuration applications were pioneered by computer companies

²⁷Source: 57. *THE APPLICATIONS OF EXPERT SYSTEMS*. [cited 2009 May]; Available from: http://www.wtec.org/loyola/kb/c1_s2.htm.

ANNEXES

as a means of facilitating the manufacture of semi-custom minicomputers (McDermott 1981). The technique has found its way into use in many different industries, for example, modular home building, manufacturing, and other problems involving complex engineering design and manufacturing.

Financial Decision Making

The financial services industry has been a vigorous user of expert system techniques. Advisory programs have been created to assist bankers in determining whether to make loans to businesses and individuals. Insurance companies have used expert systems to assess the risk presented by the customer and to determine a price for the insurance. A typical application in the financial markets is in foreign exchange trading.

Knowledge Publishing

This is a relatively new, but also potentially explosive area. The primary function of the expert system is to deliver knowledge that is relevant to the user's problem, in the context of the user's problem. The two most widely distributed expert systems in the world are in this category. The first is an advisor which counsels a user on appropriate grammatical usage in a text. The second is a tax advisor that accompanies a tax preparation program and advises the user on tax strategy, tactics, and individual tax policy.

Process Monitoring and Control

Systems falling in this class analyze real-time data from physical devices with the goal of noticing anomalies, predicting trends, and controlling for both optimality and failure correction. Examples of real-time systems that actively monitor processes can be found in the steel making and oil refining industries.

ANNEXES

Design and Manufacturing

These systems assist in the design of physical devices and processes, ranging from high-level conceptual design of abstract entities al. the way to factory floor configuration of manufacturing processes.

ANNEXES

ANNEX M- Conditional Probabilities Tables

bht/bhp = kte(C1)

| | |
|--------|-----|
| Good | 1.0 |
| Medium | 1.0 |
| Bad | 1.0 |

bhp > Min(C2)

| | |
|-----|-----|
| YES | 1.0 |
| NO | 1.0 |

bhp - whp > diffP(C3)

| | |
|-----|-----|
| Yes | 0.4 |
| No | 0.2 |

bhp/whp = kte(C4)

| | |
|--------|-----|
| Good | 1.0 |
| Medium | 1.0 |
| Bad | 1.0 |

age bht(A1)

| | |
|---------|-----|
| Neutral | 1.0 |
| Old | 1.0 |

age whp(A3)

| | |
|---------|-----|
| Neutral | 1.0 |
| Old | 1.0 |

age bhp(A2)

| | |
|---------|-----|
| Neutral | 1.0 |
| Old | 1.0 |

Sensor bht(S1)

| A1 | Neutral | | | | | | Old | | |
|--------|---------|--------|-----|--------|--------|-----|------|--------|-----|
| S2 | OK | | | NOT OK | | | OK | | |
| C1 | Good | Medium | Bad | Good | Medium | Bad | Good | Medium | Bad |
| OK | 1.0 | 0.4 | 0.0 | 0.0 | 0.6 | 1.0 | 95.0 | 35.0 | 0.0 |
| NOT OK | 0.0 | 0.6 | 1.0 | 1.0 | 0.4 | 0.0 | 5.0 | 65.0 | 1.0 |

| A1 | Old | | |
|--------|--------|--------|------|
| S2 | NOT OK | | |
| C1 | Good | Medium | Bad |
| OK | 0.0 | 55.0 | 95.0 |
| NOT OK | 1.0 | 45.0 | 5.0 |

ANNEXES

Sensor bhp(S2)

| | | | | | | | | | |
|--------|---------|--------|-----|--------|--------|-----|------|--------|-----|
| A2 | Neutral | | | | | | | | |
| C3 | Yes | | | | | | | | |
| C2 | YES | | | | | | | | |
| C4 | Good | | | Medium | | | Bad | | |
| C1 | Good | Medium | Bad | Good | Medium | Bad | Good | Medium | Bad |
| OK | 1.0 | 0.6 | 0.9 | 0.5 | 0.4 | 0.3 | 0.5 | 0.2 | 0.1 |
| NOT OK | 0.0 | 0.4 | 0.1 | 0.5 | 0.6 | 0.7 | 0.5 | 0.8 | 0.9 |

| | | | | | | | | | |
|--------|---------|--------|-----|--------|--------|-----|------|--------|-----|
| A2 | Neutral | | | | | | | | |
| C3 | Yes | | | | | | | | |
| C2 | NO | | | | | | | | |
| C4 | Good | | | Medium | | | Bad | | |
| C1 | Good | Medium | Bad | Good | Medium | Bad | Good | Medium | Bad |
| OK | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| NOT OK | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |

| | | | | | | | | | |
|--------|---------|--------|-----|--------|--------|-----|------|--------|-----|
| A2 | Neutral | | | | | | | | |
| C3 | No | | | | | | | | |
| C2 | YES | | | | | | | | |
| C4 | Good | | | Medium | | | Bad | | |
| C1 | Good | Medium | Bad | Good | Medium | Bad | Good | Medium | Bad |
| OK | 1.0 | 0.5 | 0.4 | 0.5 | 0.3 | 0.2 | 0.5 | 0.1 | 0.0 |
| NOT OK | 0.0 | 0.5 | 0.6 | 0.5 | 0.7 | 0.8 | 0.5 | 0.9 | 1.0 |

| | | | | | | | | | |
|--------|---------|--------|-----|--------|--------|-----|------|--------|-----|
| A2 | Neutral | | | | | | | | |
| C3 | No | | | | | | | | |
| C2 | NO | | | | | | | | |
| C4 | Good | | | Medium | | | Bad | | |
| C1 | Good | Medium | Bad | Good | Medium | Bad | Good | Medium | Bad |
| OK | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| NOT OK | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |

| | | | | | | | | | |
|--------|------|--------|------|--------|--------|------|------|--------|------|
| A2 | Old | | | | | | | | |
| C3 | Yes | | | | | | | | |
| C2 | YES | | | | | | | | |
| C4 | Good | | | Medium | | | Bad | | |
| C1 | Good | Medium | Bad | Good | Medium | Bad | Good | Medium | Bad |
| OK | 95.0 | 55.0 | 85.0 | 45.0 | 35.0 | 25.0 | 45.0 | 25.0 | 15.0 |
| NOT OK | 5.0 | 45.0 | 15.0 | 55.0 | 65.0 | 75.0 | 55.0 | 75.0 | 85.0 |

| | | | | | | | | | |
|--------|------|--------|-----|--------|--------|-----|------|--------|-----|
| A2 | Old | | | | | | | | |
| C3 | Yes | | | | | | | | |
| C2 | NO | | | | | | | | |
| C4 | Good | | | Medium | | | Bad | | |
| C1 | Good | Medium | Bad | Good | Medium | Bad | Good | Medium | Bad |
| OK | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| NOT OK | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |

| | | | | | | | | | |
|--------|------|--------|------|--------|--------|------|------|--------|-----|
| A2 | Old | | | | | | | | |
| C3 | No | | | | | | | | |
| C2 | YES | | | | | | | | |
| C4 | Good | | | Medium | | | Bad | | |
| C1 | Good | Medium | Bad | Good | Medium | Bad | Good | Medium | Bad |
| OK | 95.0 | 45.0 | 45.0 | 45.0 | 25.0 | 15.0 | 45.0 | 5.0 | 0.0 |
| NOT OK | 5.0 | 55.0 | 65.0 | 65.0 | 75.0 | 85.0 | 55.0 | 95.0 | 1.0 |

| | | | | | | | | | |
|--------|------|--------|-----|--------|--------|-----|------|--------|-----|
| A2 | Old | | | | | | | | |
| C3 | No | | | | | | | | |
| C2 | NO | | | | | | | | |
| C4 | Good | | | Medium | | | Bad | | |
| C1 | Good | Medium | Bad | Good | Medium | Bad | Good | Medium | Bad |
| OK | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| NOT OK | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |

ANNEXES

Sensor whp(S3)

| | | | | | | | | | |
|--------|---------|--------|-----|------|--------|-----|--------|--------|-----|
| A3 | Neutral | | | | | | | | |
| S2 | OK | | | | | | NOT OK | | |
| C3 | Yes | | | No | | | Yes | | |
| C4 | Good | Medium | Bad | Good | Medium | Bad | Good | Medium | Bad |
| OK | 1.0 | 0.3 | 0.1 | 0.3 | 0.2 | 0.0 | 1.0 | 0.6 | 0.7 |
| NOT OK | 0.0 | 0.7 | 0.8 | 0.7 | 0.8 | 1.0 | 0.0 | 0.4 | 0.3 |

| | | | | | | | | | |
|--------|---------|--------|-----|------|--------|-----|------|--------|-----|
| A3 | Neutral | | | Old | | | | | |
| S2 | NOT OK | | | OK | | | | | |
| C3 | No | | | Yes | | | No | | |
| C4 | Good | Medium | Bad | Good | Medium | Bad | Good | Medium | Bad |
| OK | 0.6 | 0.8 | 1.0 | 95.0 | 25.0 | 1.0 | 25.0 | 0.15 | 0.0 |
| NOT OK | 0.7 | 0.2 | 0.0 | 5.0 | 75.0 | 9.0 | 75.0 | 0.85 | 1.0 |

| | | | | | | |
|--------|--------|--------|------|------|--------|------|
| A3 | Old | | | | | |
| S2 | NOT OK | | | | | |
| C3 | Yes | | | No | | |
| C4 | Good | Medium | Bad | Good | Medium | Bad |
| OK | 95.0 | 55.0 | 65.0 | 55.0 | 75.0 | 95.0 |
| NOT OK | 5.0 | 45.0 | 35.0 | 75.0 | 25.0 | 5.0 |