# ENCODING FUZZY DIAGNOSIS RULES AS OPTIMISATION PROBLEMS

Antonio Sala, Alicia Esparza, Carlos Ariño and Jose V. Roig
*Systems Engineering and Control Dept., Univ. Politécnica de Valencia*
*Cno. Vera s/n, 46022 Valencia, Spain*

Keywords: Fault detection and diagnosis, fuzzy mathematical programming, approximate reasoning, optimisation.

Abstract: This paper discusses how to encode fuzzy knowledge bases for diagnostic tasks (i.e., list of symptoms produced by each fault, in linguistic terms described by fuzzy sets) as constrained optimisation problems. The proposed setting allows more flexibility than some fuzzy-logic inference rulebases in the specification of the diagnostic rules in a transparent, user-understandable way (in a first approximation, rules map to zeros and ones in a matrix), using widely-known techniques such as linear and quadratic programming.

## 1 INTRODUCTION

Many industrial activities depend on the correct operation of complex technological processes. A fault changes the behaviour of a system in such a way that it does no longer satisfy its nominal performance objectives or even the system functionality is lost. The objective of diagnosis (process monitoring) is estimating a vector of fault state parameters, $f$, from measurements of the outputs of a dynamic system, whose trajectories depend on $f$ and on time, initial conditions, external input variables, physical parameters, etc. Detecting faults with a gradation of *severity* from let's say zero to 100% may be advantageous in practice: detecting faults in its early stages enables corrective actions to be taken on time, if needed.

Different approaches to the problem appear in literature: data-based, knowledge-based or based on differential-equation analytical models. A description of many of these techniques appear in (Chiang et al., 2001; Blanke et al., 2003). The broadest conception of the problem may be set up in a probabilistic setting. In that situation, a comprehensive approach to the problem would involve estimation under nonlinear stochastic dynamics (Timmer, 2000; Khalil, 2002) as well as decision-theoretic criteria (Berger, 1985) apart, of course, of the consideration of robustness of the resulting results when subject to possibly significant modelling errors. The problem, as such, is quite complex and possibly intractable. Hence, simplified assumptions are often stated.

In quite a few practical cases, diagnostic-related knowledge is available from experts, who express it in linguistic terms ("fault F produces a pressure in pipe 3 *lower than normal*"). The meaning of some of those linguistic terms may be understood as rules involving fuzzy concepts defined on the numeric range of the physical variables being measured. This fact inspires the use of fuzzy logic in diagnosis (Angeli, 1999; Carrasco and *et. al.*, 2004). Other knowledge-based frameworks extend the basic logic reasoning schemes to uncertain reasoning (Kruse et al., 1991; Shafer and Pearl, 1990), possibilistic reasoning (Dubois and Prade, 2004; Yamada, 2004), Bayesian networks (Castillo et al., 1997; Russell and Norvig, 2003) or uncertain probability (Kyburg, 1988), or combine the approach with neural networks (Ayoubi and Isermann, 1997; Jie and Morris, 1996).

This paper presents an alternative approach: the expert knowledge stemming from some (partially simplified) properties of a nonlinear dynamic system is encoded as a constrained optimisation problem, transcoding fuzzy assertions as approximate linear equations in the linguistic domain. The idea of transforming fuzzy statements into equations also appears in (Juuso, 1999; Sala and Albertos, 2001) in a different context of system modelling.

The proposed approach seems to possesses significative advantages with respect to a classical fuzzy IF-THEN rulebase, particularly in multiple fault situations, while keeping the problem readable (reduced



Sala A., Esparza A., Ariño C. and V. Roig J. (2006).
ENCODING FUZZY DIAGNOSIS RULES AS OPTIMISATION PROBLEMS.
In *Proceedings of the Third International Conference on Informatics in Control, Automation and Robotics*, pages 34-39
DOI: 10.5220/0001203100340039

number of "rules") and computationally tractable (efficient off-the-shelf linear programming (LP) software exists able to deal with hundreds, even thousands, of constraints and variables, and quadratic programming (QP) routines are also widely available). LP is a widely known tool (Sierksma, 2001; Gass, 2003), taught in many undergraduate disciplines so that user understanding of both the rules and the inference tools implies that the approach might be useful in practical applications.

The structure of the paper is as follows: a preliminary section will justify some approximate additivity properties of systems based on linearisation. Section 3 will discuss how a fuzzy rulebase may be encoded as equations. Section 4 will discuss the available algorithms to solve them and how they should be modified for the problem in consideration. A conclusion section closes the paper.

## 2 PRELIMINARIES

**Dynamic systems.** Let us assume that a system to be diagnosed is governed by equations $\dot{x} = \psi(x, u, f, \theta, t)$ where $x$ is the system's dynamical state, $u$ is a set of known *input* variables, $f$ is the *failure* state to be estimated and $\theta$ are a set of system parameters, also assumed to be approximately known, and $t$ is the time variable (Khalil, 2002).

First, it will be assumed that the initial conditions $x(t_0)$ and inputs of standarised tests are known beforehand, as well as system-dependent parameters $\theta$. The result of such tests will be a finite collection of measurements at some time instants:

$$y = \psi(x(t_0), \theta, u, t_0, t, f)$$

where, by assumption, only $f$ is unknown. Let us denote as $q = (x(t_0), \theta, u, t_0, t)$ the set of known variables; they will be denoted as the experiment *context variables*. The diagnostic problem may then be cast as estimating $f$ from $y(q, f)$, i.e., obtaining the implicit function, if it exists $f = \gamma(q, y)$. In general, the analytic expression for $\gamma$ cannot be obtained except in the simplest of the cases. If enough sensors exist,and pre-classified data from the system ($y$,$q$,$f$) were available, a functional approximator such as a neural network might be used in order to try to learn $\gamma$ by example, or at least to learn the normal behaviour and generate some residuals in case of faults. Applications of the approach are reported in (Chow et al., 1993; Jie and Morris, 1996). However, learning for successful fault isolation in a complex system may require an impractical number of data. The approach will not be pursued further in this paper.

The diagnosis problem becomes easier if some simplifications and assumptions are made. Taking a Taylor series expansion of $\psi$ on the variable $f$, around $f = 0$, which will be defined as the "normal" situation, the result is an affine model:

$$y(q, f) = \psi(q, 0) + \frac{\partial \psi}{\partial f}(q, 0) * f + o(f^2) \qquad (1)$$

This Taylor series expansion justifies that at least approximately a linearity assumption holds:

$$y(q, f) \approx \psi(q) + C_y(q) * f \qquad (2)$$

where $C_y$ is a linear operator (a matrix whose elements depend on the known information $q$). Usually, a parameterised analytical model of the system is not available so $\psi(q)$ and $C_y(q)$ cannot be calculated. However, human operators possess knowledge that may be encoded in terms of fuzzy sets. Expressing such a knowledge in the form (2) is discussed in next section.

## 3 FUZZY DIAGNOSTIC MODELS

In the above discussed context, a collection of *fuzzy* sets mapping $y$ to the interval $[0, 1]$ are assumed to defined by an expert on the system to be diagnosed, $\mu : R^p \to [0, 1]^k$ where $p$ is the number of measurements and $k$ is the number of fuzzy concepts. Usually, those sets are denoted by user-defined linguistic labels.

Then, the expert also knows which effects each fault has in the measurements. This knowledge is usually expressed in terms of rules:

> If the isolated fault $F_i$ occurs, then abnormal symptoms $S_{i_1}, \ldots, S_{i_p}$ should be observed, being the rest normal

Those rules, in a fuzzy context, may be basically understood as

> If the *severity* of fault $F_i$ is $f_i$, $0 \le f_i \le 1$, and it is the only fault occurring in the system, then the intensity of symptoms $S_{i_1}, \ldots, S_{i_p}$ is approximately $f_i$, and the intensity of the rest of them is zero.

where $f_i = 0$ denotes fault not occurring and $f_i = 1$ denotes the fault occurring at a very significative severity level requiring user attention[1], and the intensity of the symptoms is the membership function of the suitably defined fuzzy concepts.

In a multiple-fault situation, the system is assumed to verify an expression such as (2). If the observed outputs are the fuzzified ones, it will be assumed that

[1] It is up to the user to fix a maximum value of $f_i$ (usually 1) in the optimisation procedures to be discussed, or to set $f_i = 1$ as a "landmark" point but considering higher values (more severe) possible.

the system verifies the following linear equation in the domain $[0,1]^p$ of logic values:

$$\mu_q(y) \approx C(q)f + D(q) \tag{3}$$

where $C$ and $D$ are a known function of the system's parameters, input variables and initial conditions of the experiment. The notation $\mu_q$ indicates that even the definition of the membership function may depend on *a priori* information, such as historical data, system-dependent parameters, etc. $C(q)$ will be similar to $\frac{\partial\mu}{\partial y}C_y(q)$. Note that a differentiable system indeed does verify such an equation[2] for small values of $f$ based in (2). The basic assumption here is that, with suitably defined fuzzy sets, the range in which such an equation fulfills is large enough to be useful for diagnosis.

Fuzzy sets may be defined on the *difference* between the observed readings and those in a normal situation, as an alternative to the *absolute* reference frame for concepts implicit in equation (3), so that:

$$\mu_q(y - D(q)) \approx C(q)f \tag{4}$$

In a practical situation, both types of fuzzy sets (3) or (4) may be used. In summary, the following linear equation must be solved in the diagnosis process:

$$\mu(y,q) = C(q)f \tag{5}$$

Other authors also pose matrix representations of the relationship between fuzzy faults and symptoms (for instance, (Yao and Yao, 2001) uses a fuzzy relation approach). The assumption of linearity in the logic domain, with suitable definitions of membership functions is also used, in a control context, in (Juuso, 1999).

## 3.1 Rule Encoding

The proposed expert rules are encoded in the format required in (3) in a simple way. Let us discuss several situations which will be clarified by examples.

First, the simplest setting would imply that, under a standarised test ($q$ is fixed for all diagnostic experiments), a fuzzy set denoting an "abnormal" situation is defined for every measured variable. In that case, the knowledge:

$F_i$ causes abnormality in $y_1, y_3, \ldots$

will define a *column* of matrix $C$, where elements at rows $1, 3, \ldots$ will be 1 and the rest of elements not explicitly enumerated at the above assertion will be set as zero. The usual fuzzy negation operator may be used if some variables have a fuzzy set defining the "normal" value of a variable.

[2]Formally, left and right derivatives may be needed, but details are not relevant.

**Example 1** Let us have an industrial boiler where a fault $f_1$ causes: no variation on a temperature measurement $t_1$, and increasing of temperature $t_2$. Another fault, $f_2$, causes increases in both temperatures. They both cause an increase of pressure $p$. Defining abnormally high temperatures with fuzzy sets denoted as "$T_1$ abnormal", $\mu_1$ and "$T_2$ abnormal", $\mu_2$, and "normal pressure" with another fuzzy set (such as a triangular one), $\mu_3$, the basic diagnosis equation would be:

$$\begin{pmatrix} \mu_1(t_1) \\ \mu_2(t_2) \\ 1-\mu_3(p) \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} f_1 \\ f_2 \end{pmatrix}$$

Alternatively, if the nominal normal pressure depends on a variable $q$, denoted as $P_0(q)$, (for instance, $q$ might be the load regime of the boiler), a fuzzy set may be defined on the pressure increment so that the diagnosis equation would be written as:

$$\begin{pmatrix} \mu_1(t_1) \\ \mu_2(t_2) \\ 1-\mu_3(p-P_0(q)) \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} f_1 \\ f_2 \end{pmatrix}$$

**Model errors and sensor faults.** Another situation takes into account the approximate nature of (5) allowing for an instrumental "fault" variable associated to each of the measurements. That instrumental fault variable encompasses both sensor faults and modelling errors (inaccuracy in the definition of the membership functions).

**Example 2** In the example being considered, the fault vector may be extended as:

$$\begin{pmatrix} \mu_1(t_1) \\ \mu_2(t_2) \\ 1-\mu_3(p) \end{pmatrix} = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} f_1 \\ f_2 \\ f_1^* \\ f_2^* \\ f_3^* \end{pmatrix} \tag{6}$$

where $f^*$ are three instrumental fault variables and $f$ denotes the "primary" faults.

In other situations, different faults have opposite effects on a particular variable so that its simultaneous occurrence does not deviate its measurements from the normal condition. The following example clarifies how to encode such a knowledge.

**Example 3** If, in example 1 fault 1 decreases T1, being the rest of symptoms the same as previously described, with concepts "abnormally high", $\mu_h$, and "abnormally low", $\mu_l$, defined for $T_1$, then the rulebase should be encoded as:

$$\begin{pmatrix} \mu_h(t_1) \\ \mu_l(t_1) \\ \mu_2(t_2) \\ 1-\mu_3(p) \end{pmatrix} = \begin{pmatrix} -1 & 1 \\ 1 & -1 \\ 1 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} f_1 \\ f_2 \end{pmatrix}$$

In the above example, depending on the units and membership definitions, the $-1$ terms may be a different negative number. Note that some combinations of faults may yield negative membership values; this is not a problem if inequality restrictions are considered, as discussed in section 4.1.

In any of the examples, a coefficient in matrix $C$ lower (higher) than 1 would indicate a milder (stronger) effect of the fault on the symptom, as defined by the membership function. Modifying the coefficients might be needed in a fine-tuning phase because not all the faults influence with the same intensity a particular variable. However, taking into account "sensor" errors $f^*$, an initial setting with mostly 0 and 1 coefficients in $C$ may be enough in order to achieve a reasonable output.

# 4 INFERENCE

A fault estimation $f$ is consistent with the observed symptoms if it is a solution of the basic diagnosis equation (5). Hence, fuzzy diagnosis amounts to finding the set of solutions of (5). However, some considerations need to be made: indeed, if sensor faults and modelling errors, $f^*$, are considered, the solution set includes *any* conceivable primary fault $f$, as sensor faults can accommodate any reading. For instance, in (6) a value of $f^*$ can be calculated for any sensor reading and any value of $f$. However, if sensor faults are not considered, with less faults than sensors the set of consistent faults will be usually empty, as (5) will have no solution due to modelling errors. So, the above idea must be refined for practical usability.

**Inference as optimisation.** As above discussed, sensor faults must be considered in practice. Then, (5) are equality restrictions and a criterion should be used in order to rank all the feasible solutions. Note also that inequality restrictions $f_i > 0$ implicitly apply, unless the user casts a meaning for negative fault severities.

A possible criterion to be chosen is minimising the norm of the "instrumental" fault component, *i.e.*, giving as the diagnosis solution the one that achieves less discrepancy between the measurements and the predictions $C(q)f$. The discrepancy $\mu - C(q)f$ will be denoted as *inference error*.

If the chosen norm is the Euclidean one, fuzzy inference is equivalent to a least squares problem. Let us consider an equation $\mu = Cf$ where sensor faults are also members of $f$. This is a linear system of equations with more unknowns than equations, which can be solved in the following sense:

The feasible solution $f$ that minimises the squared Euclidean norm of $Wf$, where $W$ is a diagonal weight matrix, is given by the pseudo-inverse formula (Meyer, 2001):

$$f = W^{-2}C^T(CW^{-2}C^T)^{-1}\mu \tag{7}$$

In a practical setting, a high penalisation in $W$ must be specified for the sensor fault components $f^*$. For invertibility of $W$, small positive weights in the primary faults need to be introduced[3].

If the chosen norm is the 1-norm of $Wf$ (sum of absolute value of the components), then inference can be carried out in a linear programming setting. The LP framework needs to introduce dummy variables for positive and negative sensor errors $f^* = f^+ - f^-$, $f^+ > 0, f^- > 0$, to calculate the 1-norm as the sum of $f^+ + f^-$. This change of variables is standard in LP textbooks.

Note that LS algorithms produce an "intermediate" point as a result (not a vertex of the feasible region), sharing the error between all the equations, as small errors are not significant (because of the squaring) so LS tries to reduce big errors. On the contrary, LP produce a result in a vertex of the feasible solution space, and increments from either small or big errors weight the same.

## 4.1 Constrained Optimisation

Under the proposed settings, it is implicitly assumed that a reasonable diagnostic should verify $f_i \geq 0$ in all components of the primary faults.

Also, fuzzy concepts saturate in $[0, 1]$. Hence, simultaneous faults yielding the same symptom cannot fulfill, for instance, $1 = f_1 + f_2$ if they are fully active. However, two easy options are available (or a combination of them) in that case:

- When a fuzzified sensor reading is saturated, replace the equality constraint in the diagnosis equation by an inequality ($1 \leq C^i f$, $0 \geq C^i f$, where $C^i$ denotes the $i$-th row of $C$).
- Translate an ordered fuzzy partition on a domain into numerical values (for instance, $\{$*very low, low, normal, high, very high*$\}$ into $\{-2, -1, 0, 1, 2\}$) in the spirit of the so-called linguistic equation (Juuso, 1999; Jarvensivu et al., 2001). In this way, the basic diagnosis equation (5) may involve sensor values ranging more than $[0, 1]$, but somehow keeping the linguistic meaning.

**Algorithms.** LP algorithms incorporate linear inequality restrictions seamlessly. However, the least squares formula (7) must then be discarded and quadratic programming (QP) routines used instead.

[3] Solving $y = Cf$ by standard least squares, $f = (C^TC)^{-1}C^T\mu$, is equivalent to the proposed approach when the primary-fault weights tend to zero and sensor ones are equal to the same constant.

For instance, if the sensor reading in example (4) had been $\{.2, .18, 0\}$ the output of the LS formula would have been $f_1 = 0.197$, $f_2 = -0.003$, out of the constraint space so $QP$ or non-negative least squares would have been needed.

As the involved restrictions are linear, efficient code exists for both LP and QP settings. For brevity, mostly linear programming settings will be considered in the sequel, although a similar version posed as QP would produce comparable results. For instance, commercially available LP software is able to efficiently deal with hundreds (even thousands) of variables and restrictions, allowing for large-scale implementation of the ideas in this work.

**Binary faults.** Some faults may be only either 0 or 1, without intermediate values (for instance, circuit breaker ON vs. OFF). To carry out optimisation, explicit enumeration of all the involved binary (or integer) variables and solving for each of them the optimisation on the remaining real variables may be an approach. Alternatively, mixed linear integer programming or branch-and-bound methodologies may also be applied (Sierksma, 2001).

**Managing non-unique solutions.** The optimisation routines stop at an approximately optimal point. However, there might be other optimal points or, at least, which may have a very similar value of the cost index. The situation is particularly frequent in the case of missing measurements, which amounts to deleting the corresponding row of $C$.

In order to choose between possible nonunique solutions, the weights of the different fault components should penalise each fault according to its probability: in that way, the solution would tend to be the most likely fault consistent with the sensor measurements.

**Imprecise measurements.** Some sensors may be imprecise, in the sense that a small deviation from the expected values is frequent and acceptable to assume when producing a diagnosis. In a sense, quadratic cost indices naturally take that fact into account, but LP settings need a straightforward change of variable to do that[4]. This is easily carried out, by expressing each "sensor fault" by a sum of two sub-faults, bounding the maximum value of one of them (with a small weight in the inference cost index) and setting a much larger penalisation on the deviations of the non-limited one. If the small sensor errors have zero weight, the setting is equivalent to interval measurements:

$$f^* = f_1^+ + f_2^+ - f_1^- - f_2^- \quad (8)$$

$$0 \leq f_1^+ \leq l^+,\ 0 \leq f_1^- \leq l^-,\ 0 \leq f_2^+,\ 0 \leq f_2^- \quad (9)$$

Heavy cost in $f_2^+$ and $f_2^-$, no penalisation in $f_1^+, f_1^-$ results in an interval sensor reading allowing, with no cost, diagnostics involving the actual reading, say $\sigma$, plus or minus the desired bound: $[\sigma - l^-, \sigma + l^+]$.

[4]The same change of variable may be used in QP formulations to fine-tune the cost index formula, if so wished.

**Example 4** Let us consider a knowledge base:

Fault 1 produces S1, S2. Fault 2 produces S2, S3

and a fuzzified sensor reading were $\{0.2, 0.43, 0.15\}$. The setting for minimal squared inference error (weighting by 0.5 the "confidence" on the accuracy of equation 2, because the addition of individual faults may not result in the exact addition of the results due to possible system nonlinearity) would be carried out by the following Matlab code:

```
C=[1 0;1 1;0 1];C2=[C eye(3)];
j=diag([0.01 0.01 1 .5 1]);
 cw=C2*inv(j);
 f=inv(j)*pinv(cw)*[.2 .43 .15]'
```

The result is $f_1 = 0.21$, $f_2 = 0.16$.

Alternatively, the Matlab code for the above problem minimising the 1-norm of the inference error via linear programming is:

```
 C=[1 0;1 1;0 1];
C2=[C eye(3) -eye(3)];
 j=[0.01 0.01 1 .5 1 1 .5 1];
x=linprog(j,[],[],C2,
      [.2 .43.15],zeros(8,1));
```

which produces $f_1 = 0.2$, $f_2 = 0.15$, apportioning all error to the less reliable sensor 2. Vector $j$ contains the weights for $f_1$, $f_2$ and the positive and negative components of the 3 possible sensor/modelling faults. The Matlab manual explains the meaning of the arguments to `linprog`.

## 5 CONCLUSIONS

This paper presents a methodology for approximately translating expert diagnostic knowledge into mathematical programming problems (constrained optimisation). The knowledge is a series of statements about the list of symptoms caused by the occurrence of a particular fault, expressed via linguistic statements involving fuzzy sets.

The approach deals naturally with multiple faults approximately following a linear equation in the linguistic domain. The diagnostic procedure operates satisfactorily with missing measurements or a limited number of faulty sensors.

The proposed approach can be thought of as an intermediate between IF-THEN rulebases and diagnosis based on a full mathematical model. It keeps a linguistic interpretation while allowing for many combinations of requirements difficult to be taken into account in a pure logic framework. Marginal possible intervals of fault severities are also easily calculated in the case of non-unique solutions of the optimisation problem. The problem is computationally tractable even in a large-scale framework as efficient software exists for the proposed optimisation techniques.

The presented framework has been discussed on a theoretical level with simple academic examples. Detailed comparative analysis and application to realistic diagnostic environments with a large number of "rules" is under research at this moment.

## REFERENCES


Angeli, C. (1999). Online expert system for fault diagnosis in hydraulic systems. *Expert Systems*, 16:115–120.

Ayoubi, M. and Isermann, R. (1997). Neuro-fuzzy systems for diagnosis. *Fuzzy Sets and Systems*, 89(3):289–307.

Berger, J. (1985). *Statistical Decision Theory and Bayesian Analysis*. Springer-Verlag, London.

Blanke, M., Kinnaert, M., Lunze, J., and Staroswiecki, M., editors (2003). *Diagnosis and Fault-Tolerant Control*. Springer, London.

Carrasco, E. and *et. al.*, J. R. (2004). Diagnosis of acidification states in an anaerobic wastewater treatment plant using a fuzzy-based expert system. *Control Engineering Practice*, 12(1):59–64.

Castillo, E., Gutierrez, J., and Hadi, A. (1997). *Expert Systems and Probabilistic Network Models*. Springer, London.

Chiang, L., Russell, E., and Braatz, R. (2001). *Fault Detection and Diagnosis in Industrial Systems*. Springer-Verlag.

Chow, M., Sharpe, R., and Hung, J. (1993). On the application and design consideration of artificial neural network fault detectors. *IEEE Trans. Industrial Electronics*, 40:181–198.

Dubois, D. and Prade, H. (2004). Possibilistic logic: a retrospective and prospective view. *Fuzzy Sets and Systems*, 144(1):3–23.

Gass, S. (2003). *Linear Programming: methods and applications*. Dover, 5th edition.

Jarvensivu, M., Juuso, E., and Ahavac, O. (2001). Intelligent control of a rotary kiln fired with producer gas generated from biomass. *Eng. Applic. Artif. Intelligence*, 14:629–653.

Jie, Z. and Morris, J. (1996). Process modelling and fault diagnosis using fuzzy neural networks. *Fuzzy Sets and Systems*, 79(1):127–140.

Juuso, E. (1999). Fuzzy control in process industry: the linguistic equation approach. In Verbruggen, H., Zimmermann, H.-J., and Babuska, R., editors, *Fuzzy Algorithms for Control*, pages 243–300. Kluwer, Boston.

Khalil, H. (2002). *Nonlinear Systems*. Prentice Hall, third edition.

Kruse, R., Schwecke, E., and Heinsohn, J., editors (1991). *Uncertainty and vagueness in Knowledge Based Systems: Numerical Methods (Artificial Intelligence)*. Springer-Verlag.

Kyburg, H. (1988). Higher order probabilities and intervals. *Int. J. Approximate Reasoning*, 2:195–209.

Meyer, C. (2001). *Matrix Analysis and Applied Linear Algebra*. Society for Industrial & Applied Mathematics (SIAM).

Russell, S. and Norvig, P. (2003). *Artificial Intelligence: a modern approach*. Prentice-Hall, 2nd edition.

Sala, A. and Albertos, P. (2001). Inference error minimisation: Fuzzy modelling of ambiguous functions. *Fuzzy Sets and Systems*, 121(1):95–111.

Shafer, G. and Pearl, J., editors (1990). *Readings in uncertain reasoning*. Morgan Kauffman, San Mateo (CA), USA.

Sierksma, G. (2001). *Linear and Integer Programming: Theory and Practice*. Marcel Dekker Pub., New York, 2nd edition.

Timmer, J. (2000). Parameter estimation in nonlinear stochastic differential equations. *Chaos, Solitons and Fractals*, 11(15):2571–2578.

Yamada, K. (2004). Diagnosis under compound effects and multiple causes by means of the conditional causal possibility approach. *Fuzzy Sets and Systems*, 145:183–212.

Yao, J. and Yao, J. (2001). Fuzzy decision making for medical diagnosis based on fuzzy number and compositional rule of inference. *Fuzzy Sets and Systems*, 120:351–366.