Fuzzy Justification of Heuristic Methods in Inverse Problems and in Numerical Computations, with Applications to Detection of Business Cycles From Fuzzy and Intuitionistic Fuzzy Data

Krassimir T. Atanassov[1], Vladik Kreinovich[2],
Hung T. Nguyen[3], and Berlin Wu[4]

[1]CLBME - Bulgarian Academy of Sciences
Sofia-1113, P.O.Box 12, Bulgaria
emails krat@bgcict.acad.bg, krat@bgearn.acad.bg

[2]Department of Computer Science
University of Texas at El Paso
El Paso, TX 79968, USA, vladik@cs.utep.edu

[3]Department of Mathematical Sciences
New Mexico State University
Las Cruces, NM 88003, USA, hunguyen@nmsu.edu

[4]Department of Applied Mathematics
National Chengchi University
Taipei, Taiwan, China, berlin@math.nccu.edu.tw

1 First Case Study: Inverse Problems

Inverse problems: brief reminder. One of the main problems of data processing is the *inverse problem*: we observe the signal y after it has passed through some medium, and we want to reconstruct the original signal x (i.e., we want to reconstruct the original image x from the results y of astronomical measurements).

When the original signal is strong enough, i.e., when the signal-to-noise ratio is high, the signal arrives practically un-changed ($y \approx x$), so reconstruction is rather easy (we can even take the observed signal y as a reasonably good approximation to original signal x).

The inverse problem becomes complex when the original signal x is weak, i.e., when the signal-to-noise ratio is small. In this case, we can expand the dependence of y on x into a Taylor series and neglect quadratic and higher order terms in this expansion. Thus, we have a linear dependence $y = Ax + b$ (for some matrix A and vector b), and we get a *linear* equation that we have to solve in order to reconstruct x:

$$Ax = c,$$

where $c = y - b$, y is measured, A and b are known, and x has to be determined. For example, when we reconstruct how a single x depends on time $x = x(t)$, we get a linear integral equation

$$\int A(t, t') \cdot x(t')\,\mathrm{d}t = c(t),$$

where $c(t) = y(t) - b(t)$, $y(t)$ is a measured signal, and $b(t)$ is a known correction. If we want to reconstruct an *image* x, i.e., the dependence of brightness x on *two* spatial coordinates x_1 and x_2 ($x(x_1, x_2)$), then we must solve a 2D integral equation

$$\int A(x_1, x_2, x_1', x_2') \cdot x(x_1', x_2')\,\mathrm{d}x_1 \mathrm{d}x_2 = c(x_1, x_2),$$

where $c(x_1, x_2) = y(x_1, x_2) - b(x_1, x_2)$, $y(x_1, x_2)$ is a measured (distorted) image, and $b(x_1, x_2)$ is a known correction.

Inverse problems are often ill-defined. In principle, it is well know how to solve linear equations. However, linear equations which appear in inverse problems are often difficult to solve because they are *ill-defined*: a small change in c can lead to a drastic change in the solution x (see, e.g., [6]). The reason for this is that often, the distortions in signal transmission *smoothen* the signal. As a result, e.g., the value $y(t)$ of the measured signal at a moment t depends not only on the value $x(t)$ of the original signal at the same moment of time, but also on the values in the nearby moments of time (similar, the image gets blurred). In the simplest case, when

$$y(t) = \frac{1}{2T} \cdot \int_{-T}^{T} x(t - s)\,ds,$$

one can easily explain how smoothing leads to ill-definedness: if we add a high-frequency component $\Delta x(t) = c \cdot \sin(\omega \cdot t)$ to the original signal $x(t)$, then the measured signal $y(t)$ gets changed by $\Delta y(t) \approx c/\omega$, i.e., for large ω, by a really small value. Thus, adding this very small value Δy to y may lead to a drastic change $\Delta x \approx c$ in $x(t)$, which is exactly what we mean by saying that the inverse problem is ill-defined.

A typical practical example of an ill-defined problem is an *interpolation* problem: we know the values $x(t_1)$, $x(t_2)$, $\ldots$, $x(t_n)$ of some quantity x at different moments $t_1, \ldots, x_n$, and we want to find the values $x(t)$ for *all* moments of time t. One possible extrapolation method is to find a polynomial which passes through all these points. Such a polynomial (called *Lagrange polynomial*) always exists, but it is well known that a small change in the values $x(t_i)$ can lead to a huge change in $x(t)$ for $t \neq t_i$.

In mathematical terms, "ill-defined" means that the matrix A is almost degenerate, some of its eigenvalues are very small, and as a result, the inverse matrix A^{-1} (for which $x = A^{-1}c$) has some very large eigenvalues.

From the practical viewpoint, solving ill-defined problems is difficult, because all measurements (including measurements of y) are imprecise, and small measurement inaccuracy in y can lead to a huge inaccuracy in the reconstructed signal x.

Regularization: a brief reminder. One of the main methods of solving ill-defined problems is to *regularize* these problems, i.e., e.g., to replace the original equation

$$Ax = b \tag{1}$$

by a *regularized* equation

$$Ax + \alpha \cdot x = b \tag{2}$$

for some small real number $\alpha > 0$ (see, e.g., [6]).

This replacement indeed makes the problem less ill-defined:

- As we have just mentioned, in matrix terms, ill-defined means that the eigenvalues of the inverse matrix A^{-1} can be arbitrarily large, because the eigenvalues of the matrix A can be very close to 0.

- On the other hand, e.g., when a matrix A is non-negatively defined, i.e., when all its eigenvalues are non-negative, then the eigenvalues of the new matrix $A' = A + \alpha$ are all greater than or equal to $\alpha > 0$ (i.e., cannot be too close to 0), and so all the eigenvalues of the inverse matrix $(A')^{-1}$ are bounded from above by α^{-1} (and cannot, therefore, be arbitrarily large).

How to choose a regularization parameter? The larger α, the smaller the effect of ill-definedness, i.e., the smaller the effect of the errors in y on the reconstructed value x. Hence, from the viewpoint of reducing ill-definedness, we should take α as large as possible.

However, we cannot take α to be too large, because then, the coefficients (A and $A + \alpha$) in the equations (1) and (2) become drastically different and therefore, the solution of the equation (2) becomes drastically different from the desired solution of the equation (1). So:

- we cannot choose a very small α, because then the problem will still be ill defined, and
- we cannot choose a very large α, because then the solution of the regularized problem will be too much distorted in comparison with the original signal.

In short, a choice of the parameter α can seriously affect the quality of the reconstructed signal and therefore, the problem of choosing the optimal value of α is one of the main problems of regularization.

We want to find α for which the reconstruction is the most adequate, i.e., for which the *reconstruction error*, i.e., the difference $\Delta x = \widetilde{x} - x$ between the reconstructed signal $\widetilde{x}$ and the actual signal x is the smallest possible. Therefore, to find such optimal α, let us estimate the error Δx.

Estimating the reconstruction error. Let us first consider the simplified situation, in which we know the original signal x. Then, we should get $y = Ax + b$; however, in reality, we get due to measurement errors Δy, we get a slightly different value $\widetilde{y} = Ax + b + \Delta y$. Therefore, the right-hand side $\widetilde{c} = \widetilde{y} - b$ of the equations (1) and (2) takes the form $\widetilde{c} = Ax + \Delta y$. Substituting this right-hand side into the equation (2), we get

$$\widetilde{x} = (A + \alpha)^{-1}(Ax + \Delta y).$$

Thus, the difference $\widetilde{x} - x$ between the reconstructed and the actual signal x can be represented as a sum:

$$\Delta x = \Delta_1 + \Delta_2, \tag{3}$$

where

$$\Delta_1 = (A + \alpha)^{-1}(Ax) - x, \tag{4}$$

and

$$\Delta_2 = (A + \alpha)^{-1}\Delta y. \tag{5}$$

In reality, we do not know x, but usually, we can assume that the reconstructed signal $\widetilde{x}$ is a good approximation to the actual signal x. Therefore, we can replace x by $\widetilde{x}$ in the formulas (4) and get a reasonable estimate

$$\Delta_1 = (A + \alpha)^{-1}(A\widetilde{x}) - \widetilde{x}. \tag{4a}$$

The resulting expression (3), (4a), (5) means that the reconstruction error Δx consists of two components:

- the first component Δ_1 is caused by the fact that, since $\alpha \neq 0$, the equation (2) which we are solving is somewhat different from the original equation (1); the larger α, the larger this difference, and therefore, the larger this error component Δ_1;

- the second component Δ_2 is caused by the measurement error Δy; when $\alpha = 0$, this error can be huge; the larger α, the smaller this error component.

Fuzzy approach to choosing α. We want the resulting error to be as small as possible. Therefore, we want the first error component Δ_1 to be small *and* the second error component Δ_2 to be small.

For the first component, by "small" we mean that some reasonable norm $\delta_1 = \|\Delta_1\|$ of the vector Δ_1 should be small: e.g., the largest possible component of the vector Δ_1, or the means square (i.e., l^2-norm) of this vector.

The second component Δ_2 depends on the random error Δy, so its norm $\|\Delta_2\|$ will also depend on this (unknown) error. For this component, when we say "small", we mean that, e.g., a *mathematical expectation* $\delta_2 = E[\|\Delta_2\|]$ of this norm should be small.

The smaller both components, the better. However, we cannot simply claim that we want δ_1 to be the smallest possible *and* that we want δ_2 to be the smallest possible, because δ_1 is the smallest (equal to 0) when $\alpha = 0$, but the second component is the smallest when $\alpha = \infty$. So, we cannot directly formulate the problem of choosing the best α as a *crisp* optimization problem. Instead, let us formulate it as a *fuzzy* problem (for basic notions of fuzzy logic, see, e.g., [1, 5]).

We want to find the value α for which the property "δ_1 is small *and* δ_2 is small" is satisfied to the largest extent, i.e., for which the degree of satisfying (of truth, of membership) for this property is the largest possible. This degree can be obtained from the degrees of truth that δ_1 is small and that δ_2 is small by using a t-norm (a fuzzy analogue of "and"). Let us use the simplest possible t-norm $a\&b = \min(a, b)$. Then, we are looking for the value α for which

$$\min(\mu_{\rm small}(\delta_1), \mu_{\rm small}(\delta_2)) \to \min_{\alpha}, \tag{6}$$

where $\mu_{\rm small}(z)$ is a membership function which describes the fuzzy term "small". Clearly, $\mu_{\rm small}(0) = 1$, and the function $\mu_{\rm small}(z)$ monotonely (and continuously) decreases as z increases.

Since the equation (6) uses the membership function for "small", it may seem, at first glance, that the resulting choice of α should be different for different membership functions, so we would have to select a specific membership function to make a selection of α. Interestingly, it turns out that the optimal value of α does not depend on the choice of the membership function at all:

Proposition 1. *For the optimization problem* (6)*, the optimal value of α is the value for which $\delta_1(\alpha) = \delta_2(\alpha)$.*

Comment. In other words, the optimal choice of α is when the error components are equal to each other. This choice has been earlier proposed as a successful heuristic (see, e.g., [7]). Our Proposition shows that fuzzy logic can provide a justification for this heuristic.

The very fact that fuzzy logic can provide a justification for a crisp heuristic should not be surprising: other examples of this type (and even examples related to inverse problems) are given in [2, 3, 4]).

Proof of Proposition 1. The proof of this Proposition is based on the following Lemma:

Lemma 1. *Let* $f(\alpha) = \min(g_1(\alpha), g_2(\alpha))$*, where:*

- $g_1(\alpha) \geq 0$ *is a continuous strictly increasing function for which* $g_1(0) = 0$*, and*
- $g_2(\alpha) \geq 0$ *is a continuous strictly decreasing function for which* $g_2(\alpha) \to 0$ *as* $\alpha \to \infty$*.*

Then, the function $f(\alpha)$ *attains its maximum at a single point in which* $g_1(\alpha) = g_2(\alpha)$*.*

Proof of Lemma 1. Since both functions $g_1(\alpha)$ and $g_2(\alpha)$ are continuous, their difference $d(\alpha) = g_1(\alpha) - g_2(\alpha)$ is also continuous.

- When $\alpha = 0$, we have $g_1(\alpha) = 0$ and $g_2(0) > 0$ (since g_2 is strictly decreasing and tends to 0 at ∞), so $d(0) = g_1(0) - g_2(0) < 0$.
- When $\alpha \to \infty$, we have $g_1(\alpha) > 0$ and increasing and $g_2(\alpha) \to 0$, so $d(\alpha) > 0$ for sufficiently large α.

The continuous function $d(\alpha)$ passes from a negative to a positive value, and therefore, it has to attain the 0 value somewhere; hence, there exists a value α_0 for which $d(\alpha_0) = g_1(\alpha_0) - g_2(\alpha_0) = 0$ and $g_1(\alpha_0) = g_2(\alpha_0)$.

Let us show that the function $f(\alpha)$ attains its maximum for $\alpha = \alpha_0$, and that this is the only maximum point of $f(\alpha)$. In other words, we need to show that if $\alpha \neq \alpha_0$, then $f(\alpha) < f(\alpha_0)$. We will prove this inequality by considering two possible cases: $\alpha < \alpha_0$ and $\alpha > \alpha_0$.

- If $\alpha < \alpha_0$, then, due to monotonicity of the functions g_i, we have

  $$g_1(\alpha) < g_1(\alpha_0) \text{ and } g_2(\alpha_0) < g_2(\alpha).$$

  By the choice of α_0, we have

  $$g_1(\alpha) < g_1(\alpha_0) = g_2(\alpha_0) < g_2(\alpha).$$

  Therefore,

  $$f(\alpha) = \min(g_1(\alpha), g_2(\alpha)) = g_1(\alpha) < g_1(\alpha_0) = f(\alpha_0).$$

  In other words, if $\alpha < \alpha_0$, then $f(\alpha) < f(\alpha_0)$.

- If $\alpha > \alpha_0$, then, due to monotonicity of the functions g_i, we have

$$g_1(\alpha) > g_1(\alpha_0) \text{ and } g_2(\alpha_0) > g_2(\alpha).$$

By the choice of α_0, we have

$$g_1(\alpha) > g_1(\alpha_0) = g_2(\alpha_0) > g_2(\alpha).$$

Therefore,

$$f(\alpha) = \min(g_1(\alpha), g_2(\alpha)) = g_2(\alpha) < g_2(\alpha_0) = f(\alpha_0).$$

In other words, if $\alpha > \alpha_0$, then $f(\alpha) < f(\alpha_0)$.

In both cases, we have the desired inequality, and thus, the lemma is proven.

The proposition follows from the Lemma because of the assumed monotonicity and continuity of the membership function $\mu_{\text{small}}(z)$ and the assumed monotonicity of the errors δ_1 and δ_2.

2 Second Case Study: Numerical Computations

Many numerical methods are iterative; often, these methods have a proven convergence. However, the convergence proofs are based on the idealized situation when all the real numbers are represented in the computer *precisely*, and all required elementary operations with real numbers are performed *exactly*. In this idealized situation, the more iterations we perform, the more accurate the results.

In reality, however, computers have *finite accuracy*: numbers are represented with only finitely many digits, and operations are also only approximate. As a result, rounding errors accumulate. So, with each iteration:

- on one hand, we *decrease* the error (due to the computational (process), but

- on the other hand, we *increase* the error due to roundings.

As we go from iteration to iteration, the first error decreases (and tends to 0), while the rounding error accumulates and thus increases. As a result, after sufficiently many iterations, the rounding error becomes the main error component, and the solution worsens. To get a good solution, it is, therefore, important to know when to stop the iterations.

This situation is similar to the previous one; namely, on each iteration n, the total error consists of two components:

- the component $\delta_1(n)$ which describes the ideal difference between the current approximation and the desired solution, and
- the rounding error component $\delta_2(n)$.

As n increases, the first component decreases, while the second one increases (because rounding errors accumulate). We want *both* components to be small. In other words, we want to find n for which the degree to which both components are small is the largest possible:

$$\min(\mu_{\text{small}}(\delta_1(n)), \mu_{\text{small}}(\delta_2(n))) \to \min_n. \tag{7}$$

This problem is similar to the problem (6), with the only exception that the optimized parameter n is now a *discrete* parameter. Its solution is also similar, except that in the continuous case, the optimal value α was when errors were exactly equal ($\delta_1(\alpha) = \delta_2(\alpha)$), while in the discrete case, since the parameter n is discrete, we may no longer have the exact equality $\delta_1(n) = \delta_2(n)$; it turns out, however, that the solution is attained when these values are as close as possible:

Proposition 2. *For the optimization problem (7), there exists a unique value n_0 for which $\delta_1(n_0) \le \delta_2(n_0)$ and $\delta_1(n_0+1) \ge \delta_2(n_0+1)$; the optimal value of n is either n_0 or n_0+1.*

Comment. In other words, the optimal choice of n is when the error components are "almost" equal to each other. This choice has been earlier proposed as a successful heuristic: stop iterations when the rounding error becomes of the same order as the solution error. Our Proposition shows that fuzzy logic can provide a justification for this heuristic.

Proof of Proposition 2. The proof of this Proposition is based on the following Lemma:

Lemma 2. *Let $f(n) = \min(g_1(n), g_2(n))$, where:*

- *$g_1(n) \ge 0$ is a strictly increasing sequence for which $g_1(0) = 0$, and*
- *$g_2(\alpha) \ge 0$ is a strictly decreasing sequence for which $g_2(n) \to 0$ as $n \to \infty$.*

Then, there exists a value n_0 for which $g_1(n_0) \le g_2(n_0)$ and $g_1(n_0+1) \ge g_2(n_0+1)$, and the function $f(n)$ attains its maximum at one of the points n_0 or n_0+1.

Proof of Lemma 2. Similarly to the proof of Lemma 1, let us consider the difference $d(n) = g_1(n) - g_2(n)$.

- When $n = 0$, we have $g_1(0) = 0$ and $g_2(0) > 0$ (since g_2 is strictly decreasing and tends to 0 at ∞), so $d(0) = g_1(0) - g_2(0) < 0$.
- When $n \to \infty$, we have $g_1(n) > 0$ and increasing and $g_2(n) \to 0$, so $d(n) > 0$ for sufficiently large n.

Let us show that there exists an n_0 for which $d(n_0) \le 0$ and $d(n_0+1) \ge 0$. We will prove it by reduction to a contradiction. Suppose that such an n_0 does not exist. This means that for every n_0, if $d(n_0) \le 0$, then $d(n_0+1)$ cannot be non-negative, i.e., we must have $d(n_0+1) < 0$. Since $d(0) < 0$, we can thus conclude that $d(0+1) = d(1) < 0$, $d(1+1) = d(2) < 0$, and by induction, that $d(n) < 0$ for all n, which contradicts to the fact that $d(n) > 0$ for some n. This contradiction shows that our initial assumption was wrong, and the desired n_0 exists. In other words, there exist an n_0 for which $g_1(n_0) \le g_2(n_0)$ and $g_1(n_0+1) \ge g_2(n_0+1)$.

Let us now show that the function $f(n)$ attains its maximum either for $n = n_0$ or for $n = n_0+1$. In other words, we will show that if $n \ne n_0$ and $n \ne n_0+1$, then $f(n) < f(n_0)$ or $f(n) < f(n_0+1)$ (and in both cases, $f(n) < \max(f(n_0), f(n_0+1)))$. We will prove this inequality by considering two possible cases: $n < n_0$ and $n > n_0+1$.

- If $n < n_0$, then, due to monotonicity of the functions g_i, we have
  $$g_1(n) < g_1(n_0) \text{ and } g_2(n_0) < g_2(n).$$
  By the choice of n_0, we have
  $$g_1(n) < g_1(n_0) \le g_2(n_0) < g_2(n).$$
  Therefore,
  $$f(n) = \min(g_1(n), g_2(n)) = g_1(n) < g_1(n_0) = f(n_0).$$
  In other words, if $n < n_0$, then $f(n) < f(n_0)$.
- If $n > n_0+1$, then, due to monotonicity of the functions g_i, we have
  $$g_1(n) > g_1(n_0+1) \text{ and } g_2(n_0+1) < g_2(n).$$
  By the choice of n_0, we have
  $$g_1(n) > g_1(n_0+1) \ge g_2(n_0+1) > g_2(n).$$
  Therefore,
  $$f(n) = \min(g_1(n), g_2(n)) = g_2(n) < g_2(n_0+1) = f(n_0+1).$$
  In other words, if $n > n_0+1$, then $f(n) < f(n_0+1)$.

In both cases, we have the desired inequality, and thus, the lemma is proven.

The proposition follows from the Lemma because of the assumed monotonicity and continuity of the membership function $\mu_{\rm small}(z)$ and the assumed monotonicity of the errors $\delta_1(n)$ and $\delta_2(n)$. Q.E.D.

Comment. A similar result can be applied to *discrete* regularization techniques: e.g., in interpolation problem, instead of considerating a polynomial which fits *all* the data points, we may want to consider a polynomial of a fixed order n. The optimal choice of n can then be obtained by using Proposition 2.

3 Application to Detection of Business Cycles: An Example Related to Intuitionistic Fuzzy Sets

Economy is changing in cycles: a growth period is followed by recession, and recession changes back to growth. It is extremely important to be able to predict the future economic behavior, and for this prediction, we must collect the statistics of the previous cycles. However, transitions are gradual, and it is therefore very difficult to find out when exactly growth changes into recession and vice versa. Both terms are not precisely defined, they express the expert's opinion and can be, therefore, best described by *fuzzy* sets.

Let us assume that we are analyzing a transition between growth and recession. We start with a year (let us denote it by 0) of clear growth; we know that at some following year T, we have a clear recession. We want to find the year when the change occured, i.e., a year c which was a growth year, while the next year was a recession.

For each year n from 0 to T, we have a degree of belief $g(n)$ that year n was a growth year, and a degree of belief $r(n)$ that year n was a recession year. Typically, the transition is rather monotonic, so we can assume that the values $r(n)$ strictly increase from 0 to 1, while the values of $g(n)$ strictly decrease from 1 to 0. We are interested in finding a *change point*, i.e., a year n which was a growth year, while the next one was a recession year. The degree of belief that n is a change point is equal to $\min(g(n), r(n+1))$. Thus, we arrive at the following problem:

$$\min(g(n)), r(n+1)) \to \min_{n:\ 0 \le n < T}. \tag{8}$$

This problem is similar to the problem (7), with the only exception that n only goes from 0 to $T-1$. Its solution is also similar (with a similar proof):

Proposition 3. *For the optimization problem* (8)*, there exists a unique value* n_0 *for which* $g(n_0) \ge r(n_0+1)$ *and* $g(n_0+1) \le r(n_0+1)$*; the optimal value of* n *is either* n_0 *or* n_0+1.

Comments.

- The application to Taiwan business cycle was given in [9] (see also [8]).
- This example is closely related with *intutionistic fuzzy sets.* Indeed, strictly speaking, recession is a negation ot growth, so if we would strictly follow the standard fuzzy set methodology, in which negation is described by the operation $z \to 1-z$, we would have to take $r(n) = 1 - g(n)$. In practice, however, if an expert is not sure that n was a growth year, it does not necessarily mean that he believes that n was a recession year. It is also possible that he thinks that there is not enough information to

decide whether this year was a recession or a growth year. In other words, for every year n, instead of *single* degree of confidence $g(n)$ corresponding to growth, we have it two different degrees of confidence:

– the *positive* degree of confidence $g(n)$, that n was a growth year, and

– the *negative* degree of confidence $r(n)$, that n was a recession year.

Since believing in growth means not believing in recession, we have $r(n) \leq 1 - g(n)$. Such a pair is called an *intuitionistic fuzzy set.*

Acknowledgments. This work was supported in part by NASA under cooperative agreement NCC5-209 and grant No. NAG 9-757, by NSF grants No. EEC-9322370 and DUE-9750858, by United Space Alliance, grant No. NAS 9-20000 (PWO C0C67713A6), by Future Aerospace Science and Technology Program (FAST) Center for Structural Integrity of Aerospace Systems, effort sponsored by the Air Force Office of Scientific Research, Air Force Materiel Command, USAF, under grant number F49620-95-1-0518, and by the National Security Agency under Grant No. MDA904-98-1-0564.

The authors are thankful to Gerhard Heindl and to Gennady N. Solopchenko for valuable discussions.

References

[1] G. Klir and B. Yuan, *Fuzzy sets and fuzzy logic: theory and applications.* Prentice Hall, Upper Saddle River, NJ, 1995.

[2] V. Kreinovich, C.-C. Chang, L. Reznik, and G. N. Solopchenko, "Inverse problems: fuzzy representation of uncertainty generates a regularization", In: *Proceedings of NAFIPS'92: North American Fuzzy Information Processing Society Conference, Puerto Vallarta, Mexico, December 15–17, 1992*, NASA Johnson Space Center, Houston, TX, 1992, pp. 418–426.

[3] H. T. Nguyen, V. Kreinovich, and B. Bouchon-Meunier, "Soft Computing Explains Heuristic Numerical Methods in Data Processing and in Logic Programming", *Working Notes of the AAAI Symposium on Frontiers in Soft Computing and Decision Systems*, Boston, MA, November 8–10, 1997, pp. 40–45.

[4] H. T. Nguyen, V. Kreinovich, and B. Bouchon-Meunier, "Soft Computing Explains Heuristic Numerical Methods in Data Processing and in Logic Programming", *Heuristics: The Journal of Intelligent Technology*, 1998 (to appear).

[5] H. T. Nguyen and E. A. Walker, *A first course in fuzzy logic,* CRC Press, Boca Raton, Florida, 1997.

[6] A. N. Tikhonov and V. Y. Arsenin. *Solutions of ill-posed problems,* V. H. Winston & Sons, Wasnington, DC, 1977.

[7] N. A. Zheludeva, V. Kreinovich, and G. N. Solopchenko, *Applicability criteria and accuracy estimates for heuristic methods of solving inverse problems*, Leningrad Center of New Technology "Informatika", Technical Report, Leningrad, 1989 (in Russian).

[8] B. Wu and S.-L. Hung, "Pattern recognition for nonlinear time series: with examples for distinguishing ARCH and bilinear models", *Fuzzy Sets and Systems*, 1998 (to appear).

[9] B. Wu and M.-C. Liaw, *Application of fuzzy time series analysis to change period detection*, Technical Report, Department of Mathematical Sciences and Statistics, National Chengchi University, Taipei, Taiwan, China, 1998.