Autonomous Creation of Process Cause and Effect Relationships: Metrics for Evaluation of the Goodness of Linguistic Rules

Nitin Sharma, R. Russell Rhinehart

Abstract— **Autonomous knowledge discovery requires metrics of rule base quality. This work recommends metrics useful for selecting good cause-and-effect rules from naturally occurring dynamic data such as found in historian databases in the Chemical Process Industry.**

I. INTRODUCTION

The Chemical Process Industry (CPI) uses models (knowledge) to guide and automate process management decisions (control). When derived from first principles and/or from laboratory and pilot plant experiments, ideal knowledge is of less than ideal use for real processes.

Process operators eventually derive linguistic relationships from observation and intuition. Stephanopoulos and Han [1] described this learning as "a cognitive skill of experienced process operators that fits the current facts about the process and enables the operators to assess process behavior and predict the effects of possible control actions". Oshima [2] described deficiencies in such knowledge stating that misjudgment on behalf of operators can arise from to a wrong impression of what is going on in the process.

Expert Systems and Fuzzy Logic Controllers make use of linguistic information in form of IF/THEN statements. Fuzzy Logic Controllers (FLC) are applied in: robotics (Collewet, et al. [3]), automated vehicles (Ko and Chen [4]) and process control (Rhinehart [5]). Expert systems in the CPI control, monitor (Cimander, et al. [6]), and understand process behaviors. Other applications are operator training, and planning and scheduling of operations and maintenance (Fonseca and Knapp [7]), especially after failure or abnormal running conditions.

Expert Systems made by polling information from experienced personnel are never complete. Such rules pertain only to critical or obvious trends, and often do not incorporate knowledge of events of lesser significance. Finally, knowledge collected from experts is usually in the form of static rules loosely related to the real numerical world (Kordon [8]). This rigid, non-adapting rule base does not include temporal behavior making it inadequate for complete supervisory control.

Extracting complete and dynamic knowledge is confounded by temporal characteristics of process plants. Due to lag or delay the present value of the controlled variable is not the effect of the current conditions of the system. This delay is usually variable - the result of the hydrodynamic characteristics of processes. Another temporal characteristic that must be incorporated is persistence of any event. The mixing nature of a process tends to diminish the effect of any non-persistent event that may occur, and a learning algorithm overlooking persistence may incorrectly interpret this event as a rule.

The work presented here is part of a larger project, which aims to describe an algorithm that will extract useful, complete, linguistic, temporal and mechanistic cause-and-effect rules from numerical data, such as from historian databases of process plants. Information extracted from historical data could be expressed in simple linguistic terms very similar to human understanding of the process. For example: *IF condition1 AND condition2 OR condition3 AND condition4…THEN after certain delay control variable will be such.* The statement between the IF and the THEN conjunction is the antecedent and the statement after the THEN conjunction is the consequent.

Rules can be created, by choosing/assembling different antecedents, conjunctions and consequents in a variety of ways. However, the common aspect of any rule creation algorithm or approach is the validation of rules created. To autonomously find correct rules and to create optimized rule bases shall require metrics (numerical measures) that evaluate certain desirable characteristics of a linguistic rule. This paper concentrates on recommending the right metrics, which prescribe correct, complete and mechanistic temporal linguistic rules.

II. PREVIOUS WORK

Goodness metrics can be classified into two characteristics: measures that evaluate the fitness of individual rules (Local strategies) and those that evaluate the fitness of the entire rule base (Global strategies).

GA based technologies that use local strategies include the work by Ngan, et al. [9]. The final fitness function is a weighted function of the above-mentioned metrics. Using a single fitness function is not advised when multiple objectives are being addressed because this reduces the distinguishing features. In the current work we propose a multi-objective approach for evaluation of rules.

Another example of the local approach is by Herrera, et al. [10]. Instead of simply counting the number of observations within data a "compatibility" degree was used which is calculated by taking a norm of the membership values of the antecedent and consequent parts of the rule, evaluated at every example in the dataset, using this compatibility degree various metrics are calculated. Measuring the degree to which each dataset affects a rule is less susceptible to the variation in amount of data available and measures the quality of the rule itself. Earlier, Chen and Black [11] also used a degree of the rule to decide between conflicting rules that had the same antecedent but different consequent parts. Degree was defined as the product of the individual membership values of the antecedent and consequent parts. Although both these approaches are better than simple counting, one should not combine the degree of the antecedent and consequent parts since each degree explains different features of a rules and more information can be extracted by using these separately.

An example of a global strategy is the work by Xiong and Litz [12], who introduce a consistency index of the entire fuzzy rule base, a function of the number of observations of conflicting rules in the rule base. A conflict is defined as two rules having the same antecedent but different consequents. This index is then combined with the performance of the fuzzy logic controller itself to form the global fitness function. An earlier work by Wong and Fan [13] evaluates the rule base by evaluating the performance of the FLC. Traditional performance measures such as Rise Time, Overshoot and Integral of Absolute Error are used to optimize successive rule populations in a genetic scheme. These are evaluated from a step response of the system being controlled. Also, it is usually not feasible to run control tests on already commissioned processes. This might be practical with simple mechanical objects (as shown in the example in the paper), or if we have successful detailed models of the process.

III. METHODOLOGY

The current work introduces a general strategy, based on the "truth space diagram" of a rule, to evaluate the multiple measures of goodness of linguistic rules. The objective is to find metrics that provide information on both the qualitative and the quantitative relationships between the data and the rule base.

To find the best (out of many possible) metrics for rule evaluation, a case study approach was used. First, process data from a simulator (See Figure 1(a)) was generated.
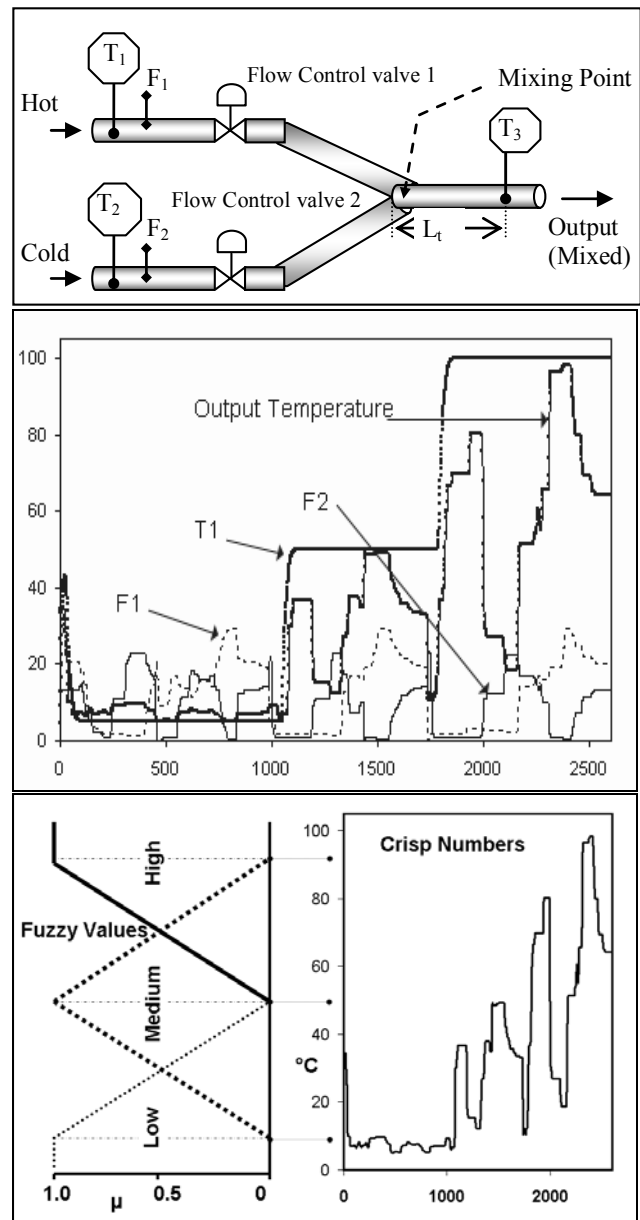

Fig. 1. a) Hot and Cold water simulator, b) Input-Output Variables, c) Fuzzy classification of Temperature with Limits at 10, 50 and 90 °C.

Then a rule base containing all possible rules (all possible combinations of antecedent and consequent parts) was created, designed metrics were calculated and the

initial rule base (using these metrics) was optimized using these metrics to find the optimal rule base. This final rule base was inspected by the human operator and the best combination of metrics decided based on the quality and compactness of the rule base. The approach followed the following broad steps.

A. Data Generation and Processing

To illustrate the concept, data was acquired from a Hot and Cold water simulator. The simulation incorporates real world dynamics such as transport and measurement delays and is capable of adding deviations such as measurement bias and process drifts that have an ARMA stochastic behavior, noise and valve "sticktion". This is a simple example, but incorporates behaviors which are representative of a majority of unit operations within the CPI. The simulation was nonlinear, had multiple inputs, and the dynamics of the response temperature (such as delay) depended upon operating conditions. For the purpose of generating data, three input variables were manipulated and the effect on one output variable was monitored (See Figure 1(b) for transient response). The manipulated input variables were temperature of hot water stream ($0 \leq T1 \leq 100\,^{\circ}$C), flow rate of hot water stream ($0 \leq F1 \leq 30$ Kg/min) and flow rate of cold water stream ($0 \leq F2 \leq 30$ Kg/min). The output variable was temperature of the mixed stream ($0 \leq T3 \leq 100\,^{\circ}$C). This input-output data was then processed in the following manner: 1) The output data was "un-delayed" by shifting data backwards into three categories – short, medium and long delay. 2) The crisp input-output data was fuzzified. Each variable was classified into three fuzzy categories – low, medium and high, using triangular membership functions:

$$\mu_x^{i,j} = \frac{a_j - x_i}{a_j - b_j}; \text{ j=1 to 3, i = 1 to n;}$$

Where x_i is the crisp numerical value of the ith input or output variable, is the fuzzy membership value of x_i in the jth fuzzy category, a_j and b_j are the fuzzy set break points for category j and n is the maximum number of datasets in the input-output data. Figure 1 (c) illustrates fuzzy classification of one variable into three fuzzy categories. To incorporate temporal information, two unique features were included into the linguistic rules. The persistence of an event was incorporated into the antecedent of the rule while the resulting delay was incorporated into the consequent of the rule.

B. Initial Rule base and Truth Space Diagrams (TSD)

The structure of the rule base resulting from the above step was as follows:

IF T1 is L/M/H **AND** F1 is L/M/H **AND** F2 is L/M/H **AND** Persistence is L/M/H …**THEN** after L/M/H delay T3 is L/M/H; Where L/M/H is Low, Medium or High. The total number of possible rules will be = (no. of fuzzy classes) [(no. of variables)] = 3^6 = 729. Hence initial rule base will have 729 rules consisting of the various combinations of the linguistic categories of each variable in the antecedent and consequent parts. The algorithm generates the statement of the rules programmatically and performs the following steps according to the statement of the rule:

Step 1: According to the rule statement R_l (where $1 \leq l \leq 729$) the persistence of the event expressed by the antecedent is calculated. The persistence of the event stated is the minimum persistence (or length of time an event persists) of any of the three input parts of the rule antecedent, i.e. *T1* is high, *F1* is low etc.

Step 2: For $1 \leq i \leq n$ calculate Truth of the antecedent and Truth of the consequent for each rule statement R_l. Truth of any statement is defined as the degree of membership of any data set or example to the linguistic terms in that statement.

$$Ta_{i,l} = \left(\mu_{T1}^{i,j} \times \mu_{F1}^{i,j} \times \mu_{F2}^{i,j} \times \mu_{Persistence}^{i,j}\right)^{\frac{1}{4}}$$
$$Tc_{i,l} = \mu_{T3}^{i,j}$$

Step 3: The truth space is a two-dimensional space bounded by the truth of the antecedent and truth of the consequent parts of a linguistic rule. The truth space hence is bounded by the region {T: $0 \leq T \leq 1$, T = truth of antecedent/consequent}, where a truth equal to 0 means absolute false and truth of 1 means absolute truth. The truth space diagram also represents a one-to-one mapping of the dataset from the real (numerical) space to a new (truth) space defined by the linguistic statement of the rule. The space can be divided into four quadrants and each quadrant provides different information about the linguistic rule.

C. Numerical Metrics and Rule base optimization

Each dataset is represented by a point on the TSD, and the location of the point demonstrates the membership of the dataset to the linguistic statement of the rule. For example, consider point A in Figure 2. The values for Ta_i and Tc_i are high for this dataset, i.e. what the antecedent states the consequent follows or the cause and effect match according to the rule statement. This reveals that the information expressed in the linguistic rule is contained in the numerical data from the simulator. Hence many points in Quadrant II of the TSD would reflect the validity of the rule. Consequently, points in Quadrant IV would show that the rule statement was false, i.e. what the antecedent of the rule expressed did not match what the consequent said, for example point B in Figure 3.
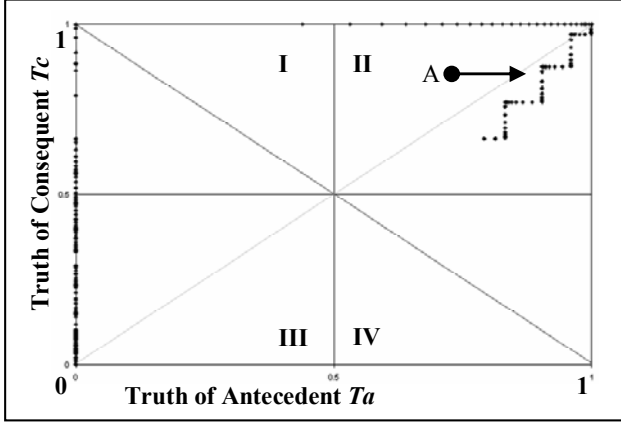
Fig. 2. TSD for R480: IF T1 is HIGH & F1 is HIGH & F2 is LOW & Persistence is HIGH THEN after MED delay T3 will be HIGH
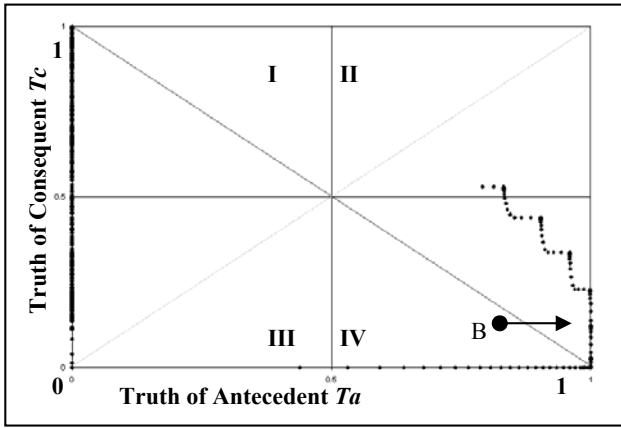

Fig. 3. TSD for R642: IF T1 is HIGH & F1 is HIGH & F2 is LOW & Persistence is HIGH THEN after LONG delay T3 will be MED

Similarly points in the Quadrant I demonstrate the incompleteness of the rule, i.e. the consequent was observed but was due to an event(s) other than the one expressed in the antecedent of the rule. Quadrant III points show the possibility of the rule, but because the value of both the truths is low, it is not possible to confidently use this information. The points that lie on the axis show that either the antecedent or the consequent was not expressed in the data or the rule was expressed insufficiently within the data.

The following concepts can be defined to assess some desired qualities of a linguistic rule:

Goodness – The consequent of a rule statement should be the actual effect of the cause expressed by the antecedent of the rule statement. Hence if the truth of the antecedent is high the truth of the consequent should also be high, subsequently points in Quadrant II show that the rule is good because in this quadrant both Ta and Tc are high. However merely counting the number of points is not correct (discussed earlier) and one should base metrics on the degree to which each point affects this quality of the rule, a metric should not only contain quantitative information but also qualitative information. The following metrics are based on the above concept of goodness:

Strength of Goodness: Tc should be approximately equal to Ta for all points in Quadrant II if a rule is good. Accordingly the RMS distance of the points from the good diagonal {line from point (0,0) to point (1,1)} of the TSD shows how good is the rule. Closer the points are to the main diagonal the better is the rule, i.e. smaller the RMS perpendicular distance of all points:

$$S_G(r) = \sqrt{\frac{\sum_{i=1}^{n_2}\left(\frac{\left|Ta_{i,j} - Tc_{i,j}\right|}{\sqrt{2}}\right)^2}{n_2}}$$

Where n_2 = no. of points in Quadrant II.

Probability of Goodness: If a rule is good it increases the probability of points to occur in Quadrant II than in other quadrants. A normal rule probability and expected number of points in Quadrant II can be calculated. A positive deviation from the expected number of points will show the positive affect of the rule statement and vice versa:

$$P_{II,\text{expected}} = \left(\frac{n_1 + n_2}{n_{tot,TSD}}\right) \times \left(\frac{n_2 + n_4}{n_{tot,TSD}}\right)$$

Where n_x is the actual number of points in Quadrant x. $n_{tot,TSD}$ is the total number of points in the TSD excluding points on the abscissa (Ta=0).

$$n_{2\text{expected}} = P_{II\text{expected}} \times n_{tot,TSD}$$

$$\sigma = \sqrt{n_{2\text{expected}} \times P_{II\text{expected}} \times \left(1 - P_{II\text{expected}}\right)}$$

$$Deviation(r) = \frac{n_2 - n_{2\text{expected}}}{\sigma}$$

Where σ is the standard deviation for a probability distribution with mean at $n_{2\text{ expected}}$.

Combined Goodness: Normally the value of Tc should be equal to the value of Ta, but the actual value of Tc is determined by the quality of the rule statement and the quality of the data. Tc will be lower if the rule statement is wrong or if deviations occur in the data (due to noise etc.). The final value for Tc shall incorporate two deviations:

$$Tc = Ta \pm \sigma_{rule} \pm \varepsilon_{data} \quad \text{If } \sigma_{rule} \gg \varepsilon_{data}$$

then σ for the rule can be calculated:

$$\sigma(r) = \sqrt{\frac{\sum_{i=1}^{n_2}\left(Ta_{i,j} - Tc_{i,j}\right)^2}{n_2}}$$

To incorporate the amount of information available the above metric can be scaled with a quantity based metric:

$$\sigma^*(r) = \sqrt{\frac{\sum_{i=1}^{n_2}\left(Ta_{i,j} - Tc_{i,j}\right)^2}{n_2}} \times \left(1 - \frac{n_2}{n_{tot,TSD}}\right)$$

Badness – If the antecedent of a rule states a cause but the consequent of the rule states a false or incorrect effect then the rule is bad or incorrect. Subsequently, if Ta is high but Tc is low, i.e. a point in Quadrant IV shows that the rule is bad. The following metrics are based on the above concept of badness:

Strength of Badness: The RMS perpendicular distance of points in Quadrant IV from the bad diagonal {line from point (0,1) to (1,0)} of the TSD shows how bad is the rule:

$$S_B(r) = \sqrt{\frac{\sum_{i=1}^{n_4}\left(\frac{\left|Ta_{i,j} + Tc_{i,j} - 1\right|}{\sqrt{2}}\right)^2}{n_4}}$$

Probability of Badness: Similarly to Probability of Goodness, expected points in Quadrant IV can be calculated and the positive deviation from this expected value will show the badness of the rule:

$$P_{IV,\exp ected} = \left(\frac{n_2 + n_3}{n_{tot,TSD}}\right) \times \left(\frac{n_2 + n_4}{n_{tot,TSD}}\right)$$

$$n_{4\exp ected} = P_{IV\exp ected} \times n_{tot,TSD}$$

$$\sigma = \sqrt{n_{4\exp ected} \times P_{IV\exp ected} \times \left(1 - P_{IV\exp ected}\right)}$$

$$Deviation(r) = \frac{n_4 - n_{4\exp ected}}{\sigma}$$

Quantity of badness: The amount of information present in the fourth quadrant can be found using a quantity based metric:

$$n_4^* = \frac{n_4}{n_{tot,TSD}}$$

Incompleteness – An effect can be the consequence of many different causes, an antecedent hence may not be completely responsible for the effect stated in the consequence of the rule. In such a case, the value for Tc will be high but the value for Ta will be low, i.e. points in Quadrant I reflect on the incompleteness of the rule. A qualitative metric can be designed to measure this property of the rule:

$$I(r) = \sqrt{\frac{2 \times \sum_{i=1}^{n1}\left[\left(0.5 - Ta_{i,j}\right)^2 + \left(0.5 - Tc_{i,j}\right)^2\right]}{n_1}}$$

This is the RMS deviation of Quadrant I points from point (0.5,0.5) or lower right hand corner of the quadrant, scaled

from 0 to 1.

Insufficiency of Data – Data may not be available to sufficiently qualify a rule. The antecedent may not be expressed in the data enough number of times to confidently state the goodness, badness or incompleteness of the rule. Hence a metric is defined to state the insufficiency of the data to express a rule:

$$In(d) = 1 - \frac{n_{tot,TSD}}{n_{tot,data}}$$

Where (d) denotes the data-set and n tot, data the total number of points in the data-set.

The objective was to design metrics that will not be affected by amount of total data available, i.e. a metric should not merely increase or decrease because of the size of the data-set. A metric should be scaled (or normalized) from 0 to 1; this improves the Pareto optimization of the rule base. A metric should be robust to noise and other deviations in the data.

The size, content and quality of the final optimized rule base depend on the metrics chosen during the optimization. Different optimization schemes were followed using various combinations of the above metrics, the initial rule base of 729 rules was optimized, and the results were analyzed on various factors such as – **Information content:** were the chosen rules correct? **Information quantity:** was all the information present in the numerical data expressed in the final rule base? **Information quality:** what is the confidence in the chosen rules?

IV. RESULTS AND DISCUSSION

The objective of the above exhaustive search methodology is to determine the goodness criteria that will accept good rules and reject bad rules. Good rules express the phenomenological mechanism. For example, refer to rule statement of Rule 480 (Figure 2). The antecedent part states that a high flow rate of water *(F1)* with high temperature *(T1)* is mixing with a low flow rate of cold water *(F2)* and the persistence of this event is high. The logical consequence is that the resultant delay will be medium and the resultant temperature will be high *(T3)*.

A good rule may not be accepted into the final rule base if the event it describes does not exist within the data from which the knowledge is being extracted. To solve this problem one must use a larger data-set which encompasses all possible process events.

A bad rule is a rule, which is inconsistent with the process phenomena. For example, the logical consequence of the antecedent of Rule 642 (Figure 3) should be that after medium delay the temperature is high. This rule is bad and is removed by the Pareto search method. Certain bad rules may be included in the final rule base due to the vagaries of numerical data such as temporary transition

periods, noise, etc. A bad rule may have a few, scarce data points within the TSD, and if these points happen to be in Quadrant II the bad rule may appear "good" and become accepted. For example, Rule 55 (see Figure 4) is a bad rule, which appears to be good due to the nature of data available.
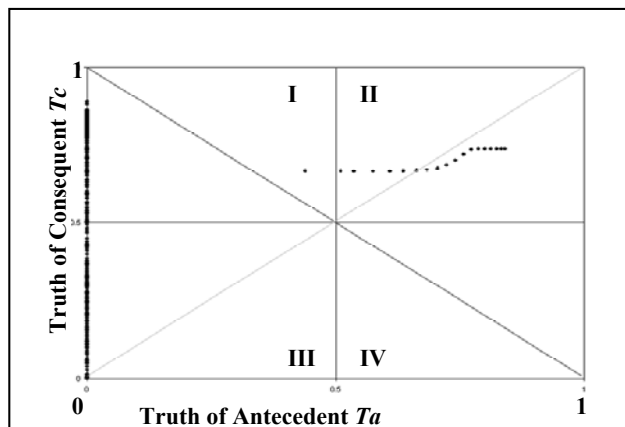

Fig. 4. TSD for R55: IF T1 is HIGH & F1 is LOW & F2 is LOW & Persistence is LOW THEN after SHORT delay T3 will be LOW

Accordingly, only rules that are corroborated by independent data instances should be accepted. A rule should be accepted into the final rule base only if $n_{tot,TSD} \geq$ θ % of $n_{tot,data}$ (total data-set). The (θ %) parameter (percent cut) serves a dual purpose: It removes bad rules, which are triggered due to the inconstancy of numerical data; and it removes those rules, which have a low support or corroboration.

The best rule base was found using the following search scheme: Insufficiency and Incompleteness were used in the first Pareto optimization and Combined Goodness and Quantity of Badness in the second. A percent cut of 1% was used in the removal of bad or unsupported rules. The choice of this parameter depends upon the nature and size of the data available and was found using a simple optimization routine, which minimizes the number of bad rules included in the final rule base. Of the 729 rules in the exhaustive rule base, this two-stage Pareto optimization selected 110 good and sufficiently expressed rules. No selected rules were bad (none were incompatible with process phenomena).

V. Conclusions

The above results provide proof of concept for Truth Space Evaluation, an effective tool for the following purposes:
1. Extracting linguistic information from numerical data.
2. Visualizing the relation between the data and the rule statement. Providing a linguistic depiction of the numerical data according to the rule statement.
3. Providing robust metrics for rule base optimization.
4. Reducing the size and variability of the data-set by using fuzzy membership and persistence.

The exhaustive search method will be ineffective for processes with a high number of input and output variables. More efficient search techniques such as Genetic Algorithms can be used to create successive rule bases and TSD can be used to provide rule base optimization and evaluation.

REFERENCES

[1] Stephanopoulos G. and Han C. Intelligent systems in process engineering: a review, Computers & Chemical Engineering, Volume 20, Issues 6-7, June 1996, Pages 743-791.
[2] O'Shima E. Computer aided plant operation, Computers & Chemical Engineering, Volume 7, Issue 4, 1983, Pages 311-329.
[3] Collewet C., Rault G., Quellec S. and Marchal P. Fuzzy adaptive controller design for the joint space control of an agricultural robot, Fuzzy Sets and Systems, Volume 99, Issue 1, October 1998, Pages 1-25.
[4] Ko K. and Chen J. C. A fuzzy-nets training scheme for controlling nonlinear systems, Computers & Industrial Engineering, Volume 31, Issues 1-2, October 1996, Pages 425-428.
[5] Rhinehart, R. R. and Murugan P. Improve Process control Using Fuzzy Logic, Chemical Engineering process, 91(11), November 1996, Pages 60-65.
[6] Cimander C., Bachinger T. and Mandenius C. Integration of distributed multi-analyzer monitoring and control in bioprocessing based on a real-time expert system, Journal of Biotechnology, Volume 103, Issue 3, August 2003, Pages 237-248.
[7] Fonseca D. J. and Knapp G. M. An expert system for reliability centered maintenance in the chemical industry, Expert Systems with Applications, Volume 19, Issue 1, July 2000, Pages 45-57.
[8] Kordon A. Hybrid Intelligent Systems for Industrial Data Analysis, Seminar given at MCEC meeting, September 2002.
[9] Ngan P. S., Wong M. L., Lam W., Leung K. S. and Cheng J. C. Y. Medical data mining using evolutionary computation, Artificial Intelligence in Medicine, Volume 16, Issue 1, May 1999, Pages 73-96.
[10] Herrera F., Lozano M. and Verdegay J. L. A learning process for fuzzy control rules using genetic algorithms, Fuzzy Sets and Systems, Volume 100, Issues 1-3, November 1998, Pages 143-158
[11] Chen J. C. and Black J. T. A fuzzy-nets in-process (FNIP) system for tool-breakage monitoring in end-milling operations, International Journal of Machine Tools and Manufacture, Volume 37, Issue 6, June 1997, Pages 783-800
[12] Xiong N. and Litz L. Reduction of fuzzy control rules by means of premise learning method and case study, Fuzzy Sets and Systems, Volume 132, Issue 2, December 2002, Pages 217-231
[13] Wong C. and Fan C. Rule mapping fuzzy controller design, Fuzzy Sets and Systems, Volume 108, Issue 3, December 1999, Pages 253-261