Pergamon

Computers chem. Engng Vol. 20, Suppl., pp. S877–S882, 1996

S0098-1354(96)00154-8
0098-1354/96 $15.00+0.00

NONLINEAR RULE BASED PROCESS CONTROL-ANALYSIS AND REDUCTION OF THE RULE SET BY NONLINEAR THEOREY

Jun-Sien Lin and Shi-Shang Jang

Chemical Engineering Department, National Tsing-Hua University
Hsin-Chu, Taiwan, R.O.C.
E-mail: jang@che.nthu.edu.tw

ABSTRACT

Abundant time-series dynamic data can be accumulated from a chemical plant during long term operations. In our previous work, these plant data were directly implemented for the purpose of model predictive control. In this work, fractal analysis is performed to reduce the size of a time-series data set for high quality nonlinear model predictive control. Results in this study indicate that on-line identification of nonlinear models is unnecessary if the disturbances to the process satisfy the fractal-equivalence condition. Simulation examples, including the dual composition control of a high-purity distillation column demonstrate that the nonlinear model predictive scheme is quite useful for those cases in which linear model predictive controller has failed.

KEYWORDS

Nonlinear Model Preictive Control; Rule Based Control; Fractals.

INTRODUCTION

Abundant time series dynamic data can be accumulated during the long term operation of chemical processes. In our pervious work [1], those plant data could be easily organized into the following input (**u**) and output (**y**) rule based model form:

If y_k is in $\mathbf{Y}_k$, and if y_{k-1} is in $\mathbf{Y}_{k-1}$, ..., and if y_{k-n} is in $\mathbf{Y}_{k-n}$, and if u_k is in $\mathbf{U}_k$, ..., and if u_{k-m} is in $\mathbf{U}_{k-m}$, then y_{k+1} is in $\mathbf{Y}_{k+1}$. (R1)

where k represents the current state and, n and m are known process orders. The above rule set can be implemented as the plant model for nonlinear model predictive control (NMPC). However, the following problems remain unsolved:

(i) If the system is not a single-input-single-output (SISO), and/or if the system orders (n and m) are high, the required rule set that provides high quality NMPC may subsequently be extremely large because it requires substantially "dense" rule set.
(ii) On-line identification of low frequency disturbances is not easily performed. The disturbances must first be measured off-line, and, then, the required data set becomes larger.

Model predictive control (MPC) has found successful advanced control scheme applications in the chemical industry since the pioneering work by Cutler and Ramaker [2]. In those works and a later substantial review work [3], only empirical models such as impulse response models and dynamic matrix models were recommended for use by the model based controllers. In linear systems, the system dynamics can be well defined by the above empirical models; this is not true for nonlinear systems. In addition, first principle physical models have been most frequently used in recent

developments of nonlinear model based control schemes [4]. However, a physical model is normally quite difficult to implement on-line. In this work, the time series rule based models (instead of physical models) are implemented as (R1).

In this work, fractal analysis of the rule set (R1) is performed to reduce the burden of storage and also search for a large rule set. Furthermore, nonlinear model predictive control using this rule set is shown to be superior to the original approach owing to the fact that the reduced rule set actually represents the process dynamics in a larger range than the original rule set. Also, analysis results indicate that, for many cases, the equivalence condition for different values of disturbances of the original rule based model is not true. Such a condition can be true in fractal transformed rule set including a very important application case -- high purity distillation systems.

NONLINEAR MODEL PREDICTIVE CONTROL

A general nonlinear lumped system is considered:

$$\begin{aligned} \dot{x} &= f(x,u,d) \\ y &= g(x,u,d) \end{aligned} \qquad (1)$$

where $x \in R^{l}$ are system states, $u \in R^{l'}$ are control variables, $y \in R^{l''}$ are on-line measurements, and $d \in R^{l'''}$ are disturbances. If the computer control system includes with a zero-order-hold element, the following functions ϕ and γ can be obtained from **f** and **g** in (1) such that:

$$\begin{aligned} x_{k+1} &= \phi(x_k, u_k, d_k) \\ y_k &= \gamma(x_k, u_k, d_k) \end{aligned} \qquad (2)$$

It has been shown in our previous work [1], a function vector **F** exists such that:

$$y_{k+1} = F(y_k, y_{k-1}, \cdots, y_{k-n}, u_k, u_{k-1}, \cdots, u_{k-m}, d_k) \qquad (3)$$

Given the fact that the known physical state space model (2) or input-output model (3), the most general model predictive control scheme can be as the following:

$$\min_{u_k, u_{k+1}, \cdots, u_{k+N_O}} \sum_{i=1}^{N_O} \varphi_i(\hat{y}_{k+i}, u_{k+i-1})$$

s. t. (2) or (3)

$$h_j\left(\hat{y}_{k+1}, \hat{y}_{k+2}, \cdots, \hat{y}_{k+N_O}, u_k, \cdots, u_{k+N_O-1}\right) \geq 0 \qquad (4)$$

$j = 1, \ldots, K$

where k represents the present time, φ_i , l=1...., N_o, is the discretized operating cost as a function of only the input and output variables, $\hat{y}_{k+i}$ is the model predicted output into the immediate future, and N_o is the interested time horizon for the immediate future. h_j, j=1,...,K, are a set of operating constraints. In this work, we set $\varphi_i = (\hat{y}_{k+1} - y_s)^T (\hat{y}_{k+1} - y_s)$, where y_s is the set point. Given the physial model (2) , (3) or the rule set (R1), equation (4) can be solved using a standard nonlinear programming software such as GRG2 as shown in our previous work [1]. However, in case that some unknown disturbance **d** arises in (2) or (3), d_k must be updated so that the estimated output $\hat{y}_{k+i}$ can be accurately predicted by the model (2) or (3). To update the model, the following identification problem must frequently be solved on-line:

$$\min_{d_k} \sum_{i=1}^{N_I} \left(\hat{y}_{k-i} - \tilde{y}_{k-i}\right)^2 \qquad (5)$$

s.t. (2) or (3)

Although (5) must be solved on-line, there are some exceptions. As indicated in Economou et al. [5], if the superposition condition is satisfied, on-line identification is not necessary. However, the above superposition condition is generally not true.

Given a rule set model in (R1), and current state (y_k, y_{k-1},..., y_{k-n},u_k,..., u_{k-m}), how the future states $\hat{y}_{k+1}$,$\hat{y}_{k+2}$,...can be estimated, the reader is referred to our previous work [1].

FRACTAL ANALYSIS OF RULE BASED MODEL

The rule based model set in the form of (R1) is next considered. The rule set is simplified into the following form:

If Δy_k is in ΔY_k, and if Δy_{k-1} is in ΔY_{k-1}, ..., and if Δy_{k-n} is in ΔY_{k-n}, and (R2)
if Δu_k is in ΔU_k, ..., and if Δu_{k-m+1} is in ΔU_{k-m+1}, then Δy_{k+1} is in ΔY_{k+1}

where $\Delta y_{k-i} = y_{k-i}-y_{k-i-1}$, and $\Delta u_{k-i} = u_{k-i}-u_{k-i-1}$. Figure 1 shows a set of rules of (R2) for the CSTR example (also the example #1 in this study) given in our previous work in the above simplified rule based model (R2). In Figure 1, the magnitudes of Δy_1 , Δy_2 and Δy_3 are plotted on the three axes for the same control actions Δu_1 , Δu_2 and Δu_3, respectively. This figure indicates that the rules are quite similar to each other except for that the triangle sizes are different. Figure 2 plots the other subsets of the rules; however, the rules form two different sets of similar triangles. Figures 1 and 2 lead to the following definition and algorithm:

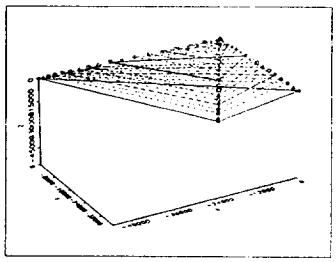


Figure 1a

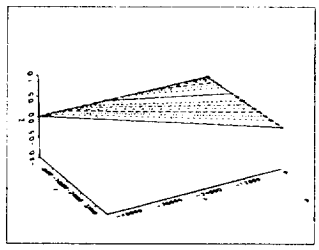


Figure 1b

Figure 1. Fractal shapes of a rule set that can be reduced to a single rule. (1a) A 3-D plot of a n=3 system. (1b) The projection of (1a) to a 2-D plane.

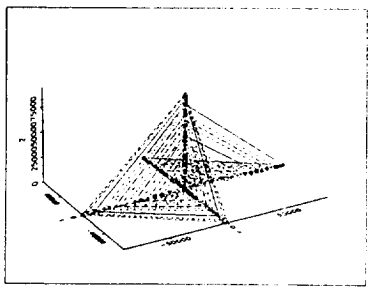


Figure 2a

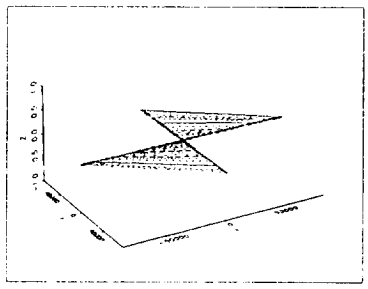


Figure 2b

Figure 2. Fractal shape of a rule set that can be reduced to two rules. (2a) A 3-D plot of a n=3 system. (2b) The projection of (4a) to a 2-D plane.

Let's denoted

$$R = \left\{ x \middle| x = \left(\Delta y_{k+1}, \Delta y_k, \Delta y_{k-1}, \cdots, \Delta y_{k-n+1}, \Delta u_k, \cdots, \Delta u_{k-m+1} \right) \right\} \quad (6)$$

as the reset dynamics set of the dynamic system.

ALGORITHM 1: (Fractal Analysis) *Consider two rules in R, that are with same control actions*

Δu_k , Δu_{k-1}, ..., Δu_{k-m+1}, *if there exists a real number* **M** *such that for any* Δy_{k+1}, Δy_k , Δy_{k-1}, ..., Δy_{k-n+1} *in rule #1 and* $\Delta y'_{k+1}$, $\Delta y'_k$, $\Delta y'_{k-1}$, ... $\Delta y'_{k-n+1}$ *in rule #2 the following is true*:

$$\frac{\Delta y_{k+1}}{\Delta y'_{k+1}} = \frac{\Delta y_k}{\Delta y'_k} = \frac{\Delta y_{k-1}}{\Delta y'_{k-1}} = \cdots = \frac{\Delta y_{k-n+1}}{\Delta y'_{k-n+1}} = M \qquad (7)$$

then one of the rules should be eliminated from the rule set.

In real applications, the exact **M** in (7) can not be found owing to consideration of model uncertainties and measuring noise. The equalities in (7) should be considered with certain range tolerances. Using the above algorithm, all of the rules in Figure 1 will obviously be eliminated except for one rule. Also, all of the rules in Figure 2 will eliminated except for two rules.

However, not all rule sets in any of the systems can be reduced to one rule or two rules. The remaining rules in the transformed rule set of **R** may not be so significantly reduced from the original **R**. Hence, the following scaling factor and experimental fractal dimension are defined as:

Definition 1. (Scaling Factor and Fractal Dimension) *Given a subset S (fractal set) of* **R** *that the rules in the subset are all with the same control actions* Δu_k , Δu_{k-1}, ..., Δu_{k-m+1} , *then the magnification factor* **M** *in the above* ***ALGORITHM 1*** *is denoted as the scaling factor of the fractal set, and the experimental fractal dimension* **D** *is denoted by the following*:

$$D = \text{slop}\left(\frac{\ln(\text{No. of rules removed from S})}{\ln(M)} \right) \qquad (8)$$

In most systems, systematic approach is unavailable to find the number of rules remaining after the analysis in **ALGORITHM 1** has been performed for **R**. In fractal analysis, the above fractal dimension reflects the "density" of the systems. The following fractal transformed rule set is defined in the following.

For simplicity of the expression, let's denote the fractal reduction of R using the methodology in **ALGORITHM 1** by :

$$S = \Im\{R\} \qquad (9)$$

Given a set of data (y_k, y_{k-1},..., y_{k-n},u_k,..., u_{k-m}), for the current state, The predictions of the future states $\hat{y}_{k+1}$,$\hat{y}_{k+2}$,... using the fractal transformed rule set S., the reader is referred to [6]. The following is hence denoted:

$$\hat{y}_{k+1} = \aleph\{S\} \qquad (10)$$

The reconstructed algorithm in this figure can also be expanded to a wider situation that requires on-line identification of the generalized NMPC (4). The following can be easily attained by:

Property 1: **(Fractal Equivalence)** *Consider the following dynamic system:*

$$y_{k+1} = F(\bullet, u_k, d) \qquad (11)$$

and n sets of rules $R_1(d_1)$, $R_2(d_2)$, ..., $R_n(d_n)$, *directly obtained from (8). In case that*

$$S = \Im(R_1) = \Im(R_2) = \cdots = \Im(R_n) \qquad (12)$$

then

$$S = \Im(R_1 \cup R_2 \cup \cdots \cup R_n) \qquad (13)$$

also

$$y_{k+1}(d_i) = \aleph(S) \qquad (14)$$

where $i=1,\ldots,n$

The validity of the above fractal transform can be summarized in the following:
(i) The rule set S is usually much smaller than **R**.

(ii) The reconstruction of the rule set using S may yield a better estimate of the future states than the original search of **R** because finding the rule to exactly match the current states in **R** is not possible. However, in S, current state can more likely be reconstructed exactly if no measuring noise exists by selecting an appropriate magnification factor **M** in (10). This indicates that **S** is actually more "dense" than **R**.
(iii) If the condition in **Property 1** is true, on-line identification is not required to solve the NMPC problem (4).
(iv) There is no need to use **R** to predict the future states,any new state of the system can be reconstructed for its original form in **R** by comparing all rules in S and inverse calculating for **M** in (7).

NUMERICAL EXAMPLES

Realistic examples are simulated in this section to demonstrate the validity of a fractal transformed rule based model. The single-input-single-output (SISO) CSTR control problem demonstrated in our previous work [1] and a high purity column that separates methanol from water are simulated to present the validity of the fractal transformed rule based NMPC scheme. For the design parameters of the high purity column, the reader is referred to [6]. This distillation column is actually a high purity column with top product x_D=0.998 and bottom x_B=0.001. The simulation is based on a rigorous material and energy balances equation of the tray-by-tray system. A comparison of two conventional controllers were made to control two compositions of the column. On-line testing indicated that a n=3, m=3 rule based model is sufficient for NMPC, although the system orders are actually very high. In this case, a comparison is made of the case in which only the top product is controlled (SISO)with that in which both compositions of top and bottom are controlled (MIMO).

One thousand rules based on the same control actions Δu_1 , Δu_2 and Δu_3, but different initial conditions are obtained from the SISO CSTR example. Figure 3 shows the experimental fractal dimensions plot for this case. The other analysis shows the same case with ten thousand the initial rules. The fractal dimensions for both cases are roughly the same (D=0.0102344 for 1,000 rules and D=0.0101234 for 10,000 rules). Figure 4 gives the rules remained, number of elements in S, after fractal reduction with different numbers of initial rule set **R** that is with the same control action. It shows that the number of remaining rules after the fractal reduction is about the same if the numbers of initial rules reaches to 15,000. This reveals the fact that, under this set of control action, this 52 elements in S may represent the dynamics of the system much more "dense" than its initial rule set **R**. Table 1 shows that the computer memories needed to save the original rule set and the fractal transformed rule set for the above examples. It can be found from from Table 1 that the memories implemented to store the rule set is drastically reduced by the fractal analysis. Figure 5 shows that the fractal transformed rule based NMPC performs much better than a well-tuned PID controller for the case of SISO CSTR

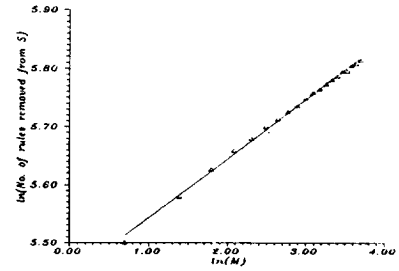
Table 1. Comparison of computer memories for storing the original set (R) and fractal transform set (S) for different examples

	Memory for R	Memory for S
CSTR	88 Mega	1.3 Mega
SISO Column	21 Mega	3 Mega
MIMO Column	80 Mega	3.1 Mega

Figure 6 compares the performances of the fractal transformed rule based control and the NMPC proposed by Economou et al.[5] for the case of SISO high purity column. We assume that both controllers are all designed based on the wrong case (x_F=0.3). Since the fractal transformed rule based model can track the column dynamics well for x_F=0.2 the rule based NMPC works fine. The NMPC using physical model with wrong parameter (x_F=0.3) may work for regulation case because the system does not deviate from its steady state too much during the transient. However, if a set point change is implemented, the system becomes unstable as shown in Figure 6. Figures 7a and 7b compare the MIMO distillation control using the NMPC with that using PI controller with 2% measuring noise. Once again, the NMPC performs much better than PI controllers.

CONCLUSION

A fractal transformed rule base model was proposed to perform NMPC in this work. The transformed model could actually represent the system dynamics in a wider range than the original rule based model in the sense that a different number of the original rules could be transformed into the same rule set. Fractal analysis results indicated that if the rule set with different values of disturbances can be transformed to the same rule set, then on-line identification would be unnecessary. Furthermore, realistic examples including high-purity distillation column were also simulated. The simulation examples demonstrated that the fractal transformed rule based is markedly superior to the conventional controllers.



Figur e 3. Fractal dimension analysis of the SISO CSTR based on 1000 rules.

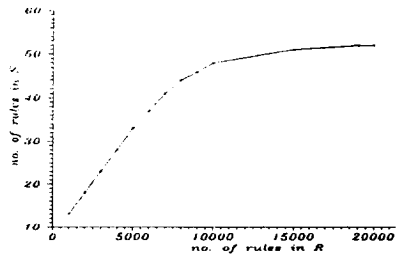


Figure 4. Numbers of rules remained in the fractal transformed set S from the original rule sets R's with different initial number.

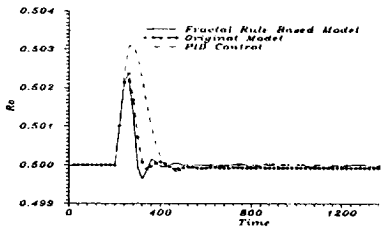


Figure5. Comparisons of regulation (Ai = 1.05) behaviors of the SISO CSTR example among (1)the fractal rule based model control, (2)original rule based model control, (3) PID control

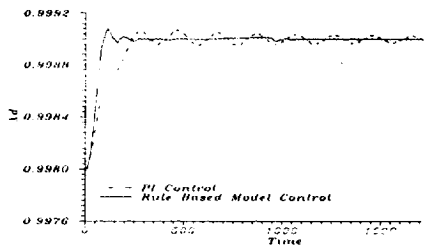


Figure 7a

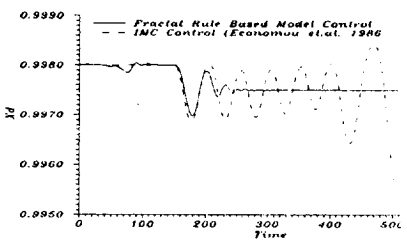


Figure 6 Comparisons between the rule bassed NMPC and NIMC (Economou et al., 1986) for the SISO case of the high purity column

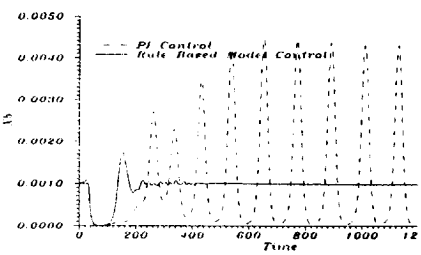


Figure 7b

Figure7. Comparisons of regulation behavior (x_F change from 0.3 to 0.2) between the fractal transformed NMPC and PI controllers for MIMO high purity column.

LITERATURE CITED

[1] Peng, C.Y. and Jang, S.S., 1994, Nonlinear Rule Based Model Predictive Control of Chemical Processes, I&EC Research, **33**, 9, 2140.

[2] Culter, C.R. and Ramaker, B.L., 1979, Dynamic Matrix Control: A Computer Control Algorithm, 86th Nat. Meet. AIChE, Houston (April).

[3] Garcia, C. E., Morari, M., 1982,Internal Model Control 1. A Unifying Review and Some New Result, Ind. Eng. Chem., Process Des. Dev., 21(2), 308-323.

[4] Jang, S.S., Joseph, B. and Mukai, H., 1987, On-Line Optimization of Constrained Multivariable Chemical Processes, AICHE J., **33**, 1, 26.

[5] Economou, G.E., Morari, M., and Palsson, B.O., 1986, Internal Model Control. 5. Extension to Nonlinear Systems, Ind. Eng. Chem. Process Des. Dev., **25**, 403-411.

[6] Peng, C.Y., 1995, Nonlinear Rule Based Model Predictive Control and Fractal Analysis, PhD Thesis, National Tsing Hua University, Hsin Chu, Taiwan, ROC.