

# Modeling of an Inverted Pendulum based on Fuzzy Clustering Techniques

E.Sivaraman

Assistant Professor,  
Annamalai University  
Annamalainagar, Tamilnadu  
India - 608002

S.Arulselvi

Associate Professor,  
Annamalai University  
Annamalainagar, Tamilnadu  
India - 608002

S.P.Natarajan

Professor,  
Annamalai University  
Annamalainagar, Tamilnadu  
India - 608002

## ABSTRACT

The inverted pendulum is a highly nonlinear and open loop unstable system. To develop an accurate model of the inverted pendulum, different linear and nonlinear methods of identification will be used. However one of the problems encountered during modeling is the collection of experimental data from the inverted pendulum system. Since the output data from the unstable system does not show enough information or dynamics of the system. This can be overcome by designing a feedback controller, which stabilize the system before identification can take place. Recently Takagi-Sugeno (T-S) fuzzy modeling based on clustering techniques have shown great progress in identification of nonlinear systems. Hence in this paper, Takagi-Sugeno (T-S) model is proposed for an inverted pendulum based on fuzzy c-means, Gustafson-Kessel (G-K) and Gath-Geva (G-G) clustering techniques. Simulation results show that Gustafson-Kessel (G-K) clustering technique produces satisfactory performance.

## General Terms

Fuzzy logic, Modeling.

## Keywords

Nonlinear, Clustering, Fuzzy, Inverted Pendulum, Takagi-Sugeno.

## 1. INTRODUCTION

Fuzzy control system, identification and design have been a subject of intense research in recent years. The heart of a fuzzy control system consists of a set of fuzzy “if – then” rules, which models the process state-control action relationship in the control domain. The design of appropriate fuzzy rules and membership is a critical bottleneck in developing powerful fuzzy control systems. Membership functions and fuzzy rules can be defined by the model developer (expert), using prior knowledge, or by experimentation, which is a typical approach in knowledge based fuzzy control [1]. Knowledge acquisition however is, a cumbersome task, and for (partially) unknown systems, human experts are not available. Therefore, data-driven construction of fuzzy membership function and rules from measured input - output data has received a lot of attention. Such modeling approaches typically seek to optimize some numerical objective function, while less attention is paid to the complexity of the resulting model in terms of the number of membership functions and rules, Babuska [2]. Takagi- Sugeno [3] proposed a heuristic method for designing Takagi - Sugeno type of fuzzy rules from input-output data, using some performance index. Subsequently, Sugeno and Tanaka reported [4] a method for on-line identification of a fuzzy system. Recently, several proposals are made to use fuzzy clustering for fuzzy system identification. An

effective approach is to partition the available data into subsets and approximate each subset by a simple piecewise linear model. Fuzzy clustering can be used as tool to partition the data where transitions between the subsets are gradual [5]. Sugeno and Yasukawa [6] uses the fuzzy c-means algorithm to cluster the output data points to determine the fuzzy rules, and use some heuristics to select the most relevant input variables in fuzzy system design. Bezdek [7] combines the ideas in [6] and [8] to develop a neural network fuzzy control system which also utilizes clustering. Typically, the specification of a real-world control problem is given by a finite set of input-output data points  $z_k = [x_k, y_k]$ , and the design task is to extract optimum membership functions and good set of fuzzy rules that best represent the input-output relationship in the data. This paper address the design issue by demonstrating good performance of the fuzzy clustering-based methods, which identifies clusters from the input-output data and constructs membership function thereby generating a fuzzy rule for each cluster. In this paper, the inverted pendulum problem has been considered as a typical representative of inherently unstable nonlinear control systems, thus it is an ideal choice for testing the modeling capability of the fuzzy c-means, Gustafson-Kessel (G-K) and Gath-Geva(G-G) clustering algorithms. Simulation results are presented using fuzzy modeling and identification toolbox [2]. The performance of the models obtained by fuzzy c- means ,G-K and G-G methods are compared in terms of RMSE.

## 2. TAKAGI-SUGENO FUZZY MODEL

Consider the identification of an unknown nonlinear system

$$y = f(x) \quad (1)$$

based on some available input-output data  $x_k = [x_{1,k}, \dots, x_{n,k}]^T$  and  $y_k$ . The index  $k=1, \dots, N$  denotes the individual data samples. While it may be difficult to find a model to describe the unknown system globally, it is often possible to construct local linear models around selected operating points. The modeling framework that is based on combining local models valid in predefined operating region is called operating regime-based modeling. In this frame work, the model is generally given by

$$\hat{y} = \sum_{i=1}^c \phi_i(x) (a_i^T x + b_i) \quad (2)$$

where  $\phi_i(x)$  is the validity function for the  $i^{\text{th}}$  operating regime and  $\theta_i = [a_i^T \ b_i]^T$  is the parameter vector of the corresponding local linear model. The operating regimes can also be represented by fuzzy sets, in which case the TS fuzzy model is obtained [6] as

$$R_i : \text{If } x \text{ is } A_i(x) \text{ then } \hat{y} = a_i^T x + b_i, [w_i] \quad (3)$$

$i=1, \dots, c.$

here, the number of rules is denoted by  $c$ ,  $A_i(x)$  is a multivariable membership function,  $a_i$  and  $b_i$  are parameters of the local linear model as shown in Fig.1, and  $w_i \in [0,1]$  is the weight of the rule.

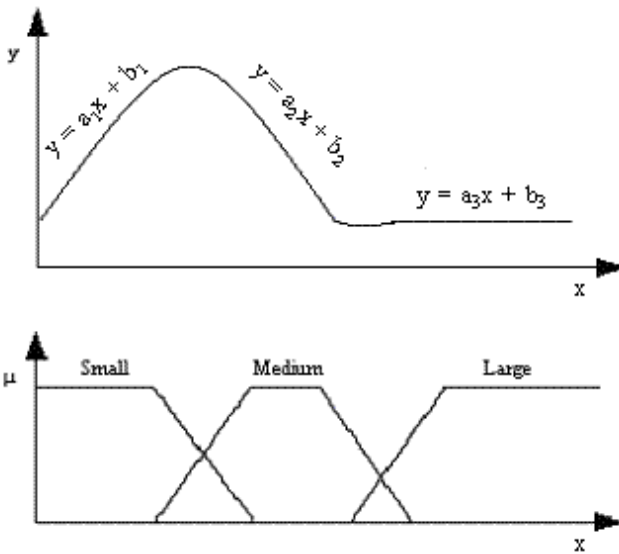


Fig. 1. Local linear model and the membership function diagram.

The value of  $w_i$  is usually chosen by the designer of the fuzzy system to represent the belief in the accuracy of the  $i^{\text{th}}$  rule. When such knowledge is not available  $w_i = 1$ ,  $A_i$  is used. The degree of fulfillment of the rule can be calculated as the product of the individual membership degrees and the rule's weight

$$\beta_i(x) = w_i A_i(x) = w_i \prod_{j=1}^n A_{i,j}(x_j) \quad (4)$$

The rules are aggregated by using the fuzzy-mean formula

$$\hat{y} = \frac{\sum_{i=1}^c \beta_i(x) (a_i^T x + b_i)}{\sum_{i=1}^c \beta_i(x)} \quad (5)$$

From (2) and (5), one can see that the fuzzy model is equivalent to the operating regime-based model when the validity function is chosen to be the normalized rule degree of fulfillment as

$$\phi_i(x) = \frac{\beta_i(x)}{\sum_{i=1}^c \beta_i(x)} \quad (6)$$

Prior to clustering, the regression structure of the model is selected, in order to properly represent the system dynamics. Problems where little prior knowledge is available are usually represented in an input-output form. After the structure is determined, clustering in the product space of the regressors and of the regressand can be applied to partition the data. Since each cluster serves as a local linear model of the system, the fuzzy c-means, Gustafson-Kessel (G-K) and Gath-Geva clustering algorithms are capable of detecting clusters which lie in linear subspaces.

### 3. FUZZY MODEL IDENTIFICATION BASED ON FUZZY C-MEANS, G-K AND G-G CLUSTERING ALGORITHMS

Forward and inverse modeling techniques helps to design model based control techniques like direct inverse, Internal Model Control and Model Predictive Control for nonlinear processes. These control techniques give satisfactory response when conventional control techniques fail. Hence, this necessitates the development of different modeling techniques based on fuzzy clustering.

#### 3.1 Fuzzy Partition

From the available input-output data pairs, the regression matrix  $X$  and the output vector  $y$  are constructed as given by

$$X^T = [x_1, \dots, x_N] \text{ and } y^T = [y_1, \dots, y_N]. \quad (7)$$

where  $N \gg n$  is the number of samples used for identification. The antecedent fuzzy sets  $A_i$  in (3) are determined by means of fuzzy clustering in the product space of the systems input and outputs.

Hence, the data set  $Z = R^{(n+1) \times N}$  to be clustered is represented as a  $(n+1) \times N$  data matrix composed from  $X$  and  $y$

$$Z^T = [X, y] \quad (8)$$

where each column  $z_k$ ,  $k = 1, 2, \dots, N$  of  $Z$  contains an input-output data pair as given by

$$Z_k = [X_k^T, y_k]^T$$

Given  $Z$  and an estimated number of clusters 'c', fuzzy clustering partitions  $Z$  into 'c' fuzzy clusters. A fuzzy partition can be represented as an ' $N \times c$ ' matrix  $U$ , whose elements  $\phi_i(x) \in [0,$

$1]$  represents the membership degree of  $Z_k$  in clusters 'i'. Hence, the  $i^{\text{th}}$  column of  $U$  contains the values of the membership function in the fuzzy partition, which is taken to be a point wise representation of the antecedent fuzzy set  $A_i$  of the  $i^{\text{th}}$  rule as given in (3). The sum of each column of  $U$  is constrained to one, but the distribution of membership among the 'c' fuzzy subsets is not considered. Also, there can be no empty clusters and no cluster

may contain all the objects. This means that the membership degrees in the partition matrix U are normalized, and for the given identification data, the membership values  $\phi_i(x)$  correspond to the normalized degree of fulfillment of the rule antecedents as given in (6). In this paper, membership functions extracted using fuzzy c-means, G-K clustering and Gath - Geva algorithms are discussed in detail. Each cluster forms one fuzzy rule.

### 3.2 Cluster Validation

Cluster validity refers to the problem whether a given fuzzy partition fits to the data all. The clustering algorithm always tries to find the best fit for a fixed number of clusters and the parameterized cluster shapes. However this does not mean that even the best fit is meaningful at all. Either the number of clusters might be wrong or the cluster shapes might not correspond to the groups in the data, if the data can be grouped in a meaningful way at all. Clustering data for different values of c, and using validity measures to assess the goodness of the obtained partitions. Different types of validity measure to find optimum number of clusters are given below

#### 3.2.1 Partition Coefficient (PC)

It measures the amount of “overlapping” between cluster. It is defined by Bezdek as follows

$$PC(c) = \frac{1}{N} \sum_{i=1}^c \sum_{j=1}^N (\mu_{ij})^2$$

where  $\mu_{ij}$  is the membership of data point j in cluster i. The disadvantage of PC is lack of direct connection to some property of the data themselves. The optimal number of cluster is at the maximum value.

#### 3.2.2 Classification Entropy (C)

It measures the fuzziness of the cluster partition only, which is similar to the partition coefficient.

$$CE(c) = -\frac{1}{N} \sum_{i=1}^c \sum_{j=1}^N \mu_{ij} \log(\mu_{ij})$$

#### 3.2.3 Partition Index(SC)

It is the ratio of the sum of compactness and separation of the clusters. It is a sum of individual cluster validity measures normalized through division by the cardinality of each cluster

$$SC(c) = \frac{\sum_{i=1}^c \sum_{j=1}^N (\mu_{ij})^m \|x_j - v_i\|^2}{N_i \sum_{k=1}^c \|v_k - v_i\|^2}$$

SC is useful when comparing different partitions having equal number of clusters. A lower value of SC indicates a better partition.

#### 3.2.4 Separation Index(S)

On the contrary of partition index (SC), the separation index uses a minimum-distance separation for partition validity

$$S(c) = \frac{\sum_{i=1}^c \sum_{j=1}^N (\mu_{ij})^2 \|x_j - v_i\|^2}{N \min_{i,k} \|v_k - v_i\|^2}$$

#### 3.2.5 Xie and Beni's Index (XB)

It aims to quantify the ratio of the total variation within clusters and the separation of clusters

$$XB(c) = \frac{\sum_{i=1}^c \sum_{j=1}^N (\mu_{ij})^m \|x_j - v_i\|^2}{N \min_{i,j} \|x_j - v_i\|^2}$$

The optimal cluster should minimize the value of the index.

#### 3.2.6 Dunn's Index (DI)

This index is originally proposed to use at the identification of “compact and well separated clusters”. So the result of the clustering has to be recalculated as it is a hard partition algorithm.

$$DI(c) = \min_{i \in c} \left\{ \min_{j \in c, i \neq j} \left\{ \frac{\min_{x \in c_i, y \in c_j} d(x, y)}{\max_{k \in c} \left\{ \max_{x, y \in c} d(x, y) \right\}} \right\} \right\}$$

The main drawback of Dunn's index is computational since calculating becomes computationally very expensive as c and N increase.

#### 3.2.7 Alternative Dunn's Index (ADI)

The aim of modifying the original Dunn's index was that the calculation becomes more simple, when the dissimilarity function between two clusters ( $\min_{x \in c_i, y \in c_j} d(x, y)$ ) is rated in

value from beneath by the triangle-non equality:

$$d(x, y) \geq |d(y, v_j) - d(x, v_j)|$$

Where  $v_j$  is the cluster centre of the j-th cluster.

$$ADI(c) = \min_{i \in c} \left\{ \min_{j \in c, i \neq j} \left\{ \frac{\min_{x_i \in c_i, x_j \in c_j} |d(y, v_j) - d(x_i, v_j)|}{\max_{k \in c} \left\{ \max_{x, y \in c} d(x, y) \right\}} \right\} \right\}$$

Note that the only difference of SC, S and XB is the approach of the separation of clusters. In the case of overlapped clusters the values of DI and ADI are not really reliable because of re-partitioning the results with the hard partition method. The above performance indexes are used to get optimum number of clusters for the following algorithms.

### 3.3 Fuzzy C-means Clustering Algorithm

Most analytical fuzzy clustering algorithms are based on optimization of the basic c-means objective function, or some modification of it. The advantage of fuzzy c-means algorithm is simplicity and easy implementation. Given the data set Z, choose the number of clusters by cluster validation as  $1 < c < N$ , the weighting exponent  $m > 1$ . The termination tolerance  $\varepsilon > 0$  and the norm-inducing matrix A. Initialize the partition matrix randomly such that  $U^{(0)} \in M_{fc}$ . The following steps are repeated using MATLAB software for  $i = 1, 2, \dots, 1$ , until the termination tolerance is achieved, then the values of membership functions [9] are obtained as given in (11).

Step (1): Compute the cluster prototypes (means)

$$V_i^{(l)} = \frac{\sum_{k=1}^N (\mu_{ik}^{(l-1)})^m Z_k}{\sum_{k=1}^N (\mu_{ik}^{(l-1)})^m}, \quad 1 \leq i \leq c \quad (9)$$

Step (2): Compute the Euclidian distances

$$D_{ikA}^2 = (Z_k - V_i^{(l)})^T A (Z_k - V_i^{(l)}), \quad (10)$$

$$1 \leq i \leq c, \quad 1 \leq k \leq N$$

Step (3): Update the partition matrix

If  $D_{ikA} > 0$  for  $1 \leq i \leq c, 1 \leq k \leq N$ ,

$$\mu_{ik}^{(l)} = \frac{1}{\sum_{i=1}^c (D_{ikA} / D_{jkA})^{2/(m-1)}}$$

otherwise

$$\mu_{ik}^{(l)} = 0 \text{ if } D_{ikA} > 0, \text{ and } \mu_{ik}^{(l)} \in [0, 1] \quad (11)$$

with  $\sum_{i=1}^c \mu_{ik}^{(l)} = 1$  until  $\| U^{(l)} - U^{(l-1)} \| < \varepsilon$

### 3.4 G-K Fuzzy Clustering Algorithm

Gustafson and Kessel extended the standard fuzzy c-means algorithm by employing an adaptive distance norm, in order to detect clusters of different geometrical shapes in one data set [7]. Each cluster has its own norm-inducing matrix  $A_i$ . The matrices  $A_i$  are used as optimization variables in the c-means functional, thus allowing each cluster to adapt the distance norm to the local

topological structure of the data. Given the data set Z, choose the number of clusters by cluster validation as  $1 < c < N$ , the weighting exponent  $m > 1$  and the termination tolerance  $\varepsilon > 0$ . Initialize the partition matrix randomly such that  $U^{(0)} \in M_{fc}$ . The following steps are repeated using MATLAB software for  $l=1, 2, \dots$ . The G-K algorithm is same as that of fuzzy c-means, only the computation of distance measure is calculated using co-variance matrices as follows:

Co-variance matrix

$$F_i = \frac{\sum_{k=1}^N (\mu_{ik}^{(l-1)})^m (Z_k - V_i^{(l)})(Z_k - V_i^{(l)})^T}{\sum_{k=1}^N (\mu_{ik}^{(l-1)})^m}, \quad (12)$$

where  $1 \leq i \leq c$

Euclidian distance

$$D^2_{ikA_i} = (Z_k - V_i^{(l)})^T * \left[ (\rho_i \det(F_i))^{1/n} F_i^{-1} \right] * (Z_k - V_i^{(l)}), \quad 1 \leq i \leq c, \quad 1 \leq k \leq N \quad (13)$$

### 3.5 G-G Fuzzy Clustering Algorithm

The fuzzy maximum likelihood estimates (FMLE) clustering algorithm employs a distance norm based on the fuzzy maximum likelihood estimates proposed by Bezdek and Dunn [9]. The Euclidian distance functions is chosen as

$$D_{ik\Sigma_i} = \frac{\sqrt{\det(\Sigma_i)}}{P_i} * \exp \left\{ \frac{1}{2} (z_k - v_i^{(l)})^T \Sigma_i^{-1} (z_k - v_i^{(l)}) \right\}, \quad (14)$$

$1 \leq i \leq c, \quad 1 \leq k \leq N$

The priori probability of selecting cluster  $i$  is given by

$$P_i = \frac{1}{N} \sum_{k=1}^N \mu_{ik}^{(l-1)}, \quad 1 \leq i \leq c. \quad (15)$$

Given the data set Z, choose the number of clusters by cluster validation as  $1 < c < N$  and the termination tolerance  $\varepsilon > 0$ . The following steps are repeated using MATLAB software for  $l=1, 2, \dots$ . The Gath - Geva algorithm is same as that of G-K, only the computation of distance norm involves an exponential term and thus decreases faster than the inner-product norm. The fuzzy covariance matrix of the  $i^{\text{th}}$  cluster is given by

$$\Sigma_i = \frac{\sum_{k=1}^N (\mu_{ik}^{(l-1)})^w (Z_k - V_i^{(l)})(Z_k - V_i^{(l)})^T}{\sum_{k=1}^N (\mu_{ik}^{(l-1)})^w}, \quad (16)$$

$1 \leq i \leq c.$

The difference between the matrix  $F_i$  in GK algorithm and the  $\sum_i$  define above is that the latter does not involve the weighting exponent  $m$ , instead of this it consists of  $w = 1$ .

#### 4. THE INVERTED PENDULUM PROBLEM

Very often the quality of a control algorithm is tested and demonstrated on the inverted pendulum problem due to its inherent instability and dynamic characteristics. Figure 2 shows the free-bodied diagram of the inverted pendulum system. It is the problem of learning, how to balance an upright pole. Its solution consists of finding the horizontal force to be applied to the cart in order to balance the pole. The cart is moving on the track with no friction. Also, the pole is tied up to the cart by a frictionless hinge. Both the cart and the pole have only one degree of freedom, i.e. each of them can move in vertical plane only. Lagrange equations can be used to derive dynamical system equations for a complicated mechanical system such as the inverted pendulum. The Lagrange equations use the kinetic and potential energy in the system to determine the dynamical equations of the inverted pendulum system.

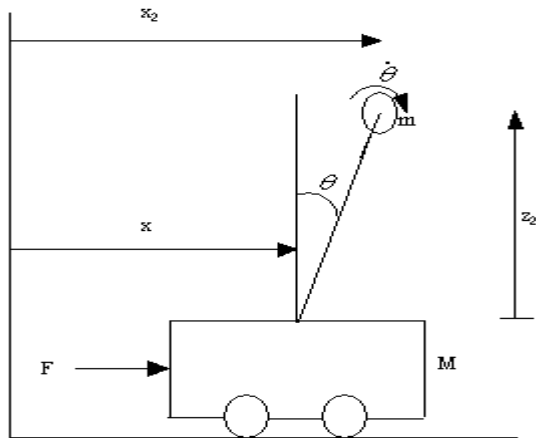


Fig. 2 Inverted pendulum

where,

- $\theta$  - Angle of the pole with respect to the vertical axis
- $\dot{\theta}$  - Angular velocity of the pole with respect to the vertical axis
- F - Force applied to the cart
- M - Mass of the cart
- m - Mass of the pole
- X - Position of the cart
- $\dot{X}$  - Velocity of the cart

The kinetic energy of the system is the sum of the kinetic energies of each mass. The kinetic energy  $E_c$  of the cart is

$$E_c = \frac{1}{2} M \dot{x}^2 \quad (17)$$

The pole can move in both the horizontal and vertical directions. So the pole kinetic energy is

$$E_p = \frac{1}{2} m (\dot{x}_2^2 + \dot{z}_2^2) \quad (18)$$

From the free bodied diagram  $x_2$  and  $z_2$  are equal to

$$x_2 = x + L(\sin \theta) \quad (19)$$

$$z_2 = L(\cos \theta) \quad (20)$$

$$\dot{z}_2 = -L(\sin \theta) \dot{\theta} \quad (21)$$

$$\dot{x}_2 = \dot{x} + L(\cos \theta) \dot{\theta} \quad (22)$$

The total kinetic energy, E of the system is equal to

$$E = E_c + E_p \quad (23)$$

Substitute Eq. (17),(18), (21)and (22) in (23)

$$E = \frac{1}{2} M \dot{x}^2 + \frac{1}{2} m \dot{x}^2 + \frac{1}{2} mL^2 \dot{\theta}^2 + mL\dot{x}(\cos \theta) \dot{\theta} \quad (24)$$

The potential energy, V of the system is stored in the pendulum so

$$V = mgz_2 \quad (25)$$

Substitute equation (20) in (25)

$$V = mgL(\cos \theta) \quad (26)$$

The Lagrangian function is

$$P = E - V$$

$$P = \frac{1}{2} (M + m) \dot{x}^2 + mL(\cos \theta) \dot{x} \dot{\theta} + \frac{1}{2} mL^2 \dot{\theta}^2 - mgL(\cos \theta) \quad (27)$$

The state-space variables of the system are  $x$  and  $\theta$ , so the lagrange equations are

$$\frac{d}{dt} \left( \frac{\partial P}{\partial \dot{x}} \right) - \frac{\partial P}{\partial x} = F \quad (28)$$

$$\frac{d}{dt} \left( \frac{\partial P}{\partial \dot{\theta}} \right) - \frac{\partial P}{\partial \theta} = 0 \quad (29)$$

Solving equation (28) and (29) the dynamic equations of the inverted pendulum is obtained given below

$$\dot{x}_1 = x_2 \quad (30)$$

$$\dot{x}_2 = \frac{\begin{pmatrix} (M + m)g(\sin x_1) \\ -mL(\sin x_1)(\cos x_1)x_2^2 - (\cos x_1)F \end{pmatrix}}{L((M + m) - m\cos^2 x_1)} \quad (31)$$

$$\dot{x}_3 = x_4 \quad (32)$$

$$\dot{x}_4 = \frac{mL(\sin x_1)x_2^2 - mg(\sin x_1)(\cos x_1)}{(M+m) - m\cos^2 x_1} + \frac{F}{(M+m) - m\cos^2 x_1} \quad (33)$$

Let the state variables

$x_1 = \theta$  ; Angle of the pole with respect to the vertical axis

$x_2 = \dot{\theta}$  ; Angular velocity of the pole with respect to the vertical axis

$x_3 = x$  ; Position of the cart

$x_4 = \dot{x}$  ; Velocity of the cart

### 5. SIMULATION RESULTS

To simulate the above equations, the mass of the cart,  $M$  is set to 1.2 kg, mass of the pendulum is set to 0.1 kg, length of the pendulum is 0.4 meters, gravitational force,  $g$  is set to 9.81 m/s. The above state equations are simulated using SIMULINK software. It is observed that the angle of the inverted pendulum shown in Fig. 3 does not give us enough information on the inverted pendulum system. The pendulum falls over quickly and

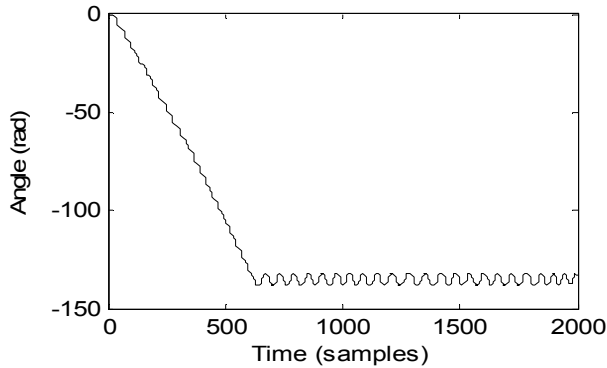


Fig. 3. Open loop response of the inverted pendulum

it found to be unstable. One of the requirements in system identification is the collection of ‘information rich’ input-output data. In order to adequately model the inverted pendulum it is necessary to stabilize it using a nonlinear feedback controller.

#### 5.1 Design of Nonlinear Feedback Controller

Using a nonlinear feedback controller, the output data will contain more information for describing the process. The following equations are the control law developed for the inverted pendulum.

$$h_1 = \frac{3}{4l} g \sin \theta \quad (34)$$

$$h_2 = \frac{3}{4l} \cos \theta \quad (35)$$

$$f_1 = m \left( l \sin \theta \dot{\theta}^2 - \frac{3}{8} g \sin 2\theta \right) \quad (36)$$

$$f_2 = M + m \left( 1 - \frac{3}{4} \cos^2 \theta \right) \quad (37)$$

$$u = \frac{f_2}{h_2} \left[ h_1 + k_1(\theta - \theta_d) + k_2\dot{\theta} + c_1(x - x_d) \right] - f_1 \quad (38)$$

Equation (38) calculates the required force  $u$ , to keep the pendulum stable. For simulation,  $k_1=25$ ,  $k_2=10$ ,  $c_1=1$  and  $c_2=2.6$  [10]. Also  $x_d = 0$  meters and  $\theta_d = 0$  rad, which are the desired position of the cart and angle of the pendulum, respectively. The inputs to this controller are the four output states of the non-linear pendulum model. The correct magnitude and force to keep the pendulum stable is calculated by the control law. The control law and closed loop model of the inverted pendulum developed by simulink. The closed loop response of the inverted pendulum with controller and force applied to the cart are shown in Fig.4 and Fig.5 respectively.

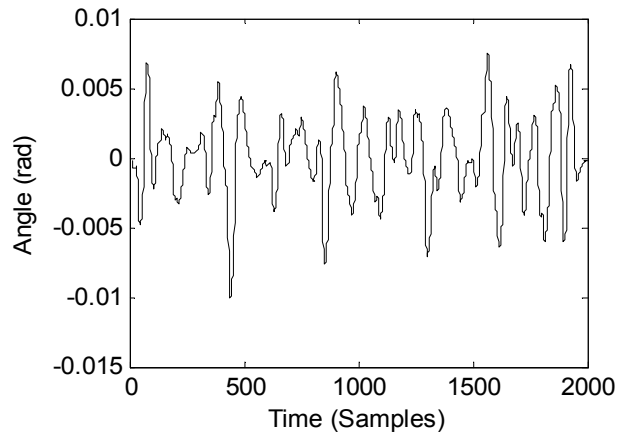


Fig. 4. Closed loop response of the inverted pendulum with Controller.

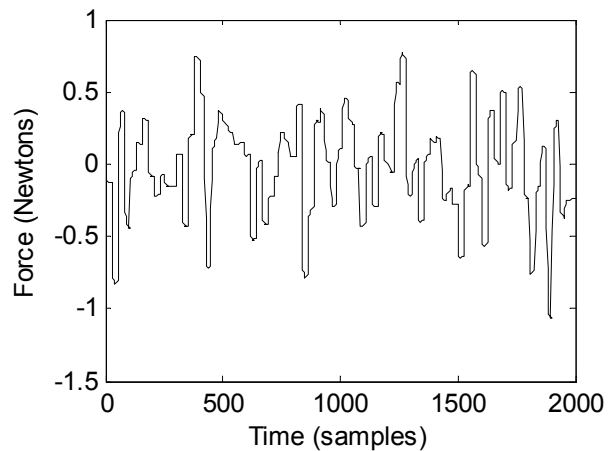


Fig.5 Force applied to the cart

## 5.2 Generation of Input-Output Data with Nonlinear Controller

Fig.6 shows that the control law is working properly. The data matrix  $Z$  was constructed from the identification data set as:

$$Z = \begin{bmatrix} u_1, u_2, \dots, u_{2000} \\ \theta_1, \theta_2, \dots, \theta_{2000} \end{bmatrix}^T$$

## 5.3 Simulation results for Cluster Validation

Validity measures given in section 3.2 were used with Gustafson-Kessel algorithm to validate the partitioning of the Inverted pendulum data with the current number of clusters. During the optimization parameters were fixed to the following values:  $m=2$ ,  $\epsilon=0.001$ ,  $\rho=1$  for each cluster,  $c \in [2 \ 14]$ . The values of the validity measures depending from the number of cluster are plotted and embraced in table.1.

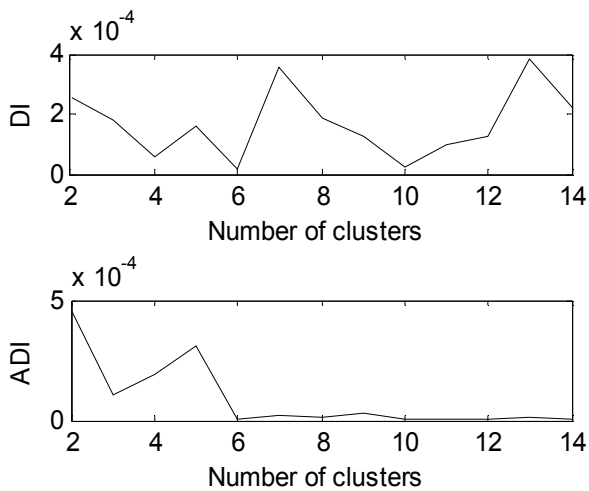


Fig.6 Values of Dunn's Index and Alternative Dunn Index.

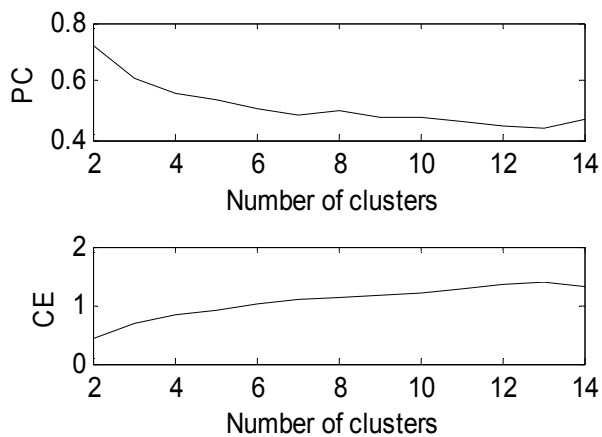


Fig. 7 Values of Partition coefficient and classification entropy.

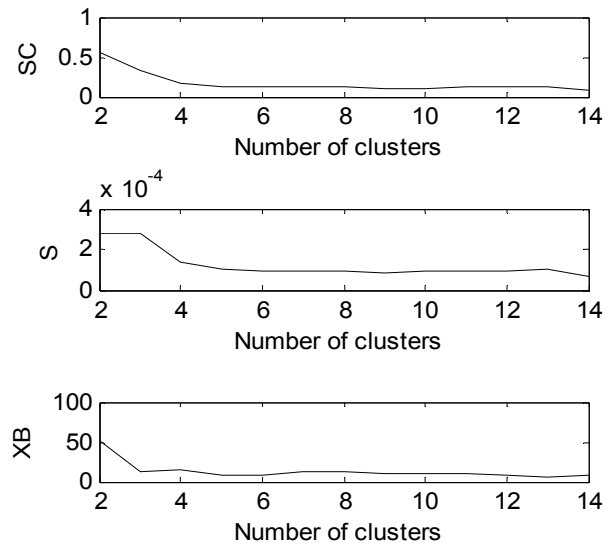


Fig. 8 Values of Partition index, Separation Index and Xie and Beni's Index.

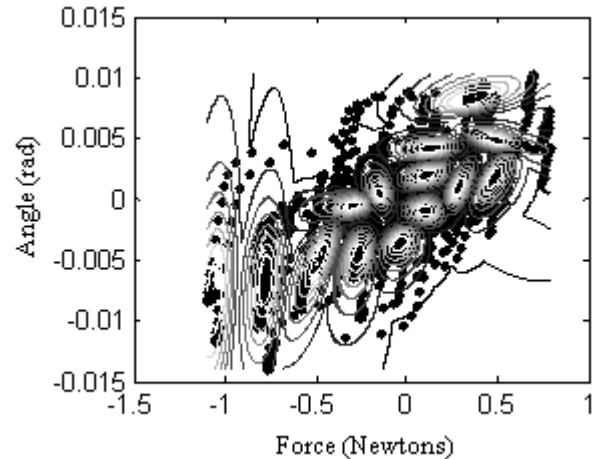


Fig.9. Results showing number of clusters for Gustafson-Kessel algorithm applied to Inverted Pendulum.

From Fig.8 SC and S hardly decreases at  $c=4$  point. The xb index reaches this local minimum at  $c=4$ . considering that SC and S are more useful, when comparing different clustering methods with the same  $c$ , we choose the optimal cluster to 4, which is confirmed by Dunn's index and Alternative Dunn index in Fig.6.

## 5.4 Simulation results implementing Fuzzy C-means algorithm

Initialization parameters of fuzzy c-means algorithm are given as: number of clusters  $c = 4$ , fuzziness of clustering  $m = 2$ , termination tolerance  $\epsilon = 0.001$ . These parameters are fixed by trial and error. Using these initialization parameters, the fuzzy c-means algorithm given in Section 3.3 is simulated. The modeling of inverted pendulum using fuzzy c-means algorithm is shown in Fig. 10. It does not exactly matching with actual process output.

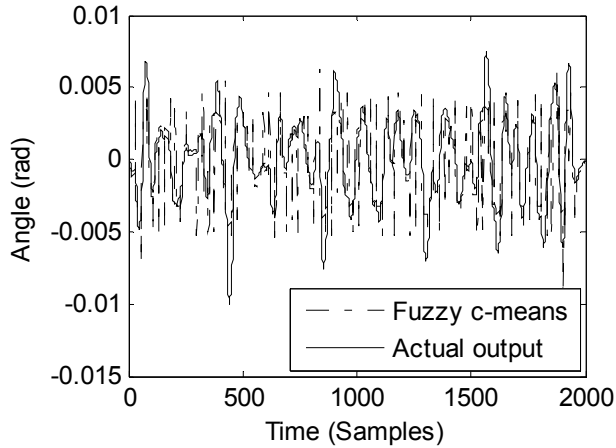


Fig.10 Simulated angle of fuzzy C-means and actual process output.

### 5.5 Simulation results implementing G-K algorithm

Initialization parameters of G-K algorithm are given as: number of clusters  $c=4$ , fuzziness of clustering  $m=2$ , termination tolerance  $\varepsilon = 0.001$ , expected cluster volumes  $\rho=1$ . These parameters are fixed by trial and error. Using these initialization parameters, the G-K algorithm given in Section 3.4 is simulated. The model of inverted pendulum using G-K algorithm is shown in Fig.11. It is evident from the response the model output is exactly matching with the process output.

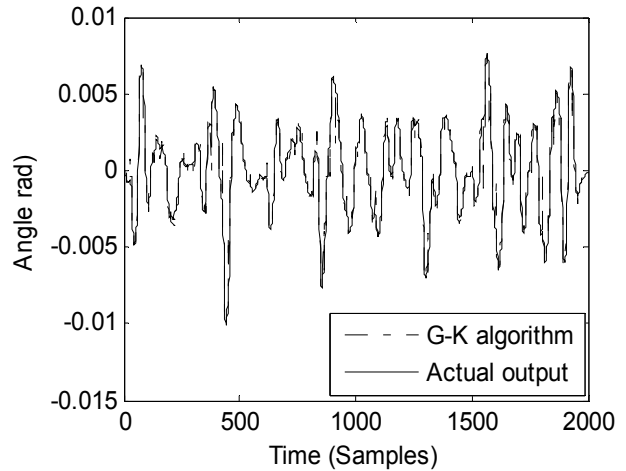


Fig. 11. Simulated angle of G-K algorithm and actual process output.

### 5.6 Simulation results implementing G-G algorithm

Initialization parameters of Gath - Geva algorithm are given as: number of clusters  $c=4$ , fuzziness of clustering  $m=2$ , termination tolerance  $\varepsilon=0.001$ . Using these initialization parameters, the Gath-Geva algorithm given in Section 3.5 is simulated. The modeling of inverted pendulum using Gath - Geva algorithm is shown in Fig. 12. It does not exactly matching with actual process output.

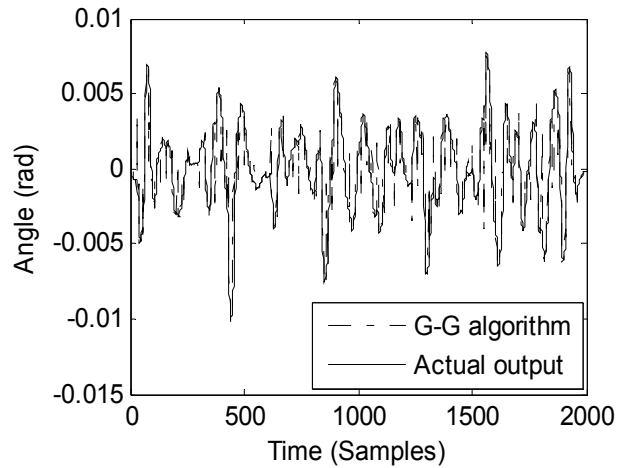


Fig.12 Simulated angle of Gath – Geva algorithm and actual process output.

**Table 1. The numerical values of validity measures.**

No. of Cluster	2	3	4	5	6	7	8	9	10	11	12	13	14
SC	0.802	0.279	0.150	0.118	0.1204	0.125	0.101	0.103	0.103	0.106	0.113	0.113	0.094
PC	0.726	0.609	0.562	0.534	0.5115	0.486	0.472	0.462	0.469	0.453	0.442	0.429	0.440
CE	0.425	0.681	0.824	0.922	1.0156	1.109	1.176	1.227	1.244	1.307	1.357	1.415	1.390
XB	50.71	19.00	7.229	7.843	7.201	10.201	10.181	9.565	5.512	6.790	5.824	4.509	6.905



### 5.7 Comparison of performance of fuzzy c-means, G-K and G-G Algorithm

The model outputs obtained using c-means, G-K and Gath – Geva algorithms are compared with actual output is shown in Fig.10, Fig 11 and Fig.12. From the result and performance table 2, it is observed that G-K algorithm gives better result by minimizing the Root Mean Square Error (RMSE). The performance index measured on the free-run test experiment is given by

$$RMSE = \sqrt{\frac{1}{N} \sum_{k=1}^N (y(k) - \hat{y}(k))^2}$$

where,  $N$  represents the number of input-output data used for validating the model,  $y(k)$  is the actual output and  $\hat{y}(k)$  is the model output.

**Table 2. Performance measure (RMSE) for a inverted pendulum.**

Modeling	Fuzzy c-means	Gath-Geva	G – K algorithm
RMSE	0.0017	0.0011	0.00049598

### 6. CONCLUSION

The design and implementation of fuzzy model based on fuzzy c-means, g-k algorithm and Gath – Geva algorithms for the inverted pendulum are discussed in this paper. The effectiveness of G-K algorithm is verified through simulation studies and the results obtained are compared with that of actual process output. By means of product-space fuzzy c-means, G-K and Gath – Geva clustering, a data set generated by a inverted pendulum system can be partitioned into fuzzy subsets of data that are locally described by linear sub models are demonstrated in this paper. Also different cluster validation methods are discussed and presented to choose optimum number of cluster. From the simulation results it is observed that the G-K clustering gives better performance for the inverted pendulum. Hence, it can be concluded that the proposed G-K algorithm can be applied to decompose any nonlinear system into local linear models and can be used to design model based control techniques such as Internal Model Control.

### 7. REFERENCES

- [1] Abonyi, R. Babuska and F. Szeifert, 2001, Fuzzy modeling with multivariate membership functions: gray-box identification and control design, *IEEE Trans. On Systems*, vol. 31, pp. 755-767.
- [2] R. Babuska, 1998, *Fuzzy Modeling for Control*, Boston, MA: Kluwer.
- [3] T. Takagi and M.Sugeno, 1985, Fuzzy identification of systems and its application to modeling and control, *IEEE Trans. Syst., Man, Cybern.*, vol SMC-15, pp. 116-132.
- [4] M. Sugeno, K. Tanaka, 1991, Successive identification of a fuzzy model and its application to prediction of a complex system, *Fuzzy sets and system*, Vol. 42, pp. 335-344.
- [5] S. Arulselvi, G.Uma and M. Chidambaram, 2004, Design of Takagi-Sugeno fuzzy model based on Gustafson Kessel clustering technique for a dc-dc converter, *4<sup>th</sup> International conference on TIMA*, CSIR complex, Chennai, pp. 221-227.
- [6] M. Sugeno, T. Yasukawa, 1993, A fuzzy logic based approach to qualitative modeling, *IEEE Trans.on fuzzy systems*, Vol. 1, No. 1, pp. 7-31.
- [7] J. Bezdek, hybrid models for fuzzy control, 1994, *Proc. Of the first Int. Symp. On Integrating Knowledge and Neural Heuristics*, pp. 3-14.
- [8] H. Berenji, Y.Y. Chen, C.C. Lee, J.S. Jang, S. Murugesan,1990,A hierarchical approach to designing approximate-reasoning based controllers for dynamical physical systems, *Proc. Of 6<sup>th</sup> Conf. on Uncertaintyin AI*, pp. 362.
- [9] J.C. Bezdek and J.C. Dunn, 1975,Optimal fuzzy partitions:A heuristic for estimating the parameters in a mixture of normal distributions, *IEEE Transactions on Computers*, pp. 835- 838.
- [10] A.Guez, J. Selinsky, 1988, A trainable neuromorphic controller, *Journal of robotic systems*, Vol.5, No.4, pp 363-388.