QUALITATIVE ANALYSIS OF DYNAMICAL SYSTEMS

by
Debi Prasad Pati

under the supervision of
Prof. Kumar Sankar Ray
ELECTRONICS & COMMUNICATION SCIENCES UNIT

a
DISSERTATION
submitted in the partial fulfillment for the degree of
Master of Technology
in
Computer Science

INDIAN STATISTICAL INSTITUTE
203, B.T ROAD.
CALCUTTA - 700035
1997

INDIAN STATISTICAL INSTITUTE
203, B.T ROAD.
CALCUTTA - 700035

Certificate of Approval

Certified that the dissertation named "**Qualitative Analysis of Dynamical Systems**" being submitted by Debi Prasad Pati, as partial fulfillment of the requirements for the degree of M.Tech(Computer Science) is a record of the work of the student, which has been carried out under my supervision.

(**Prof. Kumar Sankar Ray**)
ECSU ,ISI

Acknowledgements

I am happy to express my deepest sense of indebtness and gratitude to my guide Prof. Kumar Sankar Ray for his planning and timely guidance and regular supervision during the entire course of M.Tech dissertation.

I am thankful to Mr. Bijan Mukherjee,Mr.Abul Hussain , Mr.Tapan Kumar Dinda and Mr Abhik Mukharjee for their affection,timely help and advice during my dissertation work.

I express sense of gratitude to Head of Unit, ECSU for his kind provision of facilities during my project.

Contents

1 Introduction

Modelling of dynamical systems is broadly classified into two catagories,i.e quantitative modelling and qualitative modelling.Currently,the most research works are based on the qualitative analysis rather than quantitative analysis because of its dependace on many factors such as nature of the plant(e.g.size ,complexity,nonlinerity,time-varying parameters,dynamic interactions),control objective and specifications, and cost considerations.Graph theoritic analysis of dynamical systems,uncertain system modelling and robust control or the qualitative analysis of interconnected systems have been fields where qualitative, rather than quantitative, results are the main motivation of research.These investigations try to follow the method of a control expert. Experienced control engineers are able to solve their control task even if many details of the system dynamics are not known or deliberately neglected, because their knowledge about the principal behavioural patterns, such as the existence of oscillations, saturation effects or limit cycles, or about the current output of the process in terms of subsets of the state space, rather than accurate qualitative values, is sufficient for many control purposes.

One of the main problems in qualitative modelling is the conservatism of the results.Even for simple examples,such as mass-spring system,qualitative models yield a large set of trajectories.Although it can be proved that this set includes the qualitative description of the real system trajectory,this set also includes many behavioural forms that no physically real dynamical system can perform(spurious solutions).The main reason for this is that the qualitative model is based on inadequate information about the real system, because the quality spaces used are too coarse.

To circumvent this situation is the motivation for incorporating a new transition matrix concept which is capable of including more information about the system.This new kind of qualitative model has the form of nondeterministic or stochastic automata.Here it is assumed that the qualitative value $[x(k)]$ is received by means of a direction-wise quantiser. It can be shown that for a qualitatively given initial state $[x(0)]$, the qualitative system trajectory is ambiguous. As a cosequence of the ambiguities of system performance, nondeterministic and stochastic automata are proposed as a resonable forms of qualitative models.These kinds of models can be used to analyse the qualitative behaviour of the system and, moreover, to design a qualitative controller.Here the work is based on the design of a qualitative controller for a pole balancing problem(inverted pendulum).The design procedure as well as the simulation results are discussed step by step by considering new transition matrix.

2 Finite Automata

A finite automaton is a kind of dynamic system which,at the discrete moments of time under consideration,satisfies the following conditions,
(a) At each of these moments of time,the system subject to an input can be in one of a finite number of possible states.
(b) At these moments of time,the inputs of the system can be chosen from a finite number of possible states.
(c) At any of these moments of time, the state of the system is uniquely defined by the state of the input and state of system at the previous moment of time.

Let the set of possible states of automaton be

$$X = \{x_1, x_2, \ldots, x_n\},$$

the set of possible state of input be

$$U = \{u_1, u_2, \ldots, u_m\},$$

and the set of possible of output be

$$Y = \{y_1, y_2, \ldots, y_q\}.$$

The set X is referred to as the automaton state set,the sets U and Y are referred to as input and output state sets,or input and output alphabet, respectively. The operation of an automaton can be described by the recursive relationship

$$x(k+1) = F[x(k), u(k)], \tag{1}$$

where $x \in X$ and $u \in U$, $x(k+1)$ denotes any state automaton at instant $(k+1)$,and simillarly, $x(k)$ and $y(k)$,denote,the automaton state and the input at instant k respectively,and F is a single-valued function that relates a definite symbol from the automaton set and the other from the input alphabet.The automaton operates on a discrete time scale $t = 1, 2 \ldots k, (k+1), \ldots$.This automton with n internal states is reffered to as an n state automaton.This class of dynamic system is not described by differential equations, nor by difference equations of the general form,but by the equations of the type given in Eqn.1.The output states and the automaton states are related by the output function G,

$$y(k) = G[x(k), u(k)], \tag{2}$$

In a finite automaton,the size of the output alphabet may,but in general doesn't,equal to the number of automaton states. The above discussions define a deterministic finite automaton.It is complete in the sense that every combination of the automaton state and the input state is considered meaningful and produces a known output and next state.It is stationery in the sense the transition function F and the output function G do not depend upon the sampling instant under consideration. It is memoryless in the sense that the present output state and the next automaton state do not depend upon the past inputs,outputs,or automaton states. The transition function F and output function G of an automaton are usually characterised by three basic representations:the transition table, the transition diagram,the transition matrix.The transition table consists of two subtables that display the functional relationships defined in Eqn.1 and Eqn.2 ,as shown in fig 1.

The transition diagram for an n-state finite automaton consists of n circular vertices representing the n states with directed branches connecting these vertices.Each branch indicates the transition from one state to next and is labeled (u_i/y_j),where u_i and y_j represent,respectively,the present input and output during the transition.The transition diagram of a three -state automaton is shown in fig 2.

The transition matrix of an n-state automaton is a $n \times n$ matrix whose ijth entry is the label of the branch b_i^j from the ith state x_i to the jth state x_j if the branch b_i^j exists,and is equal to zero if the branch does not exist. The transition matrix for the three-state automaton is

$$\begin{bmatrix} u_1/y_2 & u_2/y_1 & 0 \\ 0 & u_2/y_2 & u_1/y_1 \\ u_2/y_1 & u_1/u_2 & 0 \end{bmatrix}$$

A finite automton can also be described by the state matrix

$$P_0(u_k) = [p_{ij}(u_k)], i, j = 1, 2, \ldots, n; k = 1, 2, \ldots, m; \tag{3}$$

$x(K)$ \ $U(K)$	$x(K+1)$	$y(K)$
	$u_1, u_2, \cdots u_m$	$u_1, u_2, \cdots u_m$
x_1	Entries selected from set X	Entries selected from set Y
x_2		
x_3		
$\vdots$		
x_n		

fig.1

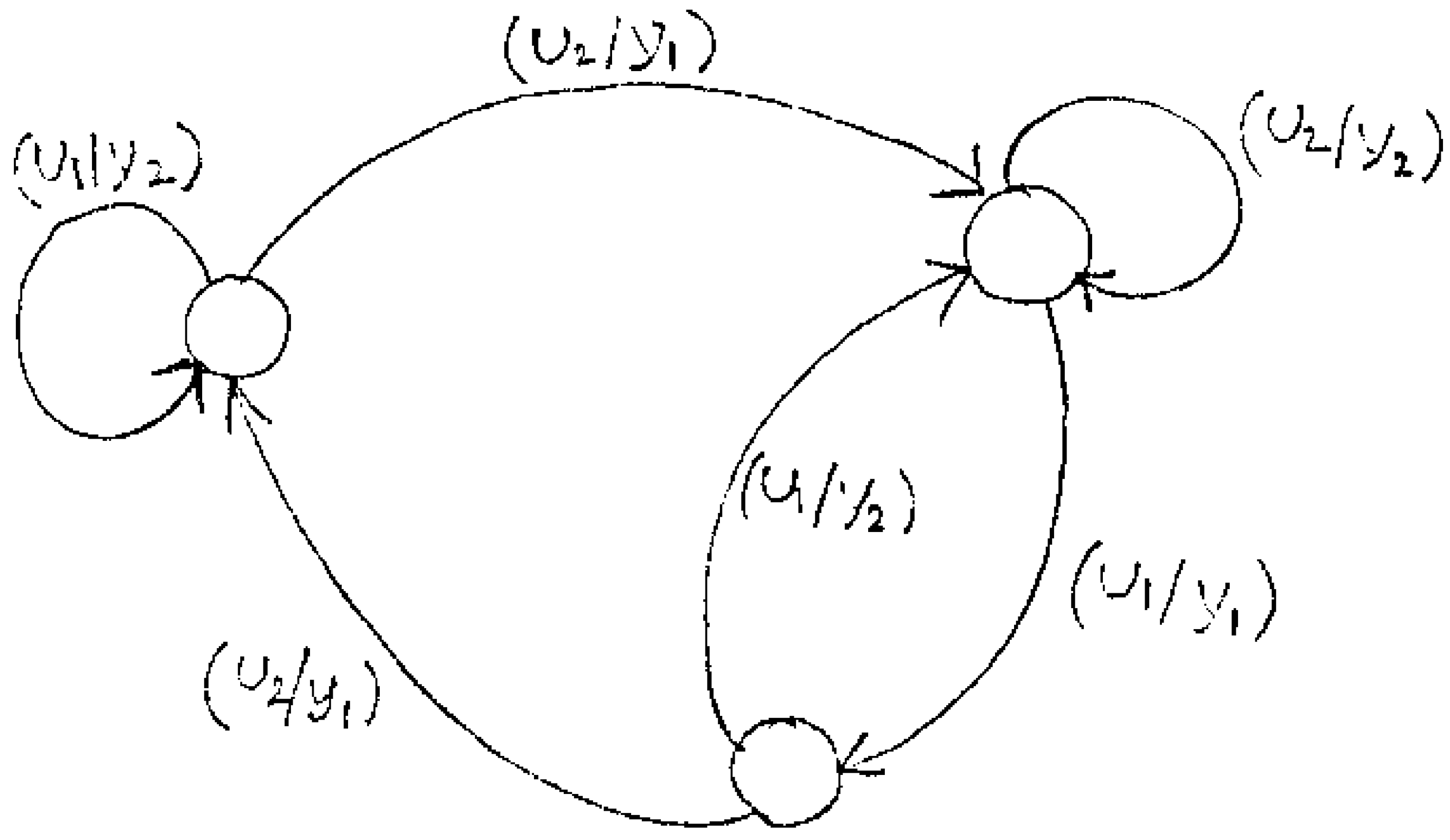


fig.2

where $p_{ij}(u_k)$ is conditional probabilty for the automaton to go from state x_i to stae x_j when the input is u_k.The state matrix $P_0(u_k)$ consists of zeros and ones,where each row of matrix contains one element equal to one for any u_k.These matrices determine the transitions from one state to another.If the automaton is in state x_i and the input u_k is applied,the next state of the automaton will be state x_j provided that

$$p_{ij}(u_k) = 1.$$

Certain automata have the property that starting with any state x_i we may transform the automaton into any other state x_j through an appropriate sequence of input alphabets.Such a automaton is refered to as a strongly connected machine.Irrespetive of the problems involved in analysis of the transition and output responses of given machine for the purpose of information processing,automata may be considered as mathematical models for information processing systems.Study of automata aims at the development of improved schemes for information systems.

3 Characterization of Stochastic Automata

A stochastic automata is a kind of discrete stochastic system that resembles a finite automaton.It possesses a finite set of internal states, can receive a set of input sysmbols, and can generate a set of output symbols.The equations defined for finite automaton may be used to characterize on n-state stochastic automaton, but the transition between states is no longer deterministic.The behaviour of the stochastic automaton is governed partly by its input,and partly by a set of probabilities associated with the internal transition of states of the automaton.For each input, the one step transition from any state x_i to any other state x_j may have a non-zero probabilty.

The probabilties of transition from state x_i to state x_j may be specified by a set $n \times n$ transition matrices,

$$P(u_1), P(u_2), \ldots, P(u_m)$$

associated with each of the input symbols.For input $u \in U$, the tranition matrix is

$$P(u) = \begin{pmatrix} p_{11}^u & p_{12}^u & \cdots & p_{1n}^u \\ p_{21}^u & p_{22}^u & \cdots & p_{2n}^u \\ \vdots & \vdots & \cdots & \vdots \\ p_{n1}^u & p_{n2}^u & \cdots & p_{nn}^u \end{pmatrix} \tag{4}$$

where the elements p_{ij}^u is given by the conditonal probability

$$p_{ij}^u = p(x_i, x_j, u)$$

which represents the transition probability that input u at any instant k will cause the automaton to go from internal state x_i at this instant to state x_j at instant $(k+1)$.If the application of input u cannot possibly cause this transition, then $p_{ij}^u = 0$. On the other hand, if x_j is the only state that can be reached from state x_i in one step, then $p_{ij}^u = 1$.In the general case,

$$0 \leq p_{ij}^u \leq 1.$$

Since the transition probabilities have the property that

$$\sum_{j=1}^{n} p_{ij}^u, i = 1, 2, \ldots, n.$$

the transition matrix $P(u)$ is a stochastic matrix, and it describes a Markov chain.The probabilistic description of the output states $y \in Y$ also requires set of output probability matrices,

$$\{Q(u_1), Q(u_2), \ldots, Q(u_m)\}$$

associated with each of the input sysmbols. The output probability matrices do not affect the internal states and do not enter into more than one step transition chain.

The stochastic matrix has the following properties,
(1) the elements are non-negative and not greater than one and
(2) the rows of this matrix sum to one.
It is noted that, if in each row of the stochastic matrix $P(u)$, there is one only one non-zero element, then this element is equal to one.Such a stochastic matrix degenerates into a state matrix.In this situation the sum of these matrices,

$$P(u_1) + P(u_2) + \ldots + P(u_m)$$

can be reduced to the transition matrix of a finite automaton in skeleton matrix form, if each of the non-zero sum terms is equal to one. In analogy to the characterization of a finite automata, a stochastic automaton may be represented by a probabilistic transition table,a set of transition diagrams or a set of transition matrices.The probabilistic transition table consists of two subtables that display the stochastic matrices and the output matrices for each input alphabet.The transition diagram of an n-state stochastic automaton for specified input-output relationship can be readily be derived from the corresponding transition matrix.Each branch is labelled with conditional probability p_{ij}.A stochastic automaton with m input symbols can be characterized by m transition diagrams. With each input u, the transition is given by Eq.4.The element p_{ij}^{u} is the transition probability that an input u will cause the automaton to go from state x_i to state x_j.A stochastic automaton with u input sumbols can be characterized by m transition matrices, $P(u_1), P(u_2), \ldots, P(u_m)$.The transition matrix for sequence of inputs can be determined by matrix multiplication. If input u is applied for q times, then the q-step transition matrix is $[P(u)]^q$.A stochastic automaton with constant inputs are called autonomous stochastic automaton,which can be interpreted as a Markov chain with the same state set.A Markov chain is said be ergodic if the associated transition diagram is strongly connected.An ergodic chain is said to be regular if there is a positive integer t_0 such that for any x_i and x_j there is an arc progression from x_i to x_j having precisely t arcs for every $t \geq t_0$. The transition matrix for an autonomous stochastic automaton may be reduced to a canonical form, indicating submatrices of closed sets.The reduction process may be continued until the set of all irreducible states is found.

4 Qualitative state measurement of continuous variable system and its characteristics

Consider a linear discrete-time continuous variable system,for which the state equation is defined as

$$x(k+1) = Ax(k) + Bu(k), \quad x(0) = x_0 \tag{5}$$

where $x = (x_1, x_2, \ldots, x_n)'$ and $u = (u_1, u_2, \ldots, u_m)'$ denote the vectors of the systems state or input variables, respectively. The prime is the symbol for vector transposition.A and B are matrices of appropriate dimensios with constant elements.For every given input sequence

$$U = (u(0), u(1), \dots, u(T-1))$$

with fixed observation horizon T, the system in Eqn.5 has the trajectory

$$X(x(0), U) = (x(0), x(1), \dots, x(T)) \tag{6}$$

where

$$x(k) = A^k x(0) + \sum_{l=0}^{k-1} A^{k-l-1} B u(l) \tag{7}$$

holds.

Now, it is assumed the state $x(k)$ cannot be measured quantitatively but is quantised by a direction-wise quantiser that maps the state variables into a set of intervals.The intervals are bounded by given values.

$$g_{ij} \ (i = 1, 2, \dots, n \ ; j \in F_i) \text{ where } F_i = \{f_i^-, f_i^- + 1, \dots, f_i^+\}$$

and defined independently for all components x_i of the state vector x.Hence , the state variables x_i that belongs to the same set

$$Q_{x_i}(z_i) = \{x_i / g_{i,z_i} \leq x_i < g_{i,z_i+1}\} \tag{8}$$

are qualitatively equivalent and represented by the same qualitative value $[x_i(k)] = z_i$. The sets $Q_{x_i}(z_i)$ are defined for

$$z_i \in F_i - \{f_i^+\}$$

The quantised state vector $[x(k)]$ is given by

$$x[k] = (x_1[k], x_2[k], \dots, x_n[k])' \tag{9}$$

where $[x_i(k)] = z_i$ holds if, and only if, $x_i(k) \in Q_{x_i}(z_i)$.

The set $Q_{x_i}(z_i)$ defined in Eqn.8 can be written as

$Q_{x_i}(z_i) = \{x_i / [x_i] = z_i\}$

For all

$$z \in Z_x = \{(z_1, z_2, \dots, z_n)' / z_i \in F_i - \{f_i^+\}\} \subseteq Z^n$$

$Q_z(z)$ can be defined by

$$Q_z(z) = Q_{z_1}(z_1) \times Q_{z_2}(z_2) \times \ldots \times Q_{z_n}(z_n) = \{x/[x] = z\}$$

(for $n = 2, z$ represents a rectangular area, otherwise it is a hyperbox of dimension n) with $\times$ denoting the cartesian product.All these sets together cover the subspace $R_z \subseteq R^n$

$$R_z = \bigcup_{z \in z_z} Q_z(z)$$

For $g_{i,f_i^-} = -\infty$ and $g_{i,f_i^+} = \infty$, the whole state space R^n is partitioned into the sets $Q_z(z) : R = \bigcup_{z \in Z_z} Q_z(z)$

In the case of equidistant intervals with interval lenth q_{z_i}

$$g_{ij} = (j - 1/2)q_{z_i}, j \in Z$$

holds. Then $Z = Z^n$ and $R^n = \bigcup_{z \in Z^n} Q_z(z)$ follow.

It is further assumed that input u_i can assume one of a set of given values u_i^j

$$u_i(k) \in Z_{u_i} = \{u_i^j / j \in G_i\}$$

where $G_i = \{g_i^-, g_i^- + 1, \ldots, g_i^+\}$.

Therefore,the element j of Z_{u_i} is the qualitative value of $u_i(k)$,i.e.

$$[u(k)] = ([u_1(k)], [u_2(k)], \ldots, [u_m(k)])'$$

where $[u_i(k)] = j$ holds if, and only if, $u_i(k) = u_i^j$ .HEnce, $[u(k)]$ belongs to the set Z_u:

$$z \in Z_u = \{(z_1, z_2, \ldots, z_m)' / z_i \in G_i\} \subseteq Z^n$$

The qualitative input sequence $[U]$ is represented by

$$[U] = ([u(0)], [u(1)], \ldots, [u(T-1)]) \quad (10)$$

The qualitative trajectory of the system in Eqn.5 is given by

$$[X(x(0), U)] = ([x(0)], [x(1)], \ldots, [x(T)]) \quad (11)$$

But, for a given initial state $x(0)$, the system defined by Eqn.5 has a unique qualitative trajectory $[X]$.Hence the nondeterminism of the qualitative behaviour is discussed below.

For the qualitative model, only the qualitative values of $[x(k)]$ and $[u(k)]$ are taken into account.The chosen initial state $x(0)$ belongs to the set $Q_x(z(0)$ for some given $z(0)$:

$$x(0) \in Q_x(z(0)) \tag{12}$$

The system input is described by a qualitative input sequence

$$V = (v(0), v(1), \ldots, v(T-1))$$

where

$$[u(k)] = v(k) \text{ for } k = 0, 1, 2, \ldots, (T-1) \tag{13}$$

holds.

It is obvious that the system defined by Eqn.5, if starts from some $x(0)$ given in Eqn.12 under the control sequence described by V and Eqn.13, then the qualitative trajectories formed from the set

$$\overline{X}(z(0), V) = \{X(x(0), U)/x(0) \in Q_x(z(0)), [U] = V\} \tag{14}$$

The model has to generate the qualitative trajectories that result from the set $\overline{X}$ and from the set $[\overline{X}]$:

$$[\overline{X}(z(0)), V)] = \{[X]/x \in \overline{X}\}$$

It has been shown that for an autonomous system i.e.Eqn.5 with $u(k) = 0$ and equidistant quantisation, the set $[\overline{x}]$ is, in general, not a singleton but has more than one element.In order to to extend this result to the class of systems in Eqn.5 considered here, the sets

$$M_x(0) = Q_x(z(0))$$

$$M_x(k) = \{Ax + Bu/x \in M_x(k-1), [u] = v(k)\} \tag{15}$$

are defined and qualitatively described by

$$[M_x(k)] = \{[x]/x \in M_x(k)\} \tag{16}$$

Obviously, the system in Eqn. 5 has a unique qualitative trajectory if, only if, $M_x(k) \subseteq Q_x(z)$ for some $z \in Z_x$ holds for all $k = 0, 1, \ldots, T$.

5 Qualitative Modelling by means of Stochastic Automata

The non-determinisim of the qualitative trajectory of the system in Eq.5 suggests the use of non-deterministic or stochastic automata as a qualitative model.Here the qualitative modelling by stochastic automaton is presented. First,the stochastic automaton $S(Z_x, Z_u, P, z(0), V)$ is considered, where Z_x denotes the set of states, Z_u the set of inputs,

$$P : Z_x \times Z_x \times Z_u \rightarrow R$$

represents the transition probability function $P(z, \overline{z}, v)$ is the probabilty that automaton which has, at time k, the state z and gets the input v, goes to the state $\overline{z}$ at time $(k+1)$, $z(0)$ the initial state and V the input sequence.

Now it is required to define a transition function H in terms of probability probability function P such that the automaton generates the set $\overline{X}$ of qualitative trajectories of the system in Eq.5.In order to achieve this, the following sets are introduced.

$$H(z, v) = \{\overline{z}/P(z, \overline{z}, v) \neq 0\} \tag{17}$$

$$M_z(0) = \{z(0)\} \tag{18}$$

$$M_z(k+1) = \{H(z, v(k))/z \in M_z(k)\}, \text{ for } k = 0, 1, \ldots, (T-1). \tag{19}$$

The performance of the stochastic automaton is described by the probability $q(z, k)$ with which the automaton is at time k in state z.

Let us define $q(k)$ as a row vector

$$(q(z_1, k), q(z_2, k), \ldots, q(z_r, k))$$

where r is the number of elements of Z_x and $P(v(k))$ is a $r \times r$ matrix which represents the probability function P for a given input $v(k)$ and is also called as transition matrix.The following relations for the automaton that has initial value $z(0) = z_0$:

$$q(z_0, 0) = 1 \text{ and } q(z, 0) = 0 \text{ for all } z \neq z_0$$

$$q(z, k+1) = \sum_{\overline{z} \in Z_x} q(\overline{z}, k) P(\overline{z}, z, v) (\text{ for } k = 0, 1, \ldots, T-1)$$

In matrix form it can be written as

$$q(k+1) = q(k)P(v(k)) \tag{20}$$

and,obviously,

$$M_z(k) = \{z/q(z,k) \neq 0\}$$

holds.The set of trajectories of stochastic automaton is given by

$$\overline{Z}(z(0),V) = \{(z(0),z(1),\ldots,z(T)/z(k+1) = H(z(k),v(k))\} \tag{21}$$

The stochastic automaton yields a better characterization of the qualitative performance of the system in Eq.5, since it generates,together with each set $M_z(k)$ a weighting function $q(z,k)$ that describes the probability of the state $z \in M_z(k)$ being really assumed by the system in Eq.5.

The additional characterization of the states of the qualitative model by the probability $q(z,k)$ makes it possible to reduce the set $\overline{Z}(z(0),V)$.If $q(z,k)$ has a low value, the state z can be assumed not to belong to the qualitative trajectory of the system in Eq.5, but to spurious solution.Therefore such states can be deleted.
Considering the Eq20,iteratively,it can be written as

$$q(k+1) = q(0)P(v(0))P(v(1))\ldots P(v(k)) \tag{22}$$

If $v(k)$ is constant for all k, then

$$q[k] = q(0)[P]^k \tag{23}$$

This above formula is used for calculating the performance or state probabilities $q(k)$ of stochastic automaton with initial state probabilities specified.The evaluation $q(k)$ requires the determination of $[p]^k$, may be specified by making use of the z-transform analysis. Taking the z-transform of Eq.20 with $v(k)$ is constant for all k,yields

$$z[Q(z) - q(0)] = Q(z)P,$$

On rearranging the above equation,the z-transform of the state probability vector is

$$Q(z) = q(0)[I - z^{-1}P]^{-1} \tag{24}$$

Let $Y(z) = [I - z^{-1}P]^{-1}$ and inverse transform of $Y(z)$ be y(k). Then the inverse transform of Eq.24 may be written as

$$q(k) = q(0)y(k). \tag{25}$$

By comparing with Eq.23, we have

$$y(k) = [P]^k,$$

which provides a convenient way to calculate the kth power of the stochastic matrix. The matrix $y(k)$ may be refered to as response matrix.The ijth entry of the response matrix represents the probability that the stochastic automaton will go to state x_j at instant k, given that it was in state x_i at time $k = 0$.The response matrix may be broken into two parts:the steady-state response matrix, and the transient response matrix:

$$y(k) = a + b(k). \tag{26}$$

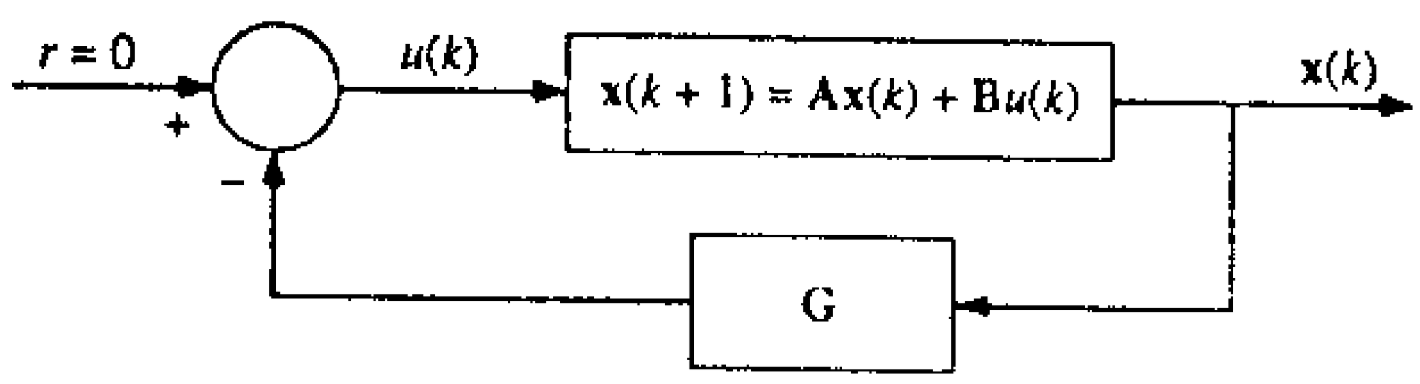
The steady-state response matrix a is independent of k.For nonrecurrent and simple recurrent stochastic automaton,the rows of matrix a are identical, and each will yield the limiting state probability vector of the automaton, since,

$$\sum_{i=1}^{n} q_i(0) = 1.$$

In the case of multiple recurrent stochastic automaton, the rows of matrix a are no longer equal, and the limiting state probability vector is now dependent upon how the automaton was started.If the automaton was started in the ith state, the ith row of the steady-state response matrix represents the limiting state probability vector.A recurrent stochastic automaton is characterized by one with a set of persistent states that are connected in such a way that the automaton makes jumps whithin this set of states indefinitely but never jumps outside the set.The transient response matrix $b(k)$ represents the transient behaviour of the automaton.It vanishes as k approaches infinity. The sum of the elements of matrix $b(k)$ in each row is zero.This is true, since the elements in row may be considered as perturbations applied to the limiting state probability vector.The elements of the ith row of matrix $b(k)$ determine the set of transient components of the state probability vector if x_i is the starting state.

6 Design of Qualitative Feedback Controller for Inverted Pendulum by New transition matrix

Here,our main motivation is to stabilise an "Inverted Pendulum" (shown in fig.5) by pushing the vehicle resonably to the left or to the right i.e by designing an appropriate feedback controller such that it can be stabilised. If the angular displacement and angular velocity can be measured precisely, then systems theoritical approach i.e quantitative approach to the control problem is resonable.First,a discrete-time model is set up and the feedback controller $u(k) = -Gx(k)$ (shown in fig.4) can be designed by the known methods.



A linear digital system with state feedback.

fig.4

But, this type of quantitative feedback controller does not take into account deterioration of the closed-loop system performance in case of bad sensor information and nonlinearities present in the system dynamics.

For the above reason, we are going to propose a method for designing a qualitative feedback controller that stabilises the unstable system in Eq.5, taking qualitative measurements of state variables,discussed in the qualitative model proposed in the previous section.The main idea of this section is as follows.

Without loss of generality, we can assume that the equilibrium state of the system in Eq.5 is given by $x = 0$ and $z = [x] = z_{eql}$. Therefore,the aim of stabilising the system is to find a qualitative controller

$$[u(k)] = g([x(k)]) \tag{27}$$

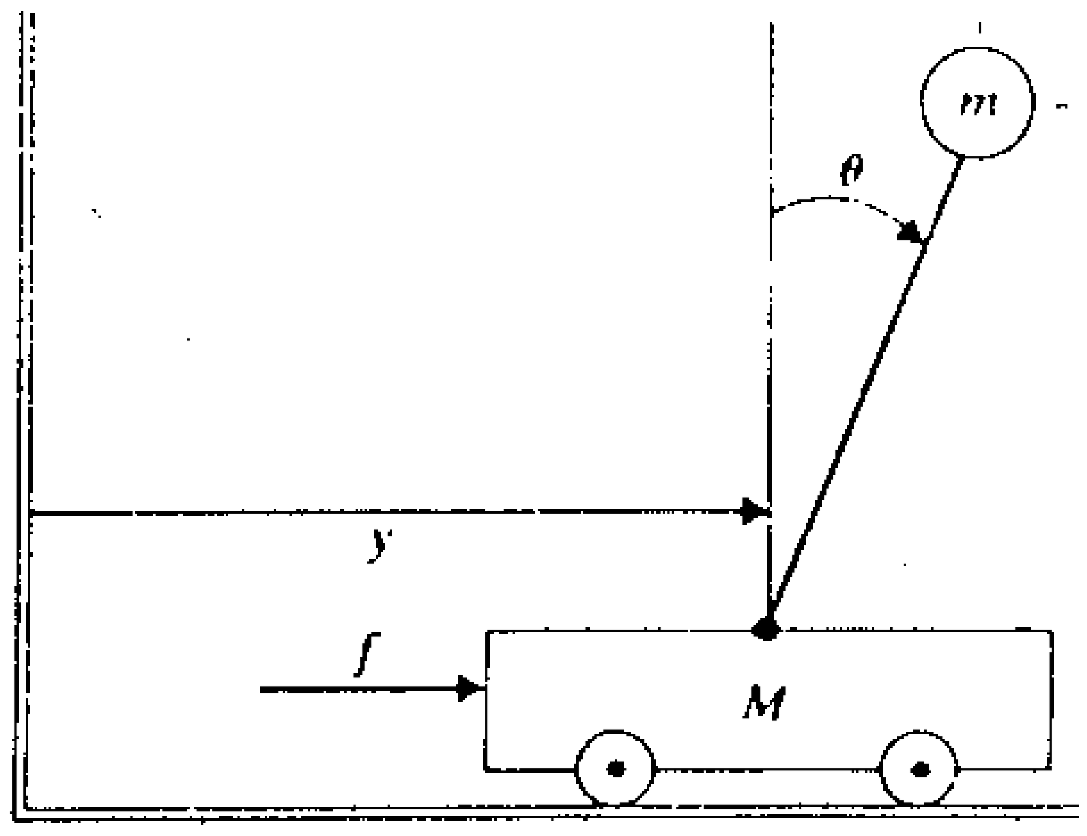
that moves the system into the equilibrium state. Since only the qualitative state $[x]$ is available,the system cannot be asymptotically stabilised as would be possible with a quantitative controller $u(k) = g(x(k))$ where $g(x(k)) = -Gx(k)$.Therefore the control aim is to hold the system in the surrounding of the equilibrium state.The control law f has to be chosen so that the probability $q(z_e ql, k)$ of the stochastic automaton, in connection with control sequence that results from Eqn.27 is maximised. Now the qualitative model of the closed loop system is given by $S_f(Z_x, Z_u, P_f, z(0), z_{eql})$ with $P_f = (p_{fij}), p_{fij} = p_{ij}(f(z_f))$

where $p_{ij}(v_k)$ is the ijth element of the matrix $P(v_k)$.

Due to the severe measurement errors,the above said qualitative approach is resonable,where the quality of the sensor data can explicitly be taken into consideration by using appropriate quantisation of the angular displacement and the angular velocity.The following are the steps to design the qualitative feedback controller for Inverted Pendulum.

6.1 Mathematical Modelling of Inverted Pendulum and its State Space Representation

The Inverted Pendulum on moving cart is shown below.



Inverted pendulum on moving cart.

fig.5

It is observed that the motion of the system is uniquely defined by the displacement of the cart from some reference point,and the angle that pendulum rod makes with respect to the vertical.Hence the system has only two degrees of freedom.

Considering the potential and kinetic energy of the above system and applying Lagranges's equations, we have

$$\begin{aligned} (M+m)\ddot{y} + ml\cos\theta\ddot{\theta} - ml\dot{\theta}^2\sin\theta = f \\ ml\cos\theta\ddot{y} - ml\sin\theta\dot{y}\dot{\theta} + ml^2\ddot{\theta} - mgl\sin\theta = 0 \end{aligned} \tag{28}$$

where
f=the force on the vehicle(moving cart)i.e input signal,
y=position of the vehicle,
θ=angular displacement,
m=mass of the bob,
M=mass of the vehicle,
and lenth of the pole.

These are the exact equations of motion of the inverted pendulum on a cart as shown in fig.5.They are nonlinear owing to the presence of the trigonometric terms $\sin\theta$ and $\cos\theta$ and the quadratic terms $\dot{\theta}^2$ and $\dot{y}\dot{\theta}$.If the pendulum is stabilised,however,then θ will be kept small.This justifies the approximations,

$$\cos\theta = 1 \text{ and } \sin\theta = 0.$$

We may also assume that $\dot{\theta}$ and $\dot{y}$ will be kept small, so the quadratic terms are negligible.Using these approximations, we obtain the linearised dynamic model

$$\begin{aligned} (M+m)\ddot{y} + ml\ddot{\theta} = f \\ m\ddot{y} + ml\ddot{\theta} - mg\theta = 0 \end{aligned} \tag{29}$$

A state variable representation coressponding to Eq.29 is obtained by defining the state vector

$$X = [x_1, x_2, x_3, x_4]'$$

Then

$$x_1 = \dot{y} \text{ and } x_2 = \dot{\theta}$$

constitute the first two dynamic equations and on solving Eq.29,for $\ddot{y}$ and $\ddot{\theta}$, we obtain two more equtions

$$x_3 = \ddot{y} = \frac{f}{M} - \frac{mg}{M}\theta$$
$$x_4 = \ddot{\theta} = -\frac{f}{Ml} + \frac{(M+m)}{Ml}g\theta$$

The four equations can be put into the standard matrix form

$$\dot{X} = AX + BU \tag{30}$$

where

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{mg}{M} & 0 & 0 \\ 0 & \frac{(M+m)g}{Ml} & 0 & 0 \end{bmatrix} \text{ and } B = \begin{bmatrix} 0 \\ 0 \\ \frac{1}{M} \\ -\frac{1}{Ml} \end{bmatrix}$$

The corresponding discrete-data model with sample and hold devices of the system given by Eq. 5 is

$$X((k+1)T) = \Phi(T)\, X(kT) + \Theta(T)U(kT) \tag{31}$$

where

$$\Phi(T) = \mathcal{L}^{-1}[(sI - A)^{-1}]\,|_{t=T} \text{ and } \Theta(T) = \int_0^T \Phi(T-\tau)B\, d\tau$$

With the following parameter values

$$m = 0.1Kg, M = 1.0Kg$$
$$l = 0.5m, g = 9.81m/sec^2$$

and the sampling period of 0.2 sec,the discrete model of Eq.30 is

$$X(k+1) = \begin{bmatrix} 1 & 0.02 & -0.001983 & 0.0 \\ 0.0 & 0.0 & 1.00425 & 0.019278 \\ 0.0 & 1.0 & -0.0196569 & -0.0019483 \\ 0.0 & 0.0 & 0.4160 & 1.00425 \end{bmatrix} X(k) + \begin{bmatrix} 0.0 \\ 0.0 \\ 0.266 \\ -0.04 \end{bmatrix} U(k) \tag{32}$$

6.2 Determination of unstable poles .

In order to design a qualitative controller,for stabilising inverted pendulum,we consider two state equations.

$$\begin{bmatrix} x_2(k+1) \\ x_4(k+1) \end{bmatrix} = \begin{bmatrix} 1.00425 & 0.019278 \\ 0.4160 & 1.00425 \end{bmatrix} \begin{bmatrix} x_2(k) \\ x_4(k) \end{bmatrix} + \begin{bmatrix} 0.0 \\ -0.04 \end{bmatrix} u(k) \tag{33}$$

It is required to know the location of poles of above discrete-data open loop system because its stability depends on the presence of the poles inside the unit circle in the z-plane(z-transform).The poles are nothing but the roots of the characterstic equation $| zI - A |$ of discrete model.Hence for the Eq.33, the poles are 1.094 and 0.91 which represents that the open loop system is unstable.Hence there is a requirement of feedback controller to stabilise the system.

6.3 Determination of Quantised Working space in phase plane plot

To define the control law,for qualitative feedback controller,the total working space (phase plane plot) is to be represented by some quantised regions.These regions are bounded by previously defined $g_{i,j}$s'.But the value of $g_{i,j}$ depends on some measurement constraint and simulation study. Therefore some $g_{i,j}$ are taken as fixed values due to measurement constraints and for others,initially,resonable values are taken.Then the control law f is defined and the $g_{i,j}$s' are adjusted from simulation study.

Here ,each state variable in Eq.33 is represented by three quantised regions. Hence 10 regions are defined by boundaries $g_{i,j}$, for $i \in \{0,1\}$ and $j \in \{-3,-2,2,3\}$.

Owing to the measurements insensitivity of 0.0175(1^0) for θ and 0.0175(1^0) per sampling period for $\dot{\theta}$, the bounds g_{ij} for $i \in \{0,1\}$ and $j \in \{-2,2\}$ are fixed :

$$g_{0,-2} = -0.0175, g_{0,2} = 0.0175$$
$$g_{1,-2} = -0.0175, g_{1,2} = 0.0175$$

and the rest are set with resonable values to define the working space:

$$g_{0,-3} = -0.2, g_{0,3} = 0.2$$
$$g_{1,-3} = -0.8, g_{1,3} = 0.8.$$

These values are adjusted if possible from simulation study.

INVERTED PENDULUM , SIMULATION RESULTS:-
Developed by D.P.Pati

Representation of Control Law
in Working space

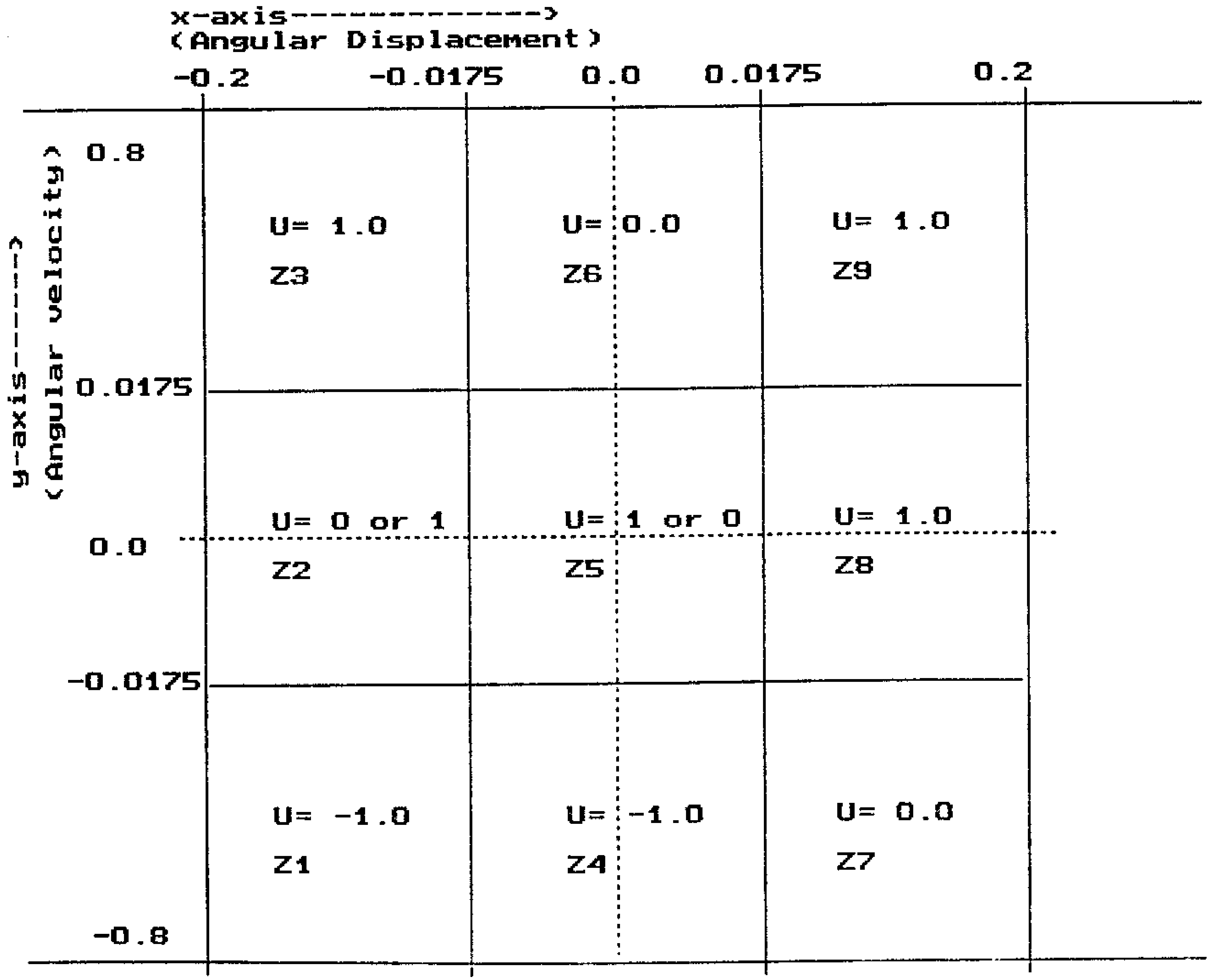


fig.6

Now the temporary working space which includes 9 regions, are numbered as :

$$z_1 = (-3, -3), z_2 = (-3, -2), z_3 = (-3, 2),$$
$$z_4 = (-2, -3), z_5 = (-2, -2), z_6 = (-2, 2),$$
$$z_7 = (2, -3), z_8 = (2, -2), z_9 = (2, 2).$$

and is shown in fig.6.

The control law f is to be defined over these regions except z_{10},the outside region of the temporary working space.

6.4 Determination of control law by New Transition Matrix

Before determining the control law,we have to know the method to determine the new transition matrix which as follows,

Suppose, z_i for $i = 1, 2, \ldots, n-1$, represent the region in the working space in which the system is stable and z_n represents region outside the working space, basically, an unstable region. Then new transition matrix $P(v(k))$ is $n \times n$ matrix where each element $p(z_i, z_j, v(k))$ represents the probability that system which is at the region z_i,after getting qualitative input v(k),goes to the region z_j by one-step transition.The transition is determined by the discrete-open loop state equations. The steps to determine the new transition matrix are

(1) For each $x(0) \in z_i$, calculate one-step transition by

$$x(1) = Ax(0) + Bv(k).$$

(2) If $x(1) \in z_j$, then c_j is updated as $c_j \leftarrow c_j \cup \{x(0)\}$, where c_j represents the set corresponds to z_j region.Initially, $c_j \leftarrow \Phi$ for $j = 1, 2, \ldots, n$.

(3)Repeat step (1) and step (2) until the total set z_i gets exhausted.

(4) Calculate

$$p(z_i, z_j, v(k)) = \frac{| c_j |}{| z_i |}$$

NEW TRANSITION MATRIX (WITHOUT CONSIDERING A FORCING FUNCTION)

INPUT 'U'=0.000000 (OPEN LOOP SYSTEM)

$$P(u) = \begin{bmatrix} 0.780 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.220 \\ 0.850 & 0.050 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.100 \\ 0.070 & 0.050 & 0.870 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.010 \\ 0.260 & 0.000 & 0.000 & 0.660 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.080 \\ 0.030 & 0.040 & 0.000 & 0.070 & 0.840 & 0.020 & 0.000 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 & 0.050 & 0.780 & 0.000 & 0.000 & 0.170 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.100 & 0.000 & 0.000 & 0.880 & 0.020 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 & 0.040 & 0.000 & 0.000 & 0.110 & 0.850 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.970 & 0.030 \\ 0.000 & 0.000 & 0.042 & 0.000 & 0.000 & 0.000 & 0.050 & 0.000 & 0.000 & 0.907 \end{bmatrix}$$

INPUT 'U'=1.000000 (OPEN LOOP SYSTEM)

$$P(u) = \begin{bmatrix} 0.370 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.630 \\ 0.900 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.100 \\ 0.540 & 0.050 & 0.400 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.010 \\ 0.100 & 0.000 & 0.000 & 0.400 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.500 \\ 0.070 & 0.000 & 0.000 & 0.930 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.470 & 0.000 & 0.360 & 0.030 & 0.000 & 0.140 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.050 & 0.000 & 0.000 & 0.500 & 0.000 & 0.000 & 0.450 \\ 0.000 & 0.000 & 0.000 & 0.040 & 0.000 & 0.000 & 0.960 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.420 & 0.030 & 0.550 & 0.000 \\ 0.000 & 0.000 & 0.077 & 0.000 & 0.000 & 0.007 & 0.007 & 0.000 & 0.092 & 0.815 \end{bmatrix}$$

fig.7

INPUT 'U'=-1.000000 (OPEN LOOP SYSTEM)

$$P(u) = \begin{bmatrix}
0.590 & 0.030 & 0.280 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.100 \\
0.000 & 0.000 & 0.900 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.100 \\
0.000 & 0.000 & 0.650 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.350 \\
0.240 & 0.000 & 0.070 & 0.360 & 0.000 & 0.330 & 0.000 & 0.000 & 0.000 & 0.000 \\
0.000 & 0.000 & 0.070 & 0.000 & 0.000 & 0.930 & 0.000 & 0.000 & 0.000 & 0.000 \\
0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.550 & 0.000 & 0.000 & 0.050 & 0.400 \\
0.000 & 0.000 & 0.000 & 0.060 & 0.000 & 0.040 & 0.460 & 0.040 & 0.400 & 0.000 \\
0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.040 & 0.000 & 0.000 & 0.960 & 0.000 \\
0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.520 & 0.480 \\
0.095 & 0.000 & 0.000 & 0.005 & 0.000 & 0.000 & 0.085 & 0.002 & 0.005 & 0.807
\end{bmatrix}$$

CLOSED LOOP SYSTEM

$$P(u) = \begin{bmatrix}
0.590 & 0.030 & 0.280 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.100 \\
0.850 & 0.050 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.100 \\
0.540 & 0.050 & 0.400 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.010 \\
0.240 & 0.000 & 0.070 & 0.360 & 0.000 & 0.330 & 0.000 & 0.000 & 0.000 & 0.000 \\
0.070 & 0.000 & 0.000 & 0.930 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \\
0.000 & 0.000 & 0.000 & 0.000 & 0.050 & 0.780 & 0.000 & 0.000 & 0.170 & 0.000 \\
0.000 & 0.000 & 0.000 & 0.100 & 0.000 & 0.000 & 0.880 & 0.020 & 0.000 & 0.000 \\
0.000 & 0.000 & 0.000 & 0.040 & 0.000 & 0.000 & 0.960 & 0.000 & 0.000 & 0.000 \\
0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.420 & 0.030 & 0.550 & 0.000 \\
0.000 & 0.000 & 0.042 & 0.000 & 0.000 & 0.000 & 0.050 & 0.000 & 0.000 & 0.907
\end{bmatrix}$$

fig.7

NEW TRANSITION MATRIX (CONSIDERING FORCING FUNCTION)

INPUT 'U'=0.000 (OPEN LOOP SYSTEM)

$$p(u) = \begin{bmatrix} 0.860 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.140 \\ 0.950 & 0.050 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \\ 0.080 & 0.050 & 0.870 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \\ 0.260 & 0.000 & 0.000 & 0.660 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.080 \\ 0.030 & 0.040 & 0.000 & 0.070 & 0.840 & 0.020 & 0.000 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 & 0.050 & 0.780 & 0.000 & 0.000 & 0.170 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.100 & 0.000 & 0.000 & 0.880 & 0.020 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 & 0.040 & 0.000 & 0.000 & 0.110 & 0.850 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.970 & 0.030 \\ 0.047 & 0.000 & 0.115 & 0.000 & 0.000 & 0.000 & 0.132 & 0.000 & 0.060 & 0.645 \end{bmatrix}$$

INPUT 'U'=1.000 (OPEN LOOP SYSTEM)

$$p(u) = \begin{bmatrix} 0.400 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.600 \\ 1.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \\ 0.550 & 0.050 & 0.400 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \\ 0.100 & 0.000 & 0.000 & 0.400 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.500 \\ 0.070 & 0.000 & 0.000 & 0.930 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.470 & 0.000 & 0.360 & 0.030 & 0.000 & 0.140 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.050 & 0.000 & 0.000 & 0.500 & 0.000 & 0.000 & 0.450 \\ 0.000 & 0.000 & 0.000 & 0.040 & 0.000 & 0.000 & 0.960 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.420 & 0.030 & 0.550 & 0.000 \\ 0.045 & 0.012 & 0.112 & 0.000 & 0.000 & 0.007 & 0.070 & 0.000 & 0.147 & 0.605 \end{bmatrix}$$

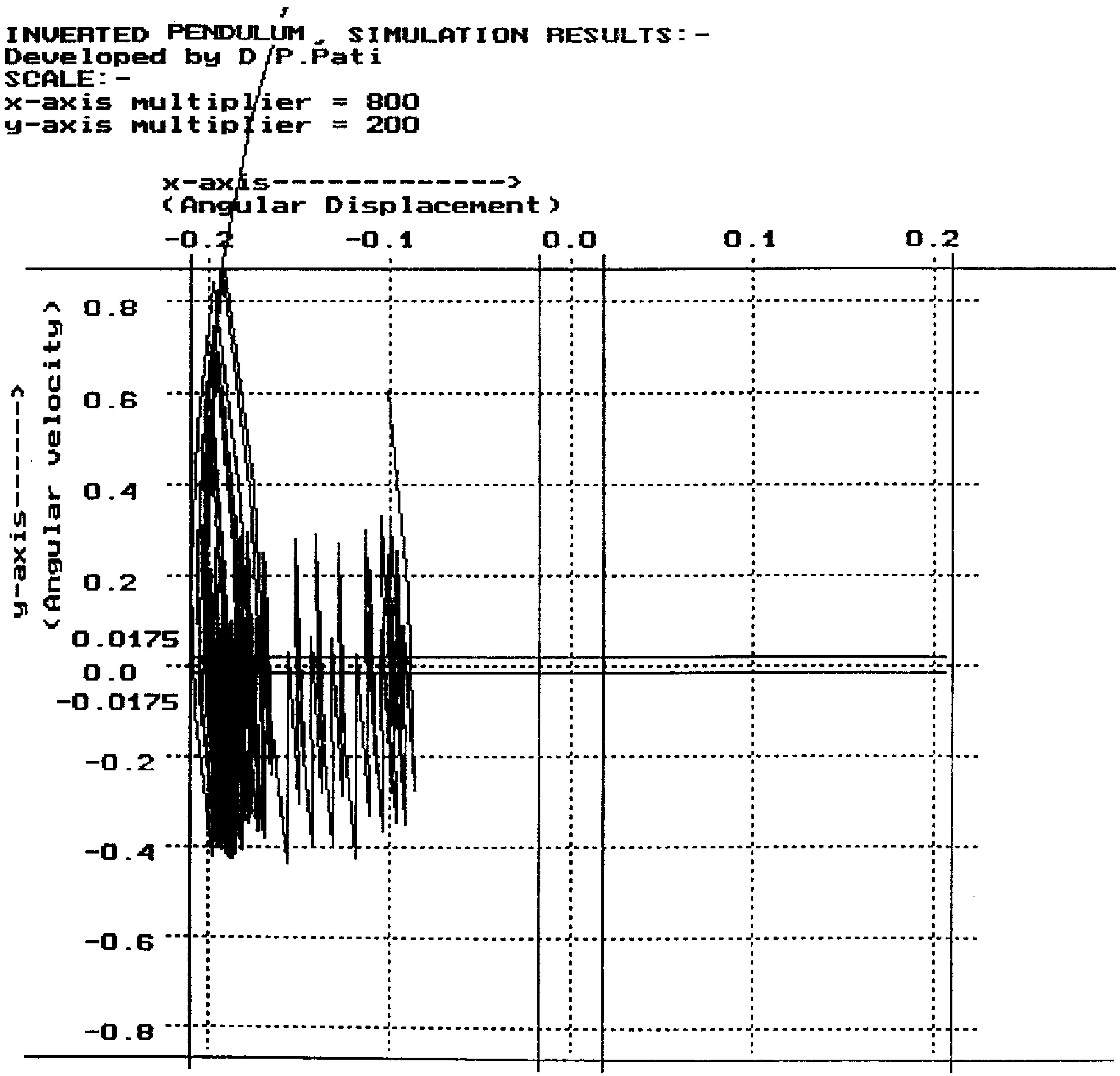
INPUT 'U'=-1.000 (OPEN LOOP SYSTEM)

$$p(u) = \begin{bmatrix} 0.660 & 0.040 & 0.300 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.020 & 0.980 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.660 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.340 \\ 0.240 & 0.000 & 0.070 & 0.360 & 0.000 & 0.330 & 0.000 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.070 & 0.000 & 0.000 & 0.930 & 0.000 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.550 & 0.000 & 0.000 & 0.050 & 0.400 \\ 0.000 & 0.000 & 0.000 & 0.060 & 0.000 & 0.040 & 0.460 & 0.040 & 0.400 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.040 & 0.000 & 0.000 & 0.960 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.520 & 0.480 \\ 0.147 & 0.000 & 0.057 & 0.005 & 0.000 & 0.000 & 0.132 & 0.015 & 0.057 & 0.585 \end{bmatrix}$$

CLOSED LOOP SYSTEM

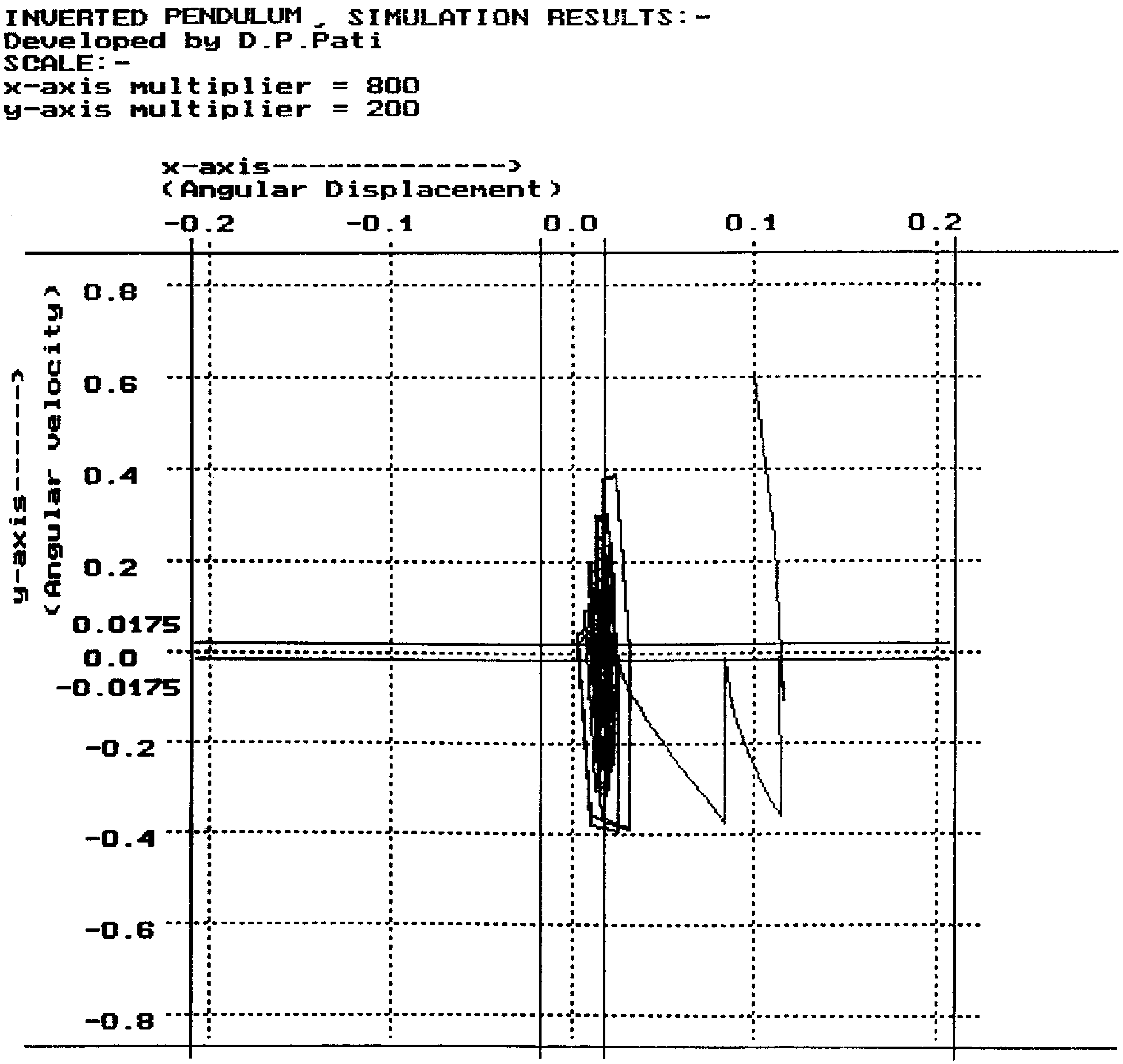
$$p(u) = \begin{bmatrix} 0.660 & 0.040 & 0.300 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \\ 0.950 & 0.050 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \\ 0.550 & 0.050 & 0.400 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \\ 0.240 & 0.000 & 0.070 & 0.360 & 0.000 & 0.330 & 0.000 & 0.000 & 0.000 & 0.000 \\ 0.070 & 0.000 & 0.000 & 0.930 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 & 0.050 & 0.780 & 0.000 & 0.000 & 0.170 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.100 & 0.000 & 0.000 & 0.880 & 0.020 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.040 & 0.000 & 0.000 & 0.960 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.420 & 0.030 & 0.550 & 0.000 \\ 0.070 & 0.012 & 0.087 & 0.000 & 0.000 & 0.000 & 0.157 & 0.000 & 0.035 & 0.637 \end{bmatrix}$$

fig.7


INVERTED PENDULUM , SIMULATION RESULTS:-
Developed by D P.Pati
SCALE:-
x-axis multiplier = 800
y-axis multiplier = 200
x-axis------------->
(Angular Displacement)
-0.2
-0.1
0.0
0.1
0.2
y-axis------>
(Angular velocity)
0.8
0.6
0.4
0.2
0.0175
0.0
-0.0175
-0.2
-0.4
-0.6
-0.8

$x = -0.1, y = 0.6$

fig.8


INVERTED PENDULUM , SIMULATION RESULTS:-
Developed by D.P.Pati
SCALE:-
x-axis multiplier = 800
y-axis multiplier = 200
x-axis------------->
(Angular Displacement)
-0.2
-0.1
0.0
0.1
0.2
y-axis------>
(Angular velocity)
0.8
0.6
0.4
0.2
0.0175
0.0
-0.0175
-0.2
-0.4
-0.6
-0.8

$x = 0.1, y = 0.6$

fig.8

for $j = 1, 2, \ldots, n$

(5) The above steps are repeated for $i = 1, 2, \ldots, n$.

The control law f is determined by considering the transition matrix for each qualitative input $v(k)$.But it is not true that the working(initially chosen) ,for which the control f is defined, is an optimal one.The final working space as well as f are set from the simulation results.

Here,the input signal $u(k)$ is taken to have three qualitative levels:

$$\begin{aligned} u(k) &= 10 \leftrightarrow v(k) = 1, \\ u(k) &= 0 \leftrightarrow v(k) = 0, \\ u(k) &= -10 \leftrightarrow v(k) = -1 \end{aligned}$$

i.e the force can be chosen to zero or maximum in directions.

Now considering the equations,

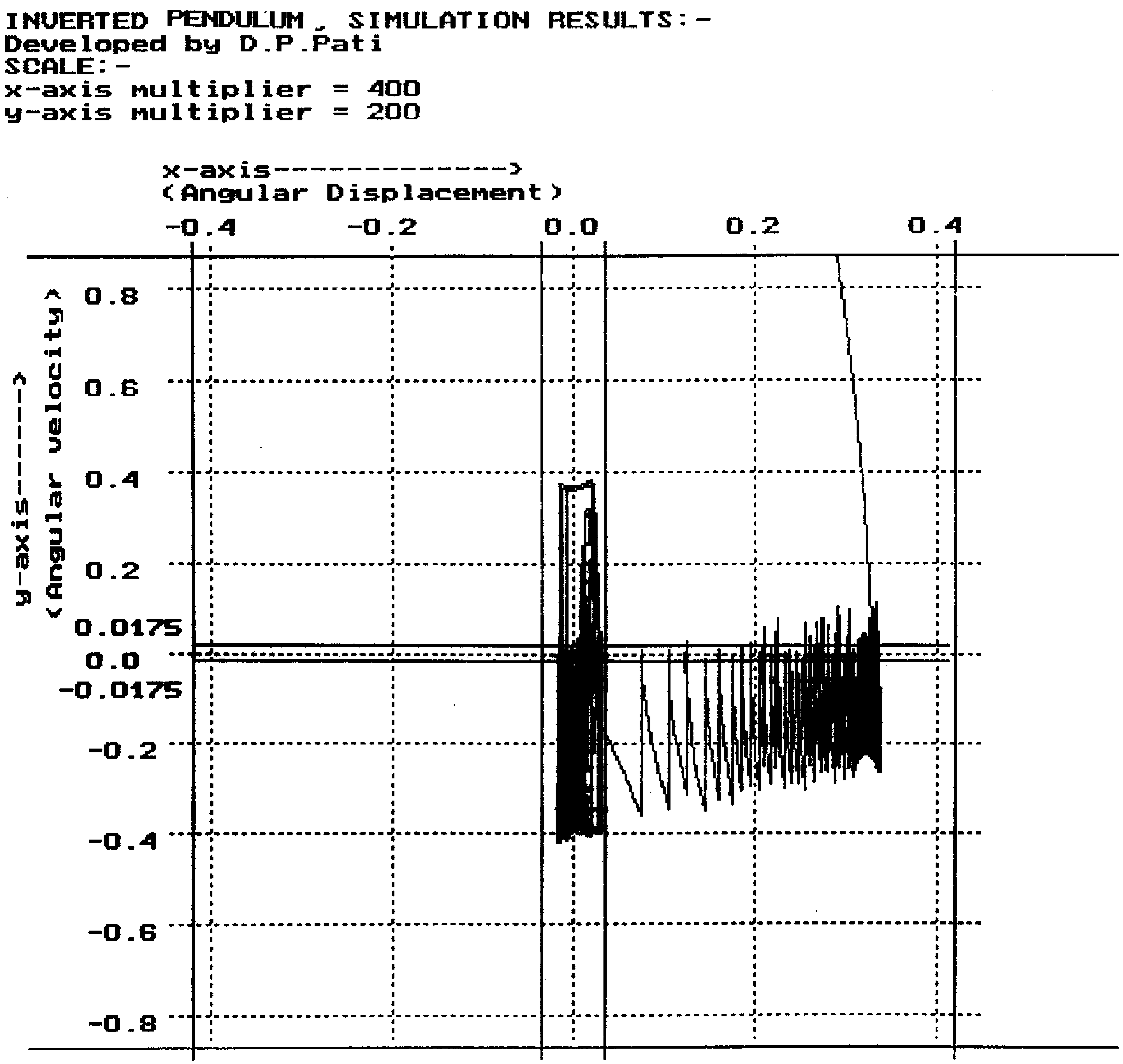
$$\begin{aligned} x_2(1) &= 1.00425x_2(0) + 0.019278x_4(0) \\ x_4(1) &= 0.4160x_2(0) + 1.00425x_4(k) - 0.4v(k) \end{aligned} \tag{34}$$

$P(0), P(1) and P(-1)$ are obtained(as shown in fig.7).By studying these three transition matrices and taking z_5 as the equlibrium state,the control law f is defined as

$$\begin{aligned} &z_1 \rightarrow v(k) = 0, z_2 \rightarrow v(k) = 0 \text{ or } 1^*, z_3 =\rightarrow v(k) = 1, \\ &z_4 \rightarrow v(k) = -1, z_5 \rightarrow v(k) = 1 \text{ or } 0^*, z_6 =\rightarrow v(k) = 0, \\ &z_7 \rightarrow v(k) = 0, z_8 \rightarrow v(k) = 1, z_9 =\rightarrow v(k) = 1, \end{aligned} \tag{35}$$

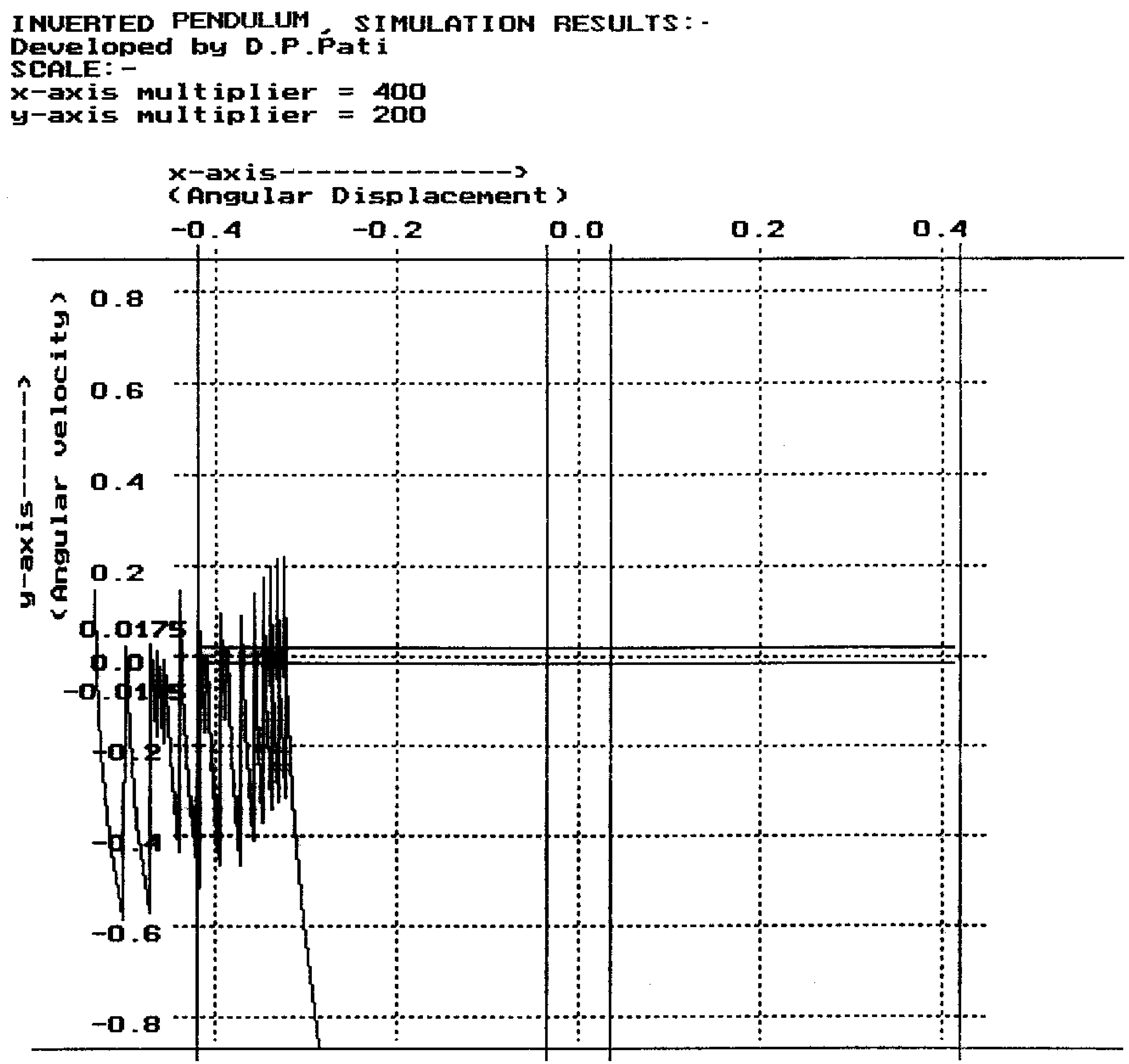
and $*$ mark represents the qualitative input value which are not taken for the simulation results shown here.

But the simulation study(shown in fig.8) shows that, if the initial point is chosen from $z_4 - z_9$, the above control law stabilises the system.Otherwise,it oscllates around the boundary $g_{0,-3}$ and goes to unstable region.We can say that above control law could not set the poles inside the unit circle for $z_1 - z_3$.In order to set the poles in the unit circle,a forcing function , $f_i = d_i e^{-n}$ is used where $d_i \geq (g_{i,j_eql} - g_{i,j})$ i.e it is taken to be greater than the distance between the equilibrium boundary to new unstable boundary where the system oscillates before going to unstable regions and n (positive real number) $< n_{cr}$



$x = 0.291, y = 0.869$

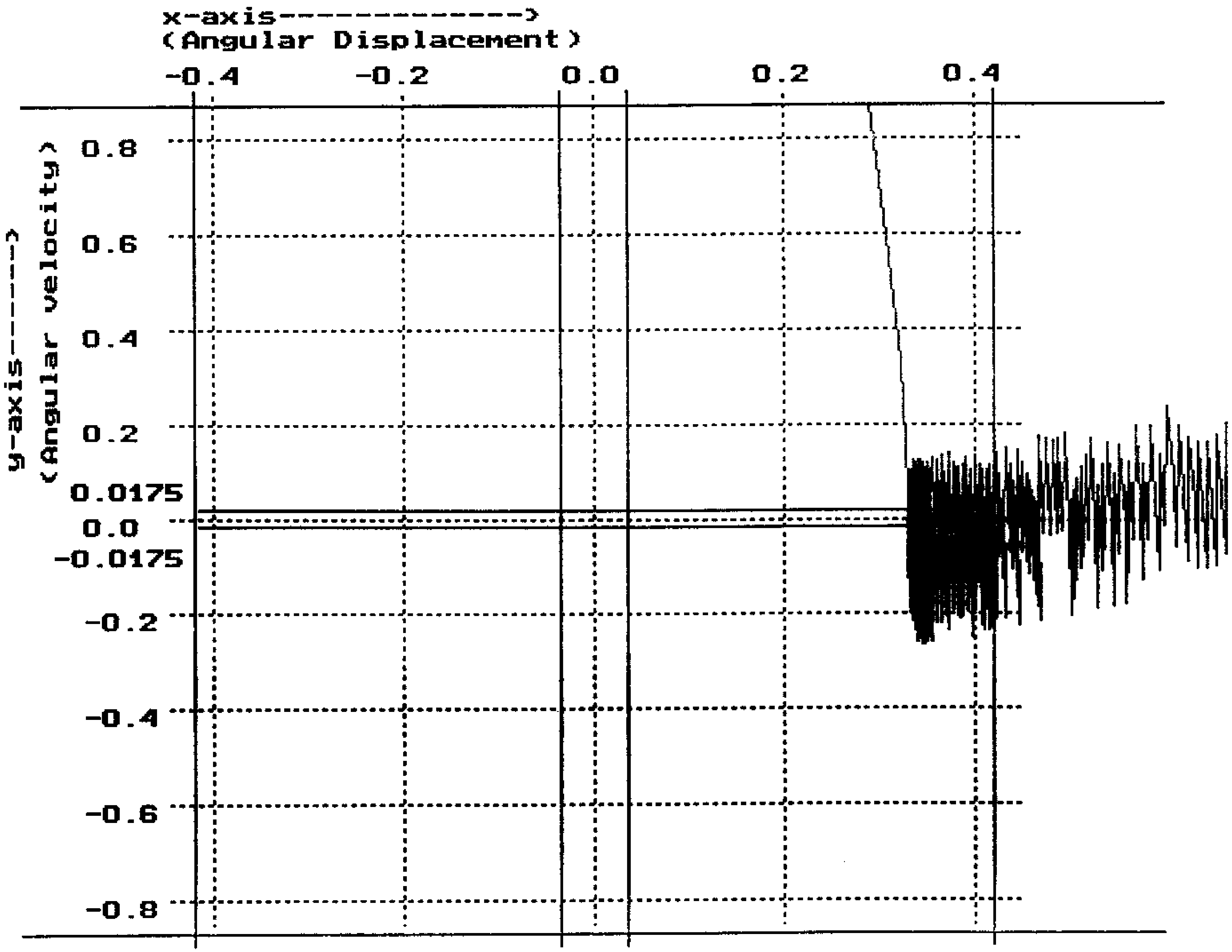
$fig.9$



$x = -0.283, y = -0.869$

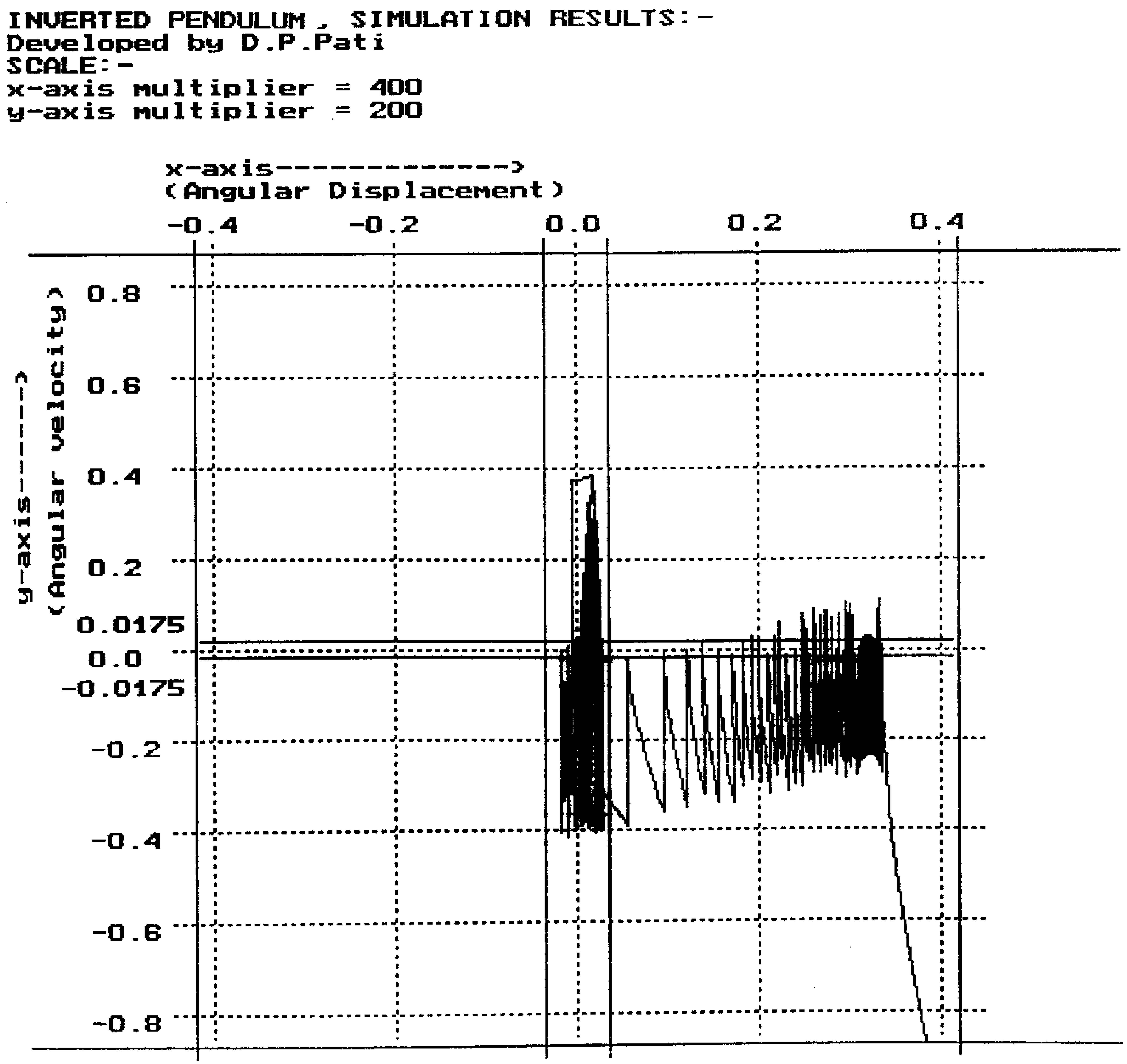
fig.10

INVERTED PENDULUM , SIMULATION RESULTS:-
Developed by D.P.Pati
SCALE:-
x-axis multiplier = 400
y-axis multiplier = 200



$x = 0.292, y = 0.869$

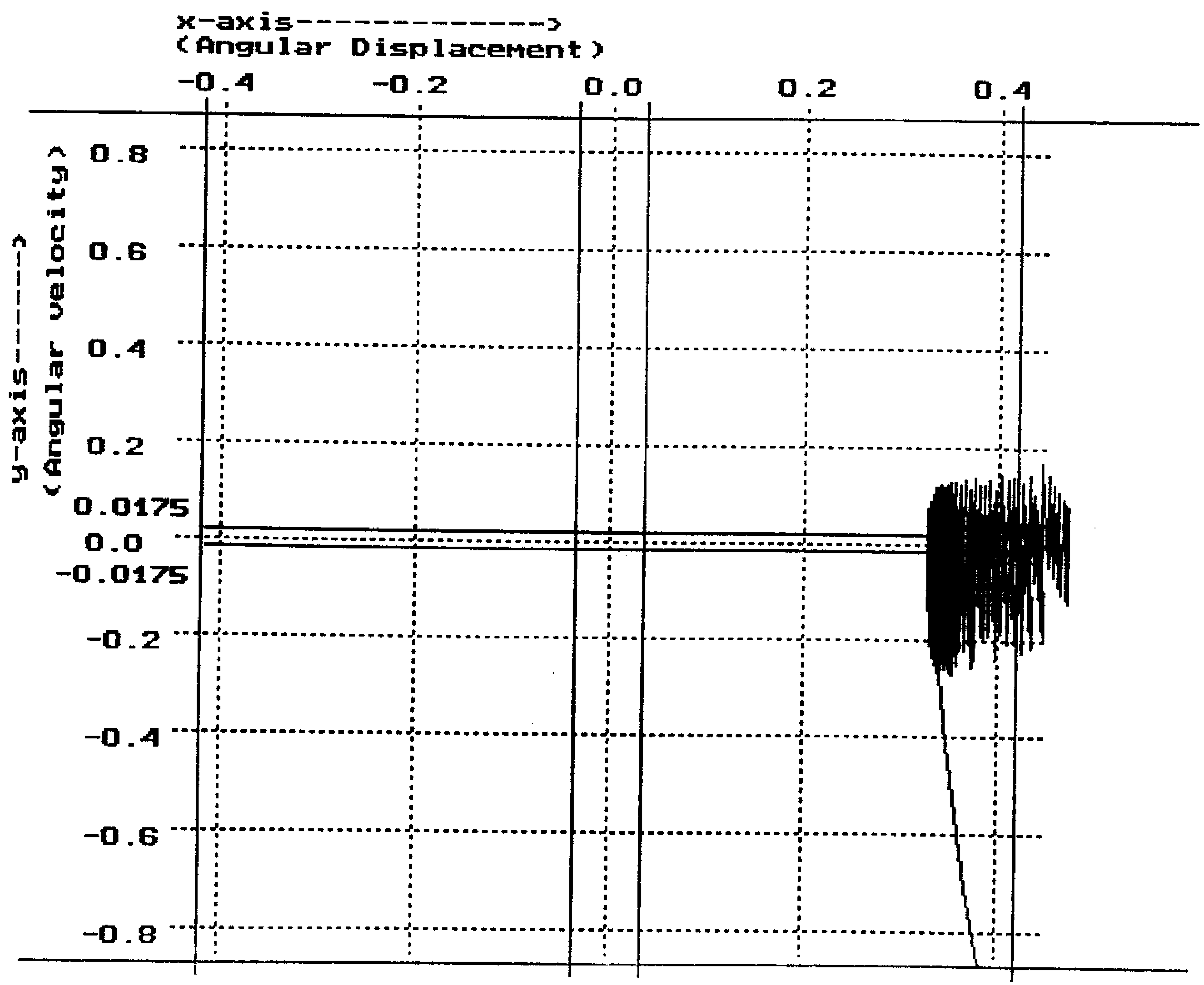
*fig.*11



$x = 0.384, y = -0.869$

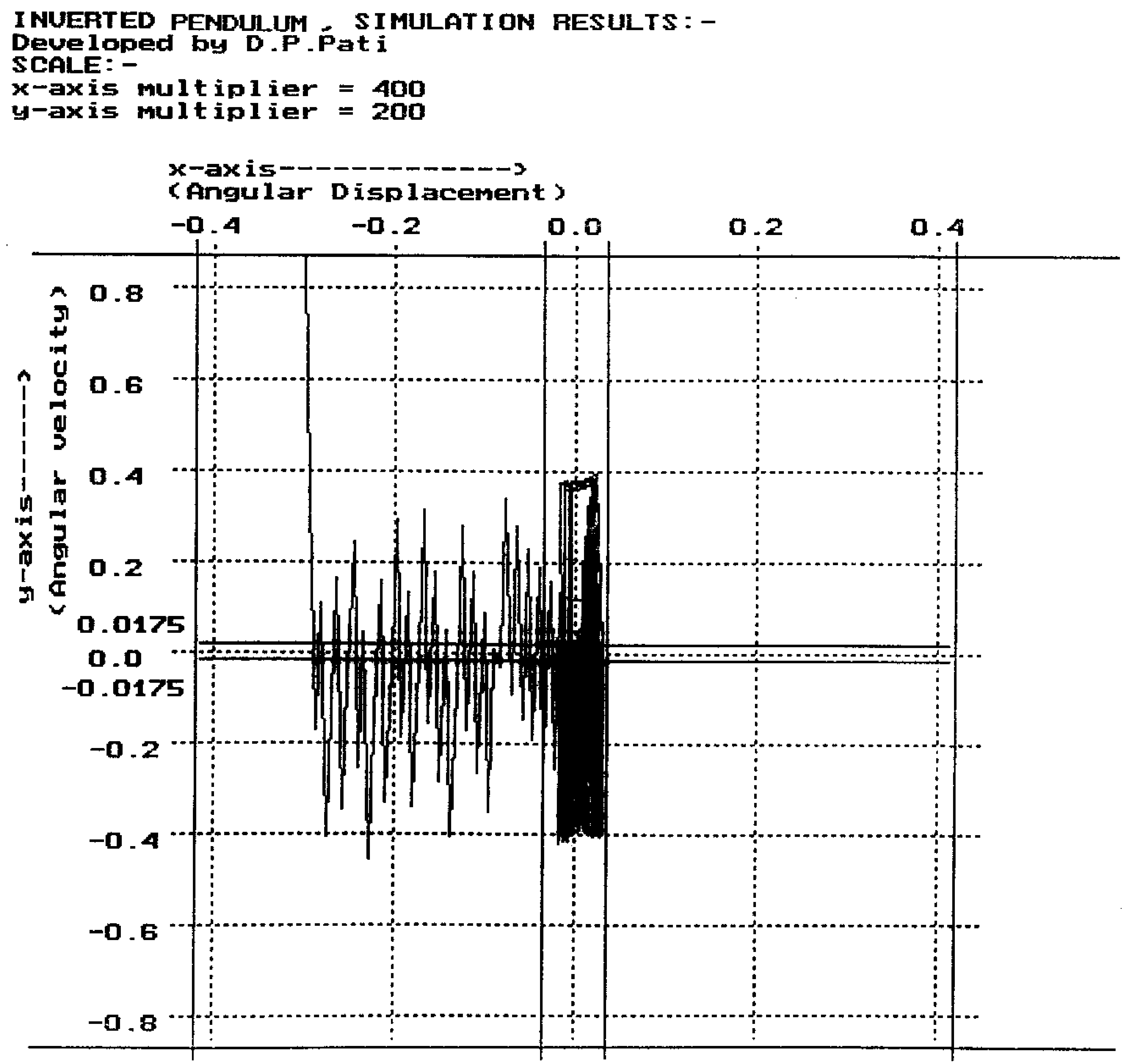
*fig.*12

INVERTED PENDULUM , SIMULATION RESULTS:-
Developed by D.P.Pati
SCALE:-
x-axis multiplier = 400
y-axis multiplier = 200



$x = 0.385, y = -0.869$

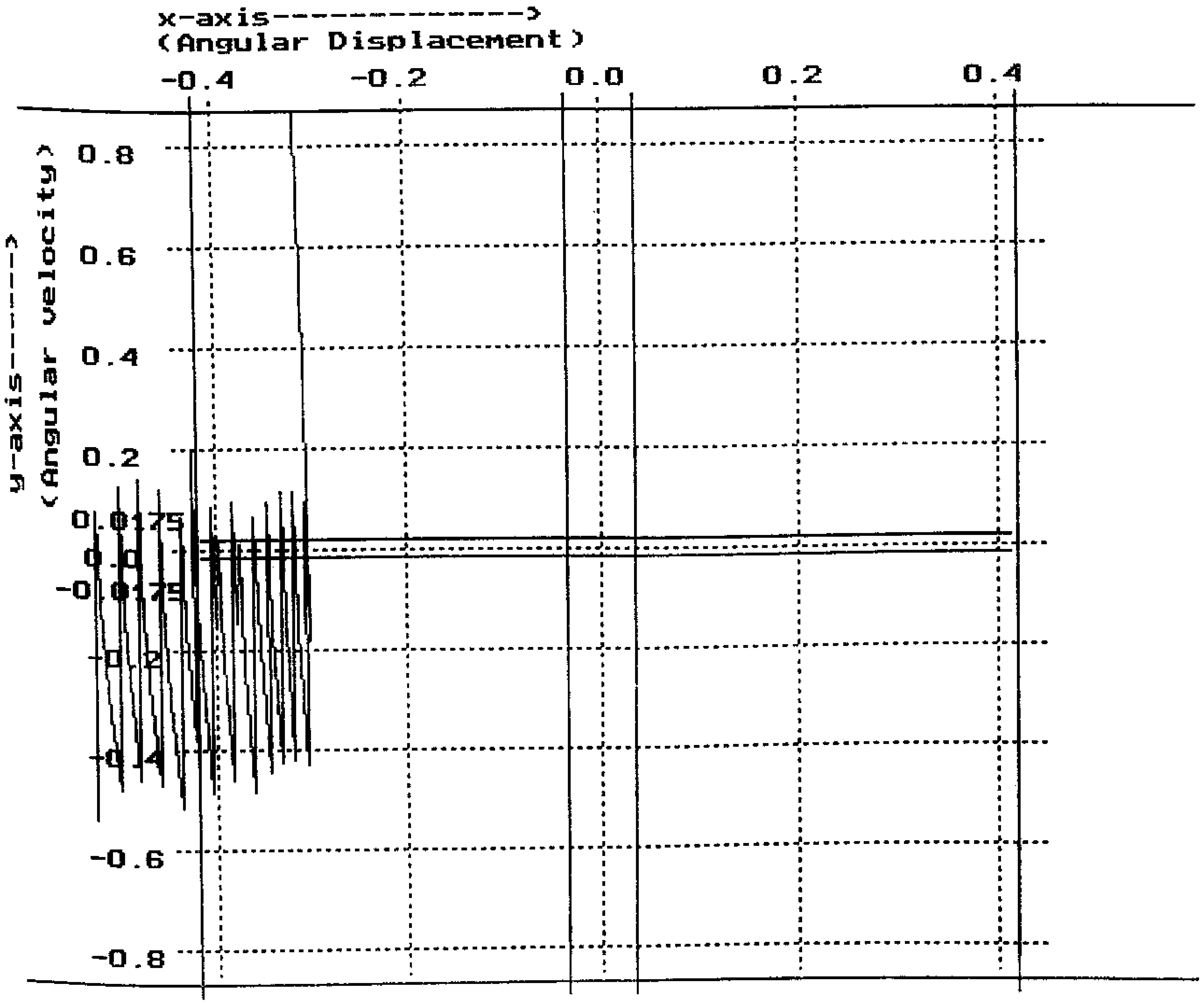
*fig.*13



$x = -0.299, y = 0.869$

fig.14

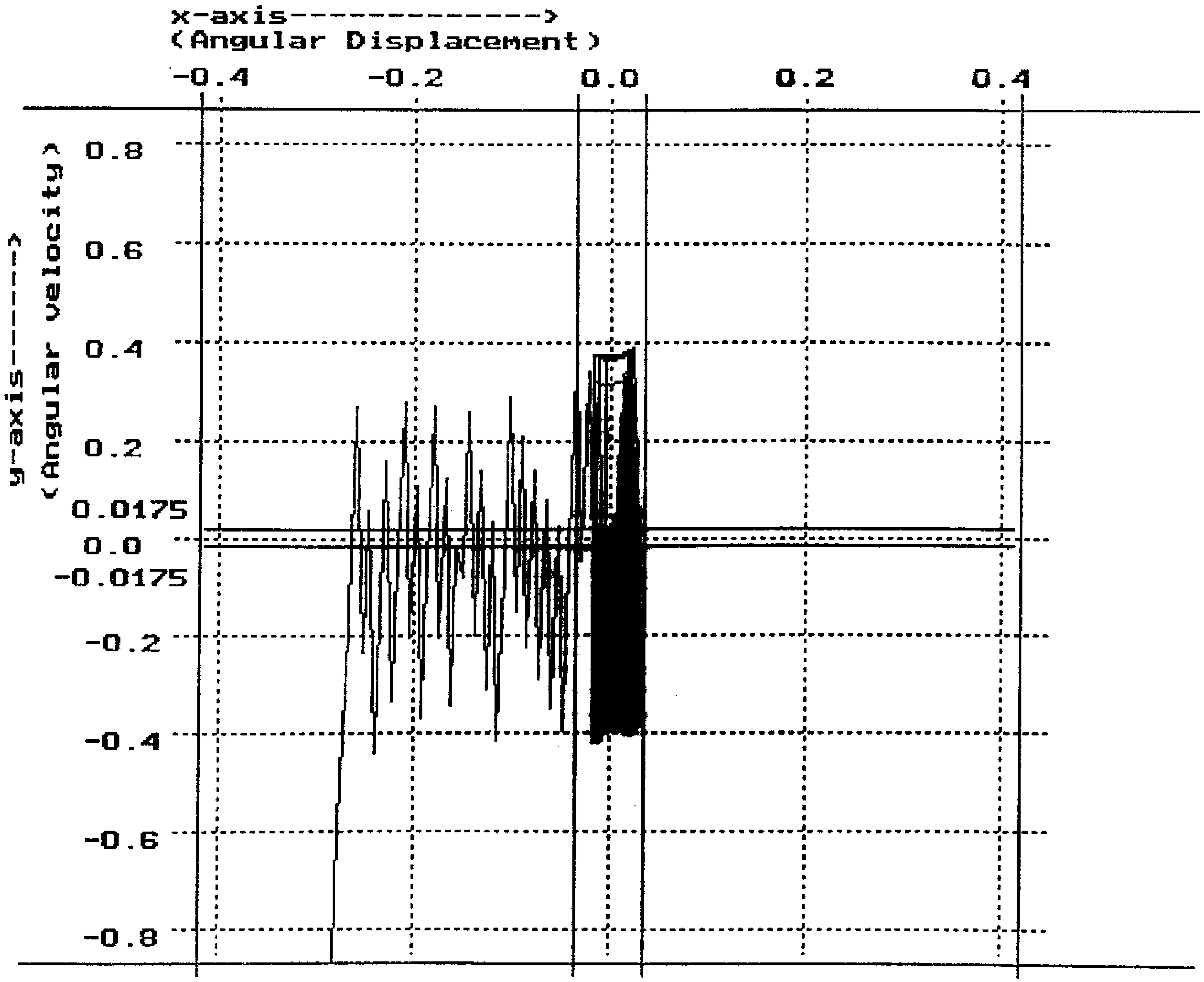
INVERTED PENDULUM , SIMULATION RESULTS:-
Developed by D.P.Pati
SCALE:-
x-axis multiplier = 400
y-axis multiplier = 200



$x = -0.31, y = 0.869$

*fig.*15

INVERTED PENDULUM , SIMULATION RESULTS:-
Developed by D.P.Pati
SCALE:-
x-axis multiplier = 400
y-axis multiplier = 200



$x = -0.282, y = -0.869$

*fig.*16

determines the rate at which it moves towards the stable region and n_{cr} determines the critcal value at which the system is no more stable.Here, the simulation study shows the state variable x_2 is responsible for driving the system into unstable region. The forcing function used here is

$$f_2 = 0.2e^{-3.5}, n_{cr} < 5.0.$$

Now one-step transition can be determined by

$$\begin{aligned} x_2(1) &= 1.00425x_2(0) + 0.019278x_4(0) + 0.2e^{-3.5} \\ x_4(1) &= 0.4160x_2(0) + 1.00425x_4(k) - 0.4v(k) \end{aligned} \tag{36}$$

and the boundaries of the working space are determined from simulation results (shown in fig.9 to fig.16) and they are

$$\begin{aligned} g_{0,-3} &= -0.28, g_{0,3} = 0.28 \\ g_{1,-3} &= -0.869, g_{1,3} = 0.869. \end{aligned} \tag{37}$$

The performance of the automaton (in Eq20) for $k = 500$, shown in fig.17 does not coincides with the simulation results ie as $k \to \infty$ then $q(\infty)$ should be in either z_4 or z_5 or z_6.But if we take 100-step transition instead of taking one-step transition for calculating new transistion matrix, then the performance of automaton and the simulation study somehow holds good as shown in fig.17.This indicates that one-step transition callculation is not appropriate for calculation of new transition matrix.We may explain it as follows:

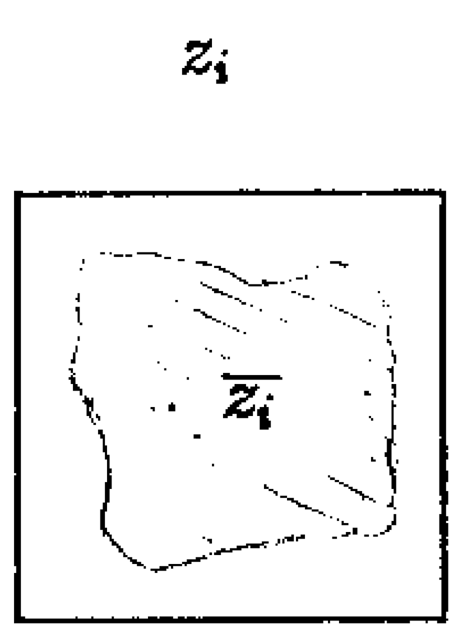


fig.18

consider a region z_i, by one step transition,each points $x(0) \in \overline{z_i}$ goes to z_i although they would have tendency to go to z_j for $j \neq i$ because it depends on the size of the region z_i and the maximum values,δx_is' possible in that region.Hence by taking more step transition instead of one-step transition,we could get better results.

NO OF ITERATION = 500 (TAKING 100 TRANSITION)

0.021	0.014	0.017	0.130	0.049	0.379	0.247	0.017	0.068	0.052
0.021	0.014	0.017	0.130	0.049	0.379	0.247	0.017	0.068	0.052
0.021	0.014	0.017	0.130	0.049	0.379	0.247	0.017	0.068	0.052
0.021	0.014	0.017	0.130	0.049	0.379	0.247	0.017	0.068	0.052
0.021	0.014	0.017	0.130	0.049	0.379	0.247	0.017	0.068	0.052
0.021	0.014	0.017	0.130	0.049	0.379	0.247	0.017	0.068	0.052
0.021	0.014	0.017	0.130	0.049	0.379	0.247	0.017	0.068	0.052
0.021	0.014	0.017	0.130	0.049	0.379	0.247	0.017	0.068	0.052
0.021	0.014	0.017	0.130	0.049	0.379	0.247	0.017	0.068	0.052
0.021	0.014	0.017	0.130	0.049	0.379	0.247	0.017	0.068	0.052

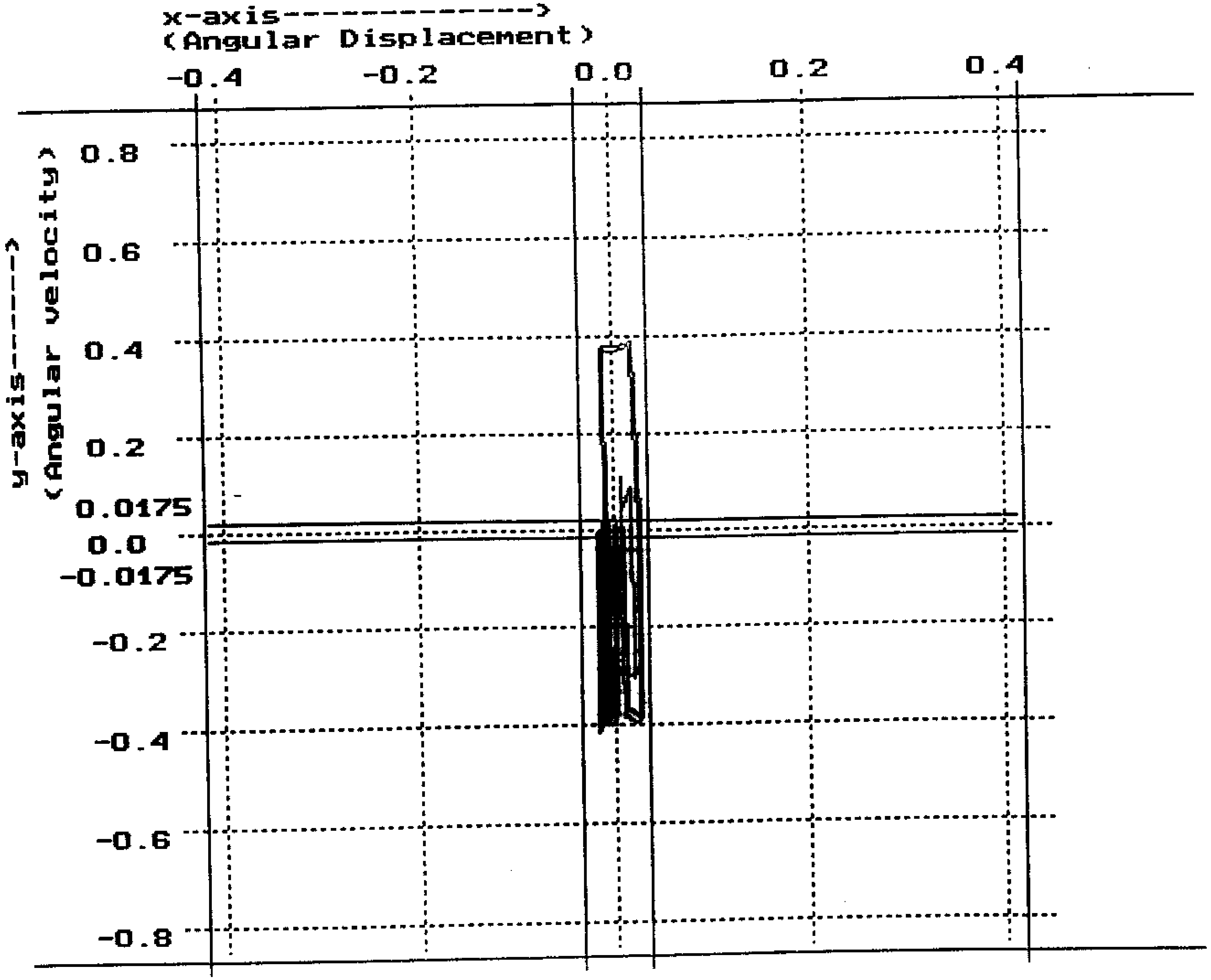
NO OF ITERATION = 500 (Taking One Transition)

0.637	0.043	0.318	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.637	0.043	0.318	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.637	0.043	0.318	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.637	0.043	0.318	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.637	0.043	0.318	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.637	0.043	0.318	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.637	0.043	0.318	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.637	0.043	0.318	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.637	0.043	0.318	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.637	0.043	0.318	0.000	0.000	0.000	0.000	0.000	0.000	0.000

Note :The ith row corresponds to z_i as starting region for the automaton.

fig.17

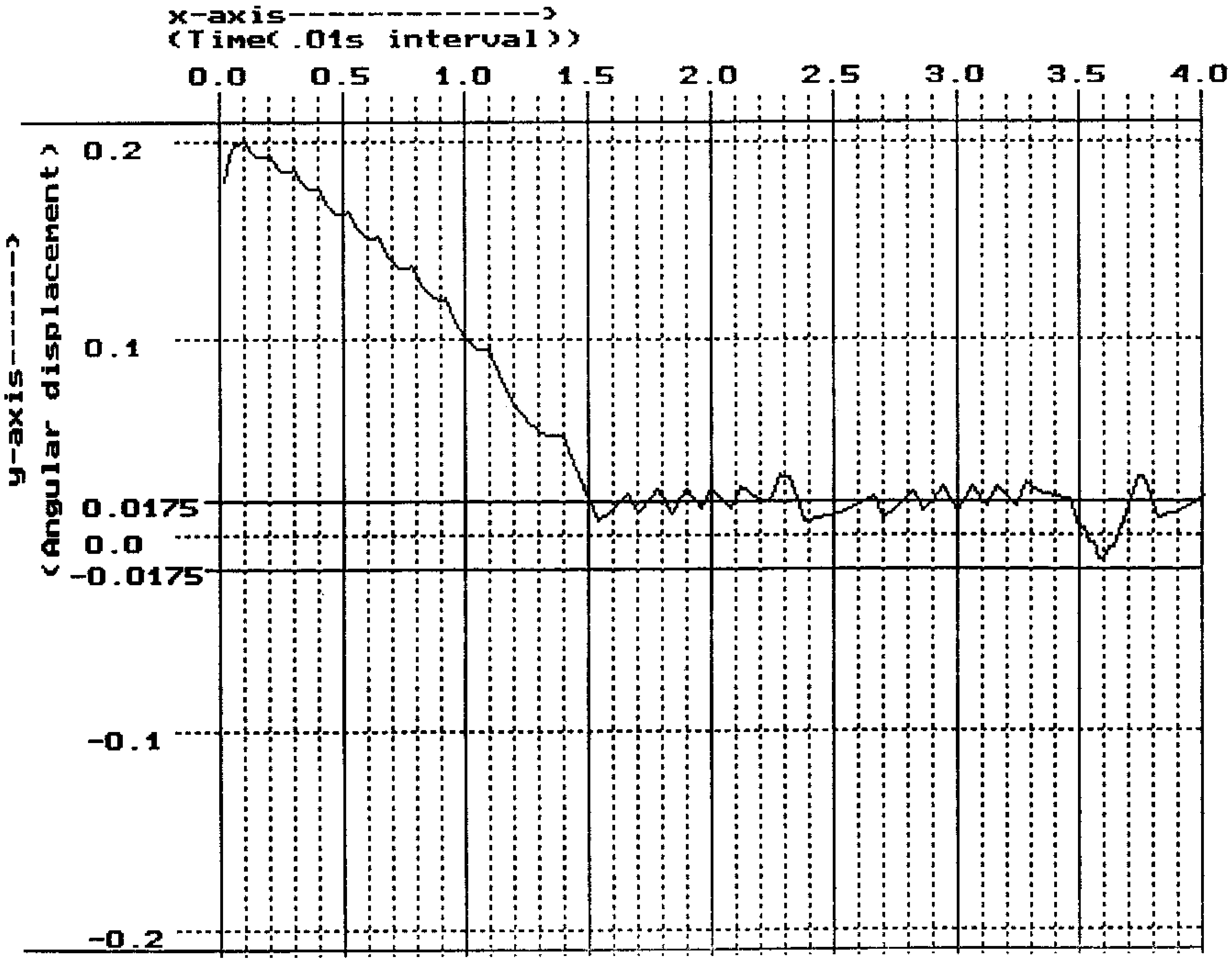
INVERTED PENDULUM , SIMULATION RESULTS:-
Developed by D.P.Pati
SCALE:-
x-axis multiplier = 400
y-axis multiplier = 200



$x = 0.0, y = 0.0$

fig.19

INVERTED PENDULUM
Developed by D.P.Pati
SCALE:-
x-axis multiplier = 100
y-axis multiplier = 800



The simulation result shows that the desired controller actually stabilises the inverted pendulum,Again it has been shown in fig.19 that the closed loop system without input,oscillates around the regions z_4, z_5 and z_6 which can be compared with a limit cycle of nonlinear system.The designed qualitative feedback controller incorporates severe nonlinearity of the closed loop system. It is also found that the equilibrium point cannot be approached asymptotically, rather remains in the vicinity of this point.

7 Conclusion.

Here, we have proposed the design of qualitative feedback controller by stochastic automata which uses the new transition concept.As the determination of new transition matrix involves, somehow directly with the dynamics of the system rather than its qualitative nature, the stochastic automata provide a resonable framework for the qualitative modelling.This new form of qualitative model makes it possible to use well known results on discrete event systems, such as manufacturing systems or computer nets.Further,we can say,the improvement on the determination of new transition matrix,may motivate the design of qualitative controller to some extent.

8 References

(1). Lunze,J.:'Qualitative modelling of continuous-variable systems by means of nondeterministic automata'Intelligent Systems Engineering, Autumn 1992.

(2).Tou,Julius T.:'Stochastic Automata and Discrete Systems Theory'Academic Press,New York,1967.

(3).Kuo,B.C.:Digital Control System.

(5).Iserman,Rolf.:Digital Control System.