DISCRETE-TIME SLIDING MODE CONTROL OF ONE-LEGGED *SLOM* HOPPING ROBOT

Ajij Sayyad [*,1] **Bharatendu Seth** [*,2]
Kurien K. Issac [*,2]

[*] *Indian Institute of Technology, Bombay, Mumbai, India*
(sayyad@sc.iitb.ac.in, seth@iitb.ac.in, kurien@iitb.ac.in)

Abstract: In the proposed study, an asymmetrical two-degree-of-freedom one-legged hopping robot - a *Springy-Legged Offset-Mass* (*SLOM*) robot - is investigated. An under-actuated model, uses a simple linear actuator to input energy within the flight duration. Further, a 3-dimensional Poincaré return map is used to construct a reduced order implicit model representing the non-linear dynamics of the robot. Using *Taylor series approximation* technique, we formulated the explicit non-linear approximate model of the proposed robot. The periodic motion obtained for the system with a fixed energy injection in each cycle is unstable. We used a different discrete-time sliding mode based state feedback stabilization strategy to accomplish sustained hopping behaviour.

Keywords: hopping robot, Poincaré map, Taylor series approximation, sliding mode control

1. INTRODUCTION

An extensive research related to one-legged hopping robots has been going on for over three decades (Sayyad *et al.*, 2007*b*). Almost in all these single-legged hopping robot configurations, the body mass is symmetrically distributed such that the geometrical axis of a telescopic leg is along the center of gravity (CG). Asymmetric 2D configurations have been considered by Shanmuganathan (Shanmuganathan, 2002) and Kuswadi et al. (Kuswadi *et al.*, 2003). While, Wei et. al. (Wei *et al.*, 2000) have considered 3D asymmetric configuration. Accounting for the asymmetry in single-legged hopping robot is a new contribution along with these studies. Silent features differing the previous studies and proposed model is highlighted in our previous study (Sayyad *et al.*, 2007*a*). We refer to these configurations as "*SLOM*" hoppers, wherein the line of action of the spring or actuator force is offset from the CG, which appears to give rise to a restoring rotational moment.

There is loss of energy in hopping robots both due to impacts and frictional affects. A periodic hopping motion is possible only if the lost energy is compensated periodically. In our model, energy is compensated in each hop by compressing the leg spring in the flight phase. There are several advantages of this approach (Sayyad *et al.*, 2007*a*). However, the capability of this input in stabilizing the SLOM, needs to be examined. The main focus of this work is how to stabilize the SLOM with this inputs using feedback.

Regarding the sliding mode based control theory, it is based on the concept of varying the structure of the controller based on the state variation of the system in order to obtain a desired response (Young *et al.*, 1999). Typically, sliding mode con-

[1] Research Scholar, IDP in Systems and Control Engg.
[2] Professor, Department of Mechanical Engg.

trol is based on state feedback. Such closed-loop control design problem involves two steps: selecting the switching manifold and reaching the switching surface within finite time. A control effort is used to switch between different structures and the system state is forced to move along a chosen manifold, called the switching manifold.

The main reason of proposed study is that the evolved dynamics of the hopping robot by choosing appropriate Poincaré section is discrete time system. Here the 'time' refers to the hop. Recently, many studies are tackling the Discrete-time Sliding Mode Control (DSMC) system design problems (Furuta, 1990)-(Bandyopadhyay and Janardhanan, 2006). A typical characteristics of DSMC systems is that the control input is applicable only at certain sampling instants and the control effort is constant over the entire sampling period. However, the system states are no longer constrained to lie upon the confined sliding surface. As a result, DSMC systems can undergo only quasi-sliding mode, i.e. the system states are allowed to remain in the vicinity of the sliding surface, called the *boundary Layer*.

The present study uses the Gao's reaching law (Gao *et al.*, 1995) and the saturation type reaching law (Misawa, 1997). The proposed DSMC designs use full state feedback and is applied to nonlinear hopping robotic system that guarantees discrete sliding mode. Moreover, to the author's knowledge, this is the first attempt to apply DSMC for single-legged hopping robot.

The remainder part of the paper is organized as follows: Section 2, describes the proposed SLOM hopper and its dynamic model. In Section 3, we determine periodic motion by constructing a 3-dimensional Poincaré map and seeking its fixed points. The Taylor series approximation approach is used to construct the discrete-time linear and non-linear approximate models. Section 4 is focused on the design of DSMC for the proposed hopping model using different state feedback based control algorithms. Section 5 discusses the results obtained in the current study.

2. THE PROPOSED *SLOM* HOPPER

In general, the hopping robots are variable structure systems where continuous motion is followed by impacts which are discrete events . The posture of proposed hopper during the stance phase is shown in the Figure 1. It may be noted that the leg spring is offset by a distance d from the CG of the body located at A. The spring force also does not act along a line passing through the CG of the system (located at C). The leg length (l) is defined to be the distance DQ. The leg motion in downward direction is restricted by a mechanical stop.

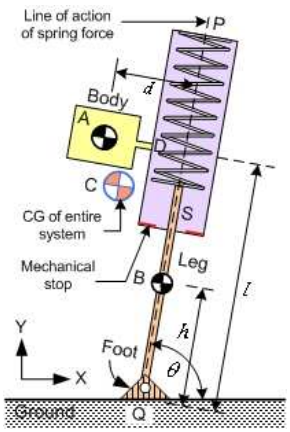

Fig. 1. A proposed *SLOM* hopping robot

2.1 Euler-Lagrangian (EL) equations of motion:

A hopping robot moves with alternating flight and stance phases. In the EL formulation, during such phase change, the dynamics of the robot does not vary, but only the constraints imposed on the robot changes. We choose $q = [\, x_b \;\; y_b \;\; l \;\; \theta \,]^T$ as the set of generalized coordinates. Here x_b and y_b are the coordinates of CG of body. The standard form of a robot model (for no external forces) is,

$$D(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = -J^T(q)\lambda_r$$
$$J(q)\dot{q} = 0 \qquad (1)$$

Where, $D \in R^{n\times n}$, $C \in R^{n\times n}$, and $G \in R^n$ are called the inertia matrix, the "coriolis and centrifugal forces" matrix and gravitational force vector respectively, and given as,

$$D = \begin{bmatrix} M & 0 & -m_l\cos\theta & m_l l_1 \\ 0 & M & -m_l\sin\theta & -m_l l_2 \\ -m_l\cos\theta & -m_l\sin\theta & m_l & -dm_l \\ m_l l_1 & -m_l l_2 & -dm_l & I \end{bmatrix},$$

$$C = \begin{bmatrix} 0 & 0 & m_l\sin\theta\dot{\theta} & m_l\dot{l}\sin\theta + m_l l_2\dot{\theta} \\ 0 & 0 & -m_l\cos\theta\dot{\theta} & -m_l\dot{l}\cos\theta + m_l l_1\dot{\theta} \\ 0 & 0 & 0 & m_l(l-h)\dot{\theta} \\ 0 & 0 & m_l(l-h)\dot{\theta} & m_l(l-h)\dot{l} \end{bmatrix},$$

$$G = \begin{bmatrix} 0 & Mg & -m_l g sin\theta + K_s(l-L_0) & -m_l l_2 \end{bmatrix}^T.$$

Note that, $M = m_b + m_l$, $I = I_b + I_l + m_l\Big((l-h)^2 + d^2\Big)$, $l_1 = (l-h)sin\theta + dcos\theta$, $l_2 = (l-h)cos\theta - dsin\theta$ and L_0, is the length l when the spring force is zero. λ_r is the generalized constraint force. J is Jacobian matrix determined based on constraints. The subscripts b and l stands for parameters related to body and leg of hopper respectively.

2.2 Equations of impact:

Due to impact there is sudden discontinuous change in the generalized velocities, while the generalized coordinates and input changes continuously. The change of the derivative of the generalized co-ordinates ($\dot{q}$) can be determined by integrating equation (1) with respect to time across the impulse.

$$D(q)(\dot{q}_+ - \dot{q}_-) = -J^T(q)\lambda_I$$
$$J(q)\dot{q}_+ = 0$$

where, $\dot{q}_+$ and $\dot{q}_-$ are respectively the post and prior generalized velocities across the impulse. Refer ((Sayyad *et al.*, 2007*a*)) for the constraints and Jacobian during different phases and events.

3. POINCARÉ MAP AND ITS TAYLOR SERIES APPROXIMATION

Our first objective is to obtain a periodic motion with the help of an energy compensation mechanism, adding fixed amount of energy in each flight phase. We have found periodic motion by seeking fixed points of Poincaré return maps. The Poincaré section used in our analysis corresponds to the instant just before take-off, when the leg reaches its fully extended state. It is defined as $l(t) = l_{takeoff}$ and $\dot{l}(t) > 0$.

The Poincaré section is represented by three variables as, $\mathbf{x} = \begin{bmatrix} \theta & \dot{l} & \dot{\theta} \end{bmatrix}^T$, which is one less than the number of state variables for stance phase, $(l, \theta, \dot{l}, \dot{\theta})$. The entire hopping cycle is divided into sub-phases ((Sayyad *et al.*, 2007*a*)). If the state at k^{th} take off is identical to the state at $(k+1)^{th}$ take off, then the cycle is repetitive. In our system, the Poincaré map is defined as a vector function $\mathbf{P} : R^3 \times R^1 \rightarrow R^3$ mapping the k^{th} take off state to the $(k+1)^{th}$ take off state as,

$$\mathbf{x}_{k+1} = \mathbf{P}(\mathbf{x}_k, u_k) \tag{2}$$

for a fixed spring compression during flight phase (referred as u, system input). Here $\mathbf{x}_k$ is the state $\mathbf{x}$ at k^{th} hop. To determine the periodic motion, we search for the fixed point of the function $\mathbf{P}$. The fixed point is a pair $(\overline{\mathbf{x}}, \overline{u})$ which satisfy following equation

$$\mathbf{x} - \mathbf{P}(\mathbf{x}, u) = 0$$

3.1 Taylor series approximation:

The hopper model using the Poincaré map in equation (2) is discrete time (hop) model as,

$$\mathbf{x}_{k+1} = f(\mathbf{x}_k, u_k) \tag{3}$$

where, $\mathbf{x} = [\, x_1 \; x_2 \; x_3]^T = [\, \theta \; \dot{l} \; \dot{\theta}]^T$, f is the function $\mathbf{P}$ defined in equation (2) and since we are dealing with single input, u is the scalar quantity. More explicitly, we can write $f(\mathbf{x}_k, u_k) = f(x_{1k}, x_{2k}, x_{3k}, u_k)$. Let's define new variables that simply measure how far we are from the fixed point $(\overline{\mathbf{x}}, \overline{u})$ of given non-linear function, as below,

$$\begin{aligned} \triangle\mathbf{x} &= \mathbf{x} - \overline{\mathbf{x}} \\ \triangle u &= u - \overline{u} \end{aligned} \tag{4}$$

Using the *Taylor series approximation*, we have,

$$\begin{aligned} f(\mathbf{x}, u) &= f(x_1, x_2, x_3, u) \\ &= \sum_{m=0}^{n} \frac{1}{m!} \Big(a + b + c + d \Big)^m f(\overline{\mathbf{x}}, \overline{u}) \end{aligned}$$

where, $a = \triangle x_1 \frac{\partial}{\partial x_1}$, $b = \triangle x_2 \frac{\partial}{\partial x_2}$, $c = \triangle x_3 \frac{\partial}{\partial x_3}$, $d = \triangle u \frac{\partial}{\partial u}$ and n is the order of approximation. The term $(\ldots)^m$ is the order-variant partial derivative operator on f. We chosen $n = 3$ (cubic approximation). Further simplification gives,

$$\begin{aligned} f(\mathbf{x}, u) = f(\overline{\mathbf{x}}, \overline{u}) &+ A\triangle\mathbf{x} + B\triangle u \\ &+ \frac{1}{2} D_1 f_1(\triangle\mathbf{x}, \triangle u) \\ &+ \frac{1}{6} D_2 f_2(\triangle\mathbf{x}, \triangle u) \end{aligned} \tag{5}$$

where, in our case, $A \in R^{3\times3}$, $B \in R^{3\times1}$, $D_1 \in R^{3\times10}$ and $f_1 \in R^{10\times1}$, $D_2 \in R^{3\times20}$ and $f_2 \in R^{20\times1}$. It is to be noted that, A, B, D_1 and D_2 are determined around $\mathbf{x} = \overline{\mathbf{x}}$ and $u = \overline{u}$ as,

$$A = \begin{bmatrix} \dfrac{\partial f(\mathbf{x},u)}{\partial x_1} & \dfrac{\partial f(\mathbf{x},u)}{\partial x_2} & \dfrac{\partial f(\mathbf{x},u)}{\partial x_3} \end{bmatrix}, \; B = \frac{\partial f(\mathbf{x},u)}{\partial u}$$

$$D_1{}^T = \begin{bmatrix} \dfrac{\partial^2 f(\mathbf{x},u)}{\partial {x_1}^2} \\ \dfrac{\partial^2 f(\mathbf{x},u)}{\partial {x_2}^2} \\ \dfrac{\partial^2 f(\mathbf{x},u)}{\partial {x_3}^2} \\ \dfrac{\partial^2 f(\mathbf{x},u)}{\partial u^2} \\ \dfrac{\partial^2 f(\mathbf{x},u)}{\partial x_1 \partial x_2} \\ \dfrac{\partial^2 f(\mathbf{x},u)}{\partial x_1 \partial x_3} \\ \dfrac{\partial^2 f(\mathbf{x},u)}{\partial x_1 \partial u} \\ \dfrac{\partial^2 f(\mathbf{x},u)}{\partial x_2 \partial x_3} \\ \dfrac{\partial^2 f(\mathbf{x},u)}{\partial x_2 \partial u} \\ \dfrac{\partial^2 f(\mathbf{x},u)}{\partial x_3 \partial u} \end{bmatrix}, \; f_1 = \begin{bmatrix} \triangle {x_1}^2 \\ \triangle {x_2}^2 \\ \triangle {x_3}^2 \\ \triangle u^2 \\ 2\triangle x_1 \triangle x_2 \\ 2\triangle x_1 \triangle x_3 \\ 2\triangle x_1 \triangle u \\ 2\triangle x_2 \triangle x_3 \\ 2\triangle x_2 \triangle u \\ 2\triangle x_3 \triangle u \end{bmatrix}$$

$$D_2{}^T = \begin{bmatrix} \frac{\partial^3 f(\mathbf{x},u)}{\partial {x_1}^3} \\ \frac{\partial^3 f(\mathbf{x},u)}{\partial {x_2}^3} \\ \frac{\partial^3 f(\mathbf{x},u)}{\partial {x_3}^3} \\ \frac{\partial^3 f(\mathbf{x},u)}{\partial u^3} \\ \frac{\partial^3 f(\mathbf{x},u)}{\partial {x_1}^2 \partial x_2} \\ \frac{\partial^3 f(\mathbf{x},u)}{\partial {x_1}^2 \partial x_3} \\ \frac{\partial^3 f(\mathbf{x},u)}{\partial {x_1}^2 \partial u} \\ \frac{\partial^3 f(\mathbf{x},u)}{\partial x_1 \partial {x_2}^2} \\ \frac{\partial^3 f(\mathbf{x},u)}{\partial {x_2}^2 \partial x_3} \\ \frac{\partial^3 f(\mathbf{x},u)}{\partial {x_2}^2 \partial u} \\ \frac{\partial^3 f(\mathbf{x},u)}{\partial x_1 \partial {x_3}^2} \\ \frac{\partial^3 f(\mathbf{x},u)}{\partial x_2 \partial {x_3}^2} \\ \frac{\partial^3 f(\mathbf{x},u)}{\partial {x_3}^2 \partial u} \\ \frac{\partial^3 f(\mathbf{x},u)}{\partial x_1 \partial u^2} \\ \frac{\partial^3 f(\mathbf{x},u)}{\partial x_2 \partial u^2} \\ \frac{\partial^3 f(\mathbf{x},u)}{\partial x_3 \partial u^2} \\ \frac{\partial^3 f(\mathbf{x},u)}{\partial x_1 \partial x_2 \partial x_3} \\ \frac{\partial^3 f(\mathbf{x},u)}{\partial x_1 \partial x_2 \partial u} \\ \frac{\partial^3 f(\mathbf{x},u)}{\partial x_1 \partial x_3 \partial u} \\ \frac{\partial^3 f(\mathbf{x},u)}{\partial x_2 \partial x_3 \partial u} \end{bmatrix}, f_2 = \begin{bmatrix} \triangle {x_1}^3 \\ \triangle {x_2}^3 \\ \triangle {x_3}^3 \\ \triangle u^3 \\ 3\triangle {x_1}^2 \triangle x_2 \\ 3\triangle {x_1}^2 \triangle x_3 \\ 3\triangle {x_1}^2 \triangle u \\ 3\triangle x_1 \triangle {x_2}^2 \\ 3\triangle {x_2}^2 \triangle x_3 \\ 3\triangle {x_2}^2 \triangle u \\ 3\triangle x_1 \triangle {x_3}^2 \\ 3\triangle x_2 \triangle {x_3}^2 \\ 3\triangle {x_3}^2 \triangle u \\ 3\triangle x_1 \triangle u^2 \\ 3\triangle x_2 \triangle u^2 \\ 3\triangle x_3 \triangle u^2 \\ 6\triangle x_1 \triangle x_2 \triangle x_3 \\ 6\triangle x_1 \triangle x_2 \triangle u \\ 6\triangle x_1 \triangle x_3 \triangle u \\ 6\triangle x_2 \triangle x_3 \triangle u \end{bmatrix}$$

With the help of equation (3)-(5) and using the relation $\overline{\mathbf{x}} = f(\overline{\mathbf{x}}, \overline{u})$, we can write the incremental discrete time *nonlinear approximate model* as,

$$\triangle \mathbf{x}_{k+1} = A\triangle \mathbf{x}_k + B\triangle u_k + \frac{1}{2} D_1 f_1(\triangle \mathbf{x}_k, \triangle u_k) + \frac{1}{6} D_2 f_2(\triangle \mathbf{x}_k, \triangle u_k) \quad (6)$$

Also, we have the *linear approximate model* as,

$$\triangle \mathbf{x}_{k+1} = A\triangle \mathbf{x}_k + B\triangle u_k \quad (7)$$

3.2 Numerical results:

The central difference approximation is used to evaluate the corresponding partial derivatives, e.g.

$$\left.\frac{\partial f(\mathbf{x},u)}{\partial x_2}\right|_{\mathbf{x}=\overline{\mathbf{x}}, u=\overline{u}} = \frac{f_F - f_A}{2 \times dx_2}$$

for state component x_2. f_F and f_A means f calculated at fore and aft of fixed point respectively as,

$$f_F = f(\overline{x}_1, \overline{x}_2 + dx_2, \overline{x}_3, \overline{u})$$
$$f_A = f(\overline{x}_1, \overline{x}_2 - dx_2, \overline{x}_3, \overline{u})$$

The fixed point obtained (Sayyad *et al.*, 2007*a*) is,

$$\bar{\mathbf{x}} = \begin{pmatrix} \bar{\theta} \\ \bar{\dot{l}} \\ \bar{\dot{\theta}} \end{pmatrix} = \begin{pmatrix} 1.3992\ rad \\ 2.9193\ m/s \\ 0.1454\ rad/s \end{pmatrix}$$

and $\bar{u} = 0.05$ m. We have chosen the following small perturbations from the fixed point:

$$dx_1 = 1.2 \times 10^{-4} \quad dx_2 = 2.0 \times 10^{-4}$$
$$dx_3 = 10 \times 10^{-4} \quad du = 0.2 \times 10^{-4}$$

We get following matrices,

$$A = \begin{bmatrix} 1.7114 & -0.03251 & 0.90066 \\ 0.91097 & 0.7947 & 0.24249 \\ 6.2283 & -0.06616 & 3.0435 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.37427 \\ 11.673 \\ 3.0052 \end{bmatrix}$$

We also calculated D_1 and D_2. In order to compare linear and non-linear appropriate model with the model originated from Poincaré return map (referred as '*actual model*'), we numerically simulated it for various cases. We enumerate some of them as follows:

- Figure 2 shows the $x_2(k+1)$ vs $x_1(k)$, the next hop-state $\dot{l}$ evaluated when only state θ is varied, keeping all other states and input u at their fixed point values.

- Figure 3 shows the error in three state variables when those next hop-state were evaluated when all states and input is varied randomly between appropriate range.

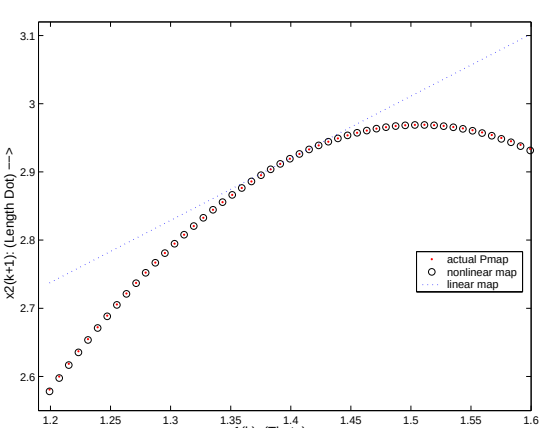

Fig. 2. Comparing actual & approximated models

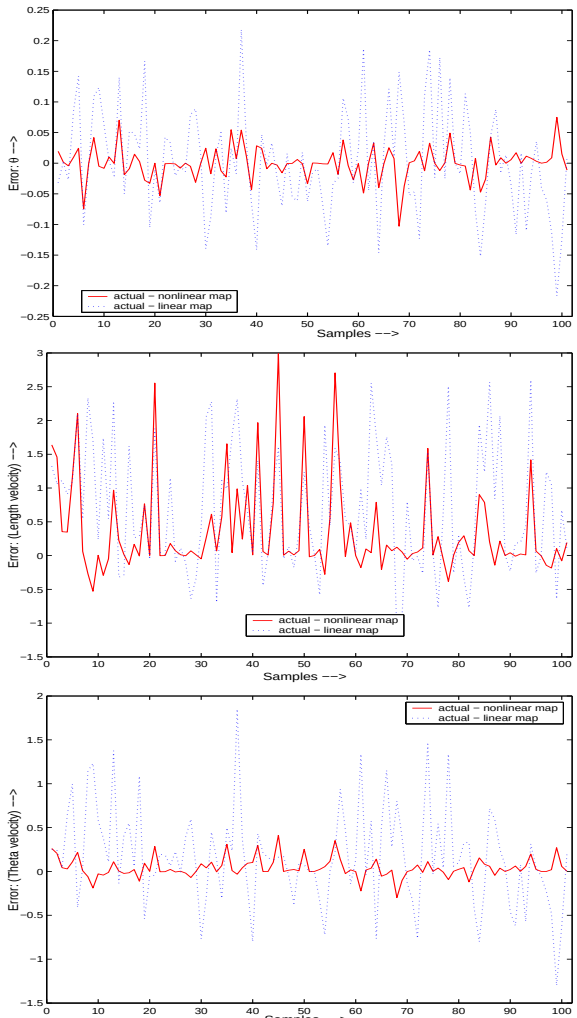

Fig. 3. Error (actual - approximate x_{k+1}) profile

4. DISCRETE-TIME STATE FEEDBACK SLIDING MODE CONTROL

Our next objective is to stabilize the hopper. Note that, the eigenvalues of state matrix A are,

$$\lambda_1 = 4.8273,\ \lambda_2 = -0.0786,\ \lambda_3 = 0.8009$$

This ensures that the open-loop system is unstable. But, (A, B) pair satisfies complete state controllability condition. In this study we designed simple state feedback based control system to stabilise hopper. We intend to design the controller for approximated incremental model and then extend it to *actual* hopper model. The schematic diagram representing this intention is shown in Figure 4. For that, we have to use equation (4). Model represented by equation (6) is considered as linear uncertain system as,

$$\triangle \mathbf{x}_{k+1} = A\triangle \mathbf{x}_k + B\triangle u_k + +\widehat{d}(x_k, u_k) \quad (8)$$

where, $\widehat{d}(x_k, u_k) = \frac{1}{2}D_1 f_1 + \frac{1}{6}D_2 f_2$, can be considered as measurable disturbance, but we found it as unmatched type disturbance (Gao *et al.*, 1995).

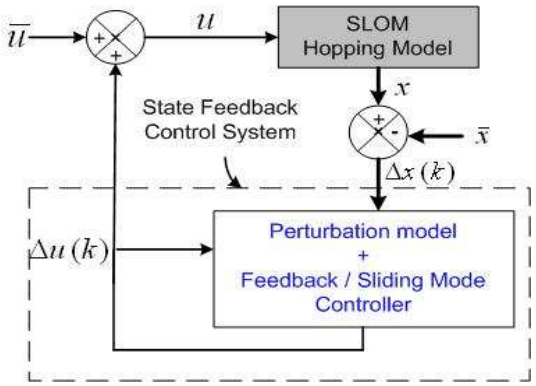

Fig. 4. Schematic diag. of feedback control system

DSMC Approach 1 (Gao *et al.*, 1995):
This approach use the linear model (7). The sliding manifold s is defined as $\{s_k = G\triangle x_k = 0\}$. Here we used the Gao's reaching law and equivalent control law as,
$s_{k+1} - s_k = -\epsilon\tau \mathrm{sgn}(s_k) - q\tau s_k$
$\triangle u_k = -(GB)^{-1}\left[GA\triangle x_k - G\triangle x_k + q\tau s_k + \epsilon\tau \mathrm{sgn}(s_k)\right]$
where, $\epsilon > 0$, $q > 0$ and $1 - q\tau > 0$

DSMC Approach 2 (Misawa, 1997):
This approach use the nonlinear model (8). Note that, the disturbance is bounded i.e. $d_L \leq \widehat{d} \leq d_U$. From Figure 3, we can calculate $\max(|\widehat{d}|)$. So, we have, $|G\widehat{d}| \leq \gamma$. The approach use saturation type reaching law and control law as,
$s_{k+1} - s_k = -K\mathrm{sat}(\frac{s_k}{\phi})$
$\triangle u_k = -(GB)^{-1}\left[GA\triangle x_k - G\triangle x_k + K\mathrm{sat}(\frac{s_k}{\phi})\right]$
where, in generality $K = \gamma + 2\Delta t\varepsilon$ and $\phi = \gamma + \Delta t\varepsilon$. Δt and ε are positive scalars, can be regarded as design parameters. The sliding manifold s is defined as the *boundary layer* so that, $\Psi = \left\{\triangle x_k \| s_k = |G\triangle x_k| \leq \phi\right\}$

In both above approaches, the switching vector G is determined using the eigenvalue assignment technique considering linear dynamics to be followed during sliding mode.

Classical Approach:
This approach use the linear model (7) and $\triangle u_k = -K_p\triangle x_k$. K_p is determined such that the eigenvalues of $(A - BK_p)$ is inside unit circle.

Simulation result:
For simulation purpose we used following controller design parameters:
$G = [0.07382 - 0.01490.3816]$ for reduced order model eigenvalues (0.8, 0.6). $\epsilon = 0.0001$, $\tau = 1$ and $q = 0.9$. While, for $\gamma = 0.5483$, we selected $\Delta t = 1$, $\varepsilon = 0.0001$, $K = 0.5603$ and $\phi = 0.5593$. Selecting eigenvalues as (0.8, 0.1, 0.6), we get $K_p = [2.4821 - 0.03801.1861]$. Figure 5 shows the typical simulated response of *actual* hopper.

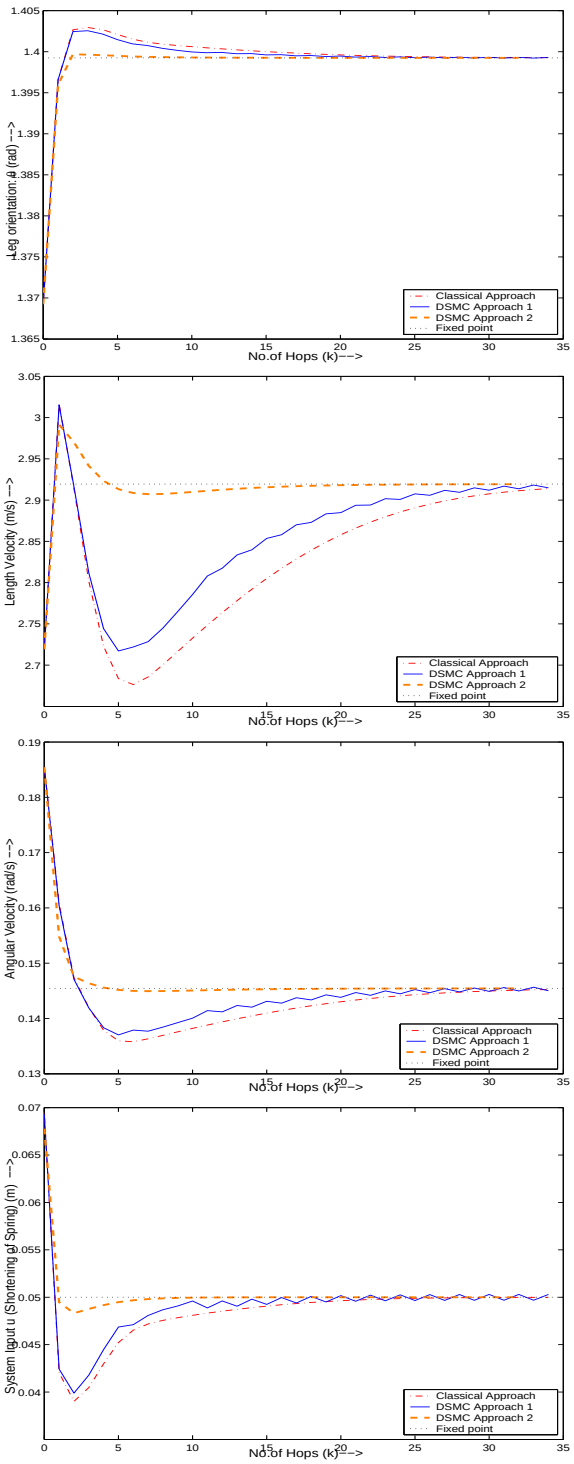

Fig. 5. Response of *actual* hopping model, when simulated for initial deviation from fixed point as: (-0.04 rad, -0.5 m/s, 0.04 rad/sec)

5. CONCLUSION

In this paper, we have proposed a planar asymmetrical hopper. In order to seek a periodic motion, we proposed an energy pumping mechanism which uses a simple linear actuator and is active only during flight. We derived a discrete dynamical system using the Poincaré map and further constructed the linear and non-linear approximate models using *Taylor series approximation*. From the simulated results, it is easy to conclude that the non-linear approximated model is better approximation to the *actual model*. Also, the explicit non-linear approximated model is in discrete time, that helped to formulate DSMC control system.

We investigated the stability with classical and DSMC based feedback strategies using simple control algorithms. It is shown that DSMC utilizing the nonlinear approximated model performs better over classical and DSMC which use only linear dynamical model. Saturation type reaching law avoid chattering as observed in Gao's reaching law case. In terms of speed of convergence, control effort required, DSMC based feedback strategy performed well. The major conclusion from this study is the possibility of stabilizing the actual non-linear model with the help of a simple feedback controller. Further research effort in DSMC system design is directed towards observer or output feedback based design.

REFERENCES

Bandyopadhyay, B. and S. Janardhanan (2006). *Discrete-time Sliding Mode Control using Multirate Output Feedback*. Springer-Verlag. Berlin.

Furuta, K. (1990). Sliding mode control of a discrete system. *System Control Letters* **14**, 145–152.

Gao, W., Y. Wang and A. Homaifa (1995). Discrete-time variable structure control systems. *IEEE Trans. Ind. Electron* **42**, 117–122.

Kuswadi, S., A. Takahashi, A. Ohnishi, M. Sampei and S. Nakaura (2003). A one linear actuator hopping robot: Modelling and control. *Advanced Robotics* **17**, 709–713.

Misawa, E. A. (1997). Discrete-time sliding mode control: The linear case. *ASME J. Dyn. Sys., Meas, and Control* **119**, 819–821.

Sayyad, A., B. Seth and K. K. Issac (2007*a*). Dynamics and control of a one-legged 2-D *SLOM* hopping robot. IFToMM World Congress. Besancon, France.

Sayyad, A., B. Seth and P. Seshu (2007*b*). Single-legged hopping robotics research- a review. *Robotica* **25**(05), 587–613.

Shanmuganathan, P. V. (2002). Dynamics and Stabilization of Under-Actuated Monopedal Hopping. Ph.d. thesis. Indian Institute of Technology, Bombay, India.

Wei, T.E., G. M. Nelson, R. D. Quinn, H. Verma and S. L. Garverick (2000). Design of a 5-cm monopod hopping robot. Vol. 3. IEEE Int. conf. on Robotics and Automation. San Francisco, CA. pp. 2828–2883.

Young, K. D., V. I. Utkin and U. Ozguner (1999). A control engineer's guide to sliding mode control. *IEEE Trans. Contr. Syst.* **7**, 328–342.