A Novel Ball on Beam Stabilizing Platform with Inertial Sensors

Part II: Hybrid Controller Design: Partial Pole Assignment & Rapid Control Prototyping

Ali Shahbaz Haider
Control Systems Laboratory
Electrical Engineering Dept
COMSATS Institute of Information Technology,
Beta Facility for Advance Automatic Control Tech.
Wah, Pakistan

Samter Ahmed
Electrical Engineering Dept
COMSATS Institute of Information Technology
Wah, Pakistan

Muhammad Bilal
Electrical Engineering Dept
COMSATS Institute of Information Technology
Wah, Pakistan

Saqib Raza
Electrical Engineering Dept
COMSATS Institute of Information Technology
Wah, Pakistan

Imran Ahmed
Electrical Engineering Dept
COMSATS Institute of Information Technology
Wah, Pakistan

***Abstract*—This research paper presents a novel controller design for one degree of freedom (1-DoF) stabilizing platform using inertial sensors. The plant is a ball on a pivoted beam. Multi-loop controller design technique has been used. System dynamics is observable but uncontrollable. The uncontrollable polynomial of the system is not Hurwitz hence system is not stabilizable. Hybrid compensator design strategy is implemented by partitioning the system dynamics into two parts: controllable subsystem and uncontrollable subsystem. Controllable part is compensated by partial pole assignment in the inner loop. Prediction observer is designed for unmeasured states in the inner loop. Rapid control prototyping technique is used for compensator design for the outer loop containing the controlled inner loop and uncountable part of the system. Real-time system responses are monitored using MATLAB/Simulink that show promising performance of the hybrid compensation technique for reference tracking and robustness against model inaccuracies.**

Keywords—stabilizing platform; ball on beam; multi-loop controller; inertial sensors; rapid control prototyping; partial pole assignment

I. INTRODUCTION

Stabilizing platforms are among challenging control systems. One of such systems is the single degree of freedom (1-DoF) ball on beam mechanism. Plant of this control problem consists of a ball capable of rolling on a beam under the action of gravity due to the inclination of the beam. The control objective is to stabilize the positions of the ball on the beam in the presence of external disturbances and to achieve ball position reference tracking. The system is open loop unstable so feedback is inevitable [1], [13].

Owing to the significance of ball on beam system a lot of research work has been dedicated to it. Classical PID controller has been implemented in [13] treating system a single input single output plant without taking in to account the internal states of the system. The observer-based model reference adaptive iterative learning controller has been demonstrated in [2]. A new technique based on geometric control has been implemented in [3], which involves designing immersion and invariance based speed and rotation angle observer for the ball and beam system. Decoupled neural fuzzy sliding mode control of the nonlinear ball on beam system has been considered in [4]. Nonlinear model predictive control for a ball and beam has been implemented in [5]. MATLAB based modeling and modulation of nonlinear ball-beam system controller has been demonstrated in [6]. A new adaptive state feedback controller for the ball and beam system is presented in [7]. Augmented state estimation and LQR control for a ball and beam system are implemented in [8]. Adaptive Neural Network for stabilization of ball on beam system has been studied in [9]. Human simulated intelligent control for ball and beam system is implemented in [10]. The Lyapunov direct method for the stabilization of the ball is presented in [11] and Energy-based balance control approach to the ball and beam system is presented in [12].

The majority of research work in the literature takes into account a reduced order model of the system by neglecting certain states in the system. In this research paper full order model of the system is stabilized using a novel method that is a hybrid of partial pole assignment and rapid control prototyping using feedback from inertial sensors. Rapid Control Prototyping is a controller testing and tuning strategy on the actual plant in the feedback loop. With the availability of low-cost high processing capability digital processors and software suits, responses of real plants can directly be obtained and evaluated for a given control law. Nowadays rapid control prototyping is industry-wide adopted because the behavior of control algorithm can directly be tested on real world plants.

This research paper is the second part of two parts research. Part-I described geometrically accurate and detailed nonlinear model of the ball on beam system followed by linearization and state space conversion. In this part-II of the research work, controller is designed for the model developed in part-I.

Organization of the paper is as follows, section-II gives a brief overview of system dynamics. Section-III comprises of multi-loop hybrid compensation design involving partial pole assignment for inner loop and rapid control prototyping for the outer loop. Section-IV presents simulation and experimental results followed by section-V describing conclusions and future work.

II. Overview of System Dynamics

Hardware platform is shown in Figure 1. Functional description for this plant is given in [1]. Position of a metallic ball capable of rolling on a beam is to be controlled. Beam consists of two parallel rods. Both rods are hollow thin cylindrical. One rod is wound by a chromium wire and the other rod has metallic conducting surface. Position of the ball is monitored by a linear potentiometer mechanism which consisting of aforementioned two rods shorted by metallic ball hence producing a voltage proportional to position of ball on the beam. An accelerometer and a rate gyro on an inertial measurement unit (IMU) board measure beam inclination angle and angular velocity respectively as shown in Figure 2.

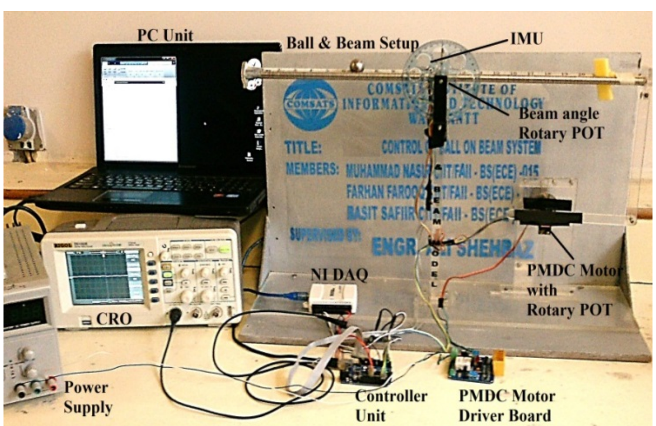

Fig. 1. Hardware platform

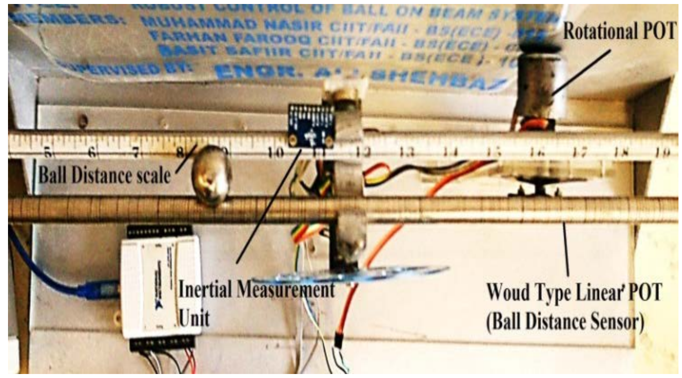

Fig. 2. Sensing mechanism

Inclination of the beam is actuated by a permanent magnet DC motor (PMDC) with its shaft coupled to a rotary potentiometer. Motor is driven by driver board. Control strategy is implemented by a digital micro controller and data acquisition card (DAQ) interfaced with MATLAB/Simulink for real time data monitoring and processing. The continuous time state space of the plant is given by (1), which has been derived in [1].

$$
\begin{aligned}
A &= \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -0.82 & 9.8 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -341.3e3 & 0 & 0 & 54.6e3 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -13.65 & 1.36 \\ 0 & 0 & 0 & 0 & 0 & -61.2e-3 & -3.26 \end{bmatrix}, \\
B &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}^{T}, \\
C &= \begin{bmatrix} 5 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2.5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3.5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 4.5 \end{bmatrix}, \\
D &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \end{bmatrix}^{T}.
\end{aligned} \tag{1}
$$

The model (1) is discretized in the MATLAB using c2d command with zero-order-hold and 0.01sec sampling interval. The discretized state space model is given by (2).

III. Controller Design

System dynamics in (2) are observable but uncontrollable. In order to stabilize the system and to achieve control objects, system in (2) is partitioned in block upper triangular configuration given by (3). The partitioning has created two subsystems as shown in Figure 3. One of these subsystems is completely controllable and observable. This subsystem is named subsystem 2 given by (4). The other subsystem is termed subsystem 1 given by (5). This partitioning into subsystems is shown in Figure 4. Our controller design strategy involves hybrid compensation in multi-loop control topology. Subsystem 2 is controlled in inner loop by unmeasured state observation followed by partial pole assignment. Controlled subsystem 2 along with subsystem 1 is compensated in outer loop using rapid control prototyping.

$$\begin{aligned}
&\underline{x_b}(k+1) = G_{bb}\underline{x_b}(k) + H_b e(k),\\
&y_b = C_{bb}\underline{x_b}(k) + D_{bb}e(k),\\
&C_{bb} = \begin{bmatrix} 3.5 & 0 & 0 \\ 0 & 0 & 4.5 \end{bmatrix},\ D_{bb} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.
\end{aligned} \tag{4}$$

$$\begin{aligned}
&\underline{x_a}(k+1) = G_{aa}\underline{x_a}(k) + G_{ab}\underline{x_b}(k) + H_a e(k),\\
&G_{ba} = \underline{0},\\
&y_b(k) = \begin{bmatrix} y_4(k)/3.5 \\ y_5(k)/4.5 \end{bmatrix}.
\end{aligned} \tag{5}$$

$$\underline{x}(k+1) = G\underline{x}(k) + He(k),$$

$$y(k+1) = C\underline{x}(k) + De(k),$$

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \\ x_4(k+1) \\ x_5(k+1) \\ x_6(k+1) \\ x_7(k+1) \end{bmatrix} = \begin{bmatrix}
1 & 9.9591\text{e-}03 & 4.8866\text{e-}04 & 1.4309\text{e-}09 & 0 & -3.8549\text{e-}11 & 2.5842\text{e-}07 \\
0 & 9.9183\text{e-}01 & 9.7599\text{e-}02 & 2.8588\text{e-}07 & 0 & -1.5289\text{e-}08 & 7.7286\text{e-}05 \\
0 & 0 & 1 & 2.9300\text{e-}06 & 0 & -4.6263\text{e-}07 & 1.5735\text{e-}03 \\
0 & 0 & 0 & 0 & 0 & -8.9984\text{e-}05 & 1.5485\text{e-}01 \\
0 & 0 & 0 & 0 & 1 & 9.3475\text{e-}03 & 6.4300\text{e-}05 \\
0 & 0 & 0 & 0 & 0 & 8.7240\text{e-}01 & 6.4300\text{e-}05 \\
0 & 0 & 0 & 0 & 0 & -5.6263\text{e-}04 & 9.6792\text{e-}01
\end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ x_4(k) \\ x_5(k) \\ x_6(k) \\ x_7(k) \end{bmatrix} + \begin{bmatrix} 6.4718\text{e-}10 \\ 2.5842\text{e-}07 \\ 7.9080\text{e-}06 \\ 1.5735\text{e-}03 \\ 2.1735\text{e-}07 \\ 6.4300\text{e-}05 \\ 9.8387\text{e-}03 \end{bmatrix} e(k), \tag{2}$$

$$\begin{bmatrix} y_1(k) \\ y_2(k) \\ y_3(k) \\ y_4(k) \\ y_5(k) \end{bmatrix} = \begin{bmatrix}
5 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 4 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 2.5 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 3.5 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 4.5
\end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ x_4(k) \\ x_5(k) \\ x_6(k) \\ x_7(k) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} e(k).$$

$$\begin{bmatrix} \underline{x}_a(k+1) \\ \underline{x}_b(k+1) \end{bmatrix} = \left[\begin{array}{c|c} G_{aa} & G_{ab} \\ \hline G_{ba} & G_{bb} \end{array}\right] \begin{bmatrix} \underline{x}_a(k) \\ \underline{x}_b(k) \end{bmatrix} + \begin{bmatrix} H_a \\ \hline H_b \end{bmatrix} e(k),$$

$$y(k) = \left[\begin{array}{c|c} C_a & C_b \end{array}\right] \begin{bmatrix} \underline{x}_a(k) \\ \underline{x}_b(k) \end{bmatrix} + De(k),$$

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \\ x_4(k+1) \\ \hline x_5(k+1) \\ x_6(k+1) \\ x_7(k+1) \end{bmatrix} = \left[\begin{array}{cccc|ccc} 1 & 9.9591\text{e-}03 & 4.8866\text{e-}04 & 1.4309\text{e-}09 & 0 & -3.8549\text{e-}11 & 2.5842\text{e-}07 \\ 0 & 9.9183\text{e-}01 & 9.7599\text{e-}02 & 2.8588\text{e-}07 & 0 & -1.5289\text{e-}08 & 7.7286\text{e-}05 \\ 0 & 0 & 1 & 2.9300\text{e-}06 & 0 & -4.6263\text{e-}07 & 1.5735\text{e-}03 \\ 0 & 0 & 0 & 0 & 0 & -8.9984\text{e-}05 & 1.5485\text{e-}01 \\ \hline 0 & 0 & 0 & 0 & 1 & 9.3475\text{e-}03 & 6.4300\text{e-}05 \\ 0 & 0 & 0 & 0 & 0 & 8.7240\text{e-}01 & 1.2503\text{e-}02 \\ 0 & 0 & 0 & 0 & 0 & -5.6263\text{e-}04 & 9.6792\text{e-}01 \end{array}\right] \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ x_4(k) \\ \hline x_5(k) \\ x_6(k) \\ x_7(k) \end{bmatrix} + \begin{bmatrix} 6.4718\text{e-}10 \\ 2.5842\text{e-}07 \\ 7.9080\text{e-}06 \\ 1.5735\text{e-}03 \\ \hline 2.1735\text{e-}07 \\ 6.4300\text{e-}05 \\ 9.8387\text{e-}03 \end{bmatrix} e(k), \quad (3)$$

$$\begin{bmatrix} y_1(k) \\ y_2(k) \\ y_3(k) \\ y_4(k) \\ y_5(k) \end{bmatrix} = \left[\begin{array}{cccc|ccc} 5 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2.5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3.5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 4.5 \end{array}\right] \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ x_4(k) \\ \hline x_5(k) \\ x_6(k) \\ x_7(k) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} e(k).$$

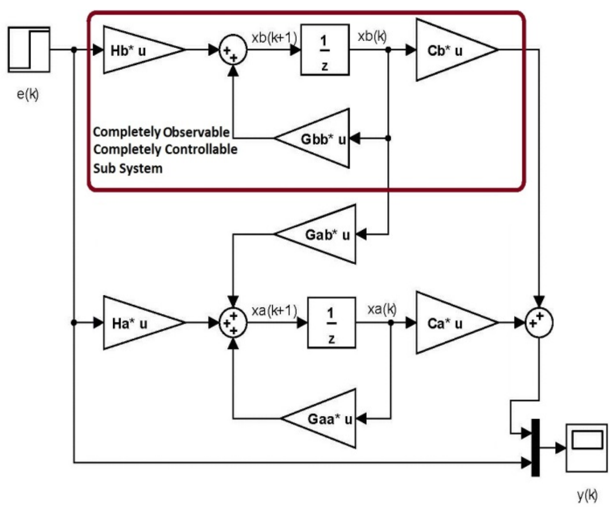

Fig. 3. System partitioning into two subsystems

A. Prediction observer for subsystem 2

In order to accomplish pole assignment for subsystem 2, we have to design observer for unmeasured states. State x_6 is unmeasured [1] in vector $\underline{x}_b$. Following the standard procedure for minimum order prediction observer design in [13], we define a similarity transformation matrix T for system in (4) such that $C_h = C_{bb}T = [\underline{I} \quad \underline{0}]$ and $x_b(k) = Tq(k)$.

$$T = \begin{bmatrix} 1/3.5 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1/4.5 & 0 \end{bmatrix}. \quad (6)$$

The new system $G_h = T^{-1}G_{bb}T, H_h = T^{-1}H_b, C_h = C_{bb}T$ is given by (7).

$$G_h = \left[\begin{array}{cc|c} 1 & 5e-5 & 2.27e-2 \\ 0 & 9.67e-1 & -2.53e-3 \\ \hline 0 & 2.77e-3 & 8.72e-1 \end{array}\right] = \begin{bmatrix} G_{haa} & G_{hab} \\ G_{hba} & G_{hbb} \end{bmatrix},$$

$$H_h = \begin{bmatrix} 7.61e-7 \\ 4.43e-2 \\ \hline 6.43e-5 \end{bmatrix} = \begin{bmatrix} H_{ha} \\ H_{hb} \end{bmatrix}, \quad (7)$$

$$C_h = \left[\begin{array}{cc|c} 1 & 0 & 0 \\ 0 & 1 & 0 \end{array}\right] = [C_{ha} \quad C_{hb}].$$

Let $K_e = [\alpha \quad \beta]$ be the observer state gain matrix. Placing the pole of observer at origin puts condition (8) on observer closed loop characteristic polynomial.

$$\left|zI - G_{hbb} + K_e G_{hab}\right| = z \qquad (8)$$

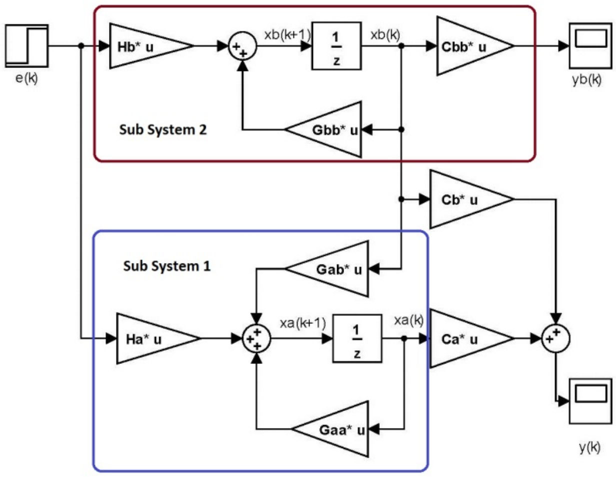

Fig. 4. Block diagram representation of subsystem1 & subsystem 2

Solution of (8) is non-unique. Assigning $\alpha = 1$ we get $\beta = -335.88$. Value $K_e = [1 \quad -335.88]$ is used in observer design algorithm (9).

$$\begin{aligned} q(k+1) &= \{H_{hb} - K_e H_{ha}\}\, e(k) \\ &+ \{G_{hba} - K_e G_{haa}\}\, y_b(k) \\ &\{G_{hbb} - K_e G_{hab}\}\{q(k) + K_e y_b(k)\} \end{aligned} \qquad (9)$$

Observed state vector is given by (10).

$$\tilde{x}_b(k) = T\left\{\begin{bmatrix} C_{hb} \\ 1 \end{bmatrix} q(k) + \begin{bmatrix} C_{ha} \\ K_e \end{bmatrix} y_b(k)\right\} \qquad (10)$$

The procedure for minimum order prediction observer design is presented diagrammatically in Figure 5.

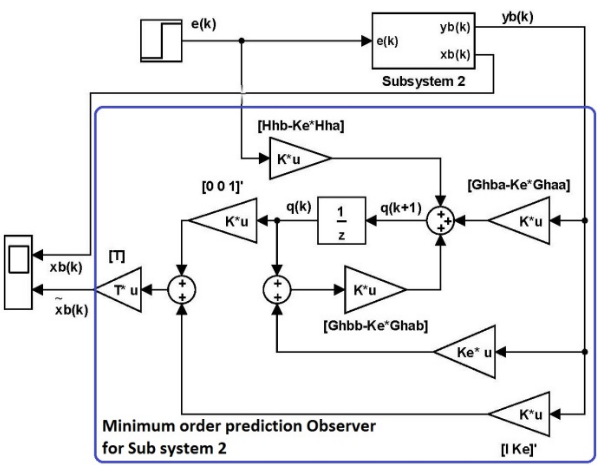

Fig. 5. Minimum order prediction observer for subsystem 2

B. Pole assignment to subsystem 2

We assign one pole at 0 and two poles at 0.8, an experimental optimal for fast response within actuator capacity. The characteristic polynomial becomes $z(z-0.8)(z-0.8)$.

Let K_b be the state gain for pole assignment then from [13] we have $K_b = \underline{\varphi}(G_{bb})M^{-1}\begin{bmatrix}0 & 0 & 1\end{bmatrix}^T$ where M is the controllability matric of subsystem 2. This expression results in state gain given by (11).

$$K_b = \begin{bmatrix}3.198e4 & 2.036e3 & 1.12e2\end{bmatrix} \qquad (11)$$

Stabilized closed loop subsystem 2 is shown in Figure 6 with new reference input $v(k)$ and signal $e(k)$ given by (12).

$$e(k) = v(k) + K_b \tilde{x}_b(k) \qquad (12)$$

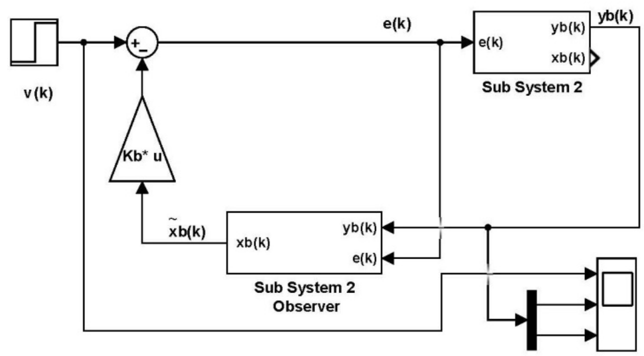

Fig. 6. Subsystem 2 stabilized by pole assignment and prediction observer

Step response of inner loop system containing observer and pole assignment is shown in Figure 7.

C. Inner loop system dynamics with stabilized subsystem 2

Using (12), (4) and (5) we get the dynamics of the overall system given by equation (13).

$$\begin{aligned} x(k+1) &= G_{s2s}x(k) + Hv(k) \\ y &= Cx(k) \end{aligned} \qquad (13)$$

System matrix in (13) with subsystem 2 stabilized is given by (14).

$$G_{s2s} = \left[\begin{array}{c|c} G_{aa} & G_{ab} \\ \hline \underline{0} & G_{bb} - H_b K_b \end{array}\right] \qquad (14)$$

To implement rapid control prototyping we treat system in (13) as single input single output system with input $v(k)$ and distance covered by ball on beam $y_1(k)$ as an output and we consider it as inner loop system given by transfer function in (15).

$$G_{IL}(z) = K_{IL} \frac{a_1 z^5 + a_2 z^4 + a_3 z^3 + a_4 z^2 + a_5 z + a_6}{z(z^5 + b_1 z^4 + b_2 z^3 + b_3 z^2 + b_4 z^1 + b_5)} \qquad (15)$$

Values of various parameters of transfer function (15) are tabulated in Table 1.

TABLE I. INNER LOOP SYSTEM PARAMETERS

Parameter	*Value*	*Parameter*	*Value*
a_1	0.0324	b_1	-4.5918
a_2	0.2928	b_2	8.4106
a_3	-0.2827	b_3	-7.6805
a_4	-0.3168	b_4	3.4965
a_5	0.2467	b_5	-0.6348
a_6	00276	K_{IL}	1e-7

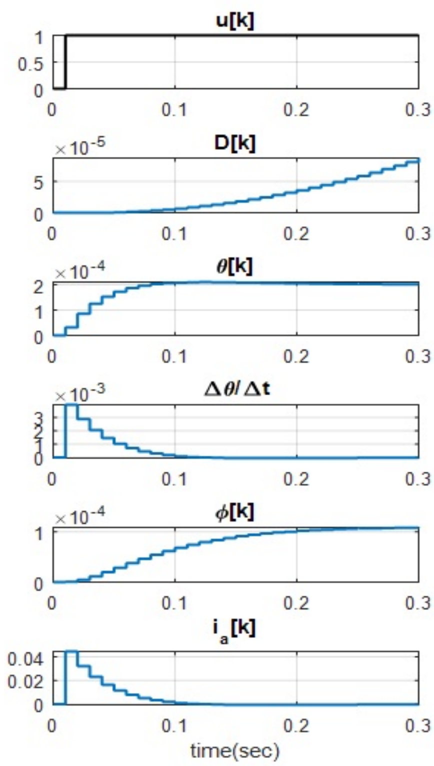

Fig. 7. Step responses for inner loop system

D. Rapid Control prototyping for inner loop

RCP implementation strategy is elucidated in Figure 8. Using hardware/software interface module i.e. NI DAQ, real plant is put into the software control loop with model compensator to be tuned. Responses of the system against various test commands are evaluated and controller parameters are adjusted accordingly until satisfactory performance is achieved.

Compensator model that has been used is given by (16).

$$C_{OL} = K_{OL} \frac{z - \varsigma}{z - \rho} \qquad (16)$$

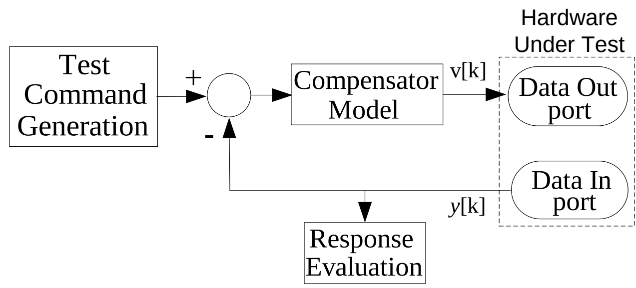

Fig. 8. Rapid Control Prototyping implementation strategy

Tuned parameter values for C_{OL} are given by (17).

$$K_{OL} = 2262.4$$
$$\varsigma = 0.9912 \qquad (17)$$
$$\rho = 0.9673$$

Overall implementation of multi loop control law that is hybrid of pole assignment and rapid control prototyping has been explained diagrammatically in Figure 9. Partial pole assignment is implemented on digital controller in inner loop followed by rapid control prototyping strategy implemented in outer loop using real time data acquisition, processing and monitoring in MATLAB. Figure 10 shows simulation of the hybrid multi-loop control algorithm. This simulation is used to obtain simulated responses in section IV. Figure 11 shows actual implementation of RCP strategy in Simulink.

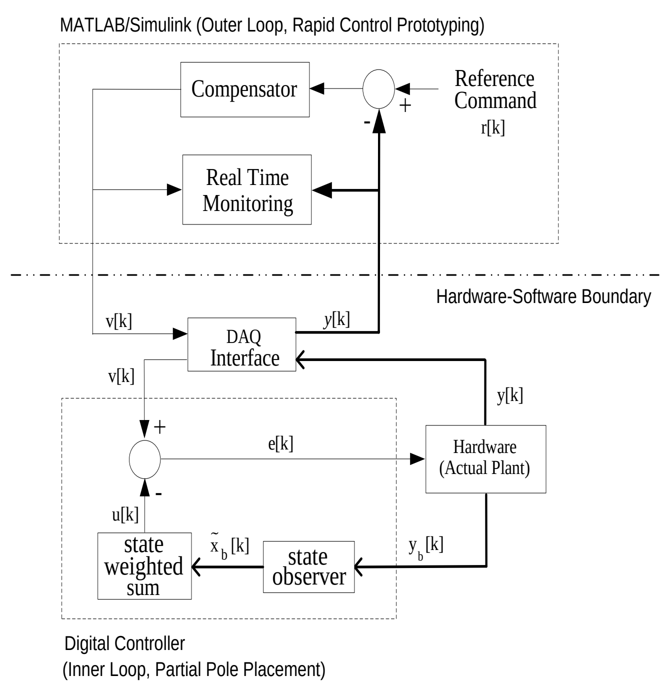

Fig. 9. Block diagram of multi loop hybrid control law implementaton

IV. SIMULATION AND EXPERIMENTAL RESULTS

The proposed hybrid multi loop control law is simulated and experimentally tested. Figure 12 shows the step response of position of the ball on beam. Actual response nearly follows simulation result. Response settles down in 1.5sec.

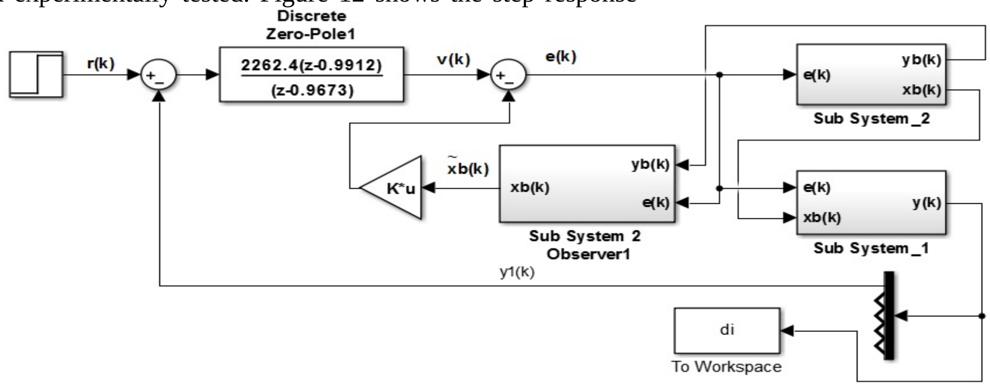

Fig. 10. Simulation of hybrid multi-loop control algorithm in Simulink

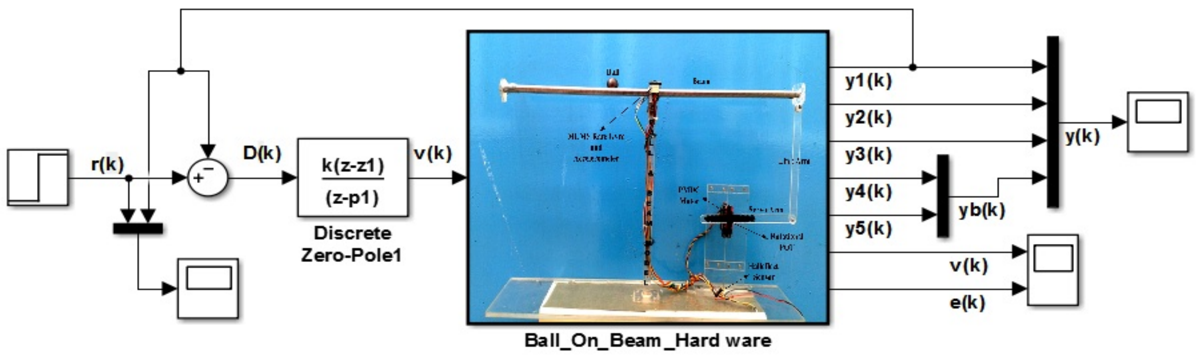

Fig. 11. Actual Rapid Control Prototyping implementation in MATLAB/Simulink

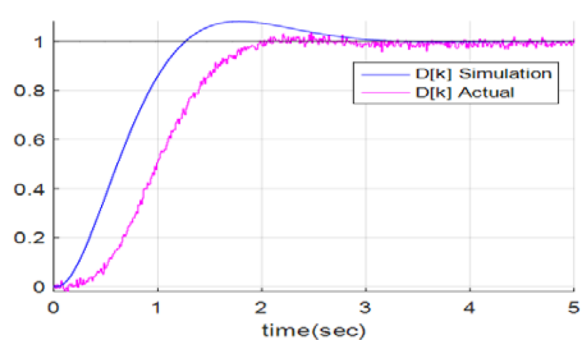

Fig. 12. Unit step response of position of ball on the beam

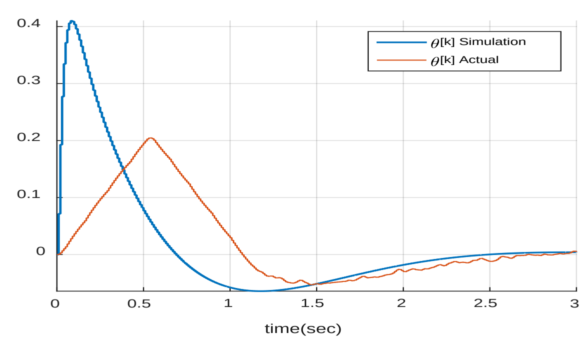

Fig. 13. Unit step response of the angle of the beam

Unit step response in Figure 12 has zero steady state error. Figure 13 shows unit step response of the beam angle. The supply limitations result in the lag in the actual response during fast transients, however the steady state response well follows the simulation response.

Figure 14 shows unit step response of servo arm angle. The lag in the actual response during fast transients is due to the supply limitations. The steady state response follows simulation response.

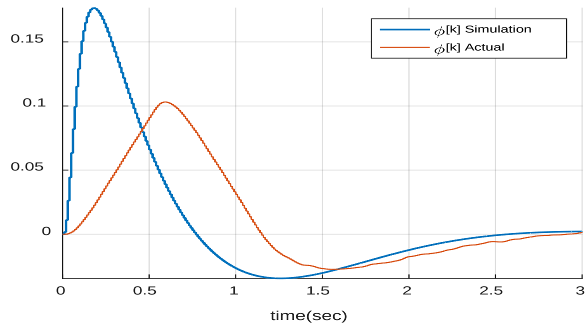

Fig. 14. Unit step response of angle of the servo arm

Figure 15 shows unit step response of PMDC motor current. Actual current waveform is limited within ±3A power supply current bounds. Figure 16 shows unit step response of beam angular velocity. Actual angular velocity of the beam is bounded by ±3A current limits of supply as shown in Figure 15. Figure 17 shows unit step response of motor input voltage. Actual input voltage waveform is bounded by ±24V power supply limits for PMDC motor driver board. Figure 18 shows the unit step response of the control algorithm signal $v(k)$ from Figure 9. The supply limitations are not included in the simulations so that we may compare actual response with ideal conditions of the simulation and monitor ideal compensator robustness against practical limitations.

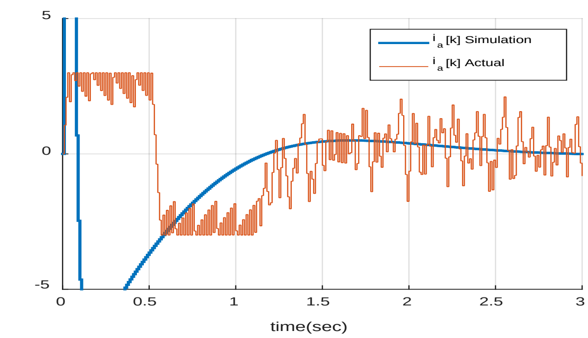

Fig. 15. Unit step response of the current of PMDC motor

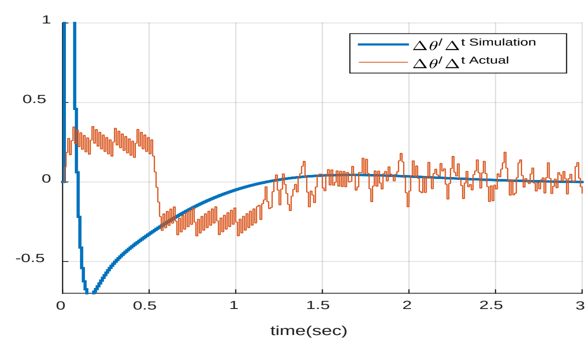

Fig. 16. Unit step response of the angular velocity of the beam

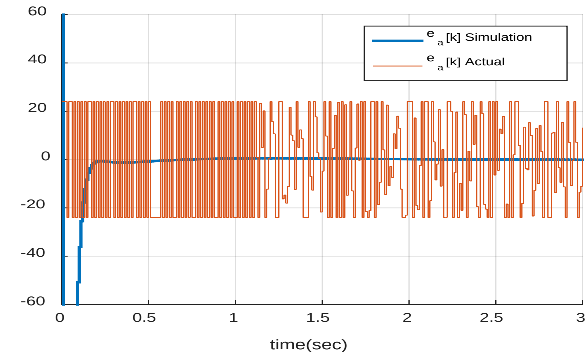

Fig. 17. Unit step response of the voltage applied to the PMDC motor

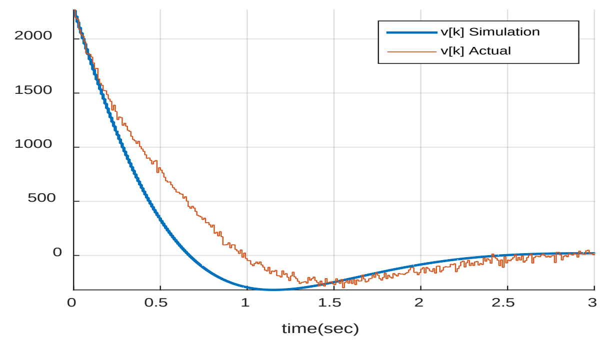

Fig. 18. Unit step response of signal v[k]

The Sinusoidal and Sawtooth reference tracking responses are shown in Figure 19 and Figure 20. Trapezoidal reference tracking response is shown in Figure 20. Actual response well follows the simulation responses with a constant steady state error for the ramp part of the reference input signal. Despite actual model has saturation limits for current and voltage yet responses well follow the simulation results. This tantamount to robustness of proposed technique against model inaccuracies.

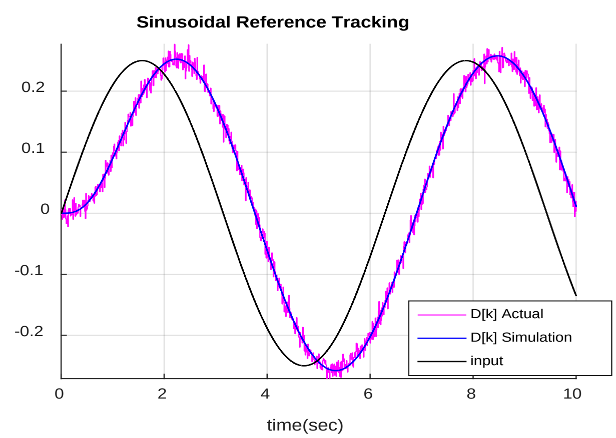

Fig. 19. Sinusoidal reference tracking response of the position of the ball

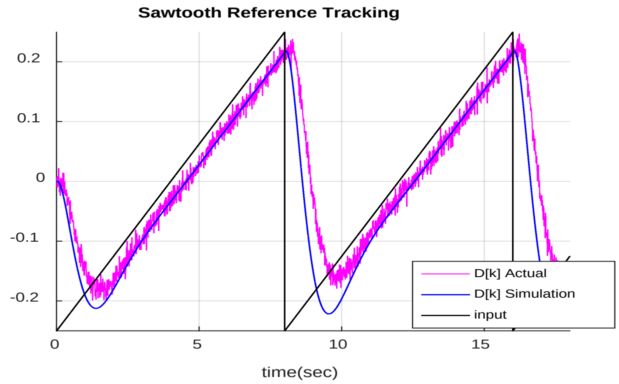

Fig. 20. Sawtooth reference tracking response for the position of the ball

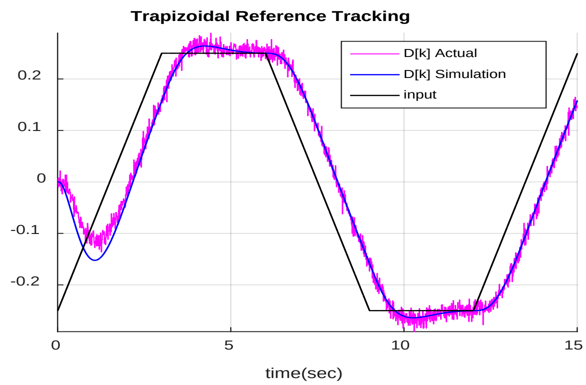

Fig. 21. Trapizoidal reference tracking response for the position of the ball

V. Conclusions

A novel compensator for the ball on beam platform is presented. Full order dynamic model of system is broken down into two parts. One part is controlled by partial pole assignment. The resulting system is compensated by rapid control prototyping. Experimental results validate that this hybrid compensator design strategy has given full control on all system outputs with system order reduction and it has given excellent results, especially regarding reference tracking and robustness against model inaccuracies.

References

[1] A. S. Haider, M. Nasir, B. Safir, and F. Farooq, "A Novel Ball on Beam Stabilizing Platform with Inertial Sensors," International Journal of Advanced Computer Science and Applications, vol. 6, no. 8, pp. 54–61, July 2015.

[2] W. Y. Chung, and C. C. Ju, "An observer-based model reference adaptive iterative learning controller for nonlinear systems without state measurement," Proc. of the 2013 International Conference on Fuzzy Theory and Its Applications, Taipei, Taiwan, pp. 171–176, Dec 2013.

[3] P. Rapp, O. Sawodny, and C. Tarin, "An immersion and invariance based speed and rotation angle observer for the ball and beam system," Proc. of the 2013 American Control Conference, Washington DC, pp. 1069–1075, June 2013.

[4] R. M. Nagarale, and B. M. Patre, "Decoupled neural fuzzy sliding mode control of nonlinear systems," Proc. of the 2013 IEEE International Conference on Fuzzy Systems, Hyderabad, India, pp.1–8, July 2013.

[5] D. Martinez, and F. Ruiz, "Nonlinear model predictive control for a Ball&Beam," Proc. of the 2012 IEEE 4th Colombian Workshop on Circuits and Systems, Barranquilla, Colombia, pp. 1–5, Nov 2012.

[6] C. Wenzhuo, S. Xiaomei, and X, Yonghui, "Modeling and modulation of nonlinear ball-beam system controller based on matlab," Proc. of the 2012 9th International Conference on Fuzzy Systems and Knowledge Discovery, Shenyang, China, pp. 2388–2391, May 2012.

[7] D. N. Kouya, and F. A. Okou, "A new adaptive state feedback controller for the ball and beam system," Proc. of the 2011 24th Canadian Conference on Electrical and Computer Engineering, Niagara Falls, Canada, pp. 247–252, May 2011.

[8] P. Z. Hua, Z. Geng, and L. C. Xiang, "Augmented state estimation and LQR control for a ball and beam system," Proc. of the 2011 6th IEEE Conference on Industrial Electronics and Applications, Beijing, China, pp. 1328–1332, June 2011.

[9] W. Wei, and X. Peng, "A Research on Control Methods of Ball and Beam System Based on Adaptive Neural Network," Proc. of the 2010 International Conference on Computational and Information Sciences, Chengdu, China, pp. 1072–1075, Dec 2010.

[10] P. Xu, L. Lei, and Z. Tan, "Human simulated intelligent control for ball and beam system," Proc. of the 2010 29th Chinese Control Conference, Beijing, China, pp. 96–100, July 2010.

[11] A. I. Carlos, "The Lyapunov direct method for the stabilisation of the ball on the actuated beam," International Journal of Control, vol 12, no. 82, pp. 2169–2178, Oct 2009.

[12] E. Lia, Z. Z. Lianga, Z. G. Hou, and M. Tana, "Energy-based balance control approach to the ball and beam system," International Journal of Control, vol. 6, no. 82, pp. 981–992, May 2009.

[13] K. Ogata. Modern Control Engineering. 3rd ed., New Jersey: Prentice Hall, 1997.