14th International Conference on
AEROSPACE SCIENCES & AVIATION TECHNOLOGY,
ASAT - 14 – May 24 - 26, 2011, Email: asat@mtc.edu.eg
Military Technical College, Kobry Elkobbah, Cairo, Egypt
Tel: +(202) 24025292 –24036138, Fax: +(202) 22621908

Robust CLOS Guidance and Control: Part-3: HIL Systems Simulation

A.N. Oda[*], G.A. El-Sheikh[†], Y.Z. El-Halwagy[*], M. Al-Ashry[*] and M.A. Abd-Altief

Abstract: The great developments in applied mathematics and computational capabilities facilitate the design and implementation of robust control. In addition, the huge developments in nanotechnology and its availability in civilian level with less cost, size and weight attract many of the researchers allover the world towards embedded systems especially the embedded flight control. Among the real applications are the guided missiles especially the antitank guided missile systems which are commanded to the line of sight (CLOS) against ground and short range targets. The present work is concerned with improving the performance of an antitank guided missile system belonging to the first generation via robust synthesis of autopilot and guidance systems. The design and analysis necessitates somehow accurate model with different uncertainties (objective of Part-1 of the paper) for the system, a robust autopilot design (objective of Part-2 of the paper) and implementation via hardware in the loop (HIL) simulation (objective of Part-3 of the paper).

This part of the paper is devoted to conduct the simulation with Jetvator hardware (pitch and yaw) in the loop and the robust autopilot obtained in the previous part of this paper. It starts by identifying the Jetvator dynamics on-line and within the closed loop from which the transfer function is used in justifying the designed autopilot via time responses and the 6DOF simulation. Then, the Jetvator hardware is implemented within the 6DOF simulation via interfacing cards using the designed robust autopilot in presence of the prescribed sources of uncertainty. The results show the model accuracy via using the HIL simulation with system-identification which has clear effect upon enhancing the system performance and gives the green light for the next step of implementing the designed robust autopilot with the HIL. The obtained results are very promising clarifying the autopilot capability to stabilize the system (in pitch and yaw) with the HIL and in presence of disturbance and measurement noise.

Keywords: Command Guidance Systems, CLOS, Robust Control, HIL, Systems Identification

1. Introduction

The objectives of autopilot and guidance synthesis necessitate the availability of accurate system models among which is the transfer function. Transfer functions representing the missile-control system dynamics in pitch and yaw planes are identified via hardware in the loop (HIL) simulation and considered for design, investigation and validation against previous work and reference flight data. This investigation includes experiment design, on-line systems identification, and evaluating the identified control system (Jetvator) model within the 6DOF

[*] Egyptian Armed Forces, Egypt
[†] Professor, , gaelsheikh@gmail.com, Tel. 02,01002682402

simulation such that the performance requirements are achieved. Hardware-In-the-Loop (HIL) is a form of real-time simulation where a real component is implemented within the loop and this component may be an electronic control unit or electromechanical element. The purpose

of HIL system is to provide all of possible electrical and mechanical stimuli needed to fully exercise the flight path trajectory without damaging the equipment and getting more accurate modeling. For the underlying missile guidance system, the missile airframe and its control system are the major building blocks that necessitate mature designs [11,12]. However, due to the nature of the missile flight, its nonlinear aerodynamic characteristics can be linearized at some conditions and consequently their transfer functions are obtained for the design and analysis. The obtained transfer dynamics are utilized in autopilot design with which the system response (time and frequency) to a step jet deflection is evaluated. The evaluation is carried out for different engagement scenarios with and without inserting the hardware in the loop, where the hardware includes the jetivator drives with the electronic driver (E-Driver) of the missile. The results show somehow accurate identified model with clear effect upon enhancing the system performance and gave the green light for robust controller design. The robustness of the obtained model diminishes the effects of un-modeled dynamics towards enhanced system performance and gives the base for robust controller design.

2. Hardware Implementation

The system model should predict the input-output response in such a way that can be used to design a control system with high confidence level to work with the real physical system. There are two ways to get a system model: first by applying the laws of physics and the second is by conducting experiments on the physical system itself with systems identification [3]. The second approach utilizes the hardware in the loop (HIL) simulation in which the autopilot section is disassembled from a real missile and interfaced with the computer simulation program via data acquisition module.

2.1 Experiment Structure

The autopilot disassembled hardware section consists of actuator and jetivator assembly, Electronic Driver and resolver locked at normal position. The actuator and jetivator assembly and EDriver are attached together via two potentiometers to feedback the nozzle movement. The input signal to the hardware assembly is the subtraction of the command current (through the wire link) and the gyro feedback signal. The output of the hardware is the nozzle movement voltage which picked up through the feedback potentiometer. This voltage is transformed into jet deflection angle inside the simulation program. The input signal to the hardware section and the potentiometer voltage are transmitted from the simulation program to the hardware section and vice versa through a multi-channel input-output data acquisition module. The hardware assembly has an external power supply and a source of pressed air necessary for the actuator servo, Fig. 1. This Figure illustrates a view for the real hardware section after disassembled from the missile and the construction of the hardware interface experiment including the control section, computer and data acquisition module, power supply, pressed air source, and level gain switch. The electronic driver has two gains; low gain for booster thrust and high gain for sustainer. They are physically controlled through pressure switch mounted on missile body and works during the missile flight. Thus, for the simulation, the pressure switch is replaced by an electric relay energized from the simulation program according to thrust value by a pulse via the data acquisition output channel. The input and output of the hardware section are recorded during the simulation run at each sample of time along the missile flight.

2.2 HIL Experimental Results

The control section simulation part is removed from the flight simulation model and replaced by acquiring the hardware via the data acquisition card. The results for the HIL experiments and the pure simulation for control section interior signals waveforms are shown in Figs. 2-5. In addition, the flight path trajectories for both cases and the reference trajectory are shown in Fig. 6,7 using target at range (R_{tx}) = 2800 [m], velocity (V_t) = 0 [m/sec], pitch LOS angle ($\theta_s = 0^\circ$), and yaw LOS angle ($\Psi_s = 0^\circ$).

The difference between simulation, reference and HIL results can be returned back to the 60 Hz desired oscillation in the jetivator movement, the noise acting in real hardware, unmodeled dynamics in control section model and the difference in applied guidance equation or guidance law. However, in case of missile guidance system, it's necessary to have a 6DOF model much close to the real system as possible.

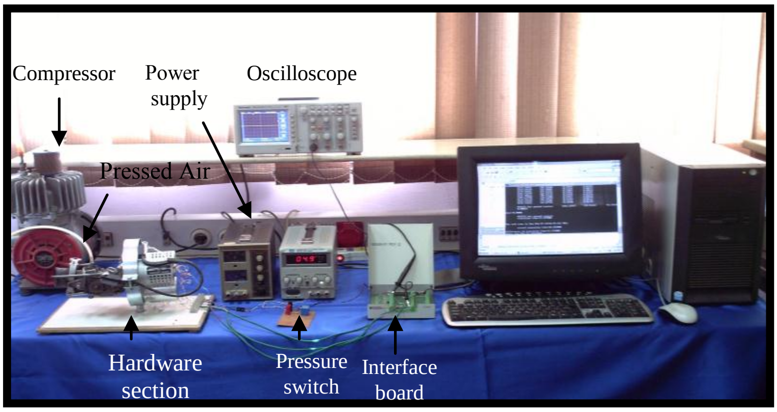


Fig. 1a: Hardware in the loop experiment

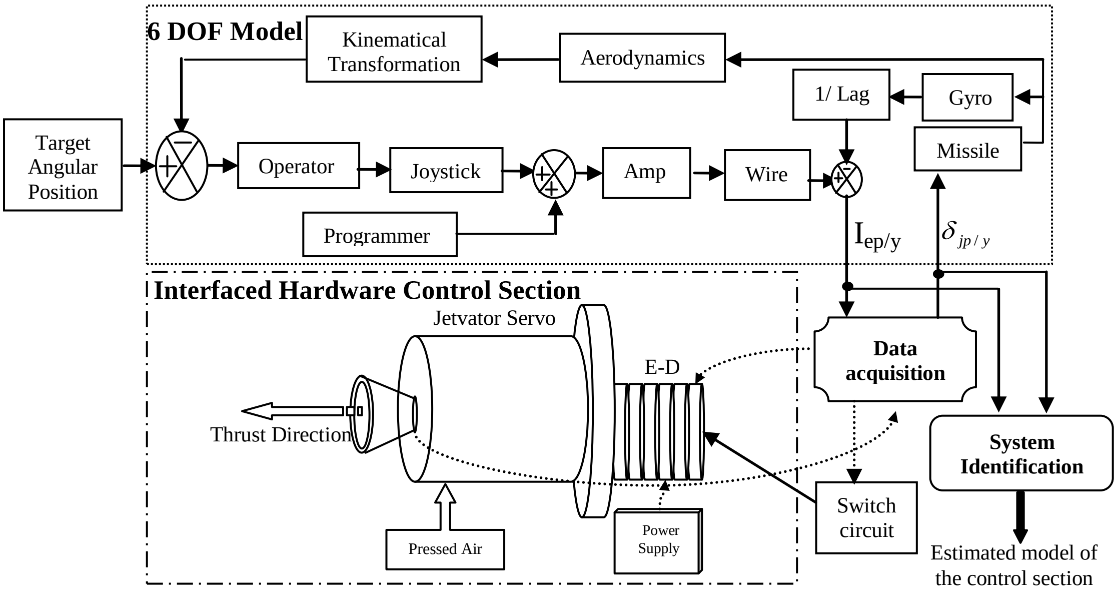
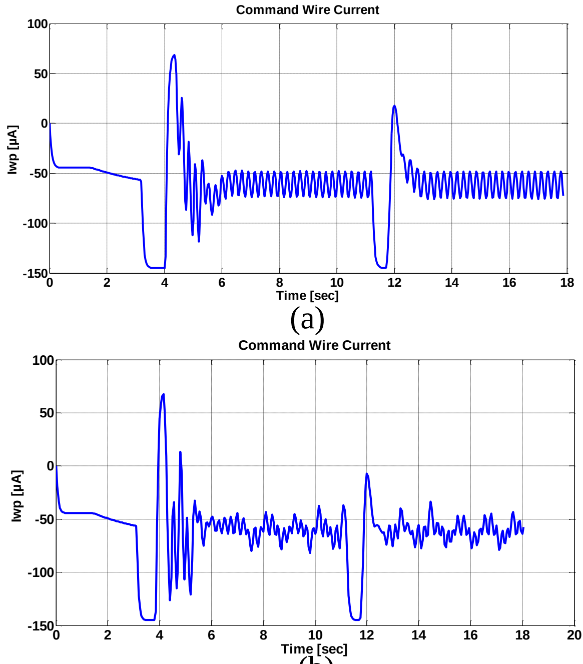


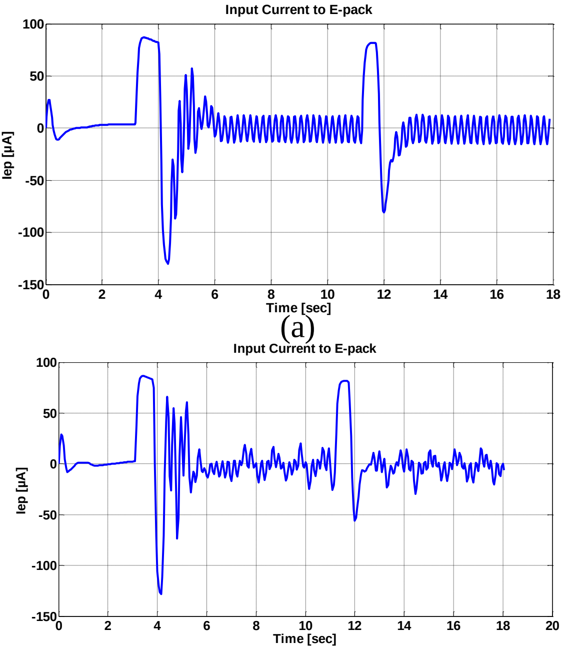
Fig. 1b: Hardware in the loop simulation



(a)

(b)

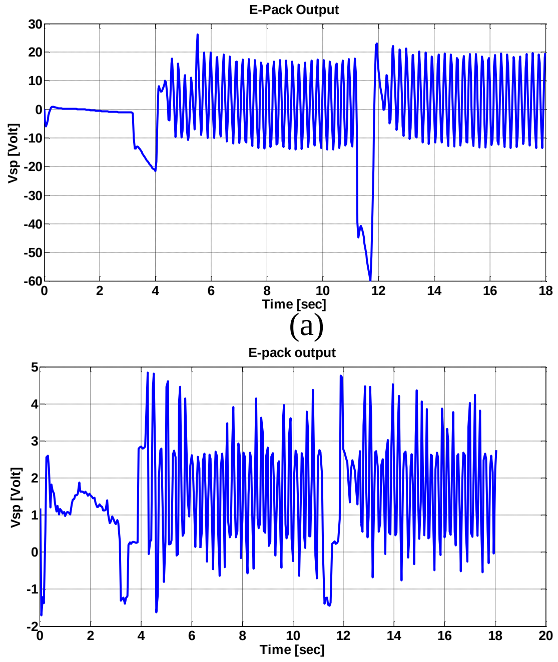
Fig. 2: Pitch wire current
(a) Simulation and (b) HIL



(a)

(b)

Fig. 3: Pitch E-Driver input current
(a) Simulation and (b) HIL



(a)

(b)

Fig. 4: Pitch E-Driver output voltage
(a) Simulation and (b) HIL

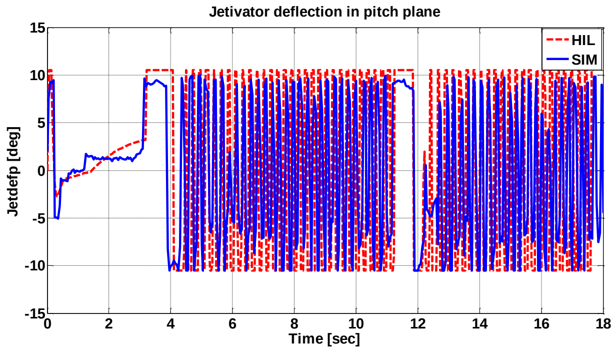


Fig. 5: Jet deflection angle
using Simulation and HIL

3. System Identification

Systems identification is the field of modelling systems dynamics from experimental data. That is, it provides mathematical models of dynamic systems based on observed input-output data. The obtained model is validated via comparing its output (predicted) with the observed data. A dynamic system can be conceptually described as shown in Fig. 8, where it is driven by input control variables U(t) and random disturbance $\xi(t)$, while the output signals is Y(t). The procedure of identifying system's parameters includes the some steps as shown in Fig. 9 [2,9,12,14]:

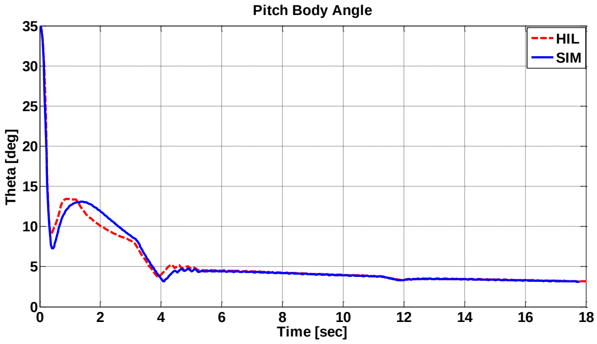


Fig. 6: Missile body angle (Simulation and HIL)

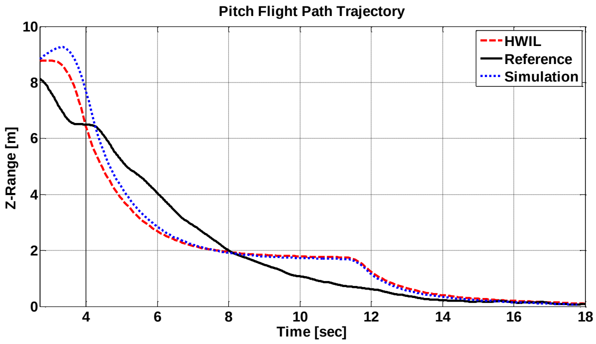


Fig. 7: Missile pitch plane trajectories: reference, 6DOF model, and HIL

- **Experiment design:** is devoted to obtain good experimental data (persistent exciting) and it includes the choice of the input and the measured output signals.
- **Model structure:** a suitable model structure is chosen using prior knowledge and trade-off approach.
- **Choice of the fit criterion:** a suitable cost function is chosen to reflect how well the model fits the experimental data.
- **Parameter estimation:** the optimization problem is solved to obtain the numerical values of the model parameters in the form of poles and zeros (polynomial or transfer function) or the state space matrices.
- **Model validation:** the model is tested in order to reveal any inadequacies against the measured output.

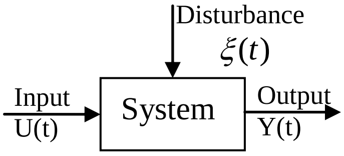


Fig. 8: A Dynamic system

Estimating the system models might be carried out using either parametric or nonparametric methods.

3.1 Nonparametric Methods

The non-parametric methods are characterized by the property that the resulting models are curves or function, which are not necessarily parameterized by a finite dimensional parameter vector [13]. The following methods can be used:

- **Transient Analysis:** With this approach the model used is either step or the impulse response of the system. The use of an impulse as input (delta dirac function $\delta(t)$) is common practice in certain application. This method does not impose any structure assumptions about the system, other than that it is linear, which can be described by the impulse response (g_k), with the property that:

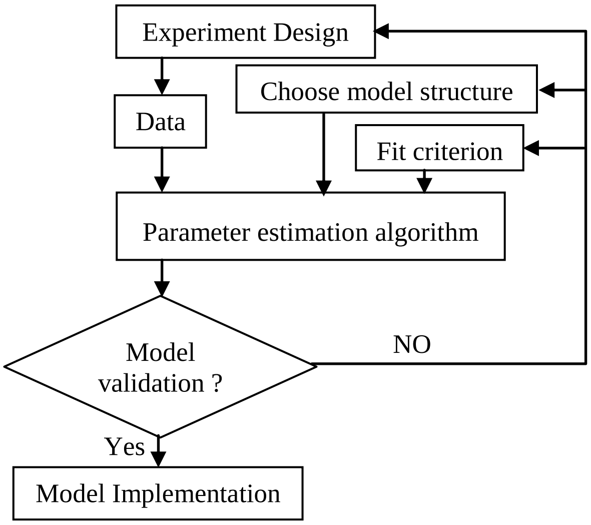


Fig. 9: System Identification procedure

$$y(t)=\sum_{k=1}^{\infty} g_k\, u(t-k) \qquad (1)$$

where u is the input, y is the output, t is the time, and k is the order of sampling or period. The name of this method derives from the fact that if the input $u(t)$ is an impulse, {i.e., $u(t)=1$ when $t=0$ and $u(t)=0$ when $t>0$} then the output $y(t)$ will be $y(t)=g_t$. It can be illustrated that transient analysis gives rough estimates of time delay. For an impulse response the time delay is equal to the first positive peak in the transient response magnitude that is greater than the confidence region for positive time values.

- **Frequency Analysis:** The frequency response of a linear system is the Fourier transform of its impulse response. This description of the system gives considerable engineering insight into its properties, where the relation between input and output is often written as:

$$\begin{aligned} y(t) &= G(q)\,u(t)+\xi(t) \\ \xi(t) &= H(q)\,e(t) \end{aligned} \qquad (2)$$

where $G(q)$ is the transfer function, $\xi(\mathrm{t})$ is the additive disturbance, $e(t)$ is a white noise, $q=e^{i\omega T}$ with ω is angular frequency and T is the sampling interval.

Correlation Analysis: The better understanding of the algorithm underlying correlation analysis is obtained via considering the description of a dynamic system given in Eq$^{\underline{n}}$ (2), in which *G(q)u(t)* notation represents the following operation: $G(q)u(t)=\sum_{k=1}^{\infty} g(k)u(t-k)$; where q is the shift operator, defined by $G(q)=\sum_{k=1}^{\infty} g(k)q^{-k}$ i.e. $q^{-1}u(t)=u(t-1)$. For impulse response, the algorithm estimates impulse response coefficients *g* for both the single- and multiple-output data. The impulse response is estimated as a high-order, non causal (finite impulse response) FIR model:

$$y(t)=g(-m)u(t+m)+...+g(-1)u(t+1)+g(0)u(t)+g(1)u(t-1)+...+g(n)u(t-n)$$

The estimation algorithm pre-filters the data such that the input is as white as possible. It then computes the correlations from the pre-filtered data to obtain the FIR coefficients. The variable *g* is also estimated for negative lags, which takes into account any non causal effects from input to output that can result from feedback. The coefficients are computed using the least-squares method.

- **Spectral Analysis:** Spectral analysis can be applied with rather arbitrary inputs, where the transfer function is obtained in the form of bode plot or other equivalent form.

As discussed before, nonparametric methods are easily to apply but give only moderately accurate models that are not easily interpreted. For high accuracy and good interpretation a parametric method has to be used. In such cases non parametric methods can be used to get a first crude model, which may give useful information, such as time delay, on how to apply the parametric method.

3.2 Parametric Methods

Alternatively, parametric methods necessitate a specific model structure in which the parameters are estimated using the observed input-output data. This approach opens up a large variety of possibilities corresponding to different structures describing the system [1,4,6,15], the simple form of which is shown in Fig. 10 as:

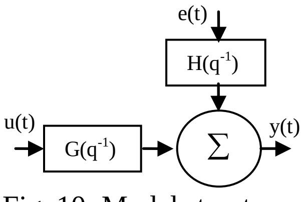


Fig. 10: Model structure

$$y(t)=G(q^{-1})\,u(t)+H(q^{-1})\,e(t) \qquad (3a)$$

where y(t) is the n_y output at time t and u(t) is the n_u input, e(t) is a sequence of independent and identically distributed random variables, the argument q^{-1} denotes the backward shift operator, and $G(q^{-1})$ and $H(q^{-1})$ are two filters with finite orders. There are several structure of different models that can be described as follows [5,8]:

- **ARX** models has the form

$$A(q^{-1})\, y(t) = G(q^{-1})\, u(t) + e(t) \qquad (3b)$$

- **ARMAX** model has the form

$$A(q^{-1})\, y(t) = G(q^{-1})\, u(t) + H(q^{-1})\, e(t) \qquad (3c)$$

- **Output-Error (OE)** models has the form

$$y(t) = \frac{G(q^{-1})}{F(q^{-1})} u(t) + e(t) \qquad (3d)$$

- **Box-Jenkins (BJ)** models has the form

$$y(t) = \frac{G(q^{-1})}{F(q^{-1})} u(t) + \frac{H(q^{-1})}{D(q^{-1})} e(t) \qquad (3e)$$

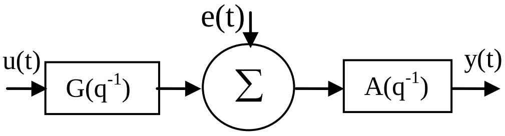


Fig. 11a: ARX model structure

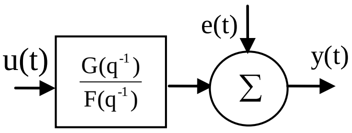


Fig. 11c: OE model structure

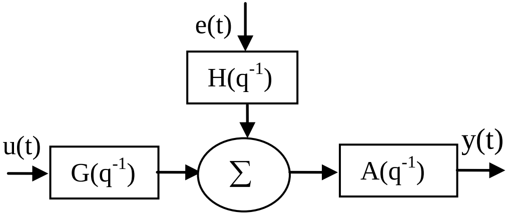


Fig. 11b: ARMAX model structure

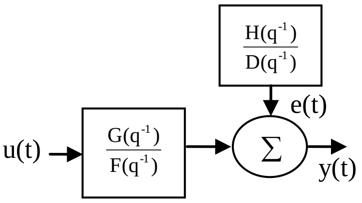


Fig. 11d: BJ model structure

State-Space: Another common way of describing linear systems is to use the state-space form:

$$\begin{aligned} \dot{x}(t) &= A\,x(t) + B\,u(t) \\ y(t) &= C\,x(t) + D\,u(t) + v(t) \end{aligned} \qquad (4)$$

where the relationship between the input $u(t)$ and the output $y(t)$ is defined via the n_x-dimensional state vector $x(t)$. In transfer function form, $\text{Eq}^{\underline{n}}$ (4) corresponds to $\text{Eq}^{\underline{n}}$ (2) with the following equality:

$$G(q) = C(q\,I_{nx} - A)^{-1} B + D \qquad (5)$$

where I_{nx} is the n_x-by-n_x identity matrix. Clearly $\text{Eq}^{\underline{n}}$ (4) can be viewed as one way to parameterize the transfer function where $G(q)$ is a function of the state space matrices *A*, *B*, *C*, and *D* in $\text{Eq}^{\underline{n}}$ (5). To further describe the character of the noise term $v(t)$ in Eq^{n} (4), a more flexible innovations form of the state-space model can be used as follows:

$$\begin{aligned} \dot{x}(t) &= A\,x(t) + B\,u(t) + K\,e(t) \\ y(t) &= C\,x(t) + D\,u(t) + e(t) \end{aligned} \qquad (6)$$

The state-space matrices *A*, *B*, *C*, *D*, and *K* in Eq$^{\underline{n}}$ (6) can be estimated directly, without first specifying any particular parameterization by efficient *subspace methods* [8]. If the sequence of state vectors *x*(*t*) were known together with *y*(*t*) and *u*(*t*), Eq$^{\underline{n}}$ (6) would be a linear regression, and *C* and *D* could be estimated by the least squares method. Then, *e*(*t*) could be determined and treated as a known signal in Eq$^{\underline{n}}$ (6), which would be another linear regression model for *A*, *B*, and *K*. One could also treat Eq$^{\underline{n}}$ (4) as a linear regression for *A*, *B*, *C*, and *D* with *y*(*t*) and *x*(*t+1*) as simultaneous outputs, and find the joint process and measurement noises as the residuals from this regression. The Kalman gain *K* could then be computed from the pertinent Riccati equation. Thus, once the states are known, the estimation of the state-space matrices is easy. All states in representations like Eq$^{\underline{n}}$ (6) can be formed as linear combinations of the *k*-step ahead predicted outputs (k = 1,2,...,n). Thus, it is a matter of finding these predictors and then selecting a basis among them. The subspace methods form an efficient and numerically reliable way of determining the predictors by projections directly on the observed data sequences [5].

System Identification Algorithm: The identification process amounts to repeatedly selecting a model structure, computing the best model in the structure, and evaluating this model's properties to justify its satisfaction. Therefore, a MATLAB environment is established to conduct the system identification steps and identify transfer function models for the implemented hardware using experimental data gained from the flight simulation program as shown in Fig. 12.

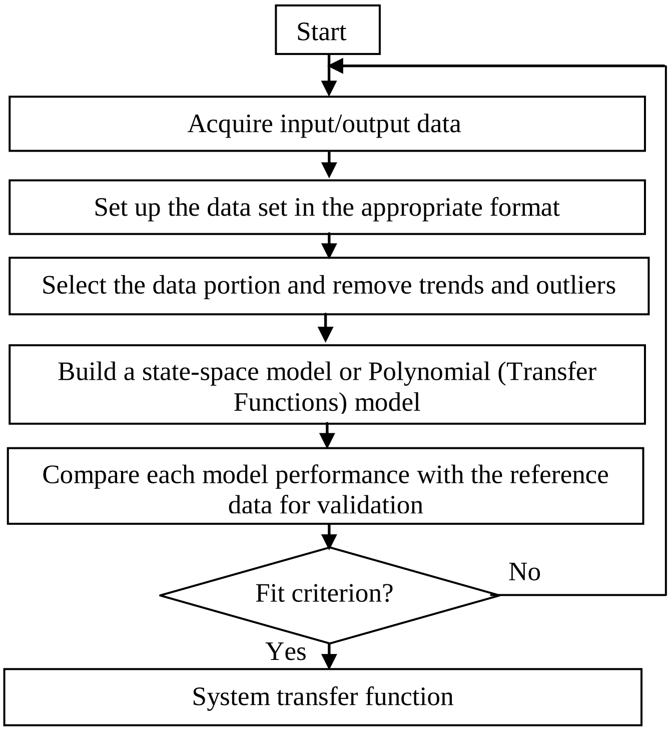

Start
Acquire input/output data
Set up the data set in the appropriate format
Select the data portion and remove trends and outliers
Build a state-space model or Polynomial (Transfer Functions) model
Compare each model performance with the reference data for validation
Fit criterion?
No
Yes
System transfer function

Fig. 12: System identification flowchart

4. Guidance System Identification and Analysis

The autopilot is a minor closed loop inside the main guidance and control system. The underlying missile system employs a gyro control autopilot as shown in Fig. 13 which describes a simplified block diagram of the missile guidance and control loop including the autopilot. The deviation of the missile LOS from that of the target is measured by the operator which generates the guidance commands according to the adopted guidance law and the instantaneous positions of the missile and target. These guidance commands are used to control the missile motion via the autopilot. The difference between the demanded and actual acceleration is produced due to mechanical limiting in the jetivator assemble to prevent excessive acceleration demands beyond the missile structural capability. In addition, it prevents the missile from taking larger angle of attack that can cause instability or stalling. The magnitude of acceleration that the missile can perform is determined by its ability to stay in one flight condition despite the aerodynamic loading. The autopilot contains a pneumatic jetivator assembly, two gyros for pitch and way channels and the electronic circuits.

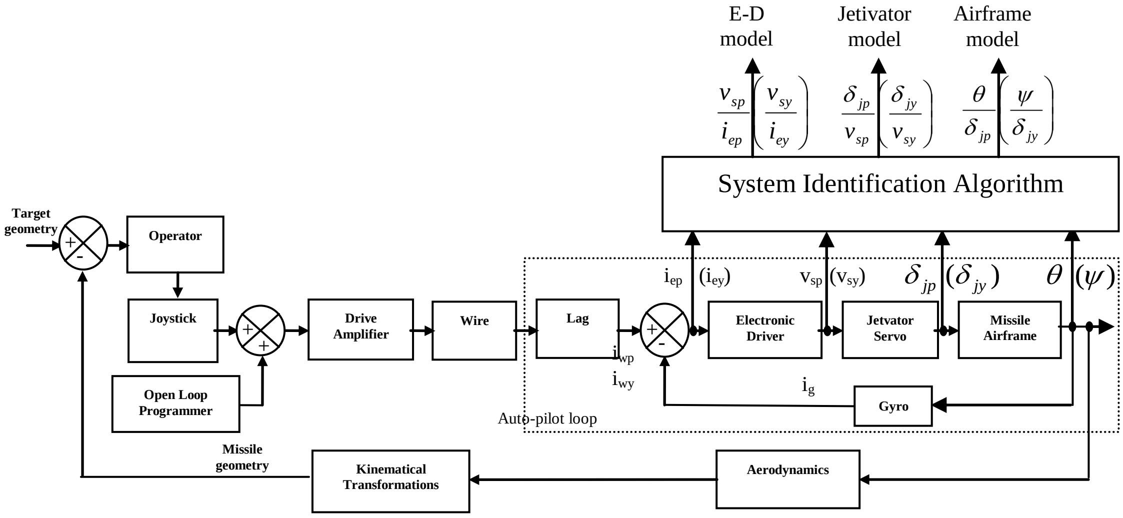


Fig. 13: Guidance and control loop

4.1 Electronic Driver (E-D)

It is the electronic circuit which drives the actuator servos and represents the autopilot controller of the underlying system as shown in Fig. 14. The command currents through the wire in both yaw and pitch channels are the electronic driver inputs while the output is a pulse width modulated (PWM) voltage used to drive a pair of solenoids in each plane. The input current at each channel is converted to the corresponding voltage with range of ± 5V through a lag circuit. This voltage is subtracted from the missile position voltage which picked up from gyros. The total voltage is chopped by 5 KHz chopper with 90° shift in yaw channel. Both signals are multiplied by low gain during the boost time and high gain in the sustain time and the gain is switched from low to high by a pressure switch mounted on the missile body. Then, the pitch and yaw command signals are summed and filtered by a 5 KHz filter. The resolver output is 5 KHz sinusoidal wave shifted according to the value of roll angle. That is, the resolver function is to pick up this information according to the roll angle. Then, a zero detection demodulating circuit pick the value of the filtered signal when the resolver voltage crosses the zero value and yields the output signal that represents the demodulated signal of yaw channel. The pitch demodulated signal is produced by applying zero detection demodulation sequence to 90° shifting of filtered signal. The previous sequence is used in order to shift the command value from pitch to yaw channels and verses while the missile flying according to the value of the roll angle. A PWM signal is produced by triangle 60Hz signal at the gate of the power amplifier to drive each pair of the solenoids in each channel. The amplifier opens when the triangle signal crosses certain level (threshold level), while at zero command the opening and closing duration are equal. When the command is received, the modulation voltage shifts the triangle signal upward or downward the amplifier opening level, so the amplifier working period change according to the value and direction of the command signal.

The transfer function which gives the relationship between the error command current (i_{ep} and i_{ey}) as input to the E-Driver and the output voltage (V_{sp} and V_{sy}) from the E-Driver in pitch and yaw planes, respectively, is derived from the implementation of the hardware control section within the simulation program. The input-output data are used in the identification program to find out the final transfer function for the E-Driver within the duration of flight time. Since the E-Driver has two modes of operation low gain at booster and high gain at sustainer, there are two transfer functions describing the E-Driver. The first one from lunching instant up to about 5.2 [sec] at the end of booster, while the second for the rest of the missile flight. That is, the identification procedure is applied to the recorded data for the first period to yield the estimated E-Driver transfer function at booster time while the second period is devoted to estimating the E-Driver transfer function at sustainer time.

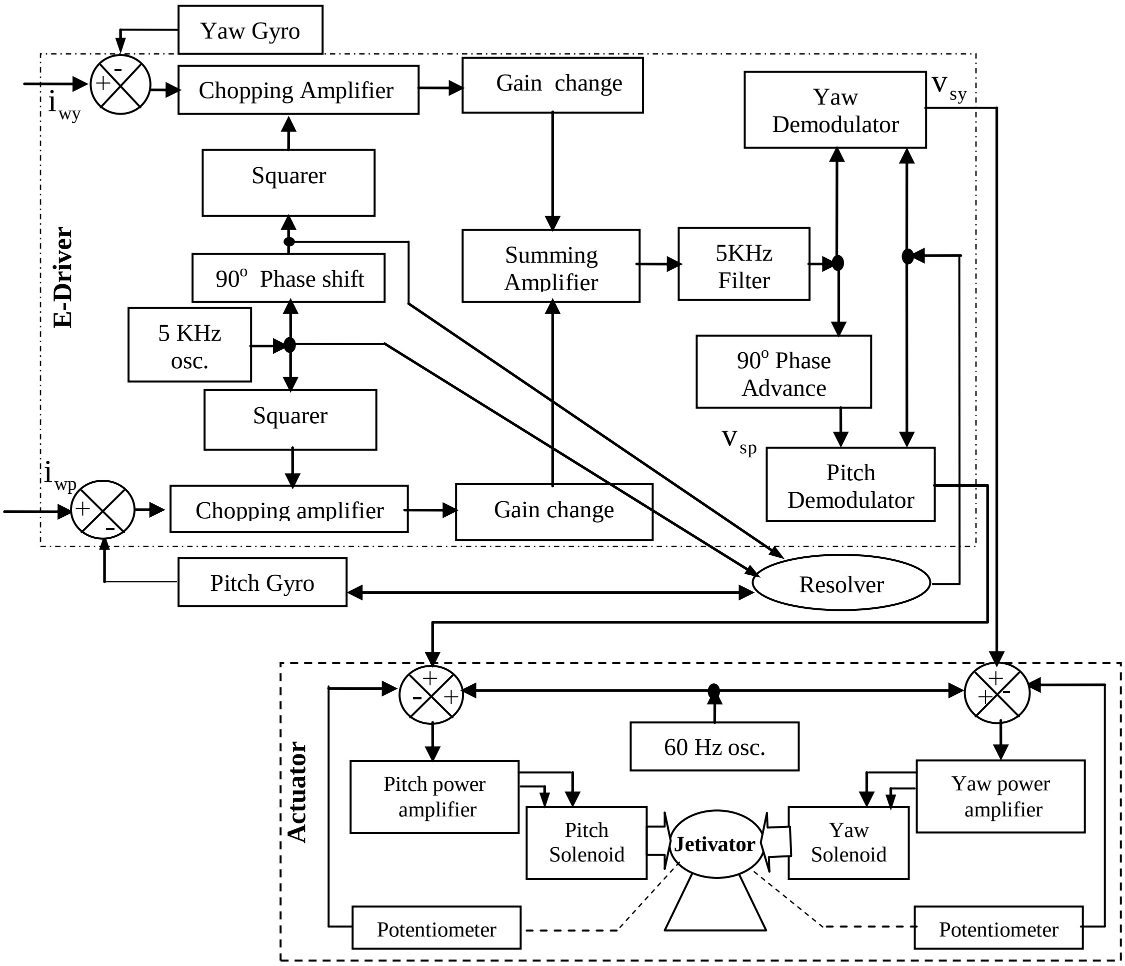


Fig. 14: E-Driver block diagram

4.1.1 Pitch Plane E-Driver

The systems identification experiment is conducted with the E-driver in closed loop performance and using different model structures including: ARX, ARMAX and state space model. Figure 15 illustrates the response via confidence interval which equals 3 standard deviations, where the filled gray region shows the confidence interval for this estimation. There is a clear indication that the impulse response takes off (leaves the uncertainty region) after (n) samples or a delay of (n) intervals. This figure reveals that the delay can be considered to be zero. Figures (16-18) illustrate the validation of outputs from the E-Driver identified models using state space, ARX, ARMAX with the HIL experiment for a target at distance X_t = 2800 [m], and LOS angels in both yaw and pitch planes is $\Psi_s = 0^\circ$ and $\theta_s = 0^\circ$, with sampling time $\Delta t = 0.05$[sec].

The identified transfer function of the missile E-Driver along its flight is of third order in the form $\frac{V_{sp}}{i_{ep}} = \frac{a_2\, s^2 + a_1 s + a_0}{s^3 + b_2\, s^2 + b_1 s + b_0}$. The obtained results illustrate that the ARMAX model is the most suitable model structure for the intended case study. Additional validation analysis was obtained as follows:

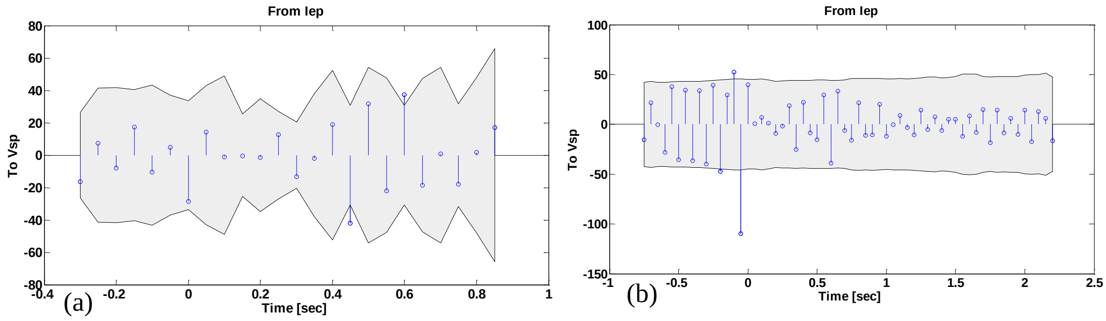


Fig. 15: Impulse response of E-Driver (a) 0-5.2 [sec] (b) 5.2-18 [sec]

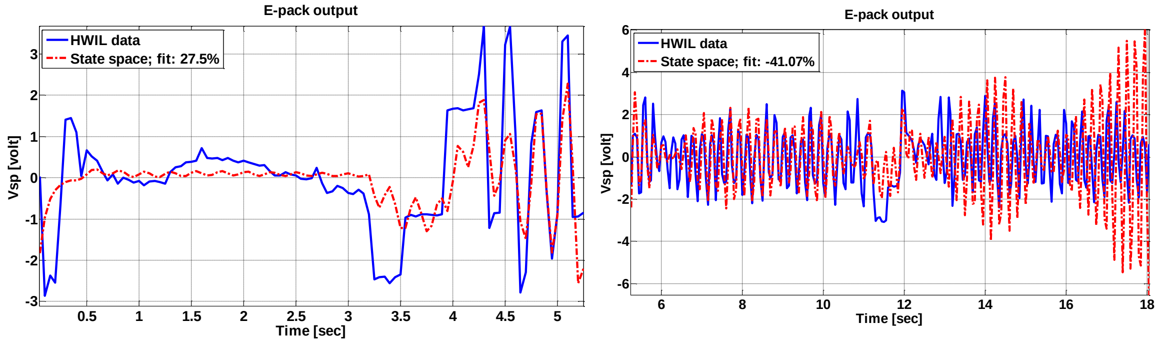


Fig. 16: Comparison between HIL data and state-space model

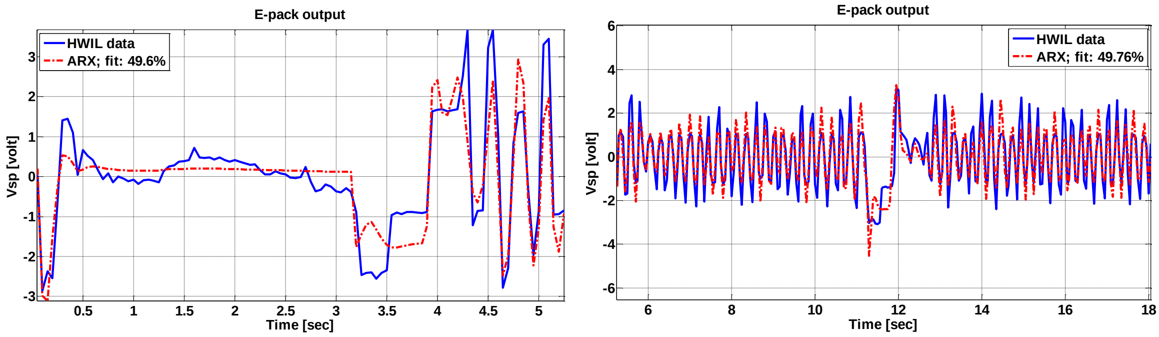


Fig. 17: Comparison between HIL data and ARX model

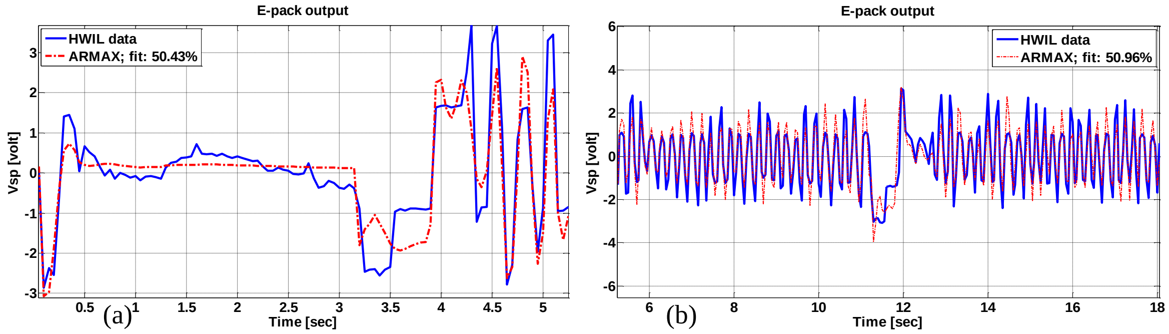


Fig. 18: HIL data and ARMAX model for E-Driver at: (a) t = 0 : 5.2 [sec], (b) t = 5.2 : 18 [sec]

- **Pole-Zero configuration**

The pole-zero configurations for the validated models was obtained by including confidence regions for the poles and zeros corresponding to 3 standard deviations, in order to determine how accurately they are estimated and also how close the poles and zeros are to each other. The pole-zero configuration of fifth, fourth and third order ARMAX model for 0-5.2 [sec] is shown in Fig. 19. This Figure illustrates that the obtained third order model have no overlap between the two complex-conjugate poles and zeros, which indicate that the obtained models have a suitable order and there is no pole-zero cancellation.

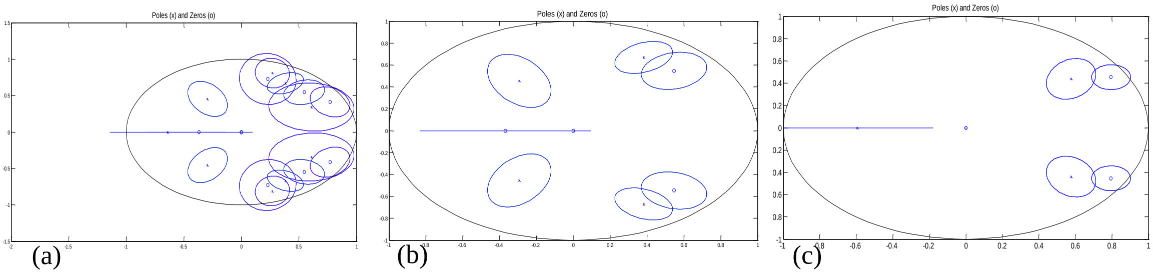


Fig. 19: Pole-zero configuration of ARMAX model at 0-5.2 sec (a) fifth (b) fourth (c) third order

- **Noise spectrum**

The noise spectrum of the obtained models is shown in Fig. 20, which shows that the ARMAX model has a best fit for most of the obtained models and has a lowest noise effect on model output.

- **Residual analysis**

A further way to gain insight into the quality of a model is to compute the residuals (e) in the output equation y = Gu + He, that could not be explained by the model. Ideally, these should be uncorrelated with the input and also mutually uncorrelated. The residual analysis for both state space and ARMAX models in both time and frequency domains at 5.2-18 [sec] are shown in Fig. (21, 22).

- **Residuals for the output error model in the time domain**

The cross correlation between residuals and input lies in the gray confidence region, indicating that there is no significant correlation and consequently the estimated model can be considered as adequate. However, the autocorrelation of (e) is significant, so it cannot be seen as white noise, i.e. the noise model (H) is not adequate.

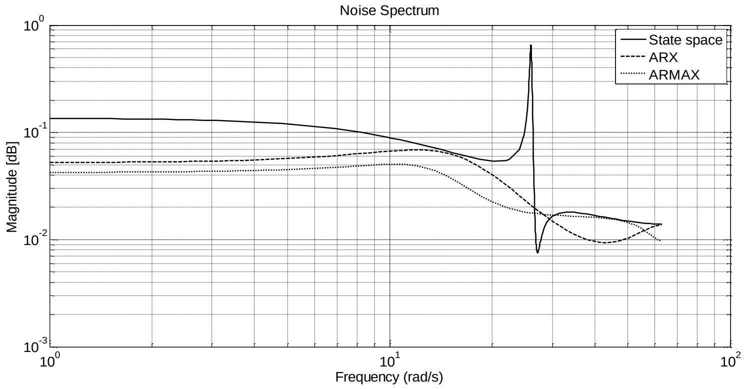


Fig.20: Noise spectrum of the obtained models

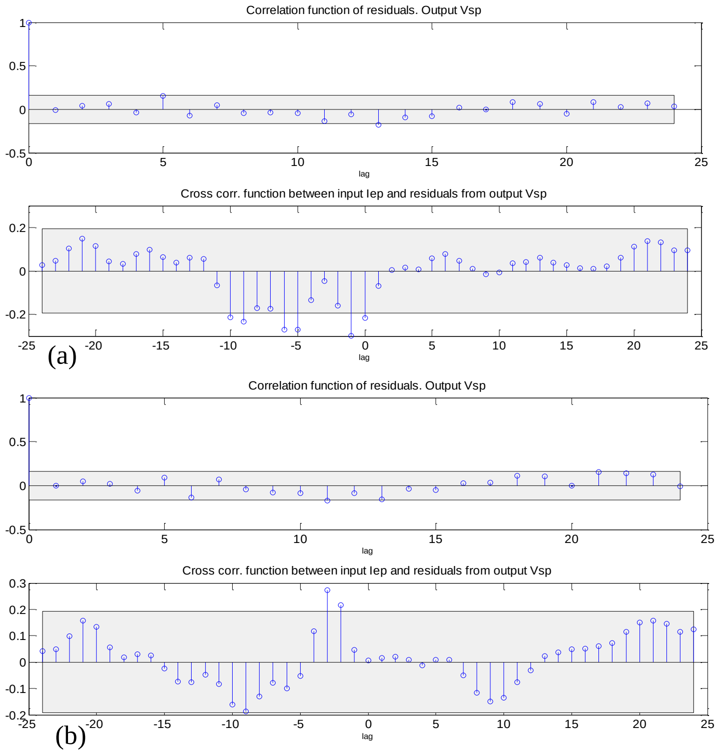


Fig. 21: Residual analysis (time domain) (a) state space (b) ARMAX

- **Residuals in the frequency domain**

The dotted curve is an estimate of the frequency function from u to e, where it lies inside the gray confidence interval and means that the obtained model is satisfactory at all frequencies.

Now, it is clear that the residual analysis is consistent with fitting results and proves that the ARMAX models have a best fit and residual analysis for most of the obtained models.

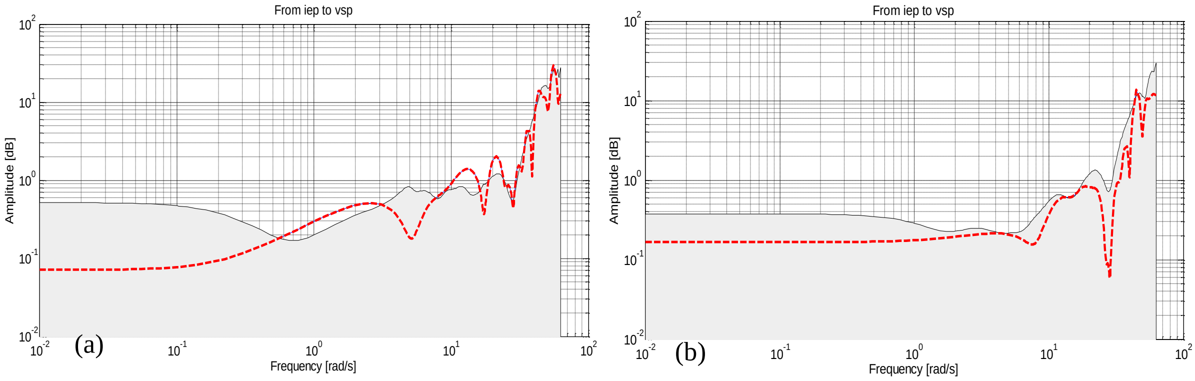


Fig. 22: Residual analysis (frequency domain) (a) state space (b) ARMAX

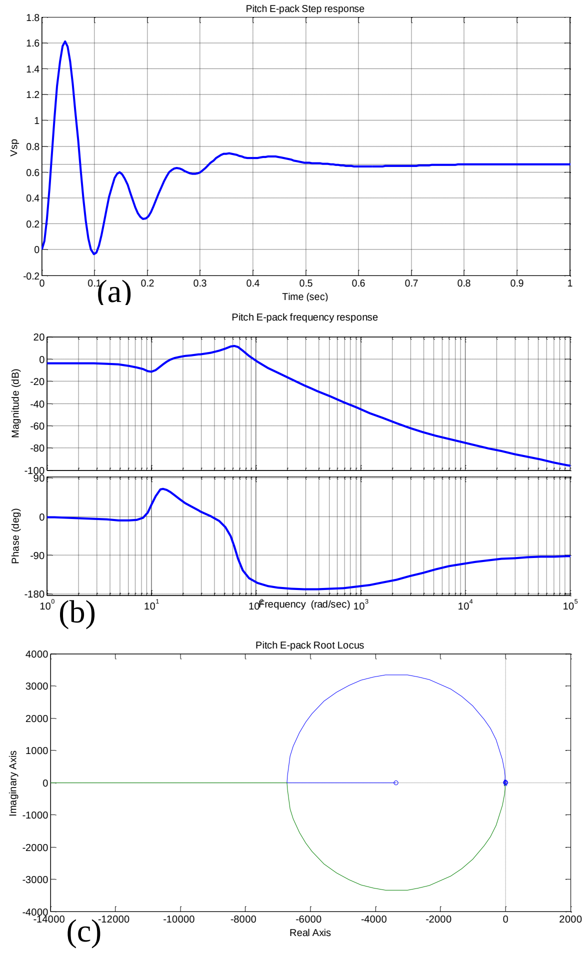
The step response, bode diagram, and the root locus for the E-Driver ARMAX pitch plane model at booster and sustainer periods are shown in Fig. (23, 24), respectively.

4.1.2 Yaw Plane E-Driver

The system identification experiment is conducted with E-driver in closed loop performance using different model structures including: ARX, ARMAX and state space model. Figures (25-27) illustrate the validation of outputs from the E-Driver identified models using state space, ARX, ARMAX with the HIL experiment for a target at distance X_t = 2800 [m], LOS angeles in both yaw and pitch planes is $\Psi_s = 0^\circ$ and $\theta_s = 0^\circ$, with sampling time $\Delta t = 0.001$[sec]. The identified transfer function of the missile E-Driver along its flight is of third order in the form $\frac{V_{sy}}{i_{ey}} = \frac{a_2\, s^2 + a_1 s + a_0}{s^3 + b_2\, s^2 + b_1 s + b_0}$. The figures (25-27) illustrate that all identified models are nearly have the same fitting value of about 52.5%. This experiment is conducted with $\Psi_s = 0^\circ$ and consequently the obtained yaw dynamics are returned back to the coupling between the control channels (pitch and yaw).

4.2 Actuator Assembly

The jetavator servo mechanism is an electro-pneumatic servo valve and plays the role of control within the missile that necessitates fast speed response, high power output and working fidelity. It has the function of moving the thrust nozzle according to the guidance command profile. The construction of the jetivator with the detailed main components is shown in Fig. (28). When one solenoid is energised, its ball is held against the pressure of the gas supply to close its inlet and opens one side of the piston to atmosphere. At the same time, the remaining solenoid is de-energised allowing the gas to pass through the other side of the piston. The solenoid valves are energised through drive amplifiers in the electronic driver. Under no-demand conditions each pair of ball valves is operated alternately at a frequency of 60 Hz with a 1:1 duty factor. This arrangement overcomes static friction and ensures rapid response (dithering). When a demand is received the duty factor is increased on one valve and decreased on the other, causing a greater mean pressure on one side of the piston that moves it with the jetivator nozzle in the required direction. The jetivator transfer function is estimated using the identification program and the measured data from the implementation of the hardware control section inside the simulation program. In this experiment, the E-Driver output voltage is the input while the nozzle deflection is the output.



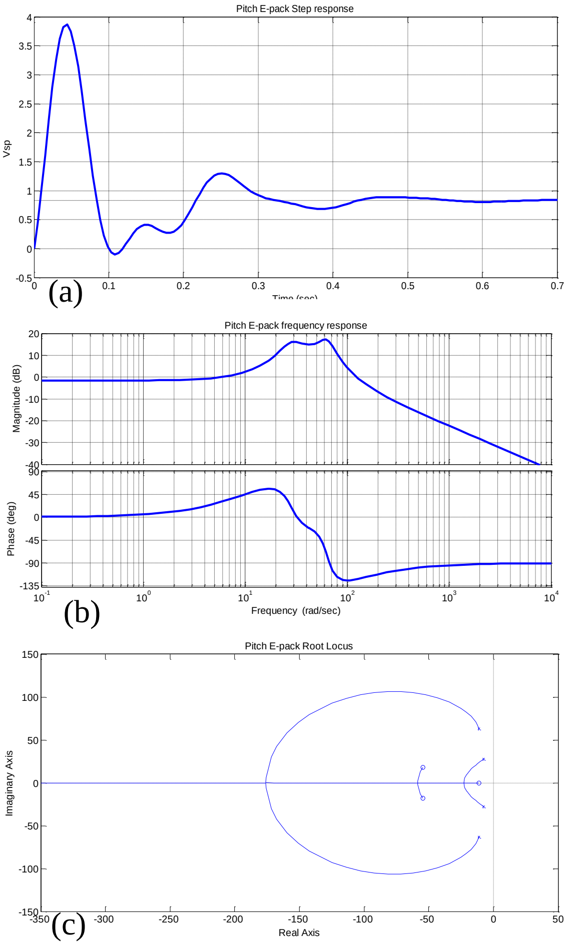
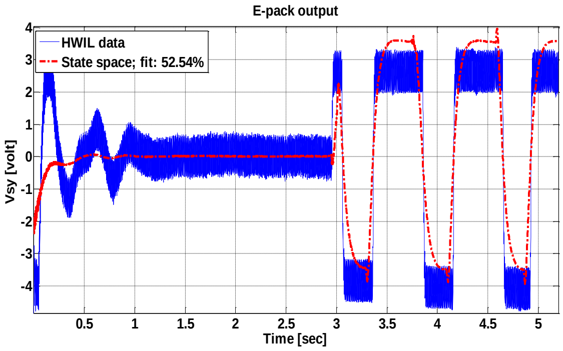


Fig. 23: ARMAX E-Driver in booster period

Fig. 24: ARMAX E-Driver in sustainer



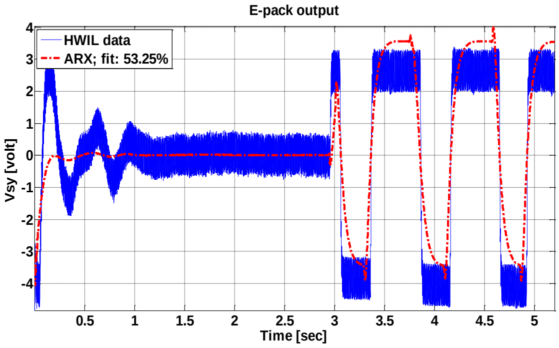


Fig. 25: HIL data and state-space model for Yaw E-Driver

Fig. 26: HIL data and ARX model for Yaw E-Driver

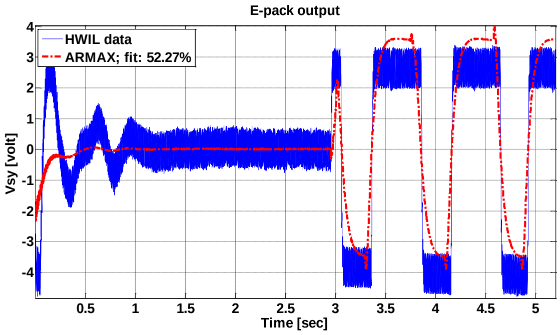


Fig. 27: HIL data and ARMAX model for Yaw E-Driver

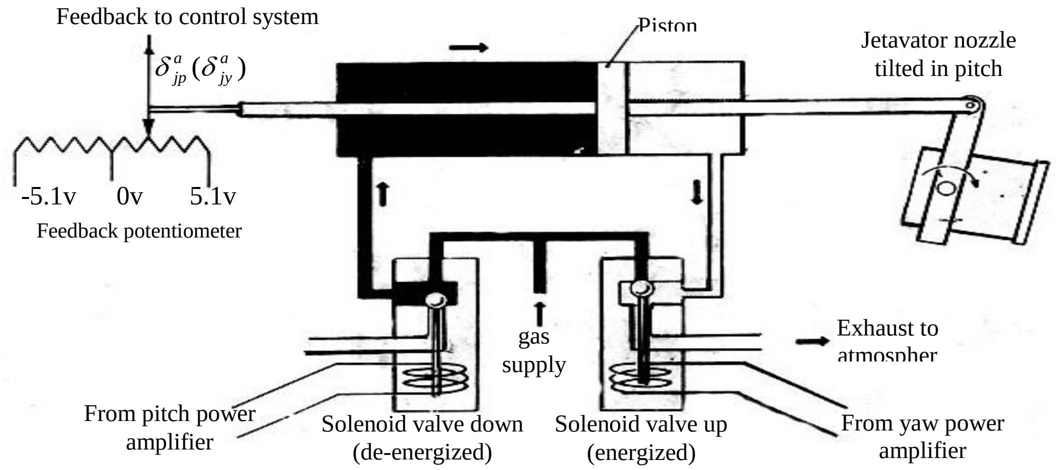

Feedback to control system
$\delta^a_{jp}\,(\delta^a_{jy})$
-5.1v
0v
5.1v
Feedback potentiometer
Piston
Jetavator nozzle tilted in pitch
gas supply
Exhaust to atmospher
From pitch power amplifier
Solenoid valve down (de-energized)
Solenoid valve up (energized)
From yaw power amplifier

Fig. 28: Jetivator Schematic

The transfer functions parameters of the jetivator are identified according to the input-output data at specified flight conditions. Figure 29 illustrates the outputs from the actuator identified model (state-space, ARX, and ARMAX) with the HIL experiment for a target at distance X_t = 2800 [m], LOS angels in yaw and pitch planes $\Psi_s = 0^\circ$ and $\theta_s = 0^\circ$, with sampling time $\Delta t = 0.05$[sec]. The identified transfer function of the missile pitch plane actuator along its flight is of 2nd order in the form $\frac{\delta_{jp}}{Vsp} = \frac{a_1 s + a_0}{s^2 + b_1 s + b_0}$. The step response and bode diagram for the jetivator ARMAX identified pitch plane model is shown in Fig. 30.

Using the same data above except that the sampling time is $\Delta t = 0.001$[sec], the identified transfer function of the missile yaw plane actuator is of 2nd order and has the form $\frac{\delta_{jy}}{Vsy} = \frac{a_1 s + a_0}{s^2 + b_1 s + b_0}$.

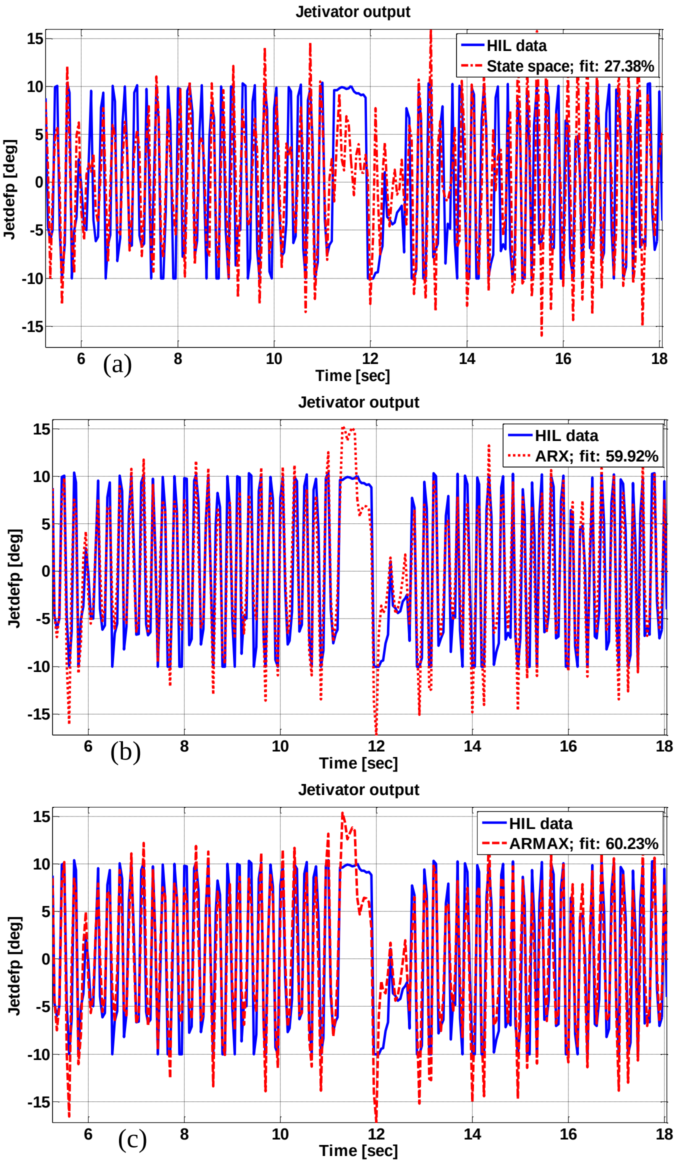


Fig. 29: Jetivator output using HIL data and identified model (a) state-space (b) ARX (c) ARMAX

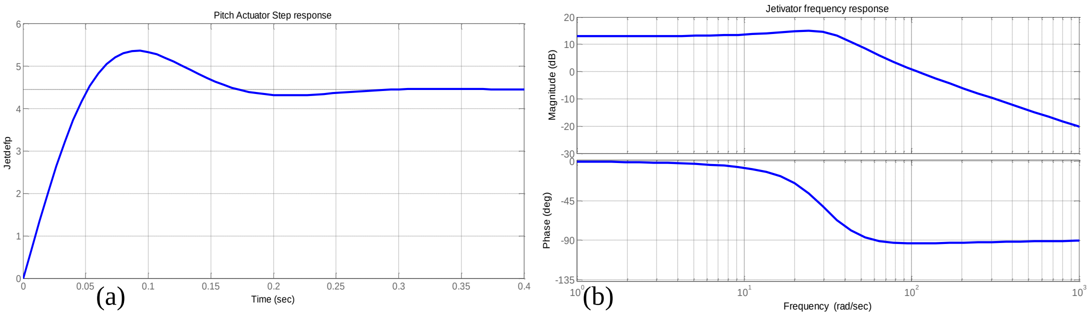


Fig. (30): ARMAX Jetivator model (a) Step response (b) Bode diagram

4.3 Missile Airframe

Accurate determination of the missile airframe response as an element within the guidance loop constitutes a corner stone in the design of a reliable and robust guidance system. In addition, it plays an important role in the execution of the guidance commands produced by the guidance computer. Thus, from the control point of view, the input to the missile airframe is the jet deflection (pitch/yaw -δ_{jp}/δ_{jy}) for thrust vector control and its output is the missile body angles (elevation/azimuth -θ/ψ). Since the missile system is time varying, there are more than one transfer function which describe the missile airframe along the flight envelope. The simulation program runs using different engagement scenarios for both the HIL and autopilot simulation model. The pitch/yaw jet deflection and corresponding body angles are recorded for a target at distance X_t = 2800 [m], LOS angles in both yaw and pitch planes $\Psi_s = 0^\circ$ and $\theta_s = 0^\circ$.

The transfer functions parameters of the airframe are identified according to the input-output data at specified flight conditions given above with sampling time $\Delta t = 0.05$[sec]. The response of pitch plane airframe identified models (state space, ARX, ARMAX) with the HIL experiment are shown in Fig. 31. The identified transfer function is of 3rd order and has the form $\dfrac{\theta}{\delta_{jp}} = \dfrac{a_2\, s^2 + a_1 s + a_0}{s^3 + b_2\, s^2 + b_1 s + b_0}$, the step response and bode diagram for which are shown in Fig. (32, 33) using ARMAX and state space at t=5:18 [sec]. In addition, using these data with sampling time $\Delta t = 0.001$[sec], the identified transfer function of the missile yaw plane airframe along its flight is of 3rd order and has the form $\dfrac{\psi}{\delta_{jy}} = \dfrac{a_2\, s^2 + a_1 s + a_0}{s^3 + b_2\, s^2 + b_1 s + b_0}$.

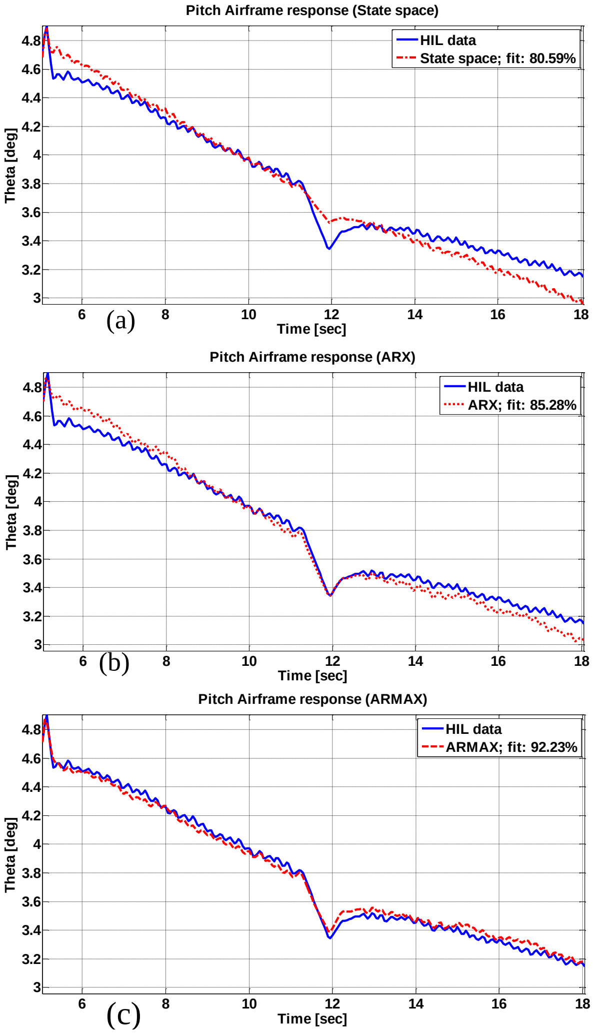


Fig. 31: Body angles using HIL data and airframe identified model

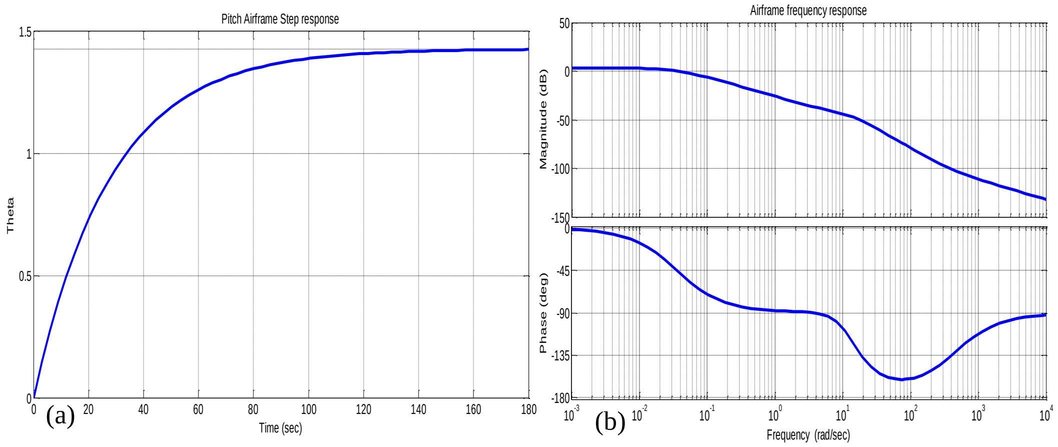


Fig. 32: ARMAX airframe model 5:18 [sec]

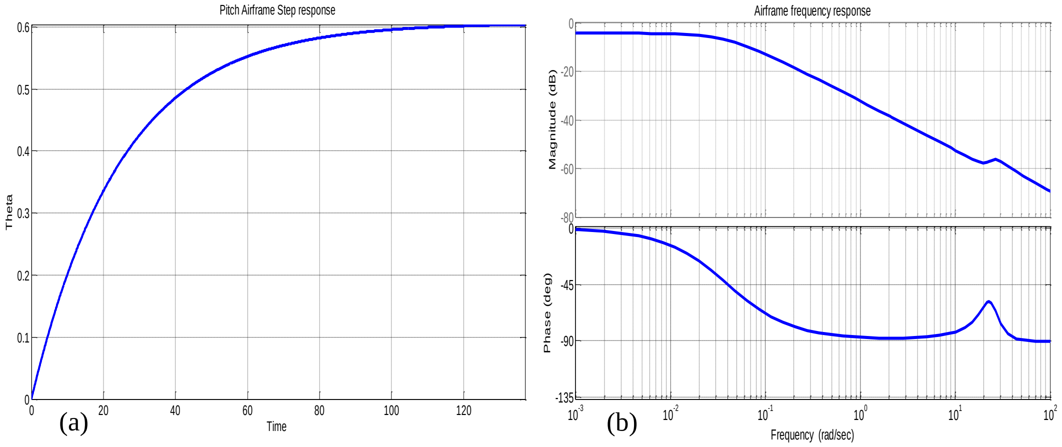


Fig. 33: State space airframe model 5:18 [sec]

5. Flight Path Performance

The effect of the identified jetivator transfer function on the flight path trajectory for conventional and robust controllers are shown in Fig. 34, which illustrates that a successful flight path trajectory is obtained via designed robust controller with the identified jetivator transfer function. The autopilot disassembled hardware section consists of actuator and jetivator assembly, Electronic Driver and resolver locked at normal position as illustrated earlier in section (2.1). In order to investigate the performance of the designed robust autopilot explained in Part-2 of the paper [12], the hardware implementation of the control section within the simulation program is shown in Fig. 35. The evaluation of the conventional autopilot and designed sub-optimal robust autopilot against different uncertainties in HIL environment is illustrated in Fig. 36.

The obtained results with HIL experiments reveal the following observations: (1) at nominal flight and low uncertainties level the flight path trajectory shows stable and successful engagements for both classical and robust autopilots. In addition, the control effort obtained with classical controller at this low level of uncertainties is lower than that obtained with robust controller. (2) Increasing the margin of uncertainties (lower thrust values – change of aerodynamic coefficients - wind speed) the classical autopilot has unstable flight path trajectory and ground impact, while the robust autopilot yields successful engagements. The control effort obtained using robust autopilots has the same level as that obtained at low uncertainties level, while the classical controller has a higher control effort. (3) The miss-distance obtained via robust autopilot especially at low tactical target range is lower than that obtained via classical autopilot and within an accepted margin.

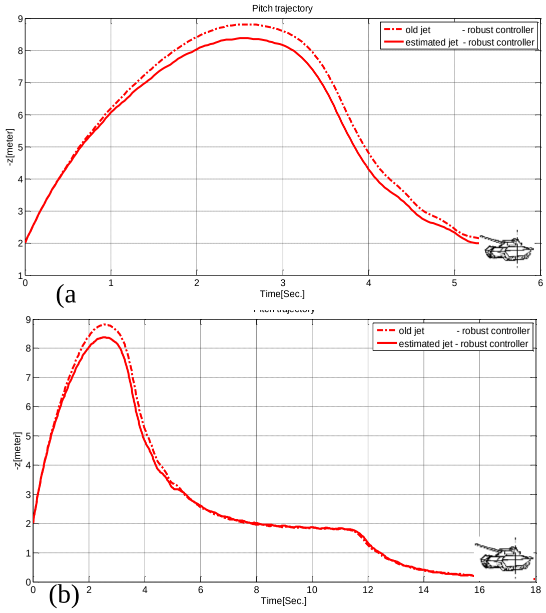


Fig. 34: Pitch trajectory for old and identified jetivator (a) 500 [m] (b) 2800 [m]

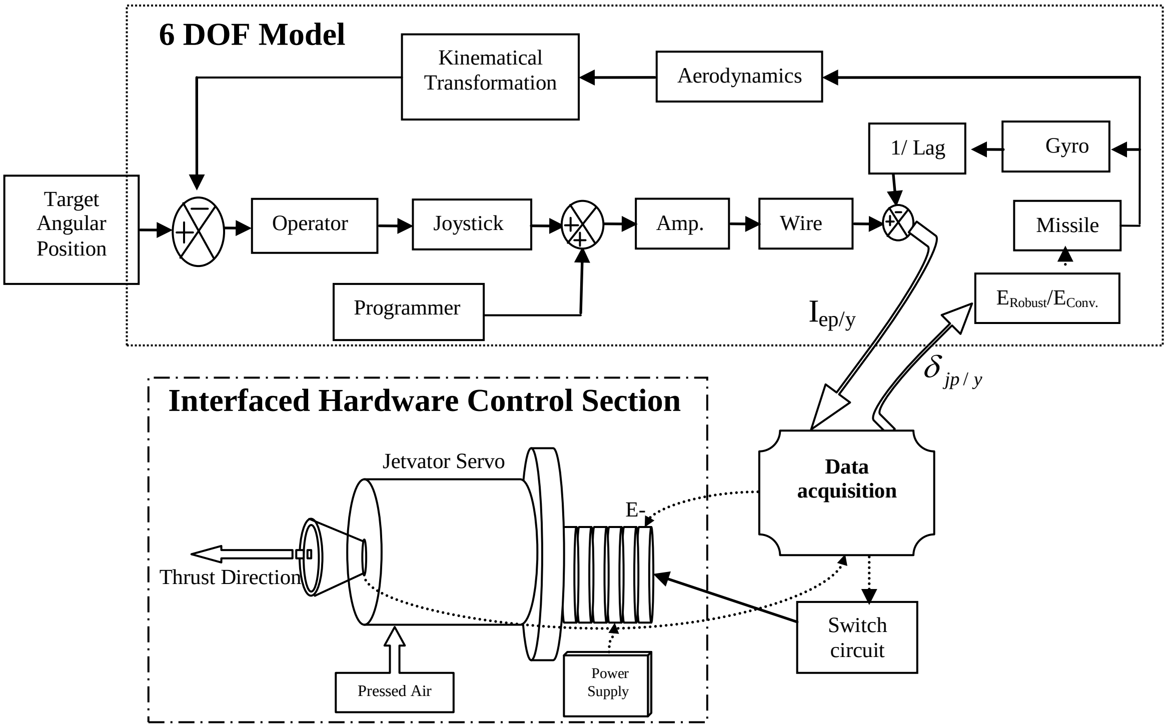
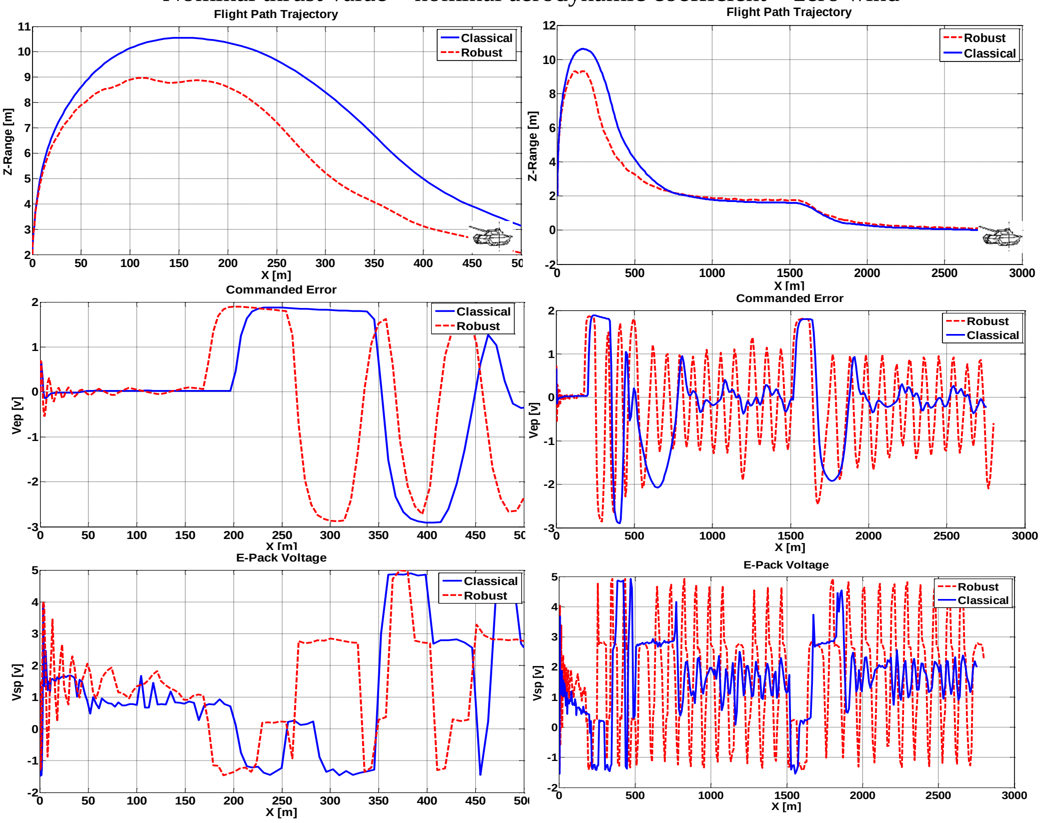


Fig. 35: Hardware in the loop simulation (Robust autopilot)

- Nominal thrust value – nominal aerodynamic coefficient – zero wind



- 85% thrust value – nominal aerodynamic coefficient – zero wind

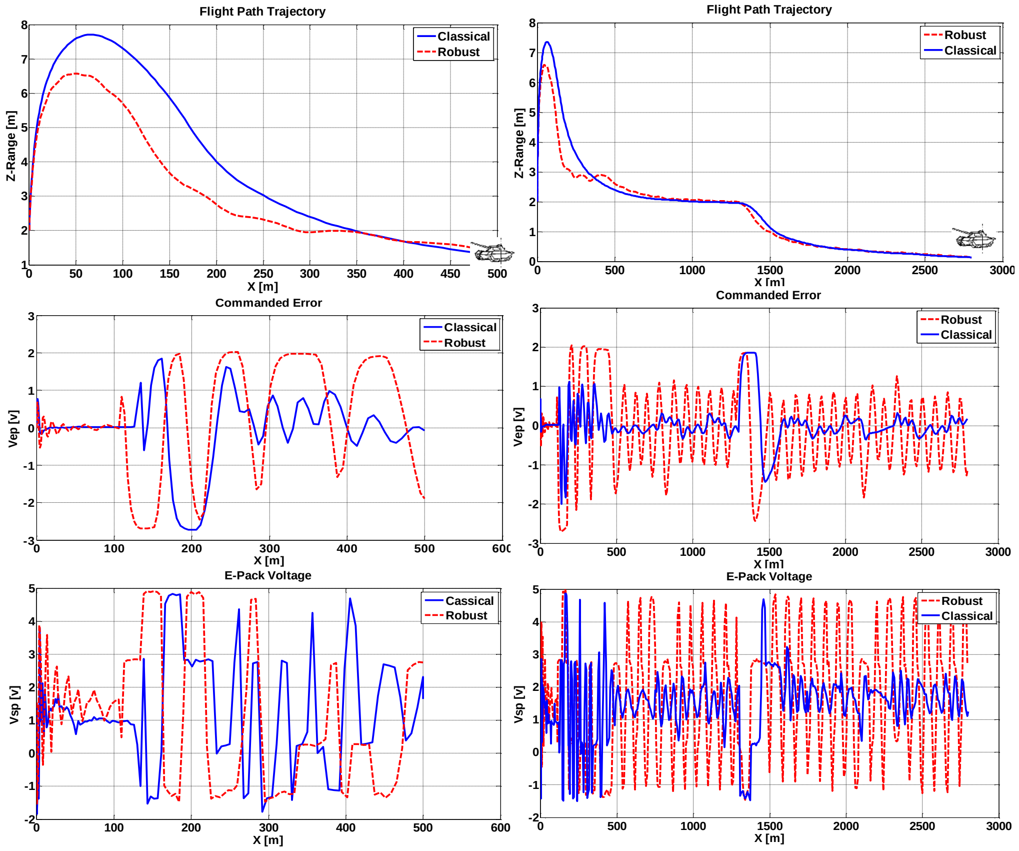
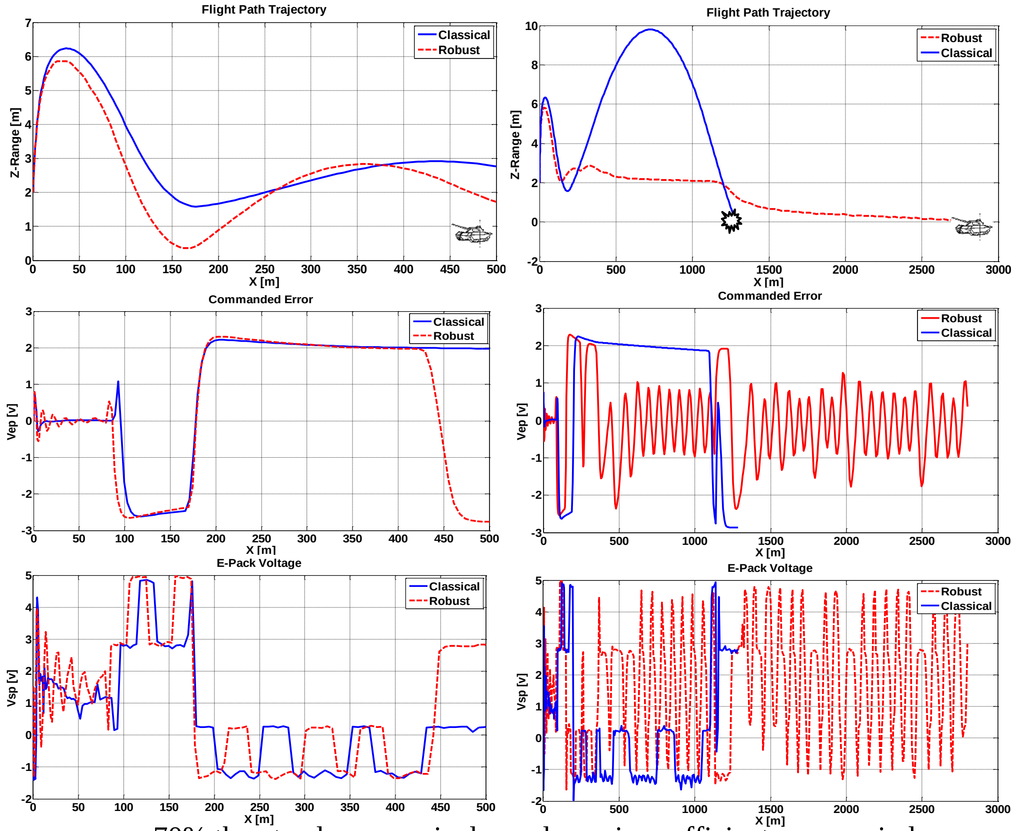


Fig. 36: HIL robust and classical Autopilot (a) Pitch trajectory (b) commanded error (c) E-Driver voltage

- 75% thrust value – nominal aerodynamic coefficient – zero wind



- 70% thrust value – nominal aerodynamic coefficient – zero wind

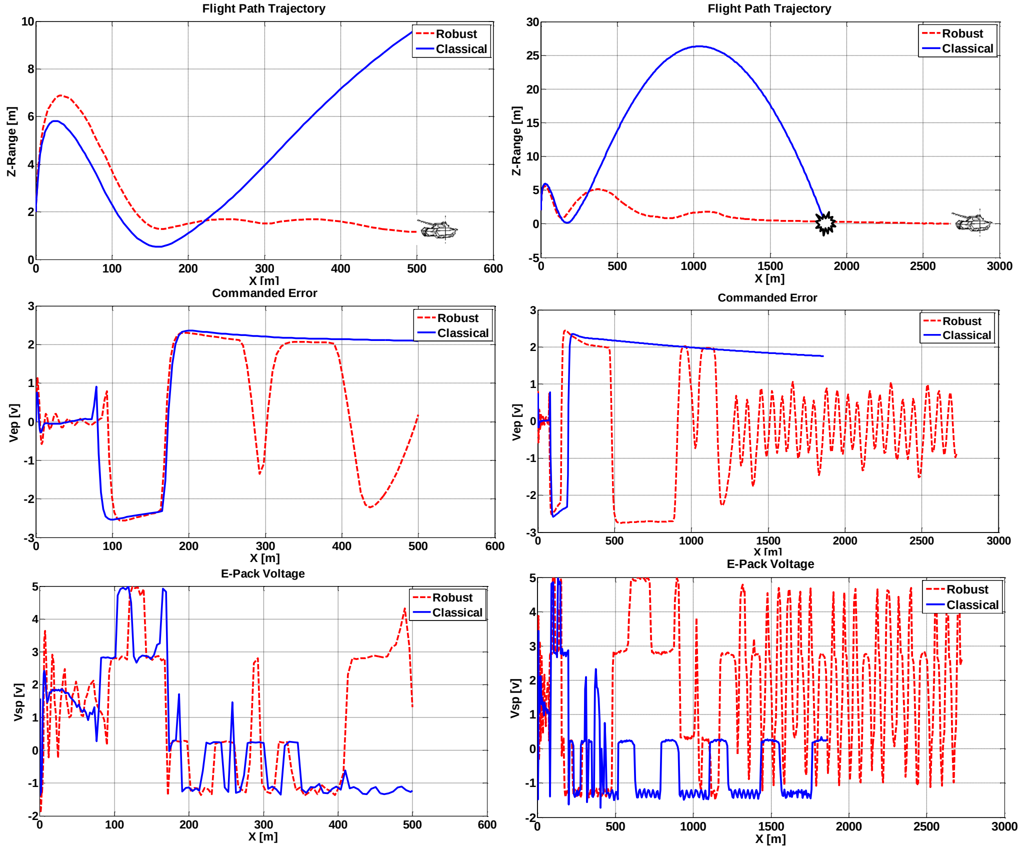
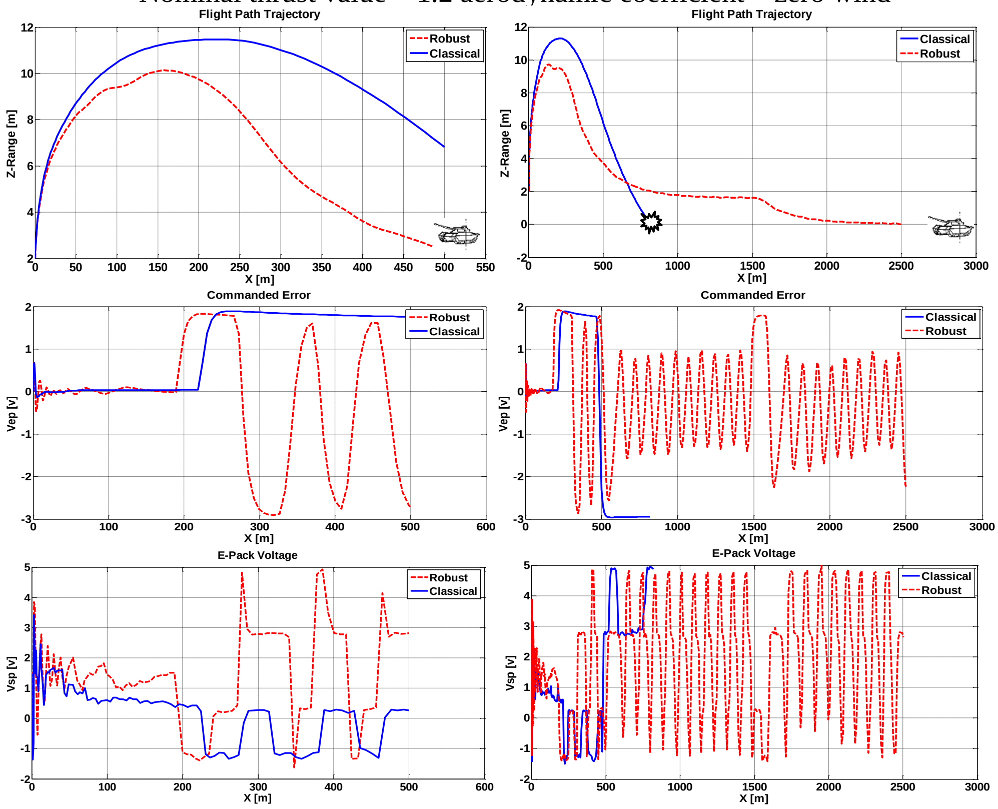


Fig. 36 (continued): HIL robust and classical Autopilot (a) Pitch trajectory (b) commanded error (c) E-Driver voltage

- Nominal thrust value – 1.2 aerodynamic coefficient – zero wind



- Nominal thrust value – nominal aerodynamic coefficient – V_{wy} = -30

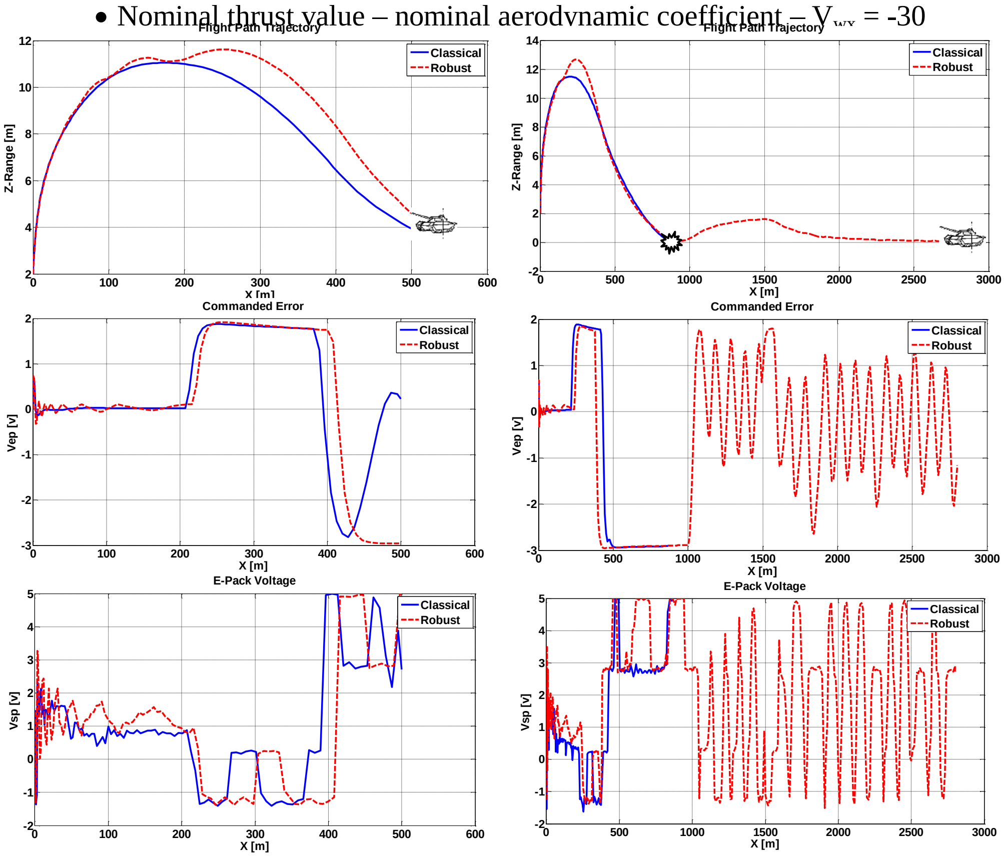


Fig. 36 (continued): HIL robust and classical Autopilot (a) Pitch trajectory Trajectory (b) commanded error (c) E-Driver voltage

6. Conclusions

This paper presented the systems identification techniques devoted to obtain good modeling for the system to enhance its performance via the HILS. The HIL experiment is described in detail complemented with systems identification theory and algorithms. Then, the system performance is evaluated using different types of models' structures and at different flight phases. The results reveal the robustness of identified models via fewer excursions in the flight path which leads to less flight time, possible increased range and less possibility to ground hit. This work has future objectives among them are the rules for selecting the weighting functions necessary for the H_∞ design, the friction in the servo of the utilized hardware (very old), adaptive robust flight control and the coupling between the two channels; pitch and yaw.

7. References

[1] Astrom, K.J., and P.Eykhoff, *System Identification-A survey*, Automatica,Vol. 7, pp.123-162,1971.

[2] Becerra, V.M., *Advanced System Identification*.

[3] Dolye J., Bruce Francis, Allen Tannenbaum, *Feedback Control Theory*, Macmillan Publishing Co., 1990.

[4] Gustvsson I., L.Ljung and T.Soderstrom, *Identification of Process in Closed Loop*, Automatica,Vol.13,pp.59-75,1977.

[5] Ljung, L., *System Identification – Theory for the User*, Prentice Hall, Upper Saddle River, N.J., 2nd edition, 1999.

[6] Ljung L., and Svante gunnarsson , *Adaptation and Tracking in System Identification* , Automatica,Vol.26,pp.7-21,1990.

[7] Maintenance and Repair Documents of the Swingfire Missile.

[8] Matlab toolboxes, *System Identification Toolbox*, 1988 – 2005 by The Math Works, Inc.

[9] Norton J.P., *An Introduction to Identification*, 1986.

[10] Robert L. Kosut, Ming K. Lau, Stephen P. Boyd, *System Identification for Robust Control Design* , European control conference,France,July 1991.

[11] Oda, A.N., G.A. El-Sheikh, Y.Z. El-Halwagy and M. Al-Ashry *Robust CLOS Guidance and Control Part-1: System Modeling and Uncertainty Evaluation,*14th International Conference on Aerospace Sciences & Aviation Technology,2010.

[12] Oda, A.N., G.A. El-Sheikh, M. Al-Ashry and Y.Z. El-Halwagy *Robust CLOS Guidance and Control Part-2: Scalar H∞ Autopilot Synthesis,*14th International Conference on Aerospace Sciences & Aviation Technology,2010.

[13] Soderstrom T., and Petre Stoica, *System Identification*, Prentice Hall,1989.

[14] Soderstrom T., I. Gustavsson, and L. Ljung , *Identifiability Conditions for Linear Systems Operating in Closed Loop*, Int. J. Control,1975,vol.21,No. 2.pp 243-255.

[15] Unbehauen, H., and G.P. Rao , *Continous-Time Approaches to SystemIdentification*, Automatica,Vol.26,pp.23-35,1990.