Available online at www.sciencedirect.com

ScienceDirect

IFAC Papers Online CONFERENCE PAPER ARCHIVE

IFAC PapersOnLine 55-37 (2022) 530–535

Physics-Informed Disturbance Estimation and Nonlinear Controller Design for a Multi-Axis Gimbal System

Damla Leblebicioglu*, Ozgur Atesoglu*,** Melih Cakmakci****

**ROKETSAN Missiles Inc., 06780, Ankara, Turkey*
(Tel: +90-312-860-5500; e-mail: {damla.leblebicioglu, ozgur.atesoglu}@roketsan.com.tr)

***Mechanical Engineering Department, Bilkent University, 06800, Ankara, Turkey*
(Tel: +90-312-290-4000; e-mail:{d.leblebicioglu, oatesoglu, melihc}@bilkent.edu.tr)

Abstract: The increasing demand for target tracking, environmental surveys, surveillance and mapping requires multi-axis gimbal systems with high tracking and stabilization performance. In this study, a new torque estimation structure is proposed to compute the complex disturbances negatively affecting the performance of the system. Two different control strategies based on Active Disturbance Rejection Control (ADRC) and *Estimated Torque Model* are implemented on a two-axis gimbal system. In the first strategy, the purpose is to improve the robustness, environmental adaptability and tracking accuracy of the system. The tuning effort of the ADRC is reduced by integrating a Neural Network (NN) based disturbance compensator (NN assisted ADRC). In the second strategy, NN is replaced with an *Estimated Torque Model* (ETM assisted ADRC), whose inputs come from plant outputs. The simulation results show that, both NN and ETM assisted control structures decrease the tracking errors. However, the improvements achieved by the physics-informed neural network-based estimator are more significant.

Keywords: *missile guidance and control, modeling, physics-informed neural network applications*

1. INTRODUCTION

Multi-axis gimbal systems are frequently used as mounts for day and infrared cameras (Ekstrand (2001); Hilkert (2008)). A multi-axis gimbal is a complex, multiple-input and multiple-output (MIMO), nonlinear system that is susceptible to effects of unknown frictional torques, external/internal disturbances and modeling uncertainties.

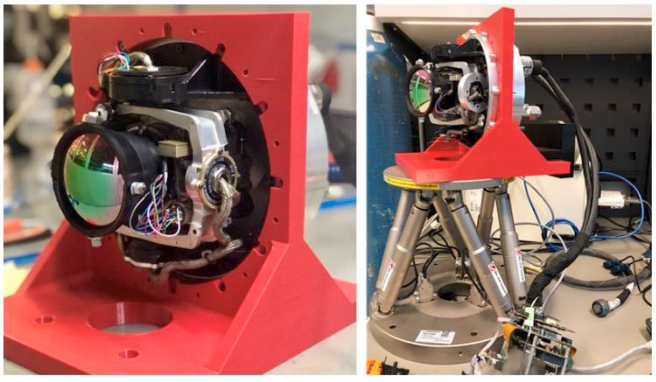

Figure 1. Two-axis gimbal system.

Various control strategies (Abdo et al. (2015); Baskin and Leblebicioglu (2017); Mao et. al (2019); Xiang and Mu (2021)) have been developed to satisfy the accurate positioning requirements during stabilization and tracking related missions. Among these, a popular and a powerful technique is ADRC, which first estimates disturbances using ESO (Extended State Observer). It is proposed by Han (2009). ADRC is a controller with sufficient adaptability that improves dynamic performance of the system. It is a method proven to be a suitable alternative for the conventional Proportional-Integral-Derivative (PID) controller. PID controllers are widely applicable to various systems due to their straightforward structure and comparably effortless design. Nonetheless, they are insufficient for compensation when nonlinear disturbances and parameter uncertainties are significant. ADRC is generally implemented to eliminate the limitations of a simple PID controller. Moreover, this method does not need any sophisticated information of the model (Bai and Zhang (2018); Zhou and Gao et al. (2018); Wang et al. (2019)). However, ADRC necessitates a certain parameter set to be adjusted for the best design, which is a challenging and a time-consuming task (Bai and Zhang (2018); Zhou and Gao et al. (2018)).

Traditional ADRC structure is quite sensitive to changes in model parameters, states of the system and inputs. Besides, controller parameters have a strong influence on the robustness and disturbance rejection capability of the system, (Bai and Zhang (2018); Liu and Zhao (2021); Qiao et al. (2020); Shen and Xu (2021)). In recent years, many researchers produced quite successful back-up solutions for these problems, especially by using NNs. For example, a classical state observer and a Radial Basis Function NN (RBFNN) is used to estimate the unknown total disturbance in Shen and Xu (2021). Similarly, an RBFNN is used in a dual channel composite and adaptive controller scheme without a state observer in Liu and Zhao (2021). Tuning problem is solved by using a NN which outputs ADRC parameters in response to changes in plant parameters and disturbances in Qiao et al. (2020).

In addition, a classical ADRC controller uses distinct ESOs for each channel. Instead, in the proposed control strategy, there

10.1016/j.ifacol.2022.11.237

is only one NN and it works as a MIMO disturbance compensator. Inputs of the NN are fed from both the azimuth and elevation axes. Outputs of the NN are supplementary accelerations for both channels. Due to authors' knowledge, the proposed controller and compensation scheme applied on a two-axis gimbal system is used for the first time. Furthermore, it is relatively simpler when compared to existing similar studies, (Liu and Zhao (2021); Shen and Xu (2021)). Moreover, another approach is also presented as an alternative to NN: necessary compensation can be provided by adding a second *ETM block* for the ADRC structure (*ETM* based ADRC controller) to replace the NN.

This study concentrates on disturbance torque estimation and proposes an alternative control methodology for a multi-axis gimbal system that will be mounted on an unmanned aerial vehicle (Leblebicioglu et al. (2021)). Our objective is to improve the position control loop performance of the gimbal system that is currently working with a conventional ADRC by adding a coupled, neural network based disturbance compensator. NN supports the ADRC by learning the acceleration required to compensate the loss arising from external/internal disturbances and parameter uncertainties of the system. As a result, gimbal follows the reference position commands ensuring higher precision on reference tracking and inner loop stabilization. First an accurate gimbal model is implemented in Simulink® and verified with the hardware set-up shown in Fig. 1. A standard ADRC structure in Han (2009) is implemented by using the *Estimated Torque Model (ETM)*. Afterwards, a NN whose inputs are plant position, velocity and acceleration and whose output is the delta acceleration commands (difference between ETM and real plant) to support the ADRC output has been embodied to ADRC. This new structure: (i) makes parameter tuning of ADRC much easier, (ii) improves tracking accuracy, (iii) increases disturbance rejection ability of the system. It works as a complimentary back-up when ADRC is not completely sufficient to meet the desired performance requirements. Also, NN provides enhanced adaptivity of the control system for time-varying reference inputs. Those inputs vary in broad range during the search, detect and track phases of the operation of the multi-axis gimbal system (iv). We consider the physical structure of the gimbal system in the ETM block and in the proposed training method of the NN.

The rest of the paper is organized as follows. Section 2 introduces the mathematical model of the gimbal dynamics and the *Estimated Torque Model* for a two-axis gimbal system. Section 3 presents the classical ADRC, NN assisted ADRC and *ETM* assisted ADRC designs for the multi-axis gimbal system model. Section 4 summarizes and compares simulation results for the suggested controllers. The final section (Section 5) lists conclusions and future work about the current implementation.

2. MATHEMATICAL MODEL OF THE SYSTEM

The mechanical structure of the two-axis gimbal system in Fig. 1 is composed of three interconnected rigid bodies: the inner ring (for the elevation motion), the outer ring (for the azimuth motion) and the base platform. The sighting camera and the micro-electromechanical system (MEMS) gyroscope are installed on the inner ring. Thus, they provide measurements corresponding to the inner gimbal reference frame F_m (with axes x_m, y_m, z_m). Outer gimbal rotates with angle ψ_a around the z_b-axis of the base platform reference frame F_b (with axes x_b, y_b, z_b). Inner gimbal rotates with angle θ_m around the y_a-axis of the outer gimbal reference frame F_a (with axes x_a, y_a, z_a).

The Euler angles of the base platform with respect to (wrt.) a chosen inertial reference frame are denoted by $\emptyset, \theta, \psi$. The angular velocity ($\bar{\omega}_{b/o}^{(b)}$) and translational acceleration ($\bar{\alpha}_{b/o}^{(b)}$) of the base platform wrt. a chosen inertial reference frame, expressed in the base platform reference frame are $\bar{\omega}_{b/o}^{(b)} = [p\ q\ r]^T$ and $\bar{a}_{b/o}^{(b)} = [a_x\ a_y\ a_z]^T$. The rotation sequence and the related angles are shown in Fig. 2. In this figure, $\bar{u}_1^{(i)}, \bar{u}_2^{(i)}, \bar{u}_3^{(i)}$ are the unit vectors along x_i, y_i, z_i directions for reference frames $i = b, a, m$.

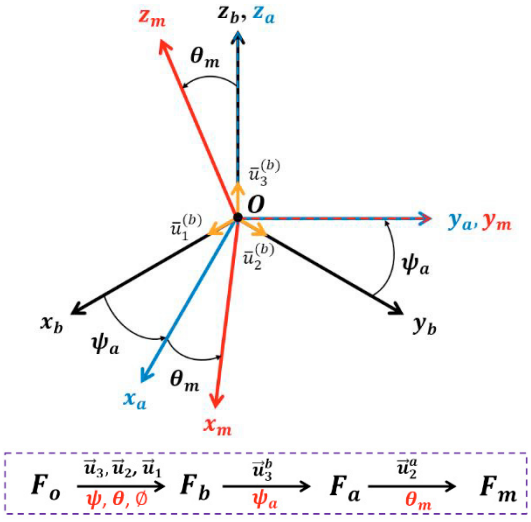

Figure 2. Two-axis gimbal system reference frames and rotation sequence definition.

The mathematical model used in this paper is first proposed by Atesoglu, et al. (2016). The dynamical equations for the multi-axis gimbal system in the matrix form are provided in Eqn. 1. Regarding modeling of two-axis gimbal systems, Eq.1 and more can be found in Leblebicioglu, et al. (2021). Given the driving torques, T_a and T_e, the relative angular accelerations of the azimuth and elevation gimbals, $\ddot{\psi}_a$, $\ddot{\theta}_m$, can be calculated.

Here, the proposed modeling technique, also ensure the calculation of the reaction forces ($\bar{F}_{a/m}^{(a)}$, $\bar{F}_{a/b}^{(a)}$) and the certain components of the reaction moments (M_{amx}, M_{amz}, M_{abx}, M_{aby}) acting on the gimbal revolute joints.

In addition, the commanded torques, T_{ac}, T_{ec}, can be calculated by inverting Eqn. 1, when the reference values of angular accelerations, $\ddot{\psi}_{ac}$, $\ddot{\theta}_{mc}$, angular velocities, $\dot{\psi}_{ac}$, $\dot{\theta}_{mc}$, and angular positions, ψ_{ac}, θ_{mc}, of the azimuth and elevation gimbals are available. This is shown in the block diagram given in Fig. 3, and named as the *Estimated Torque Model.*

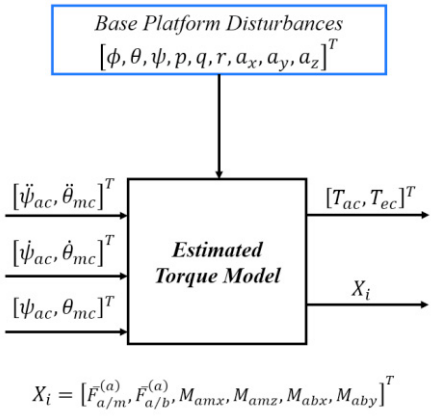

Figure 3. Block diagram of the *Estimated Torque Model*.

$$\hat{F}_{12x2}\begin{bmatrix}\ddot{\psi}_a\\ \ddot{\theta}_m\end{bmatrix}_{2x1} + \hat{R}_{12x10}\begin{bmatrix}\bar{F}^{(a)}_{a/m}\\ \bar{F}^{(a)}_{a/b}\\ M_{amx}\\ M_{amz}\\ M_{abx}\\ M_{aby}\end{bmatrix}_{10x1} = \hat{D}_{12x1} + \hat{G}_{12x2}\begin{bmatrix}T_a\\ T_e\end{bmatrix}_{2x1} \quad (1)$$

3. CONTROLLER DESIGN

3.1 Classical ADRC with ETM

The classical ADRC is composed of three parts: *Transient Profile Generator (TG), Weighted State Error Feedback (WSEF)* and *Total Disturbance Estimation and Rejection by* using *ESO.* This controller is seperately designed and constructed for distinct command channels. Equations for the azimuth and elevation channels are the same; the formulation for a single channel are given in this section.

Transient Profile Generator (TG):
This part introduces the implementation of a high-quality nonlinear derivative operation without noise. Here, v_1 is the desired trajectory, v_2 is its derivative, v is the reference input (desired position references, ψ_{ac} and θ_{mc}), r is the learning rate that speeds up or slows down the transient profile and it needs to be tuned for perfect derivative action.

$$\dot{v}_1 = v_2, \dot{v}_2 = -r sign(v_1 - v + \frac{v_2|v_2|}{2r}) \quad (2)$$

Extended State Observer (ESO):
Consider the state space respsentation of the nonlinear system expressed in Eqn. 3. An additional state variable x_3 is created to represent total disturbance, $F(t)$ in the system where $\dot{F}(t) = G(t)$, and $G(t)$ is unknown.

$$\dot{x}_1 = x_2$$
$$\dot{x}_2 = x_3 + bu, x_3 = f(x_1, x_2, w(t), t) = F(t)$$
$$y = x_1 \quad (3)$$

Note that, $\dot{x}_3 = G(t)$, according to definitions above. The state observer (ESO), is constructed with the set of equations given in Eqn. 4. z_1, z_2 and z_3 are the outputs of ESO. z_1 is the observed position, z_2 is the observed velocity and z_3 is the estimated disturbance. Explicit form of the nonlinear *fal(.)* function is given in Eqn. 5. It is used to eliminate the chattering problem highly possible to arise from the *sign()* function. The observer parameters needed to be tuned are $\beta_{01}, \beta_{02}, \beta_{03}, \alpha_1, \alpha_2, \delta$. Furthermore, a scaling parameter b_0, should also be tuned. In this study, the control input is u ($\ddot{\psi}_{ac}$,$\ddot{\theta}_{mc}$) and plant output is y (ψ_a,θ_m). Also, following Han (2009), $\alpha_1 = 0.5$ and $\alpha_2 = 0.25$ are used.

$$e = z_1 - y, fe = fal(e, \alpha_1, \delta), fe_1 = fal(e, \alpha_2, \delta)$$
$$\dot{z}_1 = z_2 - \beta_{01}e, \dot{z}_2 = z_3 + b_0 u - \beta_{02} fe, \dot{z}_3 = -\beta_{03} fe_1 \quad (4)$$

$$fal(e, \alpha, \delta) = \begin{cases} \frac{e}{\delta^{1-\alpha}}, & |e| \le \delta \\ |e|^{\alpha} sign(e), & |e| > \delta \end{cases} \quad (5)$$

Weighted State Error Feedback (WSEF):
Here, WSEF acts like a simple PD (Proportional-Derivative) type controller. Commands generated from Eqn. 2, (v_1 and v_2), are used to form the error signals (e_1 and e_2). Weighted combinations of error signals form the reference acceleration input (u_o) for the *ETM.* k_1 and k_2 are the parameters to be tuned. The block diagram in Fig. 4 shows the signal flow for $\bar{v}_1$, $\bar{v}_2$, $\bar{z}_1$ and $\bar{z}_2$, to form $\bar{e}_1$ and $\bar{e}_2$ (bar denotes concatenation of azimuth and elevation vectors). Note that $\bar{v}_1$ and $\bar{v}_2$ will be the position and velocity inputs for the *ETM.*

$$e_1 = v_1 - z_1, e_2 = v_2 - z_2$$
$$u_o = k1e_1 + k2e_2, u = (u_0 - z_3)/b_0 \quad (6)$$

ADRC with *ETM* structure is given in Fig. 4 (plant block is composed of *Gimbal Dynamics* and the DC motor model). u_{0a} and u_{0e} are the outputs of WSEF blocks from azimuth and elevation axes, respectively. The ETM block is named as *ideal* (described with $\bar{D} = \bar{0}$). There is no (i) friction torque in revolute joints, (ii) disturbance torque due to electrical cables of motors, encoders, gyroscope, camera and cooling pipes, (iii) disturbance torque due to off-diagonal inertia terms, rotation axes and CoG (center of gravity) offsets, (iv) base disturbance. However, the plant is *non- ideal*, which means, $\bar{D} \neq \bar{0}$.

3.2 NN Based ADRC Implementation

As mentioned previously, ADRC has certain parameters to be tuned. This paper proposes an intelligent controller, NN assisted ADRC structure which eases calibration and tuning of the legacy controller. It also unlocks the possible need for scheduling of the controller parameters which can be needed to cope with the time varying disturbances on the system and/or different reference signal levels. NN is employed to approximate the complex nature of possible disturbances which are present in the hardware set-up (Fig. 1) and enhance the tracking performance of the already constructed legacy ADRC structure. Addition of the proposed NN also speeds up the response and decreases tracking error on the position

control loop. The performance of the proposed controller is verified by simulations presented in Sec. 4.

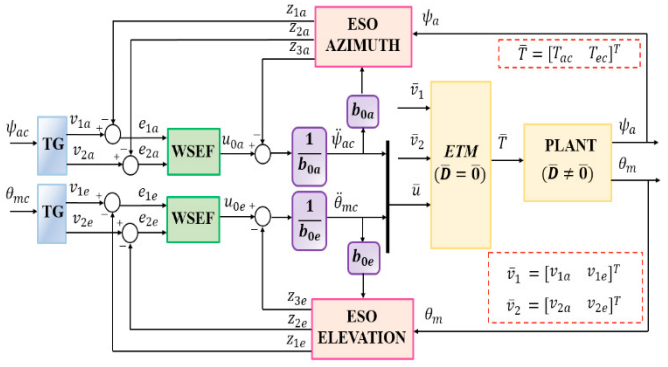

Figure 4. Block diagram of controller with TG, WSEF, ESO, ETM structure.

NN training is performed with the block diagram shown in Fig. 5. Note that, there are no disturbances in the ETM block used for training. The difference between $[\ddot{\psi}_{ac}\ \ddot{\theta}_{mc}\ ]^T$ and $[\ddot{\psi}_a\ \ddot{\theta}_m\ ]^T$ is accounted as the compensatory acceleration that needs to be fed on the control commands generated by the legacy controller. The training data set of the neural network is $\{(\bar{r}, \dot{\bar{r}}, \ddot{\bar{r}}),\ \Delta\ddot{\bar{r}}, t \in [0, t_f]\}$, where $\bar{r} = [\psi_a\ \theta_m\ ]^T$, $\dot{\bar{r}} = [\dot{\psi}_a\ \dot{\theta}_m\ ]^T$ and $\ddot{\bar{r}} = [\ddot{\psi}_a\ \ddot{\theta}_m\ ]^T$. Output of the NN is the compensatory acceleration, $\Delta\ddot{\bar{r}} = [\Delta\ddot{\psi}_a\ \Delta\ddot{\theta}_m\ ]^T$. The block diagram representation of the NN based ADRC controller is given in Fig. 6. $\Delta\ddot{\bar{r}}$ enables the plant to reach the reference position in a closed loop structure when it is added to $[\ddot{\psi}_{ac}\ \ddot{\theta}_{mc}\ ]^T$. With the implementation of NN, $u = \left(\frac{u_0 - z_3}{b_0}\right) + \Delta r$.

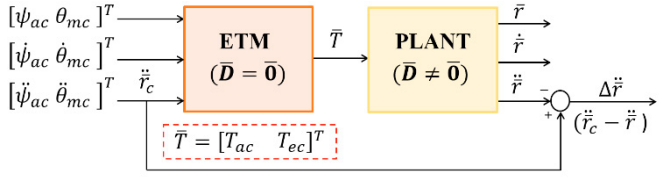

Figure 5. NN training structure.

3.3 ETM Based ADRC Implementation

As an alternative, instead of using a NN-supported-ADRC controller, complementary torque signals can also be generated from a second *ETM block* in the controller architecture. By using the output of the plant as the input for the second *ETM block*, the differential torque between the first *ETM* and the second *ETM* can be calculated and added to the system. With this approach, instead of training a NN with the data obtained from *ETM* and plant, the compensatory torque signals can be directly supplied from the secondary *ETM block.* Support will be in the form of delta torque instead of delta acceleration. The block diagram representation of the *ETM*-assisted-ADRC controller is given in Fig. 7. The total torque applied to the plant is sum of the output of the first *ETM block* ($\bar{T}$) and the difference between the outputs of the first and second *ETM blocks* ($\Delta\bar{T}$).

4. SIMULATIONS

In this section, we present the performance of the NN assisted ADRC formulation presented in Sec. 3. Experiments of the system with NN-ADRC, ETM-ADRC hybrid controllers and conventional legacy ADRC controller are carried out by using MATLAB® and Simulink® R2020b. Equations of gimbal dynamics and ETM are implemented as MATLAB® function blocks. The sampling period of the simulated system is chosen to be 0.001 seconds and the numerical integration method is ODE4 (Runge-Kutta). Properties of the simulated two-axis gimbal system, calculated from Catia v5-3DX® are given in Tabs. 1 and 2.

Table 1. Distances of the gimbal platform

Distances (in mm)	
$\bar{r}^{(a)}_{Ga/a} = [\boldsymbol{Ga_x\ Ga_y\ Ga_z}]^T$	$[0\ \ 0\ 57.5]^T$
$\bar{r}^{(m)}_{Gm/m} = [\boldsymbol{Gm_x\ Gm_y\ Gm_z}]^T$	$[0\ -44.5\ 0]^T$
$\bar{r}^{(a)}_{m/a} = [\boldsymbol{am_x\ am_y\ am_z}]^T$	$[0\ \ 44.5\ 57.5]^T$
$\bar{r}^{(b)}_{a/b} = [\boldsymbol{ba_x\ ba_y\ ba_z}]^T$	$[31.625\ \ 0\ -57.5]^T$

Table 2. Parameters of the gimbal platform

	Yaw Gimbal	*Pitch Gimbal*
$\boldsymbol{m\ (kg)}$	0.55	1.138
$\hat{\boldsymbol{J}}\ \boldsymbol{(kgm^2)}$	$J_{xx} = J_{zz} = 0.002$ $J_{yy} = 0.004$	$J_{xx} = J_{zz} = 0.004$ $J_{yy} = 0.003$
FOR limits	$\pm 45^{\circ}$	$\pm 20^{\circ}$

$\bar{r}^{(a)}_{Ga/a}, \bar{r}^{(m)}_{Gm/m}, \bar{r}^{(a)}_{m/a}, \bar{r}^{(b)}_{a/b}$ in Tab. 1, are the distance vectors between revolute joints and CoG points and revolute joints. Tab. 2 presents the mass, inertia and field-of-regard (FOR) limits of the azimuth and elevation gimbals (J_{ii} are the diagonal inertia terms along directions $i = x, y, z$). Off-diagonal inertia terms calculated from 3D model are of order 10^{-6} or 10^{-5} (they are approximately twice smaller than primary diagonal inertia terms). Although they are small enough, the off-diagonal terms are included in the gimbal models (dynamical mass unbalance is included).

The feedforward neural network designed for the simulations has 2 hidden layers, each layer is composed of 20 neurons. Layer outputs are calculated by the transfer function ***poslin (positive linear).*** The training method is chosen as ***Levenberg-Marquardt*** backpropagation. The disturbance torque in the system, used in simulations is given in Eqn. 7, $(x = a, m)$. The terms in the disturbance function are single and dual combinations of $\theta_m, \dot{\theta}_m, \psi_a, \dot{\psi}_a, \theta_m^2, \psi_a^2, \dot{\theta}_m^2, \dot{\psi}_a^2$. The state independent bias term ($k37_x$) is included as well. Coefficients are determined by curve fitting using data from the experimental set-up.

The parameters $k1_x \ldots k36_x$ are associated with (i), (ii), and the parameter $k37_x$ is associated with (iii), in Section 3.1.

$$T_{dx} = k1_x\dot{\theta}_m + k2_x\theta_m + \cdots + k34_x(\dot{\psi}_a^2\theta_m^2) + k35_x(\dot{\psi}_a^2\dot{\theta}_m^2) + k36_x(\dot{\theta}_m^2\theta_m^2) + k37_x \qquad (7)$$

The generic base disturbance data history, (p, q, r, a_x, a_y, a_z), used in simulations, is obtained from a real-time flight of the carrier vehicle.

In order to validate the effectiveness of the proposed control scheme, experiments are carried out in this section. For the experiments below, NN is trained based on the data under amplitude and frequency varying sinusoidal excitations that travel a wide range in the FOR limits of the experimental set-up. For the NN to learn the delta accelerations effectively, the training input was constructed to resemble a chirp signal (training input is a signal with varying amplitude and frequency, it also includes a phase shift). The reference position travels from 30° to 1° for the azimuth axis and from 15° to 1° for the elevation axis, with an increasing frequency of 0.5 to 5 Hz for the first 10 seconds. Then it travels back to 30° for the azimuth axis, 15° for the elevation axis while decreasing the frequency from 5 Hz to 0.5 Hz and forming sine waves with a 45° phase difference. MATLAB® Neural Network Toolbox is used for training this network. If the validation set error increases for more than 30 iterations, training stops to prevent overfitting.

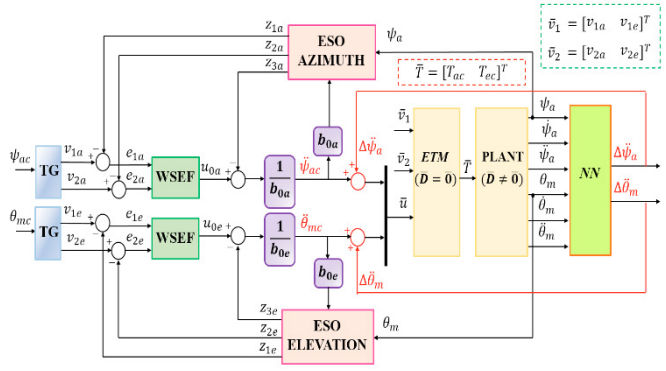

Figure 6. Block diagram of the controller with TG, WSEF, ESO, ETM and Physics-Informed NN.

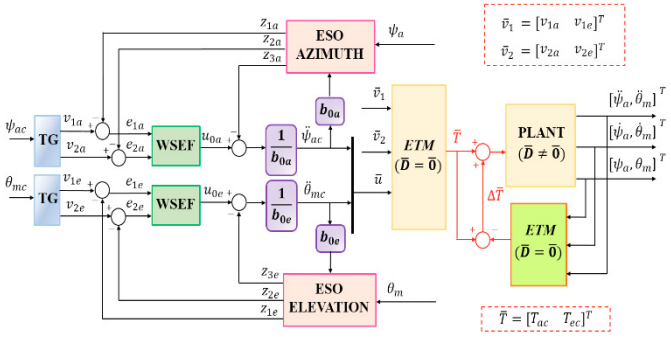

Figure 7. Block diagram of the *ETM* based ADRC structure.

In the first test case, azimuth and elevation gimbals are excited with a reference (*) of 15° and 10°, 0.5Hz sine waves along two axes. This is a low frequency and high amplitude reference that represents a possible input (for the target detection & search sequence) while gimbal is trying to locate the target. Fig. 8 gives the trajectory tracking response of the azimuth gimbal. It has been observed that, mean of the absolute normalized tracking error (MTE) of ADRC, ETM-ADRC and NN-ADRC controllers are very close to each other; they are: 4.4%, 3.5% and 2.8%, respectively. All three controllers have shown sufficient performance in terms of target tracking.

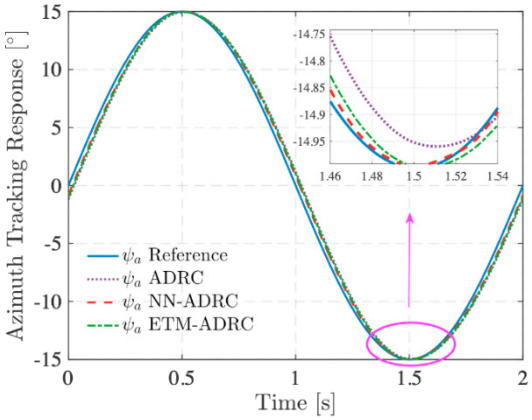

Figure 8. Azimuth gimbal position for the reference (*)

In the second test case, both azimuth and elevation gimbals are excited with a reference (**) of 0.2°, 5Hz sine wave along two axes. This input sequence is used after gimbal locks-on a desired point of interest and starts tracking the target (while gimbal aims to overcome the dominant disturbance profile created by the missile). Fig. 9 gives the trajectory tracking response of the azimuth gimbal. NN supports the ADRC controller and helps the gimbal to reach the reference position (0.2°) with a faster tracking rate. The MTE and percent improvement of hybrid controllers are given in Tab. 3 (% decrease indicates the MTE enhancement wrt. legacy ADRC). The MTE of ADRC is 17.4% for this example (which is approximately 4.1 times larger compared to MTE of *NN-ADRC* controller). This value was 1.6 in the previous example. The control accuracy of the ADRC drops considerably. This situtation shows the input dependency of ADRC parameters.

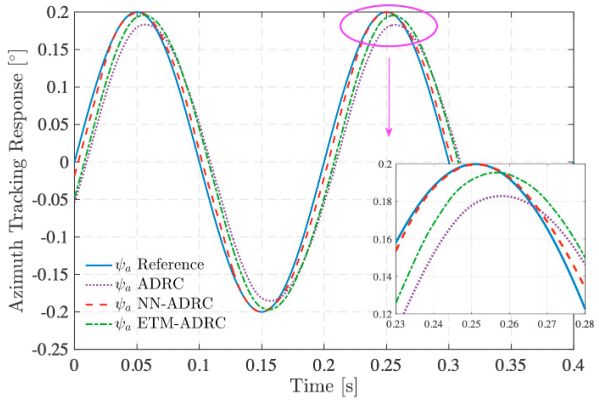

Figure 9. Azimuth gimbal position for the reference (**)

% decrease of *NN-ADRC* is slightly better than the *ETM-ADRC* structure almost in all cases (due to a possible one-step delay in the subtraction of first and second *ETM* block outputs and addition of $\Delta\bar{T}$ to the system). Further investigation indicated that, the torques calculated in NN-ADRC structure are leading those calculated in ETM-ADRC structure approximately by 8 ms as in Fig. 10. This is an obvious result, since NN-ADRC structure is calculating the additive accelerations while ETM-ADRC is calculating the additive

torques. Compensatory accelarations are injected earlier than the compensatory torques as shown in Figs. 6 and 7.

Table 3. MTE comparison for Scenario 2 (**)

	NN-ADRC	ETM-ADRC
MTE	4.2%	14%
% decrease	13.2	3.4

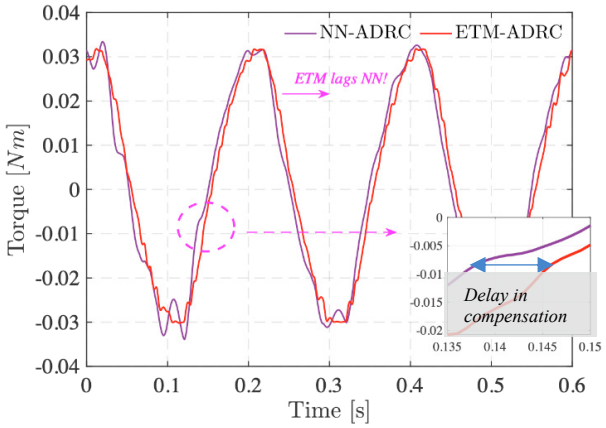

Figure 10. Azimuth torque applied to the plant for the reference (**)

5. CONCLUSIONS AND FUTURE WORK

For the simulation results provided in this study, NN is trained with frequency and amplitude varying data that travels within the FOR limits of the gimbal system. Two different approaches are presented: ETM-ADRC and the more advantageous NN-ADRC, which uses the ETM indirectly during its training and performs MIMO-based compensation for the multi-axis gimbal model, reducing the time and effort required for tuning the parameters of ADRC controller and decreasing MTE up to 4.1 times when compared to legacy ADRC. Moreover, it can decrease the MTE at least three times compared to that of the ETM-ADRC due to better utilization and estimation of the disturbance torque with less delay. Real time adjustment of the weights of NN, may give the proposed algorithm strong robustness and environmental adaptability. As a future study, we are planning to (i) implement the NN-ADRC hybrid controller on the physical system in real-time , (ii) continue the online training of NN within the physical system in real-time, (iii) apply the proposed controller design scheme on different gimbal topologies (i.e., roll-pitch gimbal).

ACKNOWLEDGEMENT

This study is a joint work between Roketsan Missiles Inc. and Bilkent University, Department of Mechanical Engineering.

REFERENCES

Abdo, M.M., Vali, A.R., Toloei, A.R., and Arvan, M.R. (2015). Improving Two-Axes Gimbal Seeker Performance Using Cascade Control Approach. *Journal of Aerospace Engineering*, 229(1), 38-55.

Atesoglu, O., Nalbantoglu, V., and Seymen, B. (2016). Gyro-Bias Estimation Filter Design for the Stabilization Accuracy Enhancement of Two Axes Gimbaled Sighting Systems. *IFAC Proceedings*, 42(2), 5011-5017.

Bai, C., Zhang, Z. (2018). A Least Mean Square Based Active Disturbance Rejection Control for an Inertially Stabilized Platform. *Optik-International Journal for Light and Electron Optics*, 174, 609-622.

Baskin, M., Leblebicioglu, K. (2017). Robust Control for Line-of-Stabilization of a Two-Axis Gimbal System. *Turkish Journal of Electrical Engineering & Computer Science*, 25(5), 3839-3854.

Ekstrand, B. (2001). Equations of Motion for a Two-Axes Gimbal System. *IEEE Transactions on Aerospace and Electronic Systems*, 37(3), 1083-1091.

Han, J. (2009). From PID to Active Disturbance Rejection Control. *IEEE Transactions on Industrial Electronics*, 56(3), 900-906.

Hilkert, J.M. (2008). Inertially Stabilized Platform Technology Concepts and Principles. *IEEE Control Systems Magazine*, 28(1), 26-46.

Leblebicioglu, D., Atesoglu, O., Derinoz, A.E., and Cakmakci, M. (2021, December 5). Learning-Based Control Compensation for Multi-Axis Gimbal Systems Using Inverse and Forward Dynamics. arXiv:2112.02561v1 [eess.SY]

Liu, W., Zhao, T. (2021). An Active Disturbance Rejection Control for Hysteresis Compensation based on Neural Networks Adaptive Control. *ISA Transactions*, 109, 81-88.

Mao, J., Li, S., Li, Q., and Yang, J. (2019). Design and Implementation of Continuous Finite-Time Sliding Mode Control for 2-DOF Inertially Stabilized Platform Subject to Multiple Disturbances. *ISA Transactions*, 84, 214-224.

Qiao, H., Meng, H., Ke, W., Gao, Q., and Wang, S. (2020). Adaptive Control of Missile Attitude Based on BP-ADRC. *Aircraft Engineering and Aerospace Technology*, 92(10), 1475-1481.

Shen, S., Xu, J. (2021). Adaptive Neural Network Based Active Disturbance Rejection Flight Control of an Unmanned Helicopter. *Aerospace Science and Technology*, 119, 107062.

Wang, F., Wang, R., Liu, E., and Zhang, W. (2019). Stabilization Control Method for Two-Axis Inertially Stabilized Platform Based on Active Disturbance Rejection Control with Noise Reduction Disturbance Observer. *IEEE Access*, 7, 99521-99529.

Xiang, B., Mu, Q. (2021). Gimbal Control of Inertially Stabilized Platform for Airbone Remote Sensing System based on Adaptive RBFNN Feedback. *IFAC Journal of Systems and Control*, 16, 100148.

Zhou, X., Gao, H., Zhao, B., and Zhao, L. (2018). A GA-Based Parameters Tuning Method for an ADRC Controller of ISP for Aerial Remote Sensing Applications. *ISA Transactions*, 81, 318-328.