Turkish Journal of Electrical Engineering and Computer Sciences

Volume 29 | Number 5 Article 16

1-1-2021

# Line-of-sight rate construction for a roll-pitch gimbal via a virtualpitch-yaw gimbal

OĞUZHAN ÇİFDALÖZ

Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2021) 29: 2502 – 2519

doi:10.3906/elk-2010-121

# Line-of-sight rate construction for a roll-pitch gimbal via a virtual pitch-yaw gimbal

**Oğuzhan ÇİFDALÖZ***
Deptartment of Electrical-Electronics Engineering, Faculty of Engineering, Çankaya University, Ankara, Turkey

**Received:** 29.10.2020 • **Accepted/Published Online:** 17.05.2021 • **Final Version:** 23.09.2021

**Abstract:** In this paper, a method to construct the line of sight rate of a target with a roll-pitch gimbal and tracker is described. Construction of line-of-sight rate is performed via utilizing a virtual pitch-yaw gimbal. Kinematics of both the roll-pitch and pitch-yaw gimbals are described. A dynamical model for the roll-pitch gimbal is developed, and a nested control structure is designed to control the angular rates and line of sight angles. A kinematic model of the tracker is developed and a tracker controller is designed to keep the target in the field of view. Conversion equations between roll-pitch and pitch-yaw gimbal configurations are provided. Finally, constructed line of sight rates are compared to true line of sight rates via simulations. Obtained results indicate that the constructed line of sight rates pertaining to a target satisfactorily converge to the actual line of sight rates.

**Key words:** Roll-pitch gimbal, virtual gimbal, line of sight rate, kinematic tracker model, gimbal dynamics, seeker control design

## 1. Introduction and problem statement

Proportional navigation (PN) is perhaps the most common method for homing guidance of short-to-medium range missiles [1–5]. Although there are variants of it, PN is based on the realization that two closing objects will collide if the line of sight (LoS) between them does not rotate with respect to an inertial (non-rotating) frame. Hence, the main objective of proportional navigation is to null the LoS rate between two objects, one being the missile and the other a target. This is achieved by commanding the missile autopilot with normal acceleration, which is *proportional* to the LoS rate measured by the on-board missile seeker [2, 3].

$$a_n = NV_c\dot{\lambda} \tag{1}$$

where $a_n$ denotes the commanded normal acceleration, $N$ the navigation constant, $V_c$ the closing velocity, and $\dot{\lambda}$ the the LoS rate measured by the on board missile seeker. As described in [3, p. 194], $N$ is based on the missile's acceleration requirements and will vary depending on target maneuvers and other system-induced tracking-error sources. In order to minimize the missile acceleration requirement, values of $N$ between 3 and 5 are usually used to obtain an acceptable miss distance intercept.

A sensor, such as an infrared detector, is mounted on the inner axis of a two-axis, sometimes a 3-axis, gimbal. Gimbals are stabilized via motors to achieve stabilization using measurements obtained from rate gyros, mounted on the inner axis. The objective is for the sensor to point a target, using the measurements

*Correspondence: oguzhanc@cankaya.edu.tr

received from an optical detector, and if available, from other resources, such as radars. In most cases, two degrees of freedom are sufficient. Most seekers are of the configuration (outer)yaw-(inner)pitch or (outer)pitch-(inner)yaw gimbals [3]. Pitch-yaw gimbals provide line of sight rate estimates more or less directly; hence, the rates measured by the rate gyros can be used to compute the normal acceleration to be commanded to the missile autopilot. Due to mechanical considerations, pitch-yaw gimbals provide a smaller field or regard (the total angle that can be covered by the gimbal) than roll-pitch gimbals [6]. However, rate gyros mounted on the inner axis of roll-pitch gimbals do not provide the necessary LoS rates, directly. Hence, LoS rates need to be constructed.

Various methods were proposed to tackle the problem of constructing the LoS rates from a roll-pitch gimbal. In [7], LoS estimation model is built based on unscented Kalman filters (UKF). Wang, et.al. propose a coordinate transformation method to compensate for the impact of rapid roll motion. A LoS rate dynamics model is derived on the transformed coordinate system, and based on the model, a LoS rate estimator with the Improved Iterated Cubature Kalman Filter algorithm (IICKF) is designed [8]. A kinematic approach is in [9]. In [10], various schemes of gyroscope configurations and control are utilized to transform the inertial angular rate into the line-of-sight (LoS) coordinate system with the aid of recursive angular velocity equations. [11] discusses a control strategy to improve the disturbance rejection ability of roll-pitch seeker using a differentiator-based disturbance compensation controller.

The problem of LoS rate estimation is not limited to roll-pitch gimbal configurations. Although this paper is specific to a roll-pitch gimbals, studies which address LoS rate estimation for other gimbal configurations is also relevant. [12] address loS rate modeling and error analysis via coordinate transformations. In [13], LoS rates are analyzed for a pitch-yaw gimbal when the platform makes roll rotations. LoS rate estimation methods for strapdown seekers, which do not utilize gimbals but are rigidly mounted on the platform, are discussed in [14–17]. [14] utilizes Extended Kalman Filters, [15] proposes a solution based on attitude dynamics and [16] utilizes adaptive unscented Kalman filters, and [17] takes into account a target performing a specific type of maneuver.

The main problem addressed in this paper is constructing the line-of-sight rates of a target when a roll-pitch gimbal is used for target tracking. While pitch-yaw gimbals provide LoS rates directly to be used for missile guidance, roll-pitch gimbals cannot. Sensor measurements mounted on a roll-pitch gimbal configuration need to be processed to construct the required LoS rates. Main contributions of this work include the development of a kinematic tracker model mounted on a roll-pitch platform and a LoS rate reconstruction scheme based on a *virtual* gimbal. A virtual gimbal is a mathematical gimbal model embedded in the line of sight rate estimation algorithm. It is purely a software model and does not physically exist. To the best of the author's knowledge, none of these were addressed directly in the literature. The kinematic tracker model describes an ideal tracker. This tracker model does not only generate the angular distance of the target from the center of the detector, it also generates roll-pitch angular errors to be processed by the outer loop controllers. As for the LoS construction scheme, a kinematic solution is described in this paper, in which a virtual gimbal and its characteristics are utilized. Described method accepts roll-pitch gimbal angles, platform body angular rates, platform Euler angles, and tracking errors to construct LoS angles and LoS rates applicable to platform guidance. Other solutions in the literature employ various Kalman filtering methods. Another advantage of the proposed solution to the Kalman filter based solutions is that, an accurate model of the gimbal dynamics is not required. Standard Kalman filter based methods would need a linearized model, extended or unscented Kalman filter methods would require nonlinear models or discretized nonlinear models. Proposed solution uses

an approximate linear model to design the controllers and relies on the robustness properties of the closed loop system. The justification for using a linear model is presented in Section 3.

The organization of the paper is as follows: Notation and preliminaries are provided in Section 2. Section 3 describes the kinematics of both roll-pitch and pitch-yaw gimbal configurations, and presents a dynamical model for the roll-pitch gimbal. Nested control structure and the structures of controllers are presented. Section 4 describes the kinematics of an ideal seeker. Section 5 presents the necessary conversion equations between the two types of gimbal configurations that are considered in the paper and the final steps to construct the LoS rate. Section 6 includes the simulation results of the proposed method. Section 7 summarizes the results and describes the future directions.

## 2. Notation and preliminaries

### 2.1. Notation

Notation adopted in this paper is due to [18, p. 22]. Rotational transformation of a vector, $\mathbf{x}^a$, in frame $F_a$ into a vector, $\mathbf{x}^b$, in frame $F_b$ is described as

$$\mathbf{x}^b = C_a^b \, \mathbf{x}^a$$

where $F_a$ and $F_b$ are orthogonal and right handed, subscript $a$ in $C_a^b$ denotes the ***reference*** frame, superscript $b$ in $C_a^b$ denotes the ***target*** frame, and $\mathbf{x}^b$ is the representation of the vector $\mathbf{x}^a$ in frame $F_b$. Since $C_a^b$ is orthonormal, and $(C_a^b)^{-1} = (C_a^b)^T = C_b^a$ from which it follows $\mathbf{x}^a = C_b^a \, \mathbf{x}^b$.

Throughout the document, the entry on the $i$th row and $j$th column of a transformation matrix $C_b^a$ will be denoted by $C_b^a(i,j)$.

The angular velocity of the $k$-frame relative to the $m$-frame, as resolved (measured) in the $p$-frame, is represented by $\omega_{mk}^p$.

*Roll*, *pitch*, and *yaw* rotations are expressed mathematically as direction cosine matrices as:

$$R_1(\phi) \triangleq \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix}, \; R_2(\theta) \triangleq \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix}, \; R_3(\psi) \triangleq \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{2}$$

### 2.2. Reference coordinate frames and transformations

Four fundamental reference frames are of importance to describe the method presented in this paper [18, p. 21]. However, two of them will play the biggest role.

*Inertial frame* ($i$-frame) has its origin at the center of the Earth and axes which are non-rotating with respect to the fixed stars; with its $z$-axis coincident with the Earth's polar axis (which is assumed to be invariant in direction).

*Earth fixed frame* ($e$-frame) has its origin at the center of the Earth and axes which are fixed with respect to the Earth, with its $z$-axis along the Earth's polar axis. Its $y$-axis lies along the intersection of the plane of the Greenwich meridian with the Earth's equatorial plane. The Earth frame rotates, with respect to the inertial frame, at a rate of $\Omega \approx 15$ deg/h about its $z$-axis.

*Navigation frame* ($n$-frame) is a local geographic frame which has its origin at the location of the navigation system (vehicle, craft, etc.), and axes aligned with the directions of north, east and the local vertical

(down). For most applications (mid/short range missiles, land and sea vehicles, etc.) earth, inertial, and navigation frames can be assumed to be coincident.

*Body fixed frame* ($b$-frame), is an orthogonal axis set which is aligned with the roll ($\phi$), pitch ($\theta$), and yaw ($\psi$) axes of the vehicle in which the navigation system is installed. The orientation of the craft's body frame is along its forward, side-wards right, and downwards axis. The origin of the body fixed frame is generally located at the center of the inertial measurement unit.

$\phi$, $\theta$, and $\psi$ are called the Euler Angles and are related to the body angular rates as follows [18, p. 42]:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi/\cos\theta & \cos\phi/\cos\theta \end{bmatrix} \begin{bmatrix} \tilde{p} \\ \tilde{q} \\ \tilde{r} \end{bmatrix} \tag{3}$$

where $\tilde{p}$, $\tilde{q}$, $\tilde{r}$ are the body roll, pitch, and yaw rates of the platform, respectively.

The transformation matrix from the navigation to body frame is given by

$$C_n^b = R_1(\phi)R_2(\theta)R_3(\psi) \tag{4}$$

Transposing $C_n^b$ yields the transformation matrix from the body to navigation frame [18, p. 41]:

$$C_b^n = \begin{bmatrix} \cos\psi\cos\theta & \cos\psi\sin\theta\sin\phi - \sin\psi\cos\phi & \cos\psi\sin\theta\cos\phi + \sin\psi\sin\phi \\ \sin\psi\cos\theta & \sin\psi\sin\theta\sin\phi + \cos\psi\cos\phi & \sin\psi\sin\theta\cos\phi - \cos\psi\sin\phi \\ -\sin\theta & \cos\theta\sin\phi & \cos\theta\cos\phi \end{bmatrix} \tag{5}$$

## 3. Gimbal models and control structure

This section describes the kinematic and dynamic models of two types of gimbal configurations: outer roll/inner pitch gimbal and the virtual outer pitch/inner yaw gimbal. In this paper, it is considered that the roll-pitch gimbal is mounted on the seeker head. The pitch-yaw gimbal, on the other hand, is a virtual (nonexistent) gimbal. However, the kinematic model of the pitch-yaw gimbal is utilized to construct the signals required for the navigation.

### 3.1. Outer roll - inner pitch gimbal

In the analysis below, $r$ denotes the outer (roll) frame, $p$ denotes the inner (pitch) frame of the roll-pitch gimbal. $\phi_{\text{rp}}$ and $\theta_{\text{rp}}$ denote the relative gimbal angles as measured by encoders. In order to combine with the dynamical model, the kinematics of the roll-pitch gimbal is derived. The angular rates as resolved in the (outer) roll axis is given by

$$\omega_{ir}^r = \omega_{ib}^r + \omega_{br}^r = C_b^r\omega_{ib}^b + \omega_{br}^r = R_1(\phi_{\text{rp}}) \begin{bmatrix} \tilde{p} \\ \tilde{q} \\ \tilde{r} \end{bmatrix} + \begin{bmatrix} \dot{\phi}_{\text{rp}} \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \tilde{p} + \dot{\phi}_{\text{rp}} \\ \tilde{q}\cos\phi_{\text{rp}} + \tilde{r}\sin\phi_{\text{rp}} \\ -\tilde{q}\sin\phi_{\text{rp}} + \tilde{r}\cos\phi_{\text{rp}} \end{bmatrix} \tag{6}$$

where $\phi_{\text{rp}}$ is the gimbal roll angle. If a *virtual* 3-axis rate gyro was mounted on the axis of rotation of the outer gimbal, it would measure $\omega_{ir}^r$.

The angular rates as resolved in the (inner) pitch axis is given by

$$\begin{bmatrix} \omega_{x_{\rm rp}} \\ \omega_{y_{\rm rp}} \\ \omega_{z_{\rm rp}} \end{bmatrix} \triangleq \omega_{ip}^{p} = R_2(\theta_{\rm rp})R_1(\phi_{\rm rp}) \begin{bmatrix} \tilde{p} \\ \tilde{q} \\ \tilde{r} \end{bmatrix} + R_2(\theta_{\rm rp}) \begin{bmatrix} \dot{\phi}_{\rm rp} \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ \dot{\theta}_{\rm rp} \\ 0 \end{bmatrix}$$

$$= \begin{bmatrix} (\tilde{p} + \dot{\phi}_{\rm rp})\cos\theta_{\rm rp} - \tilde{r}\cos\phi_{\rm rp}\sin\theta_{\rm rp} + \tilde{q}\sin\phi_{\rm rp}\sin\theta_{\rm rp} \\ \dot{\theta}_{\rm rp} + \tilde{q}\cos\phi_{\rm rp} + \tilde{r}\sin\phi_{\rm rp} \\ (\tilde{p} + \dot{\phi}_{\rm rp})\sin\theta_{\rm rp} + \tilde{r}\cos\phi_{\rm rp}\cos\theta_{\rm rp} - \tilde{q}\sin\phi_{\rm rp}\cos\theta_{\rm rp} \end{bmatrix} \tag{7}$$

where $\theta_{\rm rp}$ is the gimbal pitch angle. If a *virtual* 3-axis rate gyro was mounted on the axis of rotation of the outer gimbal, it would measure $\omega_{ip}^{p}$. If a 2-axis roll-pitch rate gyro is used, only the first two components of (7) are available as measurements. In such a case, $\dot{\phi}_{\rm rp}$ can be obtained from Line 1 of (7) as

$$\dot{\phi}_{\rm rp} = \frac{\omega_{x_{\rm rp}} + \tilde{r}\cos\phi_{\rm rp}\sin\theta_{\rm rp} - \tilde{q}\sin\phi_{\rm rp}\sin\theta_{\rm rp}}{\cos\theta_{\rm rp}} - \tilde{p} \tag{8}$$

and, $\dot{\theta}_{\rm rp}$ can be obtained from Line 2 of (7) as

$$\dot{\theta}_{\rm rp} = \omega_{y_{\rm rp}} - \tilde{q}\cos\phi_{\rm rp} - \tilde{r}\sin\phi_{\rm rp} \tag{9}$$

There is a singularity in the solutions when $\cos\theta_{\rm rp} = 0$. However, this case would normally not be a problem because, due to physical limitations in most applications, $\theta_{\rm rp}$ cannot take values close to $\pm 90^{\circ}$.

A linear dynamical model, which includes inertia and friction terms, are derived for the roll-pitch gimbal considered in this work [19, 20]. Static (coulomb) friction is treated as model uncertainty [21] and included in the simulations presented below. A block diagram of the (outer) roll axis is given in Figure 1.

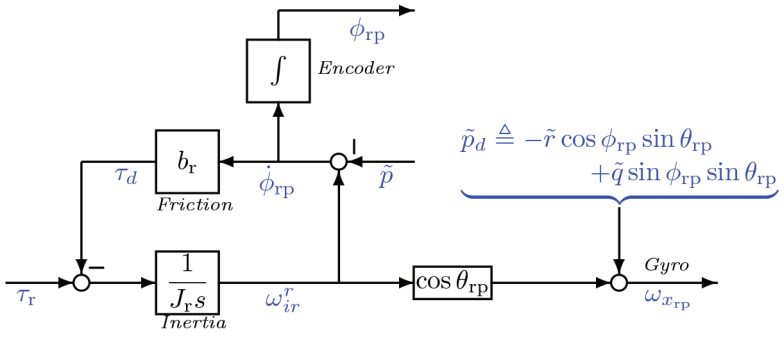



**Figure 1**. Roll (outer) axis dynamical model.

Platform angular rates are treated as a disturbance and implemented in the diagram using (6) and (7). In the block diagram, $J_{\rm r}$ denotes the roll axis inertia, $b_{\rm r}$ denotes the viscous friction constant, and $\tau_{\rm r}$ is the torque applied by the roll motor. It should be noted that this model assumes the motor dynamics are fast enough to be neglected, spring effects are negligible, the gimbal is perfectly balanced, and rigid [20, 21].

State equations are given by

$$\begin{bmatrix} \dot{\omega}_{ir}^{r} \\ \dot{\phi}_{\rm rp} \end{bmatrix} = \begin{bmatrix} -\frac{b_{\rm r}}{J_{\rm r}} & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \omega_{ir}^{r} \\ \phi_{\rm rp} \end{bmatrix} + \begin{bmatrix} \frac{1}{J_{\rm r}} & \frac{b_{\rm r}}{J_{\rm r}} \\ 0 & -1 \end{bmatrix} \begin{bmatrix} \tau_{\rm r} \\ \tilde{p} \end{bmatrix} \tag{10}$$

When the outputs are selected as $\phi_{\rm rp}$ and $\omega_{ir}^{r}$, the transfer function matrix for the system is given by

$$\begin{bmatrix} \Omega_{ir}^{r} \\ \Phi_{\rm rp} \end{bmatrix} = \begin{bmatrix} \frac{1}{J_{\rm r}s+b_{\rm r}} & \frac{b_{\rm r}}{J_{\rm r}s+b_{\rm r}} \\ \frac{1}{J_{\rm r}s^2+b_{\rm r}s} & -\frac{J_{\rm r}}{J_{\rm r}s+b_{\rm r}} \end{bmatrix} \begin{bmatrix} T_{\rm r} \\ \tilde{P} \end{bmatrix} \tag{11}$$

where the Laplace transform of variables are given with their capital letter counterparts.

Hence, the system is a 2-input 2-output system, where the inputs are the applied torque, $\tau_{\rm r}$, and the disturbance due to the roll rate of the platform, $\tilde{p}$.

The transfer function from $\tau_{\rm r}$ to $\omega_{ir}^{r}$

$$P_{\rm roll}(s) \triangleq \frac{\Omega_{ir}^{r}(s)}{T_{\rm r}(s)} = \frac{1}{J_{\rm r}s + b_{\rm r}} \tag{12}$$

is used to design an angular rate controller, $K_{\rm r}(s)$. Noting that

$$\omega_{x_{\rm rp}} = \omega_{ir}^{r}\cos\theta_{\rm rp} + \tilde{p}_d \tag{13}$$

where $\tilde{p}_d \triangleq \tilde{q}\sin\phi_{\rm rp}\sin\theta_{\rm rp} - \tilde{r}\cos\phi_{\rm rp}\sin\theta_{\rm rp}$, the design can be extended to control $\omega_{x_{\rm rp}}$. The output of $K_{\rm r}(s)$ is multiplied by $\sec\theta_{\rm rp}$ for the purpose of canceling out $\cos\theta_{\rm rp}$ (secant correction), and $\tilde{p}_d$ constitutes the output disturbance.

A block diagram of the (inner) pitch axis is given in Figure 2. Platform angular rates are treated as a disturbance and implemented in the diagram using (7). In the block diagram, $J_{\rm p}$ denotes the pitch axis inertia, $b_{\rm p}$ denotes the viscous friction constant, and $\tau_{\rm p}$ is the applied torque by the pitch motor.

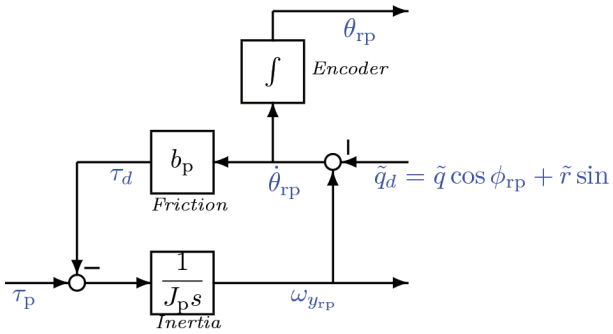



**Figure 2**. Pitch (inner) axis dynamical model.

State equations are given by

$$\begin{bmatrix} \dot{\omega}_{y_{\rm rp}} \\ \dot{\theta}_{\rm rp} \end{bmatrix} = \begin{bmatrix} -\frac{b_{\rm p}}{J_{\rm p}} & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \omega_{y_{\rm rp}} \\ \theta_{\rm rp} \end{bmatrix} + \begin{bmatrix} \frac{1}{J_{\rm p}} & \frac{b_{\rm p}}{J_{\rm p}} \\ 0 & -1 \end{bmatrix} \begin{bmatrix} \tau_{\rm p} \\ \tilde{q}_d \end{bmatrix} \tag{14}$$

When the outputs are selected as $\theta_{\rm rp}$ and $\omega_{y_{\rm rp}}$, the transfer function matrix for the system is given by

$$\begin{bmatrix} \Omega_{y_{\rm rp}} \\ \theta_{\rm rp} \end{bmatrix} = \begin{bmatrix} \frac{1}{J_{\rm p}s+b_{\rm p}} & \frac{b_{\rm p}}{J_{\rm p}s+b_{\rm p}} \\ \frac{1}{J_{\rm p}s^2+b_{\rm p}s} & -\frac{J_{\rm p}}{J_{\rm p}s+b_{\rm p}} \end{bmatrix} \begin{bmatrix} T_{\rm p} \\ \tilde{Q}_d \end{bmatrix} \tag{15}$$

Similarly, as in the outer axis, the system is a 2-input 2-output system, where the inputs are the applied torque, $\tau_{\rm p}$, and the disturbance due to the pitch and yaw rates of the platform $\tilde{q}_d \triangleq \tilde{q}\cos\phi_{\rm rp} + \tilde{r}\sin\phi_{\rm rp}$; and the outputs are the pitch angle ($\theta_{\rm rp}$) as measured by the encoder and the angular rate ($\omega_{y_{\rm rp}}$) as measured by the rate gyro located on the (inner) pitch axis of the gimbal.

The transfer function from $\tau_{\rm p}$ to $\omega_{y_{\rm rp}}$

$$P_{\rm pitch}(s) \triangleq \frac{\Omega_{y_{\rm rp}}(s)}{T_{\rm p}(s)} = \frac{1}{J_{\rm p}s + b_{\rm p}} \tag{16}$$

is used to design an angular rate controller, $K_{\rm p}(s)$.

### 3.2. Virtual outer pitch - inner yaw gimbal

In the analysis below, $p$ denotes the outer (pitch) frame, $y$ denotes the inner (yaw) frame of the pitch-yaw gimbal. $\theta_{\rm py}$ and $\psi_{\rm py}$ denote the relative gimbal angles as measured by encoders. Kinematics of the roll-pitch gimbal is derived. The angular rates as resolved in the (outer) pitch axis is given by

$$\omega_{ip}^{p} = \omega_{ib}^{p} + \omega_{bp}^{p} = C_b^p\omega_{ib}^{b} + \omega_{bp}^{p} = R_2(\theta_{\rm py})\begin{bmatrix}\tilde{p}\\ \tilde{q}\\ \tilde{r}\end{bmatrix} + \begin{bmatrix}0\\ \dot{\theta}_{\rm py}\\ 0\end{bmatrix} = \begin{bmatrix}\tilde{p}\cos\theta_{\rm py} - \tilde{r}\sin\theta_{\rm py}\\ \tilde{q} + \dot{\theta}_{\rm py}\\ \tilde{p}\sin\theta_{\rm py} + \tilde{r}\cos\theta_{\rm py}\end{bmatrix} \tag{17}$$

where $\theta_{\rm py}$ is the gimbal pitch angle. $\omega_{ip}^{p}$ can be viewed as the measurement of a virtual 3-axis rate gyro mounted on the outer axis. The angular rates as resolved in the (inner) yaw axis is given by

$$\begin{aligned}\begin{bmatrix}\omega_{x_{\rm py}}\\ \omega_{y_{\rm py}}\\ \omega_{z_{\rm py}}\end{bmatrix} \triangleq \omega_{iy}^{y} &= R_3(\psi_{\rm py})R_2(\theta_{\rm py})\begin{bmatrix}\tilde{p}\\ \tilde{q}\\ \tilde{r}\end{bmatrix} + R_3(\psi_{\rm py})\begin{bmatrix}0\\ \dot{\theta}_{\rm py}\\ 0\end{bmatrix} + \begin{bmatrix}0\\ 0\\ \dot{\psi}_{\rm py}\end{bmatrix}\\ &= \begin{bmatrix}(\tilde{q} + \dot{\theta}_{\rm py})\sin\psi_{\rm py} + \tilde{p}\cos\theta_{\rm py}\cos\psi_{\rm py} - \tilde{r}\sin\theta_{\rm py}\cos\psi_{\rm py}\\ (\tilde{q} + \dot{\theta}_{\rm py})\cos\psi_{\rm py} - \tilde{p}\cos\theta_{\rm py}\sin\psi_{\rm py} + \tilde{r}\sin\theta_{\rm py}\sin\psi_{\rm py}\\ \dot{\psi}_{\rm py} + \tilde{r}\cos\theta_{\rm py} + \tilde{p}\sin\theta_{\rm py}\end{bmatrix}\end{aligned} \tag{18}$$

where $\psi_{\rm py}$ is the gimbal yaw angle. $\omega_{iy}^{y}$ is the signal that would be measured by a 3-axis rate gyro mounted on the inner gimbal. If a 2-axis pitch-yaw rate gyro is used, only the last two components of (18) are available as measurements. In such a case, $\dot{\theta}_{\rm py}$ can be obtained from Line 2 of (18) as

$$\dot{\theta}_{\rm py} = \frac{\omega_{y_{\rm py}} + p\cos\theta_{\rm py}\sin\psi_{\rm py} - r\sin\theta_{\rm py}\sin\psi_{\rm py}}{\cos\psi_{\rm py}} - q \tag{19}$$

and, $\dot{\psi}_{\rm py}$ can be obtained from Line 3 of (18) as

$$\dot{\psi}_{\rm py} = \omega_{z_{\rm py}} - r\cos\theta_{\rm py} - p\sin\theta_{\rm py} \tag{20}$$

There is a singularity in the solutions when $\cos\psi_{\rm py} = 0$. However, in general, due to physical limitations, $\psi_{\rm py}$ cannot take values close to $\pm 90^\circ$.

It is important to note at this point is that $\omega_{y_{\rm py}}$ and $\omega_{z_{\rm py}}$ are the required LoS rates for a PN system.

### 3.3. Control structure

The structure given in Figure 3 describes the control structure of the closed loop system [22]. R/P Gimbal block includes the models developed in Figures 1 and 2. When combined, the gimbal model becomes a 3-input 4-output system. $\tau_{\mathrm{p}}$ and $\tau_{\mathrm{r}}$ denote the roll and pitch torque inputs applied to the system. As stated before the angular rate vector of the platform, $\left[\tilde{p}\ \tilde{q}\ \tilde{r}\right]^T$, is treated as a source of disturbance. $K_{\mathrm{r}}$ and $K_{\mathrm{p}}$ are inner loop rate controllers to control $\omega_{x_{\mathrm{rp}}}$ and $\omega_{y_{\mathrm{rp}}}$. Bandwidth requirements used to design the angular rate are controllers are usually high and are based on the missile and target characteristics. $K_{\mathrm{r}}^t$ and $K_{\mathrm{p}}^t$ are outer loop tracker controllers. They process LoS angle errors provided by the tracker algorithm and generate rate commands, $\omega_{x_{\mathrm{rp}}}^c$ and $\omega_{y_{\mathrm{rp}}}^c$, to feed into the inner loop. Tracker controllers are designed so that tracker closed loop bandwidths are roughly one tenth of the angular rate loop bandwidths. The bandwidth of the tracker loops are determined by the capabilities of the image processing algorithms and detectors used for target tracking. Fast tracker loops may reduce the effectiveness of tracker algorithms simply due to the fact that a single pixel may appear as a line on the detector if the gimbals move too fast.

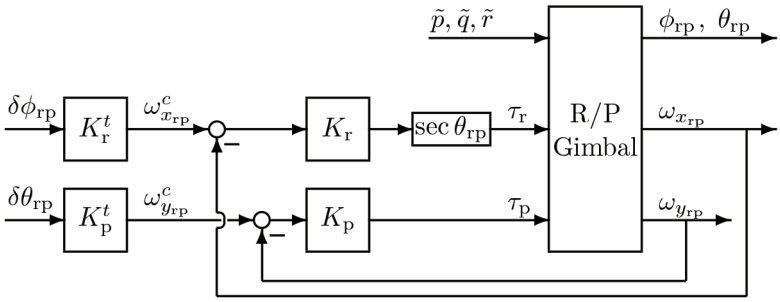



**Figure 3**. Gimbal control - closed loop structure.

$\delta\phi_{\mathrm{rp}}$ and $\delta\theta_{\mathrm{rp}}$ are the errors provided as an input to the outer loop tracker controllers. These signals need to be constructed by the tracking algorithm or by the control algorithm. Tracker model will be described next in Section 4.

## 4. Target tracking model

A tracker is an independent system of software and hardware, which identifies a target and produces the tracking error, the distance of the target from the center of the optical detector in two dimensions, usually in pixels [23]. In this section, the kinematics of a tracker will be described and the equations to generate the tracking error will be developed. In order to achieve this objective, a method to generate the true LoS angles and rates is required. One such method is described using a pitch-yaw gimbal analogy. Once the true LoS angles and rates are obtained, tracking errors can be generated using the instantaneous angular positions of the roll-pitch gimbal.

### 4.1. LoS angles and LoS rates from target position

LoS is defined as the imaginary line that stretches between an observer's eye and the object that is looked at. As shown in Figure 4, the observer is located at point $p_o$ of the coordinate frame and the object is at point $p_t$. LoS can be expressed by two successive rotations and the distance from the observer. In this paper, LoS angles, $\theta_{\mathrm{los}}$ and $\psi_{\mathrm{los}}$ are defined by an outer elevation/inner azimuth gimbal analogy, i.e. first through an elevation angle ($\theta_{\mathrm{los}}$) and then an azimuth angle ($\psi_{\mathrm{los}}$).

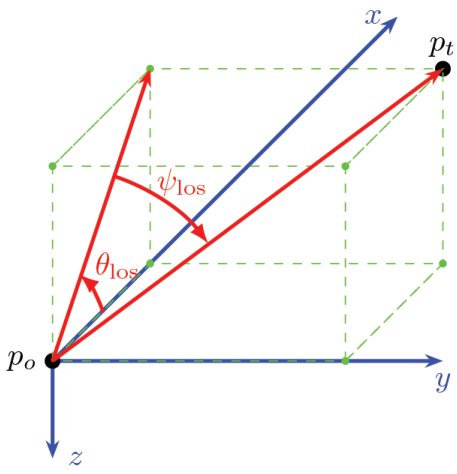



**Figure 4**. Line of sight schematic.

Let the observer and target position vectors be given in the navigation frame by $p_o = \begin{bmatrix} x_o & y_o & z_o \end{bmatrix}^T$ and $p_t = \begin{bmatrix} x_t & y_t & z_t \end{bmatrix}^T$, respectively. Relative position of the target with respect to the observer in the navigation frame is then

$$\mathbf{p}^n = \begin{bmatrix} x_t - x_o \\ y_t - y_o \\ z_t - z_o \end{bmatrix} \triangleq \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{21}$$

*Outer pitch/inner yaw gimbal analogy*: To compute $\theta_{\text{los}}$ and $\psi_{\text{los}}$, $\mathbf{p}^n$ needs to be transformed to the innermost (azimuth or *target*) frame such that the object lies on the $x$-axis of the target frame ($t$-frame) as follows

$$\mathbf{p}^t = C_n^t \mathbf{p}^n = \begin{bmatrix} r \\ 0 \\ 0 \end{bmatrix} \tag{22}$$

where $r = ||\mathbf{p}^n||$ and $C_n^t = R_3(\psi_{\text{los}}) R_2(\theta_{\text{los}})$.

(22) can be rewritten as

$$C_t^n \mathbf{p}^t = \begin{bmatrix} \cos\psi_{\text{los}} \cos\theta_{\text{los}} & -\cos\theta_{\text{los}} \sin\psi_{\text{los}} & \sin\theta_{\text{los}} \\ \sin\psi_{\text{los}} & \cos\psi_{\text{los}} & 0 \\ -\cos\psi_{\text{los}} \sin\theta_{\text{los}} & \sin\psi_{\text{los}} \sin\theta_{\text{los}} & \cos\theta_{\text{los}} \end{bmatrix} \begin{bmatrix} r \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \mathbf{p}^n \tag{23}$$

Hence, in order to obtain the LoS angles with respect to the assumed gimbal mechanization (23) needs to be solved for $\theta_{\text{los}}$ and $\psi_{\text{los}}$. Carrying out the multiplication yields

$$\begin{bmatrix} \cos\psi_{\text{los}} \cos\theta_{\text{los}} \\ \sin\psi_{\text{los}} \\ \cos\psi_{\text{los}} \sin\theta_{\text{los}} \end{bmatrix} = \frac{1}{r} \begin{bmatrix} x \\ y \\ -z \end{bmatrix} \tag{24}$$

where $r$ is the distance of the object from the observer. From (24), LoS Angles can be computed as

$$\theta_{\text{los}} = \text{atan}\left(\frac{-z}{x}\right) \quad \text{and} \quad \psi_{\text{los}} = \text{atan}\left(\frac{y}{\sqrt{x^2 + z^2}}\right) \tag{25}$$

Furthermore, if $\dot{x}$, $\dot{y}$, and $\dot{z}$ are available, true LoS rates can be computed as

$$\dot{\theta}_{\text{los}} = \frac{z\dot{x} - x\dot{z}}{x^2 + z^2} \quad \text{and} \quad \dot{\psi}_{\text{los}} = \frac{\dot{y}(x^2 + z^2) - y(x\dot{x} + z\dot{z})}{(x^2 + y^2 + z^2)\sqrt{x^2 + z^2}} \tag{26}$$

$\dot{\theta}_{\text{los}}$ and $\dot{\psi}_{\text{los}}$ are the true LoS rate signals to assess the performance of the LoS rate construction method. $\dot{x}$, $\dot{y}$, and $\dot{z}$ are not used in anyway to compute the LoS rates. They are used only for the purposes of generating the ground truth for comparison purposes.

### 4.2. Tracking angle errors

Detectors collect the incident radiation from the environment. Field-of-view (FoV) is the range of angles from which this radiation can be collected. It is usually decomposed into its vertical and horizontal components. Distance of a target from the center of FoV is measured in pixels; however, it is naturally converted into angles. In short, tracker angles measure the displacement of a target from the center of a detector [1].

In this paper, the target location will be represented by a target frame ($t$-frame). The vertical and horizontal distance of the target from the center of the detector will be denoted by $\delta\theta$ and $\delta\psi$, respectively. The detector is mounted on the innermost gimbal, i.e. the pitch frame. Hence, a transformation from the pitch frame to the target frame can be defined. However, the order of rotations in this transformation is not obvious. One could choose to compute the tracking errors ($\delta\theta$ and $\delta\psi$) simply by computing the distance of the target from the center in pixels and by converting them to angles. Or, a transformation matrix could be defined. The order of rotations in this case is unimportant since, whichever method is used by the tracking algorithm to compute $\delta\theta$ and $\delta\psi$ is not coupled in any way to the tracker controllers. Tracker controller aims to nullify the tracking error computed by the tracker, regardless of the method the tracker uses to compute the errors. The transformation matrix that will be adopted in this work is selected to be

$$C_p^t = R_3(\delta\psi)R_2(\delta\theta) = \begin{bmatrix} \cos\delta\psi & \sin\delta\psi & 0 \\ -\sin\delta\psi & \cos\delta\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\delta\theta & 0 & -\sin\delta\theta \\ 0 & 1 & 0 \\ \sin\delta\theta & 0 & \cos\delta\theta \end{bmatrix} \tag{27}$$

Finally, the tracker equation can be stated. The transformation matrix from the navigation frame to the target frame is given by

$$C_n^t = R_3(\psi_{\text{los}})R_2(\theta_{\text{los}}) = \begin{bmatrix} \cos\theta_{\text{los}} & \sin\theta_{\text{los}} & 0 \\ -\sin\theta_{\text{los}} & \cos\theta_{\text{los}} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta_{\text{los}} & 0 & -\sin\theta_{\text{los}} \\ 0 & 1 & 0 \\ \sin\theta_{\text{los}} & 0 & \cos\theta_{\text{los}} \end{bmatrix} \tag{28}$$

$C_n^t$ can be decomposed into three components as

$$C_n^t = C_p^t C_b^p C_n^b \tag{29}$$

where $C_b^p$ is defined by the gimbal angles as $C_b^p = R_2(\theta_{\text{rp}})R_1(\phi_{\text{rp}})$ and $C_n^b$ is defined by the platform Euler angles as $C_n^b = R_1(\phi)R_2(\theta)R_3(\psi)$.

(29) is rewritten to compute the tracker angles as

$$C_p^t = C_n^t C_b^n C_p^b \tag{30}$$

The right hand side of (30) can be computed numerically in the tracker model. One final step involves the inclusion of a roll angle in $C_p^t$. This roll angle, denoted $\delta\phi_{\mathrm{L}}$, is the induced roll angle due to combined pitch and yaw rotations. With the inclusion of $-\delta\phi_{\mathrm{L}}$,

$$C_p^t = R_1(-\delta\phi_{\mathrm{L}})R_3(\delta\psi)R_2(\delta\theta) \tag{31}$$

$$= \begin{bmatrix} \cos\delta\psi\cos\delta\theta & \sin\delta\psi & -\cos\delta\psi\sin\delta\theta \\ -\cos\delta\phi_{\mathrm{L}}\cos\delta\theta\sin\delta\psi - \sin\delta\phi_{\mathrm{L}}\sin\delta\theta & \cos\delta\phi_{\mathrm{L}}\cos\delta\psi & \cos\delta\phi_{\mathrm{L}}\sin\delta\psi\sin\delta\theta - \cos\delta\theta\sin\delta\phi_{\mathrm{L}} \\ \cos\delta\phi_{\mathrm{L}}\sin\delta\theta - \cos\delta\theta\sin\delta\phi_{\mathrm{L}}\sin\delta\psi & \sin\delta\phi_{\mathrm{L}}\cos\delta\psi & \sin\delta\phi_{\mathrm{L}}\sin\delta\psi\sin\delta\theta + \cos\delta\phi_{\mathrm{L}}\cos\delta\theta \end{bmatrix} \tag{32}$$

Equating (30) and (32), tracker angles can be computed as

$$\delta\theta = \operatorname{atan}\left(\frac{-C_p^t(1,3)}{C_p^t(1,1)}\right) \quad \text{and} \quad \delta\psi = \arcsin\left(C_p^t(1,2)\right) \tag{33}$$

Since $\delta\theta$ and $\delta\psi$ are computed via the tracker, and $\phi_{\mathrm{rp}}$ and $\theta_{\mathrm{rp}}$ are available through encoder measurements, $\delta\phi_{\mathrm{rp}}$ and $\delta\theta_{\mathrm{rp}}$ can now be computed. As described in Section 3 Figure 3, $\delta\phi_{\mathrm{rp}}$ and $\delta\theta_{\mathrm{rp}}$ are the inputs to the closed loop system. The transformation matrix from the body frame to the target frame is given by

$$C_b^t = C_p^t C_b^p = R_3(\delta\psi)R_2(\delta\theta)R_2(\theta_{\mathrm{rp}})R_1(\phi_{\mathrm{rp}}) \tag{34}$$

Let the coordinates of a target in the body frame is given by $x^b$. Then, $x^t = C_b^t x^b = \begin{bmatrix} r & 0 & 0 \end{bmatrix}^T$ will locate the target on the $x$-axis of the target frame, where $r$ is the distance to the target. Let $\phi_{\mathrm{rp}}^c$ and $\theta_{\mathrm{rp}}^c$ be the required (desired) roll and pitch angles that also locate the target on the $x$-axis of the target frame. Then,

$$x^t = R_2(\theta_{\mathrm{rp}}^c)R_1(\phi_{\mathrm{rp}}^c)x^b = \begin{bmatrix} r & 0 & 0 \end{bmatrix}^T$$

Combining these two equations yield

$$R_1^T(\phi_{\mathrm{rp}}^c)R_2^T(\theta_{\mathrm{rp}}^c)x^t = C_t^b x^t \Longrightarrow \begin{bmatrix} \cos\theta_{\mathrm{rp}}^c \\ \sin\phi_{\mathrm{rp}}^c\sin\theta_{\mathrm{rp}}^c \\ -\cos\phi_{\mathrm{rp}}^c\sin\theta_{\mathrm{rp}}^c \end{bmatrix} = \begin{bmatrix} C_t^b(1,1) \\ C_t^b(2,1) \\ C_t^b(3,1) \end{bmatrix} \tag{35}$$

Solving (35) for $\phi_{\mathrm{rp}}^c$ and $\theta_{\mathrm{rp}}^c$ yields

$$\phi_{\mathrm{rp}}^c = \operatorname{atan}\left(\frac{-C_t^b(2,1)}{C_t^b(3,1)}\right) \pm k\pi \quad \text{and} \quad \theta_{\mathrm{rp}}^c = (-1)^k \arccos\left(C_t^b(1,1)\right) \tag{36}$$

where $k$ is any integer. Finally, expressions for $\delta\phi_{\mathrm{rp}}$ and $\delta\theta_{\mathrm{rp}}$ can be obtained as

$$\delta\phi_{\mathrm{rp}} = \phi_{\mathrm{rp}}^c - \phi_{\mathrm{rp}} \tag{37}$$

$$\delta\theta_{\mathrm{rp}} = \theta_{\mathrm{rp}}^c - \theta_{\mathrm{rp}} \tag{38}$$

## 5. LoS rate construction

Gimbal models described in Section 3 and the tracker model described in Section 4 provide the necessary variables to close the gimbal control loops for target tracking. Tracker model generates $\delta\phi_{\mathrm{rp}}$ and $\delta\theta_{\mathrm{rp}}$ from

target position $(x, y, z)$, platform Euler angles $(\phi, \theta, \psi)$, and gimbal angles $(\phi_{\text{rp}}, \theta_{\text{rp}})$. The control loops, in turn, generate gimbal angles $(\phi_{\text{rp}}, \theta_{\text{rp}})$ and gyro measurements $(\omega_{x_{\text{rp}}}, \omega_{y_{\text{rp}}})$. Given these two measurements (from encoders and rate gyros), $\dot{\phi}_{\text{rp}}$ and $\dot{\theta}_{\text{rp}}$ can be computed using (8) and (9).

Assuming the tracker/gimbal loop is closed and the target is being tracked, constructing the LoS rate will require (18), which describes virtual pitch/yaw gimbal's rate gyro measurements.

Missing variables to construct LoS rates are $\theta_{\text{py}}$, $\psi_{\text{py}}$ and their derivatives. This will be done by relating the roll-pitch gimbal angles to the virtual pitch-yaw gimbal angles [24]. The transformation matrix from the body frame to the roll-pitch gimbal's inner axis is given by

$$C_b^p = R_2(\theta_{\text{rp}}) R_1(\phi_{\text{rp}}) \tag{39}$$

The transformation matrix from the body frame to the pitch-yaw gimbal's inner axis is given by

$$C_b^y = R_3(\psi_{\text{py}}) R_2(\theta_{\text{py}}) \tag{40}$$

Let the coordinates of a target be given in the body frame by $x^b$ and let both gimbals point to the target, i.e. the target is aligned along the $x$-axes of each gimbal. Then, where $r$ denotes the distance to target

$$C_b^p x^b = R_2(\theta_{\text{rp}}) R_1(\phi_{\text{rp}}) x^b = C_b^y x^b = R_3(\psi_{\text{py}}) R_2(\theta_{\text{py}}) x^b = \begin{bmatrix} r \\ 0 \\ 0 \end{bmatrix}. \tag{41}$$

$$\begin{bmatrix} \cos\psi_{\text{py}} \cos\theta_{\text{py}} \\ \sin\psi_{\text{py}} \\ -\cos\psi_{\text{py}} \sin\theta_{\text{py}} \end{bmatrix} = \begin{bmatrix} \cos\theta_{\text{rp}} \\ \sin\phi_{\text{rp}} \sin\theta_{\text{rp}} \\ -\cos\phi_{\text{rp}} \sin\theta_{\text{rp}} \end{bmatrix} \tag{42}$$

Given $\phi_{\text{rp}}$ and $\theta_{\text{rp}}$, (42) has a unique solution given by

$$\theta_{\text{py}} = \operatorname{atan}(\cos\phi_{\text{rp}} \tan\theta_{\text{rp}}) \quad \text{and} \quad \psi_{\text{py}} = \operatorname{atan}\left( \frac{\sin\phi_{\text{rp}} \sin\theta_{\text{rp}}}{\sqrt{1 - \sin\phi_{\text{rp}}^2 \sin\theta_{\text{rp}}^2}} \right) \tag{43}$$

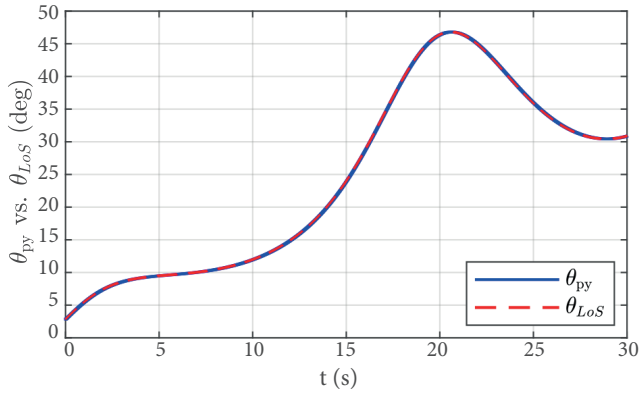
Furthermore, time derivatives of $\theta_{\text{py}}$ and $\psi_{\text{py}}$ can be obtained from (43), and given by

$$\dot{\theta}_{\text{py}} = \frac{\cos\phi_{\text{rp}} \dot{\theta}_{\text{rp}} - \sin\phi_{\text{rp}} \sin\theta_{\text{rp}} \cos\theta_{\text{rp}} \dot{\phi}_{\text{rp}}}{1 - \sin\phi_{\text{rp}}^2 \sin\theta_{\text{rp}}^2} \quad \text{and} \quad \dot{\psi}_{\text{py}} = \frac{\cos\phi_{\text{rp}} \sin\theta_{\text{rp}} \dot{\phi}_{\text{rp}} + \sin\phi_{\text{rp}} \cos\theta_{\text{rp}} \dot{\theta}_{\text{rp}}}{\sqrt{1 - \sin\phi_{\text{rp}}^2 \sin\theta_{\text{rp}}^2}} \tag{44}$$

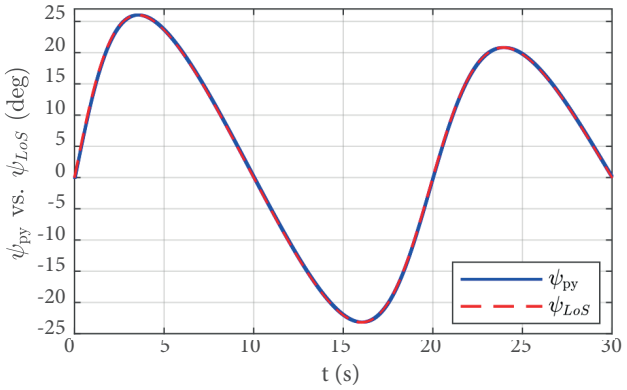
Finally, (18) can be computed to obtain the required LoS rates, $\omega_{y_{\text{py}}}$ and $\omega_{z_{\text{py}}}$.

## 6. Simulation results

**Scenario 1.** In order to validate the results obtained in this paper, a rather aggressive tracking scenario is constructed. The object being tracked is assumed to be a maneuvering aircraft traveling at a speed of about 250 m/s, initially about 1000 m away from the observer. Using the rate gyro outputs ($\omega_{x_{\text{rp}}}$ and $\omega_{y_{\text{rp}}}$), roll-pitch gimbal angles ($\phi_{\text{rp}}$ and $\theta_{\text{rp}}$) and platform angular rates $(\tilde{p}, \tilde{q}, \tilde{r})$, LoS angles and rates are constructed.




(a) Constructed and true LoS angles: $\theta_{py}$ and $\theta_{los}$




(b) Constructed and true LoS angles: $\psi_{py}$ and $\psi_{los}$

**Figure 5**. Constructed and true LoS angles.

Constructed LoS angles vs. the true LoS angles are plotted in Figures 5a and 5b. As described before, $\theta_{py}$ and $\psi_{py}$ are virtual gimbal angles. $\theta_{los}$ and $\psi_{los}$ are the true LoS angles, computed directly from the relative position of the target with respect to the platform on which gimbal is located.

Results obtained from the LoS construction method agree well with the true values, with some error, $\delta\theta$ and $\delta\psi$. This error is the tracking error shown in Figure 6. (Due to the scaling of the graph, error is not discernible in Figure 5.) These angles should be viewed as the distance of the target form the center of the detector, usually in terms of pixels. This is the primary output of a tracking algorithm.

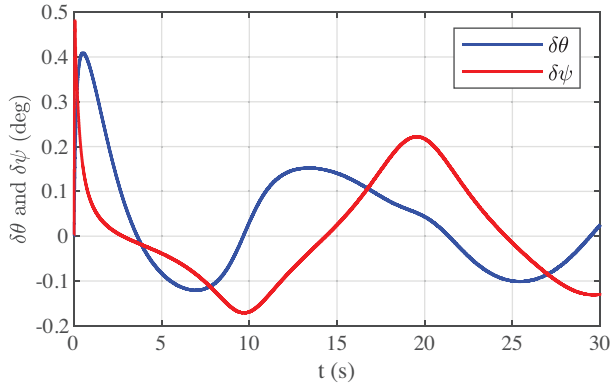
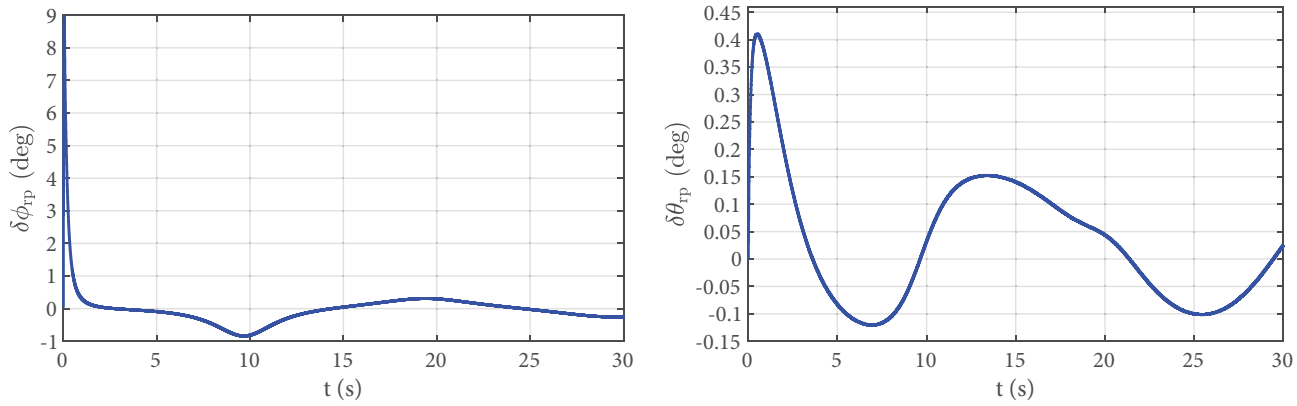



**Figure 6**. Virtual gimbal delta angles: $\delta\theta$ and $\delta\psi$.

However, since, $\delta\theta$ and $\delta\psi$ cannot be used directly to close the tracker control loop. They need to be converted into $\delta\phi_{rp}$ and $\delta\theta_{rp}$ angles and processed by the tracker controllers, $K_p^t$ and $K_r^t$. $\delta\phi_{rp}$ and $\delta\theta_{rp}$ angles that correspond to $\delta\theta$ and $\delta\psi$ angles are shown in Figure 7a and 7b, respectively.

A control design for the roll-pitch gimbal is designed to keep the target at the center of FoV. The outer (tracker) controllers ($K_p^t$ and $K_r^t$) are P type controllers and the tracker loops are designed so that the 3 dB closed loop bandwidth is about 5 Hz. (Although a PI type tracker controller would perform better, a P controller is selected so that the time responses that will be presented is more informative.)

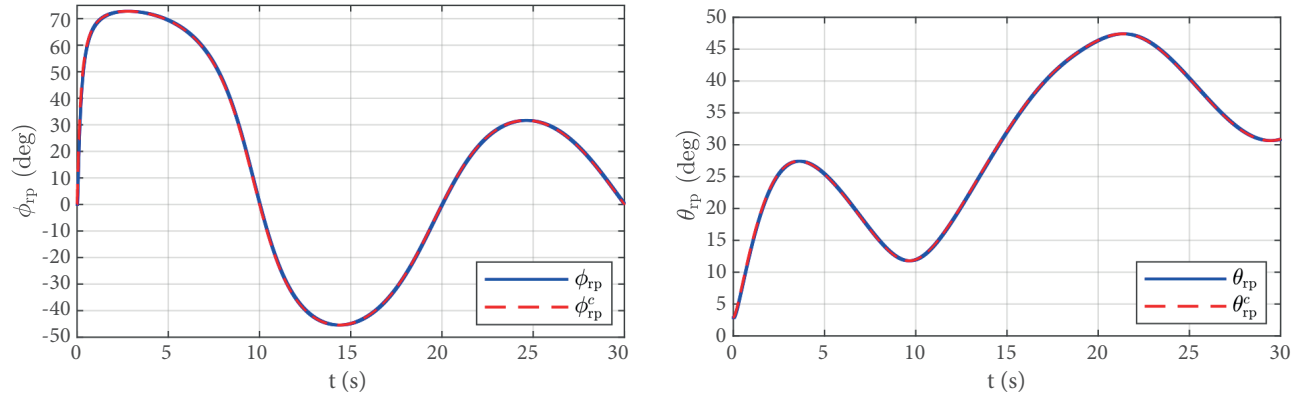
The outputs of the tracker controllers, $K_p^t$ and $K_r^t$, are commanded angular rates to the inner (rate)




(a) Gimbal delta angles: $\delta\varphi_{rp}$ (b) Gimbal delta angles: $\delta\theta_{rp}$

**Figure 7**. Roll-pitch gimbal angle errors.

controllers, $K_p$ and $K_r$. These controllers process the difference between the commanded rates and the measurements received from a two-axis gyroscope mounted on the inner (pitch) axis of the gimbal. The inner loop (rate) controllers ($K_p$ and $K_r$) are PI and the 3dB bandwidths of the inner loops are about 70 Hz. Commanded and actual roll and pitch gimbal angles are given in Figures 8a and 8b. Although not detectable due the scaling of the graphs in Figure 7 there is a discrepancy between the commanded and actual roll-pitch gimbal angles. This discrepancy is in fact presented in Figure 7.



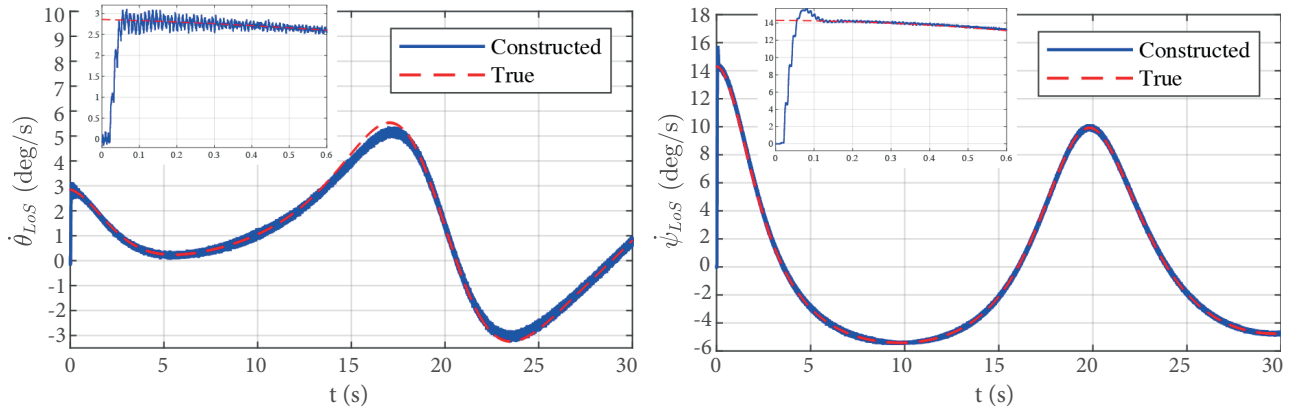

(a) Commanded and actual roll angle: $\varphi^c_{rp}$ and $\varphi_{rp}$ (b) Commanded and actual pitch angle: $\theta^c_{rp}$ and $\theta_{rp}$

**Figure 8**. Roll-pitch gimbal angles and commands.

Finally, constructed LoS rates and true LoS rates are presented in Figures 9a and 9b. Zoomed in figures are also provided to show the convergence in less than 200 ms. The true LoS rates, similar to the true LoS angle case, are computed directly from the relative position and velocity of the target with respect to the platform on which gimbal is located. Constructed rates are the main result of the algorithm described in this paper. Results show a good agreement between the constructed and the true values, with some error, especially in the roll axis of the gimbal.

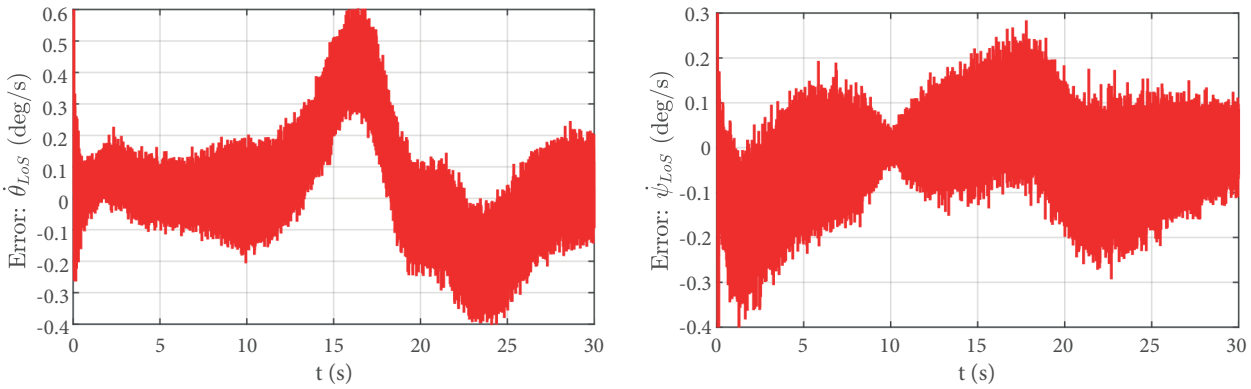
LoS rate estimation errors are presented in Figures 10a and 10b. As mentioned, the simulation includes a rate gyro model with dynamics and noise. Due to noise, the error signal is a fairly noisy signal and is of the

order magnitude of the noise. The simulation scenario considered employs a maneuvering fast target and the target is also close to the observer.




(a) Constructed and true LoS rates: $\dot{\theta}_{py}$ and $\dot{\theta}_{LoS}$ (b) Constructed and true LoS rates: $\dot{\psi}_{py}$ and $\dot{\psi}_{LoS}$

**Figure 9**. Constructed and true LoS rates.




(a) LoS rate estimationerror: $\dot{\theta}_{LoS}$ (b) LoS rate estimation error: $\dot{\psi}_{LoS}$

**Figure 10**. LoS rate estimation error.

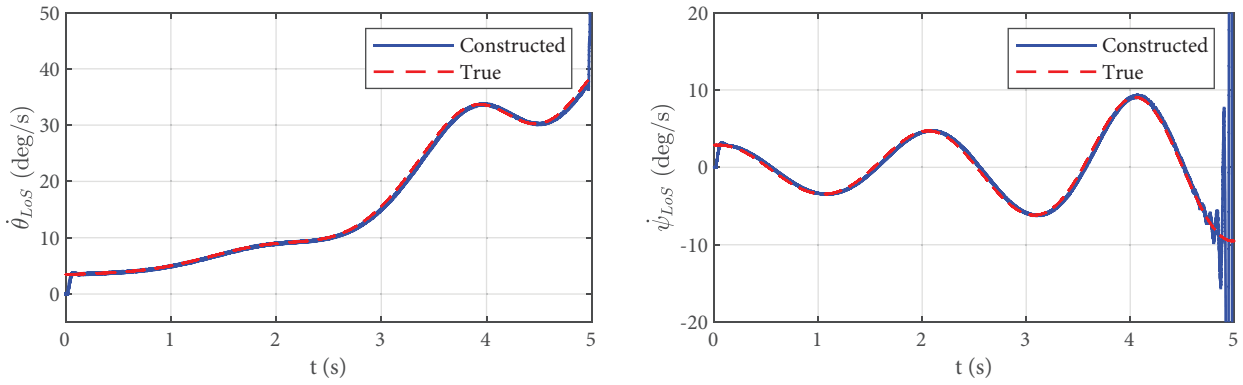
All of these factors result in high magnitude and rapidly changing LoS rates. Reducing the speed of the target and/or placing it further away from the sensor would result in smaller LoS rates, and the method described would display smaller estimation errors.

**Scenario 2.** Any target scenario where either the target is slower or further away or both is a less aggressive scenario than the one presented in the paper. The second target scenario is picked to demonstrate the limitations of the proposed method. The object being tracked is assumed to be a maneuvering aircraft traveling at a varying speed between about 160 and 250 m/s, initially about 1000 m away from the observer, and approaching the tracking system amid high slalom maneuvers. Using the rate gyro outputs ($\omega_{x_{\text{rp}}}$ and $\omega_{y_{\text{rp}}}$), roll-pitch gimbal angles ($\phi_{\text{rp}}$ and $\theta_{\text{rp}}$) and platform angular rates ($\tilde{p}, \tilde{q}, \tilde{r}$), LoS angles and rates are constructed.

Constructed LoS rates and true LoS rates are presented in Figures 11a and 11b. The true LoS rates are computed directly from the relative position and velocity of the target with respect to the platform on which

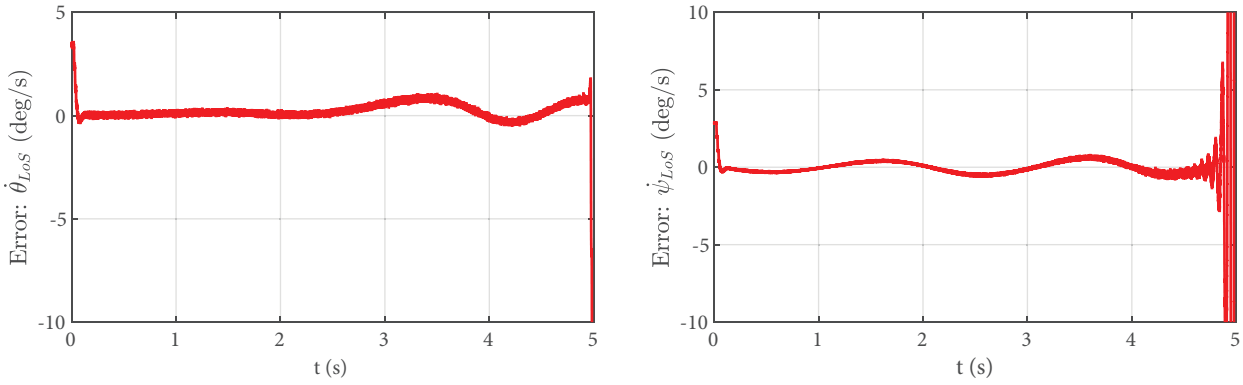
gimbal is located. Constructed rates are the main result of the algorithm described in this paper. Initially, results show a good agreement between the constructed and the true values. However, around $t = 5$ seconds the estimation quickly fails. This is due to the fact that target flies over the gimbal system, and falls outside of the gimbal angular range, namely the field of regard.

LoS rate estimation errors are presented in Figures 12a and 12b. As mentioned, the simulation includes a rate gyro model with dynamics and noise. As can be seen, initially, the estimation errors are fairly small. However, as the target nears the gimbal while maneuvering, the error becomes larger, and eventually the errors are not contained. The target is lost.




(a) Constructed and true LoS rates: $\dot{\theta}_{py}$ and $\dot{\theta}_{LoS}$ (b) Constructed and true LoS rates: $\dot{\psi}_{py}$ and $\dot{\psi}_{LoS}$

**Figure 11**. Constructed and true LoS rates.




(a) LoS rate estimationerror: $\dot{\theta}_{LoS}$ (b) LoS rate estimation error: $\dot{\psi}_{LoS}$

**Figure 12**. LoS rate estimation error.

As expected from any tracking system, the proposed method have limitations. Those limitations may be due insufficient bandwidth or narrow field of regard. However, there are environmental factors which effect any tracking systems performance. These include atmospheric effects, thermal effects, the Sun's position with respect to the tracker and the target, etc. These effects were not included in the simulations.

## 7. Summary and discussion

A kinematic tracker model to use on a roll-pitch gimbal and a LoS rate construction method for roll-pitch gimbals are described in this paper. A virtual pitch-yaw gimbal devised and the kinematic relationship between the two gimbal configurations are utilized to construct the LoS rate. The virtual gimbal purely exist in the algorithm and does not exist physically. An ideal (kinematic) tracker model was developed in order to supply the gimbal control loops with required error inputs. Next, given the platform body rate sensor measurements, actual roll-pitch gimbal angle measurements, and their computed rates are used to construct the LoS rate. The constructed LoS rates are visulaized for two different target scenarios: one a close range, high speed, maneuvering target; and another scenario where the target leaves the field of regard of the gimbal.

There are a number of issues that are left outside scope of this paper. An important one is the synchronicity of measurements. When the measured data does not belong to the same time instant, certain measures need to be taken. Another important topic is measurement errors, i.e., sensor errors and also errors that stem from the physical implementation of any algorithm such as the quantization error. In a complex system, many subsystems operate at different sampling rates. This multi-rate nature of the LoS rate construction method also needs to examined. Since it has been addresses by many researchers, the infamous zenith pass problem of roll-pitch gimbal configurations is left outside the scope. Future directions include addressing some of the issues above and also including a missile autopilot and guidance system to better assess the performance of the proposed method.

## References

[1] Ekstrand B. Tracking filters and models for seeker applications. IEEE Transactions on Aerospace and Electronic Systems 2001; 37 (3): 965-977. doi: 10.1109/7.953250

[2] Shneydor NA. Missile guidance and pursuit: kinematics, dynamics and control. Chichester, West Sussex, England: Horwood Publishing Limited, 1998.

[3] Siouris GM. Missile guidance and control systems. New York, NY, USA: Springer, 2003.

[4] Waldmann J. Line-of-sight rate estimation and linearizing control of an imaging seeker in a tactical missile guided by proportional navigation. IEEE Transactions on Control Systems Technology 2002; 10 (4): 556-567. doi: 10.1109/TCST.2002.1014675

[5] Ra WS, Whang IH, Ahn JY. Robust horizontal line-of-sight rate estimator for sea skimming anti-ship missile with two-axis gimballed seeker. IEE Proceedings Radar Sonar Navigation 2005; 152 (1): 9-15. doi: 10.1049/ip-rsn:20041177

[6] Hilkert JM, Miller R, Mooty G. Gimbal system configurations and line-of-sight control techniques for small UAV applications. In: Proceedings SPIE 8713 Airborne Intelligence, Surveillance, Reconnaissance Systems and Applications; Baltimore, MA, USA; 2013. pp. 1-15.

[7] Liu X, Mo B. Line-of-sight estimation for missile with roll-pitch seeker. In: Chinese Control Conference; Wuhan, China; 2018. pp. 4957-4961.

[8] Wang X, Mo B, Li X, Liu F. A line-of-sight rate estimation method for roll-pitch gimballed infrared seeker. Optik - International Journal for Light & Electron Optics 2019; 192: 162935. doi: 10.1016/j.ijleo.2019.162935

[9] Yang C, Zhang N, Jia H. LOS rate reconstruction and application of roll-pitch seeker. In: International Conference on Electrical and Control Engineering; Yichang, China; 2011. pp. 2307-2310 (in Chinese).

[10] Liu H, Zhu NC, Liu B, Jia HG. LOS stabilization and gyro configuration analysis for roll-pitch seeker. Applied Mechanics and Materials 2013; 397-400: 530-535. doi: 10.4028/www.scientific.net/AMM.397-400.530

[11] Liu X, Mo B, Liu F. Line-of-sight stabilization of roll-pitch seeker using differentiator-based disturbance compensation control. Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering 2020; 234 (7): 1326-1339. doi:10.1177/0954410020902665

[12] Gao Q, Sun Q, Qu F, Wang J, Han X et al. Line-of-sight rate modeling and error analysis of inertial stabilized platforms by coordinate transformation. Proceedings of the Institution of Mechanical Engineers, Part B: Engineering Manufacture 2019; 233 (5): 1317-1322. doi: https://doi.org/10.1177/0954405417716955

[13] Kim J, Hong J, Park K, Ryoo C. Line-of-sight rate analysis for a missile with two-axes gimbal seeker mounted on the nose cone platform. In: Conference of the Society of Instrument and Control Engineers of Japan; Kanazawa, Japan; 2017. pp. 458-463.

[14] Maley JM. Line of sight rate estimation for guided projectiles with strapdown seekers. In: AIAA Guidance, Navigation, and Control Conference; Kissimmee, Florida, USA; 2015. pp. 1-24.

[15] Yingchao X, Jun Z, Bin Z. Line-of-sight rate estimation based on attitude dynamics for strap-down seeker. In: Chinese Control And Decision Conference; Nanchang, China; 2019. pp. 2078-2083.

[16] Mi W, Shan J, Liu Y. Adaptive unscented kalman filter based line of sight rate for strapdown seeker. In: Chinese Automation Congress; Xi'an, China; 2018. pp. 886-891.

[17] Guo X, Lan J. Line-of-sight rate estimation for barrel-roll maneuvering target tracking. In: International Conference on Information Fusion; Rustenburg, South Africa; 2020. pp. 1-8.

[18] Titterton DH, Weston JL. Strapdown Inertial Navigation Technology. Reston, VA, USA: The Institution of Electrical Engineers, 2004.

[19] Ekstrand B. Equations of motion for a two-axes gimbal system. IEEE Transactions on Aerospace and Electronic Systems 2001; 37 (3): 1083-1091. doi: 10.1109/7.953259

[20] Hilkert JM. Inertially stabilized platform technology. IEEE Control Systems Magazine 2008; 28 (1): 26-46. doi: 10.1109/MCS.2007.910256

[21] Kürkçü B, Kasnakoğlu C, Efe MÖ. Disturbance/uncertainty estimator based robust control of nonminimum phase systems. IEEE/ASME Transactions on Mechatronics 2018; 23 (4): 1941-1951. doi: 10.1109/TMECH.2018.2835658

[22] Hurák Z, Řezáč M. Combined line-of-sight inertial stabilization and visual tracking: application to an airborne camera platform. In: Joint IEEE Conference Decision and Control and Chinese Control Conference; Shanghai, China; 2009. pp. 8458-8463.

[23] Masten MK. Inertially stabilized platforms for optical imaging systems. IEEE Control Systems Magazine 2008; 28 (1): 47-64. doi: 10.1109/MCS.2007.910201

[24] Zipfel PH. Modeling and Simulation of Aerospace Vehicle Dynamics. Reston, VA, USA: AIAA, 2007.