NASA/TM–20240012525

MSFC-TVC | TB-03: Derivation of Thrust Vector Control (TVC) Actuator-Force / Gimbal-Torque Transformation Matrix

Nathaniel A. Stepp
NASA Marshall Spaceflight Center, Huntsville, AL

December 2024

NASA STI Program Report Series

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA scientific and technical information (STI) program plays a key part in helping NASA maintain this important role.

The NASA STI program operates under the auspices of the Agency Chief Information Officer. It collects, organizes, provides for archiving, and disseminates NASA's STI. The NASA STI program provides access to the NTRS Registered and its public interface, the NASA Technical Reports Server, thus providing one of the largest collections of aeronautical and space science STI in the world. Results are published in both non-NASA channels and by NASA in the NASA STI Report Series, which includes the following report types:

- TECHNICAL PUBLICATION. Reports of completed research or a major significant phase of research that present the results of NASA Programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counterpart of peer-reviewed formal professional papers but has less stringent limitations on manuscript length and extent of graphic presentations.

- TECHNICAL MEMORANDUM. Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.

- CONTRACTOR REPORT. Scientific and technical findings by NASA-sponsored contractors and grantees.

- CONFERENCE PUBLICATION. Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or co-sponsored by NASA.

- SPECIAL PUBLICATION. Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.

- TECHNICAL TRANSLATION. English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services also include organizing and publishing research results, distributing specialized research announcements and feeds, providing information desk and personal search support, and enabling data exchange services.

For more information about the NASA STI program, see the following:

- Access the NASA STI program home page at http://www.sti.nasa.gov

NASA/TM–20240012525

MSFC-TVC | TB-03: Derivation of Thrust Vector Control (TVC) Actuator-Force / Gimbal-Torque Transformation Matrix

Nathaniel A. Stepp
NASA Marshall Spaceflight Center, Huntsville, AL

National Aeronautics and
Space Administration

Marshall Space Flight Center (MSFC)
Huntsville, AL

December 2024

Available from:

NASA STI Program / Mail Stop 050
NASA Langley Research Center
Hampton, VA 23681-2199

Thrust Vector Control Technical Bulletin (TB-03):

Derivation of Thrust Vector Control (TVC) Actuator-Force / Gimbal-Torque Transformation Matrix

Nathaniel A. Stepp

Technical Assistant / TVC Subject-Matter-Expert (SME)

NASA Marshall Space Flight Center
Thrust Vector Control System Integration and Components Branch (MSFC-ER63)

nathaniel.a.stepp@nasa.gov

Created On: 27 August 2020

Revised On: 10 December 2024 @ 13:58

1 Introduction

Thrust vector control (TVC) systems for rocket engine propulsion traditionally use a simple linear relationship to convert between actuator forces and torques about the engine gimbal's center-of-rotation (COR). As shown in Equation (1), the torque about the gimbal COR is proportional to the applied actuator force and the TVC moment arm (MA)—the perpendicular distance between the TVC actuator's line-of-action (LOA) and the engine gimbal's COR.

$$|\vec{\tau}_g| = |\vec{\mathbf{F}}_a|\,|\vec{\mathbf{L}}_{MA}| \quad \Longleftrightarrow \quad |\vec{\mathbf{F}}_a| = \frac{|\vec{\tau}_g|}{|\vec{\mathbf{L}}_{MA}|} \tag{1}$$

While this fundamental relationship remains valid and accurate in a two-dimensional (2D), one-degree-of-freedom (1-DOF) context—particularly in its non-linear formulation as described in the ER63 Technical Bulletin TB-02 (*Derivation of Thrust Vector Control (TVC) Engine-Gimbal / Actuator Moment-Arm Geometry*)—it becomes limited when extended to three-dimensional (3D), two-degree-of-freedom (2-DOF) analyses. In 3D, out-of-plane angular displacements can arise, causing the gimbaled engine plane-of-motion to become non-coplanar with the respective actuator plane-of-motion. Such misalignments occur due to the rod-end (RE) and/or tail-stock (TE) mounting geometry of individual TVC actuators.

These geometric complexities lead to inaccuracies in calculating engine torques and corresponding actuator forces if only the traditional TVC MA relationship is employed. For higher gimbal angular displacements (e.g., greater than 5 degrees), these inaccuracies become more pronounced, necessitating a more robust mathematical framework.

This document presents a comprehensive 3D (2-DOF) derivation of a transformation matrix that accurately converts between gimbaled engine torques and TVC actuator forces. By incorporating the necessary geometric and rotational considerations, this new approach corrects the limitations of the traditional TVC MA method. Subsequent sections compare the newly formulated approach to the traditional method, demonstrating its enhanced accuracy and reliability for a broad range of gimbaled engine conditions.

Acronym Definitions:

The following table lists acronyms used throughout this report, in alphabetical order.

COM	Center of Rotation
DCM	Direction Cosine Matrix
DOF	Degree of Freedom
LOA	Line of Action
MA	Moment Arm
TVC	Thrust Vector Control

Variable Definitions

The following table lists variables in their order of appearance, providing concise definitions and reference frames.

$\vec{\tau}_g$	Gimbaled engine torque vector about the gimbal COR, expressed in the vehicle-fixed coordinate frame.
$\vec{\mathbf{F}}_a$	TVC actuator force vector acting along the actuator line-of-action (LOA), referenced from the gimbal COR in the vehicle-fixed coordinate frame.
$\vec{\mathbf{L}}_{MA}$	TVC moment arm (MA) vector, defined as the perpendicular distance between the gimbal COR and the TVC actuator LOA, expressed in the vehicle-fixed coordinate frame.
$\mathbf{R_x}$	Roll rotation transformation matrix from the gimbaled engine coordinate frame to the vehicle-fixed coordinate frame.
ϕ_c	Gimbaled engine roll gimbal angle (roll constraint angle).
σ	Intermediate variable for calculating the roll gimbal angle (constraint).
γ	Intermediate variable for calculating the roll gimbal angle (constraint).
$\mathbf{R_y}$	Pitch rotation transformation matrix from the gimbaled engine coordinate frame to the vehicle-fixed coordinate frame.
θ	Gimbaled engine pitch gimbal angle.
$\mathbf{R_z}$	Yaw rotation transformation matrix from the gimbaled engine coordinate frame to the vehicle-fixed coordinate frame.
ψ	Gimbaled engine yaw gimbal angle.
$\mathbf{R}$	Complete rotation transformation matrix from the gimbaled engine coordinate frame to the vehicle-fixed coordinate frame (combining roll, pitch, and yaw).
$\vec{\mathbf{L}}_{LOA_p}$	Pitch TVC actuator LOA position vector, referenced from the gimbal COR in the vehicle-fixed coordinate frame.
$\vec{\mathbf{L}}_{TS_p}$	Pitch TVC actuator tail-stock position vector, referenced from the gimbal COR in the vehicle-fixed coordinate frame.
$\vec{\mathbf{L}}_{RE_p}$	Pitch TVC actuator rod-end position vector, originally defined in the gimbaled engine coordinate frame and transformed into the vehicle-fixed coordinate frame.
$\hat{\mathbf{L}}_{LOA_p}$	Pitch TVC actuator LOA unit position vector, referenced from the gimbal COR in the vehicle-fixed coordinate frame.

f_{a_p}	Pitch TVC actuator force magnitude.
$\vec{\mathbf{T}}_{(\tau\|f)_p}$	Torque/force transformation vector for the pitch TVC actuator, referenced from the gimbal COR in the vehicle-fixed coordinate frame.
$\vec{\tau}_{g_p}$	Torque vector generated by the pitch TVC actuator force about the gimbal COR, perpendicular to the pitch actuator plane-of-motion and expressed in the vehicle-fixed coordinate frame.
$\vec{\tau}_{g_y}$	Torque vector generated by the yaw TVC actuator force about the gimbal COR, perpendicular to the yaw actuator plane-of-motion and expressed in the vehicle-fixed coordinate frame.
$\vec{\mathbf{T}}_{(\tau\|f)_y}$	Torque/force transformation vector for the yaw TVC actuator, referenced from the gimbal COR in the vehicle-fixed coordinate frame.
f_r	Gimbaled engine roll null force.
f_{a_y}	Yaw TVC actuator force magnitude.
$\mathbf{T}$	Complete torque/actuator force transformation matrix, referenced from the gimbal COR in the vehicle-fixed coordinate frame.
$\vec{\mathbf{F}}$	Combined TVC actuators force vector, including the roll force component and individual pitch and yaw actuator forces.
$\vec{\tau}_{g_E}$	Gimbaled engine torque vector about the gimbal COR, defined in the gimbaled engine coordinate frame.

2 Relevant TVC System Geometry

A clear understanding of the geometric relationships between the gimbaled engine, its actuators, and the vehicle coordinate system is essential for accurate force-to-torque transformations. Figure (1) illustrates a typical gimbaled rocket engine propulsion system, showing both pitch and yaw TVC actuators (denoted by subscripts "p" and "y," respectively) in three-dimensional (3D) space. The coordinate frames and positional references introduced here provide the foundation for the transformation derivations presented in subsequent sections.

Key geometric elements include:

- **Gimbal Center-of-Rotation (COR):** The origin for both the gimbaled engine and the vehicle-fixed coordinate frames.
- **Gimbaled Engine Angular Displacements and Torque Vectors:** Defined about the gimbal COR in the vehicle-fixed coordinate system.
- **TVC Actuator Tail-Stock Position Vectors:** Referenced to the gimbal COR and expressed in the vehicle-fixed coordinate frame.
- **TVC Actuator Rod-End Position Vectors:** Referenced to the gimbal COR but defined in the gimbaled engine coordinate frame.
- **TVC Actuator Line-of-Action (LOA) Vectors:** Extending from each actuator's tail-stock to its rod-end attachment point. Unit LOA vectors indicate the direction along which actuator forces are applied.
- **TVC Actuator Force Vectors and Magnitudes:** Acting along the LOA vectors, linking actuator loads to resulting gimbal torques.
- **Transformation Vectors for Torque/Force Conversion:** Defined normal to the 3D plane formed by the COR and actuator attachment points, supporting accurate and consistent torque-to-force transformations.

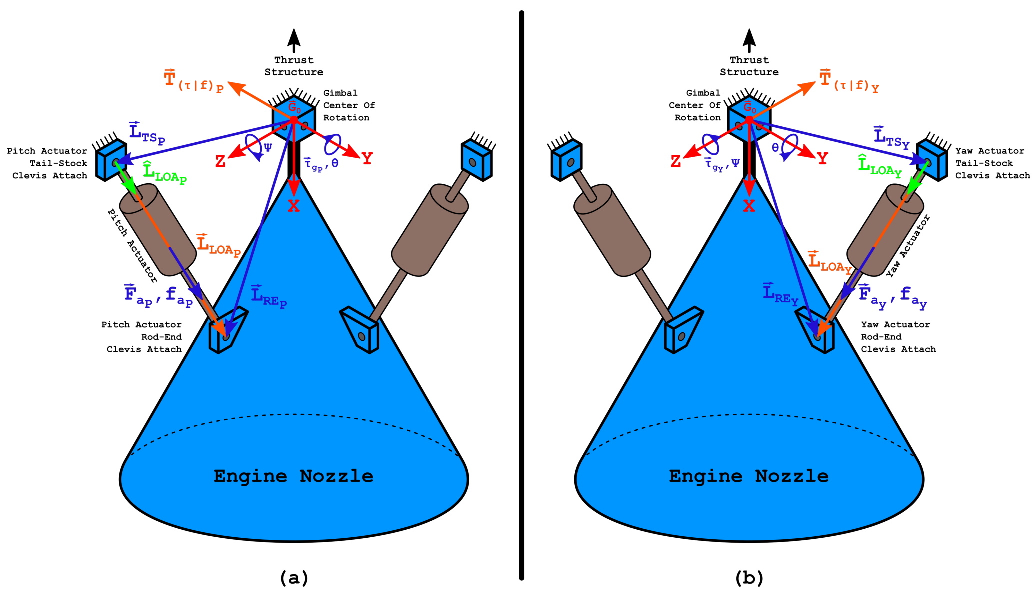

Figure 1: *Relevant TVC geometry for calculating gimbal torque/actuator force transformations. (a) Pitch actuator. (b) Yaw actuator.*

With these geometric definitions in place, we are equipped to handle the complexities inherent in 3D TVC analyses. The following sections will build upon this geometric framework, leading toward the derivation of a complete 3D transformation matrix for converting between gimbaled engine torques and actuator forces.

3 Limitations of the TVC MA for Gimbal Torque / Actuator Force Transformations

An earlier ER63 TVC Technical Bulletin (*Derivation of Thrust Vector Control (TVC) Moment Arm*) introduced a method for converting between gimbaled engine torques about the gimbal center-of-rotation (COR) and TVC actuator forces. This approach relies on the TVC moment arm (MA) concept and is accurate for two-dimensional (2D), one-degree-of-freedom (1-DOF) scenarios, provided that no significant angular displacement occurs out of the primary plane of motion.

However, in three-dimensional (3D), two-degree-of-freedom (2-DOF) analyses, the simplified TVC MA approach becomes limited. The following subsections detail why these inaccuracies arise and what their implications are for more complex gimbaled engine configurations.

3.1 Non-Coplanar Planes of Motion

The traditional TVC MA method assumes that the TVC actuator plane of motion and the gimbaled engine plane of motion remain co-planar. In 3D analyses, out-of-plane angular displacements commonly occur, causing these planes to become misaligned. When the gimbaled engine's rotation in one plane affects the angular orientation in another, the direct linear relationship between actuator forces and gimbal torques no longer holds. This misalignment introduces inaccuracies if only the original 2D MA method is employed.

3.2 Non-Planar Actuator Mounting Geometry

In addition to out-of-plane rotations, the physical placement of the TVC actuators' tail-stock and rod-end mountings can introduce further complexity. If these points do not lie in the same reference plane with respect to the gimbal COR, the actuator's effective plane of motion shifts relative to the engine's gimbaled plane of motion. Figure (2) illustrates an example where the actuator mounting geometry results in non-planar conditions, compounding the inaccuracies introduced by the basic MA approach.

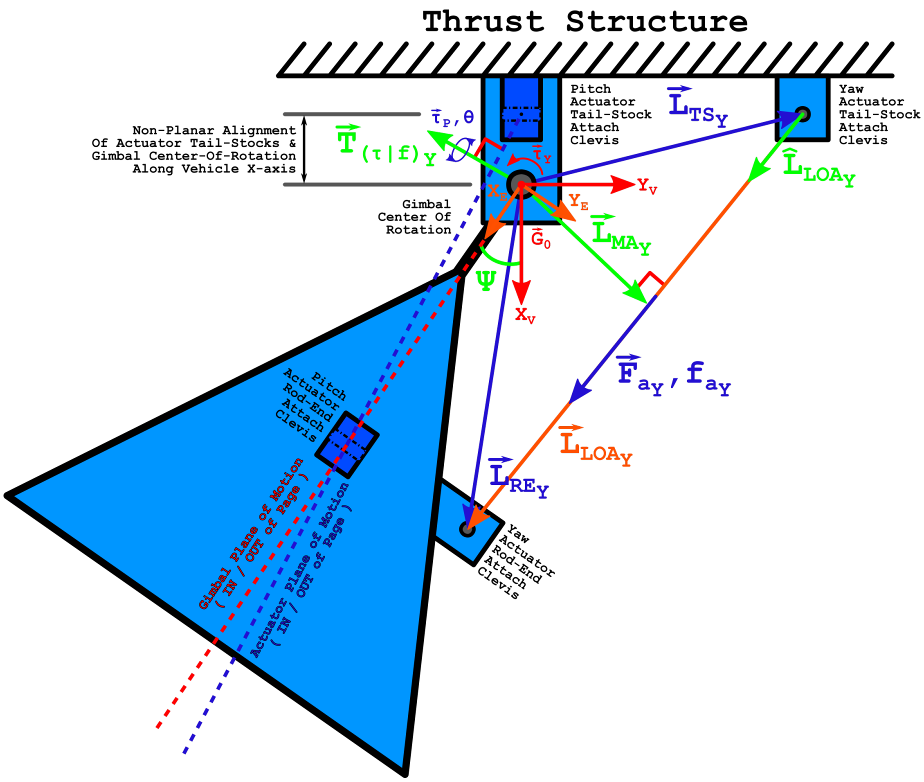

Figure 2: *A 2D view (XY-plane) of the gimbaled engine propulsion system, demonstrating how out-of-plane angular displacements and non-planar alignments between the TVC actuator and engine planes of motion degrade the accuracy of traditional MA-based predictions.*

3.3 Reaction Loads and High Angular Displacements

When the TVC actuator and engine planes of motion diverge, not all the actuator's force directly generates the desired gimbal torque. Instead, a portion of the force is reacted by the gimbal joint or even by adjacent actuators, introducing complex internal load paths. These reaction loads mean that the straightforward MA-derived relationships are no longer entirely accurate—particularly at higher gimbal angular displacements (e.g., beyond 2–3 degrees), where such discrepancies become more pronounced.

3.4 Need for a More Robust Approach

While the traditional TVC MA approach can still be applied to 3D (2-DOF) systems, it should only be used for small angular displacements if precise load predictions are required. As the gimbal angles increase, the limitations and inaccuracies grow, making it difficult to rely solely on MA-based methods.

The subsequent section presents a more comprehensive mathematical formulation that addresses these limitations. By incorporating a full 3D rotation transformation and accounting for the geometric complexities, this new approach enables accurate conversions between TVC actuator forces and gimbaled engine torques under a wide range of operating conditions.

4 Derivation of TVC Actuator Force / Gimbal Torque Transformation Matrix:

In this section, we derive a three-dimensional (3D) transformation matrix capable of accurately converting between TVC actuator forces and gimbaled engine torques about the engine's gimbal center-of-rotation, defined in the fixed vehicle coordinate system frame. This derivation starts with a rotation transformation matrix using a Direction Cosine Matrix (DCM) approach, and then incorporates a roll constraint relationship to maintain the mechanical constraints of the engine gimbal joint assembly. Finally, we define actuator line-of-action (LOA) vectors, actuator forces, and the resulting gimbal torques, culminating in a single transformation matrix.

4.1 Defining the 3D Rotation Transformation Matrix

First, a 3D rotation transformation matrix is defined. This matrix allows us to represent quantities, originally defined in the rotating gimbaled engine coordinate system frame, in the fixed vehicle coordinate system frame. The rotation changes as a function of gimbal angular displacement (pitch, yaw, and a derived roll constraint).

We use a DCM approach for the rotation matrix formulation. Below, Equations (2) through (5) define this transformation:

- Equation (2) represents the "roll" transformation, characterized by the rotation angle ϕ_c.
- Equation (3) represents the "pitch" transformation, characterized by the rotation angle θ.
- Equation (4) represents the "yaw" transformation, characterized by the rotation angle ψ.
- The product and subsequent transpose of Equations (2), (3), and (4), in an X-Y-Z rotation order, is shown in Equation (5). This represents the transformation from the gimbaled engine coordinate system frame to the fixed vehicle coordinate system frame.

$$\mathbf{R_x} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi_c & \sin\phi_c \\ 0 & -\sin\phi_c & \cos\phi_c \end{bmatrix} \tag{2}$$

$$\mathbf{R_y} = \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \tag{3}$$

$$\mathbf{R_z} = \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4}$$

$$\mathbf{R} = \Big(\mathbf{R_x}\,\mathbf{R_y}\,\mathbf{R_z}\Big)^T \tag{5}$$

4.2 Incorporating the Roll Constraint Relationship

While Equations (3) and (4) directly reference the TVC pitch and yaw rotation angles (θ and ψ, respectively), Equation (2) involves a ”roll” constraint rotation angle (ϕ_c). A naive assumption might be to set $\phi_c = 0$ due to the mechanical roll constraint of the gimbal joint. However, this would lead to an incorrect rotation transformation matrix, since the effective ”roll” orientation depends on the intermediate transformations.

For example, if $\theta = 90°$ and $\psi = 90°$, assuming $\phi_c = 0°$ would yield a 90° error in the resultant roll orientation. To correctly model the mechanical nature of the gimbal joint assembly, we must derive ϕ_c as a function of θ and ψ. This relationship is given in Equation (6), with intermediate variables defined in Equations (7) and (8).

$$\phi_c = \tan^{-1}\left(\frac{-\cos\theta\ \sin\psi\ \sin\theta\ (\sigma-1)}{\gamma}, \frac{\sigma - \Big(\cos^2\theta\ \sin^2\psi\ (\sigma-1)\Big)}{\gamma}\right) \tag{6}$$

$$\sigma = \Big(\cos\psi\ \cos\theta\Big)\sqrt{\sin^2\theta + \Big(\cos\psi\ \cos\theta\Big)^2 + \Big(\cos\theta\ \sin\psi\Big)^2} \tag{7}$$

$$\gamma = \sin^2\theta + \Big(\cos\theta\ \sin\psi\Big)^2 \tag{8}$$

4.3 Defining the Actuator LOA Position and Unit Vectors

With the rotation matrix now fully defined (Equation (5)) and the correct roll constraint incorporated, we can define the actuator line-of-action (LOA) position vector. The LOA vector points from the actuator tail-stock to the rod-end clevis attachment location, and must be expressed in the fixed vehicle coordinate system frame. This is achieved via simple vector addition as shown in Equation (9):

$$\vec{\mathbf{L}}_{LOA_p} = \vec{\mathbf{L}}_{TS_p} - \Big(\mathbf{R} * \vec{\mathbf{L}}_{RE_p}\Big) \tag{9}$$

To facilitate further calculations, we define the LOA unit vector by normalizing $\vec{\mathbf{L}}_{LOA_p}$:

$$\hat{\mathbf{L}}_{LOA_p} = \frac{\vec{\mathbf{L}}_{LOA_p}}{|\vec{\mathbf{L}}_{LOA_p}|} \tag{10}$$

4.4 Actuator Force Definition and Resulting Gimbal Torques

Next, using the LOA unit vector from Equation (10), we define the actuator force vector. As shown in Equation (11), this force is simply the actuator force magnitude multiplied by the LOA unit vector:

$$\vec{\mathbf{F}}_{a_p} = f_{a_p}\, \hat{\mathbf{L}}_{LOA_p} = f\, \frac{\vec{\mathbf{L}}_{LOA_p}}{|\vec{\mathbf{L}}_{LOA_p}|} \tag{11}$$

With the actuator force vector defined, we can determine the engine gimbal torque vector about the gimbal center-of-rotation in the vehicle frame. This torque results from the cross product of the transformed actuator rod-end position vector and the actuator force vector (Equation (12)):

$$\vec{\tau}_{g_p} = \left(\mathbf{R} * \vec{\mathbf{L}}_{RE_p}\right) \times \vec{\mathbf{F}}_{a_p} \tag{12}$$

Substituting the force vector expression from Equation (11) into Equation (12) gives Equation (13):

$$\vec{\tau}_{g_p} = \left(\mathbf{R} * \vec{\mathbf{L}}_{RE_p}\right) \times \left(f\, \frac{\vec{\mathbf{L}}_{LOA_p}}{|\vec{\mathbf{L}}_{LOA_p}|}\right) \tag{13}$$

We then factor out f_{a_p} and rearrange terms (Equation (14)):

$$\vec{\tau}_{g_p} = f_{a_p} \left[\left(\mathbf{R} * \vec{\mathbf{L}}_{RE_p}\right) \times \vec{\mathbf{L}}_{LOA_p}\right] \frac{1}{|\vec{\mathbf{L}}_{LOA_p}|} \tag{14}$$

Substituting the LOA vector definition from Equation (9) into Equation (14) yields Equation (15):

$$\vec{\tau}_{g_p} = f_{a_p} \left[\left(\mathbf{R} * \vec{\mathbf{L}}_{RE_p}\right) \times \left((\mathbf{R} * \vec{\mathbf{L}}_{RE_p}) - \vec{\mathbf{L}}_{TS_p}\right)\right] \frac{1}{|\ (\mathbf{R} * \vec{\mathbf{L}}_{RE_p}) - \vec{\mathbf{L}}_{TS_p}\ |} \tag{15}$$

After simplifying the cross product and using vector cancellation, we obtain Equation (16):

$$\vec{\tau}_{g_p} = f_{a_p} \left(\vec{\mathbf{L}}_{TS_p} \times (\mathbf{R} * \vec{\mathbf{L}}_{RE_p})\right) \frac{1}{|\ (\mathbf{R} * \vec{\mathbf{L}}_{RE_p}) - \vec{\mathbf{L}}_{TS_p}\ |} \tag{16}$$

Rearranging Equation (16) to explicitly show the torque-to-force ratio yields Equation (17):

$$\frac{\vec{\tau}_{g_p}}{f_{a_p}} = \frac{\vec{\mathbf{L}}_{TS_p} \times (\mathbf{R} * \vec{\mathbf{L}}_{RE_p})}{|\ (\mathbf{R} * \vec{\mathbf{L}}_{RE_p}) - \vec{\mathbf{L}}_{TS_p}\ |} = \vec{\mathbf{T}}_{(\tau|f)_p} \tag{17}$$

4.5 Forming the Complete 3D Transformation Matrix

By applying Equations (2) through (17) for each of the gimbaled engine's pitch and yaw TVC actuators, and concatenating those partial transformation vectors along with a $[1, 0, 0]^T$ vector (representing the engine gimbal joint roll matrix constraint), a complete 3D transformation matrix is defined. This matrix transforms actuator force magnitudes into gimbaled engine torques about the COR in the fixed vehicle coordinate system frame. Equation (18) shows this formulation:

$$\sum_{i=1}^{n} \vec{\tau}_g(i) = \underbrace{\left(\vec{\tau}_{g_p} + \vec{\tau}_{g_y}\right)}_{\vec{\tau}_g} = \underbrace{\begin{bmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} & \vec{\mathbf{T}}_{(\tau|f)_p}(3,1) & \vec{\mathbf{T}}_{(\tau|f)_y}(3,1) \end{bmatrix}}_{\mathbf{T}} \underbrace{\begin{bmatrix} f_r \\ f_{a_p} \\ f_{a_y} \end{bmatrix}}_{\vec{\mathbf{F}}} \tag{18}$$

Finally, symbolically generalizing Equation (18) results in Equation (19). Using a matrix inversion operation, one can convert between TVC actuator forces and gimbaled engine torques about the COR in the fixed vehicle coordinate system frame:

$$\vec{\tau}_g = \mathbf{T} * \vec{\mathbf{F}} \iff \vec{\mathbf{F}} = \mathbf{T}^{-1} * \vec{\tau}_g \tag{19}$$

5 Comparison Example of TVC MA & Actuator Force / Gimbal Torque Transformation Matrix:

Using the variables and parameters defined in Table 1, we can compare the predicted TVC pitch actuator force based on a given gimbaled engine torque vector about the gimbal COR (defined in the gimbaled engine coordinate frame). Specifically, we contrast the results obtained using the traditional TVC MA method (as described in the ER63 Technical Bulletin *Derivation of Thrust Vector Control (TVC) Moment Arm*) with those derived from the 3D conversion/transformation matrix approach presented in this technical bulletin.

Note: The vector values provided in Table 1 follow a standard $[X, Y, Z]$ coordinate convention, as depicted in Figure 1.

Table 1: Values defined for example calculations comparing linear vs. non-linear TVC MA approaches.

$\vec{\mathbf{G}}_0$	=	[0.00, 0.00, 0.00] inches
$\vec{\mathbf{L}}_{RE_p}$	=	[12.00, 0.00, 15.00] inches
$\vec{\mathbf{L}}_{TS_p}$	=	[−7.00, 0.00, 10.00] inches
$\vec{\mathbf{L}}_{RE_y}$	=	[12.00, 15.00, 0.00] inches
$\vec{\mathbf{L}}_{TS_y}$	=	[−7.00, 10.00, 0.00] inches
θ	=	0 degrees
ψ	=	−15 to +15 degrees (in 0.1 degree increments)
$\vec{\tau}_{g_E}$	=	[0.00, 10,000, 0.00] in*lbf

Applying the calculation methods described in the previous section of this document, the values from Table 1, and Equations 2 through 19, we obtain pitch actuator force magnitudes as a function of gimbaled engine yaw angular displacement. The results are computed using both the TVC MA analysis method and the transformation matrix approach derived herein. Additionally, the percent error between the two calculated force magnitudes is computed.

Figure 3 presents these co-plotted results, showing both the pitch actuator force predictions and the associated percent error versus yaw angular displacement.

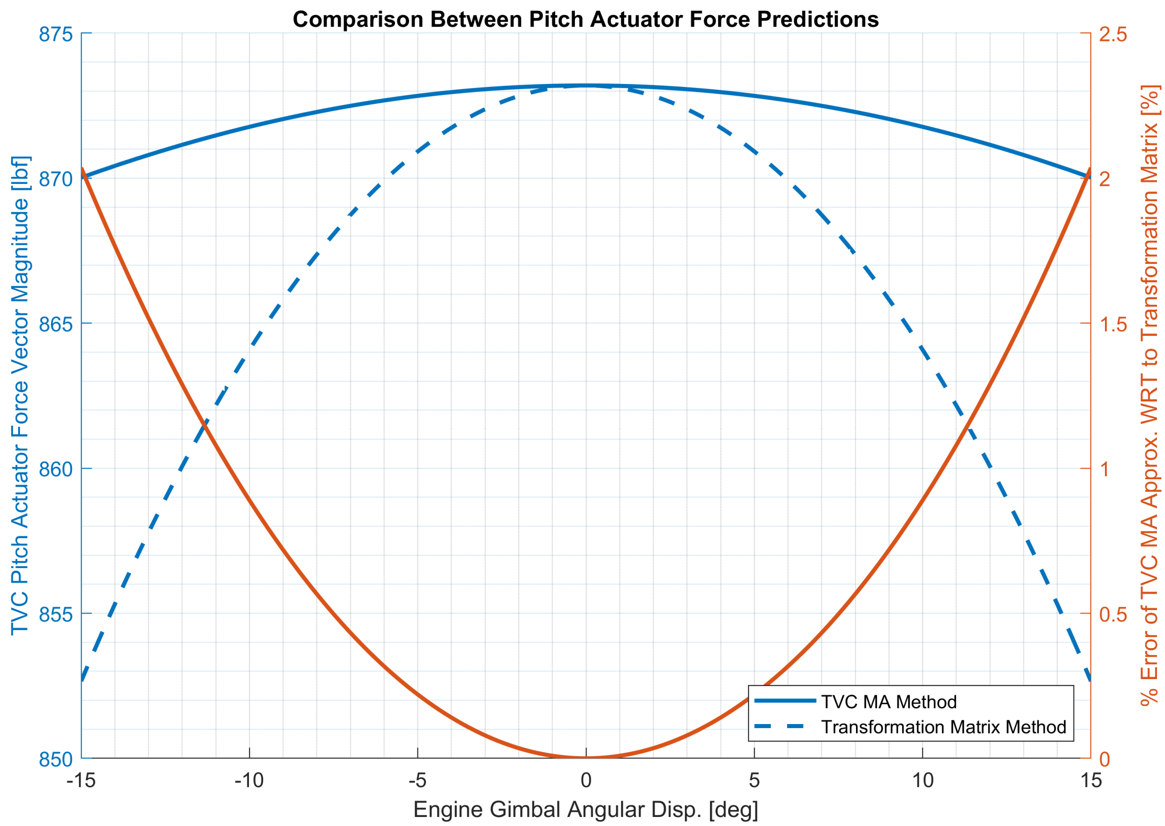

Figure 3: *Comparison of pitch actuator reaction forces due to an applied constant torque about the engine gimbal COR, calculated using the traditional TVC MA approach (Equation 1) and the transformation matrix approach (Equations 18 and 19).*

Examining Figure 3, the variability between the traditional TVC MA calculations and the fully accurate 3D transformation matrix results becomes apparent. For the specified scenario, the pitch actuator force predicted by the TVC MA method deviates from the transformation matrix prediction by approximately 2% at a 15-degree yaw displacement. This discrepancy underscores the importance of considering the geometric complexities discussed earlier. Loads analysts must evaluate whether this level of uncertainty is acceptable or if the more robust 3D transformation matrix approach is warranted for improved accuracy.

From a mathematical and analytical perspective, the transformation matrix approach, as captured by Equation 19, provides a concise and accurate formulation for performing 3D load analyses. Compared to extending the simpler TVC MA method into three dimensions, the transformation matrix approach ensures reliable load predictions, even at higher gimbal angular displacements.

Appendix A: MATLAB Script Used for Example

```matlab
%% FILE INFORMATION:

% FILENAME:     ER63_TVC_TB03_ForceTorqueTransform_FIG3.m
% AGENCY:       National Aeronautics & Space Administration (NASA)
% CENTER:       Marshall Spaceflight Center (MSFC)
% DEPARTMENT:   Propulsion Systems (ER)
% BRANCH:       TVC Systems Integration & Components Branch (ER63)
% PROGRAM:      General / Organization Academics
% PROJECT:      ER63 TVC Technical Bulletin Series
% COMPONENT:    TB-03 | TVC-Force/Torque-Transform | Figure-3
% CREATED BY:   Nathaniel A. Stepp
%               nathaniel.a.stepp@nasa.gov
% CREATED ON:   23 September 2020
% UPDATED ON:   10 December 2024

%% SCRIPT SETUP:

% Closing all open figure windows, clearing workspace variables,
% clearing text in command window, and formatting numerical command
% window output to "Long":
    close all; clear; clc; format longG; % Workspace Cleanup/Setup

% Setting script-global figure counter variable to "0":
    fig = 0; % Figure Number Counter

% Definition of plotting format / output parameters:
    figFileName = 'ER63-TVC-TB03_ForceTorqueTransform_FIG3';
    markerSpacePercent = 0.01;
    plottingSizeType = 'Inches';
    SIZE_X = 9;
    SIZE_Y = 7;
    padPercent = 0.05;

%% PARAMETER DEFINITIONS:

% Definition of gimbal center-of-rotation null position vector
% (coordinate system origin):
    G_0    = [  0.00;    0.00;    0.00 ]; % [in]

% Definition of TVC pitch actuator rod-end attach location position
% vector:
    L_RE_P = [ 12.00;    0.00;   15.00 ]; % [in]

% Definition of TVC pitch actuator tail-stock attach location position
% vector:
    L_TS_P = [ -7.00;    0.00;   10.00 ]; % [in]

% Definition of TVC yaw actuator rod-end attach location position
% vector:
    L_RE_Y = [ 12.00;   15.00;    0.00 ]; % [in]

% Definition of TVC yaw actuator tail-stock attach location position
% vector:
```

```
    L_TS_Y = [ -7.00;  10.00;   0.00 ]; % [in]

% Definition of pitch ("theta") range of gimbal angular motion:
    theta = 0; % [deg]

% Definition of yaw ("psi") range of gimbal angular motion::
    psi = -15:0.1:15; % [deg]

% Definition of externally applied torque-vector about engine gimbal
% center-of-rotation (COR) WRT engine gimbal frame:
    tau = [ 0; 10000; 0]; % [in*lbf]


%% CALCULATIONS:

% Definition of variable pre-allocations:
    F_a_P_1 = zeros(numel(theta), numel(psi));
    F_a_P_2 = zeros(numel(theta), numel(psi));

for i = 1:numel(theta)
for j = 1:numel(psi)

% Calculation of Direction-Cosine-Matrix (DCM) rotation matrix for given
% input "theta" and "psi" pitch/yaw gimbal angle vector elements:
    R_GIMB = calc_R_GIMB(theta(i), psi(j));

% Applying DCM rotation matrix to given input torque vector (WRT gimbaled
% engine frame) such to transform it to be represented WRT the
% vehicle-frame:
    tau_T = R_GIMB*tau;

% Pitch TVC Moment-Arm (MA) calculations:
    % Calculation of pitch actuator rod-end position vector WRT
    % vehicle-frame:
        L_RE_Y_T = R_GIMB*L_RE_Y; % [in]
    % Calculation of pitch actuator line-of-action unit vector (from
    % tail-stock to rod-end):
        L_ULOA_Y = (L_TS_Y - L_RE_Y_T)/norm(L_TS_Y - L_RE_Y_T); % [ND]
    % Calculation of pitch actuator TVC moment arm:
        L_MA_Y = L_TS_Y + dot((G_0 - L_TS_Y),L_ULOA_Y)*L_ULOA_Y; % [in]

% Yaw TVC Moment-Arm (MA) calculations:
    % Calculation of pitch actuator rod-end position vector WRT
    % vehicle-frame:
        L_RE_P_T = R_GIMB*L_RE_P; % [in]
    % Calculation of pitch actuator line-of-action unit vector (from
    % tail-stock to rod-end):
        L_ULOA_P = (L_TS_P - L_RE_P_T)/norm(L_TS_P - L_RE_P_T); % [ND]
    % Calculation of pitch actuator TVC moment arm:
        L_MA_P = L_TS_P + dot((G_0 - L_TS_P),L_ULOA_P)*L_ULOA_P; % [in]

% Calculation of engine torque to actuator force transformation matrix:
    % Definition of roll torque vector per unit force:
        TUA2F_X = [1;0;0];
    % Calculation of pitch torque vector per unit force:
```

```
        TAU2F_Y = cross(L_TS_P,L_RE_P_T)/norm(L_RE_P_T - L_TS_P);
    % Calculation of yaw torque vector per unit force:
        TAU2F_Z = cross(L_TS_Y,L_RE_Y_T)/norm(L_RE_Y_T - L_TS_Y);
    % Definition of transformation matrix, relating actuator forces (and
    % gimbal joint "roll" reaction forces) to engine gimbal torques
    % (specified in the vehicle-frame) about the gimbal center (T = A*F
    % -> F = A\T):
        forceTransform = [TUA2F_X, TAU2F_Y, TAU2F_Z];

% TBD:
    F_a_P_1(i,j) = norm(tau_T)/norm(L_MA_P);

% TBD:
    F_a_T = forceTransform\tau_T;

% TBD:
    F_a_P_2(i,j) = F_a_T(2);

end
end

% Calculation of percent-error between estimated pitch actuator forces via
% TVC MA vs. transformation matrix methods over entire input range of
% gimbal angular motion:
    percentError = (abs(F_a_P_1 - F_a_P_2)./F_a_P_2)*100;

%% FIGURES:

% Generating force-estimation / percent-error comparison plot:
        fig = fig + 1;
        figure(fig);
        set(gcf,'Color','White','Units',plottingSizeType,...
                'OuterPosition',[0, 0.04,SIZE_X,SIZE_Y]);
        hold on;
        yyaxis left;
        plot(psi, F_a_P_1, '-',...
                 psi, F_a_P_2, '--',...
                 'LineWidth', 2);
        ylabel('TVC Pitch Actuator Force Vector Magnitude [lbf]');
        yyaxis right;
        plot(psi, percentError, '-', 'LineWidth', 2);
        grid on;
        grid minor
        title('Comparison Between Pitch Actuator Force Predictions');
        ylabel('% Error of TVC MA Approx. WRT to Transformation Matrix [%]');
        xlabel('Engine Gimbal Angular Disp. [deg]');
        legend('TVC MA Method', 'Transformation Matrix Method',...
                   'Location', 'SE');

% Saving plotted figure to "*.png" image file:
        saveas(gcf,[figFileName '.fig'])
        exportgraphics( ...
                gcf, horzcat(figFileName, '.png'), ...
                'BackgroundColor' ,'w', ...
                'Resolution', 400 ...
```

```
        );

%% SUPPORT / UTILITY FUNCTION(S):

function R_GIMB = calc_R_GIMB (theta, psi)

% Calculation of the constrained gimbal joint implicit Euler roll
% angle as a function of the Euler pitch and yaw angles:
    if theta == 0 && psi == 0 % Null case
        phi_C = 0; % [rad]
    else % Non-null case
        sigma = (cosd(psi)*cosd(theta))/sqrt(sind(theta)^2 + ...
            (cosd(psi)*cosd(theta))^2 + (cosd(theta)*sind(psi))^2);
        gamma = sind(theta)^2 +(cosd(theta)*sind(psi))^2;
        phi_C = atan2d(-(cosd(theta)*sind(psi)*sind(theta)*...
            (sigma - 1))/(gamma),sigma - (cosd(theta)^2*...
            sind(psi)^2*(sigma - 1))/(gamma));
    end
% Pre-calculation of sines and cosines of gimbal roll, pitch, and
% yaw angles (specified in degrees):
    sx = sind(phi_C);
    sy = sind(theta);
    sz = sind(psi);
    cx = cosd(phi_C);
    cy = cosd(theta);
    cz = cosd(psi);
% Definition of "yaw" rotation matrix (about engine body-frame
% z-axis):
    R_GIMB_Z = [ cz,  sz,   0;...
                -sz,  cz,   0;...
                  0,   0,   1 ];
% Definition of "pitch" rotation matrix (about first
% intermediate-frame y-axis):
    R_GIMB_Y = [ cy,   0, -sy;
                  0,   1,   0;
                 sy,   0,  cy ];
% Definition of "roll" rotation matrix (about second
% intermediate-frame x-axis):
    R_GIMB_X = [  1,   0,   0;...
                  0,  cx,  sx;...
                  0, -sx,  cx ];
% Calculation of rotation matrix from engine body-frame to
% inertial frame (Direction Cosine Matrix method):
    R_GIMB = transpose( R_GIMB_X * R_GIMB_Y * R_GIMB_Z );

end
```